(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0081663
(43) 공개일자 2021년07월02일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2006.01) *G06F 13/16* (2006.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06F 13/1605 (2013.01)
(21) 출원번호 10-2019-0173846
(22) 출원일자 2019년12월24일
심사청구일자 없음

(71) 출원인
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
박용하
경기도 성남시 분당구 동판교로 156, 9142동 202호 (삼평동, 봇들마을9단지금호어울림아파트)
(74) 대리인
특허법인 무한

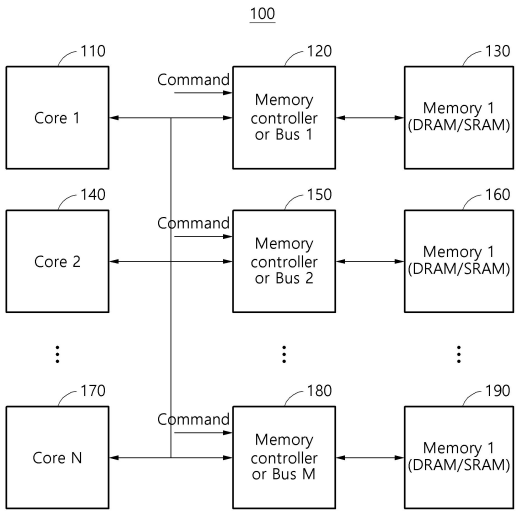
전체 청구항 수 : 총 24 항

(54) 발명의 명칭 **인터커넥트 장치, 인터커넥트 장치의 동작 방법 및 인터커넥트 장치를 포함하는 AI(Artificial Intelligence) 가속기 시스템**

(57) 요 약

일 실시예에 따른 인터커넥트 장치는 프로세싱 코어로부터 명령을 수신하고, 명령을 기초로, 메모리에 저장된 데이터들에 대한 누적 연산 및 프로세싱 코어에서 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행하며, 적어도 하나의 연산의 수행 결과를 제공한다.

대 표 도 - 도1



(52) CPC특허분류
G06F 13/1668 (2013.01)

명 세 서

청구범위

청구항 1

프로세싱 코어(processing core)로부터 명령(command)을 수신하고, 상기 명령을 기초로, 메모리(memory)에 저장된 데이터들에 대한 누적 연산(accumulation operation) 및 상기 프로세싱 코어에서 처리된 결과들에 대한 집합 연산(aggregation operation) 중 적어도 하나의 연산을 수행하며, 상기 적어도 하나의 연산의 수행 결과를 제공하는,

인터커넥트 장치(interconnect device).

청구항 2

제1항에 있어서,

상기 명령은

상기 누적 연산 및 상기 집합 연산 각각을 위한 오퍼레이션 코드(operation code; OP code)와 상기 오퍼레이션 코드에 따른 동작을 수행하기 위한 데이터들이 저장된 어드레스 정보(address information); 및

상기 데이터들이 저장된 상기 메모리의 어드레스 정보

중 어느 하나를 포함하는,

인터커넥트 장치.

청구항 3

제1항에 있어서,

상기 인터커넥트 장치는

상기 프로세싱 코어로부터 수신한 명령을 저장 및 전달하는 커맨드 모듈;

상기 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달하는 어드레스 모듈; 및

상기 명령에 따라 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈

을 포함하는,

인터커넥트 장치.

청구항 4

제3항에 있어서,

상기 읽기 데이터 모듈은

상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 읽기 데이터 모듈에 저장된 데이터를 합산하는 가산기(adder); 및

상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 합산한 데이터 중 어느 하나를 상기 읽기 데이터 모듈로 제공하는 멀티플렉서(MUX)

를 포함하는,

인터커넥트 장치.

청구항 5

제4항에 있어서,

상기 읽기 데이터 모듈은

상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 읽기 데이터 모듈에 저장된 데이터를 곱하는 곱셈기(multiplier)를 더 포함하는,

인터커넥트 장치.

청구항 6

제3항에 있어서,

상기 인터커넥트 장치는

상기 명령을 기초로 상기 커맨드 모듈에게 제어 신호를 제공하고, 상기 프로세싱 코어로부터 수신한 어드레스 정보에 기초한 메모리의 어드레스를 상기 어드레스 모듈에 제공하는 제어 모듈

을 더 포함하고,

상기 커맨드 모듈은

상기 제어 신호를 상기 읽기 데이터 모듈 및 상기 메모리에 전송하는,

인터커넥트 장치.

청구항 7

제6항에 있어서,

상기 제어 모듈은

상기 어드레스 정보에 따른 상기 메모리의 소스 어드레스(source address)를 저장하는 레지스터(register);

상기 명령에 따라 상기 소스 어드레스에 기초한 카운팅(counting)을 수행하는 카운터 레지스터(counter register); 및

상기 카운팅 결과에 기초하여 결정한 상기 어드레스를 상기 어드레스 모듈로 제공하는 컨트롤러(controller)

를 포함하는,

인터커넥트 장치.

청구항 8

제1항에 있어서,

상기 인터커넥트 장치는

상기 명령에 기초하여 제어 신호를 생성하고, 상기 프로세싱 코어로부터 수신한 어드레스 정보에 기초하여 메모리의 어드레스를 결정하는 제어 모듈;

상기 명령을 기초로, 상기 어드레스에 대응하는 신호를 생성하는 상기 어드레스 모듈;

상기 제어 모듈로부터 수신한 제어 신호를 상기 메모리로 전송하는 상기 커맨드 모듈; 및

상기 제어 신호에 기초하여 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈

을 포함하고,

상기 읽기 데이터 모듈은 복수의 서브 데이터 모듈들을 포함하고,

상기 제어 신호에 기초하여, 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈에 저장된 누적 데이터- 상기 누적 데이터는 상기 메모리로부터 읽어온 데이터 및 상기 어느 하나의 서브 데이터 모듈의 데이터

를 누적한 것임- 를 상기 프로세싱 코어에게 제공하는,

인터커넥트 장치.

청구항 9

제8항에 있어서,

상기 읽기 데이터 모듈은

상기 제어 신호에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈의 데이터를 합산하는 가산기;

상기 제어 신호에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 합산한 데이터 중 어느 하나의 데이터를 출력하는 제1 멀티플렉서(MUX);

상기 제어 신호에 따라, 상기 제1 멀티플렉서에서 출력된 데이터를 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈로 출력하는 디멀티플렉서(DEMUX); 및

상기 제어 신호에 기초하여, 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈의 데이터를 출력하는 제2 멀티플렉서

를 포함하는, 인터커넥트 장치.

청구항 10

제8항에 있어서,

상기 제어 모듈은

상기 어드레스 정보에 따른 상기 메모리의 소스 어드레스를 저장하는 레지스터;

상기 명령에 따라 상기 소스 어드레스에 기초한 카운팅을 수행하는 카운터 레지스터; 및

상기 카운팅 결과에 기초하여 결정한 상기 어드레스를 상기 어드레스 모듈로 제공하고, 상기 제어 신호를 상기 커맨드 모듈로 제공하는 컨트롤러

를 포함하는,

인터커넥트 장치.

청구항 11

제1항에 있어서,

상기 인터커넥트 장치는

상기 명령에 기초하여 제어 신호를 생성하고, 상기 프로세싱 코어로부터 수신한 어드레스 정보에 기초하여 메모리의 어드레스를 결정하는 제어 모듈;

상기 명령을 기초로, 상기 어드레스에 대응하는 신호를 생성하는 상기 어드레스 모듈;

상기 제어 모듈로부터 수신한 제어 신호를 상기 메모리로 전송하는 상기 커맨드 모듈; 및

상기 제어 신호에 기초하여 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈

을 포함하고,

상기 읽기 데이터 모듈은

상기 메모리부터 수신한 데이터를 누적한 누적 데이터를 저장하는 제1 서브 데이터 모듈; 및

상기 메모리로부터 읽어온 데이터를 저장하는 제2 서브 데이터 모듈

을 포함하고,

상기 읽기 데이터 모듈은

상기 제어 신호에 기초하여, 상기 누적 데이터 및 상기 데이터 중 어느 하나의 데이터를 출력하는,

인터커넥트 장치.

청구항 12

제1항에 있어서,

상기 인터커넥트 장치는

상기 프로세싱 코어로부터 수신한 명령을 저장 및 전달하는 커맨드 모듈;

상기 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달하는 어드레스 모듈; 및

상기 명령에 따라 상기 프로세싱 코어에서 처리된 결과 데이터 또는 상기 프로세싱 코어로부터 수신한 처리된 결과 데이터들을 누적한 누적 데이터를 상기 메모리에 전송하는 쓰기 데이터 모듈

을 포함하는,

인터커넥트 장치.

청구항 13

제19항에 있어서,

상기 쓰기 데이터 모듈은

상기 프로세싱 코어에서 처리된 결과 데이터 및 상기 메모리에 저장되는 데이터를 합산하는 가산기; 및

상기 명령에 기초하여 상기 프로세싱 코어에서 처리된 결과 데이터 및 상기 합산한 데이터 중 어느 하나의 데이터를 상기 메모리로 제공하는 멀티플렉서(MUX)

를 포함하는,

인터커넥트 장치.

청구항 14

제13항에 있어서,

상기 쓰기 데이터 모듈은

상기 합산한 데이터를 나누는 디바이더(divider) 또는 상기 합산한 데이터를 한 비트(bit) 씩 이동시키는 쉬프트 레지스터(shift register)

를 더 포함하는,

인터커넥트 장치.

청구항 15

제1항에 있어서,

상기 인터커넥트 장치는

상기 프로세싱 코어로부터 수신한 명령을 저장 및 전달하는 커맨드 모듈;

상기 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달하는 어드레스 모듈;

상기 명령에 따라 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈; 및

상기 명령에 따라 상기 프로세싱 코어로부터 수신한 데이터 또는 상기 프로세싱 코어로부터 수신한 데이터들을

누적한 누적 데이터를 상기 메모리에 전송하는 쓰기 데이터 모듈

을 포함하는,

인터커넥트 장치.

청구항 16

제15항에 있어서,

상기 읽기 데이터 모듈은

상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 읽기 데이터 모듈에 저장된 데이터를 합산하는 가산기; 및

상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 합산한 데이터 중 어느 하나를 상기 읽기 데이터 모듈로 제공하는 멀티플렉서

를 포함하는,

인터커넥트 장치.

청구항 17

제15항에 있어서,

상기 쓰기 데이터 모듈은

상기 프로세싱 코어로부터 수신한 데이터 및 상기 메모리에 저장되는 데이터를 합산하는 가산기; 및

상기 명령에 기초하여 상기 프로세싱 코어로부터 수신한 데이터 및 상기 합산한 데이터 중 어느 하나의 데이터를 상기 메모리로 제공하는 멀티플렉서

를 포함하는,

인터커넥트 장치.

청구항 18

제1항에 있어서,

상기 인터커넥트 장치는

DMA(Direct Memory Access) 방식으로 상기 메모리에 액세스하는,

인터커넥트 장치.

청구항 19

제1항에 있어서,

상기 프로세싱 코어는

CPU(Central Processing Unit), GPU(Graphic Processing Unit), 및 NPU(Neural Processing Unit) 중 어느 하나를 포함하는,

인터커넥트 장치.

청구항 20

제1항에 있어서,

상기 메모리는

SRAM(Static Random Access Memory) 및 DRAM(Dynamic Random Access Memory), 플래시 메모리(flash memory) 중 어느 하나를 포함하는,

인터커넥트 장치.

청구항 21

복수의 프로세싱 코어들(Cores), 복수의 인터커넥트 장치들, 및 메모리를 포함하는 AI 가속기 시스템에 있어서,

상기 복수의 인터커넥트 장치들은 상기 복수의 프로세싱 코어들 및 상기 메모리와 연결되고,

상기 복수의 인터커넥트 장치들 중 어느 하나의 인터커넥트 장치는

상기 복수의 프로세싱 코어들 중 적어도 하나의 프로세싱 코어로부터 명령을 수신하고, 상기 명령을 기초로, 상기 메모리에 저장된 데이터들에 대한 누적 연산 및 상기 적어도 하나의 프로세싱 코어에서 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행하며, 상기 연산 결과를 상기 메모리 또는 상기 적어도 하나의 프로세싱 코어에게 제공하는,

AI 가속기 시스템.

청구항 22

프로세싱 코어로부터 명령을 수신하는 단계;

상기 명령을 기초로, 메모리에 저장된 데이터들에 대한 누적 연산 및 상기 프로세싱 코어에서 분산 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행하는 단계; 및

상기 연산 결과를 전송하는 단계

를 포함하는,

인터커넥트 장치의 동작 방법.

청구항 23

제22항에 있어서,

상기 명령은

상기 누적 연산 및 상기 집합 연산 각각을 위한 오퍼레이션 코드와

상기 오퍼레이션 코드에 따른 동작을 수행하기 위한 데이터들이 저장된 어드레스 정보; 및

상기 데이터들이 저장된 상기 메모리의 어드레스 정보

중 어느 하나를 포함하는,

인터커넥트 장치의 동작 방법.

청구항 24

하드웨어와 결합되어 제22항 내지 제23항 중 어느 하나의 항의 방법을 실행시키기 위하여 컴퓨터 판독 가능한 기록매체에 저장된 컴퓨터 프로그램.

발명의 설명

기 술 분 야

[0001] 아래의 실시예들은 인터커넥트 장치, 인터커넥트 장치의 동작 방법 및 인터커넥트 장치를 포함하는 AI(Artificial Intelligence) 가속기 시스템에 관한 것이다.

배 경 기 술

[0002] 인공 지능(Artificial Intelligence; AI) 기술이 발전함에 따라 인공 지능만을 위한 독자적인 하드웨어의 필요성이 증가하고 있다. 인공 지능은 예를 들어, 특정한 연산을 통해 추론과 학습을 수행할 수 있다. 이와 같이

인공 지능을 구현 하고 실행하기 위한 전용 하드웨어로서 다양한 장치들이 개발되고 있다.

[0003] 인공 지능을 위한 전용 하드웨어는 예를 들어, CPU(Central Processing Unit), GPU(Graphics Processing Unit) 등에 의해 구현될 수도 있고, 용도 변경이 가능한 FPGA(Field Programmable Gate Array), 및 ASIC(Application Specific Integrated Circuit) 등에 의해 구현될 수도 있다.

발명의 내용

해결하려는 과제

과제의 해결 수단

[0004] 일 실시예에 따르면, 인터커넥트 장치(interconnect device)는 프로세싱 코어(processing core)로부터 명령(command)을 수신하고, 상기 명령을 기초로, 메모리(memory)에 저장된 데이터들에 대한 누적 연산(accumulation operation) 및 상기 프로세싱 코어에서 처리된 결과들에 대한 집합 연산(aggregation operation) 중 적어도 하나의 연산을 수행하며, 상기 적어도 하나의 연산의 수행 결과를 제공한다.

[0005] 상기 명령은 상기 누적 연산 및 상기 집합 연산 각각을 위한 오퍼레이션 코드(operation code; OP code)와 상기 오퍼레이션 코드에 따른 동작을 수행하기 위한 데이터들이 저장된 어드레스 정보(address information); 및 상기 데이터들이 저장된 상기 메모리의 어드레스 정보 중 어느 하나를 포함할 수 있다.

[0006] 상기 인터커넥트 장치는 상기 프로세싱 코어로부터 수신한 명령을 저장 및 전달하는 커맨드 모듈; 상기 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달하는 어드레스 모듈; 및 상기 명령에 따라 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈을 포함할 수 있다.

[0007] 상기 읽기 데이터 모듈은 상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 읽기 데이터 모듈에 저장된 데이터를 합산하는 가산기(adder); 및 상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 합산한 데이터 중 어느 하나를 상기 읽기 데이터 모듈로 제공하는 멀티플렉서(MUX)를 포함할 수 있다.

[0008] 상기 읽기 데이터 모듈은 상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 읽기 데이터 모듈에 저장된 데이터를 곱하는 곱셈기(multiplier)를 더 포함할 수 있다.

[0009] 상기 인터커넥트 장치는 상기 명령을 기초로 상기 커맨드 모듈에게 제어 신호를 제공하고, 상기 프로세싱 코어로부터 수신한 어드레스 정보에 기초한 메모리의 어드레스를 상기 어드레스 모듈에 제공하는 제어 모듈을 더 포함하고, 상기 커맨드 모듈은 상기 제어 신호를 상기 읽기 데이터 모듈 및 상기 메모리에 전송할 수 있다.

[0010] 상기 제어 모듈은 상기 어드레스 정보에 따른 상기 메모리의 소스 어드레스(source adress)를 저장하는 레지스터(register); 상기 명령에 따라 상기 소스 어드레스에 기초한 카운팅(counting)을 수행하는 카운터 레지스터(counter register); 및 상기 카운팅 결과에 기초하여 결정한 상기 어드레스를 상기 어드레스 모듈로 제공하는 컨트롤러(controller)를 포함할 수 있다.

[0011] 상기 인터커넥트 장치는 상기 명령에 기초하여 제어 신호를 생성하고, 상기 프로세싱 코어로부터 수신한 어드레스 정보에 기초하여 메모리의 어드레스를 결정하는 제어 모듈; 상기 명령을 기초로, 상기 어드레스에 대응하는 신호를 생성하는 상기 어드레스 모듈; 상기 제어 모듈로부터 수신한 제어 신호를 상기 메모리로 전송하는 상기 커맨드 모듈; 및 상기 제어 신호에 기초하여 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈을 포함하고, 상기 읽기 데이터 모듈은 복수의 서브 데이터 모듈들을 포함하고, 상기 제어 신호에 기초하여, 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈에 저장된 누적 데이터- 상기 누적 데이터는 상기 메모리로부터 읽어온 데이터 및 상기 어느 하나의 서브 데이터 모듈의 데이터를 누적한 것임- 를 상기 프로세싱 코어에게 제공할 수 있다.

[0012] 상기 읽기 데이터 모듈은 상기 제어 신호에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈의 데이터를 합산하는 가산기; 상기 제어 신호에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 합산한 데이터 중 어느 하나의 데이터를 출력하는 제1 멀티플렉서(MUX); 상기 제어 신호에 따라, 상기 제1 멀티플렉서에서 출력된 데이터를 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서

브 데이터 모듈로 출력하는 디멀티플렉서(DEMUX); 및 상기 제어 신호에 기초하여, 상기 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈의 데이터를 출력하는 제2 멀티플렉서를 포함할 수 있다.

[0013] 상기 제어 모듈은 상기 어드레스 정보에 따른 상기 메모리의 소스 어드레스를 저장하는 레지스터; 상기 명령에 따라 상기 소스 어드레스에 기초한 카운팅을 수행하는 카운터 레지스터; 및 상기 카운팅 결과에 기초하여 결정한 상기 어드레스를 상기 어드레스 모듈로 제공하고, 상기 제어 신호를 상기 커맨드 모듈로 제공하는 컨트롤러를 포함할 수 있다.

[0014] 상기 인터커넥트 장치는 상기 명령에 기초하여 제어 신호를 생성하고, 상기 프로세싱 코어로부터 수신한 어드레스 정보에 기초하여 메모리의 어드레스를 결정하는 제어 모듈; 상기 명령을 기초로, 상기 어드레스에 대응하는 신호를 생성하는 상기 어드레스 모듈; 상기 제어 모듈로부터 수신한 제어 신호를 상기 메모리로 전송하는 상기 커맨드 모듈; 및 상기 제어 신호에 기초하여 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈을 포함하고, 상기 읽기 데이터 모듈은 상기 메모리부터 수신한 데이터를 누적한 누적 데이터를 저장하는 제1 서브 데이터 모듈; 및 상기 메모리로부터 읽어온 데이터를 저장하는 제2 서브 데이터 모듈을 포함하고, 상기 읽기 데이터 모듈은 상기 제어 신호에 기초하여, 상기 누적 데이터 및 상기 데이터 중 어느 하나의 데이터를 출력할 수 있다.

[0015] 상기 인터커넥트 장치는 상기 프로세싱 코어로부터 수신한 명령을 저장 및 전달하는 커맨드 모듈; 상기 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달하는 어드레스 모듈; 및 상기 명령에 따라 상기 프로세싱 코어에서 처리된 결과 데이터 또는 상기 프로세싱 코어로부터 수신한 처리된 결과 데이터들을 누적한 누적 데이터를 상기 메모리에 전송하는 쓰기 데이터 모듈을 포함할 수 있다.

[0016] 상기 쓰기 데이터 모듈은 상기 프로세싱 코어에서 처리된 결과 데이터 및 상기 메모리에 저장되는 데이터를 합산하는 가산기; 및 상기 명령에 기초하여 상기 프로세싱 코어에서 처리된 결과 데이터 및 상기 합산한 데이터 중 어느 하나의 데이터를 상기 메모리로 제공하는 멀티플렉서를 포함할 수 있다.

[0017] 상기 쓰기 데이터 모듈은 상기 합산한 데이터를 나누는 디바이더(divider) 또는 상기 합산한 데이터를 한 비트(bit) 씩 이동시키는 쉬프트 레지스터(shift register)

[0018] 를 더 포함할 수 있다.

[0019] 상기 인터커넥트 장치는 상기 프로세싱 코어로부터 수신한 명령을 저장 및 전달하는 커맨드 모듈; 상기 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달하는 어드레스 모듈; 상기 명령에 따라 상기 메모리로부터 읽어온 데이터 또는 상기 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 상기 프로세싱 코어에게 전송하는 읽기 데이터 모듈; 및 상기 명령에 따라 상기 프로세싱 코어로부터 수신한 데이터 또는 상기 프로세싱 코어로부터 수신한 데이터들을 누적한 누적 데이터를 상기 메모리에 전송하는 쓰기 데이터 모듈을 포함할 수 있다.

[0020] 상기 읽기 데이터 모듈은 상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 읽기 데이터 모듈에 저장된 데이터를 합산하는 가산기; 및 상기 명령에 따라, 상기 메모리로부터 읽어온 데이터 및 상기 합산한 데이터 중 어느 하나를 상기 읽기 데이터 모듈로 제공하는 멀티플렉서를 포함할 수 있다.

[0021] 상기 쓰기 데이터 모듈은 상기 프로세싱 코어로부터 수신한 데이터 및 상기 메모리에 저장되는 데이터를 합산하는 가산기; 및 상기 명령에 기초하여 상기 프로세싱 코어로부터 수신한 데이터 및 상기 합산한 데이터 중 어느 하나의 데이터를 상기 메모리로 제공하는 멀티플렉서를 포함할 수 있다.

[0022] 상기 인터커넥트 장치는 DMA(Direct Memory Access) 방식으로 상기 메모리에 액세스할 수 있다.

[0023] 상기 프로세싱 코어는 CPU(Central Processing Unit), GPU(Graphic Processing Unit), 및 NPU(Neural Processing Unit) 중 어느 하나를 포함할 수 있다.

[0024] 상기 메모리는 SRAM(Static Random Access Memory) 및 DRAM(Dynamic Random Access Memory), 및 플래시 메모리(flash memory) 중 어느 하나를 포함할 수 있다.

[0025] 일 실시예에 따르면, 복수의 프로세싱 코어들, 복수의 인터커넥트 장치들, 및 메모리를 포함하는 AI 가속기 시스템에서 상기 복수의 인터커넥트 장치들은 상기 복수의 프로세싱 코어들 및 상기 메모리와 연결되고, 상기 복수의 인터커넥트 장치들 중 어느 하나의 인터커넥트 장치는 상기 복수의 프로세싱 코어들 중 적어도 하나의 프로세싱 코어로부터 명령을 수신하고, 상기 명령을 기초로, 상기 메모리에 저장된 데이터들에 대한 누적 연산 및

상기 적어도 하나의 프로세싱 코어에서 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행하며, 상기 연산 결과를 상기 메모리 또는 상기 적어도 하나의 프로세싱 코어에게 제공한다.

[0026] 일 실시예에 따르면, 인터커넥트 장치의 동작 방법은 프로세싱 코어로부터 명령을 수신하는 단계; 상기 명령을 기초로, 메모리에 저장된 데이터들에 대한 누적 연산 및 상기 프로세싱 코어에서 분산 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행하는 단계; 및 상기 연산 결과를 전송하는 단계를 포함한다.

[0027] 상기 명령은 상기 누적 연산 및 상기 집합 연산 각각을 위한 오퍼레이션 코드와 상기 오퍼레이션 코드에 따른 동작을 수행하기 위한 데이터들이 저장된 어드레스 정보; 및 상기 데이터들이 저장된 상기 메모리의 어드레스 정보 중 어느 하나를 포함할 수 있다.

도면의 간단한 설명

[0028] 도 1은 일 실시예에 따른 인터커넥트 장치를 포함하는 AI 가속기 시스템의 구조를 설명하기 위한 도면.

도 2 내지 도 7은 실시예들에 따른 인터커넥트 장치의 구조를 설명하기 위한 도면들.

도 8은 일 실시예에 따른 AI 가속기 시스템 자원의 이용(Utilization)과 레이턴시(Latency) 간의 관계를 설명하기 위한 그래프.

도 9는 일 실시예에 따라 분산 처리에 참여하는 프로세싱 코어들 간의 브로드캐스트(broadcast) 동작을 설명하기 위한 도면.

도 10은 일 실시예에 따른 평면 구조(flat structure)의 AI 가속기 시스템을 도시한 도면.

도 11은 일 실시예에 따른 계층적 구조(hierarchical structure)의 AI 가속기 시스템을 도시한 도면.

도 12는 일 실시예에 따른 딥 러닝 추천 모델(Deep Learning Recommendation Model)을 도시한 도면.

도 13은 일 실시예에 따른 인터커넥트 장치의 동작 방법을 나타낸 흐름도.

발명을 실시하기 위한 구체적인 내용

[0029] 이하에서, 첨부된 도면을 참조하여 실시예들을 상세하게 설명한다. 그러나, 특허출원의 범위가 이러한 실시예들에 의해 제한되거나 한정되는 것은 아니다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0030] 아래 설명하는 실시예들에는 다양한 변경이 가해질 수 있다. 아래 설명하는 실시예들은 실시 형태에 대해 한정하려는 것이 아니며, 이들에 대한 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.

[0031] 실시예에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 실시예를 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 명세서 상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0032] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 실시예가 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다. 이하, 실시예들을 첨부된 도면을 참조하여 상세하게 설명한다. 각 도면에 제시된 동일한 참조 부호는 동일한 부재를 나타낸다.

[0034] 도 1은 일 실시예에 따른 인터커넥트 장치를 포함하는 AI 가속기 시스템의 구조를 설명하기 위한 도면이다. 도 1을 참조하면, 복수의 프로세싱 코어들(110, 140, 170), 복수의 인터커넥트 장치들(120,150,180), 및 메모리(130,160,190)를 포함하는 AI 가속기 시스템(100)이 도시된다.

[0035] 일 실시예에 따른 AI 가속기 시스템(100)에 포함된 복수의 인터커넥트 장치들(120,150,180)에는 복수의 프로세싱 코어들(110, 140, 170)과 메모리(130, 160, 190) 간의 데이터가 경유될 수 있다. 복수의 프로세싱 코어들(110, 140, 170)은 예를 들어, CPU(Central Processing Unit), GPU(Graphic Processing Unit), 및 NPU(Neural

Processing Unit) 중 어느 하나를 포함할 수 있다. 메모리(130,160,190)는 예를 들어, SRAM(Static Random Access Memory), DRAM(Dynamic Random Access Memory), 및 플래시 메모리(flash memory) 중 어느 하나를 포함할 수 있다.

[0036] AI 가속기 시스템(100)은 예를 들어, SoC(System on Chip)의 형태로 구성될 수도 있고, 하나의 시스템으로 구성될 수도 있다.

[0037] 복수의 인터커넥트 장치들(120,150,180)은 복수의 프로세싱 코어들(110, 140, 170) 사이를 연결하는 버스 인터커넥트(Bus Interconnect)와 같은 버스 컴포넌트 또는 프로세싱 코어들(110, 140, 170)과 메모리(130, 160, 190) 사이를 연결하는 메모리 컨트롤러(Memory controller)와 같은 인터커넥트 컴포넌트에 해당할 수 있다. 복수의 인터커넥트 장치들(120,150,180)은 예를 들어, DMA(Direct Memory Access) 방식으로 메모리(130, 160, 190)에 액세스할 수 있다.

[0038] 일 실시예에 따른 프로세싱 코어들(110, 140, 170)은 예를 들어, MAC(Multiplication and ACcumulation) 연산이 아니라, 가산기(Adder)와 멀티플렉서(MUX)에 의한 연산을 통해 버스(Bus) 및 메모리 컨트롤러의 기능을 유지 및/또는 확장함으로써 저밀도(Sparse) 고용량의 메모리 액세스 특성을 가지는 AI 연산에 최적화될 수 있다.

[0039] 복수의 인터커넥트 장치들(120,150,180)은 복수의 프로세싱 코어들(110, 140, 170)로부터 예를 들어, 메모리(130, 160, 190)에 저장된 데이터들에 대한 누적 연산(accumulation operation) 및 적어도 하나의 프로세싱 코어에서 처리된 결과들에 대한 집합 연산(aggregation operation) 등과 같은 연산 기능을 수행할 수 있는 명령(command)을 수신할 수 있다.

[0040] 복수의 인터커넥트 장치들(120,150,180)은 복수의 프로세싱 코어들(110, 140, 170)로부터 메모리 액세스(Access)를 위한 요청(Request)을 수신함에 있어, 예를 들어, 메모리에 대한 읽기(Read) 및/또는 쓰기(Write) 신호에 더하여 추가적인 명령을 수신하거나, 또는 새로운 명령을 수신할 수 있다.

[0041] 일 실시예에 따른 명령은 예를 들어, 누적 연산 및 집합 연산 각각을 위한 오퍼레이션 코드(operation code; OP code), 및 오퍼레이션 코드에 따른 동작을 수행하기 위한 데이터들이 저장된 어드레스 정보(address information) 중 적어도 하나를 포함할 수 있다. 오퍼레이션 코드는 누적 연산의 수행을 위한 누적 연산 명령, 집합 연산의 수행을 위한 집합 연산 명령, 누적 연산에 수반되는 메모리에 대한 읽기 명령, 집합 연산에 수반되는 메모리에 대한 쓰기 명령, 데이터들을 저장하는 메모리에 대한 리셋(Reset) 명령; 및 데이터들을 저장하는 레지스터에 대한 리셋 명령 중 어느 하나 또는 이들의 조합에 대응할 수 있다. 실시예에 따라서, 오퍼레이션 코드는 누적 연산 또는 집합 연산의 수행 여부를 나타내는 플래그 비트(flag bit)를 포함할 수 있다. 어드레스 정보는 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스를 지시하는 정보를 포함할 수 있다. 어드레스를 지시하는 정보는 누적 연산의 수행을 위한 적어도 하나의 레지스터(register)의 인덱스(index)를 포함할 수 있다.

[0042] 복수의 인터커넥트 장치들(120,150,180)은 복수의 프로세싱 코어들(110, 140, 170)로부터 수신한 명령에 의해 메모리에 대한 액세스뿐만 아니라 다양한 연산을 수행할 수 있다.

[0043] 복수의 인터커넥트 장치들(120,150,180)이 수행하는 연산은 예를 들어, 추천(Recommendation) 시에 활용되는 SparseLengthSum과 같이 벡터(Vector)화된 데이터를 메모리(130, 160, 190)에서 읽어 요소 별 합산(element wise summation)을 수행하는 누적 연산(Accumulation operation)을 포함할 수 있다. 누적 연산은 요소 별 합산 이외에도 메모리(130, 160, 190)에 저장된 데이터들에 대한 요소 별 곱(element wise product), 및 가중합(weighted sum) 등과 같은 요소 별 연산(element wise operation)을 포함할 수 있다.

[0044] 또한, 복수의 인터커넥트 장치들(120,150,180)이 수행하는 연산은 예를 들어, 학습 과정에서 분산 처리된 그래디언트(Gradient)를 집합(Aggregation)하는 집합 연산을 포함할 수 있다. 복수의 인터커넥트 장치들(120,150,180)은 메모리 (130,160,190)에 쓰는 복수개의 그래디언트를 집합 연산을 통해 한번에 수행함으로써 메모리 액세스 횟수를 줄일 수 있다.

[0045] 실시예에 따라서, 복수의 프로세싱 코어들(110, 140, 170)은 서로 연결되어 하나의 시스템처럼 동작하는 클러스터(cluster)를 구성할 수 있다. 복수의 프로세싱 코어들(110, 140, 170)이 하나의 클러스터를 구성하는 경우, 예를 들어, 프로세싱 코어(140)는 주 코어(master core)로 동작하고, 나머지 프로세싱 코어들(110,170)은 종속 코어(slave core)로 동작할 수 있다. 주 코어로 동작하는 프로세싱 코어(140)는 종속 코어로 동작하는 프로세싱 코어들(110,170)에서 분산 처리된 결과들에 대한 집합 연산을 수행하거나, 또는 프로세싱 코어들(110,170)에게 동일한 데이터를 전달할 수 있다. 프로세싱 코어(140)는 예를 들어, 브로드캐스팅(broadcasting) 명령에 의

해 클러스터에 포함된 복수의 프로세싱 코어들 각각에게 동일한 데이터를 전달할 수 있다.

[0046] 이 때, 주 코어로 동작하는 프로세싱 코어(140)와 종속 코어로 동작하는 프로세싱 코어들(110, 170)은 서로 인터커넥트 장치를 통해 연결될 수 있다. 일 예로, 인터커넥트 장치는 종속 코어로 동작하는 프로세싱 코어들(110, 170)에서 분산 처리된 결과들을 수신하여, 집합 연산을 수행한 뒤, 그 결과를 주 코어로 동작하는 프로세싱 코어(140)에 전달할 수 있다. 또는, 인터커넥트 장치는 프로세싱 코어(140)로부터 수신되는 브로드캐스팅 명령에 반응하여, 프로세싱 코어(140)로부터 수신되는 데이터를 종속 코어로 동작하는 프로세싱 코어들(110, 170)로 전파할 수 있다.

[0047] 이러한 동작은 예를 들어, 추천 시나리오에 의해 복수의 프로세싱 코어들(110, 140, 170)에서 병렬적으로 추론 작업을 수행하고, 그 결과에 대한 집합 연산을 수행하는 경우, 및/또는 복수의 프로세싱 코어들(110, 140, 170) 각각이 서로 다른 사용자들에 대응하는 데이터를 처리하는 경우에 수행될 수 있다.

[0048] 일 실시예에 따른 복수의 인터커넥트 장치들(120,150,180)은 복수의 프로세싱 코어들(110, 140, 170)로부터 수신한 명령(들)에 의해 메모리 칩의 구조 변경없이, 메모리들(130, 160, 190)에 저장된 데이터들에 대한 누적 연산 및 프로세싱 코어들(110, 140, 170)에서 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행하고, 연산의 수행 결과를 메모리(130, 160, 190) 및/또는 프로세시 코어들(110, 140, 170)에게 제공할 수 있다.

[0049] 일 실시예에 따른 메모리들(130, 160, 190)은 예를 들어, 하나의 메모리를 구성하는 메모리 뱅크들(memory banks), 또는 하나의 메모리의 서브 세트들(sub sets)에 해당할 수 있다. 복수의 인터커넥트 장치들(120,150,180) 각각의 구조는 아래의 도 2 내지 도 7을 참조하여 구체적으로 설명한다.

[0051] 도 2는 일 실시예에 따른 누적 연산을 수행하는 인터커넥트 장치의 구조를 도시한 도면이다. 도 2를 참조하면, 일 실시예에 따른 인터커넥트 장치(120)는 읽기 데이터 모듈(210), 커맨드 모듈(220), 어드레스 모듈(230), 멀티플렉서(MUX)(240), 가산기(250), 데이터 포트(260), 제어 포트(270), 및 어드레스 포트(280)를 포함할 수 있다. 일 실시예에서 읽기 데이터 모듈(210), 커맨드 모듈(220), 및 어드레스 모듈(230)은 예를 들어, 큐(Queue)로 구성될 수 있다. 데이터 포트(260), 제어 포트(270), 및 어드레스 포트(280)는 예를 들어, 레지스터(register) 또는 버퍼(buffer)로 구성될 수 있다.

[0052] 인터커넥트 장치(120)는 누적 연산을 수행함에 있어 메모리에 대한 읽기 동작을 수반할 수 있다. 인터커넥트 장치(120)는 가산기(250)와 멀티플렉서(240)를 이용하여 읽기 데이터 모듈(210)에 저장된 데이터와 메모리에서 읽어와 데이터 포트(260)에 저장한 데이터에 대한 연산(예를 들어, 합산, 요소 별 합산, 가중합)을 수행한 후, 연산 결과를 프로세싱 코어로 전달할 수 있다.

[0053] 읽기 데이터 모듈(210)은 프로세싱 코어로부터 수신한 명령에 따라 메모리로부터 읽어온 데이터 또는 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 데이터 포트(260)를 통해 프로세싱 코어에게 전송할 수 있다. 이때, 읽기 데이터 모듈(210)은 멀티플렉서(MUX)(240) 및 가산기(adder)(250)를 포함할 수 있다.

[0054] 멀티플렉서(240)는 프로세싱 코어의 명령(예를 들어, select 신호)에 따라, 메모리로부터 읽어온 데이터 및 합산한 데이터 중 어느 하나를 읽기 데이터 모듈(210)로 제공할 수 있다. 가산기(250)는 명령에 따라, 메모리로부터 읽어온 데이터 및 읽기 데이터 모듈에 저장된 데이터를 합산할 수 있다. 가산기(250)의 합산 결과는 멀티플렉서(240)의 하나의 입력으로 인가될 수 있다. 예를 들어, 프로세싱 코어의 명령이 메모리로부터 읽어온 데이터를 출력하도록 하는 명령(select = 0)인 경우, 멀티플렉서(240)는 데이터 포트(260)를 통해 메모리로부터 읽어온 데이터를 읽기 데이터 모듈(210)로 전달할 수 있다. 이와 달리, 프로세싱 코어의 명령이 합산된 데이터를 출력하도록 하는 명령(select = 1)인 경우, 멀티플렉서(240)는 데이터 포트(260)를 통해 메모리로부터 읽어온 데이터와 읽기 데이터 모듈(210)에 저장된 데이터를 가산기(250)에서 합산한 결과를 읽기 데이터 모듈(210)로 전달할 수 있다. 예를 들어, 합산된 데이터를 출력하도록 하는 명령(select = 1)이 멀티플렉서(240)에 지속적으로 전달되는 경우, 읽기 데이터 모듈(210)은 멀티플렉서(240)로부터 출력되는 누적 합산된 데이터를 수신할 수 있다.

[0055] 커맨드 모듈(220)은 프로세싱 코어로부터 수신한 명령을 저장 및 전달할 수 있다. 커맨드 모듈(220)은 프로세싱 코어로부터 수신한 명령을 멀티플렉서(240)로 인가하는 한편, 제어 포트(270)를 통해 메모리로 전달할 수 있다.

[0056] 어드레스 모듈(230)은 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달할 수 있다. 어드레스 모듈(230)은 프로세싱 코어로부터 수신한 어드레스 정보를 어드레스 포트(280)를 통해 메모리로 전달할 수 있다. 이때, 어드레스 정보는 메모리의 로우(Row) 및 컬럼(Column) 어드레스를 나타낼 수 있다.

[0057] 데이터 포트(260), 제어 포트(270), 및 어드레스 포트(280) 각각은 레지스터로 구성될 수 있다. 예를 들어, CPU와 같은 프로세싱 코어는 자체적으로 데이터를 저장할 방법이 없기 때문에 메모리로 직접 데이터를 전송할 수 없다. 때문에 연산을 위해서는 레지스터를 거쳐야 하며, 이를 위해 레지스터는 특정 주소를 가리키거나, 값을 읽어올 수 있다. 일 실시예에서 데이터 포트(260), 제어 포트(270), 및 어드레스 포트(280) 각각은 메모리의 특정 주소를 가리키거나 특정 주소로부터 값을 읽어올 수 있다.

[0058] 실시예에 따라서, 읽기 데이터 모듈(210)은 프로세싱 코어로부터 수신한 명령에 따라, 메모리로부터 읽어온 데이터 및/또는 읽기 데이터 모듈(210)에 저장된 데이터를 곱하는 곱셈기(multiplier)(미도시)를 더 포함할 수도 있다. 이 경우, 곱셈기는 예를 들어, 메모리로부터 읽어온 데이터와 해당 데이터를 트랜스포즈한(transposed) 데이터를 곱할 수 있다.

[0060] 도 3은 다른 실시예에 따른 누적 연산을 수행하는 인터커넥트 장치의 구조를 도시한 도면이다. 도 3을 참조하면, 일 실시예에 따른 인터커넥트 장치(120)는 제어 모듈(310), 읽기 데이터 모듈(320), 커맨드 모듈(330), 어드레스 모듈(340), 멀티플렉서(MUX)(350), 가산기(360), 데이터 포트(370), 제어 포트(380), 및 어드레스 포트(390)를 포함할 수 있다.

[0061] 제어 모듈(310)은 프로세싱 코어로부터 수신한 명령을 기초로 커맨드 모듈(330)에게 제어 신호를 제공하고, 프로세싱 코어로부터 수신한 어드레스 정보에 기초한 메모리의 어드레스를 어드레스 모듈(340)에 제공할 수 있다. 제어 모듈(310)은 예를 들어, 컨트롤러(controller)(311), 카운터 레지스터(counter register)(313), 및 주소 저장 레지스터(register)(315)를 포함할 수 있다.

[0062] 컨트롤러(311)는 카운터 레지스터(313)의 카운팅 결과에 기초하여 결정한 어드레스를 어드레스 모듈(340)로 제공할 수 있다. 컨트롤러(311)는 예를 들어, 데이터 전송 시작 명령을 내릴 수 있는 비트(bit)나, 어떤 전송 방식으로 데이터를 전송할 것인지를 정해줄 수 있는 비트 등을 정의하는 컨트롤 레지스터(control register)를 포함할 수 있다. 전송 방식은 예를 들어, 단일 어드레스 모드(Single Address Mode)와 버스트 어드레스 모드(Burst Address Mode)를 포함할 수 있다.

[0063] 단일 어드레스 모드에 따르면, 컨트롤러(311)는 메모리의 데이터를 한번에 읽고 쓸 수 있다. 이때, 메모리에서 데이터를 읽어오는 주소는 소스 어드레스에 해당하고, 메모리에 데이터를 쓰는 주소는 데스티네이션 어드레스에 해당할 수 있다. 컨트롤러(311)는 예를 들어, 카운터 레지스터(313)의 카운트 값을 1씩 줄여나가면서 카운터 값이 0이 될 때까지 메모리로부터 데이터를 읽어오거나 쓸 수 있다. 카운터 값이 0이 되면, 컨트롤러(311)는 프로세싱 코어에게 예를 들어, DMA_INT와 같은 인터럽스 신호를 전송할 수 있다.

[0064] 버스트 어드레스 모드에 따르면, 컨트롤러(311)는 어드레스 정보에 기초한 메모리의 어드레스에 따라 카운터 레지스터의 카운트 값을 1씩 줄여가면서, 시작부터 끝까지 메모리의 읽기 및 쓰기를 계속적으로 반복할 수 있다. 컨트롤러(311)는 단일 어드레스 모드와 마찬가지로 카운터 값이 0이 되면, 프로세싱 코어에게 인터럽스 신호를 전송할 수 있다.

[0065] 카운터 레지스터(313)는 프로세싱 코어로부터 수신한 명령에 따라 소스 어드레스에 기초한 카운팅(counting)을 수행할 수 있다. 주소 저장 레지스터(315)는 프로세싱 코어로부터 수신한 어드레스 정보에 따른 메모리의 소스 어드레스(source address)를 저장할 수 있다.

[0066] 커맨드 모듈(330)은 제어 모듈(310)로부터 수신한 제어 신호를 읽기 데이터 모듈(320)로 전송하는 한편, 포트(380)를 통해 메모리에 전송할 수 있다.

[0067] 그 밖의 읽기 데이터 모듈(320), 어드레스 모듈(340), 멀티플렉서(350), 가산기(360), 데이터 포트(370), 제어 포트(380), 및 어드레스 포트(390)의 동작은 도 2를 통해 전술한 읽기 데이터 모듈(210), 어드레스 모듈(230), 멀티플렉서(240), 가산기(250), 데이터 포트(260), 제어 포트(270), 및 어드레스 포트(280)의 동작과 동일하므로 해당 부분의 설명을 참조하기로 한다.

[0069] 도 4는 다른 실시예에 따른 누적 연산을 수행하는 인터커넥트 장치의 구조를 도시한 도면이다. 도 4를 참조하면, 일 실시예에 따른 인터커넥트 장치(120)는 제어 모듈(410), 제2 멀티플렉서(420), 읽기 데이터 모듈(430), 디멀티플렉서(435), 제1 멀티플렉서(440), 가산기(443), 데이터 포트(445), 커맨드 모듈(450), 어드레스 모듈(460), 제어 포트(470), 및 어드레스 포트(480)를 포함할 수 있다.

[0070] 제어 모듈(410)은 프로세싱 코어의 명령에 기초하여 제어 신호를 생성하고, 프로세싱 코어로부터 수신한 어드레스 정보에 기초하여 메모리의 어드레스를 결정할 수 있다. 도 3의 제어 모듈(310)과 마찬가지로, 제어 모듈(410)은 컨트롤러(411), 카운터 레지스터(413), 및 주소 저장 레지스터(415)를 포함할 수 있다. 제어 모듈(410)의 각 구성 요소의 동작은 제어 모듈(310)의 각 구성 요소의 동작과 동일하므로 해당 부분의 설명을 참조하기 한다.

[0071] 어드레스 모듈(460)은 프로세싱 코어의 명령을 기초로, 어드레스에 대응하는 신호를 생성할 수 있다. 어드레스에 대응하는 신호는 어드레스 포트(480)를 통해 메모리로 전달될 수 있다.

[0072] 커맨드 모듈(450)은 제어 모듈(410)로부터 수신한 제어 신호를 메모리로 전송할 수 있다. 제어 신호는 제어 포트(470)를 통해 메모리로 전달될 수 있다.

[0073] 읽기 데이터 모듈(430)은 제어 모듈(410)로부터 수신한 제어 신호에 기초하여 메모리로부터 읽어온 데이터 또는 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 프로세싱 코어에게 전송할 수 있다. 읽기 데이터 모듈(430)은 복수의 서브 데이터 모듈들(431, 433, 435, 437)을 포함할 수 있다. 이때, 서브 데이터 모듈들(431, 433, 435, 437) 각각에는 예를 들어, 도 4에 도시된 것과 같이 0, 1, 2, 3과 같은 인덱스가 부여될 수 있다.

[0074] 읽기 데이터 모듈(430)은 제어 모듈(410)로부터 수신한 제어 신호에 기초하여, 복수의 서브 데이터 모듈들(431, 433, 435, 437) 중 어느 하나의 서브 데이터 모듈에 저장된 누적 데이터를 프로세싱 코어에게 제공할 수 있다. 여기서, 누적 데이터는 메모리로부터 읽어온 데이터 및 어느 하나의 서브 데이터 모듈의 데이터를 누적한 것일 수 있다.

[0075] 읽기 데이터 모듈(430)은 예를 들어, 제2 멀티플렉서(420), 디멀티플렉서(DEMUX)(435), 제1 멀티플렉서(440), 및 가산기(443)를 포함할 수 있다.

[0076] 가산기(443)는 제어 모듈(410)의 제어 신호에 따라, 메모리로부터 읽어온 데이터 및 복수의 서브 데이터 모듈들(431, 433, 435, 437) 중 어느 하나의 서브 데이터 모듈의 데이터를 합산할 수 있다.

[0077] 제1 멀티플렉서(440)는 제어 모듈(410)의 제어 신호에 따라, 메모리로부터 읽어온 데이터 및 합산한 데이터 중 어느 하나의 데이터를 출력할 수 있다.

[0078] 디멀티플렉서(DEMUX)(435)는 제어 모듈(410)의 제어 신호에 따라, 제1 멀티플렉서(440)에서 출력된 데이터를 복수의 서브 데이터 모듈들(431, 433, 435, 437) 중 어느 하나의 서브 데이터 모듈로 출력할 수 있다.

[0079] 제2 멀티플렉서(420)는 제어 모듈(410)의 제어 신호에 기초하여, 복수의 서브 데이터 모듈들 중 어느 하나의 서브 데이터 모듈의 데이터를 출력할 수 있다.

[0081] 도 5는 다른 실시예에 따른 누적 연산을 수행하는 인터커넥트 장치의 구조를 도시한 도면이다. 도 5를 참조하면, 일 실시예에 따른 인터커넥트 장치(120)는 제어 모듈(510), 제2 멀티플렉서(520), 읽기 데이터 모듈(530), 제1 멀티플렉서(540), 디멀티플렉서(550), 데이터 포트(560), 커맨드 모듈(570), 어드레스 모듈(580), 제어 포트(590), 및 어드레스 포트(595)를 포함할 수 있다. 도 5의 커맨드 모듈(570), 어드레스 모듈(580), 제어 포트(590), 및 어드레스 포트(595)의 동작은 도 4의 커맨드 모듈(450), 어드레스 모듈(460), 제어 포트(470), 및 어드레스 포트(480)의 동작과 동일하므로 해당 부분의 설명을 참조하기로 한다.

[0082] 제어 모듈(510)은 프로세싱 코어의 명령에 기초하여 제어 신호를 생성하고, 프로세싱 코어로부터 수신한 어드레스 정보에 기초하여 메모리의 어드레스를 결정할 수 있다.

[0083] 어드레스 모듈(580)은 명령을 기초로, 어드레스에 대응하는 신호를 생성하고, 어드레스 포트(590)를 통해 메모리로 전달할 수 있다.

[0084] 커맨드 모듈(570)은 제어 모듈(510)로부터 수신한 제어 신호를 제어 포트(590)를 통해 메모리로 전송할 수

있다.

[0085] 읽기 데이터 모듈(530)은 제어 모듈(510)로부터 수신한 제어 신호에 기초하여 메모리로부터 읽어온 데이터 또는 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 프로세싱 코어에게 전송할 수 있다. 이때, 읽기 데이터 모듈(530)은 예를 들어, 제1 서브 데이터 모듈(531) 및 제2 서브 데이터 모듈(533)을 포함할 수 있다. 제1 서브 데이터 모듈(531)은 메모리부터 수신한 데이터를 누적한 누적 데이터를 저장할 수 있다. 제2 서브 데이터 모듈(533)은 메모리로부터 읽어온 데이터를 저장할 수 있다.

[0086] 읽기 데이터 모듈(530)은 제어 모듈(510)로부터 수신한 제어 신호에 기초하여, 제1 서브 데이터 모듈(531)에 저장된 누적 데이터 및 제2 서브 데이터 모듈(533)에 저장된 데이터 중 어느 하나의 데이터를 출력할 수 있다.

[0087] 예를 들어, 메모리로부터 읽어온 데이터가 데이터 포트(560)를 통해 디멀티플렉서(550)로 입력되었다고 하자. 디멀티플렉서(550)는 제어 모듈(510)로부터 수신한 제어 신호에 따라 데이터를 제2 서브 데이터 모듈(533) 또는 제1 멀티플렉서(540)로 제공할 수 있다.

[0088] 예를 들어, 디멀티플렉서(550)는 제어 신호(select = 0)에 따라, 메모리로부터 읽어온 데이터를 제2 서브 데이터 모듈(533)로 제공할 수 있다.

[0089] 이와 달리, 디멀티플렉서(550)는 제어 모듈(510)로부터 수신한 제어 신호(예를 들어, select = 1)에 따라 데이터를 제1 멀티플렉서(540)로 전달할 수 있다. 이때, 제1 멀티플렉서(540)로 전달된 데이터는 제어 모듈(510)의 제어 신호에 따라 제1 멀티플렉서(540)를 통해 그대로 출력되거나, 또는 가산기(545)에 의해 제1 서브 데이터 모듈(531)에 저장된 데이터와 합산되어 출력될 수 있다. 예를 들어, 제어 신호(select = 0)에 따라, 제1 멀티플렉서(540)는 메모리로부터 읽어온 데이터를 그대로 제1 서브 데이터 모듈(531)로 제공할 수 있다. 이와 달리, 제어 신호(select = 1)에 따라, 제1 멀티플렉서(540)는 가산기(545)에 의해 합산된 데이터를 제1 서브 데이터 모듈(531)로 제공할 수 있다.

[0090] 이에 따라, 제1 서브 데이터 모듈(531)에는 합산된 데이터가 저장될 수도 있고, 또는 메모리로부터 읽어온 데이터가 저장될 수도 있다. 또한, 제2 서브 데이터 모듈(533)에는 메모리로부터 읽어온 데이터가 저장될 수 있다. 제2 멀티플렉서(520)는 제어 모듈(510)의 제어 신호에 따라 제1 서브 데이터 모듈(531)에 저장된 데이터를 출력할 수도 있고, 또는 제2 서브 데이터 모듈(533)에 저장된 데이터를 출력할 수도 있다.

[0091] 예를 들어, 디멀티플렉서(550)에 대한 제어 신호가 select = 1이고, 제1 멀티플렉서(540)에 대한 제어 신호가 select = 1인 경우, 제1 서브 데이터 모듈에는 합산된 데이터가 저장될 수 있다. 이때, 제2 멀티플렉서(520)에 대한 제어 신호가 select = 1이라면, 제2 멀티플렉서(520)는 프로세싱 코어에게 합산된 데이터를 제공할 수 있다.

[0093] 도 6은 일 실시예에 따른 집합 연산을 수행하는 인터커넥트 장치의 구조를 도시한 도면이다. 도 6을 참조하면, 일 실시예에 따른 인터커넥트 장치(600)는 쓰기 데이터 모듈(610), 커맨드 모듈(620), 어드레스 모듈(630), 가산기(640), 멀티플렉서(650), 데이터 포트(660), 제어 포트(670), 및 어드레스 포트(680)를 포함할 수 있다. 일 실시예에서 쓰기 데이터 모듈(610), 커맨드 모듈(620), 및 어드레스 모듈(630)은 예를 들어, 큐로 구성될 수 있다. 또한, 데이터 포트(660), 제어 포트(670), 및 어드레스 포트(680)는 예를 들어, 레지스터 또는 버퍼로 구성되 수 있다.

[0094] 일 실시예에 따른 인터커넥트 장치(600)는 집합 연산을 수행함에 있어 메모리에 대한 쓰기 동작을 수반할 수 있다. 쓰기 동작 시에, 인터커넥트 장치(600)는 가산기(640)와 멀티플렉서(650)를 이용하여 데이터 포트(660)에 저장된 데이터와 프로세싱 코어로부터 수신한 데이터에 대한 연산(예를 들어, 합산, 요소 별 합산, 가중합)을 수행한 후, 연산 결과를 데이터 포트(660)를 거쳐 메모리에 전달할 수 있다.

[0095] 쓰기 데이터 모듈(610)은 프로세싱 코어의 명령에 따라 프로세싱 코어에서 처리된 결과 데이터 또는 프로세싱 코어로부터 수신한 처리된 결과 데이터들을 누적한 누적 데이터를 메모리에 전송할 수 있다. 쓰기 데이터 모듈(610)은 가산기(640) 및 멀티플렉서(650)를 포함할 수 있다. 가산기(640)는 프로세싱 코어에서 처리된 결과 데이터 및 메모리에 저장되는 데이터를 합산할 수 있다. 멀티플렉서(650)는 프로세싱 코어의 명령에 기초하여 프로세싱 코어에서 처리된 결과 데이터 및 합산한 데이터 중 어느 하나의 데이터를 데이터 포트(660)를 통해 메모리로 제공할 수 있다.

[0096] 커맨드 모듈(620)은 프로세싱 코어로부터 수신한 명령을 저장 및 전달할 수 있다. 커맨드 모듈(620)은 프로세

싱 코어로부터 수신한 명령을 멀티플렉서(650) 및 데이터 포트(660)로 전달할 수 있다. 또한, 커맨드 모듈(620)은 제어 포트(670)를 통해 명령을 메모리로 전달할 수 있다.

[0097] 어드레스 모듈(630)은 프로세싱 코어로부터 수신한 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달할 수 있다. 어드레스 모듈(630)은 어드레스 정보를 어드레스 포트(680)를 통해 메모리로 전달할 수 있다.

[0098] 실시예에 따라서, 쓰기 데이터 모듈(610)은 합산한 데이터, 다시 말해 프로세싱 코어에서 처리된 결과 데이터 및 메모리에 저장되는 데이터를 합산한 데이터를 나누는 디바이더(divider) 또는 합산한 데이터를 한 비트(bit)씩 이동시키는 쉬프트 레지스터(shift register)를 더 포함할 수 있다.

[0099] 추론 과정과 비교할 때, 학습 과정은 메모리 대역폭(Band Width; BW) 요구량 및 연산량 많아 분산 처리가 일반적이지만, 그 과정에서 분산 처리된 그래디언트를 취합하는 과정이 성능 제약의 원인 중 하나가 될 수 있다. 일 실시예에서는 메모리의 쓰기 동작을 수행하는 과정에서 여러 프로세싱 코어들에서 처리된 결과를 취합하는 집합 연산을 별도의 간섭(Coherency) 장치 및/또는 동기화(Synchronization) 장치 없이 구현함으로써 데이터 트래픽 감소에 따라 레이턴시를 감소시키는 한편, 전력 소비 또한 감소시킬 수 있다. 또한, 일 실시예에 따르면, 프로세싱 코어에 공급되는 메모리 대역폭을 완화시켜 다른 연산에 활용할 수 있도록 함으로써 전체적인 시스템의 성능 또한 개선할 수 있다.

[0101] 도 7은 일 실시예에 따른 누적 연산 및 집합 연산을 수행하는 인터커넥트 장치의 구조를 도시한 도면이다. 도 7을 참조하면, 일 실시예에 따른 인터커넥트 장치(700)는 읽기 데이터 모듈(705), 가산기(715), 멀티플렉서(720), 읽기 데이터 포트(725), 쓰기 데이터 모듈(730), 가산기(735), 멀티플렉서(740), 쓰기 데이터 포트(745), 커맨드 모듈(750), 제어 포트(755), 어드레스 모듈(760), 및 어드레스 포트(765)를 포함할 수 있다.

[0102] 읽기 데이터 모듈(705)은 프로세싱 코어의 명령에 따라 읽기 데이터 포트(725)를 통해 메모리로부터 읽어온 데이터 또는 메모리로부터 읽어온 데이터들을 누적한 누적 데이터를 프로세싱 코어에게 전송할 수 있다. 읽기 데이터 모듈(705)은 가산기(715) 및 멀티플렉서(720)를 포함할 수 있다. 가산기(715)는 프로세싱 코어의 명령에 따라, 읽기 데이터 포트(725)를 통해 메모리로부터 읽어온 데이터 및 읽기 데이터 모듈에 저장된 데이터를 합산할 수 있다. 멀티플렉서(720)는 프로세싱 코어의 명령에 따라, 읽기 데이터 포트(725)를 통해 메모리로부터 읽어온 데이터 및 합산한 데이터 중 어느 하나를 읽기 데이터 모듈(705)로 제공할 수 있다.

[0103] 쓰기 데이터 모듈(730)은 프로세싱 코어의 명령에 따라 프로세싱 코어로부터 수신한 데이터 또는 프로세싱 코어로부터 수신한 데이터들을 누적한 누적 데이터를 쓰기 데이터 포트(745)를 통해 메모리에 전송할 수 있다. 쓰기 데이터 모듈(730)은 가산기(735) 및 멀티플렉서(740)를 포함할 수 있다. 가산기(735)는 프로세싱 코어로부터 수신한 데이터 및 메모리에 저장되는 데이터를 합산할 수 있다. 멀티플렉서(740)는 프로세싱 코어의 명령에 기초하여 프로세싱 코어로부터 수신한 데이터 및 합산한 데이터 중 어느 하나의 데이터를 쓰기 데이터 포트(745)를 통해 메모리로 제공할 수 있다.

[0104] 커맨드 모듈(750)은 프로세싱 코어로부터 수신한 명령을 저장 및 전달할 수 있다. 커맨드 모듈(750)은 프로세싱 코어로부터 수신한 명령을 읽기 데이터 모듈(705)과 멀티플렉서(720)로 전달하거나, 및/또는 쓰기 데이터 모듈(730) 및 멀티플렉서(740)로 전달할 수 있다. 또한, 커맨드 모듈(750)은 프로세싱 코어로부터 수신한 명령을 제어 포트(755)를 통해 메모리로 전달할 수 있다.

[0105] 어드레스 모듈(760)은 프로세싱 코어의 명령에 따른 동작을 수행하기 위한 데이터들이 저장된 메모리의 어드레스 정보를 저장 및 전달할 수 있다. 어드레스 모듈(760)은 메모리의 어드레스 정보를 어드레스 포트(765)를 통해 메모리로 전달할 수 있다.

[0106] 일 실시예에 따르면, 예를 들어, 추천과 같은 추론 과정에서 많은 횟수의 메모리 액세스를 통해 읽은 데이터를 처리하는 경우, 매번 프로세싱 코어가 메모리에 액세스하는 대신에 메모리 컨트롤러나 중간에 위치한 컴포넌트와 같은 인터커넥트 장치에서 직접 처리하고 연산 결과만을 프로세싱 코어로 전송함으로써 데이터 트래픽을 감소시킬 수 있다. 또한, 일 실시예에 따르면, 인터커넥트 장치는 학습 과정에서 분산 처리된 그래디언트를 해당 가중치(Weight)가 저장된 메모리 채널(Memory Channel)에 쓰는 과정에서 해당 쓰기 데이터 모듈에 저장된 그래디언트를 집합(aggregation) 하는 기능을 통해서 메모리 액세스 횟수를 줄일 수 있다. 인터커넥트 장치는 이를 통해 데이터 트래픽을 감소시키고, 레이턴시 감소에 따라 자원(resource)의 성능을 증가시키는 한편, 전력 또한

감소시킬 수 있다. 자원과 레이턴시 간의 관계는 아래의 도 8을 참조하여 설명한다.

[0108] 도 8은 일 실시예에 따른 AI 가속기 시스템 자원의 이용과 레이턴시 간의 관계를 설명하기 위한 그래프이다. 도 8의 그래프에서 X 축은 AI 가속기 시스템 자원(resource)의 이용(Utilization)에 해당하고, Y 축은 레이턴시(Latency)에 해당할 수 있다.

[0109] 도 8의 그래프에서 AI 가속기 시스템 자원의 이용과 레이턴시 간의 관계가 지수 함수의 형태를 나타내는 것을 볼 수 있다. 일 실시예에서는 인터커넥트 장치에 의해 프로세싱 코어의 액세스 횟수 및/또는 메모리의 액세스 횟수 등과 같은 AI 가속기 시스템 자원의 이용 횟수를 줄임으로써 레이턴시를 크게 감소시킬 수 있다. 또한, 인터커넥트 장치에 의한 레이턴시 감소에 따라 자원의 성능을 증가시키는 한편, 전력 또한 감소시킬 수 있다.

[0111] 도 9는 일 실시예에 따라 분산 처리에 참여하는 프로세싱 코어들 간의 브로드캐스트(broadcast) 동작을 설명하기 위한 도면이다. 도 9를 참조하면, 일 실시예에 따른 AI 가속기 시스템에 포함된 복수의 프로세싱 코어들 중 하나의 클러스터(cluster)를 구성하는 프로세싱 코어들(910, 920, 930, 940)이 도시된다.

[0112] 예를 들어, 연산을 수행하기 위한 데이터가 여러 메모리에 분산 저장되어 인터커넥트 장치에서 연산이 완결되지 못할 경우, 인터커넥트 장치(들)는 프로세싱 코어(들)로부터 새로운 명령 또는 메모리에 대한 읽기/쓰기 명령에 추가적인 명령을 수신할 수 있다. 인터커넥트 장치(들)는 새로운 명령 또는 추가적인 명령에 의해 읽기/쓰기와 같은 메모리 액세스뿐만 아니라 해당 연산의 중간 결과를 생성할 수 있다. 인터커넥트 장치(들)로부터 중간 결과를 수신한 AI 가속기 시스템의 다른 컴포넌트들은 중간 결과에 대한 추가 연산을 수행하여 최종 결과를 생성할 수 있다.

[0113] 예를 들어, 프로세싱 코어(910)가 클러스터의 주 코어이고, 프로세싱 코어들(920, 930, 940)이 프로세싱 코어(910)의 종속 코어들이라고 하자. 주 코어로 동작하는 프로세싱 코어(910)는 종속 코어들로 동작하는 프로세싱 코어들(920, 930, 940)에서 분산 처리된 결과들에 대한 집합 연산을 수행할 수 있다. 또한, 프로세싱 코어(910)는 클러스터에 포함된 프로세싱 코어들(920, 930, 940)과 분산 처리를 수행하는 경우, 예를 들어, 브로드캐스팅(broadcasting) 명령에 의해 클러스터에 포함된 복수의 프로세싱 코어들 각각에게 처리 결과를 전달할 수 있다.

[0114] 여기서, '브로드캐스팅 명령'은 예를 들어, 연산 결과를 분산 처리에 참여하는 프로세싱 코어들에게 브로드캐스팅하는 기능을 포함할 수 있다. 예를 들어, 추천 과정에서 SparseLengthSum 연산 결과는 브로드캐스팅 명령을 통해 이를 참조하는 클러스터의 복수의 프로세싱 코어들에게 동시에 전달될 수 있다. 또한, 학습 과정에서는 갱신된 파라미터(Parameter)는 브로드캐스팅 명령을 통해 분산 학습에 참여하는 복수의 프로세싱 코어들에게 동시에 전달될 수 있다.

[0116] 도 10은 일 실시예에 따른 평면 구조(flat structure)의 AI 가속기 시스템을 도시한 도면이다. 도 10을 참조하면, 일 실시예에 따른 AI 가속기 시스템(1000)의 구조가 도시된다. AI 가속기 시스템(1000)은 복수의 NPU 코어들(1010), 복수의 메모리 컨트롤러(1030), 및 복수의 메모리(1050)를 포함할 수 있다. 여기서, 복수의 NPU 코어들(1010)은 일 실시예에 따른 프로세싱 코어의 일 예시에 해당하며, CPU 코어들 또는 MPU 코어들이 NPU 코어들을 대체할 수 있다. 또한, 복수의 메모리 컨트롤러(1030)는 일 실시예에 따른 인터커넥트 장치의 일 예시에 해당하며, 버스 컴포넌트가 메모리 컨트롤러를 대체할 수도 있다.

[0117] 일 실시예에 따른 AI 가속기 시스템(1000)의 메모리 컨트롤러들(1030)은 복수의 NPU 코어들(1010) 중 적어도 하나의 프로세싱 코어로부터 수신한 명령을 기초로, 메모리들(1050)에 저장된 데이터들에 대한 누적 연산 및 적어도 하나의 프로세싱 코어에서 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행할 수 있다. 메모리 컨트롤러들(1030)은 연산 결과를 메모리들(1050) 또는 적어도 하나의 프로세싱 코어에게 제공할 수 있다.

[0118] 일 실시예에 따르면, AI 가속기 시스템(1000)은 평면 구조를 통해 DMA(Direct Memory Access) 장치와 연결되어 MAC 연산을 수행하는 복수의 NPU 코어들(1010)의 상위 계층의 SoC(System on Chip) 구성 요소들에 대한 추가적인 컴퓨팅 기능을 제공함으로써 분산 및/또는 병렬 추론 및 학습과정에서 집합 연산, 비선형 필터(Non Linear Filter; NFL) 함수, 및/또는 그래디언트 집합 갱신(Gradient Aggregation-Update) 등을 수행할 수 있다. 여기

서, 상위 계층의 SoC 구성 요소들은 예를 들어, CPU, 메모리 컨트롤러(Memory Controller), 네트워크 스위치(Network Switch), 라우터(Router) 또는 다른 레벨의 NPU 코어 등을 포함할 수 있다.

[0120] 도 11은 일 실시예에 따른 계층적 구조(Hierarchical structure)의 AI 가속기 시스템을 도시한 도면이다. 도 11을 참조하면, 복수의 NPU 코어들(1100), 메모리 컨트롤러(1130), 및 메모리(1150)를 포함하는 AI 가속기 시스템(1100)이 도시된다.

[0121] 실시예에 따라서, 복수의 NPU 코어들(1100), 메모리 컨트롤러(1130), 및 메모리(1150)는 평면 구조가 아니라, 각각이 서로 다른 계층을 구성하는 계층적 구조로 구성될 수 있다. 이 경우, 복수의 NPU 코어들(1100) 또한, 복수의 계층들로 구성된 계층적 구조로 구성될 수 있다.

[0123] 도 12는 일 실시예에 따른 딥 러닝 추천 모델(deep learning recommendation mode; DLRM)을 도시한 도면이다. 도 12를 참조하면, 일 실시예에 따른 딥 러닝 추천 모델(1200)이 도시된다.

[0124] 일 실시예에 따른 딥 러닝 추천 모델(1200)의 입력은 예를 들어, 밀도가 높고 성긴(sparse) 기능으로 구성될 수 있다. 도 12에서 도시된 딥 러닝 추천 모델(1200)의 첫번째 입력은 조밀한 특징들로서, 부동 소수점 값으로 구성된 벡터일 수 있다. 또한, 딥 러닝 추천 모델(1200)의 두번째 및 세번째 입력은 임베드 테이블(embed table)의 희소 색인(sparse indices) 목록으로서, 부동 소수점 값으로 구성된 벡터로 구성될 수 있다. 입력된 벡터는 삼각형으로 표시되는 MLP(Multilayer Perceptron) 네트워크로 전달될 수 있다. 경우에 따라 벡터는 연산자(Ops)를 통해 상호 작용할 수도 있다. 여기서, MLP 네트워크는 예를 들어, fully connected layers로 구성될 수 있다.

[0125] 딥 러닝 추천 모델(1200)은 예를 들어, p=[p1,...,pk])와 같은 성긴 색인 목록에 대해 z = Op(e1,...,ek)와 같은 embedding lookup을 수행하고, e1=E[:,p1], ..., ek=E[:,pk]와 같은 벡터를 획득할 수 있다. 여기서, 연산자는 Op는 Sum(e1,...,ek) = e1 + ... + ek, 또는 Dot(e1,...,ek) = [e1'e1 + ... + e1'ek + ... + ek'e1 + ...+ ek'ek]일 수 있다. '는 transpose operation을 나타낸다.

[0127] 도 13은 일 실시예에 따른 인터커넥트 장치의 동작 방법을 나타낸 흐름도이다. 도 13을 참조하면, 일 실시예에 따른 인터커넥트 장치는 프로세싱 코어로부터 명령을 수신한다(1310). 이때, 명령은 예를 들어, 누적 연산 및 집합 연산 각각을 위한 오퍼레이션 코드; 및 오퍼레이션 코드에 따른 동작을 수행하기 위한 데이터들이 저장된 어드레스 정보 중 적어도 하나를 포함할 수 있다.

[0128] 인터커넥트 장치는 단계(1310)에서 수신한 명령을 기초로, 메모리에 저장된 데이터들에 대한 누적 연산 및 프로세싱 코어에서 분산 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행한다(1320).

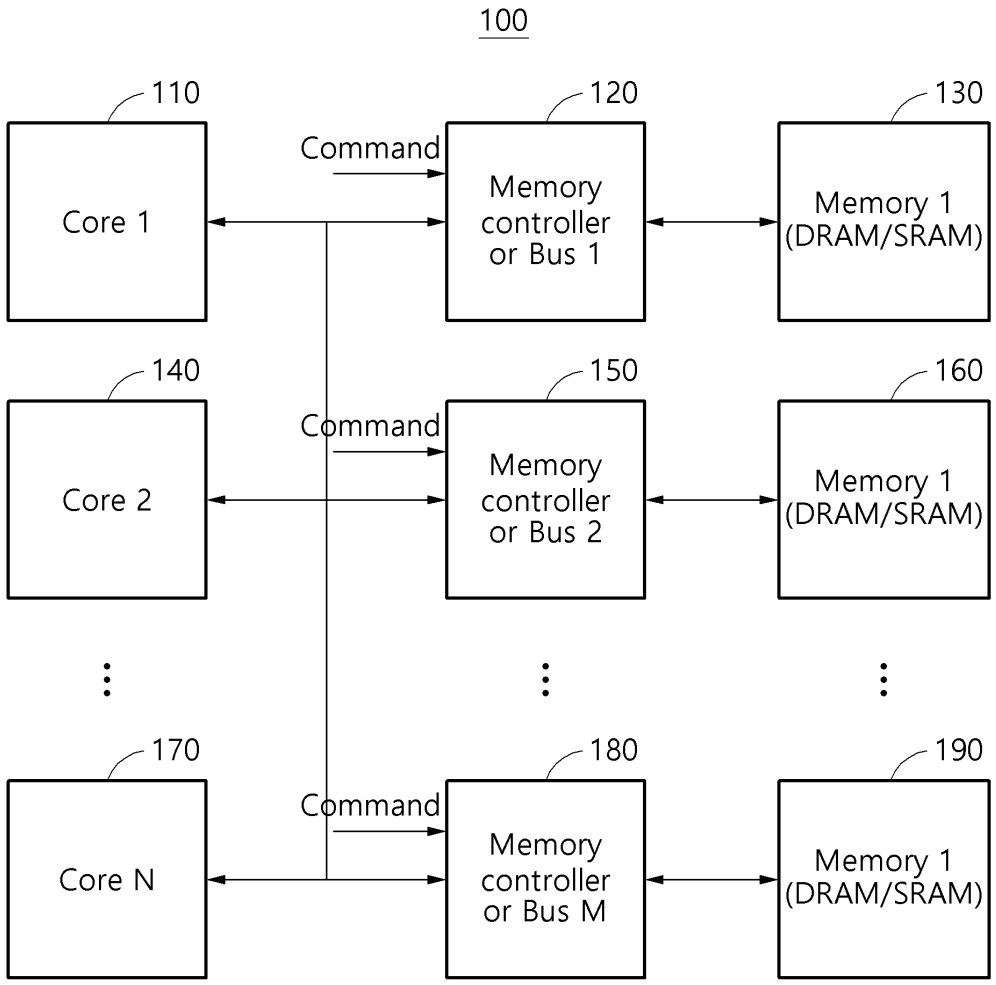
[0129] 인터커넥트 장치는 단계(1320)에서 수행한 연산 결과를 전송한다(1330). 인터커넥트 장치는 예를 들어, 연산 결과를 메모리, 및 해당 프로세싱 코어에게 전송할 수도 있고, 다른 프로세싱 코어 또는 다른 프로세싱 코어들로 전송할 수 있다.

[0131] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic media), CD-ROM, DVD와 같은 광기록 매체(optical media), 플롭티컬 디스크(floptical disk)와 같은 자기-광 매체(magneto-optical media), 및 롬(ROM), 램(RAM), 플래시 메모리 등과 같은 프로그램 명령을 저장하고 수행하도록 특별히 구성된 하드웨어 장치가 포함된다. 프로그램 명령의 예에는 컴파일러에 의해 만들어지는 것과 같은 기계어 코드뿐만 아니라 인터프리터 등을 사용해서 컴퓨터에 의해서 실행될 수 있는 고급 언어 코드를 포함한다. 상기된 하드웨어 장치는 실시예의 동작을 수행하기 위해 하나 이상의 소프트웨어 모듈로서 작동하도록 구성될 수 있으며, 그 역도 마찬가지이다.

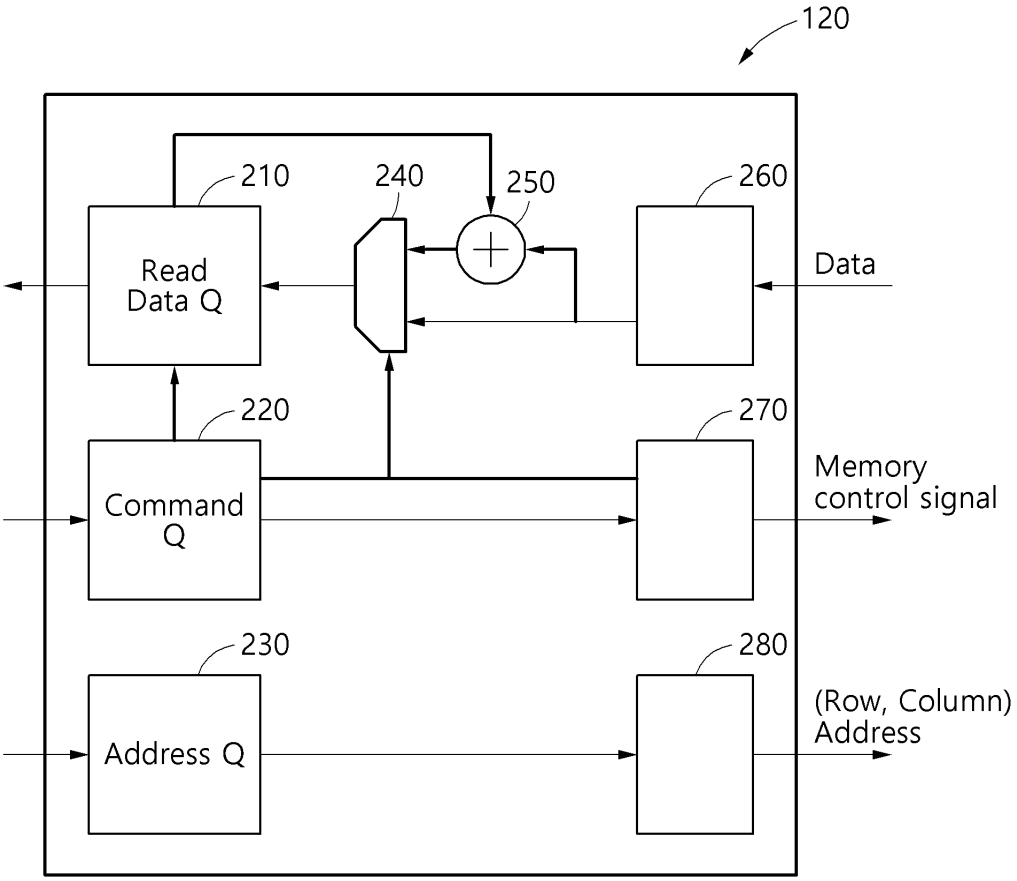
[0132] 이상과 같이 실시예들이 비록 한정된 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기를 기초로 다양한 기술적 수정 및 변형을 적용할 수 있다. 예를 들어, 설명된 기술들이 설명된 방법과 다른 순서로 수행되거나, 및/또는 설명된 시스템, 구조, 장치, 회로 등의 구성요소들이 설명된 방법과 다른 형태로 결합 또는 조합되거나, 다른 구성요소 또는 균등물에 의하여 대치되거나 치환되더라도 적절한 결과가 달성될 수 있다. 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 청구범위의 범위에 속한다.

도면

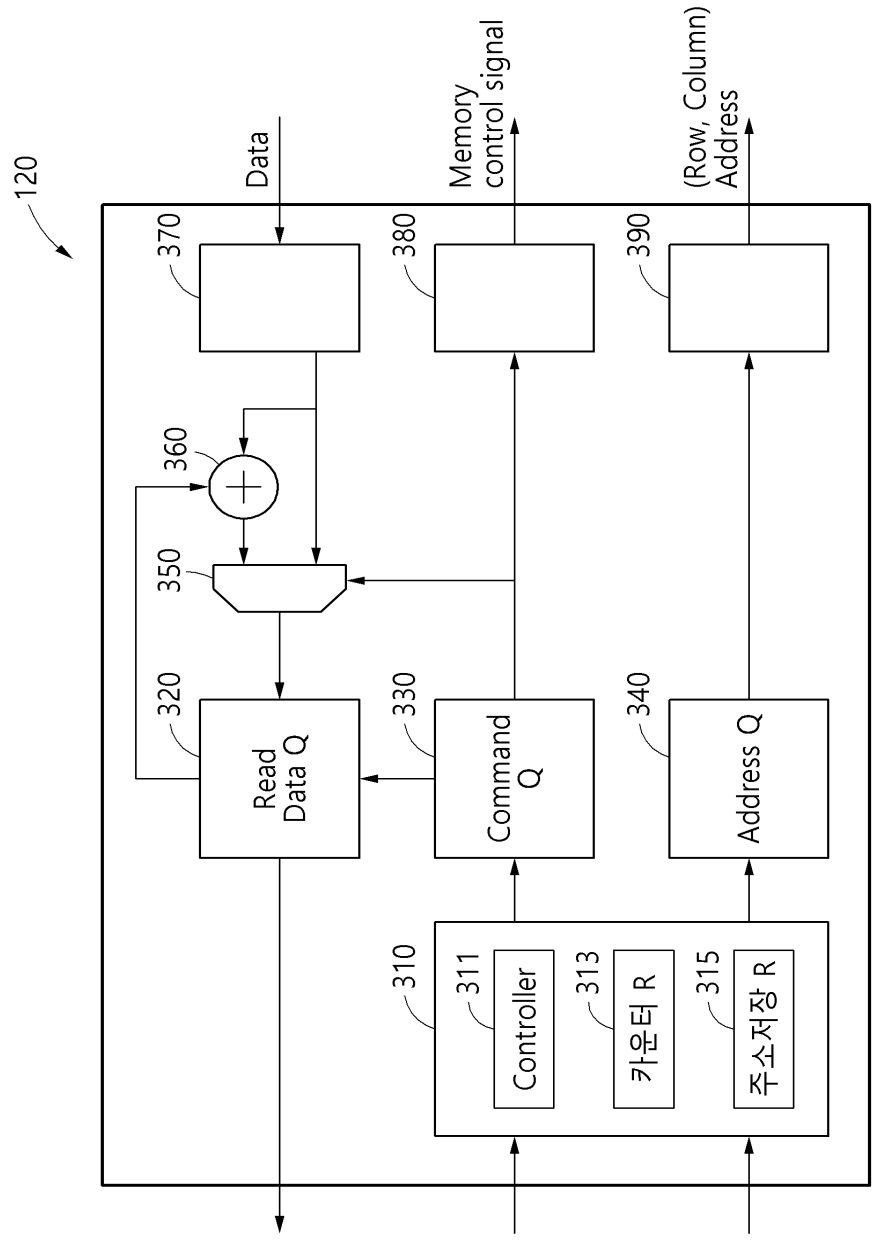
도면1



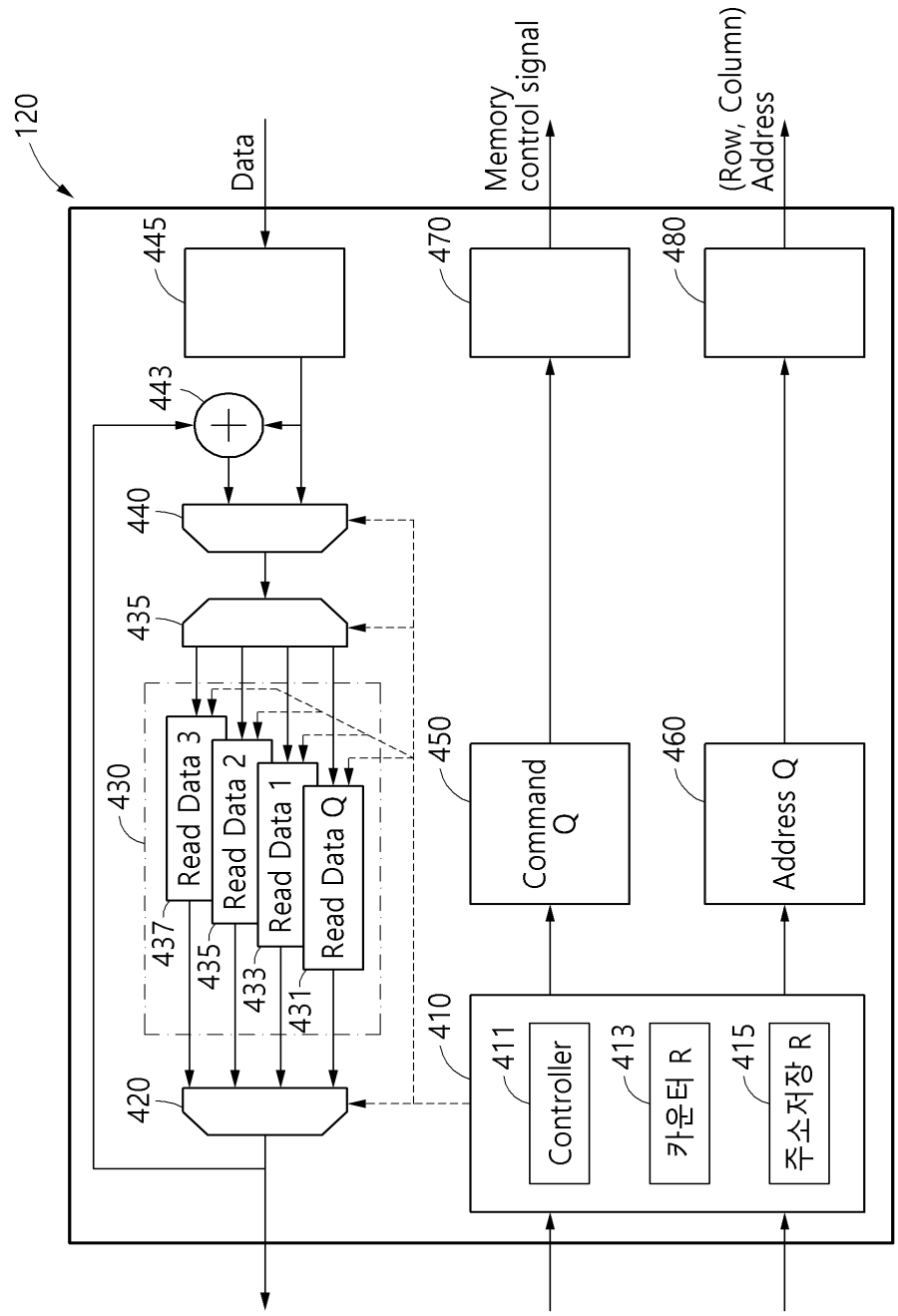
도면2


120
210
240
250
260
Read
Data Q
Data
220
270
Command
Q
Memory
control signal
230
280
Address Q
(Row, Column)
Address

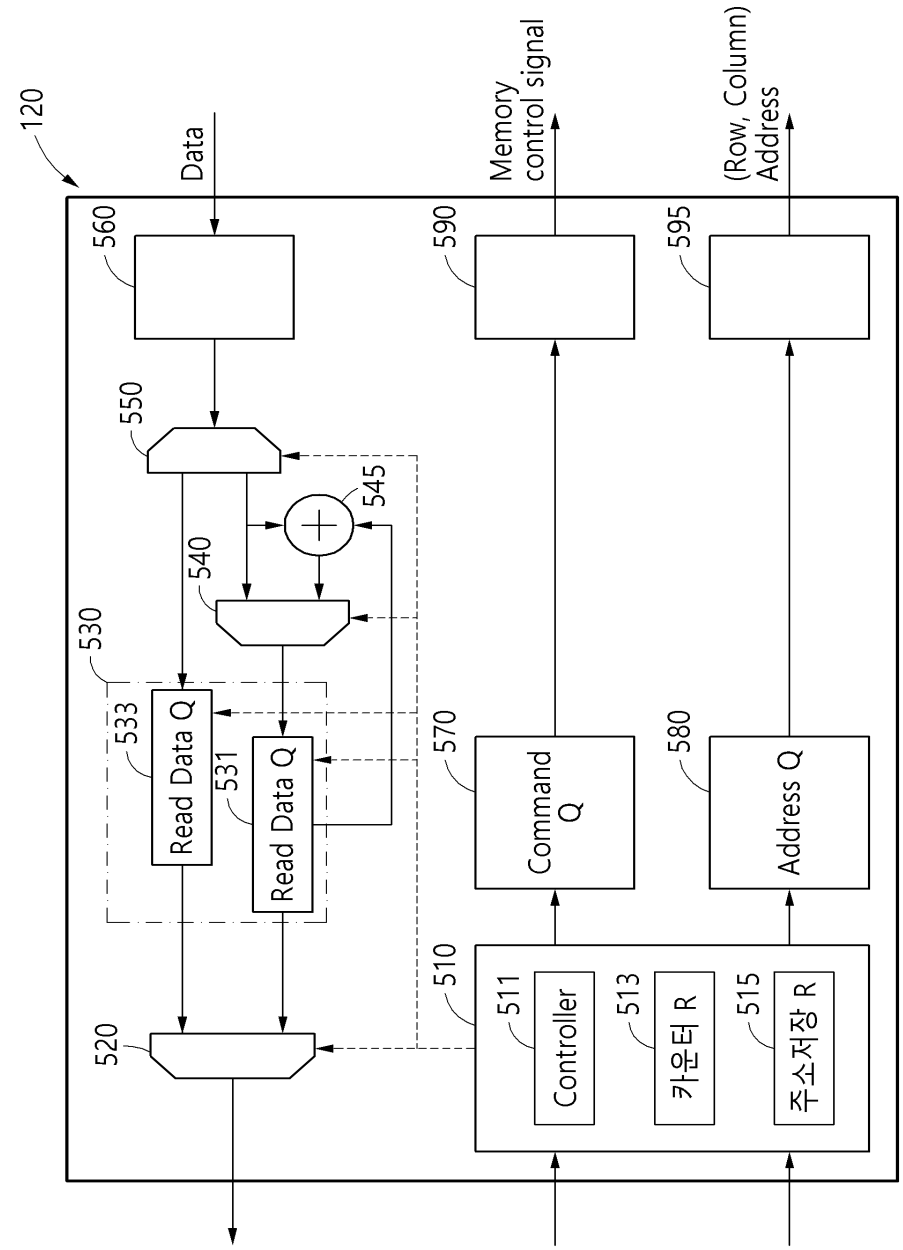
도면3


120
Data
Memory control signal
(Row, Column) Address
370
380
390
360
350
320
330
340
Read Data Q
Command Q
Address Q
310
311
Controller
313
카운터 R
315
주소저장 R

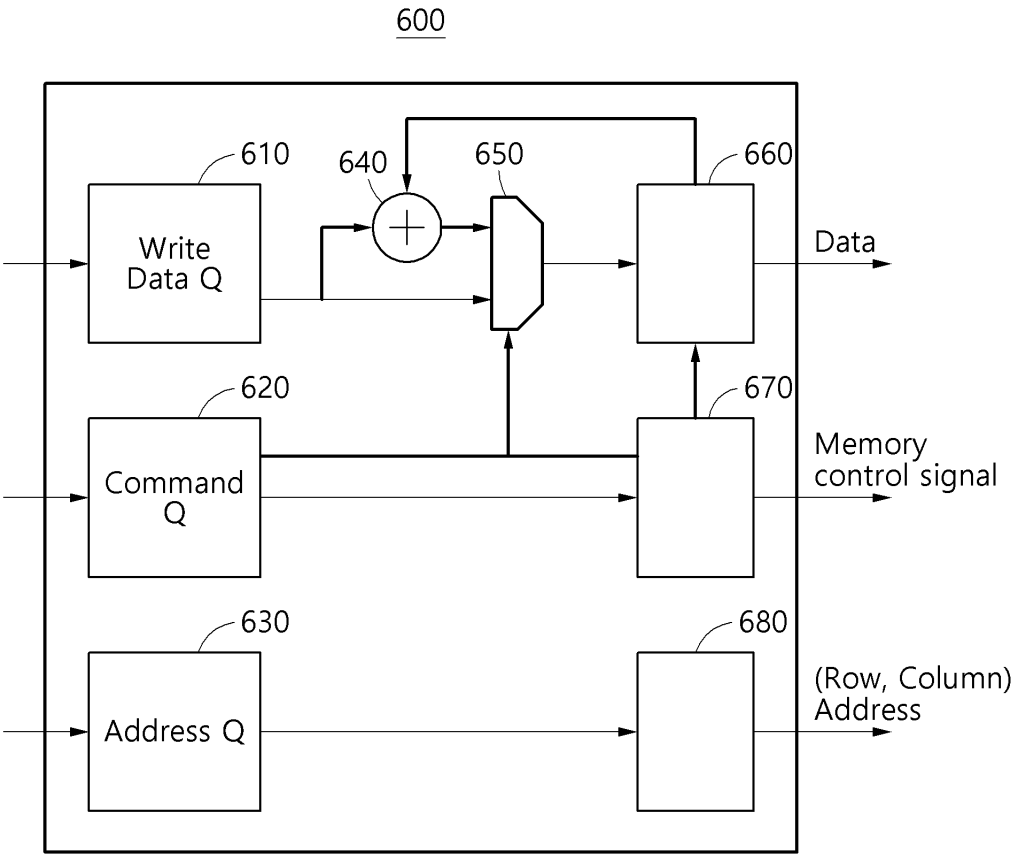
도면4


120
Data
Memory control signal
(Row, Column) Address
445
470
480
443
440
435
430
Read Data 3
Read Data 2
Read Data 1
Read Data Q
437
435
433
431
450
Command Q
460
Address Q
420
410
411
Controller
413
카운터 R
415
주소저장 R

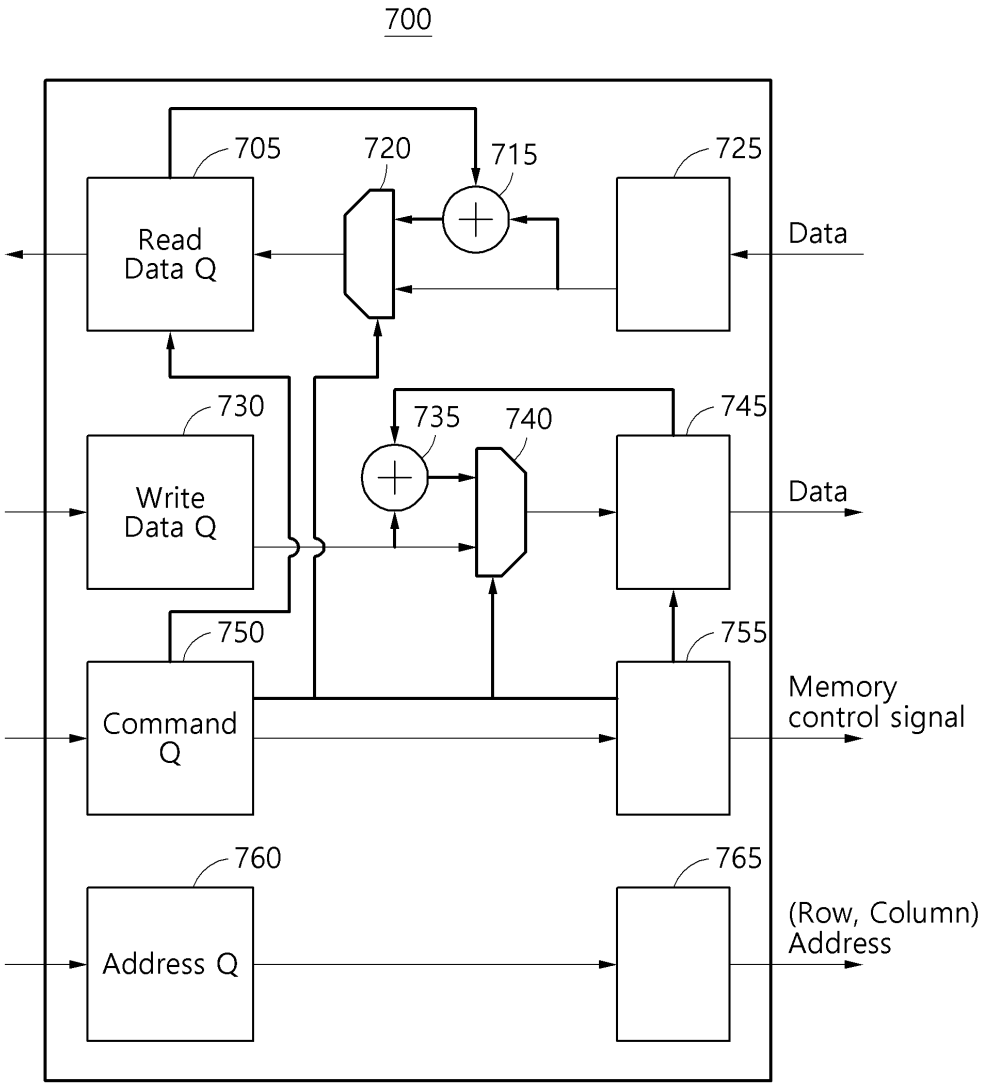
도면5


120
Data
Memory control signal
(Row, Column) Address
560
590
595
550
545
540
530
533
Read Data Q
531
Read Data Q
570
Command Q
580
Address Q
520
510
511
Controller
513
카운터 R
515
주소저장 R

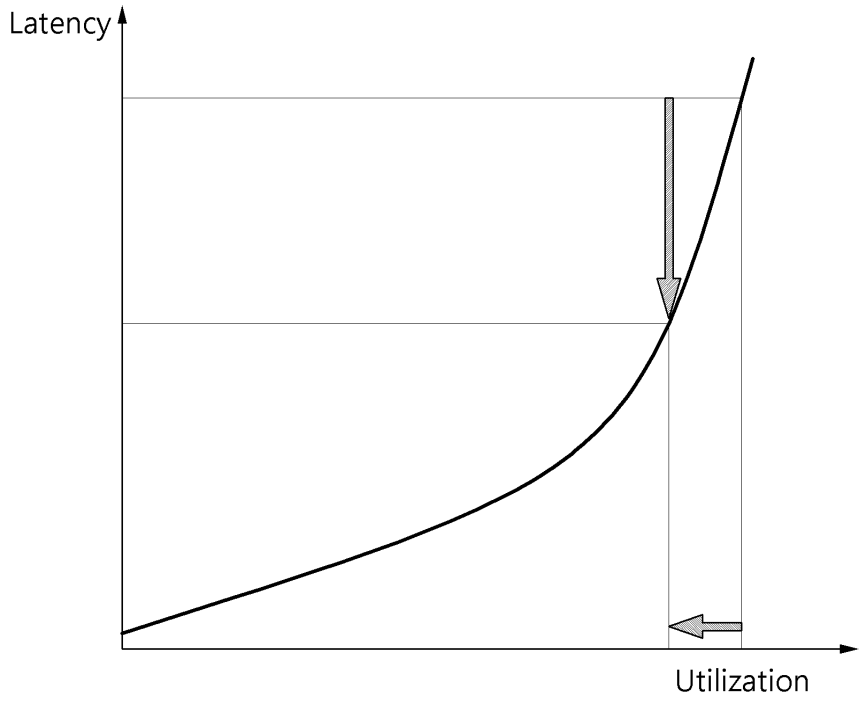
도면6


600
610
640
650
660
Write
Data Q
Data
620
670
Command
Q
Memory
control signal
630
680
Address Q
(Row, Column)
Address

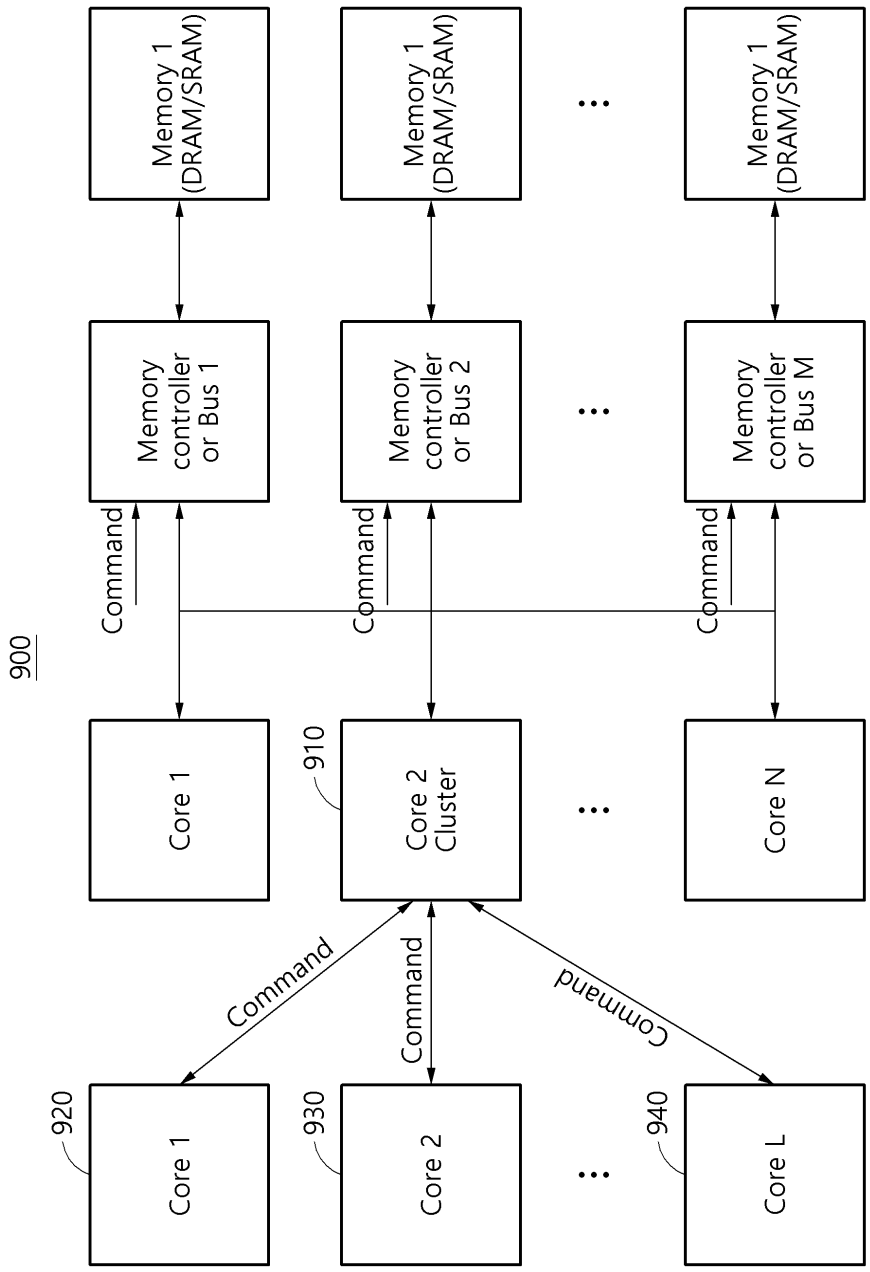
도면7


700
705
720
715
725
Read
Data Q
Data
730
735
740
745
Write
Data Q
Data
750
755
Command
Q
Memory
control signal
760
765
Address Q
(Row, Column)
Address

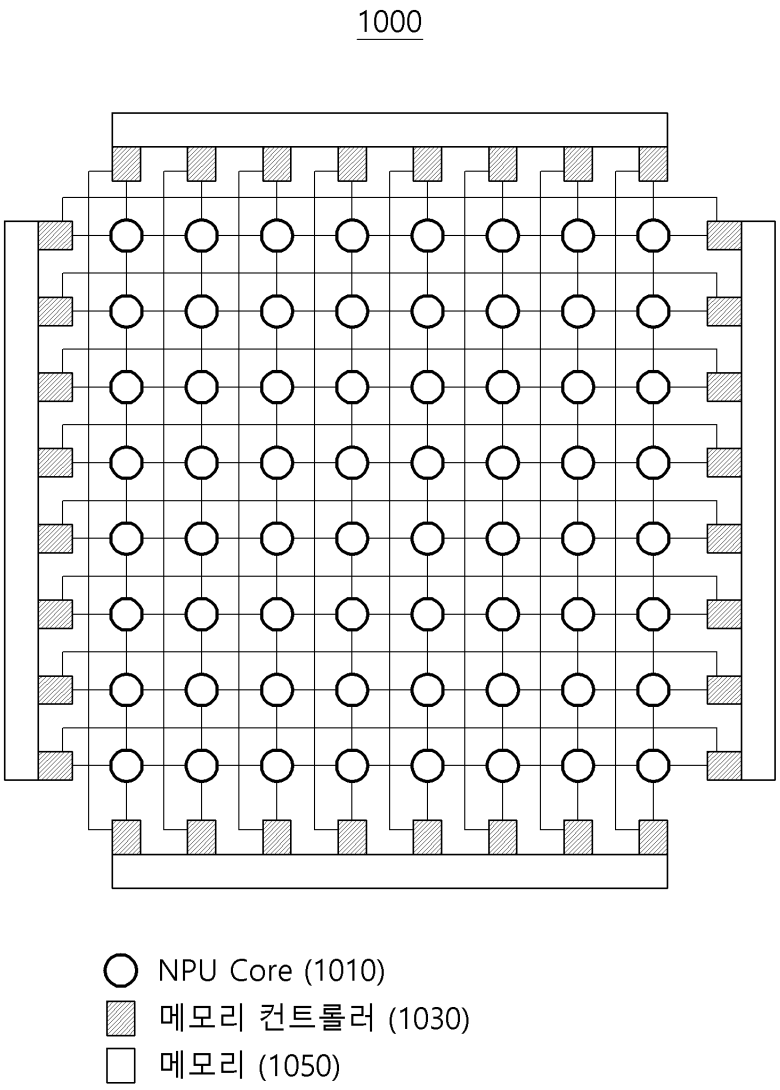
도면8



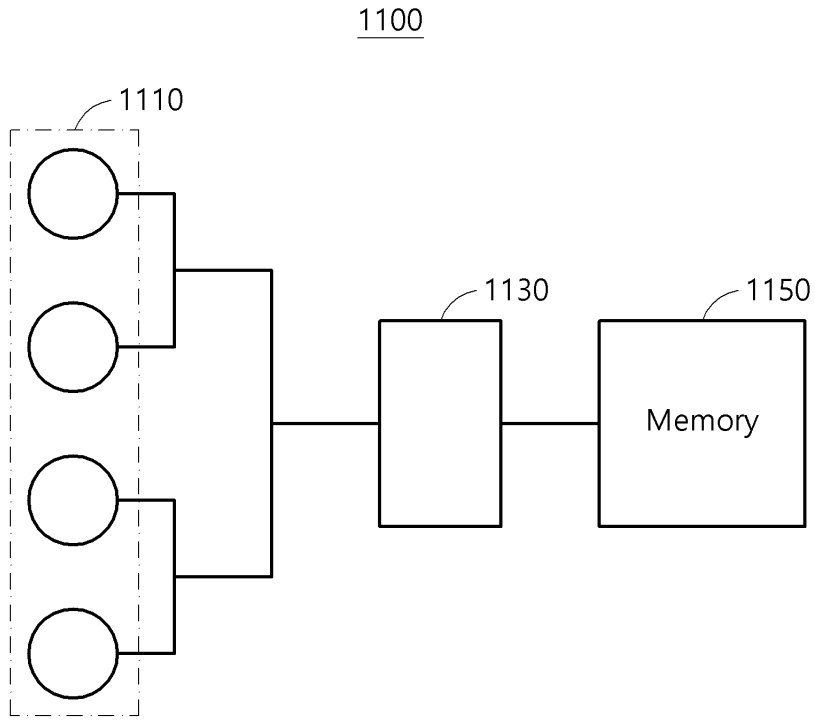
도면9


900
Memory 1 (DRAM/SRAM)
Memory 1 (DRAM/SRAM)
Memory 1 (DRAM/SRAM)
Memory controller or Bus 1
Memory controller or Bus 2
Memory controller or Bus M
Command
Command
Command
Core 1
910
Core 2 Cluster
Core N
Command
Command
Command
920
Core 1
930
Core 2
940
Core L

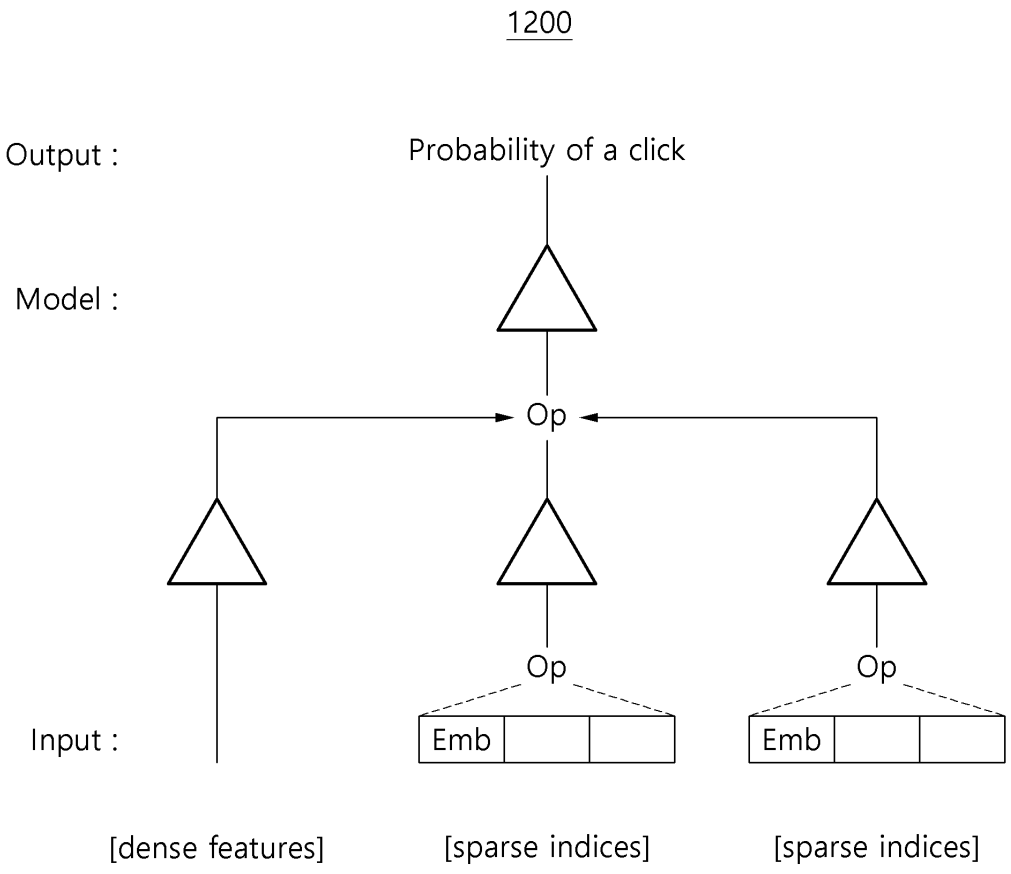
도면10



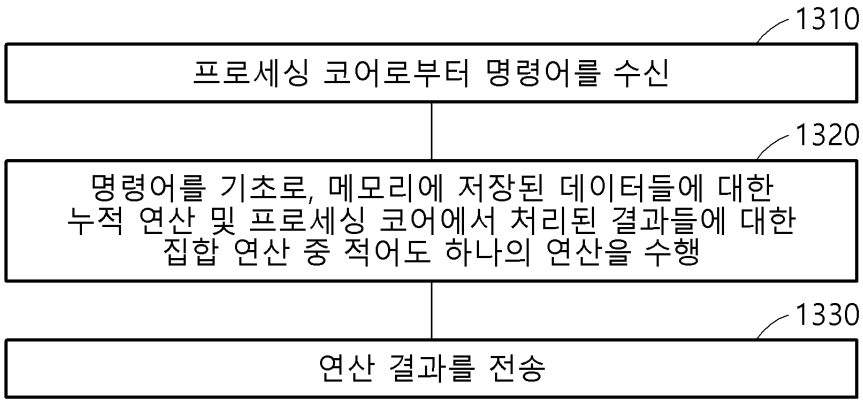
도면11


1100
1110
1130
1150
Memory

도면12


1200
Output :
Probability of a click
Model :
Op
Op
Op
Input :
Emb
Emb
[dense features]
[sparse indices]
[sparse indices]

도면13


1310
프로세싱 코어로부터 명령어를 수신
1320
명령어를 기초로, 메모리에 저장된 데이터들에 대한 누적 연산 및 프로세싱 코어에서 처리된 결과들에 대한 집합 연산 중 적어도 하나의 연산을 수행
1330
연산 결과를 전송