



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년04월26일
 (11) 등록번호 10-1615773
 (24) 등록일자 2016년04월20일

(51) 국제특허분류(Int. Cl.)
 G11C 16/34 (2006.01) G11C 16/06 (2006.01)
 G11C 16/10 (2006.01) G11C 7/02 (2006.01)
 (21) 출원번호 10-2009-7005011
 (22) 출원일자(국제) 2007년09월07일
 심사청구일자 2012년09월04일
 (85) 번역문제출일자 2009년03월10일
 (65) 공개번호 10-2009-0074733
 (43) 공개일자 2009년07월07일
 (86) 국제출원번호 PCT/US2007/077940
 (87) 국제공개번호 WO 2008/031074
 국제공개일자 2008년03월13일
 (30) 우선권주장
 11/530,392 2006년09월08일 미국(US)
 (뒷면에 계속)
 (56) 선행기술조사문헌
 JP2006228407 A
 JP1995240098 A
 US20060126390 A1
 US5996041 A

(73) 특허권자
샌디스크 테크놀로지스, 인코포레이티드
 미국 텍사스 75024 플라노 노스 달라스 파크웨이
 6900 투 리가시 타운 센터
 (72) 발명자
리, 안
 미국, 캘리포니아 95035, 밀피타스, 케베네어 드
 라이브 695
풍, 유핀 카윙
 미국, 캘리포니아 94539, 프레몬트, 브루스 드라
 이브 2938
모클레시, 니마
 미국, 캘리포니아 95032, 로스 가토스, 쉘린다 웨
 이 14285
 (74) 대리인
양영준, 백만기, 정은진

전체 청구항 수 : 총 30 항

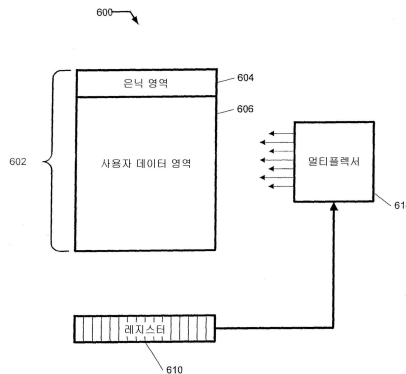
심사관 : 한선경

(54) 발명의 명칭 **플래시 메모리에서 사이클링 효과들에 대한 의사 랜덤 및 명령 구동 비트 보상 및 이를 위한 방법**

(57) 요약

플래시 메모리 EEPROM 내에서 용이하게 구현되는 랜덤화는 특정 데이터 패턴들의 반복적이고 장기간 저장에 기인한, NAND 스트링 저항효과, 프로그램 교란, 사용자 관독 교란, 및 플로팅 게이트간 커플링을 감소시킨다. 랜덤화는 상이한 실시예들에서, 코드로 발생된 의사(pseudo) 랜덤화이거나 사용자에게 의해 구동된 랜덤화일 수 있다. 사용자에게 의해 구동되는 명령들은 이의 타이밍을 예측될 수는 없으나, 고 수준의 랜덤화를 트리거하고 달성하기 위해 사용될 수 있다. 데이터의 인코딩 방식을 랜덤하게 변경하는 것은 특정 데이터 패턴들의 반복적이고 장기간의 저장을 방지한다. 사용자가 긴 기간 동안 동일 정보를 저장하거나, 반복하여 이를 저장하기를 원할지라도, 서로 다른 인코딩 방식들로 랜덤하게 인코딩될 것이며, 그러므로 데이터 패턴은 다양해질 것이다.

대표도 - 도21



(30) 우선권주장

11/530,399 2006년09월08일 미국(US)

11/852,229 2007년09월07일 미국(US)

명세서

청구범위

청구항 1

플래시 메모리 EEPROM을 포함하는 저장 장치에 있어서,

상기 EEPROM은:

은닉 영역 및 사용자 데이터 영역을 포함하는 플래시 메모리 어레이;

복수의 비트 레지스터를 포함하는 주변 회로;

를 포함하고,

복수의 랜덤(random) 또는 의사 랜덤(pseudo-random) 비트 코드는 상기 레지스터에 저장되고,

상기 저장 장치는 페이지를 인코딩하기 위해 사용된 인코딩 방식을 반전시키거나 상기 인코딩 방식을 그대로 유지하기 위해 한 페이지의 데이터의 페이지 주소에 대한 코드의 각 비트를 비교하기 위한 비교기를 더 포함하는,

저장 장치.

청구항 2

제 1항에 있어서, 상기 코드는 먼저 상기 어레이의 상기 은닉 영역에 저장되고, 이어서 상기 레지스터로 전송되는, 저장 장치.

청구항 3

제 2항에 있어서, 상기 전송은 상기 EEPROM의 기동시 행해지는, 저장 장치.

청구항 4

제 1항에 있어서, 상기 EEPROM은 멀티플렉서를 더 포함하고, 상기 비교기는 극성 비트(polarity bit)를 발생하며, 상기 극성 비트는 상기 멀티플렉서를 통해 다수의 페이지들에 할당되는, 저장 장치.

청구항 5

복수의 페이지들의 사용자 데이터를 포함하는 비휘발성 NAND 플래시 메모리 칩에 있어서, 상기 칩은,

메모리 어레이와,

유한 상태의 머신 클럭 신호(machine clock signal)와,

명령 클럭 신호(command clock signal)와,

상기 유한 상태의 머신 클럭 신호 및 상기 명령 클럭 신호를 기준으로 발생하는 극성 비트와,

상기 극성 비트의 제 1 값에 대한 제 1 인코딩 유형과,

상기 극성 비트의 제 2 값에 대한 제 2 인코딩 유형을

포함하는, 비휘발성 NAND 플래시 메모리 칩.

청구항 6

제 5항에 있어서, 상기 극성 비트는 상기 명령 클럭 신호의 상승 에지에서 상기 유한 상태의 머신 클럭 신호를 기준으로 발생하는, 비휘발성 NAND 플래시 메모리 칩.

청구항 7

삭제

청구항 8

복수의 페이지들의 데이터를 포함하는 비휘발성 메모리에서, 방법은,

복수의 비트들을 포함하는 코드를 상기 비휘발성 메모리에 저장하는 단계와 - 상기 복수의 비트들 각각은 상기 복수의 페이지들 중 한 페이지의 데이터의 엔코딩이 반전될 것인지 아니면 그대로 둘 것인지를 정의함 -,

상기 비휘발성 메모리로부터 상기 코드를 판독하는 단계와,

상기 비휘발성 메모리로부터의 상기 코드를 상기 복수의 비트들의 코드를 저장하기 위해 할당된 레지스터에 전송하는 단계와,

상기 레지스터에 상기 코드를 저장하는 단계와,

상기 복수의 페이지들 중 한 페이지의 페이지 주소를 상기 코드의 각각의 비트에 대하여 상기 코드의 비트와 비교함으로써 극성 신호를 발생시키는 단계와,

발생된 상기 극성 신호에 기초하여 상기 복수의 페이지들 중 한 서브세트의 페이지들의 엔코딩을 변경하는 단계를

포함하는, 방법.

청구항 9

제 8항에 있어서, 상기 극성 신호가 상기 복수의 페이지들의 그룹에 입력되고, 상기 그룹의 상기 엔코딩이 상기 코드의 단일 비트에 의해 결정되게 상기 극성 신호를 멀티플렉싱하는 단계를 더 포함하는, 방법.

청구항 10

제 8항에 있어서, 상기 코드는 17 또는 그 이상의 비트들을 포함하는, 방법.

청구항 11

제 8항에 있어서, 상기 코드를 상기 비휘발성 메모리에 저장하는 단계는 상기 비휘발성 메모리의 은닉 부분에 상기 코드를 저장하는 단계를 포함하는, 방법.

청구항 12

제 8항에 있어서, 상기 비휘발성 메모리로부터 상기 코드를 판독하는 단계는 상기 메모리의 기동시 행해지는, 방법.

청구항 13

제 8항에 있어서, 하나 이상의 페이지들의 사용자 데이터를 판독하는 단계를 더 포함하고, 상기 사용자 데이터의 판독은 상기 페이지 주소에 의해 판독될 각 페이지의 상기 극성을 판정하는 단계를 포함하는, 방법.

청구항 14

메모리 제어기 및 플래시 메모리 어레이를 포함하는 플래시 저장 장치에서, 상기 어레이는 사용자 및 그외 데이터를 저장하기 위해 사용되고, 상기 어레이에서 이행되는 방법은,

상기 어레이에 상기 사용자 데이터를 프로그램 또는 판독하기 위한 사용자 명령을 수신하는 단계와;

상기 사용자 명령 수신시, 상기 사용자 데이터의 엔코딩을 랜덤화하기 위해서 한 블록의 상기 사용자 데이터에 대한 복수의 비트들을 포함하는 코드를 사용하는 단계를 포함하고,

상기 코드의 상기 비트들 각각은 상기 블록의 하나 이상의 페이지들의 그룹이 그대로 저장되는지 아니면 저장되기에 앞서 반전되는지를 결정하는 코드를 사용하는,

방법.

청구항 15

제 14항에 있어서, 상기 코드에 의해 조각된 상기 사용자 데이터의 세그먼트(segment)와 함께 상기 코드를 저장하는 단계를 더 포함하는, 방법.

청구항 16

제 14항에 있어서, 상기 코드는 17 이상의 비트들을 포함하는, 방법.

청구항 17

제 14항에 있어서, 상기 코드는 사용자 명령들 사이에서 업데이트되는, 방법.

청구항 18

제 14항에 있어서, 상기 코드가 블록 내 페이지들의 수보다 적은 비트들을 포함한다면, 상기 코드는 하나 이상의 그룹의 하나 이상의 페이지들에 할당되는, 방법.

청구항 19

제 15항에 있어서, 상기 방법은 상기 세그먼트에 대해 저장된 상기 코드를 판독하는 단계와, 비트단위로 상기 사용자 데이터를 상기 코드의 적용 전에 있던 상기 엔코딩 방식으로 되돌리는 단계를 더 포함하는, 방법.

청구항 20

비휘발성 메모리에서, 방법은,

비트 스트링을 보유하는 시프트 레지스터를 제공하는 단계;

사용자에 의해 발생된 명령 신호의 상승에지에서, 상기 시프트 레지스터가 한 비트씩 시프트시키는 단계;

상기 시프트 레지스터의 위치에 저장된 값을 모니터링하는 단계;

상기 시프트 레지스터의 모니터링 된 위치에서의 제 1 값에 대하여, 사용자 데이터의 부분의 엔코딩 방식을 반전시키는 단계; 및

상기 시프트 레지스터의 모니터링 된 위치에서의 제 2 값에 대하여, 사용자 데이터의 상기 부분의 엔코딩 방식을 그대로 유지하는 단계

를 포함하는, 방법.

청구항 21

제 20항에 있어서, 상기 부분은 하나 이상의 페이지들을 포함하는, 방법.

청구항 22

제 20항에 있어서, 상기 사용자에 의해 발생된 명령신호는 프로그램 명령인, 방법.

청구항 23

제 20항에 있어서, 상기 사용자에 의해 발생된 명령신호는 판독 명령인, 방법.

청구항 24

제 20항에 있어서, 상기 사용자에 의해 발생된 명령신호는 소거 명령인, 방법.

청구항 25

제 20항에 있어서, 상기 시프트 레지스터의 출력의 상기 제 1 값 또는 제 2 값을 사용자 데이터의 각 부분에 대해 상기 비휘발성 메모리에 저장하는 단계를 더 포함하는, 방법.

청구항 26

제 25항에 있어서, 주어진 부분의 데이터에 대해 상기 비휘발성 메모리로부터 상기 저장된 값을 판독하는 단계와,

상기 저장된 값이 상기 제 1 값이라면, 상기 부분의 판독에 앞서 상기 엔코딩 방식을 본래 상태로 다시 반전시

키기 위해 상기 데이터의 부분을 인코딩하기 위해 사용된 상기 인코딩 방식을 반전시키는 단계를,
상기 저장된 값이 상기 제 2 값이라면, 상기 부분의 판독에 앞서 상기 인코딩 방식을 그대로 유지하는 단계를
더 포함하는, 방법.

청구항 27

복수의 페이지들의 사용자 데이터를 포함하는 NAND 플래시 메모리 칩에서, 방법은,
칩에서 사용자 명령을 수신하였을 때, 제 1 및 제 2 상태를 가지는 상기 플래시 메모리 칩의 유한 상태 머신 클
럭을 기준으로 하는 단계와,
상기 유한 상태 머신 클럭이, 기준이 되는 상기 제 1 상태에 있다면, 사용자 데이터의 부분을 인코딩하기 위해
사용된 인코딩 방식을 반전시키는 단계를,
상기 유한 상태 머신 클럭이, 기준이 되는 상기 제 2 상태에 있다면, 상기 인코딩 방식을 그대로 유지하는 단계
를
포함하는, 방법.

청구항 28

제 27항에 있어서, 상기 유한 상태 머신 클럭은 사용자 명령에 의해 발생된 신호의 상승에지에서 기준이 되는,
방법.

청구항 29

제 27항에 있어서, 상기 사용자 명령은 캐시 프로그램 명령인, 방법.

청구항 30

제 27항에 있어서, 상기 부분은 하나 이상의 페이지들의 사용자 데이터를 포함하는, 방법.

청구항 31

제 27항에 있어서, 상기 유한 상태 머신 클럭의 상태의 표시는 시프트 레지스터의 위치에 저장되는, 방법.

발명의 설명

기술 분야

[0001] 본 발명은, 일반적으로 디지털 장치들에 및 이로부터 큰 파일들을 저장 및 전송하기 위해 사용되는 메모리 카드
들 및 휴대 유니버설 직렬 버스("USB") 플래시 메모리 드라이브들과 같은 휴대 대량 저장 장치들에 관한
것이고, 보다 구체적으로 이들 드라이브들에서 반복적으로 저장되는 비트 패턴들의 영향을 최소화하는 것에 관
한 것이다.

배경 기술

[0002] 플래시 메모리 카드들 및 드라이브들의 용량이 증가함에 따라, 메모리 어레이 내에 메모리 셀들의 스케일은 계
속하여 감소한다. 특히 NAND 종류의 고밀도 어레이 내에, 어레이의 한 셀 또는 부분에 저장되는 전하는 이웃 셀
의 판독 또는 프로그램 동작에 영향을 미칠 수 있다. 이것은 판독 또는 프로그램 교란 및 셀 커플링으로서 알려
진 것이다.

[0003] 셀 커플링, 교란들, 그리고 일반적으로 NAND 플래시의 동작과 구조에 관한 더 많은 정보에 대해서, 모든 목적들
을 위해 전체를 참조로 여기 포함하는 미국특허 출원번호들로서, "Method for Non-Volatile Memory With
Background Data Latch Caching During Program Operations" 명칭의 11/381,995; "Method for Non-Volatile
Memory With Background Data Latch Caching During Erase Operations" 명칭의 11/382,001; "Method for Non-
Volatile Memory With Background Data Latch Caching During Read Operations" 명칭의 11/381,994; 현재 특
허 6,870,768인, "Techniques for Reducing Effects of Coupling Between Storage Elements of Adjacent Rows
of Memory Cells" 명칭의 10/923,320; 및 "Reducing Floating Gate to Floating Gate Coupling Effect" 명칭

의 11/021,872를 참조한다.

- [0004] 플래시 메모리는 플래시 메모리의 어떤 블록들에 동일 데이터 패턴을 몇 번이고 반복하여 끊임없이 저장하기 위해 어떤 사용자들에 의해 흔히 사용된다. 결과는, 소거되었지만 결코 프로그램되지 않고 남은 어떤 비트들이 있을 것이라는 것이다. 또한, 항시 프로그램되지만 드물게 소거되는 어떤 비트들도 있을 것이다. 이들 지속적인 데이터 패턴들은 이들이 교란들 및 이외에 곤란들로서 이를테면 플로팅 게이트 간 효과들, NAND 스트링 저항 효과, 그리고 감소한 메모리 내구성 및 신뢰성, 등을 야기할 수 있는 점에서 문제가 된다.
- [0005] 특히 소형 폼 팩터(form factor) 카드로서 패키지 된 EEPROM 및 플래시 EEPROM 형태인, 전하를 비휘발성으로 저장할 수 있는 고체 상태 메모리는 최근에 다양한 이동 및 휴대장치들, 특히 정보 기기들 및 소비자 전자제품들에서 선택되는 저장 장치가 되었다. 역시 고체상태 메모리인 RAM(랜덤 액세스 메모리)과는 달리, 플래시 메모리는 비휘발성이어서, 이의 저장된 데이터를 전기가 턴 오프 된 후에도 보존한다. 더 높은 비용에도, 플래시 메모리는 대량 저장 애플리케이션들에서 점점 더 사용되고 있다. 하드 드라이브들 및 플로피 디스크들과 같은 회전 자기매체에 기반한 종래의 대량 저장 장치는 이동 및 휴대 환경엔 적합하지 않다. 이것은 디스크 드라이브들이 부피가 크고 기계적 고장이 나기 쉽고 큰 레이턴시(latency) 및 고 파워 요구조건들을 갖는 경향이 있기 때문이다. 이들 바람직하지 못한 속성들은 디스크 기반의 저장 장치를 대부분의 이동 및 휴대 애플리케이션들에서 사용되지 못하게 한다. 한편, 내장되고 탈착 가능한 카드 형태의 플래시 메모리는 소형 크기, 낮은 전력소모, 고속 및 고 신뢰성 특징들 때문에 이동 및 휴대용 환경에서 이상적으로 적합하다.
- [0006] EEPROM 및 전기적으로 프로그램가능한 판독 전용 메모리(EPROM)는 소거될 수 있고 이들의 메모리 셀들에 새로운 데이터가 기입 또는 "프로그램"되게 할 수 있는 비휘발성 메모리이다. 이들 둘 다는 소스영역과 드레인 영역 간에 반도체 기관의 채널 영역 내 채널 영역 상에 위치되는, 전계 효과 트랜지스터 구조에서 플로우팅(비접속된) 도전성 게이트를 이용한다. 이때, 제어 게이트는 플로우팅 게이트 위에 제공된다. 트랜지스터의 임계 전압 특성은 플로우팅 게이트 상에 보존되는 전하량에 의해 제어된다. 즉, 플로우팅 게이트 상에 소정 레벨의 전하에 대해서, 소스영역과 드레인 영역 간에 도통시키기 위하여 트랜지스터가 턴 "온"되기 전에 제어 게이트에 인가되어야 하는 대응하는 전압(임계)이 있다.
- [0007] 플로우팅 게이트는 한 범위의 전하들을 유지할 수 있고, 따라서 임계 전압 윈도우 내에서 임의의 임계 전압 레벨로 프로그램될 수 있다. 임계 전압 윈도우의 크기는 플로우팅 게이트에 프로그램될 수 있는 전하들의 범위에 대응하게 될 장치의 최대 및 최소 임계 레벨들에 의해 경계가 정해진다. 임계 윈도우는 일반적으로 메모리 장치의 특성들, 동작 조건들 및 이력(history)에 좌우된다. 윈도우 내에서 각각의 구별되는 분해가능한(resolvable) 임계전압 레벨 범위는 원리적으로 셀의 유한 메모리 상태를 지정하는데 사용될 수 있다.
- [0008] 메모리 셀로서 작용하는 트랜지스터는 전형적으로 두 메커니즘들 중 한 메커니즘에 의해 "프로그램된" 상태로 프로그램된다. "핫(hot) 전자 주입"에서, 드레인에 인가되는 고전압은 기관 채널 영역을 횡단하는 전자들을 가속시킨다. 동시에, 제어 게이트에 인가되는 고전압은 얇은 게이트 유전체를 통해서 플로우팅 게이트 로 핫 전자들을 가져간다. "터널링 주입"에서, 고전압은 기관에 대해서 제어 게이트로 인가된다. 이에 따라, 전자들은 기관으로부터 개재된 플로우팅 게이트로 당겨진다.
- [0009] 메모리 장치는 다수의 메커니즘에 의해 소거될 수 있다. EPROM에 있어서, 메모리는 자외선 조사에 의해 플로우팅 게이트로부터 전하를 제거함으로써 대량으로 소거될 수 있다. EEPROM에 있어서, 얇은 산화물을 통해서 기관 채널 영역으로 터널링하기 위하여(즉, 파울러-노다임 터널링) 플로우팅 게이트에서 전자들을 유도하도록 메모리 셀은 고전압을 제어 게이트에 대해서 기관에 인가함으로써 전기적으로 소거될 수 있다. 전형적으로, EEPROM은 한 바이트씩 소거될 수 있다. 플래시 EEPROM에 있어서, 메모리는 한꺼번에 또는 한번에 하나 이상의 블록들을 전기적으로 소거될 수 있는데, 여기서 블록은 메모리의 512 바이트들 또는 그 이상으로 구성될 수 있다.

발명의 상세한 설명

- [0010] 비휘발성 메모리 셀들의 예들
- [0011] 메모리 장치들은 전형적으로 카드상에 설치될 수 있는 하나 이상의 메모리 칩들을 포함한다. 각 메모리 칩은 디코더들 및 소거, 기입 및 판독 회로들과 같은 주변 회로들에 의해 지원되는 메모리 셀들의 어레이를 포함한다. 더욱 복잡한 메모리 장치들은 지능적이고 더 높은 레벨의 메모리 동작들 및 인터페이싱을 수행하는 제어기에 따른다. 오늘날 사용되는 많은 상업적으로 성공한 비휘발성 고체상태 메모리 장치들이 존재한다. 이들 메모리 장치들은 다양한 유형들의 메모리 셀들을 사용할 수 있는데, 각 유형은 하나 이상의 전하 저장 소자를 갖는다.

- [0012] 도 1a 내지 도 1e는 비휘발성 메모리 셀들의 다양한 예들을 개요적으로 도시한 것이다.
- [0013] 도 1a는 전하를 저장하기 위한 플로우팅 게이트를 갖는 EEPROM 셀 형태의 비휘발성 메모리를 개요적으로 도시한 것이다. 전기적으로 소거가능하고 프로그램가능한 판독 전용 메모리(EEPROM)는 EPROM과 유사한 구조를 갖지만 UV 조사에 노출될 필요없이 적절한 전압들의 인가시 플로우팅 게이트로부터 전기적으로 전하를 로딩하고 제거하는 메커니즘을 부가적으로 제공한다. 이들을 제조하는 이와 같은 셀들 및 방법들의 예들이 미국 특허 5,595,924에 제공된다.
- [0014] 도 1b는 선택 게이트 및 제어 또는 스티어링(steering) 게이트 둘 다를 갖는 플래시 EEPROM 셀을 개요적으로 도시한 것이다. 메모리 셀(10)은 소스(14)와 드레인(16) 확산들 사이에 "스플릿-채널(split-channel)"(12)을 갖는다. 셀은 직렬의 2개의 트랜지스터들(T1, T2)로 효율적으로 형성된다. T1은 플로우팅 게이트(20) 및 제어 게이트(30)를 갖는 메모리 트랜지스터로서 작용한다. 플로우팅 게이트는 선택가능한 전하량을 저장할 수 있다. 채널의 T1의 부분을 통해서 흐를 수 있는 전류량은 제어 게이트(30) 상의 전압과 개재된 플로우팅 게이트(20) 상에 존재하는 전하량에 좌우된다. T2는 선택 게이트(40)를 갖는 선택 트랜지스터로서 작용한다. T2가 선택 게이트(40)에서 전압에 의해 턴 온 될 때, 이는 채널의 T1 부분에서 전류를 소스와 드레인 사이를 통과시킨다. 선택 트랜지스터는 제어 게이트의 전압에 관계없이 소스-드레인 채널을 따라서 스위치를 제공한다. 한가지 이점은 플로우팅 게이트들에서 전하 공핍(포지티브)으로 인해 제로 제어 게이트 전압에서도 여전히 도통하는 이들 셀들을 턴 오프시키도록 사용될 수 있다는 것이다. 다른 이점은 소스측 주입 프로그래밍이 더욱 손쉽게 구현되도록 한다는 것이다.
- [0015] 스플릿-채널 메모리 셀의 한 가지 간단한 실시예는 선택 게이트 및 제어 게이트가 도 1b에 도시된 점선으로 개요적으로 도시된 바와 같은 동일한 워드 라인에 접속되는 경우이다. 이는 채널의 일부분 위에 위치되는 전하 저장 소자(플로우팅 게이트), 그리고 다른 채널 부분뿐만 아니라 전하 저장 소자 위에 위치되는 제어 게이트 구조(이는 워드 라인의 부분이다)를 가짐으로써 달성된다. 이는 직렬의 2개의 트랜지스터들을 갖는 셀을 유효하게 형성하는데, 한 트랜지스터(메모리 트랜지스터) 전하 저장 소자 상의 전하량과 채널 부분을 통해서 흐를 수 있는 전류량을 제어하는 워드 라인 상의 전압과의 조합이고, 다른 한 트랜지스터(선택 트랜지스터)는 게이트로서 작용하는 워드 라인 만을 갖는다. 이와 같은 셀들의 예들, 메모리 시스템들에서 이들의 용도들 및 이들을 제조하는 방법들이 미국 특허들 5,070,032, 5,095,344, 5,315,541, 5,343,063, 및 5,661,053에 개시되어 있다.
- [0016] 도 1b에 도시된 스플릿-채널 셀의 더욱 자세한 실시예는 선택 게이트 및 제어 게이트가 독립적이고 이들 간에 점선으로 연결되지 않았을 때이다. 한 구현은 워드라인에 수직한 제어 (또는 스티어링) 라인에 접속되는 셀들의 어레이에서 하나의 컬럼(column)의 제어 게이트들을 갖는다. 이 효과는 선택된 셀을 판독 또는 프로그래밍할 때와 동시에 워드 라인이 2가지 기능들을 수행해야 하는 것을 경감시킨다는 것이다. 이들 2가지 기능들은 (1) 선택 트랜지스터를 턴 온 또는 턴 오프시키는 적절한 전압을 필요로 하게 되는 것으로 선택 트랜지스터의 게이트로서 작용하는 것과, (2) 워드라인과 전하 저장 소자간의 전계(용량성) 커플링을 통해서 원하는 레벨로 전하 저장 소자의 전압을 구동시키는 것이다. 단일 전압에 의해 최적의 방식으로 이들 기능들 둘 다를 수행하는 것은 어렵다. 제어 게이트 및 선택 게이트의 개별적 제어에 의해, 워드라인은 기능(1)만을 수행하는 한편, 부가된 제어 라인이 기능(2)을 수행하도록 한다. 이 성능은 타깃으로 하는 데이터에 맞게 프로그래밍 전압이 조정되는 고성능 프로그래밍의 설계를 할 수 있게 한다. 플래시 EEPROM 어레이에서 독립적인 제어(또는 스티어링) 게이트들의 사용이 예를 들면 미국 특허 5,313,421 및 6,222,762에 기술되어 있다.
- [0017] 도 1c는 듀얼 플로우팅 게이트들 및 독립적인 선택 및 제어 게이트들을 갖는 또 다른 플래시 EEPROM 셀을 개요적으로 도시한 것이다. 메모리 셀(10)은 유효하게 직렬의 3개의 트랜지스터들을 갖는다는 것을 제외하면 도 1b의 메모리 셀과 유사하다. 이 유형의 셀에서, 2개의 저장 소자들(즉, T1-좌(left) 및 T1-우(right)의 저장 소자)는 이들 간에 선택 트랜지스터(T1)와 함께 소스 및 드레인 확산들 간의 채널 위에 포함된다. 메모리 트랜지스터들은 각각, 플로우팅 게이트들(20, 20') 및 제어 게이트들(30, 30')을 갖는다. 선택 트랜지스터(T2)는 선택 게이트(40)에 의해 제어된다. 어떤 한 시간에, 한 쌍의 메모리 트랜지스터들 중 단지 한 트랜지스터가 판독 또는 기입을 위하여 액세스된다. 저장 유닛 T1-좌가 액세스될 때, T2 및 T1-우 둘 다는 턴 온 되어 채널의 T1-좌의 부분에서 전류를 소스와 드레인 간을 통과시킨다. 유사하게, 저장 유닛 T1-우가 액세스되고 있을 때, T2 및 T1-좌는 턴 온 된다. 소거는 플로우팅 게이트 내에 저장된 전자들이 선택 게이트 폴리실리콘으로 터널링될 수 있도록, 선택 게이트 폴리실리콘의 부분을 플로우팅 게이트에 근접하게 하고 상당한 포지티브 전압(예를 들어, 20V)을 선택 게이트에 인가함으로써 달성된다.
- [0018] 도 1d는 NAND 셀로 구성된 메모리 셀들의 스트링을 개요적으로 도시한 것이다. NAND 셀(50)은 소스들 및 드레인

들에 의해 데이지-체인되는(daisy-chained) 일련의 메모리 트랜지스터들(M1, M2, ..., Mn)(n=4, 8, 16 또는 그 이상)로 이루어진다. 한 쌍의 선택 트랜지스터들(S1, S2)은 NAND 셀의 소스 단자(54) 및 드레인 단자(56)를 통해서 외부로 메모리 트랜지스터를 체인의 접속을 제어한다. 메모리 어레이에서, 소스 선택 트랜지스터(S1)가 턴 온 될 때, 소스 단자는 소스 라인에 결합된다. 유사하게, 드레인 선택 트랜지스터(S2)가 턴 온 될 때, NAND 셀의 드레인 단자는 메모리 어레이의 비트라인에 결합된다. 이 체인의 각 메모리 트랜지스터는 의도된 메모리 상태를 나타내도록 주어진 전하량을 저장하기 위하여 전하 저장 소자를 갖는다. 각 메모리 트랜지스터의 제어 게이트는 판독 및 기입 동작들에 대해 제어한다. 선택 트랜지스터들(S1, S2) 각각의 제어 게이트는 소스 단자(54) 및 드레인 단자(56) 각각을 통해서 NAND 셀에 제어 액세스를 제공한다.

[0019] NAND 셀 내의 어드레스된 메모리 트랜지스터가 프로그래밍 동안 판독되어 검증될 때, 이의 제어 게이트는 적절한 전압을 공급받는다. 동시에, NAND 셀(50) 내 비어드레스된 메모리 트랜지스터들의 나머지는 자신들의 제어 게이트들 상에 충분한 전압을 인가함으로써 완전히 턴 온 된다. 이 방식으로, 개별적인 메모리 트랜지스터의 소스로부터 NAND 셀의 소스 단자(54)까지의 도통 경로가 효율적으로 생성되고 마찬가지로 이 셀의 개별적인 메모리 트랜지스터의 드레인으로부터 드레인 단자(56)까지의 도통 경로가 효율적으로 생성된다. 이와 같은 NAND 셀 구조들을 갖는 메모리 장치들이 미국 특허 5,570,315, 5,903,495, 6,046,935에 개시되어 있다.

[0020] 도 1e는 전하를 저장하기 위한 유전층을 갖는 비휘발성 메모리를 개요적으로 도시한 것이다. 앞서 설명된 도전성 플로우팅 게이트 소자들 대신에, 유전층이 사용된다. 유전성 저장 소자를 이용하는 이와 같은 메모리 장치들은 Eitan 등이 2000년 11월 IEEE Electron Device Letters, vol. 21, no. 11, pp. 543-545호에 발표한 "NROM: A Novel Localized Trapping, 2-Bit Nonvolatile Memory Cell"에 개시되어 있다. ONO 유전층은 소스 및 드레인 확산들 간의 채널에 걸쳐서 확장되어 있다. 하나의 데이터 비트를 위한 전하는 드레인에 인접한 유전층에 국부화(local) 되고 다른 데이터 비트를 위한 전하는 소스에 인접한 유전층에 국부화 된다. 예를 들어, 미국 특허 5,768,192 및 6,011,725는 2개의 이산화규소층들 사이에 개재된 트래핑 유전체(trapping dielectric)를 갖는 비휘발성 메모리 셀을 개시한다. 복수-상태 데이터 저장 장치는 유전체에 공간적으로 분리된 전하 저장 영역들의 2진 상태들을 개별적으로 판독함으로써 구현된다.

[0021] 메모리 어레이

[0022] 메모리 장치는 전형적으로, 행들(row) 및 컬럼들(column)로 배열되고 워드라인들 및 비트라인들에 의해 어드레스가능한 메모리 셀들의 2차원 어레이로 이루어진다. 이 어레이는 NOR 형 또는 NAND 형 아키텍처에 따라서 형성될 수 있다.

[0023] NOR 어레이

[0024] 도 2는 메모리 셀들의 NOR 어레이의 예를 도시한다. NOR형 아키텍처를 갖는 메모리 장치들은 도 1b 또는 도 1c에 도시된 유형의 셀들로 구현되었다. 메모리 셀들의 각각의 행은 데이지-체인 방식으로 자신들의 소스들 및 드레인들에 의해 접속된다. 이 설계는 가상 그라운드 설계라고도 한다. 각 메모리 셀(10)은 소스(14), 드레인(16), 제어 게이트(30) 및 선택 게이트(40)를 갖는다. 한 행의 셀들은 워드라인(42)에 접속된 자신들의 선택 게이트들을 갖는다. 컬럼의 셀들은 선택된 비트라인들(34, 36)에 각각 접속되는 소스들 및 드레인들을 갖는다. 메모리 셀들이 독립적으로 제어되는 제어 게이트 및 선택 게이트를 갖는 일부 실시예들에서, 스티어링 라인(30)은 또한 컬럼 내의 셀들의 제어 게이트들을 접속한다.

[0025] 많은 플래시 EEPROM 장치들은 메모리 셀들로 구현되는데, 각 메모리 셀은 이의 제어 게이트 및 선택 게이트가 함께 접속되어 형성되어 있다. 이 경우에, 각 행을 따라서 셀들의 제어 게이트들 및 선택 게이트들 모두를 단지 접속하는 워드라인 및 스티어링 라인들이 필요로 되지 않는다. 이들 설계들의 예들은 미국 특허 5,172,338 및 5,418,752에 개시되어 있다. 이들 설계들에서, 워드라인은 행 선택과, 그리고 제어 게이트 전압을 판독 또는 프로그래밍하기 위하여 행 내 모든 셀들에 공급하는 것인, 2가지 기능들을 필수적으로 수행한다.

[0026] NAND 어레이

[0027] 도 3은 도 1d에 도시된 것과 같은 메모리 셀들의 NAND 어레이의 예를 도시한 것이다. NAND 셀들의 각 컬럼을 따라서, 비트라인은 각 NAND 셀의 드레인 단자(56)에 결합된다. NAND 셀들의 각 행을 따라서, 소스 라인은 이들의 모든 소스 단자들(54)을 접속시킬 수 있다. 또한, 행을 따른 NAND 셀들의 제어 게이트들은 일련의 대응하는 워드라인들에 접속된다. NAND 셀들의 전체 행은 접속된 워드라인들을 통해서 자신들의 제어 게이트들 상의 적절한 전압들에 의해 한 쌍의 선택 트랜지스터들(도 1d 참조)을 턴 온시킴으로써 어드레스될 수 있다. NAND 셀의 체인 내에 메모리 트랜지스터가 판독될 때, 체인을 통해 흐르는 전류가 필수적으로, 판독되는 셀에 저장된 전하 레벨

에 좌우되도록, 이 체인 내의 나머지 메모리 트랜지스터들은 자신들의 연관된 워드 라인들을 통해서 좀처럼 턴 온 되지 않다. 메모리 시스템의 일부로서 NAND 아키텍처 어레이 및 이의 동작의 예는 미국특허 5,570,315, 5,774,397 및 6,046,935에서 알 수 있다.

[0028] 블록 소거

[0029] 전하 저장 메모리 장치들의 프로그래밍은 자신의 전하 저장 소자들에 더 많은 전하를 추가하는 결과가 될 수 있을 뿐이다. 그러므로 프로그램 동작 전, 전하 저장 소자의 기존 전하는 제거(또는 소거)되어야만 된다. 소거 회로들(도시되지 않음)은 메모리 셀들의 하나 이상의 블록들을 소거하도록 제공된다. EEPROM과 같은 비휘발성 메모리는 셀들의 전체 어레이 또는 이 어레이의 셀들의 상당한 그룹들이 모두 전기적으로(즉, 플래시로) 소거될 때 "플래시" EEPROM 이라 칭한다. 일단 소거되면, 셀들의 그룹은 재프로그램될 수 있다. 함께 소거가능한 셀들의 그룹은 하나 이상의 어드레스가능한 소거 단위로 구성될 수 있다. 소거 단위 또는 블록은 전형적으로 데이터의 하나 이상의 페이지들을 저장하는데, 이 페이지는 프로그래밍 및 판독 단위이지만, 하나 이상의 페이지가 단일 동작으로 프로그램되거나 판독될 수도 있다. 각 페이지는 전형적으로, 하나 이상의 데이터 섹터들을 저장하는데, 이 섹터의 크기는 호스트 시스템에 의해 규정된다. 일 예는 자기 디스크 드라이브들에 확립된 표준에 따라, 한 섹터의 512 바이트 사용자 데이터 및 이에 더하여 사용자 데이터 및/또는 이를 저장하는 블록에 관한 몇 바이트들의 오버헤드 정보이다.

[0030] 판독/기입 회로들

[0031] 통상의 2상태 EEPROM 셀에서, 적어도 하나의 전류 구분점(breakpoint) 레벨은 2개의 영역들로 도통 윈도우를 분할하도록 설정된다. 셀이 소정의 고정된 전압들을 인가함으로써 판독될 때, 이의 소스/드레인 전류는 구분점 레벨(또는 기준 전류 (I_{REF}))과 비교됨으로써 메모리 상태로 결정된다. 판독된 전류가 구분점 레벨의 전류보다 높으면, 셀은 한 논리 상태(예를 들면, "제로" 상태)에 있는 것으로 판정된다. 한편, 전류가 구분점 레벨의 전류보다 작으면, 이 셀은 다른 논리 상태(예를 들면, "1" 상태)에 있는 것으로 판정된다. 따라서, 이와 같은 2 상태 셀은 1비트의 디지털 정보를 저장한다. 외부에서 프로그램될 수 있는 기준 전류원은 종종 구분점 레벨 전류를 발생시키기 위한 메모리 시스템의 부분으로서 제공된다.

[0032] 메모리 용량을 증가시키기 위하여, 플래시 EEPROM 장치들은 반도체 기술 상태들이 진보됨에 따라서 점점 더 고밀도로 제조되고 있다. 저장 용량을 증가시키는 또 다른 방법은 각 메모리 셀이 2 이상의 상태들을 저장하게 하는 것이다.

[0033] 복수-상태 또는 복수-레벨 EEPROM 메모리 셀에 대해서, 도통 윈도우는 하나 이상의 구분점에 의해 2개 이상의 영역들로 분할됨으로써, 각 셀이 하나 이상의 데이터 비트를 저장할 수 있도록 한다. 주어진 EEPROM 어레이가 저장할 수 있는 정보는 각 셀이 저장할 수 있는 상태들의 수에 따라서 증가된다. 복수-상태 또는 복수-레벨 메모리 셀들을 갖는 EEPROM 또는 플래시 EEPROM이 미국 특허 5,172,338호에 개시되어 있다.

[0034] 실제로, 셀의 메모리 상태는 통상적으로 기준 전압이 제어 게이트에 인가될 때 셀의 소스 전극과 드레인 전극들을 지나는 도통 전류를 감지함으로써 판독된다. 따라서, 셀의 플로우팅 게이트 상에 각각 주어진 전하에 대해서, 고정된 기준 제어 게이트 전압에 대한 대응하는 도통 전류가 검출될 수 있다. 유사하게, 플로우팅 게이트 상으로 프로그램가능한 전하 범위는 대응하는 임계 전압 윈도우 또는 대응하는 도통 전류 윈도우를 규정한다.

[0035] 대안적으로, 분할된 전류 윈도우 중 도통 전류를 검출하는 대신에, 제어 게이트에서 테스트 하에 소정 메모리 상태에 대한 임계 전압을 설정하거나 도통 전류가 임계 전류보다 낮거나 높은지를 검출하는 것이 가능하다. 한 구현에서, 임계 전류에 대한 도통 전류의 검출은 도통 전류가 비트 라인 용량을 통해서 방전하는 레이트를 검사함으로써 달성된다.

[0036] 도 4는 플로우팅 게이트가 선택적으로 한번에 저장할 수 있는 4개의 상이한 전하들(Q1-Q4)에 대한 제어 게이트 전압(V_{CG})과 소스-드레인 전류(I_D)간의 관계를 도시한다. 4개의 실선 I_D 대 V_{CG} 곡선들은 4개의 가능한 메모리 상태들에 각각 대응하는 메모리 셀의 플로우팅 게이트 상에 프로그램될 수 있는 4개의 가능한 전하 레벨들을 나타낸다. 예로서, 셀들의 모집단의 임계 전압 윈도우는 0.5V 내지 3.5V 범위일 수 있다. 6개의 메모리 상태들은 임계 윈도우를 0.5V 간격으로 5개의 영역들로 임계 윈도우를 분할함으로써 경계가 정해질 수 있다. 예를 들면, 2 μA 의 기준 전류(I_{REF})가 도시된 바와 같이 사용된다면, Q1으로 프로그램되는 셀은 이의 곡선이 $V_{CG}=0.5V$ 및 1.0V로 경계가 정해진 임계 윈도우의 영역에서 I_{REF} 와 교차하기 때문에 메모리 상태 "1"에 있는 것으로 간주될 수 있

다. 유사하게, Q4는 메모리 상태 "5"에 있다.

- [0037] 상기 설명으로부터 알 수 있는 바와 같이, 메모리 셀이 저장할 상태들이 많으면 많을수록, 임계 윈도우는 더욱 세밀하게 분할된다. 이는 필요로 되는 분해능을 달성할 수 있도록 프로그래밍 및 관독 동작들의 높은 정밀도를 필요로 할 것이다.
- [0038] 미국특허 4,357,685는 셀이 주어진 상태로 프로그램될 때, 증분적 전하를 플로팅 게이트에 부가할 때마다, 연속적인 프로그래밍 전압 펄스들이 가해지는 2상태 EPROM을 프로그램하는 방법을 개시한다. 펄스들 간에서, 구분점 레벨에 대한 소스-드레인 전류를 판정하기 위해서 셀은 다시 관독되거나 검증된다. 전류 상태가 원하는 상태에 도달한 것으로 검증될 때 프로그래밍은 중단된다. 사용된 프로그래밍 펄스 트레인(train)은 증가하는 기간(period) 또는 진폭을 가질 수 있다.
- [0039] 종래 기술의 프로그래밍 회로들은 단순히, 타겟 상태에 도달될 때까지 소거된 또는 그라운드 상태부터 임계 윈도우를 단계적으로 거쳐(step through) 프로그래밍 펄스들을 인가한다. 실제로, 적절한 분해능을 허용하기 위해서, 각 분할된 또는 경계가 정해진 영역은 거처가기 위해서 적어도 약 5개의 프로그래밍 단계들을 필요로 할 것이다. 이 수행은 2상 메모리 셀들엔 수락될 수 있다. 그러나 복수-상태 셀들에 있어서, 필요로 되는 단계들의 수는 분할들의 수에 따라서 증가하므로, 프로그래밍 정밀도 또는 분해능이 증가되어야 한다. 예를 들면, 16 상태 셀은 타겟 상태로 프로그램하기 위하여 평균적으로 적어도 40 프로그래밍을 필요로 할 수 있다.
- [0040] 도 5는 행 디코더(130) 및 컬럼 디코더(160)를 통해서 관독/기입 회로들(170)에 의해 액세스될 수 있는 메모리 어레이(100)의 전형적인 배열을 갖는 메모리 장치를 개요적으로 도시한 것이다. 도 2 및 도 3과 관련하여 설명된 바와 같이, 메모리 어레이(100) 내의 메모리 셀의 메모리 트랜지스터는 한 세트의 선택된 워드라인(들) 및 비트라인(들)을 통해서 어드레스될 수 있다. 행 디코더(130)는 하나 이상의 워드라인들을 선택하고 컬럼 디코더(160)는 하나 이상의 비트라인들을 선택하여, 적절한 전압들을 어드레스된 메모리 트랜지스터의 각 게이트들에 인가한다. 관독/기입 회로들(170)은 어드레스된 메모리 트랜지스터들의 메모리 상태들을 관독 또는 기입(프로그래밍)하도록 제공된다. 관독/기입 회로들(170)은 비트라인들을 통해서 어레이 내 메모리 소자들에 접속될 수 있는 다수의 관독/기입 모듈들을 포함한다.
- [0041] 도 6a는 개개의 관독/기입 모듈(190)의 개요적 블록도이다. 근본적으로, 관독 또는 검출 동안에, 감지 증폭기는 선택된 비트라인을 통해 접속된 어드레스된 메모리 트랜지스터의 드레인을 통해 흐르는 전류를 판정한다. 전류는 메모리 트랜지스터에 저장된 전하 및 이의 제어 게이트 전압에 따른다. 예를 들면, 복수-상태 EEPROM 셀에서, 이의 플로팅 게이트는 몇개의 서로 다른 레벨들 중 하나에 충전될 수 있다. 4-레벨 셀에 있어서, 이것은 2비트의 데이터를 저장하는데 사용될 수 있다. 감지 증폭기에 의해 검출된 레벨은 레벨-비트 전환 로직에 의해, 데이터 래치에 저장될 한 세트의 데이터 비트들로 전환된다.
- [0042] 관독 기입 성능 및 정확도에 영향을 미치는 요인들
- [0043] 관독 및 프로그램 성능을 개선시키기 위하여, 어레이 내의 복수의 전하 저장 소자들 또는 메모리 트랜지스터들은 병렬로 관독되거나 프로그램된다. 따라서, 메모리 소자들의 논리적 "페이지"는 함께 관독되거나 프로그램된다. 기존 메모리 아키텍처들에서, 한 행은 전형적으로 몇개의 인터리빙된 페이지들을 포함한다. 한 페이지의 모든 메모리 소자들은 함께 관독되거나 프로그램될 것이다. 컬럼 디코더는 인터리빙된 페이지들 중 각 페이지를 대응하는 수의 관독/기입 모듈들에 선택적으로 접속시킬 것이다. 예를 들면, 한 구현에서, 메모리 어레이는 532 바이트들(512 바이트들과 이에 더하여 20 바이트의 오버헤드들)의 페이지 크기를 갖도록 설계된다. 각 컬럼이 드레인 비트 라인을 포함하고 행당 2개의 인터리빙된 페이지들을 포함한다면, 이는 8512 컬럼들에 이르고, 각 페이지는 4256 컬럼들과 관련된다. 모든 우수 비트 라인들 또는 기수 비트 라인들 중 하나와 병렬로 관독 또는 기입하도록 접속될 수 있는 4256 감지 모듈들이 존재하게 될 것이다. 이 방식으로, 병렬로 데이터의 4256 비트들(즉, 532 바이트들)의 페이지는 메모리 소자들의 페이지로부터 관독되거나 이 페이지에 프로그램된다. 관독/기입 회로들(170)을 형성하는 관독/기입 모듈들은 다양한 아키텍처들로 배열될 수 있다.
- [0044] 도 5를 참조하면, 관독/기입 회로들(170)은 뱅크들의 관독/기입 스택들(180)로 구성된다. 각각의 관독/기입 스택(180)은 한 스택의 관독/기입 모듈들(190)이다. 메모리 어레이에서, 컬럼 간격은 이를 점유하는 하나 또는 2개의 트랜지스터들의 크기에 의해 결정된다. 그러나 도 6a로부터 알 수 있는 바와 같이, 관독/기입 모듈의 회로는 더 많은 트랜지스터들 및 회로 요소들로 구현될 것이며 따라서 많은 컬럼들 상에 공간을 점유할 것이다. 점유된 컬럼들 중에 하나 이상의 컬럼에 사용하기 위해서, 복수의 모듈들은 수직으로 적층된다.
- [0045] 도 6b는 한 스택의 관독/기입 모듈들(190)에 의해 종래에 구현되는 도 5의 관독/기입 스택을 도시한 것이다. 예

를 들면, 판독/기입 모듈은 16개의 컬럼들 상에 확장할 수 있고, 그러면 한 스택의 8개의 판독/기입 모듈들을 가진 판독/기입 스택(180)은 8개의 컬럼들에 병렬로 사용하기 위해 사용될 수 있다. 판독/기입 스택은 컬럼 디코더를 통해뱅크 중에 8개의 기수 (1, 3, 5, 7, 9, 11, 13, 15) 컬럼들 또는 8개의 우수 (2, 4, 6, 8, 10, 12, 14, 16) 컬럼들에 결합될 수 있다.

[0046] 앞에 언급된 바와 같이, 종래의 메모리 장치들은 한번에 모든 우수 또는 모든 기수 비트 라인들 상에서 대량 병렬 방식으로 동작됨으로써 판독/기입 동작들을 개선한다. 한 행이 2개의 인터리빙된 페이지들로 구성되는 이러한 아키텍처는 판독/기입 회로들의 블록에 맞추어야 하는 문제를 경감시키도록 도울 것이다. 또한, 이것은 비트-라인간 용량성 커플링을 제어하는 것을 고려한다. 블록 디코더는 우수 페이지 또는 기수 페이지 중 하나에 한 세트의 판독/기입 모듈들을 멀티플렉스하도록 사용된다. 이 방식으로, 한 세트 비트 라인들이 판독되거나 프로그램될 때마다, 인터리빙 세트는 바로 이웃한 인접 커플링을 최소화하게 그라운드될 수 있다.

[0047] 그러나 인터리빙 페이지 아키텍처는 적어도 3가지 면에서 단점이 있다. 첫째, 이는 부가적인 멀티플렉싱 회로를 필요로 한다. 두 번째 이는 수행성능면에서 느리다. 워드 라인에 의해 또는 행으로 접속되는 메모리 셀들의 판독 또는 프로그램을 완료하기 위하여, 2개의 판독 또는 2개의 프로그램 동작들이 필요로 된다. 세 번째, 2개의 이웃들이 이를테면 기수 및 우수 페이지들에서 개별적으로와 같이, 상이한 시간들에서 프로그램될 때 플로팅 게이트 레벨에서 이웃하는 전하 저장 소자들간의 필드 커플링(field coupling)과 같은 다른 교란 영향들을 해결함에 있어 최적이지 않다.

[0048] 이웃 필드 커플링의 문제는 메모리 트랜지스터들간에 훨씬 더 밀접한 간격으로 더욱 더 명백해진다. 메모리 트랜지스터에서, 전하 저장소자는 채널 영역과 제어 게이트간에 개재된다. 채널 영역에 흐르는 전류는 제어 게이트와 전하 저장소자에서의 전계에 의해 기여된 결과적인 전계의 함수이다. 훨씬 증가하는 밀도로, 메모리 트랜지스터들은 서로 더 가깝게 형성된다. 그러면 이웃한 전하소자들로부터의 전계는 영향받는 셀의 결과적인 필드에의 현저한 기여자(contributor)가 된다. 이웃한 필드는 이웃들의 전하 저장소자들에 프로그램된 전하에 따른다. 이러한 교란하는 전계는 이웃들의 프로그램된 상태들에 따라 변하기 때문에 사실상 동적이다. 이에 따라, 영향받는 셀은 이웃들의 변하는 상태들에 따라 다른 시간에 다르게 판독될 수 있다.

[0049] 페이지를 인터리빙하는 종래의 아키텍처는 이웃한 플로팅 게이트 커플링에 의해 야기되는 오류를 더욱 심하게 한다. 우수 페이지 및 기수 페이지가 서로 무관하게 프로그램되고 판독되기 때문에, 페이지는 한 세트의 조건 하에 프로그램될 수 있으나, 한편으로 인터리빙 페이지에 일어난 것에 따라, 완전히 다른 한 세트의 조건 하에 다시 판독될 수 있다. 판독 오류들은 증가하는 밀도로 더욱 심해질 것이며, 복수-상태 구현을 위한 임계 윈도우의 보다 정확한 판독 동작 및 간격이 넓은 분할을 요구한다. 성능은 악화할 것이며 복수-상태 구현에서 잠재적 용량은 제한된다.

[0050] 미국특허공개 US-2004-0060031-A1는 대응하는 한 블록의 메모리 셀들을 병렬로 판독 및 기입하기 위해 큰 블록의 판독/기입 회로들을 구비한 고성능이면서도 콤팩트한 비휘발성 메모리 장치를 개시한다. 특히, 메모리 장치는 한 블록의 판독/기입 회로들에 용장성을 최소한으로 감소시키는 아키텍처를 갖는다. 전력뿐만 아니라 공간에서 현저한 절약은 한 블록의 판독/기입 모듈들을, 상당히 더 작은 공통 부분들의 세트들과 시간-다중화식으로 상호작용하면서 병렬로 동작하는 한 블록의 판독/기입 모듈 핵심 부분들에 재분포시킴으로써 달성된다. 특히, 복수의 감지 증폭기들과 데이터 래치들간에 판독/기입 회로들간에 데이터 처리는 공유된 프로세서에 의해 수행된다.

[0051] 그러므로 고성능 및 고용량 비휘발성 메모리에 대한 일반적인 필요성이 있다. 특히, 콤팩트하고 효율적인, 그러면서도 판독/기입 회로들간에 데이터를 처리에 있어 매우 융통성이 있는 개선된 프로세서를 구비한 향상된 판독 및 프로그램 성능을 가진 콤팩트한 비휘발성 메모리에 대한 필요성이 있다.

[0052] 발명의 요약

[0053] 본 발명은 NAND 스트링 저항효과를 감소시키고, 메모리 내구성 및 신뢰성을 증가시킬 뿐만 아니라, 프로그램 교란 또는 사용자 판독 교란을 야기할 수 있는 특성의 데이터 패턴들을 감소 또는 제거할 것이다. 또한, 플로팅 게이트 간 커플링의 문제를 감소시킬 것이다.

[0054] 본 발명의 일 면은 플래시 메모리 칩 또는 어레이에 저장된 데이터의 의사-랜덤 화를 구현하는 방법을 포함한다. 이 방법은 플래시 메모리의 회로에 대한 최소의 수정만을 요구하며, 그러면서도 이와 아울러 데이터 저장의 랜덤화 대폭 증가시킬 것이며, 그러므로 어레이의 성능을 증가시킬 것이다. 데이터의 랜덤화는 메모리 내에서 쉽게 활성화 또는 비활성화될 수 있다. 또한, 의사 랜덤화를 구동하는 시퀀스는 계속적으로 바뀔 수 있

어, 시스템의 융통성을 더 증가시킨다.

- [0055] 일부 실시예들에서, 사용자 작용은 데이터의 엔코딩에 잠재적 변화를 트리거한다. 사용자 작용의 타이밍은 완전히 예측 불가능하기 때문에, 사용자에게 의해 구동되는 트리거의 결과로서 이용되는 엔코딩 방식 또한 예측할 수 없고 근본적으로 전적으로 랜덤하다.
- [0056] 다양한 랜덤화 방법론들 및 실시예들은 메모리 칩(EEPROM)에 구현된다. 이것은 이들이 칩과 통신하는 메모리 제어기로 일어나는 것이 아니라 메모리 칩 자체 내에서 일어남을 말해준다. 이것은 흔히 시스템 레벨에서 구현되고 데이터가 메모리 칩 내에 저장되는 방법을 변경하기 위해 제어기를 이용하는 마모 레벨링(wear leveling)으로서 알려진 문제를 해결하는 다른 기술과는 다르다.
- [0057] 발명의 또 다른 면은 복수의 페이지들의 사용자 데이터를 포함하는 비휘발성 NAND 플래시 메모리 칩이다. 칩은 메모리 어레이, 유한 상태 머신 클럭신호, 명령 클럭신호, 및 유한 상태 머신 클럭신호와 명령 클럭신호를 기준으로 발생하는 극성 비트를 포함한다. 제 1 엔코딩 유형은 극성 비트의 제 1 값에 대해 사용되며, 제 2 엔코딩 유형은 극성 비트의 제 2 값에 대해 사용된다. 서로 다른 엔코딩 유형들의 이용은 같은 데이터가 실제로 반복적으로 또는 장기간동안 어레이 내에 저장될지라도, 어레이 내 동일 데이터 패킷의 저장 기간을 감소시킨다.
- [0058] 발명의 또 다른 면은 복수의 페이지들의 데이터를 포함하는 비휘발성 NAND 플래시 메모리에서의 방법에 관한 것이다. 방법은, 비휘발성 메모리 내에, 복수의 비트들을 포함하는 코드를 저장하는 것을 포함하며, 이 복수의 비트들의 각각의 비트는 한 페이지의 복수의 비트들의 데이터의 엔코딩이 반전될 것인지 아니면 그대로 둘 것인지를 규정한다. 또한, 이것은 비휘발성 메모리로부터 코드를 판독하는 것과, 코드의 복수의 비트들을 저장하기 위해 할당된 레지스터에 비휘발성 메모리로부터의 코드를 전송하는 것과, 레지스터에 코드를 저장하는 것과, 복수의 페이지들 중 한 페이지의 페이지 주소를 코드의 각 비트에 대한 코드의 비트와 비교함으로써 극성 신호를 발생하는 것과, 발생된 극성 비트에 기초하여 복수의 페이지들의 일부분의 엔코딩을 변경하는 것을 포함한다.

실시예

- [0094] 도 7a는 본 발명의 개선된 프로세서가 구현되는, 한 बैं크(bank)의 분할된 판독/기입 스택들(stack)을 구비한 콤팩트 메모리 장치를 개요적으로 도시한 것이다. 메모리 장치는 2차원 어레이의 메모리 셀들(300), 제어회로(310), 및 판독/기입 회로들(370)을 포함한다. 메모리 어레이(300)는 행 디코더(330)를 통해서 워드라인들에 의해 그리고 컬럼 디코더(360)를 통해서 비트라인들에 의해 어드레스될 수 있다. 판독/기입 회로들(370)은 한 बैं크의 분할된 판독/기입 스택들(400)로서 구현되고 한 블록("페이지"라고도 함)의 메모리 셀들이 병렬로 판독되거나 프로그램되도록 한다. 바람직한 실시예에서, 페이지는 인접한 한 행의 메모리 셀들로부터 구성된다. 한 행의 메모리 셀들이 복수의 블록들 또는 페이지들로 분할되는 또 다른 실시예에서, 블록 멀티플렉서(350)는 개별적인 블록들에 판독/기입 회로들(370)을 멀티플렉스하기 위하여 제공된다.
- [0095] 제어회로(310)는 메모리 어레이(300) 상에 메모리 동작들을 수행하도록 판독/기입 회로들(370)과 협동한다. 제어회로(310)는 상태 머신(312), 온-칩 어드레스 디코더(314), 및 전력 제어 모듈(316)을 포함한다. 상태 머신(312)은 메모리 동작들의 칩 레벨 제어를 제공한다. 온-칩 어드레스 디코더(314)는 호스트 또는 메모리 제어기에 의해 사용되는 것 간의 어드레스 인터페이스를 디코더들(330, 360)에 의해 사용되는 하드웨어 어드레스에 제공한다. 전력 제어 모듈(316)은 메모리 동작들 동안 워드라인들 및 비트라인들에 공급되는 전력과 전압을 제어한다.
- [0096] 도 7b는 도 7a에 도시된 콤팩트 메모리 장치의 바람직한 배열을 도시한 것이다. 각종 주변 회로들에 의한 메모리 어레이(300)에의 액세스는 각 축의 액세스 라인들 및 회로가 절반으로 감소되도록 어레이의 서로 대향하는 축들 상에서 대칭적인 방식으로 구현된다. 따라서, 행 디코더는 행 디코더들(330A, 330B)로 분할되고 컬럼 디코더는 컬럼 디코더들(360A, 360B)로 분할된다. 한 행의 메모리 셀들이 복수의 블록들로 분할되는 실시예에서, 블록 멀티플렉서(350)는 블록 멀티플렉서들(350A, 350B)로 분할된다. 유사하게, 판독/기입 회로들은 하부로부터 비트라인들로 접속하는 판독/기입 회로들(370A) 및 어레이(300)의 최상부로부터 비트라인들에 연결되는 판독/기입 회로들(370B)로 분할된다. 이 방식으로, 판독/기입 모듈들의 밀도와, 따라서 분할된 판독/기입 스택들(400)의 밀도는 근본적으로 1/2만큼 감소된다.
- [0097] 도 8은 도 7a에 도시된 판독/기입 스택 내 기본 성분들의 일반적인 배열을 개요적으로 도시한 것이다. 발명의 일반적인 아키텍처에 따라, 판독/기입 스택(400)은 k 비트라인들을 감지하기 위한 한 스택의 감지 증폭기들(212), I/O 버스(231)를 통한 데이터의 입력 또는 출력을 위한 I/O 모듈(440), 입력 또는 출력 데이터를 저장하기 위한 한 스택의 데이터 래치들(430), 판독/기입 스택(400)간에 데이터를 처리하고 저장하는 공통 프로세서

(500), 및 스택 성분들간에 통신을 위한 스택 버스(421)를 포함한다. 관독/기입 회로들(370)간에 스택 버스 제어기는 관독/기입 스택들간에 각종 성분들을 제어하기 위해 라인들(411)을 통해 제어 및 타이밍 신호들을 제공한다.

- [0098] 도 9는 도 7a 및 도 7b에 도시된 관독/기입 회로들 간에 관독/기입 스택들의 한 바람직한 배열을 도시한 것이다. 각각의 관독/기입 스택(400)은 일 그룹의 k 비트라인들에 병렬로 동작한다. 페이지가 $p=r*k$ 비트라인들을 갖고 있다면, r개의 관독/기입 스택들(400-1, ..., 400-r)이 있을 것이다.
- [0099] 병렬로 동작하는 한 बैं크의 분할된 관독/기입 스택들(400) 전체는 한 행을 따라 p셀들의 블록(또는 페이지)이 병렬로 관독 또는 프로그램되게 한다. 이에 따라, 한 행의 셀들 전체에 대해 p개의 관독/기입 모듈들이 있을 것이다. 각 스택이 k개의 메모리 셀들에 사용되고 있기 때문에, बैं크 내 관독/기입 스택들의 총 수는 $r=p/k$ 로 주어진다. 예를 들면, r이 बैं크 내 스택들의 수이면, $p=r*k$ 이다. 일 예로서 메모리 어레이는 $p=512$ 바이트(512x8 비트), $k=8$ 을 가질 수 있고 따라서 $r=512$ 이다. 바람직한 실시예에서, 블록은 연속한 전체 한 행의 셀들이다. 또 다른 실시예에서, 블록은 행 내에 한 부분의 셀들이다. 예를 들면, 한 부분의 셀들은 전체 행의 1/2 또는 전체 행의 1/4일 수 있다. 한 부분의 셀들은 연속한 인접한 셀들이거나 하나 건너 한 셀, 또는 소정 수로 건너 있는 셀들일 수도 있을 것이다.
- [0100] 400-1과 같은 각 관독/기입 스택은 한 부분의 k개의 메모리 셀들에 병렬로 사용되는 한 스택의 감지 증폭기들(212-1 내지 212-k)을 필수로 포함한다. 바람직한 감지 증폭기는 개시된바 전체를 참조로 여기에 포함시키는 미국특허공개 2004-0109357-A1에 개시되어 있다.
- [0101] 스택 버스 제어기(410)는 라인들(411)을 통해 관독/기입 회로(370)에 제어 및 타이밍 신호들을 제공한다. 스택 버스 제어기 자체는 라인들(311)을 통해 메모리 제어기(310)에 따른다. 각 관독/기입 스택(400)간에 통신은 SABus(422)와 Dbus(423)를 포함하는 스택 버스에 의해서 행해지고 스택 버스 제어기(410)에 의해 제어된다. 제어 라인(411)은 스택 버스 제어기(410)로부터 관독/기입 스택(400-1)의 성분에 제어 및 클럭 신호를 제공한다.
- [0102] 바람직한 실시예에서, 스택 버스는 공통 프로세서(500)와 한 스택의 감지 증폭기들(212)간에 통신을 위한 SABus(422), 그리고 프로세서와 한 스택의 데이터 래치들(430) 간에 통신을 위한 Dbus(423)로 분할된다.
- [0103] 한 스택의 데이터 래치들(430)은 스택에 연관된 각각의 메모리 셀마다 하나씩인 데이터 래치들(430-1 내지 430-k)로 구성된다. I/O 모듈(440)은 데이터 래치들이 I/O 버스(231)를 통해 외부와 데이터를 교환할 수 있게 한다.
- [0104] 또한, 공통 프로세서는 오류 상태와 같은 메모리 동작의 상태를 나타내는 상태신호의 출력을 위한 출력(507)을 포함한다. 상태신호는 OR 결선(Wired-Or) 구성으로 플래그 버스(FLAG BUS)(509)에 연결된 n-트랜지스터(550)의 게이트를 구동하기 위해 사용된다. 플래그 버스는 제어기(310)에 의해 프리차지(precharge)되는 것이 바람직하며 관독/기입 스택들 중 어느 하나에 의해 상태신호가 발현되었을 때 풀 다운(pull down) 될 것이다.
- [0105] 도 10은 도 9에 도시된 공통 프로세서의 개선된 실시예를 도시한 것이다. 공통 프로세서(500)는 프로세서 버스, 외부 회로들과의 통신을 위한 PBUS(505), 입력 로직(510), 프로세서 래치 PLatch(520) 및 출력 로직(530)을 포함한다.
- [0106] 입력 로직(510)은 PBUS로부터 데이터를 수신하고 신호라인들(411)을 통해 스택 버스 제어기(410)로부터의 제어 신호들에 따라 논리 상태들 "1", "0" 또는 "Z"(플로팅) 중 하나로 변환된 데이터로서 BSI 노드에 출력한다. 셋/리셋 래치 PLatch(520)는 BSI를 래치하여 MTCH 및 MTCH*로서 한 쌍의 상보 출력 신호들이 된다.
- [0107] 출력 로직(530)은 MTCH 및 MTCH* 신호들을 수신하고 신호라인들(411)을 통해 스택 버스 제어기(410)로부터의 제어신호들에 따라 논리 상태들 "1", "0" 또는 "Z"(플로팅) 중 하나로 변환된 데이터를 PBSU(505) 상에 출력한다.
- [0108] 어떤 한 시간에 공통 프로세서(500)는 주어진 메모리 셀에 관계된 데이터를 처리한다. 예를 들면, 도 10은 비트 라인(1)에 결합된 메모리 셀에 대한 경우를 도시한 것이다. 대응하는 감지 증폭기(212-1)는 감지 증폭기 데이터가 나타나는 노드를 포함한다. 바람직한 실시예에서, 노드는 데이터를 저장하는 SA 래치(214-1) 형태를 취한다. 유사하게, 대응하는 한 세트의 데이터 래치들(430-1)은 비트라인(1)에 결합된 메모리 셀에 연관된 입력 또는 출력 데이터를 저장한다. 바람직한 실시예에서, 한 세트의 데이터 래치들(430-1)은 n 비트들의 데이터를 저장하기 위한 충분한 데이터 래치들(434-1, ..., 434-n)을 포함한다.
- [0109] 공통 프로세서(500)의 PBUS(505)는 전송 게이트(501)가 한 쌍의 상보 신호들(SAP, SAN)에 의해 활성화되었을 때 SBUS(422)를 통해 SA 래치(214-1)에 액세스할 수 있다. 유사하게, PBUS(505)는 전송 게이트(502)가 한 쌍의 상보 신호들(DTP, DTN)에 의해 활성화되었을 때 DBUS(423)를 통해 한 세트의 데이터 래치들(430-1)에 액세스할 수

있다. 신호들(SAP, SAN, DTP, DTN)이 스택 버스 제어기(410)로부터 제어신호들의 일부로서 명백하게 도시되었다.

[0110] 도 11a는 도 10에 도시된 공통 프로세서의 입력 로직의 바람직한 실시예를 도시한 것이다. 입력 로직(520)은 PBUS(505) 상의 데이터를 수신하고, 제어신호들에 따라, 출력 BSI가 동일하거나, 반전되거나, 플로팅되게 한다. 출력 BSI 노드는 근본적으로 전송 게이트(522)의 출력에 의해서, 또는 Vdd에 직렬의 p-트랜지스터(524, 525)를 포함하는 풀-업(pull-up) 회로에 의해서, 또는 그라운드에 직렬의 n-트랜지스터(526, 527)를 포함하는 풀-다운(pull-down) 회로에 의해 영향을 받는다. 풀-업 회로는 각각 신호들 PBUS 및 ONE에 의해 제어되는 p-트랜지스터(524, 525)에 게이트들을 갖는다. 풀-다운 회로는 각각 신호들 ONEB<1> 및 PBUS에 의해 제어되는 n-트랜지스터(526, 527)에 게이트들을 갖는다.

[0111] 도 11b는 도 11a의 입력 로직의 진리표를 도시한 것이다. 로직은 PBUS, 그리고 스택 버스 제어기(410)로부터의 제어 신호들의 일부인 제어신호들(ONE, ONEB<0>, ONEB<1>)에 의해 제어된다. 근본적으로, 3개의 전송 모드들로서 패스스루(PASSTHROUGH), 반전(INVERTED), 및 플로팅(FLOATED)이 지원된다.

[0112] BSI가 입력 데이터와 동일한 경우인 패스스루 모드의 경우에, 신호들 ONE은 논리 "1"에 있고, ONEB<0>은 "0"에, ONEB<1>은 "0"에 있다. 이것은 풀-업 또는 풀-다운을 비활성화하지만 전송 게이트(522)가 PBSU(505) 상의 데이터를 출력(523)에 전달할 수 있게 할 것이다. BSI가 입력 데이터의 반전인 반전 모드의 경우에, 신호들 ONE은 "0"에 있고, ONEB<0>은 "1"에, ONEB<1>은 "1"에 있다. 이것은 전송 게이트(522)를 비활성화할 것이다. 또한, PBUS가 "0"에 있을 때, 풀-다운 회로는 비활성화될 것이지만 풀-업 회로는 활성화되어, BSI는 "1"에 있게 될 것이다. 유사하게, PBUS가 "1"에 있을 때, 풀-업 회로는 비활성화되거나 풀-다운 회로는 활성화되어, BSI는 "0"에 있게 될 것이다. 마지막으로, 플로팅 모드의 경우에, 출력 BSI는 신호들 ONE이 "1"에, ONEB<0>이 "1"에, 그리고 ONEB<1>이 "0"에 있게 함으로써 플로팅될 수 있다. 플로팅 모드는 실제로는 사용되지 않을지라도, 완전하게 하기 위해서 열거되었다.

[0113] 도 12a는 도 10에 도시된 공통 프로세서의 출력 로직의 바람직한 실시예를 도시한 것이다. 입력 로직(520)으로부터 BSI 노드의 신호는 프로세서 래치 PLatch(520)에 래치된다. 출력 로직(530)은 PLatch(520)의 출력으로부터 데이터 MTCH 및 MTCH*을 수신하고 제어신호들에 따라, 패스스루, 반전 또는 플로팅 모드에서와 같이 PBUS 상에 출력한다. 즉, 4개의 브랜치들은 PBUS(505)에 대한 드라이버들로서 작용하여, 이를 능동적으로 하이(HIGH), 로우(LOW) 또는 플로팅 상태로 가져간다. 이것은 4 브랜치 회로들에 의해서, 즉 PBUS(505)에 대한 2개의 풀-업 및 2개의 풀-다운 회로들에 의해서 달성된다. 제 1 풀-업 회로는 Vdd에 직렬의 p-트랜지스터들(531, 532)를 포함하며, MTCH가 "0"에 있을 때 PBUS를 풀업할 수 있다. 제 2 풀-업 회로는 그라운드에 직렬의 p-트랜지스터들(533, 534)를 포함하고 MTCH가 "1"에 있을 때 PBUS를 풀업할 수 있다. 유사하게, 제 1 풀-다운 회로는 Vdd에 직렬의 n-트랜지스터들(535, 536)을 포함하며, MTCH가 "0"에 있을 때 PBUS를 풀다운할 수 있다. 제 2 풀-다운 회로는 그라운드에 직렬의 n-트랜지스터(537, 538)를 포함하고, MTCH가 "1"에 있을 때 PBUS를 풀다운할 수 있다.

[0114] 발명의 한 특징은 PMOS 트랜지스터들로 풀-업 회로와 NMOS 트랜지스터들로 풀-다운 회로들을 구성하는 것이다. NMOS에 의한 인출(pu11)이 PMOS의 인출보다 훨씬 강하기 때문에, 풀-다운은 어떠한 경합들에서도 풀-업을 항상 압도할 것이다. 즉, 노드 또는 버스는 풀-업 또는 "1" 상태로 항상 내정될 수 있고, 요망된다면, 풀-다운에 의해 언제든지 "0" 상태로 바뀔 수 있다.

[0115] 도 12b는 도 12a의 출력 로직의 진리표를 도시한 것이다. 로직은 입력 로직으로부터 래치된 MTCH, MTCH*와 스택 버스 제어기(410)로부터의 제어신호들의 일부는 제어신호들(PDIR, PINV, NDIR, NINV)에 의해 제어된다. 4개의 동작 모드들로서 패스스루(PASSTHROUGH), 반전(INVERTED), 플로팅(FLOATED), 프리차지(PRECHARGE)가 지원된다.

[0116] 플로팅 모드에서, 4개의 모든 브랜치들은 비활성화된다. 이것은 역시 디폴트 값들인, 신호들 PINV = 1, NINV = 0, PDIR = 1, NDIR = 0을 취함으로써 달성된다. 패스스루 모드에서, MTCH=0일 때, 이것은 PBUS=0을 요구할 것이다. 이것은 NDIR=1은 제외하고 모든 제어신호들을 이들의 디폴트값들에 있게 하고, n-트랜지스터들(535, 536)을 사용한 풀-다운 브랜치를 활성화하는 것 만에 의해 달성된다. MTCH = 1일 때, 이것은 PBUS=1을 요구할 것이다. 이것은 PINV=0은 제외하고 모든 제어신호들을 이들의 디폴트값들에 있게 하고, p-트랜지스터들(533, 534)을 사용한 풀-업 브랜치를 활성화하는 것 만에 의해 달성된다. 반전 모드에서, MTCH=0일 때, 이것은 PBUS=1을 요구할 것이다. 이것은 PDIR=0은 제외하고 모든 제어신호들을 이들의 디폴트값들에 있게 하고, p-트랜지스터들(531, 532)을 사용한 풀-업 브랜치를 활성화하는 것 만에 의해 달성된다. MTCH=1일 때, 이것은 PBUS=0을 요구할 것이다. 이것은 NINV=1은 제외하고 모든 제어신호들을 이들의 디폴트값들에 있게 하고, n-트랜지스터들(537, 538)을 사용한 풀-다운 브랜치를 활성화하는 것 만에 의해 달성된다. 프리차지 모드에서, PDIR=0 및 PINV=0의

제어 신호 설정은 MTCH=0일 때 p-트랜지스터(531, 532)를 사용한 풀-업 브랜치를 활성화하거나 MTCH=1일 때 p-트랜지스터(533, 543)를 사용한 풀-업 브랜치를 활성화할 것이다.

- [0117] 공통 프로세서 동작들은 전체를 여기 참조로 포함시키는 2004년 12월 29일에 출원된 미국특허출원번호 11/026,536에 보다 자세히 전개되어 있다.
- [0118] 캐시 동작들에서 데이터 래치들의 사용
- [0119] 본 발명의 다수의 면들은 내부 메모리가 판독, 기입, 또는 소거와 같은 다른 동작들을 하고 있는 동안에 데이터를 입력 및 출력할 캐시 동작들을 위해 도 10에서 위에 기술된 판독/기입 스택들의 데이터 래치들을 이용한다. 위에 기술된 아키텍처들에서, 데이터 래치들은 다수의 물리적 페이지들에 의해 공유된다. 예를 들면, 모든 워드 라인들에 의해 공유되는, 비트라인들의 판독/기입 스택들에서, 한 동작이 진행되고 있는 동안, 이들 래치들 중 어느 것이 비어있다면, 이들은 동일 또는 다른 워드라인에서 추후의 동작들을 위해 데이터를 캐시할 수 있고, 이것이 다른 동작 뒤로 은닉될 수 있기 때문에 전송시간을 절약할 수 있다. 이것은 서로 다른 동작들 또는 동작들의 국면들을 파이프라인하는 량을 증가시킴으로써 성능을 개선할 수 있다. 일 예에서, 캐시 프로그램 동작에서, 한 페이지의 데이터를 프로그램하는 동안 다른 한 페이지의 데이터가 로딩될 수 있어, 전송시간을 절약할 수 있다. 또 다른 예에 있어서, 한 실시예에서, 한 워드라인 상에서 판독 동작은 또 다른 워드라인 상에 기입 동작에 삽입되어, 데이터 기입이 계속되는 동안 판독으로부터의 데이터가 메모리로부터 전송될 수 있게 한다.
- [0120] 이것은 기입 또는 그외 동작이 제 1 페이지의 데이터에 대해 진행되고 있는 동안에 동일 블록 내 또 다른 페이지로부터의, 그러나 다른 워드라인 상의 데이터가 토글 아웃(toggle out)되게(예를 들면, ECC 동작을 위해서) 하는 것에 유의한다. 동작들의 이러한 국면 간 파이프라이닝(pipelining)은 데이터 전송에 필요한 시간이 제 1 페이지의 데이터에 대한 동작 뒤로 은닉되게 한다. 보다 일반적으로, 이것은 한 동작의 일부가 다른, 전형적으로 더 긴, 동작의 국면들 사이에 삽입될 수 있게 한다. 또 다른 예는, 이를테면 소거 펄스 전 또는 소거의 후반 부분으로서 사용되는 소프트(soft) 프로그래밍 국면 전과 같이, 예를 들어 소거 동작의 국면들 사이에 감지 동작을 삽입하는 것이다.
- [0121] 논의된 일부 동작들에 대해 상대적 시간들이 필요하게 하기 위해서, 위에 기술된 시스템에 대한 한 세트의 대표적인 시간값들은 다음처럼 취해질 수 있다.
- [0122] 데이터 기입: ~700 μ s(하위 페이지 ~ 600 μ s, 상위 페이지 800 μ s)
- [0123] 2진 데이터 기입: ~200 μ s
- [0124] 소거: ~2,500 μ s
- [0125] 판독: ~20 - 40 μ s
- [0126] 판독 및 토글 아웃 데이터: 2KB 데이터, ~80 μ s; 4KB ~160 μ s; 8KB ~320 μ s.
- [0127] 이들 값들은 밑에 타이밍도들에 연루된 상대적 시간들을 알려주기 위한 참조를 위해 사용될 수 있다. 상이한 국면들에 긴 동작이 있다면, 주요 면은 래치들이 사용될 수 있을 경우 판독/기입 스택들의 공유된 래치들을 사용하여 보다 신속한 동작에 개입할 것이다. 예를 들면, 판독은 프로그램 또는 소거 동작에 삽입될 수 있거나, 2진 프로그램이 소거에 삽입될 수 있다. 주요 실시예들은 예를 들면 토글 아웃 및 수정된 데이터의 판독이 데이터 기입의 검증 국면에 삽입되는 경우, 동일 판독 기입 스택들을 공유하는 또 다른 페이지에 대한 프로그램 동작 동안 한 페이지에 대해 데이터를 토글 인 및/또는 아웃할 것이다.
- [0128] 열린 데이터 래치들의 이용성은 많은 방법들로 발생할 수 있다. 일반적으로, 셀 당 n 비트를 저장하는 메모리에 있어서, 각 비트라인에 대해 n개의 이러한 데이터 래치들이 필요로 될 것이지만, 이들 모든 래치들이 항상 필요로 되는 것은 아니다. 예를 들면, 상위 페이지/하위 페이지 포맷으로 데이터를 저장하는 셀 당 2비트 메모리에서, 하위 페이지를 프로그램하는 동안 2개의 데이터 래치들이 필요로 될 것이다. 보다 일반적으로, 복수의 페이지들을 저장하는 메모리들에 있어서, 모든 래치들은 최상위 페이지를 프로그램할 때만 필요하게 될 것이다. 이것은 다른 래치들을 캐시 동작들을 위해 사용될 수 있는 상태가 되게 한다. 또한, 최상위 페이지를 기입하는 동안에도, 기입 동작의 검증 국면으로부터 여러 가지 상태들이 제거되므로, 래치들은 자유롭게 될 것이다. 구체적으로, 일단 최상위 상태만이 검증될 것으로 남아있게 되면, 검증 목적을 위해 하나의 래치만이 필요하게 되고 다른 것들은 캐시 동작들을 위해 사용될 수 있다.
- [0129] 다음 논의는 위에서 포함되었던 본원과 동시에 출원된 "Use of Data Latches in Multi-Phase Programming of

Non-Volatile Memories" 명칭의 미국특허 출원에 기술된 바와 같이, 셀 당 2비트들을 저장하고 각 비트라인 상의 데이터를 위한 2개의 래치들과 신속 패스(pass) 기입을 위한 하나의 추가의 래치를 구비하는 4 상태 메모리에 기초할 것이다. 하위 페이지를 기입하거나, 소거하거나, 사후 소거 소프트웨어 프로그램을 행하는 동작들은 기본적으로 2진 동작이며 데이터 래치들 중 한 래치를 방면시킬 수 있고, 이 래치는 데이터를 캐시하는데 사용할 수 있다. 유사하게, 상위 페이지 또는 풀(full) 시퀀스 기입을 행하는 경우, 최상위 레벨을 제외한 모든 레벨이 일단 검증되었으면, 한 상태만이 검증할 필요가 있고 메모리는 데이터를 캐시하는데 사용될 수 있는 래치를 자유롭게 할 수 있다. 이것이 어떻게 사용될 수 있는가의 예는 이를테면 카피(copy) 동작에서와 같이 한 페이지를 프로그램할 때, 같은 한 세트의 비트라인들 상에 또 다른 워드라인과 같이, 같은 한 세트의 데이터 래치들을 공유하는 또 다른 페이지의 판독은 기입의 검증 국면 동안 섞여들 수 있다. 이어서 주소는 기입되는 페이지로 전환될 수 있어, 기입 프로세스는 다시 시작해야 할 필요없이 중지된 곳을 회복할 수 있게 된다. 기입이 계속하는 동안, 삽입된 판독 동안 캐시된 데이터가 토글 아웃 되거나, 체크되거나, 수정되고, 앞선 기입 동작이 일단 완료되면 다시 기입을 위해 존재할 수 있게 다시 전송될 수 있다. 이러한 종류의 캐시 동작은 제 2 페이지의 토글 아웃 및 수정이 제 1 페이지의 프로그래밍 뒤로 은닉될 수 있게 한다.

[0130] 제 1 예로서, 2비트 메모리를 위한 캐시 프로그램 동작은 단일 페이지(하위 페이지/상위 페이지 포맷) 프로그램 모드에서 동작한다. 도 13은 다른 요소들은 논의를 단순화하기 위해 생략하고 2비트 실시예에서 본 논의에 관계된 어떤 특정한 요소들을 나타낸, 도 10을 간이화하여 도시한 것이다. 이들은 데이터 I/O 라인(231)에 접속되는 데이터 래치 DL0(434-0), 선(423)에 의해 공통 프로세서(50)에 접속되는 데이터 래치 DL1(434-1), 선(435)에 의해 다른 데이터 래치들에 공통으로 접속되는 데이터 래치 DL2(434-2), 및 선(422)에 의해 공통 프로세서(50)에 접속되는 감지 증폭 데이터 래치 DLS(214)를 포함한다. 도 13의 여러 요소들은 하위 페이지의 프로그래밍 동안 이들 요소들의 사용에 따라 표기되었다. 래치 DL2(434-2)은 본원과 동시에 출원된 "Use of Data Latches in Multi-Phase Programming of Non-Volatile Memories" 명칭의 미국특허출원에 기술된 바와 같이, 신속 패스 기입 모드에서 하위 검증(VL)을 위해 사용되는데, 래지스터의 포함과, 이것이 포함될 때 신속 패스 기입을 사용하는 것을 포함하는 것은 선택적이지만, 본 실시예는 이 래지스터를 포함할 것이다.

[0131] 하위 페이지의 프로그래밍은 다음 단계들을 포함할 수 있다.

[0132] (1) 프로세서는 데이터 래치들 DL0(434-0)을 디폴트값 "1"으로 리셋함으로써 시작한다. 이러한 관례는 프로그램 되지 않을 선택된 행에 셀들이 프로그램 금지될 것이기 때문에 부분적 페이지 프로그래밍을 단순화하기 위해 사용된다.

[0133] (2) 프로그램 데이터는 I/O 라인(231)을 따라 DL0(434-0)에 공급된다.

[0134] (3) 프로그램 데이터는 DL1(434-1) 및 DL2(434-2)에 전송될 것이다(이 래치가 포함되고 신속 패스 기입이 구현된다면).

[0135] (4) 일단 프로그램 데이터가 DL1(434-1)에 전송되면, 데이터 래치 DL0(434-0)은 "1"로 리셋될 수 있고, 프로그램 시간 동안, 다음 데이터 페이지가 I/O 라인(231)을 따라 DL0(434-0)에 로딩될 수 있어, 제 1 페이지가 기입되고 있는 동안 제 2 페이지가 캐시될 수 있게 된다.

[0136] (5) 일단 제 1 페이지가 DL1(434-1)에 로딩되면, 프로그래밍이 시작할 수 있다. DL1(434-1)의 데이터는 후속 프로그램으로부터 셀의 록아웃(lockout)을 위해 사용된다. DL2(434-2)의 데이터는 본원과 동시에 출원된 "Use of Data Latches in Multi-Phase Programming of Non-Volatile Memories" 명칭의 미국특허출원에 기술된 바와 같이, 신속 패스 기입의 제 2 국면으로의 이행(transition)을 제어하는 하위 검증 록아웃을 위해 사용된다.

[0137] (6) 일단 프로그래밍이 시작되면, 프로그래밍 펄스 후에, DL2(434-2)를 업데이트하기 위해 하위 검증의 결과가 사용되며; 상위 검증의 결과는 DL1(434-1)을 업데이트하기 위해 사용된다. (이 논의는 하위 페이지 프로그래밍이 A 상태에 대한 것인 경우, "통상의" 코딩에 기초한다. 이 코딩, 및 이외 코딩들은 본원과 동시에 출원된 "Use of Data Latches in Multi-Phase Programming of Non-Volatile Memories" 명칭의 미국특허출원 및 2005년 3월 16일에 출원된 "Non-Volatile Memory and Method with Power-Saving Read and Program-Verify Operations" 명칭의 미국특허출원에 더 논의되어 있다. 다른 코딩들에의 본 논의의 확장은 용이하게 이어진다).

[0138] (7) 프로그래밍이 완료되었는지를 판정함에 있어, 행(또는 프로그램의 적절한 물리적 단위)의 셀들의 DL1(434-1) 래지스터들만이 체크된다.

[0139] 일단 하위 페이지가 기입되면, 상위 페이지가 프로그램될 수 있다. 도 14는 도 13과 동일한 요소들을 도시하나, 하위 페이지 데이터가 판독되는 상위 페이지 프로그램을 위한 래치 할당을 나타낸다. (이 설명은 다시 통상의

코딩을 사용하며, 따라서 상위 페이지의 프로그래밍은 B 상태 및 C 상태에 대한 것이다). 상위 페이지의 프로그래밍은 다음의 단계들을 포함할 수 있다.

- [0140] (1) 일단 하위 페이지가 프로그래밍을 종료하면, 상위 페이지(또는 다음 페이지) 기입은 (실행되지 않은) 캐시 프로그램 명령들이 유지되는 상태 머신 제어기로부터의 신호로 시작할 것이다.
- [0141] (2) 프로그램 데이터는 DL0(434-0)(하위 페이지 기입동안 단계 (3)에서 로딩되었던 곳)에서 DL1(434-1) 및 DL2(434-2)로 전송될 것이다.
- [0142] (3) 하위 페이지 데이터는 어레이로부터 관독되어 DL0(434-0)에 놓여질 것이다.
- [0143] (4) DL1(434-1) 및 DL2(434-2)는 다시 각각 검증 하이 및 검증 로우 록아웃 데이터를 위해 사용된다. 래치 DL0(434-0)(하위 페이지 데이터를 유지하는)은 프로그램 참조 데이터로서 체크되나, 검증 결과들로 업데이트되지 않는다.
- [0144] (5) B 상태를 검증하는 부분으로서, 하위 검증 VBL에서 감지한 후에, 이에 따라 데이터가 DL2(434-2)에서 업데이트될 것이며, DL1(434-1) 데이터는 하이 검증 VBH 결과들로 업데이트된다. 유사하게, C 검증은 각각의 VCL 및 VCH 결과들로 래치들 DL2(434-2) 및 DL1(434-1)을 업데이트하는 대응하는 명령들을 가질 것이다.
- [0145] (6) 일단 B 데이터가 완료되면, C 상태에 대한 검증만이 수행될 필요가 있으므로 하위 페이지 데이터(참조를 위해 DL0(434-0)에 유지된)는 필요로 되지 않는다. DL0(434-0)은 "1"로 리셋되고 또 다른 페이지의 프로그램 데이터는 I/O 라인(231)으로부터 로딩되어 래치 DL0(434-0)에 캐시될 수 있다. 공통 프로세서(500)는 C 상태만이 검증되어야 한다는 지시를 설정할 수 있다.
- [0146] (7) 상위 페이지 프로그래밍이 완료되었는지를 판정함에 있어, B 상태에 대해서, 래치들 DL1(434-1) 및 DL0(434-0) 둘 다 체크된다. 일단 셀들이 B 상태로 프로그램되고 C 상태만이 검증되고 있으면, 프로그램되지 않은 어떤 비트들이 있는지 알기 위해 래치 DL1(434-1)의 데이터만이 체크될 필요가 있다.
- [0147] 이러한 배열 하에서, 단계 6에서, 래치 DL0(434-0)은 더 이상 요구되지 않으며 다음 프로그래밍 동작을 위해 데이터를 캐시하는데 사용될 수 있는 것에 유의한다. 또한, 신속 패스 기입을 사용하는 실시예들에서, 일단 제 2의 저속의 프로그래밍 국면에 진입되면, 데이터를 캐시하기 위해 래치 DL2(434-2)를 사용할 수 있게 할 수도 있을 것이지만, 그러나 실제로는 이것은 종종 이 특징을 구현하는데 요구되는 추가의 오버헤드를 정당화하지 않는 꽤 짧은 시간동안 이러한 식으로만 가능할 수 있는 경우이다.
- [0148] 도 15는 바로 앞의 몇 단락들에 기술된 단일 페이지 모드에서 캐시 프로그램의 많은 면들을 예시하기 위해 사용될 수 있다. 도 15는 메모리에 내부적으로 어떤 이벤트들이 일어나는가(밑에 "실제 비지(Busy)" 라인)와 메모리 외부에서 알 수 있는(위에 "캐시 비지" 라인) 것의 상대적 타이밍을 도시한 것이다.
- [0149] 시간 t_0 에서, 선택된 워드라인(WLn)에 프로그램될 하위 페이지가 메모리에 로딩된다. 이것은 제 1 하위 페이지의 데이터가 사전에 캐시되지 않았으며, 후속되는 페이지들에 대해서도 그와 같을 것으로 가정한다. 시간 t_1 에서, 하위 페이지는 로딩이 종료되고 메모리는 이것을 기입하기 시작한다. 이것은 이 시점에서 2진 동작과 동등하기 때문에, 상태 A만이 검증될 필요가 있고("pvfyA"), 여기에서는 시간 t_2 에서 WLn에 프로그램될 상위 페이지들로서 취해진 다음 페이지의 데이터를 수신하기 위해 데이터 래치 DL0(434-0)가 사용될 수 있는데, 이 상위 페이지는 결국 하위 페이지의 프로그래밍 동안 래치 DL0(434-0)에 캐시된다. 상위 페이지는 시간 t_3 에서 로딩을 종료하고 t_4 에서 하위 페이지가 종료하는 즉시 프로그램될 수 있다. 이 배열 하에서, 모든 데이터(하위 및 상위 페이지)가 프로그래밍의 물리적 단위(여기에서는, 워드라인(WLn))에 기입될지라도, 메모리는 이하 기술되는 폴시퀀스 실시예와는 달리, 상위 페이지 데이터가 기입될 수 있기 전에 시간 t_3 에서 시간 t_4 까지 기다려야 한다.
- [0150] 상위 페이지의 프로그래밍은 시간 t_4 에서 시작하며, 여기서 초기에는 B 상태만이 검증되며("pvfyB"), C 상태가 t_5 에서 추가된다("pvfyB/C"). 일단 B 상태가 t_6 에서 더 이상 검증되고 있지 않게 되면, C 상태만이 검증될 필요가 있고("pvfyC") 래치 DL0(434-0)는 자유롭게 된다. 이것은 상위 페이지가 프로그래밍을 종료하는 동안 다음 데이터가 캐시될 수 있게 한다.
- [0151] 언급된 바와 같이, 캐시 프로그램을 사용하는 단일 페이지 알고리즘에 따라, 도 15에 도시된 바와 같이, 상위 페이지 데이터가 시간 t_3 에서 사용될 수 있을지라도, 메모리는 이 데이터를 기입하기를 시작하기 전에 시간 t_4 까

지 기다려야 한다. 미국특허출원 11/013,125에 보다 완전하게 전개되어 있는 바와 같은 풀 시퀀스 프로그램 동작으로 전환에서, 일단 상위 페이지가 가용하다면 상위 및 하위 페이지 데이터는 동시에 프로그램될 수 있다.

- [0152] 풀 시퀀스(로우에서 폴로의 전환) 기입에서 캐시 프로그램을 위한 알고리즘은 위에서처럼 하위 페이지 프로그램 부터 시작한다. 결국, 단계들 (1)-(4)는 단일 페이지 프로그램 모드에서 하위 페이지 프로세스에 관한 것이다.
- [0153] (1) 프로세스는 데이터 래치들 DL0(434-0)을 디폴트값 "1"으로 리셋함으로써 시작한다. 이러한 관례는 프로그램 되지 않을 선택된 행에 셀들이 프로그램 금지될 것이기 때문에 부분적 페이지 프로그래밍을 단순화하기 위해 사용된다.
- [0154] (2) 프로그램 데이터는 I/O 라인(231)을 따라 DL0(434-0)에 공급된다.
- [0155] (3) 프로그램 데이터는 DL1(434-1) 및 DL2(434-2)에 전송될 것이다(이 래치가 포함되고 신속 패스 기입이 구현된다면).
- [0156] (4) 일단 프로그램 데이터가 DL1(434-1)에 전송되면, 데이터 래치 DL0(434-0)은 "1"로 리셋될 수 있고, 프로그램 시간 동안, 다음 데이터 페이지가 I/O 라인(231)을 따라 DL0(434-0)에 로딩될 수 있어, 제 1 페이지가 기입되고 있는 동안 제 2 페이지가 캐시될 수 있게 된다.
- [0157] 일단 제 2 페이지의 데이터가 로딩되고, 이것이 기입되는 하위 페이지의 상위에 대응하고 하위 페이지는 아직 프로그램을 마치지 않았으면, 풀 시퀀스 기입으로의 전환이 이행될 수 있다. 이 논의는 이러한 알고리즘에서 데이터 래치들의 사용에 중점을 두며, 이외 많은 상세들은 함께 계류중인 같이 양도된 미국특허출원 11/013,125에 더 완전하게 전개되어 있다.
- [0158] (5) 상위 페이지 데이터가 래치 DL0(434-0)에 로딩된 후에, 한 페이지는 하위 페이지이고 또 한 페이지는 상위 페이지인 2개의 페이지들이 동일 워드라인 및 동일 블록 상에 있는지를 체크하기 위해서 어드레스 블록에서 판단이 행해질 것이다. 그러하다면, 프로그램 상태 머신은 허용된다면 풀 시퀀스 프로그램으로 하위 페이지 프로그램의 전환을 트리거할 것이다. 어떤 미결정된 검증이 완료된 후에, 이행이 행해진다.
- [0159] (6) 하위 페이지에서 풀 시퀀스로 프로그램 시퀀스가 변경되었을 때 전형적으로 일부 동작 파라미터들이 변경될 것이다. 실시예에서 이들은 다음을 포함한다.
- [0160] (i) 펄스 검증 사이클들의 수에 대한 최대 프로그램 루프는, 하위 페이지 데이터가 록아웃되지 않았다면 하위 페이지 알고리즘의 루프에서 풀 시퀀스의 루프로 변경될 것인데, 그러나 완료된 프로그램 루프들 수는 전환에 의해 리셋되지 않을 것이다.
- [0161] (ii) 도 16에 도시된 바와 같이, 프로그래밍 파형은 하위 페이지 프로그래밍 프로세스에서 사용되는 값 VPGM_L로 시작한다. 프로그래밍 파형이 풀 시퀀스로의 전환에서, 상위 페이지 프로세스에서 사용되는 시작 값 VPGM_U를 초과하는 곳까지 진행되었다면, 계단은 위쪽으로 계단을 계속하기에 앞서 VPGM_U까지 다시 낮출 것이다.
- [0162] (iii) 프로그램 펄스의 스텝 크기 및 최댓값을 결정하는 파라미터들은 변경되지 않는다.
- [0163] (7) 메모리 셀들의 현 상태의 풀 시퀀스 관독은 복수-레벨 코딩을 위해 올바른 데이터가 프로그램되어질 것을 보장하기 위해 수행되어야 한다. 이것은 하위 페이지 프로그래밍에서 전에 록아웃되어 있을 수도 있지만, 이들의 상위 페이지 데이터를 고려하기 위해서 추가 프로그래밍을 필요로 하는 상태들이, 풀 시퀀스가 시작될 때 확실하게 프로그램 금지되지 않게 한다.
- [0164] (8) 신속 패스 기입이 활성화된다면, 상위 페이지 프로그램 데이터가 전에 A 상태만에 대해 하위 검증에 기초하였기 때문에, 상위 페이지 프로그램 데이터를 반영하기 위해서 래치 DL2(434-2)의 데이터도 업데이트될 것이다.
- [0165] (9) 이어서 프로그래밍은 복수-레벨, 풀 시퀀스 프로그램 알고리즘으로 재개한다. 하위 페이지 프로세스에서 프로그램 파형이 상위 페이지 시작 레벨을 넘어 증가하였다면, 도 16에 도시된 바와 같이, 전환 시간에 파형은 이 레벨로 다시 낮추어진다.
- [0166] 도 17은 하위 페이지에서 풀 시퀀스로 전환 기입 프로세스에 연루된 상대적 시간들을 개요적으로 나타낸 것이다. 시간 t_3 까지, 프로세스는 도 15에서 프로세스에 대해 위에 기술된 바와 같다. t_3 에서 상위 페이지의 데이터가 로딩되었으며 A 상태들과 함께 B 상태들을 포함하게 검증 프로세스가 전환되는 풀 시퀀스 알고리즘으로의 이행이 행해진다. 모든 A 상태들이 일단 록아웃되었으면, 검증 프로세스는 t_4 에서 B 및 C 상태들에 대해 체

크하는 것으로 전환한다. 일단 B 상태들이 t_5 에서 검증되었으면, C 상태만이 체크될 필요가 있고 레지스터는 자유롭게 되어 캐시 비지 라인 상에 나타난 바와 같이 다음 워드라인(WL_{n+1}) 상에 하위 페이지와 같은, 프로그램될 다음 데이터를 로딩할 수 있게 된다. 시간 t_6 에서, 이 다음 데이터 세트는 캐시되어 있으며 이전 세트에 대한 C 데이터의 프로그래밍은 t_7 에서 끝내고, 이 다음 데이터 세트는 프로그래밍을 시작한다. 또한, 워드라인(WL_{n+1}) 상에 (여기에서) 하위 페이지가 프로그래밍되고 있는 동안, 다음 데이터(이들테면 대응하는 상위 페이지 데이터)는 열린 래치 DL0(434-0)에 로딩될 수 있다.

- [0167] 풀 시퀀스 기입 동안에, 독립적으로 하위 페이지 및 상위 페이지 상태를 알려주도록 상태 보고가 이행된다. 프로그램 시퀀스의 끝에서, 미완료된 비트들이 있다면, 물리 페이지의 스캔(scan)이 수행될 수 있다. 제 1 스캔은 미완료된 상위 페이지 데이터에 대해 래치 DL0(434-0)을 체크할 수 있고, 제 2 스캔은 미완료된 하위 페이지 데이터에 대해 DL1(434-1)을 체크할 수 있다. B 상태의 검증이 DL0(434-0) 및 DL1(434-1) 데이터 둘 다를 변경시킬 것이기 때문에, 비트의 임계값이 A 검증 레벨보다 높다면 DL1(434-1) 데이터 "0"이 "1"로 변경되도록 A 상태 검증이 수행되어야 한다. 이 사후 검증은 레벨 이하의 프로그램된 임의의 B 레벨들이 A 레벨에서 통과하는지를 체크할 것이며, 이들이 A 레벨에서 통과한다면, 오류는 상위 페이지 상에만 있고 하위 페이지 상에는 없으며, 이들이 A 레벨에서 통과하지 않는다면, 하위 및 상위 페이지들 둘 다는 오류를 갖는다.
- [0168] 캐시 프로그램 알고리즘이 사용된다면, A 및 B 데이터가 프로그램된 후에, C 상태가 래치 DL1(434-1)에 보내어져 프로그래밍을 마칠 것이다. 이 경우, 래치의 스캔은 하위 페이지가 어떠한 실패된 비트들도 없이 이미 프로그램을 통과하였을 것이기 때문에 하위 페이지에 대해선 필요하지 않다.
- [0169] 본 발명의 또 다른 한 세트의 실시예들은 페이지 카피 동작들에 관한 것으로, 이 경우 데이터 세트는 한 위치에서 다른 위치로 재배치된다. 데이터 재배치 동작의 여러 가지 양상은, 2004년 5월 13일에 출원된 미국 특허 출원 번호 10/846,289; 2004년 12월 21일에 출원된 번호 11/022,462; 2004년 8월 9일에 출원된 번호 10/915,039; 및 미국 특허 6,266,273에 기술되어 있고, 상기 문서는 본 명세서에 참조로 포함되어 있다. 데이터가 한 위치에서 다른 위치로 카피될 때, 데이터는 체크되거나(예를 들면, 오류에 대해서), 업데이트되거나(이들테면 헤더를 업데이트함), 둘 다를 위해서(이들테면 검출된 오류를 정정하는 것) 종종 토글 아웃된다. 이러한 전송들은 가비지 수거 동작(garbage collection operation)에서 데이터를 통합하는 것이다. 본 발명의 주요 면은 열린 레지스터에 대한 데이터 관독이 기입 동작의 검증 국면 동안 삽입될 수 있도록 하고, 이러한 캐시된 데이터가 이어서 기입 동작이 계속될 때 메모리 장치로부터 전송되며, 이에 따라 데이터를 토글 아웃하기 위한 시간이 기입 동작 뒤로 은닉되게 하는 것이다.
- [0170] 다음은 캐시 페이지 카피 동작의 2개의 실시예들을 제시한다. 두 경우들에 있어서, 신속 패스 기입 구현을 사용하는 구현이 기술된다. 도 18은 프로세스가 진행될 때 래치들의 배열의 처분을 나타낸다.
- [0171] 캐시 페이지 카피의 제 1 버전은 하위 페이지에 기입할 것이며 다음의 단계들을 포함할 수 있는데, 여기서 관독 주소들은 M, M+1, ...으로 표기되었고 기입 주소들은 N, N+1, ...으로 표기되었다.
- [0172] (1) 카피될 페이지("페이지 M")가 래치 DL1(434-1)로 읽혀진다. 이것은 상위 페이지 또는 하위 페이지의 데이터일 수 있다.
- [0173] (2) 이어서 페이지 M이 DL0(434-0)으로 전송된다.
- [0174] (3) DL0(434-0) 내의 데이터는 토글 아웃 및 수정되고, 이후에 이것은 다시 래치로 전송된다.
- [0175] (4) 이어서 프로그램 시퀀스가 시작될 수 있다. 하위 페이지 N에 기입될 데이터가 DL1(434-1) 및 DL2(434-2)에 전송된 후에, 래치 DL0(434-0)은 데이터를 캐시할 준비가 된다. 이 하위 페이지가 프로그램될 것이다. 이 실시예에 있어서, 프로그램 상태 머신은 여기에서 멈출 것이다.
- [0176] (5) 이어서 카피될 다음 페이지가 DL0(434-0)으로 읽혀진다. 이어서 프로그래밍은 재개될 수 있다. 단계(4)의 끝에서 멈춘 상태 머신은 프로그램 시퀀스를 처음부터 다시 시작할 것이다.
- [0177] (6) 프로그래밍은 하위 페이지가 끝날 때까지 계속된다.
- [0178] 카피 목적지 페이지 주소는 기입이 하위 페이지로인지 아니면 상위 페이지로인지를 결정할 것이다. 프로그램 주소가 상위 페이지 주소이라면, 프로그래밍 시퀀스는 프로그래밍이 종료할 때까지 중지되지 않을 것이며 단계(5)의 관독은 기입이 완료된 후 실행될 것이다.

- [0179] 제 2 캐시 카피 방법에서, 프로그램/검증 프로세스는 판독 동작을 삽입하기 위해 중단될 수 있고, 이어서 중단되었던 지점을 회복하여, 기입동작을 다시 시작할 수 있다. 이 인터리브된 감지 동작 동안 판독되었던 데이터는 재개된 기입 동작이 계속되는 동안 토글 아웃될 수 있다. 또한, 이 제 2 프로세스는 일단 C 상태만이 검증되고 있고 각 비트라인 상에 한 래치가 열리게 되면 페이지 카피 메커니즘이 상위 페이지 또는 풀 시퀀스 기입 프로세스에서 사용될 수 있게 한다. 제 2 캐시 페이지 카피 동작은 제 1 경우에서와 동일한 첫 번째 3개의 단계들부터 시작하는데, 그러나 이후는 다르다. 이것은 다음의 단계들을 포함할 수 있다.
- [0180] (1) 카피될 페이지("페이지 M")가 래치 DL1(434-1)로 읽혀진다. 이것은 하위 또는 상위 페이지일 수 있다.
- [0181] (2) 이어서 페이지 M으로부터 데이터가 DL0(434-0)으로 전송된다. (전처럼, N 등은 판독 주소에 대해 기입 주소, M 등을 나타낼 것이다.)
- [0182] (3) 이어서 DL0(434-0) 내의 데이터가 토글 아웃되고, 수정된 후 래치로 다시 전송된다.
- [0183] (4) 상태 머신 프로그램은 판독 명령이 입력될 때까지 무한 대기 상태로 갈 것이며 이어서 또 다른 페이지, 예를 들면 다음 페이지 M+1를 래치 DL0(434-0)로의 판독이 시작될 것이다.
- [0184] (5) 일단 단계(4)의 판독이 완료되면, 주소는 단계들(1-3)에서 데이터를 페이지 N(여기에서는, 하위 페이지)로 프로그램하기 위해서 다시 워드라인 및 블록 주소로 전환되고, 프로그래밍이 재개된다.
- [0185] (6) 페이지 M+1의 판독이 종료된 후에, 데이터는 토글 아웃되고, 수정되고 복귀될 수 있다. 일단 프로세스가 완료되면, 기입은 2개의 페이지들이 동일 WL 상에 대응하는 상위 및 하위 페이지들이라면 풀 시퀀스 동작으로 전환될 수 있다.
- [0186] (7) 일단 A 및 B 레벨들이 풀 시퀀스 기입에서 행해지면, DL0(434-0) 내에 데이터는 앞에서 기술된 정규 캐시 프로그램에서처럼, DL1(434-1)로 전송될 것이며, 또 다른 페이지(예를 들면, 페이지 M+2)에 대한 판독 명령이 발행될 수 있다. 단일 페이지에서 풀 시퀀스로의 전환이 없다면, 하위 페이지는 기입을 종료할 것이며 이어서 상위 페이지가 시작될 것이다. B 레벨 상태가 완전히 행해진 후에, 동일 DL0(434-0) 내지 DL1(434-1) 데이터 전송이 일어날 것이며, 상태 머신은 페이지 M+2에 대한 판독 명령을 기다리는 상태로 갈 것이다.
- [0187] (8) 일단 판독 명령이 도착하면, 주소는 판독 주소로 전환되고 다음 페이지(페이지 M+2)가 판독된다.
- [0188] (9) 일단 판독이 완료되면, 주소는 기입이 종료될 때까지 다시 이전 상위 페이지 주소(프로그램 주소 N+1)로 전환될 것이다.
- [0189] 위에 언급된 바와 같이, 실시예들은 메모리 셀들의 각각에 프로그램될 수 있는 데이터(여기에서는 2비트의)를 유지하는데 사용되는 래치들 DL0(434-0) 및 DL1(434-1) 외에도 신속 패스 기입 기술의 하위 검증을 위해 사용되는 래치 DL2(434-2)를 포함한다. 일단 하위 검증이 통과되면, 래치 DL2(434-2)도 자유롭게 될 수 있고 본 실시예들에서 행해지지 않을지라도 데이터를 캐시하는데 사용될 수 있다.
- [0190] 도 19a 및 도 19b는 제 2 캐시 페이지 카피 방법의 상대적 타이밍을 도시한 것이며, 도 19b는 풀 시퀀스 기입 전환을 가진 알고리즘을 도시한 것이며 도 19a는 이를 갖는 않는 알고리즘을 도시한 것이다. (도 19a 및 도 19b 둘 다는 두 부분들 구성되는데, 첫번째 위에 부분은 t_0 에 대응하는 점선의 수직선 A에서 시작하고 t_5 에 대응하는 점선의 수직선 B에서 끝나며, 두 번째 아랫부분은 위에 부분의 연속이며 t_5 에 대응하는 점선의 수직선 B에서 시작한다. 두 경우들에 있어서, 시간 t_5 에서 선 B는 하래 부분에서 위에 부분과 동일하고, 두 선들 상에 표시되게 한두 부분들을 잇는 것일 뿐이다).
- [0191] 도 19a는 이 예에서 하위 페이지인 것으로 취해지는 제 1 페이지(페이지 M)의 판독부터 시작하고, 어떠한 데이터도 사전에 캐시되어 있지 않은 것으로 가정하며 단일 페이지 모드에서 동작하고, 상위 페이지를 기입하기를 시작하기 전에 하위 페이지가 기입을 끝마칠 때까지 기다리는 프로세스를 도시한 것이다. 프로세스는 시간 t_0 에서 여기에서는 이 코딩에서 A 및 C 레벨들에서 판독에 의해 감지되는 하위인 페이지 M의 판독(페이지 M(L) 감지)부터 시작한다. 시간 t_1 의 시간에서 판독이 완료되고 페이지 M은 토글 아웃되고 체크 또는 수정될 수 있다. 시간 t_2 에서 시작하여 다음 페이지(여기에서는 페이지 M+1로서, 하위 페이지 M과 동일한 물리적 페이지에 대응하는 상위 페이지)가 B 레벨에서 판독함으로써 감지되며, 프로세스는 시간 t_3 에서 끝난다. 이때, 제 1 페이지(페이지 M에서 온)(하위)는 페이지 N에 메모리로 다시 프로그램될 준비가 되고 페이지 M+1로부터 판독된 데이터는 래치 내에 유지되어 있으며 수정/체크되기 위해 전송될 수 있다. 이들 프로세스들 둘 다는 같은 시간, 여기

에서는 t_3 에서 시작할 수 있다. 위에 기술된 전형적인 시간 값들을 사용하여, 페이지 M+1으로부터의 데이터는 시간 t_4 까지 토글 아웃 되고 수정되었지만, 풀 시퀀스 전환을 구현하지 않는 실시예에 있어서, 메모리는 페이지 N+1에 제 2의 판독된 한 페이지의 데이터(페이지 M+1에서 온)를 기입하기를 시작하기 위해 페이지 N이 시간 t_5 에서 끝날 때까지 기다릴 것이다.

[0192] 페이지 N+1이 상위 페이지이기 때문에, 이의 기입은 처음엔 B 레벨에서 검증부터 시작하며, C 레벨은 t_6 에서 추가된다. 일단 타겟 상태 B를 갖는 저장 요소들이 모두가 시간 t_7 에서 록아웃되면(또는 최대 카운트에 도달되면), B 상태 검증은 행해지지 않는다. 위에 기술된 바와 같이, 본 발명의 몇몇의 주요 면들에 따라, 이것은 데이터 래치가 자유롭게 되게 하며, 진행중의 기입 동작은 보류되며, 판독 동작(보류된 프로그램/검증 동작과는 다른 주소에서)이 삽입되고, 기입은 중단된 곳에서 재개하며, 삽입된 기입 동작에서 감지된 데이터는 개재된 기입 동작이 계속 진행되는 동안 토글 아웃 될 수 있다.

[0193] 시간 t_7 에서 삽입된 기입 동작은 여기에서는 하위 페이지 M+2에 대해 수행된다. 이 감지는 시간 t_8 에서 종료되며 페이지 N+1의 기입은 다시 회복하며 페이지 M+2로부터 데이터는 동시에 토글 아웃 되고 수정된다. 이 예에서, 페이지 N+1은 페이지 M+2가 시간 t_{10} 에서 종료되기 전에 시간 t_9 에서 프로그래밍을 종료한다. 시간 t_{10} 에서 페이지 M+2에서 오는 데이터의 기입이 시작될 수도 있을 것이지만, 이 실시예에서는 대신에 페이지 M+3의 판독이 먼저 실행되어, 이 페이지의 데이터가 토글 아웃 및 수정되게 함으로써 시간 t_{11} 에서 시작하는, 페이지 M+2로부터 오는 데이터의 페이지 N+2에의 기입 뒤로 은닉되게 한다. 이어서 프로세스는 도면의 앞의 부분들에서처럼 계속 되는데, 그러나 페이지 번호들이 바뀔에 따라, 카피 프로세스가 중지될 때까지 시간 t_{11} 은 시간 t_3 에 대응하고 시간 t_{12} 는 시간 t_4 에 대응하고, 등등이 된다.

[0194] 도 19b는 하위 페이지인 것으로 취해지는 페이지 M인 하위 페이지의 판독부터 시작하고 어떠한 데이터도 이전에 캐시되어 있지 않다고 가정하는 프로세스를 다시 도시한다. 도 19b는 시간 t_4 에서 풀 시퀀스 기입으로의 전환을 구현함으로써 도 19a과는 다르다. 이것은 도 19a의 시간(t_5-t_4)까지 프로세스를 개략적으로 가속시킨다. 시간 t_4 (도 19a에서 $=t_5$), 풀 시퀀스 전환에 관계된 다양한 변경들은 앞에 기술된 바와 같이 구현된다. 그렇지 않다면, 프로세스는 시간 t_7 과 시간 t_{12} 사이에서 발견되는 본 발명의 면들을 포함하여, 도 19a의 프로세스와 유사하다.

[0195] 둘 다에서 상태들이 주어진 시간에 검증되는 것들로서 페이지 카피 프로세스들 및 데이터 기입을 수반하는 여기 기술된 그외의 기술들은 참조로 여기 포함시키는 미국특허공개번호 US-2004-0109362-A1에 기술하는 방침을 따라, 지능형으로 선택될 수 있다. 예를 들면, 풀 시퀀스 기입에서, 기입 프로세스는 A 레벨만을 검증하는 것을 시작할 수 있다. A가 검증된 후에, 어떤 비트들이 통과하였는지를 알기 위해 체크된다. 그러하다면, B 레벨이 검증 국면에 추가될 수 있다. A 레벨 검증은 이를 타겟 값들로서 갖는 모든 저장 유닛들이 검증된 후에(또는 설정가능한 파라미터에 기초하여 최대 카운트를 제외하고) 제거될 것이다. 유사하게, B 레벨에서 검증들 후에, C 레벨의 검증이 추가될 수 있고, B 레벨 검증은 이를 타겟 값들로서 갖는 모든 저장 유닛들이 검증된 후에(또는 설정가능한 파라미터에 기초하여 최대 카운트를 제외하고) 제거된다.

[0196] 배경 데이터를 다른 동작들을 위해 캐시하는 프로그래밍 동작이 바람직한 복수-상태 코딩에 관하여 기술된다.

[0197] 4-상태 메모리를 위한 바람직한 "LM" 코딩

[0198] 도 20은 2비트 논리 코드("LM" 코드)로 인코딩된 4-상태 메모리의 프로그래밍 및 판독을 도시한 것이다. 이 코드는 무장애(fault-tolerance)를 제공하며 유핀 효과에 기인한 이웃 셀 커플링을 완화시킨다. 도 20은 각 메모리 셀이 LM 코드를 사용하여 2비트의 데이터를 저장할 때 4-상태 메모리 어레이의 임계 전압 분포들을 도시한 것이다. LM 코딩은 상위 및 하위 비트들이 상태들 "A" 및 "C"에 대해 반전되는 점에서 종래의 그레이 코드와는 다르다. "LM" 코드는 미국특허 6,657,891에 개시되어 있으며 전하들에 큰 변화를 요구하는 프로그램 동작들을 피함으로써 이웃한 플로팅 게이트들간에 전계효과 커플링을 감소시키는데에 이점이 있다.

[0199] 코딩은 상위 및 하위의 2비트들이 개별적으로 프로그램 및 판독될 수 있게 설계된다. 하위 비트를 프로그래밍할 때, 셀의 임계 레벨은 프로그램되지 않은 영역에 남아 있거나 임계 윈도우의 "하위 중간" 영역으로 옮겨진다. 상위 비트를 프로그래밍할 때, 이들 두 영역들 중 어느 하나에 임계 레벨은 약간 더 높은 레벨로 그러나 임계

원도우의 1/4 이하로 더 높아진다.

[0200] 데이터 패턴들의 의사(pseude) 및 사용자 구동의 랜덤화

[0201] 메모리 EEPROM 또는 칩 및 랜덤화의 방법의 다양한 실시예들은 증가된 NAND 스트링 저항, 감소된 내구성 및 신뢰성, 및 요구되지 않는 커플링과 같은 반복적인 데이터 저장 패턴들에 기인한 문제들을 최소화하려는 것이다. 본 발명의 의사 랜덤화 기술들은 실용적이며 데이터 처리 역량 관점에서, 이들은 구현하기에 비용이 많이 들지 않는다.

[0202] 본 발명은 플래시 메모리 칩 상에 저장된 데이터의 의사-랜덤화 및 사실상 사용자에게 기초한 랜덤화를 구현하는 상이한 실시예들 및 방법을 포함한다. 모든 실시예들은 플래시 EEPROM에 구현될 간단하고 약간의 회로 수정들만을 요구하는 이점이 있다. 이것은, 랜덤화 기술들 및 회로가 계산 집약적이지 않으며 있다고 해도 약간의 수행 불이익으로 구현되기 때문에 주목할 만하다. 본 발명의 해결책들은 랜덤화가 언제든지 용이하게 활성화 또는 비활성화될 수 있는 점에서 유연하다. 또한, 어떤 실시예들에서 이용되는 의사 랜덤화의 패턴은 많은 방법들로 다양하게 될 수 있고 시간적으로 쉽게 변경될 수 있다.

[0203] 도 21은 랜덤화 프로세스에 관계된 EEPROM 또는 메모리 칩의 주 성분들을 도시한 것이다. 칩(600)은 메모리 어레이(602), 주변 회로에 레지스터(들)(610), 및 멀티플렉서(614)를 포함한다. 칩(600)의 다른 성분들은 추가한 도면들에 도시되고 이들을 참조하여 기술될 것이다. 레지스터(610)는 복수의 비트들을 유지할 수 있고 복수의 레지스터들을 포함할 수 있다. 일부 실시예들에서 이것은 시프트 레지스터로서 기능한다. 메모리 어레이(602)는 은닉된 영역(604) 및 사용자 데이터 영역(606)을 포함한다. 은닉된 영역은 펌웨어, 그리고 메모리 동작 제어 코드들과 같은 그의 다른 오버헤드 데이터를 저장하는데 사용될 수 있다. NAND 아키텍처에서, 앞에서 기술되었던 바와 같이, 데이터는 블록들로 구성되고, 이들 각각은 복수의 페이지들의 데이터를 포함할 수 있다. 어떤 실시예들에서는 레지스터(610)도, 멀티플렉서(614)도 없을 것이다.

[0204] 본 발명의 다양한 실시예들은 프로그램 교란들 또는 사용자 판독 교란들을 야기할 수 있는 특정 데이터 패턴들의 장기간의 반복된 저장을 감소 또는 제거할 것이다. 이것은 의사 랜덤 메커니즘들에 의해서 또는 사용자에게 의해 트리거된 랜덤화에 의해서 데이터의 엔코딩을 가변시킴으로써 그와 같이 행한다. 사용자 활동의 타이밍은 완전히 예측할 수 없기 때문에, 활동을 트리거로서 사용하는 것은 엔코딩 방식들의 사실상의 랜덤 시퀀스를 가져온다. 실시예들 각각은 NAND 스트링 저항 효과를 감소시키고, 메모리 내구성 및 신뢰성을 증가시키고, 플로팅 게이트간 커플링의 문제를 감소시킬 것이다.

[0205] 실시예들 각각은 플래시 EEPROM의 회로에 대한 최소의 수정만을 요구하며, 그러면서도 이와 아울러 데이터 저장의 랜덤화를 대폭 증가시킬 것이며, 그러므로 EEPROM의 성능을 증가시킬 것이다. 데이터의 랜덤화는 어레이 내에서 쉽게 활성화 또는 비활성화될 수 있다. 또한, 의사 랜덤화를 위한 시퀀스는 계속적으로 바뀔 수 있어 시스템 내에서 융통성을 제공한다.

[0206] 일 실시예에서, 0 또는 1일 수 있는 코드 또는 한 시퀀스의 비트들은 어레이(602)의 은닉된 영역(604)에 저장된다. 코드가 저장되는 은닉된 영역(604)의 부분은 "ROM 블록"이라 지칭될 수 있다. 코드는 2 이상의 비트들을 포함할 수 있으나, 바람직하게는 17 또는 그 이상의 비트들을 포함한다. 비트들이 많을수록 랜덤화는 더 클 것이다. 칩(600)이 기동되었을 때, 값이 레지스터(610)에 로딩된다. 레지스터 내 각 비트는 특정의 페이지 주소에 할당된다. 각 비트는 페이지의 페이지 주소와 비교되고, 비교에 기초하여, 페이지의 데이터의 엔코딩은 페이지에 대해 반전되거나 같은 상태(통과된)에 있게 될 것이다. 예를 들면, 비트에 대한 0 값은 데이터의 엔코딩 방식은 동일한 채로 있게 될 것임을 나타내는데 사용될 수 있고, 반면 레지스터 내 1의 값은 페이지 내 데이터의 엔코딩은 반전될 것임을 나타낼 수 있다. 코드가 블록 내 페이지들의 수보다 적은 비트들을 포함한다면, 코드는 하나 이상의 그룹의 하나 이상의 페이지들에 적용될 수도 있다. 즉, 코드는 모든 페이지들이 비교될 때까지 일련으로 반복하여 사용될 수 있다. 코드는 사이클들간에 변경될 수도 있다. 대안적으로, 코드는 멀티플렉서(614)를 통해 멀티플렉싱될 수 있고 따라서 한 코드의 한 비트는 사용자 데이터 영역(602)에 저장된 복수 페이지들의 데이터의 엔코딩을 결정할 것이다. 코드의 각 비트는 이것이 사용자 데이터의 어떤 부분에 대해 사용되는 엔코딩의 극성(polarity)을 변경하게 작용하기 때문에, 극성 비트라 칭해질 수 있다. 이것이 도 22a에 도시되었다. 이 경우에, 엔코딩은 페이지 주소에 기초하므로, 페이지들 0, N이 1의 극성을 가지며, 반면 페이지들 1, n+1은 극성 0을 가지며, 페이지들 2, n+2는 극성 1을 가지며, 등등임이 알려지게 된다. 그러므로 엔코딩이 페이지 주소에 기초하는 실시예에서, 용장 목적들을 위해 행해질 수 있을지라도, 페이지와 함께 극성 비트를 저장하는 것은 필요하지 않다.

[0207] 밑에 나타내고 도 22b에 전제된 표 1은 사용자 데이터의 부분들에 레지스터(610) 내 코드의 극성 비트들의 적용을 예시한다. 사용자 데이터의 임의의 부분이 비교되고 특정 극성 비트에 연관될 수 있으나, 기술된 바람직한 실시예들은 기본 유닛으로서 페이지를 예시한다.

[0208] [표 1]

레지스터 위치	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
코드 (극성 비트)	1	0	1	0	0	1	1	1	1	0	0	0	1	1	0	1	0
UD 오리지널 엔코딩	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
UD 후속 엔코딩	1	1	1	1	0	0	1	0	1	1	0	1	1	0	0	0	0
상태	ER		ER		B		C		ER		A		C		B		

[0209]

[0210] 표에서 알 수 있는 바와 같이, 코드의 각각의 (극성) 비트는 비트의 원래의 (데이터) 엔코딩이 동일한 채로 있게 될 것인지 아니면 바뀌게 될 것인지를 결정할 것이다. 예를 들면, 레지스터 위치 1을 보면, 이 위치에서 극성 코드는 1의 값을 갖는다. 따라서, 1이 데이터가 반전될 것임을 나타내는 실시예에서, 0으로서 저장된 사용자 데이터의 원 비트는 1의 값으로 반전될 것이다. 테이블은 복수 상태 셀을 예시하며, 여기서 2 비트들은 상태를 정의하는데 사용된다. 상태들은 도 20에 도시되었고, 도 20에서 알 수 있는 바와 같이, 상위 및 하위 비트는 상태들을 정의한다. 도 20에 도시된 유형의 2비트 또는 4상태 메모리 셀에서, (1:1)은 소거 ("ER") 또는 비-프로그램된 ("U") 상태를 정의하며, (0:1)은 상태 A를 정의하며, (0:0)은 상태 B를 정의하며, (1:0)은 상태 C를 정의한다. 상위 및 하위 비트들은 물리적으로 단일 메모리 셀 내에 위치될 수 있다. 동일 또는 상이한 코드가 또 다른 일 그룹의 데이터에 적용될 수도 있을 것이므로 비트 17에 대응하는 일 그룹의 데이터가 상태를 결정하기 위해 적용되는 다음 코드의 비트 1에 대응하는 데이터에 관련하여 사용될 것이다. 각 극성 레지스터는 대응하는 페이지 상에 모든 데이터에 대한 극성을 제어할 것이다. 하위 및 상위 비트들은 바람직하게는 동일 물리적 워드 라인들 상에 위치된다. 표 1에 주어진 예는 많은 워드라인들에 걸쳐 단순 패턴을 랜덤 패턴(들)으로 전환하기 위한 극성 비트들의 기능을 예시한다. 랜덤화는 동일 NAND 체인 구조상에 위치한 데이터에 대해 달성되고, 이의 예가 이 개념을 예시하기 위해 도 22c에 제공되어 있다.

[0211] 도 22c에서, 주어진 셀의 하위 및 상위 비트들은 NAND 스트링 또는 체인의 예시된 셀들의 각각에 예시되었다. 도시된 NAND 스트링은 단순히 예이며, 물론 스트링에 더 많은 또는 적은 셀들이 있을 수 있고, 도시된 것과는 다른 구조들이 본 발명에 사용될 수도 있다. 예를 들면, 3, 4, 또는 그 이상의 비트들을 저장하는 셀이 채용될 수도 있다. 또한, 특히 복수 상태 메모리에서, 비트 반전의 개념을 예시하기 위해 비트 레벨에 관한 적용이 보여졌을지라도, 극성 비트는 바람직하게는 한 페이지 또는 더 많은 페이지의 데이터에 적용하는 것을 상기해야 한다. 도 22c에서, 극성 비트들은 사용자 데이터의 각 비트에 적용되며, 극성 비트에 의해 반전 또는 통과된, 결과적인 사용자 데이터는 보관된(saved) 데이터로서 표기된다. 보관된 데이터는 이어서 메모리 어레이에 기입되고 랜덤화 동작의 결과로서 저장될 데이터이다. 알 수 있는 바와 같이, 도면의 우측에 나타난 상태들은 셀의 상위 및 하위 비트에 의해 정의된다. 도 22c에 사용된 "보관된 데이터"라는 용어는 표 1 및 도 22b에서 "사용자 데이터(UD) 후속 엔코딩"이라 칭해진 것에 대응한다.

[0212] 또 다른 실시예에서, 레지스터(610)는 시프트 레지스터로서 구성된다. 이러한 실시예에서, 코드의 모든 비트들이 한번에 사용되었던 실시예와는 대조적으로, 레지스터의 한 비트만이 한번에 사용될 것이다. 명령이 사용자에 의해 발행되었을 때, 시프트 레지스터는 다음 비트로 시프트할 것이다. 들어오는 사용자 페이지에 대해 사용되는 극성 비트는 마지막 레지스터 출력으로부터 온 것일 것이다. 이것은 명령의 상승에지에서 행해지는 것이 바람직하다. 트리거 명령은 프로그램 명령, 캐시 프로그램 명령, 판독 명령, 소거 명령, 또는 그의 사용자에 의해 발행된 명령일 수 있다. 프로그램 명령 신호의 예가 도 23b에 도시되었다. 명령에 연관된 클럭 신호가 도시되었으며, 명령의 실제화는 사용자 요청에 의해 트리거될 것이며, 이의 타이밍 및 유형은 예측할 수 없고 근본적으로 랜덤하다. 도 23a는 극성 비트를 결정하는 입력들 중 하나로서 사용자 명령에 연관된 클럭 신호를 도시한 것이다. 도면에서 다른 입력은 사용자 데이터이다. 극성 비트의 적용은 앞에서 기술된 바와 같이, 데이터의 엔코딩을 반전하거나 현재대로 놔둘 것이다.

[0213] 도 23c은 데이터 반전을 위한 제어 회로의 예를 도시한 것이다. 단일 인버터를 가진 데이터 경로는 엔코딩을 반전시키게 될 것이며, 반면 직렬로 2개의 인버터들을 가진 경로는 데이터 엔코딩 방식이 변경되지 않게 될 것이다. 이러한 경우에, 일 그룹의 데이터에 연관된 극성 비트는 이 일 그룹의 데이터와 함께 저장될 것이다. 예를

들면, 도 23d에서 알 수 있는 바와 같이, 한 페이지의 데이터(630)에 대해서, 사용자 영역(636)에 데이터에 대한 극성 비트(632)는 페이지(630)의 은닉 영역(634)으로 프로그램될 것이다. 페이지(630)가 판독될 때, 극성 비트는(632) 전송되어 출력 데이터를 제어하기 위해 래치될 것이며, 도 23e에 도시된 회로에 의해 달성되는 바와 같이, 반전되었다면 인코딩 방식을 다시 되돌릴 것이다. 이에 따라, 페이지의 극성은 이의 원 인코딩으로 되돌려질 것이다.

[0214] 시프트 레지스터에 의해 이용되는 코드의 패턴은 다양할 수 있고 서로 다른 적용들을 위해 수정될 수 있다. 모든 비트들이 0으로 설정된다면(0이 무 변화를 나타내는 경우에), 랜덤화는 비활성화될 것이다. 레지스터 내 비트들의 패턴은 의사 랜덤한 반면, 사용자 행동은 예측할 수 없고, 임의의 주어진 시간에 결과적인 극성 또한, 예측할 수 없고 랜덤하다. 사용자 행동의 두 예들은 다음과 같다. 1) 사용자는 몇개의 페이지들을 프로그램하며 몇 페이지들을 판독 또는 프로그램하기 위해서 또는 몇개의 블록들을 소거하기 위해 다른 주소로 도약하며, 마지막 프로그램이 행해졌던 블록으로 다시 되돌아가 더 많은 페이지들을 프로그램을 계속하며, 2) 사용자는 또 다른 주소로 도약함이 없이 순차로 모든 페이지들을 프로그램한다. 경우 1에서, 새로운 극성 비트가 각각의 사용자 명령에 대해 트리거될 수 있고, 반면 경우 2에서 순차적 프로그래밍이 이용되고 한 극성 비트에 기초할 것이다. 그러므로 사용자가 저장하기를 원하는 원 데이터가 두 경우들에 있어서 동일할 수 있을지라도, 메모리 내 최종 프로그램된 데이터는 이들 두 경우들에서 여러 개개의 페이지들 중 적어도 일부 및 그룹들의 페이지들에 대해 다를 것이다. EEPROM은 전형적으로 제어기 칩에 의해 제어되며 "사용자"의 행동들의 일부는 제어기 칩의 행동일 수 있는 것에 유의한다.

[0215] 또 다른 실시예에서, 극성 비트는 앞에서 기술된 캐시 프로그램 동작과 같이, 사용자 명령의 결과로서 랜덤하게 발생된다. 이 실시예는 동기화되지 않는 2개의 입력들을 이용한다. 첫 번째는 앞에서 언급된 바와 같이 예측불가한 것인 사용자 명령들의 타이밍이다. 두 번째는 유한 상태 머신 클럭이다. 어떤 메모리 시스템들에서, 유한 상태 머신 클럭은 어떤 시간들에서만 활성이 되며(예를 들면, 캐시 동작들 동안), 반면 다른 시스템들에서 이것은 항상 활성일 수 있다. 이 실시예의 이 기술은 메모리 시스템의 유한 상태 머신 클럭이 활성일 때는 언제나 활용가능하다.

[0216] 사용자 명령 클럭 신호의 상승에지에서, 유한 상태 머신("finite state machine"; "FSM") 클럭의 레벨 또는 상태가 참조된다. 상태는 도 24a에 보인 바와 같이, 하이이거나 로우일 수 있다. 로우 상태는 0의 극성 비트에 대응할 수 있다(반대의 대응이 가능할지라도). 시간 t=0에서, FSM은 로우이며 이에 따라 극성 비트는 0이 될 것이며, 앞에서 언급된 바와 같이, 데이터 인코딩의 무 변화를 나타낸다. 시간 t=1에서, FSM은 하이이며, 극성 비트는 1이 될 것이며, 반면 시간 t=3에서, FSM은 다시 로우 상태에 있다. 일부 실시예들에서, 실행 명령이 발행되어 이것이 감지되는 즉시 은닉 영역(634)에 극성 비트 632가 로딩된다. 다른 실시예들에서 이것은 시스템의 또 다른 메모리에 일시적으로 저장될 수도 있다. 도 24b는 위에 기술된 바와 같은 극성 비트를 결정하기 위한 회로를 도시한 것이다. 인버터는 상승에지에서 트리거되는 것이 바람직할 것이다.

[0217] 발명의 실시예들이 기술되었지만, 본 발명은 이들 예시적인 실시예들로 제한되는 것은 아니며 첨부된 청구항들에 의해 정의됨을 알아야 할 것이다.

산업상 이용 가능성

[0218] 상술한 바와 같이, 본 발명은, 일반적으로 디지털 장치들에 및 이로부터 큰 파일들을 저장 및 전송하기 위해 사용되는 메모리 카드들 및 휴대 유니버설 직렬 버스("USB") 플래시 메모리 드라이브들과 같은 휴대 대량 저장 장치들, 보다 구체적으로는 이들 드라이브들에서 반복적으로 저장되는 비트 패턴들의 영향을 최소화하는 것을 제공하는데 사용된다.

도면의 간단한 설명

[0059] 도 1a 내지 도 1e는 비휘발성 메모리 셀들의 여러 예들을 개요적으로 도시한 도면이다.

[0060] 도 2는 메모리 셀들의 NOR 어레이의 예를 도시한 도면이다.

[0061] 도 3은 도 1d에 도시된 바와 같은 메모리 셀들의 NAND 어레이의 예를 도시한 도면이다.

[0062] 도 4는 플로팅 게이트가 어떤 한 시간에 저장하고 있을 수 있을 수 있는 4개의 서로 다른 전하들(Q1-Q4)에 대한 소스-드레인 전류와 제어 게이트 전압 간의 관계를 도시한 도면이다.

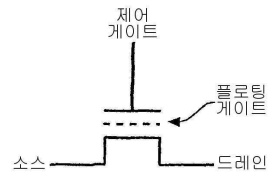
[0063] 도 5는 행 및 컬럼 디코더들을 통해서 판독/기입 회로들에 의해 액세스될 수 있는 메모리 어레이의 전형적인 배

열을 개요적으로 도시한 도면이다.

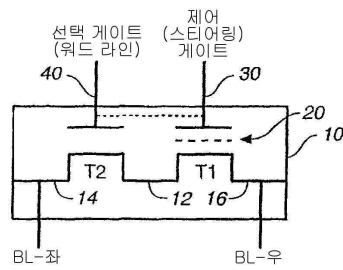
- [0064] 도 6a는 개개의 판독/기입 모듈의 개요적 블록도이다.
- [0065] 도 6b는 한 스택의 판독/기입 모듈들에 의해 통상적으로 구현되는 도 5의 기입/판독 스택을 도시한 것이다.
- [0066] 도 7a는 본 발명의 개선된 프로세서가 구현되는, 한 बैं크의 분할된 판독/기입 스택들을 구비한 콤팩트 메모리 장치를 개요적으로 도시한 것이다.
- [0067] 도 7b는 도 7a에 도시된 콤팩트 메모리 장치의 바람직한 배열을 도시한 도면이다.
- [0068] 도 8은 도 7a에 도시된 판독/기입 스택 내 기본 성분들의 일반적인 배열을 개요적으로 도시한 것이다.
- [0069] 도 9는 도 7a 및 도 7b에 도시된 판독/기입 회로들간에 판독/기입 스택들의 한 바람직한 배열을 도시한 것이다.
- [0070] 도 10은 도 9에 도시된 공통 프로세서의 개선된 실시예를 도시한 것이다.
- [0071] 도 11a는 도 10에 도시된 공통 프로세서의 입력 로직의 바람직한 실시예를 도시한 것이다.
- [0072] 도 11b는 도 11a의 입력 로직의 진리표를 도시한 것이다.
- [0073] 도 12a는 도 10에 도시된 공통 프로세서의 출력 로직의 바람직한 실시예를 도시한 것이다.
- [0074] 도 12b는 도 12a의 출력 로직의 진리표를 도시한 것이다.
- [0075] 도 13은 본 발명의 2비트 실시예에서 본 논의에 관계된 어떤 특정한 요소들을 도시한 도 10을 간이화한 것이다.
- [0076] 도 14는 하위 데이터 데이터가 판독되는 상위 페이지 프로그램에 대해 도 13과 동일한 요소들에 대한 래치 배열을 나타낸 것이다.
- [0077] 도 15는 단일 페이지 모드에서 캐시 프로그램의 면들을 도시한 것이다.
- [0078] 도 16은 풀(full) 시퀀스로 하위 페이지의 전환에서 사용될 수 있는 프로그래밍 과정을 도시한 것이다.
- [0079] 도 17은 풀 시퀀스 전환에 의한 캐시 프로그램 동작에서 상대적 타이밍을 도시한 것이다.
- [0080] 도 18은 캐시 페이지 카피 동작에서 래치들의 사용을 기술한 것이다.
- [0081] 도 19a 및 도 19b는 캐시 페이지 카피 동작들에서 상대적 타이밍들을 도시한 것이다.
- [0082] 도 20은 각 메모리 셀이 LM 코드를 사용하여 2비트의 데이터를 저장할 때 4-상태 메모리 어레이의 임계 전압 분포들을 도시한 것이다.
- [0083] 도 21은 EEPROM 또는 메모리 칩(600)의 어떤 성분들의 개요적 블록도이다.
- [0084] 도 22a는 페이지 주소에 의한 여러 페이지들의 데이터의 엔코딩 방식 및 극성 비트를 도시한 것이다.
- [0085] 도 22b는 사용자 데이터의 엔코딩을 변환하는데 사용되는 17비트 코드의 예의 적용을 예시한 표이다.
- [0086] 도 22c는 NAND 체인/스트링의 메모리 셀들에 저장된 상위 및 하위 비트들에 극성 비트들의 적용을 도시한 것이다.
- [0087] 도 23a는 명령 클럭 신호의 기능으로서 엔코딩 방식 판정을 도시한 것이다.
- [0088] 도 23b는 명령의 클럭신호이다.
- [0089] 도 23c는 데이터 엔코딩 판정 및 반전을 위한 제어회로의 실시예를 도시한 것이다.
- [0090] 도 23d는 극성 비트가 저장된 한 페이지의 사용자 데이터를 도시한 것이다.
- [0091] 도 23e는 판독 동작들을 위해 잠재적으로 반전된 데이터의 엔코딩을 되돌리기 위한 제어회로의 실시예를 도시한 것이다.
- [0092] 도 24a는 명령 클럭 신호의 상승에지에서 FSM 클럭의 값을 나타낸 명령 클럭 신호 및 유한 상태 머신 클럭 신호의 시간선이다.
- [0093] 도 24b는 도 24a에 도시된 명령 클럭 신호의 기능으로서 극성 비트를 판정하는 회로를 도시한 것이다.

도면

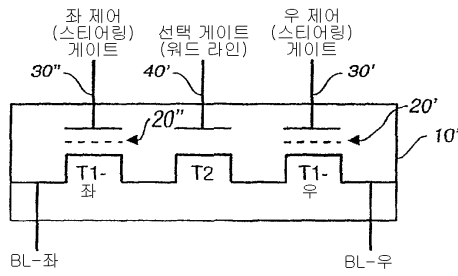
도면1a



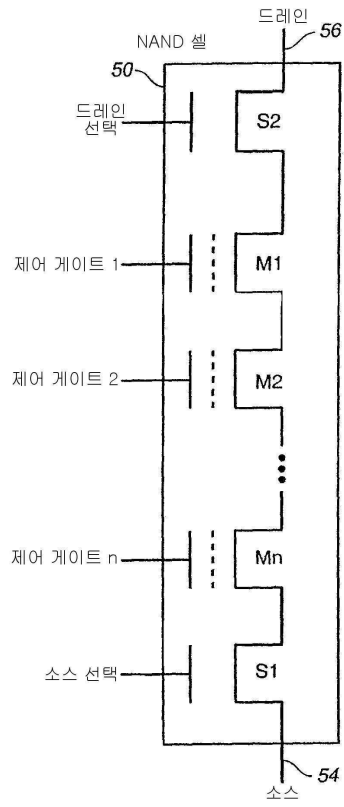
도면1b



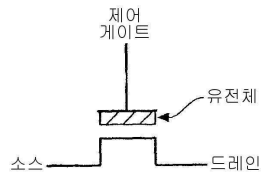
도면1c



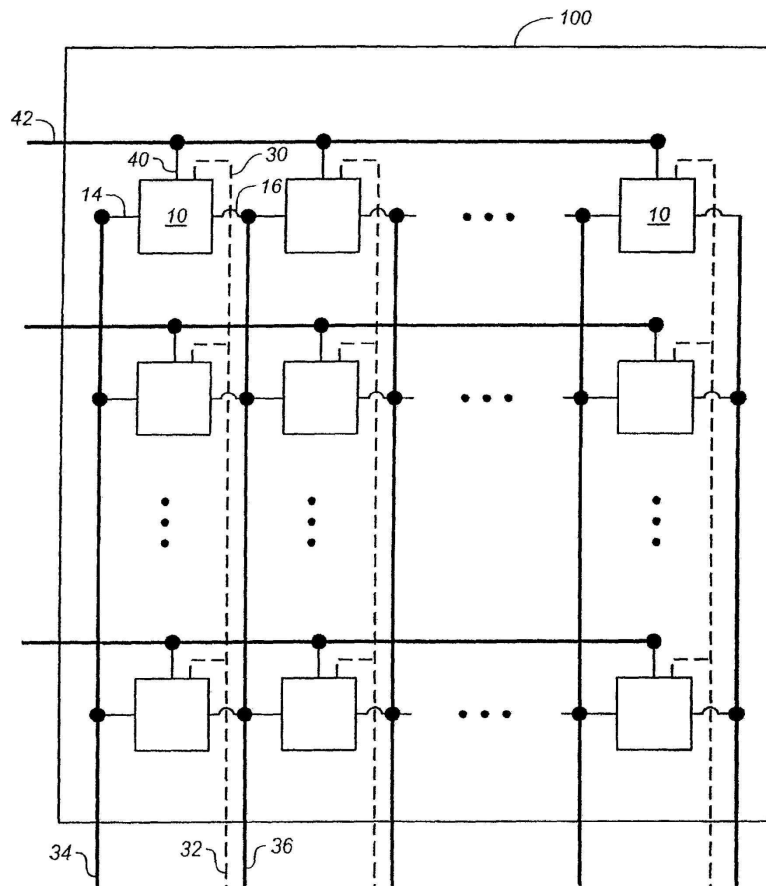
도면1d



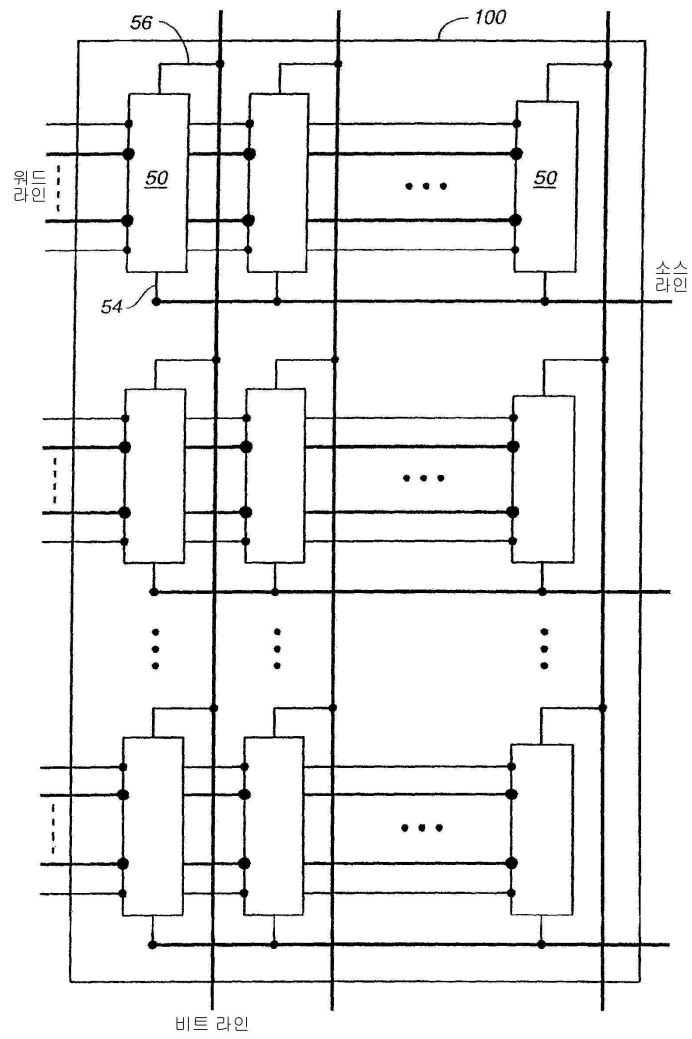
도면1e



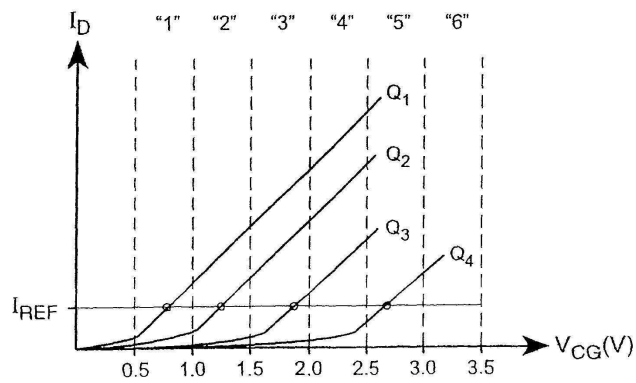
도면2



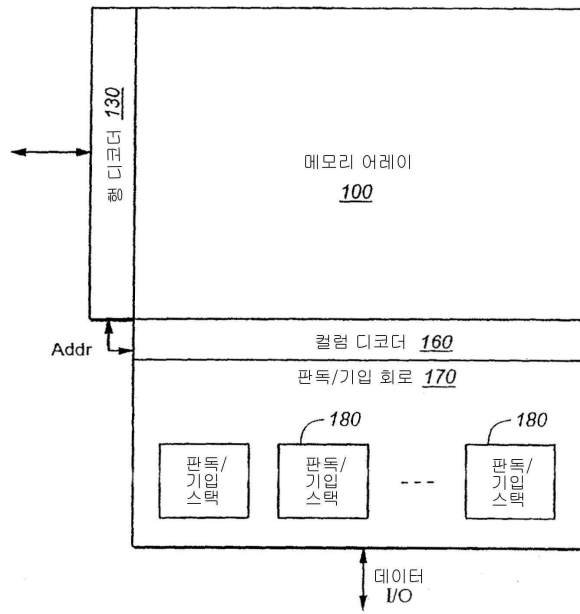
도면3



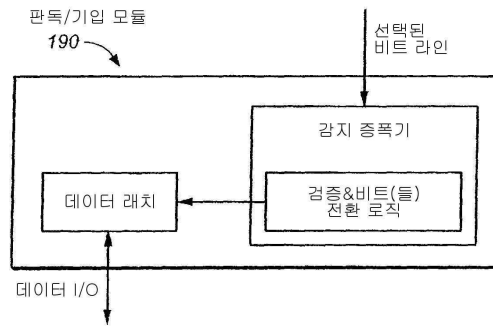
도면4



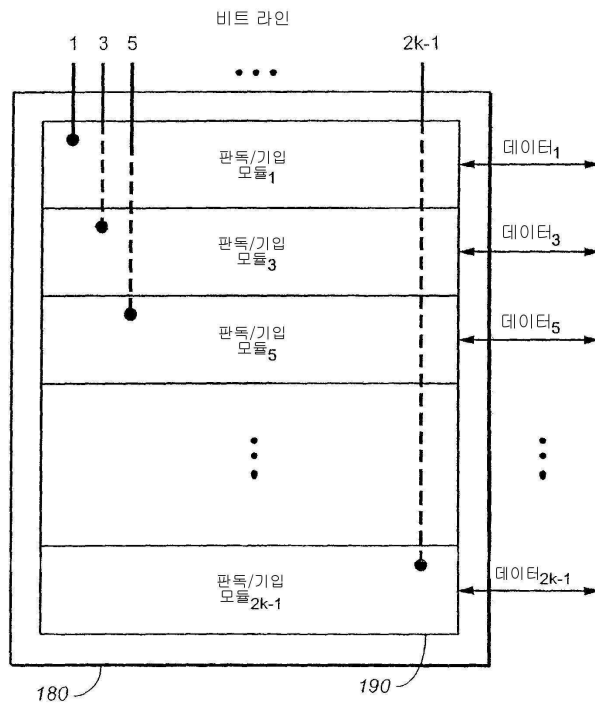
도면5



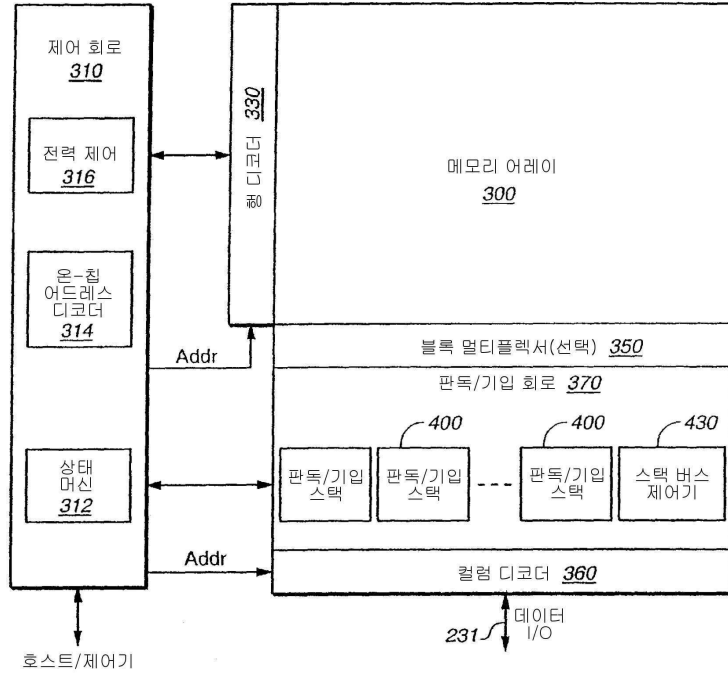
도면6a



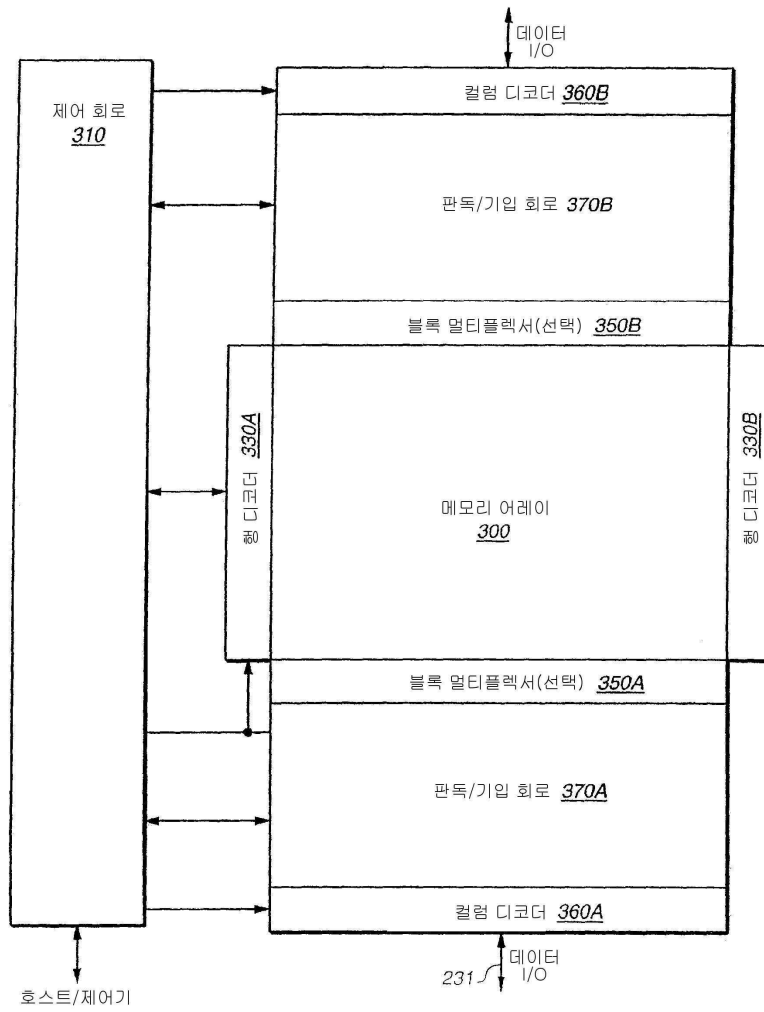
도면6b



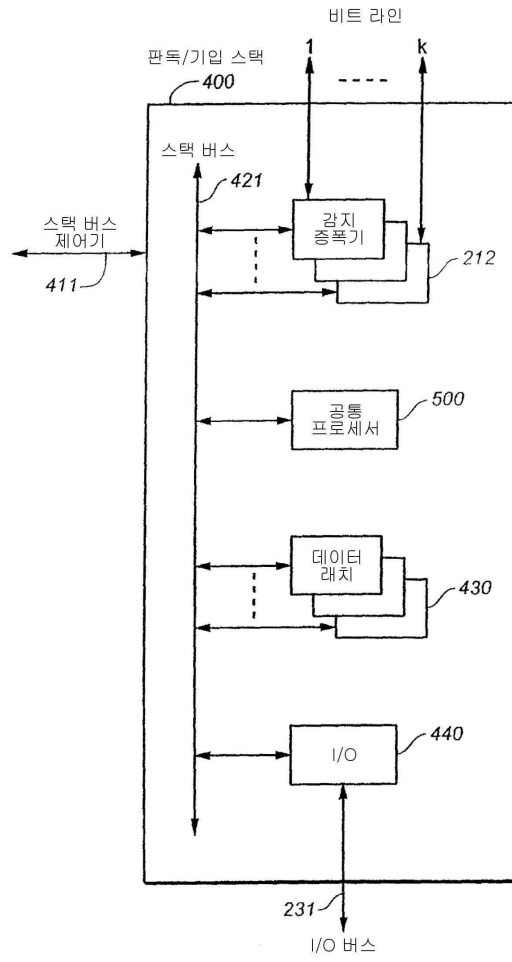
도면7a



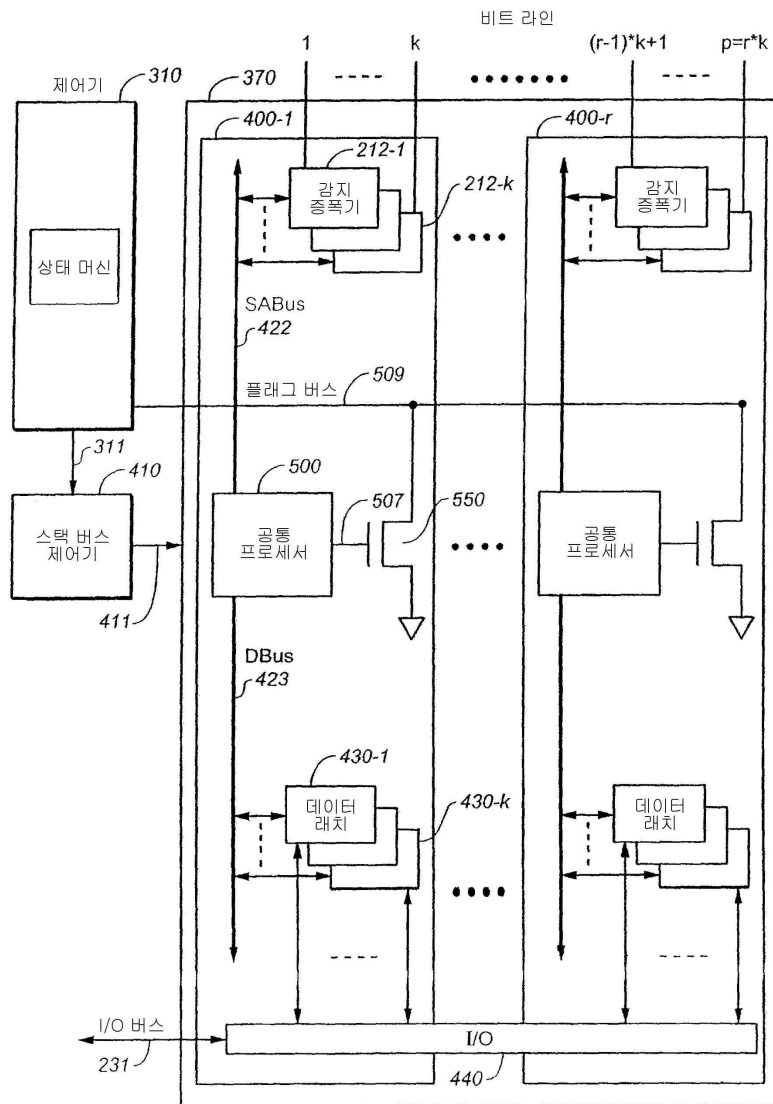
도면7b



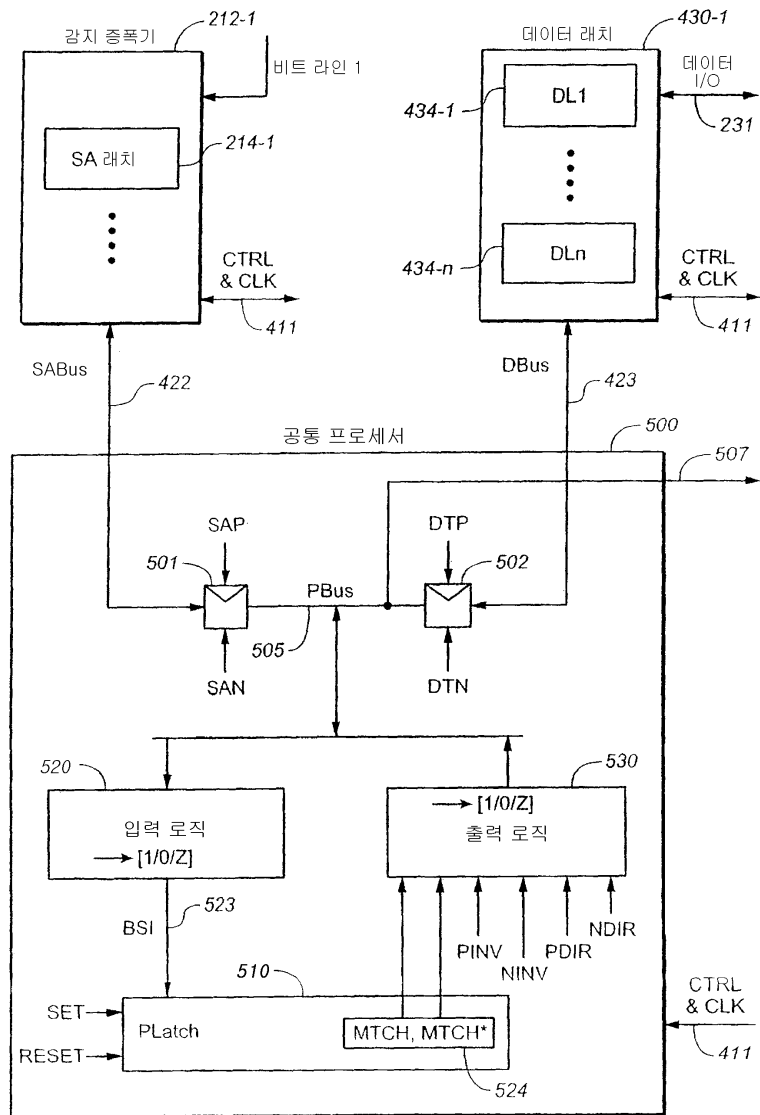
도면8



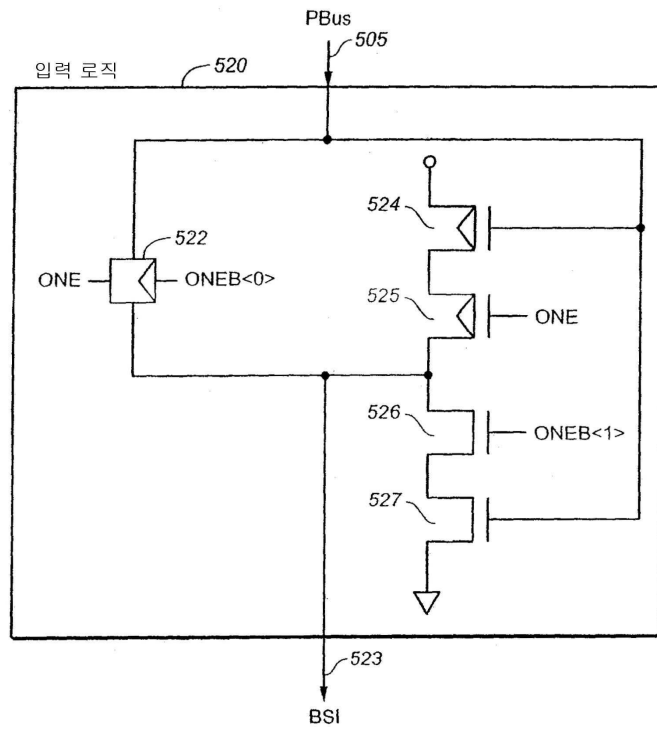
도면9



도면10



도면11a

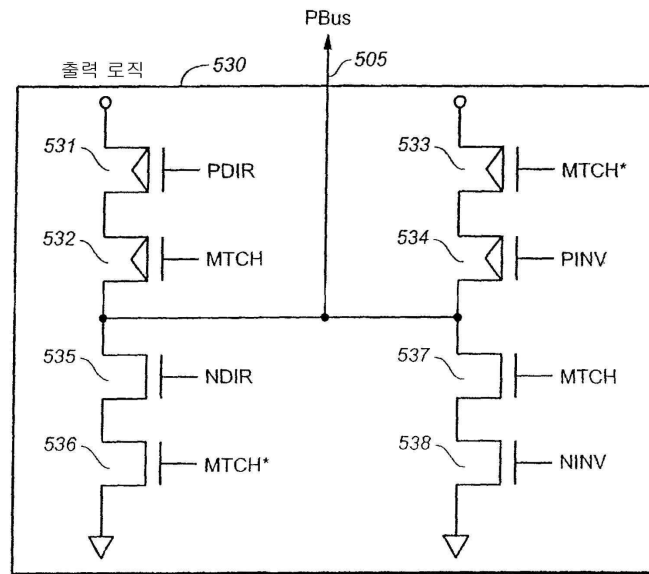


도면11b

입력 로직 진리표

전송 모드	ONE	ONEB<0>	ONEB<1>	PBus (Input)	BSI (Output)
패스스루	1	0	0	PBus	PBus
반전	0	1	1	PBus	PBus*
플로트	1	1	0	PBus	플로트

도면12a



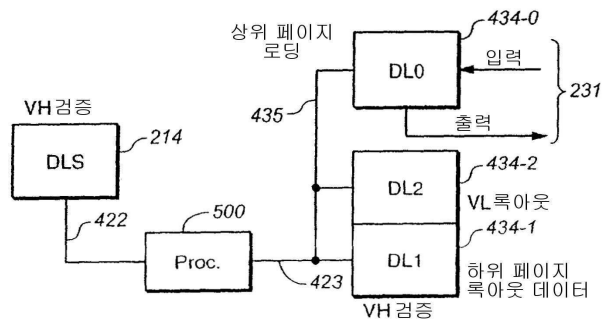
도면12b

출력 로직 진리표

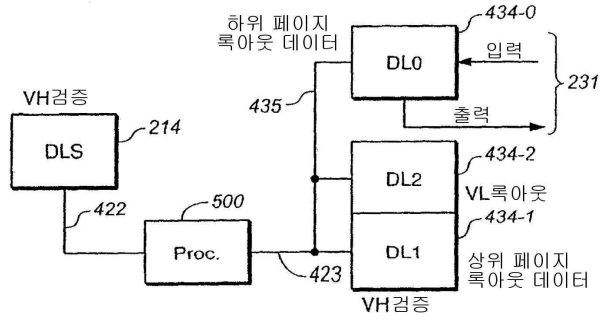
전송 모드	PINV	NINV	PDIR	NDIR	MTCH	PBus (Output)
패스스루	D	D	D	1	0	0
	0	D	D	D	1	1
반전	D	D	0	D	0	1
	D	1	D	D	1	0
플로트	D	D	D	D	X	Z
프리차지	0	D	0	D	X	1

(디폴트 값 : PINV=1, NINV=0, PDIR=1, NDIR=0)

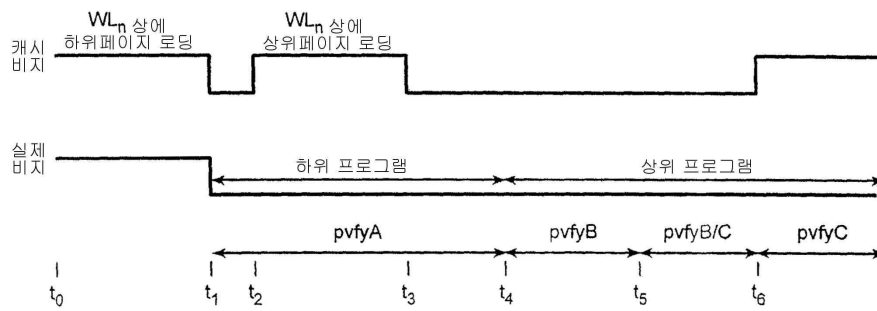
도면13



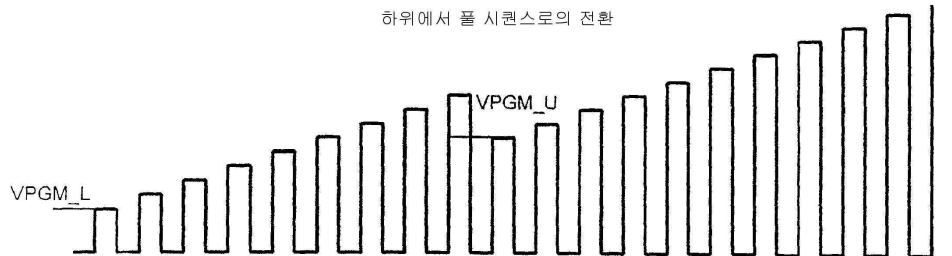
도면14



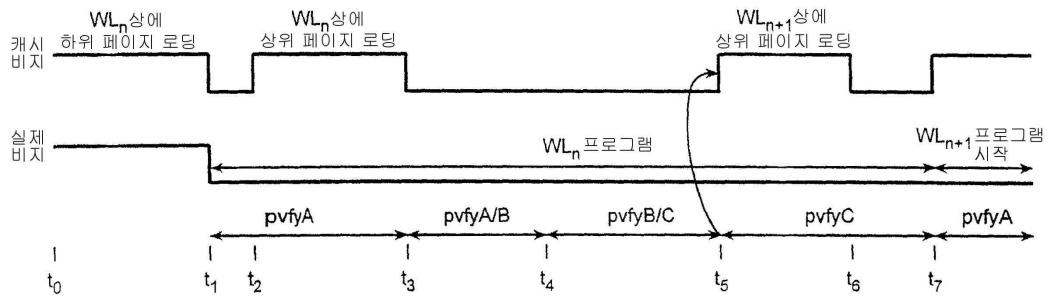
도면15



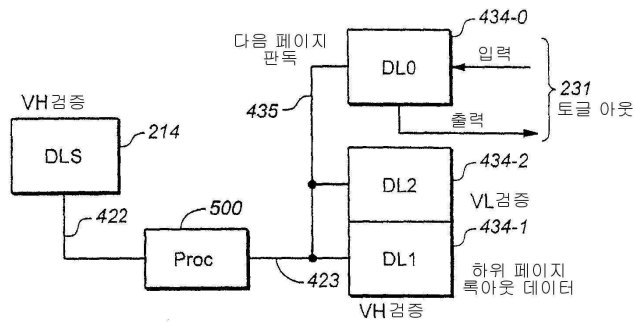
도면16



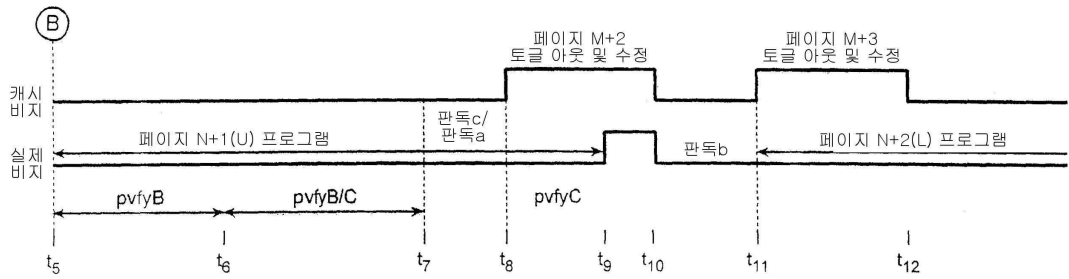
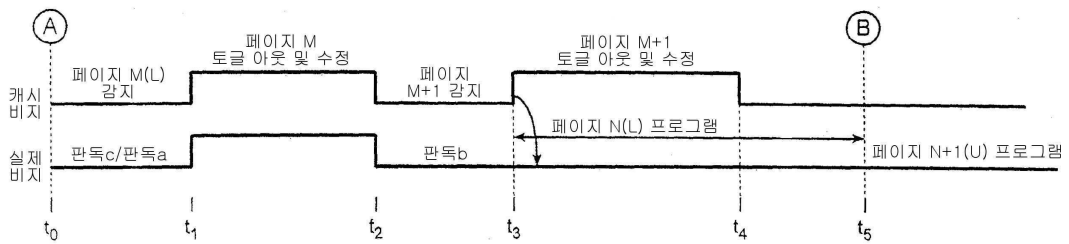
도면17



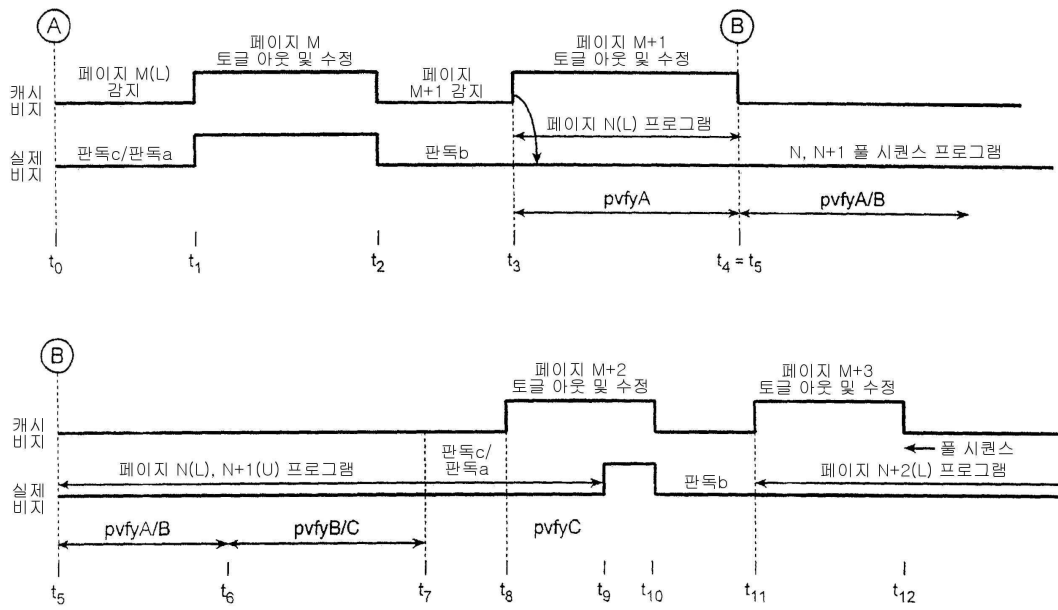
도면18



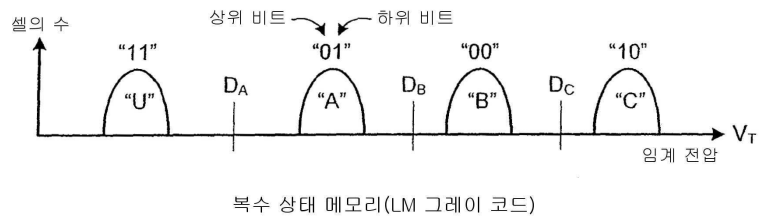
도면19a



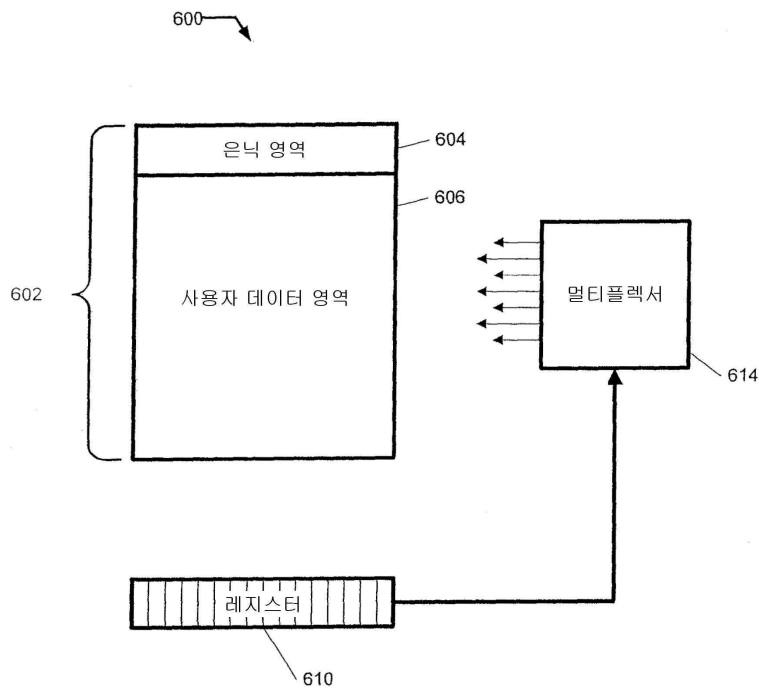
도면19b



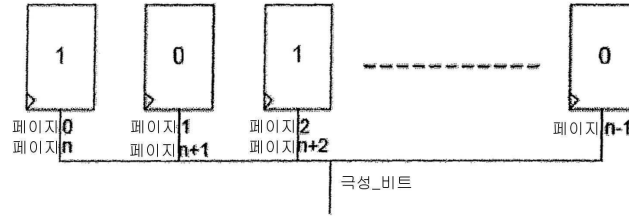
도면20



도면21



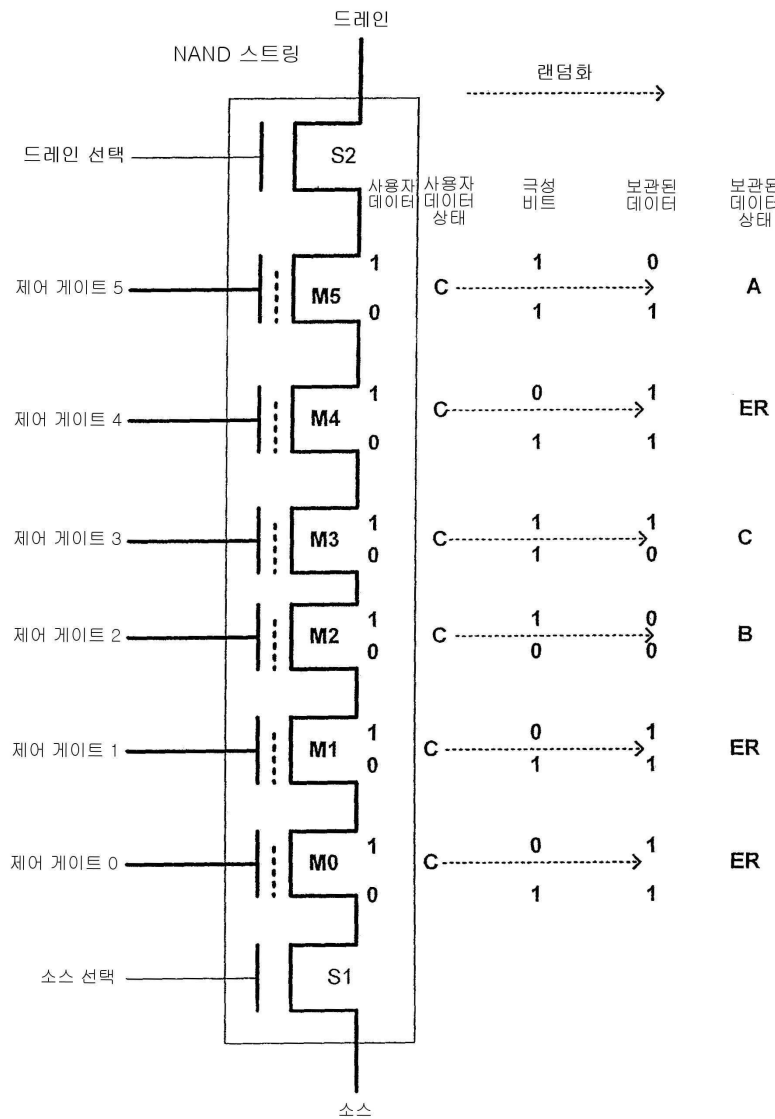
도면22a



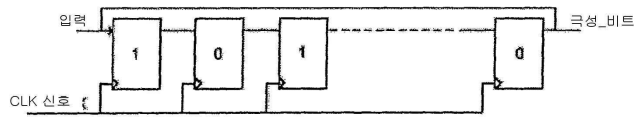
도면22b

레지스터 위치	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
코드 (극성 비트)	1	0	1	0	0	1	1	1	1	0	0	0	1	1	0	1	0
UD 오리지널 엔코딩	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
UD 후속 엔코딩	1	1	1	1	0	0	1	0	1	1	0	1	1	0	0	0	0
상태	ER		ER		B		C		ER		A		C		B		

도면22c



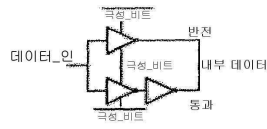
도면23a



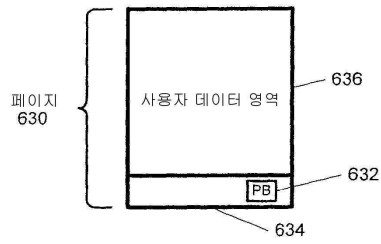
도면23b



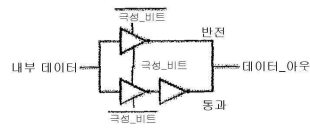
도면23c



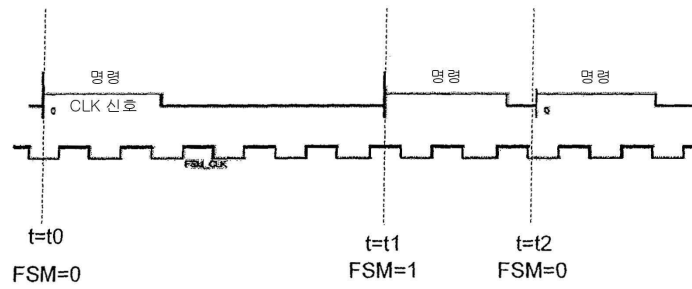
도면23d



도면23e



도면24a



도면24b

