

논문 2012-49SD-4-4

MLC NAND-형 플래시 메모리를 위한 고장검출 테스트 알고리즘

(Fault Test Algorithm for MLC NAND-type Flash Memory)

장기웅*, 황필주*, 장훈**

(Gi-Ung Jang, Phil-Joo Hwang, and Hoon Chang)

요약

임베디드 시스템의 저장매체 시장에서 플래시 메모리가 점유율을 높여나가고 시스템 내에서 대부분의 면적을 차지하게 되면서, 시스템 신뢰도에 무거운 영향을 미치고 있다. 플래시 메모리는 셀 배열구조에 따라 NOR/NAND-형으로 나뉘어져 있고 플로팅 게이트 셀의 Reference 전압의 갯수 따라 SLC(Single Level Cell)와 MLC(Multi Level Cell)로 구분된다. NAND-형 플래시 메모리는 NOR-형에 비해 속도는 느린 편이지만 대용량화가 쉽고 가격이 저렴하다. 또한 MLC NAND-형 플래시 메모리는 대용량 메모리의 수요가 급격히 높아진 모바일 시장의 영향으로 멀티미디어 데이터 저장의 목적으로 널리 채용되고 있다. 이에 따라 MLC NAND-형 플래시 메모리의 신뢰성을 보장하기 위해 고장 검출 테스트의 중요도 커지고 있다. 전통적인 RAM에서부터 SLC 플래시 메모리를 위한 테스트 알고리즘은 많은 연구가 있었고 많은 고장을 검출해 내었다. 하지만 MLC 플래시 메모리의 경우 고장검출을 위한 테스트 시도가 많지 않았기 때문에 본 논문은 SLC NAND-형 플래시 메모리에서 제안된 기법을 확장한 MLC NAND-형 플래시 메모리를 위한 고장검출 알고리즘을 제안하여 이러한 차이를 줄이려는 시도이다.

Abstract

As the flash memory has increased the market share of data storage in imbedded system and occupied the most of area in a system, It has a profound impact on system reliability. Flash memory is divided NOR/NAND-type according to the cell array structure, and is classified as SLC(Single Level Cell)/MLC(Multi Level Cell) according to reference voltage. Although NAND-type flash memory is slower than NOR-type, but it has large capacity and low cost. Also, By the effect of demanding mobile market, MLC NAND-type is widely adopted for the purpose of the multimedia data storage. Accordingly, Importance of fault detection algorithm is increasing to ensure MLC NAND-type flash memory reliability. There are many researches about the testing algorithm used from traditional RAM to SLC flash memory and it detected a lot of errors. But the case of MLC flash memory, testing for fault detection, there was not much attempt. So, In this paper, Extend SLC NAND-type flash memory fault detection algorithm for testing MLC NAND-type flash memory and try to reduce these differences.

Keywords : MLC NAND-type Flash memory, Fault Test, Algorithm, Memory Test

I. 서론

플래시 메모리는 모바일 장치와 매스 스토리지 산업

에서 중요한 부분을 차지하고 있다. 비휘발성, 갱신가능, 저비용, 저전력, 높은 신뢰도와 고속 등의 특징은 플래시를 매우 매력적인 반도체 메모리로 만들었다. 역사적으로, 1 bit를 저장하기 위해 하나의 트랜지스터가 필요하다. 스케일 축소 기술로 인해 모든 노력은 같은 지역 제약 내에서 메모리 밀도 증가에 치중되고 있다. 이런 노력은 셀 당 데이터를 2bit이상 저장할 수 있는 MLC(Multi Level Cell) 플래시 기술을 낳았다. 보통 2

* 학생회원, ** 정회원, 숭실대학교

(Department of Computer, Soongsil University)

※ 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NO. 2011-0010065)

접수일자: 2012년2월20일, 수정완료일: 2012년3월30일

bit의 데이터가 Intel 사의 [1], [2], [3] 과 NEC 사의 [4] 에서 플래시 장치에서 저장되었다. 이것은 어떤 플래시 어플리케이션도 동일한 배열 영역에서 이중 밀도를 가질 수 있게 함으로써 혁신을 이루었다.

전통적인 단일-레벨 셀(Single Level Cell) 플래시 메모리들은 프로그램 교란에 의해 대부분 영향을 받으며, 테스트 방법은 주로 마치 테스트가 중점적으로 채택되고 있다. 가장 효과적인 고장 모델과 테스트 알고리즘은 [5], [6], [7]에서 마치(march)-like 알고리즘으로 SRAM과 DRAM 테스트 알고리즘으로부터 확장제안되었다. [8]에서 제안된 고장 시뮬레이터는 고장 범위 및 테스트 어플리케이션 시간 측면에서 가장 효과적인 의사-마치 알고리즘 중의 하나이다.

Flash-March^[5], EF^[6]와 March-FT^[7]과 같은 전통적인 알고리즘들은 성공적으로 플래시 셀의 고장을 검출하기 위한 읽기/쓰기 오퍼레이션을 채용하였다. [3]에서 제안된 감지 스키마는 기준 전류에 대한 배열 셀 전류를 비교하여 멀티 레벨을 감지하는 효과적인 수단이다. 이러한 모든 기여 중에서 몇 가지, [9], [10], [11] 내 디자인의 온칩(On-Chip) 테스트에 용이하도록 되어 있다. 최종적으로, [12]는 SLC 플래시의 교란 고장을 모두 진단하고 구별하는 대각선(Diagonal) 알고리즘을 제안하였다. 이런 모든 알고리즘들은 오직 2개의 상태 ("0"과 "1")에 대해서만 고려되어 있다. 이들은 MLC 플래시 배열을 테스트할 수 없고, 고장도 다르게 일어나므로 다양한 레벨을 갖는 테스트에 대한 요구가 발생하게 되었다.

MLC를 위한 테스트 방법론들은 몇몇의 실리콘 실험실의 사후 아날로그 프로빙(Probing)으로 제한되어 있었다. MLC 플래시 메모리가 발명된 후에, 마치 알고리즘을 이용하여 MLC 배열을 테스트하거나 진단한 시도는 직접적 SLC의 방식이 적용되지 않으므로 많지 않다. 본 논문은 복잡한 MLC 고장 테스트와 진단 문제에 대한 간단한 해결책 제안으로 이런 차이를 줄여나가는 시도이다.

플래시 메모리는 일반적으로 셀 배열에 따라 NOR-형과 NAND-형으로 나뉜다^[13]. NOR-형 플래시 메모리는 RAM처럼 셀별 접근이 가능하며 속도가 빠르지만 대용량화가 어렵고 도입 비용이 비싸다. NAND-형 플래시 메모리는 페이지(Page)별로 접근되며 집적도가 높아 대용량화가 용이하지만 속도가 느린 단점이 있다. 하지만 NAND-형 플래시 메모리는 하드디스크에 비해

빠른 속도와 대용량화, 지속적인 가격하락으로 인해 널리 쓰이는 저장매체인 SSD(Solid State Drive)에도 쓰이고 있다. 메모리 영역은 시스템 내에서 차지하는 비율과 함께 신뢰도에 주는 영향이 크기 때문에 NAND-형 플래시메모리 테스트의 중요성이 커지고 있다.

2. 논문의 구성

II장에서는 NAND-형 플래시 메모리의 셀 구조와 배열 구조에 따른 특성을 분석하고, 기존의 SLC NAND-형 플래시 메모리의 고장유형을 살펴본다. MLC 플래시 메모리의 구조와 오퍼레이션의 차이를 소개하고 MLC NAND-형 플래시 메모리에 맞도록 변경된 패턴과 마치 알고리즘을 제안한다. III장에서는 제안한 마치 알고리즘의 복잡도와 함께 고장 진단 시간을 평가하고, 마지막으로 IV장에서는 제시한 MLC NAND-형 플래시메모리를 위한 고장검출 테스트알고리즘의 연구결과를 종합하여 결론을 맺는다.

II. 본 론

1. NAND-형 플래시 메모리 구조

NAND-형 플래시 메모리를 구성하는 플로팅 게이트 셀(Floating gate cell)의 구조는 [그림 1]과 같다. 플로팅 게이트 셀의 구조는 플로팅 폴리실리콘 게이트(polysilicon gate)가 추가된 것 외에는 전통적인 MOS(metal-oxide-semiconductor) 트랜지스터와 유사하다. 플로팅게이트는 터널 옥사이드(tunnel oxide)라 불리는 고 품질의 옥사이드 레이어(Oxide Layer)에 의해 실리콘 기판(Silicon - Substrate)과 절연되어 있는 반면, ONO(Oxide-Nitride-Oxide) 레이어와 인터폴리 절연체(interpoly insulator)에 의해 윗부분과 절연되어 있다^[14]. 실리콘 기판에 존재하는 전하(Electric Charge)들은 플로팅 게이트 셀에 전압을 인가했을 때 생성되는 전기장에 의해 플로팅 게이트에 갇힐 수 있다. Control Gate에 고전압을 인가하면 전하들이 절연체를 넘어서는 활동을 보이게 된다. 그 후 전압인가를 해제하면 플로팅 게이트로 넘어왔던 전하들이 절연체로 둘러싸인 플로팅 게이트에 갇히게 되는 것이다.

이처럼 플로팅 게이트 셀에 데이터를 저장하는 방식은 플로팅 게이트와 실리콘 기판사이에 저장된 전하가 이동되고, 전하의 위치(실리콘 기판 혹은 플로팅 게이트)에 따라 출력되는 전압에 의해 구분되므로 논리적

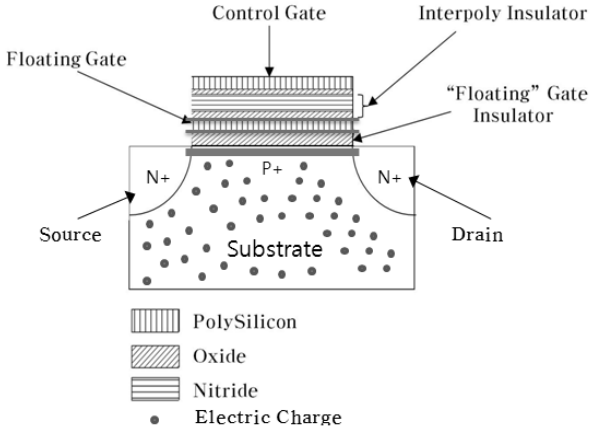


그림 1. 플로팅 게이트 셀 구조
Fig. 1. Structure of Floating Gate Cell.

으로 데이터의 추가기록을 위해서는 전하들의 위치를 실리콘 기판으로 돌려놓아야 한다. 이것은 플래시 메모리의 특징인 Write Once 특성이다. 플래시 메모리는 이 특징 때문에 메모리 셀에 값을 단 한번만 기록할 수 있고 재기록을 위해서는 셀의 소거작업이 이루어져야 한다.

NAND-형 플래시 메모리의 셀 배열구조는 [그림 2]와 같다. SL은 Select Line을, WL은 Word Line을 각각 의미한다. [그림 2]에서 몇몇 플래시 메모리 셀들이 SL 사이에 있는 것을 볼 수 있다. 이 셀들의 숫자는 NAND-형 플래시 메모리의 블록 단위를 의미한다. NAND-형 플래시 메모리의 페이지(Page)는 행을 나타내는 단위로, 하나의 워드라인에 연결된 메모리 셀들의 집합이다. NOR-형과는 달리 데이터를 읽거나 기록할 때("0"기록) 페이지 단위로 수행하게 된다. Write Once 특성 때문에 덮어쓰는 작업도 이루어질 수 없다. 플래시 메모리의 데이터를 변경하는 동작은 program, read, flash가 있다. 플래시 메모리 셀의 초기 값은 "1"이다. 이에 따라 program 동작은 플래시 메모리 셀에 "0"을 기록하게 되고, flash 는 소거동작으로, 플래시 메모리 셀에 "1"을 기록하는 작업이다. 하지만 flash 동작은 플래시 메모리의 구조적 특성에 따라 블록단위로 수행된다. 플래시 메모리를 구성하는 플로팅-게이트 셀은 한 블록위에서 하나의 실리콘 기판을 공유하고 있다^[13]. 플로팅게이트 셀의 값을 "1"로 만들기 위해서는 실리콘 기판에 바이어스보다 높은 고전압을 인가해야 하는데 실리콘 기판 공유되어 있기 때문에 블록내의 모든 플로팅-게이트 셀이 영향을 받게 된다. 이 특징은 플래시 메모리의 고장을 검출하기 위한 테스트를 수행할 때

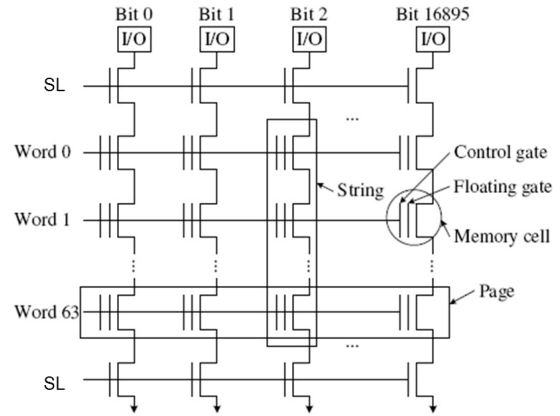


그림 2. NAND-형 플래시 메모리 셀 배열
Fig. 2. Cell Array of NAND-type Flash memory.

"1"을 기록하는 작업에 대한 처리가 필요함을 의미한다.

2. NAND-형 플래시 메모리의 고장유형

NAND-형 플래시 메모리의 고장 유형에 따라 많은 알고리즘들이 제안 되어 있다. 이 중 [15]에서는 NAND-형 플래시 메모리의 특성을 고려하고 효과적으로 에러를 검출할 수 있는 NAND 마치 패턴과 NAND 마치 알고리즘을 제안하였다. 제안한 알고리즘은 기존보다 50%이상의 시간적 성능향상을 보여주었다. 표 1은 플래시 메모리에서 발생할 수 있는 고장 유형을 나타내고 있다. 플래시메모리에서는 기존의 RAM에서 사용되던 고장모델과 함께 교란(Disturb) 고장이 발생할 수 있다^[14].

램스타일 고장은 각각의 셀에서 발생하고 RAM과 NOR/NAND-형 플래시메모리에서 모두 발생할 수 있

표 1. RAM-Style 고장과 Disturb 고장
Table 1. RAM-Style Faults and Disturb Faults.

RAM-Style 고장	
Stuck-At Fault(SAF)	셀이 0이나 1로 고정됨
Transition Fault(TF)	셀이 0(1) → 1(0)로 변화 실패
Stuck-Open Fault(SOF)	셀에 접근 불가
Address Decoder Fault(AF)	셀과 주소가 1대 1 매칭 안됨
Disturb 고장	
Word-Line Erase Disturb(WED)	0 → 1
Word-Line Program Disturb(WPD)	1 → 0
Bit-Line Erase Disturb(BED)	0 → 1
Bit-Line Program Disturb(BPD)	1 → 0

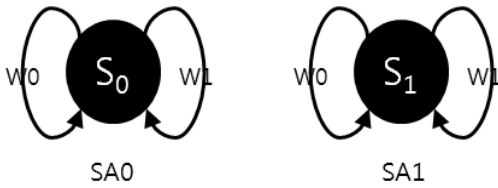


그림 3. RAM Style 고장 : 고착고장
Fig. 3. RAM Style Fault : SAF.

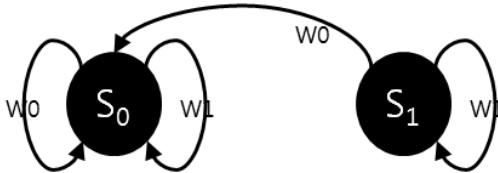


그림 4. RAM Style 고장 : 천이고장
Fig. 4. RAM Style Fault : TF.

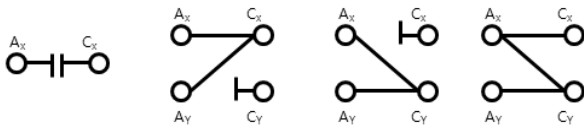


그림 5. RAM Style 고장 : 주소 고장
Fig. 5. RAM Style Fault : AF.

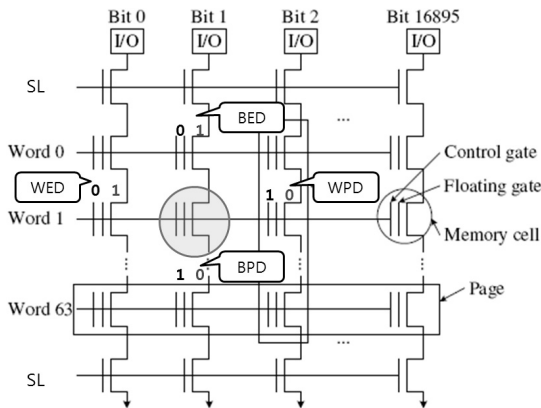


그림 6. 교란고장
Fig. 6. Disturb Faults.

다. 예를 들면 램스타일 고장은 SAF, TF, SOF 그리고 AF를 포함하고 있다. SAF를 갖는 셀은 “0”이나 “1”로 고정되어 버린다[그림 3]. TF를 갖는 셀은 자신이 갖는 값 “0”(“1”)을 “1”(“0”)로 변환하지 못한다[그림 4]. SOF를 갖는 셀은 셀과 관련된 라인들의 고장으로 접근할 수 없게 된다. 결과적으로 반환된 값은 감지증폭기에 의해 이전페이지의 값으로만 출력된다. AF는 셀과 주소가 1대 1 매칭을 하지 못하는 것으로 4가지 형태를 띠고 있다. 첫째는 고장주소가 어느 셀과도 연결되어 있지 않은 경우이고, 둘째는 하나의 셀에 고장주소가 2

개 이상 연결된 경우이다. 셋째는 고장주소가 2개 이상의 셀과 연결되어 있는 경우이고 넷째는 1대1로 연결된 셀들 중 특정 주소하나가 다른 주소와 연결된 셀과도 연결된 경우이다[그림 5].

램 스타일 고장과 다르게 교란고장은 플래시 메모리에서만 발생한다. 결함 셀은 같은 워드라인이나 비트라인의 특정 셀에 접근될 때 교란을 받을 수 있게 된다. WED/WPD는 같은 워드라인의 특정 셀에 접근할 때 결함 셀이 예기치 않게 소거(Erasing)되거나 프로그램(Programming) 되는 고장이다. BED/BPD는 같은 비트라인의 특정 셀에 접근될 때 결함 셀이 예기치 않게 소거되거나 프로그램 되는 고장을 말한다[그림 2-6].

3. MLC NAND-형 플래시 메모리의 구조

SLC 플래시는 하나의 셀에 1bit가 저장되는 반면 MLC 플래시는 하나의 셀에 2bit가 저장된다^[16]. [그림 7]는 SLC NAND-형 메모리가 Reference Point를 이용하여 저장 비트를 구분하는 것을 보여준다. SLC NAND-형 플래시 메모리는 하나의 Reference 전압을 기준으로 각각의 비트는 “Programmed”나 “Erased”의 상태를 갖는다. [표 2]는 SLC에서 셀 상태를 나타낸다. MLC NAND-형 플래시 메모리는 [그림 8]처럼 3개의 Reference 전압을 기준으로 00, 01, 10, 11의 상태를 [표 3]과 같이 갖는다.

SLC와 MLC 형 플래시 메모리는 이와 같은 플로팅 게이트 셀의 상태 차이가 존재하기 때문에 2.3에서와 같은 알고리즘은 MLC NAND-형 플래시 메모리에서는 모든 셀 상태에 대한 고장을 검출할 수 없다.

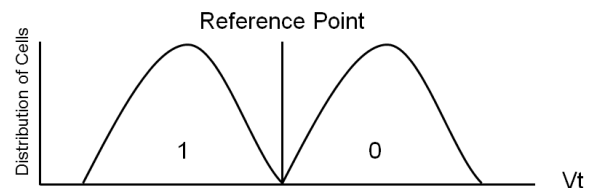


그림 7. SLC 기준 전압
Fig. 7. SLC Reference Points.

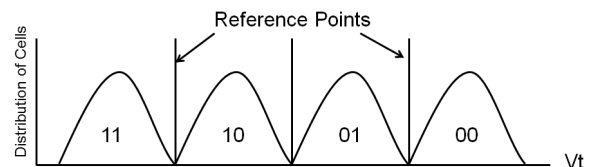


그림 8. MLC 기준 전압
Fig. 8. MLC Reference Points.

표 2. SLC 셀 상태
Table 2. SLC Cell State.

Value	State
0	Programmed
1	Erased

표 3. MLC 셀 상태
Table 3. MLC Cell State.

Value	Programmed
00	Fully Programmed
01	Partially Programmed
10	Partially Erased
11	Fully Erased

4. 제안하는 MLC NAND-형 플래시 메모리 테스트 패턴

MLC NAND-형 플래시 메모리의 고장들을 테스트 하기 위해 [15]에서 제안된 NAND Parallel 패턴을 쉽게 확장할 수 있다. MLC NAND-형 플래시 메모리에서는 “00”, “01”, “10”, “11”의 상태들 간에 전환을 고려하고 중복을 배제하여 [표 4]와 같이 총 12개의 조합을 생성해야 한다. 본 논문에서는 “00”, “01”, “10”, “11”의 셀 상태에 대한 이해를 돕기 위해 A, B, C, D의 상태들로 표현한다.

[그림 9]는 [15]에서의 SLC NAND-형 플래시 메모리에 사용된 것과 같이 각각 MLC NAND-형 플래시 메모리 셀의 상태에 따른 오류들을 검출 할 수 있도록 구성된 MLC NAND(A-B) 패턴을 나타내고 있다. 다른 조합(A-C, A-D, B-C, B-D, C-D)의 경우에도 마찬가지로 패턴이 존재한다.

위 패턴을 포함한 MLC NAND 패턴들은 셀에 저장되는 값이 “0”, “1”이던 SLC 플래시 메모리와 달리 “00”, “01”, “10”, “11”로 구성된 MLC 플래시 메모리 셀의 특성을 고려하여 모든 상태에 대한 고장을 검출할 수 있도록 확장된 것이다. [15]의 알고리즘에서와 같이

표 4. MLC NAND-형 플래시 메모리 테스트 패턴 조합
Table 4. Test Pattern Combination for NAND-type Flash memory.

	A	B	C	D
A		A-B	A-C	A-D
B	B-A		B-C	B-D
C	C-A	C-B		C-D
D	D-A	D-B	D-C	

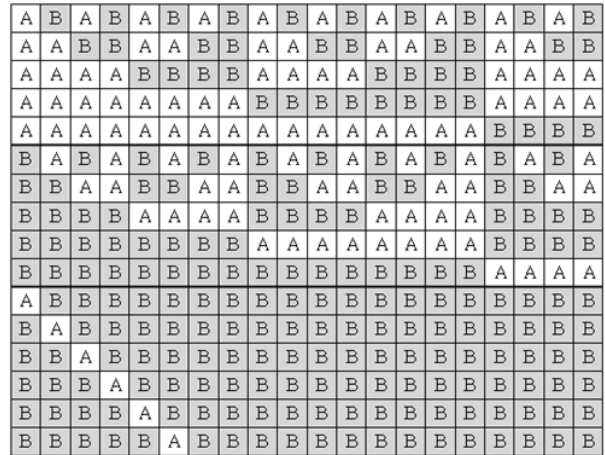


그림 9. MLC NAND(A-B) 패턴
Fig. 9. MLC NAND(A-B) Pattern.



그림 10. Inverse MLC NAND(A-B) 패턴
Fig. 10. Inverse MLC NAND(A-B) Pattern.

Disturb 고장까지 모두 검출하려면 제안한 패턴들의 Inverse 패턴이 필요하다. Inverse MLC NAND(B-A) 패턴은 [그림 10]과 같다. Inverse MLC NAND(C-A, D-A, C-B, D-B, D-C) 패턴의 경우에도 마찬가지로 존재한다.

5. 제안하는 MLC NAND-형 마치 테스트 알고리즘

MLC NAND 테스트 Pattern 을 이용하여 MLC NAND-형 플래시 메모리를 테스트하기 위해서 전통적으로 테스트에 주로 쓰이는 마치 테스트 알고리즘을 제안한다. 오퍼레이션 중 rD(rC, rB, rA)는 페이지로부터 “D”(“C”, “B”, “A”)을 읽는 것이고 wD(wC, wB, wA)는 페이지 단위로 “D”(“C”, “B”, “A”)을 쓰기 작업이 이루어지며, MLCNAND(x)는 MLCNAND(A-B), MLCNAND(A-C), MLCNAND(A-D), MLCNAND(B-C), MLCNAND

(B-D), MLCNAND(C-D) 등 MLCNAND 마치 패턴에 따라 바뀌며 비트패턴을 페이지 단위로 읽거나 쓰게 한다. $\overline{MLC}_{NAND(x)}$ 는 각 Inverse 패턴을 의미한다.

$$\begin{aligned} \text{MLC NAND-March}(x) = \{ \\ (e); \downarrow(r11, w\overline{MLC}_{NAND(x)}, r\overline{MLC}_{NAND(x)}); \\ \quad \Downarrow(r\overline{MLC}_{NAND(x)}); \\ (e); \downarrow(r11, w\overline{MLC}_{NAND(x)}, r\overline{MLC}_{NAND(x)}); \\ \quad \Downarrow(r\overline{MLC}_{NAND(x)}); \\ (e); \downarrow(r11, w00, r00) \} \end{aligned} \quad (1)$$

알고리즘 (1)은 패턴 조합 x에 따라 총 6번의 수행에 적용된다. MLC NAND-형 플래시 메모리의 테스트를 위해서는 6가지 알고리즘을 모두 사용하게 된다. 6개의 마치 엘리먼트로 이루어진 알고리즘 (1)은 각 패턴 조합별로 SAF, TF, AF, SOF, Word-Line Disturb Fault의 검출이 가능하며 각각 플래시칩 블록을 모두 소거하는 Erase 오퍼레이션으로 시작한다. 첫 번째 마치 엘리먼트는 모든 플래시 칩을 소거하고, 두 번째 마치 엘리먼트는 모든 셀에서 “D”를 읽고 패턴 MLCNAND(x)를 기록한 후 다시 읽어낸다. 세 번째 마치 엘리먼트는 기록한 패턴 MLCNAND(x)를 다시 한 번 읽음으로써 MLCNAND(x)를 기록하고 읽는데 DF가 발생하지 않았는지 알 수 있게 한다. 네 번째 마치 엘리먼트는 블록을 다시 “D”로 Erase하고 다섯 번째 마치 엘리먼트에서 Inverse 패턴을 기록한다. 다섯 번째 마치 엘리먼트에서 Inverse 패턴을 기록하는 목적은 MLC NAND-형 플래시 메모리 패턴의 아래 부분을 차지하는 Diagonal 패턴의 기록되지 남은 셀들에 의한 워드라인 교란 고장을 비롯한 DF를 검출하기 때문이다. 두 번째와 다섯 번째 마치 엘리먼트에서 쓰이고 읽혀진 MLCNAND(x)와 Inverse MLCNAND(x)는 AF 중 첫 번째와 세 번째 유형은 100% 검출이 가능하지만 SLC의 경우처럼 두 번째와 네 번째 유형은 일부만 검출 가능하다. 여섯 번째 마치 엘리먼트는 Inverse 패턴일 때 DF를 검출한다. 일곱 번째 마치 엘리먼트는 패턴에 “D”가 기록되지 않고 소거동작으로만 나타날 수 있는 MLCNAND(A-B), MLCNAND(A-C), MLCNAND(B-C)에서 필요하다. 일곱 번째 마치 엘리먼트까지 진행되면 메모리 셀은 주어진 패턴의 2가지 값과 “D”에 대한 셀 값의 변화가 일어나면서 SAF, TF를 검출가능하게 된다. 예를 들면 MLCNAND(A-B)가 수행되면 (0, 0) 셀의 값은 첫 번째 마치 엘리먼트에서 “D”, 두 번째에서는 “A”, 네 번

째에서는 “D”, 다섯 번째에서는 “B”, 일곱 번째에서는 “D”, 여덟 번째에서는 “A”로 바뀌게 된다. 셀에 SAF가 있는 경우에는 셀이 “A”, “B”, “D” 중 하나의 값으로 고정되어 검출된다. TF가 있는 경우에도 “A”, “B”, “D” 간의 상태변화가 모두 일어나는 과정을 거치기 때문에 (단, “D”를 제외한 모든 상태는 “D”를 거쳐야 다른 값으로 전환이 가능하다) 이를 보고 검출할 수 있다. 여덟 번째 마치 엘리먼트까지 진행되면서 셀 값이 기대한 값이 아닌, 직전의 WL이나 BL에서 검출되었던 값이 계속 검출되면 SOF임을 알 수 있다.

III. 실험

1. MLC NAND-March(x) 알고리즘 복잡도

제안한 MLC NAND-March(x) 알고리즘의 복잡도를 계산한다.

NAND-형 플래시 메모리의 배열 크기는 가로(m) × 세로(n)이며 “N”으로 표현한다. “A”, “B”, “C”, “D”에 대한 읽기 및 프로그램의 동작 속도는 같다고 가정하였다. 또한, 읽기 시간은 “r”, 프로그램 시간은 “p”, 소거 시간은 “e”로 표현한다. 제안한 MLC NAND-March(x)로부터 구할 수 있는 알고리즘 복잡도는 다음과 같다.

$$C = 3e + 8N(r) + 3N(p) \quad (2)$$

MLC NAND-March(x)는 첫 번째, 네 번째, 일곱 번째 마치 엘리먼트가 소거 동작으로 이루어져 있어 3번의 “e”가 계산된다. 또한 첫 번째, 네 번째, 일곱 번째 마치 엘리먼트를 제외한 모든 엘리먼트에 있는 읽기 동작으로 8번의 “r”이 계산된다. 마지막으로 두 번째, 다섯 번째, 여덟 번째에 있는 쓰기 동작으로 3번의 “p”가 추가 되었으며 MLC NAND-March(x)는 6번 수행되어야 하기 때문에 복잡도 C에 6배의 수치가 필요하다.

2. MLC NAND-March(x) 알고리즘 고장진단 시간

알고리즘 복잡도를 구한 후에는 고장 진단 시간을 산출할 수 있다. [17]에서 발표된 2Mbit 플래시 배열의 이론적인 수치를 바탕으로 계산하였다. 각 동작에 소요되는 시간은 각각 “r” = 1.25ns, “p” = 1.5us, “e” = 200ms이다. 복잡도 C에 따른 진단 시간은 9.84초이다.

IV. 결 론

NAND-형 플래시 메모리는 기존 저장장치에 비해 빠른 속도와 저전력 등의 이유로 품질향상을 위한 연구가 계속되고 있다. 본 논문에서는 대용량화가 용이하고 생산단가가 저렴해 많은 보급이 늘어나고 있으며 신뢰도 측면에서 중요한 역할을 차지하는 MLC NAND-형 플래시 메모리의 구조적 특성에 맞춰 고장을 검출할 수 있도록, 기존의 SLC에서 제안되었던 패턴을 MLC NAND-형 플래시 메모리를 위해 확장하였다. 고장진단을 위해 필요한 본 논문의 고장검출 알고리즘은, 기존에 제안된 기법의 장점을 그대로 수용하고 MLC NAND-형 플래시 메모리에도 사용할 수 있도록 함으로써 메모리로 인한 컴퓨팅 시스템 신뢰도를 검증할 수 있을 것으로 기대된다. 또한 MLC NAND-형 플래시 메모리의 수율향상에도 기여할 수 있을 것이다.

제안한 알고리즘은 MLC NAND-형 플래시 메모리에 적용하기 위한 시도로써 다소 이론적인 성격이 강하다. 향후에는 더욱 많은 고장을 검출하고 복잡도를 줄여나가는 연구가 필요하며 더욱 많은 시도가 이루어질 수 있도록 범용적인 시뮬레이터의 연구도 필요하다.

참 고 문 헌

[1] A. Fazio, et al., Intel Strata Flash Memory Technology Overview.
<http://www.intel.com/design/flash>.

[2] M. Bauer et al. "A Multi-level Cell 32Mb Flash Memory,," In Proceedings of 42nd International Solid-State Circuits Conference, pp 132-133, Feb 1995.

[3] D. Elmhurst and M. Goldman, "A 1.8V 128 Mb 125 MHz Multilevel Cell Flash Memory With Flexible Read While Write," IEEE Journal of Solid State Circuits, 38(11):1929- 1933, November 2003.

[4] M. Okhawa et al., "A 98mm² 3.3V, 64Mb Flash Memory with FN-NOR Type 4-Level Cell," In Proceedings of 43rd International Solid-State Circuits Conference, pp 36-37, February 1996.

[5] M. G. Mohammed and K. K. Saluja, "Flash Memory Disturbances: Modeling and Test," Proceedings of 19th VLSI Test Symposium, 2001, pp 218-224, April 2001.

[6] M. G. Mohammad, K. K. Saluja, and A. Yap, "Testing Flash Memories," In Proceedings of

Thirteenth Int'l Conference on VLSI Design, pp 406-411, 2000.

[7] J. Yeh, C. et al, "Flash Memory Built-In Self Test Using March-Like Algorithms," In Proceedings of the First IEEE Intl. Workshop on Electronic Design, Test and Applications, pp 137-141, 2002.

[8] K. L. Cheng, J. C. Yeh, C. W. Wang, C.T. Huang, and C. W. Wu, "RAMSES-FT: A Fault Simulator for Flash Memory Testing and Diagnostics," In Proceedings of 20th IEEE VLSI Test Symposium, pp 281-286, 2002.

[9] C. T. Huang, J. C. Yeh, Y. Y. Shih, R. F. Huang, and C. W. Wu, "On Test and Diagnostics of Flash Memories," In Proceedings of 13th Asian Test Symposium, pp 260-265, November 2004.

[10] C. L. Su, R. F. Huang, and C. W. Wu, "A Processor-Based Built-In Self-Repair Design for Embedded Memories," In Proceedings of 12th Asian Test Symposium, pp 366-371, November 2003.

[11] Y. Horng, J. Huang, and T. Chang, "March Test and On-Chip Test Circuit of Flash Memories," In Proceedings of 43rd Midwest Symposium on Circuits and Systems, volume 1, pp 128-131, August 2000.

[12] S. K. Chiu, J. C. Yeh, C. H. Huang, and C. W. Wu, "Diagonal Test and Diagnostic Schemes for Flash Memories," In Proceedings of International Test Conference, pp 37-46, 2002.

[13] Yu-Ying Hsiao, Chao-Hsun Chen, and Cheng-Wen Wu, "Built-In Self-Repair Schemes for Flash Memories", IEEE Transactions on computer-aided design of integrated circuits and systems, Vol. 29, No. 8, August 2010.

[14] Pavan P, Bez R, Olivo P, Zanoni E. "Flash memory cells-.an overview," Proc IEEE 1997;85(8):1248-.71

[15] Tei-Wei Kuo, Po-Chun Huang, Yuan-Hao Chang, Chia-Ling Ko, Chih-Wen Hsueh, "An efficient fault detection algorithm for NAND flash memory," Proc ACM SIGAPP Applied Computing Review, Vol 11, Issue 2, 2011.

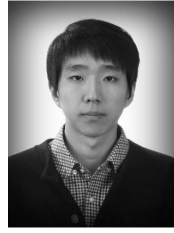
[16] Advantech, Comparing SLC and MLC Flash Technologies and Structure, September, 2009.

[17] S. K. Chiu, J. C. Yeh, C. H. Huang, and C. W. Wu, "Diagonal Test and Diagnostic Schemes for Flash Memories," In Proceedings of International Test Conference, pp 37-46, 2002.

저 자 소 개



장 기 응(학생회원)
 2010년 목포대학교 컴퓨터공학과
 학사 졸업.
 2012년 숭실대학교 대학원
 컴퓨터학과 석사 졸업.
 <주관심분야 : 컴퓨터구조, 메모리
 테스트, VLSI 설계 및 테스트, 임
 베디드 시스템>



황 필 주(학생회원)
 2010년 숭실대학교 전자계산원
 학사 졸업.
 2011년~현재 숭실대학교 대학원
 컴퓨터학과 석사과정.
 <주관심분야 : 컴퓨터구조, 메모리
 테스트, VLSI 설계 및 테스트, 임
 베디드 시스템, 메모리 에러정정>



장 훈(정회원)
 1987년 서울대학교 공대
 전자공학과 학사 졸업.
 1989년 서울대학교 공대
 전자공학과 석사 졸업.
 1993년 University of Texas at
 Austin 박사 졸업.

1991년 IBM Inc. Senior Member of Technical Staff.

1993년 Motorola Inc. Senior Member of Technical Staff.

1994년~현재 숭실대학교 컴퓨터학부 교수.

<주관심분야 : 컴퓨터구조, 메모리 테스트, VLSI
 설계 및 테스트, 임베디드 시스템>