(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2024-0077672
(43) 공개일자 2024년06월03일

(51) 국제특허분류(Int. Cl.)
G06N 3/063 (2023.01) *G06F 15/78* (2006.01)
G06F 7/544 (2017.01)
(52) CPC특허분류
G06N 3/063 (2013.01)
G06F 15/7867 (2013.01)
(21) 출원번호 10-2022-0159781
(22) 출원일자 2022년11월24일
심사청구일자 2022년11월24일

(71) 출원인
인하대학교 산학협력단
인천광역시 미추홀구 인하로 100(용현동, 인하대학교)
(72) 발명자
서영교
경기도 부천시 길주로77번길 19-45, 920호(상동, 디아뜨갤러리4)
김종혁
서울특별시 서대문구 수색로 100, 110동 801호(북가좌동, DMC 래미안 e편한세상)
(74) 대리인
양성보

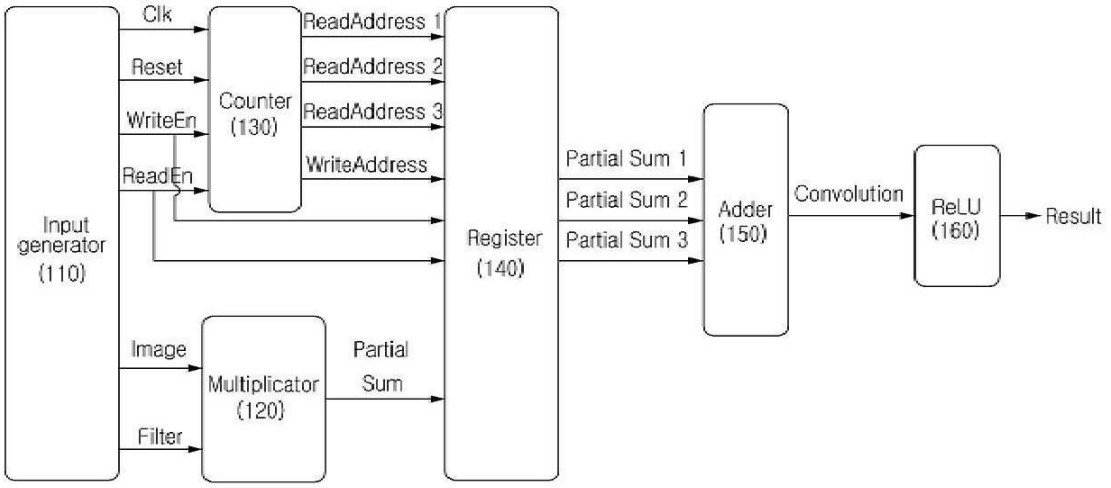
전체 청구항 수 : 총 8 항

(54) 발명의 명칭 **FPGA를 활용한 CNN 가속기 하드웨어 설계**

(57) 요 약

FPGA를 활용한 CNN 가속기 하드웨어 및 그 동작 방법이 제시된다. 일 측면에 있어서, 본 발명에서 제안하는 FPGA를 활용한 CNN 가속기 하드웨어에 있어서, CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석하기 위해, FPGA에 입력하기 위한 입력값들을 출력하는 입력 생성부(Input generator), 입력 이미지와 필터값을 곱한 값을 출력하는 곱셈부(Multiplicator), 레지스터가 쓰기 및 읽기 동작을 수행하도록 하는 신호를 출력하는 카운터, 레지스터로부터 전달 받은 값을 더하여 ReLU에 전달하는 가산기 및 입력 이미지에 대한 패턴 분석 결과 값을 도출하는 ReLU를 포함한다.

대 표 도



(52) CPC특허분류
G06F 7/5443 (2013.01)

이 발명을 지원한 국가연구개발사업
과제고유번호 1711155747
과제번호 2021M3F3A2A01037531
부처명 과학기술정보통신부
과제관리(전문)기관명 한국연구재단
연구사업명 신소자원천기술개발
연구과제명 [Ezbaro] Dynamic logic 기반 고성능 STMG 로직 회로 설계
기 여 율 20/100
과제수행기관명 인하대학교
연구기간 2022.01.01 ~ 2022.12.31
이 발명을 지원한 국가연구개발사업
과제고유번호 1415179211
과제번호 20009972
부처명 산업통상자원부
과제관리(전문)기관명 한국산업기술평가관리원
연구사업명 차세대지능형반도체기술개발
연구과제명 고효율 초저전력 경량 엣지 디바이스용 소자회로 및 SoC 개발(3차년도)
기 여 율 20/100
과제수행기관명 한국전자기술연구원
연구기간 2022.01.01 ~ 2022.12.31
이 발명을 지원한 국가연구개발사업
과제고유번호 1711169337
과제번호 2020R1F1A1051529
부처명 과학기술정보통신부
과제관리(전문)기관명 한국연구재단
연구사업명 기본연구
연구과제명 [Ezbaro] CMOS 메모리와 차세대 메모리를 이용한 저전력 저면적 PIM (Processing In Memory) 설계
기 여 율 20/100
과제수행기관명 인하대학교
연구기간 2022.03.01 ~ 2023.02.28
이 발명을 지원한 국가연구개발사업
과제고유번호 1711173059
과제번호 2022M3I7A2085479
부처명 과학기술정보통신부
과제관리(전문)기관명 한국연구재단
연구사업명 원천기술개발사업
연구과제명 [Ezbaro] (2022M3I7A2079155) 강유전체 메모리 기반 PIM 소자, 어레이, 구동회로 개발
기 여 율 20/100
과제수행기관명 인하대학교
연구기간 2022.04.20 ~ 2022.12.31
이 발명을 지원한 국가연구개발사업
과제고유번호 1711152938
과제번호 2021-0-02052-002
부처명 과학기술정보통신부
과제관리(전문)기관명 정보통신기획평가원
연구사업명 대학ITRC센터육성지원사업(교육운영수익)
연구과제명 [Ezbaro][정부] 스마트 모빌리티를 위한 인공지능 시스템반도체 핵심 기술 개발 및 인력 양성(2차년도)
기 여 율 20/100
과제수행기관명 인하대학교 산학협력단
연구기간 2022.01.01 ~ 2022.12.31

명 세 서

청구범위

청구항 1

CNN 가속기 하드웨어에 있어서,

CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석하기 위해,

FPGA에 입력하기 위한 입력값들을 출력하는 입력 생성부(Input generator);

입력 이미지와 필터값을 곱한 값을 출력하는 곱셈부(Multiplicator);

레지스터가 쓰기 및 읽기 동작을 수행하도록 하는 신호를 출력하는 카운터;

레지스터로부터 전달 받은 값을 더하여 ReLU에 전달하는 가산기; 및

입력 이미지에 대한 패턴 분석 결과 값을 도출하는 ReLU

를 포함하는 CNN 가속기 하드웨어.

청구항 2

제1항에 있어서,

상기 카운터는,

쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 레지스터에 쓰기 동작을 수행하기 위한 쓰기주소(WriteAddress)를 출력하고,

읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 레지스터로부터 읽기 동작을 수행하기 위한 복수의 읽기주소(ReadAddress1, ReadAddress2, ReadAddress3)를 출력하는

CNN 가속기 하드웨어.

청구항 3

제2항에 있어서,

상기 레지스터는,

쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 카운터가 출력하는 쓰기주소에 곱셈부에서 출력되는 부분합을 저장하고,

읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 카운터가 출력하는 읽기주소로부터 값을 출력하여 가산기에 전달하는

CNN 가속기 하드웨어.

청구항 4

제2항에 있어서,

각각의 쓰기주소 및 읽기주소는 링 카운터(Ring counter)와 인코더(encoder)가 이어져 클럭 신호마다 주소를 출력하는

CNN 가속기 하드웨어.

청구항 5

CNN 가속기 하드웨어 동작 방법에 있어서,

CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석하기 위해,

입력 생성부(Input generator)를 통해 FPGA에 입력하기 위한 입력값들을 출력하는 단계;

곱셈부(Multiplicator)를 통해 입력 이미지와 필터값을 곱한 값을 출력하는 단계;

카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계;

가산기를 통해 레지스터로부터 전달 받은 값을 더하여 ReLU에 전달하는 단계; 및

ReLU를 통해 입력 이미지에 대한 패턴 분석 결과 값을 도출하는 단계

를 포함하는 CNN 가속기 하드웨어 동작 방법.

청구항 6

제5항에 있어서,

상기 카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계는,

쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 레지스터에 쓰기 동작을 수행하기 위한 쓰기주소(WriteAddress)를 출력하고,

읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 레지스터로부터 읽기 동작을 수행하기 위한 복수의 읽기주소(ReadAddress1, ReadAddress2, ReadAddress3)를 출력하는

CNN 가속기 하드웨어 동작 방법.

청구항 7

제6항에 있어서,

상기 카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계는,

쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 카운터가 출력하는 쓰기주소에 곱셈부에서 출력되는 부분합을 저장하고,

읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 카운터가 출력하는 읽기주소로부터 값을 출력하여 가산기에 전달하는

CNN 가속기 하드웨어 동작 방법.

청구항 8

제6항에 있어서,

상기 카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계는,

각각의 쓰기주소 및 읽기주소는 링 카운터(Ring counter)와 인코더(encoder)가 이어져 클럭 신호마다 주소를 출력하는

CNN 가속기 하드웨어 동작 방법.

발명의 설명

기 술 분 야

[0001] 본 발명은 FPGA를 활용한 CNN 가속기 하드웨어 및 그 동작 방법에 관한 것이다.

배 경 기 술

[0002] 합성곱 신경망이라고 불리는 CNN(Convolution Neural Network)는 이름에서도 알 수 있듯이 컨볼루션(Convolution)을 이용한 신경망(Neural Network) 모델이다. CNN은 학습 과정에서 주어진 데이터를 통해 특징을 스스로 학습하여 이미지를 분류하고 특징을 자동으로 추출한다. 이러한 장점들을 가지고 있기 때문에 자율주행

자동차, 얼굴 인식과 같이 객체 인식 분야 혹은 컴퓨터 비전 CNN은 널리 사용되고 있다. CNN에서는 패턴 분석을 위해 입력값인 이미지에 대해서 필터를 겹쳐가면서 합성곱을 구하고, 그 합성곱의 값을 활성 함수를 거쳐 얻어낸 값을 활용하는데 이것을 컨볼루션 레이어(Convolution Layer) 라고 한다[1,2].

선행기술문헌

비특허문헌

[0003] (비특허문헌 0001) [1] A. Krizhevsky. I. Sutskever and G.E. Hinton, "Imagenet classification with deep convolutional neural networks", Advances in neural information processing systems, pp. 1097-1105, 2012.

(비특허문헌 0002) [2] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." arXiv preprint arXiv:1409.1556, 2014.

(비특허문헌 0003) [3] Vinod Nair and Geoffrey E. Hinton. "Rectified linear units improve restricted Boltzmann machines", ICML, pp. 807-814, 2010.

발명의 내용

해결하려는 과제

[0004] 본 발명이 이루고자 하는 기술적 과제는 CNN에서의 이미지 분석을 위해 이미지와 필터간의 합성곱을 수행하고 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석하기 위한 CNN 가속기 하드웨어 및 그 동작 방법을 제공하는데 있다.

과제의 해결 수단

[0005] 일 측면에 있어서, 본 발명에서 제안하는 FPGA를 활용한 CNN 가속기 하드웨어에 있어서, CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석하기 위해, FPGA에 입력하기 위한 입력값들을 출력하는 입력 생성부(Input generator), 입력 이미지와 필터값을 곱한 값을 출력하는 곱셈부(Multiplicator), 레지스터가 쓰기 및 읽기 동작을 수행하도록 하는 신호를 출력하는 카운터, 레지스터로부터 전달 받은 값을 더하여 ReLU에 전달하는 가산기 및 입력 이미지에 대한 패턴 분석 결과 값을 도출하는 ReLU를 포함한다.

[0006] 상기 카운터는 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 레지스터에 쓰기 동작을 수행하기 위한 쓰기주소(WriteAddress)를 출력하고, 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 레지스터로부터 읽기 동작을 수행하기 위한 복수의 읽기주소(ReadAddress1, ReadAddress2, ReadAddress3)를 출력한다.

[0007] 상기 레지스터는 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 카운터가 출력하는 쓰기주소에 곱셈부에서 출력되는 부분합을 저장하고, 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 카운터가 출력하는 읽기주소로부터 값을 출력하여 가산기에 전달한다.

[0008] 상기 각각의 쓰기주소 및 읽기주소는 링 카운터(Ring counter)와 인코더(encoder)가 이어져 클럭 신호마다 주소를 출력한다.

[0009] 또 다른 일 측면에 있어서, 본 발명에서 제안하는 FPGA를 활용한 CNN 가속기 하드웨어 동작 방법에 있어서, CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석하기 위해, 입력 생성부(Input generator)를 통해 FPGA에 입력하기 위한 입력값들을 출력하는 단계, 곱셈부(Multiplicator)를 통해 입력 이미지와 필터값을 곱한 값을 출력하는 단계, 카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계, 가산기를 통해 레지스터로부터 전달 받은 값을 더하여 ReLU에 전달하는 단계 및 ReLU를 통해 입력 이미지에 대한 패턴 분석 결과 값을 도출하는 단계를 포함한다.

발명의 효과

[0010] 본 발명의 실시예들에 따르면 FPGA를 활용한 CNN 가속기 하드웨어 및 그 동작 방법을 통해 CNN에서의 이미지 분석을 위해 이미지와 필터간의 합성곱을 수행하고 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석할 수 있다.

도면의 간단한 설명

[0011] 도 1은 본 발명의 일 실시예에 따른 FPGA를 활용한 CNN 가속기 하드웨어의 회로 구성을 나타내는 도면이다.

도 2는 본 발명의 일 실시예에 따른 FPGA를 활용한 CNN 가속기 하드웨어의 동작 방법을 설명하기 위한 흐름도이다.

도 3은 본 발명의 일 실시예에 따른 CNN 레이어 시뮬레이션 파형을 나타내는 도면이다.

도 4는 본 발명의 또 다른 실시예에 따른 CNN 레이어 시뮬레이션 파형을 나타내는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0012] CNN은 이미지 분석을 위해 패턴을 찾기 유용한 알고리즘으로, 데이터를 통해 이미지를 직접 학습하여 패턴을 활용해 이미지를 분류한다. CNN은 기존 사용되던 DNN의 단점을 보완하여 공간적, 지역적 정보를 유지한 채 패턴을 분석한다. 이러한 장점 때문에 CNN은 자율주행 자동차, 얼굴인식과 같은 객체 인식 분야 등에서 널리 사용되고 있다. CNN에서는 이미지 분석을 위해 이미지와 필터간의 합성곱을 수행하고 활성 함수를 거쳐 값을 도출하는데 본 발명에서는 이 과정을 FPGA를 활용하여 설계하고 확인한다. 이하, 본 발명의 실시 예를 첨부된 도면을 참조하여 상세하게 설명한다.

[0014] 도 1은 본 발명의 일 실시예에 따른 FPGA를 활용한 CNN 가속기 하드웨어의 회로 구성을 나타내는 도면이다.

[0015] 제안하는 FPGA를 활용한 CNN 가속기 하드웨어는 CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석한다.

[0016] 본 발명의 일 실시예에 따른 FPGA를 활용한 CNN 가속기 하드웨어의 회로는 입력 생성부(Input generator)(110), 곱셈부(Multiplicator)(120), 카운터(Counter)(130), 레지스터(Register)(140), 가산기(Adder)(150) 및 ReLU(160)을 포함한다.

[0017] 본 발명의 실시예에 따른 입력 생성부(110)는 FPGA에 입력하기 위한 입력값들을 출력한다.

[0018] 본 발명에서는 위에서 설명한 컨볼루션 레이어(Convolution Layer)를 하나를 FPGA를 활용하여 설계한다. 먼저 FPGA에 원하는 입력값을 넣어주기 위해서 입력값 출력을 위한 입력 생성부(Input generator)를 설계했다. 본 발명의 실시예에 따른 입력 생성부(Input generator) 회로는 베릴로그에서의 테스트벤치(testbench)처럼 활용 가능하다.

[0019] 본 발명의 실시예에 따른 곱셈부(120)는 입력 이미지와 필터값을 곱한 값을 출력한다.

[0020] 본 발명의 실시예에 따른 카운터(130)는 레지스터(140)가 쓰기 및 읽기 동작을 수행하도록 하는 신호를 출력한다.

[0021] 본 발명의 실시예에 따른 카운터(130)는 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 레지스터(140)에 쓰기 동작을 수행하기 위한 쓰기주소(WriteAddress)를 출력한다.

[0022] 본 발명의 실시예에 따른 카운터(130)는 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 레지스터(140)로부터 읽기 동작을 수행하기 위한 복수의 읽기주소(ReadAddress1, ReadAddress2, ReadAddress3)를 출력한다.

[0023] 본 발명의 실시예에 따른 카운터(130)는 링 카운터(Ring counter)와 인코더(encoder)가 이어져 각각의 쓰기주소 및 읽기주소를 클럭 신호마다 출력한다.

[0024] 본 발명의 실시예에 따른 레지스터(140)는 7bit 데이터를 15개 저장 가능하다.

[0025] 본 발명의 실시예에 따른 레지스터(140)는 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 카운터(130)가 출력하는 쓰기주소에 곱셈부(120)에서 출력되는 부분합을 저장한다.

[0026] 본 발명의 실시예에 따른 레지스터(140)는 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 카운터(130)가 출력하는 읽기주소로부터 값을 출력하여 가산기(150)에 전달한다.

[0027] 본 발명의 실시예에 따른 가산기(150)는 레지스터(140)로부터 전달 받은 3개의 값을 더하여 ReLU(160)에 전달한다.

[0028] 본 발명의 실시예에 따른 ReLU(160)는 입력 이미지에 대한 패턴 분석 결과 값을 도출한다. 본 발명의 실시예에 따른 ReLU(160)는 딥러닝에서 활용되는 활성화 함수(Activation function) 중 하나이다. 본 발명의 실시예에 따른 ReLU 함수는 입력이 양수이면 그대로 출력하고, 음수이면 0을 출력 한다[3].

[0029] $$h(x) = \begin{cases} x & (x > 0) \\ 0 & (x \leq 0) \end{cases} \quad (1)$$

[0030] 식(1)과 같이 가산기(150)로부터 전달받은 합성곱 결과가 양수면 그 대로 출력, 음수면 0을 출력한다.

[0032] 도 2는 본 발명의 일 실시예에 따른 FPGA를 활용한 CNN 가속기 하드웨어의 동작 방법을 설명하기 위한 흐름도이다.

[0033] 제안하는 FPGA를 활용한 CNN 가속기 하드웨어의 동작 방법은 CNN에서의 패턴 분석을 위한 입력 이미지에 대하여 필터를 겹쳐가면서 합성곱을 구하고, 상기 합성곱의 값에 대하여 활성 함수를 거쳐 값을 도출하는 과정을 FPGA를 활용하여 분석한다.

[0034] 본 발명의 일 실시예에 따른 FPGA를 활용한 CNN 가속기 하드웨어의 동작 방법은 입력 생성부(Input generator)를 통해 FPGA에 입력하기 위한 입력값들을 출력하는 단계(210), 곱셈부(Multiplicator)를 통해 입력 이미지와 필터값을 곱한 값을 출력하는 단계(220), 카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계(230), 가산기를 통해 레지스터로부터 전달 받은 값을 더하여 ReLU에 전달하는 단계(240) 및 ReLU를 통해 입력 이미지에 대한 패턴 분석 결과 값을 도출하는 단계(250)를 포함한다.

[0035] 단계(210)에서, 입력 생성부를 통해 FPGA에 입력하기 위한 입력값들을 출력한다.

[0036] 본 발명에서는 위에서 설명한 컨볼루션 레이어(Convolution Layer)를 하나를 FPGA를 활용하여 설계한다. 먼저 FPGA에 원하는 입력값을 넣어주기 위해서 입력값 출력을 위한 입력 생성부(Input generator)를 설계했다. 본 발명의 실시예에 따른 입력 생성부(Input generator) 회로는 베릴로그에서의 테스트벤치(testbench)처럼 활용 가능하다.

[0037] 단계(220)에서, 곱셈부를 통해 입력 이미지와 필터값을 곱한 값을 출력한다.

[0038] 단계(230)에서, 카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행한다.

[0039] 본 발명의 실시예에 따른 카운터는 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 레지스터에 쓰기 동작을 수행하기 위한 쓰기주소(WriteAddress)를 출력한다.

[0040] 본 발명의 실시예에 따른 카운터는 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 레지스터로부터 읽기 동작을 수행하기 위한 복수의 읽기주소(ReadAddress1, ReadAddress2, ReadAddress3)를 출력한다.

[0041] 본 발명의 실시예에 따른 카운터는 링 카운터(Ring counter)와 인코더(encoder)가 이어져 각각의 쓰기주소 및 읽기주소를 클럭 신호마다 출력한다.

[0042] 본 발명의 실시예에 따른 레지스터는 7bit 데이터를 15개 저장 가능하다.

[0043] 본 발명의 실시예에 따른 레지스터는 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 카운터가 출력하는 쓰기주소에 곱셈부에서 출력되는 부분합을 저장한다.

[0044] 본 발명의 실시예에 따른 레지스터는 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 카운터가 출력하는 읽기주소로부터 값을 출력하여 가산기에 전달한다.

[0045] 단계(240)에서, 가산기를 통해 레지스터로부터 전달 받은 값을 더하여 ReLU에 전달한다.

[0046] 단계(250)에서, ReLU를 통해 입력 이미지에 대한 패턴 분석 결과 값을 도출한다. 본 발명의 실시예에 따른 ReLU는 딥러닝에서 활용되는 활성화 함수(Activation function) 중 하나이다. 본 발명의 실시예에 따른 ReLU 함수는

입력이 양수이면 그대로 출력하고, 음수이면 0을 출력 한다.

[0048] 도 3은 본 발명의 일 실시예에 따른 CNN 레이어 시뮬레이션 파형을 나타내는 도면이다.

[0049] 도 3은 0ns - 180ns의 CNN 레이어 시뮬레이션 파형을 나타내는 도면이다.

[0050] 도 4는 본 발명의 또 다른 실시예에 따른 CNN 레이어 시뮬레이션 파형을 나타내는 도면이다.

[0051] 도 4는 180ns - 260ns의 CNN 레이어 시뮬레이션 파형을 나타내는 도면이다.

[0052] 본 발명의 실시예에 따른 입력 생성부(Input generator)를 활용해 입력 이미지와 필터 15개의 조합을 출력하도록 하여 레이어의 입력으로 사용했다.

[0053] 먼저 베릴로그로 시뮬레이션을 돌려본 결과, 쓰기 인에이블 신호(WriteEn)가 하이(high)일 때, 레지스터(Register)에 값들이 저장되고, 읽기 인에이블 신호(ReadEn)가 하이(high)일 때, 레지스터(Register)에서 값들을 읽어와 연산하여, 결과값을 출력하는 것을 확인했다. 이를 FPGA에서 확인하기 위해 베릴로그 코드를 FPGA에 다운로드하여 FPGA에서 동작을 확인했다. 출력은 FPGA의 LED에서 확인할 수 있도록 했다.

[0054] 예를 들어, 출력이 00001101(2) 가 나온 경우, 이는 시뮬레이션 파형에서 225ns에서의 출력인 0d(16)일 때가 FPGA에 출력된 것이고, 출력이 00011101(2) 가 나온 경우, 이는 시뮬레이션 파형에서 245ns에서의 출력인 1d(16) 이 FPGA 에 출력된 것이다.

[0056] 이상에서 설명된 장치는 하드웨어 구성요소, 소프트웨어 구성요소, 및/또는 하드웨어 구성요소 및 소프트웨어 구성요소의 조합으로 구현될 수 있다. 예를 들어, 실시예들에서 설명된 장치 및 구성요소는, 예를 들어, 프로세서, 콘트롤러, ALU(arithmetic logic unit), 디지털 신호 프로세서(digital signal processor), 마이크로컴퓨터, FPGA(field programmable gate array), PLU(programmable logic unit), 마이크로프로세서, 또는 명령(instruction)을 실행하고 응답할 수 있는 다른 어떠한 장치와 같이, 하나 이상의 범용 컴퓨터 또는 특수 목적 컴퓨터를 이용하여 구현될 수 있다. 처리 장치는 운영 체제(OS) 및 상기 운영 체제 상에서 수행되는 하나 이상의 소프트웨어 애플리케이션을 수행할 수 있다. 또한, 처리 장치는 소프트웨어의 실행에 응답하여, 데이터를 접근, 저장, 조작, 처리 및 생성할 수도 있다. 이해의 편의를 위하여, 처리 장치는 하나가 사용되는 것으로 설명된 경우도 있지만, 해당 기술분야에서 통상의 지식을 가진 자는, 처리 장치가 복수 개의 처리 요소(processing element) 및/또는 복수 유형의 처리 요소를 포함할 수 있음을 알 수 있다. 예를 들어, 처리 장치는 복수 개의 프로세서 또는 하나의 프로세서 및 하나의 콘트롤러를 포함할 수 있다. 또한, 병렬 프로세서(parallel processor)와 같은, 다른 처리 구성(processing configuration)도 가능하다.

[0057] 소프트웨어는 컴퓨터 프로그램(computer program), 코드(code), 명령(instruction), 또는 이들 중 하나 이상의 조합을 포함할 수 있으며, 원하는 대로 동작하도록 처리 장치를 구성하거나 독립적으로 또는 결합적으로(collectively) 처리 장치를 명령할 수 있다. 소프트웨어 및/또는 데이터는, 처리 장치에 의하여 해석되거나 처리 장치에 명령 또는 데이터를 제공하기 위하여, 어떤 유형의 기계, 구성요소(component), 물리적 장치, 가상 장치(virtual equipment), 컴퓨터 저장 매체 또는 장치에 구체화(embody)될 수 있다. 소프트웨어는 네트워크로 연결된 컴퓨터 시스템 상에 분산되어서, 분산된 방법으로 저장되거나 실행될 수도 있다. 소프트웨어 및 데이터는 하나 이상의 컴퓨터 판독 가능 기록 매체에 저장될 수 있다.

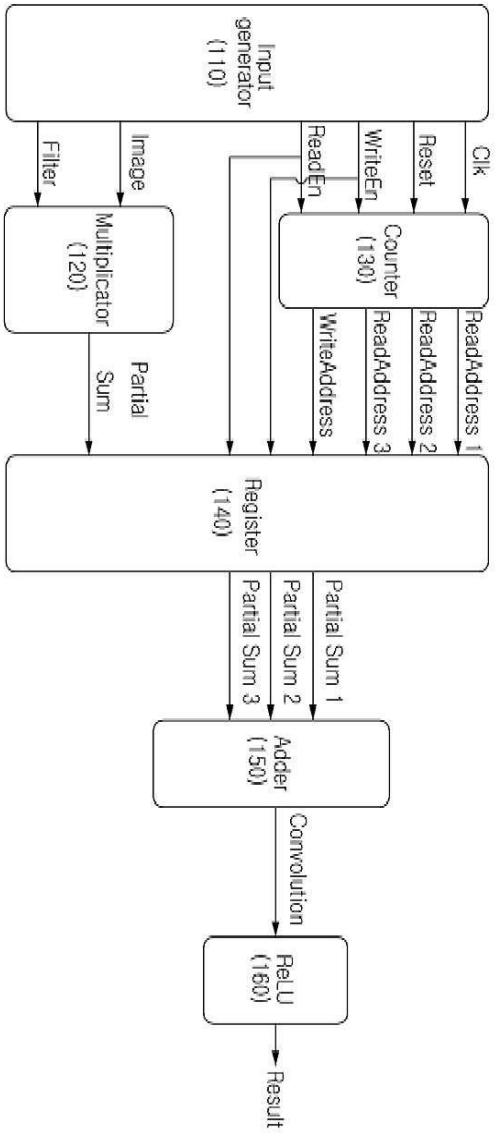
[0058] 실시예에 따른 방법은 다양한 컴퓨터 수단을 통하여 수행될 수 있는 프로그램 명령 형태로 구현되어 컴퓨터 판독 가능 매체에 기록될 수 있다. 상기 컴퓨터 판독 가능 매체는 프로그램 명령, 데이터 파일, 데이터 구조 등을 단독으로 또는 조합하여 포함할 수 있다. 상기 매체에 기록되는 프로그램 명령은 실시예를 위하여 특별히 설계되고 구성된 것들이거나 컴퓨터 소프트웨어 당업자에게 공지되어 사용 가능한 것일 수도 있다. 컴퓨터 판독 가능 기록 매체의 예에는 하드 디스크, 플로피 디스크 및 자기 테이프와 같은 자기 매체(magnetic 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.

[0059] 이상과 같이 실시예들이 비록 한정된 실시예와 도면에 의해 설명되었으나, 해당 기술분야에서 통상의 지식을 가진 자라면 상기의 기재로부터 다양한 수정 및 변형이 가능하다. 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.

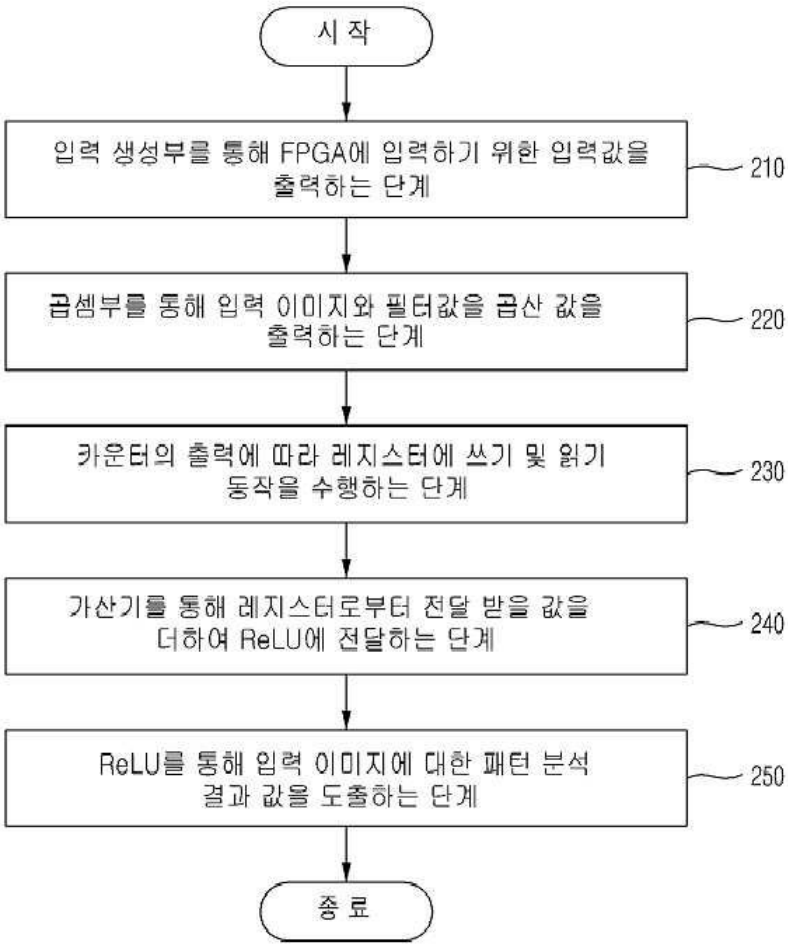
[0060] 그러므로, 다른 구현들, 다른 실시예들 및 특허청구범위와 균등한 것들도 후술하는 특허청구범위의 범위에 속한다.

도면

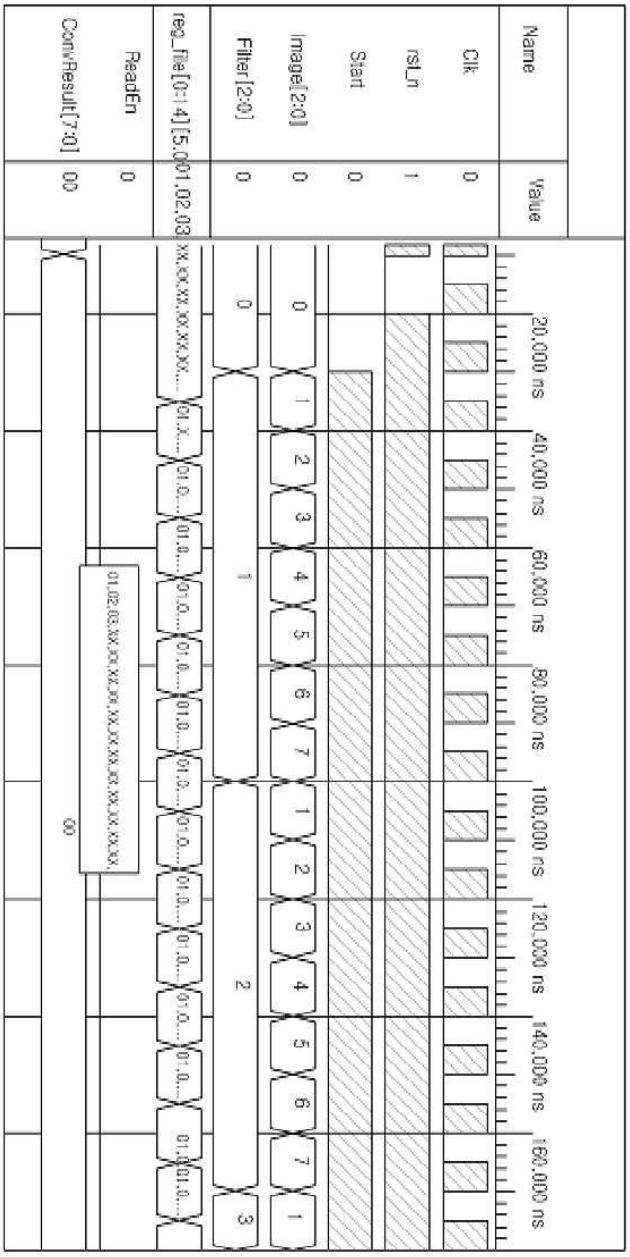
도면1



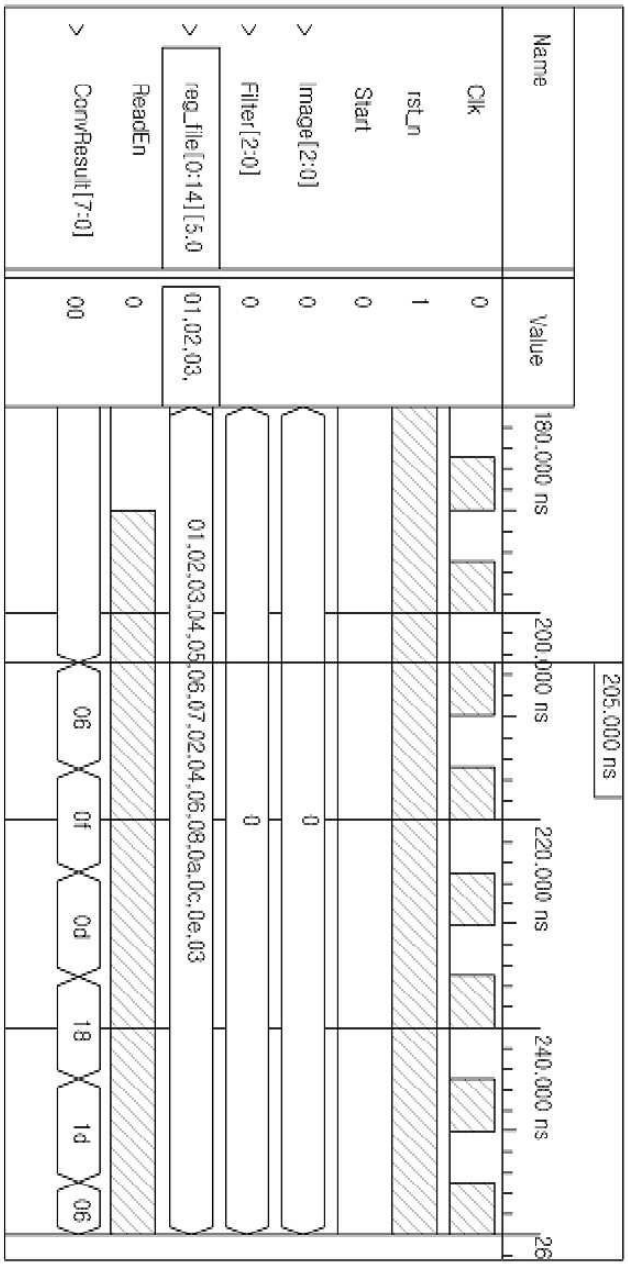
도면2


시작
입력 생성부를 통해 FPGA에 입력하기 위한 입력값을 출력하는 단계
210
곱셈부를 통해 입력 이미지와 필터값을 곱산 값을 출력하는 단계
220
카운터의 출력에 따라 레지스터에 쓰기 및 읽기 동작을 수행하는 단계
230
가산기를 통해 레지스터로부터 전달 받을 값을 더하여 ReLU에 전달하는 단계
240
ReLU를 통해 입력 이미지에 대한 패턴 분석 결과 값을 도출하는 단계
250
종료

도면3


Name
Value
Clk
rstLn
Start
Image[2:0]
Filter[2:0]
reg_file[0:14][5:0]
ReadEn
ConvResult[7:0]
20,000 ns
40,000 ns
60,000 ns
80,000 ns
100,000 ns
120,000 ns
140,000 ns
160,000 ns

도면4


Name
Value
Clk
rst_n
Start
Image[2:0]
Filter[2:0]
reg_file[0:14][5:0
ReadEn
ConvResult[7:0]
0
1
0
0
0
01,02,03,
0
00
180.000 ns
200.000 ns
205.000 ns
220.000 ns
240.000 ns
01,02,03,04,05,06,07,02,04,06,08,0a,0c,0e,03
06
0f
0d
18
1d
06