



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2017년08월14일  
(11) 등록번호 10-1768356  
(24) 등록일자 2017년08월08일

(51) 국제특허분류(Int. Cl.)  
G06F 3/0488 (2013.01) G06F 3/044 (2006.01)  
(52) CPC특허분류  
G06F 3/0488 (2013.01)  
G06F 3/044 (2013.01)  
(21) 출원번호 10-2015-7034063  
(22) 출원일자(국제) 2013년07월01일  
심사청구일자 2015년11월30일  
(85) 번역문제출일자 2015년11월30일  
(65) 공개번호 10-2016-0003199  
(43) 공개일자 2016년01월08일  
(86) 국제출원번호 PCT/CN2013/078571  
(87) 국제공개번호 WO 2015/000101  
국제공개일자 2015년01월08일  
(56) 선행기술조사문헌  
US20120120002 A1\*  
US07664720 B1\*  
US20130257761 A1\*  
\*는 심사관에 의하여 인용된 문헌

(73) 특허권자  
인텔 코포레이션  
미합중국 캘리포니아 95054 산타클라라 미션 칼리지 블러바드 2200  
(72) 발명자  
유안 카이닝  
중국 쑤엔 200241 31 상하이 지상 로드 넘버 880  
(74) 대리인  
제일특허법인

전체 청구항 수 : 총 8 항

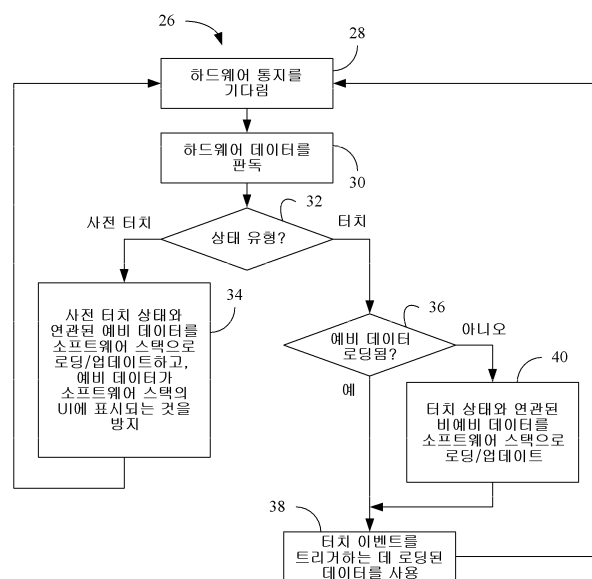
심사관 : 장진환

(54) 발명의 명칭 신속 응답 용량성 터치 스크린 디바이스

(57) 요약

시스템 및 방법은 터치 스크린에 관하여 사전 터치 상태를 식별하고, 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하는 것을 제공할 수 있다. 추가적으로, 터치 스크린에 관하여 터치 상태가 식별되면 터치 이벤트를 트리거하는 데 예비 데이터가 사용될 수 있다. 일례로서, 예비 데이터는 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것이 방지된다.

대표도 - 도4



(52) CPC특허분류  
G06F 2203/04101 (2013.01)

---

## 명세서

### 청구범위

#### 청구항 1

터치 스크린과,

적어도 부분적으로 고정 기능 하드웨어(fixed-functionality hardware)로 구현되는 로직을 포함하되,

상기 로직은,

상기 터치 스크린에 관하여, 좌표 데이터 및 캐패시턴스 값에 관한 하드웨어 통지를 포함하는 사전 터치 상태(pre-touch condition)를 식별하고 - 상기 로직은 상기 사전 터치 상태를 식별하기 위해 상기 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했는지 결정하고, 상기 사전 터치 임계치는 잡음 임계치보다 크고 상기 터치 상태에 대응하는 터치 임계치보다 작음 - ,

상기 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하며 - 상기 로딩은 상기 좌표 데이터 및 상기 캐패시턴스 값을 하나 이상의 레지스터 및 메모리 위치 중 적어도 하나에 기록하는 것을 포함하고, 상기 예비 데이터는 상기 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩되어서 새로운 소프트웨어 데이터 구조를 생성함 - ,

상기 예비 데이터가 상기 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하기 위해 상기 소프트웨어 스택에 상기 로딩된 예비 데이터가 예비적인 것임을 통지하고,

상기 터치 스크린에 관하여 터치 상태가 식별되면 상기 예비 데이터를 사용하여 터치 이벤트를 트리거하는

시스템.

#### 청구항 2

삭제

#### 청구항 3

삭제

#### 청구항 4

제 1 항에 있어서,

상기 로직은 상기 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 상기 예비 데이터를 업데이트하는

시스템.

#### 청구항 5

삭제

#### 청구항 6

삭제

#### 청구항 7

적어도 부분적으로 고정 기능 하드웨어로 구현되는 로직을 포함하는 장치로서,

상기 로직은,

터치 스크린에 관하여, 좌표 데이터 및 캐패시턴스 값에 관한 하드웨어 통지를 포함하는 사전 터치 상태를 식별하고 - 상기 로직은 상기 사전 터치 상태를 식별하기 위해 상기 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했는지 결정하고, 상기 사전 터치 임계치는 잠음 임계치보다 크고 상기 터치 상태에 대응하는 터치 임계치보다 작음 - ,

상기 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하며 - 상기 로딩은 상기 좌표 데이터 및 상기 캐패시턴스 값을 하나 이상의 레지스터 및 메모리 위치 중 적어도 하나에 기록하는 것을 포함하고, 상기 예비 데이터는 상기 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩되어서 새로운 소프트웨어 데이터 구조를 생성함 - ,

상기 예비 데이터가 상기 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하기 위해 상기 소프트웨어 스택에 상기 로딩된 예비 데이터가 예비적인 것임을 통지하고,

상기 터치 스크린에 관하여 터치 상태가 식별되면 상기 예비 데이터를 사용하여 터치 이벤트를 트리거하는

장치.

#### 청구항 8

삭제

#### 청구항 9

삭제

#### 청구항 10

제 7 항에 있어서,

상기 로직은 상기 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 상기 예비 데이터를 업데이트하는

장치.

#### 청구항 11

삭제

#### 청구항 12

삭제

#### 청구항 13

터치 스크린에 관하여, 좌표 데이터 및 캐패시턴스 값에 관한 하드웨어 통지를 포함하는 사전 터치 상태를 식별하는 단계 - 상기 사전 터치 상태를 식별하는 단계는 상기 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했는지 결정하는 단계를 포함하고, 상기 사전 터치 임계치는 잠음 임계치보다 크고 상기 터치 상태에 대응하는 터치 임계치보다 작음 - 와,

상기 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하는 단계 - 상기 로딩하는 단계는 상기 좌표 데이터 및 상기 캐패시턴스 값을 하나 이상의 레지스터 및 메모리 위치 중 적어도 하나에 기록하는 단계를 포함하고, 상기 예비 데이터는 상기 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩되어서 새로운 소프트웨어 데이터 구조를 생성함 - 와,

상기 예비 데이터가 상기 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하기 위해 상기 소프트웨어 스택에 상기 로딩된 예비 데이터가 예비적인 것임을 통지하는 단계와,

상기 터치 스크린에 관하여 터치 상태가 식별되면 상기 예비 데이터를 사용하여 터치 이벤트를 트리거하는 단계를 포함하는

방법.

#### 청구항 14

삭제

#### 청구항 15

삭제

#### 청구항 16

제 13 항에 있어서,

상기 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 상기 예비 데이터를 업데이트하는 단계를 더 포함하는

방법.

#### 청구항 17

삭제

#### 청구항 18

삭제

#### 청구항 19

명령어 세트를 포함하는 비일시적 컴퓨터 판독가능 저장 매체로서,

상기 명령어 세트는, 디바이스에 의해 실행되는 경우 상기 디바이스로 하여금

터치 스크린에 관하여, 좌표 데이터 및 캐패시턴스 값에 관한 하드웨어 통지를 포함하는 사전 터치 상태를 식별하게 하고 - 상기 사전 터치 상태를 식별하게 하는 것은 상기 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했는지 결정하게 하는 것을 포함하고, 상기 사전 터치 임계치는 잡음 임계치보다 크고 상기 터치 상태에 대응하는 터치 임계치보다 작음 - ,

상기 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하게 하며 - 상기 로딩하게 하는 것은 상기 좌표 데이터 및 상기 캐패시턴스 값을 하나 이상의 레지스터 및 메모리 위치 중 적어도 하나에 기록하게 하는 것을 포함하고, 상기 예비 데이터는 상기 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩되어서 새로운 소프트웨어 데이터 구조를 생성함 - ,

상기 예비 데이터가 상기 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하기 위해 상기 소프트웨어 스택에 상기 로딩된 예비 데이터가 예비적인 것임을 통지하고,

상기 터치 스크린에 관하여 터치 상태가 식별되면 상기 예비 데이터를 사용하여 터치 이벤트를 트리거하게 하는 컴퓨터 판독가능 저장 매체.

#### 청구항 20

삭제

## 청구항 21

삭제

## 청구항 22

제 19 항에 있어서,

상기 명령어 세트는, 실행되는 경우, 디바이스로 하여금 상기 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 상기 예비 데이터를 업데이트하게 하는

컴퓨터 판독가능 저장 매체.

## 청구항 23

삭제

## 청구항 24

삭제

## 청구항 25

삭제

## 발명의 설명

### 기술 분야

[0001] 실시예는 전반적으로 터치 스크린 디바이스에 관한 것이다. 보다 구체적으로, 실시예는 신속 응답 용량성 터치 스크린 디바이스에 관한 것이다.

### 배경 기술

[0002] 터치 스크린은 컴퓨팅 플랫폼 상의 커서 이동, 스크롤 동작 및 줌 동작과 같은 다양한 사용자 인터페이스(UI) 기반 기능을 수행하는 데 사용될 수 있다. 전통적으로, 터치 스크린의 패널과 접촉하는 손가락은 터치 스크린으로부터 하드웨어 데이터를 판독하고, 새로운 소프트웨어 데이터 구조를 생성하며, UI 기반 기능을 지원하는 다양한 소프트웨어 계층에 통지하는 것과 같은 소프트웨어 리액션을 개시할 수 있다. 게다가, 소프트웨어 계층은 차례로 더 많은 데이터 구조를 생성할 수 있다. 이들 소프트웨어 리액션이 발생하게 하는데 필요한 시간은 터치 스크린 디바이스의 응답 시간을 상당히 느리게 해서 최적이 아닌 사용자 경험을 야기할 수 있다.

### 도면의 간단한 설명

[0003] 실시예의 다양한 이점은 후속 설명 및 첨부된 청구범위를 판독하고 후속 도면을 참조함으로써 당업자에게 자명해질 것이다.

도 1은 실시예에 따른 사전 터치 상태의 예의 도시이다.

도 2는 실시예에 따라 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하는 방안의 예의 블록도이다.

도 3은 실시예에 따라 사전 터치 상태를 식별하는 방안의 예의 도시이다.

도 4는 실시예에 따라 터치 이벤트를 트리거하는 방법의 예의 흐름도이다.

도 5는 실시예에 따른 시스템의 예의 블록도이다.

### 발명을 실시하기 위한 구체적인 내용

- [0004] 이제 도 1을 참조하면, 사용자의 손가락(10)이 터치 스크린(14)의 용량성 패널(12) 근처에 있는 사전 터치 상태 시나리오가 도시된다. 도시된 예에서, 손가락(10)이 터치 스크린(14)의 패널(12)과 실제로 접촉하기 전에, 손가락(10)과 패널(12) 사이의 캐패시턴스( $C_f$ )의 변화가 검출될 수 있다. 보다 상세히 논의되는 바와 같이, 도시된 사전 터치 상태를 식별하고 사전 터치 상태와 연관된 예비 데이터를 레버리징(leveraging)하는 것은 터치 스크린(14)의 응답 시간이 상당히 감소되게 할 수 있다.
- [0005] 예컨대, 도 2는 사용자의 손가락이 터치 스크린과 실제로 접촉하기 전에 사전 터치 상태와 연관된 예비 데이터(16)가 소프트웨어 스택(18)(18a 내지 18f)으로 로딩되는 방안을 설명한다. 예비 데이터(16)는 예컨대, X-Y 좌표 데이터, 캐패시턴스 값 등을 포함할 수 있다. 예비 데이터(16)를 소프트웨어 스택(18)으로 로딩하는 것은 예컨대, 터치 스크린(14)으로부터 예비 데이터(16)를 판독하는 것, 새로운 소프트웨어 데이터 구조를 생성하는 것 및 사용자 인터페이스(UI) 기반 기능을 지원하는 다양한 소프트웨어 계층에 통지하는 것과 같은 특정 처리 기능이 터치 스크린(14)에 관하여 실제 터치 상태의 검출 이전에 처리되게 할 수 있다.
- [0006] 검증된 터치 상태가 발생할 때까지 예비 데이터(16)가 UI에 표시되는 것이 방지될 수 있다는 점에 특히 유의한다. 그러한 방안은 후속 터치 상태가 사전 터치 상태에 계속되지 않는 경우에 발생할 수 있는 긍정 오류(false positive)를 방지할 수 있다. 그럼에도, 실질적인 이점은 예컨대, 운영 체제(OS) 커널 계층(18b) 및/또는 애플리케이션 계층(18f)과 같은 소프트웨어 스택(18) 내의 계층이 전처리할 수 있게 하고 잠재적인 터치 이벤트를 발생 이전에 준비할 수 있게 함으로써 구현될 수 있다. 하드웨어(HW)/펌웨어(FW) 계층(18a), 프로토콜 계층(18c), 디바이스 드라이버 계층(18d) 및/또는 미들웨어 계층(18e)과 같은 다른 계층은 예비 데이터(16)로부터 이익을 얻는 프레임워크를 가질 수 있다. 예비 데이터(16)가 UI에 표시되지 않게 하는 것은, 예비 데이터(16)가 터치 상태가 아닌 사전 터치 상태와 연관된 것이라는 것을 소프트웨어 스택(18) 내의 하나 이상의 계층에 통지함으로써 달성될 수 있다. 예컨대, 그러한 통지 수신시에, 애플리케이션 계층(18f) 내의 서비스는 수신된 좌표를 적합한 자바스크립트 객체, 메소드 및/또는 연산으로 변환하도록 적합한 데이터 구조를 생성할 수 있지만, 터치 상태 및/또는 이벤트의 통지가 수신될 때까지 서비스의 UI를 통한 예비 데이터(16)의 처리를 완료할 수 있다.
- [0007] 도 3은 실질적인 성능 이점이 본 명세서에 설명된 사전 터치 기술을 통해 획득될 수 있음을 설명한다. 일반적으로, 사용자가 터치 스크린의 패널(12)과 상호작용하는 시나리오에 대해 시간에 걸쳐 등가 캐패시턴스 곡선이 도시된다. 보다 구체적으로, 사전 터치 임계치(20)가 수립될 수 있으며, 사전 터치 임계치(20)는 잠음 임계치(22)를 초과하고 터치 임계치(24) 미만이다. 도시된 예에서, 손가락(10)은 초기에 거리( $D_1$ )에 있으며 이는 캐패시턴스 값이 잠음 임계치(22) 미만이 되게 한다. 따라서, 터치 스크린 패널(12)은 시간( $t_0$ ) 이전에 도시된 시구간 동안아이들(idle)로 간주될 수 있다. 그러나, 시간( $t_0$ )에, 캐패시턴스 값이 사전 터치 임계치(20)를 초과하게 하도록 도시된 손가락(10)은 터치 스크린 패널(12)에 충분히 근접한다(예컨대, 거리  $D_2$ 에). 그러므로, 시간( $t_0$ )에, 사전 터치 상태와 연관된 예비 데이터(예컨대, 좌표 데이터, 캐패시턴스 값)는 시간( $t_0$ )에 터치 스크린 패널(12)과 연관된 소프트웨어 스택으로 로딩될 수 있으며, 예비 데이터를 로딩하는 것은 소프트웨어 스택의 하나 이상의 계층이 후속 터치 상태를 준비하는 것을 가능하게 할 수 있다.
- [0008] 도시된 예에서, 손가락(10)은 시간( $t_1$ )에 터치 스크린 패널(12)과 접촉하여, 캐패시턴스 값이 터치 임계치(24)를 초과하게 하고 터치 상태가 식별되게 한다. 사전 터치 상태에 응답하여 예비 데이터가 이전에 로딩되었으므로, 터치 이벤트는 시간( $t_1$ )과 실질적으로 동일한(또는 무시해도 될 정도로 상이한) 시간일 수 있는 시간( $t_2$ )에 트리거될 수 있다. 그러므로, 도시된 해결책은 터치 이벤트가 실제 터치 상태와 거의 동시에 트리거될 수 있게 한다. 이와 반대로, 종래의 해결책에서는, 시간( $t_0$ )에 어떠한 처리도 발생할 수 없고 시간( $t_1$ )과 시간( $t_2$ ) 사이에 상당한 시간이 존재할 수 있다.
- [0009] 이제 도 4를 참조하면, 터치 이벤트를 트리거하는 방법(26)이 도시된다. 방법(26)은 RAM(random access memory), ROM(read only memory), PROM(programmable ROM), 펌웨어, 플래시 메모리 등과 같은 머신 또는 컴퓨터 판독가능 저장 매체, 예컨대, PLA(programmable logic array), FPGA(field programmable gate array), CPLD(complex programmable logic device)와 같은 구성가능 로직, ASIC(application specific integrated circuit), CMOS(complementary metal oxide semiconductor) 또는 TTL(transistor-transistor logic) 기술과 같은 회로 기술을 사용하는 고정 기능 하드웨어 로직 또는 이들의 임의의 조합에 저장된 로직 명령어 세트로서 구현될 수 있다. 예컨대, 방법(26)으로 도시된 동작을 수행하는 컴퓨터 프로그램 코드는 자바, 스몰토크, C++

등과 같은 객체 지향 프로그래밍 언어 및 "C" 프로그래밍 언어 또는 유사한 프로그래밍 언어와 같은 종래의 절차 프로그래밍 언어를 포함하는 하나 이상의 프로그래밍 언어의 임의의 조합으로 기록될 수 있다.

[0010] 도시된 처리 블록(28)은 예컨대, 터치 스크린(14)(도 1)과 같은 터치 스크린으로부터 하드웨어 통지를 기다리는 것을 제공한다. 일례에서, 하드웨어 통지는 OS 커널 루틴에 의해 처리될 수 있는 인터럽트 요청(IRQ)이다. 하드웨어 통지에 응답하여 블록(30)에서 하드웨어 데이터(예컨대, 좌표 데이터, 캐패시턴스 값)가 터치 스크린으로부터 판독될 수 있되, 도시된 블록(32)은 하드웨어 통지를 발생시키는 상태의 유형을 결정한다. 그러므로 블록(32)에서의 결정은 하드웨어 통지의 유형을 식별하는 것을 수반할 수 있다.

[0011] 이에 관하여, 터치 스크린과 연관된 하드웨어 및/또는 펌웨어는 일반적으로 사전 터치 임계치 또는 터치 임계치가 초과하였는지 여부를 판정하도록 구성될 수 있다. 그러므로, 하드웨어 통지의 생성은 하나 이상의 캐패시턴스 값을 임계치와 비교하는 것을 수반할 수 있다. 비교가 하드웨어에 의해 처리되면, 2 개의 하드웨어 비교기 세트가 사용될 수 있다. 비교가 펌웨어에 의해 처리되면, 하나의 하드웨어 비교기로 충분할 수 있다. 사전 터치 임계치 또는 터치 임계치가 초과하였는지 여부를 판정하는 데 펌웨어를 사용하는 것은 펌웨어가, OS 또는 다른 소프트웨어의 호스트 프로세서 시간이나 구동 프로세스, 스케줄 등을 점유하지 않는 개별 루틴일 수 있다는 점에서 유리할 수 있다.

[0012] 사전 터치 상태가 식별되면, 블록(34)에서 사전 터치 상태와 연관된 예비 데이터로 고려될 수 있는 하드웨어 데이터는 소프트웨어 스택으로 로딩되고/되거나 업데이트될 수 있다. 예비 데이터를 소프트웨어 스택으로 로딩하는 것은 예컨대, 예비 데이터를 소프트웨어 스택에 노출되는 하나 이상의 레지스터 및/또는 메모리 위치에 기록하는 것을 수반할 수 있다. 예비 데이터를 업데이트하는 것은 식별된 사전 터치 상태가 이전 사전 터치 상태에 이어지는 경우 발생할 수 있다. 그러한 경우에, 이전에 로딩된 예비 데이터를 적합한 레지스터 및/또는 메모리 위치에 덮어쓰거나 보충하는 데 가장 최근의 예비 데이터가 사용될 수 있다. 로딩되고/되거나 업데이트된 예비 데이터는 예컨대, 소프트웨어 스택의 하나 이상의 계층에 데이터가 사실상 예비임을 통지함으로써 소프트웨어 스택과 연관된 UI에 표시되지 않을 수 있다. 그러므로 계층은 예비 데이터가 사용자의 시각으로부터 보이지 않게 하는 것에 대해 협력할 수 있다.

[0013] 반면에, 블록(32)에서 터치 상태가 식별(예컨대, 터치 임계치가 초과되고, 터치 하드웨어 통지가 수신됨)된다고 결정되면, 도시된 블록(36)은 예비 데이터가 터치 상태를 위해 이미 로딩되었는지 여부를 판정한다. 그러므로, 블록(36)에서의 판정은 예비 데이터가 충분히 최신(예컨대, 적합한 타이머가 만료하지 않음)인지 여부를 판정하는 것을 수반할 수 있다. 예비 데이터가 이미 로딩되었으면, 블록(38)에서 터치 이벤트를 트리거하는 데 로딩된 데이터가 즉시 사용될 수 있다. 그렇지 않으면, 블록(40)에서 터치 상태와 연관된 비예비 데이터로 고려될 수 있는 하드웨어 데이터가 소프트웨어 스택으로 로딩되고/되거나 업데이트될 수 있다. 예비 데이터와 마찬가지로, 비예비 데이터를 로딩하는 것은 소프트웨어 스택에 노출되는 하나 이상의 레지스터 및/또는 메모리 위치에 좌표 데이터 및/또는 캐패시턴스 값을 기록하는 것을 수반할 수 있다. 블록(40)을 바이패스하는 것은 상당한 응답 시간 감소 및 성능 개선을 가능하게 할 수 있다.

[0014] 도 5는 시스템(42)을 도시한다. 시스템(42)은 컴퓨팅 기능(예컨대, 개인 디지털 보조기기/PDA, 랩톱, 스마트 태블릿), 통신 기능(예컨대, 무선 스마트폰), 이미징 기능, 미디어 재생 기능(예컨대, 스마트 텔레비전/TV) 또는 이들의 임의의 조합(예컨대, 모바일 인터넷 디바이스/MID)을 가진 모바일 플랫폼의 일부일 수 있다. 도시된 예에서, 시스템(42)은 시스템에 전력을 공급하는 배터리(58) 및 시스템 메모리(48)와 통신할 수 있는 IMC(integrated memory controller)를 가진 프로세서(44)를 포함한다. 시스템 메모리(48)는 예컨대, DIMMs(dual inline memory modules), SODIMMs(small outline DIMMs) 등과 같은 하나 이상의 메모리 모듈로서 구성되는 예컨대, DRAM(dynamic random access memory)을 포함할 수 있다.

[0015] 도시된 시스템(42)은, 때때로 칩셋의 사우스브릿지로서 지칭되며, 호스트 디바이스로서 기능하고 예컨대, 터치 스크린(52) 및 대용량 저장장치(54)(예컨대, 하드 디스크 드라이브/HDD, 광학 디스크, 플래시 메모리 등)와 통신할 수 있는 입출력(IO) 모듈(50)을 또한 포함한다. 도시된 프로세서(44)는 터치 스크린(52)에 관하여 사전 터치 상태를 식별하고, 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하며, 터치 상태가 터치 스크린(52)에 관하여 식별되면 그 예비 데이터를 사용하여 터치 이벤트를 트리거하도록 구성되는 로직(56)을 실행할 수 있다. 로직(56)은 이와 달리 프로세서(44)의 외부에서 구현될 수 있다. 추가적으로, 프로세서(44) 및 IO 모듈(50)은 SoC(system on chip)로서 동일한 반도체 다이 상에 함께 구현될 수 있다.

[0016] 추가 주석 및 예



- [0017] 예 1은 시스템에 전력을 제공하는 배터리와, 터치 스크린과, 적어도 부분적으로 고정 기능 하드웨어로 구현되는 로직을 포함하는 터치 이벤트 관리 시스템을 포함할 수 있으며, 로직은 터치 스크린에 관하여 사전 터치 상태를 식별하고, 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하며, 터치 스크린에 관하여 터치 상태가 식별되면 예비 데이터를 사용하여 터치 이벤트를 트리거한다.
- [0018] 예 2는 예 1의 시스템을 포함할 수 있되, 로직은 예비 데이터가 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지한다.
- [0019] 예 3은 예 1의 시스템을 포함할 수 있되, 예비 데이터는 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩된다.
- [0020] 예 4는 예 1 내지 예 3 중 어느 하나의 시스템을 포함할 수 있되, 로직은 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 예비 데이터를 업데이트한다.
- [0021] 예 5는 예 1 내지 예 3 중 어느 하나의 시스템을 포함할 수 있되, 로직은 사전 터치 상태를 식별하기 위해 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했다고 결정한다.
- [0022] 예 6은 예 5의 시스템을 포함할 수 있되, 사전 터치 임계치는 잡음 임계치보다 크고 터치 상태에 대응하는 터치 임계치보다 작다.
- [0023] 예 7은 적어도 부분적으로 고정 기능 하드웨어로 구현되는 로직을 포함하는 터치 이벤트 관리 장치를 포함할 수 있고, 로직은 터치 스크린에 관하여 사전 터치 상태를 식별하고, 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하며, 터치 스크린에 관하여 터치 상태가 식별되면 예비 데이터를 사용하여 터치 이벤트를 트리거한다.
- [0024] 예 8은 예 7의 장치를 포함할 수 있되, 로직은 예비 데이터가 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지한다.
- [0025] 예 9는 예 7의 장치를 포함할 수 있되, 예비 데이터는 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩된다.
- [0026] 예 10은 예 7 내지 예 9 중 어느 하나의 장치를 포함할 수 있되, 로직은 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 예비 데이터를 업데이트한다.
- [0027] 예 11은 예 7 내지 예 9 중 어느 하나의 장치를 포함할 수 있되, 로직은 사전 터치 상태를 식별하기 위해 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했다고 결정한다.
- [0028] 예 12는 예 11의 장치를 포함할 수 있되, 사전 터치 임계치는 잡음 임계치보다 크고 터치 상태에 대응하는 터치 임계치보다 작다.
- [0029] 예 13은 터치 스크린에 관하여 사전 터치 상태를 식별하는 단계와, 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하는 단계와, 터치 스크린에 관하여 터치 상태가 식별되면 예비 데이터를 사용하여 터치 이벤트를 트리거하는 단계를 포함하는 터치 이벤트 관리 방법을 포함할 수 있다.
- [0030] 예 14는 예 13의 방법을 포함할 수 있되, 예비 데이터가 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하는 단계를 더 포함한다.
- [0031] 예 15는 예 13의 방법을 포함할 수 있되, 예비 데이터는 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나 이상으로 로딩된다.
- [0032] 예 16은 예 13 내지 예 15 중 어느 하나의 방법을 포함할 수 있되, 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 예비 데이터를 업데이트하는 단계를 더 포함한다.
- [0033] 예 17은 예 13 내지 예 15 중 어느 하나의 방법을 포함할 수 있되, 사전 터치 상태를 식별하는 단계는 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했다고 결정하는 단계를 포함한다.
- [0034] 예 18은 예 17의 방법을 포함할 수 있되, 사전 터치 임계치는 잡음 임계치보다 크고 터치 상태에 대응하는 터치 임계치보다 작다.
- [0035] 예 19는 명령어 세트를 포함하는 비일시적 컴퓨터 판독가능 저장 매체를 포함할 수 있고, 명령어 세트는 디바이스에 의해 실행되는 경우 디바이스로 하여금, 터치 스크린에 관하여 사전 터치 상태를 식별하고, 사전 터치 상

태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하며, 터치 스크린에 관하여 터치 상태가 식별되면 예비 데이터를 사용하여 터치 이벤트를 트리거하게 한다.

- [0036] 예 20은 예 19의 비밀시적 컴퓨터 판독가능 저장 매체를 포함할 수 있되, 명령어 세트는, 실행되는 경우, 디바이스로 하여금 예비 데이터가 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하게 한다.
- [0037] 예 21은 예 19의 비밀시적 컴퓨터 판독가능 저장 매체를 포함할 수 있되, 예비 데이터는 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩된다.
- [0038] 예 22는 예 19 내지 예 21 중 어느 하나의 비밀시적 컴퓨터 판독가능 저장 매체를 포함할 수 있되, 명령어 세트는, 실행되는 경우, 디바이스로 하여금 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 예비 데이터를 업데이트하게 한다.
- [0039] 예 23은 예 19 내지 예 21 중 어느 하나의 비밀시적 컴퓨터 판독가능 저장 매체를 포함할 수 있되, 명령어 세트는, 실행되는 경우, 디바이스로 하여금 사전 터치 상태를 식별하기 위해 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했다고 결정하게 한다.
- [0040] 예 24는 예 23의 비밀시적 컴퓨터 판독가능 저장 매체를 포함할 수 있되, 사전 터치 임계치는 잡음 임계치보다 크고 터치 상태에 대응하는 터치 임계치보다 작다.
- [0041] 예 25는 터치 스크린에 관하여 사전 터치 상태를 식별하는 수단과, 사전 터치 상태와 연관된 예비 데이터를 소프트웨어 스택으로 로딩하는 수단과, 터치 스크린에 관하여 터치 상태가 식별되면 예비 데이터를 사용하여 터치 이벤트를 트리거하는 수단을 포함하는 터치 이벤트 관리 장치를 포함할 수 있다.
- [0042] 예 26은 예 25의 장치를 포함할 수 있되, 예비 데이터가 소프트웨어 스택과 연관된 사용자 인터페이스에 표시되는 것을 방지하는 수단을 더 포함한다.
- [0043] 예 27은 예 26의 장치를 포함할 수 있되, 예비 데이터는 소프트웨어 스택의 운영 체제 계층 또는 애플리케이션 계층 중 하나로 로딩된다.
- [0044] 예 28은 예 25 내지 예 27 중 어느 하나의 장치를 포함할 수 있되, 터치 상태 또는 후속 사전 터치 상태 중 하나 이상에 기초하여 예비 데이터를 업데이트하는 수단을 더 포함한다.
- [0045] 예 29는 예 25 내지 예 27 중 어느 하나의 장치를 포함할 수 있되, 사전 터치 상태를 식별하기 위해 터치 스크린의 캐패시턴스 값이 사전 터치 임계치를 초과했다고 결정하는 수단을 더 포함한다.
- [0046] 예 30은 예 29의 장치를 포함할 수 있되, 사전 터치 임계치는 잡음 임계치보다 크고 터치 상태에 대응하는 터치 임계치보다 작다.
- [0047] 그러므로, 본 명세서에 설명된 기술은 손가락이 터치 스크린 표면에 실제로 놓이기 전에 터치 스크린에 관하여 손가락 움직임을 추적할 수 있다. 그 결과, 다양한 소프트웨어 계층이 추적된 예비 데이터에 기초하여 터치 이벤트를 트리거할 준비가 되어 있을 수 있다. 따라서, 실질적인 성능 개선이 달성될 수 있다. 그러한 개선은 게임 애플리케이션 및 응답 시간이 관심 영역인 다른 UI 기반 애플리케이션에 특히 유리할 수 있다.
- [0048] 실시예는 반도체 집적 회로(IC) 칩의 모든 유형과 함께 사용하기 위해 적용가능하다. IC 칩의 예는 프로세서, 제어기, 칩셋 구성요소, 프로그램가능 로직 어레이(PLA), 메모리 칩, 네트워크 칩, 시스템 온 칩(SoC), SSD/NAND 제어기 ASIC 등을 포함하지만, 이들로 제한되지 않는다. 또한, 일부 도면에서, 신호 도전체 라인은 선으로 표현될 수 있다. 일부는 더 많은 구성 신호 경로를 나타내기 위해 상이할 수 있고, 구성 신호 경로의 개수를 나타내는 숫자 라벨을 가지며/가지거나 1차 정보 흐름 방향을 나타내는 하나 이상의 종단에서의 화살표를 가질 수 있다. 그러나, 이는 제한적인 방법으로 해석되어서는 안 된다. 오히려, 그러한 추가 세부사항은 회로의 쉬운 이해를 가능하게 하도록 하나 이상의 예시적인 실시예와 함께 사용될 수 있다. 임의의 표시된 신호 라인은 추가 정보를 갖든 갖지 않든 다수의 방향으로 이동할 수 있는 하나 이상의 신호를 실제로 포함할 수 있고, 임의의 적합한 유형의 신호 스킴, 예컨대, 상이한 쌍들로 구현되는 디지털 또는 아날로그 라인, 광섬유 라인 및/또는 싱글 엔드형 라인으로 구현될 수 있다.
- [0049] 예시적인 사이즈/모델/값/범위가 주어질 수 있지만, 실시예는 동일한 것으로 제한되지 않는다. 제조 기술(예컨대, 포토리소그래피)이 시간이 지남에 따라 완성되므로, 더 작은 크기의 디바이스가 제조될 수 있을 것으로 예상된다. 또한, 잘 알려져 있는 IC 칩 및 다른 구성요소에 대한 전원/접지 접속은 간결한 예시 및 논의를 위해 그리고 실시예의 특정 양상을 불명료하게 하지 않도록 도면 내에 도시될 수도 도시되지 않을 수도 있다. 또한,

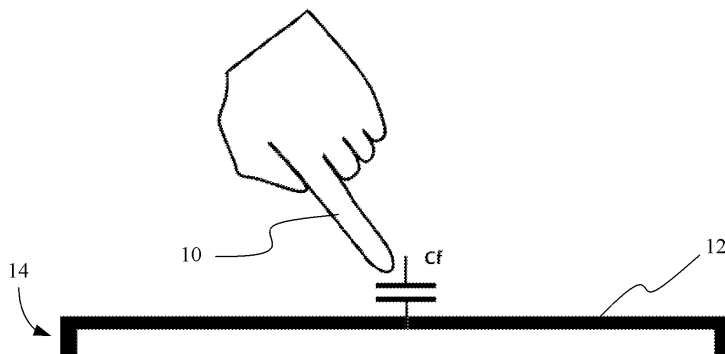
배치는 실시예를 불명료하지 않게 하기 위해 블록도 형태로 또한 그러한 블록도 배치의 구현에 관한 세부사항이 실시예가 구현되는 플랫폼에 크게 의존한다, 즉, 그러한 세부사항이 당업자의 범위 내에 충분히 있어야 한다는 사실을 고려하여 도시될 수 있다. 특정 세부사항(예컨대, 회로)은 예시적인 실시예를 설명하도록 제시되는 경우, 실시예가 이들 특정 세부사항 없이 또는 특정 세부사항의 변경으로 실시될 수 있음은 당업자에게 자명해야 한다. 그러므로 설명은 제한 대신에 예시적인 것으로 간주된다.

[0050] 용어 "연결된"은 본 명세서에서 문제의 구성요소들 사이의 임의의 유형의 관계를 직접 또는 간접적으로 지칭하는 데 사용될 수 있고, 전기, 기계, 유체, 광학, 전자기, 전기기계 또는 다른 접속에 적용할 수 있다. 또한, 용어 "제 1", "제 2" 등은 본 명세서에서 논의를 용이하게 하는 데에만 사용될 수 있고 이와 달리 나타내지 않는 한 어떠한 특정 시간 또는 연대 의미도 전달하지 않는다.

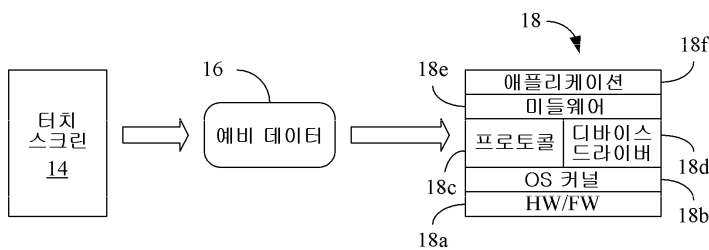
[0051] 당업자는 전술한 설명으로부터 실시예의 광범위한 기술이 다양한 형태로 구현될 수 있음을 알 것이다. 그러므로, 실시예는 특정 예시와 관련하여 설명되었지만, 도면, 명세서 및 후속하는 청구범위의 학습을 통해 다른 변형이 당업자에게 자명해질 것이므로 실시예의 진정한 범위는 제한되어서는 안 된다.

## 도면

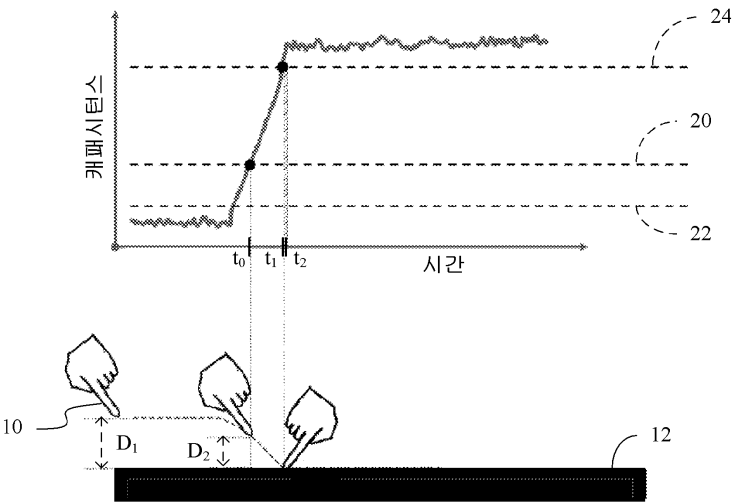
### 도면1



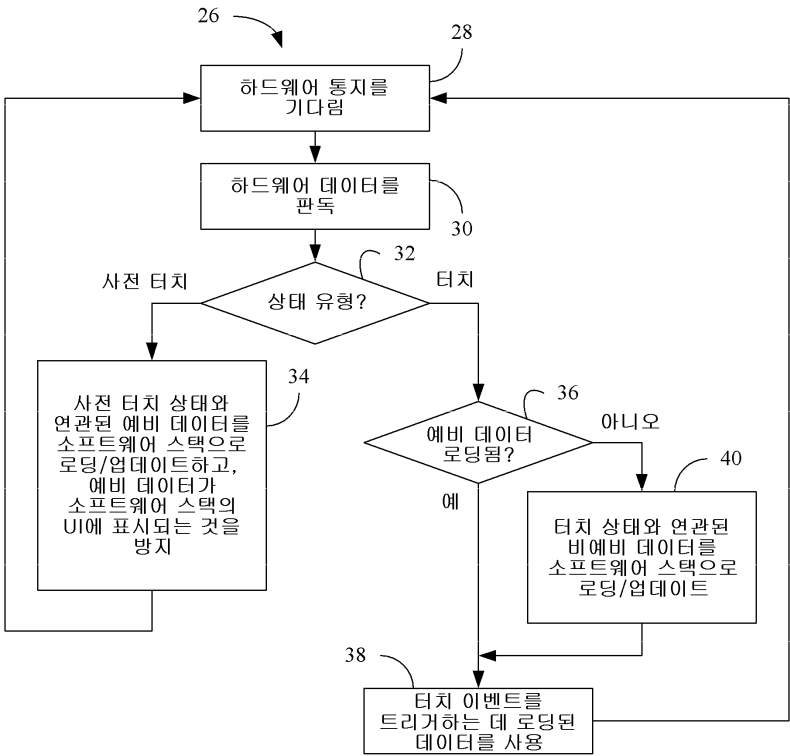
### 도면2



도면3



도면4



도면5

