(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2025-0058999
(43) 공개일자 2025년05월02일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2023.01) *G06F 7/544* (2017.01)
G06N 3/04 (2023.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06F 7/5443 (2013.01)
(21) 출원번호 10-2023-0142830
(22) 출원일자 2023년10월24일
심사청구일자 2023년10월24일

(71) 출원인
울산과학기술원
울산광역시 울주군 언양읍 유니스트길 50
(72) 발명자
이규호
울산광역시 울주군 언양읍 유니스트길 50
박건희
울산광역시 울주군 언양읍 유니스트길 50
정희창
울산광역시 울주군 언양읍 유니스트길 50
(74) 대리인
특허법인 무한

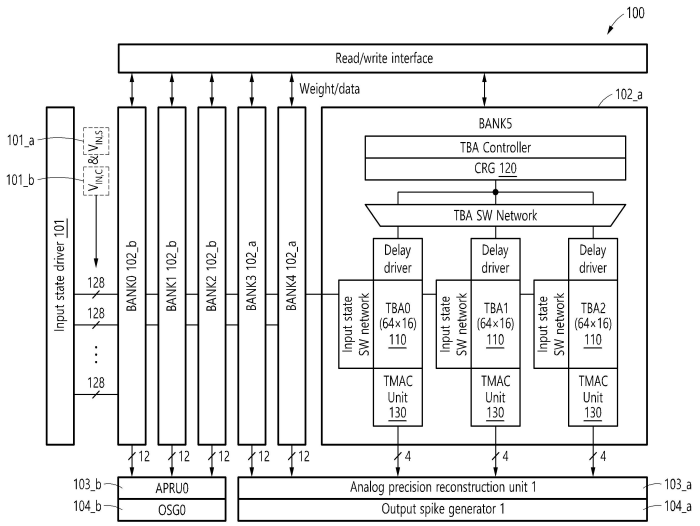
전체 청구항 수 : 총 14 항

(54) 발명의 명칭 **컴퓨팅 인 메모리 장치를 포함하고 시간 영역 연산을 활용하는 스파이킹 뉴럴 네트워크 연산 가속기**

(57) 요 약

일 실시예에 따른 SNN 연산 가속 장치는 서로 반대되는 방향으로 연결된 한 쌍의 인버터(inverter)들을 포함하고, 설정된 가중치를 가지는 정적 랜덤 액세스 메모리(static random access memory), 상기 한 쌍의 인버터들의 제1 단과 게이트(gate)를 통해 연결된 제1 트랜지스터, 상기 한 쌍의 인버터들의 제2 단과 게이트를 통해 연결되고, 일단에서 입력 신호를 수신 가능한 제2 트랜지스터, 및 상기 제1 트랜지스터 및 제2 트랜지스터와 게이트를 통해 연결되고, 지연 커패시터(delay capacitor) 및 출력 라인 사이에 연결되는 지연 트랜지스터를 가지는 비트셀(bitcell)을 포함할 수 있고, 상기 비트셀은, 상기 입력 신호의 값 및 상기 가중치에 기초하여 상기 출력 라인의 기준 신호를 지연시킬 수 있다.

대 표 도 - 도1



(52) CPC특허분류
G06N 3/049 (2023.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711194743
과제번호 00212222
부처명 과학기술정보통신부
과제관리(전문)기관명 한국연구재단
연구사업명 개인기초연구(과기정통부)
연구과제명 소형 인공지능을 위한 복합 다치로직 기반 초저전력, 초고효율 프로세서 구조 개발
기 여 율 70/100
과제수행기관명 울산과학기술원
연구기간 2023.03.01 ~ 2024.02.29
이 발명을 지원한 국가연구개발사업
과제고유번호 1711193271
과제번호 2022-0-01037-002
부처명 과학기술정보통신부
과제관리(전문)기관명 정보통신기획평가원
연구사업명 PIM인공지능반도체핵심기술개발(설계)
연구과제명 DRAM 기반 고성능 PIM 메모리 반도체 기술 개발
기 여 율 30/100
과제수행기관명 한국과학기술원
연구기간 2023.01.01 ~ 2023.12.31

명 세 서

청구범위

청구항 1

스파이킹 뉴럴 네트워크(spiking neural network) 연산 가속 장치에 있어서,

서로 반대되는 방향으로 연결된 한 쌍의 인버터(inverter)들을 포함하고, 설정된 가중치를 가지는 정적 랜덤 액세스 메모리(static random access memory),

상기 한 쌍의 인버터들의 제1 단과 게이트(gate)를 통해 연결된 제1 트랜지스터,

상기 한 쌍의 인버터들의 제2 단과 게이트를 통해 연결되고, 일단에서 입력 신호를 수신 가능한 제2 트랜지스터, 및

상기 제1 트랜지스터 및 상기 제2 트랜지스터와 게이트를 통해 연결되고, 지연 커패시터(delay capacitor) 및 출력 라인 사이에 연결되는 지연 트랜지스터

를 가지는 비트셀(bitcell)

을 포함하고,

상기 비트셀은,

상기 입력 신호의 값 및 상기 가중치에 기초하여 상기 출력 라인의 기준 신호를 지연시키는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 2

제1항에 있어서,

상기 비트셀은,

상기 제1 트랜지스터의 일단 및 상기 제2 트랜지스터의 일단에 공급 전압이 인가되는 것에 응답하여, 상기 지연 커패시터의 전압을 접지 전압(ground voltage)으로 리셋(reset)하는

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 3

제1항에 있어서,

상기 제2 트랜지스터는,

상기 설정된 가중치의 값이 제1 논리 값을 가지는 것에 응답하여, 상기 입력 신호를 상기 지연 트랜지스터의 게이트로 전달하고,

상기 지연 트랜지스터는,

상기 입력 신호의 값이 제1 논리 값을 가지는 것에 응답하여, 상기 출력 라인을 따른 상기 기준 신호가 지연되도록 상기 지연 커패시터를 상기 출력 라인에 연결하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 4

제1항에 있어서,

상기 지연 트랜지스터는,

상기 입력 신호의 값이 제2 논리 값을 가지는 것에 응답하여, 상기 출력 라인을 상기 지연 커패시터로부터 분리하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 5

제1항에 있어서,

상기 제2 트랜지스터는,

상기 설정된 가중치의 값이 제2 논리 값을 가지는 것에 응답하여, 상기 지연 커패시터에 의한 지연이 방지되도록 상기 지연 트랜지스터의 게이트를 향하는 상기 입력 신호를 차단(block)하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 6

제1항에 있어서,

상기 비트셀을 포함하는 복수의 비트셀들이 행들(rows) 및 열들(columns)로 배열되는 비트셀 어레이; 및

복수의 출력 라인들

을 포함하고

상기 복수의 출력 라인들의 각각은,

상기 열들 중 해당하는 열에 배열된 비트셀들과 연결되는

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 7

제6항에 있어서,

상기 복수의 비트셀들에 의해 지연된 기준 신호는,

입력 신호들의 값들 및 가중치들 간의 MAC(multiply-accumulate computation) 결과에 대응하는 지연 시간만큼, 원본 기준 신호 대비 지연된 신호이고,

상기 스파이킹 뉴럴 네트워크 연산 가속 장치는,

상기 원본 기준 신호 및 상기 지연된 기준 신호에 기초하여, 상기 MAC 결과에 대응하는 상기 지연 시간을 지시하는 출력 신호를 생성하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 8

제6항에 있어서,

복수의 입력 라인들

을 더 포함하고,

상기 복수의 입력 라인들 각각은,

상기 행들 중 해당하는 행에 배열된 비트셀들과 연결되는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 9

제6항에 있어서,

복수의 비트 자리들(bit positions)에 개별적으로 대응하는 복수의 시간 MAC(Multiply-and-Accumulate) 회로들을 더 포함하고,

상기 복수의 시간 MAC 회로들의 각각은,

MAC 결과에 대응하는 지연 시간을 지시하는 출력 신호를 전하로 변환하는 시간 대 전하 변환기(Time-to-Charge Converter); 및

상기 변환된 전하에서 해당하는 비트 자리에 따른 전하량을 보존하는 비트 전하 보존 회로

를 포함하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 10

제9항에 있어서,

상기 비트 전하 보존 회로는,

제1 커패시턴스를 갖는 제1 커패시터;

제2 커패시턴스를 갖는 제2 커패시터; 및

상기 제1 커패시터 및 상기 제2 커패시터 사이에 배치되는 스위치

를 포함하고,

상기 제1 커패시터의 제1 커패시턴스와 상기 제2 커패시터의 제2 커패시턴스의 합은 미리 결정된 값에 대응하며,

상기 비트 전하 보존 회로는,

상기 지연 시간을 지시하는 상기 출력 신호가 공급되는 동안 상기 제1 커패시터와 상기 제2 커패시터에 전하를 충전하고,

상기 지연 시간을 지시하는 상기 출력 신호가 종료하는 것에 응답하여, 상기 스위치를 이용하여 상기 제1 커패시터로부터 상기 제2 커패시터를 분리(disconnect)함으로써 상기 제2 커패시터에서 상기 해당하는 비트 자리에 따른 전하량을 저장하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 11

제10항에 있어서,

상기 제2 커패시터에 저장된 상기 해당하는 비트 자리에 따른 전하량을 합산하는 합산 커패시터

를 더 포함하는,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 12

제9항에 있어서,

가중치에 대응하는 전하량에 기초하여, 페이즈 코딩된 입력의 복수의 비트 자리들에 개별적으로 대응하는 복수의 시간 스텝(time-step)의 각각마다 상기 페이즈 코딩된 입력의 해당하는 비트 자리의 입력 값에 대한 MAC 결과를 생성하는 APR(Analog Precision Reconstruction) 유닛

을 더 포함하고,

상기 APR 유닛은,

복수의 시간 스텝들 별로, 복수의 가중치들 및 해당하는 입력 비트 자리의 입력 값들에 기초한 MAC 결과들의 합산에 대응하는 전하량에 기초하는 전하를 각각 저장하는 복수의 차지 홀더들(charge holders) 및

상기 복수의 차지 홀더들에 저장된 전하들로부터 복원 커패시터로의 전하 공유를 상기 복수의 시간 스텝들 별로 순차적으로 수행함으로써 커널 중 한 커널 원소에 대응하는 최종 MAC 결과를 생성하는 복원 메모리 회로

를 포함하는

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 13

제12항에 있어서,

상기 복수의 차지 홀더들에 포함된 커패시터의 커패시턴스 값과

상기 복원 메모리 회로에 포함된 커패시터의 커패시턴스 값이 동일한,

스파이킹 뉴럴 네트워크 연산 가속 장치.

청구항 14

제12항에 있어서,

상기 페이즈 코딩된 입력의 각 시간 스텝에 대응하는 MAC 결과를 축적하고,

상기 축적된 결과 값이 미리 결정된 임계값을 초과하는 것에 응답하여,

출력 스파이크를 생성하는 출력 스파이크 생성기(output spike generator)

를 더 포함하는

스파이킹 뉴럴 네트워크 연산 가속 장치.

발명의 설명

기 술 분 야

[0001] 이하, 스파이킹 뉴럴 네트워크 연산을 가속하는 기술에 관하여 설명한다.

배 경 기 술

[0003] 현재 폰노이먼(von Neumann) 컴퓨터 아키텍처에서 프로세서와 메모리 간 방대한 데이터의 잦은 이동은 긴 딜레

이와 큰 전력 소모를 야기하여 칩 성능을 제한시킨다. 현재 소프트웨어 기반의 딥신경망 연산은 고성능 CPU, GPU, ASIC 등의 AI 가속기 하드웨어가 사용되고 있다.

[0004] 스파이킹 뉴럴 네트워크(Spiking neural networks, SNN) 연산을 위한 컴퓨팅 인 메모리(computing in memory, CIM) 장치는 데이터가 저장된 메모리 소자의 위치에서 바로 연산을 진행하며 뉴런 회로 간의 연결 강도(예: 시냅스 가중치)를 메모리 소자에 저장하고 갱신할 수 있다. SNN 연산 방식은 인공지능, 빅데이터, 센서네트워크, 패턴/사물 인식 등에 적용될 수 있다.

발명의 내용

과제의 해결 수단

[0005] 일 실시예에 따른 스파이킹 뉴럴 네트워크(spiking neural network) 연산 가속 장치는, 서로 반대되는 방향으로 연결된 한 쌍의 인버터(inverter)들을 포함하고, 설정된 가중치를 가지는 정적 랜덤 액세스 메모리(static random access memory), 상기 한 쌍의 인버터들의 제1 단과 게이트(gate)를 통해 연결된 제1 트랜지스터, 상기 한 쌍의 인버터들의 제2 단과 게이트를 통해 연결되고, 일단에서 입력 신호를 수신 가능한 제2 트랜지스터, 및 상기 제1 트랜지스터 및 제2 트랜지스터와 게이트를 통해 연결되고, 지연 커패시터(delay capacitor) 및 출력 라인 사이에 연결되는 지연 트랜지스터를 가지는 비트셀(bitcell)을 포함하고, 상기 비트셀은, 상기 입력 신호의 값 및 상기 가중치에 기초하여 상기 출력 라인의 기준 신호를 지연시킬 수 있다.

[0006] 상기 비트셀은, 상기 제1 트랜지스터의 일단 및 상기 제2 트랜지스터의 일단에 공급 전압이 인가되는 것에 응답하여, 상기 지연 커패시터의 전압을 접지 전압(ground voltage)으로 리셋(reset)할 수 있다.

[0007] 상기 제2 트랜지스터는, 상기 설정된 가중치의 값이 제1 논리 값을 가지는 것에 응답하여, 상기 입력 신호를 상기 지연 트랜지스터의 게이트로 전달하고, 상기 지연 트랜지스터는, 상기 입력 신호의 값이 제1 논리 값을 가지는 것에 응답하여, 상기 출력 라인을 따른 상기 기준 신호가 지연되도록 상기 지연 커패시터를 상기 출력 라인에 연결할 수 있다.

[0008] 상기 지연 트랜지스터는, 상기 입력 신호의 값이 제2 논리 값을 가지는 것에 응답하여, 상기 출력 라인을 상기 지연 커패시터로부터 분리할 수 있다.

[0009] 상기 제2 트랜지스터는, 상기 설정된 가중치의 값이 제2 논리 값을 가지는 것에 응답하여, 상기 지연 커패시터에 의한 지연이 방지되도록 상기 지연 트랜지스터의 게이트를 향하는 상기 입력 신호를 차단(block)할 수 있다.

[0010] 상기 비트셀을 포함하는 복수의 비트셀들이 행들(rows) 및 열들(columns)로 배열되는 비트셀 어레이; 및 복수의 출력 라인들을 포함하고 상기 복수의 출력 라인들의 각각은, 상기 열들 중 해당하는 열에 배열된 비트셀들과 연결될 수 있다.

[0011] 상기 복수의 비트셀들에 의해 지연된 기준 신호는, 입력 신호들의 값들 및 가중치들 간의 MAC(multiply-accumulate computation) 결과에 대응하는 지연 시간만큼, 원본 기준 신호 대비 지연된 신호이고, 상기 스파이킹 뉴럴 네트워크 연산 가속 장치는, 상기 원본 기준 신호 및 상기 지연된 기준 신호에 기초하여, 상기 MAC 결과에 대응하는 상기 지연 시간을 지시하는 출력 신호를 생성할 수 있다.

[0012] 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치는, 복수의 입력 라인들을 더 포함하고, 상기 복수의 입력 라인들 각각은, 상기 행들 중 해당하는 행에 배열된 비트셀들과 연결될 수 있다.

[0013] 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치는, 복수의 비트 자리들(bit positions)에 개별적으로 대응하는 복수의 시간 MAC(Multiply-and-Accumulate) 회로들을 더 포함하고, 상기 복수의 시간 MAC 회로들의 각각은, MAC 결과에 대응하는 지연 시간을 지시하는 상기 출력 신호를 전하로 변환하는 시간 대 전하 변환기(Time-to-Charge Converter); 및 상기 변환된 전하에서 해당하는 비트 자리에 따른 전하량을 보존하는 비트 전하 보존 회로를 포함할 수 있다.

[0014] 상기 비트 전하 보존 회로는, 제1 커패시턴스를 갖는 제1 커패시터; 제2 커패시턴스를 갖는 제2 커패시터; 및 상기 제1 커패시터 및 상기 제2 커패시터 간에 배치되는 스위치를 포함하고, 상기 제1 커패시터의 제1 커패시턴스와 상기 제2 커패시터의 제2 커패시턴스의 합은 미리 결정된 값에 대응하며, 상기 비트 전하 보존부는, 상기 지연 시간을 지시하는 상기 출력 신호가 공급되는 동안 상기 제1 커패시터와 상기 제2 커패시터에 전하를 충전하고, 상기 지연 시간을 지시하는 상기 출력 신호가 종료하는 것에 응답하여, 상기 스위치를 이용하여 상기 제1

커패시터로부터 상기 제2 커패시터를 분리(disconnect)함으로써 상기 제2 커패시터에서 상기 해당하는 비트 자리에 따른 전하량을 저장할 수 있다.

[0015] 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치는, 상기 제2 커패시터에 저장된 상기 해당하는 비트 자리에 따른 전하량을 합산하는 합산 커패시터를 더 포함할 수 있다.

[0016] 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치는, 가중치에 대응하는 전하량에 기초하여, 페이즈 코딩된 입력의 복수의 비트 자리들에 개별적으로 대응하는 복수의 시간 스텝(time-step)의 각각마다 상기 페이즈 코딩된 입력의 해당하는 비트 자리의 입력 값에 대한 MAC 결과를 생성하는 APR(Analog Precision Reconstruction) 유닛을 더 포함하고, 상기 APR 유닛은, 복수의 시간 스텝들 별로, 복수의 가중치들 및 해당하는 입력 비트 자리의 입력 값들에 기초한 MAC 결과들의 합산에 대응하는 전하량에 기초하는 전하를 각각 저장하는 복수의 차지 홀더들(charge holders) 및 상기 복수의 차지 홀더들에 저장된 전하들로부터 복원 커패시터로의 전하 공유를 상기 복수의 시간 스텝들 별로 순차적으로 수행함으로써 커널 중 한 커널 원소에 대응하는 최종 MAC 결과를 생성하는 복원 메모리 회로를 포함할 수 있다.

[0017] 상기 복수의 차지 홀더들에 포함된 커패시터의 커패시턴스 값과 상기 복원 메모리 회로에 포함된 커패시터의 커패시턴스 값이 동일할 수 있다.

[0018] 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치는, 상기 페이즈 코딩된 MAC 결과를 축적하고, 상기 축적된 결과 값이 미리 결정된 임계값을 초과하는 것에 응답하여, 출력 스파이크를 생성하는 출력 스파이크 생성기(output spike generator)

[0019] 를 더 포함할 수 있다.

도면의 간단한 설명

[0021] 도 1은 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치의 아키텍처를 도시한다.

도 2는 일 실시예에 따른 SNN 연산 가속 장치에 의한 SNN 연산 가속 방법 흐름도다.

도 3은 APR 유닛에 입력하기 위한 페이즈 코딩된 신호를 개략적으로 나타낸다.

도 4는 SNN 연산 가속 장치가 입력 액티베이션과 가중치 커널 간의 커널 연산을 통해 MAC 결과를 획득하고, 획득된 MAC 결과가 APR 유닛으로 시간 스텝에 따라 순차적으로 입력되는 것을 도시한다.

도 5는, 일 실시예에 따른 시간 영역 비트셀 어레이 구조를 도시한다.

도 6은 일 실시예에 따른 기준 신호를 지연할 수 있는 비트셀의 구조를 도시한다.

도 7은 TBA에 포함된 하나의 출력 라인을 따라 흐르는 기준 신호가 비트셀들에 의해 지연되는 것 및 기준 신호의 지연이 누적된 MAC 펄스를 형성하는 것을 나타낸다.

도 8은, 출력 라인 별 출력 신호의 비트 정확도를 고려하기 위한 시간 MAC 회로의 구성을 나타낸다.

도 9는 시간 MAC 회로의 순차적인 동작을 설명하는 도면이다.

도 10은 APR 유닛의 구성을 개략적으로 도시한다.

도 11은 APR 유닛의 출력 액티베이션 생성 타이밍도를 나타낸다.

도 12a는 출력 라인에 연결된 복수의 비트셀들로 인한 기준 신호의 딜레이에 기초하여 형성된 스파이크 신호의 누적 결과를 도시한다.

도 12b는 시간 MAC 회로에서 MAC 값에 따른 전압을 나타낸다.

도 12c는APR 유닛에서의 최종 MAC 결과에 따른, 최종 MAC 결과의 전압을 나타낸다.

도 12d는복수의 시간 스텝 동안 APR 유닛에서 가중치에 대응하는 MAC 결과를 최종 MAC 결과로 재구성하는 과정을 그래프를 통해 도시한다.

도 13은 일 실시예에 따른 SNN 연산 가속 장치에 포함된 9T1C 구조의 비트셀 레이아웃을 나타낸다.

도 14는 일 실시예에 따른 SNN 연산 가속 장치의 레이아웃을 나타낸다.

발명을 실시하기 위한 구체적인 내용

[0022] 실시예들에 대한 특정한 구조적 또는 기능적 설명들은 단지 예시를 위한 목적으로 개시된 것으로서, 다양한 형태로 변경되어 구현될 수 있다. 따라서, 실제 구현되는 형태는 개시된 특정 실시예로만 한정되는 것이 아니며, 본 명세서의 범위는 실시예들로 설명한 기술적 사상에 포함되는 변경, 균등물, 또는 대체물을 포함한다.

[0023] 제1 또는 제2 등의 용어를 다양한 구성요소들을 설명하는데 사용될 수 있지만, 이런 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 해석되어야 한다. 예를 들어, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소는 제1 구성요소로도 명명될 수 있다.

[0024] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다.

[0025] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 설명된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함으로 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0026] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 해당 기술분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0027] 이하, 실시예들을 첨부된 도면들을 참조하여 상세하게 설명한다. 첨부 도면을 참조하여 설명함에 있어, 도면 부호에 관계없이 동일한 구성 요소는 동일한 참조 부호를 부여하고, 이에 대한 중복되는 설명은 생략하기로 한다.

[0029] 스파이킹 뉴럴 네트워크(Spiking Neural Network, SNN)는 인간 뇌에서 뉴런들의 시냅스 연결을 모델링하여 고에너지 효율을 위한 유망한 알고리즘으로서 활발히 연구되고 있다. SNN의 다양한 모델링 방법들 중 스파이크 입력 및 스파이크 출력을 다루는 통합-실행 뉴런 모델(Integrate-and-Fire neuron model, IF)이 널리 사용된다. IF 모델의 막전위는 매 시간 스텝(time step)에서 스파이크 입력의 양에 따라 누적되고, IF 모델은 막전위가 특정 임계값을 초과하면 출력 스파이크를 발화(fire)한다. SNN은 기존의 심층 뉴럴 네트워크(deep neural networks)에 비해 에너지 효율이 높은 것으로 알려져 있다. 이러한 이유로, 여러 저전력 프로세서들이 제안되었으나, 시스템 전력 소비량은 여전히 높다(>20 mW).

[0030] 컴퓨팅 인 메모리(computing in memory, CIM) 아키텍처는 에너지 효율성을 향상시키기 위해 다른 하드웨어 측면에서 적극적으로 연구되고 있다. 폰 노이만 아키텍처 기반의 기존 프로세서들은 연산보다는 데이터 이동에 많은 에너지를 소비하기 때문에 저전력 SNN 가속에 적합하지 않다. 반면, CIM은 읽기(read) 작업과 동시에 메모리 코어 내부의 시냅스 연산이 가능하여 데이터 이동으로 인한 에너지를 줄일 수 있다. 따라서, CIM은 SNN 구현에 적합하다. CIM은 연산하는 도메인에 따라 아날로그 CIM과 디지털 CIM으로 분류될 수 있다. 디지털 CIM은 프로세스, 전압 및 온도 노이즈에 강하지만, 디지털 로직(digital logic)이 큰 면적을 차지하고, 긴 연산 레이턴시(computation latency)를 갖기 때문에 제한된 메모리 용량을 갖는 단점이 있다. 반면 아날로그 CIM은 노이즈에 취약하지만, 매크로(macro)의 전체 비트셀을 동시에 활용함으로써, 에너지 효율을 높일 수 있다는 큰 장점이 있다. 아날로그 CIM은 아날로그 도메인에서 프로세싱 후 데이터 변환을 위해, 일반적으로 전체 전력 소비량의 50%를 차지하며 많은 전력을 소비하는 아날로그-디지털 컨버터(ADC)를 필요로 한다. 그러나 IF 모델은 막전위를 임계값과 비교함으로써 아날로그 값을 디지털 비트들로 변환한다. 따라서, 전력 소모가 심한 ADC는 SNN의 저전력 소비를 가능하게 하는 간단한 비교기(comparator)로 대체될 수 있다.

[0031] 일반적인 아날로그 CIM은 입력과 가중치를 비트셀에서 곱하고, 비트라인(BL)을 통해 그 결과 값들을 누적하여 심층 신경망 연산의 대부분을 차지하는 곱셈-누적(MAC) 연산을 수행할 수 있다. BL을 통해 값을 누적하는 방법은 전류 모드, 전압 모드, 전하 공유, 및 전하 커플링이 있다. 그러나, 이러한 방법들은, BL의 최대 전압이 제한되어 있기 때문에, 표현해내는 값의 전압 간격(voltage interval)이 매우 좁다. 그 결과 판독 정확도가 저하

되고 정교한 ADC가 필요하므로 더 많은 전력 소비가 필요하다. 이러한 전력 오버헤드를 해결하기 위해 종래에 시간 영역 처리(time-domain processing)가 제안되었다. 단순히 시간 스텝을 늘려서 높은 판독 정확도로 광범위한 MAC 값들을 계산할 수 있다. 또한, ADC를 TDC(time-to-digital converter)로 대체하면 에너지 효율이 향상된다.

[0032] 종래에는 1비트 이진 신호의 상승 에지(rising edge) 및 하강 에지(falling edge)를 모두 사용하는 시간 영역의 연산을 제안했다. 그러나, 해당 방법은 최대 동작 주파수를 2 MHz로 제한했다. 게다가, 이미지 분류에서 높은 정확도(accuracy)를 달성하는 더 복잡하고 심층적인 뉴럴 네트워크에는 멀티 비트 정확도(multi bit precision)가 필요하다. 그러나, 멀티 비트 CIM은 개별 비트들을 멀티 비트로 변환하기 위해 TDC, 시프터(shifter), 및 가산기(adder)를 사용하여 여러 사이클을 소모한다.

[0033] 이에 따라, 본 명세서에서는 일 실시예에 따른 높은 에너지 효율의 SNN 가속을 달성하기 위한 SNN 연산 가속 장치를 제안한다. 해당 장치는: 1) 저전력 MAC 연산을 위한 시간 영역 비트셀 어레이(Time-Domain Bitcell Array); 2) 시프터 및 가산기 없이 멀티 비트 정확도를 지원하는 TMAC(Time-Domain Multiply-and Accumulate) 회로; 3) 페이즈 코딩된 입력 스파이크들을 컨트롤하기 위한 APR(Analog Precision Reconstruction) 유닛을 특징으로 가진다.

[0034] 스파이킹 뉴럴 네트워크(Spiking neural networks, SNN)(이하, 'SNN') 모델은 생물학적 뇌를 구성하는 뉴런이 출력을 결정하는 메커니즘을 모방하는 인공 신경망의 한 종류이다. 예를 들어, 뉴런이 출력을 결정하는 메커니즘이란, 뉴런이 이전 뉴런으로부터 시냅스를 통해 신호를 전달받고, 전달받은 신호를 축적함으로써 축적된 신호가 임계값(예: 막전위)을 초과하는 경우 활동 전위(action potential) 또는 스파이크(spike)라 불리는 전기적 펄스를 생성하는 것을 포함할 수 있다. SNN 모델은 스파이킹 뉴런(spiking neuron), 시냅스(synapse), 및 뉴런과 시냅스 간의 상호 연결(interconnection)으로 구성될 수 있다. SNN 모델은 시냅스를 통해 연결된 하나의 뉴런에서 다른 뉴런으로 스파이크 형태의 정보를 전달할 수 있다. 예를 들어 SNN 모델 중 하나인 Leaky Integrate and Fire(LIF) 모델은, 대상 뉴런과 연결된 이전 뉴런들로부터의 스파이크를 누적할 수 있다. LIF 모델은 누적된 스파이크들의 값이 임계치(예: 임계 전하량 또는 임계 전압)를 넘으면 스파이크를 생성하고, 대상 뉴런에 누적된 스파이크 값을 리셋할 수 있다. 대상 뉴런에 의해 생성된 스파이크는 대상 뉴런과 시냅스를 통해 연결된 다음 뉴런으로 전파될 수 있다. 이상에서 설명한 LIF 모델은 SNN 모델링 방법의 일 예에 불과하며, 본 명세서에서 구현되는 SNN 모델은 이에 제한되지 않는다. 이하 첨부된 도면들에서 SNN 연산 가속 장치에 대해 설명한다.

[0036] 도 1은 일 실시예에 따른 스파이킹 뉴럴 네트워크 연산 가속 장치의 아키텍처를 도시한다.

[0037] 일 실시예에 따른 SNN 연산 가속 장치(100)는, 시간 영역에서 SNN 연산을 수행함으로써, SNN 연산을 가속하는 장치일 수 있다. 예를 들어, SNN 연산 가속 장치(100)는 입력 신호(예: 펄스 신호)를 지연(delay)시킬 수 있다. SNN 연산 가속 장치(100)는 입력 신호와 지연된 신호(예: 입력 펄스가 지연된 펄스) 간의 연산을 통해, 입력 신호와 지연된 신호 간의 시간 차이를 펄스 폭으로 하는 스파이크 신호를 생성할 수 있다. 참고로, 스파이크 신호는, 입력 신호와 지연된 신호 간의 XOR 연산을 통해 생성될 수 있다. 스파이크 신호의 펄스 폭은 SNN 연산에서 스파이크가 누적된 값(예: 스파이크 값)에 대응할 수 있다. 다시 말해, 스파이크 값이 시간 영역의 값으로서 표현되어 처리될 수 있다. SNN 연산 가속 장치(100)에 의해 입력 신호를 지연시키는 방법은 이하 도 5 내지 도 7에서 구체적으로 설명한다.

[0038] SNN 연산 가속 장치(100)는, 입력 상태 드라이버(input state driver)(101), 시간 영역 CIM 뱅크(102_a, 102_b), APR(Analog Precision Reconstruction) 유닛(103_a, 103_b), 및 출력 스파이크 생성기(Output Spike Generator, OSG)(이하, 'OSG')(104_a, 104_b)를 포함할 수 있다. 예를 들어, SNN 연산 가속 장치(100)는 하나의 입력 상태 드라이버(101)와 연결된 복수의 입력 라인들 각각과 개별적으로 연결된 6개의 시간 영역 CIM 뱅크(102_a, 102_b)들을 포함할 수 있다. 6개의 시간 영역 CIM 뱅크(102_a, 102_b)는 각각 시간 영역 CIM 뱅크 0 내지 시간 영역 CIM 뱅크 5로 구별될 수 있다. SNN 연산 가속 장치(100)는 3개의 시간 영역 CIM 뱅크(102_a)와 연결되는 APR 유닛(103_a) 및 OSG(104_a) 한 쌍과 다른 3개의 시간 영역 CIM 뱅크(102_b)와 연결되는 APR 유닛(103_b) 및 OSG(104_b) 한 쌍을 포함할 수 있다.

[0039] 입력 상태 드라이버(101)는 입력 신호(101_a, 101_b)를 생성하고, 입력 라인을 통해 입력 신호를 시간 영역 CIM 뱅크(102_a, 102_b)로 전달할 수 있다. 예를 들어, 입력 상태 드라이버(101)에 의해 생성된 입력 신호(101_a,

101_b)는 비트 정보(예: 1 또는 0) 또는 전압에 대응하는 펄스 신호를 나타낼 수 있다. SNN 연산 가속 장치(100)는 입력 상태 드라이버(101)로부터 멀티 비트 입력을 생성하도록, 복수의 입력 라인들을 더 포함할 수 있다. 복수의 입력 라인들 각각은 시간 영역 비트셀 어레이(time domain bitcell array, TBA)(이하, 'TBA')(110)의 행들 중 해당하는 행에 배열된 비트셀들과 연결될 수 있다.

[0041] 시간 영역 CIM 뱅크(102_a, 102_b)는 입력 상태 드라이버(101)로부터 수신된 입력 신호(101_a, 101_b)에 기초하여, 시간 지연을 포함하는 MAC 펄스를 생성할 수 있다. 시간 지연을 포함하는 MAC 펄스는 입력 신호(101_a, 101_b)와 TBA에 저장된 가중치 간의 연산 결과에 기초하여, 기준 신호를 지연함으로써 생성된 신호일 수 있다. 시간 영역 CIM 뱅크(102_a, 102_b)는 생성된 MAC 펄스에 기초하여, 입력 신호에 대한 커널 연산을 수행할 수 있다. 예를 들어, 하나의 시간 영역 CIM 뱅크는 3×1 커널 연산을 수행할 수 있다. 시간 영역 CIM 뱅크의 커널 연산에 대한 구체적인 설명은 이하 도 4에서 설명한다.

[0042] 시간 영역 CIM 뱅크(102_a, 102_b)는 TBA(110), TBA 컨트롤러, TBA 스위치 네트워크, 기준 신호 생성기(Computation Reference Generator, CRG)(이하, 'CRG')(120), 및 시간 MAC(Multiply-and-Accomulate) 회로(130)를 포함할 수 있다.

[0043] TBA(110)는 일 실시예에 따른 9T1C 구조의 비트셀을 포함하며, 비트셀은 비트 가중치를 저장할 수 있다. TBA(110)에 포함된 복수의 비트셀들로 멀티 비트 가중치를 저장할 수 있다. 예를 들어, TBA(110)에서 비트셀들이 하나의 행에 16개 배열된 경우, TBA(110)의 해당 행은 4개의 4비트 가중치를 저장할 수 있다. TBA(110)의 구조 및 TBA(110)에 포함된 비트셀의 구조에 대하여 이하 도 5 및 도 6에서 설명한다.

[0044] 시간 영역 CIM 뱅크(102_a, 102_b)는 복수의 TBA(110)를 포함할 수 있다. 예를 들어, 시간 영역 CIM 뱅크(102_a, 102_b)는 3개의 서로 다른 TBA를 포함할 수 있다. 하나의 시간 영역 CIM 뱅크에 포함된 3개의 서로 다른 TBA(110)는 TBA 0, TBA 1, TBA 2로 구분될 수 있다.

[0045] SNN 연산 가속 장치(100)는 시간 영역 CIM 뱅크(102_a, 102_b)에 복수의 TBA(110)가 포함된 경우, TBA 컨트롤러 및 TBA 스위치 네트워크를 통해 각 TBA(110)를 병렬적으로 작동할 수 있다.

[0046] CRG(120)는 시간 지연을 포함하는 스파이크 신호 생성을 위한 기준 신호를 생성할 수 있다. CRG(120)는 출력 라인을 통해 TBA(110)와 연결될 수 있다. 예를 들어, 하나의 시간 영역 CIM 뱅크에 포함된 CRG(120)는, 시간 영역 CIM 뱅크에 포함된 복수의 TBA(110)와 출력 라인을 통해 연결될 수 있다. 복수의 TBA(110)와 연결된 출력 라인 각각으로부터 서로 다른 지연 폭을 포함하는 스파이크 신호가 출력될 수 있다.

[0047] 시간 MAC 회로(130)는, 출력 라인을 통해, 기준 신호에 기초하여 지연된 폭을 갖는 스파이크 신호를 수신할 수 있다. SNN 연산 가속 장치(100)는 시간 MAC 회로(130)를 통해 TBA(110)에 포함된 복수의 가중치 비트의 비트 위치를 고려할 수 있다. 예를 들어, TBA(110)가 하나의 행에 4개의 4비트 가중치를 저장하는 경우, 시간 MAC 회로(130)는 시간 지연을 포함하는 스파이크 신호를 4비트 정확도로 변환할 수 있다.

[0048] APR(Analog Precision Reconstruction) 유닛(103_a, 103_b)은 시간 영역 CIM 뱅크(102_a, 102_b)로부터 수신된 전하를 시간 스텝에 대응하도록 정렬함으로써, 페이즈 코딩된 입력 신호들(예: 입력 스파이크)에 대한 아날로그 정확도를 보상할 수 있다. 입력 신호의 페이즈 코딩 및 APR 유닛(103_a, 103_b)의 구체적인 동작 방법은 이하 도 3 및 도 4에서 설명한다.

[0049] 일 실시예에 따른, SNN 연산 가속 장치(100)는 페이즈 코딩된 입력 신호의 각 시간 스텝에 대응하는 MAC 결과를 축적하고, 축적된 결과 값이 미리 결정된 임계값을 초과하는 것에 응답하여, 출력 스파이크를 생성하는 OSG(104_a, 104_b)를 더 포함할 수 있다. OSG(104_a, 104_b)는 APR 유닛(103_a, 103_b)으로부터 수신된 전하가 임계값을 초과하는 경우 출력 스파이크를 생성할 수 있다. OSG(104_a, 104_b)에 의해 생성된 출력 스파이크는 발화 시간과 함께 디지털 비트들로 저장될 수 있다.

[0050] SNN 연산 가속 장치(100)에 포함된 3개의 시간 영역 CIM 뱅크(102_a)와 3개의 시간 영역 CIM 뱅크(102_b)는 개별적으로 동작할 수 있다. 예를 들어, SNN 연산 가속 장치(100)가 하나의 시간 영역 CIM 뱅크로 3×1 커널 연산하는 경우, 한 쌍의 APR 유닛(103_a)과 OSG(104_a)는 3개의 시간 영역 CIM 뱅크(102_a)와 함께 동작할 수 있다. SNN 연산 가속 장치(100)를 파이프라인 구조로 설계하기 위해, 3개의 시간 영역 CIM 뱅크(102_a)는, 다른 3개의 시간 영역 CIM 뱅크(102_b)가 가중치를 업데이트 하는 동안, 멀티 비트 MAC 연산을 위해 동작할 수 있다.

[0052] 도 2는 일 실시예에 따른 SNN 연산 가속 장치에 의한 SNN 연산 가속 방법 흐름도다.

[0053] 단계(210)에서, SNN 연산 가속 장치는 입력 신호와 비트셀에 저장된 가중치 간의 연산 결과에 기초하여, 원본 기준 신호를 지연할 수 있다. 예를 들어, 비트셀에 저장된 가중치가 [1, 0]에 해당하고, 입력 신호가 스파이크 신호에 해당하는 경우, 비트셀이 활성화될 수 있다. 활성화된 비트셀은 출력 라인을 따라 흐르는 원본 기준 신호를 지연시킬수 있다. 비트셀 활성화에 의한 원본 기준 신호의 지연에 대한 설명은 이하 도 7에서 구체적으로 설명한다.

[0054] 단계(220)에서, SNN 연산 가속 장치는 지연된 기준 신호와 원본 기준 신호 간의 시간 차이를 폭으로 하는 MAC 펄스를 생성할 수 있다. 예를 들어, SNN 연산 가속 장치는 원본 기준 신호와 지연된 기준 신호 간의 XOR 연산을 통해 MAC 펄스를 생성할 수 있다. 생성된 MAC 펄스는 원본 기준 신호의 상승 에지(edge)에서 지연된 기준 신호의 상승 에지 간의 차이를 폭으로 하는 스파이크 신호일 수 있다.

[0055] 단계(230)에서, SNN 연산 가속 장치는 가중치의 비트 정확도(예: 멀티 비트 가중치의 각 비트 자리)를 고려하기 위해 미리 결정된 커패시턴스를 갖는 커패시터에, 생성된 MAC 펄스에 기초하여 전하를 저장할 수 있다. SNN 연산 가속 장치는 복수의 시간 MAC 회로들을 포함할 수 있다. 시간 MAC 회로들은 생성된 MAC 펄스에 기초하여, 커패시터들에 전하를 저장할 수 있다. 예를 들어, SNN 연산 가속 장치는 4비트 정확도를 고려하기 위해 1C, 2C, 4C, 및 8C 크기의 커패시턴스를 갖는 커패시터에 전하를 저장할 수 있다. 각 커패시턴스 크기에 대응하는 커패시터에 저장된 전하는 가중치의 비트 자리에 대응할 수 있다. 시간 MAC 회로에 대한 구체적인 구조는 이하 도 8에서 설명한다.

[0056] 단계(240)에서, SNN 연산 가속 장치는 저장된 전하들을 합산 커패시터를 통해 합산하고, 합산된 전하들을 페이즈 코딩함으로써 각 시간 스텝에 대응하는 최종 MAC 결과를 획득할 수 있다. SNN 연산 가속 장치는 합산된 전하들을 페이즈 코딩하여, 순차적으로 APR 유닛에 입력할 수 있다. SNN 연산 가속 장치는 APR 유닛에 순차적으로 입력된 합산된 전하들에 기초하여, 페이즈 코딩된 입력 신호의 각 시간 스텝에 대응하는 최종 MAC 결과를 획득할 수 있다. 예를 들어, APR 유닛에 $1000_{(2)}$의 입력이 페이즈 코딩되어 입력되는 경우, 제1 시간 스텝에 LSB에 해당하는 0이 APR 유닛에 입력될 수 있다. 그 후 제4 시간 스텝에 MSB에 해당하는 1이 APR 유닛에 입력될 수 있다. SNN 연산 가속 장치는 LSB에 해당하는 신호에서 MSB에 해당하는 신호까지 순차적으로 APR 유닛에 입력함으로써 입력의 각 시간 스텝에 대응하는 최종 MAC 결과를 획득할 수 있다. APR 유닛의 동작에 대하여 이하 도 10 내지 도 11에서 설명한다.

[0057] 단계(250)에서, SNN 연산 가속 장치는, 획득된 최종 MAC 결과를 축적함으로써, 축적된 MAC 결과가 미리 결정된 임계치를 넘는 경우 출력 스파이크를 생성할 수 있다. SNN 연산 가속 장치는 최종 MAC 결과를 축적하기 위해 단계(210) 내지 단계(240)을 반복할 수 있다.

[0059] 도 3은 APR 유닛에 입력하기 위한 페이즈 코딩된 신호를 개략적으로 나타낸다.

[0060] 비교 실시예는 입력 신호를 입력 값의 수만큼의 입력 스파이크로 변환하는 레이트 코딩(rate coding) 방식을 채택할 수 있다. 그러나 레이트 코딩 방식은 비트 정확도(bit precision)가 커지면 신호 처리를 위한 레이턴시(latency)가 직접적으로 늘어난다.

[0061] 일 실시예에 따른, SNN 연산 가속 장치는, 레이트 코딩 방식 대신, 페이즈 코딩(phase coding) 방식을 통해 입력 신호(300)를 처리할 수 있다. 참고로, 페이즈 코딩 방식은 각 시간 스텝(301)이 비트 위치를 나타내고, 4개의 시간 스텝(301)으로 4비트 입력을 나타낼 수 있다. 예를 들어, 입력 신호(300)는 디지털 입력 또는 아날로그 입력 중 하나일 수 있다. 예를 들어, 디지털 입력은 4 비트 입력으로서, $1011_{(2)}$, $0110_{(2)}$, 또는 $1010_{(2)}$에 대응할 수 있다. 다른 예를 들어, 아날로그 입력은 픽셀 값을 포함할 수 있다. 본 명세서에서는, 이하 디지털 입력을 기준으로 설명한다.

[0062] SNN 연산 가속 장치는, 페이즈 코딩 방식을 통해, 4비트 입력 신호(310)에 포함된 각 비트를 시간 스텝(301) 각각과 대응할 수 있다. 예를 들어, $1011_{(2)}$의 디지털 입력 신호(310)가 페이즈 코딩 된 경우, LSB(Least Significant Bit)에 해당하는 비트(311a)가 시간 스텝 0에 대응하는 스파이크(311b)로 변환될 수 있고, MSB(Most Significant Bit)에 해당하는 비트(312a)가 시간 스텝 3에 대응하는 스파이크(312b)로 변환될 수 있

다. 일 실시예에 따른 SNN 연산 가속 장치는 시간 스텝 0에 대응하는 스파이크(311b)부터 시간 스텝 3에 대응하는 스파이크(312b)까지 순차적으로 APR 유닛에 입력할 수 있다.

[0064] 도 4는 SNN 연산 가속 장치가 입력 액티베이션과 가중치 커널 간의 커널 연산을 통해 MAC 결과를 획득하고, 획득된 MAC 결과가 APR 유닛으로 시간 스텝에 따라 순차적으로 입력되는 것을 도시한다.

[0065] 일 실시예에 따른, SNN 연산 가속 장치는 입력 액티베이션(input activation)(401)에 가중치 커널(402)을 적용함으로써 MAC 결과에 대응하는 출력 액티베이션(output activation)을 생성할 수 있다.

[0066] 입력 액티베이션(401)은 복수의 입력 채널 맵들을 포함할 수 있다. 입력 액티베이션(401)은 예를 들어, 64개의 입력 채널 맵들을 가질 수 있다. 예를 들어, 입력 액티베이션(401)이 64개의 입력 채널 맵들을 포함하는 경우, SNN 연산 가속 장치의 각 시간 영역 CIM 뱅크는 64개의 행 라인들을 따라 배치된 비트셀들을 포함할 수 있다. 입력 액티베이션(401)에 포함된 복수의 입력 채널 맵들 각각은 서로 다른 복수의 입력 값들을 포함할 수 있다. 예를 들어, 서로 다른 복수의 입력 값들은, SNN 연산 가속 장치에 포함된 입력 라인들을 통해 TBA로 전달되는 입력 신호에 대응할 수 있다. 예를 들어, 서로 다른 복수의 입력 값들은 도 1의 해당 입력 신호(101_a)에 대응할 수 있다. 예를 들어, 각 입력 채널 맵은 복수의 입력 값들(예: IA_0 내지 IA_{63})을 포함할 수 있다.

[0067] 가중치 커널(402)은 입력 액티베이션(401)에 포함된 입력 채널 맵들의 수만큼의 채널 가중치 커널들을 포함할 수 있다. 예를 들어, 입력 액티베이션(401)에 포함된 입력 채널 맵들이 64개인 경우, 채널 가중치 커널들도 64개일 수 있다. 각 채널 가중치 커널은 복수의 4비트 가중치(예: W_0 내지 W_8)를 포함할 수 있다. 예를 들어, 채널 가중치 커널에 포함된 제1 가중치(404), 제2 가중치(405), 및 제3 가중치(406)는 4비트 가중치에 대응할 수 있다.

[0068] 가중치 커널(402)의 채널 가중치 커널은 3×3 크기를 가질 수 있다. 3×3 크기의 가중치 커널(402)의 제1 행(407)에 포함된 제1 가중치(404), 제2 가중치(405), 및 제3 가중치(406)는 하나의 시간 영역 CIM 뱅크에 저장될 수 있다. 예를 들어, 제1 가중치(404)는 시간 영역 CIM 뱅크 0의 TBA 0에 저장될 수 있다. 제2 가중치(405)는 시간 영역 CIM 뱅크 0의 TBA 1에 저장될 수 있다. 제3 가중치(406)는 시간 영역 CIM 뱅크 0의 TBA 2에 저장될 수 있다.

[0069] 일 실시예에 따른 SNN 연산 가속 장치가, 입력 액티베이션(401)에 가중치 커널(402)을 적용하는 경우, 입력 액티베이션(401)에 포함된 입력 값들과 가중치 커널(402)에 포함된 가중치들 간의 연산 결과는 MAC 결과에 대응하는 출력 액티베이션에 매핑될 수 있다. 예를 들어, 입력 액티베이션(401)에 포함된 제1 입력 값(409)과 가중치 커널(402)에 포함된 제1 가중치(404) 간의 연산 결과가 출력 값(451a)(예: OA_0)에 매핑될 수 있다. 예를 들어, 입력 액티베이션(401)에 포함된 제2 입력 값(410)과 가중치 커널(402)에 포함된 제2 가중치(405) 간의 연산 결과가 출력 값(452a)(예: OA_1)에 매핑될 수 있다.

[0070] 일 실시예에 따른 SNN 연산 가속 장치는 출력 액티베이션에 포함된 출력 값을 페이즈 코딩하여 APR 유닛에 입력할 수 있다. SNN 연산 가속 장치는 페이즈 코딩된 출력 값을 개별 시간 스텝 마다 순차적으로 APR 유닛에 입력할 수 있다. 예를 들어, SNN 연산 가속 장치는 제1 타임 스텝(460)에 출력 값(451a, 452a, 453a)을 병렬적으로 APR 유닛(470, 471, 472)에 입력할 수 있다. SNN 연산 가속 장치는 제1 타임 스텝(460) 후의 제2 타임 스텝(461)에, 제2 타임 스텝(461)에 출력 값(451b, 452b, 453b)을 병렬적으로 APR 유닛(470, 471, 472)에 입력할 수 있다. APR 유닛(470, 471, 472) 각각은 3×1 크기의 가중치 커널 연산을 처리할 수 있다. 일 실시예에 따른, SNN 연산 가속 장치는 APR 유닛(470, 471, 472)을 병렬로 동작함으로써 3×3 크기의 가중치 커널 연산을 처리할 수 있다. SNN 연산 가속 장치는 APR 유닛(470, 471, 472)을 병렬로 동작함으로써, 32 사이클 안에서 4개 출력 채널들에 대응하는 하나의 컨볼루션 출력을 반환할 수 있으며, 순차적 구현에 비해 처리량이 83% 향상될 수 있다. SNN 연산 가속 장치는 사이클 낭비 없이 페이즈 코딩에 따라 APR 유닛(470, 471, 472)을 파이프라이닝(pipelining)함으로써 1.69배 높은 에너지 효율을 달성할 수 있다.

[0072] 도 5는, 일 실시예에 따른 시간 영역 비트셀 어레이 구조를 도시한다.

[0073] 일 실시예에 따른 SNN 연산 가속 장치는, 시간 영역 비트셀 어레이(Time-domain bitcell array, TBA)(이하,

'TBA')(500)를 포함할 수 있다.

[0074] TBA(400)는 복수의 비트셀이 행들(rows) 및 열들(columns)로 배열되는 비트셀 어레이, 복수의 비트셀들 각각에 개별적으로 연결된 비트라인과 워드라인, 및 복수의 출력 라인을 포함할 수 있다. 예를 들어, TBA(500)은 64개의 행들 및 16개의 열들에 비트셀이 배열된 비트셀 어레이일 수 있다.

[0075] 비트셀은 입력 라인과 연결될 수 있다. TBA(500)의 동일한 행에 위치된 비트셀들 각각은 동일한 입력 라인과 연결될 수 있다. 비트셀들은 가중치를 저장할 수 있다. 예를 들어, TBA(500)의 하나의 행에 16개의 비트셀이 배열된 경우, TBA(500)의 하나의 행은 4비트 가중치 4개를 저장할 수 있다. SNN 연산 가속 장치는 입력 라인을 통해 수신된 입력 신호와 비트셀들에 저장된 가중치에 기초하여 비트셀들을 활성화할 수 있다. 비트셀의 구조 및 활성화 방법에 대하여 이하 도 6을 참조한다.

[0076] 비트셀들은 복수의 출력 라인들을 통해, CRG(Computation Reference Generator)(미도시)와 연결될 수 있다. 복수의 출력 라인들 각각은 비트셀 어레이의 열들 중 해당하는 열에 배열된 비트셀들과 연결될 수 있다. 이하에서는 하나의 열에 배열된 비트셀들(501)과 해당 열의 비트셀들(501)과 연결된 하나의 출력 라인(502)을 기준으로 설명한다. SNN 연산 가속 장치는 CRG(미도시)에서 생성된 기준 신호가 출력 라인(502)을 통과하는 경우 출력 라인(502)과 연결된 비트셀들(501)의 활성화 여부에 따라 기준 신호를 지연할 수 있다. SNN 연산 가속 장치는 지연된 신호와 기준 신호 간의 시간 차이에 기초하여 누적된 스파이크 수를 결정할 수 있다. 각각의 비트셀들(501)은 기준 신호에 대한 동일한 지연 셀로 동작할 수 있고, 출력 라인(502)으로부터 출력되는 펄스 신호(예: 스파이크)의 폭은 출력 라인(502)과 연결된 비트셀들(501) 중 활성화된 비트셀의 수에 비례할 수 있다. 구체적인 비트셀의 구조 및 비트셀에 의한 기준 신호의 지연에 대하여 도 6 및 도 7에서 설명한다.

[0078] 도 6은 일 실시예에 따른 기준 신호를 지연할 수 있는 비트셀의 구조를 도시한다.

[0079] 일 실시예에 따른 SNN 연산 가속 장치는 비트셀(600)을 포함할 수 있다. 일 실시예에 따른 비트셀(600)은 서로 반대되는 방향으로 연결된 한 쌍의 인버터(inverter)들을 포함하고, 설정된 가중치를 가지는 정적 랜덤 액세스 메모리(static random access memory)(603)를 포함할 수 있다. 정적 랜덤 액세스 메모리(603)에 포함된, 한 쌍의 인버터들의 제1 단(610)은 비트라인(Bit Line, BL)(601)과 트랜지스터를 통해 연결될 수 있고, 제2 단(611)은 비트라인(602)과 트랜지스터를 통해 연결될 수 있다. 일 실시예에 따른 SNN 연산 가속 장치는 비트라인(601)과 비트라인(602)에 서로 반대되는 데이터를 인가할 수 있다. 예를 들어, 비트라인(601)에 1에 해당하는 데이터가 인가된 경우, 비트라인(602)에는 0에 해당하는 데이터가 인가될 수 있다. 비트라인(601)에 1에 해당하는 데이터가 인가된 경우, 제1단(610)에는 1에 해당하는 데이터가 저장되고, 동시에 제2단(611)에는 0에 해당하는 데이터가 저장될 수 있다. 제1단(610) 및 제2단(611)에 저장된 데이터는 정적 랜덤 액세스 메모리(603)에 저장된 가중치에 해당할 수 있다. 예를 들어, 제1단(610)에 1, 제2단(611)에 0이 저장된 경우, 정적 랜덤 액세스 메모리(603)에 저장된 가중치는 [1, 0]에 해당할 수 있다. 다른 예를 들어, 비트라인(601)에 0에 해당하는 데이터가 인가되는 경우, 비트라인(602)에는 1에 해당하는 데이터가 인가될 수 있다. 비트라인(601) 및 비트라인(602)에 각각 0에 해당하는 데이터가 인가되고, 1에 해당하는 데이터가 인가된 경우, 정적 랜덤 액세스 메모리(603)에 저장된 가중치는 [0, 1]에 해당할 수 있다.

[0080] 일 실시예에 따른 비트셀(600)은, 한 쌍의 인버터들의 제1 단(610)과 게이트(gate)를 통해 연결된 제1 트랜지스터(620)를 포함할 수 있다. 제1 트랜지스터(620)는 일단에서 입력 신호(630)를 수신하고, 타단에서 제2 트랜지스터(621)와 연결될 수 있다. 예를 들어, 제1 트랜지스터(620)는 PMOS(p-channel metal oxide semiconductor)에 해당할 수 있다. 비트셀(600)은, 한 쌍의 인버터들의 제2 단(611)과 게이트를 통해 연결되고, 일단에서 입력 신호(631)를 수신 가능한 제2 트랜지스터(621)를 포함할 수 있다. 예를 들어, 제2 트랜지스터(621)는 PMOS에 해당할 수 있다. 제1 트랜지스터(620) 및 제2 트랜지스터(621)의 동작 여부는 정적 랜덤 액세스 메모리(603)에 저장된 가중치에 의해 결정될 수 있다. 제1 트랜지스터(620) 및 제2 트랜지스터(621)의 동작 여부에 대하여 이하 표 1 내지 표 3에서 설명한다.

[0081] 일 실시예에 따른 비트셀(600)은 제1 트랜지스터(620) 및 제2 트랜지스터(621)와 게이트를 통해 연결되고, 지연 커패시터(delay capacitor)(640) 및 출력 라인(650)사이에 연결되는 지연 트랜지스터(660)를 가질 수 있다. 지연 트랜지스터(660)는 NMOS(n-channel metal oxide semiconductor)에 해당할 수 있다. 지연 트랜지스터(660)의 게이트(gate)단은 제1 트랜지스터(620) 및 제2 트랜지스터(621) 사이의 연결 노드에 연결될 수 있다. 지연 트랜지스터(660)의 동작 여부는 제1 트랜지스터(620) 및 제2 트랜지스터(621)의 작동 여부 및 입력 신호(630,

631)의 값에 기초하여 지연 트랜지스터(660)의 게이트단에 전달되는 값에 의해 결정될 수 있다. 예를 들어, 정적 랜덤 액세스 메모리(603)에 저장된 가중치가 [1, 0]인 경우, 제1단(610)에 연결된 제1 트랜지스터(620)는 입력 신호(630)가 지연 트랜지스터(660)의 게이트로 인가되는 것을 차단하고, 제2 단(611)에 연결된 제2 트랜지스터(621)는 입력 신호(631)가 지연 트랜지스터(660)의 게이트로 인가되도록 할 수 있다. 지연 트랜지스터(660)의 게이트로 인가되는 입력 신호(631)가 스파이크(예를 들어 1)인 경우, 지연 트랜지스터(660)는 지연 커패시터(640)와 출력 라인(650)을 연결할 수 있다. 지연 트랜지스터(660)의 동작에 의한 지연 커패시터(640)와 출력 라인(650)의 연결 여부는 이하 표 1 내지 표 3에서 구체적으로 설명한다.

[0082] 일 실시예에 따른 비트셀(600)은 9T1C의 구조일 수 있다. 예를 들어, 비트셀(600)은 한 쌍의 인버터의 제1단(610) 및 제2단(611)에 개별적으로 연결된 제1 트랜지스터(620) 및 제2 트랜지스터(621)를 갖는 6T 구조의 정적 랜덤 액세스 메모리(603)와 지연 커패시터(630) 및 출력 라인(650)과 연결된 NMOS 구조의 지연 커패시터(660)를 포함할 수 있다.

[0083] 일 실시예에 따른 비트셀(600)은, 입력 신호(630, 631)의 값 및 가중치에 기초하여 출력 라인(650)의 기준 신호를 지연시킬 수 있다. 위에서 설명한 바와 같이, 정적 랜덤 액세스 메모리(603)에 저장된 가중치는 비트라인(601, 602)에 인가된 데이터에 의해 결정될 수 있다. 저장된 가중치에 따라 제1 트랜지스터(620) 및 제2 트랜지스터(621)의 작동 여부가 결정될 수 있다. 예를 들어, 제1 트랜지스터(620)가 작동되는 경우, 제1 트랜지스터(620)는 입력 신호(630)를 지연 트랜지스터(660)의 게이트로 전달하여 지연 트랜지스터(660)의 작동 여부를 결정할 수 있다. 다른 예를 들어, 제2 트랜지스터(621)가 작동되는 경우, 제2 트랜지스터(621)는 입력 신호(631)를 지연 트랜지스터(660)의 게이트로 전달하여 지연 트랜지스터(660)의 작동 여부를 결정할 수 있다. 예를 들어, 정적 랜덤 액세스 메모리(603)에 저장된 가중치가 [1, 0]에 해당하면, 제2 트랜지스터(631)가 작동되고, 스파이크(예: 1에 해당하는 비트 값)에 해당하는 입력 신호(631)가 제2 트랜지스터(631)를 통해 지연 트랜지스터(660)의 게이트로 전달될 수 있다. 지연 트랜지스터(660)의 게이트로 스파이크에 해당하는 입력 신호(631)가 전달되면, 출력 라인(650)과 지연 커패시터(640)가 연결될 수 있다. 출력 라인(650)을 따라 흐르는 기준 신호는, 출력 라인(650)과 연결된 지연 커패시터(640)를 충전함으로써 기준 신호를 지연시킬 수 있다.

[0085] 아래의 표 1은, 입력 신호의 값과 가중치에 따른 기준 신호의 지연 여부를 나타낸다.

표 1

[0086]

피연산자 테이블(operand table)		
입력 신호 / 가중치	LOW (Non-Spike)	HIGH (Spike)
LOW (제1단=L, 제2단=H)	LOW	LOW
HIGH (제1단=H, 제2단=L)	LOW	**Delay**

[0087] 표 1에 따르면, 일 실시예에 따른 비트셀(600)은 입력 신호(630, 631)의 값과 가중치에 기초하여 기준 신호를 지연할 수 있다. 비트셀(600)에 포함된 제2 트랜지스터(621)는 정적 랜덤 액세스 메모리(603)에 설정된 가중치의 값이 제1 논리 값(예: Q=H)을 가지는 것에 응답하여, 입력 신호(631)를 지연 트랜지스터(660)의 게이트로 전달할 수 있다. 예를 들어, 정적 랜덤 액세스 메모리(603)에 저장된 가중치의 제1 논리 값은 제1단(610)에 1에 해당하는 데이터 또는 H(high)에 해당하는 데이터가 저장되고, 제2 단(611)에 0에 해당하는 데이터 또는 L(low)에 해당하는 데이터가 저장된 경우를 나타낼 수 있다. 가중치의 값이 제1 논리 값을 나타내는 경우, PMOS에 해당하는 제1 트랜지스터(620)는 꺼지고(off), 제2 트랜지스터(621)는 켜(on)질 수 있다. 제2 트랜지스터(621)가 켜지면, 제2 트랜지스터를 통해 입력 신호(631)가 지연 트랜지스터(660)의 게이트로 전달될 수 있다.

[0088] 예를 들어, 제2 트랜지스터(621)를 통해 지연 트랜지스터(660)의 게이트로 전달된 입력 신호(631)가 스파이크(spike) 신호에 해당하는 경우(예: 1에 해당하는 비트 값), NMOS에 해당하는 지연 트랜지스터(660)는 켜질(ON) 수 있다. 지연 트랜지스터(660)는, 입력 신호(631)의 값이 제1 논리 값(예: 1에 해당하는 비트 값, 또는 High에 해당하는 스파이크 신호)을 가지는 것에 응답하여, 출력 라인(650)을 따르는 기준 신호가 지연되도록 지연 커패시터(640)를 출력 라인(650)에 연결할 수 있다.

[0089] 다른 예를 들어, 제2 트랜지스터(621)는 설정된 가중치의 값이 제2 논리 값을 가지는 것에 응답하여, 지연 커패

시터(640)에 의한 지연이 방지되도록 지연 트랜지스터(660)의 게이트를 향하는 입력 신호(621)를 차단(block)할 수 있다. 예를 들어, 정적 랜덤 액세스 메모리(603)에 저장된 가중치의 제2 논리 값은, 제1단(610)에 0에 해당하는 데이터 또는 L(low)에 해당하는 데이터가 저장되고, 제2단(611)에 1에 해당하는 데이터 또는 H(high)에 해당하는 데이터가 저장된 경우를 나타낼 수 있다. 가중치의 값이 제2 논리 값을 나타내는 경우, PMOS에 해당하는 제1 트랜지스터(620)는 켜지고, 제2 트랜지스터(621)는 꺼질 수 있다. 제2 트랜지스터(621)가 꺼지면, 입력 신호(631)가 지연 트랜지스터(660)의 게이트로 전달될 수 없으므로, 지연 트랜지스터(660)이 켜질 수 없고, 지연 커패시터(640)와 출력 라인(650)의 연결이 방지될 수 있다.

[0090] 다른 예를 들어, 지연 트랜지스터(660)는 입력 신호(631)의 값이 제2 논리 값을 가지는 것에 응답하여, 출력 라인(650)을 지연 커패시터(640)로부터 분리할 수 있다. 예를 들어, 입력 신호(631)의 제2 논리 값은, 입력 신호(631)에 인가되는 신호가 0에 해당하는 데이터 또는 스파이크 신호가 없는 상태에 대응하는 L(low)에 해당하는 데이터일 수 있다. 제2 논리 값의 입력 신호(631)가 제2 트랜지스터(621)를 통해 지연 트랜지스터(660)의 게이트로 전달될 경우, NMOS에 해당하는 지연 트랜지스터(660)는 꺼질(off)수 있다. 지연 트랜지스터(660)가 꺼지는 경우(동작하지 않는 경우), 지연 커패시터(640)와 출력 라인(650) 사이의 연결이 방지될 수 있다.

[0092] 아래의 표 2는 입력 신호에 의해 결정되는 비트셀(600)의 동작 상태를 나타낸다.

표 2

[0093]

입력 상태 드라이버 논리(input state driver logic)			
입력 신호 / 출력 신호	LOW (Non-Spike)	HIGH (Spike In)	Pre-Charge
$V_{IN,S}$	GND	V_{DD}	V_{DD}
$V_{IN,C}$	GND	GND	V_{DD}

[0094] 표 2에 따르면, 일 실시예에 따른 비트셀(600)의 동작 상태는 프리 차지(pre-charge) 단계와 연산 단계로 구별될 수 있다. 연산 단계는, 입력 신호가 스파이크 신호에 해당하는 경우와 스파이크 신호에 해당하지 않는 경우로 구별될 수 있다.

[0095] 예를 들어, 프리 차지 단계는, 입력 신호(630, 631)가 모두 공급 전압(예: V_{DD})에 대응하는 신호인 경우에 대응할 수 있다.

[0096] 예를 들어, 연산 단계 중 입력 신호가 스파이크 신호에 해당하는 경우는, 입력 신호(631)가 공급 전압에 대응하는 신호이고, 입력 신호(630)가 접지 전압(ground voltage)에 대응하는 신호인 경우에 대응할 수 있다.

[0097] 예를 들어, 연산 단계 중 입력 신호가 스파이크 신호에 해당하진 않는 경우는, 입력 신호(630, 631)가 모두 접지 전압에 대응하는 신호인 경우에 대응할 수 있다.

[0099] 표 3은 표 2에 따른 비트셀의 동작 상태에 따른, 지연 트랜지스터(660)의 게이트단의 전압 및 출력 라인(650)을 따라 흐르는 기준 신호의 지연 여부를 나타낸다.

표 3

[0100]

출력 상태 논리(output state logic)			
입력 신호 / 출력 신호	LOW (Non-Spike)	HIGH (Spike In)	Pre-Charge
지연 트랜지스터의 게이트단	GND	V_{DD}	V_{DD}
출력 라인	N/C (Not Con.)	**Delay**	GND

[0101] 프리 차지 단계에서, 비트셀(600)은, 제1 트랜지스터(620)의 일단 및 제2 트랜지스터(621)의 일단에 공급 전압(예: V_{DD})이 인가되는 것에 응답하여, 지연 커패시터(640)의 전압을 접지 전압(ground voltage)으로 리셋

(reset)할 수 있다. 예를 들어, 입력 신호(630, 531)가 모두 공급 전압(예: V_{DD})인 경우, 정적 랜덤 액세스 메모리(603)에 저장된 가중치와 무관하게, 지연 트랜지스터(660)의 게이트에 공급 전압이 인가될 수 있다. NMOS에 해당하는 지연 트랜지스터(660)는 게이트에 인가된 공급 전압에 의해 지연 커패시터(640)와 출력 라인(650)을 연결할 수 있다. 프리 차지 단계에서, 지연 트랜지스터(660)의 일단과 연결된 출력 라인(650) 노드는 접지 전압을 가지므로, 지연 커패시터(640)의 전압을 접지 전압으로 리셋(reset)할 수 있다.

[0102] 위의 표 1 및 표 2를 참조하여 설명된 바와 같이, 정적 랜덤 액세스 메모리(603)에 저장된 가중치가 제1 논리 값(예: 제1단(610)에 1에 해당하는 데이터, 제2단(611)에 0에 해당하는 데이터가 저장된 경우)에 대응하고, 연산 단계 중 입력 신호가 스파이크 신호(예: 입력 신호(631)가 공급 전압이고, 입력 신호(630)가 접지 전압에 해당하는 경우)에 해당하는 경우, 지연 트랜지스터의 게이트단에 공급 전압이 인가될 수 있다. 지연 트랜지스터의 게이트 단에 공급 전압이 인가되면, 지연 커패시터(640)와 출력 라인(650)이 지연 트랜지스터(660)를 통해 연결될 수 있고, 출력 라인(650)을 따라 흐르는 기준 신호가 지연될 수 있다.

[0104] 도 7은 TBA에 포함된 하나의 출력 라인을 따라 흐르는 기준 신호가 비트셀들에 의해 지연되는 것 및 기준 신호의 지연이 누적된 MAC 펄스를 형성하는 것을 나타낸다.

[0105] 일 실시예에 따른 SNN 연산 가속 장치는 복수의 비트셀들이 행들 및 열들로 배열된 비트셀 어레이 및 복수의 출력 라인들을 포함할 수 있다. SNN 연산 가속 장치에 포함된 복수의 출력 라인들 각각은 어레이의 열들 중 해당하는 열에 배열된 비트셀들과 연결될 수 있다. 이하에서는, 해당하는 열에 배열된 비트셀들과 연결된 하나의 출력 라인(700)을 기준으로 설명한다.

[0106] 출력 라인(700)을 따라 흐르는 원본 기준 신호(701)는 출력 라인(700)과 연결된 비트셀들에 의해 지연될 수 있다. 복수의 비트셀들에 의해 지연된 기준 신호(702)는, 입력 신호들의 값들 및 가중치들 간의 MAC(multiply-accumulate computation) 결과에 대응하는 지연 시간만큼, 원본 기준 신호 대비 지연된 신호를 나타낼 수 있다.

[0107] 표 1 내지 표 3에서 설명한 바와 같이, 비트셀에 저장된 가중치가 제1 논리 값(예: 가중치가 [1, 0])에 해당하고, 연산 단계 중 입력 신호가 스파이크 신호인 경우에, 비트셀에 의해 출력 라인(700)을 따라 흐르는 원본 기준 신호(701)가 지연될 수 있다. 원본 기준 신호(701)를 지연시키는 비트셀을 활성화 상태의 비트셀이라 지시하는 경우, 출력 라인(700)에 연결된 비트셀 중 활성화 상태의 비트셀 수만큼 원본 기준 신호(701)가 지연될 수 있다. 예를 들어, 출력 라인(700)과 연결된 제1 비트셀이 활성화된 경우, 제1 비트셀에 의해, 원본 기준 신호(701)가 지연된 제1 지연 신호(710)가 생성될 수 있다. 제1 지연 신호(710)는 출력 라인(700)을 따라 흐를 수 있다. 제1 지연 신호(710)는 활성화된 제2 비트셀에 의해, 지연될 수 있다. 제1 지연 신호(710)가 지연되어 생성된 제2 지연 신호(711)는 원본 기준 신호(701)에 비해 제1 비트셀 및 제2 비트셀에 의한 지연이 누적된 신호 일 수 있다. 제2 지연 신호(711)는 출력 라인을 따라 흐를 수 있고, 활성화되지 않은 제3 비트셀은 제2 지연 신호(711)를 지연시키지 않을 수 있다. 제2 지연 신호(711)가 지연되지 않은 신호(712)는 출력 라인(700)을 따라 흐를 수 있다. 원본 기준 신호(710)가 출력 라인(700)을 따라 흐를 때, 출력 라인(700)과 연결된 비트셀들의 활성화 여부에 따라 원본 기준 신호(710)의 지연을 축적할 수 있다. 다시 말해, 출력 라인(700)과 연결된 비트셀들 중 활성화된 비트셀들의 수에 따라, 원본 기준 신호(701)에서 활성화된 비트셀의 수만큼의 지연이 누적된 최종 지연된 기준 신호(702)가 생성될 수 있다. 최종 지연된 기준 신호(702)는 원본 기준 신호(701)와 신호의 폭은 동일하나, 신호의 상승 에지(edge) 시점이, 활성화된 비트셀들의 수에 대응하여 지연된 신호일 수 있다.

[0108] 일 실시예에 따른 SNN 연산 가속 장치는, 원본 기준 신호(701) 및 지연된 기준 신호(702)에 기초하여, MAC 결과에 대응하는 지연 시간을 지시하는 출력 신호(703)를 생성할 수 있다. 예를 들어, SNN 연산 가속 장치는, 원본 기준 신호(701) 및 최종 지연된 기준 신호(702) 간에 XOR 연산을 적용함으로써, 출력 신호(703)를 생성할 수 있다. 출력 신호(703)는 원본 기준 신호(701)의 상승 에지 시점과 동일한 시점을 상승 에지로 하고, 최종 지연된 기준 신호(702)의 상승 에지 시점과 동일한 시점을 하강 에지로 하는 펄스 신호(pulse signal) 일 수 있다. 다시 말해, 출력 신호(703)는 비트셀에 의한 지연이 누적된 펄스 신호일 수 있다.

[0109] 일 실시예에 따른 SNN 연산 가속 장치는, 출력 라인(700)을 통해 다른 원본 기준 신호를 인가하기 전에, 도 5의 표 2 및 표 3에서 설명한 비트셀들의 프리 차지(pre-charge) 단계(750)를 동작할 수 있다. SNN 연산 가속 장치는 프리 차지 단계(750)에서 비트셀들의 지연 커패시터들을 리셋(reset)할 수 있다.

[0111] 도 8은, 출력 라인 별 출력 신호의 비트 정확도를 고려하기 위한 시간 MAC 회로의 구성을 나타낸다.

[0112] SNN 연산 가속 장치의 각 출력 라인 별 출력 신호는, 출력 라인에 연결된 비트셀들 중 활성화된 비트셀의 수에 비례하는 지연을 포함할 수 있다. 예를 들어, 제1 출력 라인에 연결된 비트셀들 중 활성화된 비트셀의 수가 10개이고, 제2 출력 라인에 연결된 비트셀들 중 활성화된 비트셀의 수가 10개인 경우, 제1 출력 라인을 통해 생성된 출력 신호는 제2 출력 라인을 통해 생성된 출력 신호와 폭이 같고, 상승 에지 및 하강 에지의 시점이 동일한 펄스 신호 (예: 스파이크)일 수 있다. 다시 말해, 출력 신호는 비트셀들에 의해 축적된 지연을 나타낼 수 있지만, 출력 라인 별 4비트 가중치의 정확도(weight precision)를 포함하지 않을 수 있다. 비교 실시예에 따르면, 시프터(shifter)와 덧셈기(adder)를 이용하여, 출력 신호들을 멀티 비트로 재구성하기 위해 여러 번의 반복을 수행해야 한다. 일 실시예에 따른, SNN 연산 가속 장치에 포함된 시간 MAC 회로는 한 사이클의 아날로그 영역에서 4비트 가중치를 재구성할 수 있다.

[0113] 일 실시예에 따른 SNN 연산 가속 장치는, 복수의 비트 자리들(bit positions)에 개별적으로 대응하는 복수의 시간 MAC(Multiply-and-Accumulate) 회로들(800)을 더 포함할 수 있다. 복수의 시간 MAC 회로들(800)은 MAC 결과에 대응하는 지연 시간을 지시하는 출력 신호들(810)을 TBA로부터 수신할 수 있다. 예를 들어, TBA가 4비트 가중치 4개를 포함하기 위해, TBA의 한 행에 총 16개의 비트셀을 포함하는 경우, TBA에 포함된 출력 라인들(820)은 총 16개일 수 있다. 예를 들어, 출력 라인들(820)은 TBA의 좌측 열에서부터 우측 열까지 출력 라인 0 내지 출력 라인 15로 구별될 수 있다. TBA에 포함된 16개의 출력 라인들(820)은 4개씩 시간 MAC 회로들(800)에 연결될 수 있다. 예를 들어, 출력 라인들(820)은 0 내지 3, 4 내지 7, 8 내지 11, 및 12 내지 15로 4개씩 시간 MAC 회로들(800)과 연결될 수 있다.

[0114] 일 실시예에 따른, 복수의 시간 MAC 회로들(800)의 각각은, MAC 결과에 대응하는 지연 시간을 지시하는 출력 신호들(810)을 전하로 변환하는 시간 대 전하 변환기(Time-to-Charge Converter)(830) 및 변환된 전하에서 해당하는 비트 자리에 따른 전하량을 보존하는 비트 전하 보존 회로(840)를 포함할 수 있다.

[0115] 시간 대 전하 변환기(830)는, MAC 결과에 대응하는 지연 시간을 지시하는 출력 신호들(810)을 전하로 변환할 수 있다. 예를 들어, 스파이크 신호에 해당하는 출력 신호들(810)은 시간 대 전하 변환기(830)에 포함된 트랜지스터들의 동작 여부를 결정함으로써, 출력 신호들(810)을 전하로 변환할 수 있다.

[0116] 비트 전하 보전 회로(840)는 제1 커패시턴스를 갖는 제1 커패시터(850), 제2 커패시턴스를 갖는 제2 커패시터(860), 및 제1 커패시터(850) 및 제2 커패시터(860) 간에 배치되는 스위치(870)를 포함할 수 있다. 비트 전하 보전 회로(840)에 포함된 제2 커패시터(860)의 크기에 기초하여, 출력 신호(810)의 해당하는 비트 자리를 지시할 수 있다. 예를 들어, 제2 커패시터(860)의 커패시턴스가 8C인 경우 가중치의 비트 자리 중 MSB를 지시할 수 있다. 제2 커패시터(860)의 커패시턴스가 1C인 경우 가중치의 비트 자리 중 LSB를 지시할 수 있다.

[0117] 제1 커패시터(850)의 제1 커패시턴스와 제2 커패시터(860)의 제2 커패시턴스의 합은 미리 결정된 값에 대응할 수 있다. 예를 들어, 제1 시간 MAC 회로(801)는 1C의 커패시턴스를 갖는 제1 커패시터(850) 및 8C의 커패시턴스를 갖는 제2 커패시터(860)를 포함할 수 있다. 제2 시간 MAC 회로는 5C의 커패시턴스를 갖는 제1 커패시터 및 4C의 커패시턴스를 갖는 제2 커패시터를 포함할 수 있다. 제3 시간 MAC 회로는 7C의 커패시턴스를 갖는 제1 커패시터 및 2C의 커패시턴스를 갖는 제2 커패시터를 포함할 수 있다. 제4 시간 MAC 회로(802)는 8C의 커패시턴스를 갖는 제1 커패시터 및 1C의 커패시턴스를 갖는 제2 커패시터를 포함할 수 있다. 제1 시간 MAC 회로(801) 내지 제4 시간 MAC 회로(802)에 각각 포함된 제1 커패시터의 커패시턴스 및 제2 커패시터의 커패시턴스의 합은 9C에 해당할 수 있다.

[0118] SNN 연산 가속 장치는 복수의 시간 MAC 회로들(800)과 연결된 합산 커패시터(미도시)를 더 포함할 수 있다. 예를 들어, 합산 커패시터는 제1 시간 MAC 회로(801) 내지 제4 시간 MAC 회로(802)와 연결될 수 있다. SNN 연산 가속 장치는, 시간 MAC 회로들(800) 각각에 포함된 제2 커패시터(860)에 저장된 전하량을 합산 커패시터를 통해 합산할 수 있다.

[0120] 도 9는 시간 MAC 회로의 순차적인 동작을 설명하는 도면이다.

[0121] 일 실시예에 따른 SNN 연산 가속 장치는 시간 MAC 회로(901, 902, 903, 904)에 조절 신호(911)를 인가하고, 시간 MAC 회로(901, 902, 903, 904)에 포함된 스위치(920, 921)의 동작을 조절함으로써, 시간 MAC 회로(901,

902, 903, 904)의 동작을 조절할 수 있다.

[0122] 시간 MAC 회로(901)에 로우(low) 상태의 출력 신호(910)가 인가될 수 있다. SNN 연산 가속 장치는 시간 MAC 회로(901)의 스위치(920)의 연결을 끊을 수 있고, 시간 MAC 회로(901)에 로우(low) 상태의 조절 신호(911)를 인가할 수 있다. 비트 전하 보존 회로(930)는 제1 커패시터(940) 및 제2 커패시터(950)를 공급 전원(예: V_{DD})으로 충전할 수 있다.

[0123] 시간 MAC 회로(902)에 지연 시간을 지시하는 출력 신호(910)가 공급되는 동안, 비트 전하 보존 회로(930)는 제1 커패시터(940)와 제2 커패시터(941)에 전하를 충전할 수 있다. 예를 들어, 시간 MAC 회로(902)에 하이(high) 상태의 출력 신호(910)가 인가될 수 있다. SNN 연산 가속 장치는 하이(high) 상태의 출력 신호(910)가 시간 MAC 회로(902)에 인가되는 동안, 스위치(920)를 연결할 수 있다. 스위치(920)가 연결되는 경우, 시간 MAC 회로(902)에 싱킹 패스(sinking path)(950)가 형성될 수 있다. 시간 MAC 회로(902)에 형성된 싱킹 패스(950)로 인하여, 비트 전하 보존 회로(930)에 인가되는 전압이 공급 전원 보다 작은 값으로 감소할 수 있다.

[0124] 시간 MAC 회로(903)에 지연 시간을 지시하는 출력 신호(910)의 인가가 종료하면, SNN 연산 가속 장치는 스위치(920)의 연결을 끊고, 시간 MAC 회로(903)에 하이(high) 상태의 조절 신호(911)를 인가할 수 있다. 시간 MAC 회로(903)의 스위치(920) 연결이 끊기는 것과 인가된 하이(high) 상태의 조절 신호(911)에 기초하여, 비트 전하 보존 회로(930)에 인가되는 전압이 유지될 수 있다.

[0125] 시간 MAC 회로(904)에 지연 시간을 지시하는 출력 신호(910)가 종료하는 것에 응답하여, 비트 전하 보존 회로(930)는 스위치(921)를 이용하여 제1 커패시터(940)로부터 제2 커패시터(941)를 분리(disconnect)함으로써 제2 커패시터(941)에서 해당하는 비트 자리에 따른 전하량을 저장할 수 있다. 비트 전하 보존 회로(930)는 스위치(921)의 연결이 종결되는 것에 응답하여, 제1 커패시터(940) 및 제2 커패시터(941) 간의 전하 공유(charge sharing)를 종료할 수 있다.

[0126] 일 실시예에 따른 SNN 연산 가속 장치는 시간 MAC 회로(901 내지 904)의 동작을 반복함으로써 멀티 비트 입력에 대한 가중치의 비트 자리를 고려할 수 있다.

[0128] 도 10은 APR 유닛의 구성을 개략적으로 도시한다.

[0129] 일 실시예에 따른 SNN 연산 가속 장치는 APR(analog precision reconstruction) 유닛(1000)을 더 포함할 수 있다. APR 유닛(1000)은, 시간 영역 CIM 뱅크에 포함된 시간 MAC 회로로부터 가중치에 대응하는 전하량을 수신할 수 있다. 예를 들어, APR 유닛(1000)은 64개의 행들(rows) 및 16개의 열들(columns)에 배열된 비트셀을 포함하는 TBA로부터 생성된 시간 지연을 지시하는 출력 신호에 기초하여, 시간 MAC 회로를 통해 4개의 4비트 가중치에 대응하는 전하량을 수신할 수 있다. APR 유닛(1000)은 서로 다른 시간 영역 CIM 뱅크에 포함된 시간 MC 회로로부터 가중치에 대응하는 전하량을 병렬적으로 수신할 수 있다.

[0130] APR 유닛(1000)은 복수의 시간 스텝들 별로, 복수의 가중치들 및 해당하는 입력 비트 자리의 입력 값들에 기초한 MAC 결과들의 합산에 대응하는 전하량에 기초하는 전하를 각각 저장하는 복수의 차지 홀더들(charge holders)(1001, 1002, 1003, 1004)을 포함할 수 있다. 예를 들어, 복수의 차지 홀더들(1001, 1002, 1003, 1004)은, 복수의 시간 스텝들에 대응하고 MAC 결과들의 합산에 해당하는 전하를 개별적으로 저장할 수 있다. 예를 들어, 차지 홀더(1001)는 MAC 결과들의 합산에 해당하는 전하 중 제1 시간 스텝에 대응하는 전하를 저장할 수 있다. 차지 홀더(1002)는 MAC 결과들의 합산에 해당하는 전하 중 제2 시간 스텝에 대응하는 전하를 저장할 수 있다. 차지 홀더(1003)는 MAC 결과들의 합산에 해당하는 전하 중 제3 시간 스텝에 대응하는 전하를 저장할 수 있다. 차지 홀더(1004)는 MAC 결과들의 합산에 해당하는 전하 중 제3 시간 스텝에 대응하는 전하를 저장할 수 있다.

[0131] APR 유닛(1000)은 복수의 차지 홀더들(1001, 1002, 1003, 1004)에 저장된 전하들로부터 복원 커패시터(1011)로의 전하 공유를 복수의 시간 스텝들 별로 순차적으로 수행함으로써 커널 중 한 커널 원소에 대응하는 최종 MAC 결과를 생성하는 복원 메모리 회로(1010)를 포함할 수 있다. 복원 메모리 회로(1010)는 매 클럭(clock) 마다 차지 홀더들(1001, 1002, 1003, 1004) 각각과 순차적으로 전하 공유를 통해 4개의 시간 스텝에서 전하를 축적할 수 잇다. 예를 들어, 복원 메모리 회로(1010)는 제1 시간 스텝에서 차지 홀더(1001)와 전하 공유할 수 있다. 복원 메모리 회로(1010)는 제2 시간 스텝에서 차지 홀더(1001)와 전하 공유를 유지하면서 차지 홀더(1002)와 전하 공유할 수 있다. 복원 메모리 회로(1010)는 제3 시간 스텝에서, 차지 홀더(1001) 및 차지 홀더(1002)와 전

하 공유를 유지하면서, 차지 홀더(1003)와 전하 공유할 수 있다. 복원 메모리 회로(1010)는 제4 시간 스텝이서, 이전 시간 스텝에서 전하 공유한 차지 홀더(1001, 1002, 1003)과의 전하 공유를 유지하면서, 차지 홀더(1004)와 전하 공유할 수 있다.

[0132] 일 실시예에 따른 APR 유닛(1000)은 복수의 차지 홀더들(1001, 1002, 1003, 1004)에 포함된 커패시터(1012)의 커패시턴스 값과 복원 메모리 회로(1010)에 포함된 커패시터(1011)의 커패시턴스 값이 동일할 수 있다. 커패시터(1012)의 커패시턴스와 커패시터(1011)의 커패시턴스 값이 동일한 경우, 최종 MAC 결과(1020)는 아래 수학식 1에 의해 각 시간 스텝 별로 구별되는 전압값을 가질 수 있다.

수학식 1

[0133]
$$V_{Reconst.} = (\frac{V_0}{16} + \frac{V_1}{8} + \frac{V_2}{4} + \frac{V_3}{2})$$

[0134] 위의 수학식 1에서 $V_{RECONST}$는 복원 메모리 회로(1010)의 커패시터(1011)에 저장된 전압에 대응할 수 있다. V_0는 제1 시간 스텝에 차지 홀더(1001)에 저장된 전하에 의한 전압에 대응할 수 있다. V_1은 제2 시간 스텝에 차지 홀더(1002)에 저장된 전하에 의한 전압에 대응할 수 있다. V_2는 제3 시간 스텝에 차지 홀더(1003)에 저장된 전하에 의한 전압에 대응할 수 있다. V_3는 제4 시간 스텝에 차지 홀더(1004)에 저장된 전하에 의한 전압에 대응할 수 있다.

[0135] 일 실시예에 따른 SNN 연산 가속 장치는 차지 홀더들(1001, 1002, 1003, 1004)과

[0136] 복원 메모리 회로(1010)간의 전하 공유(charge sharing)에 기초하여, 페이즈 코딩된 입력 신호에 대응하는 전압 값을 획득할 수 있다. 예를 들어, 제1 시간 스텝에 LSB에 해당하는 전하에 의한 전압 V_0가 차지 홀더(1001)의 커패시터(1012)에 저장될 수 있다. V_0는 복원 메모리 회로(1010)의 커패시터(1011)와 전하 공유됨으로써 1/2배 되어 커패시터(1011)에 저장될 수 있다. 예를 들어, 제1 시간 스텝에 커패시터(1011)에 저장된 전하의 전압은 $V_0/2$ 일 수 있다. 그 후 차지 홀더(1001)에 저장된 전하가 리셋(reset)될 수 있다. 예를 들어, 차지 홀더(1001)에 저장된 전하는 리셋 신호(예: RST0)에 기초하여 리셋될 수 있다.

[0137] 제2 시간 스텝에 LSB의 다음 비트(bit) 자리에 해당하는 전하에 의한 V_1이 차지 홀더(1002)에 저장될 수 있다. V_1은 복원 메모리 회로(1010)의 커패시터(1011)와 전하 공유됨으로써 1/2배되어 커패시터(1011)에 저장될 수 있다. 동시에 제1 시간 스텝에 커패시터(1011)에 저장된 기존의 전하에 의한 전압 $V_0/2$가 복원 메모리 회로(1010) 및 차지 홀더(1002) 간의 전하 공유로 인해 1/2배 되어, $V_0/4$의 전압으로 커패시터(1011)에 저장될 수 있다. 예를 들어, 제2 시간 스텝에 커패시터(1011)에 저장된 전하에 의한 전압은 $V_0/4+ V_1/2$에 대응할 수 있다. 그 후 차지 홀더(1002)에 저장된 전하가 리셋될 수 있다.

[0138] 참고로, 제2 시간 스텝에 차지 홀더(1001)의 커패시터(1012)에 다른 데이터의 LSB에 해당하는 전하가 저장되고, 복원 메모리 회로(1010)과는 다른 복원 메모리 회로(예: 복원 메모리 회로 1 또는 복원 메모리 회로 2 중 적어도 하나)와 전하 공유를 함으로써, APR 유닛은 병렬적으로 동작할 수 있다.

[0139] 제3 시간 스텝에 MSB 바로 전 비트 자리에 해당하는 전하에 의한 전압 V_2가 차지 홀더(1003)에 저장될 수 있다. V_2는 복원 메모리 회로(1010)의 커패시터(1011)와 전하 공유 도힘으로써 1/2배되어 커패시터(1011)에 저장될 수 있다. 동시에 제2 시간 스텝에 커패시터(10110)에 저장된 기존의 전하에 의한 전압이 복원 메모리 회로(1010) 및 차지 홀더(1003) 간의 전하 공유로 인해 1/2배되어 커패시터(1011)에 저장될 수 있다. 예를 들어, 제3 시간 스텝에 커패시터(1011)에 저장된 전하에 의한 전압은, $V_0/8+ V_1/4+V_2/2$에 대응할 수 있다. 그 후 차지 홀더(1003)에 저장된 전하가 리셋될 수 있다.

[0140] 제4 시간 스텝에 MSB에 해당하는 전하에 의한 전압 V_3이 차지 홀더(1004)에 저장될 수 있다. V_3은 복원 메모리 회로(1010)의 커패시터(1011)와 전하 공유됨으로써 1/2배 되어 커패시터(1011)에 저장될 수 있다. 동시에 제3

시간 스텝에 커패시터(1011)에 저장된 기존의 전하에 의한 전압이, 복원 메모리 회로(1010) 및 차지 홀더(1004) 간의 전하 공유로 인해 1/2배 되어 커패시터(1011)에 저장될 수 있다. 예를 들어, 제4 시간 스텝에 커패시터(1011)에 저장된 전하에 의한 전압은, 수학식 1의 $V_{RECONST}$에 대응할 수 있다.

[0141] 차지 홀더들(1001, 1002, 1003, 1004) 각각과 복원 메모리 회로(1010) 간의 전하 공유 횟수(예: 1/2배된 횟수)에 따라, APR 유닛(1000)은 수신된 가중치에 대응하는 전하량에 기초하여, 페이즈 코딩된 입력의 복수의 비트 자리들에 개별적으로 대응하는 복수의 시간 스텝의 각각마다 페이즈 코딩된 입력의 해당하는 비트 자리의 입력 값에 대한 최종 MAC 결과(1020)를 생성할 수 있다. 예를 들어, $V_0/16$에 해당하는 전압 값은 입력 신호의 LSB에 대응하는 최종 MAC 결과(1020)일 수 있다. 예를 들어, $V_3/2$에 해당하는 전압 값은 입력 신호의 MSB에 대응하는 최종 MAC 결과(1020)일 수 있다. APR 유닛(1000)은 복수의 복원 메모리 회로들을 포함할 수 있다. 예를 들어, APR 유닛(1000)은 3개의 복원 메모리 회로들을 포함할 수 있다. 3개의 복원 메모리 회로들은 모든 복원 메모리 회로가 서로 다른 뱅크로부터 4개의 MAC 결과에 대응하는 전하를 수신할 때까지 병렬적으로 동작할 수 있다.

[0143] 도 11은 APR 유닛의 출력 액티베이션 생성 타이밍도를 나타낸다.

[0144] 일 실시예에 따른 SNN 연산 가속 장치는 복수의 시간 영역 CIM 뱅크를 포함할 수 있다. SNN 연산 가속 장치는 복수의 시간 영역 CIM 뱅크를 병렬적으로 동작 시킬 수 있다.

[0145] 시간 영역 CIM 뱅크는 복수의 TBA를 포함할 수 있다. SNN 연산 가속 장치는 복수의 TBA를 병렬적으로 동작 시킬 수 있다. SNN 연산 가속 장치는 복수의 TBA 각각과 개별적으로 연결되는 시간 MAC 회로를 포함할 수 있다. APR 유닛(1100)은 시간 MAC 회로로부터 복수의 가중치 비트 정확도가 고려된 MAC 결과를 수신할 수 있다. APR 유닛(1100)은 복수의 복원 메모리 회로들(1101, 1102, 1103)을 포함할 수 있다. 예를 들어, APR 유닛(1100)은 3개의 복원 메모리 회로들(1101, 1102, 1103)을 포함할 수 있다. APR 유닛(1100)은 모든 복원 메모리 회로들(1101, 1102, 1103)이 서로 다른 시간 영역 CIM 뱅크로부터 4개의 MAC 결과를 수신할 때까지 복원 메모리 회로들(1101, 1102, 1103)을 병렬적으로 동작할 수 있다. 예를 들어, 복원 메모리 회로(1101)는 제1 시간 스텝(예: TS 0) 내지 제4 시간 스텝(예: TS 3) 동안, 제1 시간 영역 CIM 뱅크로부터 가중치 비트 정확도가 고려된 MAC 결과를 수신할 수 있다. 복원 메모리 회로(1102)는 제1 시간 스텝 내지 제4 시간 스텝 동안, 제2 시간 영역 CIM 뱅크로부터 가중치 비트 정확도가 고려된 MAC 결과를 수신할 수 있다. 복원 메모리 회로(1103)는 제1 시간 스텝 내지 제4 시간 스텝 동안, 제3 시간 영역 CIM 뱅크로부터 가중치 비트 정확도가 고려된 MAC 결과를 수신할 수 있다.

[0146] APR 유닛(1100)은 복원 메모리 회로(1101)에 수신된 MAC 결과에 기초하여 입력 신호의 비트 자리에 대응하는 복수의 시간 스텝들 각각에 개별적으로 대응하는 출력 액티베이션을 생성할 수 있다. 복원 메모리 회로들(1101, 1102, 1103)이 병렬적으로 동작됨으로써, 한 사이클(1120)에 복원 메모리 회로들(1101, 1102, 1103) 각각으로부터 출력 액티베이션이 생성될 수 있다. 예를 들어, APR 유닛(1100)은 서로 다른 3개의 시간 영역 CIM 뱅크로부터 수신된 MAC 결과에 기초하여 6사이클 동안 출력 액티베이션을 생성할 수 있다. APR 유닛(1100)에 포함된 복수의 차지 홀더들이 복원 메모리 회로들(1101, 1102, 1103) 각각에 대한 직렬 연결을 마친 후, OSG(1110)에 배치된 합산 커패시터와의 최종 전하 공유가 1사이클(1120) 수행될 수 있다. 합산 커패시터의 전압이 미리 결정된 임계값 전압을 넘는 경우, OSG(1110)은 출력 스파이크를 생성할 수 있다. 따라서, 4비트 가중치를 포함하는 3×3 커널의 출력은 매 8번째 클럭 사이클 마다 획득될 수 있다.

[0148] 도 12a는 출력 라인에 연결된 복수의 비트셀들로 인한 기준 신호의 딜레이에 기초하여 형성된 스파이크 신호의 누적 결과를 도시한다.

[0149] 그래프(1200a)에서는 출력 라인에서의 스파이크 신호의 수에 따른 시간 지연을 나타낸다. 하나의 비트셀은 평균(1201a), 출력 라인을 따라 흐르는 기준 신호를 0.1ns 만큼 지연시킬 수 있다. 출력 라인으로부터 출력 라인에 연결된 복수의 비트셀들로 인한 기준 신호의 지연에 기초하여 생성된 출력 신호의 지연 범위는, 각 출력 라인 별 스파이크들의 수(예: 출력 라인에 연결된 활성화된 비트셀의 수)에 따라 [1.53ns, 7.75ns]이다.

[0150] 도 12b는 시간 MAC 회로에서 MAC 값에 따른 전압을 나타낸다.

[0151] 그래프(1200b)에서는 시간 MAC 회로에서 MAC 값에 따른 MAC 결과의 전압 값을 나타낸다. MAC 결과의 전압은 MAC 값 마다 평균(1201b) 0.2mV씩 변할 수 있다. MAC 결과의 전압 값의 범위는 [271.63mV, 81.17mV]에 해당한다.

[0152] 도 12c는APR 유닛에서의 최종 MAC 결과에 따른, 최종 MAC 결과의 전압을 나타낸다.

[0153] 그래프(1200c)에서는 APR 유닛에서의 최종 MAC 결과가 0에서 14400의 범위로 증가할 때, 최종 MAC 결과의 전압이 130.58mV에서 47.13mV로 감소하는 것을 나타낸다. 그래프(1200c)에서는 최종 MAC 출력 당 전압 변동 평균(1201c)이 0.01mV 미만인 6.00 μV에 해당한다.

[0154] 도 12d는복수의 시간 스텝 동안 APR 유닛에서 가중치에 대응하는 MAC 결과를 최종 MAC 결과로 재구성하는 과정을 그래프를 통해 도시한다.

[0155] 그래프(1200d)에서는 4개의 시간 스텝 동안 APR 유닛에 입력된 가중치에 대응하는 MAC 결과가 최종 MAC 결과로 재구성함으로써, 시간에 따른 전압 곡선을 포함한다. 복수의 비트들을 갖는 가중치에 대응하는 MAC 결과의 LSB는 제1 시간 스텝(1201d)에서 시간 MAC 회로에서부터 APR 유닛으로 입력된다. MSB는 제4 시간 스텝(1204d)에서 시간 MAC 회로에서부터 APR 유닛으로 입력된다.

[0156] 예를 들어, 전압 곡선(1210d)은 시간 MAC 회로에서 MAC 결과의 LSB가 제1 시간 스텝(1201d)에 시간 MAC 회로에서부터 APR 유닛으로 입력된 경우, 최종 MAC 결과의 전압이 39.73mV 상승하는 것을 나타낸다. 전압 곡선(1210d)은 LSB의 다음 비트에 대응하는 MAC 결과가 제2 시간 스텝(1202d)에 APR 유닛으로 입력되는 경우, 20.09mV의 전압 상승을 나타낸다. 다시 말해, 제1 시간 스텝에서의 전압 상승에 비해 제2 시간 스텝에서의 전압 상승이 약 절반으로 줄어들 수 있다.

[0158] 도 13은 일 실시예에 따른 SNN 연산 가속 장치에 포함된 9T1C 구조의 비트셀 레이아웃을 나타낸다.

[0159] 비트셀 레이아웃(1300)은 6T 정적 랜덤 액세스 메모리(이하, 'SRAM')(1301), NMOS, PMOS 및 커패시터(1302)를 포함할 수 있다. 비트셀 레이아웃(1300)은 가로로 2.76 μm, 세로로 1.66 μm의 길이를 나타낼 수 있다. 비트셀 레이아웃(1300)에서 가로 길이 2.76 μm의 6T SRAM(1301)이 포함될 수 있다. 비트셀 레이아웃(1300)에서 NMOS 및 PMOS는 6T SRAM(1301)의 가로 방향 축을 따라, 6T SRAM(1301)의 가로와 인접하게 배열될 수 있다. 비트셀 레이아웃(1300)에서 커패시터(1302)는 1fF의 커패시턴스를 가질 수 있다. 비트셀 레이아웃(1300)에서 커패시터(1302)는 6T SRAM(1301)상에 적층될 수 있다. 비트셀 레이아웃(1300)에 따르면, 비교 실시예에 비해 3개 적은 트랜지스터를 사용함으로써 25%의 면적 오버헤드를 줄일 수 있다.

[0161] 도 14는 일 실시예에 따른 SNN 연산 가속 장치의 레이아웃을 나타낸다.

[0162] 레이아웃(1400)에서 일 실시예에 따른SNN 연산 가속 장치는 65nm CMOS 로직 기술로 구현될 수 있다. SNN 연산 가속 장치는 626 μm ×271 μm를 차지할 수 있다.

[0163] 이하 표 4는 일 실시예에 따른 SNN 연산 가속 장치의 사양을 나타낸다.

표 4

[0164]

Process	65 nm CMOS
Area	0.17 mm^2
Supply	1.0 V
Frequency	50 Mhz
Power	656.64 μW
Capacity	2.25 KB
Energy Efficiency	701.7 TOPS/W

[0165] 표 4에 따르면, 일 실시예에 따른 SNN 연산 가속 장치는 1.0V 공급 전압 및 50 Mhz 작동 주파수에서 실행되고, 656.64㎼의 전력을 소모할 수 있다. SNN 연산 가속 장치의 에너지 효율은 701.7 TOPS/W를 달성할 수 있다.

[0166] 이하 표 5는 일 실시예에 따른 SNN 연산 가속 장치와 비교 실시예들의 사양을 항목에 따라 비교한다.

표 5

[0167]

	비교 실시예 1	비교 실시예 2	비교 실시예 3	SNN 연산 가속 장치
Technology	7nm	28nm	40nm	65nm
Comp. Domain	Current	Current	Time	Time
In-Memory Comp. Frequency (Mhz)	N/A	N/A	0.5	50
Power Supply(V)	1.0	1.1	0.9	1.0
Cell Structure	8T	8T	12T	9T1C
Input/Weigh Precision	4b / 4b	4b / 4b	1b / 1b	4b / 4b
Energy Efficiency of Macro (TOPS/W)	351	413.8	716	701.7

[0168] 표 5에서, 비교 실시예 1은 M. E. Sinangil et al., "A 7-nm Compute-in-Memory SRAM Macro Supporting Multi-Bit Input, Weight and Output and Achieving 351 TOPS/W and 372.4 GOPS," IEEE Journal of Solid-State Circuits, vol. 56, no. 1, pp. 188-198, Jan. 2021. 문헌의 기재에 따라 구현된 장치일 수 있다. 비교 실시예 2는 S. Kim, S. Kim, S. Um, S. Kim, K. Kim and H. -J. Yoo, "Neuro-CIM: A 310.4 TOPS/W Neuromorphic Computing-in-Memory Processor with Low WL/BL activity and Digital-Analog Mixed-mode Neuron Firing," in Proc. IEEE Symp. VLSI Circuits (VLSIC), 2022, pp. 38-39. 문헌의 기재에 따라 구현된 장치일 수 있다. 비교 실시예 3은 J. Song et al., "TD-SRAM: Time-Domain-Based In-Memory Computing Macro for Binary Neural Networks," IEEE Transactions on Circuits and Systems I: Regular Papers, vol. 68, no. 8, pp. 3377-3387, Aug. 2021. 문헌의 기재에 따라 구현된 장치일 수 있다. 제안된 SNN 연산 가속 장치는 시간 영역 연산을 수행할 수 있다. 비교 실시예 1 및 비교 실시예 2는 일 실시예에 따른 SNN 연산 가속 장치와 동일한 입력 및 가중치 정확도로 전류 영역(current domain) 연산을 수행할 수 있다. 제안된 SNN 연산 가속 장치는 연산 단계에서 싱크(sink) 또는 소스(source) 경로가 없을 수 있다. 따라서, 일 실시예에 따른 SNN 연산 가속 장치는 표 5의 비교 실시예 1 보다 1.99배 더 높은 에너지 효율을 달성할 수 있다. 또한, SNN 연산 가속 장치는 표 5의 비교 실시예 2 보다 1.69배 더 높은 에너지 효율을 달성할 수 있다. 제안된 레이아웃(1400) 설계 방식에 따르면, SNN 연산 가속 장치는, 표 5의 비교 실시예 3에 비해, IF(integrate and fire) 모델을 적용한 TDC(time to digital converter)를 제거함으로써 에너지 효율의 손실이 2%에 불과하고, 메모리 계산 속도가 100배 향상될 수 있다. 일 실시예에 따른 SNN 연산 가속 장치는, 또한, 비교 실시예 3이 1비트 입력 및 1비트

의 가중치 정확도로 인해 복잡한 신경망을 지원할 수 없는데 반해, 4 비트 정확도를 지원할 수 있다.

[0169] 위에서 설명한 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 또는 복수의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

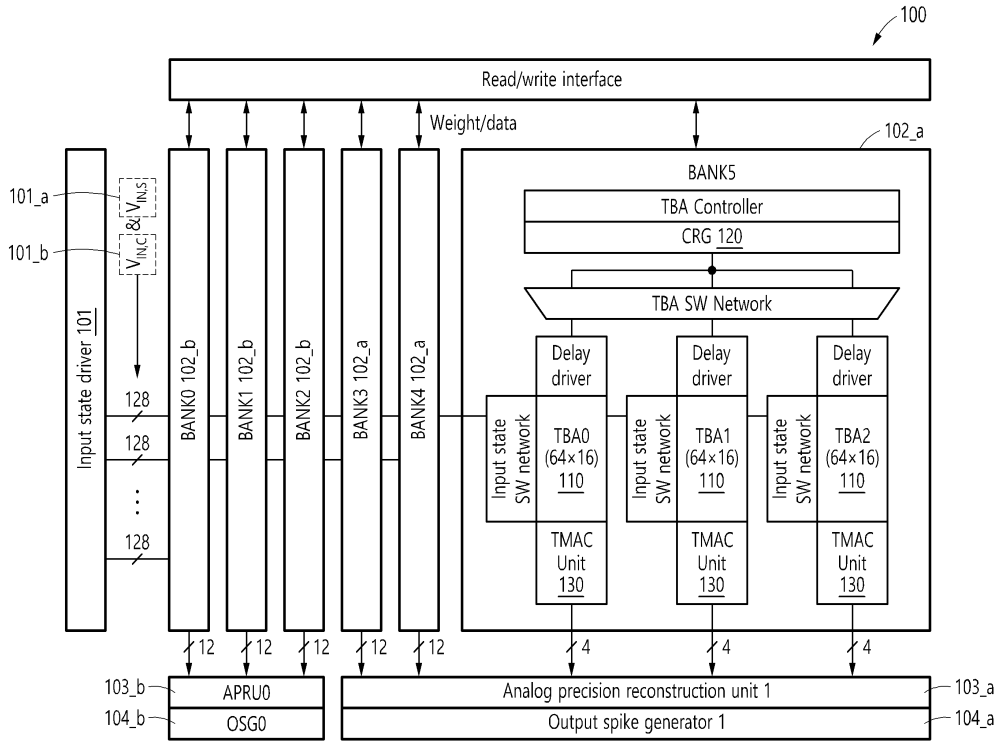
[0170] 본 문서에서, "A 또는 B", "A 및 B 중 적어도 하나", "A 또는 B 중 적어도 하나", "A, B 또는 C", "A, B 및 C 중 적어도 하나", 및 "A, B, 또는 C 중 적어도 하나"와 같은 문구들 각각은 그 문구들 중 해당하는 문구에 함께 나열된 항목들 중 어느 하나, 또는 그들의 모든 가능한 조합을 포함할 수 있다.

[0171] 이상과 같이 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 이를 기초로 다양한 기술적 수정 및 변형을 적용할 수 있다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다.

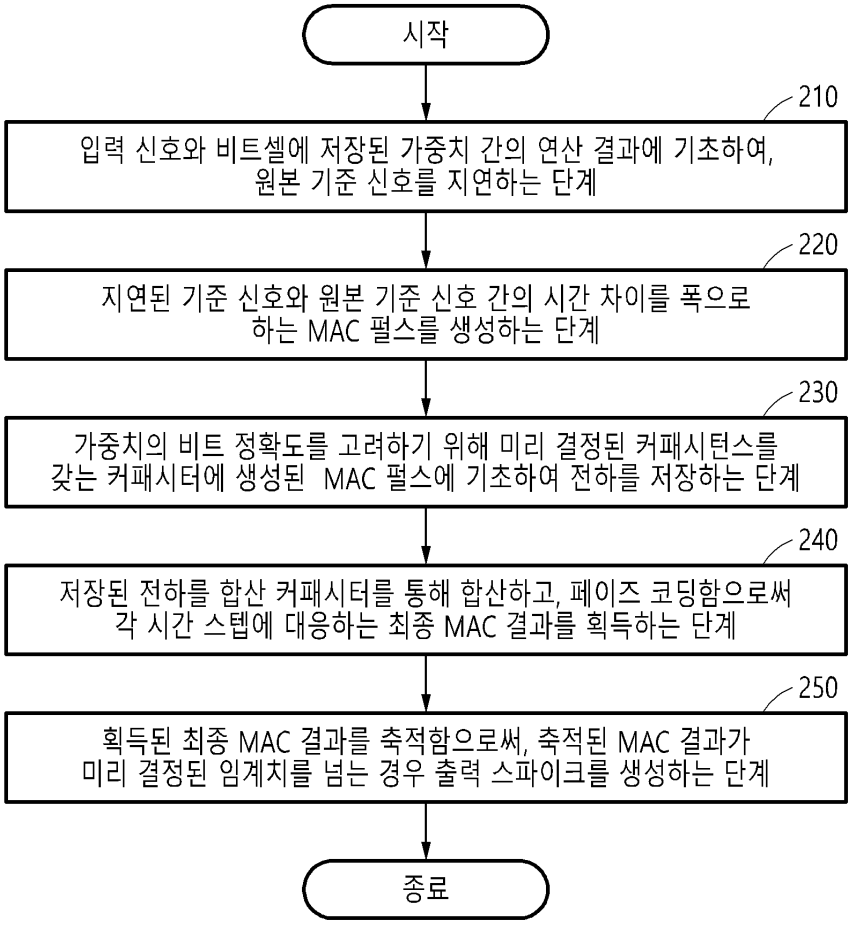
[0172] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

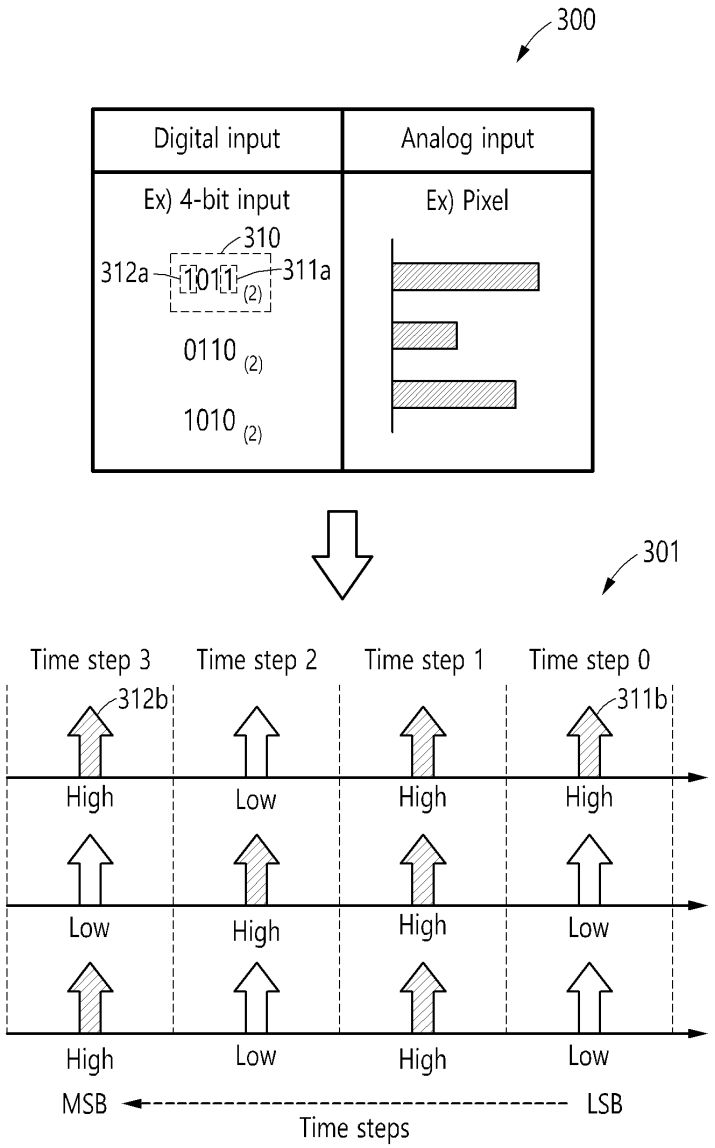
도면1



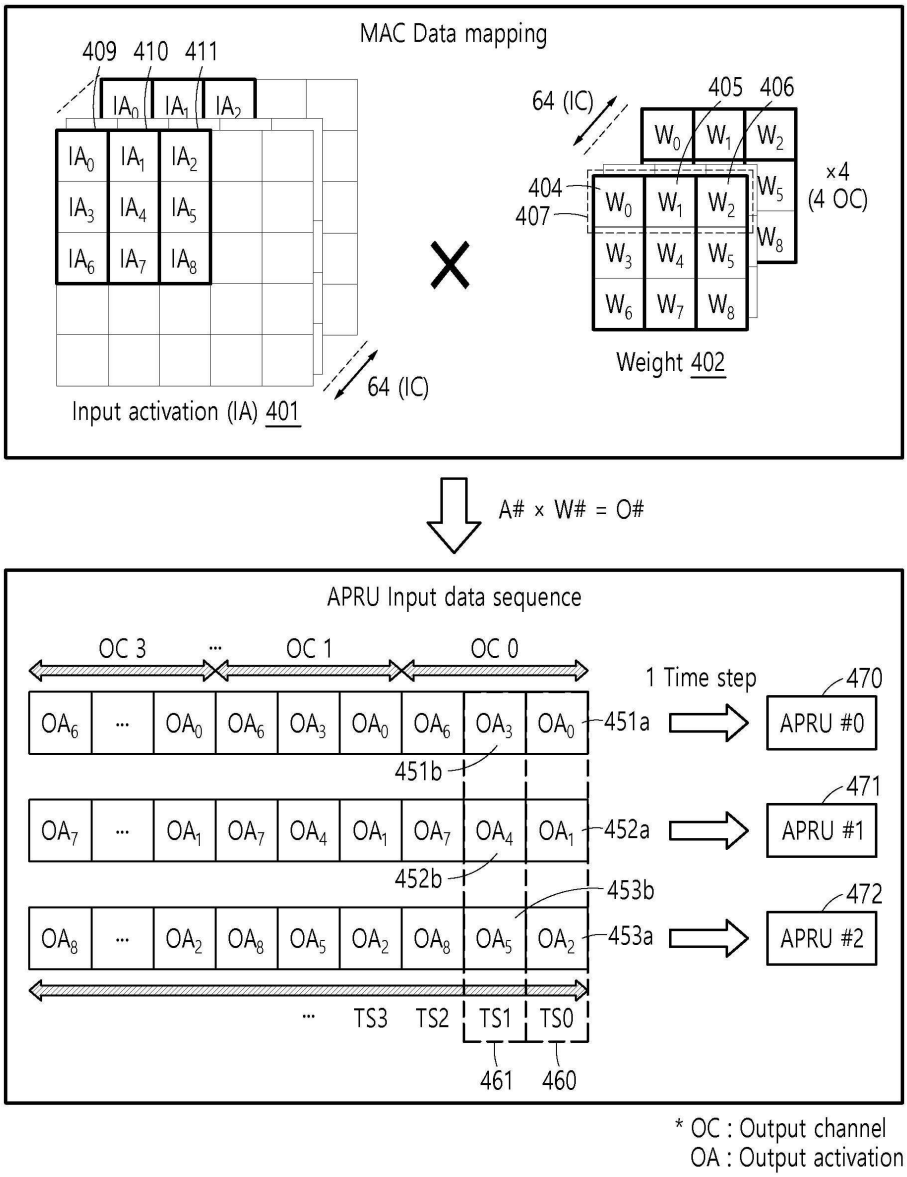
도면2


시작
210
입력 신호와 비트셀에 저장된 가중치 간의 연산 결과에 기초하여, 원본 기준 신호를 지연하는 단계
220
지연된 기준 신호와 원본 기준 신호 간의 시간 차이를 폭으로 하는 MAC 펄스를 생성하는 단계
230
가중치의 비트 정확도를 고려하기 위해 미리 결정된 커패시턴스를 갖는 커패시터에 생성된 MAC 펄스에 기초하여 전하를 저장하는 단계
240
저장된 전하를 합산 커패시터를 통해 합산하고, 페이즈 코딩함으로써 각 시간 스텝에 대응하는 최종 MAC 결과를 획득하는 단계
250
획득된 최종 MAC 결과를 축적함으로써, 축적된 MAC 결과가 미리 결정된 임계치를 넘는 경우 출력 스파이크를 생성하는 단계
종료

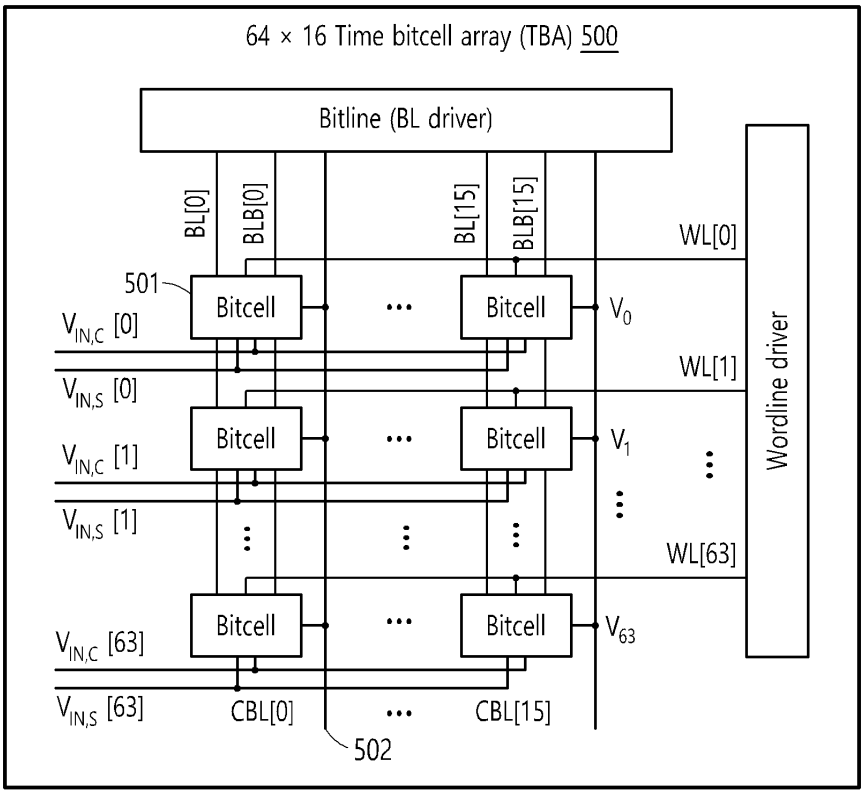
도면3


300
Digital input
Analog input
Ex) 4-bit input
Ex) Pixel
310
312a
1011 (2)
311a
0110 (2)
1010 (2)
301
Time step 3
Time step 2
Time step 1
Time step 0
312b
311b
High
Low
High
High
Low
High
High
Low
High
Low
High
Low
MSB
LSB
Time steps

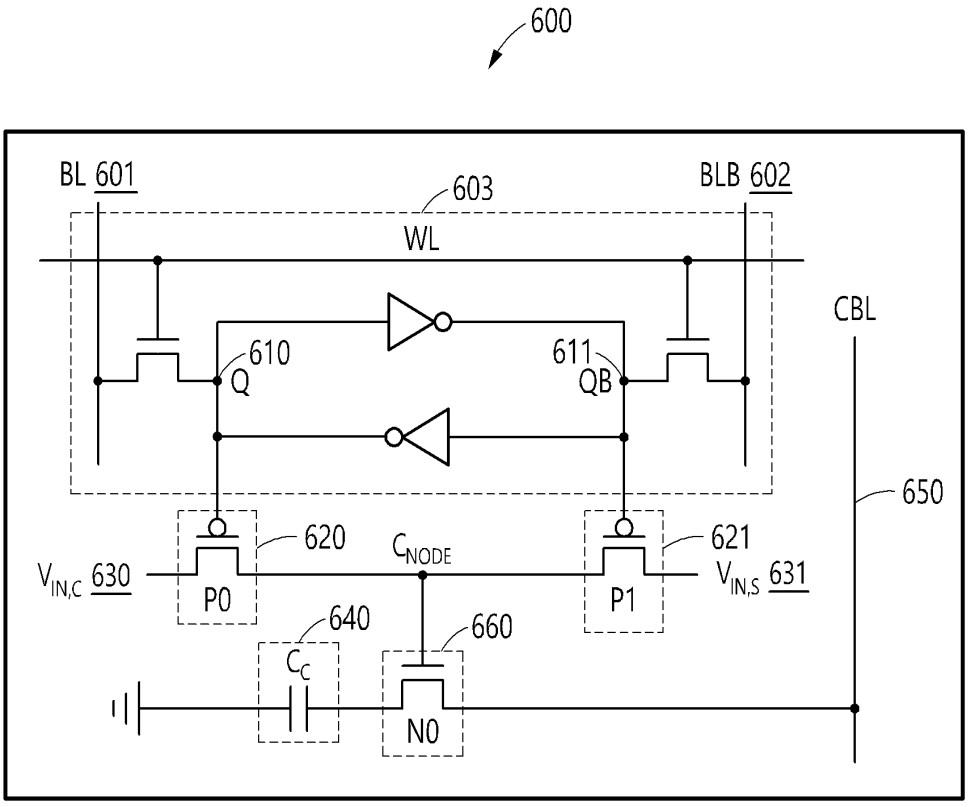
도면4


MAC Data mapping
409 410 411
IA0 IA1 IA2
IA3 IA4 IA5
IA6 IA7 IA8
64 (IC)
Input activation (IA) 401
64 (IC)
405 406
W0 W1 W2
404
407
W3 W4 W5
W6 W7 W8
×4
(4 OC)
Weight 402
A# × W# = O#
APRU Input data sequence
OC 3
OC 1
OC 0
1 Time step
470
APRU #0
451a
451b
471
APRU #1
452a
452b
453b
472
APRU #2
453a
TS3 TS2 TS1 TS0
461 460
* OC : Output channel
OA : Output activation

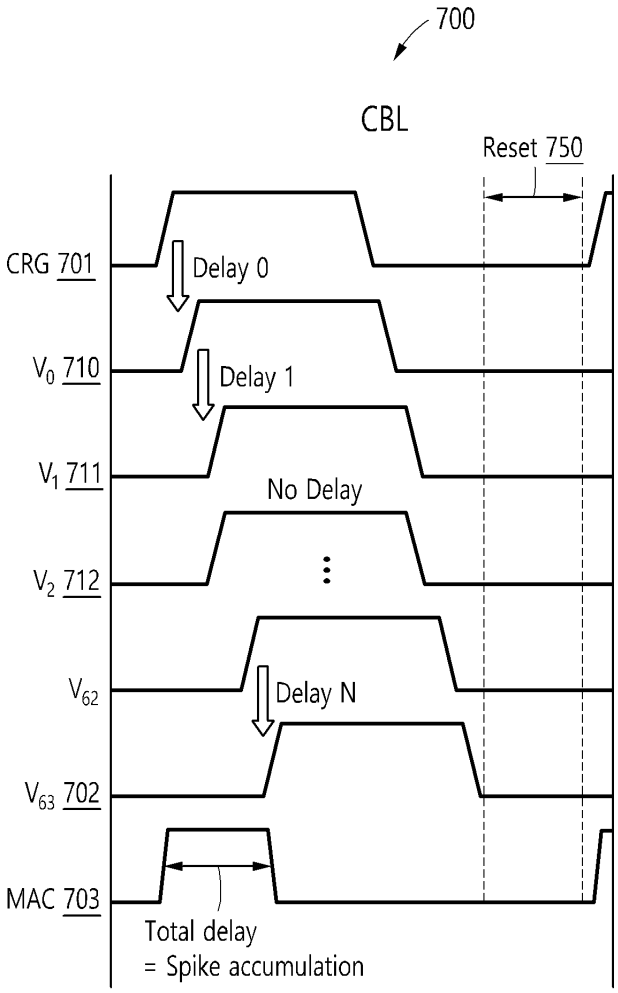
도면5


64 × 16 Time bitcell array (TBA) 500
Bitline (BL driver)
BL[0]
BLB[0]
BL[15]
BLB[15]
WL[0]
WL[1]
WL[63]
Wordline driver
501
Bitcell
$V_{IN,C}$ [0]
$V_{IN,S}$ [0]
$V_{IN,C}$ [1]
$V_{IN,S}$ [1]
$V_{IN,C}$ [63]
$V_{IN,S}$ [63]
V_0
V_1
V_{63}
CBL[0]
CBL[15]
502

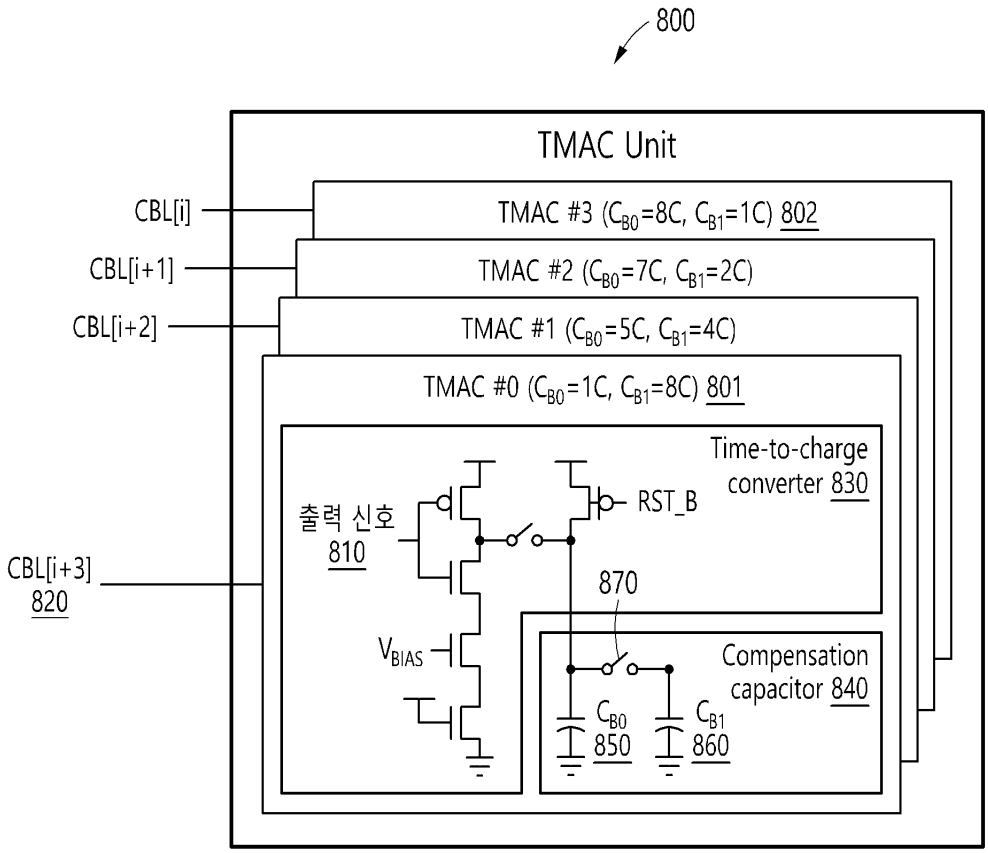
도면6


600
BL 601
BLB 602
603
WL
CBL
610
Q
611
QB
650
620
C_{NODE}
621
$V_{IN,C}$ 630
$V_{IN,S}$ 631
P0
P1
640
660
C_C
N0

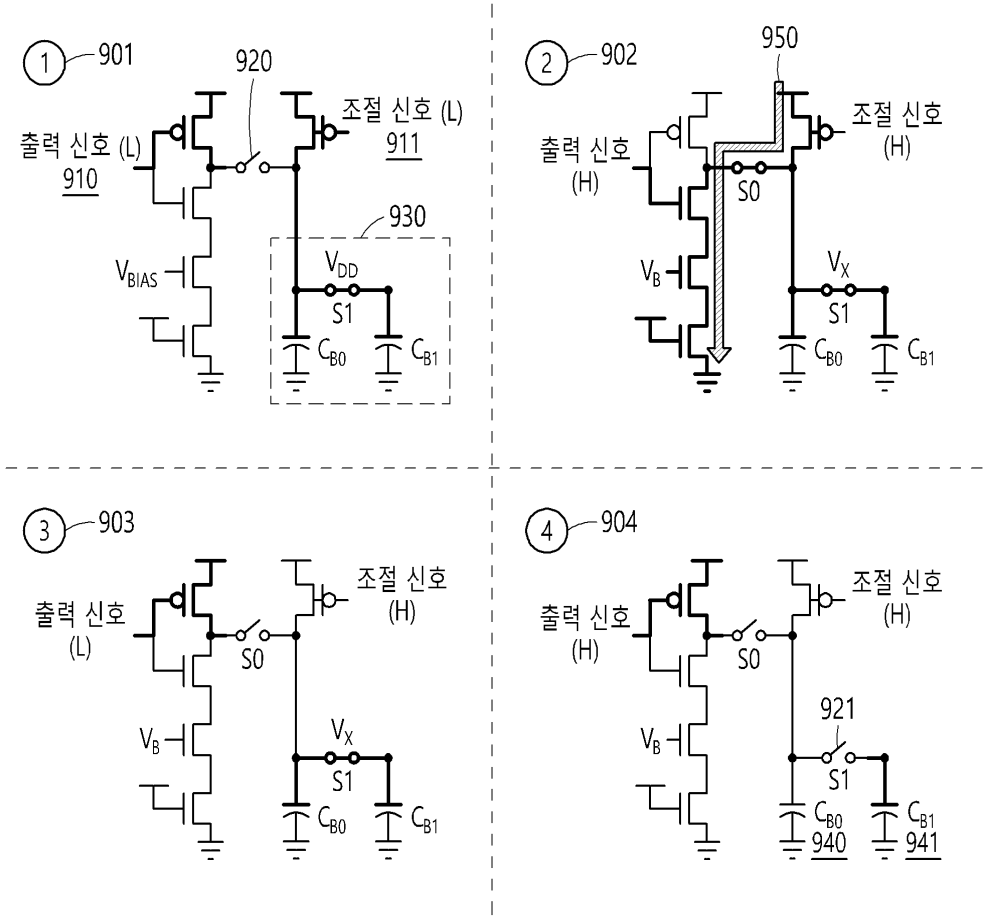
도면7


700
CBL
Reset 750
CRG 701
Delay 0
V0 710
Delay 1
V1 711
No Delay
V2 712
V62
Delay N
V63 702
MAC 703
Total delay
= Spike accumulation

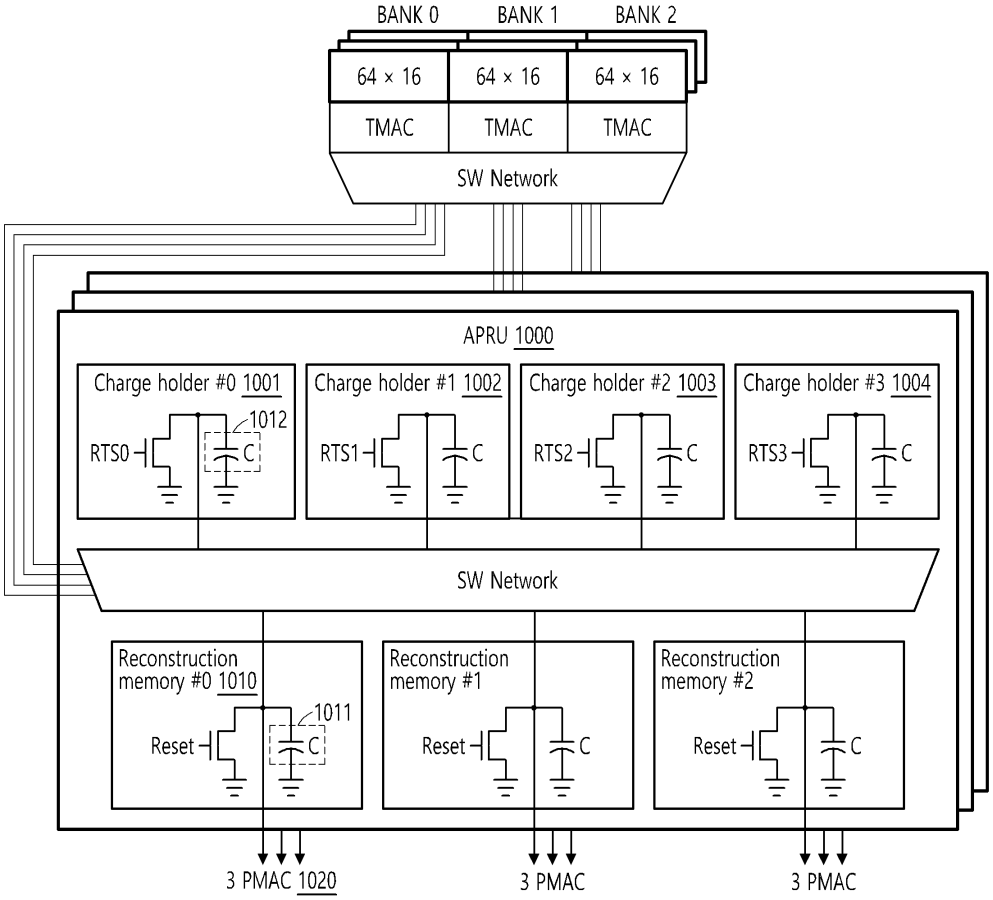
도면8


800
TMAC Unit
CBL[i]
CBL[i+1]
CBL[i+2]
CBL[i+3]
820
TMAC #3 (C_{B0}=8C, C_{B1}=1C) 802
TMAC #2 (C_{B0}=7C, C_{B1}=2C)
TMAC #1 (C_{B0}=5C, C_{B1}=4C)
TMAC #0 (C_{B0}=1C, C_{B1}=8C) 801
Time-to-charge converter 830
출력 신호 810
RST_B
870
V_{BIAS}
Compensation capacitor 840
C_{B0} 850
C_{B1} 860

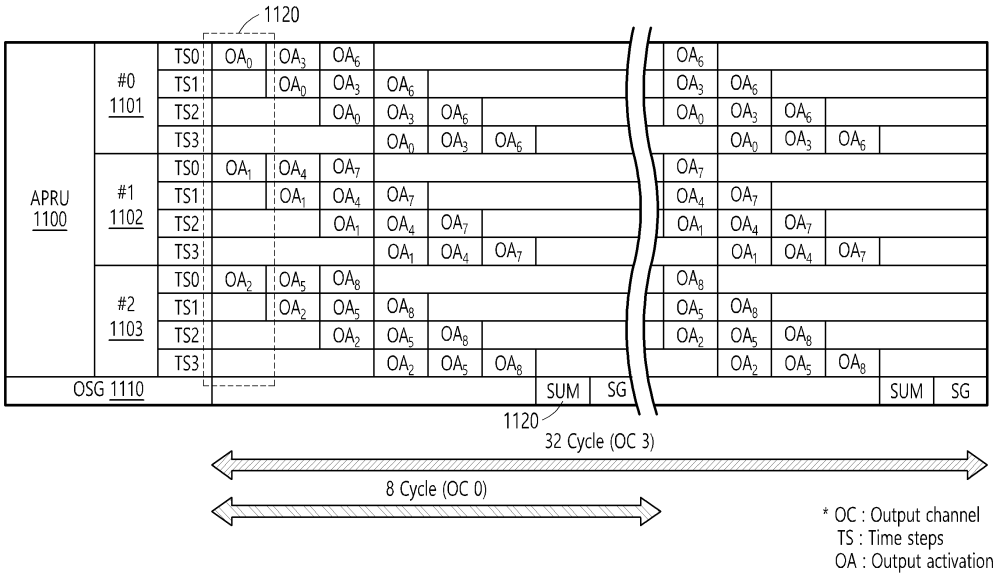
도면9


1
901
920
출력 신호 (L)
910
조절 신호 (L)
911
930
VBIAS
VDD
S1
CB0
CB1
2
902
950
출력 신호
(H)
조절 신호
(H)
S0
VB
VX
S1
CB0
CB1
3
903
출력 신호
(L)
조절 신호
(H)
S0
VB
VX
S1
CB0
CB1
4
904
출력 신호
(H)
조절 신호
(H)
S0
921
VB
S1
CB0
940
CB1
941

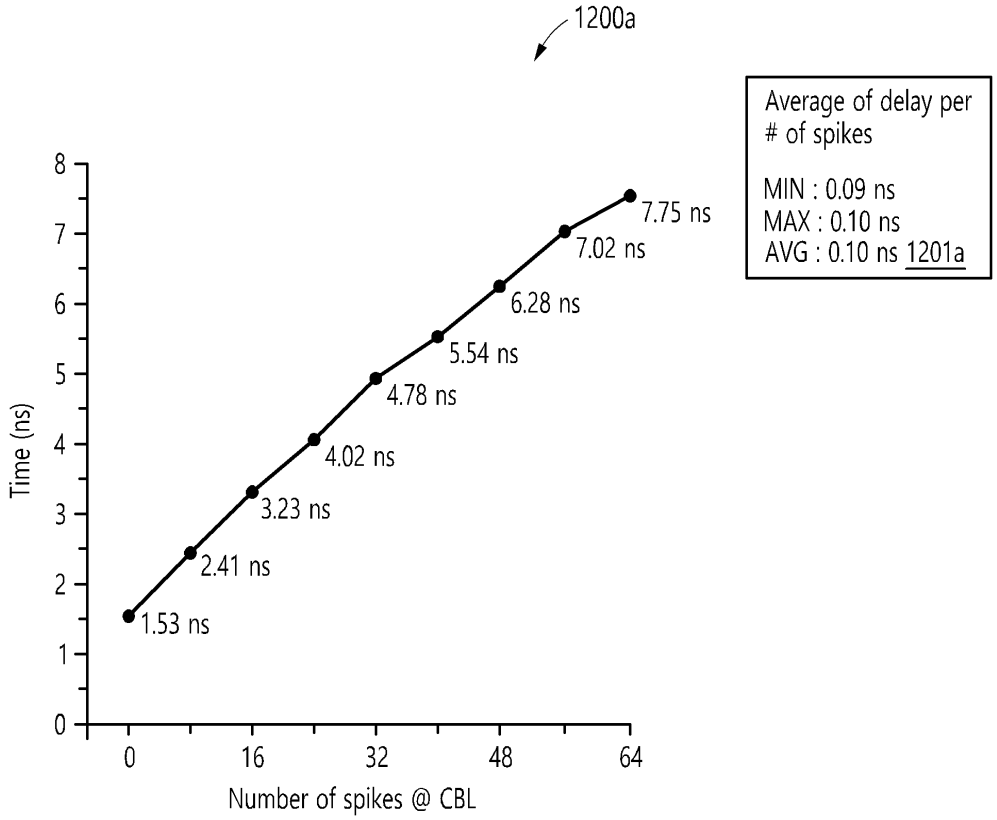
도면10


BANK 0
BANK 1
BANK 2
64 × 16
64 × 16
64 × 16
TMAC
TMAC
TMAC
SW Network
APRU 1000
Charge holder #0 1001
Charge holder #1 1002
Charge holder #2 1003
Charge holder #3 1004
1012
RTS0
RTS1
RTS2
RTS3
C
SW Network
Reconstruction memory #0 1010
Reconstruction memory #1
Reconstruction memory #2
1011
Reset
3 PMAC 1020
3 PMAC
3 PMAC

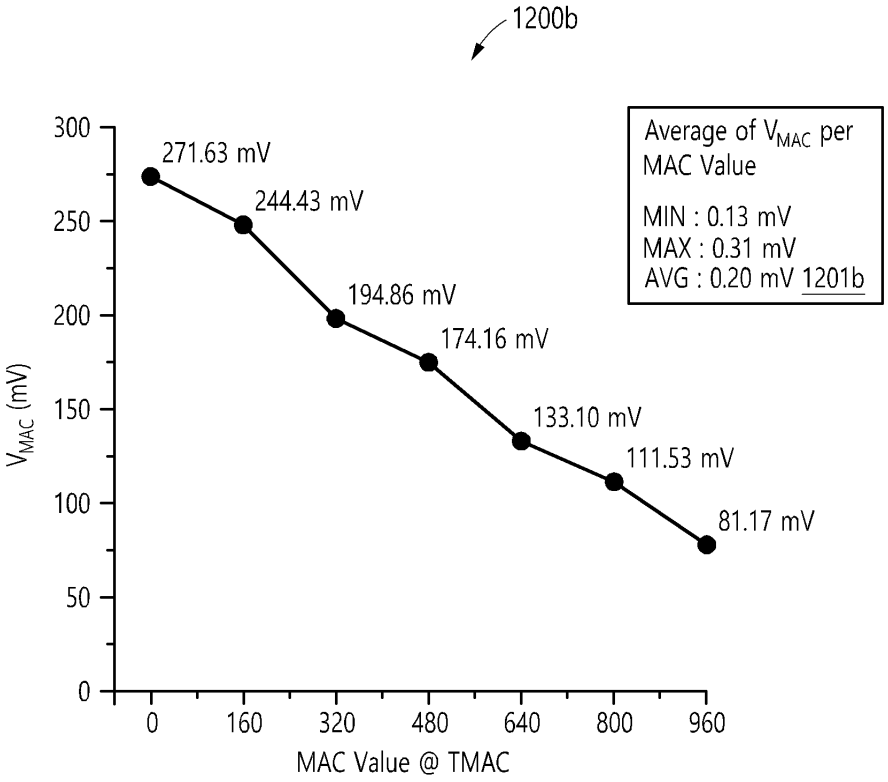
도면11


1120
APRU 1100
#0 1101
#1 1102
#2 1103
TS0
TS1
TS2
TS3
OSG 1110
SUM
SG
1120
32 Cycle (OC 3)
8 Cycle (OC 0)
* OC : Output channel
TS : Time steps
OA : Output activation

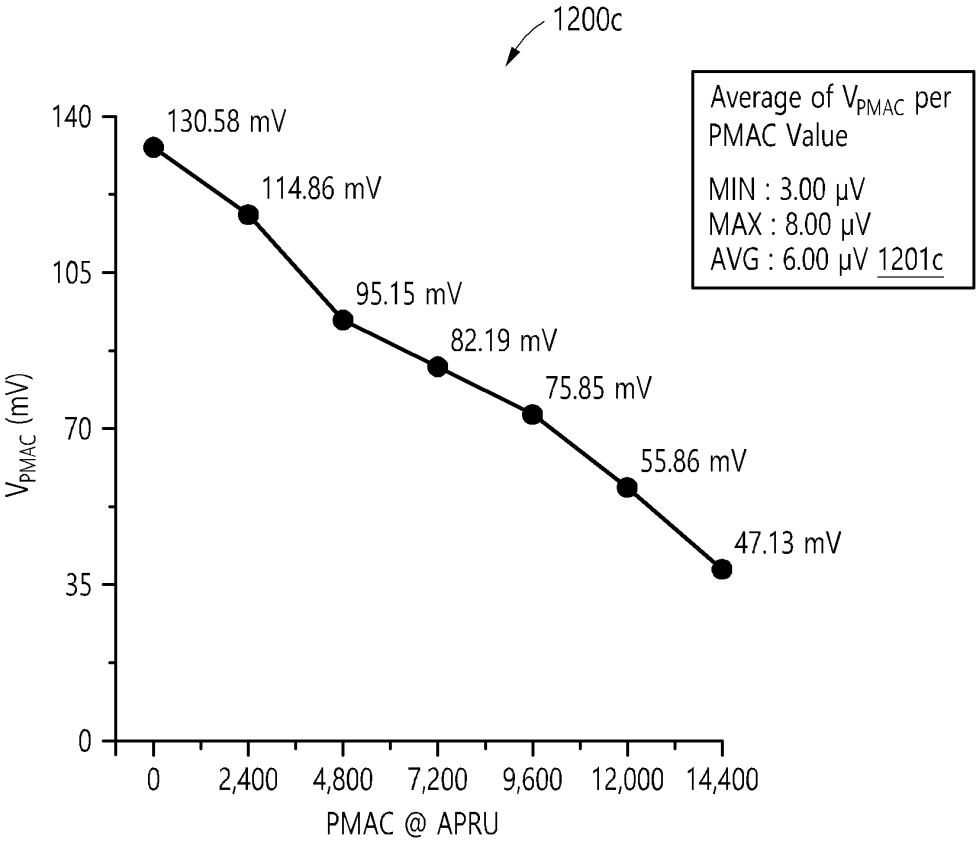
도면12a


1200a
Average of delay per # of spikes
MIN : 0.09 ns
MAX : 0.10 ns
AVG : 0.10 ns 1201a
Time (ns)
Number of spikes @ CBL
1.53 ns
2.41 ns
3.23 ns
4.02 ns
4.78 ns
5.54 ns
6.28 ns
7.02 ns
7.75 ns

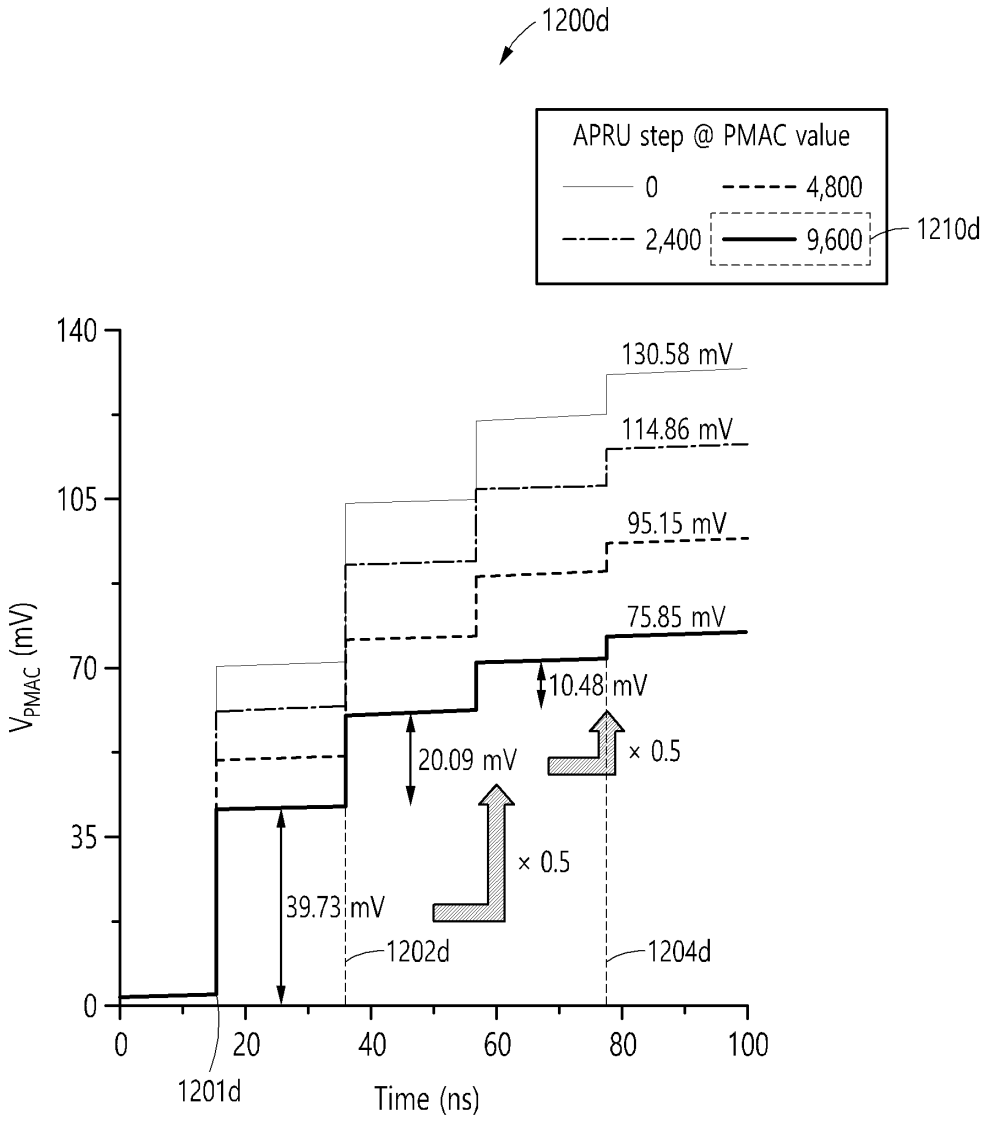
도면12b


1200b
Average of V_{MAC} per MAC Value
MIN : 0.13 mV
MAX : 0.31 mV
AVG : 0.20 mV 1201b
V_{MAC} (mV)
MAC Value @ TMAC
271.63 mV
244.43 mV
194.86 mV
174.16 mV
133.10 mV
111.53 mV
81.17 mV

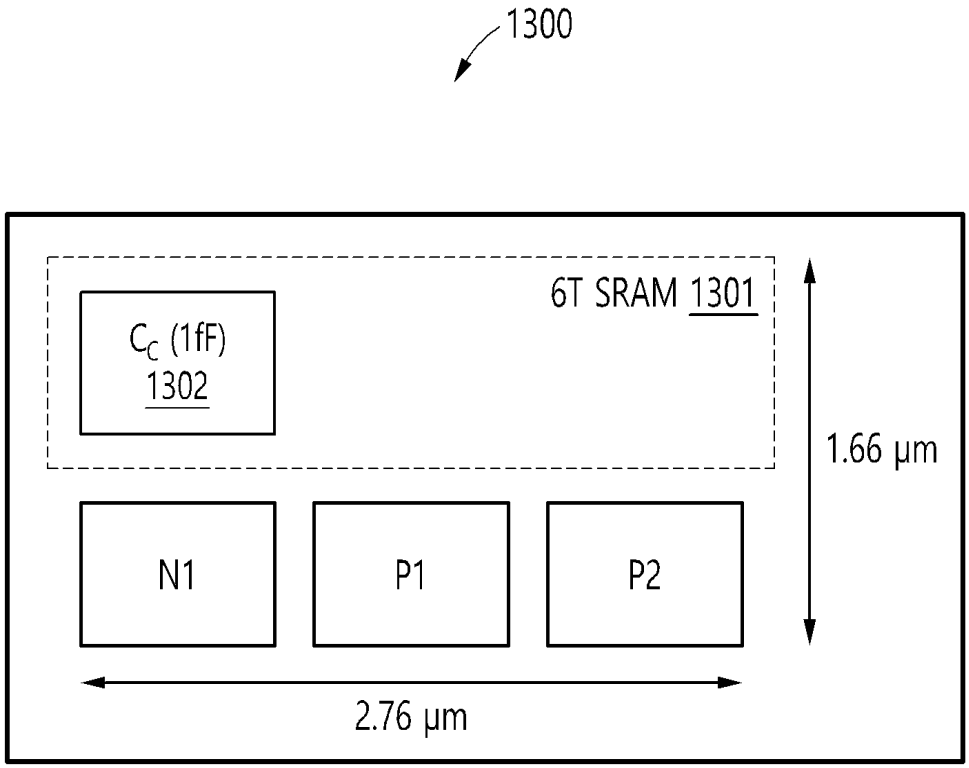
도면12c


1200c
Average of V_{PMAC} per PMAC Value
MIN : 3.00 μV
MAX : 8.00 μV
AVG : 6.00 μV 1201c
130.58 mV
114.86 mV
95.15 mV
82.19 mV
75.85 mV
55.86 mV
47.13 mV
140
105
70
35
0
V_{PMAC} (mV)
0
2,400
4,800
7,200
9,600
12,000
14,400
PMAC @ APRU

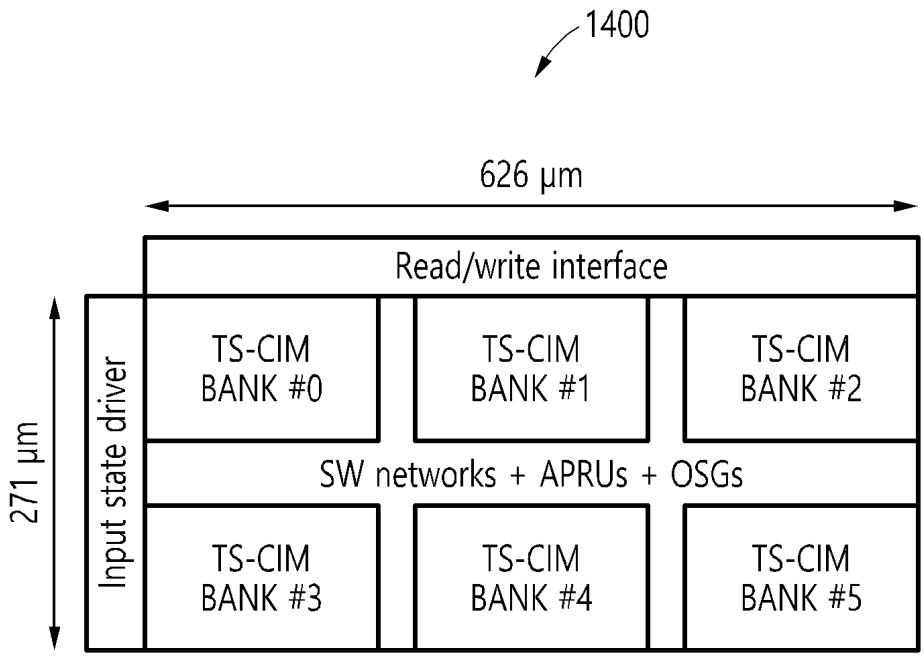
도면12d


1200d
APRU step @ PMAC value
0
4,800
2,400
9,600
1210d
140
105
70
35
0
V_{PMAC} (mV)
130.58 mV
114.86 mV
95.15 mV
75.85 mV
10.48 mV
20.09 mV
× 0.5
× 0.5
39.73 mV
1202d
1204d
1201d
0
20
40
60
80
100
Time (ns)

도면13


1300
6T SRAM 1301
C_C (1fF)
1302
1.66 μm
N1
P1
P2
2.76 μm

도면14


1400
626 μm
Read/write interface
TS-CIM
BANK #0
TS-CIM
BANK #1
TS-CIM
BANK #2
SW networks + APRUs + OSGs
TS-CIM
BANK #3
TS-CIM
BANK #4
TS-CIM
BANK #5
Input state driver
271 μm