Testing content addressable memories with physical fault models*

Ma Lin(马麟)[1,2,†], Yang Xu(杨旭)[1], Zhong Shiqiang(钟石强)[1], and Chen Yunji(陈云霁)[1]

(1 Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

(2 Graduate University of the Chinese Academy of Sciences, Beijing 100049, China)

Abstract: Content addressable memory (CAM) is widely used and its tests mostly use functional fault models. However, functional fault models cannot describe some physical faults exactly. This paper introduces physical fault models for write-only CAM. Two test algorithms which can cover 100% targeted physical faults are also proposed. The algorithm for a CAM module with N-bit match output signal needs only $2N$+$2L$+4 comparison operations and $5N$ writing operations, where N is the number of words and L is the word length. The algorithm for a HIT-signal-only CAM module uses $2N$+$2L$+5 comparison operations and $8N$ writing operations. Compared to previous work, the proposed algorithms can test more physical faults with a few more operations. An experiment on a test chip shows the effectiveness and efficiency of the proposed physical fault models and algorithms.

Key words: content addressable memory; test algorithm; physical fault model

DOI: 10.1088/1674-4926/30/8/085001 **PACC:** 0150K **EEACC:** 1265D

1. Introduction

Content addressable memory (CAM) is widely used in computing and communication applications. With the development of very large scale integration (VLSI) design, high performance microprocessors and system on chips (SOCs) need high density large capacity CAM design. Testing the CAM becomes more important.

A CAM cell is composed of a storage part and a comparison part. Most CAM fault models are functional fault models, which are introduced in Refs. [1–6], especially focusing on the faults in comparison part. For a CAM structure which has read ports, the faults in the storage part of CAM can be tested by March-like SRAM-based algorithms. Zhao gives build-in self-test (BIST) algorithms for CAM with both a read port and a write port in Ref. [1]. $10N$+$2L$ operations are needed to test concurrent read and comparison operation CAM and $12N$+$2L$ operations to test non-concurrent read and comparison operation CAM. Li uses $11N$+$2L$ operations to test CAM with a read port in Ref. [2]. Zhao gives a test algorithm using two data backgrounds to test CAM with a read port, which needs $20N$+$2L$ operations, in Ref. [3]. We proposed an instruction implementable algorithm to test CAM with a read port in Ref. [4]. For write-only CAM which does not have a read port, the faults in comparison part and storage part can be tested only by comparison operations. A full scale study of both the comparison part and storage part is needed to test write-only CAM. Lin builds functional fault models for write-only CAM, and gives an algorithm which needs $8N$+$4L$+1 in Ref. [5] and $10N$+$3L$+2 operations in Ref. [6] to test write-only CAM. Du uses $3N$+$2L$+2 operations to test the delay fault of CAM in Ref. [7]. Bhaysar builds a simple BIST logic which uses a linear feedback shift register (LFSR) to generate the test pattern in Ref. [8]. However, it needs $NL\log_2 N$ operations to finish the test.

The functional fault models are intuitive for CAM structure. However, they cannot describe some physical faults clearly, and for the delay faults in CAM, even though some functional test algorithms can test them, no functional fault models can analyze them. Therefore current functional fault models do not suit these physical faults. For example, the stuck-open fault of transistor M1 or M2 in Fig. 1 is difficult to classify into any type of functional fault. The statistic probability of these faults is not less than the other faults. If both M1 and M2 are opened, the value of G is out of control and may remain at the previous value. In this situation, it is not a simple functional fault, and it needs an operation sequence to test the fault. Furthermore, physical fault modes can also uncover delay faults in the CAM circuits. For write-only CAM, there is only a comparison operation to observe the test result. A physical faults analysis of a full-scale CAM circuit is needed.

In this paper, physical faults based on the write-only CAM structure are studied. A full-scale study of both the comparison part and the storage part is used. Physical faults and their functional behaviors are analyzed. The physical fault models can cover all the functional fault models, but some physical faults cannot be detected by any current functional fault models. The physical fault models can also cover the delay faults. Test approaches to detect them are also introduced. Two test algorithms which can test all the physical faults

* Project supported by the National Natural Science Foundation of China (No. 60603049), the National High Technology Research and Development Program of China (Nos. 2008AA110901, 2007AA01Z112, 2009AA01Z125), the State Key Development Program for Basic Research of China (No. 2005CB321600), and the Beijing Natural Science Foundation (No. 4072024).

† Corresponding author. Email: lma@ict.ac.cn

Received 23 February 2009, revised manuscript received 26 March 2009

© 2009 Chinese Institute of Electronics

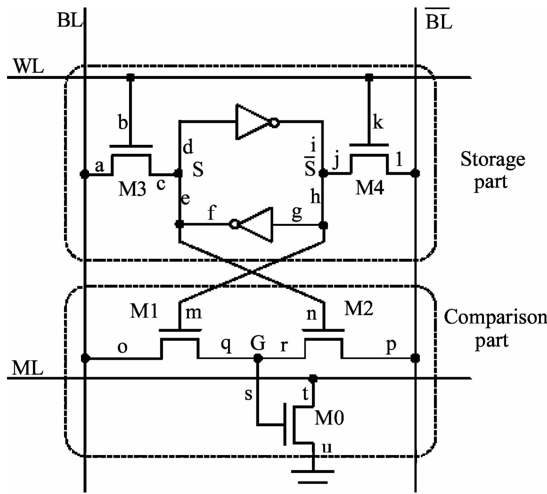

Fig. 1. CAM cell structure.

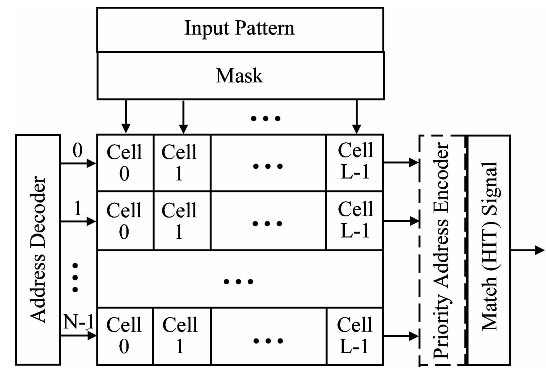

Fig. 2. CAM array.

involved in this paper are given. One of the proposed algorithms has been used in a CAM BIST design of a test chip. Experiment shows that the test algorithm and the physical fault models are efficient and effective for write-only CAM testing.

2. Physical fault models for CAM testing

A normal write-only CAM cell structure is shown in Fig. 1. It is composed of a storage part and a comparison part. The storage part is similar to an RAM (random access memory) cell. For writing operation, WL (write line) is pulled up. Transistors M3 and M4 are on and connect BL, $\overline{\text{BL}}$ (bit line) to S, $\overline{\text{S}}$ (state), respectively. The data is saved to the CAM cell. For comparison operation, ML (match line) is pre-charged to HIGH. S and $\overline{\text{S}}$ determine which transistor M1 or M2 is on or open. If BL equals S and $\overline{\text{BL}}$ equals $\overline{\text{S}}$, the G value is pulled down and transistor M0 is open. ML is still HIGH, which means match. Otherwise, the G value is pulled up and sets M0 on. ML is pulled down, meaning mismatch.

A CAM array consists of N words and a CAM word consists of L-bit CAM cells, as shown in Fig. 2. The CAM cells in one word share one WL and one ML. Every writing operation updates L bits CAM cells in a word, and the word address comes from the address decoder. The input pattern for comparison operation is masked by the MASK register if the corresponding bit is set to 0. Comparison results of masked bits do not affect other CAM cells. Comparison operation compares all the words concurrently and results in LOW ML if there is more than one CAM cell mismatching in a word. All the MLs of every word are assembled to an N-bit match signal. For some designs not allowing multi-match, if there is more than one bit ONE in the match signal, PAE (priority address encoder) circuits are needed. Also, some designs need only a one-bit output HIT signal which is the result of OR operation of all MLs.

Functional fault models are widely used in CAM testing. In Ref. [6] several functional fault models are introduced, such as the stuck-matched fault, the stuck-mismatched fault, the conditional-match fault, the partial-match fault, the equivalence-mismatch fault, the inequivalence-match fault, and so on. In other research, similar fault models are used. From the functional opinion, there are four classes of faults: storing 0 comparing 0 (S0C0) fault, storing 0 comparing 1 (S0C1) fault, storing 1 comparing 0 (S1C0) fault and storing 1 comparing 1 (S1C1) fault. If the four classes are tested, the functional faults are all detected. However, these four classes of faults are limited to one cycle of operations, and our research shows that functional fault models can not cover all the physical faults.

In the paper, it is assumed that (1) writing and comparison operations can not be executed in the same cycle and (2) there is only one fault in a CAM cell. The location of the fault is unknown but the fault is stable. The terminology used in this paper is as follows. C_s^v is used to represent comparison operation, where s means CAM cell state and v means the value to be compared. When s is unknown or unconcerned, it can be ignored. W_a^v represents a writing operation, where a is the writing address and v is the writing value. If the writing address is not useful it can be ignored. The compare value and writing value can be 1, 0, or X (unconcerned).

The physical faults discussed in this paper are classified as transition stuck faults, signal stuck-at faults, circuit open faults and signal bridge faults. The MASK register faults are also discussed with stuck-at faults. Faults crossing the cells are not discussed in this paper, and the faults of the two inverters in the storage part are not discussed here because these faults can be observed when testing other faults.

2.1. Transistor stuck faults

Transistor stuck faults include stuck-open faults and stuck-on faults. In the CAM cell structure, there are five transistors. The stuck faults of transistors are listed in Table 1, where the first column indicates the transistor which has the stuck fault, the second column shows the type of fault, and the last column gives the test approaches to test the fault.

For an M1 stuck-open fault, when the CAM cell state is 0, $\overline{\text{S}}$ is 1, M1 should be on; however, the fault makes M1 open. At the same time, M2 is also open, and the value of G is out of control. The G signal may remain at the previous value. If the previous value is 0, the fault can be detected by sequence $C_1^1 W^0 C_0^1$. If the previous value is 1, the fault can be detected by sequence $C_1^0 W^0 C_0^0$. The operations in the sequence do not

Table 1. Transistor stuck faults.

Trans	Defect	Test approach
M0	Stuck-Open	$\{C_1^0\}$ or $\{C_0^1\}$
M1	Stuck-Open	$\{C_1^0 W^0 C_0^0\}$ or $\{C_1^1 W^0 C_0^1\}$ $\{C_1^0\}$ or $\{C_0^1\}$
M2	Stuck-Open	$\{C_0^1 W^1 C_1^1\}$ or $\{C_0^0 W^1 C_1^0\}$ $\{C_1^0\}$ or $\{C_0^1\}$
M3	Stuck-Open	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
M4	Stuck-Open	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
M0	Stuck-On	$\{C_0^0\}$ or $\{C_1^1\}$
M1	Stuck-On	$\{C_1^1\}$ or $\{C_1^0\}$
M2	Stuck-On	$\{C_0^1\}$ or $\{C_0^0\}$
M3	Stuck-On	$\{C_0^1\}$ or $\{C_1^0\}$
M4	Stuck-On	$\{C_0^1\}$ or $\{C_1^0\}$

Table 2. Signal stuck-at faults.

Signal	Defect	Test approach
ML	Stuck-at-0	$\{C_0^0\}$ or $\{C_1^1\}$
WL	Stuck-at-0	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
BL	Stuck-at-0	$\{C_0^1\}$
$\overline{\text{BL}}$	Stuck-at-0	$\{C_1^0\}$
S	Stuck-at-0	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
$\overline{\text{S}}$	Stuck-at-0	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
G	Stuck-at-0	$\{C_0^1\}$ or $\{C_1^0\}$
ML	Stuck-at-1	$\{C_0^1\}$ or $\{C_1^0\}$
WL	Stuck-at-1	$\{C_0^1\}$ or $\{C_1^0\}$
BL	Stuck-at-1	$\{W^0 C^0\}$ or $\{W^0 C^1\}$
$\overline{\text{BL}}$	Stuck-at-1	$\{W^1 C^0\}$ or $\{W^1 C^1\}$
S	Stuck-at-1	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
$\overline{\text{S}}$	Stuck-at-1	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
G	Stuck-at-1	$\{C_0^0\}$ or $\{C_1^1\}$

need to be executed back to back. Comparison operations can be inserted between the last operation and the second operation. The G value may also become 0 because of leakage. This is a functional stuck-match fault. Then C_0^1 or C_1^0 can test the fault. An M2 stuck-open fault is similar to the M1 stuck-open fault. The test sequence is $C_0^1 W^1 C_1^1$ or $C_0^0 W^1 C_1^0$.

For M3 and M4 stuck-open faults, the fault makes the delay timing of the writing operation increase. However, the fault may not influence the function behavior if the operation frequency is not high enough. In a high speed test, a comparison operation next to a writing operation which changes the CAM cell state can test the faults.

2.2. Signal stuck-at faults

The signals in a CAM cell are four in/out signals BL, $\overline{\text{BL}}$, WL, ML, and three state signals S, $\overline{\text{S}}$ and G. The stuck-at faults of these signals are listed in Table 2, where the first column indicates the signal which has the stuck fault, the second column shows the type of fault, and the last column gives the test approaches to test the fault.

For a BL stuck-at-1 fault, value 1 can be written to the CAM cell correctly. When writing value 0, BL and $\overline{\text{BL}}$ are both 1, and it is unknown which value 0 or 1 is written to the cell. If value 1 is written, the fault can be detected by W^0C^0 or W^0C^1. If value 0 is written, the fault can be detected by a value 0 comparison operation, i.e. W^0C^0. A $\overline{\text{BL}}$ stuck-at-1 fault is similar to the BL stuck-at-1 fault.

Table 3. Circuit open faults.

Open point	Test approach
a, l	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
b, k	$\{C_0^1\}$ or $\{C_1^0\}\{W^0 C^x\}$ or $\{W^1 C^x\}$
c, j	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
d, e, f, g, h, j	$\{W^0 C^x\}$ or $\{W^1 C^x\}$
m	$\{C_1^1\}$ or $\{C_1^0\}$ or $\{C_0^1\}$ $\{C_1^0 W^0 C_0^0\}$ or $\{C_1^1 W^0 C_0^1\}$
n	$\{C_0^1\}$ or $\{C_0^0\}$ $\{C_0^1 W^1 C_1^1\}$ or $\{C_0^0 W^1 C_0^0\}$
o, q	$\{C_1^0 W^0 C_0^0\}$ or $\{C_1^1 W^0 C_0^1\}$
p, r	$\{C_0^1 W^1 C_1^1\}$ or $\{C_0^0 W^1 C_1^0\}$ $\{C_1^0\}$ or $\{C_0^1\}$
s	$\{C_1^0\}$ or $\{C_0^1\}$
	$\{C_0^0\}$ or $\{C_1^1\}$
t	$\{C_1^0\}$ or $\{C_0^1\}$
u	$\{C_1^0\}$ or $\{C_0^1\}$

2.3. Circuit open faults

As shown in Fig. 1, there are 21 circuit open points (a–u) of the circuit. The circuit open faults of the CAM cell do not include the four in/out signals. These four signals are connected to all row or column CAM cells. An open fault of WL causes some CAM cells write errors, and it may be the same for the WL stuck-at-0/1 fault. BL and $\overline{\text{BL}}$ open faults can also be the same as the stuck-at-0/1 faults for some CAM cells. The open fault of ML causes comparison result lost, and can be detected by comparison operations. The circuit open faults are listed in Table 3.

For a, l, c, j point open faults, the fault may cause writing operation time increasing. So the test approaches need to be executed in high speed.

For b, k point open faults, the fault causes the state of M3 to be unknown. If the fault can turn on M3, the comparison value will write to the CAM cell. The result of the comparison operation will be MATCH. It can be tested by C_0^1 or C_1^0. If the fault cannot turn on M3, it causes the writing operation to need more executing time, which may lead to writing error. The fault can be detected by high speed or at-speed test of W^0C^x or W^1C^x.

For point d, e, f, g, h and i open faults, these six points are in the push-pull inverter pair of the storage part. When an open fault occurs at these points, it destroys the storage part of the CAM cell. The CAM cell cannot store the value and the voltage of S and $\overline{\text{S}}$ come down, so M1 and M2 cannot be turned on. This makes the comparison operation get the same result whenever it should be match or mismatch. So the fault can be detected by W^0C^x or W^1C^x. The fault is not similar to delay faults. Actually, because the voltage drop of S and $\overline{\text{S}}$ needs some time, the fault may not be detected if there are frequent writing operations. But because the drop time is uncertain, a slower test speed, which may be the valid lowest frequency

Table 4. Signal bridge faults.

Signal	Stronger	Test approach
WL and ML	WL	$\{C_0^0\}$ or $\{C_1^1\}$
WL and ML	ML	$\{C_0^1\}$ or $\{C_1^0\}$
WL and BL	WL	$\{C_0^1\}$ or $\{W^0C^x\}$
WL and BL	BL	$\{W^0C^x\}$ or $\{C_0^1\}$
WL and $\overline{\text{BL}}$	WL	$\{C_1^0\}$ or $\{W^1C^x\}$
WL and $\overline{\text{BL}}$	$\overline{\text{BL}}$	$\{W^1C^x\}$ or $\{C_0^1\}$
WL and S	WL	$\{C_1^0\}$ or $\{C_1^1\}$ or $\{W^0C^x\}$
WL and $\overline{\text{S}}$	WL	$\{C_0^1\}$ or $\{C_0^0\}$ or $\{W^1C^x\}$
WL and G	WL	$\{C_0^1\}$ or $\{C_1^0\}$
WL and BL	ML	$\{C_0^0\}$
WL and BL	BL	$\{C_0^0\}$ or $\{C_0^1\}$
WL and $\overline{\text{BL}}$	ML	$\{C_1^1\}$
WL and $\overline{\text{BL}}$	$\overline{\text{BL}}$	$\{C_1^1C_1^0\}$
WL and S	ML	$\{C_0^0\}$ or $\{C_0^1\}$
WL and $\overline{\text{S}}$	ML	$\{C_1^0\}$ or $\{C_1^1\}$
WL and G	ML	$\{C_0^0\}$ or $\{C_1^1\}$
WL and $\overline{\text{BL}}$	BL	$\{C_1^0\}$ or $\{C_1^1\}$
WL and $\overline{\text{BL}}$	$\overline{\text{BL}}$	$\{C_0^1\}$ or $\{C_0^0\}$
WL and S	BL	$\{C_1^0\}$ or $\{C_0^1\}$
WL and $\overline{\text{S}}$	BL	$\{C_0^0\}$ or $\{C_1^1\}$
WL and G	BL	$\{C_1^0\}$ or $\{C_1^1\}$
$\overline{\text{BL}}$ and S	$\overline{\text{BL}}$	$\{C_1^1\}$ or $\{C_0^0\}$
$\overline{\text{BL}}$ and $\overline{\text{S}}$	$\overline{\text{BL}}$	$\{C_0^1\}$ or $\{C_1^0\}$
$\overline{\text{BL}}$ and G	$\overline{\text{BL}}$	$\{C_0^1\}$ or $\{C_0^0\}$
S and $\overline{\text{S}}$		$\{C_0^x\}$ or $\{C_1^x\}$
S and G	S	$\{C_0^1\}$ or $\{C_1^1\}$
$\overline{\text{S}}$ and G	$\overline{\text{S}}$	$\{C_1^0\}$ or $\{C_0^0\}$

of the CAM design, is needed to detect the fault.

For m, n point open faults, the state of M1 or M2 is unknown. If the fault makes the transistor open, it is same as an M1, M2 stuck-open fault; if the fault sets the transistor to on, it is same as an M1, M2 stuck-on fault.

For o, q point open faults, when M1 is on, the fault is active. Because M2 is open at the same time, the value of G cannot be controlled. This is similar to an M1 stuck-open fault. The r, p point open fault is similar to an M2 stuck-open fault.

2.4. Signal bridge faults

Signal bridge faults in the CAM cell are made by short defects between two of seven signals. When two signals have the same value, the fault is masked. Otherwise the bridge fault makes two signals have same value. The shorted value is dependent on the driving strength of the signals, and a stronger signal will dominate the shorted value. Some bridge faults cause more power consumption and can be detected by power testing. The signal bridge faults are listed in Table 4. The first column is the signal pair where the short defect is located. The second column indicates the stronger signal and the last column gives the test approaches. WL, BL, $\overline{\text{BL}}$ are input signals. S and $\overline{\text{S}}$ are weak drive signals. G is an internal signal. Normally, the first three signals are stronger than the last three signals, and G is the weakest signal. In this paper, if the first three signals meet the other signals, the last three signals are not treated as stronger drive signals. The G signal is not treated as the stronger signal at anytime. ML needs to pre-charge in a comparison operation. In that time ML is the stronger drive signal, and its value is 1.

S and $\overline{\text{S}}$ short faults mean that the CAM cell cannot store the value, and S and $\overline{\text{S}}$ are at the same voltage. So the fault makes the comparison operation get the same result whenever it is match or mismatch. The fault can be detected by C_0^x or C_1^x.

2.5. MASK register faults

For exact description, it is assumed that when a bit of the MASK register is 0, the corresponding bit of the comparison data is masked. The stuck-at faults of the MASK register are noted in this paper. The stuck-at-0 fault can be detected in the normal test programs which do not use the MASK register. The fault masks the comparison result. The stuck-at-1 fault of the MASK register can be detected by setting the MASK register to 0 and making all the CAM cells mismatch. When the fault exists, the comparison operation results in mismatch; otherwise it should be match.

3. Proposed CAM test algorithms

3.1. Primary test approaches

After studying the physical fault models, it can be found that, although there are many fault models and every fault model has its own test approaches, the test approaches can be classified into several primary test approaches (PTAs).

(1). Match comparison: C_0^0 and C_1^1.
(2). Mismatch comparison: C_0^1 and C_1^0.
(3). Write error test: W^0C^x and W^1C^x.
(4). Test sequence: $C_1^0W^0C_0^0$ or $C_0^1W^1C_1^1$.
(5). Test sequence: $C_0^0W^1C_1^0$ or $C_1^1W^0C_0^1$.
(6). MASK register test: $C_{M=0}$.

For PTA3, write error test, the writing operation should change the CAM cell state. The comparison operation needs to be next to the writing operation. The test approaches should be executed in high speed or at-speed to test the faults which increase the writing time. PTA3 also needs to be executed at slower speed to test the open faults of the pull-push inverter pair circuits. Because the period between two writing operations can not be determined easily, the slower speed test can be finished by executing the CAM test in different frequencies.

For PTA4 and PTA5, the test sequences do not require one operation right after another operation. Between the first comparison operation and writing operation there can be other writing operations. Also, between the writing operation and the last comparison operation, other comparison operations can even be inserted.

PTA6 is to test the MASK register stuck-at-1 fault. $M = 0$ means every bit of the MASK register is set to 0. The comparison operation needs every CAM cell to have a mismatch result. The MASK register stuck-at-0 fault can be tested through

normal comparison operations.

When every CAM cell finishes six PTAs, all faults discussed in Section 2 can be detected.

3.2. Test algorithms

Two algorithms are proposed here. One is for the CAM design which has an N-bit match output signal, and the other is for the CAM design which has only a HIT output signal.

3.2.1. N-bit match signal algorithm

For the CAM module which has an N-bit match signal for comparison operations, the test algorithm (Algo. A) is shown below.

Pass 1: $\left\{W_k^0\right\}_{k=0}^{N-1}$

Pass 2: $\left\{C^{2^L-1}\right\}$

Pass 3: $\left\{W_k^{2^L-1}; C^{2^L-1}\right\}_{k=0}^{N-1}$

Pass 4: $\left\{C^0\right\}$

Pass 5: $\left\{W_k^0; C^0\right\}_{k=0}^{N-1}$

Pass 6: $\left\{W_k^{2^L-1}\right\}_{k=0}^{N-1}$

Pass 7: $\left\{C^{2^L-1-2^k}\right\}_{k=0}^{L-1}$

Pass 8: $\left\{C^{2^L-1}\right\}$

Pass 9: $\left\{W_k^0\right\}_{k=0}^{N-1}$

Pass 10: $\left\{C^{2^k}\right\}_{k=0}^{L-1}$

Pass 11: $\left\{C_{M=0}^{2^L-1}\right\}$

Pass1 initializes the CAM array. The comparison operation in the kth iteration of Pass3 executes C_1^1 for the current writing operation, and for subsequent iterations it executes C_0^1. The first C_0^1 is Pass2. So Pass2 and Pass3 complete the test sequence $C_0^1W^1C_1^1$. In a similar way, Pass4 and Pass5 complete the test sequence $C_1^0W^0C_0^0$. PTA4 is tested. At the same time, PTA1 and PTA3 are also tested in Pass3 and Pass5.

After Pass5, the CAM array is flushed to 0. The last comparison operation in Pass5 makes a C_0^0 for all the CAM cells. Pass6 writes value 1 to all the CAM cells. The Pass7 is named the walk-0 test in previous presentations. In the kth iteration, except the kth bit, all cells with the value 1 are compared. If there is a fault in the kth bit, it can be detected. Pass7 executes C_1^0. So the $C_0^0W^1C_1^0$ is completed. In a similar way, Pass8, Pass9 and Pass10 complete the $C_1^1W^0C_0^1$. PTA5 is tested. At the same time, PTA2 is also tested in Pass7 and Pass10.

$M = 0$ in Pass11 means that all bits of the input pattern are masked. Comparison with 1 will return a match result. If there is a stuck-at-1 fault in the MASK register, there will be a mismatch result, and the fault is detected. So PTA6 is tested. For high speed and low speed testing of PTA3, the algorithm needs to be executed in both high and low clock frequencies.

The algorithm finishes all six PTAs, and it needs $5N$ writing operations and $2N+2L+4$ comparison operations.

3.2.2. HIT signal only algorithm

For the CAM design which has only the a HIT output signal for comparison operations, the test algorithm (Algo. B) is shown below.

Pass 1: $\left\{W_k^0\right\}_{k=0}^{N-1}$

Pass 2: $\left\{C^{2^L-1}\right\}$

Pass 3: $\left\{W_k^{2^L-1}; C^{2^L-1}; W_k^0\right\}_{k=0}^{N-1}$

Pass 4: $\left\{W_k^{2^L-1}\right\}_{k=0}^{N-1}$

Pass 5: $\left\{C^0\right\}$

Pass 6: $\left\{W_k^0; C^0; W_k^{2^L-1}\right\}_{k=0}^{N-1}$

Pass 7: $\left\{C^{2^L-1}\right\}$

Pass 8: $\left\{W_k^0\right\}_{k=0}^{N-1}$

Pass 9: $\left\{C^{2^k}\right\}_{k=0}^{L-1}$

Pass 10: $\left\{C^0\right\}$

Pass 11: $\left\{W_k^{2^L-1}\right\}_{k=0}^{N-1}$

Pass 12: $\left\{C^{2^L-1-2^k}\right\}_{k=0}^{L-1}$

Pass 13: $\left\{C_{M=0}^{2^L-1}\right\}$

Pass1 initializes the CAM array. The comparison operation in the kth iteration of Pass3 executes C_1^1 for the current writing operation, and for subsequent iterations it executes C_0^1. The first C_0^1 is Pass2. So Pass2 and Pass3 complete the test sequence $C_0^1W^1C_1^1$. In a similar way, Pass5 and Pass6 complete the test sequence $C_1^0W^0C_0^0$. PTA4 is tested. At the same time, PTA1 and PTA3 are also tested in Pass3 and Pass6.

After Pass6, the CAM array is flushed to 1. The comparison operation in Pass7 makes a C_1^1 for all the CAM cells. Pass8 writes value 0 to all the CAM cells. Pass9 is named the walk-1 test in previous presentations. In the kth iteration, except the kth bit, all cells with the value 0 are compared. If there is a fault in the kth bit, it can be detected. Pass9 executes C_1^0. So the $C_1^1W^0C_0^1$ is completed. In a similar way, Pass10, Pass11 and Pass12 complete the $C_0^0W^1C_1^0$. PTA5 is tested. At the same time, PTA2 is also tested in Pass9 and Pass12. Pass13 is same as the first algorithm. It tests PTA6. For high speed and low speed testing of PTA3, the algorithm needs to be executed in both high and low clock frequencies.

The comparison operations in Pass3 and Pass6 result in one matched word for fault-free circuits and no matched words for fault circuits. The comparison operations in Pass7 and Pass10 match all the words. But these two passes are not used to detect the fault. They are used to prepare for Pass8 and Pass11, so their results can be ignored. So the second algorithm is suitable for one-bit HIT output CAM design.

The algorithm finishes all six PTAs, and it needs $8N$ writing operations and $2N+2L+5$ comparison operations.

3.3. Algorithm comparison

A comparison between the algorithms is listed in Table 5. In the table, the second column is the test algorithm in Ref. [6] and the third column is the test algorithm in Ref. [7]. The proposed algorithm with N-bit match signal is listed in the 4th column and the algorithm with HIT-signal-only is listed in the

Table 5. Algorithms comparison.

	Algo. [6]	Algo. [7]	Algo. A	Algo. B
PTA1	√	√	√	√
PTA2	√	√	√	√
PTA3	√	×	√	√
PTA4	×	×	√	√
PTA5	×	×	√	√
PTA6	√	×	√	√
Comp	$2N$+$2L$+2	$2L$+2	$2N$+$2L$+4	$2N$+$2L$+5
Write	$6N$	$3N$	$5N$	$8N$

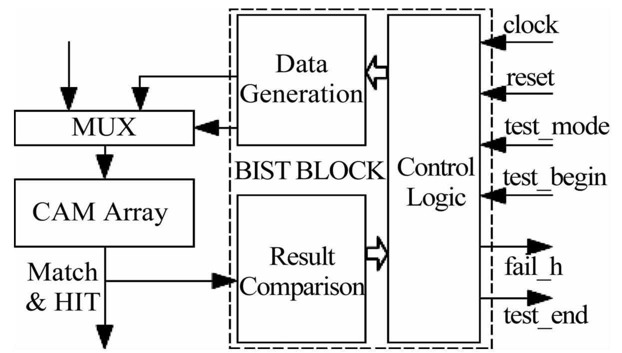

Fig. 3. CAM BIST logic.

last column. The last two lines are comparison operations and writing operations which the algorithms need.

It is shown that, because PTA4 and PTA5 can not be described in functional fault models, they are only tested in proposed algorithms. PTA4 and PTA5 can test physical faults such as transistor M1 or M2 stuck-open faults and some circuit open faults. The probability of these faults is not less than other physical faults. It is necessary to test them.

The algorithm in Ref. [6] and the proposed algorithm B are both HIT-signal-only suitable. It can be found that the proposed algorithm can test all the introduced physical faults with the cost of only 3 more comparison operations and $2N$ more writing operations. The algorithm in Ref. [7] is used to test the delay faults in the CAM circuits; however, it actually can not find delay faults in writing operations because it does not have a comparison operation right after the CAM word writing operation. The proposed algorithms can test all the delay faults in Ref. [7] and the delay faults in writing operations.

4. Experiment and results

The proposed algorithm (Algo. A) is implemented in a 64×64 bits CAM BIST circuit in a test chip. The CAM design in the test chip is a fully custom design in a 65 nm process. The frequency of the test chip is 800 MHz. The BIST circuit based on the algorithm is running at the same frequency. The CAM design uses both the 64-bit Match signal and the one-bit HIT signal as the CAM output signals. The BIST block structure is shown in Fig. 3. If a fault is detected, the BIST output fail_h will be pulled up for one clock cycle, and then it is returned to zero. The BIST output test_end will be pulled up when the test is finished. After synthesis, the BIST circuit has 115 registers and 3717 cell instances, and the BIST circuit area is $300 \times 80 \mu m^2$.

The test on the assembled chips shows that the BIST block works correctly. The CAM BIST is running at 800 MHz and 30 kHz separately. The test results show that the fail_h is pulled up at 67-320, 451 cycles. When the V_{dd} voltage is raised to 1.8 V, all the tests can pass correctly. According to the algorithm, the test fails at Pass3, Pass4, Pass5 and Pass8. The last comparison in Pass5 is correct. Studying the Match signal, it is found that the C_1^0, C_1^1 comparison operations fail. This is the S stuck-at-0 fault or write 1 error. The state of the CAM cells can not be changed to 1. After analysis by the circuit designers, the problem is found: it is because of the error delay timing of BL and WL. In the design, the BL path goes through the sample circuits and the WL path goes through the decoder circuits. BL should be ready before WL is pulled up, and keeps its value until the WL is pulled down. But the foundry gives inexact SPICE model parameters. The circuit designers get a correct result with inexact parameters, and the delay time of BL and WL is perfect in the post-layout simulation as shown in Fig. 4(a). The first line is WL, the second line is BL and the last line is the state of the CAM cell. After the foundry updates the SPICE model parameters, the BL is pulled down before WL, and value 1 can not be written to the CAM cell. The post-layout simulation is incorrect as shown in Fig. 4(b). After raising the voltage, the gap between the BL and WL paths is reduced as the last column shows, and does not influence the writing operation. The value 1 can be written to the CAM core cell correctly. This matches the fault analysis. These inexact parameters mean that the CAM module of the test chip can not work correctly in the chip test. The experiment shows that analysis of the circuit level faults of CAM is effective. It is helpful to analyze the faults. Compared to other BIST algorithms, the proposed algorithms can test more faults.

5. Conclusion and future work

The paper focuses on the write-only CAM structure. It discusses physical fault models and introduces new algorithms based on the physical fault models. Compared to functional fault models, the physical fault models are closer to circuit defects, and fault analyzing at the physical level is more holistic than in the functional fault models. Physical faults which can not be detected by any current functional fault model are also introduced. The probability of these faults is not less than other physical faults. The physical fault models can cover full-scale faults, and are suitable for CAM circuits. Furthermore, the physical fault models can cover delay faults. We propose two algorithms based on physical fault models. The algorithm for a CAM module with N-bit match output signal needs only $2N$+$2L$+4 comparison operations and $5N$ writing operations to cover 100% faults involved in the paper. The algorithm for the HIT-signal-only CAM module uses $2N$+$2L$+5 comparison operations and $8N$ writing operations to cover 100% faults involved in the paper. More physical faults can be detected by the proposed algorithms with the cost of only a few more op-

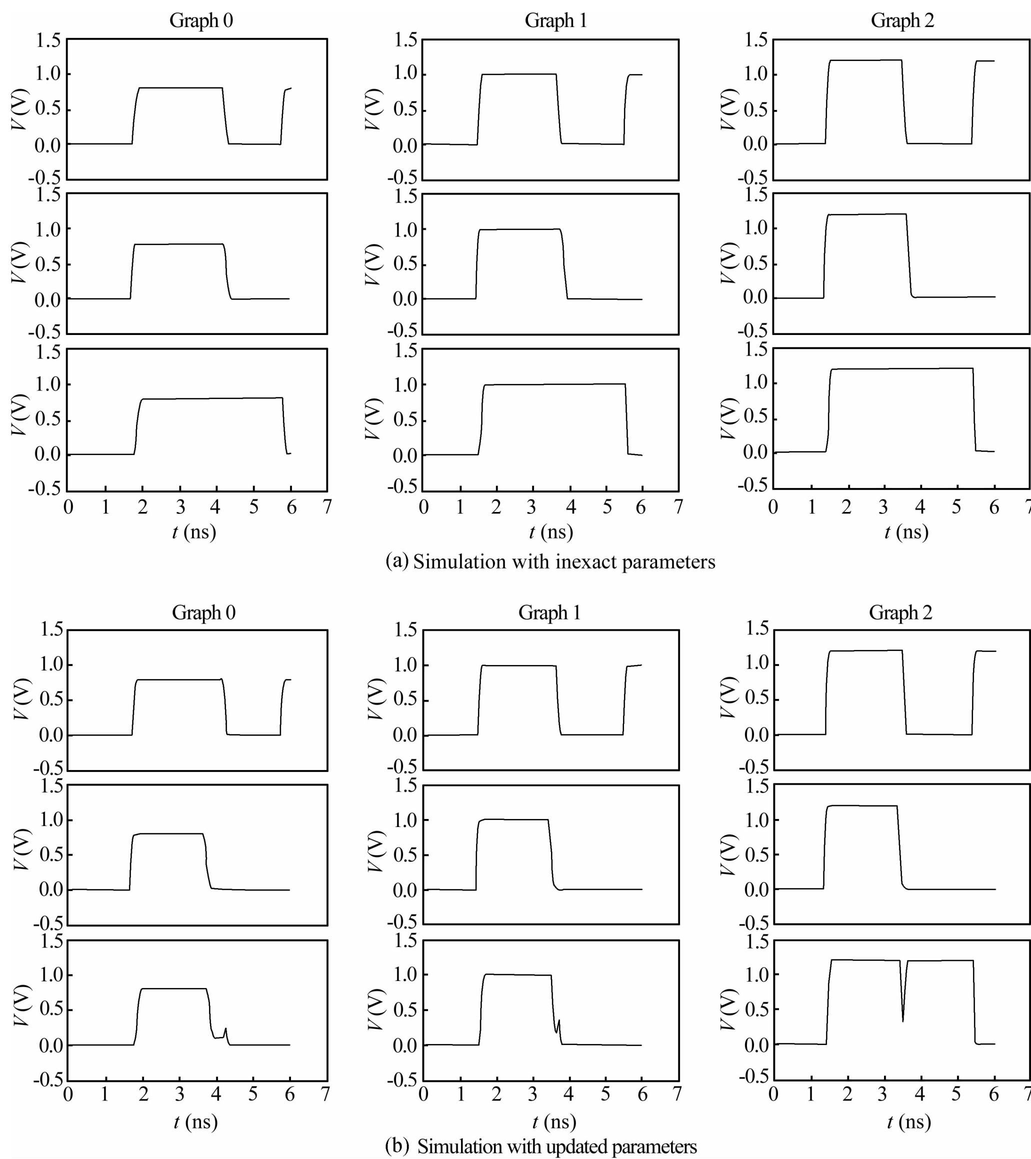

Fig. 4. Post-layout simulation waveform.

erations. Experimenting on a test chip shows that the BIST design using the proposed algorithm works correctly and one design error is found.

However, the fault models discussed in the paper are within the CAM cell. Our future work includes physical fault models that cross CAM cells and their corresponding test algorithms.

References

[1] Zhao J, Irrinki S, Puri M, et al. Testing SRAM based content addressable memories. IEEE Trans Comput, 2000, 49(10): 1054

[2] Li J F, Tzeng R S, Wu C W. Testing and diagnosing embedded content addressable memories. Proc IEEE VLSI Test Symposium, Monterey, CA, 2002: 389

[3] Zhao Xuemei, Ye Yizheng, Chen Cunxu, et al. Built-in self-test algorithm for embedded cache. Journal of Computer-Aided Design & Computer Graphics, 2005, 17(1): 110 (in Chinese)

[4] Ma Lin, Chen Yunji, Su Menghao, et al. Testing content addressable memories using instructions and marchlike algorithms. Proc IEEE International Conference on Electronics, Circuits and Systems, Malta, 2008: 774

[5] Lin K J, Wu C W. Functional testing of content addressable memories. IEEE Int Workshop Memory Technology, Design and Testing, San Jose, 1998: 70

[6] Lin K J, Wu C W. Testing content-addressable memories using functional fault models and March-like algorithms. IEEE Trans Comput Aided Design Integr Circuits Syst, 2000, 19(5): 577

[7] Du X, Reddy S M, Rayhawk J, et al. Testing delay faults in embedded CAMs. Proc Asian Test Symposium, Xi'an, China, 2003: 378

[8] Bhaysar D K. A built-in self-test method for write-only content addressable memories. Proc IEEE VLSI Test Symposium, Palm Springs, CA, 2005: 9