An informational approach to quantizing the effectiveness of test data arrays for static memory devices

Ju S Akinina¹, V E Bolnokin², S V Tyurin¹, A A Akinin¹, and S A Popov¹

¹ Voronezh State Technical University, 14 Moscow av., Voronezh, 394000, Russian Federation
² Mechanical Engineering Research Institute of the Russian Academy of Sciences, 4, Small Kharitonyevsky lane, Moscow, 101000, Russian Federation

E-mail: julakinn@mail.ru

Abstract. While testing of the Static Random Access Memory (SRAM) the effective detection of potential faults is mostly determined by the structure and the algorithm of test arrays of data generation. This article proposes a multiplicative indicator \(W\), which allows us to quantify the quality of test arrays of data. It is necessary to represent test arrays of data as a binary matrix, where each column corresponds to a certain bit of SRAM, and the rows correspond to test patterns. In this case, the process of sequential formation of test patterns is identified with the percolation process on a rectangular grid. It is assumed that percolation occurs at the moment when the maximization of the binary antagonisms both on the rows and on the columns of the binary matrix is provided. In this case, the binary sequences in all columns of the matrix must be different. Preliminary experimental data allows to suggest that the evaluation of the quality of test arrays of data based on the proposed indicator \(W\) can simultaneously be considered as the maximum efficiency of detecting a certain class of potential SRAM faults in a certain order of reading test data.

1. Introduction
The reliability of the functioning of various automated systems is mostly determined by the reliability of the used memory devices. A significant proportion of storage devices is static memory (SRAM), which has the highest performance with significant information volumes. These features of SRAM are caused to use in them the MBIST (Memory Built-In Self Test) technology.

Nowadays many variants of the practical implementation of the MBIST technology are known, for example, from [1, 2, 3, 4, 5]. In turn, MBIST tools implement algorithms of March tests, the theoretical foundations of the construction are presented in [6]. In [6] it is showed how the effectiveness of March tests depends on the bit structure of test arrays of data (test patterns), the order of writing test data, and reading the content of bits or words of the tested memory. In general, the effectiveness of tests is determined on the base of SRAM simulation with embedded faults as the ratio of the number of detected faults to the total number of embedded faults [6, 7]. This makes it much more difficult to independently search for high-quality test arrays not only for March tests, but also for managed random tests [5].
This article offers an information approach based on the provisions of percolation theory to an Autonomous quantitative assessment of the effectiveness of test arrays of data, which does not depend on the method of their formation – deterministic, algorithmic or random.

2. The formation of test arrays of data as a percolation process

From a mathematical point of view, percolation theory is a branch of probability theory that studies the properties of connected components of random graphs [8]. From the physical point of view, percolation processes include such processes that are called critical [9]. They are characterized by a critical point where certain properties change sharp. The physics of each critical phenomenon is unique, but they also have common features, the most important of which is that near the critical point, the physical system seems to break up into blocks with different properties, and the size of individual blocks grows indefinitely when approaching the critical point [9]. The outlines of the blocks are random. The blocks are arranged randomly, so looking at their instant images; it is difficult to distinguish any patterns. However, "on average "this geometry of images, which is called the" geometry of disorder", has quite certain properties.

It is convenient to represent test arrays of data in the form of a test matrix (T-matrix) (1):

\[
T = \begin{pmatrix}
a_{11} & a_{12} & \cdots & a_{1N} \\
a_{21} & a_{22} & \cdots & a_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
a_{L1} & a_{L2} & \cdots & a_{LN}
\end{pmatrix},
\]

where \( a_{ij} = \{0,1\} \); \( N \) – the number of controlled points (memory elements); \( L \) – the number of test sets; \( i \) – the current number of the test set (row); \( j \) - the current number of the controlled point (column). If we identify \( a_{ij} = 0 \) with an unpainted cell, and \( a_{ij} = 1 \) with a colored cell, then the structure of various functional tests can be geometrically represented as a rectangular grid, which is typical for the cell-based percolation [10].

The numerical characteristic of such grids in percolation theory is the fraction (\( p \)) of the colored squares. With such model representation of the \( T \)– matrix, the purpose of its study is to find a bit structure for which

\[
p \approx p_c,
\]

where \( p_c \) is the critical concentration, called the percolation threshold.

Grid models have the feature that a set of strict statements and numerical relations which have a universal character for percolation processes is proved for them [8, 9, 10].

The probability of joining clusters occurrence depends on the size of the grid. It is theoretically proved [8, 9, 10] that for a grid of finite size, the connecting cluster occurs at different concentrations, but if the size of the grid is directed to infinity, then the critical concentration becomes quite definite.

Figure 1 shows the approximate dependence of the probability \( P \) of percolation on the percentage of filled cells \( p \) [11]. The smooth curve in the figure 1 corresponds to a grid of finite size, and the stepped curve corresponds to an infinitely large grid.

![Figure 1. The probability P of percolation depending on the percentage p of filled cells.](image-url)
Let's go back to (1) and imagine that this is a graphical representation of a single-bit SRAM containing $N = 8$ memory elements, when forming $L = 8$ binary test vectors. Physically, binary 1 and 0 correspond the certain values of electric potentials $\phi_1$ and $\phi_0$. It is natural to assume that the interaction of these potentials in various combinations provoke the manifestation of the potential SRAM faults. In this case, an element or group of memory elements can act as an "aggressor", changing the contents of another element or group of elements that are "victim". Then it can be said that in order to effectively detect faults of the mutual influence of memory cells (elements), it is necessary to maximize the amount of antagonisms, both between all possible pairs of cells (elements) and between their various cluster formations, using a series-parallel change of test templates. In the future, we will distinguish antagonisms between pairs of memory elements (binary antagonisms) and all other antagonisms (cluster antagonisms).

3. The method of the quantization of the effectiveness of test arrays of data

The amount of binary antagonisms in a $k$-bit binary code can be determined from the following ratio (3):

$$S^{k,r} = C^2_k - (C^2_r + C^2_{k-r}) = r(k - r),$$

where $C^2_k$ – the number of pairs without repetitions between all bits of $k$-bit code;

$C^2_r$ – the number of pairs without repetitions between $r$-bits of the code that have the value of logical 1;

$C^2_{k-r}$ – the number of pairs without repetitions between $(k-r)$ - bits of the code that have the value of logical 0.

The maximization of the ratio (3) is achieved with the condition $r = k/2$, i.e. (4):

$$S_{\text{max}}^{k,r} = k^2 / 4$$

Taking into account (3) and (4), it is advisable to go to the relative indicator of the amount of binary antagonisms in the $k$-bit binary code (5):

$$P_{k,r} = r(k-r)/k_2^2 = 4 r(k-r)/k^2 = 4 p^1 p^0,$$

where $p^1$ is the fraction of single values in the binary code; $p^0$ is the fraction of zero values, and $1 \geq P_{k,r} \geq 0$.

Applying the ratio (5) to each row and to each column of the $T$-matrix, we can find private indicators of quality $W_1$ and $W_2$ that characterize the average values of the relative amount of binary antagonisms for rows and columns, respectively. The $W_2$ indicator will indicate the average frequency of changing values in the test templates. However, another private quality indicator $W_3$ is required, the numerical value of which would allow for individual distinctiveness of each individual SRAM element. To do this, it is enough to have the unique binary sequences in the columns of the $T$-matrix.

Then, the quantitative evaluation of the effectiveness of test arrays of data can be characterized by the following generalized indicator (6):

$$W = W_1 + W_2 + W_3,$$

where

$$W_1 = \frac{4 \sum_{i=1}^{L} p^1_i p^0_j}{L}, \quad W_2 = \frac{4 \sum_{j=1}^{N} p^1_j p^0_j}{N}, \quad W_3 = \frac{L \times n}{L \times N},$$
where $l$ – is the number of distinguishable rows (a consecutive group of identical rows is considered a single row);

$n$ – the number of distinguishable columns;

$L$ – the total number of rows;

$N$ – the total number of columns.

It should be noted that the ratio (6) is only valid if the SRAM with any organization will be pre-represented as one-bit.

Let’s take a look at a graphical representation of the $W$ (6) dependency obtained using special software that allows to consistently generate a given number of ($L$) random test patterns and change the number of ($N$) bits in the test patterns. In each test template, the generated single and zero values have an equally probable distribution. In the course of such statistical experiments, it was found that the type of functional dependence (6) and General regularities are preserved when the bit volumes of SRAM change. Figure 2 shows a typical type of graphical dependency (6) for the SRAM with a total volume of 128 bits (i.e. $N=128$).

![Figure 2. The graphical dependence of $W$ for RAM with a total volume of 128 bits.](image)

The analysis of the graphical dependence shows:

- graphical representation of the dependence $W$ is very close to the dependence of the probability $P$ of percolation on finite grids (see figure 1);
- there is an inflection point ($L \approx 7$) corresponding to some phase transition and equal $L_{\text{min}} = \text{int}(\log_2 N + 0.5)$ for any $N$;
- after the point $L \approx 14$ (i.e. $L = 2L_{\text{min}}$), when the test matrix approaches the square grid, the generalized efficiency index tends to 1, as with any percolation process on a bounded square grid;
- if the test matrix reaches the dimension of the square grid, the value $W$ reaches values greater than 0.97.

The value $L_{\text{min}}$ indicates that with such amount of binary bits, individual distinctness of each SRAM element can be provided in principle. However, only when $L = 2L_{\text{min}}$ individual distinctiveness can be provided by binary sequences in the columns of the $T$-matrix, each of which contains an equal number of ones and zeros. For this purpose, it is enough, for example, to use $L_{\text{min}}$ first test templates and then their inverse copies. Thus, the point $L_{\text{min}}$ is represented as a critical point, after which the entire SRAM seems to break up into individual memory elements. This feature allows us to consider the process of forming effective test arrays of data as a percolation process.
4. Preliminary results of experimental studies

Using the proposed generalized indicator \( W(6) \), we compare the quality of test arrays of data that are formed by the following March tests (table 1).

**Table 1. Comparison the quality of test arrays.**

| Test array | Test source | Data |
|------------|-------------|------|
| MATS+      | [12]        | \{↑(w0); ↑(r0, w1); ↓(r1,w0)\} |
| March C-   | [13]        | \{↑(w0); ↑(r0, w1); ↑(r1, w0); ↓(r0, w1); ↓(r1, w0); ↓(r0)\} |
| March RAW  | [14]        | \{↑(w0); ↑(r0, w0,r0,w1,r1); ↑(r1,w1,r1,w0,r0); ↓(r0,w0,r0,w1,r1); ↓(r1,w1,r1,w0,r0); ↓(r0)\} |

For a formal description of the March test the notation proposed in [15, 6] is used: \( \uparrow(w0) \) – writing 0 in all memory cells in an arbitrary order of addresses; \( \uparrow(r0, w1) \) – reading 0 and writing 1 in each memory cell in ascending order of addresses; \( \uparrow(r1, w0) \) – reading 1 and writing 0 to each memory cell in ascending order of addresses; \( \downarrow(r0, w1) \); \( \downarrow(r1, w0) \) is similar to the actions in descending order of addresses; \( \downarrow(r0) \) – reading from all the memory cells in a random order.

When forming the test arrays of data, all reading operations were ignored. The March element \( \uparrow(w0) \) was represented by a single null pattern. Each bit change in the memory content is treated as the next pattern. For example, for the MATS+ test, the sequence of test patterns for 4-bit memory looks like this: 0000; 1000; 1100; 1110; 1111; 1110; 1100; 1000. As a result of the conducted experiments, the following quantitative estimates of test data sets were obtained:

\[
W(\text{MATS+}) = 0.44; \quad W(\text{March C-}) = 0.65; \quad W(\text{March RAW}) = 0.65.
\]

In [16], the following experimental values of the covering capacity of the same tests are given:

\[
P(\text{MATS+}) = 43\%; \quad P(\text{March C-}) = 47\%; \quad P(\text{March RAW}) = 67\%.
\]

It should be noted that the test arrays of data are similar for the March C- and March RAW tests. The difference is in the March RAW test there are additional operations w0 and w1 in memory cells that have the same values, as well as additional operations r0 and r1.

Nowadays it is not possible to evaluate the effectiveness of March tests using the \( W(6) \) indicator. However, based on the preliminary results, we can compare the quality of test arrays of data using the \( W \) indicator and choose more efficient ones due to this indicator.

5. Conclusion

The proposed method of autonomous (i.e. preliminary) evaluation of the quality of test arrays of data for the SRAM provides the possibility of their quantitative evaluation of efficiency regardless of the method of data generation-deterministic, algorithmic or random. The practical use of this method expands the possibilities of searching for high-quality test arrays, both for March tests and for managed random and pseudo-random tests.

Preliminary experimental data allows to suggest that the evaluation of the quality of test arrays of data based on the proposed method can simultaneously be considered as the maximum efficiency of detecting a certain class of potential SRAM faults with a certain order of reading the test data.

References

[1] Kumar V S and Manimegalai R 2015 Efficient Memory Built in Self Test Address Generator Implementation. *International Journal of Applied Engineering Research* 10(7) 16797-813

[2] Van de Goor A J, Kukner H and Hamdioui S 2011 Optimizing Memory BIST Address Generator Implementations *6th International Conference on Design & Technology of Integrated Systems in Nanoscale Era* pp 1-6
[3] Varghese L and Suranya G 2014 Test Pattern Generation Using LFSR with Reseeding Scheme for BIST Designs International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering 3(5) 452-58

[4] Koteswaramma D, MuraliKrishna K, Sailaja M and Yedukondalu U 2014 Memory Testing and Repairing using MBIST with complete Programmability IOSR Journal of Electronics and Communication Engineering 9(7) 80-3

[5] Mrozek I and Yarmolik V 2016 Multiple Controlled Random Testing Fundamenta Informaticae 144 23-43

[6] Van de Goor A J 1991 Testing Semiconductor Memories: Theory and Practicechester (Wiley) 512

[7] Benso A, Di Carlo S, Di Natale G and Prinetto P 2002 Specification and design of a new memory fault simulator IEEE 11th AsianTest Symposium (ATS) 92-7

[8] Kesten H 1982 Percolation Theory for Mathematics (Springer) 432

[9] Efros A L 1986 Physics and Geometry of Disorder: Percolation Theory (Mir, Moscow) 259

[10] Gould H, Tobochnik J and Christian W 1988 An Introduction to Computer Simulation Methods: Application to Physical Systems (MA: Addison-Wesley)

[11] Saberi A A 2015 Recent advances in percolation theory and its applications Physics Reports 578 1-32

[12] Nair R 1979 Comments on "An Optimal Algorithm for Testing Stuck-at Faults in Random Access Memories" IEEE Transactions on Computers 28(3) 258-61

[13] Marinescu M 1982 Simple and Efficient Algorithms for Functional RAM Testing IEEE International Test Conference 236-9

[14] Hamdioui S, Al-ars Z and Goor A J Van de 2002 Testing static and dynamic faults in random access memories IEEE VLSI Test Symposium 395-400

[15] Suk D S and Reddy S M 1981 A march test for functional faults in semiconductors random-access memories IEEE Transactions on Computers 30(12) 982-5

[16] Linder M, Eder A, Oberländer K and Huch M 2011 Effectiveness of memory test algorithms and analysis of fault distribution in SRAMs IEEE Proceedings on ETS’11 1-6