(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2019-0104740
(43) 공개일자 2019년09월11일

(51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01) *G11C 5/14* (2006.01)
(52) CPC특허분류
G06F 3/061 (2013.01)
G06F 3/0616 (2013.01)
(21) 출원번호 10-2018-0025223
(22) 출원일자 2018년03월02일
심사청구일자 2018년06월04일

(71) 출원인
고려대학교 산학협력단
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)
(72) 발명자
김선욱
경기 남양주시 도농로 34 (도농동, 부영그린타운 2014 동 2003 호)
한미선
서울 동대문구 무학로 193 (용두동, 삼영오피스텔)
(74) 대리인
이기성

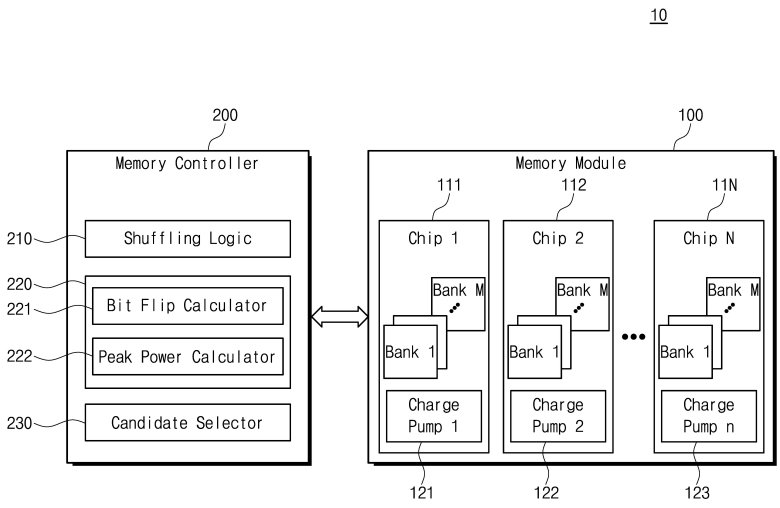
전체 청구항 수 : 총 16 항

(54) 발명의 명칭 **메모리 장치 및 메모리 장치의 인코딩 방법**

(57) 요 약

본 발명은 메모리 장치에 관한 것으로, 좀 더 자세하게는 가변 저항 메모리 셀들을 포함하는 메모리 장치에 관한 것이다. 본 출원의 일 실시 예에 따른 메모리 장치는 적어도 하나의 차지 펌프를 포함하는 메모리 모듈; 및 상기 메모리 모듈을 제어하는 컨트롤러를 포함하며, 상기 컨트롤러는 해시를 이용하여, 쓰기 요청된 데이터에 대응하는 복수의 데이터 후보를 생성하는 셔플링 로직; 상기 메모리 모듈에 저장된 데이터와 상기 복수의 데이터 후보의 비교 결과에 기초하여, 상기 복수의 데이터 후보 각각에 대한 비트 플립의 개수 및 파워 소모를 연산하는 계산부; 및 상기 비트 플립의 개수와 상기 피크 파워 소모에 기초하여, 상기 복수의 데이터 후보 중 상기 메모리 모듈에 저장될 데이터를 선택하는 후보 선택부를 포함한다. 본 출원의 실시 예에 따른 메모리 장치는 향상된 쓰기 성능 및 메모리 수명을 가질 수 있다.

대 표 도



(52) CPC특허분류
G06F 3/0658 (2013.01)
G11C 5/145 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 10052653
부처명 산업통상자원부
연구관리전문기관 한국산업기술평가원
연구사업명 산업기술혁신사업
연구과제명 Processing in Memory 반도체 아키텍처 및 데이터 집약적 응용처리를 위한 병렬처리 기술 개발
기 여 율 1/1
주관기관 전자부품연구원
연구기간 2015.06.01 ~ 2020.05.31

명 세 서

청구범위

청구항 1

적어도 하나의 차지 펌프를 포함하는 메모리 모듈; 및

상기 메모리 모듈을 제어하는 컨트롤러를 포함하며,

상기 컨트롤러는

해시 후보를 이용하여, 쓰기 요청된 데이터에 대응하는 복수의 데이터 후보를 생성하는 셔플링 로직;

상기 메모리 모듈에 저장된 데이터와 상기 복수의 데이터 후보의 비교 결과에 기초하여, 상기 복수의 데이터 후보 각각에 대한 비트 플립의 개수 및 파워 소모를 연산하는 계산부; 및

상기 비트 플립의 개수와 상기 피크 파워 소모에 기초하여, 상기 복수의 데이터 후보 중 상기 메모리 모듈에 저장될 데이터를 선택하는 후보 선택부를 포함하는, 메모리 장치.

청구항 2

제1 항에 있어서,

상기 셔플링 로직은 상기 쓰기 요청된 데이터의 비트 위치와 상기 해시에 대한 논리 연산 결과에 기초하여, 상기 복수의 데이터 후보를 생성하는, 메모리 장치.

청구항 3

제1 항에 있어서,

상기 계산부는

상기 메모리 모듈에 저장된 데이터와 상기 복수의 데이터 후보를 비교 결과에 기초하여, 비트 플립의 개수를 계산하는 비트 플립 계산부; 및

상기 비트 플립 정보에 기초하여, 상기 복수의 데이터 후보 각각에 대한 피크 파워 소모량을 계산하는 피크 파워 계산부를 포함하는, 메모리 장치.

청구항 4

제3 항에 있어서,

상기 메모리 모듈은

복수의 칩들; 및

상기 복수의 칩들에 대응하는 복수의 차지 펌프들을 포함하며,

상기 피크 파워 계산부는 상기 복수의 칩들 각각에 대한 파워 소모량을 계산하고, 상기 피크 파워 소모량은 상기 복수의 칩들 각각에 대한 파워 소모량 중 가장 큰 파워 소모량인, 메모리 장치.

청구항 5

제3 항에 있어서,

상기 메모리 모듈은

복수의 칩들; 및

상기 복수의 칩들에 전원을 공급하는 차지 펌프를 포함하며,

상기 피크 파워 계산부는 상기 복수의 칩들 전체에서 소모되는 파워 소모량을 합산하여 상기 피크 파워 소모량

을 계산하는, 메모리 장치.

청구항 6

제3 항에 있어서,

상기 후보 선택부는 상기 비트 플립 정보에 기초하여, 상기 복수의 데이터 후보들 중 적어도 한 개의 데이터 후보를 선택하는, 메모리 장치.

청구항 7

제6 항에 있어서,

상기 후보 선택부는 상기 복수의 데이터 후보들 중 최소 비트 플립의 개수를 갖는 데이터 후보 및 상기 최소 비트 플립의 개수를 갖는 데이터 후보보다 소정 개수 많은 비트 플립을 갖는 데이터 후보를 선택하는, 메모리 장치.

청구항 8

제6 항에 있어서,

상기 후보 선택부는 상기 피크 파워 소모량 정보에 기초하여, 상기 적어도 한 개의 데이터 후보 중 상기 메모리 모듈에 저장될 데이터를 선택하는, 메모리 장치.

청구항 9

제8 항에 있어서,

상기 후보 선택부는 상기 적어도 한 개의 데이터 후보 중 피크 파워 소모량이 적은 데이터 후보를 상기 메모리 모듈에 저장될 데이터로 선택하는, 메모리 장치.

청구항 10

제1 항에 있어서,

상기 메모리 모듈은 상 변화 메모리 셀들을 포함하는, 메모리 장치.

청구항 11

해시를 이용하여, 쓰기 요청된 데이터에 대응하는 복수의 데이터 후보를 생성하는 단계;

쓰기 요청된 데이터가 저장될 메모리 셀들에 저장된 데이터와 상기 복수의 데이터 후보의 비교 결과에 기초하여, 상기 복수의 데이터 후보 각각에 대한 비트 플립의 개수 및 피크 파워 소모량을 계산하는 단계;

상기 비트 플립의 개수에 기초하여, 상기 복수의 데이터 후보 중 적어도 한 개의 데이터 후보를 선택하는 단계; 및

상기 피크 파워 소모량에 기초하여, 상기 적어도 한 개의 데이터 후보 중 쓰기 동작이 수행될 데이터를 선택하는, 메모리 장치의 인코딩 방법.

청구항 12

제11 항에 있어서,

상기 쓰기 요청된 데이터의 비트 위치와 상기 해시에 대한 논리 연산 결과에 기초하여, 상기 복수의 데이터 후보를 생성하는, 메모리 장치의 인코딩 방법.

청구항 13

제11 항에 있어서,

상기 쓰기 요청된 데이터는 복수의 칩들 각각에 분산되어 저장되며,

상기 피크 파워 소모량은 상기 복수의 칩들 각각에서 소모되는 파워 소모량 중 가장 큰 파워 소모량인, 메모리

장치의 인코딩 방법.

청구항 14

제11 항에 있어서,

상기 쓰기 요청된 데이터는 복수의 칩들 각각에 분산되어 저장되며,

상기 피크 파워 소모량은 상기 복수의 칩들 각각에서 소모되는 파워 소모량의 합인, 메모리 장치의 인코딩 방법.

청구항 15

제11 항에 있어서,

상기 복수의 데이터 후보 중 적어도 한 개의 데이터 후보를 선택하는 단계는

상기 복수의 데이터 후보들 중 최소 비트 플립의 개수를 갖는 데이터 후보를 선택하는 단계; 및

상기 최소 비트 플립의 개수를 갖는 데이터 후보보다 소정 개수 많은 비트 플립을 갖는 데이터 후보를 선택하는 단계를 포함하는, 메모리 장치의 인코딩 방법.

청구항 16

제11 항에 있어서,

상기 적어도 한 개의 데이터 후보 중 쓰기 동작이 수행될 데이터를 선택하는 단계는, 상기 적어도 한 개의 데이터 후보 중 피크 파워 소모량이 적은 데이터 후보를 상기 쓰기 동작이 수행될 데이터로 선택하는, 메모리 장치의 인코딩 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 메모리 장치에 관한 것으로, 좀 더 자세하게는 가변 저항 메모리 셀들을 포함하는 메모리 장치에 관한 것이다.

배 경 기 술

[0002] 반도체 메모리 장치(semiconductor memory device)는 데이터를 저장해 두고 필요할 때 꺼내어 읽어볼 수 있는 기억장치이다. 반도체 메모리 장치는 크게 휘발성 메모리 장치(Volatile memory device)와 불휘발성 메모리 장치(Nonvolatile mmory device)로 구분된다.

[0003] 휘발성 메모리 장치는 전원 공급이 차단되면 저장하고 있던 데이터가 소멸되는 메모리 장치이다. 휘발성 메모리 장치에는 SRAM, DRAM, SDRAM 등이 있다. 불휘발성 메모리 장치는 전원 공급이 차단되어도 저장하고 있던 데이터가 소멸하지 않는 메모리 장치이다. 불휘발성 메모리 장치에는 ROM, PROM, EPROM, EEPROM, 플래시 메모리 장치, PRAM, MRAM, RRAM, FRAM 등이 있다.

발명의 내용

해결하려는 과제

[0004] 본 출원의 목적은 쓰기 성능이 향상되면서, 동시에 메모리 수명을 연장할 수 있는 메모리 장치 및 그것의 동작 방법을 제공하는데 있다.

과제의 해결 수단

[0005] 본 출원의 일 실시 예에 따른 메모리 장치는 적어도 하나의 차지 펌프를 포함하는 메모리 모듈; 및 상기 메모리 모듈을 제어하는 컨트롤러를 포함하며, 상기 컨트롤러는 해시를 이용하여, 쓰기 요청된 데이터에 대응하는 복수의 데이터 후보를 생성하는 셔플링 로직; 상기 메모리 모듈에 저장된 데이터와 상기 복수의 데이터 후보의 비교

결과에 기초하여, 상기 복수의 데이터 후보 각각에 대한 비트 플립의 개수 및 피크 파워 소모를 연산하는 계산부; 및 상기 비트 플립의 개수와 상기 피크 파워 소모에 기초하여, 상기 복수의 데이터 후보 중 상기 메모리 모듈에 저장될 데이터를 선택하는 후보 선택부를 포함한다.

[0006] 실시 예에 있어서, 상기 셔플링 로직은 상기 쓰기 요청된 데이터의 비트 위치와 상기 해시에 대한 논리 연산 결과에 기초하여, 상기 복수의 데이터 후보를 생성한다.

[0007] 실시 예에 있어서, 상기 계산부는 상기 메모리 모듈에 저장된 데이터와 상기 복수의 데이터 후보를 비교 결과에 기초하여, 비트 플립의 개수를 계산하는 비트 플립 계산부; 및 상기 비트 플립 정보에 기초하여, 상기 복수의 데이터 후보 각각에 대한 피크 파워 소모량을 계산하는 피크 파워 계산부를 포함한다.

[0008] 실시 예에 있어서, 상기 메모리 모듈은 복수의 칩들; 및 상기 복수의 칩들에 대응하는 복수의 차지 펌프들을 포함하며, 상기 피크 파워 계산부는 상기 복수의 칩들 각각에 대한 파워 소모량을 계산하고, 상기 피크 파워 소모량은 상기 복수의 칩들 각각에 대한 파워 소모량 중 가장 큰 파워 소모량이다.

[0009] 실시 예에 있어서, 상기 메모리 모듈은 복수의 칩들; 및 상기 복수의 칩들에 전원을 공급하는 차지 펌프를 포함하며, 상기 피크 파워 계산부는 상기 복수의 칩들 전체에서 소모되는 파워 소모량을 합산하여 상기 피크 파워 소모량을 계산한다.

[0010] 실시 예에 있어서, 상기 후보 선택부는 상기 비트 플립 정보에 기초하여, 상기 복수의 데이터 후보들 중 적어도 한 개의 데이터 후보를 선택한다.

[0011] 실시 예에 있어서, 상기 후보 선택부는 상기 복수의 데이터 후보들 중 최소 비트 플립의 개수를 갖는 데이터 후보 및 상기 최소 비트 플립의 개수를 갖는 데이터 후보보다 소정 개수 많은 비트 플립을 갖는 데이터 후보를 선택한다.

[0012] 실시 예에 있어서, 상기 후보 선택부는 상기 피크 파워 소모량 정보에 기초하여, 상기 적어도 한 개의 데이터 후보 중 상기 메모리 모듈에 저장될 데이터를 선택한다.

[0013] 실시 예에 있어서, 상기 후보 선택부는 상기 적어도 한 개의 데이터 후보 중 피크 파워 소모량이 적은 데이터 후보를 상기 메모리 모듈에 저장될 데이터로 선택한다.

[0014] 실시 예에 있어서, 상기 메모리 모듈은 상 변화 메모리 셀들을 포함한다.

[0015] 본 출원의 일 실시 예에 따른 메모리 장치의 인코딩 방법은 해시를 이용하여, 쓰기 요청된 데이터에 대응하는 복수의 데이터 후보를 생성하는 단계; 쓰기 요청된 데이터가 저장될 메모리 셀들에 저장된 데이터와 상기 복수의 데이터 후보의 비교 결과에 기초하여, 상기 복수의 데이터 후보 각각에 대한 비트 플립의 개수 및 피크 파워 소모량을 계산하는 단계; 상기 비트 플립의 개수에 기초하여, 상기 복수의 데이터 후보 중 적어도 한 개의 데이터 후보를 선택하는 단계; 및 상기 피크 파워 소모량에 기초하여, 상기 적어도 한 개의 데이터 후보 중 쓰기 동작이 수행될 데이터를 선택한다.

발명의 효과

[0016] 본 출원의 실시 예에 따른 메모리 장치는 향상된 쓰기 성능 및 메모리 수명을 가질 수 있다.

도면의 간단한 설명

[0017] 도 1은 본 출원의 일 실시 예에 따른 메모리 장치를 보여주는 블록도이다.

도 2는 도 1의 메모리 장치의 메모리 셀의 리셋 및 셋 동작 시의 전류 레벨을 보여주는 도면이다.

도 3 및 도 4는 도 1의 셔플링 로직의 구조 및 동작을 설명하기 위한 도면들이다.

도 5는 도 1의 메모리 장치의 동작을 좀 더 자세히 설명하기 위한 도면이다.

도 6은 도 1의 메모리 장치의 중간 인코딩 후보 결정 방법의 일 예를 설명하기 위한 도면이다.

도 7은 도 1 내지 도 6에서 설명된 메모리 장치의 동작을 좀 더 자세히 설명하는 순서도이다.

도 8은 본 출원의 다른 실시 예에 따른 메모리 장치를 보여주는 블록도이다.

도 9는 도 8의 메모리 장치의 동작을 설명하기 위한 도면이다.

도 10은 도 8 및 도 9에서 설명된 메모리 장치의 동작을 좀 더 자세히 설명하는 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0018] 이하에서는, 본 출원의 기술 분야에서 통상의 지식을 가진 자가 본 출원의 기술적 사상을 용이하게 이해할 수 있을 정도로 상세히 설명하기 위하여, 본 출원의 기술적 사상이 첨부된 도면을 참조하여 설명될 것이다.

[0019] 도 1은 본 출원의 일 실시 예에 따른 메모리 장치(10)를 보여주는 블록도이다. 도 1을 참조하면, 메모리 장치(10)는 메모리 모듈(100) 및 메모리 컨트롤러(200)를 포함한다.

[0020] 메모리 모듈(100)은 메모리 컨트롤러(200)의 쓰기 요청에 응답하여, 쓰기 요청된 데이터를 저장한다. 메모리 모듈(100)은 메모리 컨트롤러(200)의 읽기 요청에 응답하여, 읽기 요청된 데이터를 독출한다. 메모리 모듈(100)은 복수의 칩들(111~11N)을 포함한다.

[0021] 복수의 칩들(111~11N)은 각각 적어도 하나의 뱅크들(BANK 1~BANK M)을 포함한다. 각각의 뱅크는 데이터를 저장하기 위한 메모리 셀들을 포함한다. 예를 들어, 각각의 뱅크는 복수의 상 변화 메모리 셀들(Phase Change Memory cells)을 포함할 수 있다. 다만, 이는 예시적인 것이며, 저항이 가변 될 수 있다면, 각각의 뱅크에 포함된 메모리 셀들의 종류는 특별히 제한되지 않는다.

[0022] 복수의 칩들(111~11N)은 각각 필요한 전력을 제공하는 차지 펌프들(121~12N)을 포함할 수 있다. 예를 들어, 제1 칩(111)은 제1 차지 펌프(121)를 포함하며, 제1 차지 펌프(121)는 읽기 및/또는 쓰기 동작 시에 필요한 전력을 제1 칩(111)에 제공할 수 있다. 마찬가지로, 제2 내지 제N 차지 펌프(122~12N)는 각각 제2 내지 제N 칩(112~11N)의 구동에 필요한 전력을 각각 제공할 수 있다.

[0023] 메모리 컨트롤러(200)는 메모리 장치(10)의 전반적인 동작을 제어한다. 예를 들어, 읽기 동작이 수행되는 경우, 메모리 컨트롤러(200)는 읽기 요청된 데이터가 메모리 모듈(100)로부터 독출되도록 메모리 장치(10)를 제어할 것이다. 다른 예로, 쓰기 동작이 수행되는 경우, 메모리 컨트롤러(200)는 쓰기 요청된 데이터가 메모리 모듈(100)에 저장되도록 메모리 장치(10)를 제어할 것이다.

[0024] 본 출원의 기술적 사상에 따른 실시 예에 있어서, 메모리 컨트롤러(200)는 쓰기 요청된 데이터를 셔플링하여 복수의 인코딩 후보들(encoding candidates)을 생성하고, 비트 플립의 개수와 피크 파워 소모량에 기초하여 해당 복수의 인코딩 후보들 중 최적의 인코딩 후보를 선택하여 쓰기 동작을 수행한다. 이와 같이, 비트 플립의 개수와 피크 파워 소모량에 기초하여 최적의 인코딩 후보를 선택하여 쓰기 동작을 수행함으로써, 쓰기 성능을 향상시킬 수 있을 뿐만 아니라, 메모리의 수명을 연장시킬 수 있다.

[0025] 좀 더 자세히 설명하면, 도 1과 같이 칩(chip)마다 파워 예산(power budget)의 제한이 있는 메모리 장치(10)의 경우, 각 칩(chip)마다 차지 펌프가 구비되어 쓰기 동작 시에 필요한 전압과 파워를 공급한다. 그러나 가변 변화 메모리의 경우에 쓰기 동작을 수행하기 위하여 필요한 파워량이 크며, 이에 따라 파워 예산 내에서 동시에 수행할 수 있는 쓰기 요청의 수가 제한이 되게 된다.

[0026] 한편, 메모리 셀들로 상 변화 메모리가 사용되는 경우, 메모리 셀에 저장되는 데이터에 따라 필요한 전력 소모량이 달라질 수 있다. 예를 들어, 도 2에 도시된 바와 같이, 상 변화 메모리의 데이터를 "1"에서 "0"으로 리셋(reset)하는데 소모되는 전력량이 데이터를 "0"에서 "1"로 셋(set)하는데 소모되는 전력량보다 클 수 있다. 이와 같이 데이터 "0"에 대한 쓰기 동작을 수행할 때 필요한 전력 소모량과 데이터 "1"에 대한 쓰기 동작을 수행할 때 필요한 전략 소모량이 다른 것은 "쓰기 비대칭(write asymmetry)"라고 칭해질 수 있다.

[0027] 본 출원의 기술적 사상의 실시 예에 따른 메모리 장치(10)는 이와 같은 쓰기 비대칭을 고려하여, 쓰기 요청 당 피크 파워 소모량을 줄임으로써 파워 예산 내에서 동시에 수행할 수 있는 쓰기 요청의 개수를 증가시킬 수 있다. 더욱이, 본 출원의 기술적 사상에 따른 메모리 장치(10)는 쓰기 요청된 데이터와 기존에 저장된 데이터의 비트 플립의 개수를 줄임으로써, 메모리의 수명을 향상시킬 수 있다.

[0028] 이를 위하여, 본 출원의 기술적 사상의 실시 예에 따른 메모리 컨트롤러(200)는 셔플링 로직(210), 계산부(220) 및 후보 선택부(230)를 포함한다.

[0029] 셔플링 로직(210)은 쓰기 요청된 데이터를 해시(hash)로 셔플링(shuffling)하여 인코딩 후보들(encoding candidates)를 생성한다. 예를 들어, 셔플링 로직(210)은 쓰기 요청된 데이터의 각 비트 위치를 해시 후보와 XOR 연산을 수행하고, 그 결과를 셔플링된 데이터의 새로운 비트 위치로 결정할 수 있다. 이 경우, 새로운 비트 위치에 의하여 변경된 데이터는 "데이터 후보"로 칭해지며, 해당 데이터 후보에 대응하는 해시는 "해시 후보"라

칭해질 수 있다. 또한, 데이터 후보와 해시 후보는 통칭하여 “인코딩 후보”라 칭해질 수 있다. 셔플링 로직(210)의 구조 및 동작은 이하의 도 3 및 도 4에서 좀 더 자세히 설명될 것이다.

[0030] 계산부(220)는 셔플링 로직(210)으로부터 쓰기 요청된 데이터에 대응하는 인코딩 후보들을 수신한다. 또한, 계산부(220)는 쓰기 요청된 데이터가 저장될 메모리 셀들에 저장되어 있는 기존 데이터(이하, "기 입력 데이터") 및 해당 기 입력 데이터에 대응하는 해시(이하, “기 입력 해시”)에 대한 정보를 수신한다.

[0031] 계산부(220)는 데이터 후보들 및 해시 후보들을 기 입력 데이터 및 기 입력 해시의 정보와 비교함으로써, 비트 플립의 개수(the numberof bit flips) 및 피크 파워 소모량(peak power consumption)을 계산한다. 이를 위하여, 계산부(220)는 비트 플립 계산부(221) 및 피크 파워 계산부(222)를 포함한다.

[0032] 비트 플립 계산부(221)는 데이터 후보들 및 해시 후보들을 기 입력 데이터 및 기 입력 해시의 정보와 비교함으로써, 비트 플립의 개수를 계산한다. 예를 들어, 기 입력 데이터가 "1101"이고, 이에 대응하는 기 입력 해시가 "00"이라고 가정하자. 그리고 인코딩 후보의 데이터 후보가 "0001"이고, 이에 대응하는 해시(이하, "해시 후보")가 "00"이라고 가정하자. 이 경우, 기 입력 데이터가 "1101"에서 "0001"로 변경되었고, 해시는 "00"으로 변경이 없으므로, 비트 플립 계산부(221)는 비트 플립의 개수를 "2"로 계산할 것이다.

[0033] 피크 파워 계산부(222)는 비트 플립 정보에 기초하여, 복수의 인코딩 후보들 각각에 필요한 피크 파워 소모량을 계산한다. 예를 들어, 피크 파워 계산부(222)는 쓰기 비대칭, 즉 데이터 "0"에 대한 쓰기 동작이 수행될 때와 데이터 "1"에 대한 쓰기 동작이 수행될 때에 필요한 소모량이 다르다는 점을 고려하여, 복수의 인코딩 후보들 각각에 필요한 피크 파워 소모량을 계산할 것이다.

[0034] 후보 선택부(230)는 비트 플립 계산부(221)로부터 각 인코딩 후보들에 대한 비트 플립 정보를 수신하고, 피크 파워 계산부(222)로부터 각 인코딩 후보들에 대한 피크 파워 정보를 수신한다. 후보 선택부(230)는 비트 플립 정보와 피크 파워 정보에 기초하여, 복수의 인코딩 후부들 중 최종적으로 쓰기 동작이 수행될 인코딩 후보를 선택한다.

[0035] 예를 들어, 후보 선택부(230)는 비트 플립 정보에 기초하여 복수의 인코딩 후보들 중 적어도 한 개의 인코딩 후보를 먼저 선택할 수 있다. 예를 들어, 후보 선택부(230)는 최소 비트 플립의 개수를 갖는 인코딩 후보와, 상기 최소 피트 필립의 개수보다 조금 많은 비트 플립의 개수를 갖는 인코딩 후보를 중간 인코딩 후보들로 선택할 수 있다. 이후, 후보 선택부(230)는 상기 중간 인코딩 후보들 중 가장 적은 파워 소모를 갖는 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택할 수 있다.

[0036] 이 경우, 최종적으로 선택된 인코딩 후보는 비트 플립의 개수가 적을 뿐만 아니라 피크 파워 소모량 역시 크지 않다. 따라서, 메모리의 수명이 연장되며, 파워 예산 내에서 동시에 가능한 많은 쓰기 동작이 수행될 수 있다.

[0037] 도 3 및 도 4는 도 1의 셔플링 로직(210)의 구조 및 동작을 설명하기 위한 도면들이다.

[0038] 도 3 및 도 4를 참조하면, 셔플링 로직(210)은 쓰기 요청된 데이터를 해시 후보와 XOR 연산을 수행함으로써, 데이터 후보를 생성할 수 있다.

[0039] 좀 더 자세히 설명하면, 쓰기 요청된 데이터가 "0001"이라고 가정된다. 또한, 쓰기 요청된 데이터 "0001"의 비트 위치들(b0~b3)은 각각 "00", "01", "10", "11"이라고 가정된다. 또한, 해시 후보는 "01"이라고 가정된다.

[0040] 이 경우, 먼저 쓰기 요청된 데이터의 각각의 비트 위치와 해시 후보에 대한 XOR 연산을 수행함으로써 새로운 비트 위치(new bit position)가 결정된다. 예를 들어, 쓰기 요청된 데이터의 원래의 비트 위치(original bit position) 중에서 첫 번째 비트 위치(b0)는 두 번째 비트 위치(b1)로, 두 번째 비트 위치(b1)는 첫 번째 비트 위치(b0)로, 세 번째 비트 위치(b2)는 네 번째 비트 위치(b3)로, 네 번째 비트 위치(b3)는 세 번째 비트 위치(b2)로 각각 변경된다. 또한, 변경된 비트 위치에 따라 비트 데이터 역시 변경되며, "0010"의 데이터 후보가 생성된다.

[0041] 도 5는 도 1의 메모리 장치(10)의 동작을 좀 더 자세히 설명하기 위한 도면이다.

[0042] 설명의 편의상, 쓰기 요청된 데이터는 "0001"이라고 가정된다. 또한, 쓰기 요청된 데이터가 저장될 메모리 셀의 기 입력 데이터는 "1101"이며, 기 입력 데이터에 대응하는 기 입력 해시는 "00"이라고 가정된다. 또한, 하나의 칩에는 두 비트의 데이터가 저장된다고 가정된다. 그리고 데이터 "0"에 대한 쓰기 동작이 수행될 때에 480 마이크로 와트의 전력이 소모되고, 데이터 "1"에 대한 쓰기 동작이 수행될 때에 90 마이크로 와트의 전력이 소모된다고 가정될 것이다.

[0043] 도 5를 참조하면, 셔플링 로직(210)은 쓰기 요청된 데이터 및 해시 후보들로 셔플링하여 인코딩 후보들을 생성한다. 이 경우, 해시 후보들은 쓰기 요청된 데이터의 비트 너비(bit width)를 고려하여 결정될 수 있다. 예를 들어, 쓰기 요청된 데이터가 L 비트인 경우, 해시 후보는 "0~(L-1)"로 결정될 수 있다. 즉, 도 5와 같이, 쓰기 요청된 데이터가 "4" 비트인 경우, 해시 후보 후보들은 "0~3"일 수 있으며, 이에 따라 해시 후보들은 각각 "00", "01", "10", "11"로 생성될 수 있다.

[0044] 예를 들어, 해시 후보가 "00"인 경우, 셔플링 로직(210)은 쓰기 요청된 "0001"의 데이터와 "00"의 해시 후보를 논리 조합하여, 데이터 후보가 "0001"인 제1 인코딩 후보(Encoding Candidate 1)를 생성할 것이다. 이 경우, 각 칩이 두 비트의 데이터를 저장하기 때문에, "0001"의 데이터 중 "00"은 제1 칩(chip 1)에 대응되고, "01"은 제2 칩(chip)에 대응되며, 해시 후보 “00” 은 제3 칩(chip 3)에 대응될 것이다.

[0045] 이와 유사하게, 해시 후보가 "01"인 경우, 셔플링 로직(210)은 데이터 후보가 "0010"인 제2 인코딩 후보(Encoding Candidate 2)를 생성할 것이다. 또한, 해시 후보가 "10"인 경우에 셔플링 로직(210)은 데이터 후보가 "0100"의 제3 인코딩 후보(Encoding Candidate 3)를 생성하고, 해시 후보가 "11"인 경우에 셔플링 로직(210)은 데이터 후보가 "1000"인 제4 인코딩 후보(Encoding Candidate 4)를 생성할 것이다.

[0046] 이후, 계산부(220)는 복수의 인코딩 후보들 각각에 대한 비트 플립의 개수 및 피크 파워 소모량을 계산할 것이다.

[0047] 좀 더 구체적으로, 비트 플립 계산부(221)는 인코딩 후보의 데이터 후보를 기 입력 데이터와 비교하고, 인코딩 후보의 해시 후보를 기 입력 해시와 비교함으로써, 비트 플립의 개수를 계산할 것이다.

[0048] 예를 들어, 제1 인코딩 후보의 경우, 제1 데이터 후보 및 제1 해시 후보는 각각 "0001" 및 "00"이고, 기 입력 데이터 및 기 입력 해시는 각각 "1101" 및 "00"이다. 따라서, 이 경우, 비트 플립 계산부(221)는 비트 플립의 개수를 "2"로 계산할 것이다. 마찬가지로, 제2 인코딩 후보의 경우에 비트 플립의 개수는 "5"로 계산되고, 제3 인코딩 후보의 경우에 비트 플립의 개수는 "3"으로 계산되며, 제4 인코딩 후보의 경우에 비트 플립의 개수는 "4"로 계산될 것이다.

[0049] 또한, 피크 파워 계산부(222)는 비트 플립 정보에 기초하여, 칩 별 피크 파워 소모량(per-chip peak power consumption)을 계산할 것이다.

[0050] 예를 들어, 제1 인코딩 후보의 경우, 제1 칩(chip 1)에 대응하는 두 개 비트가 플립되며, 두 개 비트 모두 "0"에 해당한다. 이 경우, 데이터 "0"에 대한 쓰기 동작 시에 소요되는 파워가 480 마이크로 와트이기 때문에, 제1 인코딩 후보의 칩 별 피크 파워 소모량은 960 마이크로 와트로 계산된다.

[0051] 또한, 제2 인코딩 후보의 경우, 제1 칩(chip)에 대응하는 두 개 비트가 모두 "0"으로 플립되고, 제2 칩(chip)에 대응하는 두 개 비트가 각각 "1" 및 "0"으로 플립되며, 제3 칩(chip)에 대응하는 한 개 비트가 "1"로 플립된다. 이 경우, 데이터 "0"에 대한 쓰기 동작 시에 소요되는 파워가 480 마이크로 와트이고, 데이터 "1"에 대한 쓰기 동작 시에 소요되는 파워가 90 마이크로 와트이기 때문에, 제1 칩(chip 1)에는 960 마이크로 와트가 필요하고, 제2 칩(chip 2)에는 560 마이크로 와트가 필요하며, 제3 칩(chip 3)에는 90 마이크로 와트가 필요하다. 이 경우, 피크 파워 계산부(222)는 제2 인코딩 후보의 칩 별 피크 파워 소모량을 960 마이크로로 계산한다.

[0052] 이와 유사하게, 제3 인코딩 후보 및 제4 인코딩 후보의 칩 별 피크 파워 소모량은 각각 480 마이크로 와트 및 480 마이크로 와트로 계산될 것이다.

[0053] 후보 선택부(230)는 비트 플립의 개수와 피크 파워 소모량에 기초하여, 최종적으로 쓰기 동작이 수행될 인코딩 후보를 선택한다.

[0054] 좀 더 자세히 설명하면, 먼저 후보 선택부(230)는 최소 비트 플립의 개수를 갖는 인코딩 후보와, 이보다 좀 더 많은 비트 플립의 개수를 갖는 인코딩 후보를 중간 인코딩 후보들로 선택한다.

[0055] 예를 들어, 최소 비트 플립의 개수를 갖는 인코딩 후보와, 이보다 하나 많은 비트 플립의 개수를 갖는 인코딩 후보가 중간 인코딩 후보들로 선택된다면, 후보 선택부(230)는 "2"의 비트 플립 개수를 갖는 제1 인코딩 후보와, "3"의 비트 플립 개수를 갖는 제3 인코딩 후보를 중간 인코딩 후보들로 선택한다.

[0056] 이후, 후보 선택부(230)는 중간 인코딩 후보들 중 피크 파워 소모량이 가장 적은 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택한다.

[0057] 예를 들어, 제1 인코딩 후보의 피크 파워 소모량이 960 마이크로 와트이고, 제3 인코딩 후보의 피크 파워 소모

량이 480 마이크로 와트이기 때문에, 후보 선택부(230)는 제3 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택할 것이다.

[0058] 상술한 바와 같이, 본 출원의 실시 예에 따른 메모리 장치(10)는 쓰기 요청된 데이터에 대응하는 복수의 인코딩 후보들을 생성하고, 칩 별 피크 파워 소모량 및 비트 플립의 개수를 고려하여 복수의 인코딩 후보들 중 최종적으로 쓰기 동작이 수행될 인코딩 후보를 선택한다. 이에 따라, 본 출원의 실시 예에 따른 메모리 장치(10)는 메모리 수명을 향상시킬 수 있을 뿐만 아니라, 쓰기 성능을 향상시킬 수 있다.

[0059] 한편, 상술한 설명은 예시적인 것이며, 본 출원의 기술적 사상은 이에 한정되지 않음이 이해될 것이다. 예를 들어, 도 5에서, 중간 인코딩 후보를 결정할 때에, 최소 비트 플립의 개수보다 하나 또는 N개 많은 인코딩 후보가 중간 인코딩 후보로 결정된다고 설명하였다. 다만, 이는 예시적인 것이며, 중간 인코딩 후보를 찾는 방법은 설계자에 의하여 변경될 수 있음이 이해될 것이다. 예를 들어, 중간 인코딩 후보를 결정할 때에, 최소 비트 플립의 개수보다 2 번째 또는 N번째 적은 비트 플립의 개수를 갖는 인코딩 후보가 중간 인코딩 후보로 결정될 수도 있다.

[0060] 또한, 도 1 내지 도 5에서는, 최종적으로 쓰여질 인코딩 후보를 선택할 때에, 비트 플립의 개수를 고려하여 인코딩 후보들 중 적어도 두 개의 인코딩 후보들(즉, 중간 인코딩 후보들)을 선택하는 것으로 설명하였다. 다만, 이는 예시적인 것이며, 중간 인코딩 후보를 설정하는 방법에 따라, 중간 인코딩 후보로 하나의 인코딩 후보만이 선택될 수도 있다. 이는 이하의 도 6에서 좀 더 자세히 설명될 것이다.

[0061] 도 6은 도 1의 메모리 장치(10)의 중간 인코딩 후보 결정 방법의 일 예를 설명하기 위한 도면이다.

[0062] 설명의 편의상, 쓰기 요청된 데이터는 "00111001"이라고 가정되고, 기 입력 데이터 및 기 입력 해시는 각각 "00000110" 및 "001"이라고 가정된다.

[0063] 도 6을 참조하면, 쓰기 요청된 데이터가 8비트이기 때문에, 8개의 해시 후보가 생성된다. 이후, 쓰기 요청된 데이터의 비트 위치와 해시 후보 사이의 논리 연산에 의하여 8개의 인코딩 후보들(Candidate 0~Candidate 7)가 형성된다. 이때, 최소 비트 플립의 개수는 "2"로 계산된다.

[0064] 이 경우, 최소 비트 플립의 개수보다 좀 더 많은 비트 플립의 개수를 갖는 인코딩 후보를 찾아 중간 인코딩 후보로 결정하는 방법은 설계자에 의하여 다양하게 설정될 수 있다.

[0065] 예를 들어, 최소 비트 플립의 개수보다 하나, 또는 X개 많은 비트 플립을 갖는 인코딩 후보가 중간 인코딩 후보로 결정될 수 있다(이하, "+X 기법").

[0066] “+X 기법” 의 경우에, 만약 X가 "1"이라면, 최소 비트 플립인 "2"보다 하나 많은 비트 플립이 없으므로, 중간 인코딩 후보로는 하나의 인코딩 후보(candidate 1)만이 결정될 것이다. 다른 예로, 만약 X가 "2"라면, 중간 인코딩 후보로는 "2"의 비트 플립 개수를 갖는 인코딩 후보(candidate 1)와 "4"의 비트 플립 개수를 갖는 인코딩 후보(candidate 2)가 중간 인코딩 후보로 결정될 것이다.

[0067] 다른 예로, 최소 비트 플립의 개수보다 2번째 또는 Y 번째 적은 비트 플립을 갖는 인코딩 후보가 중간 인코딩 후보로 결정될 수 있다(이하, "+Yth 기법").

[0068] “+Yth 기법” 의 경우에, 만약 Y가 "2"라면, 최소 비트 플립 개수를 갖는 인코딩 후보(candidate 1)와 두 번째로 적은 비트 플립 개수를 갖는 인코딩 후보(candidate 2)가 중간 인코딩 후보로 결정될 수 있다.

[0069] 이와 같이, "+X 기법"의 경우에는 X 값에 따라 +X의 비트 플립 개수를 갖는 인코딩 후보가 존재하지 않을 수 있으나, "+Yth 기법"의 경우에는 Y 번째 작은 비트 플립을 갖는 인코딩 후보가 항상 존재한다.

[0070] 도 7은 도 1 내지 도 6에서 설명된 메모리 장치(10)의 동작을 좀 더 자세히 설명하는 순서도이다.

[0071] S110 단계에서, 셔플링 로직(210)은 해시를 이용하여 쓰기 요청된 데이터에 대한 셔플링 동작을 수행함으로써, 복수의 인코딩 후보들을 생성한다.

[0072] S120 단계에서, 계산부(220)는 복수의 인코딩 후보들 각각에 대한 비트 플립의 개수 및 칩 별 피크 파워 소모량을 계산한다.

[0073] S130 단계에서, 선택부(230)는 최소 비트 플립의 개수보다 조금 더 많은 비트 플립을 갖는 인코딩 후보를 찾는다. 이때, 최소 비트 플립의 개수를 갖는 인코딩 후보와, 최소 비트 플립의 개수보다 조금 많은 비트 플립을 갖는 인코딩 후보는 중간 인코딩 후보들로 결정된다.

[0074] S140 단계에서, 선택부(230)는 중간 인코딩 후보들 중 최소 피크 파워 소모량을 갖는 중간 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택한다.

[0075] 한편, 상술한 설명은 예시적인 것이며, 본 출원의 기술적 사상은 이에 한정되지 않음이 이해될 것이다. 예를 들어, 도 1 내지 도 7에서는, 각 칩에 대응하는 차지 펌프가 구비되며, 이에 딸 칩 별 파워 예산이 제한되는 것으로 설명되었다. 다만, 이는 예시적인 것이며, 복수의 칩들이 하나의 차지 펌프를 공유하도록 메모리 모듈이 구현될 수도 있다. 이는 이하의 도 8 내지 도 10을 참조하여, 좀 더 자세히 설명될 것이다.

[0076] 도 8은 본 출원의 다른 실시 예에 따른 메모리 장치(20)를 보여주는 블록도이다. 도 8의 메모리 장치(20)는 도 1의 메모리 장치(10)와 유사하다. 따라서, 동일하거나 유사한 구성은 동일하거나 유사한 참조번호를 사용하여 표기되었으며, 중복되는 설명은 간략한 설명을 위하여 이하 생략될 것이다.

[0077] 도 8을 참조하면, 메모리 장치(20)는 메모리 모듈(300) 및 메모리 컨트롤러(400)를 포함한다. 메모리 모듈(300)은 복수의 칩들(311~31N) 및 상기 복수의 칩들(311~31N)에 전원을 공급하는 차지 펌프(320)를 포함한다. 메모리 컨트롤러(400)는 셔플링 로직(410), 계산부(420) 및 후보 선택부(430)를 포함한다.

[0078] 도 1 내지 도 7의 메모리 장치(20)와 달리, 도 8의 메모리 장치(20)의 메모리 모듈(300)은 복수의 칩들(311~31N)에 전원을 공급하는 하나의 차지 펌프(320)를 구비한다. 따라서, 메모리 장치(20)의 파워 예산(power budget)은 메모리 모듈 별로 제한된다.

[0079] 이에 따라, 메모리 컨트롤러(400)의 피크 파워 계산부(422)는 인코딩 후보 각각에 대한 파워 소모량을 계산하고, 후보 선택부(430)는 복수의 인코딩 후보들 중 파워 소모량이 가장 적은 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택한다.

[0080] 도 9는 도 8의 메모리 장치(20)의 동작을 설명하기 위한 도면이다. 도 9의 메모리 장치(20)의 동작은 도 5의 메모리 장치(10)의 동작과 유사하다. 따라서, 동일하거나 유사한 구성요소는 동일하거나 유사한 참조번호를 사용하여 표기되었으며, 중복되는 설명은 간략한 설명을 위하여 생략될 것이다.

[0081] 도 9를 참조하면, 셔플링 로직(410)은 쓰기 요청된 데이터와 해시 후보들을 이용하여, 복수의 인코딩 후보들(Encoding candidate 1~Encoding candidate 4)을 생성한다.

[0082] 비트 플립 계산부(421)는 복수의 인코딩 후보들 각각을 기 입력 데이터 및 기 입력 해시와 비교함으로써, 비트 플립의 개수를 계산한다.

[0083] 피크 파워 계산부(422)는 복수의 인코딩 후보들 각각에 대한 파워 소모량을 계산한다.

[0084] 예를 들어, 제1 인코딩 후보(Encoding Candidate 1)의 경우, 두 개 비트가 플립되며, 두 개 비트 모두 "0"에 해당한다. 이 경우, 데이터 "0"에 대한 쓰기 동작 시에 소요되는 파워가 480 마이크로 와트이기 때문에, 제1 인코딩 후보의 파워 소모량은 960 마이크로 와트로 계산된다.

[0085] 또한, 제2 인코딩 후보의 경우, 다섯 개 비트가 플립되며, 이 중 세 개의 비트는 "0"으로 플립되고, 두 개의 비트는 "1"로 플립된다. 이 경우, 데이터 "0"에 대한 쓰기 동작 시에 소요되는 파워가 480 마이크로 와트이고, 데이터 "1"에 대한 쓰기 동작 시에 소요되는 파워가 90 마이크로 와트이기 때문에, 피크 파워 계산부(422)는 제2 인코딩 후보의 파워 소모량을 1620 마이크로 와트로 계산한다.

[0086] 이와 유사하게, 제3 인코딩 후보 및 제4 인코딩 후보의 파워 소모량은 각각 1050 마이크로 와트 및 1140 마이크로 와트로 계산될 것이다.

[0087] 이후, 후보 선택부(430)는 비트 플립의 개수와 파워 소모량에 기초하여, 최종적으로 쓰기 동작이 수행될 인코딩 후보를 선택한다.

[0088] 예를 들어, 후보 선택부(430)는 최소 비트 플립의 개수를 갖는 제1 인코딩 후보와, 이보다 하나 많은 비트 플립의 개수를 갖는 제3 인코딩 후보를 중간 인코딩 후보들로 선택하고, 이 중 파워 소모량이 가장 적은 제1 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택할 것이다.

[0089] 도 10은 도 8 및 도 9에서 설명된 메모리 장치(20)의 동작을 좀 더 자세히 설명하는 순서도이다.

[0090] S210 단계에서, 셔플링 로직(410)은 해시를 이용하여 쓰기 요청된 데이터에 대한 셔플링 동작을 수행함으로써, 복수의 인코딩 후보들을 생성한다.

[0091] S220 단계에서, 계산부(420)는 복수의 인코딩 후보들 각각에 대한 비트 플립의 개수 및 파워 소모량을 계산한다.

[0092] S230 단계에서, 선택부(430)는 최소 비트 플립의 개수보다 조금 더 많은 비트 플립을 갖는 인코딩 후보를 찾는다. 이때, 최소 비트 플립의 개수를 갖는 인코딩 후보와, 최소 비트 플립의 개수보다 조금 많은 비트 플립을 갖는 인코딩 후보는 중간 인코딩 후보들로 결정된다.

[0093] S240 단계에서, 선택부(430)는 중간 인코딩 후보들 중 파워 소모량이 가장 적은 중간 인코딩 후보를 최종적으로 쓰기 동작이 수행될 인코딩 후보로 선택한다.

[0094] 한편, 본 발명의 범위 또는 기술적 사상을 벗어나지 않고 본 발명의 구조가 다양하게 수정되거나 변경될 수 있음은 이 분야에 숙련된 자들에게 자명하다. 상술한 내용을 고려하여 볼 때, 만약 본 발명의 수정 및 변경이 아래의 청구항들 및 동등물의 범주 내에 속한다면, 본 발명이 이 발명의 변경 및 수정을 포함하는 것으로 여겨진다.

부호의 설명

[0095] 10, 20: 메모리 장치

100, 300: 메모리 모듈

200, 400: 메모리 컨트롤러

111, 112, 11N, 311, 312, 31N: 칩

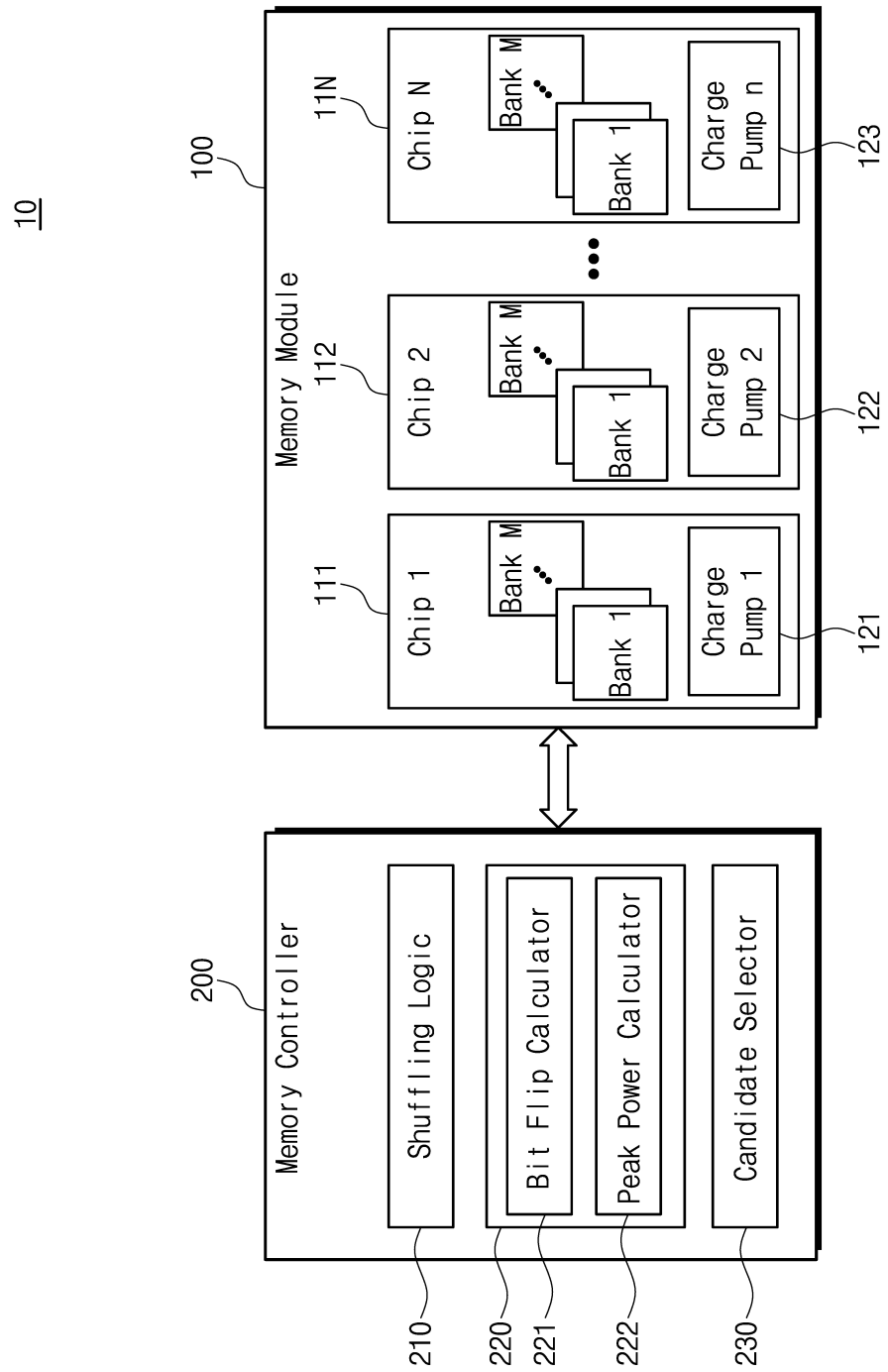
210, 410: 셔플링 로직

220, 420: 계산부

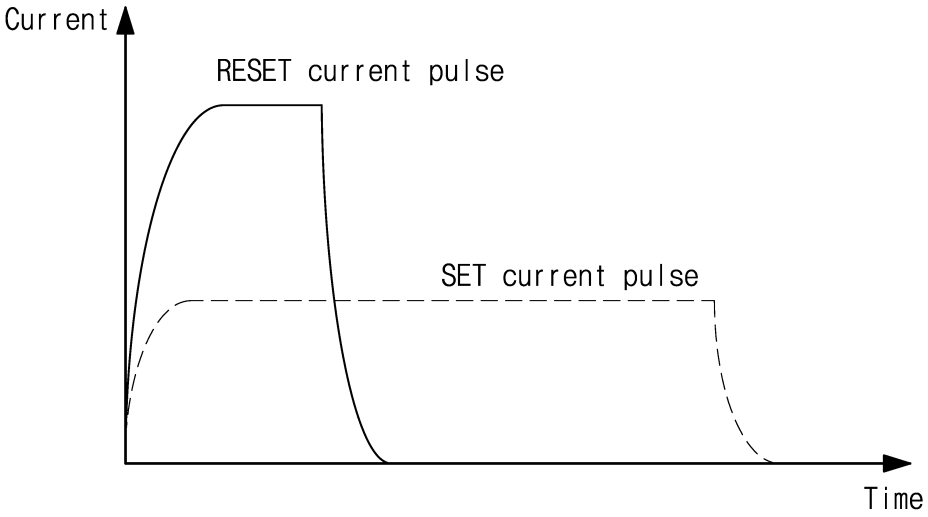
230, 430: 후보 선택부

도면

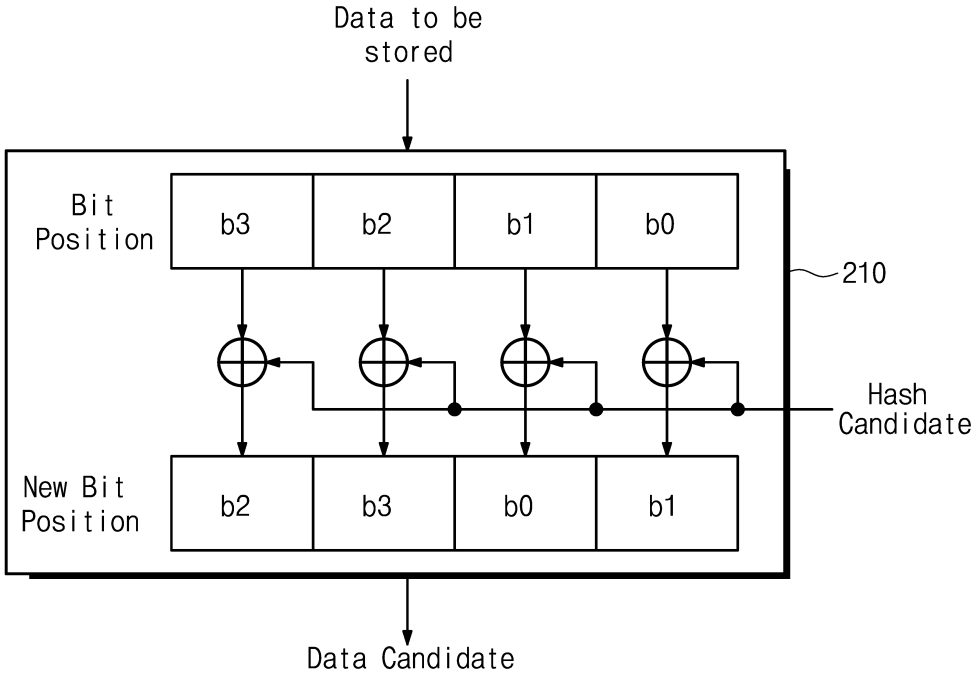
도면1


10
100
Memory Module
111
Chip 1
Bank M
Bank 1
Charge Pump 1
121
112
Chip 2
Bank M
Bank 1
Charge Pump 2
122
11N
Chip N
Bank M
Bank 1
Charge Pump n
123
200
Memory Controller
Shuffling Logic
Bit Flip Calculator
Peak Power Calculator
Candidate Selector
210
220
221
222
230

도면2


Current
RESET current pulse
SET current pulse
Time

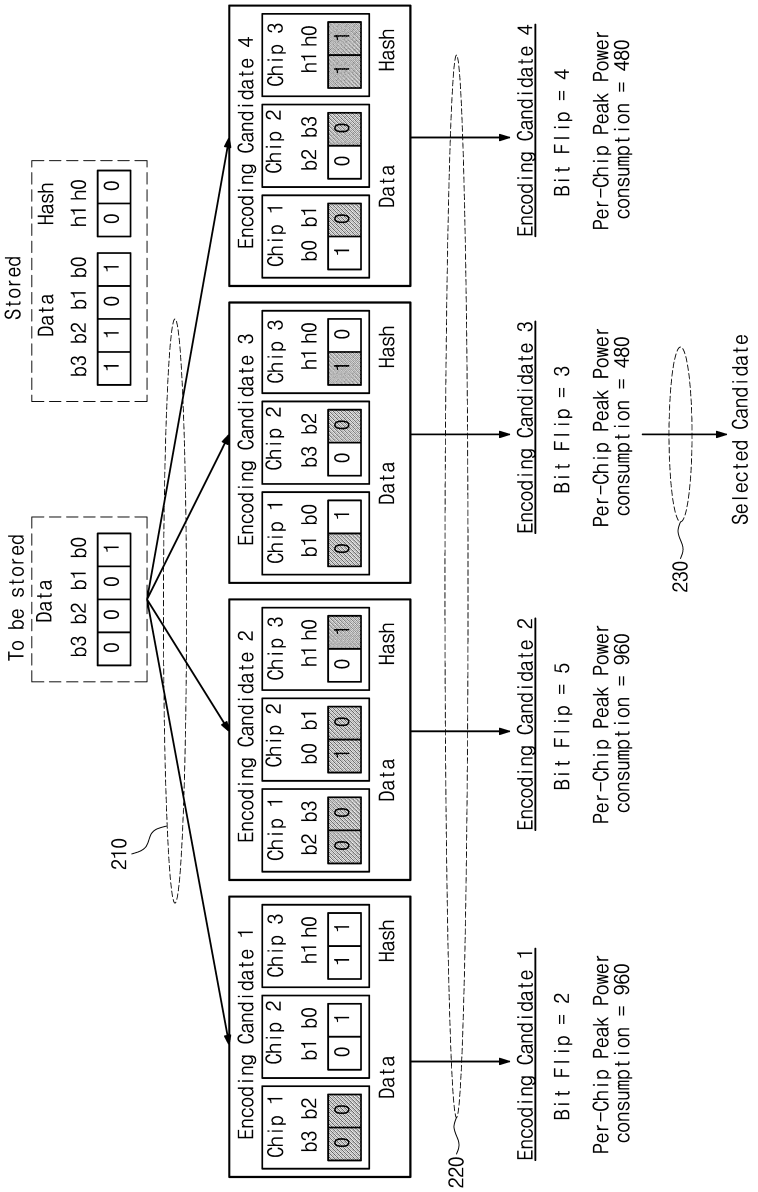
도면3


Data to be stored
Bit Position
b3
b2
b1
b0
210
Hash Candidate
New Bit Position
b2
b3
b0
b1
Data Candidate

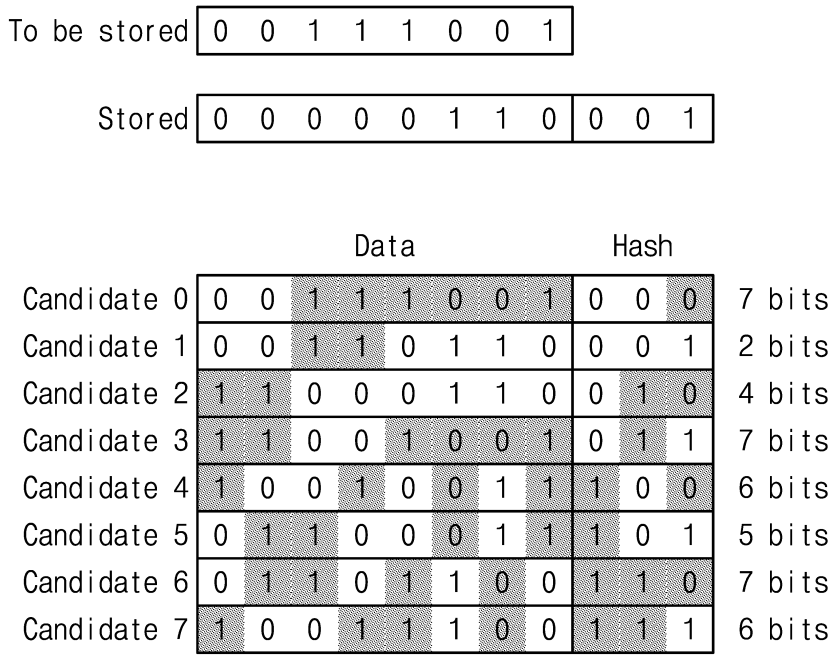
도면4

Original Bit Position	11 (b3)	10 (b2)	01 (b1)	00 (b0)
Hash Canditate	01	01	01	01
New Bit Position	10 (b2)	11 (b3)	00 (b0)	01 (b1)
Data to be stored	0	0	0	1
Data candidate	0	0	1	0

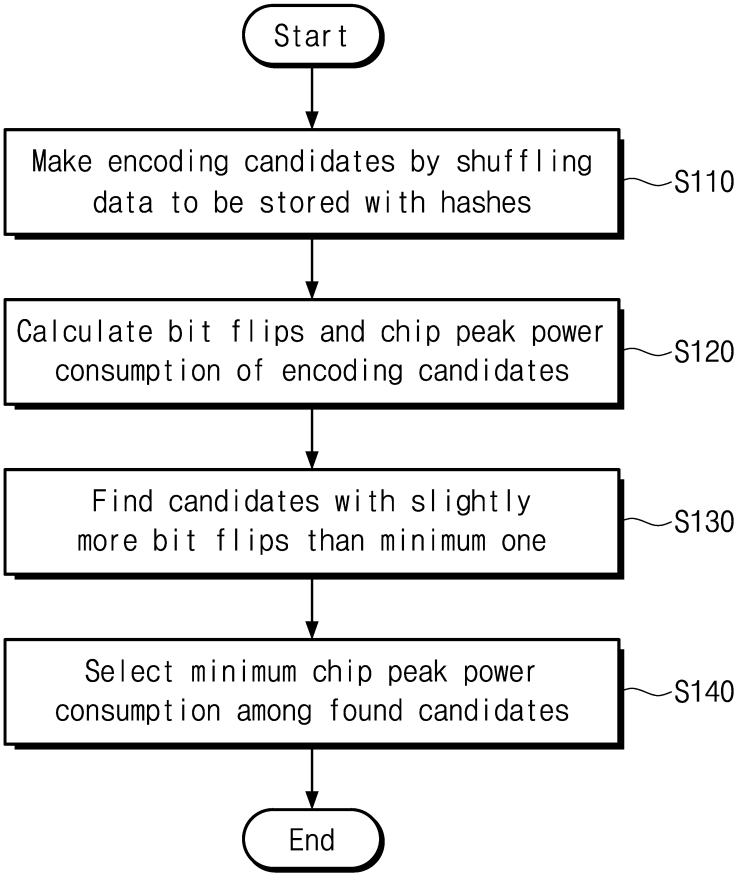
도면5



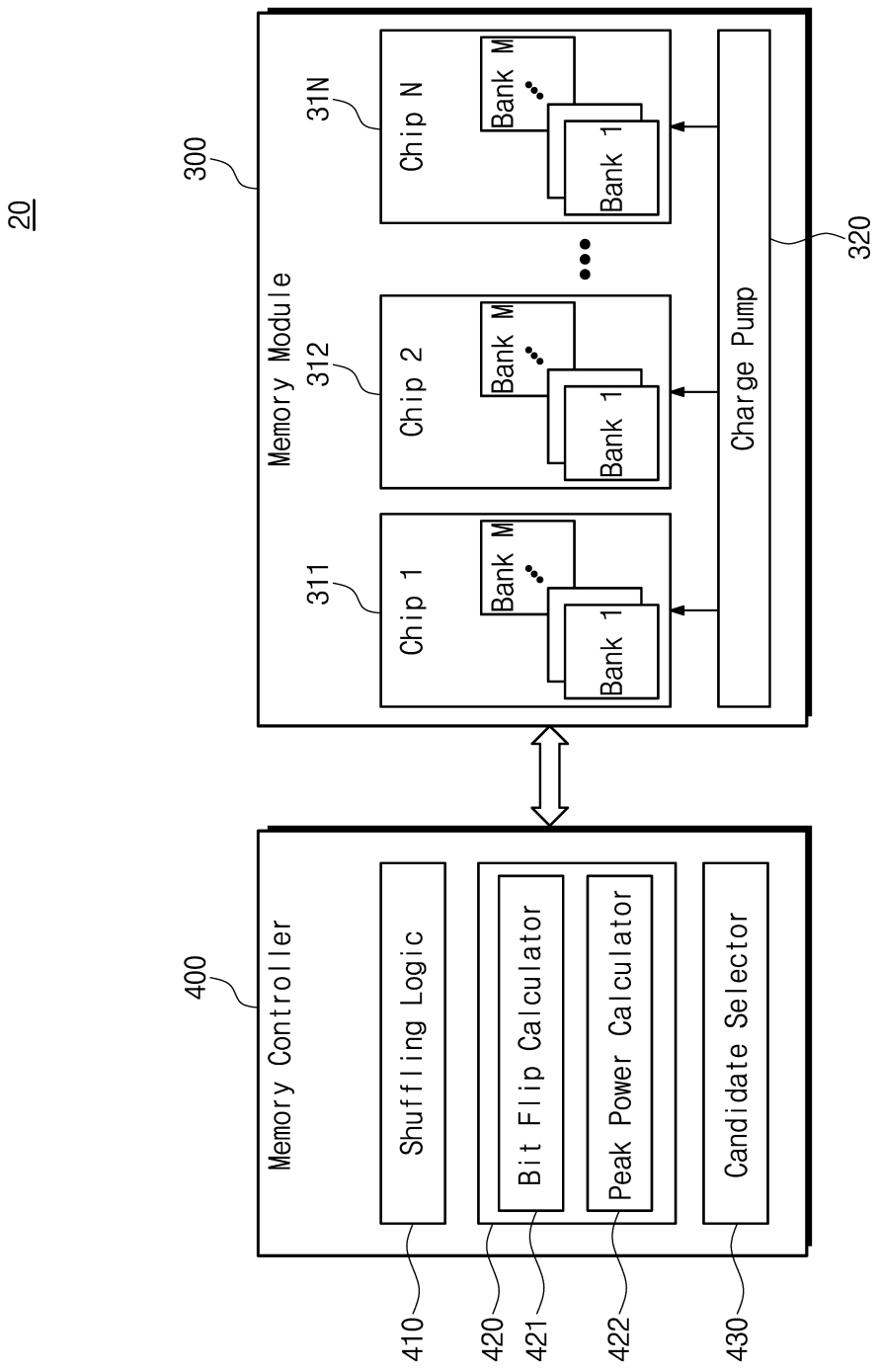
도면6


To be stored
0 0 1 1 1 0 0 1
Stored
0 0 0 0 0 1 1 0 0 0 1
Data
Hash
Candidate 0
0 0 1 1 1 0 0 1 0 0 0
7 bits
Candidate 1
0 0 1 1 0 1 1 0 0 0 1
2 bits
Candidate 2
1 1 0 0 0 1 1 0 0 1 0
4 bits
Candidate 3
1 1 0 0 1 0 0 1 0 1 1
7 bits
Candidate 4
1 0 0 1 0 0 1 1 1 0 0
6 bits
Candidate 5
0 1 1 0 0 0 1 1 1 0 1
5 bits
Candidate 6
0 1 1 0 1 1 0 0 1 1 0
7 bits
Candidate 7
1 0 0 1 1 1 0 0 1 1 1
6 bits

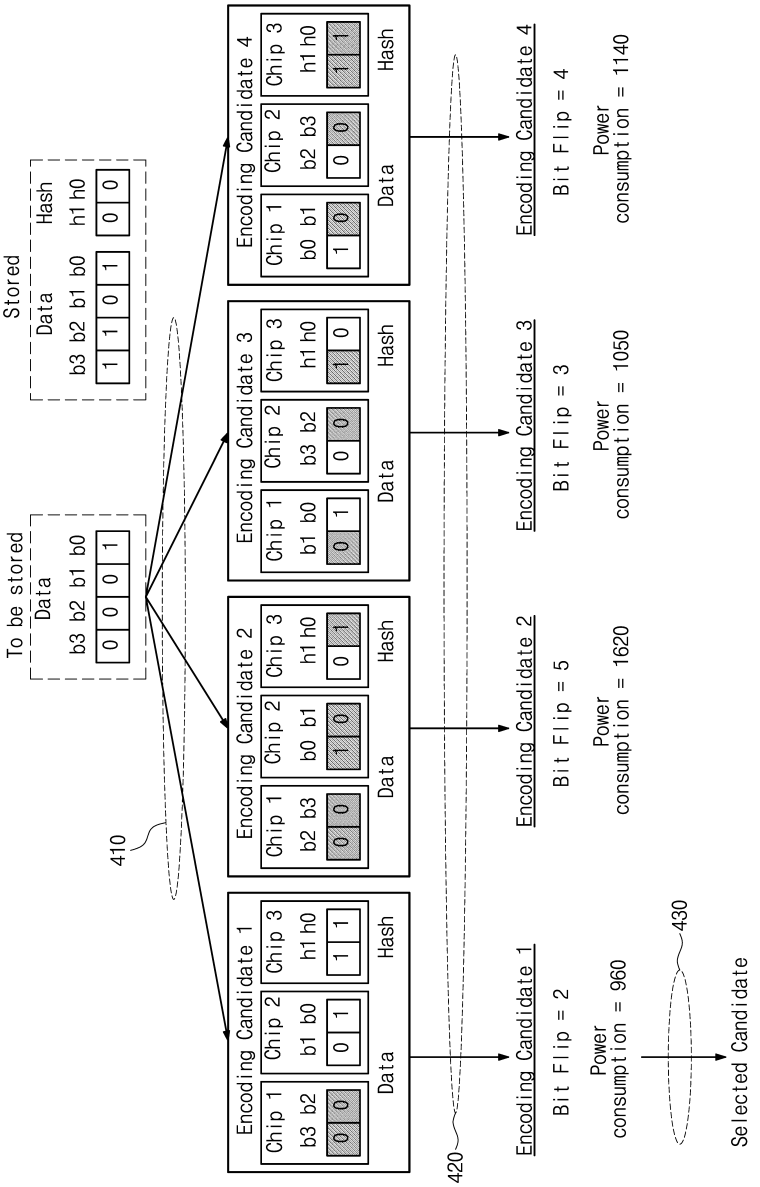
도면7


Start
Make encoding candidates by shuffling data to be stored with hashes
S110
Calculate bit flips and chip peak power consumption of encoding candidates
S120
Find candidates with slightly more bit flips than minimum one
S130
Select minimum chip peak power consumption among found candidates
S140
End

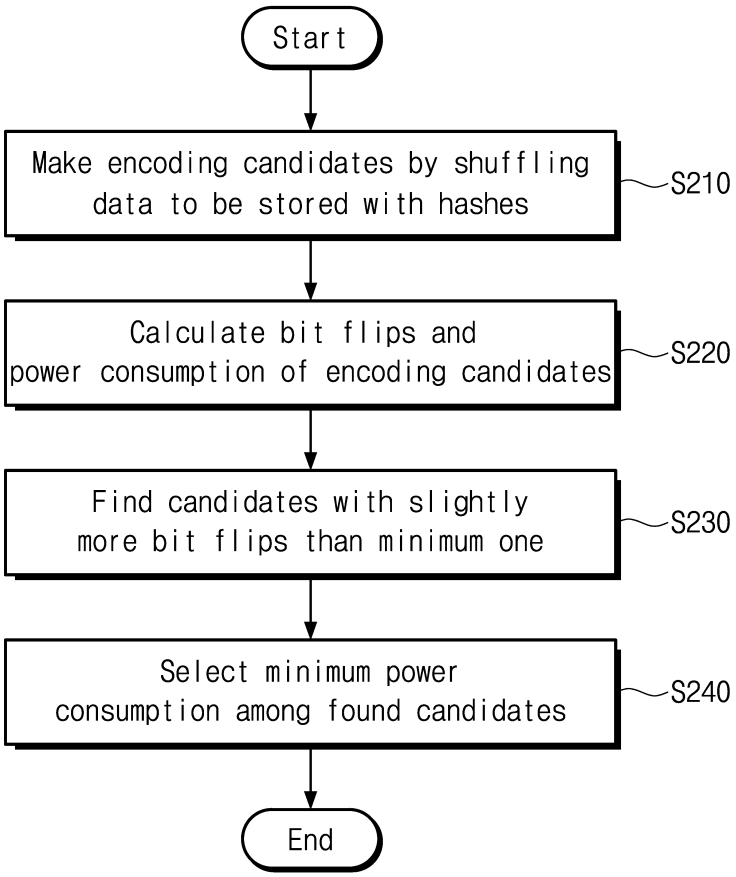
도면8


20
300
Memory Module
31N
Chip N
Bank M
Bank 1
312
Chip 2
Bank M
Bank 1
311
Chip 1
Bank M
Bank 1
Charge Pump
320
400
Memory Controller
Shuffling Logic
Bit Flip Calculator
Peak Power Calculator
Candidate Selector
410
420
421
422
430

도면9


To be stored
Data
b3 b2 b1 b0
0 0 0 1
Stored
Data Hash
b3 b2 b1 b0 h1 h0
1 1 0 1 0 0
410
Encoding Candidate 1
Chip 1 Chip 2 Chip 3
b3 b2 b1 b0 h1 h0
0 0 0 1 1 1
Data Hash
Encoding Candidate 2
Chip 1 Chip 2 Chip 3
b2 b3 b0 b1 h1 h0
0 0 1 0 0 1
Data Hash
Encoding Candidate 3
Chip 1 Chip 2 Chip 3
b1 b0 b3 b2 h1 h0
0 1 0 0 1 0
Data Hash
Encoding Candidate 4
Chip 1 Chip 2 Chip 3
b0 b1 b2 b3 h1 h0
1 0 0 0 1 1
Data Hash
420
Encoding Candidate 1
Bit Flip = 2
Power consumption = 960
Encoding Candidate 2
Bit Flip = 5
Power consumption = 1620
Encoding Candidate 3
Bit Flip = 3
Power consumption = 1050
Encoding Candidate 4
Bit Flip = 4
Power consumption = 1140
430
Selected Candidate

도면10


Start
Make encoding candidates by shuffling data to be stored with hashes
S210
Calculate bit flips and power consumption of encoding candidates
S220
Find candidates with slightly more bit flips than minimum one
S230
Select minimum power consumption among found candidates
S240
End