IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 1, 2021, accepted August 27, 2021, date of publication September 6, 2021, date of current version September 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3110594

Fault Coverage Re-Evaluation of Memory Test Algorithms With Physical Memory Characteristics

KISEOK LEE, JEONGHWAN KIM, AND SANGHYEON BAEG, (Member, IEEE)
Department of Electronics Engineering, Hanyang University, Ansan 15588, Republic of Korea

Corresponding author: Sanghyeon Baeg (bau@hanyang.ac.kr)

This work was supported in part by Korea Institute for Advancement of Technology (KIAT) through the Korean Government [Ministry of Trade, Industry and Energy (MOTIE)] (The competency development program for industry specialist) under Grant P0012451, and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning under Grant NRF-2020R1H1A2103043.

ABSTRACT A memory fault model (FM) is an abstraction of the physical mechanism of memory failure. When the physical failure mechanisms are not fully represented in FMs, the coverage of the FMs can be different from that of the failure mechanisms. However, it is impractical (or impossible) to model every electrical aspect of the failure mechanisms with one or more FMs. This problem has become even worse with emerging technologies. Thus, in this study, the fault coverage (FC) consequences are investigated when the physical memory characteristics are not properly linked to the FMs or even test algorithms. Three physical characteristics were considered for this exploration: electrical masking, address scrambling, and electrical neighborhoods. To this end, memory fault simulations were performed, and the test algorithms were re-evaluated in terms of FC. Simulations were performed on the 1 kB area of the example SRAM model; three classes of FMs (56 static faults (SFs), 126 dynamic faults (DFs), and 192 neighborhood pattern-sensitive faults (NPSFs)) were simulated for FC evaluation; and March MSS, March MD2, and March 12N were used to re-evaluate the FCs of SFs, DFs, and NPSFs, respectively. From the simulation results, we observed the negative impact of physical characteristics on FC. When masking was considered, FC reductions of 10.72% SFs and 9.52% DFs were observed; when address scrambling was not available, an FC reduction of 80.21% NPSFs was observed. Finally, considering electrical neighborhood changes depending on the physical memory structure, an FC reduction of 41.67% NPSFs was observed.

INDEX TERMS Fault coverage (FC) re-evaluation, electrical masking, address scrambling, electrical neighborhood, memory fault model (FM).

I. INTRODUCTION

Since the invention of the first commercialized random-access memory (RAM) [1] in the early 1970s, the demand for memory has grown extraordinarily along with the evolving needs of the industry. In particular, the demand for high-speed, high-integration, and low-power memories is increasing at an unprecedented rate as cloud computing, artificial intelligence (AI), and fifth generation (5G) communication, among others, are positioned as the major contributors in the fourth industrial revolution. To provide memories that meet these requirements, the importance of testing memories inevitably increases daily.

A memory test algorithm is used to perform memory tests. The primary goal of the test algorithm is to detect faults associated with target memory fault models (FMs). Memory FMs are a means of explaining the process of faults occurring in memories; this process is described as the dynamics of interactions within (among) the memory cell(s). Inductive fault analysis (IFA) [2] is widely used as a fault modeling method that directly inserts resistive defects into simulated circuit models. Fault modeling is achieved by performing simulations for defective circuit models, then describing the difference between the predicted behavior and the actual behavior of the circuit, which directly depends on the sizes and locations of the defects inserted into the memory cells or peripherals [3]–[5]. A number of existing memory FMs apply this method.

However, as semiconductor technology gradually advances and new memory structures are continuously developed, it is unrealistic to model every aspect of memory failure mechanism. For example, it is infeasible to describe all aspects

The associate editor coordinating the review of this manuscript and approving it for publication was Cristian Zambelli.

of the radiation-induced or rowhammer failure mechanisms because the sources of mechanisms are diverse and interrelated, and the effects of mechanisms are expressed as a probability density according to the type/strength/frequency of the sources as well as memory constraints [6]–[8]. An FM is an abstraction of a failure mechanism [9], [10]; when the failure mechanisms are not properly blended into the FMs, test algorithms generated based on these FMs cannot properly test the memory failure mechanisms. This results in reduced fault coverages (FCs) of the test algorithms.

To bridge the gap between the FMs and failure mechanisms, in this study, three physical memory characteristics (electrical masking, address scrambling, and electrical neighborhoods) were considered; the FCs of existing memory test algorithms were re-evaluated considering these characteristics to examine the impact of memory characteristics on FC. When the test algorithms were evaluated based on one of the physical characteristics, their FCs were clearly reduced. The resulting reductions in FC provided in this work show that it is very difficult to accurately understand and model the failure mechanisms occurring in memories. At the same time, this study also demonstrates that the test algorithms developed based on such imprecise understanding of the mechanisms are no longer complete.

The remainder of this paper is organized as follows. Section II introduces the memory model, FMs, and test algorithms used in this work. Section III presents the FCs of the test algorithms introduced in Section II. In Section IV, the FCs of the test algorithms in Section III are re-evaluated considering the physical memory characteristics. Finally, Section V concludes the paper.

II. MEMORY MODEL, FAULT MODELS, TEST ALGORITHMS

This section explains the memory model, FMs, and test algorithms, which are the primitive inputs used in the in-house memory fault simulator. The fault simulator was developed in collaboration with a third-party company and validated by comparing the FCs of published test algorithms. Each of the primitive inputs used in this work is discussed in the following subsections.

A. SAMPLE MEMORY MODEL

The cell-level structure of memory can be described using a memory model interface. The model supports scrambling both address and data bits [11] to design an arbitrary cell-level structure of memory and port mappings of the cells [12], [13].

A bank of quad data rate II SRAM (QDRII SRAM) [14] is modeled as the example memory used in this work. Fig. 1 shows a bank with eight blocks in the QDRII SRAM model. Blocks with the same number have the same address mapping, and there are four different address mappings in the bank (address scrambled). The cells in a bank are connected to four bank data input/output (I/O) ports, which will eventually be mapped to four data I/O ports in the memory. The port connections through the banks are also scrambled (data scrambled). In this work, address scrambling information is mainly used as a key consideration in the implementation of test algorithms.

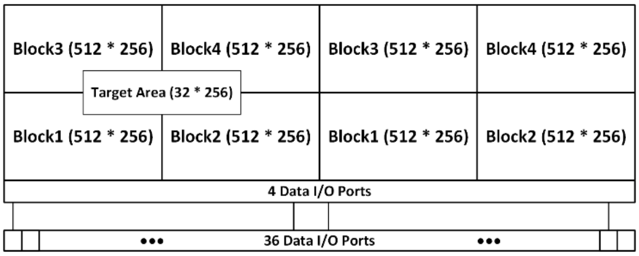


FIGURE 1. **8-block structure for a bank in QDRII SRAM model.**

The fault simulation is performed by specifying a target area that can be chosen by users. In this example, the target area was selected to include the partial areas of four blocks, as shown in Fig. 1. The size of the target area is 1 kB (32 × 256 cells). The cells in the target area are also connected to four I/O ports; thus, the target area can be thought of as 4-bit word-oriented memory.

B. TARGET FAULT MODELS AND BACKGROUND

The memory FMs used in this study are divided into three classes: static faults (SFs), dynamic faults (DFs), and neighborhood pattern-sensitive faults (NPSFs). SFs are FMs where the corresponding faults are sensitized by at most one memory operation [15], [16], whereas DFs are FMs where the corresponding faults are sensitized by multiple operations [15], [17]. NPSFs are FMs in which the base cell (victim cell) is affected by deleted neighborhood (DN) cells (aggressor cells) [9], [18]. Conventional NPSFs can be classified as Type-1 and Type-2 NPSFs depending on the numbers and locations of DN cells. In this work, only Type-1 NPSFs were considered.

Based on the number of cells involved, FMs can be classified into single- and multiple-cell faults; the former category represents faults sensitized by the state or operation of a single cell, whereas the latter indicates the faults sensitized by the values or operations of multiple cells. Coupling faults (CFs) and NPSFs belong to the multiple-cell fault class because faults are sensitized by the interaction between two cells and multiple cells, respectively.

In this work, the notation of FMs adopts the fault primitive concept in [15]–[17]. $<S/F/R>$ is the notation for single-cell faults, where S is the sensitizing operation sequence (SOS) that excites the fault, F is the faulty cell value, and R is the logic value when the last SOS is a read operation. As far as CFs are concerned, the fault primitive of $<S_A; S_V/F/R>$ is used, where S_A and S_V indicate the SOSs for the aggressor and victim cells, respectively. S can be written as $\alpha(w_a\beta)(r_b\gamma), \ldots,$ where $\alpha, \beta, \gamma, \ldots, \in \{0, 1\}$ are the cell values and $a, b, \ldots, \in \{1, 2, \ldots, \}$ are the temporal units; w and r indicate the write and the read operations, respectively. For example, $0\,w_1 1$ means that the fault can be sensitized when the value of 1 is written in the cell after a

TABLE 1. List of static faults.

FM name	FM subcase	Fault primitive
Stuck-at fault (SAF)[a]	SAF-x	< * / x’ / – >
Transition fault (TF)	TF-xwx’	< xwx’ / x / – >
State fault (STF)	STF-x	< x / x’ / – >
Write disturb fault (WDF)	WDF-xwx	< xwx / x’ / – >
Read disturb fault (RDF)	RDF-xrx	< xrx / x’ / x’ >
Deceptive read disturb fault (DRDF)	DRDF-xrx	< xrx / x’ / x >
Incorrect read fault (IRF)	IRF-xrx	< xrx / x / x’ >
Idempotent CF (CFid)	CFid-xwx’-y	< xwx’ ; y / y’ / - >
State coupling fault (CFst)	CFst-x-y	< x ; y / y’ / - >
Inversion CF (CFinv)	CFinv-xwx’	< xwx’ ; a[b] / a’ / - >
Disturb CF (CFds)	CFds-xwx-y CFds-xwx’-y CFds-xrx-y	< xwx ; y / y’ / - > < xwx’ ; y / y’ / - > < xrx ; y / y’ / - >
Transition CF (CFtr)	CFtr-x-ywy’	< x ; ywy’ / y / - >
Write disturb CF (CFwd)	CFwd-x-ywy	< x; ywy / y’ / - >
Read disturb CF (CFrd)	CFrd-x-yry	< x ; yry / y’ / y’ >
Deceptive read disturb CF (CFdrd)	CFdrd-x-yry	< x ; yry / y’ / y >
Incorrect read CF (CFir)	CFir-x-yry	< x ; yry / y / y’ >

[a] In this case, a special character was adopted, as used in [15]. The notation “*” denotes any operation or state that can sensitize the SAFs.

[b] This is an arbitrary value, which can be either 0 or 1.

TABLE 2. List of dynamic faults.

FM name	FM subcase	Fault primitive
Dynamic read disturb fault (dRDF)	dRDF-xwyry dRDF-xrxrx	< xwyr$_1$y / y’ / y’ > < xrxr$_1$x / x’ / x’ >
Dynamic deceptive read disturb fault (dDRDF)	dDRDF-xwyry dDRDF-xrxrx	< xwyr$_1$y / y’ / y > < xrxr$_1$x / x’ / x >
Dynamic incorrect read fault (dIRF)	dIRF-xwyry dIRF-xrxrx	< xwyr$_1$y / y / y’ > < xrxr$_1$x / x / x’ >
Dynamic transition fault (dTF)	dTF-xwywy’ dTF-xrxwx’	< xwyw$_1$y’ / y / – > < xrxw$_1$x’ / x / – >
Dynamic write disturb fault (dWDF)	dWDF-xwywy dWDF-xrxwx	< xwyw$_1$y / y’ / – > < xrxw$_1$x / x’ / – >
Dynamic disturb CF (dCFds)	dCFds-xwyry-z dCFds-xwywy-z dCFds-xwywy’-z dCFds-xrxwy-z dCFds-xrxrx-y	< xwyr$_1$y ; z / z’ / - > < xwyw$_1$y ; z / z’ / - > < xwyw$_1$y’ ; z / z’ / - > < xrxw$_1$y ; z / z’ / - > < xrxr$_1$x ; y / y’ / - >
Dynamic read disturb CF (dCFrd)	dCFrd-x-ywzrz dCFrd-x-yryry	< x ; ywzr$_1$z / z’ / z’ > < x ; yryr$_1$y / y’ / y’ >
Dynamic deceptive read disturb CF (dCFdrd)	dCFdrd-x-ywzrz dCFdrd-x-yryry	< x ; ywzr$_1$z / z’ / z > < x ; yryr$_1$y / y’ / y >
Dynamic incorrect read CF (dCFir)	dCFir-x-ywzrz dCFir-x-yryry	< x ; ywzr$_1$z / z / z’ > < x ; yryr$_1$y / y / y’ >
Dynamic transition CF (dCFtr)	dCFtr-x-ywzwz’ dCFtr-x-yrywy’	< x ; ywzw$_1$z’ / z / - > < x ; yryw$_1$y’ / y / - >
Dynamic write disturb CF (dCFwd)	dCFwd-x-ywzwz dCFwd-x-yrywy	< x ; ywzw$_1$z / z’ / - > < x ; yryw$_1$y / y’ / - >

temporal unit of 1 if the current cell value is 0. If the timing is irrelevant, the temporal unit is omitted.

Tables 1 and 2 list the SFs and DFs included in this work, respectively [15]–[17]. The first column shows the FM names and abbreviations of the corresponding FMs. The leading letter “d” in the FM name abbreviations was used to denote DFs. The second column presents all subcases of the corresponding FMs, where $x, y, z \in \{0, 1\}$, and the third column shows the corresponding fault primitives for each FM subcase. The apostrophe (’) following x, y, or z indicates the logical negation of its value.

TABLE 3. List of neighborhood pattern-sensitive faults.

FM name	FM subcase	Fault primitive
Active NPSF	NPSF-xyz↑-t	< xyz↑; t / t’ >
	NPSF-xyz↓-t	< xyz↓; t / t’ >
	NPSF-x↑yz-t	< x↑yz; t / t’ >
	NPSF-x↓yz-t	< x↓yz; t / t’ >
	NPSF-↑xyz-t	< ↑xyz; t / t’ >
	NPSF-↓xyz-t	< ↓xyz; t / t’ >
	NPSF-xy↑z-t	< xy↑z; t / t’ >
	NPSF-xy↓z-t	< xy↓z; t / t’ >
Passive NPSF	NPSF-xyzt-↑	< xyzt ; ↑ / 0 >
	NPSF-xyzt-↓	< xyzt ; ↓ / 1 >
Static NPSF	NPSF-xyzt-0	< xyzt ; 0 / 1 >
	NPSF-xyzt-1	< xyzt ; 1 / 0 >

Table 3 illustrates all of the NPSFs considered in this work. The NPSF notations in [18] were adopted and expressed as $<S_N S_W S_E S_S; S_B/F>$, where S_N, S_W, S_E, S_S, S_B, $F \in \{\uparrow, \downarrow, 0, 1\}$; ↑ and ↓ indicate the up- and down-transition write operations, respectively; S_N, S_W, S_E, and S_S designate the SOSs for the DN cells on the north, west, east, and south of the base cell, respectively; and S_B and F refer to the SOS and the faulty behavior, respectively, of the base cell. In Table 3, cell values are denoted as x, y, z, and t, where $x, y, z, t \in \{0, 1\}$.

By providing the FMs in the tables in this section, the number of FMs of each FM class for FC evaluation in this work is clearly disclosed. The numbers of FM subcases in Tables 1, 2, and 3 are 56, 126, and 192, respectively. For the rest of the discussions, the total numbers of subcases (56, 126, and 192) are assumed as the total set when discussing the FC results.

C. TEST ALGORITHMS WITH 100% FC OF SFs, DFs, AND NPSFs

Table 4 shows the March algorithms used in this work to perform comparative FC analyses with varying physical memory characteristics. The notations of the March algorithms in [10] are used. The algorithms were selected to cover 100% of the three different classes of FMs mentioned in the previous subsection. March MSS, March MD2, and March 12N algorithms were targeted to detect all SFs [16] (refer to Table 1), two-operation DFs [17] (refer to Table 2), and NPSFs [18] (refer to Table 3), respectively; March MD2 also detects FMs detected by March MSS. In this work, the March algorithms were applied to the 4-bit word-oriented memory unless specified otherwise (refer to Fig. 1).

In addition to the March algorithm specification in Table 4, the algorithm can be implemented through an address sequence (AS) and a data background (DB), as defined in [19]. Any number of implemented algorithms can be included in building a test suite (TS). In this work, three TSs, namely the March MSS TS, March MD2 TS, and March 12N TS, were constructed.

TABLE 4. March algorithms used in the fault simulations.

March algorithm	Description
March MSS [16]	$\{\Uparrow(wD);\ \Uparrow(rD, rD, w\overline{D}, w\overline{D});\ \Uparrow(r\overline{D}, r\overline{D}, wD, wD);$ M0 M1 M2 $\Downarrow(rD, rD, w\overline{D}, w\overline{D});\ \Downarrow(r\overline{D}, r\overline{D}, wD, wD);\ \Uparrow(\mathrm{r}D)\}$ M3 M4 M5
March MD2 [17]	$\{\Uparrow(wD);$ M0 $\Uparrow(rD, w\overline{D}, w\overline{D}, r\overline{D}, w\overline{D}, w\overline{D}, r\overline{D}, wD, wD, rD,$ $wD, wD, rD, wD, w\overline{D}, wD, w\overline{D});$ M1 $\Uparrow(r\overline{D}, wD, wD, rD, wD, wD, rD, w\overline{D}, w\overline{D}, r\overline{D},$ $w\overline{D}, w\overline{D}, r\overline{D}, w\overline{D}, wD, w\overline{D}, wD);$ M2 $\Downarrow(rD, w\overline{D}, r\overline{D}, w\overline{D}, r\overline{D}, r\overline{D}, r\overline{D}, wD, rD, wD,$ $rD, rD, rD, wD, w\overline{D}, wD, w\overline{D});$ M3 $\Downarrow(r\overline{D}, wD, rD, wD, rD, rD, rD, w\overline{D}, r\overline{D}, w\overline{D},$ $r\overline{D}, r\overline{D}, r\overline{D}, w\overline{D}, wD, w\overline{D}, wD);$ M4 $\Uparrow(\mathrm{r}D)\}$ M5
March 12N [18]	$\{\Uparrow(wD);\ \Uparrow(rD, w\overline{D}, wD);\ \Uparrow(rD, w\overline{D});$ M0 M1 M2 $\Uparrow(r\overline{D}, wD);\ \Uparrow(r\overline{D}, wD, w\overline{D});\ \Uparrow(rD)\}$ M4 M5 M6

The March MSS TS consists of eight implemented March MSS algorithms, and each algorithm is implemented by one of the eight exhaustive DBs for the 4-bit word [10] and a Count AS, in which eight-exhaustive DBs for the 4-bit word are referred to as 4-bit word DBs. In the Count AS, the logical address simply increases (decreases) from *0* to $n-1$ ($n-1$ to *0*), where n is the number of logical addresses; the physical address movement of the memory by the Count AS directly depends on the address scrambling of the memory. The March MD2 TS is built in the same manner; it consists of eight implemented March MD2 algorithms with 4-bit word DBs and the Count AS.

The March 12N TS also consists of eight implemented March 12N algorithms, and each algorithm is implemented by one of the eight dedicated DBs for complete NPSF detection [18] and the FastY AS [19], [20]; in the rest of this paper, the eight dedicated DBs for NPSF detection are referred to as the NPSF DBs. To properly generate the FastY AS and NPSF DBs, address scrambling information is required.

III. FAULT SIMULATION RESULTS OF SFs, DFs, AND NPSFs

In this section, the results of the fault simulations performed using the FMs and TSs are presented. Furthermore, the simulated results were validated by comparing the FCs of published results.

In this work, the FM minimum coverage (FMMC) is proposed as a new metric to show the percentage of target FMs that have an FC above a threshold, which is referred to as the minimum FC. FMMC(x) can be expressed as

$$FMMC(x) = \left(\frac{Number\ of\ FMs\ with\ an\ FC\ larger\ than\ or\ equal\ to\ x}{Total\ target\ FMs}\right) \times 100, \qquad (1)$$

where x is the minimum FC value, which can be any integer value from 100, 99, ..., 1, 0 (unit: %). The FC for an FM can be expressed as

$$FC = \left(\frac{Number\ of\ detected\ faults\ in\ an\ FM}{Total\ target\ faults\ in\ an\ FM}\right) \times 100. \qquad (2)$$

Fig. 2 shows the FMMCs of SFs for March MSS, March MD2, and March 12N TSs. The x-axis shows the minimum FC values from 100% to 0%, and the y-axis shows the FMMC(x) of the corresponding x values.

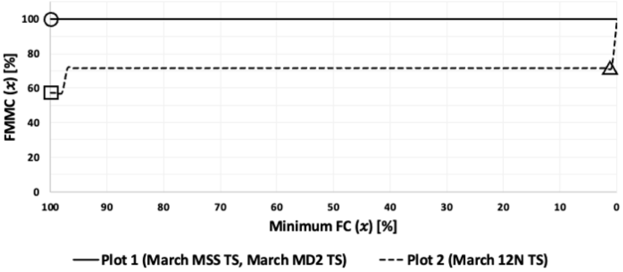


FIGURE 2. FMMCs of SFs by March MSS, March MD2, and March 12N TSs.

The FMMCs for both March MSS and MD2 TSs are the same, as shown in Plot 1 of Fig. 2. In Plot 1, FMMC(100) is 100% (see the circle), which means that the two TSs have 100% FCs for all SFs. These results match the published results in [16] and [17].

In the rest of this paper, the term "lucky FMMC" is used to refer to the FMMC for FMs that are not intended to be detected by the TS. Accordingly, the March 12N TS has the lucky FMMC for the SFs. The lucky FMMC(100) is equal to 57.14% (see the square), which indicates that 32 SFs have 100% FCs. The lucky FMMC(1) is equal to 71.43% (see the triangle), which shows that 16 SFs have less than 1% FCs. The lucky FMMC result is understandable because SFs were not detection targets of the March 12N TS.

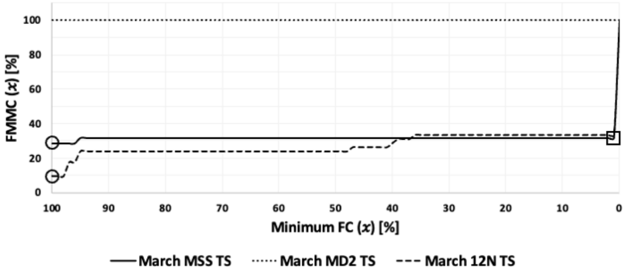


FIGURE 3. FMMCs of DFs by March MSS, March MD2, and March 12N TSs.

Fig. 3 shows the FMMCs of DFs for the three TSs. The FMMC(100) of March MD2 TS is 100%, which means that March MD2 TS has 100% FCs for all DFs. This result concurs with the results reported in [17].

The March MSS and March 12N TSs have the lucky FMMCs for the DFs. The lucky FMMCs(100) are 28.57% and 9.52% for the March MSS and March 12N TSs, respectively (see the circles). Thus, 36 DFs and 12 DFs can have 100% FCs by the March MSS and March 12N TSs, respectively. The lucky FMMCs(1) are 31.74% and 33.33%, which indicates that 86 FMs and 84 FMs can have less than 1% FCs by the March MSS and March 12N TSs, respectively (see the square). Unlike the lucky FMMC result in Fig. 2, the March MSS and March 12N TSs have relatively low lucky FMMCs for all minimum FC values.

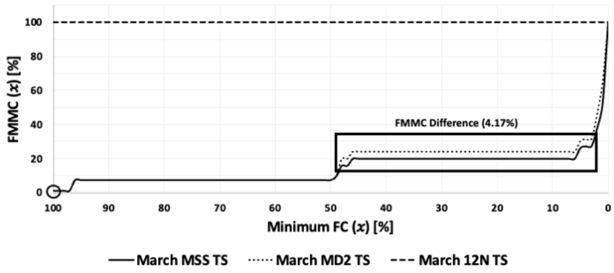


FIGURE 4. FMMCs of NPSFs by March MSS, March MD2, and March 12N TSs.

Fig. 4 shows the FMMCs of the NPSFs for the three TSs. The FMMC(100) of March 12N TS is 100%, which means that all NPSFs can be fully detected by the March 12N TS; this FMMC result is the same as the result reported in [18].

Fig. 4 shows the low lucky FMMCs for the March MSS and March MD2 TSs. The lucky FMMCs(100) are identical and equal to 1.04%, which means that only two FMs have 100% FCs. It is interesting to note that although the complexity of the March MD2 TS (560N) is approximately four times greater than that of March MSS (144N), their FCs hardly differ. The lucky FMMC values are the same until the minimum FC of 49%, and below the minimum FC of 49%, the lucky FMMC difference remains nearly constant at 4.17% (see the square). From Figs. 3 and 4, it can be observed that the test algorithms can have almost equally low FC values for non-target FMs regardless of their complexities.

IV. FAULT COVERAGE EXAMINATION WITH PHYSICAL MEMORY CHARACTERISTICS

In this section, the change in FC is explored when the physical characteristics of memory are additionally considered. Such coverage analysis provides the FC penalty when the electrical and structural characteristics of memory are not properly considered. It is not uncommon to see that such constraints are not properly considered when testing memories.

A. ELECTRICAL MASKING EFFECTS

In this subsection, the FCs of March MSS and March MD2 are re-evaluated considering the electrical cell level variations due to the defect size.

As described in Section I, faulty RAM behavior highly depends on the size of the resistive defects. For example, [3] shows the different electrical cell levels for repeated non-transition write operations after the first transition write operations ($1w0w0, \ldots,$ or $0w1w1, \ldots,$); according to the defect size, the number of write-operation attempts to write successful 0 or 1 values varies.

When such an electrical cell level variation is blended in the FMs, some of the SFs (DFs) may not be detected by March MSS (March MD2). We can consider one specific TF-0w1 case when March MSS is performed with the all-zero DB [19], [20] (also called solid DB). TF-0w1 can be sensitized by the first $w1$ in M1 as it fails to undergo a transition from 0 to 1; the state of the cell can be electrically close to 0.5 V_{DD}, depending on the defect size. If the second $w1$ in M1 is applied in such an electrical state, the effect of TF-0w1 is likely to disappear with a successful cell-state transition. The TF-1w0 effect can also disappear in this way after successive $w0$ operations in M2.

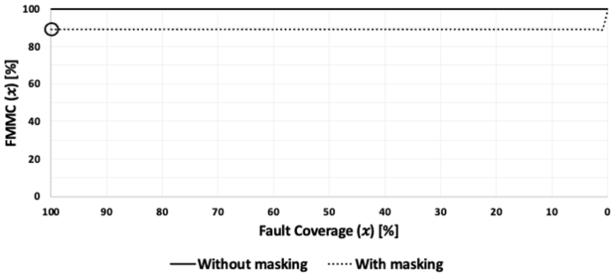


FIGURE 5. FMMCs of SFs by the March MSS TS with/without considering electrical masking.

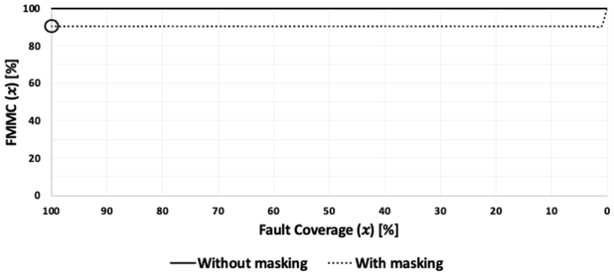


FIGURE 6. FMMCs of DFs by the March MD2 TS with/without considering electrical masking.

The above TF-0w1 example is a representative case where the fault is masked by the electrical level uncertainties when the SOS of the FM is applied twice in a row without detecting a fault after the first SOS is applied. In this work, we refer to this masking case as electrical masking. Because electrical masking can occur sufficiently considering variable defect sizes, it is necessary to re-evaluate FC with electrical masking in mind.

Figs. 5 and 6 show the FMMCs of SFs and DFs yielded by the March MSS and March MD2 TSs, respectively, for the FMs with/without considering the electrical masking characteristic. Without electrical masking, March MSS and March MD2 provided 100% FCs for all SFs and DFs, respectively, as demonstrated in the previous section. However, when the electrical masking characteristic existed in memory, the FMMCs of SFs and DFs dropped to 89.28% and 90.48%, respectively, for all FC values (see the circles in Figs. 5 and 6), which means that there are 10.72% SFs and 9.52% DFs that have FCs less than 1% or were never detected. As expected, TFs (TF-0w1 and TF-1w0) could not be detected by the March MSS TS when the electrical masking characteristic was considered.

B. ADDRESS SCRAMBLING EFFECTS

The utilization of address scrambling allows the use of the inherent characteristics of the memory to design/implement efficient tests or schemes to detect/correct target FMs or errors [11], [18], [21], [22]. However, because access to address scrambling information is highly limited, in most situations, test algorithms are implemented without such information. Thus, FC analysis considering the address scrambling effect is very valuable; in this subsection, the address scrambling effect on FC is investigated by implementing the March 12N algorithm according to the availability of address scrambling and re-evaluating FCs of NPSFs with implemented algorithms.

As mentioned in Section II-C, March 12N should be implemented eight times with the FastY AS and NPSF DBs to construct the March 12N TS, so address scrambling information is required. If the March 12N TS is constructed without address scrambling, it is easy to predict the FC reduction in NPSFs due to the incomplete algorithm implementations. In this study, when March 12N was implemented without scrambling information, the NPSF DBs generated by logical addresses and the Count AS were used. Note that the two least significant bits (LSBs) of the physical row and column addresses must be used for the NPSF DB generation [18], [23]; because the physical addresses cannot be directly manipulated without scrambling information, the two LSBs of the logical row and column addresses were used instead.

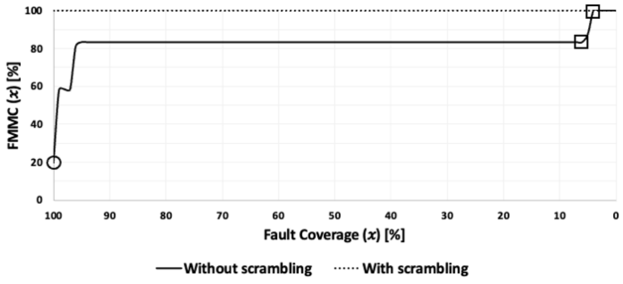


FIGURE 7. FMMCs of NPSFs for the March 12N implemented with/without address scrambling.

Fig. 7 shows the FMMCs of NPSFs by the March 12N TS constructed with/without the availability of internal memory physical structure, i.e., address scrambling. Fig. 7 reveals that severe FC reduction occurs if address scrambling is not available when March 12N is implemented. The FMMC(100) is significantly reduced to 19.79% (see the circle), and we can observe that FMMC(6) = 83.3% and FMMC(4) = 100% (see the squares), which means that 32 NPSFs have 4%–6% FCs; these FC values cannot be observed when the algorithm is completely implemented with address scrambling.

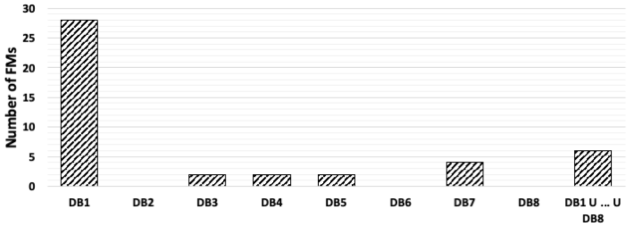


FIGURE 8. Number of NPSFs with 100% FCs for each March 12N algorithm implemented without scrambling information.

Fig. 8 shows the number of NPSFs with 100% FCs for each of the March 12N algorithms implemented without scrambling information; each of the implemented algorithms is identified by the number after "DB." Additionally, the last column in Fig. 8 shows the number of NPSFs with 100% FCs only if all FCs measured from DB1, DB2, DB8 are combined. From Fig. 7 (see the circle) and Fig. 8 (see the last column), it can be seen that the number of NPSFs with 100% FCs is 38 (=192 × 0.1979) and 6 of them have 100% FCs only if the FCs from different DBs are combined; a total of 32 FMs have 100% FCs when the FCs from different DBs are individually considered.

Fig. 8 demonstrates that the NPSFs that are not sensitive to address scrambling are mostly detected by DB1; the corresponding DB is the all-zero DB, which is not dependent on address scrambling; thus, there is no reduction in FC by the algorithm implemented with this DB. The remaining DBs are dependent on address scrambling, resulting in FC reductions in many NPSFs, as shown in Fig. 7, when the DBs are generated with the logical addresses.

C. ELECTRICAL NEIGHBORHOOD EFFECTS

Generally, NPSFs consider the Type-1 neighborhood, as shown in Fig. 9(a). Such a neighborhood consideration means that the four cells closest to the base cell have the strongest electrical interferences with the base cell, which is suitable when memory cell arrays are composed of a rectangular grid. Therefore, Type-1 NPSFs are very likely to occur, and various studies have been performed to test Type-1 NPSFs [9], [24]–[26].

However, the physically adjacent Type-1 neighborhood does not always indicate the strongest electrical interferences among the cells; in this study, the strongest electrical interferences among the cells are referred to as the electrical

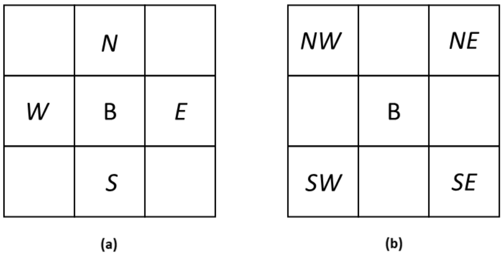


FIGURE 9. (a) Type-1 and (b) diagonal neighborhoods for NPSFs.

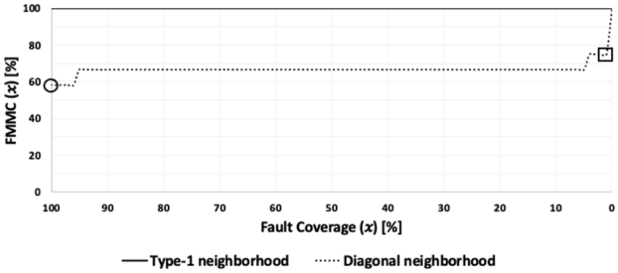
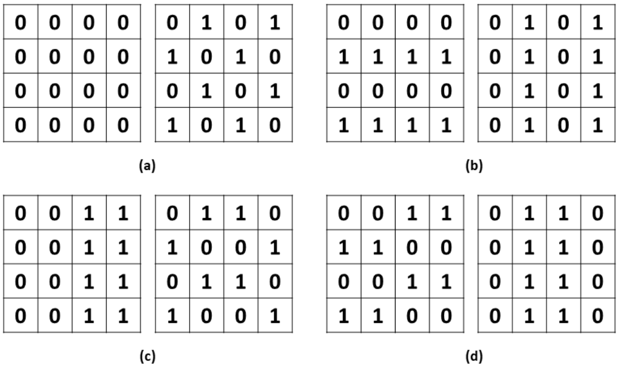


FIGURE 10. FMMCs of two-neighborhood NPSFs by the March 12N TS.



0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

0	1	0	1
1	0	1	0
0	1	0	1
1	0	1	0

(a)

0	0	0	0
1	1	1	1
0	0	0	0
1	1	1	1

0	1	0	1
0	1	0	1
0	1	0	1
0	1	0	1

(b)

0	0	1	1
0	0	1	1
0	0	1	1
0	0	1	1

0	1	1	0
1	0	0	1
0	1	1	0
1	0	0	1

(c)

0	0	1	1
1	1	0	0
0	0	1	1
1	1	0	0

0	1	1	0
0	1	1	0
0	1	1	0
0	1	1	0

(d)

FIGURE 11. Four pairs of DBs that provide identical FCs for diagonal NPSFs.

neighborhood. According to the physical structure of the memory, [27] and [28] proposed the delta-type and t-type neighborhoods, respectively, to properly reflect electrical neighborhoods, and previous studies have shown that electrical neighborhoods should be defined differently depending on the physical layout of the memory. Accordingly, this subsection considers the diagonal neighborhood as the one possible electrical neighborhood, as shown in Fig. 9(b); then, the FCs of the March 12N TS are re-evaluated with respect to diagonal NPSFs (=NPSFs with the diagonal neighborhood) to examine the effect of the electrical neighborhood change on FC.

Fig. 10 shows the FMMCs of NPSFs for the two neighborhoods shown in Fig. 9, measured by the March 12N TS. When the Type-1 neighborhood changes to the diagonal neighborhood, the FMMC(100) is reduced to 58.33% (see the circle), which means that the FCs of 80 NPSFs are reduced. The FMMC(1) is 75% (see the square), indicating that 48 NPSFs are not detected at all or less than 1% are detected.

The reduction in FC occurs because there are some DBs that provide identical FCs for diagonal NPSFs. Fig. 11 shows four pairs of DBs; two DBs from each pair provide the same FCs for the diagonal NPSFs. For example, each of the algorithms implemented by the two DBs in Fig. 11(a) yields the same FCs for the diagonal NPSFs. Figs. 10 and 11 demonstrate that the FCs of NPSFs provided by the existing test algorithms are clearly reduced when the electrical neighborhood of NPSFs is changed owing to the change in the memory physical mechanism for any reason.

V. CONCLUSION

In this study, fault coverages of the existing memory test algorithms were re-evaluated by considering three memory physical characteristics: electrical masking, address scrambling, and electrical neighborhoods. To this end, the fault simulations for three fault model classes, including 56 static faults, 126 dynamic faults, and 192 neighborhood pattern-sensitive faults (NPSFs), were performed on the example SRAM model; March MSS, March MD2, and March 12N algorithms targeting 56 static faults, 126 dynamic faults, and 192 NPSFs, respectively, were selected. For the fault simulation, a fault simulator from a third party was used.

The simulation results were presented in two parts. The first part of the result showed the fault coverages for the fault models targeted by each algorithm, and it was confirmed that the same fault coverages as in the previous studies were observed. In addition, the lucky fault coverage was measured, which is the fault coverage for fault models that the test algorithm is not intended to detect. From the perspective of lucky fault coverage, it was observed that the algorithms could have the same level of low coverage regardless of their complexities.

Furthermore, we discussed the fault coverage reduction in the test algorithms when the memory physical characteristics were considered. When electrical masking was considered, fault coverage reductions of 10.72% static faults and 9.52% dynamic faults were observed for March MSS and March MD2, respectively. When address scrambling was not available or the changed electrical neighborhood was properly considered, reductions in fault coverages of 80.21% and 41.67% NPSFs, respectively, were observed for March 12N.

Memory test algorithms are typically developed to detect target fault models that require fully addressing the memory failure mechanism. However, fully representing the semiconductor/circuit failure mechanisms in fault models is not always feasible. In this work, such deficiencies were demonstrated by showing fault coverage reductions in existing algorithms according to the physical characteristics of the memory. This suggests that the conventional test algorithm development process that detects target fault models is no longer complete and needs improvement.

REFERENCES

[1] D. Klein, "The history of semiconductor memory: From magnetic tape to NAND flash memory," *IEEE Solid-State Circuits Mag.*, vol. 8, no. 2, pp. 16–22, Spring 2016.

[2] A. Jee and F. J. Ferguson, "Carafe: An inductive fault analysis tool for CMOS VLSI circuits," in *Proc. IEEE VLSI Test Symp.*, Atlantic City, NJ, USA, Apr. 1993, pp. 92–98.

[3] Z. Al-Ars, S. Hamdioui, A. J. van de Goor, and S. Al-Harbi, "Influence of bit-line coupling and twisting on the faulty behavior of DRAMs," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 12, pp. 2989–2996, Dec. 2006.

[4] S. Hamdioui, M. Taouil, and N. Z. Haron, "Testing open defects in memristor-based memories," *IEEE Trans. Comput.*, vol. 64, no. 1, pp. 247–259, Jan. 2015.

[5] S. Hamdioui, Z. Al-Ars, and J. A. van de Goor, "Opens and delay faults in CMOS RAM address decoders," *IEEE Trans. Comput.*, vol. 55, no. 12, pp. 1630–1639, Dec. 2006.

[6] R. C. Baumann, "Radiation-induced soft errors in advanced semiconductor technologies," *IEEE Trans. Device Mater. Rel.*, vol. 5, no. 3, pp. 305–316, Sep. 2005.

[7] K. Park, C. Lim, D. Yun, and S. Baeg, "Experiments and root cause analysis for active-precharge hammering fault in DDR3 SDRAM under 3× nm technology," *Microelectron. Rel.*, vol. 57, pp. 39–46, Feb. 2016.

[8] A. J. Walker, S. Lee, and D. Beery, "On DRAM rowhammer and the physics of insecurity," *IEEE Trans. Electron Devices*, vol. 68, no. 4, pp. 1400–1410, Apr. 2021.

[9] A. J. van de Goor, *Testing Semiconductor Memories: Theory and Practice*, 1st ed. Chichester, U.K.: Wiley, 1991, pp. 54–56.

[10] L.-T. Wang, C.-W. Wu, and X. Wen, *VLSI Test Principles and Architectures: Design for Testability*, 1st ed. San Francisco, CA, USA: Morgan Kaufmann, 2006.

[11] A. J. van de Goor and I. Schanstra, "Address and data scrambling: Causes and impact on memory tests," in *Proc. 1st IEEE Int. Workshop Electron. Design, Test Appl.*, Christchurch, New Zealand, Jan. 2002, pp. 128–136.

[12] B. Keeth, R. J. Baker, B. Johnson, and F. Lin, *DRAM Circuit Design: Fundamental and High-Speed Topics*, 2nd ed. Hoboken, NJ, USA: Wiley, 2007.

[13] R. D. Adams, *High Performance Memory Testing: Design Principles, Fault Modeling and Self-Test*, 1st ed. Boston, MA, USA: Kluwer, 2003.

[14] *72M-Bit QDRII SRAM Datasheet, M19958EJ2V0DS00*, Renesas Electron., Tokyo, Japan, Mar. 2010.

[15] A. J. van de Goor and Z. Al-Ars, "Functional memory faults: A formal notation and a taxonomy," in *Proc. IEEE VLSI Test Symp.*, Montreal, QC, Canada, Apr./May 2000, pp. 281–289.

[16] G. Harutunyan, V. A. Vardanian, and Y. Zorian, "Minimal March tests for unlinked static faults in random access memories," in *Proc. IEEE VLSI Test Symp.*, Palm Springs, CA, USA, May 2005, pp. 53–59.

[17] G. Harutunyan, V. A. Vardanian, and Y. Zorian, "Minimal March tests for detection of dynamic faults in random access memories," *J. Electron. Test., Theory Appl.*, vol. 23, no. 1, pp. 55–74, Feb. 2007.

[18] K.-L. Cheng, M.-F. Tsai, and C.-W. Wu, "Neighborhood pattern-sensitive fault testing and diagnostics for random-access memories," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 21, no. 11, pp. 1328–1336, Nov. 2002.

[19] S. Hamdioui, Z. Al-Ars, A. J. van de Goor, and R. Wadsworth, "Impact of stresses on the fault coverage of memory tests," in *Proc. IEEE Int. Workshop Memory Technol., Design, Test.*, Taipei, Taiwan, Aug. 2005, pp. 103–108.

[20] S. Hamdioui, J. A. van de Goor, J. D. Reyes, and M. Rodgers, "Memory test experiment: Industrial results and data," *IEE Proc., Comput. Digit. Tech.*, vol. 153, no. 1, pp. 1–8, Jan. 2006.

[21] S. Khan, D. Lee, and O. Mutlu, "PARBOR: An efficient system-level technique to detect data-dependent failures in DRAM," in *Proc. IEEE/IFIP Int. Conf. Dependable Syst. Netw. (DSN)*, Toulouse, France, Jun. 2016, pp. 239–250.

[22] M. Kim, J. Choi, H. Kim, and H.-J. Lee, "An effective DRAM address remapping for mitigating rowhammer errors," *IEEE Trans. Comput.*, vol. 68, no. 10, pp. 1428–1441, Oct. 2019.

[23] R. R. Julie, W. H. Wan Zuha, and R. M. Sidek, "12N test procedure for NPSF testing and diagnosis for SRAMs," in *Proc. IEEE Int. Conf. Semiconductor Electron.*, Johor Bahru, Malaysia, Nov. 2008, pp. 430–435.

[24] J. P. Hayes, "Testing memories for single-cell pattern-sensitive faults," *IEEE Trans. Comput.*, vol. C-29, no. 3, pp. 249–254, Mar. 1980.

[25] D. S. Suk and S. M. Reddy, "Test procedures for a class of pattern-sensitive faults in semiconductor random-access memories," *IEEE Trans. Comput.*, vol. C-29, no. 6, pp. 419–429, Jun. 1980.

[26] V. Yarmolik, Y. Klimets, and S. Demidenko, "March PS(23N) test for DRAM pattern-sensitive faults," in *Proc. Asian Test Symp.*, Singapore, 1998, pp. 354–357.

[27] Y. Sfikas, Y. E. Tsiatouhas, and S. Hamdioui, "Layout-based refined NPSF model for DRAM characterization and testing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 6, pp. 1446–1450, Jun. 2014.

[28] D. Kang, S. Park, and S. Cho, "An efficient built-in self-test algorithm for neighborhood pattern and bit-line sensitive," *ETRI J.*, vol. 26, no. 6, pp. 520–534, Dec. 2004.

KISEOK LEE received the B.S. degree in electrical and communication engineering from Hanyang University, South Korea, in 2013, where he is currently pursuing the Ph.D. degree in electronics and communication engineering.

His research interests include memory test pattern optimization, memory fault modeling, and memory fault diagnosis. He is also conducting research related to random patterns to determine their benefits. His work also focuses on noise-inducing patterns in very large-scale integration (VLSI) circuits and systems with interconnect test patterns.

JEONGHWAN KIM received the B.S. degree in electronic engineering from Hoseo University, Asan, South Korea, in 2019. He is currently pursuing the Ph.D. degree in electronic engineering with Hanyang University.

His research interests include memory test methodologies, memory failure analysis at the system level, and memory fault diagnosis. He is also conducting research on memory reliability in a server-class system environment.

SANGHYEON BAEG (Member, IEEE) received the B.S. degree in electronic engineering from Hanyang University, Seoul, South Korea, in 1986, and the M.S. and Ph.D. degrees in electrical and computer engineering from The University of Texas at Austin, in 1988 and 1991, respectively. From 1994 to 1997, he was a Staff Researcher with Samsung Electronics Company Ltd., Kihung, South Korea. In 1995, he was dispatched to Samsung Semiconductor, Inc., San Jose, CA, USA, and worked as a member of the Technical Staff. In 1997, he joined Cisco Systems, Inc., San Jose, and worked as a Hardware Engineer, a Technical Leader, and a Hardware Manager. Since 2004, he has been working as a Professor with the School of Electronics and Communication Engineering, Hanyang University, Ansan, South Korea. His work has focused on reliable computing, soft errors, low-power content-addressable memory (CAM), and VLSI DFT implementation and methodologies. He is the holder of many U.S. patents in these fields. He was a recipient of the Inventor Recognition Award from Semiconductor Research Cooperation, in 1993. He was an IEEE 1149.6 Working Group Member, in 2003. He has been served as an Organizing Member for the Institute of Semiconductor Test of Korea, since 2012.