



등록특허 10-2500633



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2023년02월15일  
(11) 등록번호 10-2500633  
(24) 등록일자 2023년02월13일

(51) 국제특허분류(Int. Cl.)  
G06N 3/063 (2023.01) G06N 3/08 (2023.01)  
(52) CPC특허분류  
G06N 3/063 (2013.01)  
G06N 3/08 (2023.01)  
(21) 출원번호 10-2020-0055739  
(22) 출원일자 2020년05월11일  
심사청구일자 2020년05월11일  
(65) 공개번호 10-2021-0137663  
(43) 공개일자 2021년11월18일  
(56) 선행기술조사문헌  
KR1020190051766 A\*  
US20120259804 A1  
\*는 심사관에 의하여 인용된 문헌

(73) 특허권자  
연세대학교 산학협력단  
서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)  
(72) 발명자  
정성욱  
서울특별시 서대문구 연세로 50, 연세대학교 제3공학관 C513(신촌동)  
박지아  
서울특별시 서대문구 연세로 50, 연세대학교 제3공학관 C421(신촌동)  
(뒷면에 계속)  
(74) 대리인  
민영준

전체 청구항 수 : 총 14 항

심사관 : 이준상

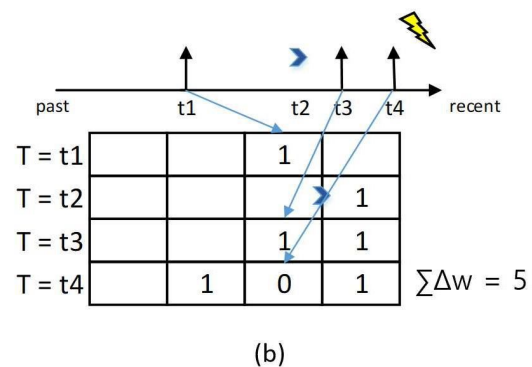
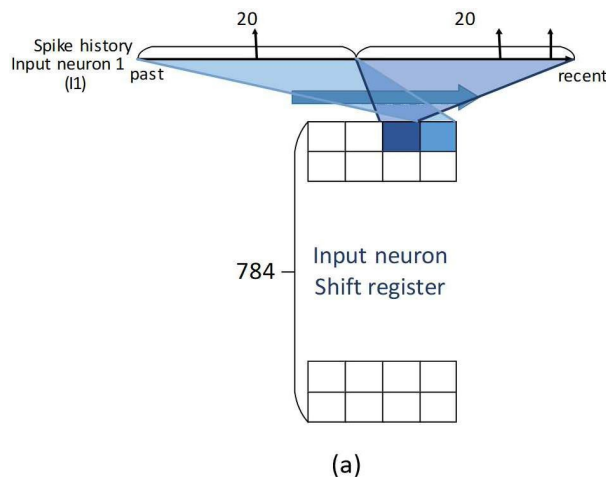
(54) 발명의 명칭 스파이킹 신경망 학습 장치 및 방법

(57) 요약

본 발명은 클럭 신호를 인가받아 카운트하여 기지정된 시간 간격 단위로 시프트 신호를 출력하는 카운터, 다수의 입력 뉴런의 개수에 대응하는 개수와 양자화 STDP 함수에 의해 구분된 시간 구간인 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖고, 대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 기지정된 값을 가산하는

(뒷면에 계속)

대표도



다수의 입력 시프트 레지스터, 다수의 출력 뉴런의 개수에 대응하는 개수와 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖고, 대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 기지정된 값을 가산하는 다수의 출력 시프트 레지스터 및 다수의 입력 시프트 레지스터와 다수의 출력 시프트 레지스터에 저장된 값을 누적 가중치 변화량으로 인가받아 다수의 시냅스에 저장된 가중치에 가산 또는 감산하여 가중치를 업데이트하는 가중치 업데이트부를 포함하여, 소형, 저비용으로 용이하게 하드웨어로 구현할 수 있고, 학습 윈도우 크기를 용이하게 가변할 수 있는 스파이킹 신경망 학습 장치 및 방법을 제공할 수 있다.

(72) 발명자

**주성환**

서울특별시 서대문구 연세로 50, 연세대학교 제3공학관 C421(신촌동)

**김재윤**

서울특별시 서대문구 연세로 50, 연세대학교 제3공학관 C421(신촌동)

## 명세서

### 청구범위

#### 청구항 1

다수의 입력 뉴런과 다수의 출력 뉴런 및 각각 가중치가 저장된 다수의 시냅스를 포함하는 스파이킹 신경망을 학습시키기 위한 학습 장치에 있어서,

상기 다수의 입력 뉴런의 개수에 대응하는 개수와 양자화 STDP 함수에 의해 구분된 시간 구간인 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖고, 대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 기지정된 값을 가산하는 다수의 입력 시프트 레지스터;

상기 다수의 출력 뉴런의 개수에 대응하는 개수와 상기 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖고, 대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 기지정된 값을 가산하는 다수의 출력 시프트 레지스터; 및

상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터에 저장된 값을 누적 가중치 변화량으로 인가받아 다수의 시냅스에 저장된 가중치에 가산 또는 감산하여 가중치를 업데이트하는 가중치 업데이트부를 포함하되,

상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터 각각은

상기 다수의 양자화 구간의 개수에 기지정된 비트 수가 추가된 크기를 갖는 스파이킹 신경망 학습 장치.

#### 청구항 2

삭제

#### 청구항 3

삭제

#### 청구항 4

제1항에 있어서,

클럭 신호를 인가받아 카운트하여 기지정된 시간 간격 단위로 시프트 신호를 출력하는 카운터를 더 포함하고,

상기 카운터는

상기 다수의 양자화 구간의 시간 구간에 대응하는 시간 간격 단위로 상기 시프트 신호를 출력하며,

상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터 각각은

상기 시프트 신호가 인가되면, 저장된 값을 최하위 비트 방향으로 1비트 단위로 시프트시키는 스파이킹 신경망 학습 장치.

#### 청구항 5

제4항에 있어서, 상기 다수의 입력 시프트 레지스터 각각은

대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 상기 다수의 양자화 구간의 개수에 대응하는 비트 위치에 1의 값을 가산하는 스파이킹 신경망 학습 장치.

#### 청구항 6

제5항에 있어서, 상기 다수의 입력 시프트 레지스터 각각은

상기 다수의 출력 뉴런 중 적어도 하나의 출력 뉴런에서 출력 스파이크가 발생되면, 저장된 값을 입력 누적 가

중치 변화량으로서 상기 가중치 업데이트부로 출력하는 스파이킹 신경망 학습 장치.

#### 청구항 7

제6항에 있어서, 상기 다수의 출력 시프트 레지스터 각각은

대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 상기 다수의 양자화 구간의 개수에 대응하는 비트 위치에 1의 값을 가산하는 스파이킹 신경망 학습 장치.

#### 청구항 8

제7항에 있어서, 상기 다수의 출력 시프트 레지스터 각각은

상기 다수의 입력 뉴런 중 적어도 하나의 입력 뉴런으로 입력 스파이크가 인가되면, 저장된 값을 출력 누적 가중치 변화량으로서 상기 가중치 업데이트부로 출력하는 스파이킹 신경망 학습 장치.

#### 청구항 9

제8항에 있어서, 상기 가중치 업데이트부는

상기 다수의 입력 시프트 레지스터 각각에 저장된 값을 인가받아, 상기 다수의 시냅스 중 출력 스파이크를 발생한 출력 뉴런에 연결된 시냅스의 가중치에 가산하고,

상기 다수의 출력 시프트 레지스터 각각에 저장된 값을 인가받아, 상기 다수의 시냅스 중 입력 스파이크가 인가된 입력 뉴런에 연결된 시냅스의 가중치에 감산하는 스파이킹 신경망 학습 장치.

#### 청구항 10

다수의 입력 뉴런과 다수의 출력 뉴런 및 각각 가중치가 저장된 다수의 시냅스를 포함하는 스파이킹 신경망을 학습시키기 위한 학습 방법에 있어서,

대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 상기 다수의 입력 뉴런의 개수에 대응하는 개수와 양자화 STDP 함수에 의해 구분된 시간 구간인 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖는 입력 시프트 레지스터에 기지정된 값을 가산하는 단계;

대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 상기 다수의 출력 뉴런의 개수에 대응하는 개수와 상기 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖는 출력 시프트 레지스터에 기지정된 값을 가산하는 단계; 및

상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터에 저장된 값을 누적 가중치 변화량으로 인가받아 다수의 시냅스에 저장된 가중치에 가산 또는 감산하여 가중치를 업데이트하는 단계를 포함하되,

상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터 각각은

상기 다수의 양자화 구간의 개수에 기지정된 비트 수가 추가된 크기를 갖는 스파이킹 신경망 학습 방법.

#### 청구항 11

삭제

#### 청구항 12

삭제

#### 청구항 13

제10항에 있어서,

클럭 신호를 인가받아 카운트하여 기지정된 시간 간격 단위로 시프트 신호를 출력하는 단계를 더 포함하고,

상기 시프트 신호를 출력하는 단계는

상기 다수의 양자화 구간의 시간 구간에 대응하는 시간 간격 단위로 상기 시프트 신호를 출력하고,  
상기 시프트 신호가 인가되면, 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터 각각에 저장된 값을 최하위 비트 방향으로 1비트 단위로 시프트시키는 단계를 더 포함하는 스파이킹 신경망 학습 방법.

#### 청구항 14

제13항에 있어서, 상기 입력 시프트 레지스터에 기지정된 값을 가산하는 단계는

대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 상기 다수의 양자화 구간의 개수에 대응하는 비트 위치에 1의 값을 가산하는 스파이킹 신경망 학습 방법.

#### 청구항 15

제14항에 있어서, 상기 스파이킹 신경망 학습 방법은

상기 다수의 출력 뉴런 중 적어도 하나의 출력 뉴런에서 출력 스파이크가 발생되면, 상기 다수의 입력 시프트 레지스터에 저장된 값을 입력 누적 가중치 변화량으로서 출력하는 단계를 더 포함하는 스파이킹 신경망 학습 방법.

#### 청구항 16

제15항에 있어서, 상기 출력 시프트 레지스터에 기지정된 값을 가산하는 단계는

대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 상기 다수의 양자화 구간의 개수에 대응하는 비트 위치에 1의 값을 가산하는 스파이킹 신경망 학습 방법.

#### 청구항 17

제16항에 있어서, 상기 스파이킹 신경망 학습 방법은

상기 다수의 입력 뉴런 중 적어도 하나의 입력 뉴런으로 입력 스파이크가 인가되면, 저장된 값을 출력 누적 가중치 변화량으로서 출력하는 단계를 더 포함하는 스파이킹 신경망 학습 방법.

#### 청구항 18

제17항에 있어서, 상기 가중치를 업데이트하는 단계는

상기 다수의 입력 시프트 레지스터 각각에 저장된 값을 인가받아, 상기 다수의 시냅스 중 출력 스파이크를 발생한 출력 뉴런에 연결된 시냅스의 가중치에 가산하는 단계; 및

상기 다수의 출력 시프트 레지스터 각각에 저장된 값을 인가받아, 상기 다수의 시냅스 중 입력 스파이크가 인가된 입력 뉴런에 연결된 시냅스의 가중치에 감산하는 단계를 포함하는 스파이킹 신경망 학습 방법.

### 발명의 설명

### 기술 분야

[0001] 본 발명은 스파이킹 신경망의 학습 장치에 관한 것으로, 스파이크 타이밍 의존 가소성을 시프트 누적기를 사용하여 하드웨어적으로 구현하는 스파이킹 신경망의 학습 장치에 관한 것이다.

### 배경 기술

[0002] 뉴로모픽(Neuromorphic) 기술은 인간의 신경구조를 하드웨어적으로 모방하기 위한 기술로서, 기존 컴퓨팅 아키텍처가 인지처리 기능을 수행함에 있어 인간에 비해 효율성이 매우 낮고 전력 소모가 크다는 한계를 극복하기 위해 제안되었다.

[0003] 뉴로모픽 기술에는 대표적으로 스파이킹 신경망(Spiking Neural Network: 이하 SNN)이 있다. SNN은 인간의 두뇌가 뉴런(Neuron)-시냅스(Synapse) 구조를 가지고 있고, 뉴런과 뉴런을 잇는 시냅스가 스파이크 형태의 전기 신호로 정보를 전달한다는 특징을 모방하여 고안된 신경망이다. 이러한 SNN은 스파이크가 전송되는 타이밍 차에 기초하여 정보를 처리한다.

- [0004] 도 1은 스파이킹 신경망의 동작 개념을 설명하기 위한 도면이다.
- [0005] 도 1의 (b) 도시된 바와 같이, SNN은 다수의 입력 뉴런(I1 ~ I3)과 다수의 출력 뉴런(O1 ~ O3) 및 다수의 입력 뉴런(I1 ~ I3)과 다수의 출력 뉴런(O1 ~ O3)을 서로 잇는 다수의 시냅스를 포함한다. 그리고 다수의 시냅스 각각은 학습에 의해 획득되는 가중치를 갖는다. 도 1에서는 일 예로 3개의 입력 뉴런(I1 ~ I3)과 하나의 출력 뉴런(O2) 사이를 잇는 3개의 시냅스가 갖는 가중치(W12, W22, W32)를 도시하였다.
- [0006] 다수의 입력 뉴런(I1 ~ I3)에는 (a)와 같이 서로 다른 타이밍(T1, T2, T4)에 입력 스파이크(Input spike) 신호가 인가된다. 입력 스파이크는 입력 데이터를 기지정된 방식으로 인코딩하여 스파이크 트레인(spike train)의 형태로 획득된 신호이다. 그리고 입력 스파이크가 인가된 입력 뉴런(I1 ~ I3)은 시냅스를 따라 다수의 출력 뉴런(O1 ~ O3)으로 스파이크를 전달한다. 이때, 각 시냅스의 가중치가 입력 스파이크에 가중되어 전달되고, 가중치가 가중된 입력 스파이크가 전달된 다수의 출력 뉴런(O1 ~ O3) 각각은 입력 스파이크가 전달된 타이밍과 시냅스에 할당된 가중치에 기반하여, 서로 다른 크기로 누적된 스파이크의 세기가 기지정된 발화 문턱값( $V_{th}$ ) 이상이면, 출력 스파이크(Output spike) 신호를 출력한다.
- [0007] 도 1에서 (c)를 살펴보면, 제1 타이밍(T1)과 제2 타이밍(T2)에 각각 제1 및 제3 입력 뉴런(I1, I3)으로 인가된 입력 스파이크에 시냅스의 가중치(W12, W32)가 가중되어 제2 출력 뉴런(O2)으로 전달되고, 제2 출력 뉴런(O2)은 가중치(W12, W32)가 가중되고 누적된 입력 스파이크의 전압이 발화 문턱값( $V_{th}$ ) 이상이므로, 제3 타이밍(T3)에서 출력 스파이크를 출력한다.
- [0008] 이와 같은 SNN에서는 주로 인간 신경망의 동작 특성을 모방한 LIF(Leaky-Integrate and Fire) 뉴런 모델이 이용된다. LIF 뉴런 모델은 입력되는 입력 스파이크에 따른 가중치를 누적(Integrate)하고, 누적된 가중치가 기지정된 기준 레벨 이상이면, 발화(Fire)하여 다음 뉴런으로 전달할 새로운 입력 스파이크를 생성하며, 입력 스파이크가 전달되지 않으면 누적된 입력 스파이크의 세기가 누설(Leaky)되어 시간이 흐를수록 점차 약해지도록 구성된 모델이다.
- [0009] 한편, SNN 또한 인공 신경망의 일종이므로 뉴런을 잇는 시냅스의 가중치를 업데이트하기 위한 학습이 수행되어야 하며, 특히 LIF 뉴런 모델에 기반하는 SNN의 학습에는 일반적으로 스파이크 타이밍 의존 가소성(Spike-timing-dependent plasticity: 이하 STDP) 기법이 이용되고 있다.
- [0010] STDP 기법은 출력 스파이크가 발생하는 타이밍과 입력 스파이크가 시냅스를 통해 전달된 타이밍 사이의 시간 차( $\Delta t$ )에 따라 대응하는 시냅스의 가중치를 기지정된 가중치 변화량( $\Delta w$ )만큼 증가 또는 감소시켜 업데이트 함으로써 학습을 수행하는 기법이다.
- [0011] 다만 기존에는 STDP 기법을 하드웨어로 구현하는 경우에, 출력 스파이크가 발생하는 타이밍과 입력 스파이크 전달되는 타이밍 사이의 시간 차( $\Delta t$ )를 계산하기 위해, 기지정된 기간 동안 다수의 입력 뉴런 각각으로 입력 스파이크 신호가 인가된 시간 정보인 입력 스파이크 히스토리를 저장하기 위한 다수의 입력 히스토리 레지스터와 기지정된 기간 동안 다수의 출력 뉴런 각각에서 출력 스파이크 신호가 발생한 시간 정보인 출력 스파이크 히스토리를 저장하기 위한 다수의 출력 히스토리 레지스터가 요구되었다. 또한 다수의 입력 히스토리 레지스터와 다수의 출력 히스토리 레지스터에 저장된 입력 스파이크 히스토리와 출력 스파이크 히스토리로부터 계산되는 누적 가중치 변화량( $\sum \Delta w$ )을 저장하기 위한 별도의 변화량 레지스터가 요구되었다.
- [0012] 이때 입력 히스토리 레지스터와 출력 히스토리 레지스터는 각각 기지정된 시간 구간에 대응하는 크기로 구현되어 입력 뉴런의 개수와 출력 뉴런의 개수에 대응하는 개수로 구비되어야 하고, 변화량 레지스터 또한 입력 뉴런의 개수와 출력 뉴런의 개수에 각각에 대응하는 개수로 구비되어야 한다.
- [0013] 따라서 STDP 기법을 하드웨어로 구현하고자 하는 경우, 요구되는 레지스터의 개수가 매우 많아, 하드웨어 소모가 크다는 단점이 있다. 즉 하드웨어로 구현이 어렵고, 제조 비용이 크게 상승한다는 문제가 있다.

## 선행기술문헌

### 특허문헌

- [0014] (특허문헌 0001) 한국 공개 특허 제10-2018-0062934호 (2018.06.11 공개)

## 발명의 내용

### 해결하려는 과제

- [0015] 본 발명의 목적은 소형, 저비용으로 용이하게 하드웨어로 구현할 수 있는 스파이킹 신경망 학습 장치 및 방법을 제공하는데 있다.
- [0016] 본 발명의 다른 목적은 STDP 기법에 따른 학습 수행 시 학습 윈도우 크기를 용이하게 가변할 수 있는 스파이킹 신경망 학습 장치 및 방법을 제공하는데 있다.

### 과제의 해결 수단

- [0017] 상기 목적을 달성하기 위한 본 발명의 일 실시예에 따른 스파이킹 신경망 학습 장치는 다수의 입력 뉴런과 다수의 출력 뉴런 및 각각 가중치가 저장된 다수의 시냅스를 포함하는 스파이킹 신경망을 학습시키기 위한 학습 장치에 있어서, 클럭 신호를 인가받아 카운트하여 기지정된 시간 간격 단위로 시프트 신호를 출력하는 카운터; 상기 다수의 입력 뉴런의 개수에 대응하는 개수와 양자화 STDP 함수에 의해 구분된 시간 구간인 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖고, 대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 기지정된 값을 가산하는 다수의 입력 시프트 레지스터; 상기 다수의 출력 뉴런의 개수에 대응하는 개수와 상기 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖고, 대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 기지정된 값을 가산하는 다수의 출력 시프트 레지스터; 및 상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터에 저장된 값을 누적 가중치 변화량으로 인가받아 다수의 시냅스에 저장된 가중치에 가산 또는 감산하여 가중치를 업데이트하는 가중치 업데이트부를 포함한다.
- [0018] 상기 카운터는 상기 다수의 양자화 구간의 시간 구간에 대응하는 시간 간격 단위로 상기 시프트 신호를 출력할 수 있다.
- [0019] 상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 레지스터 각각은 상기 다수의 양자화 구간의 개수에 기지정된 비트 수가 추가된 크기를 가질 수 있다.
- [0020] 상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 레지스터 각각은 상기 시프트 신호가 인가되면, 저장된 값을 최하위 비트 방향으로 1비트 단위로 시프트시킬 수 있다.
- [0021] 상기 다수의 입력 시프트 레지스터 각각은 대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 상기 다수의 양자화 구간의 개수에 대응하는 비트 위치에 1의 값을 가산할 수 있다.
- [0022] 상기 다수의 입력 시프트 레지스터 각각은 상기 다수의 출력 뉴런 중 적어도 하나의 출력 뉴런에서 출력 스파이크가 발생되면, 저장된 값을 입력 누적 가중치 변화량으로서 상기 가중치 업데이트부로 출력할 수 있다.
- [0023] 상기 다수의 출력 시프트 레지스터 각각은 대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 상기 다수의 양자화 구간의 개수에 대응하는 비트 위치에 1의 값을 가산할 수 있다.
- [0024] 상기 다수의 출력 시프트 레지스터 각각은 상기 다수의 입력 뉴런 중 적어도 하나의 입력 뉴런으로 입력 스파이크가 인가되면, 저장된 값을 출력 누적 가중치 변화량으로서 상기 가중치 업데이트부로 출력할 수 있다.
- [0025] 상기 가중치 업데이트부는 상기 다수의 입력 시프트 레지스터 각각에 저장된 값을 인가받아, 상기 다수의 시냅스 중 출력 스파이크를 발생한 출력 뉴런에 연결된 시냅스의 가중치에 가산하고, 상기 다수의 출력 시프트 레지스터 각각에 저장된 값을 인가받아, 상기 다수의 시냅스 중 입력 스파이크가 인가된 입력 뉴런에 연결된 시냅스의 가중치에 감산할 수 있다.
- [0026] 상기 다른 목적을 달성하기 위한 본 발명의 다른 실시예에 따른 스파이킹 신경망 학습 방법은 다수의 입력 뉴런과 다수의 출력 뉴런 및 각각 가중치가 저장된 다수의 시냅스를 포함하는 스파이킹 신경망을 학습시키기 위한 학습 방법에 있어서, 클럭 신호를 인가받아 카운트하여 기지정된 시간 간격 단위로 시프트 신호를 출력하는 단계; 대응하는 입력 뉴런으로 입력 스파이크가 인가되면, 상기 다수의 입력 뉴런의 개수에 대응하는 개수와 양자화 STDP 함수에 의해 구분된 시간 구간인 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖는 입력 시프트 레지스터에 기지정된 값을 가산하는 단계; 대응하는 출력 뉴런에서 출력 스파이크가 발생되면, 상기 다수의 출력 뉴런의 개수에 대응하는 개수와 상기 다수의 양자화 구간의 개수 및 크기에 대응하는 크기를 갖는 출력 시프트 레지스터에 기지정된 값을 가산하는 단계; 및 상기 다수의 입력 시프트 레지스터와 상기 다수의 출력 시프트 레지스터에 저장된 값을 누적 가중치 변화량으로 인가받아 다수의 시냅스에 저장된 가중치에 가산 또는 감



산하여 가중치를 업데이트하는 단계를 포함한다.

### 발명의 효과

[0027] 따라서, 본 발명의 실시예에 따른 스파이킹 신경망 학습 장치 및 방법은 입력 히스토리 레지스터와 출력 히스토리 레지스터를 필요로 하지 않고, 누적 가중치 변화량을 계산하여 시냅스의 가중치를 업데이트하여, 소형, 저비용으로 용이하게 하드웨어로 구현할 수 있고, 학습 윈도우 크기를 용이하게 가변할 수 있다.

### 도면의 간단한 설명

[0028] 도 1은 스파이킹 신경망의 동작 개념을 설명하기 위한 도면이다.  
 도 2는 STDP 기법을 이용한 스파이킹 신경망의 학습 개념을 설명하기 위한 도면이다.  
 도 3 내지 도 5는 양자화된 STDP 함수를 이용하여 학습되는 SNN의 일예를 나타낸다.  
 도 6은 본 발명의 일 실시예에 따른 스파이킹 신경망의 개략적 구조를 나타낸다.  
 도 7은 도 6의 가중치 학습부의 상세 구성을 나타낸다.  
 도 8은 도 7의 가중치 학습부에서 입력 시프트 레지스터의 동작을 설명하기 위한 도면이다.  
 도 9는 학습 윈도우의 크기 및 양자화 구간의 개수와 시프트 레지스터의 크기 사이의 관계를 설명하기 위한 도면이다.  
 도 10은 본 발명의 일 실시예에 따른 스파이킹 신경망 학습 방법을 나타낸다.

### 발명을 실시하기 위한 구체적인 내용

[0029] 본 발명과 본 발명의 동작상의 이점 및 본 발명의 실시에 의하여 달성되는 목적을 충분히 이해하기 위해서는 본 발명의 바람직한 실시예를 예시하는 첨부 도면 및 첨부 도면에 기재된 내용을 참조하여야만 한다.

[0030] 이하, 첨부한 도면을 참조하여 본 발명의 바람직한 실시예를 설명함으로써, 본 발명을 상세히 설명한다. 그러나, 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며, 설명하는 실시예에 한정되는 것이 아니다. 그리고, 본 발명을 명확하게 설명하기 위하여 설명과 관계없는 부분은 생략되며, 도면의 동일한 참조부호는 동일한 부재임을 나타낸다.

[0031] 명세서 전체에서, 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라, 다른 구성요소를 더 포함할 수 있는 것을 의미한다. 또한, 명세서에 기재된 "...부", "...기", "모듈", "블록" 등의 용어는 적어도 하나의 기능이나 동작을 처리하는 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현될 수 있다.

[0032] 도 2는 STDP 기법을 이용한 스파이킹 신경망의 학습 개념을 설명하기 위한 도면이다.

[0033] 도 2에서도 도 1과 유사하게, (a)는 다수의 입력 뉴런(I1 ~ I3)에 입력 스파이크가 인가되는 타이밍을 나타내고, (b)는 다수의 입력 뉴런(I1 ~ I3)과 하나의 출력 뉴런(O2) 사이에 연결되는 시냅스의 가중치(W12, W22, W32)를 나타내며, (c)는 STDP 기법에서 출력 스파이크가 발생하는 타이밍과 입력 스파이크가 인가되는 타이밍 사이의 시간 차이( $\Delta t$ )에 따라 대응하는 시냅스에서 업데이트되는 가중치 변화량( $\Delta w$ )을 나타낸다.

[0034] (c)에 그래프로 도시된 바와 같이, STDP 기법에서는 기지정된 STDP 함수에 따라 출력 뉴런(O2)이 출력 스파이크를 발생하기 이전 입력 스파이크를 전달한 시냅스는 출력 스파이크의 발생과 관련성이 높다고 판단하여 가중치(W)를 증가( $\Delta w > 0$ )시키는 장기 강화(Long Term Potentiation: 이하 LTP)와 출력 스파이크가 발생된 이후 입력 스파이크를 전달한 시냅스는 출력 스파이크의 발생과 관련성이 낮다고 판단하여 가중치(W)를 감소( $\Delta w < 0$ )시키는 장기 저하(Long Term Depression: LTD)로 구성될 수 있다. 여기서 가중치 변화량( $\Delta w$ )의 절대값( $|\Delta w|$ )은 도 2의 (c)에 도시된 바와 같이, 출력 스파이크와 입력 스파이크 사이의 타이밍 차이( $\Delta t$ )의 절대값( $|\Delta t| = |t_{\text{outputspike}} - t_{\text{inputspike}}|$ )이 작을 수록 크다. 즉 제2 출력 뉴런(O2)에서 출력 스파이크가 발생된 제3 타이밍(T3)을 기준으로 제1 입력 뉴런(I1)에 입력 스파이크가 인가된 제1 타이밍(T1)과의 타이밍 차이보다 제3 입력 뉴런(I3)에 입력 스파이크가 인가된 제2 타이밍(T2)과의 타이밍 차이가 더 작으므로, 가중치 변화량( $\Delta w$ )의 절대값이 더 크게 나타남을 알 수 있다. 그리고 제2 입력 뉴런(I2)에는 출력 스파이크가 발생된 이후인 제4 타이밍에 입력 스파이크가 인가되므로, 가중치 변화량( $\Delta w$ )이 음의 값으로 나타남을 알 수 있다.



- [0035] 다만, 도 2의 (c)와 같이 연속적인 STDP 함수에 따른 가중치 변화량( $\Delta w$ )을 그대로 하드웨어로 구현하는 것은 매우 어렵기 때문에 일반적으로 양자화(quantization)된 STDP 함수를 적용한다.
- [0036] 도 3 내지 도 5는 양자화된 STDP 함수를 이용하여 학습되는 SNN의 일례를 나타낸다.
- [0037] 도 3에서는 784개의 입력 뉴런(I1~ I784)과 784개의 출력 뉴런(O1 ~ O784) 및 입력 뉴런(I1~ I784)과 출력 뉴런(O1 ~ O784)을 각각 서로 연결하는 시냅스를 포함하는 SNN을 도시하였다. 그리고 SNN의 학습을 위해 입력 뉴런(I1~ I784)의 개수에 대응하는 개수의 입력 히스토리 레지스터(Input history register)와 출력 뉴런(O1 ~ O784)의 개수에 대응하는 개수의 출력 히스토리 레지스터(Input history register)가 더 구비된다. 입력 히스토리 레지스터는 기지정된 시간 구간 동안 입력 스파이크가 인가된 타이밍을 저장하기 위한 레지스터이고, 출력 히스토리 레지스터는 기지정된 시간 구간 동안 출력 스파이크가 발생된 타이밍을 저장하기 위한 레지스터이다. 또한 입력 히스토리 레지스터와 출력 히스토리 레지스터는 시프트 레지스터로 구현되어 인가된 입력 스파이크 또는 발생된 출력 스파이크를 기지정된 시간 간격 단위(여기서는 클럭 또는 사이클 단위)로 이동시킨다.
- [0038] 도 4에서는 STDP 함수가 20클럭과 40 클럭을 기준으로 서로 다른 가중치 변화량( $\Delta w$ )을 갖도록 2개의 양자화 구간으로 양자화된 STDP 함수를 도시하였으며, 이 경우, 입력 히스토리 레지스터와 출력 히스토리 레지스터 각각은 입력 스파이크가 인가된 타이밍 또는 출력 스파이크가 발생된 타이밍을 40클럭까지 저장할 수 있어야 한다. 즉 입력 히스토리 레지스터와 출력 히스토리 레지스터는 양자화된 STDP 함수에 의해 지정된 시간 구간에 대응하는 크기를 가져야 한다.
- [0039] 도 4에서 시간 차( $\Delta t$ )가 0인 위치를 기준으로 양의 값을 갖는 LTP에서 가중치 변화량( $\Delta w$ )이 0이 될 때까지, 또는 음의 값을 갖는 LTD에서 가중치 변화량( $\Delta w$ )이 0이 될 때까지를 각각 학습 윈도우(learning window)라고 한다. 도 4에서는 일 예로 LTP 학습 윈도우와 LTD 학습 윈도우 각각이 20클럭 단위로 2개의 양자화 구간으로 구분되는 것으로 도시하였으나, 양자화된 STDP 함수에서 학습 윈도우의 크기와 양자화 구간의 개수 및 각 양자화 구간에 대응하는 가중치 변화량( $\Delta w$ )은 조절될 수 있다.
- [0040] 그리고 도 5의 (a)에 도시된 바와 같이, 다수의 입력 히스토리 레지스터에 저장된 입력 스파이크 타이밍은 다수의 출력 뉴런 중 특정 출력 뉴런에서 출력 스파이크가 발생되면, 도 4의 양자화된 STDP 함수의 LTP에 기반하여 타이밍에 따라 서로 다르게 적용되는 가중치 변화량( $\Delta w$ )이 가산되어 입력 변화량 레지스터에 저장된다. 여기서 입력 변화량 레지스터는 다수의 입력 히스토리 레지스터 각각에 대응하는 개수로 구비되어 대응하는 입력 히스토리 레지스터에 저장된 입력 스파이크 타이밍에 따른 가중치 변화량( $\Delta w$ )을 가산한 누적 가중치 변화량( $\sum \Delta w$ )을 획득하고, 획득된 누적 가중치 변화량( $\sum \Delta w$ )을 대응하는 시냅스의 가중치( $w$ )에 더하여 시냅스의 가중치( $w + \sum \Delta w$ )를 업데이트 한다.
- [0041] 또한 도 5의 (b)에 도시된 바와 같이, 다수의 출력 히스토리 레지스터에 저장된 출력 스파이크 타이밍은 다수의 입력 뉴런 중 특정 입력 뉴런에서 입력 스파이크가 인가되면, 도 4의 양자화된 STDP 함수의 LTD에 기반하여 타이밍에 따라 서로 다르게 적용되는 가중치 변화량( $\Delta w$ )이 가산되어 출력 변화량 레지스터에 저장된다. 출력 변화량 레지스터 또한 다수의 출력 히스토리 레지스터 각각에 대응하는 개수로 구비되어, 대응하는 출력 히스토리 레지스터에 저장된 출력 스파이크 타이밍에 따른 가중치 변화량( $\Delta w$ )을 가산한 누적 가중치 변화량( $\sum \Delta w$ )을 획득하고, 획득된 누적 가중치 변화량( $\sum \Delta w$ )을 대응하는 시냅스의 가중치( $w$ )에 더하여 시냅스의 가중치( $w + \sum \Delta w$ )를 업데이트 한다.
- [0042] 결과적으로, 도 3 내지 도 5에서는 기지정된 크기로 입력 히스토리 레지스터와 출력 히스토리 레지스터가 다수의 입력 뉴런 및 다수의 출력 뉴런 각각에 대응하는 개수로 요구될 뿐만 아니라, 입력 변화량 레지스터와 출력 변화량 레지스터 또한 입력 뉴런 및 출력 뉴런의 개수로 요구된다. 따라서 이를 하드웨어로 구현하는 경우, 대량의 레지스터가 필요하게 된다.
- [0043] 도 6은 본 발명의 일 실시예에 따른 스파이킹 신경망의 개략적 구조를 나타낸다.
- [0044] 도 6을 참조하면, SNN은 입력 뉴런부(100)와 가중치 시냅스부(200), 출력 뉴런부(300) 및 가중치 학습부(400)를 포함한다.
- [0045] 입력 뉴런부(100)는 다수의 입력 뉴런(미도시)을 포함하고, 입력 데이터가 인가되면, 인가된 입력 데이터를 기지정된 방식에 따라 인코딩하여 스파이크 트레인(spike train)의 형태의 입력 스파이크로 변환하여 대응하는 입력 뉴런으로 인가한다.
- [0046] 이때 입력 뉴런부(100)는 포아송 인코딩(Poisson encoding) 기법에 따라 입력 데이터를 인코딩하여 입력 스파이크

크로 변환할 수 있다. 포아송 인코딩은 SNN에서 입력 데이터를 변환하기 위해 주로 사용되는 인코딩 기법으로 포아송 분포(Poisson distribution)에 따른 확률 기반하여 각각 시간차를 두고 발생하는 희소 입력 스파이크로 생성될 수 있다. 일 예로 입력 뉴런부(100)는 이미지에서 대응하는 각 픽셀의 픽셀값들을 입력 데이터로 인가 받고, 인가된 픽셀값에 따라 시간차를 두고 발생하는 입력 스파이크를 출력할 수 있다.

[0047] 여기서 입력 뉴런부(100)는 이진 데이터의 형식으로 입력 스파이크를 생성하는 것으로 가정한다. 일 예로 입력 스파이크가 발생되면 1의 값을 출력하고, 입력 스파이크가 발생되지 않으면 0의 값을 출력하도록 설정될 수 있다.

[0048] 또한 입력 뉴런부(100)는 이진 데이터 형식으로 다수의 입력 스파이크를 인가받도록 구성될 수도 있으며, 이 경우 입력 뉴런부(100)는 다수의 입력 스파이크를 버퍼링하여 출력하는 버퍼로 구현될 수 있다.

[0049] 그리고 입력 스파이크가 인가된 다수의 입력 뉴런 각각은 인가된 입력 스파이크를 가중치 시냅스부(200)의 대응하는 시냅스로 전달한다.

[0050] 가중치 시냅스부(200)는 SNN에서 두뇌의 시냅스를 모방한 구성으로, 다수의 가중치를 저장하는 메모리 어레이와 유사한 구조를 가질 수 있다. 일 예로 가중치 시냅스부(200)는 SRAM(Static Random access memory) 등으로 구현될 수 있다.

[0051] 가중치 시냅스부(200)의 다수의 시냅스 각각은 대응하는 가중치를 저장하고, 입력 뉴런부(100)에서 입력 스파이크가 인가되면, 인가된 입력 스파이크에 대응하는 가중치를 출력 뉴런부(300)로 출력한다. 여기서 가중치 시냅스부(200)에 저장된 다수의 가중치는 가중치 학습부(400)에 의해 업데이트된다.

[0052] 가중치 시냅스부(200)의 다수의 시냅스 각각은 입력 뉴런부(100)로부터 시간차를 두고 발생되어 인가되는 다수의 입력 스파이크 각각에 응답하여 대응하는 가중치를 반복적으로 출력 뉴런부(300)의 대응하는 출력 뉴런으로 전달한다. 즉 입력 뉴런부(100)에서 입력 스파이크가 발생될 때마다 저장된 가중치 중 대응하는 가중치를 출력 뉴런부(300)로 출력한다.

[0053] 출력 뉴런부(300)의 다수의 출력 뉴런 각각은 가중치 시냅스부(200)에서 출력되는 가중치의 누적 합에 기초하여 다수의 출력 스파이크를 생성한다. 다수의 출력 뉴런은 상기한 LIF 뉴런 모델에 기반하여 구성될 수 있다.

[0054] 다수의 출력 뉴런 각각은 적어도 하나의 입력 스파이크가 발생될 때마다 대응하는 시냅스부로부터 인가되는 가중치( $w$ )를 모두 합하고, 가중 입력 스파이크의 합에 이전 획득된 누적 스파이크를 더 가산한다. 이때 출력 뉴런부(300)는 입력 스파이크의 합에 이전 획득된 누적 스파이크를 그대로 더하는 것이 아니라, LIF 기법의 누설을 구현하기 위해, 이전 획득된 누적 스파이크에 기지정된 누설 팩터( $\lambda$ )(여기서  $0 < \lambda < 1$ 인 실수)를 가중한 누설 누적 스파이크를 더하여 누적 스파이크를 획득할 수 있다.

[0055] 그리고 누적 스파이크가 기지정된 발화 문턱값( $V_{th}$ )보다 전압 레벨이 높으면, 출력 스파이크를 발화하여 출력한다.

[0056] 가중치 학습부(400)는 SNN을 학습시키기 위한 학습 장치에 해당하는 구성으로 상기한 STDP 기법에 따라 가중치 시냅스부(200)의 다수의 시냅스 각각에 대한 가중치를 업데이트하여 시냅스를 학습시킨다. STDP 기법에 따라 가중치를 학습시키는 가중치 학습부(400)는 출력 스파이크가 발생된 시점을 기준으로 이전 기지정된 기간 동안 입력 뉴런부(100)으로부터 가중치 시냅스부(200)로 입력 스파이크가 입력된 시간 정보를 기반으로 가중치를 증가시킨다. 한편, 가중치 학습부(400)는 입력 뉴런부(100)으로부터 가중치 시냅스부(200)로 입력 스파이크가 입력되면, 입력 스파이크가 입력된 이전 기지정된 기간 동안 출력 스파이크가 발생된 시간 정보를 기반으로 가중치를 감소시킨다. 즉 가중치 학습부(400)는 인가되는 입력 스파이크와 출력 스파이크의 발생 시간에 기반하여 가중치를 증가 또는 감소시킨다.

[0057] 이때 가중치 학습부(400)는 상기한 바와 같이, 양자화 STDP 함수에 기반하여 가중치( $w$ )의 가중치 변화량( $\Delta w$ )을 조절하여 가중치를 업데이트 함으로써 시냅스를 학습시킬 수 있다.

[0058] 도 7은 도 6의 가중치 학습부의 상세 구성을 나타내고, 도 8은 도 7의 가중치 학습부에서 입력 시프트 레지스터의 동작을 설명하기 위한 도면이다.

[0059] 도 7을 참조하면, 가중치 학습부(400)는 카운터(410)와 입력 시프트 레지스터부(420), 출력 시프트 레지스터부(430) 및 가중치 업데이트부(440)를 포함할 수 있다.

[0060] 카운터(410)는 클럭 신호( $clk$ )를 인가받아 카운트하고, 카운트된 클럭 신호의 개수가 기지정된 개수에

도달하면, 시프트 신호(sf)를 입력 시프트 레지스터부(420)와 출력 시프트 레지스터부(430)로 출력한다.

- [0061] 여기서 카운터(410)는 양자화 STDP 함수에 의해 지정된 시간 구간에 따라 클럭 개수를 카운트하여 시프트 신호(sf)를 출력하도록 설정될 수 있다. 일 예로 도 4에 도시된 바와 같이, 양자화 STDP 함수가 20 클럭(20clk) 단위로 가중치 변화량( $\Delta w$ )을 가변하는 경우, 카운터(410)는 20 클럭(20clk)마다 시프트 신호(sf)를 출력하도록 설정될 수 있으며, 양자화 STDP 함수가 40 클럭(40clk) 단위로 가중치 변화량( $\Delta w$ )을 가변하는 경우, 카운터(410)는 40 클럭(40clk)마다 시프트 신호(sf)를 출력하도록 설정될 수 있다.
- [0062] 입력 시프트 레지스터부(420)와 출력 시프트 레지스터부(430)는 각각 입력 뉴런부(100)의 입력 뉴런의 개수에 대응하는 개수의 입력 시프트 레지스터와 출력 뉴런부(300)의 출력 뉴런의 개수에 대응하는 개수의 출력 시프트 레지스터를 포함한다. 그리고 입력 시프트 레지스터와 출력 시프트 레지스터 각각은 양자화 STDP 함수에서 지정되는 학습 윈도우의 크기와 학습 윈도우 내에 가중치 변화량( $\Delta w$ )이 구분되는 양자화 구간의 개수에 대응하는 크기를 갖는다.
- [0063] 본 발명에서 입력 시프트 레지스터와 출력 시프트 레지스터는 학습 윈도우에 포함되는 양자화 구간의 개수와 기 지정된 비트수(예를 들면 2비트)의 합에 대응하는 비트수를 갖도록 구현될 수 있다. 여기서 기지정된 비트수는 학습 윈도우에서 양자화 구간의 크기에 따라 조절될 수 있다.
- [0064] 일 예로 도 4에서와 같이, 40 클럭 크기의 학습 윈도우가 서로 다른 2개의 가중치 변화량( $\Delta w$ )을 갖는 2개의 양자화 구간으로 구분되는 경우를 가정하였다. 이 경우, 입력 시프트 레지스터와 출력 시프트 레지스터는 각각 4( $= 2+2$ )비트 크기의 시프트 레지스터로 구현될 수 있다. 그러나 만일 학습 윈도우가 60 클럭 크기를 갖고 3개의 양자화 구간으로 구분되는 경우, 입력 시프트 레지스터와 출력 시프트 레지스터는 각각 5( $= 3+2$ )비트 크기의 시프트 레지스터로 구현될 수도 있다. 또한 학습 윈도우가 1000 클럭 크기를 갖고 2개의 양자화 구간으로 구분되는 경우, 입력 시프트 레지스터와 출력 시프트 레지스터는 각각 5( $= 2+3$ )비트 크기의 시프트 레지스터로 구현될 수도 있다.
- [0065] 학습 윈도우의 크기 및 양자화 구간의 개수와 입력 시프트 레지스터와 출력 시프트 레지스터의 크기 사이의 관계는 후술하도록 한다.
- [0066] 다수의 입력 시프트 레지스터 각각은 대응하는 뉴런으로 입력 스파이크(IS)가 인가되면, 입력 시프트 레지스터의 기지정된 비트 위치에 1의 값을 가산한다. 즉 입력 시프트 레지스터에 저장된 위치에 미리 지정된 값을 가산한다. 입력 시프트 레지스터는 양자화 STDP 함수에서 LTP에 기반하여 가중치 변화량( $\Delta w$ )을 가산하며, 이에 학습 윈도우를 구분한 다수의 양자화 구간 중 시간 차( $\Delta t$ )가 0인 위치를 초기 양자화 구간에 대응하는 가중치 변화량( $\Delta w$ )을 이전 저장된 값에 가산한다. 입력 시프트 레지스터부(420)는 입력 스파이크(IS)가 인가될 때마다 동일한 값을 가산하므로, 매번 동일 비트 위치에 1을 가산한다. 그리고 카운터(410)에서 시프트 신호(sf)가 인가될 때마다 저장된 값을 지정된 방향으로 시프트시킨다.
- [0067] 도 8에서 (a)는 입력 시프트 레지스터의 일 예를 나타내고, (b)는 입력 스파이크가 인가되는 타이밍에 따라 입력 시프트 레지스터에 저장된 값을 나타낸다. 도 8에서는 입력 시프트 레지스터부(420)가 입력 뉴런의 개수에 대응하여 784개의 입력 시프트 레지스터를 포함하는 경우를 가정하여 도시하였다. 그리고 도 8에서도 입력 시프트 레지스터가 도 4의 양자화 STDP 함수를 기반으로 동작하는 것으로 가정한다.
- [0068] 도 4의 예에서 초기 양자화 구간의 가중치 변화량( $\Delta w$ )은 +2이므로, 입력 시프트 레지스터는 입력 스파이크(IS)가 인가되면, 2에 대응하는 비트 위치에 1의 값을 가산한다. 이에 도 8의 (b)에서 제1 타이밍( $T = t_1$ )에 인가된 입력 스파이크에 응답하여, 입력 시프트 레지스터는 최우측의 최하위 비트(Least Significant Bit: LSB) 위치로부터 좌측 첫번째 위치한 비트 위치에 1을 가산하였다.
- [0069] 그리고 카운터(410)로부터 제2 타이밍( $T = t_2$ )에 시프트 신호(sf)가 인가되면, 입력 시프트 레지스터는 각 비트 위치에 저장된 값을 최하위 비트 방향으로 1비트 시프트시킨다. 시프트 레지스터에서 최하위 비트 방향으로 1비트 시프트시키는 것은 수학적으로 시프트 레지스터에 저장된 값을 2로 나누는 연산을 수행하는 것과 유사하므로, 입력 시프트 레지스터에 저장된 값은 1로 감소된다.
- [0070] 한편 제3 타이밍( $T = t_3$ )과 제4 타이밍( $T = t_4$ )에 각각 입력 스파이크가 다시 인가되면, 입력 시프트 레지스터는 제3 및 제4 타이밍( $t_3, t_4$ )에 인가된 입력 스파이크 각각에 응답하여, 연속적으로 +2를 가산한다.
- [0071] 이와 같이 입력 시프트 레지스터는 입력 스파이크가 인가될 때마다 기지정된 값을 가산하고, 카운터(410)로부터 시프트 신호(sf)가 인가될 때마다 저장된 값을 2로 나누는 연산이 수행되도록 비트 시프트를 수행한다. 이는

출력 스파이크가 발생되기 이전 입력 스파이크를 전달한 시냅스는 출력 스파이크의 발생과 관련성이 높다고 판단하여 가중치( $w$ )를 증가시키는 LTP를 구현하는 것으로, 입력 시프트 레지스터에 항상 누적 가중치 변화량( $\sum \Delta w$ )을 저장하고 있는 것으로 볼 수 있다.

[0072] 그리고 다수의 출력 뉴런 중 적어도 하나의 출력 뉴런에서 출력 스파이크(OS)가 발생되면, 입력 시프트 레지스터는 현재 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 출력한다.

[0073] 한편, 도시하지 않았으나, 출력 시프트 레지스터는 입력 시프트 레지스터와 유사하게, 출력 스파이크가 인가될 때마다 기지정된 값을 가산하고, 카운터(410)로부터 시프트 신호(sf)가 인가될 때마다 저장된 값을 2로 나누는 연산이 수행되도록 비트 시프트를 수행한다. 그리고 이러한 동작은 출력 스파이크가 발생된 이후 입력 스파이크를 전달한 시냅스는 출력 스파이크의 발생과 관련성이 낮다고 판단하여 가중치( $w$ )를 감소가시키는 LTD를 구현하는 것으로, 출력 시프트 레지스터에도 항상 누적 가중치 변화량( $\sum \Delta w$ )을 저장하고 있는 것으로 볼 수 있다.

[0074] 다만 출력 시프트 레지스터의 경우, 양자화 STDP 함수에 따라 음의 값을 갖는 누적 가중치 변화량( $\sum \Delta w$ )이 저장되어야 하므로, 누적 가중치 변화량( $\sum \Delta w$ )의 절대값( $|\sum \Delta w|$ )이 저장된 것으로 볼 수 있다. 그리고 다수의 입력 뉴런 중 적어도 하나의 입력 뉴런에 입력 스파이크(IS)가 인가되면, 입력 시프트 레지스터는 현재 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 출력한다.

[0075] 여기서는 설명의 편의를 위해, 입력 시프트 레지스터에 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 입력 누적 가중치 변화량으로, 그리고 출력 시프트 레지스터에 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 출력 누적 가중치 변화량으로 구분하여 표현할 수 있다.

[0076] 가중치 업데이트부(440)는 입력 시프트 레지스터에서 입력 누적 가중치 변화량( $\sum \Delta w$ )이 인가되면, 출력 스파이크(OS)가 발생된 출력 뉴런에 연결된 다수의 시냅스의 현재 가중치( $w(t)$ )를 인가받고, 인가된 가중치( $w(t)$ )에 가중치 변화량( $\Delta w$ )을 가산하여 업데이트 가중치( $w(t+1)$ )를 획득한다.

[0077] 반면, 출력 시프트 레지스터에서 출력 누적 가중치 변화량( $\sum \Delta w$ )이 인가되면, 입력 스파이크(IS)가 발생된 입력 뉴런에 연결된 다수의 시냅스의 현재 가중치( $w(t)$ )를 인가받고, 인가된 가중치( $w(t)$ )에 가중치 변화량( $\Delta w$ )을 감산하여 업데이트 가중치( $w(t+1)$ )를 획득한다.

[0078] 그리고 획득된 업데이트 가중치( $w(t+1)$ )를 대응하는 시냅스의 현재 가중치( $w(t)$ )를 대체하여 저장함으로써 가중치를 업데이트 한다. 즉 SNN을 학습시킨다.

[0079] 상기한 바와 같이, 본 실시예에서는 입력 시프트 레지스터부(420)의 다수의 입력 시프트 레지스터와 출력 시프트 레지스터부(430)의 다수의 출력 시프트 레지스터가 작은 크기의 시프트 레지스터로 구현됨에도 각각 LTP와 LTD를 위한 누적 가중치 변화량을 항상 저장하고 있으므로, 입력 스파이크가 인가되거나 출력 스파이크가 발생되면 즉시 누적 가중치 변화량( $\sum \Delta w$ )을 가중치 업데이트부(440)로 전달하여 대응하는 시냅스의 가중치를 실시간으로 업데이트 할 수 있다.

[0080] 다만, 입력 시프트 레지스터와 출력 시프트 레지스터 각각이 입력 스파이크(IS)가 인가된 정확한 타이밍과 출력 스파이크(OS)가 발생된 정확한 타이밍을 저장하지 않고, 카운터(410)로부터 인가되는 시프트 신호(sf)에 따라 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 감소시키는데 따른 누적 가중치 변화량( $\sum \Delta w$ )의 오차가 일부 발생할 수 있다. 그러나 상기한 바와 같이, 포아송 인코딩에 따라 생성되는 입력 스파이크(IS)와 출력 스파이크(OS)에 응답하여 발생하는 출력 스파이크(OS)는 매우 낮은 확률로 발생되기 때문에, 시프트 신호(sf)에 따라 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 감소시키는데 따른 오차는 크게 발생되지 않는다. 일반적으로 SNN에서 양자화 STDP 함수의 학습 윈도우 구간 동안, 포아송 인코딩에 따라 입력 스파이크(IS)가 발생하는 경우는 10% 미만이다. 즉 대부분의 학습 윈도우 구간에 입력 스파이크(IS)는 1개 이하로 존재한다. 이는 출력 스파이크(OS)에서도 유사하게 나타난다. 따라서 시프트 신호(sf)에 따라 저장된 누적 가중치 변화량( $\sum \Delta w$ )을 감소시키는데 따른 오차는 시냅스의 학습에 큰 영향을 미치지 않는다.

[0081] 반면, 입력 시프트 레지스터와 출력 시프트 레지스터를 이용하여 누적 가중치 변화량( $\sum \Delta w$ )을 획득하는 경우, 도 3에 비해 요구되는 레지스터의 크기를 크게 줄일 수 있다.



표 1

Hardware			Fig. 5	Fig. 8
Register	Shift register	Input neuron	$784 \times 40$	$784 \times 4$
		Output neuron	$256 \times 40$	$256 \times 4$
	Register	Input neuron	$784 \times 5$	No need
		Output neuron	$256 \times 5$	No need
Total			46800	4160

[0082]

[0083]

표 1은 도 5와 도 8의 방식으로 가중치 학습부(400)를 구현하는 경우에, 요구되는 레지스터의 크기를 나타낸다. 표 1에서는 입력 뉴런의 개수가 784개이고, 출력 뉴런의 개수가 256개인 경우를 가정하였다. 이 경우, 도 5와 같이 가중치 학습부(400)를 구현하는 경우, 시프트 레지스터로 구현되는 입력 히스토리 레지스터와 출력 히스토리 레지스터가 각각 입력 뉴런과 출력 뉴런의 개수만큼 구비될 뿐만 아니라, 각각 학습 윈도우의 크기(40 clk)에 대응하는 크기를 가져야 한다. 또한 입력 변화량 레지스터와 출력 변화량 레지스터가 누적 가중치 변화량( $\sum \Delta w$ )을 저장할 수 있는 크기를 가져야 한다. 이에 도 5의 방식을 이용하는 경우, 가중치 학습부(400)는 총 46800 크기의 레지스터가 요구된다. 그에 반해 도 8과 같이, 입력 시프트 레지스터와 출력 시프트 레지스터를 이용하여 가중치 학습부(400)를 구성하는 경우, 총 4160 크기의 레지스터가 요구된다. 즉 가중치 학습부(400)에 요구되는 레지스터의 크기를 91.1%나 줄일 수 있다.

[0084]

도 9는 학습 윈도우의 크기 및 양자화 구간의 개수와 시프트 레지스터의 크기 사이의 관계를 설명하기 위한 도면이다.

[0085]

상기한 바와 같이, 입력 시프트 레지스터와 출력 시프트 레지스터의 크기는 학습 윈도우의 크기와 학습 윈도우 내에 가중치 변화량( $\Delta w$ )이 구분되는 양자화 구간의 개수에 따라 조절될 수 있다. 도 9의 (a)에 도시된 바와 같이, 학습 윈도우 내에 양자와 구간이 2개로 구분되고 학습 윈도우의 전체 크기가 40 클럭(40 clk)인 경우, 입력 시프트 레지스터와 출력 시프트 레지스터는 입력 스파이크가 인가되거나 출력 스파이크가 발생되면, (b)에 도시된 바와 같이 최하위 비트(LSB)의 좌측 첫번째 비트 위치에 1의 값을 저장해야 한다. 그러나, (c)와 같이 학습 윈도우 내에 양자와 구간이 2개로 구분되고 학습 윈도우의 전체 크기가 40 클럭(40 clk)인 경우, (d)와 같이 최하위 비트(LSB)의 좌측 두번째 비트 위치에 1의 값을 저장해야 한다. 즉 학습 윈도우 내에 양자화 구간의 개수가 증가할수록 입력 시프트 레지스터와 출력 시프트 레지스터의 크기는 증가되어야 한다. 또한 양자화 구간의 개수에 대응하는 비트 수를 제외한 나머지 비트 수는 양자화 구간 내에 발생하는 입력 스파이크 또는 출력 스파이크의 모든 개수를 표현할 수 있어야 한다. 일 예로, 도 9의 (b) 및 (d)에서는 양자화 구간에 대응하는 비트 수로 각각 2비트와 3비트가 할당되었으므로, 나머지 비트 수는 2비트이다. 그리고 2비트로 표현 가능한 최대 개수는 3이다. 따라서 도 9의 (b) 및 (d)에 표시된 입력 시프트 레지스터와 출력 시프트 레지스터는 양자와 구간 내에 4개 이상의 입력 스파이크 또는 출력 스파이크가 발생되면, 누적 가중치 변화량( $\sum \Delta w$ )에 오류를 발생시키게 된다.

[0086]

상기한 바와 같이 포아송 인코딩의 경우, 하나의 양자화 구간에 하나의 입력 스파이크 또는 출력 스파이크가 발생될 확률조차 낮으므로, 일반적으로 양자와 구간 내에 4개 이상의 입력 스파이크 또는 출력 스파이크가 발생될 가능성은 극히 희박하다고 볼 수 있다. 이에 본 실시예에서는 입력 시프트 레지스터와 출력 시프트 레지스터가 양자화 구간의 개수와 기지정된 비트수(여기서는 2비트)의 합에 대응하는 비트수를 갖는 시프트 레지스터로 구현되는 것으로 가정하였다. 그럼에도 불구하고, 비록 SNN에서 적용될 가능성이 매우 낮지만 각 양자화 구간의 크기가 매우 크다면(예를 들면 100 clk), 기지정된 비트수를 증가시킴으로써, 누적 가중치 변화량( $\sum \Delta w$ )에 오류를 발생시키는 것을 용이하게 방지할 수 있다.

[0087]

이와 같이, 본 실시예에 따른 가중치 학습부(400)는 학습 윈도우의 크기와 학습 윈도우 내의 양자화 구간의 개수가 조절되더라도, 입력 시프트 레지스터와 출력 시프트 레지스터의 크기를 매우 작게 증가시킴으로써 용이하게 대응할 수 있다.

[0088]

도 10은 본 발명의 일 실시예에 따른 스파이킹 신경망 학습 방법을 나타낸다.

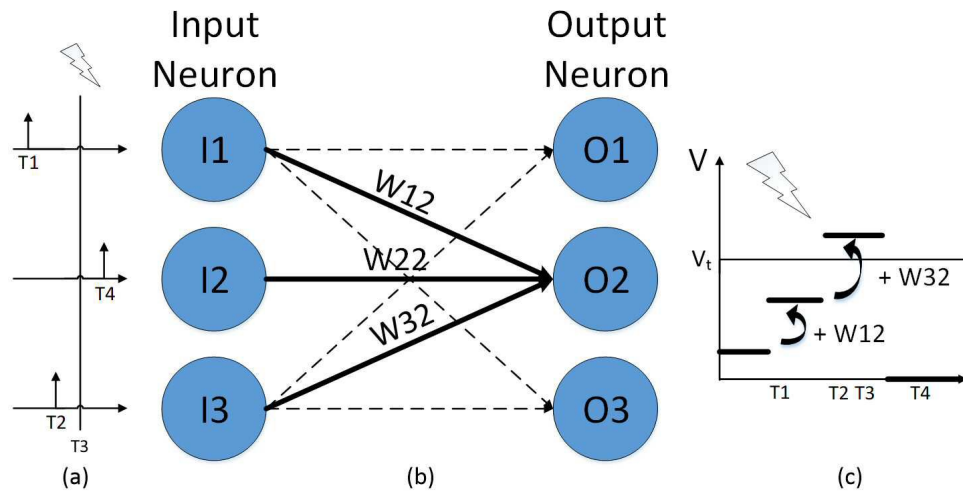
- [0089] 도 6 내지 도 9를 참조하여, 도 10의 SNN 학습 방법을 설명하면, 우선 양자화 STDP 함수의 학습 윈도우를 구성하는 양자화 구간 개수에 대응하는 기지정된 비트수를 갖도록 구성된 다수의 입력 시프트 레지스터 각각은 입력 뉴런부(100)의 다수의 입력 뉴런 중 대응하는 입력 뉴런으로 입력 스파이크(IS)가 인가되는지 판별한다(S11). 만일 입력 스파이크(IS)가 인가되는 것으로 판별되면, 양자화 STDP 함수의 학습 윈도우를 구성하는 양자화 구간 개수에 대응하는 값을 기저장된 값에 가산한다(S12). 이때, 입력 시프트 레지스터는 양자화 구간 개수에 대응하는 비트 위치에 1을 가산하여 양자화 구간 개수에 대응하는 값을 가산할 수 있다.
- [0090] 한편, 입력 시프트 레지스터와 대응하는 비트수를 갖는 다수의 출력 시프트 레지스터 각각은 입력 스파이크(I S)에 응답하여, 기저장된 값을 출력 누적 가중치 변화량으로 획득한다(S13).
- [0091] 그리고 출력 시프트 레지스터에 대응하는 출력 뉴런에 연결된 다수의 시냅스 각각의 현재 가중치(w)에 획득된 출력 누적 가중치 변화량( $\sum \Delta w$ )을 감산하여 각 시냅스의 가중치를 업데이트 한다(S14).
- [0092] 또한 다수의 출력 시프트 레지스터 각각은 출력 뉴런부(300)의 다수의 출력 뉴런 중 대응하는 출력 뉴런에서 출력 스파이크(OS)가 발생되는지 판별한다(S15). 만일 출력 스파이크(OS)가 발생되면, 양자화 STDP 함수의 학습 윈도우를 구성하는 양자화 구간 개수에 대응하는 값을 기저장된 값에 가산한다(S16). 이때, 출력 시프트 레지스터 또한 양자화 구간 개수에 대응하는 비트 위치에 1을 가산하여 양자화 구간 개수에 대응하는 값을 가산할 수 있다.
- [0093] 한편, 다수의 입력 시프트 레지스터 각각은 출력 스파이크에 응답하여, 기저장된 값을 입력 누적 가중치 변화량으로 획득한다(S17).
- [0094] 그리고 입력 시프트 레지스터에 대응하는 입력 뉴런에 연결된 다수의 시냅스 각각의 현재 가중치(w)에 획득된 출력 누적 가중치 변화량( $\sum \Delta w$ )을 가산하여 각 시냅스의 가중치를 업데이트 한다(S18).
- [0095] 또한 만일 카운터(410)가 기지정된 클럭 개수에 대응하는 주기로 발생하는 시프트 신호(sf)가 인가되는지 판별한다(S19). 만일 시프트 신호(sf)가 인가되면, 다수의 입력 시프트 레지스터와 다수의 출력 시프트 레지스터 각각은 저장된 값을 최하위 비트 방향으로 1비트 시프트시켜, 입력 누적 가중치 변화량( $\sum \Delta w$ )과 출력 누적 가중치 변화량( $\sum \Delta w$ )을 감소시킨다(S20).
- [0096] 본 발명에 따른 방법은 컴퓨터에서 실행시키기 위한 매체에 저장된 컴퓨터 프로그램으로 구현될 수 있다. 여기서 컴퓨터 판독가능 매체는 컴퓨터에 의해 액세스 될 수 있는 임의의 가용 매체일 수 있고, 또한 컴퓨터 저장 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함하며, ROM(판독 전용 메모리), RAM(랜덤 액세스 메모리), CD(컴팩트 디스크)-ROM, DVD(디지털 비디오 디스크)-ROM, 자기 테이프, 플로피 디스크, 광데이터 저장장치 등을 포함할 수 있다.
- [0097] 본 발명은 도면에 도시된 실시예를 참고로 설명되었으나 이는 예시적인 것에 불과하며, 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다.
- [0098] 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 청구범위의 기술적 사상에 의해 정해져야 할 것이다.

### 부호의 설명

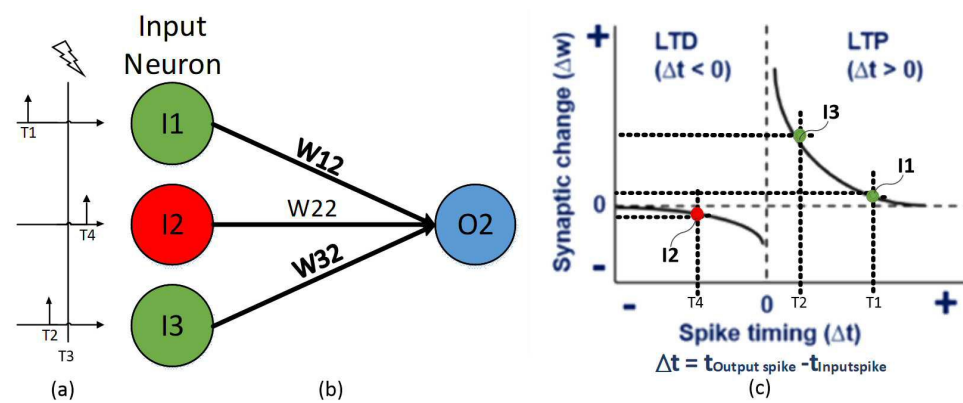
- [0099]
- |                   |                   |
|-------------------|-------------------|
| 100: 입력 뉴런부       | 200: 가중치 시냅스부     |
| 300: 출력 뉴런부       | 400: 가중치 학습부      |
| 410: 카운터          | 420: 입력 시프트 레지스터부 |
| 430: 출력 시프트 레지스터부 | 440: 가중치 업데이트부    |

도면

도면1

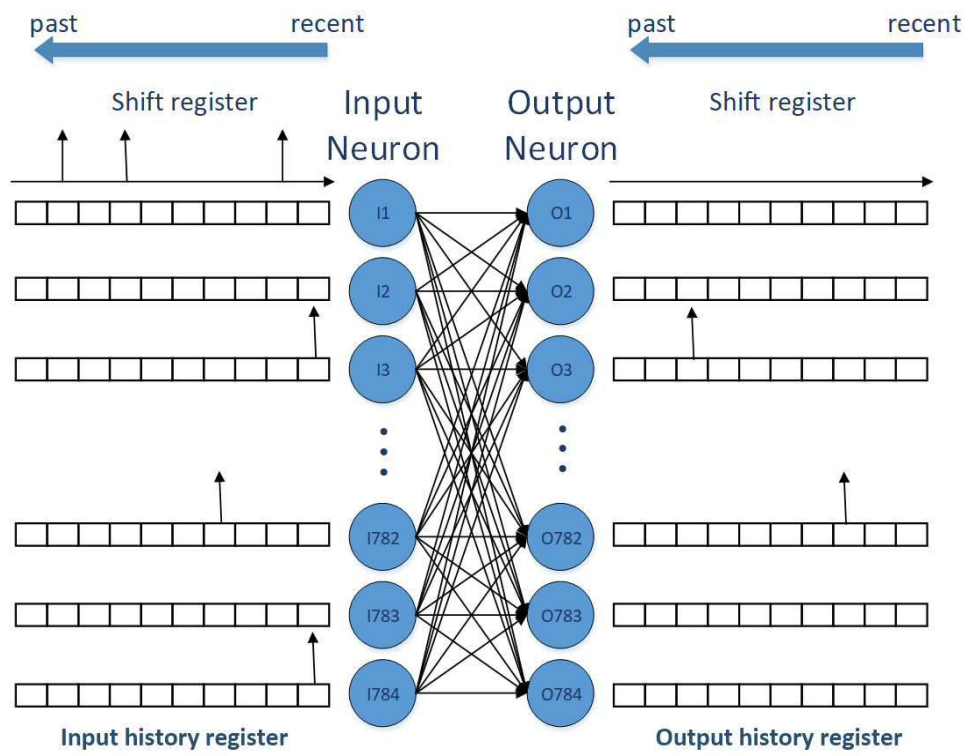


도면2

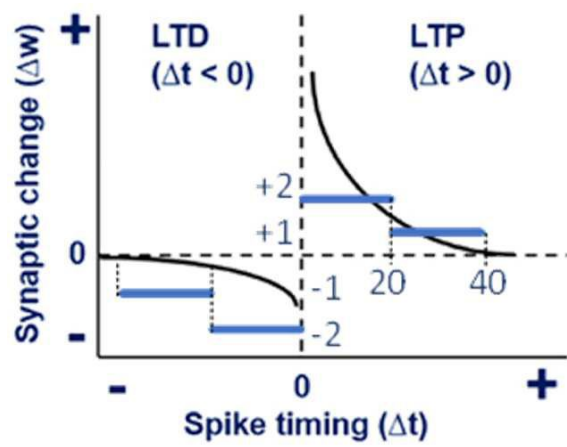




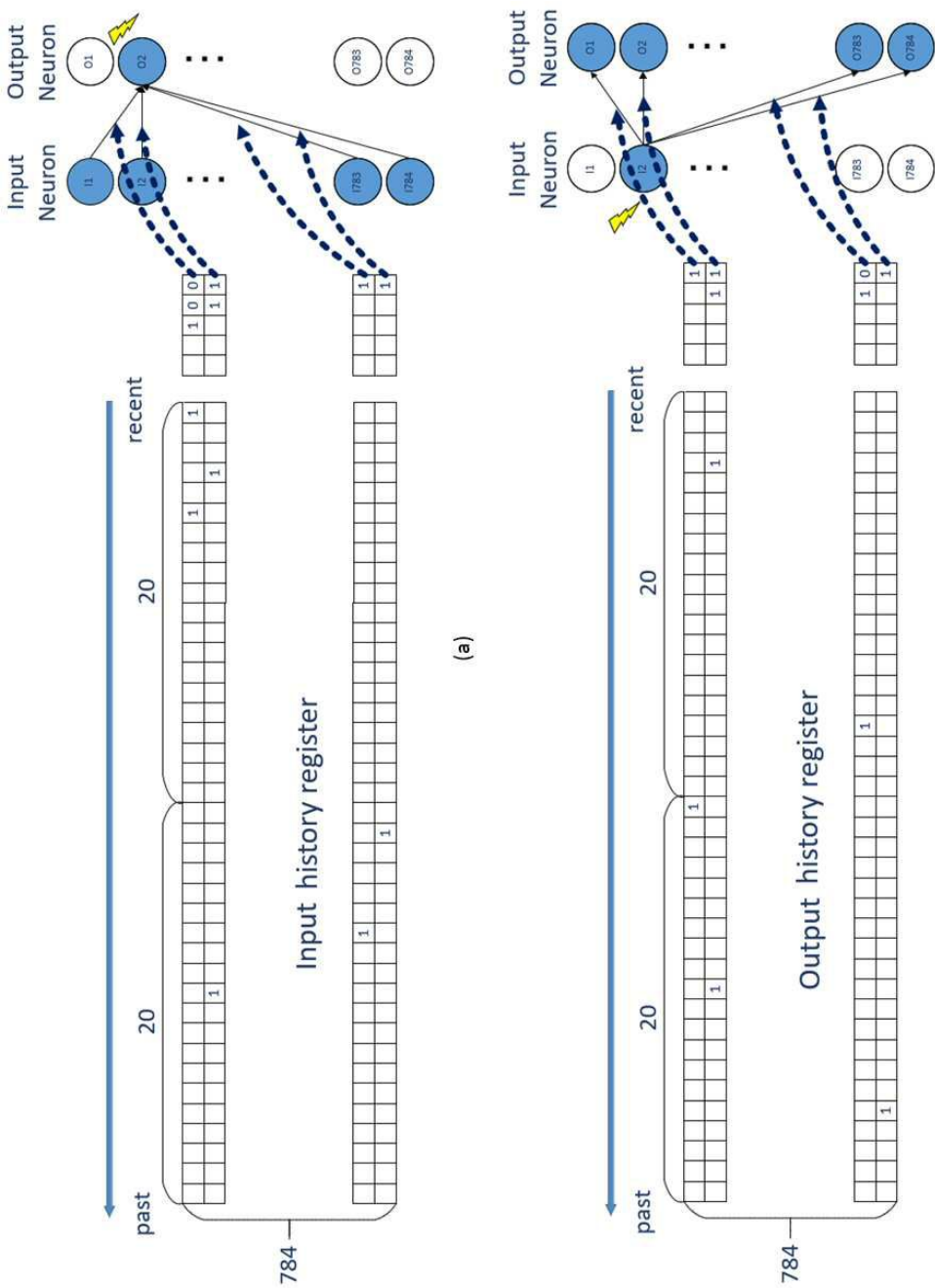
도면3



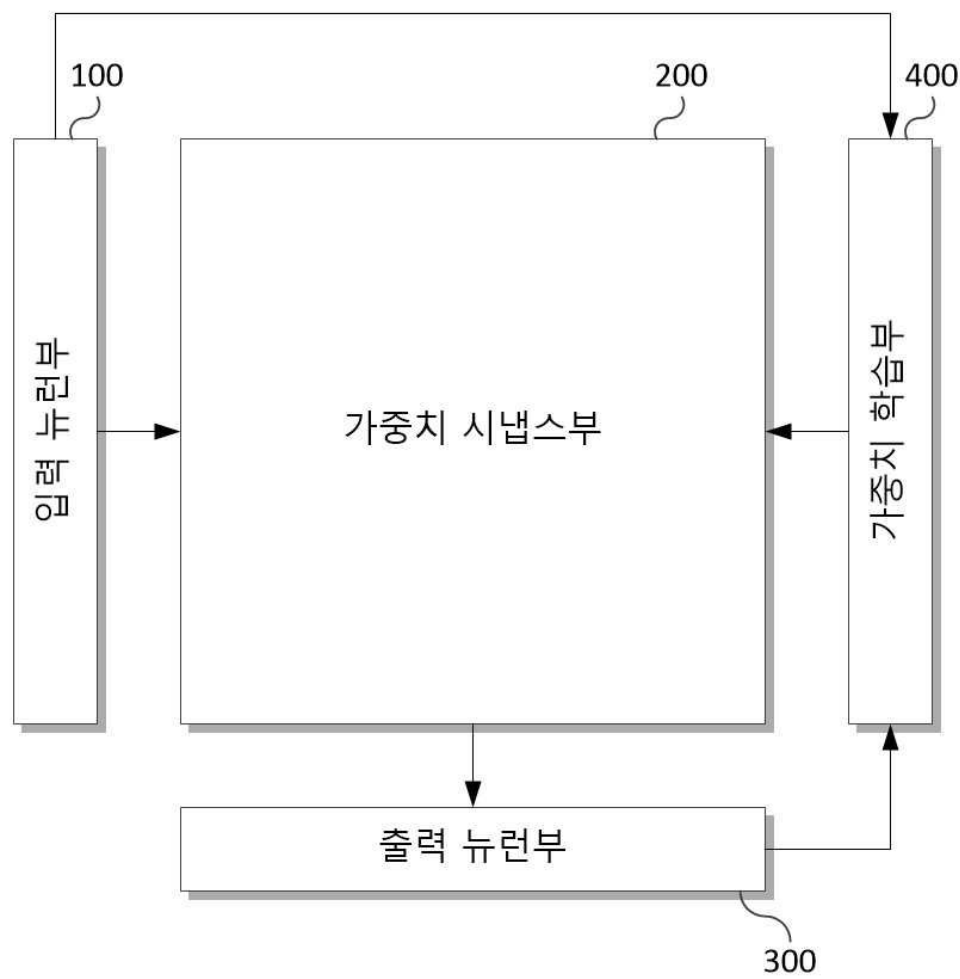
도면4



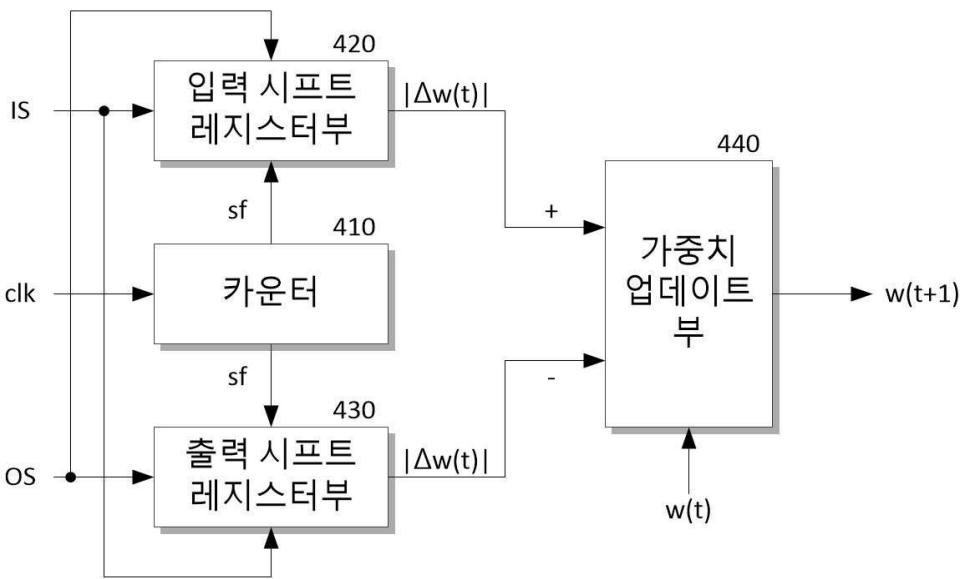
도면5



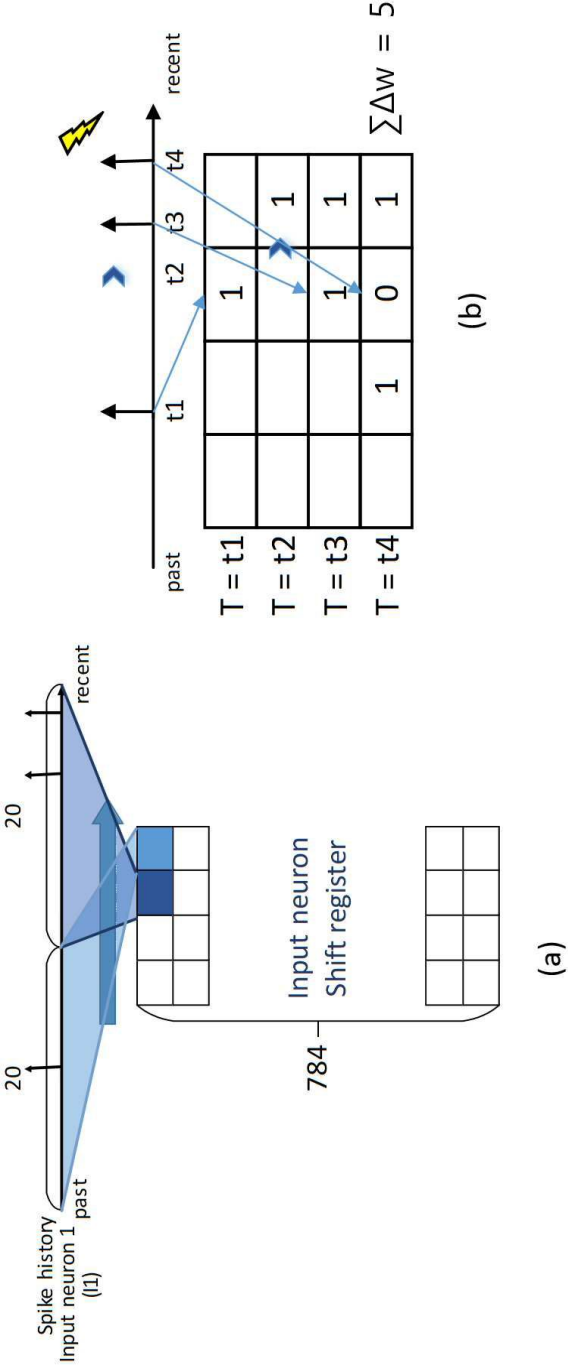
도면6



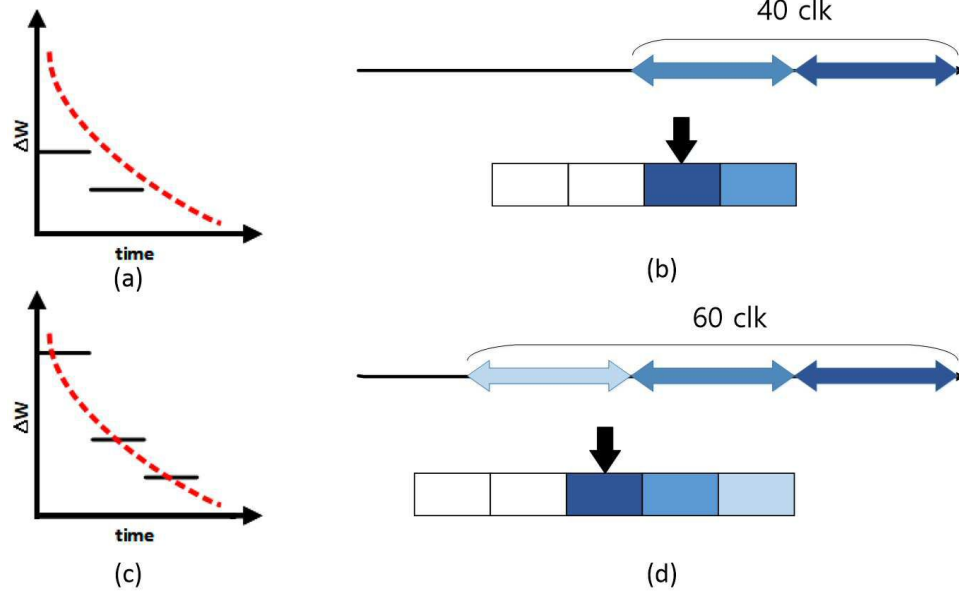
도면7



도면8



도면9



도면10

