A BIST Algorithm for Bit/Group Write Enable Faults in SRAMs

Saman Adham & Benoit Nadeau-Dostie, LogicVision, Inc.

Abstract

The use of group (or bit) write enable in memories is becoming very common in embedded memories. The circuitry used to achieve these functions need be thoroughly tested for different kind of defects using specific test sequence. However, most BIST algorithms assume that these write enables are forced active during the global write cycle in the BIST run. This paper presents a serial interface BIST algorithm that is used to test defect on bit/group write enables of these memories.

.

1. Introduction

Static Random Access Memories (SRAM) is being extensively used in system on chip (SOC) designs. It is common to see tens or even hundreds of SRAMs integrated into chips to perform different functions [1]. With the wide used of SRAMs new functionalities are being added to the memories to address specific design requirements. Among those is the ability to perform partial write operation where only specific portion of the memory word is written. This is achieved by disabling the memory internal write circuitry from writing the whole word by using write enable control ports. These write enable ports may control individual bits or a group of bits in a word. However, using the bit or group write enables does not necessarily replace the need for global write enable. Most memories maintain the global write enable port to initiate a write operation to the memory.

The existence of bit and group write enable functionality in SRAMs present new testing challenges. Treating these ports as global write enable surely reduces the fault coverage on the circuitry associated with them.

This paper presents a BIST algorithm and test circuit architecture for detecting defects in bit and group write enable that are internal to the memory. In section 2 we present the defects of interest and the fault models used. The algorithm for detecting bit write enable defects is presented next. Section 4 extends the algorithm for the detection of the group write enable faults. The memory interface circuit architecture is introduced in section 5. Conclusions are provided in section 6.

2. Bit and Group Write Enable Defects and Fault Models

Figure 1 shows a typical memory implementation to support bit write enable function. The active high write enable logic is implemented at the input data path (before the bit line decode circuit).

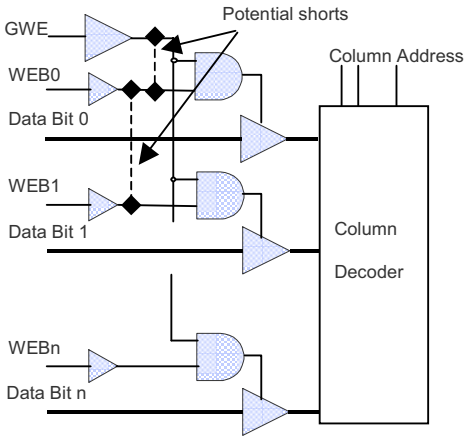

Fig. 1 A Bit write Enable implementation

Here the global write enable (GWE) is used to control all bit write enables (WEB). A data bit is written to the memory only when the global write enable and the specific bit write enable are active.

Controlling the WEBs and the GWE from one source in test mode will mask stuck active faults on the WEBs. Shorts between the WEB will also be masked. Furthermore, shorts between GWE and any WEB will not be detected, Wired-OR (for active low write enable signals); Wired-AND (for active high write enable signals) short between the Global Write Enable (GWE) and the bit write enable of the Most Significant Bit (MSB) will not be tested.

The coverage is further reduced if the write enable circuitry is implemented after the column decoder. The defects in this case will affect all bit lines in the memory array.

Traditionally test engineers use functionally generated patterns to test for these defects. The patterns perform read and write operations with different data and bit write enable values. However, generating these patterns is becoming too complicated for deeply embedded memories.

3. Detection Algorithm for Bit Write Enable

To detect the defects of interest, GWE and WEBs are controlled independently. The detection of these defects is achieved using a serial interfacing technique [2]. The memory data inputs and outputs are connected to form a shift path from one bit to the next. The first bit receives a single bit of data that will be used to apply the test data. Data is shifted from the LSB to the MSB.

Records of the 2004 International Workshop on Memory Technology, Design and Testing (MTDT'04)
1087-4852/04 $20.00 © 2004 **IEEE**

The algorithm is composed of two steps and applied to multiple addresses since some memories are designed to have the bit write enable logic after the column decoder, the algorithm is applied to all columns in a single row address.

The Algorithm:

Step(1): for all columns of row 0.
(RxW1)(R1W1)

Step(2): for all columns of row 0
$((R1W0)(R1W_m0))^B$ (R0W0)
$((R0W_m1)(R0W1))^B$ (R1W1)

Step (1) is used to initialize row 0 to all 1 value. Step (2) is used to detect all bit write enable shorts.

where:

B indicates that the operation is bit-serial and is performed on the full word width

W_m0 or **W_m1** indicates a write operation performed using a bit write enable mask. The bit write enable mask value is read from the memory itself.
The value 0 or 1 is applied to the serial input (SIMEM) of the memory (Din).

The test data input of all other bits is determined by the last value read

The first portion of step (2) **$((R1W0)(R1W_m0))^B$ (R0W0)** tests for all possible combinations of wired-AND (Wired-OR) shorts between write enables (bit and global). The second portion of the phase **$((R0W_m1)(R0W1))^B$ (R1W1)** tests for all Wired-OR (Wired-AND) shorts.

Table 1 shows the step-by-step execution of step (2) of the algorithm (shifting from LSB to MSB). Bit write enable signals are assumed to be active high in the table. For active low signals, the values indicated in the third column (bit write enable) must be inverted and the faults detected in the first half of the phase are Wired-OR faults whereas the faults detected in the second half of the phase are Wired-AND faults.

The Wired-OR (Wired-AND) short between the WEB of the MSB and the Global Write Enable (GWE) is not detected. Also, in some memory implementations, it is possible that branches of the GWE within a column are driven by more than one buffer. This leads to additional bits of the word for which the Wired-OR (Wired-AND) short to the GWE is not detected.

In order to detect these faults, the operations of Step (2) is repeated, the test data is shifted from the most significant bit to the least significant bit.

Table 1 Bit Write Enable Fault Detection

Op.	Input	WEB	Mem Cont.	Faulty Mem Cont.	Comments
R1W0	011	000	111	011	WEB0 SA1
R1Wm0	011	111	011	---	
R1W0	011	000	011	001	WEB1 SA1
R1Wm0	001	011	001	011	Wired-AND short WEB0 to {GWE, WEB1}
R1W0	000	000	001	000	Wired-AND short WEB1 to {GWE, WEB2}, Wired-AND short WEB2 to {WEB0}
R1Wm0	000	001	000	001	Wired-AND short WEB1 to {GWE,WEB2} Wired-AND short WEB2 to WEB0}
R0W0	000	000	000		Any of the Wired-AND faults will cause the compare of the MSB to 0 to fail
R0Wm1	100	000	000	100	Wired-OR short WEB0 to {GWE} or WEB0 SA1
R0W1	100	111	100	110	propagate the faults to serial output
R0Wm1	110	100	100	110	Wired-OR short WEB1 {GWE, WEB0} or WEB1 SA1
R0W1	110	111	110	111	propagate the faults to serial output
R0Wm1	111	110	110	111	Wired-OR

Records of the 2004 International Workshop on Memory Technology, Design and Testing (MTDT'04)
1087-4852/04 $20.00 © 2004 **IEEE**

					short WEB2 {GWE, WEB0, WEB1} or WE2 SA1
R0W1	111	111	111	Xx1	Any of the Wired-OR shorts will cause the compare to 0 to fail at this point
R1W1	111	111	111	---	only initializes last output

4. Group Write Enable Support

The case of group write enable is handled using the same algorithm. Let's assume that our memory is 3 times as wide (i.e. 9 bits) controlled as 3 groups of 3 bits each. Let's further assume that WEG0 is the write enable for DI0 to DI2, WEG1 controls DI3-DI5 and WEG2 controls DI 6-DI8.

The operations are summarized in the Table 2 (shifting from LSB to MSB). Some of the operations are redundant as far as the detection of faults on write enables is concerned.

At the end of the first set of operation (row 1 in Table 2) all Wired-AND short WEG0 to {GWE}is detected. Similarly Wired-AND shorts between WEG1,2 and {GWE} are detected after, at the end of operations, in row 2 and 3 of Table 2 respectively. Any of the Wired-AND faults will cause the compare of the MSB to 0 to fail when the R0W0 operation is performed (row 4).

At the end of the operations in row 5 Wired-OR short WEG0 to {GWE} will be detected. In row 6 Wired-OR short WEG1 {GWE, WEG0} is detected. Wired-OR short WEG2 {GWE, WEG0, WEG1} are detected at the end of operations of row 7. Furthermore, any of the Wire-OR shorts will cause the compare to 0 to fail at this point and before in many cases. The last operation (R1W1) is added to reinitialize the row.

To complete the algorithm coverage, the sequence of operations in Table 2 is repeated while shifting from MSB to LSB. In this case the group write enable is controlled from the least significant bit of the group instead of the most significant bit.

Op.	Din	Group WE	Mem Cont.	F.Mem Cont.
R1W0	011111111	000	111111111	011111111
R1Wm0	011111111	111	011111111	001111111
R1W0	001111111	000	011111111	000111111
R1Wm0	001111111	111	001111111	000011111
R1W0	000111111	000	001111111	000001111
RWm0	000111111	111	000111111	000000111
R1W0	000011111	100	000111111	000011111
R1Wm0	000011111	111	000011111	000001111
R1W0	000001111	100	000011111	000000111
R1Wm0	000001111	111	000001111	000000011
R1W0	000000111	100	000001111	000000001
R1Wm0	000000111	111	000000111	000000000
R1W0	000000011	110	000000111	000000011
R1Wm0	000000011	111	000000011	000000001
R1W0	000000001	110	000000011	000000000
R1Wm0	000000001	111	000000001	000000000
R1W0	000000000	110	000000001	000000000
R1Wm0	000000000	111	000000000	000000000
R0W0	000000000	000	000000000	xxxxxx111
R0Wm1	100000000	000	000000000	100000000
R0W1	100000000	111	100000000	110000000
R0Wm1	110000000	000	100000000	111000000
R0W1	110000000	111	110000000	111100000
R0Wm1	111000000	000	110000000	111110000
R0W1	111000000	111	111000000	111111000
R0Wm1	111100000	100	111000000	111100000
R0W1	111100000	111	111100000	111110000
R0Wm1	111110000	100	111100000	111111000
R0W1	111110000	111	111110000	111111100
R0Wm1	111111000	100	111110000	111111110
R0W1	111111000	111	111111000	111111111
R0Wm1	111111100	110	111111000	111111100
R0W1	111111100	111	111111100	111111110
R0Wm1	111111110	110	111111100	111111111
R0W1	111111110	111	111111110	111111111
R0Wm1	111111111	110	111111110	111111111
R0W1	111111111	111	111111111	111111111
R1W1	111111111	111	111111111	----------

5. Memory Interface Circuit Architecture

Special memory interface circuitry is needed to support the detection algorithms presented in section 3 and 4. Figure 2 illustrates the memory interface circuit for bit write enables for LSB to MSB shift direction. The bit write enable mask is generated from the memory data read of the same bit on the serial path. The WE_Mask signal is generated by the BIST FSM. This indicates that the operation is using the WEB mask value rather than the global write enable signal.

Table 2 Group Write Enable Fault Detection

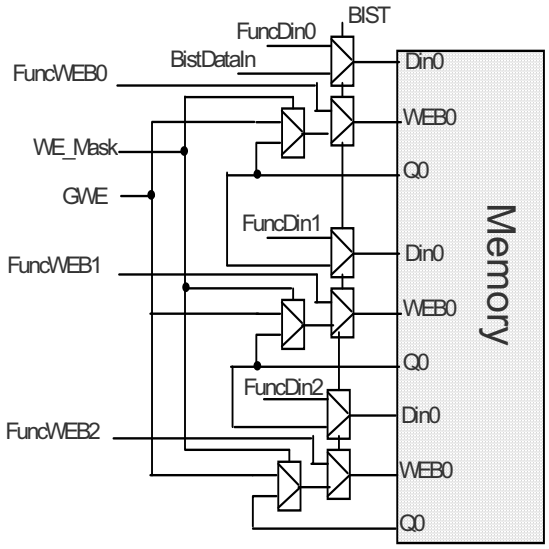

Figure 2 Memory Interface Circuitry for Bit Write Enable Testing

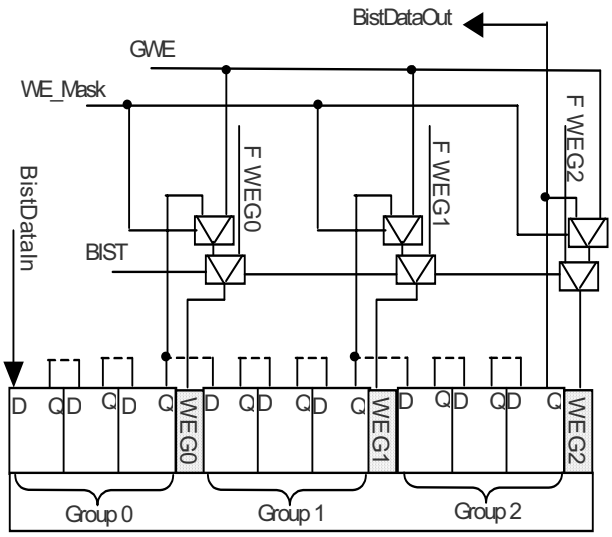

Figure 3 Memory Interface Circuit for Group Write Enable Testing

Few additional multiplexors are added to accommodate the bi-directional shifting.

Figure 3 shows the conceptual memory interface architecture when group write enable is used (shifting from LSB to MSB), where the dotted lines represent the serial interfacing chain.

The hardware is slightly modified to control a group write enable by the most significant output corresponding to the most significant input it is controlling (or the least significant output depending on the shifting direction).

6. Conclusions

This paper presented a new BIST algorithm to detect defects in the bit and group write enable circuitry. The algorithm is based on serially shifting data into the memory. Using an internally generated write enable mask enables the detection of the failures. The memory interface circuitry required by the algorithm is also presented. The new algorithm detects all write enable stuck-active faults and shorts. As a side benefit of applying this algorithm is that shorts within the memory data-path are also detected.

7. References

[1] J. Jayabalan. and J. Povazanec , “Integration of SRAM Redundancy into Production Test”, Proceedings of International Test Conference, 2002, pp. 187-193.

[2] B. Nadeau-Dostie, A. Slibert, V.K. Agarwal, “Serail Interfacing for Embedded Memory Testing”, IEEE Design and Test of computers, 4/1990, pp. 52-63.

[3] D. Adams, “High Performance Memory Testing”, Kluwer Academic Pub., USA, 2003.

[4] B. Nadeau-Dostie, “Design for At-Speed Test, Diagnosis and Measurement”, Kluwer Academic Pub., USA, 2000.

[5] Nadeau-Dostie, Benoit and Adham, Saman M.I., “Method and Test Circuitry for Testing Memory Internal Write Enable”, US Patent Pending.