Fault Models and Tests for a 2-Bit-per-Cell MLDRAM

MICHAEL REDEKER
University of Hanover
BRUCE F. COCKBURN
DUNCAN G. ELLIOTT
University of Alberta

Multilevel DRAM technology may become a cost-effective way to increase semiconductor memory storage density. The authors develop an MLDRAM fault model using both manual analysis and analog simulation. They also propose several alternative testing strategies and possible design-for-testability enhancements.

OVER THE LAST 25 YEARS, the storage capacity of monolithic dynamic random-access memories (DRAMs) has increased by six orders of magnitude through a combination of design innovations and improvements of existing technology.[1,2] Design innovations include the folded bit-line cell array arrangement and compact, high-capacitance, three-dimensional cell designs such as the stacked capacitor and trench capacitor cells. Technology improvements include increases in die area and, most importantly, reductions in minimum feature size.[1,2] Further increases in DRAM storage density through the use of still smaller feature sizes face serious challenges from increasingly high fabrication facility costs. Monolithic digital systems increasingly employ embedded DRAMs; however, some of the density advantage of DRAMs over static RAMs is lost because conventional logic processes support only planar cell capacitor designs.

An often-proposed design innovation for achieving higher storage density without further reductions in feature size or complex 3D cell structures is to hold more than one bit per storage cell. The potential for storing more than two resolvable analog voltages on the capacitance of a storage cell facilitates this idea in DRAMs.

In this article, we describe a fault model for a 2-bit-per-cell MLDRAM proposed by Gillingham.[3] We derived the fault model using manual analysis and confirmed it with analog SPICE simulation. We modified accurate circuit models obtained from MOSAID Technologies to produce the effects of physical defects similar to those reported for 1-bit-per-cell DRAMs with similar cell array layouts. We also propose an efficient test for the fault model and possible MLDRAM design-for-testability enhancements.

MLDRAM design problems

Three fundamental problems in multilevel DRAM (MLDRAM) design are 1) creating the different analog voltages to be stored, 2) creating the various reference voltages necessary to interpret previously stored signals, and 3) managing the effects of the reduced noise margins resulting from more closely spaced signal levels.

Researchers have proposed many MLDRAM designs.[2-6] The scheme described by Gillingham has several attractive characteristics. For example, it solves problems 1 and 2 with charge-sharing techniques that locally generate analog reference signals for both the writing and sensing operations. Cell signal sensing involves comparing signals on matched sub-bit lines. Also, the same cell that originally held the stored signal is used to generate the reference signal required to sense the least significant bit. These features

0740-7475/99/$10.00 © 1999 IEEE

provide robustness against process variation effects. A further advantage is that the MLDRAM reuses many elements of a conventional 1-bit-per-cell DRAM, including most of the cell array layout, the basic sense amplifier design, and the precharge to mid-voltage ($V_{pre} = 1/2\ V_{DD}$).

Random cell accesses, however, are slower in Gillingham's MLDRAM than in a conventional DRAM because of the multiple steps required in the writing and sensing operations. For example, an MLDRAM's random cell read time t_{RAC} is roughly twice that of a DRAM implemented in a similar semiconductor technology. However, after the random-access time penalty, accesses to additional data on the same internal cell page can be as fast in an MLDRAM as in a conventional DRAM. Thus, the maximum data bandwidths for MLDRAMs and DRAMs should be similar. The reduced noise margins of all MLDRAMs potentially increase their vulnerability to process variations, soft errors, and excessive cell leakage current. Possible solutions include higher-capacitance storage cells,[2] careful attention to cell layout to eliminate potential noise sources, and an increased refresh rate.

MLDRAM operation

The operation of Gillingham's MLDRAM is detailed elsewhere,[3] so we briefly summarize it here. Figure 1 illustrates the encoding of the four 2-bit Boolean pairs 00, 01, 10, and 11 as equally spaced voltages in the range ground to V_{DD}. We refer to the leftmost bit as the most significant bit (MSB) and the rightmost bit as the least significant bit (LSB). Figure 1 shows the positive signal encoding; to obtain the negative signal encoding, we complement the MSB and LSB values. Note that the noise margins of this four-level encoding are only one third those of a two-level encoding using voltages 0 and V_{DD}.

Sensing the voltage signal stored in a cell requires connecting the cell capacitance C_c to a sense amplifier via a bit line. A bit-line capacitance C_b is typically 8 to 13 times larger than a single cell's C_c.[1] Thus, charge sharing reduces the cell signal's strength by a factor of $C_c/(C_c + C_b)$. Recovering the 2-bit logical value from the attenuated cell signal is a two-step destructive read process: First, a sense amplifier compares the cell signal with the mid-reference level $1/2\ V_{DD}$ to determine the MSB. Then, depending on the outcome of the first comparison, the cell signal is compared with suitably attenuated copies of either $5/6\ V_{DD}$ or $1/6\ V_{DD}$ to determine the LSB

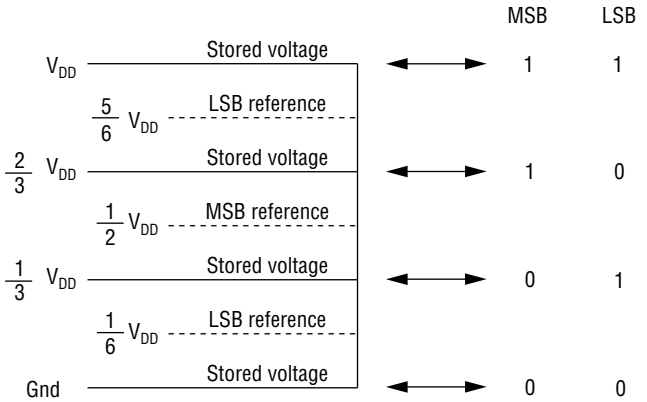

Figure 1. *Voltage encoding for a four-level MLDRAM.*[3]

(see Figure 1). After a cell's contents have been sensed, the cell's analog voltage signal must be restored to the nominal level for the data just read, or written with the nominal voltage level for newly supplied data.

Figure 2 shows the circuit diagram for one bit-line pair containing a column of storage cells (in the figure, the cell column runs horizontally, following common practice). One can view the circuit as two sub-bit-line pairs (left and right) that interact via a switch matrix controlled by signals *C*, *C**, *X*, and *X**. The left (right) sub-bit-line pair comprises true *BL* (*BR*) and complement *BL** (*BR**) sub-bit lines connected to the noninverting and inverting terminals, respectively, of the left (right) sense amplifier. All sub-bit lines have the same capacitance C_b. Typically, many bit-line pairs are grouped to form left and right cell arrays that share peripheral circuitry. Each cell in an array consists of a capacitor connected from a common biased plate node to the source node of an access transistor; each access transistor's drain node connects to a sub-bit line.

As shown in Figure 2, we assume that the MLDRAM uses PMOS access transistors. (Although using PMOS rather than NMOS access transistors changes the polarity of the wordline signals, it has no other effect on our analysis.) When a cell in the left (right) cell array is read, the MSB is recovered at a left (right) sense amplifier, and the LSB is recovered at the corresponding right (left) sense amplifier. With respect to an addressed cell in either the left or right arrays, *MSB side* refers to the array containing the cell, whereas *LSB side* refers

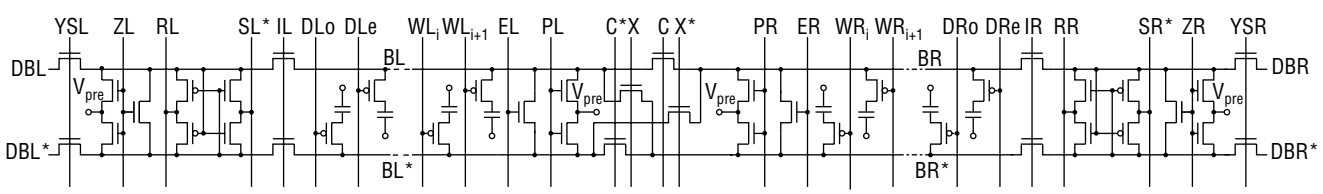

Figure 2. *MLDRAM circuit schematic.*[3]

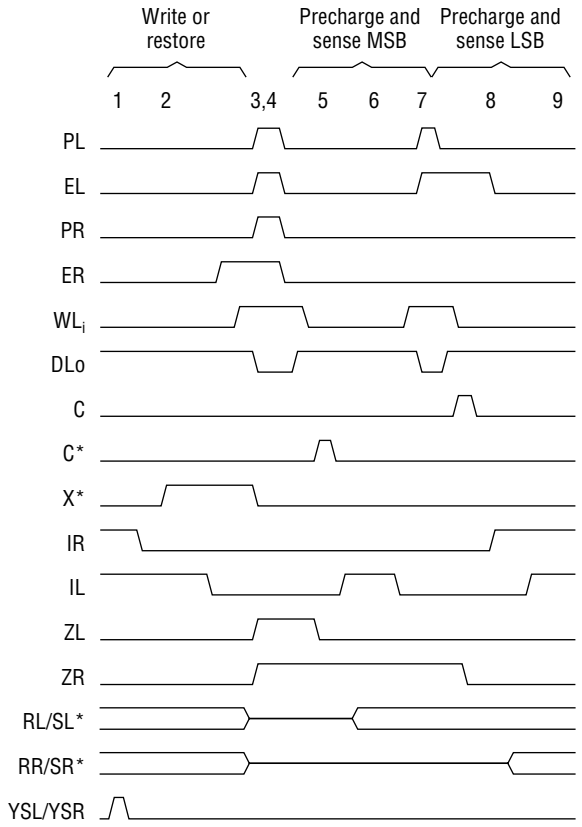

Figure 3. *Control waveform timing diagram.*

to the array on the other side of the switch matrix.

Word lines connected to the gate nodes of cell access transistors control access to a row of cells (in Figure 2, the rows run vertically). We assume the word lines are numbered $WL_1, \ldots, WL_{i+1}$ and $WR_1, \ldots, WR_{i+1}$ for the left and right cell arrays. Transitions on dummy word lines cancel out bit-line noise induced, via the gate-to-source capacitance of the cell access transistors, by opposite transitions on the regular word lines. Dummy word line *DLo* (*DRo*) is manipulated during accesses to cells connected to odd-numbered word lines in the left (right) array. Similarly, dummy word line *DLe* (*DRe*) is manipulated during accesses to cells connected to even-numbered word lines. Using dummy word lines this way is an important precaution in view of the MLDRAM's reduced noise margins.

Figure 2 also shows additional control signals required to produce the correct sequence of events during cell accesses. Signals *IL* and *IR* control pass transistors that permit the cell array to be isolated from the sense amplifiers. Signals *EL* and *ER* control the equalize circuits located on either side of the switch matrix. Similarly, *PL* and *PR* control separate left and right precharge circuits. A peripheral regulation circuit maintains the precharge voltage V_{pre} at 1/2 V_{DD}. *ZL* and *ZR* control similar equalize and precharge circuits for the sense amplifiers. Finally, column select signals *YSL* and *YSR* control the connection between the left and right sub-bit-line pairs and the left (*DBL* and *DBL**) and right (*DBR* and *DBR**) differential data buses.

Figure 3 shows the control waveforms that cause a write (or restore) operation to the cell at WL_i and *BL** followed by a read operation. This cell uses the negative signal encoding because it is accessed via the complement sub-bit line *BL**. The word line signals are active low because of the PMOS access transistors. This waveform sequence proved more convenient for studying the external effects of faults than the sense-and-restore waveform sequence given by Gillingham.[3] We summarize the corresponding events for the bit-line pair here.

Write a new 2-bit value into the addressed cell:

1. The column address decoder (not shown in Figure 2) asserts column select signals *YSL* and *YSR*. The new MSB and LSB values are driven onto the left and right sub-bit lines from the left and right data buses.
2. Control signal *IR* goes low and disconnects the right sense amplifier, leaving *BR** (one sub-bit-line capacitance) carrying the LSB. A control circuit (not shown in Figure 2) asserts *X** to connect *BL** and *BR*, thus allowing both sub-bit lines (two sub-bit-line capacitances) to carry the MSB. The left sense amplifier is disconnected, and *BR**, *BL**, and *BR* share the stored charge. The resulting signal voltage, which encodes both the LSB and MSB, is (2/3 MSB + 1/3 LSB)V_{DD}. Deasserting WL_i traps this signal in the cell.

Isolate the cell and enter storage mode:

3. The control circuit causes all dummy word lines to be held asserted and all word lines to be deasserted. Then the left and right sub-bit lines are isolated from each other, and the left and right sense amplifiers are disconnected.
4. All sub-bit lines are precharged to $V_{pre} = 1/2\ V_{DD}$. All sense amplifiers are powered down, and their outputs are precharged to mid-voltage.

Read back the signal from the addressed cell:

5. First WL_i, then *C**, are asserted to transfer the cell signal onto both sub-bit lines *BL** and *BR**. Charge sharing attenuates the cell signal by $C_c/(C_c + 2C_b) \cong C_c/2C_b$.
6. The left sense amplifier compares the weak cell signal on *BL** with the mid-voltage MSB reference on *BL*. The left

sense amplifier then amplifies this signal and applies a full-swing differential MSB signal onto *BL* and *BL**. Control signal *IL* goes low, leaving the recovered MSB latched in the left sense amplifier.

7. The LSB reference on BR is created by first isolating the full-swing MSB signal in the addressed cell. Then the left sub-bit lines are equalized and precharged. The MSB cell signal is distributed onto sub-bit lines *BL**, *BL*, and *BR* by asserting *EL* and *C*. The resulting signal on *BR* is then isolated. Gillingham[3] shows that this signal voltage approximately equals 1/6 V_{DD} (if $\overline{MSB} = 0$) or 5/6 V_{DD} (if $\overline{MSB} = 1$) attenuated by the factor $C_c/2C_b$.
8. The right sense amplifier compares the weak cell signal on *BR** with the LSB reference on *BR*. The right sense amplifier then amplifies and applies a full-swing differential LSB signal onto *BR* and *BR**.
9. At this point, we have an opportunity for page mode access via the data buses. The left and right sub-bit-line pairs are isolated from each other while they hold the MSB and LSB values latched by the left and right sense amplifiers.

An analogous sequence of steps allows accesses to cells on sub-bit lines *BL*, *BR*, and *BR**. The corresponding left and right circuit elements switch roles during accesses into the right cell array. The positive signal encoding (shown in Figure 1) is in effect during accesses of cells connected to the true sub-bit lines *BL* and *BR*.

Fault model development

To develop tests for a new DRAM, one usually starts with a superset of the tests found effective for earlier parts.[8] But this strategy is unlikely to be as effective for an MLDRAM, whose circuit schematic and operation differ from those of a conventional 1-bit-per-cell DRAM. We therefore decided to develop a new model of the expected faulty behaviors. Then we could design tests that efficiently target faults in this model. Inductive fault analysis is a defect simulation method that infers a fault model from the circuit layout and knowledge of the process defect distributions.[7] We chose to avoid this method's computational expense by using manual analysis to predict faulty behaviors directly from the circuit schematic. We confirmed the resulting behaviors, whenever practical, by SPICE simulations. This is a reasonable MLDRAM strategy since many elements of the cell array layout are identical or similar to those in a conventional DRAM. The main differences in behavior should follow from the differences in the control signals and the schematic.

Several assumptions underlie our MLDRAM fault model. First, we assume that the cell array's physical layout is very similar to that of a conventional folded bit-line DRAM. Thus, we assume that sub-bit lines are laid out in parallel on the same metal layer,[9] and that word lines are laid out orthogonally to the bit lines on one or more separate layers. (Word lines are usually implemented in gate polysilicon with parallel metal shunts or straps.) Second, we assume that physical defects in an MLDRAM will be similar to those known to occur in DRAMs.[9-11] Thus, we assume that stuck nodes, missing contacts, interconnect breaks, and short circuits between adjacent or crossing interconnections will be relatively frequent. For example, word-line-to-word-line and bit-line-to-bit-line shorts should be relatively common. To simplify our analysis, we consider both stuck-at-0 and stuck-at-1 faults affecting the control signals; this allows us to abstract away details of the circuits that produce the control signals. Finally, we assume that defects affecting peripheral circuitry—row and column decoders, control logic, voltage generation and regulation circuits, and so on—will cause catastrophic failures readily detected by any reasonably thorough test of the cell arrays.[8] Therefore, we do not directly model defects in the peripheral circuits.

We call a physical defect clean if it produces faulty behavior that can propagate deterministically to a circuit output as an observable Boolean error.[12] Unclean defects often arise when contention exists at a circuit node joining two conducting paths to different power supply voltages. As with DRAMs, many physical defects in MLDRAMs are unclean. Typical situations producing unclean faults include the following:

- *Two sub-bit lines connected to the same sense amplifier are shorted together.* Depending on the details of the sense amplifier (such as transistor sizing), the outputs settle at the same, possibly undefined, mid-voltage value.
- *Sense amplifier enabled with no differential signal on the sub-bit-line pair.* This is a metastable situation that resolves itself randomly to one of two stable Boolean states. In a chip, each sense amplifier likely has a finite input offset that causes the situation to resolve itself preferentially to one state.
- *Two sub-bit lines connected to different sense amplifiers are shorted together.* This also is a metastable situation that tends to resolve itself unpredictably to one of two possible states.

In an MLDRAM, the recovered MSB's state determines which reference voltage is used to sense the LSB. Thus, a defect that randomizes the MSB also randomizes the recovered LSB. The converse is not true; a random LSB will not affect the recovered MSB's state.

Results of manual analysis. For each physical defect, we predicted how the circuit would appear to behave at the external Boolean MLDRAM data interface. We identified

Table 1. *Fault model mapping.*

Physical defect	Fault class
Short between word line and cell capacitor	1
Short between sub-bit line and cell capacitor	1, 7
Short between two cell capacitors	3, 5
No connection between sub-bit line and cell	1
Cell access transistor stuck open	1
Cell access transistor stuck on	1, 7
Excessive cell leakage current	1
Interrupted word line (WL_i, WR_i)	2, 7
Interrupted sub-bit line (*BL*, *BL**, *BR*, *BR**)	2, 6, 7
Short between adjacent word lines	3, 8
Short between adjacent sub-bit lines	2, 5, 8
Short between word line and sub-bit line	2, 8
Stuck word line (WL_i, WR_i)	2, 7, 8
Stuck dummy word line (*DLo*, *DLe*, *DRo*, *DRe*)	7, 8
Stuck bit-line precharge control (*PL*, *PR*)	2, 6, 7, 8
Stuck bit-line equalize control (*EL*, *ER*)	2, 8
Stuck sense amplifier isolation control (*IL*, *IR*)	2, 8
Stuck sense amplifier precharge control (*ZL*, *ZR*)	2, 6
Stuck switch matrix signal (*C*, *C**, *X*, or *X**)	2, 4, 8

eight classes of distinct erroneous behavior. Table 1 shows the resulting mapping from defects to fault classes.

Now, we informally describe and justify the eight fault class definitions:[13]

1. *Isolated stuck bits: cells contain fixed values.* A short between a word line and a cell capacitor's storage node causes the word line signal to overwhelm the cell signal. This causes the cell's MSB to appear stuck and the LSB to appear fixed at some random value (the word line signal drives both the LSB reference and the cell signal). Excessive leakage current from a cell capacitor causes both MSB and LSB to settle at fixed implementation-specific values. Several defects null out the cell signal, leaving the precharge voltage to be sensed; the observable result is a stuck MSB and LSB. These defects include a short between a sub-bit line and a cell capacitor, no connection between a sub-bit line and a cell, a cell access transistor stuck open, and a cell access transistor stuck on.
2. *Multiple stuck bits: clusters or alignments of cells contain fixed values.* A stuck (or an interrupted) word line causes the precharge voltage to be sensed for all cells connected to that word line (to the floating word-line segment). The observable result is that the MSB and LSB of those cells are independently random (they are likely to be fixed states in real devices). A short from a word line to a sub-bit line produces stuck MSBs and LSBs along the shorted sub-bit-line pair, since the word-line signal is sensed instead of the cell signal during MSB sensing and LSB reference-level generation. (As in conventional DRAMs,[9] a short between a word line and a sub-bit line can also cause complex faulty behaviors if the sense amplifier's write signal communicates to the shorted word line and thus causes additional cells to be unexpectedly connected to sub-bit lines.) Similar arguments show that the following other defects also can cause multiple stuck bits affecting MSBs and/or LSBs:[14] 1) MSB-side or LSB-side sense amplifier isolation control stuck at off, 2) *C* (*C**) stuck at on, 3) sense amplifier isolation control stuck at on, 4) interrupted sub-bit line, 5) short between adjacent sub-bit lines with the same sense amplifier, 6) stuck-at-on bit-line precharge control or equalize control, 7) stuck-at-on sense amplifier precharge control, and 8) stuck-at-on switch matrix signal *X* or *X**.
3. *State coupling within a pair of adjacent cells on different word lines.* A short between adjacent word lines causes pairs of cells on the two word lines to be coupled as follows: Let cells *i* and *j* be two cells on the two shorted word lines that share the same sub-bit-line pair. Without loss of generality, assume that cell *i* was written more recently than cell *j*. If cells *i* and *j* connect to different sub-bit lines in a pair, the MSB of cell *i* is forced into the LSB of cell *i* and both the MSB and LSB of cell *j* (see Redeker[13] for details). If cells *i* and *j* connect to the same sub-bit line, the 2-bit (MSB, LSB) state of cell *i* is forced into cell *j*. If such a short interconnects the storage nodes of cells *i* and *j* on complementary sub-bit lines, writing (MSB, LSB) into cell *i* forces cell *j* into the state ($\overline{\text{MSB}}$, $\overline{\text{LSB}}$), and vice versa.
4. *State coupling from the MSB to the LSB of the same cell.* If switch control *C* (*C**) is stuck at off, the LSB of all cells on a true (complement) sub-bit line is forced to be the complement of their MSB. Likewise, the LSB of all cells on a complement (true) sub-bit line is forced to be identical to their MSB. If switch control *X* or *X** is stuck at off, the open switch prevents the LSB signal from being charge-shared with the MSB during writing. The result is that the LSB of all cells appears identical to the MSB.
5. *Pattern sensitivities within a pair of cells on the same word line.* A short between adjacent sub-bit lines connecting to different sense amplifiers can produce several kinds of pattern sensitivities. What kind depends on where the short is with respect to the addressed cells and on the logic encoding used for the two shorted sub-bit lines. During sensing, cell signals and reference levels interfere with each other. As a result, the contents of cell pairs appear to interact so that fewer than 16 different state combinations can be stored and read back from the two cells (see Redeker[13] for details). Similar pattern sensi-

tivities arise from shorts between the storage nodes of two cells accessed using the same word line.

6. *Pattern sensitivities involving the last MSB or LSB value written*. Suppose we have an interrupted sub-bit line with the open on the same (opposite) side of the switch matrix as the addressed cell. The resulting floating sub-bit line causes the last value written to any cell on the same sub-bit-line pair to determine the cell's MSB (LSB). Suppose a sense amplifier precharge control signal is stuck at off and an addressed cell is on the same (opposite) side of the switch matrix as the affected sense amplifier. The capacitance of the sense amplifier nodes holds the last MSB (LSB) data written to that sub-bit-line pair. A stuck-at-off bit-line precharge control signal prevents erasure of the cell signal last written to a cell on the same bit-line pair. The residual cell signal interferes with correct sensing of the next accessed cell. The next sensed MSB appears to be the last written MSB or its complement; the LSB is the same as the MSB or its complement, or is a random value.[13]
7. *Degraded noise margins*. Several defects can degrade the already narrow noise margins of an MLDRAM, possibly causing errors. A word line stuck at enabled or an interrupted word-line segment floating at enabled causes many cells on other word lines to have degraded noise margins due to unbalanced sub-bit-line capacitances. Similarly, a short between a sub-bit line and cell capacitor or a stuck-on cell access transistor degrades noise margins of other cells on the same sub-bit-line pair due to sub-bit-line capacitance mismatch. A dummy word line stuck at off or stuck at on degrades noise margins because of the resulting unbalanced word-line noise injection into the sub-bit lines. If the dummy word line is stuck at on, noise margins further degrade because sub-bit-line capacitances are unmatched, causing offsets in cell signals and LSB reference levels. An interrupted sub-bit line across the switch matrix from the accessed cell also degrades noise margins because of unmatched sub-bit-line capacitances. A stuck-at-off precharge control will not prevent sub-bit lines from being equalized at near mid-voltage, but the precharge voltage could drift from $V_{DD}/2$, causing reduced noise margins.
8. *Faults detectable by an I_{DDQ} measurement*. Several defects should be readily detectable as excessive quiescent power supply current I_{DDQ} (as well as through Boolean testing). When the sense amplifiers are powered up, a short between adjacent sub-bit lines with the same sense amplifier produces high I_{DDQ} as soon as the sense amplifier connects to the sub-bit-line pair. The same is true for stuck-at-on bit-line precharge control or equalize control signals. If switch control *C* (*C**) is stuck at on, high I_{DDQ} occurs during writes to cells on a complement (true) sub-bit line. If sense amplifier isolation control *IL* (*IR*) is stuck at on, high I_{DDQ} occurs when *EL* (*ER*) is enabled. Shorts between adjacent word lines cause high I_{DDQ} when one of the two word lines is asserted. A short from a word line to a sub-bit line causes high I_{DDQ} during precharge operations and whenever a restore operation causes an amplified cell signal to conflict with the driven signal on the shorted word line. Stuck or dummy word lines cause elevated I_{DDQ} if the word-line driver is functional and the associated word line is shorted to a supply node.

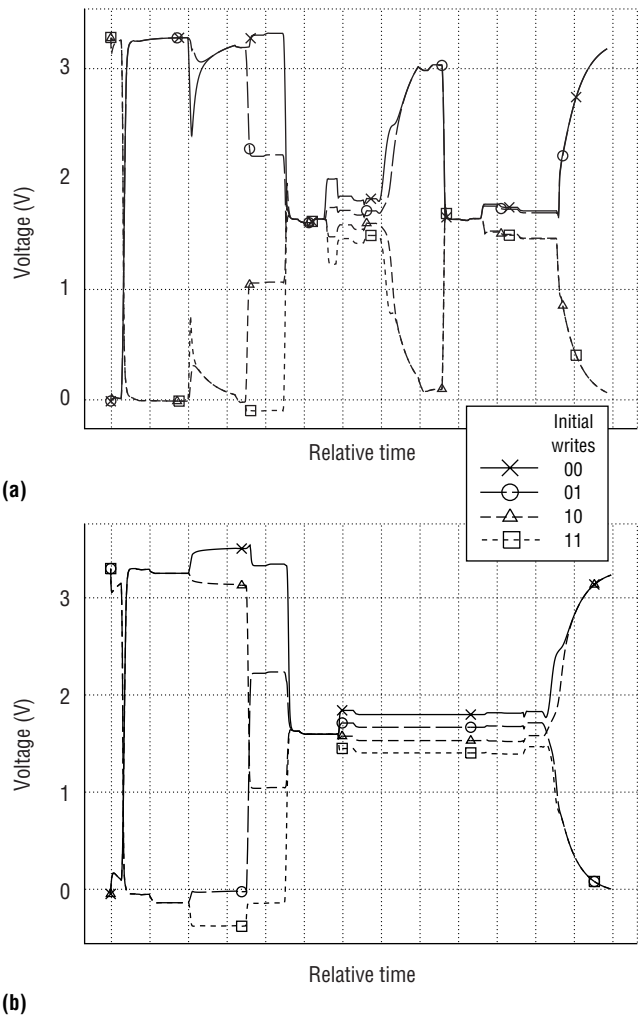

Figure 4. *Bit-line signals for a defect-free MLDRAM: MSB bit line* ***(a)****; LSB bit line* ***(b)****.*

Selected circuit simulation results. Figure 4 shows the SPICE simulation waveforms of the MSB and LSB sub-bit lines for a cell on a complement sub-bit line in a defect-free MLDRAM sequenced with the control waveforms from Figure

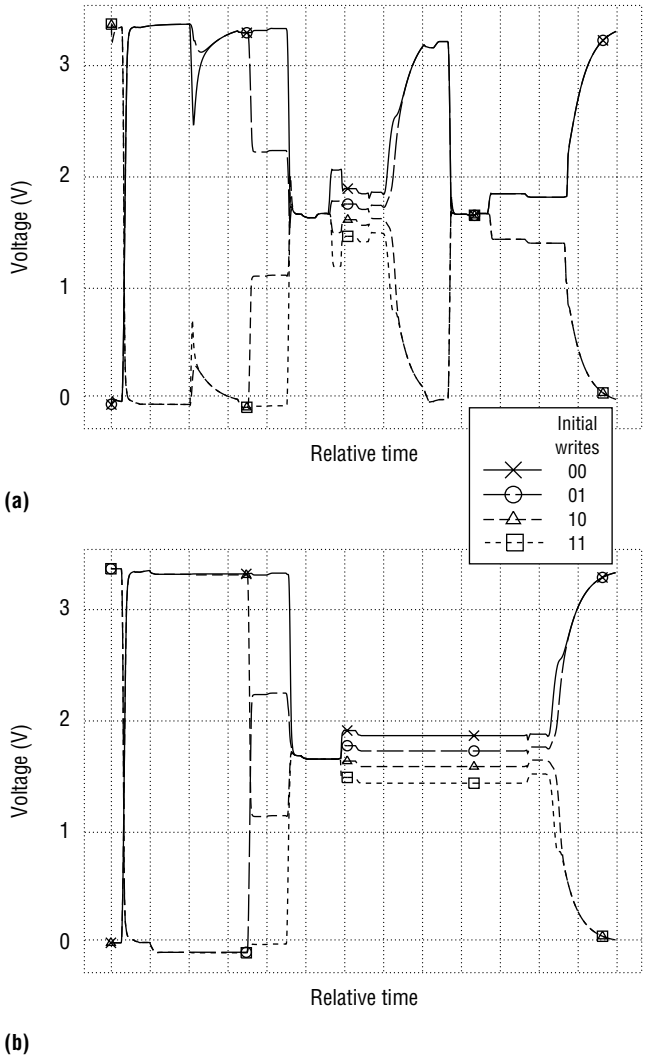

Figure 5. *Bit-line signals when C is stuck at 0: MSB bit line* ***(a)****; LSB bit line* ***(b)****.*

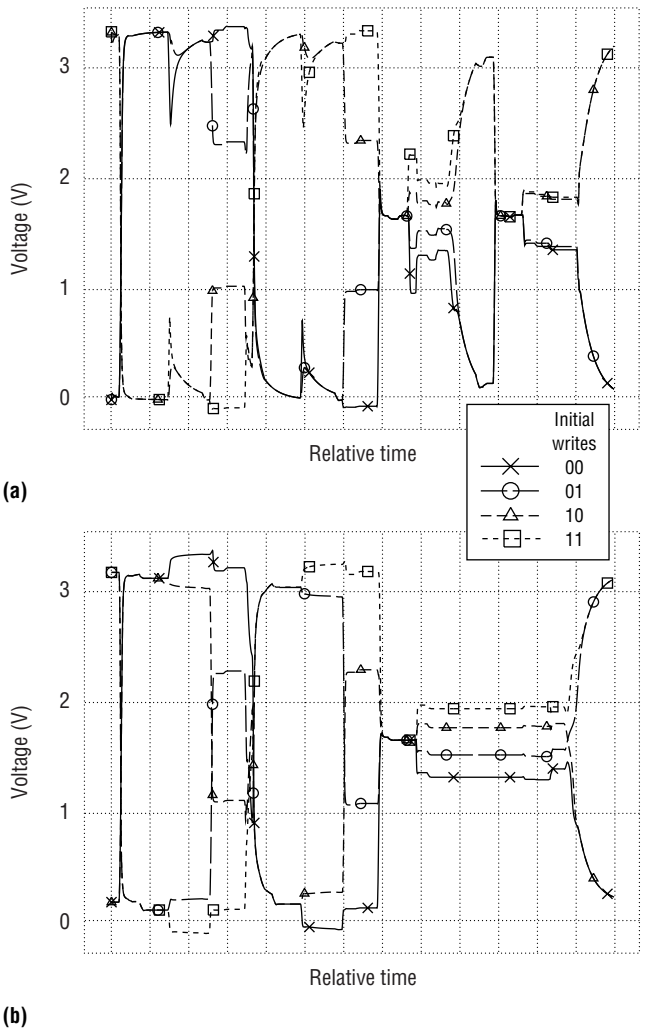

Figure 6. *WL-WL short causing one possible state-coupling behavior: MSB bit line* ***(a);*** *LSB bit line* ***(b)****.*

3. (MOSAID provided extracted DRAM simulation models. For an indication of the absolute timescale of Figures 4-7, see Gillingham.[3] We made no attempt to minimize MLDRAM access time.) The initial 40% of the plots shows superimposed bit-line signals corresponding to initial writes of 00, 01, 10, and 11. Immediately after the write operation, the sub-bit-line signals are equalized and precharged in preparation for a sensing operation. The middle 30% of the plots shows the MSB sensing, and the final 30% shows the LSB sensing.

Our manual analysis predicts that state coupling (fault class 4) will occur from the MSB to the LSB of the same cell if a switch control signal is stuck at off. Figure 5 shows the corresponding sub-bit-line signals from a SPICE simulation in which control signal *C* was fixed at 0 (off). Superficially the waveforms may not appear erroneous, but there is a problem affecting LSB sensing (note that the middle two signals no longer cross over). When we activate signal *C* to create and hold the LSB reference voltage on the appropriate sub-bit line (*BL* or *BR*), the voltage remains incorrectly at 1/2 V_{DD} if *C* is stuck at 0. Thus, the LSB that is read will always have the same value as the MSB; in other words, there is an MSB-to-LSB state-coupling fault.

Figures 6 and 7 show simulated bit-line signals that illustrate state-coupling behavior involving two cells (fault class 3). This behavior is the result of two adjacent word lines, say WL_i and WL_j, being shorted together. Let *i* and *j* be two cells on the same sub-bit-line pair, accessed using WL_i and WL_j, respectively. In Figure 6, we assume that cells *i* and *j* connect to the same sub-bit line; in Figure 7, we assume the two cells connect to different sub-bit lines. Both figures show the

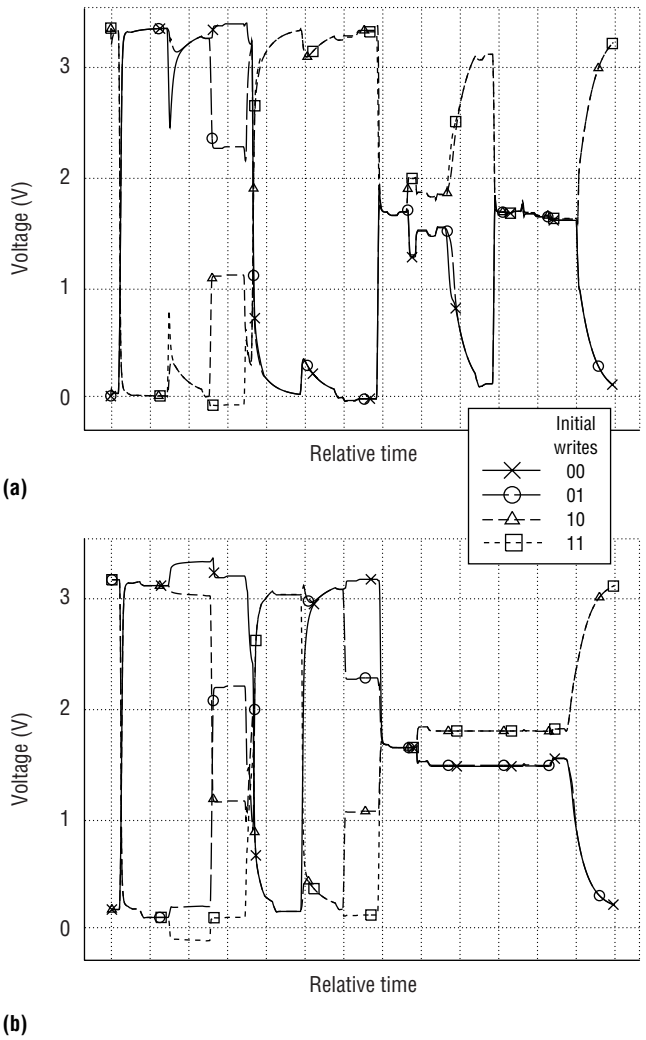

Figure 7. *WL-WL short causing a second kind of state-coupling behavior: MSB bit line* ***(a);*** *LSB bit line* ***(b)****.*

bit-line signals for cell *i* when the value (MSB, LSB), where MSB, LSB ∈ {0,1}, is written into cell *i*, then the complementary value ($\overline{\text{MSB}}$, $\overline{\text{LSB}}$) is written into cell *j*, and, finally, cell *i* is read back to verify its contents. The signals are identified with cell *i* values 00, 01, 10 and 11. Note that we assume negative signal encoding. The waveforms in Figure 6 may at first look correct until we remember that during sensing (the last half), the recovered MSB and LSB values are those stored in cell *j*, not those stored in cell *i*. Thus, this simulation represents the case in which the state of one cell is forced into a second cell.

Figure 7 shows that while the correct four voltages are formed and stored in cell *i*, the write to cell *j* runs into problems. On the MSB side, the MSB bit overwrites the LSB bit for cell *j*; thus, only 00 (the upper two curves) and 11 (the lower two curves) can be stored in cell *j*. Moreover, when cell *i* is sensed by its MSB and LSB side sense amplifiers, both its MSB and LSB are identical to the MSB value written to cell *j*. So this simulation represents the case in which cell *j*'s MSB transfers into cell *j*'s LSB and both the MSB and LSB of cell *i*.

Table 2. *Data tiles for the proposed 8m (4n) march test. Data value codes: 0 = 00, 1 = 01, 2 = 10, 3 = 11.*

Tile	BL	16 adjacent word lines
1	Even	0321 2103 3210 1032
	Odd	3012 3012 3012 3012
2	Even	2103 3210 1032 0321
	Odd	0123 0123 0123 0123
3	Even	3210 1032 0321 2103
	Odd	1230 1230 1230 1230
4	Even	1032 0321 2103 3210
	Odd	2301 2301 2301 2301

MLDRAM testing strategies

We now consider the problem of testing an MLDRAM for the expected faults. First we describe an efficient Boolean test for the cell array of an MLDRAM in its original form. Then we outline possible design-for-testability enhancement opportunities for Gillingham's MLDRAM.

Testing the unmodified MLDRAM. A useful first step in test design is to determine a lower bound on the length of any test that could detect all the modeled faults. For the MLDRAM fault model, the possibility of class 5 pattern sensitivities implies that the tester must write and then read back and verify all 16 possible joint states of pairs of adjacent cells. Hence, each cell must be written and read at least four times, once for each possible 2-bit state. Consequently, any test that detects all single or multiple faults in the model must contain at least 8*m* operations, *m* denoting the number of 2-bit storage cells. We found a Boolean test of precisely this minimum length that detects all singly occurring faults from the first six fault classes—all the clean faults in the model. This test also happens to provide worst-case sensing conditions to check for degraded noise margins.

Our test involves alternately writing into the MLDRAM and then reading back and verifying four different complete data loads. The test thus requires 8*m* memory accesses (read or write). If the MLDRAM stores *n* bits, the test contains only 4*n* operations because *n* equals 2*m* for a 2-bit-per-cell design. This rather short test length is an important benefit of the ability to simultaneously read and verify 2 bits per cell. We specify each memory load as a tiling[8] using the 2 × 16 rectangular tiles shown in Table 2.

The tiles' short dimension must align along adjacent paired sub-bit lines (memory array columns), and the long dimension along groups of 16 adjacent word lines (memory array rows). In addition, the tiles must abut so as to completely cover all cells in the array. During each pass through memory, cells must be addressed in the same order. Cells must be accessed sequentially along each bit line, bit line by bit line, in what is often called fast-Y address order (X and Y typically denote row and column address fields, respectively). We assume that physically adjacent word lines have Y addresses that differ by one. Otherwise, we assume that the tester descrambles Y addresses to ensure that physically adjacent word lines are accessed consecutively.

Theorem. The test detects all the considered defects that cause clean faults in the MLDRAM fault model.

Proof outline. We prove this claim by showing that the test detects single instances of all faults of classes 1–6 or their underlying defects. Inspection verifies that all 32 cell positions in a tile region are loaded with the four values 00, 01, 10, and 11 after all four tiles have been written and read back. Hence, the test detects all defects that cause faults of classes 1, 2, and 4.

We can verify that for all tile positions that are horizontal neighbors (including those going from the 16th to the first columns), adjacent MSB values will be written and verified with all four combinations: 0 and 0, 0 and 1, 1 and 0, and 1 and 1. Hence, the test detects all class 3 faults. Further, the test's fast-Y address ordering, with the four MSB transitions just noted, ensures that all class 6 faults are detected.

We also can verify that all 16 possible ordered pairs from $\{0, 1, 2, 3\}^2 = \{00, 01, 10, 11\}^2$ are present in the columns of each of the four data tiles in Table 2. Thus, all 16 different state combinations will be written to and read back from cell pairs connected to the same word line and to sub-bit lines connected to adjacent sense amplifiers. Hence, our test detects all sub-bit-line shorts that cause class 5 faults. So ends our proof.

Testing should detect degraded noise margins most easily during LSB sensing rather than MSB sensing because of the smaller differential signals involved. The LSBs of the tile data values form alternating word-line-aligned bars of 0's and 1's (both true and complement), as well as checkerboards (both true and complement). (Checkerboard patterns are frequently used to detect array noise and reduced noise margins in conventional DRAMs.[15]) This observation supports our claim that the proposed test provides worst-case LSB sensing conditions and should therefore be effective at testing for degraded noise margins. One could readily enhance the test to detect excessive cell leakage currents by inserting an appropriate delay (four delays in all) between the times when new data is written and read back from memory. This delay must be chosen so as to give marginally excessive leakage currents sufficient time, typically tens of milliseconds, to cause observable errors. We would need characterization experiments and detailed process data to establish the most effective delay time.

MLDRAM testability enhancements. How might we modify an MLDRAM to enhance its testability? If MLDRAM technology is to be used in a commodity design in competition with conventional 1-bit-per-cell DRAM, we could provide standard DRAM fast access modes, such as page mode. Using essentially the same peripheral circuits as in a DRAM, these fast modes would speed the writing and reading of test patterns to the cell array. In addition, parallel test modes similar to those available in many commercial DRAM parts could be implemented in an MLDRAM. We can also consider massively parallel test modes that turn the sense amplifiers into parallel test pattern generators, since our test uses only four different test patterns. However, such test modes have not appeared economically viable for conventional DRAMs, and it is unlikely that the situation would be significantly different for most MLDRAM implementations.

For test time reductions beyond those resulting from conventional page modes and parallel test modes, we could exploit the inherent structural parallelism of the two MLDRAM cell arrays. A first testing phase using standard patterns could test the left and right cell arrays in parallel as conventional DRAMs. Test time would decrease by roughly a factor of four because of the two-fold parallelism of the cell arrays and the simpler, single-step DRAM sensing operation. Modified control waveforms that keep the switch matrix pass transistor switches open would be a convenient way of isolating the two cell arrays.

After the cell arrays are tested, a second phase would test the switch matrix. To verify the operation of the switches controlled by the X, X^*, C, and C^* signals, this phase would perform write and verify cycles to cells on true and complement sub-bit lines. The test mode just outlined would require changes to the timing chains and control logic, but it would avoid much more costly changes to the larger cell arrays.

The simple structure of the proposed test is conducive to implementation in a built-in self-test circuit. Redeker[13] describes a BIST MLDRAM implementation using our $4n$ test. The test pattern generator's complexity was only about 100 gates. Simple Boolean operations on the row address, column address, and tile number form the test data.

THE WORK DESCRIBED HERE is only the first step in an ongoing detailed characterization of Gillingham's MLDRAM. Our future work will include a more detailed simulation-based sensitivity analysis of the basic design. We are fabricating a

test chip, and the resulting devices will permit much more detailed and realistic characterization studies. Uncovered design weaknesses may motivate the development of new tests. We plan to investigate further the design-for-testability enhancements mentioned here to better understand the tradeoffs among test time, fault coverage, fault diagnosability, design complexity, and silicon area increase. D&T

Acknowledgments

We presented an earlier version of this article at the 1998 IEEE International Workshop on Memory Technology, Design, and Testing. This research was supported by Micronet R&D under a STRAD (Short-Term Research and Development) grant, MOSAID Technologies Inc., the Canadian Microelectronics Corporation, and the Natural Sciences and Engineering Research Council of Canada under grant OGP0105567.

We thank Roger Colbeck, Peter Gillingham, John Wu, and Randy Torrance of MOSAID Technologies for their technical and financial support of research on multilevel DRAM at the University of Alberta. We are indebted to Gershom Birk and Albert Kwong at the University of Alberta for their assistance in getting the CAD tools and simulators to work properly with the circuit models. We also thank Dirk Niggemeyer and Joachim Mucha of the University of Hanover for proposing and then helping to arrange Michael Redeker's visit to Edmonton. Finally, we thank the anonymous referees for their helpful comments.

References

1. A.K. Sharma, *Semiconductor Memories: Testing, Technology, and Reliability*, IEEE Press, Piscataway, N.J., 1997.
2. T. Okuda and T. Murotani, "A Four-Level Storage 4-Gb DRAM," *IEEE J. Solid-State Circuits*, Vol. 32, No.11, Nov. 1997, pp. 1743-1747.
3. P. Gillingham, "A Sense and Restore Technique for Multilevel DRAM," *IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing*, Vol. 43, No. 7, July 1996, pp. 483-486.
4. D.A. Rich, "A Survey of Multivalued Memories," *IEEE Trans. Computers*, Vol. C-35, No. 2, Feb. 1986, pp. 99-106.
5. M. Horiguchi et al., "An Experimental Large-Capacity Semiconductor File Memory Using 16-Levels/Cell Storage," *IEEE J. Solid-State Circuits*, Vol. 23, No. 1, Feb. 1988, pp. 27-33.
6. T. Furuyama et al., "An Experimental 2-bit/Cell Storage DRAM for Macrocell or Memory-on-Logic Application," *IEEE J. Solid-State Circuits*, Vol. 24, No. 2, Apr. 1989, pp. 388-393.
7. J.P. Shen et al., "Inductive Fault Analysis of CMOS Integrated Circuits," *IEEE Design & Test of Computers*, Dec. 1985, Vol. 2, No. 4, pp. 13-26.
8. A.J. van de Goor, *Testing Semiconductor Memories: Theory and Practice*, John Wiley & Sons, New York, 1991.
9. H.-D. Oberle, M. Maue, and P. Muhmenthaler, "Enhanced Fault Modeling for DRAM Test and Analysis," *Proc. IEEE VLSI Test Symp.*, IEEE Computer Society Press, Los Alamitos, Calif., 1991, pp. 149-154.
10. M. Sachdev and M. Vertraelen, "Development of a Fault Model and Test Algorithms for Embedded DRAM," *Proc. IEEE Int'l Test Conf.*, IEEE CS Press, 1993, pp. 815-824.
11. A. Chan et al., "Electrical Failure Analysis in High Density DRAMs," *Records IEEE Int'l Workshop Memory Technology, Design and Testing*, IEEE CS Press, 1994, pp. 26-31.
12. B.F. Cockburn and J.A. Brzozowski, "Switch-Level Testability of the Dynamic Logic PLA," *Integration*, Vol. 9, No. 1, Feb. 1990, pp. 49-80.
13. M. Redeker, "Fault Models and Tests for a Prototype 4-level DRAM," Dipl.-Ing. thesis, Theoretical Electronics Institute, Design and Test Division, University of Hanover, Germany, 1998 (in English).
14. M. Redeker, B.F. Cockburn, and D.G. Elliott, "Fault Models and Test Strategies for a Two-Bit per Cell DRAM," *Proc. IEEE Int'l Workshop Memory Technology, Design and Testing*, IEEE CS Press, 1998, pp. 84-90.
15. Y. Konishi et al., "Analysis of Coupling Noise Between Adjacent Bit Lines in Megabit DRAMs," *IEEE J Solid-State Circuits*, Vol. 24, No. 1, Feb. 1989, pp. 35-42.

Michael Redeker is a test engineer in the LCD Development Group at Philips Semiconductors AG in Zurich, Switzerland. He received a Dipl.-Ing. degree in electrical engineering from the University of Hanover, Germany. He conducted research on multilevel memory fault modeling and testing for his final-year thesis while visiting the University of Alberta, where he participated in the work described in this article. Redeker is a member of the IEEE Computer Society.

Bruce F. Cockburn's biography and photo appear on page 21.

Duncan G. Elliott's biography and photo appear on page 41.

Send questions and comments about this article to Bruce F. Cockburn, University of Alberta, Dept. of Electrical and Computer Engineering, Edmonton, AB T6G 2G7, Canada; cockburn@ee.ualberta.ca.