Parallel Testing of Parametric Faults in a Three-Dimensional Dynamic Random-Access Memory

PINAKI MAZUMDER

Abstract—**This paper presents a testable design of dynamic random-access memory (DRAM) architecture which allows one to access multiple cells in a word line simultaneously. The technique utilizes the two-dimensional (2D) organization of the DRAM and the resulting speedup of the conventional algorithms is considerable. This paper specifically investigates the failure mechanisms in the three-dimensional (3D) DRAM with trench-type capacitor. As opposed to the earlier approaches for testing parametric faults that employed sliding diagonal-type tests with $O(n^{3/2})$ complexity, the algorithms discussed in this paper are different and have $O(\sqrt{n/p})$ complexity, where p is the number of subarrays within the DRAM chip. These algorithms can be applied externally from the chip and also they can be easily generated for built-in self-test (BIST) applications.**

I. Introduction

SEMICONDUCTOR dynamic random-access memory (DRAM) is the highest beneficiary of the rapid growth of VLSI technology. As the device feature width is decreasing every year, the DRAM size is quadrupling every two to three years. Recently Nippon Telegraph and Telephone (NTT) Company has announced the development of 16-Mbit DRAM's, and by the turn of the decade it is expected that several manufacturers will fabricate 64-Mbit DRAM's employing 0.5-μm technology [1]. This enormous prospect of DRAM development cannot be economically exploited unless cost-efficient testing strategies are evolved to arrest the polynomial growth of testing cost with the increasing DRAM size. Conventional test algorithms like the sliding diagonal test and the GALDIA are employed to test the leakage currents and the faults caused by the variation of processing technology. The sliding diagonal test, which uses $4N^{3/2}$ memory cycles, requires over 8 h to test the ac parametric faults in a 16-Mbit DRAM chip having a 100-ns memory cycle time. The main objective of this paper is to demonstrate how, by employing a new design-for-testability approach, parametric faults in a DRAM can be tested in parallel. The new algorithms test multiple cells in a memory simultaneously and thereby the overall test complexity is reduced by more than 1000 times.

The problem of parallel testing has been addressed by other researchers in the past. In order to reduce the test time McAdams *et al.* [2] fabricated a 1-Mbit CMOS three-dimensional (3D) DRAM with design-for test functions. They partitioned the memory into eight subarrays and tested them concurrently. Their scheme simultaneously writes the same data on eight cells, which are identically located inside the different subarrays. During a READ operation when any of these eight cells is addressed, all eight cells are simultaneously accessed and their contents are compared by using a two-mode 8-bit parallel comparator. The resulting scheme which compares one cell from each subarray will be called in this paper *inter-subarray single-cell comparison* (Scheme 1). By using this scheme they reduced the test time by a factor of 5.2 times. Shah *et al.* [3] used a similar 16-bit parallel comparator in their 4-Mbit DRAM with trench-transistor cell and essentially reduced the test time complexity to that of a 256-kbit DRAM.

You and Hayes [4] have introduced the concept of parallel testing within the subarrays by applying simultaneously single-cell inter-subarray comparison over multiple cells within a subarray. The resulting scheme will be called here *inter-subarray multiple-cell comparison* (Scheme 2). They have reconfigured the memory subarray of size s bits into an s-bit cyclic shift register where the data recirculate whenever a READ operation is done. The reconfiguration was accomplished by introducing pass transistors on the bit lines which deteriorate both the sensitivity of the sense amplifiers by V_T (threshold voltage of the MOS devices) and the access time of the DRAM in normal mode of operation. In order to reduce the routing complexity, it is desirable to compare the adjacent subarrays only. Thus by this scheme the occurrence of a fault is detected by comparing only two cells which are simultaneously read. If both the cells are identically faulty, it fails to detect the fault. Moreover, the reconfiguration scheme is tailored to introduce parallelism in the sliding diagonal test and the proposed design of parallel testing cannot be adapted for a large class of functional faults, like coupling and static and dynamic pattern-sensitive faults [5].

Manuscript received December 14, 1987; revised March 23, 1988. This work was supported in part by the Army Research Office under the URI program, Contract DAAL03-87-K-0007, and by the Semiconductor Research Corporation.

The author is with the Center for High-Frequency Microelectronics, Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48105.

IEEE Log Number 8822147.

0018-9200/88/0800-0933$01.00 ©1988 IEEE

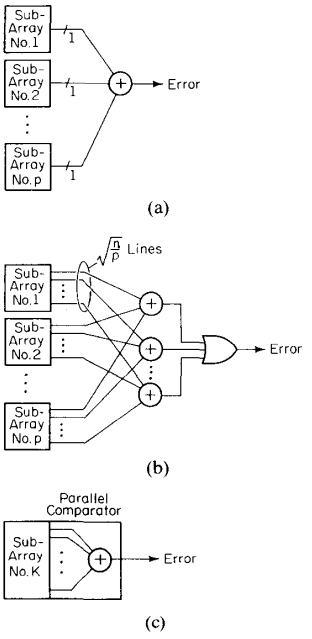


Fig. 1. Strategies for parallel memory testing. (a) Inter-subarray single-cell comparison. (b) Inter-subarray multiple-cell comparison. (c) Intra-subarray multiple-cell comparison.

This paper proposes a design-for-testability approach which does not make any inter-subarray comparisons. It writes the same data on multiple cells in a word line of a subarray in parallel, and in READ mode it compares these simultaneously written cells within a subarray. The resulting test scheme is called *intra-subarray multiple-cell comparison* (Scheme 3). These three schemes are shown in Fig. 1 and compared in Table I. The advantage of the proposed technique is that it is not constrained to any specific test procedure and the test vectors can be applied externally or generated by a built-in self-testing (BIST) circuit. It can speed up any existing test procedure of $O(n)$ test length by a factor of $O(\sqrt{pn})$. The paper investigates the parametric faults in a DRAM and proposes $O(\sqrt{n/p})$ algorithms to test the different parametric faults. Unlike in [4] the proposed technique does not modify the memory plane to introduce parallelism in the diagonal test, and thereby the normal memory performance (viz., access time, sense-amplifier sensitivity) does not degrade. The proposed design-for-testability technique employs very little overhead (only $2\sqrt{pn} + p\log_2 n - p\log_2 p + 12p$ transistors if the DRAM is organized into p square subarrays each of size $\sqrt{n/p} \times \sqrt{n/p}$) and needs only one transistor to fit within the 3λ inter-celler pitch width of the vertically integrated 3D DRAM. It will be shown that the modified bit-line decoders in Scheme 3 need $2\log_2\sqrt{n/p}$ extra transistors in each subarray to enable parallel access. Each 0/1 detector used in a subarray to test the occurrence of a fault will be implemented by $2\sqrt{n/p} + 12$ transistors. The circuits for the modified bit-line decoders and the 0/1 detectors are designed such that each transistor can fit within the pitch width of the high-density memory.

The rest of the paper has been organized as follows. Section II proposes a new design-for-testability technique which allows multiple cells to be tested in one memory cycle. Section III enumerates the different faults which occur due to variations in processing technology in a 3D DRAM; the algorithms for testing these parametric faults are presented in Section IV. The main contribution of this work is to propose a design-for-testability technique and to demonstrate how the parametric faults in a 3D DRAM using trench-type memory cells can be tested in parallel.

II. Design for Testability

The organization of the testable DRAM with augmented hardware is shown in Fig. 2. The memory is organized as a $b \times w = n$ matrix, where b is the number of bit lines and w is the number of word lines. The normal 1-out-of-b decoder is modified to select multiple bit lines during test mode. In test mode, it divides the b bit lines into g groups such that the bit line i belongs to group j, where $j = i \pmod g$. Thus a WRITE operation in test mode results in writing the content of the data-in buffer on all cells at the crosspoints of the selected word line and the bit lines in group j. In READ mode, the contents of the cells located at the crosspoints of the selected word-line and bit-line groups (say j) are read in parallel. Thus, a ZERO or ONE is entered in the data-out buffer if all the multiple-accessed cells contain ZERO or ONE, respectively. If the contents of all the cells are not identical, the data-out buffer may store a ZERO or ONE. It should be noted that it is not correct to assume that the resulting operation will be a wired-OR or wired-AND. On the contrary, it depends on the number of ZERO's and ONE's in the multiple-accessed cells. If almost all the multiple-accessed cells contain ONE's except a very few which contain ZERO's, then a ONE will be entered in the data-out buffer when the cells are read in parallel. A ZERO would have been entered if almost all the cells contained ZERO's and a few cells contained ONE's. To circumvent this problem, the contents of all the cells in a group are compared by a parallel comparator. In the event that all the cells do not have identical contents, the parallel comparator triggers an error latch to indicate that a fault has been detected by the test.

In contrast to the bit-line decoder, the word-line decoder is not modified and word lines are accessed one at a time. A parallel-word READ operation is not meaningful because two or more cells will be sensed by the same sense amplifier resulting in a wired-OR or a wired-AND operation. A parallel-WRITE operation through multiple word lines would require the sense amplifier to drive many cells at a time. For a moderate-size DRAM, this introduces high WRITE-cycle time delay. By increasing the physical size of the sense-amplifier driver, delay can be improved to a certain extent, however this increases power consumption, and because of its large gate capacitance, sense-amplifier

TABLE I
COMPARISON OF THREE PARALLEL TESTING STRATEGIES

Criterion	Scheme 1	Scheme 2	Scheme 3
Performance -- Parallelism	p	$O(\sqrt{pn})$	$O(\sqrt{pn})$
Degradation -- in Access Time	None	Large	Negligible
in Sense Amp Sensitivity	None	Large	None
Architecture Modification -- Decoder	Not Modified	Modified	Modified
Memory Plane	Not Modified	Modified	Not Modified
Comparator Size	One p-bit	$\sqrt{n/p} \times p$-bit	One $\sqrt{n/p}$-bit 0/1-Detector/Subarray
Routing Complexity	None	High	Very Low
Reliability	Moderate	Low, if only two cells are mutually compared; Moderate, if all p cells are mutually compared	Very High
Fault Coverage	Functional and Parametric	Only Parametric	Functional and Parametric

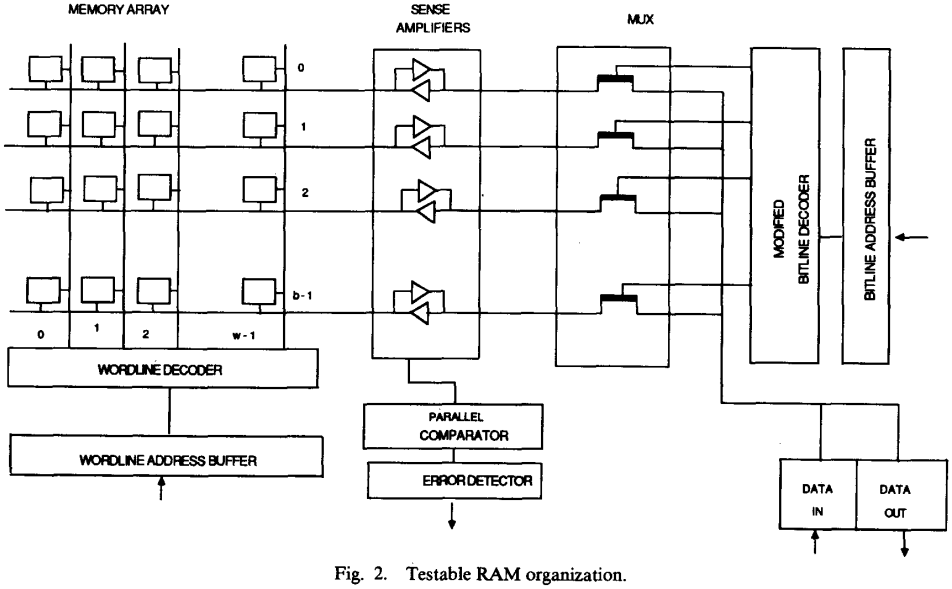


Fig. 2. Testable RAM organization.

slew rate decreases. Parallel bit-line READ and WRITE do not suffer from this drawback.

A. Modified DRAM Circuit

The modified CMOS decoder circuit is shown in Fig. 3. Transistors $Q_1, \cdots, Q_7$ with the transmission gate constitute a normal decoder circuit. In the clock phase ϕ_P, the transistor Q_1 turns on to precharge the common line connected to the address decoding transistors. If all the address bits, $a_0, \cdots, a_{k-1}$ are zeros, transistor Q_6 pulls up the OUT to ONE, and the corresponding bit line is selected. The signal ϕ_{EN} enables the transmission gate so that the decoder selects the bit line only after all address lines have changed. Transistors Q_8 and Q_9 have been added so that in the test mode the decoder output can be selected by applying $\overline{\text{SELECT}} = 0$ independent of the input address. In the normal mode of operation, $\overline{\text{SELECT}} = 1$ and the decoder output is selected by the address input $a_0, \cdots, a_{k-1}$. The operation of the modified decoder is shown by the voltage waveforms in Fig. 4. The modified decoder is simulated using SPICE and the degradation in decoding time due to addition of the extra transistors has been found to be approximately 0.1 ns.

The parallel comparator, which is essentially a multibit 0/1 detector, monitors the output of sense amplifiers connected to bit lines which are selected in parallel and detects the concurrent occurrence of either ZERO's or ONE's. If a selected bit line is different from the others, it triggers the error latch indicating the occurrence of a fault. Fig. 5 shows the parallel comparator and error detector. The p-channel transistors $T_1, \cdots, T_{m-1}$ are connected in parallel and detect concurrent occurrence of ONE in the bit lines. The n-channel transistors $P_1, \cdots, P_{m-1}$ are also connected in parallel and detect concurrent occurrence of ZERO in the bit lines. Transistors T_0 and P_0 are the precharge transistors while transistor P_m is the discharge transistor which remains cut off during the precharge phase and turns on during discharge clock phase $\phi 2$. Since the bit lines are divided into $g = 2$ classes, pass transistors

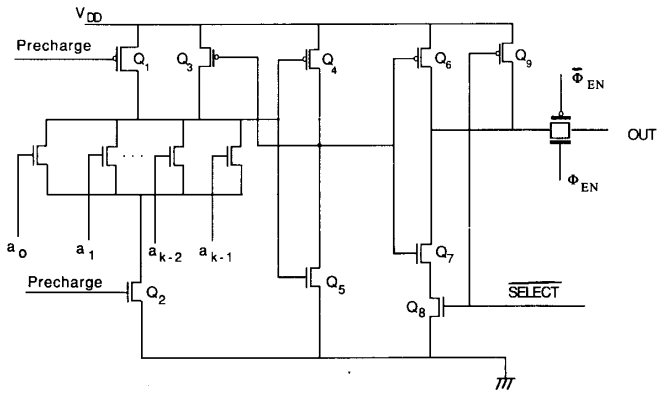


Fig. 3. Modified decoder circuit.

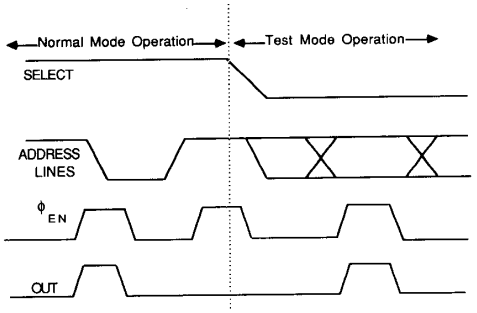


Fig. 4. Operation of modified decoder circuit.

are introduced so that only the odd or even bit lines are compared simultaneously. Signals L_1 and L_2 select these bit lines. Transistors S_0, S_1, and S_2 form a coincidence detector. If all the selected bit lines are ZERO or ONE, then either S_1 or S_2 conducts and the output of the detector is ZERO. The output of the coincidence detector is connected to an error latch through the pass transistor S_4 which isolates the error latch during the phase $\phi 1$. It may be noted that during the precharge phase the transistor S_0 will be directly shorted through the error amplifier if S_4 does not isolate the coincidence detector from the error amplifier. During phase $\phi 2$, the output of the coincidence detector is connected to the error amplifier through S_4. The error amplifier consists of transistors $V_0, \cdots, V_3$. The error latch output is ERROR = 0, when the selected bit lines are identical. If the bit lines are not identical, then both S_1 and S_2 remain cut off and the detector output is ONE. This triggers the error latch to set its output to ERROR = 1. During the WRITE phase and normal mode of operation, the error latch is clamped to zero by V_4. The error detector is inhibited by the discharge transistor P_m during the start of the READ phase when the sense-amplifier outputs are not identical because of sluggish changes in some of the sense amplifiers.

The design-for-testability approach has been applied over an experimental DRAM chip of 16 kbit. The chip overhead was found to be less than 1 percent. For multi-megabit DRAM's, this overhead will be less than 0.5 percent. The degradation in performance due to the modified decoder was less than 0.2 ns in memory cycle time. This is slightly larger than the value obtained from SPICE simulation of the modified decoder.

III. Parametric Faults in a 3D DRAM

A typical configuration of the 3D trench-type memory cell [3], [6] with p^+ sidewall doping is shown in Fig. 6. The access transistor is a PMOS transistor located within an n-well diffused over the p^+ substrate. A deep trench capacitor extends from the planarized surface through the n-well into the p^+ substrate. A conducting strap connects the p^+-doped polysilicon storage electrode inside the trench to the p^+ source region of the access transistor. With a thin composite insulator separating the polysilicon from the bulk silicon surrounding the trench, the storage capacitance comes primarily from the portions of the four trench sidewalls in the p^+ substrate region and the trench bottom. Some additional capacitance results from the four trench walls intersecting the n-well. The grounded p^+ substrate provides a very solid reference potential to the capacitor plate. The leakage currents due to process parameter variation have also been shown in the diagram. These currents are divided into four components: 1) weak-inversion current I_W from the storage area to the bit line; 2) field-inversion current I_F between the two adjacent cells; 3) gate leakage current I_G due to pin-hole defects in the gate oxide; and 4) the dark current I_B between the storage area and the p-type substrate. The weak-inversion current can degrade a stored ZERO by flow of minority carriers from the trench capacitor to the positively biased bit lines. Dark current which flows from the trench capacitor to the p^+ substrate can degrade stored ONE. It may also be observed that the cell forms a vertical parasitic FET device which occurs between the storage node and the substrate along the trench wall, gated by the node polysilicon as shown in Fig. 6.

The effects of the leakage currents result in parametric faults such as the bit-line voltage imbalance and the bit-line

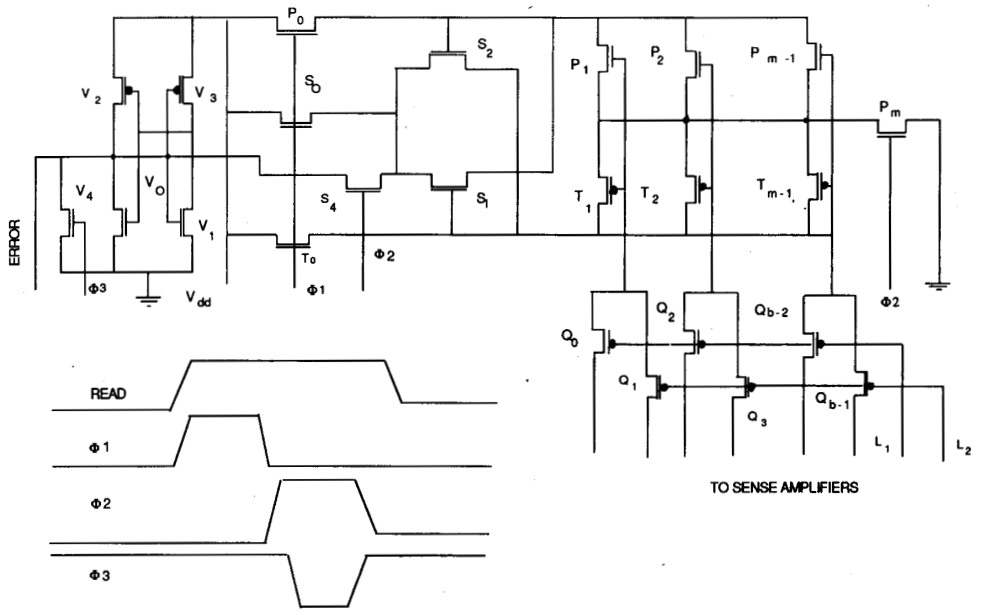


Fig. 5. Parallel comparator with error detector ($g = 2$).

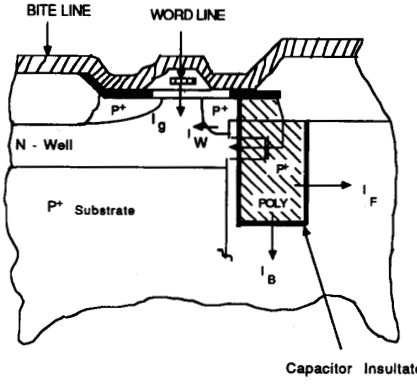


Fig. 6. 3D trench-type memory cell.

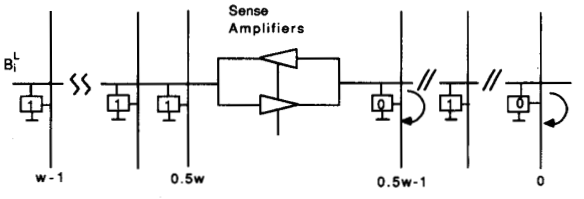


Fig. 7. Bit-line voltage imbalance.

to word-line crosstalk. The other types of parametric faults emanate due to a wide variation of timing signals in the decoding, address buffer, and peripheral circuits, such as the sense amplifiers. Incorrect timing between decoder enable, precharge clock, and decoder address signals may cause multiple-address selection. These parametric faults are described here briefly.

A. Bit-Line Voltage Imbalance

A typical memory array organization utilizes the differential amplifiers for sensing the signal partitioning of each array into two identical subarrays (called left and right in this paper) as shown in Fig. 7. Each bit line in the array is split into two halves and they are sensed by a differential pair of sense amplifiers. One of the cells in each half acts as a reference cell and its voltage is compared with the selected cell on the same word line, but on the other half. Thus in Fig. 7 when a cell in the right half on the bit line B_i^R is selected for reading, the reference cell on left-half bit line B_i^L is utilized for comparison. The bit-line voltage in B_i^L is clamped to a reference voltage (which is close to precharge voltage V_p) and is compared by the sense amplifier with the voltage of the bit-line voltage B_i^R, which will be near to the supply voltage if the selected cell on B_i^R contains a ONE, or near to the ground potential if the selected cell contains a ZERO. Thus if the difference of the voltages between the two bit lines is larger than a threshold value, the sense amplifier can correctly distinguish the state of the selected cell during a READ operation. When most of the cells in the left half of the memory subarray contain one type data (say, ONE) and most of the cells in the right half contain the opposite type data (ZERO), during a READ cycle the precharge voltage on the two halves of the bit lines will be different. This is illustrated in Fig. 7, where all the cells connected to the bit line B_i^L contain ONE and all the cells connected to the bit line B_i^R, except one which is connected to the word line W_j, contain ZERO. If the cell containing ONE in the right half of the memory is read, at first bit lines B_i^L and B_i^R

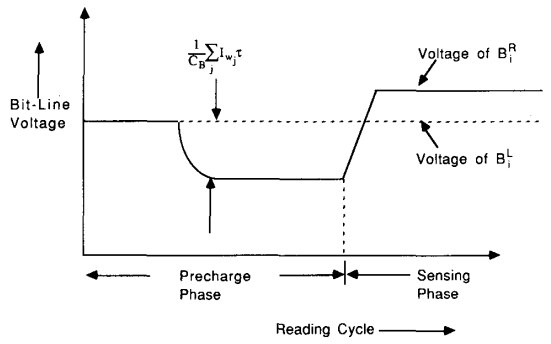


Fig. 8. Degradation of precharge voltage level due to leakage currents.

will be precharged to a voltage V_p. But due to the weak-inversion currents in the right-half cells, the precharge level will be degraded to

$$v_p = V_p - (1/C_B) \sum_{j=0}^{.5w-1} I_{W_j}\tau$$

where C_B is the capacitance of the bit line, I_{W_j} is the weak-inversion current in C_{ij}, and τ is the time interval for precharge during READ operation. This is illustrated in Fig. 8. If the weak-inversion currents are sufficiently large, the degradation in precharge voltage will be sufficiently high, i.e., $v_p \ll V_p$. Consequently, when the word line W_j is selected to read the content of the cell C_{ij}, which contains ONE, the bit-line voltage of B_i^R will be near to V_p and the contents of C_{ij} will be read as ZERO by the sense amplifier (because the ratio of bit-line capacitance and cell capacitance is usually greater than 15). This is illustrated in Fig. 7 where the difference in bit-line voltages is very small and thereby the sense amplifier incorrectly reads the contents of C_{ij}.

B. Bit-Line to Word-Line Crosstalk

From Fig. 9 it can be seen that there is an overlap between the bit line and the word line, since they are orthogonal to each other. This overlapping forms a coupling between the bit lines and the word lines so that when the bit-line voltages change due to precharging and restoring operations during a READ cycle, the unselected word lines may be inadvertently turned on. This coupling is maximum if all the cells in the selected word line contain ONE and if some of the cells of the coupled unselected word line contain ZERO, which will be degraded due to the weak-inversion current of the access transistors. In the precharging phase, at first the bit-line voltage increases from ZERO to V_p which is coupled as a noise voltage V_{n_j} to a word line W_j, where $0 \leqslant j \neq i \leqslant w-1$. If all the cells in the word line W_i contain ONE, then if W_i is selected its voltage will increase, coupling a noise voltage V_{v_j} to W_j. The effect of these superimposed noise voltages may generate a sufficient weak-inversion current such that a stored ZERO in a cell on the unselected word line W_j may be degraded [7].

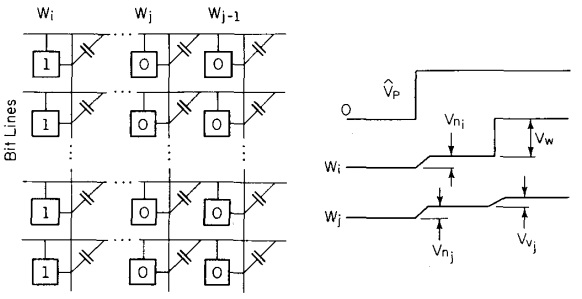


Fig. 9. Bit-line to word-line crosstalk.

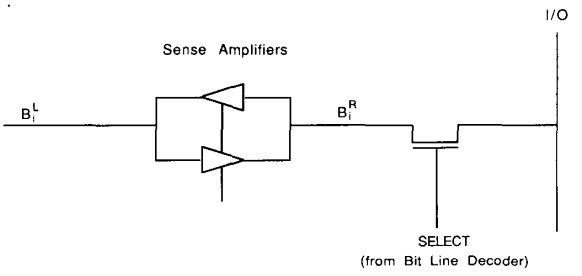


Fig. 10. Single-ended WRITE.

C. Single-Ended WRITE

In a DRAM employing single-ended WRITE technique, a single I/O line is used to write into the bit lines. In Fig. 10 it can be seen that writing on the right bit line B_i^R is controlled by the I/O line, while writing on the left bit line B_i^L is controlled by the sense amplifier. The ZERO level on B_i^R is determined by the input driver, but the ZERO level of B_i^L is determined by the sense amplifiers. Thus the level of ZERO in the two halves may be different, and this asymmetry may result in pattern-sensitive faults.

D. Multiple Selection

In the decoder circuit in Fig. 3, the precharge clock ϕ_p and the decoder enable clock ϕ_{EN} should be nonoverlapping in the sense that always ϕ_p and $\phi_{EN} = 0$. If due to incorrect timing they overlap, multiple selection may occur.

E. Transmission-Line Effect

In two-layer interconnect technology, either the bit lines or the word lines are made of metal and the other polysilicon or diffusion. Usually, the poly and diffusion lines have quadratic signal propagation delay (i.e., the signal propagation delay in an interconnect of length l is proportional to l^2, because delay is due to the product of its resistance and capacitance). Because of the high resistivity in these interconnects the cells at the periphery of the chips, away from the sense amplifiers, are delivered a weak signal and thereby may fail. By inserting the repeaters at

suitable intervals, the signal strength and delay may be improved, but complexity is added to the layout. In order to check that all the cells in the array satisfy the limits of the stipulated memory cycle time, the transmission-line effect should be tested.

IV. Testing Strategy and Algorithms

In order to test all of the above faults, it is necessary to identify the circumstances in which each of them is likely to be maximum. It can be easily noted that the field-inversion current I_F which occurs between two adjoining storage cells is maximum if the four adjacent cells of a base cell contain opposite data to that of the base cell, i.e., a checkerboard-type pattern can test the effect of a field-inversion current. Similarly, the effects of a dark current and gate short can be tested by the checkerboard pattern, because the presence of these leakage currents manifests in the form of cell stuck-at ZERO or ONE. Algorithm 1 is a parallel version of the checkerboard test, which tests the above three leakage currents in addition to the parametric faults due to the single-ended WRITE and the transmission-line effects. Since in Algorithm 1 each cell in a word line is tested for both ZERO and ONE data, the stuck-at ZERO and stuck-at ONE faults are also tested. Moreover, if a bit line or a word line is faulty (namely, broken, stuck to ground, or cannot be precharged), Algorithm 1 will detect the fault. It may be noted that a faulty bit line will be incorrectly compared by the parallel 0/1 detector to set ERROR = 1. A faulty word line where all the bits are identical may not be detected by the parallel 0/1 detector. But it can be easily detected by monitoring the output of the data-out buffer, if the entire memory is organized into a single array, otherwise, by mutually comparing the data-out values of all subarrays. It may be added that if the entire word line is faulty, then the expected value at the data-out buffer will be different from the obtained one if and only if the word line is tested for both ZERO and ONE data value. If a word line is completely fault free, then the expected value of data will always be the same as the obtained one. If only one or two bits are faulty in a word line, then the expected value may be different from the obtained one, but it will certainly be detected by the 0/1 detector.

Algorithm 1: Parametric Checkerboard Test

1) Use complementing address sequence from word line W_0 until all word lines are scanned; write in two steps a pattern of (01)* if the word line is even and two steps a pattern of (10)* if the word line is odd.
2) Freeze the clock for the entire refresh interval τ_R for testing *static refresh*.
3) Use a complementing address sequence from word line W_0 until all word lines are scanned; compare in parallel all even and odd bit lines to check ERROR = 0.
4) Read continuously any arbitrary word line for the entire refresh interval τ_R to check ERROR = 0. This test checks the effect of temperature rise and tests the *dynamic refresh*.
5) Use the complementing address sequence from word line W_0 until all word lines are scanned; compare in parallel all even and odd bit lines to check ERROR = 0.
6) Read continuously another distinct word line for the entire refresh interval τ_R to check ERROR = 0. This test checks the effect of temperature rise and tests the dynamic refresh.
7) Use the complementing address sequence from word line W_0 until all word lines are scanned; compare in parallel all even and odd bit lines to check ERROR = 0.
8) Repeat steps 1–7, with opposite data.

The effect of a weak-inversion current is maximum when all the cells in a bit line, except one, contain ZERO. If the cell which contains ONE is addressed for a READ operation, the weak-inversion currents in other cells will tend to degrade the precharge level of the bit line and thereby the cell containing ONE will be sensed as ZERO by the sense amplifier. Since the bit-line capacitance is typically 10–20 times the capacitance of an individual cell, the stored ONE may not be sufficient to replenish the degraded precharge level. The testing strategy needs to test each memory cell so that when it is ONE all its bit-line neighbors will contain ZERO.

In order to test the bit-line voltage imbalance, it is necessary to write ZERO (ONE) on the cells at the bit line on the left half of the subarray and write ONE (ZERO) on the cells at the bit line on the right half of the subarray. Thus the test to detect the weak-inversion current can be utilized to test the bit-line voltage imbalance by testing the left and right subarrays with opposite background data. It may be noted that the test also detects faults due to single-ended WRITE. Algorithm 2 tests all the above faults.

Algorithm 2: Parallel Parametric Walking Test

1) Initialize the entire memory writing ZERO in all locations.
2) Select two arbitrary word lines W_i and W_j and read them alternately for one refresh interval. Check if the ERROR = 0 during the entire refresh interval.
3) For all word lines starting from W_0, compare in parallel all even and odd bit lines to check ERROR = 0.
4) Initialize the entire memory, writing ONE in all locations.
5) Select two arbitrary word lines W_p and W_q and read them alternately for one refresh interval. Check if the ERROR = 0 during the entire refresh interval.
6) For all word lines starting from W_0, compare in parallel all even and odd bit lines to check ERROR = 0.
7) Initialize the memory such that the left subarray contains ZERO in all locations and right subarray contains ONE in all locations.
8) For all word lines starting from W_0, do the following: i) write a pattern of (01)* in the selected word line; ii) parallel compare and check if ERROR = 0; iii) initialize all the cells in the selected word line to ZERO if it is on left half, otherwise to ONE.
9) For all word lines starting from W_0, do the following: i) write a pattern of (10)* in the selected word line; ii) parallel compare and check if ERROR = 0; iii) initialize all the cells in the selected word line to ZERO if it is on left half, otherwise to ONE.
10) Repeat steps 7–9 with complementary bit patterns.

Algorithm 3, which runs a marching pattern of ONE at the background of ZERO and a marching pattern of ZERO at the background of ONE in each word line, will detect the multiple-access faults in the word-line decoder by comparing the READ data with the expected data. Since the algorithm employs parallel writing by accessing all the even bit lines and odd bit lines in a single memory cycle,

TABLE II
ALGORITHMS AND THEIR COVERAGE OF PARAMETRIC FAULTS

Fault Type	Algorithm 1	Algorithm 2	Algorithm 3
Weak-Inversion Current	No	Yes	Yes
Field-Inversion Current	Yes	No	No
Dark Current	Yes	No	No
Gate Short	Yes	Yes	Yes
Multiple Selection	No	No	Yes
Single-Ended Write	Yes	Yes	Yes
Bit-Line Voltage Imbalance	No	Yes	No
Bit Line to Word Line Crostalk	No	Yes	No
Transmission-Line Effect	Yes	No	No

multiple access in the bit-line decoder will not be tested by Algorithm 2. A separate algorithm is needed to test the bit-line decoders and is given in Algorithm 3.

Algorithm 3: Bit-Line and Word-Line Decoder Test

/* Bit-Line Decoder Multiple Access Test */
1) Write in parallel ZERO in all cells on the arbitrarily selected word line W_j.
2) Read and compare in parallel all the cells on W_j.
3) Starting from the cell at the crosspoint of B_0 and W_j, for each cell on W_j, at first write ONE and read the cell (one cell at a time in ascending order of the bit line).
4) Starting from the cell at the crosspoint of $B_{\sqrt{n}-1}$ and W_j, for each cell on W_j, at first write ONE and read the cell (one cell at a time in descending order of the bit line).

/* Word-Line Decoder Multiple Access Test */
5) Write in parallel ZERO in all cells on the bit line B_i.
6) Read and compare in parallel all the cells on B_i.
7) Starting from the cell at the crosspoint of W_0 and B_i, for each cell on B_i, at first write ONE and read the cell (one cell at a time in ascending order of the word line).
8) Starting from the cell at the crosspoint of $W_{\sqrt{n}-1}$ and B_i, for each cell on B_i, at first write ONE and read the cell (one cell at a time in descending order of the word line).

It can be seen that Algorithm 1 takes altogether $10w\tau_A + 6\tau_R$ time to complete all the steps, where τ_A is the average memory cycle time and τ_R is the refresh interval. Algorithm 2 takes $20w\tau_A + 2\tau_R$ time to test the entire DRAM. Finally, Algorithm 3 takes $(4w+2)\tau_A$ time to test the multiple-access faults in the decoder logic. Hence, altogether $(34w+2)\tau_A + 8\tau_R$ time is needed to test all the parametric faults in the DRAM. Table II depicts the different types of parametric faults and how those are covered by these algorithms.

The above algorithms are tested at the rated maximum and minimum power-supply voltages. But, in addition to the worst-case measurements, it is necessary to test the memory when the supply voltage changes rapidly due to an impressed noise voltage. Noise spikes have high slew rates and they may occur during a READ or WRITE memory cycle causing the operation to fail. If the effect of the noise spike is to lower the supply voltage, the capacitive bias of the dynamic logic may be higher than the supply bias, and this may result in a failure of a READ or WRITE operation. Similarly, if the noise spike increases the supply voltage to a high value, the capacitive bias voltage may be sufficiently lower than the supply bias resulting in a faulty READ or WRITE operation. In order to test the effect of this power-supply voltage transition, the entire memory should be tested for both the cases when the supply voltage rapidly increases and again when it quickly decreases. The typical slew rate is about a few microseconds. Algorithm 4 tests the memory for the above faults.

Algorithm 4: Power-Supply Voltage Transition Test

1) Write in parallel ZERO in the entire memory at maximum supply voltage.
2) For all word lines starting from W_0, do the following: i) write a pattern of (01)* in the selected word line at maximum supply voltage; ii) rapidly reduce the supply voltage to minimum; iii) parallel compare and check if ERROR = 0.
3) For all word lines starting from $W_{\sqrt{n}-1}$, do the following: i) write a pattern of (10)* in the selected word line at maximum supply voltage; ii) rapidly reduce the supply voltage to minimum; iii) parallel compare and check if ERROR = 0.
4) Write in parallel ZERO in the entire memory at minimum supply voltage.
5) For all word lines starting from W_0, do the following: i) write a pattern of (01)* in the selected word line at minimum supply voltage; ii) rapidly increase the supply voltage to maximum; iii) parallel compare and check if ERROR = 0.
6) For all word lines starting from $W_{\sqrt{n}-1}$, do the following: i) write a pattern of (10)* in the selected word line at minimum supply voltage; ii) rapidly increase the supply voltage to maximum; iii) parallel compare and check if ERROR = 0.

V. CONCLUSIONS

The main objective of this paper is to propose a modified DRAM architecture which enhances the speed for testing the occurrence of parametric faults in the 3D DRAM employing trench-type capacitor. A novel testable architecture is proposed to test multiple cells in a word line simultaneously resulting in a speedup by a factor of $O(\sqrt{n/p})$. The testable design has been proposed to fit within the interceller pitch width of 3λ in the 3D DRAM. It employs only an additional $2\sqrt{pn} + p\log_2 n - p\log_2 p + 12p$ transistors and has very low overhead. Above all, the technique is not tailored to test only one specific test procedure and it can speed up the conventional test algorithms for functional faults [8]–[10].

A number of parametric faults, that manifest due to a variation of processing parameters and also due to a critical circuit design, are enumerated in this paper. Leakage currents in a 3D DRAM have been identified, and test procedures have been designed to test in parallel the faults due to these leakage currents. In addition to the faults due to leakage currents, circuit-related design weaknesses (like the multiple selection, single-ended write, etc.) are also tested by the algorithms described here. The effects of rapid supply-voltage transitions, which cause a faulty READ or WRITE due to disparity in capacitive voltages in the dynamic logic and the supply bias, are comprehensively tested in this paper. Unlike the earlier approaches [11], [12] which employed diagonal tests having $O(n^{3/2})$ complexity, the algorithms described here have $O(\sqrt{n/p})$ complexity, and thereby a dramatic improvement by a factor proportional to the size of the DRAM can be achieved. For a 16-Mbit DRAM organized into 16

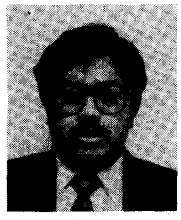
quadrants, the diagonal test algorithm will need more than a few hours as opposed to the proposed algorithms which will need only a few milliseconds. Moreover, the fault coverage by these algorithms is far superior to that of the diagonal test procedure. It may also be noted that these algorithms can be easily generated for built-in self-test (BIST) applications.

ACKNOWLEDGMENT

The author wishes to thank Prof. J. H. Patel and Prof. W. K. Fuchs of the Coordinated Science Laboratory, University of Illinois, Urbana, for their helpful suggestions on the design-for-testability strategy in Section II.

REFERENCES

[1] L. L. Lewyn and J. D. Meindl, "Physical limits of VLSI dRAM's," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 231–241, Feb. 1985.
[2] H. McAdams *et al.*, "A 1-Mbit CMOS dynamic RAM with design-for test functions," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 635–641, Oct. 1986.
[3] A. H. Shah *et al.*, "A 4-Mbit DRAM with trench-transistor cell," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 618–627, Oct. 1986.
[4] Y. You and J. P. Hayes, "A self-testing dynamic RAM chip," *IEEE J. Solid-State Circuits*, vol. SC-20, pp. 428–435, Feb. 1985.
[5] M. S. Abadir and H. K. Reghbati, "Functional testing of semiconductor random-access memories," *ACM Computing Surveys*, vol. 15, no. 3, pp. 175–198, Sept. 1983.
[6] N. C. Lu *et al.*, "A substrate-plate trench-capacitor (SPT) memory cell for dynamic RAM's," *IEEE J. Solid-State Circuits*, vol. SC-21, pp. 627–634, Oct. 1986.
[7] H. Masuda *et al.*, "A 5-V-only 64K dynamic RAM based on high S/N design," *IEEE J. Solid-State Circuits*, vol. SC-15, pp. 846–853, Oct. 1980.
[8] P. Mazumder, J. H. Patel, and W. K. Fuchs, "Design and algorithms for parallel testing of random-access and content-addressable memories," in *Proc. Design Automation Conf.*, July 1987, pp. 688–694.
[9] P. Mazumder and J. H. Patel, "An efficient built-in self-testing of random-access memory," in *Proc. Int. Test Conf.*, Sept. 1987, pp. 1072–1077.
[10] J. Inoue *et al.*, "Parallel testing technology for VLSI memories," in *Proc. Int. Test Conf.*, Sept. 1987, pp. 1066–1071.
[11] M. A. Breuer and A. D. Friedman, *Diagnosis and Reliable Design of Digital Systems*. Los Angeles: Woodland Hills, 1976.
[12] T. C. Lo and M. R. Guidry, "An integrated test concept for switched-capacitor dynamic MOS RAM's," *IEEE J. Solid-State Circuits*, vol. SC-12, pp. 693–703, Dec. 1977.



Pinaki Mazumder received the B.Sc. degree in physics from Gauhati University, India, the B.S.E.E. degree from the Indian Institute of Science, Bangalore, the M.Sc. degree in computer science from the University of Alberta, Canada, and the Ph.D. degree in electrical and computer engineering from the University of Illinois.

He worked two years as a Research Assistant at the Coordinated Science Laboratory, University of Illinois, and over six years at Bharat Electronics Ltd. (a collaborator of RCA-GE) in the area of integrated circuit design and applications. During the summers of 1985 and 1986, he worked as a Member of the Technical Staff at AT&T Bell Laboratories, Indian Hill, Naperville, IL, in the area of hardware synthesis from system-level behavioral description. Presently he is working as an Assistant Professor at the Department of Electrical Engineering and Computer Science of the University of Michigan, Ann Arbor. His research interests include VLSI testing, computer-aided design, parallel architecture, and image processing.

Dr. Mazumder is a member of Phi Kappa Phi.