



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2023-0021854  
(43) 공개일자 2023년02월14일

(51) 국제특허분류(Int. Cl.)  
G06N 10/00 (2022.01) B82Y 10/00 (2017.01)  
(52) CPC특허분류  
G06N 10/00 (2022.01)  
B82Y 10/00 (2013.01)  
(21) 출원번호 10-2021-0103746  
(22) 출원일자 2021년08월06일  
심사청구일자 2021년08월06일

(71) 출원인  
고려대학교 산학협력단  
서울특별시 성북구 안암로 145, 고려대학교 (안암동5가)  
(72) 발명자  
허준  
서울특별시 강남구 광평로10길 6, 206동 103호(일원동, 한솔마을아파트)  
강유진  
서울특별시 성북구 지봉로21길 22, 401호(보문동2가)  
(74) 대리인  
이대호, 박진홍

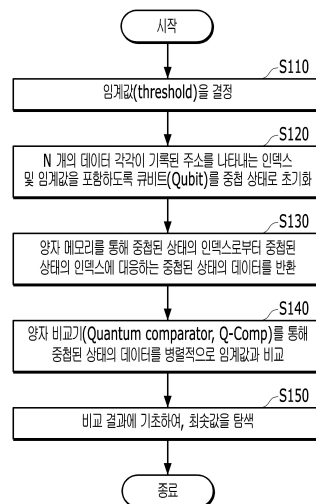
전체 청구항 수 : 총 13 항

(54) 발명의 명칭 **최솟값을 탐색하기 위한 방법**

(57) 요약

본 개시의 몇몇 실시예에 따른, 적어도 하나의 프로세서를 포함하는 컴퓨팅 장치에 의해 N(N은 자연수) 개의 데이터에서 최솟값을 탐색하기 위한 최솟값 탐색 방법이 개시된다. 상기 최솟값을 탐색하기 위한 방법은 임계값(threshold)을 결정하는 단계; 상기 N 개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 상기 임계값을 포함하도록 큐비트(Qubit)를 중첩 상태로 초기화하는 단계; 양자 메모리를 통해 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환하는 단계; 양자 비교기(Quantum comparator, Q-Comp)를 통해 상기 중첩된 상태의 데이터를 병렬적으로 상기 임계값과 비교하는 단계; 및 상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계를 포함할 수 있다.

대표도 - 도2



이 발명을 지원한 국가연구개발사업

|             |                               |
|-------------|-------------------------------|
| 과제고유번호      | 1711139206                    |
| 과제번호        | 2021-0-01810                  |
| 부처명         | 과학기술정보통신부                     |
| 과제관리(전문)기관명 | 정보통신기획평가원                     |
| 연구사업명       | 대학ICT연구센터육성지원사업               |
| 연구과제명       | 양자세계를 연결하는 초신뢰 양자인터넷 요소 기술 개발 |
| 기 여 율       | 50/100                        |
| 과제수행기관명     | 고려대학교산학협력단                    |
| 연구기간        | 2021.07.01 ~ 2028.12.31       |

이 발명을 지원한 국가연구개발사업

|             |                            |
|-------------|----------------------------|
| 과제고유번호      | 1711126139                 |
| 과제번호        | 2020-2018-0-01402          |
| 부처명         | 과학기술정보통신부                  |
| 과제관리(전문)기관명 | 정보통신기획평가원                  |
| 연구사업명       | 대학ICT연구센터육성지원사업            |
| 연구과제명       | 인공지능을 위한 양자컴퓨팅 기초 원천 기술 연구 |
| 기 여 율       | 50/100                     |
| 과제수행기관명     | 한국과학기술원                    |
| 연구기간        | 2021.01.01 ~ 2021.12.31    |

---

## 명세서

### 청구범위

#### 청구항 1

적어도 하나의 프로세서를 포함하는 컴퓨팅 장치에 의해  $N$ ( $N$ 은 자연수) 개의 데이터에서 최솟값을 탐색하기 위한 최솟값 탐색 방법으로서,

임계값(threshold)을 결정하는 단계;

상기  $N$  개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 상기 임계값을 포함하도록 큐비트(Qubit)를 중첩 상태로 초기화하는 단계;

양자 메모리를 통해 중첩된 상태의 인덱스로부터 상기 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환하는 단계;

양자 비교기(Quantum comparator, Q-Comp)를 통해 상기 중첩된 상태의 데이터를 병렬적으로 상기 임계값과 비교하는 단계; 및

상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계;

를 포함하는,

최솟값을 탐색하기 위한 방법.

#### 청구항 2

제 1 항에 있어서,

상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계는,

데이터베이스에서 상기 임계값 보다 작은 데이터와 관련된 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시키고, 그리고 그로버 연산부(Grover operator)를 통해 상기 표기된 상태의 확률을 증폭시키는 동작을 기 설정된 횟수만큼 반복 수행하는 단계;

상기 반복 수행결과 측정된 인덱스를 반환하는 단계;

상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 작은 경우, 상기 반환된 인덱스와 관련된 데이터를 이용하여 상기 임계값을 업데이트하는 단계; 및

상기 업데이트된 임계값을 이용하여, 상기 최솟값 탐색 방법을 반복 수행하는 단계;

를 포함하는,

최솟값을 탐색하기 위한 방법.

#### 청구항 3

제 1 항에 있어서,

상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계는,

데이터베이스에서 상기 임계값 보다 작은 데이터와 관련된 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시키고, 그리고 그로버 연산부(Grover operator)를 통해 상기 표기된 상태의 확률을 증폭시키는 동작을 기 설정된 횟수만큼 반복 수행하는 단계; 및

상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 큰 경우, 상기 임계값을 이용하여 상기 최솟값 탐색

방법을 반복 수행하는 단계;  
 를 포함하는,  
 최솟값을 탐색하기 위한 방법.

#### 청구항 4

제 2 항 또는 제 3 항에 있어서,  
 상기 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행하는 단계;  
 를 더 포함하는,  
 최솟값을 탐색하기 위한 방법.

#### 청구항 5

제 4 항에 있어서,  
 상기 기 설정된 시간( $\tau_{th}$ )은 수학적식  $\tau_{th} = 1.25\pi\sqrt{N} + 1.4(\log_2 N)^2$ 에 따라 결정되는,  
 최솟값을 탐색하기 위한 방법.

#### 청구항 6

제 4 항에 있어서,  
 상기 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행한 경우, 상기 측정된 인덱스를 반환하는 단계;  
 및  
 상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 작은 경우, 상기 반환된 인덱스와 관련된 데이터를 최  
 소값으로 결정하는 단계;  
 를 포함하는,  
 최솟값을 탐색하기 위한 방법.

#### 청구항 7

제 4 항에 있어서,  
 상기 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행한 경우, 상기 측정된 인덱스를 반환하는 단계;  
 및  
 상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 큰 경우, 상기 임계값과 관련된 데이터를 최솟값으로  
 결정하는 단계;  
 를 포함하는,  
 최솟값을 탐색하기 위한 방법.

#### 청구항 8

제 1 항에 있어서,

상기 임계값(threshold)을 결정하는 단계는,  
 상기 N 개의 데이터 중 임의의 하나를 상기 임계값으로 결정하는 단계;  
 를 포함하는,  
 최솟값을 탐색하기 위한 방법.

## 청구항 9

제 2 항 또는 제 3 항에 있어서,  
 상기 기 설정된 횟수는 수학적식  $O(\sqrt{N})$ 에 따라 결정되고,  
 상기 O는 점근 표기법(asymptotic notation)에 따라 시간 복잡도를 나타내는 함수인,  
 최솟값을 탐색하기 위한 방법.

## 청구항 10

제 2 항 또는 제 3 항에 있어서,  
 상기 비교 결과에 기초하여, 데이터베이스에서 상기 임계값 보다 작은 데이터의 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시키고, 그리고 그로버 연산부(Grover operator)를 통해 상기 표기된 상태의 확률을 증폭시키는 동작을 기 설정된 횟수만큼 반복 수행하는 단계는,  
 평균 진폭보다 작은 진폭을 갖는 상태의 확률을 증폭시키는 단계;  
 를 포함하는,  
 최솟값을 탐색하기 위한 방법.

## 청구항 11

제 1 항에 있어서,  
 양자 비교기(Quantum comparator, Q-Comp)를 통해 상기 중첩된 상태의 데이터를 병렬적으로 상기 임계값과 비교하는 단계는,  
 상기 임계값 및 상기 중첩된 상태의 데이터의 보수를 덧셈 연산하여 캐리(carry) 비트가 발생하는지 여부를 인식하는 단계; 및  
 상기 캐리 비트의 발생 여부에 기초하여, 상기 중첩된 상태의 데이터 및 상기 임계값의 크기를 비교하는 단계;  
 를 포함하는,  
 최솟값을 탐색하기 위한 방법.

## 청구항 12

제 11 항에 있어서,  
 상기 캐리 비트의 발생 여부에 기초하여, 상기 중첩된 상태의 데이터 및 상기 임계값의 크기를 비교하는 단계는,  
 상기 캐리 비트가 발생했다고 인식된 경우, 상기 임계값이 상기 중첩된 상태의 데이터보다 크다고 인식하는 단계;

를 포함하는,  
최솟값을 탐색하기 위한 방법.

### 청구항 13

제 11 항에 있어서,  
상기 캐리 비트의 발생 여부에 기초하여, 상기 중첩된 상태의 데이터 및 상기 임계값의 크기를 비교하는 단계는,  
상기 캐리 비트가 발생하지 않았다고 인식된 경우, 상기 중첩된 상태의 데이터가 상기 임계값보다 크다고 인식하는 단계;  
를 포함하는,  
최솟값을 탐색하기 위한 방법.

## 발명의 설명

### 기술 분야

- [0001] 본 개시는 양자 컴퓨터를 위한 탐색 알고리즘 구현에 관한 것으로, 구체적으로 양자 최솟값 찾기 알고리즘(Quantum minimum searching algorithm, QMSA)을 Quantum RAM(QRAM)과 양자 비교기(Quantum comparator, Q-comp)를 활용하여 구현하는 방법에 관한 것이다.
- [0002] 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 정보통신·방송 연구개발사업의 일환으로 수행하였다(2021-0-01810, 양자세계를 연결하는 초신뢰 양자인터넷 요소 기술 개발).
- [0003] 또한, 본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학ICT연구센터지원사업의 연구결과로 수행되었다(IITP-2020-2018-0-01402).

### 배경 기술

- [0005] 양자 컴퓨터(quantum computer)는 얽힘(entanglement)이나 중첩(superposition)과 같은 양자역학적인 현상을 활용하여 데이터를 처리하는 장치일 수 있다. 이러한 양자 컴퓨터는 고전 컴퓨터에 비해 우수한 성능을 발휘할 수 있을 것으로 예상되어, 활발한 개발이 이루어지고 있다. 그러나, 양자 컴퓨터는 구동 방식에 있어서 고전 컴퓨터와는 다소 상이한 부분이 존재하여, 실질적인 문제를 해결하기 위한 알고리즘 또한 새롭게 개발될 필요가 있다.
- [0006] 양자 컴퓨터에서 사용할 수 있는 대표적인 알고리즘으로는 1996년 뢰브 그로버(Lov Grover)에 의해 제안된 그로버 탐색 알고리즘(Grover Search Algorithm)이 존재한다. 여기서, 그로버 탐색 알고리즘은 정렬되지 않은 데이터베이스에서 특정 데이터를 찾는 알고리즘으로서, 고전 무작위 탐색 알고리즘에 비해 적은 복잡도를 지니는 알고리즘일 수 있다. 그러나, 이러한 그로버 탐색 알고리즘 및 기타 현재의 양자 알고리즘들의 경우, 데이터를 비교하는 부분에 있어서 고전 메모리, 즉 RAM(Random Access Memory)을 필요로 하는 부분이 존재할 수 있다.

## 선행기술문헌

### 특허문헌

- [0008] (특허문헌 0001) 일본 공개특허 2005-78407

## 발명의 내용

### 해결하려는 과제

- [0009] 본 개시는 전술한 배경기술에 대응하여 안출된 것으로, 고전 메모리를 필요로 하는 부분을 양자 메모리로 대체한 양자 컴퓨터를 이용하여, 최솟값을 탐색하기 위한 방법을 제공하고자 한다.
- [0010] 본 개시의 기술적 과제들은 이상에서 언급한 기술적 과제로 제한되지 않으며, 언급되지 않은 또 다른 기술적 과제들은 아래의 기재로부터 당업자에게 명확하게 이해될 수 있을 것이다.

### 과제의 해결 수단

- [0012] 전술한 바와 같은 과제를 해결하기 위한 본 개시의 일 실시예에 따라, 적어도 하나의 프로세서를 포함하는 컴퓨팅 장치에 의해  $N$ ( $N$ 은 자연수) 개의 데이터에서 최솟값을 탐색하기 위한 최솟값 탐색 방법이 개시된다. 상기 최솟값을 탐색하기 위한 방법은 임계값(threshold)을 결정하는 단계; 상기  $N$  개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 상기 임계값을 포함하도록 큐비트(Qubit)를 중첩 상태로 초기화하는 단계; 양자 메모리를 통해 중첩된 상태의 인덱스로부터 상기 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환하는 단계; 양자 비교기(Quantum comparator, Q-Comp)를 통해 상기 중첩된 상태의 데이터를 병렬적으로 상기 임계값과 비교하는 단계; 및 상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계; 를 포함할 수 있다.
- [0013] 또한, 상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계는, 데이터베이스에서 상기 임계값 보다 작은 데이터와 관련된 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시키고, 그리고 그로버 연산부(Grover operator)를 통해 상기 표기된 상태의 확률을 증폭시키는 동작을 기 설정된 횟수만큼 반복 수행하는 단계; 상기 반복 수행결과 측정된 인덱스를 반환하는 단계; 상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 작은 경우, 상기 반환된 인덱스와 관련된 데이터를 이용하여 상기 임계값을 업데이트하는 단계; 및 상기 업데이트된 임계값을 이용하여, 상기 최솟값 탐색 방법을 반복 수행하는 단계; 를 포함할 수 있다.
- [0014] 또한, 상기 비교 결과에 기초하여, 상기 최솟값을 탐색하는 단계는, 데이터베이스에서 상기 임계값 보다 작은 데이터와 관련된 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시키고, 그리고 그로버 연산부(Grover operator)를 통해 상기 표기된 상태의 확률을 증폭시키는 동작을 기 설정된 횟수만큼 반복 수행하는 단계; 및 상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 큰 경우, 상기 임계값을 이용하여 상기 최솟값 탐색 방법을 반복 수행하는 단계; 를 포함할 수 있다.
- [0015] 또한, 상기 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행하는 단계; 를 더 포함할 수 있다.
- [0016] 또한, 상기 기 설정된 시간( $\tau_{th}$ )은 수학적식  $\tau_{th} = 1.25\pi\sqrt{N} + 1.4(\log_2 N)^2$ 에 따라 결정될 수 있다.
- [0017] 또한, 상기 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행한 경우, 상기 측정된 인덱스를 반환하는 단계; 및 상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 작은 경우, 상기 반환된 인덱스와 관련된 데이터를 최솟값으로 결정하는 단계; 를 포함할 수 있다.
- [0018] 또한, 상기 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행한 경우, 상기 측정된 인덱스를 반환하는 단계; 및 상기 반환된 인덱스와 관련된 데이터가 상기 임계값 보다 큰 경우, 상기 임계값과 관련된 데이터를 최솟값으로 결정하는 단계; 를 포함할 수 있다.
- [0019] 또한, 상기 임계값(threshold)을 결정하는 단계는, 상기  $N$  개의 데이터 중 임의의 하나를 상기 임계값으로 결정하는 단계; 를 포 포함할 수 있다.
- [0020] 또한, 상기 기 설정된 횟수는 수학적식  $O(\sqrt{N})$ 에 따라 결정되고, 상기  $O$ 는 점근 표기법(asymptotic notation)에 따라 시간 복잡도를 나타내는 함수일 수 있다.
- [0021] 또한, 상기 비교 결과에 기초하여, 데이터베이스에서 상기 임계값 보다 작은 데이터의 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시키고, 그리고 그로버 연산부(Grover operator)를 통해 상기 표기된 상태의 확률을 증폭시키는 작을 기 설정된 횟수만큼 반복 수행하는 단계는, 평균 진폭보다 작은 진폭을 갖는 상태의 확률을

증폭시키는 단계; 를 포함할 수 있다.

[0022] 또한, 양자 비교기(Quantum comparator, Q-Comp)를 통해 상기 중첩된 상태의 데이터를 병렬적으로 상기 임계값과 비교하는 단계는, 상기 임계값 및 상기 중첩된 상태의 데이터의 보수를 덧셈 연산하여 캐리(carry) 비트가 발생하는지 여부를 인식하는 단계; 및 상기 캐리 비트의 발생 여부에 기초하여, 상기 중첩된 상태의 데이터 및 상기 임계값의 크기를 비교하는 단계; 를 포함할 수 있다.

[0023] 또한, 상기 캐리 비트의 발생 여부에 기초하여, 상기 중첩된 상태의 데이터 및 상기 임계값의 크기를 비교하는 단계는, 상기 캐리 비트가 발생했다고 인식된 경우, 상기 임계값이 상기 중첩된 상태의 데이터보다 크다고 인식하는 단계; 를 포함할 수 있다.

[0024] 또한, 상기 캐리 비트의 발생 여부에 기초하여, 상기 중첩된 상태의 데이터 및 상기 임계값의 크기를 비교하는 단계는, 상기 캐리 비트가 발생하지 않았다고 인식된 경우, 상기 중첩된 상태의 데이터가 상기 임계값보다 크다고 인식하는 단계; 를 포함할 수 있다.

[0026] 본 개시에서 얻을 수 있는 기술적 해결 수단은 이상에서 언급한 해결 수단들로 제한되지 않으며, 언급하지 않은 또 다른 해결 수단들은 아래의 기재로부터 본 개시가 속하는 기술분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

### 발명의 효과

[0028] 본 개시의 몇몇 실시예에 따르면, 현재 데이터를 비교하는 부분을 양자 메모리와 양자 비교기로 교체함에 따라 더 적은 복잡도로 최솟값을 탐색할 수 있도록 한다.

[0029] 본 개시에서 얻을 수 있는 효과는 이상에서 언급한 효과로 제한되지 않으며, 언급하지 않은 또 다른 효과들은 아래의 기재로부터 본 개시가 속하는 기술분야에서 통상의 지식을 가진 자에게 명확하게 이해될 수 있을 것이다.

### 도면의 간단한 설명

[0031] 다양한 양상들이 이제 도면들을 참조로 기재되며, 여기서 유사한 참조 번호들은 총괄적으로 유사한 구성요소들을 지칭하는데 이용된다. 이하의 실시예에서, 설명 목적을 위해, 다수의 특정 세부사항들이 하나 이상의 양상들의 총체적 이해를 제공하기 위해 제시된다. 그러나, 그러한 양상(들)이 이러한 특정 세부사항들 없이 실시될 수 있음은 명백할 것이다. 다른 예시들에서, 공지의 구조들 및 장치들이 하나 이상의 양상들의 기재를 용이하게 하기 위해 블록도 형태로 도시된다.

도 1은 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치의 일례를 설명하기 위한 블록 구성도이다.

도 2는 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치가 N개의 데이터에서 최솟값을 탐색하기 위한 최솟값 탐색 방법의 일례를 설명하기 위한 흐름도이다.

도 3은 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치가 비교 결과에 기초하여 최솟값을 탐색하는 방법의 일례를 설명하기 위한 흐름도이다.

도 4는 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치(100)가 반환된 인덱스에 기초하여, 최솟값을 결정하는 방법의 일례를 설명하기 위한 흐름도이다.

도 5는 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치가 중첩된 상태의 데이터를 병렬적으로 임계값과 비교하는 방법의 일례를 설명하기 위한 흐름도이다.

도 6은 고전 메모리를 이용하는 종래의 최솟값 탐색 알고리즘을 설명하기 위한 흐름도이다.

도 7은 본 개시의 몇몇 실시예에 따른 양자 메모리 이용하는 최솟값 탐색 방법의 일례를 설명하기 위한 흐름도이다.

도 8은 본 개시의 몇몇 실시예에 따른 최솟값을 탐색하기 위한 방법을 통해 최솟값을 탐색한 결과 데이터를 설

명하기 위한 도면이다.

### 발명을 실시하기 위한 구체적인 내용

- [0032] 다양한 실시예들 및/또는 양상들이 이제 도면들을 참조하여 개시된다. 하기 설명에서는 설명을 목적으로, 하나 이상의 양상들의 전반적 이해를 돕기 위해 다수의 구체적인 세부사항들이 개시된다. 그러나, 이러한 양상(들)은 이러한 구체적인 세부사항들 없이도 실행될 수 있다는 점 또한 본 개시의 기술 분야에서 통상의 지식을 가진 자에게 감지될 수 있을 것이다. 이후의 기재 및 첨부된 도면들은 하나 이상의 양상들의 특정한 예시적인 양상들을 상세하게 기술한다. 하지만, 이러한 양상들은 예시적인 것이고 다양한 양상들의 원리들에서의 다양한 방법들 중 일부가 이용될 수 있으며, 기술되는 설명들은 그러한 양상들 및 그들의 균등물들을 모두 포함하고자 하는 의도이다. 구체적으로, 본 명세서에서 사용되는 "실시예", "예", "양상", "예시" 등은 기술되는 임의의 양상 또는 설계가 다른 양상 또는 설계들보다 양호하다거나, 이점이 있는 것으로 해석되지 않을 수도 있다.
- [0033] 이하, 도면 부호에 관계없이 동일하거나 유사한 구성 요소는 동일한 참조 번호를 부여하고 이에 대한 중복되는 설명은 생략한다. 또한, 본 명세서에 개시된 실시예를 설명함에 있어서 관련된 공지 기술에 대한 구체적인 설명이 본 명세서에 개시된 실시예의 요지를 흐릴 수 있다고 판단되는 경우 그 상세한 설명을 생략한다. 또한, 첨부된 도면은 본 명세서에 개시된 실시예를 쉽게 이해할 수 있도록 하기 위한 것일 뿐, 첨부된 도면에 의해 본 명세서에 개시된 기술적 사상이 제한되지 않는다.
- [0034] 비록 제 1, 제 2 등이 다양한 소자나 구성요소들을 서술하기 위해서 사용되나, 이들 소자나 구성요소들은 이들 용어에 의해 제한되지 않음은 물론이다. 이들 용어들은 단지 하나의 소자나 구성요소를 다른 소자나 구성요소와 구별하기 위하여 사용하는 것이다. 따라서, 이하에서 언급되는 제 1 소자나 구성요소는 본 발명의 기술적 사상 내에서 제 2 소자나 구성요소 일 수도 있음은 물론이다.
- [0035] 다른 정의가 없다면, 본 명세서에서 사용되는 모든 용어(기술 및 과학적 용어를 포함)는 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에게 공통적으로 이해될 수 있는 의미로 사용될 수 있을 것이다. 또 일반적으로 사용되는 사전에 정의되어 있는 용어들은 명백하게 특별히 정의되어 있지 않는 한 이상적으로 또는 과도하게 해석되지 않는다.
- [0036] 더불어, 용어 "또는"은 배타적 "또는"이 아니라 내포적 "또는"을 의미하는 것으로 의도된다. 즉, 달리 특정되지 않거나 문맥상 명확하지 않은 경우에, "X는 A 또는 B를 이용한다"는 자연적인 내포적 치환 중 하나를 의미하는 것으로 의도된다. 즉, X가 A를 이용하거나; X가 B를 이용하거나; 또는 X가 A 및 B 모두를 이용하는 경우, "X는 A 또는 B를 이용한다"가 이들 경우들 어느 것으로도 적용될 수 있다. 또한, 본 명세서에 사용된 "및/또는"이라는 용어는 열거된 관련 아이템들 중 하나 이상의 아이템의 가능한 모든 조합을 지칭하고 포함하는 것으로 이해되어야 한다.
- [0037] 또한, "포함한다" 및/또는 "포함하는"이라는 용어는, 해당 특징 및/또는 구성요소가 존재함을 의미하지만, 하나 이상의 다른 특징, 구성요소 및/또는 이들의 그룹의 존재 또는 추가를 배제하지 않는 것으로 이해되어야 한다. 또한, 달리 특정되지 않거나 단수 형태를 지시하는 것으로 문맥상 명확하지 않은 경우에, 본 명세서와 청구범위에서 단수는 일반적으로 "하나 또는 그 이상"을 의미하는 것으로 해석되어야 한다.
- [0038] 더불어, 본 명세서에서 사용되는 용어 "정보" 및 "데이터"는 종종 서로 상호교환 가능하도록 사용될 수 있다.
- [0039] 어떤 구성 요소가 다른 구성 요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성 요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성 요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성 요소가 다른 구성 요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성 요소가 존재하지 않는 것으로 이해되어야 할 것이다.
- [0040] 이하의 설명에서 사용되는 구성 요소에 대한 접미사 "모듈" 및 "부"는 명세서 작성의 용이함만이 고려되어 부여되거나 혼용되는 것으로서 그 자체로 서로 구별되는 의미 또는 역할을 갖는 것은 아니다.
- [0041] 본 개시의 목적 및 효과, 그리고 그것들을 달성하기 위한 기술적 구성들은 첨부되는 도면과 함께 상세하게 후술되어 있는 실시예들을 참조하면 명확해질 것이다. 본 개시를 설명하는데 있어서 공지 기능 또는 구성에 대한 구체적인 설명이 본 개시의 요지를 불필요하게 흐릴 수 있다고 판단되는 경우에는 그 상세한 설명을 생략할 것이다. 그리고 후술되는 용어들은 본 개시에서의 기능을 고려하여 정의된 용어들로써 이는 사용자, 운용자의 의도 또는 관례 등에 따라 달라질 수 있다.

- [0042] 그러나 본 개시는 이하에서 개시되는 실시예들에 한정되는 것이 아니라 서로 다른 다양한 형태로 구현될 수 있다. 단지 본 실시예들은 본 개시가 완전하도록 하고, 본 개시가 속하는 기술분야에서 통상의 지식을 가진 자에게 개시의 범주를 완전하게 알려주기 위해 제공되는 것이며, 본 개시는 청구항의 범주에 의해 정의될 뿐이다. 그러므로 그 정의는 본 명세서 전반에 걸친 내용을 토대로 내려져야 할 것이다.
- [0044] 본 개시에서 최솟값을 탐색하기 위한 방법은 임계값과 현재 데이터를 비교하는 부분이 양자 메모리(또는, Quantum Random Access Memory, QRAM) 및 양자 비교기(Quantum comparator, Q-Comp)를 통해 구현됨에 따라 더 적은 복잡도로 최솟값이 탐색될 수 있다. 여기서, 양자 메모리는 고전 데이터와 양자상태의 데이터 간 인터페이스로서의 역할을 수행할 수 있다. 구체적으로, 양자 메모리는 중첩된 상태의 주소를 입력 받으면 대응되는 중첩된 상태의 데이터의 반환이 가능할 수 있다. 중첩 상태란 여러 가지 상태가 동시에 하나의 데이터를 나타내는 것을 의미할 수 있다. 이 경우, 양자 비교기는 양자 메모리를 통해 불러온 중첩된 상태의 데이터 모두에 대한 비교를 병렬적(parallel)으로 수행할 수 있다. 이하, 본 개시에 따른 최솟값을 탐색하기 위한 방법에 대해 도 1 내지 도 8을 통해 설명한다.
- [0046] 도 1은 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치의 일례를 설명하기 위한 블록 구성도이다.
- [0047] 도 1을 참조하면, 컴퓨팅 장치(100)는 프로세서(110), 양자 메모리(120), 양자 비교기(130) 및 그로버 연산부(140)를 포함할 수 있다. 다만, 상술한 구성 요소들은 컴퓨팅 장치(100)를 구현하는데 있어서 필수적인 것은 아니어서, 양자 컴퓨팅 장치(100)는 위에서 열거된 구성요소들 보다 많거나, 또는 적은 구성요소들을 가질 수 있다.
- [0048] 컴퓨팅 장치(100)는 최솟값을 탐색하기 위해 요구되는 양자 연산의 수행이 가능한 양자 컴퓨팅 장치일 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0049] 본 개시에서, 컴퓨팅 장치(100)는 고전 메모리 대신 양자 메모리(120)를 포함하고, 또한 고전 비교기 대신 양자 비교기(130)를 포함할 수 있다.
- [0050] 구체적으로, 그로버(Grover)가 양자 컴퓨터에서 사용할 수 있는 탐색 알고리즘을 제안한 이후, Durr와 Høyer에 의해 최솟값을 찾기 위한 양자 알고리즘이 제안되었다. 그러나, Durr와 Høyer에 의해 제안된 최솟값을 찾기 위한 알고리즘은 데이터와 임계값을 비교하는 부분에 있어서 고전 비교기를 필요로 하였다. 반면, 본 개시에 따른 컴퓨팅 장치(100)는 양자 메모리(120) 및 양자 비교기(130)를 포함함으로써, 데이터와 임계값을 비교하는 부분을 양자 메모리(120) 및 양자 비교기(130)를 통해 수행할 수 있다. 즉, 고전 컴퓨터의 연산이 필요했던 부분을 양자 회로를 통한 연산으로 대체함으로써, 최솟값 탐색 방법의 전 과정이 하나의 컴퓨팅 장치(100)로 모듈화될 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0051] 한편, 프로세서(110)는 통상적으로 컴퓨팅 장치(100)의 전반적인 동작을 제어할 수 있다. 프로세서(110)는 위에서 살펴본 구성요소들을 통해 입력 또는 출력되는 신호, 데이터, 정보 등을 처리하거나 데이터베이스(미도시)에 저장된 응용 프로그램을 구동함으로써, 사용자에게 적절한 정보 또는 기능을 제공 또는 처리할 수 있다.
- [0052] 본 개시에서, 프로세서(110)는  $N$ ( $N$ 은 자연수) 개의 데이터 각각이 기록된 주소를 나타내는 인덱스(index) 및 임계값(threshold)을 포함하도록 큐비트(Qubit)를 중첩 상태로 초기화할 수 있다. 여기서,  $N$ 개의 데이터는 최솟값을 탐색하기 위한 대상이 되는 데이터일 수 있다. 큐비트는 양자 컴퓨터의 기본 단위로서, 고전 컴퓨터의 비트(bit)에 대응하는 단위일 수 있다. 고전 컴퓨터는 정보를 0과 1의 비트 단위로 처리하고 저장하는 반면 양자 컴퓨터는 정보를 0과 1의 상태를 동시에 갖는 큐비트 단위로 처리하고 저장할 수 있다. 그리고, 프로세서(110)는 큐비트를 중첩 상태로 초기화한 이후, 양자 메모리(120) 및 양자 비교기(130)를 이용하여, 최솟값을 탐색할 수 있다. 이하, 본 개시에 따른 프로세서(110)가 최솟값을 탐색하는 구체적인 방법은 도 2 내지 도 7을 통해 설명한다.
- [0053] 한편, 양자 메모리(120)는 고전 메모리에 대응하는 모듈일 수 있다. 일례로, 양자 메모리(120)는 양자 랜덤 액세스 메모리(Quantum Random Access Memory, QRAM)일 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0054] 본 개시에서, 양자 메모리(120)는 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환할 수 있다. 여기서, 중첩된 상태의 인덱스는 중첩된 상태의 데이터 각각의 주소를 나타내는 것일

수 있다. 실시예에 따라, 양자 메모리(120)는  $O(\log_2 N)$ 의 복잡도를 가질 수 있다. 여기서,  $O$ 는 점근 표기법(asymptotic notation)에 따라 시간 복잡도를 나타내는 함수일 수 있다. 또한,  $N$ 은 데이터의 개수일 수 있다. 다만, 이에 한정되는 것은 아니다.

[0055] 한편, 양자 비교기(130)는 중첩된 상태의 데이터를 병렬적으로 임계값과 비교할 수 있다.

[0056] 구체적으로, 종래의 Durr와 Høyer의 최솟값을 찾기 위한 양자 알고리즘을 예로 들면, 임계값과 데이터를 비교하기 위해서는 우선 고전 메모리를 통해 해당 인덱스에 대응되는 데이터를 반환해야 했다. 그리고, 고전 비교기를 통해 반환된 데이터 및 임계값을 비교한 후 결과 값을 다시 양자 컴퓨터에 전송해야 했다. 그러나, 본 개시에 따른 컴퓨팅 장치(100)는 양자 메모리(120)를 활용하기 때문에, 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환할 수 있다. 이에 따라, 양자 비교기(130)는 중첩된 상태의 데이터를 병렬적으로 임계값과 비교할 수 있다. 다만, 이에 한정되는 것은 아니다.

[0057] 한편, 그로버 연산부(Grover operator, 140)는 데이터베이스에서 임계값 보다 작은 데이터와 관련된 인덱스가 표기된 상태의 위상(phase)이 반전된 경우, 표기된 상태의 확률을 증폭시킬 수 있다. 실시예에 따라, 데이터베이스는 오라클<sup>®</sup>일 수 있다.

[0058] 구체적으로, 오라클<sup>®</sup>은 일련의 큐비트에 대해 작용하여 해를 위상으로 반환하는 양자 서브루틴일 수 있다. 이러한 서브루틴은 몇몇 매개 변수에 더해 오라클<sup>®</sup>을 받아서 일련의 양자 연산을 적용하고, 양자 서브루틴에 대한 호출을 기본 게이트처럼 취급하는 양자 알고리즘에 대한 입력일 수 있다. 그리고, 그로버 연산부(140)는 표기된 또는 반환된 상태의 확률을 증폭시킬 수 있다. 실시예에 따라, 그로버 연산부(140)는 진폭 증폭 오라클<sup>®</sup>일 수 있다. 이 경우, 그로버 연산부(140)는 평균 진폭보다 작은 진폭을 갖는 상태의 확률을 증폭시킬 수 있다. 다만, 이에 한정되는 것은 아니다. 이하, 본 개시에 따른 그로버 연산부(140) 및 데이터베이스는 오라클<sup>®</sup>에 대응될 수 있는 바, 구체적인 내용에 대한 설명은 본 출원에서 전체가 참조로 통합되는 본 출원의 발명자의 논문 “Quantum Minimum Searching Algorithm and Circuit Implementation, DOI: 10.1109”에서 구체적으로 논의된다.

[0059] 상술한 구성에 따르면, 컴퓨팅 장치(100)는 종래의 최솟값을 찾기 위한 알고리즘에 있어서, 데이터와 임계값을 비교하는 부분을 양자 메모리(120) 및 양자 비교기(130)로 대체할 수 있다. 따라서, 컴퓨팅 장치(100)는 더 적은 복잡도로 최솟값을 탐색할 수 있다. 더하여, 중간에 측정 과정을 거쳐 고전 컴퓨터의 연산이 필요했던 부분을, 양자 회로를 통한 연산으로 대체하면서 최솟값을 탐색하기 위한 전 과정을 하나의 양자 컴퓨터로 모듈화할 수 있다.

[0060] 이하에서는 본 개시에 따른 컴퓨팅 장치(100)가 최솟값을 탐색하는 방법에 대해 설명한다.

[0062] 도 2는 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치가 N개의 데이터에서 최솟값을 탐색하기 위한 최솟값 탐색 방법의 일례를 설명하기 위한 흐름도이다.

[0063] 도 2를 참조하면, 컴퓨팅 장치(100)의 프로세서(110)는 임계값(threshold)을 결정할 수 있다(S110).

[0064] 일례로, 프로세서(110)는 N 개의 데이터 중 임의의 하나를 상기 임계값으로 결정할 수 있다. 다른 일례로, 프로세서(110)는 N 개의 데이터에 포함되지 않는 임의의 값을 임계값으로 결정할 수도 있다. 또 다른 일례로, 프로세서(110)는 사용자로부터의 입력에 기초하여 임계값을 결정할 수도 있다. 또 다른 일례로, 프로세서(110)는 사전에 수행되었던 최솟값 탐색의 결과를 참조하여, 임계값을 결정할 수도 있다. 다만, 이에 한정되는 것은 아니다.

[0065] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 N 개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 임계값을 포함하도록 큐비트(또는, 메모리)를 중첩 상태로 초기화할 수 있다(S120).

[0066] 예를 들어, 프로세서(110)는 하기의 수학식에 의해 큐비트를 초기화할 수 있다.

### 수학식 1

$$\psi = \sum_i \frac{1}{\sqrt{N}} |T[i]\rangle |T[y]\rangle$$

[0067]

[0068] 여기서,  $\psi$ 는 큐비트일 수 있다. 그리고,  $N$ 은 데이터의 개수일 수 있다. 그리고,  $i$ 는 인덱스일 수 있다. 그리고,  $T[i]$ 는 인덱스에 대응하는 데이터일 수 있다. 그리고,  $y$ 는 임계값일 수 있다. 그리고,  $T[y]$ 는 임계값에 대응하는 데이터일 수 있다. 다만, 이에 한정되는 것은 아니다.

[0069] 한편, 컴퓨팅 장치(100)의 양자 메모리(120)는 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환할 수 있다(S130).

[0070] 예를 들어, 양자 메모리(120)는 하기의 수학식 2와 같은 중첩된 상태의 인덱스를 입력 받은 경우, 하기의 수학식 3과 같은 중첩된 상태의 데이터를 반환할 수 있다. 다만, 이에 한정되는 것은 아니다.

### 수학식 2

$$\sum_{i=0}^{N-1} |i\rangle$$

[0071]

### 수학식 3

$$\sum_{i=0}^{N-1} |T[i]\rangle$$

[0072]

[0073] 한편, 컴퓨팅 장치(100)의 양자 비교기(130)는 중첩된 상태의 데이터를 병렬적으로 임계값과 비교할 수 있다(S140).

[0074] 본 개시의 실시예에 따라 양자 메모리(120)는 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환할 수 있기 때문에, 양자 비교기(130)는 중첩된 상태의 데이터를 병렬적으로 임계값과 비교할 수 있다. 다만, 이에 한정되는 것은 아니다.

[0075] 한편, 본 개시의 몇몇 실시예에 따르면, 양자 비교기(130)는 캐리(carry) 비트의 발생 여부에 기초하여, 중첩된 상태의 데이터 및 임계값을 비교할 수 있다. 여기서, 캐리는 최상위 비트(Most Significant Bit, MSB)에서 자리 올림이 발생하는 것을 의미할 수 있다. 즉, 본 개시에서의 캐리 비트가 발생했다는 의미는 최상위 비트에서 자리 올림이 발생했다는 의미일 수 있다. 다만, 이에 한정되는 것은 아니다. 이하, 본 개시에 따른 양자 비교기(130)가 캐리 비트의 발생 여부에 기초하여, 중첩된 상태의 데이터 및 임계값을 비교하는 방법은 도 5를 통해 설명한다.

[0076] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 양자 비교기(130)가 중첩된 상태의 데이터를 병렬적으로 임계값과 비교한 비교 결과에 기초하여, 최솟값을 탐색할 수 있다(S150).

[0077] 일례로, 프로세서(110)는 비교 결과 반환된 인덱스와 관련된 데이터가 임계값 보다 작은 경우, 반환된 인덱스와 관련된 데이터를 최솟값으로 결정할 수 있다. 다만, 이에 한정되는 것은 아니다.

[0078] 한편, 본 개시의 몇몇 실시예에 따르면, 프로세서(110)는 비교 결과 반환된 인덱스와 관련된 데이터가 임계값 보다 큰 경우, 상술한 최솟값 탐색 방법을 반복 수행할 수 있다. 이하, 본 개시에 따른 프로세서(110)가 최솟값 탐색 방법을 반복 수행하는 방법은 도 3을 통해 설명한다.

[0079] 한편, 본 개시의 몇몇 실시예에 따르면, 컴퓨팅 장치(100)의 프로세서(110)는 상술한 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행할 수 있다. 여기서, 기 설정된 시간은 프로세서(110)가 N 개의 데이터 중에서 도출한 결과가 최솟값인 확률이 충분히 높은 시간일 수 있다.

[0080] 한편, 본 개시의 몇몇 실시예에 따르면, 기 설정된 시간( $\tau_{th}$ )은 하기의 수학적 식 4에 따라 결정될 수 있다.

#### 수학적 식 4

$$[0081] \quad \tau_{th} = 1.25\pi\sqrt{N} + 1.4(\log_2 N)^2$$

[0082] 여기서,  $N$ 은 데이터의 개수일 수 있다.

[0083] 만약, 프로세서(110)가 기 설정된 시간( $\tau_{th}$ )이 아닌 임의의 시간 동안 최솟값 탐색 방법을 반복 수행한 경우, 그 시간이 충분하지 못하다면,  $N$  개의 데이터 중 최솟값을 탐색하지 못한 확률이 높을 수 있다. 즉, 프로세서(110)가 임의의 시간 동안 최솟값 탐색 방법을 반복 수행하여 반환한  $N$ 개의 데이터 중 하나의 데이터가  $N$  개의 데이터에서 최솟값이 아닐 수 있다. 반대로, 프로세서(110)가  $N$  개의 데이터 중 최솟값을 탐색했을 확률이 충분히 높은 시간이 경과한 이후에도 최솟값 탐색 방법을 반복 수행하는 경우, 리소스(resource)에 낭비가 발생할 수 있다. 따라서, 프로세서(110)는 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행할 수 있다. 다만, 이에 한정되는 것은 아니다.

[0084] 상술한 구성에 따르면, 컴퓨팅 장치(100)는 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환할 수 있다. 이에 따라, 컴퓨팅 장치(100)는 중첩된 상태의 데이터를 병렬적으로 임계값과 비교할 수 있다. 이 경우, 컴퓨팅 장치(100)가 최솟값을 탐색하기 위해 소요하는 기 설정된 시간( $\tau_{th}$ )은 도 종래의 시간에 비해 줄어들 수 있다. 일례로, 고전 메모리를 이용하는 종래의 기술의 경우, 최솟값을 탐색하기 위해 소요되는 시간은  $\tau_{th} = 22.5\sqrt{N} + 1.4(\log_2 N)^2$  일 수 있다. 즉, 본 발명에 따른 컴퓨팅 장치(100)에 의해 최솟값을 탐색하는데 소요되는 시간에 비해 긴 시간일 수 있다.

[0085] 한편, 본 개시의 몇몇 실시예에 따르면, 컴퓨팅 장치(100)는 중첩된 상태의 데이터가 병렬적으로 임계값과 비교된 비교 결과에 기초하여, 최솟값을 탐색할 수 있다. 이하, 도 3을 통해 본 개시에 따른 컴퓨팅 장치(100)가 비교 결과에 기초하여 최솟값을 탐색하는 방법에 대해 설명한다.

[0087] 도 3은 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치가 비교 결과에 기초하여 최솟값을 탐색하는 방법의 일례를 설명하기 위한 흐름도이다.

[0088] 도 3을 참조하면, 컴퓨팅 장치(100)는 데이터베이스에서 임계값 보다 작은 데이터와 관련된 인덱스를 표기하고, 표기된 상태의 위상(phase)을 반전시킬 수 있다. 그리고, 컴퓨팅 장치(100)는 그로버 연산부(140)를 통해 표기된 상태의 확률을 증폭시키는 동작을 기 설정된 횟수만큼 반복 수행할 수 있다(S151). 여기서, 기 설정된 횟수는 표기된 상태의 확률을 최대로 키우기 위해 결정된 횟수 일 수 있다.

[0089] 일례로, 기 설정된 횟수는 하기의 수학적 식에 따라 결정될 수 있다.

#### 수학적 식 5

$$[0090] \quad O(\sqrt{N})$$

[0091] 여기서,  $O$ 는 점근 표기법(asymptotic notation)에 따라 시간 복잡도를 나타내는 함수일 수 있다. 다만, 이에 한정되는 것은 아니다.

[0092] 종래의 고전 무작위 알고리즘의 경우, 확률을 증폭시키는 횟수는  $O(N)$  번일 수 있다. 이는  $N$ 개의 데이터에 대응하는 횟수로서, 즉  $N$ 개의 데이터가 존재하는 경우에 있어서 최대 반복 횟수일 수 있다. 반면, 본 발명에 따르면, 프로세서(110)는 최대 반복 횟수 보다 작은 횟수인  $O(\sqrt{N})$ 의 횟수만큼 단계(S151)를 반복 수행할 수 있다. 다만, 이에 한정되는 것은 아니다.

- [0093] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 반복 수행결과 측정된 인덱스를 반환할 수 있다(S152). 여기서, 프로세서(110)가 인덱스를 반환한다는 의미는 큐비트의 상태를 관측한다는 의미일 수 있다. 환언하자면, 인덱스 또는 데이터 중 적어도 하나가 결정되는 것을 의미할 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0094] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 반환된 인덱스와 관련된 데이터가 임계값 보다 작은 경우(S153, Yes), 반환된 인덱스와 관련된 데이터를 이용하여 임계값을 업데이트할 수 있다(S154). 그리고, 프로세서(110)는 업데이트된 임계값을 이용하여, 최솟값 탐색 방법을 반복 수행할 수 있다(S155).
- [0095] 실시예에 따라, 프로세서(110)는 N 개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 업데이트된 임계값을 포함하도록 큐비트를 중첩 상태로 초기화할 수 있다. 그리고, 프로세서(110)는 단계 S130, S140, S151, S152 및 S153을 반복 수행할 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0096] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 반환된 인덱스와 관련된 데이터가 임계값 보다 큰 경우(S153, No), 임계값을 이용하여 최솟값 탐색 방법을 반복 수행할 수 있다(S156).
- [0097] 실시예에 따라, 프로세서(110)는 N 개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 업데이트되지 않은 임계값을 포함하도록 큐비트를 중첩 상태로 초기화할 수 있다. 그리고, 프로세서(110)는 단계 S130, S140, S151, S152 및 S153을 반복 수행할 수 있다. 또는, 프로세서(110)는 단계 S120의 수행 없이, 단계 S130, S140, S151, S152 및 S153을 반복 수행할 수 있다.
- [0098] 한편, 상술한 단계 S155 및 S156에 따르면, 컴퓨팅 장치(100)의 프로세서(110)는 최솟값 탐색 방법을 무한히 수행하게 될 수도 있다. 따라서, 프로세서(110)는 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안만 반복 수행할 수 있다.
- [0099] 구체적으로, 프로세서(110)는 최솟값 탐색 방법을 수행하는 동안 기 설정된 시간( $\tau_{th}$ )이 경과했는지 여부를 인식할 수 있다.
- [0100] 일례로, 프로세서(110)는 단계 S120 내지 S150 중 어느 하나의 단계가 수행된 경우, 기 설정된 시간( $\tau_{th}$ )이 경과했는지 여부를 인식할 수 있다. 다른 일례로, 프로세서(110)는 단계 S151 내지 S156 중 어느 하나의 단계가 수행된 경우, 기 설정된 시간( $\tau_{th}$ )이 경과했는지 여부를 인식할 수 있다. 그리고, 프로세서(110)는 기 설정된 시간( $\tau_{th}$ )이 경과되지 않았다고 인식한 경우, 최솟값 탐색 방법을 반복 수행할 수 있다. 또는, 프로세서(110)는 설정된 시간( $\tau_{th}$ )이 경과됐다고 인식한 경우, 측정된 인덱스를 반환할 수 있다. 그리고, 프로세서(110)는 반환된 인덱스에 기초하여, 최솟값을 결정할 수 있다. 이하, 도 4를 통해 본 개시에 따른 컴퓨팅 장치(100)가 반환된 인덱스에 기초하여, 최솟값을 결정하는 방법에 대해 설명한다.
- [0102] 도 4는 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치(100)가 반환된 인덱스에 기초하여, 최솟값을 결정하는 방법의 일례를 설명하기 위한 흐름도이다.
- [0103] 도 4를 참조하면, 컴퓨팅 장치(100)의 프로세서(110)는 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행한 경우, 측정된 인덱스를 반환할 수 있다(S210). 여기서, 프로세서(110)가 인덱스를 반환한다는 의미는 큐비트의 상태를 관측한다는 의미일 수 있다. 환언하자면, 인덱스 또는 데이터 중 적어도 하나가 결정되는 것을 의미할 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0104] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 반환된 인덱스와 관련된 데이터가 임계값 보다 작은 경우(S220, Yes), 반환된 인덱스와 관련된 데이터를 최솟값으로 결정할 수 있다(S230).
- [0105] 환언하자면, 프로세서(110)는 상태가 관측되어 반환된 인덱스와 관련된 데이터 또는 상태가 관측되어 반환된 데이터를 최솟값으로 결정할 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0106] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 반환된 인덱스와 관련된 데이터가 임계값 보다 큰 경우(S220, No), 임계값과 관련된 데이터를 최솟값으로 결정할 수 있다(S240).
- [0107] 환언하자면, 프로세서(110)는 상태가 관측되어 반환된 임계값과 관련된 데이터를 최소값으로 결정할 수 있다. 다만, 이에 한정되는 것은 아니다.

- [0108] 상술한 구성에 따르면, 컴퓨팅 장치(100)의 프로세서(110)는 최솟값 탐색 방법을 기 설정된 시간( $\tau_{th}$ ) 동안 반복 수행한 경우, 측정된 인덱스를 반환할 수 있다. 그리고, 프로세서(110)는 반환된 인덱스에 기초하여, 최솟값을 결정할 수 있고, 상기 최솟값은 N 개의 데이터 중에서 가장 작은 값일 수 있다.
- [0109] 한편, 본 개시의 몇몇 실시예에 따르면, 컴퓨팅 장치(100)는 캐리 비트의 발생 여부에 기초하여, 중첩된 상태의 데이터를 병렬적으로 임계값과 비교할 수 있다. 이하, 도 5를 통해 본 개시에 따른 컴퓨팅 장치(100)가 중첩된 상태의 데이터를 병렬적으로 임계값과 비교하는 방법에 대해 설명한다.
- [0111] 도 5는 본 개시의 몇몇 실시예에 따른 컴퓨팅 장치가 중첩된 상태의 데이터를 병렬적으로 임계값과 비교하는 방법의 일례를 설명하기 위한 흐름도이다.
- [0112] 도 5를 참조하면, 컴퓨팅 장치(100)는 양자 비교기(130)를 통해 임계값 및 중첩된 상태의 데이터의 보수를 덧셈 연산하여 캐리 비트가 발생하는지 여부를 인식할 수 있다(S141). 다만, 양자 비교기(130)가 수행하는 연산은 덧셈 연산에 한정되는 것은 아니고, 뺄셈 연산일 수도 있다.
- [0113] 구체적으로, 양자 비교기(130)는 양자 메모리(120)를 통해 반환된 중첩된 상태의 데이터의 보수를 구할 수 있다. 실시예에 따라, 양자 비교기(130)는 1의 보수법을 이용하여, 중첩된 상태의 데이터의 보수를 구할 수 있다. 여기서, 1의 보수법은 각 자릿수의 값이 모두 1인 수에서, 주어진 2진수를 빼거나 또는 비트를 반전시켜 보수를 구하는 방법일 수 있다. 예를 들어, 숫자 "2" 를 이진법으로 나타내는 경우, "010" 으로 표현될 수 있고, 이 경우, 보수는 "101" 일 수 있다. 프로세서(110)는 중첩된 상태의 데이터의 보수를 구한 경우, 보수 및 임계값의 덧셈 연산을 수행하여, 캐리 비트가 발생하는지 여부를 인식할 수 있다. 여기서, 캐리는 최상위 비트에서 자리 올림이 발생하는 것을 의미할 수 있다. 즉, 본 개시에서의 캐리 비트가 발생했다는 의미는 최상위 비트에서 자리 올림이 발생했다는 의미일 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0114] 한편, 본 개시의 몇몇 실시예에 따르면, 컴퓨팅 장치(100)는 양자 비교기(130)를 통해 임계값의 보수 및 중첩된 상태의 데이터를 덧셈 연산하여 캐리 비트가 발생하는지 여부를 인식할 수도 있다.
- [0115] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 캐리 비트의 발생 여부에 기초하여, 중첩된 상태의 데이터 및 임계값의 크기를 비교할 수 있다.
- [0116] 일례로, 프로세서(110)는 캐리 비트가 발생했다고 인식된 경우(S142, Yes), 임계값이 중첩된 상태의 데이터보다 크다고 인식할 수 있다(S143).
- [0117] 예를 들어, 임계값이 "4" 이고, 중첩된 상태의 데이터가 "2" 라고 가정하자. 이 경우, "4 + 2의 보수" 를 이진법으로 표현한다면 " $(100)_2 + ((010)_2)'$ " 로 표현될 수 있다. 또한, "010" 의 보수는 "101" 이기 때문에, " $(100)_2 + ((010)_2)'$ " "는 " $(100)_2 + ((101)_2)$ " 로 표현될 수 있다. 이에 따라, " $(100)_2 + ((101)_2)$ " 의 연산을 수행하는 경우, " $(1001)_2$ " 의 값이 도출될 수 있다. 여기서, 최상위 비트에 발생한 자리 올림 1은 캐리일 수 있다. 이 경우, 프로세서(110)는 캐리 비트가 발생했다고 인식했기 때문에, 임계값인 "4" 가 중첩된 상태의 데이터인 "2" 보다 크다고 인식할 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0118] 한편, 프로세서(110)는 캐리 비트가 발생하지 않았다고 인식된 경우(S142, No), 중첩된 상태의 데이터가 임계값보다 크다고 인식할 수 있다(S144).
- [0119] 예를 들어, 임계값이 "4" 이고, 중첩된 상태의 데이터가 "6" 이라고 가정하자. 이 경우, "4 + 6의 보수" 를 이진법으로 표현한다면 " $(100)_2 + ((110)_2)'$ " 로 표현될 수 있다. 또한, "110" 의 보수는 "001" 이기 때문에, " $(100)_2 + ((110)_2)'$ " "는 " $(100)_2 + ((001)_2)$ " 로 표현될 수 있다. 이에 따라, " $(100)_2 + ((001)_2)$ " 의 연산을 수행하는 경우, " $(0101)_2$ " 의 값이 도출될 수 있다. 이와 같은 경우, 프로세서(110)는 캐리 비트가 발생하지 않았다고 인식할 수 있다. 따라서, 프로세서(110)는 중첩된 상태의 데이터인 "6" 이 임계값인 "4" 보다 크다고 인식할 수 있다. 다만, 이에 한정되는 것은 아니다.
- [0120] 상술한 구성에 따르면, 컴퓨팅 장치(100)는 양자 비교기(130)를 통해 임계값 및 중첩된 상태의 데이터의 보수를 덧셈 연산하여 캐리 비트가 발생하는지 여부에 기초하여, 중첩된 상태의 데이터 및 임계값의 크기를 비교할 수 있다. 종래의 "Thomas Haner, Martin Roetteler, Krysta M. Svore "Factoring using  $2n+2$  qubits with toffoli based modular multiplication", arXiv:1611.07995v2, Jun. 2017" 논문에서 제시된 양자 상수 덧셈기

는  $n$ -비트의 상수  $c$ 와 변수  $b$ 의 덧셈 연산  $r = c + b$ 를 수행하고,  $r$ 의 최상위 비트(캐리 비트)  $y$ 만 반환할 수 있었다. 조금 구체적으로, 종래의 양자 상수 덧셈기는  $c$ 의 각 자리수가 1인 경우에만 CNOT 게이트와 NOT 게이트를 추가하여 회로를 구성하며,  $n-1$ 개의 보조 큐비트  $q$ 를 사용할 수 있었다. 반면, 본 발명은 1의 보수법에서 두 수의 뺄셈이 음수인 경우는 캐리 비트가 발생하지 않는 것에 착안하여 변수  $a$ 와 변수  $b$ 의 덧셈기를 비교기로 변환할 수 있었다. 다만, 이에 한정되는 것은 아니다.

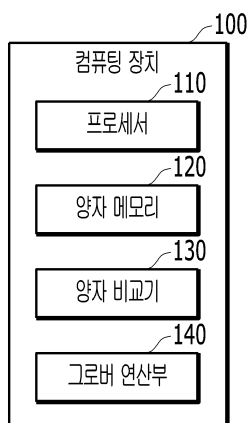
- [0122] 도 6은 고전 메모리를 이용하는 종래의 최솟값 탐색 알고리즘을 설명하기 위한 흐름도이다. 도 7은 본 개시의 몇몇 실시예에 다른 양자 메모리 이용하는 최솟값 탐색 방법의 일례를 설명하기 위한 흐름도이다.
- [0123] Durr와 Høyer에 의해 제안된 종래의 최솟값을 찾기 위한 양자 알고리즘은 임계값과 데이터를 비교하기 위해서 고전 메모리 및 고전 비교기를 필요로 하였다. 일례로, 도 5를 참조하면, 종래의 최솟값을 찾기 위한 알고리즘은 단계 S300에 있어서 고전 메모리 또는 고전 비교기를 필요로 할 수 있었다.
- [0124] 구체적으로, 종래의 최솟값을 찾기 위한 알고리즘은 임계값 및 데이터의 비교를 하는 동작을 수행하기 위해 고전 메모리를 통해 먼저 해당 인덱스에 대응하는 데이터를 반환할 수 있었다. 그리고, 반환된 데이터를 이용하여 임계값의 비교를 수행한 결과를 양자 컴퓨터에 전송하는 형태일 수 있었다.
- [0125] 반면, 본 개시에서는 고전 메모리를 필요한 부분을 양자 메모리(120)를 통해 수행할 수 있었다. 구체적으로, 도 7을 참조하면 컴퓨팅 장치(100)는 양자 메모리(120)를 인터페이스로 사용하여 단계 S400을 수행할 수 있다.
- [0126] 실시예에 따라, 컴퓨팅 장치(100)의 프로세서(110)는  $N$  개의 데이터 각각이 기록된 주소를 나타내는 인덱스 및 임계값을 포함하도록 메모리(또는, 큐비트)를 중첩 상태로 초기화할 수 있다(S410). 그리고, 컴퓨팅 장치(100)는 양자 메모리(120)를 인터페이스로 사용하여 중첩된 상태의 인덱스로부터 중첩된 상태의 인덱스에 대응하는 중첩된 상태의 데이터를 반환할 수 있다. 이 경우, 컴퓨팅 장치(100)의 프로세서(110)는 데이터가 임계값보다 큰지 여부를 결정할 수 있다(S420).
- [0127] 실시예에 따라, 컴퓨팅 장치(100)는 양자 비교기(130)를 통해 중첩된 상태의 데이터를 병렬적으로 임계값과 비교하여 임계값보다 작은 중첩된 상태의 데이터를 알아낼 수 있고, 그리고 데이터베이스(또는, 오라클<sup>®</sup>)에서 임계값 보다 작은 데이터와 관련된 인덱스 또는 데이터의 위상(phase)을 반전시킬 수 있다(S430).
- [0128] 한편, 컴퓨팅 장치(100)의 그로버 연산부(140)는 평균 진폭보다 작은 진폭을 갖는 상태의 확률을 증폭시킬 수 있다(S440). 실시예에 따라, 그로버 연산부(140)는 기 설정된 횟수인  $O(\sqrt{N})$  회만큼 상태의 확률을 증폭시키기 위한 동작을 반복 수행할 수 있다.
- [0129] 한편, 컴퓨팅 장치(100)의 프로세서(110)는 그로버 연산부(140)를 통해 상태의 확률을 증폭시키기 위한 동작을 기 설정된 횟수만큼 반복 수행한 경우(S450, No), 반복 수행결과 측정된 인덱스 또는 데이터의 상태를 측정할 수 있다(S460). 그리고, 컴퓨팅 장치(100)의 프로세서(110)는 반복 수행결과 측정된 인덱스 또는 데이터를 반환할 수 있다(S470).
- [0130] 상술한 구성에 따르면, 본 발명의 컴퓨팅 장치(100)는 기존의 고전 비교기와 RAM이 필요한 방식을 양자 비교기(130) 및 양자 메모리(120)로 대체하여, 복잡도를 감소시키고 또한 모듈화를 가능하게 하였다. 더하여, 본 발명에 따른 최솟값을 탐색하기 위한 방법은 기 설정된 시간( $\tau_{th} = 22.5\sqrt{N} + 1.4(\log_2 N)^2$ ) 만에 수행될 수 있다. 이는, 양자 메모리(120)를 인터페이스로 이용함으로써, 중첩된 상태의 인덱스로부터 중첩된 상태의 데이터를 불러올 수 있기 때문이다. 반면, 종래의 최솟값을 찾기 위한 알고리즘은 본 발명에 따른 기 설정된 시간 보다 긴 시간( $\tau_{th} = 22.5\sqrt{N} + 1.4(\log_2 N)^2$ ) 동안 수행되어야 할 수 있다. 종래의 최솟값을 찾기 위한 알고리즘에서는 양자 메모리(120) 및 양자 비교기(130)를 사용하기가 어렵기 때문에, 고전 비교기와 RAM을 사용함에 따라 발생되는 차이일 수 있다.
- [0132] 도 8은 본 개시의 몇몇 실시예에 따른 최솟값을 탐색하기 위한 방법을 통해 최솟값을 탐색한 결과 데이터를 설명하기 위한 도면이다. 도 8에 도시된 히스토그램(220)의 x축은 데이터의 값을 나타낼 수 있고, y축은 각각의 데이터가 최솟값으로 탐색된 횟수일 수 있다. 그리고, 도 8에 도시된 히스토그램(220)은 5-큐비트 시뮬레이션

결과에 대한 히스토그램일 수 있다.

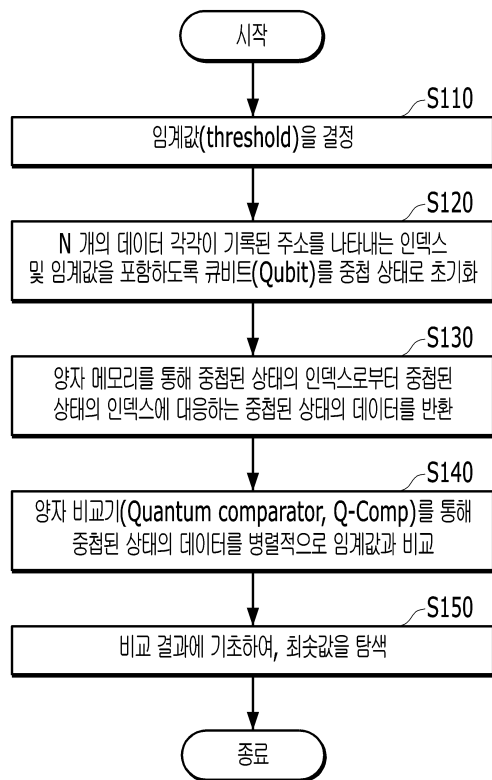
- [0133] 표(210)를 참조하면, 본 개시에 따른 최솟값 탐색 방법을 시도한 총 횟수(total number of trials)는 500번일 수 있다. 본 개시에서는, 임계값을 임의의 크기로 바뀌가며 최솟값 탐색 방법을 500번 수행할 수 있었다.
- [0134] 예상 반복 횟수(또는 예상 반복 시간)(Expected number of iterations)는 57일 수 있다. 구체적으로, 본 실험에 사용된 데이터의 개수는 32개 일 수 있다(0 내지 31). 이 경우, 실시예에 따른 기 설정된 시간( $\tau_{th} = 1.25\pi\sqrt{32} + 1.4(\log_2 32)^2$ )에 따라, 예상 반복 횟수(Expected number of iterations)는 57일 수 있다.
- [0135] 실험 결과, 실패 비율(Failure rate(%))는 0.8%일 수 있었다.
- [0136] 구체적으로, 히스토그램(220)을 참조하면, 0 내지 31의 크기를 갖는 데이터 중(본 실험은 인덱스 값이 데이터에 대응된다( $i = T[i]$ )고 가정하였다) 0의 크기를 갖는 데이터가 최솟값으로 결정된 횟수(211)는 496번 일 수 있다. 그리고, 1의 크기를 갖는 데이터가 최솟값으로 결정된 횟수(212)는 1번 일 수 있다. 그리고, 2의 크기를 갖는 데이터가 최솟값으로 결정된 횟수(213)는 3번 일 수 있다. 종합하여 보면, 최솟값을 탐색하지 못하고 실패한 비율은 0.8%일 수 있다.
- [0138] 제시된 실시예들에 대한 설명은 임의의 본 개시의 기술 분야에서 통상의 지식을 가진 자가 본 개시를 이용하거나 또는 실시할 수 있도록 제공된다. 이러한 실시예들에 대한 다양한 변형들은 본 개시의 기술 분야에서 통상의 지식을 가진 자에게 명백할 것이며, 여기에 정의된 일반적인 원리들은 본 개시의 범위를 벗어남이 없이 다른 실시예들에 적용될 수 있다. 그리하여, 본 개시는 여기에 제시된 실시예들로 한정되는 것이 아니라, 여기에 제시된 원리들 및 신규한 특징들과 일관되는 최광의의 범위에서 해석되어야 할 것이다.

## 도면

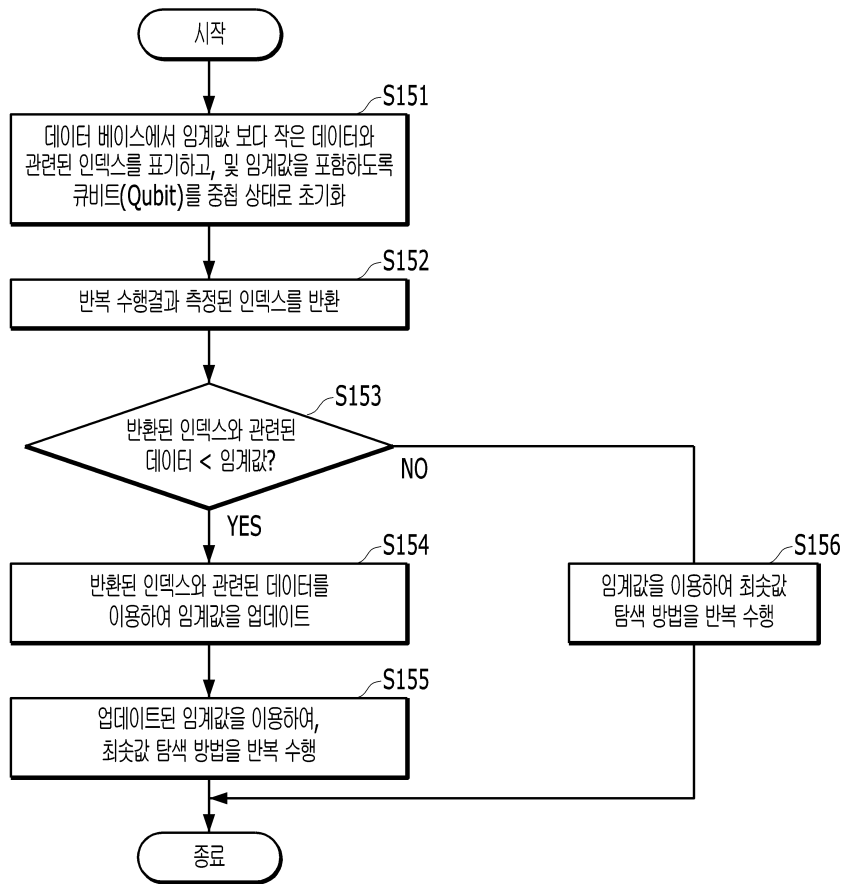
### 도면1



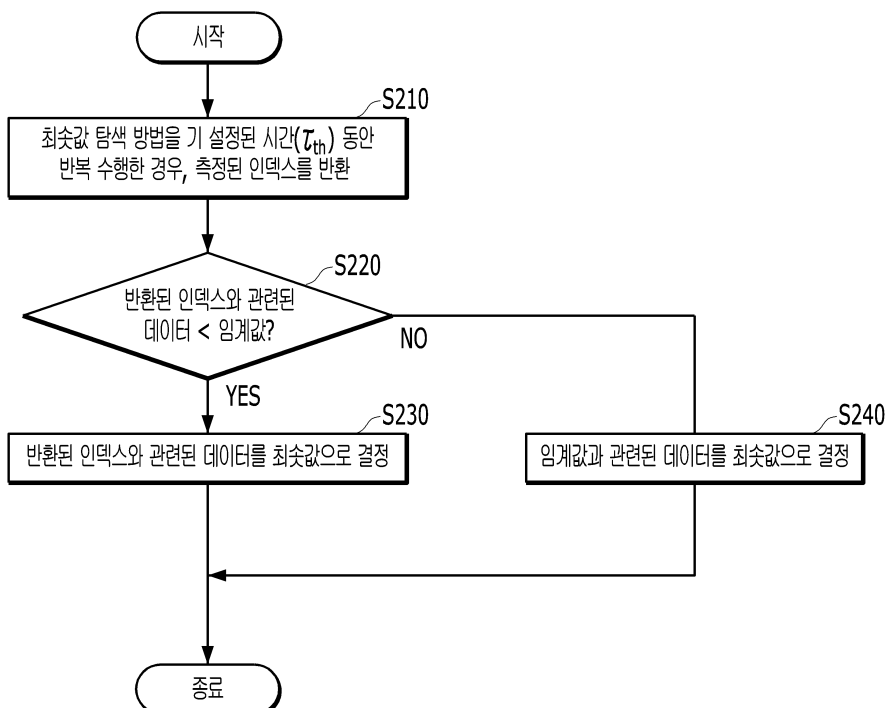
도면2



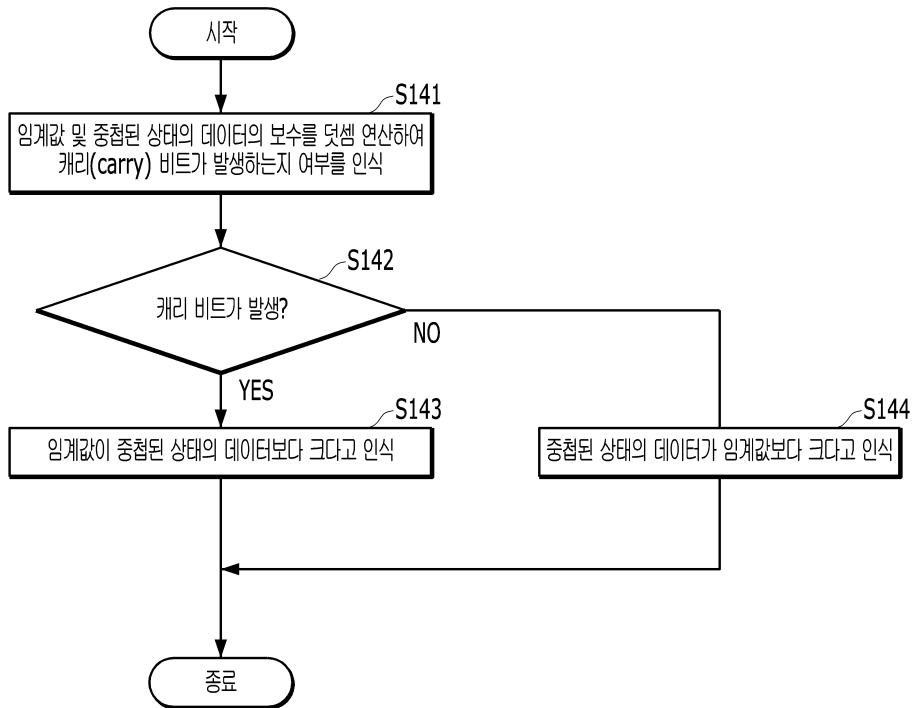
도면3



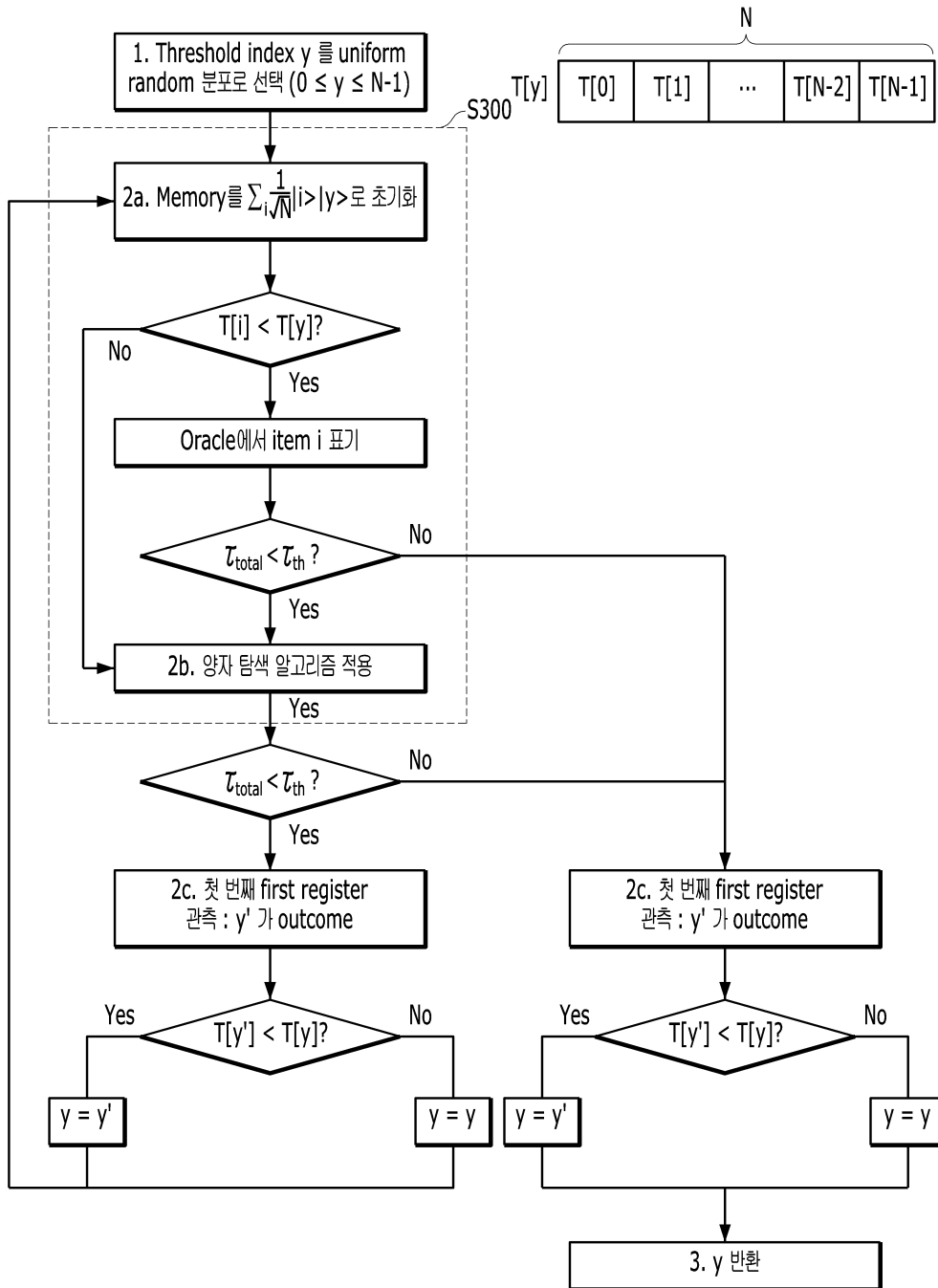
도면4



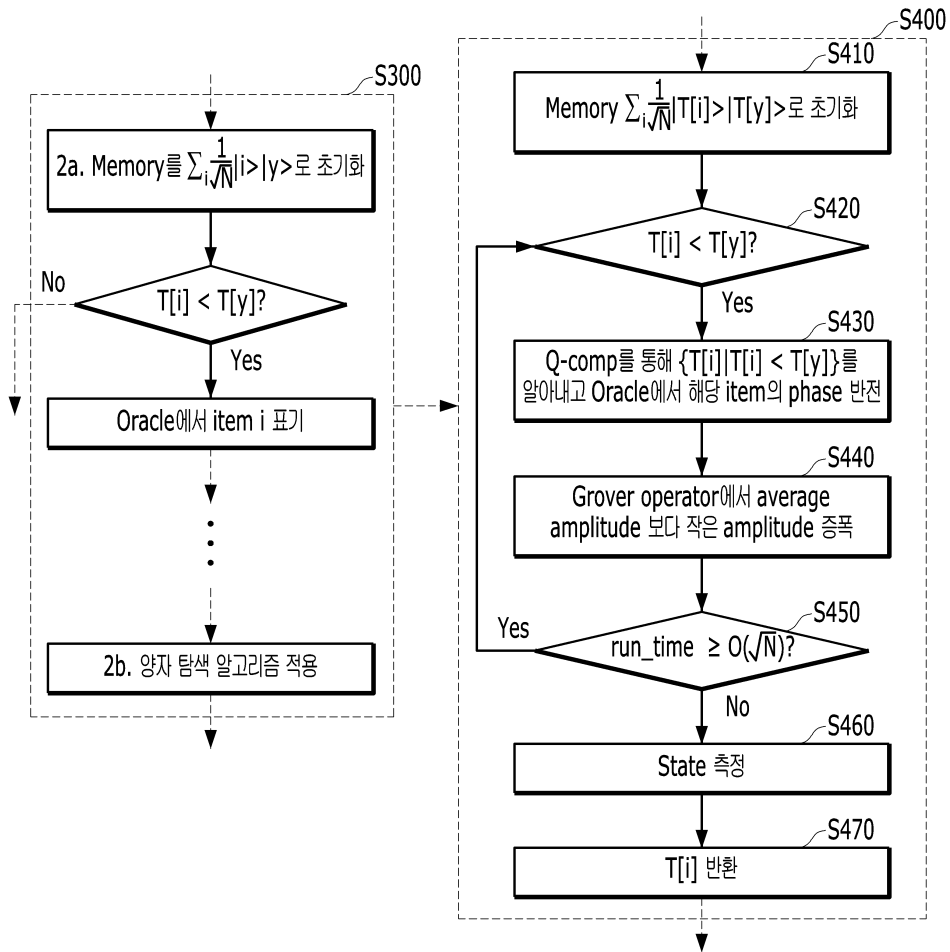
도면5



도면6



도면7



도면8

