이중 포트 메모리를 위한 고장 진단 알고리즘

Fault Diagnosis Algorithm for Dual Port Memories

朴漢源, 姜成昊
Hanwon Park and Sung-ho Kang
이중 포트 메모리를 위한 고장 진단 알고리듬

(Fault Diagnosis Algorithm for Dual Port Memories)

朴漢源*, 姜成炅*
(Hanwon Park and Sungho Kang)

요 약
현재 다양한 분야에서 이중 포트 메모리의 사용이 증가함에 따라 이중 포트 메모리의 고장을 진단하기 위한 효율적인 고장 진단 알고리듬의 필요성이 증대되고 있다. 따라서 본 논문에서는 이중 포트 메모리에서의 효율적인 고장 진단 알고리듬을 제시하여 이중 포트 메모리에서 발생하는 거의 모든 종류의 고장에 대한 진단을 가능하게 한다. 또한 진단 과정에서 착오를 일으키지 않고 다양한 고장 모델을 구별하며 고장과 관련된 위치를 정확하게 확인하는 것이 가능하다. 새로운 진단 알고리듬을 사용함으로서 이중 포트 메모리에서의 고장 진단 과정은 효과적으로 수행될 수 있으며 이전의 다른 연구들과의 성능 평가를 통해 효율성을 확인할 수 있다.

Abstract
As dual port RAMs are widely used in the various applications, the need for an efficient algorithm to diagnose faults in dual port RAMs is increased. In this paper we propose an efficient algorithm that can diagnose all kinds of faults in dual port RAMs. In addition, the new algorithm can distinguish various fault models and locate the position related to each fault. Using the new algorithm, fault diagnosis for dual port RAMs can be performed efficiently and the performance evaluation with previous approaches proves the efficiency of the new algorithm.

Key words: dual port memory, fault model, fault diagnosis.

I. 서 론
메모리의 용량과 집적도의 급속한 증가로 인해 필요로 점점적으로 메모리 생산 기술 뿐 아니라 테스트 관련 기술의 개발에 대한 필요성이 대두되었다. 특히 메모리 수리 과정을 통해 재생산을 하기 위해서는 테스트를 통해 검출한 고장을 정확하고 고장의 종류와 위치를 정확하게 결정할 수 있는 고장 진단 방법의 개발이 필수적이다. 하지만 지금까지의 연구는 대부분 메모리의 재

* 正會員, 延世大學校 電気電子工学科
(Dep. of Electrical and Electronic Eng., Yonsei Univ.)
※ 본 연구는 2001년도 시스템집적도반도체기술개발 사업의 과제 연구비에 의하여 연구되었음.

수지: 2001년 5월 28일, 수정일: 2002년 2월 7일

스트하기 위한 방법들을 다루어 왔다[1-5]. 또한 현재 다양한 분야에서 널리 사용되고 있는 이중 포트 메모리에 대한 고장 모델과 테스트 방법에 대한 연구는 어느 정도 발표된 바가 있다[6-8], 하지만 이중 포트 메모리에서의 고장 진단 방법에 관한 연구는 전혀 이루어지지 않아 이에 관한 연구의 필요성이 절실한 상황이다. 따라서 본 논문에서는 이중 포트 메모리에서의 효율적인 고장 진단 방법을 제안한다.


따라서 본 논문에서는 이중 포트 메모리에서 일반적으로 사용되는 고장 모델들에 근거하여 테스트에서 고장이 검출될 경우 고장의 종류와 고장과 관련된 포트 셀의 위치를 확인할 수 있는 진단 방법을 제안한다. 이는 각각의 고장 모델들을 위해 새로운 진단 패턴을 제안하고 기존의 테스트 알고리즘과 동일함으로써 가능하게 하였다. 이러한 이중 포트 메모리에 대한 진단 방법에 근거하여 단일 포트 메모리에 대한 진단 방법도 분석한다. 또 일반적인 진단 알고리즘뿐만 아니라 테스트 알고리즘의 수행 결과를 이용한 진단 방법을 재구현하여 테스트 정보의 분석을 이용한 진단 방법의 확보를 가능하게 한다.

본 논문은 다음과 같이 구성되어 있다. II장에서는 진단 과정에서 고려하는 고장 모델들에 대해 설명한다. III장에서는 고려하는 고장 모델들의 진단을 위한 패턴을 분석한다. IV장에서는 이중 포트 메모리의 고장을 고려한 진단 알고리즘을 제안하고 이에 따라 단일 포트 메모리에서의 진단 방법 및 테스트 정보의 분석을 이용한 진단 방법을 제안한다. V장에서는 기존의 제안된 메모리 진단 방법들과 본 논문에서 제안한 진단 방법들을 비교한 성능 평가 비교 및 결과를 보여준다. 그리고 VI장은 결론으로써, 본 논문의 연구 내용을 요약한다.

II. 고장 모델

메모리 테스트를 위해서는 발생할 수 있는 물리적 결함을 모델링하여 고장 모델의 정의하고 테스트를 통해 이를 검출할 수 있어야 한다. 따라서 테스트에서 고장이 검출될 경우 효율적인 고장 진단을 위해서는 고려해야 할 고장 모델을 결정하여야 한다. 이중 포트 메모리는 두 개의 포트를 통해 메모리에 접근하여 데이터를 입력할 수 있는 구조이므로 일반적인 단일 포트 메모리에서의 고장 뿐만 아니라 새로운 고장 모델이 필요하다. 본 연구에서는 이중 포트 메모리에서의 고장 모델의 단일 포트 관련 고장과 이중 포트 관련 고장으로 나누어 다음과 같은 고장 모델들을 고려하기로 한다.

1. 단일 포트 관련 고장(IPF)
   (1) 고착 고장, 천이 고장

고착 고장은 하나 또는 그 이상의 메모리 셀이나 채이 고착되어서 값이 0 혹은 1로 항상 고정되어 있는 상태의 고장을 말한다. 따라서 이러한 고장을 검출하려면 메모리 셀에서 0과 1의 값을 잃는 동작을 수행하여야 한다. 그리고 천이 고장은 고장 고장의 특정한 경우로 할 수 있는데 채이 0에서 1으로 혹은 1에서 0으로의 채이 쓰기 동작을 수행하지 못하고 실패하는 고장은 말한다. 그러므로 이 고장은 검출하기 위해서는 0에서 1로의 채이 쓰기 동작과 1에서 0으로의 채이 쓰기 동작을 수행한 후 각각의 값을 잃는 동작을 수행해야 한다.

(2) 결합고장

두 개 또는 그 이상의 셀 사이에서의 비정상적인 연결이나 인해 발생하는 고장으로 한 셀에서의 동작이
다른 셀에 영향을 받지 않은 구멘을 발생시키는 고장은 편안하다. 하나의 셀에서 편 쓰기 동작을 할 때 결합된 전류의 영향이 강해지므로 1 또는 0로 바뀌는 형태의 고장이다. 4자리의 다른 형태의 고장이 존재할 수 있다. 즉 셀 ij가 편각합성, ij가 결합합성이라는 형태의 고장은 마찬가지로 1 또는 0으로 바뀌게 되는 구멍을 의미한다. 따라서 이 고장은 고장 고장이나 천이 고장과 달리 테스트에서 고장이 검출되는 셀 외에도 고장과 관련된 셀이 존재하게 된다. 따라서 편안한 과정에서는 고장이 검출되는 셀 뿐만 아니라 고장과 관련된 셀 모두를 확인하는 과정이 필요하다.

(3) NPSF

셀의 내용이 메모리 배열 내의 다른 셀들의 내용에 따라 영향을 받는 것으로 파란한 네이트 키와 편각합성 결합 휴지(capacitively coupled noise)등에 의해서 발생한다. 이러한 결합 영향이 고장의 원인들은 메모리가 고장의 영향에 따라 셀의 간격이 줄어다 더욱 반복적으로 발생하게 되어 응용상 메모리의 큰 문제가 되고 있다. 이러한 NPSF는 기존 결합슈어의 이웃한 여러 셀들의 특정 때단이 천이 쓰기 동작이 기존형에 영향을 가져 오던 모습의 형태로 나타난다. 이러한 고장의 경우 여러 가지로 일반화될 수 있는데 각각 이웃셀들의 특정 때단에서의 고장 고장 천이 고장, 결합 고장과 유사한 고장의 특성을 갖는다.

(4) 어드레스 디코더 고장

어드레스 디코더 고장은 주소 디코더에서 발생하는 고장으로 나타낸다. 이들 고장은 크게 다음과 같이 분류될 수 있다.

1. 고장 1: 특정한 주소를 가지고 어떤 메모리 셀도 접근할 수 없는 고장
2. 고장 2: 특정한 셀이 접근 불가능한 고장
3. 고장 3: 특정한 주소가 여러 개의 셀을 동시에 접근하는 고장
4. 고장 4: 특정한 셀이 여러 개의 주소로 접근이 가능한 고장

2. 이중 포트 관련 고장(2A2)
(1) Inter port fault
이중 포트 메모리는 두 개의 포트 A, B를 통해 독립적으로 메모리 셀 어레이에 접근하여 데이터를 동시에 입력할 수 있도록 하기 때문에 가능하다. 이를 위해 포트 A와 포트 B를 위한 워드 라인이 별도로 존재하게 된다. 따라서 이중 포트 메모리의 구조상 서로 다른 포트 A 와 B의 워드 라인단지의 단락이나 카페턴스 결합 등 의 고장이 존재할 수 있는데, 이를 inter-port fault라 한다[14]. 이는 다음과 같은 형태로 바뀌(mapping)될 수 있다.

- 같은 행 내의 OR-type(AND-type) 워드 라인 단락
- 다른 행 내의 OR-type(AND-type) 워드 라인 단락
- 이러한 서로 다른 포트들의 워드 라인단지에 존재할 수 있는 단락은 동일한 행 내에서의 경우 및 서로 인접한 행들의 경우를 포함하여 서로 다른 행들간의 단락은 여기에서 고려하지 않는다. 동일한 행 내에서의 고장과 이웃한 행들간의 고장을 나타내면 그림 1과 같다.

![그림 1. Inter-port fault의 예](a) 동일한 행에서의 고장  (b) 이웃한 행들간의 고장

Fig. 1. Example of inter port fault.

(2) AP2
이중 포트 메모리에서는 서로 다른 포트들을 통해 메모리 셀에 접근하므로 각각의 포트에 대한 어드레스 디코더가 별도로 존재하게 독립적으로 다른 어드레스 디코더를 통해 다른 어드레스의 메모리 셀에 접근하여 데이터를 읽고 쓰는 동작을 수행할 수 있게 된다. 따라서 기존의 단일 포트 메모리의 어드레스 디코더 고장

(214)
외의 다른 고장 모델이 필요하여 아중 포트 모드리에
서의 어드레스 디코더 고장은 하나의 포트와 관련된
어드레스 디코더 고장 APB1과 두 포트의 어드레스 디코
더 관련 고장 APF로 나눌 수 있다. 여기서 APB1은 하
나의 포트의 어드레스 디코더에서의 고장은 나타낸다.
따라서 이러한 형태의 고장의 경우 단일 포트 메모리
에서의 고장과 가고 테스트 방법 또한 동일하다. 반면
APF는 두 포트의 어드레스 디코더의 타인들과의 단자
이나 간섭에 의해 발생하는 고장으로 일반적인 단일
포트 메모리에서의 고장과 차이가 존재한다[15].

III. 진단 패턴

II장에서는 아중 포트 메모리에서의 고장 진단을 위
해 고려한 고장 모델들에 대해 살펴 보았다. 이 장에서
는 고려하는 고장들에 대해 진단 과정 수행을 위해 필요
한 패턴들을 분석한다. 우선 이 고장이 하나의 세제
단 국립된 종류의 단일 선택 고장인지 여러 선택 판단이
있는 다중 선택 고장인지 여부를 규정할 수 있어야 하며
또 일반적인 단일 포트 관련 고장인지 아중 포트 관련
고장인지도 구분되어 아중 포트 메모리에서 발생하는
고장에 대해 그 종류와 위치에 대한 분석이 명확하게
어길 수 있어야 한다.

I. 단일 포트 관련 고장 진단 (PTN-1P)

(1) PTN-ST : 고차 고장, 천이 고장은 진단 외한 패턴

\[ W_0(0), R_0(0), W_1(1), R_1(1), W_2(0), R_2(0) \]

여기서 \( W_i \)는 세 (i, j)에 쓰기 동작을 수행하는 것을
 의미하며 \( R_i \)는 우선 세 (i, j)의 값을 읽는 동작을 의미
한다. 고장 (i, j)에 대화 없이 0을 읽고 읽는 동작
을 수행한다. 이러한 동작들을 수행함으로서 고장이
진단 고장인지 고장이나 천이 고장과 같은 단일원
고장인지지를 알 수 있다. 우선 여러 동작들을 수행할 때
고장이 검출되지 않는다면 고장이나 천이 고장과 같은
단일원 고장이 아니라고 진단 고장이나 NPSF와
같은 다중 선택 고장에 대한 진단 과정이 필요하다. 이와
는 단일 진단 이외의 \( R_0(0) \)이나 \( W_1(1) \)이 실패하여 고
장이 검출되지나 고장 고장, 천이 고장과 같은 단일 선택
고장으로 진단된다. 하지만 이 경우 고장이 발생한 손
간 세 (i, j)의 이웃셀들의 특정 패턴들 때문에 위의 동
작들이 실패하는 경우가 발생할 수도 있음을 못 안
한다. 따라서 세 (i, j)의 이웃셀들의 값을 바꾸어 준 후 위의 동작들을 다
시 수행함으로서 고장 고장, 천이 고장과 같은 단일원
고장인지 NPSF인지가 결정되므로 진단 과정은 완료된다.

(2) PTN-C : 결합 고장 위한 패턴

(1)의 과정에서 고장이 검출되지 않은 경우들에 대해
아래와 같은 패턴들을 가해 주는 진단 과정을 수행한
다. 이 과정에서는 메모리에서 존재하는 고장이 결합
고장인 경우 고장의 원인이 되는 결합원셀의 위치를
확인한다.

\[ W_0(0), W_1(1), R_0(0), W_0(0), R_0(0) \]

\[ W_1(1), W_0(0), R_1(0), W_1(1), R_1(0) \]

여기서 \( W_i \)는 세 (i, j)에 쓰기 동작을 수행하는 것을
 의미하며 \( W_i \)은 우선 세 (i, j)를 제외한 모든 세들에 쓰기
동작을 수행하는 것을 의미한다. 우선 세 (i, j)에 0이나
1의 값을 쓰고 다른 세들에 0을 쓰고 후 다시 1을 쓰는
동작을 수행한 후 세 (i, j)의 값을 읽어서 확인한다.
그리고 다시 0을 쓰고 세 (i, j)의 값을 읽어서 확인한
다. 이러한 동작을 어드레스를 변환시키면서 반복적으
로 수행하는 중에 고장이 검출되지 않는다면 결합 고
장이 아니므로 NPSF의 검출에 대한 진단을 수행하여
야 한다. 만약 위의 동작을 수행하는 중 고장이 검출된
다만 결합원셀 뿐 아니라 결합원셀의 위치도 확인된
다고 생각할 수 있다. 하지만 이 경우 결합원셀이 기존설
의 이웃셀중에 하나일 경우에는 고장이 발생한 순간
세 (i, j)의 이웃셀들의 특정한 값을 가지 상태에서만
위의 동작이 실시되는 것 가능성이 있으므로 결합
고장이 아니라 active NPSF일 수도 있다. 따라서 1.1의
경우와 마찬가지로 이웃설들의 패턴들을 바꾸어 준 후
위의 패턴들을 다시 수행함으로서 결합 고장인지 active
NPSF인지를 결정하고 진단 과정을 완료하게 된다.

(3) PTN-N : NPSF를 위한 패턴

(1)과 (2)의 과정을 통해 검출되지 않는 고장일 경우
아래와 같이 좀 더 복잡한 형태의 고장인 NPSF에 대
한 진단 과정을 수행하게 된다. 여기서 테스트에서는
고장이 검출된 세 (i, j)는 기존설이 되고 상하 좌우의 인
접한 세들이 이웃설이 된다. 이러한 NPSF의 검출을 위
해 필요한 패턴들은 다음과 같다.

\( \text{Active NPSF} \)

\[ W_0(D) \]

\[ \text{APPLY(ANPSF)} \]

\[ R_0(D) \]
여기서 D는 데이터 0 혹은 1을 의미하며 APPLY (ANPSF)는 active NPSF의 검출을 위해 필요한 콘텐
들을 이용함에 가하는 것을 의미한다. 필요한 콘텐은
이웃점들에 하나의 천이 쓰기 동작을 포함한 가능한
모든 상태를 의미한다. 따라서 이 경우에는 기준에 0
과 1의 값을 쓰고 이웃점들에서 하나의 천이 동작을
포함한 모든 콘텐들에 대해 각각 기준에 대해 0과 1
을 읽는 동작을 수행하여 검출 가능하다.

(2) Passive NPSF

APPLY(PNPSF)

\[ W_d(0), R_d(0), W_d(1), R_d(1), W_d(0), R_d(0) \]

passive NPSF: 고장의 경우는 active NPSF의 경우와
날리 이웃점들 처럼 기운 동작을 고려할 필요 없이 가능한 모든 상태에 대해 각각 빈 동작을 기
준에 수행하고 읽기 동작을 수행하여 값을 확인하면
된다. 따라서 APPLY(PNPSF)는 이웃점들에 천이 동
작을 고려할 필요없이 가능한 모든 상태를 적용하면
된다.

(3) Static NPSF

\[ W_d(D) \]

APPLY(SNPSF)

\[ R_d(D) \]

이 경우도 passive NPSF와 마찬가지로 이웃점들에
가능한 모든 상태를 가하는 콘텐이 필요하다. 기준에
0과 1의 값을 쓰고 이웃점들에 모든 콘텐들에 대해 각
각 기준에 대해 0과 1을 읽는 동작을 수행하여 검출
가능하다.

(4) PTN-A: 어드레스 디코더 고장의 경우 폐전

(1)에서 (3)까지의 과정을 통해서 고장의 종류와 위치가 진단되지 않는다면 어드레스 디코더 고장일 가능
성이다 이를 확인하기 위해 어드레스 디코더 고장
에 대한 테스트를 수행한다. 이 고장은 메모리에서 어
드레스 디코더 부분에서 관련된 고장으로 대부분의 경
우 메모리 셀에서의 고장으로 매핑(mapping)된다. 다음
과 같은 조건을 만족하는 테스트를 수행한 때 어드레
스 디코더 고장은 검출할 수 있다.

\[ R(D), \ldots ,W(D)) \]

\[ R(D), \ldots ,W(D)) \]

앞의 진단 과정에서는 위와 같은 콘텐들을 모두
인가하지 않았기 때문에 모든 어드레스 디코더 고장을
검출할 수는 없다. 그리고 모든 어드레스 디코더 고장
을 진단하기는 어려우므로, 앞에서 말한 모든 고장들에
대한 진단 폐쇄으로도 검출이 되지 않을 경우에 한해
위에서 말한 조건을 만족하는 가장 짧은 길이의
MATS의 콘텐들들을\[ W_d(0), R_d(0) \]를 이용하여 어드레스 디코더 고장
에 대한 진단을 수행한다.

2. 이중 포트 관련 고장 진단 (PTN-2PF)

(1) PTN-IP: Inter port fault를 위한 폐전

이중 포트 메모리의 서로 다른 포트들의 위드 라인
동작의 단락인 inter port fault는 동일한 내역의 고장과
이웃한 행들간의 고장으로, 나뉘어지게 되며 이 고장들
은 and type 과 or type이 존재하게 된다. 진단 과정에
서는 먼저 동일한 행에서의 고장에 대한 진단 폐전
을 먼저 하고 이를 통해할 경우에는 이웃한 행들간의
고장에 대한 폐전을 가하여서 inter port fault에 대해
고장의 종류와 위치를 진단하게 된다.

우선 동일한 행 내에서 inter port fault의 진단을 위
한 폐전은 다음 표 1과 같다.

표 1에서 A와 B는 이중 포트 메모리의 서로 다
른 포트를 의미하며, (i, j-1)이나 (i, j-2)는 메모리에서
의 어드레스를 의미한다. 따라서 \( W_{A}(0) \)는 포트 A를
통해 어드레스 (i, j-1)의 셀에 0을 쓰는 동작의 수행을
이미지하여 \( R_{B}(0) \)는 포트 A를 통해 어드레스 (i, j-1)
인 셀로부터 0의 값을 읽는 동작을 수행하는 것을 의
미한다. 동일한 행 내에서의 고장의 경우 OR type을
에로 들어 진단 폐전을 설명하면 다음과 같다.

| OR type | \( W_A(0)_{i,j-1}, W_A(1)_{i,j-2}, R_B(0)_{i,j-1}, R_B(0)_{i,j-2} \) |
| AND type | \( W_A(1)_{i,j-1}, W_A(0)_{i,j-2}, R_B(1)_{i,j-1}, R_B(0)_{i,j-2} \) |

우선 A 포트를 통해 어드레스 (i, j-1)의 셀에 0을 쓰
고 (i, j-2)의 셀에 1을 씌 후 A 포트를 통해 셀 (i,
j-1)의 값을 읽는 동작을 수행하고 동시에 B 포트를 통
해서는 셀 (i, j-2)의 값을 읽는 동작을 수행한다. 고장
이 존재하지 않을 때는 \( R_B(0)_{i,j-2} \)의 수행시 짧았으며

表 1. Diagnosis patterns for inter port fault at same row.
기대했던 0의 값을 얻게 된다. 하지만 만약 셀 (i, j-2)에 포트 A의 위드 라인과 포트 B의 위드 라인 사이에 단락 고장이 존재할 경우에는 B포트를 통해 어드레스 (i, j-2)의 값을 읽는 Rb(x)i-2 동작을 수행할 때 단락 고장 때문에 B 포트의 위드 라인 뿐만 아니라 A 포트의 위드 라인까지 선택되게 된다. 따라서 Rb(x)i-2와 동시에 수행하는 Rb(x)i-1 동작을 수행할 때 고장이 존재하지 않을 경우 정상적으로 0의 값을 얻게 되지만 고장이 존재하는 경우는 A 포트 (i, j-1)의 위드 라인 뿐 아니라 (i, j-2)에 해당하는 위드 라인도 선택되므로 Rb(x)i-1의 수행 결과가 0이 아닌 1이 되게 된다. 따라서 어드레스 (i, j-2)에 해당하는 행 내에서 OR type의 inter port fault는 진단이 이루어지지게 된다. AND type 고장의 경우도 이와 유사하게 표 1과 같은 패턴으로 진단 과정이 수행될 수 있다.

이러한 동일한 행 내에서의 고장의 경우와 달리 이웃한 행들간의 고장의 경우는 고장과 관련된 위치가 하나의 셀이나 행과 관련된 것이 아닌 두 개의 이웃한 행과 관련이 있으므로 이를 위한 별도의 진단 패턴이 필요하게 된다. 이중의 두 행들간의 고장에 대한 진단을 위한 패턴은 다음 표 2와 같다.

<table>
<thead>
<tr>
<th>OR type</th>
<th>W_A(0)ij, W_A(1)i-1, W_A(1)ij-1, \left( \frac{R_b(0)ij-1}{R_b(x)ij-2} \right)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND type</td>
<td>W_A(1)ij, W_A(0)i-1, W_A(0)ij-1, \left( \frac{R_A(1)ij}{R_b(x)ij-2} \right)</td>
</tr>
</tbody>
</table>

표 2에서 알 수 있듯이 이중에서의 inter port fault에 대한 진단 패턴은 동일한 행 내에서의 고장에 대한 패턴과는 차이가 있다. OR type을 예로 들어 설명하면 다음과 같다. 우선 A 포트를 통해 셀 (i, j)에 0을 쓰고 셀 (i, j-1)에 1을 쓴다. 그리고 셀 (i, j-2)에도 포트 A를 통해 1을 쓰는 동작을 수행한다. 그리고 A 포트를 통해 셀에서 0을 읽는 Rb(x)ij 동작과 B 포트를 통해 (i, j-2) 셀을 읽는 Rb(x)ij-2 동작을 동시에 수행하게 된다. 고장이 존재하지 않는 경우에는 A 포트를 통해 읽기 동작에서 정상적으로 기대했던 0의 값을 얻게 된다. 그러나 만약 셀 (i, j-1)과 셀 (i, j-2)의 행들 사이에 단락이 존재한다면 Rb(x)ij-2 동작을 수행할 때 B 포트의 셀 (i, j-2)의 행의 위드 라인이 선택될 때 j-1과 j-2 사이에 존재하는 단락 때문에 A 포트 j-1의 위드 라인도 선택되게 된다. 따라서 Rb(x)ij-2 동작을 Rb(x)ij-1 동작과 동시에 수행할 때 고장이 존재하지 않는 경우는 정상적으로 0의 값을 얻게 되지만 고장이 존재하는 경우에는 j-1에 해당하는 A 포트의 위드 라인도 선택되게 되므로 A 포트를 통해 0의 값을 읽는 동작을 실패하고 1의 값을 얻게 되어 서로 다른 행들간의 inter port fault에 대한 진단이 이루어지게 된다. AND type 고장의 경우도 이와 유사하게 수행될 수 있다. 동일한 행에서 서로 다른 행에서의 inter port fault에 대한 각각의 진단 패턴의 수행의 예는 그림 2와 같다.

(a) 동일한 행 내에서의 (b) 이웃한 행들간의 고장 고장

그림 2. Inter-port fault에 대한 진단 패턴 수행의 예

Fig. 2. Example of diagnosis procedure for the inter port fault.

(2) PTN-AF2: 이중 포트 관련 어드레스 다이코더 고장에 대한 패턴

이중 포트 관련된 어드레스 다이코더 고장인 AF2는 두 개의 어드레스 다이코더에서 일어날 수 있는 단락이 나 간섭에 의한 것으로 다양한 종류의 결합이 존재할 수 있지만 그 고장들 중 일부는 검출을 위해 두 개의 포트를 사용할 필요가 없고 하나의 포트만을 통해 일반적인 AFI에 대한 패턴들을 이용하여 가능하다는 특징을 가진다. 따라서 이중 포트 관련 어드레스 고장인 AF2에서 AFI에 대한 패턴으로 검출 가능한 고장을
표 3. AF2의 간략화된 형태
Table 3. Simplified form of AF2.

<table>
<thead>
<tr>
<th>Fault</th>
<th>Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>If ( A_a(x) : A_b(y) ) then ( C_a(x) : C_b(y) )</td>
<td>Fault E</td>
</tr>
<tr>
<td>If ( A_a(x) : A_b(z) ) ( z \neq y ) then ( C_a(x) : C_b(z) ) ( C_b(y) )</td>
<td>Fault F</td>
</tr>
<tr>
<td>If ( A_a(x) : A_b(y) ) then ( C_a(x) : C_b(\phi) )</td>
<td>Fault G</td>
</tr>
</tbody>
</table>

그림 3. 이중 포트 메모리에서 AF2를 위한 패턴
Fig. 3. Patterns for AF2 of dual port RAMs.

IV. 이중 포트 메모리의 고장 진단 알고리듬

1. 이중 포트 메모리의 진단 과정

우선 이중 포트 메모리를 위한 전체 진단 과정을 위해서는 단일 포트 관련 고장 진단 과정인 PTN-IPT를 각각의 포트를 통해 수행한다. 그리고 이중 포트 관련 고장 진단 과정인 PTN-2PF를 수행한다. 이중 포트 메모리에서의 전체적인 진단 방법은 다음과 같다.

여기에 IIP는 단일 포트와 관련된 고장을 의미하고 2PF는 이중 포트와 관련된 고장을 의미한다. IIP에는 고장 고장, 안이 고장, 결함 고장, NSF, 이드레스 디코더 고장 등이 속하고, 2PF에는 inter-port fault, 이드레스 디코더 고장이 속한다. 또, PTN - IIP는 단일 포트 관련 고장인 IIP에 대한 진단 패턴을 의미하며...
PTN - 2PF는 2PF에 대한 진단 패턴을 의미한다. 따라서 A 단계를 통해 우선 이중 포트 메모리에서 단일 포트 관련 고장들은 모두 진단이 가능하다. 그리고 단일 포트 관련 고장에 대한 전단 과정의 A 단계를 통해 서로 진단이 이루어지지 않을 경우에는 이중 포트 관련 고장에 대한 진단 과정인 B 단계를 통해 진단 과정을 수행한다. 그러므로 그림 4와 같은 과정을 수행함으로서 이중 포트 메모리에서의 모든 고장들에 대한 진단이 이루어지게 된다.

2. 이중 포트 메모리를 위한 고장 진단 알고리듬
(1) 단일 포트 관련 고장을 위한 알고리듬.
앞에서 이중 포트 메모리에서 발생하는 모든 고장에 대한 패턴들과 개략적인 진단 과정을 알아보았다. 여기에서는 그림의 이중 포트 메모리 진단 과정에서 A단계를 위한 단일 포트 관련 고장들에 대한 상세한 알고리듬을 제안한다. 단일 포트 관련 고장들에 대한 알고리들은 다음 그림 5와 같다.

그림 5에서는 고차 고장, 전이 고장, 결합 고장, NPSF, 어드레스 디코더 고장 등이 단일 포트 관련 고장들에 대한 진단 패턴을 사용하여 고장 진단을 수행하게 된다. 이러한 과정은 각각의 포트들을 통해 수행함으로서 이중 포트 메모리에서 하나의 포트와 관련된 고장들에 대한 진단이 이루어지게 된다.

(2) 이중 포트 관련 고장에 대한 알고리듬.
이중 포트 메모리에서는 그림 5와 같은 하나의 포트 와 관련이 있는 고장들에 대한 알고리즘의 수행을 통해서 진단이 이루어지지 않은 과정이 존재한다. 이는 두 개의 포트와 관련된 고장으로 본 논문에서는 II장에서 설명한 바와 같이 inter port fault와 AF2를 고려한다. 이러한 이중 포트 관련 고장들에 대한 알고리즘은 다음 그림 6과 같다.

그림 6. 이중 포트 관련 고장을 위한 알고리듬
Fig. 6. Fault diagnosis algorithm for 2PF.

그림 6에서는 이중 포트 관련 고장을 위한 진단 패턴인 PTN-IP와 PTN-AF2를 사용하여 고장 진단을 수행한다. 이러한 과정을 수행함으로서 이중 포트 메모리에서 두 개의 포트와 관련된 고장들에 대한 진단이 이루어지게 된다.
(3) 이중 포트 메모리의 고장 진단 알고리듬

앞의 2.1와 2.2에서는 하나의 포트만 관리된 고장들
과 두 개의 포트와 관련된 고장들의 진단을 위한 알고리듬을 제안하였다. 이를 바탕으로 이중 포트 메모리에
서 고장 진단을 위한 알고리듬은 다음 그림 7과 같다.

그림 7의 진단 과정은 체계적으로 설명하면 다음과 같다.
우선 PTN-ST의 패턴은 단일 셋 고장인지 여부를 판
단하기 위해 사용된다. PTN-ST의 과정을 통과한 경우
단일 셋 고장이 아닌 다중 셋 고장으로 예측되어
PTN-C의 과정을 수행하게 되며 실패할 경우에는 단일
셋 고장으로 예상되지만 NFSF일 가능성이 존재하므로
위에서 설명한 것처럼 이웃של의 패턴을 바꾸어 주는
PTN-ST/N을 통해 고장이 친이 고장인지 NFSF인
지 진단을 완료하게 된다. 다음으로 PTN-ST를 통과한
경우에 대해 PTN-C의 패턴을 인가하게 되는데 이를
통해 결합 고장을 발생하게 하는 결합원수를 찾을 수
있다. PTN-C의 과정에서 실패하는 경우 역시 결합 고
장이 NFSF일 가능성이 있으므로 이웃של의 패턴을
바꾸어 주는 PTN-C/N를 통해 결합 고장인지 NFSF인
지 진단하는 과정을 완료하게 된다. 그리고 PTN-C
의 과정을 통과할 경우에는 NFSF에 대한 테스트인
PTN-N의 과정을 수행하게 된다. 여기서 실패할 경우
에는 NFSF로 진단 과정이 완료되며 통과할 경우에는
이두레스 디코더 고장에 대한 테스트인 PTN-A의 과
정을 수행하게 된다. 진단 과정의 마지막 과정인
PTN-A에서 실패할 경우에는 이두레스 디코더 고장으로
진단이 완료된다. 하지만 만약 이 과정에서도 통과하게
된다면 이중 포트 관련 고장들로 예측될 수 있으므로
PTN-IP와 PTN-AF의 과정을 통해 이중 포트 관련
고장에 대한 진단을 수행하게 된다. 그림 7의 고장 진
단 과정에서 PTN-ST, PTN-C, PTN-N, PTN-A는 단
일 포트 관련 고장들에 대한 진단 패턴들로 PTN-IPF
를 의미한다. III장에서 보여준 바와 같이 이러한 패턴
들을 이용한 과정들을 각각의 포트를 통해 수행함으로
서 단일 포트 고장들에 진단 과정이 완료된다. 하
지만 메모리로서의 고장이 단일 포트 관련 고장이 아
닌 두 포트와 관련 고장이라면 이러한 과정들로만
진단 과정이 완료되지 않는다. 따라서 이중 포트 관련
고장들에 대한 진단 패턴은 PTN-IP와 PTN-AF를 이
용한 과정을 통해 고장의 진단 과정을 수행하게 된다.
우선 PTN-IP를 이용하여 동일한 행에서 혹은 이웃한
행에서 단락이 존재할 경우 그러한 고장의 종류 및 위
치에 대한 진단이 이루어지게 된다. 만약 이 과정에서
도 진단이 이루어지지 않을 경우에는 두 포트의 이드
레스 디코더와 관련된 PTN-AF2를 이용한 과정을 수
행하여 최종적으로 진단 과정을 완료하게 된다. 따라서
이러한 과정들을 통해 이중 포트 메모리에서 고장이
발생할 경우 대부분의 고장에 대해 그 종류와 고장이
검출되는 셋 뿐만 아니라 고장과 관련된 셋 등의 고장
의 위치까지 진단하는 것이 가능하다.

그림 7. 이중 포트 메모리 고장 진단 알고리듬
Fig. 7. Fault diagnosis algorithm for dual port
RAMs.
정의 2)과 같은 진단 알고리즘을 통해 검출되는 고장의 종류를 결정하고 고장과 관련된 모든 세부를 확인할 수 있음을 알려준다. 이 점에서는 테스트 방법에 대한 정보의 분석을 이용한 고장 진단 방법을 알아본다. 진단 알고리즘은 수행하는 과정이나 각 세부에 대한 결과값을 얻을 수 있는 경우를 말한다. 이러한 경우에는 메모리를 테스트하는 알고리즘에 따른 분석을 바탕으로 알고리즘 수행 시 각 세부에 대한 결과를 진단 과정에 이용함으로서 고장의 예측 가능성을 증가시킨다.

메모리 테스트를 위한 알고리즘의 각각의 응기 동작에서 고장이 없는 경우와 고장이 존재하는 경우로, 각 고장이 어떠한 종류의 고장 모델인지에 따라 알고리즘에서 응기 동작을 수행하였을 때 나타나는 결과가 달라질 수 있다. 따라서 고려하는 각각의 고장 모델에 따라 테스트 알고리즘의 응기 동작에서 나타나는 결과에 따라 분석보다는 테스트 알고리즘을 메모리에 기록한 결과 나타나는 결과를 판단하는 것으로서 메모리에 있어서 고장에 대한 예측 가능성이 높이지 않지만 진단 과정 또한 일반적인 진단 과정과 다른 형태로 변화될 수 있다. 이러한 경우에는 응기 동작에서 고장이 검출되어 테스트 알고리즘에 중지되는 시점의 마치 요소(march element)에서 최초로 발생하여 검출될 수 있는 고장에 대한 분석이 필요하다.

기여지는 메모리의 테스트 알고리즘으로 임의의 마치 요소를 가지는 알고리즘에 사용된 경우 고장이 발생했을 때 진단하는 방법에 대해 알아본다. 우선 이론에서는 진단 과정에서 고려하는 각각의 고장 모델과 이를 검출하는 마치 요소들에 대한 분석이 필요하다. 그리고 테스트에서 고장이 검출된 부분의 마치 요소에 따라 진단 과정을 수행해야 한다. 이러한 테스트 정보를 이용한 고장 진단 방법은 다음 그림 8과 같다.

그림 8의 테스트 정보의 분석을 이용한 고장 진단 방법을 통해서 진단 과정에서 고려하는 각각의 고장 모델을 테스트 실 시 검출할 수 있는 포인트는 다음과 같이 표 4와 같이 나타날 수 있다. 표 4에서는 진단 과정에서 고려하는 고장들과 이를 테스트 알고리즘에서 검출하는 테스트 변환을 나타낸다. 그러므로 마치 형태의 테스트 알고리즘을 통해 고장을 검출할 경우 고장을 검출한 응기 동작이 포함된 마치 요소들 중 표 4에 포함되는 것들에 따른 고장으로 예측될 수 있고 그에 따라 진단 과정을 수행하기 위해 각 고장에 대한 변형된 진단을 위한 변환은 표 5와 같이 나타날 수 있다. 따라서 마치 알고리즘을 이용한 테스트를 수행하는 경우 고장이 검출되면 고장을 검출하는 마치 요소에 따라
표 5. 고장 모델들에 따른 변형된 진단 패턴들

<table>
<thead>
<tr>
<th>고장 모델</th>
<th>변형된 진단 패턴</th>
</tr>
</thead>
<tbody>
<tr>
<td>SA0</td>
<td>$W_d(1), R_d(1)$</td>
</tr>
<tr>
<td>SA1</td>
<td>$W_d(0), R_d(0)$</td>
</tr>
<tr>
<td>TF†</td>
<td>$W_d(0), R_d(0), W_d(1), R_d(1)$</td>
</tr>
<tr>
<td>TF†</td>
<td>$W_d(1), R_d(1), W_d(0), R_d(0)$</td>
</tr>
</tbody>
</table>

따라서 테스트 과정이나 결과에 대한 정보를 이용한 고장 진단 과정은 그림 5와 7의 진단 알고리듬을 수행함에 있어 각각의 일반적인 진단 패턴을 대신 표 4와 표 5에 의해 그림 8에서 나타내는 바와 같이 예측되는 고장 모델들에 따른 변형된 진단 패턴들을 사용해서 좀 더 간략화될 수 있으며 진단 과정에 소요되는 시간을 줄일 수 있다.

V. 성능 평가 비교 및 분석


표 6. 다양한 진단 방법들의 비교  
Table 6. Comparison of various diagnosis methods.

<table>
<thead>
<tr>
<th>방법들</th>
<th>[12]</th>
<th>[13]</th>
<th>제안된 방법</th>
</tr>
</thead>
<tbody>
<tr>
<td>고장 모델</td>
<td>고장, 고장, 천이 고장, 결합 고장</td>
<td>고장, 고장, 천이 고장, 어드레스 디코더 고장, 결합 고장</td>
<td>고장, 고장, 천이 고장, 어드레스 디코더 고장, 결합 고장</td>
</tr>
<tr>
<td>진단 범위</td>
<td>고장 관련 셀 불가능</td>
<td>고장 관련 셀 가능</td>
<td>고장 관련 셀 가능</td>
</tr>
<tr>
<td>제한 요소</td>
<td>고장 관련 셀에 대한 진단 불가능</td>
<td>초기 마치 테스트의 필요</td>
<td>고장 관련 모든 셀에 대한 진단으로 인한 소요 시간 증가 (O(N^2))</td>
</tr>
<tr>
<td>진단 대상 메모리</td>
<td>단일 포트 메모리</td>
<td>단일 포트 메모리</td>
<td>단일 포트 메모리</td>
</tr>
</tbody>
</table>

에서의 효과적인 고장 진단을 가능하게 한다. 본 연구에서 제안한 메모리 진단 방법과 기존의 제안된 진단 방법들의 종합적인 성능 비교 평가 결과는 표 6과 같다.

VI. 결론

메모리 테스트를 통해 고장이 검출될 경우 고장 셀을 이론의 센로 고체화 하는 수리 과정을 통해 수율을 향상시키는 것이 메모리 생산에서 중요한 요소이다. 그러므로 내장형 고용량 메모리를 위한 내장된 자체 수리 기법 (Built-in self repair : BISR)의 개발을 위한 기반 기술을 확보하고 메모리의 수리를 통해 수율을 향상시키기 위해서는 메모리 테스트와 더불어 검출된 고장의 종류와 위치를 정확하게 결정하기 위한 진단 방법의 연구가 필요하다.

그러나 기존의 진단 방법들은 단일 포트 메모리의 경우에 국한되어 있어 이중 포트 메모리에서 발생할 수 있는 고장들은 고려하지 못하여 고장의 종류 및 위치를 효과적으로 진단 할 수 없다. 따라서 본 연구에서는 이중 포트 메모리에서 존재할 수 있는 다양한 고장 모델을 고려하는 효율적인 고장 진단 방법을 제안하였다.

본 연구에서는 이중 포트 메모리에서 발생 가능한 대부분의 고장 모델들을 고려하고 이에 근거하여 각각의 고장들에 대한 진단을 위한 패턴들을 분석하였다. 또한 바탕으로 테스트에서 고장이 검출될 경우 고장의 종류를 구별하는 것을 가능하게 하는 진단 방법을 제안하고 단지 테스트에서 고장이 검출되는 셀 뿐만 아니라 고장과 관련이 있을 수 있는 셀들을 포함하여 고장과 관련된 모든 셀들을 확인함으로써 이중 포트 메모리에서의 고장의 종류 및 위치의 분석 방식을 확보하였다. 또한 단일 포트 메모리를 위한 고장 진단 방법과 테스트 정지의 분석을 이용한 고장 진단 방법도 제시하였다. 따라서 제안된 고장 진단 알고리듬을 통해 이중 포트 메모리 및 단일 포트 메모리에서 발생하는 거의 모든 종류의 고장들에 대한 진단이 가능함을 알 수 있다. 이를 통해 이중 포트 메모리에서 고장의 발생 특성을 분석 방법을 확보하는 것을 가능하게 하고 고장이 검출된 메모리의 효율적인 재생산을 위한 진단 과정을 가능하게 하였다. 그리고 제안된 고장 진단 방법과 이전의 다른 연구들과의 성능 평가는 본 논문의 이중 포트 메모리에 대한 고장 진단 방법의 효율성을 보여준다.

참고문헌


저자 소개

朴 滿源(正會員)
1976년 4월 18일 생, 1999년 : 연대 전기공학과 졸업. 1999년~현재 연대 전기전자공학과 석사과정

姜 成 磊(正會員)
1963년 4월 13일 생, 1986년 2월 : 서울대 공대 제어기계공학과 졸업. 1988년 5월 : The University of Texas at Austin 전기 및 컴퓨터공학과 졸업(학사). 1992년 5월 : The University of Texas at Austin 전기 및 컴퓨터공학과 졸업(碩士), 미국 Schlumberger 연구원, Motorola 실임 연구원. 현재 연대 공과대학 전기전자공학과 부교수