



(19) 대한민국특허청(KR)  
(12) 등록특허공보(B1)

(45) 공고일자 2023년01월26일  
(11) 등록번호 10-2491361  
(24) 등록일자 2023년01월18일

(51) 국제특허분류(Int. Cl.)  
G10L 25/48 (2013.01) B60R 16/023 (2006.01)  
B60W 50/00 (2006.01) B60W 50/14 (2020.01)  
G06F 9/455 (2018.01) G10L 15/22 (2006.01)  
(52) CPC특허분류  
G10L 25/48 (2013.01)  
B60R 16/023 (2013.01)  
(21) 출원번호 10-2020-0161584  
(22) 출원일자 2020년11월26일  
심사청구일자 2020년11월26일  
(65) 공개번호 10-2022-0073473  
(43) 공개일자 2022년06월03일  
(56) 선행기술조사문헌  
JP2018063133 A\*  
(뒷면에 계속)

(73) 특허권자  
주식회사 드림에이스  
대구광역시 달성군 유가읍 테크노상업로 84, 802호  
(72) 발명자  
이원  
서울특별시 성동구 왕십리로 58, 615호(성수동1가)  
(74) 대리인  
특허법인다나

전체 청구항 수 : 총 5 항

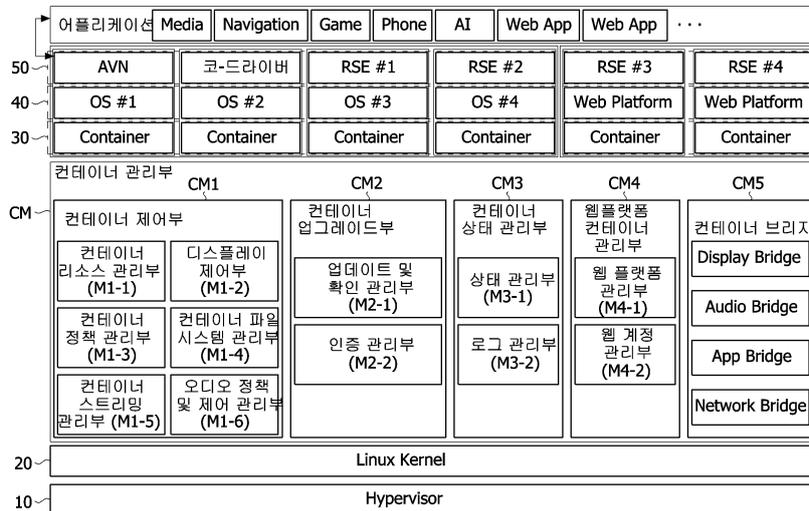
심사관 : 홍경아

(54) 발명의 명칭 차량 가상화 구조 기반의 시스템 제어 방법 및 장치

(57) 요약

실시예는 제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신하는 단계; 수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교하는 단계; 및 비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어로 송신하는 단계;를 포함하고, 상기 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재되는 차량 가상화 구조 기반의 사운드 제어 방법을 개시한다.

대표도 - 도2



(52) CPC특허분류

*B60W 50/14* (2013.01)  
*G06F 9/45533* (2013.01)  
*G10L 15/22* (2013.01)  
*B60W 2050/0094* (2013.01)  
*B60W 2050/146* (2013.01)

(56) 선행기술조사문헌

KR1020160054931 A\*  
JP2007183110 A  
KR1020170016902 A  
KR101027415 B1

\*는 심사관에 의하여 인용된 문헌

---

**명세서**

**청구범위**

**청구항 1**

제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신하는 단계;

수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교하는 단계; 및

비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어로 송신하는 단계;를 포함하고,

상기 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재되고,

상기 제1 사운드 출력정보의 우선순위는 상기 제2 사운드 출력정보의 우선순위보다 높고,

상기 우선순위를 비교하는 단계 이후에,

상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중인지 여부 확인하는 단계; 및

상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중이라면 상기 제1 사운드 출력정보의 레벨에 따라 상기 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력 여부 및 출력 시간을 조절하여 출력하는 단계;를 더 포함하고,

상기 출력정보의 레벨은 제1 레벨 및 상기 제1 레벨보다 낮은 제2 레벨을 포함하고,

상기 제1 사운드 출력정보의 레벨이 제1 레벨이면 상기 제1 사운드 출력정보만 출력하고, 상기 제2 사운드 출력정보의 출력을 차단하는 단계;를 더 포함하고,

상기 제1 사운드 출력정보의 레벨이 제2 레벨이면, 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보를 제1 시간 동안 출력하는 단계; 및

상기 제2 사운드 출력정보를 제1 시간에 연속하는 제2 시간 동안 출력하는 단계;를 포함하고,

상기 제1 시간 동안 상기 제1 사운드 출력정보의 크기는 상기 제2 사운드 출력정보의 크기보다 큰 차량 가상화 구조 기반의 사운드 제어 방법.

**청구항 2**

제1항에 있어서,

상기 제1 사운드 클라이언트는 클러스터 운영체제에 탑재되고,

상기 제2 사운드 클라이언트는 AVN(Audio Video Navigation) 운영체제, 코-드라이버(Co-Driver) 운영체제, RSE(Rear Seat Entertainment) 운영체제에 탑재되는 차량 가상화 구조 기반의 사운드 제어 방법.

**청구항 3**

삭제

**청구항 4**

삭제

**청구항 5**

제1항에 있어서,

상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중이 아니라면 상기 제1 사운드 출력정보를 출력한 후 상기 제2 사운드 출력정보를 출력하는 단계;를 더 포함하는 차량 가상화 구조 기반의 사운드 제어 방법.

**청구항 6**

삭제

**청구항 7**

삭제

**청구항 8**

삭제

**청구항 9**

제1항에 있어서,

상기 제2 사운드 출력정보만을 수신하면 상기 송신하는 단계에서 상기 제2 사운드 출력정보를 상기 하드웨어로 송신하는 차량 가상화 구조 기반의 사운드 제어 방법.

**청구항 10**

제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신하는 수신부;

수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교하는 비교부; 및

비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어로 송신부;를 포함하고,

상기 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재되고,

상기 제1 사운드 출력정보의 우선순위는 상기 제2 사운드 출력정보의 우선순위보다 높고,

상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중인지 여부 확인하고, 상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중이라면 상기 제1 사운드 출력정보의 레벨에 따라 상기 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력 여부 및 출력 시간을 조절하여 출력하는 판단부;를 더 포함하고,

상기 출력정보의 레벨은 제1 레벨 및 상기 제1 레벨보다 낮은 제2 레벨을 포함하고,

상기 판단부는, 상기 제1 사운드 출력정보의 레벨이 제1 레벨이면 상기 제1 사운드 출력정보만 출력하고, 상기 제2 사운드 출력정보의 출력을 차단하고, 상기 제1 사운드 출력정보의 레벨이 제2 레벨이면, 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보를 제1 시간 동안 출력하며, 상기 제2 사운드 출력정보를 제1 시간에 연속하는 제2 시간 동안 출력하는 것을 판단하고,

상기 제1 시간 동안 상기 제1 사운드 출력정보의 크기는 상기 제2 사운드 출력정보의 크기보다 큰 차량 가상화 구조 기반의 사운드 제어 장치.

**발명의 설명**

**기술 분야**

본 발명은 실시에는 차량 가상화 구조 기반의 시스템 제어 방법 및 장치에 관한 것이다.

**배경 기술**

[0001]

- [0002] 차량은 보다 높은 수준의 전산화로 점차 이동하고 있다. 대부분의 차량 작업, 기능 및 작업은 현재 컴퓨터 제어 중이거나 전산 장치로 모니터링될 수 있다. 다시 말해, 디지털 시대가 도래함에 따라 소비자들은 휴대폰과 태블릿 컴퓨터 사이와 유사한 수행을 차량 내에서도 수행할 수 있다. 예컨대, 터치 및 햅틱 피드백, 자연 언어 음성 상호 작용, 근접 감지 및 버튼 및 컨트롤을 갖춘 지능형 사용자 인터페이스가 있는 차량용 인포테인먼트 시스템이 현재 각광받고 있다. 이에, 차량 내의 정보, 통신 내비게이션, 동시에 엔터테인먼트를 간소화하고 운전 사용에 맞게 조정하고 있다.
- [0003] 이러한 기술 상황에 맞춰, 차량 내에는 복수의 디스플레이 장치가 존재하며, 디스플레이 장치를 실행하는 소프트웨어도 다수 존재한다. 또한, 차량 에서 사용자 또는 탑승자는 복수의 디스플레이 장치에 사용 자유도가 향상되고 있다.
- [0004] 다만, 차량 내에서 운영체제 간의 업데이트가 어렵고, 웹 플랫폼 컨테이너 사용이 사용자 별 독립적이지 못한 한계가 존재한다.
- [0005] 나아가, 차량 내에서 사용자 간 음향이 섞여 주행의 위험도가 증가해지는 문제점이 있다.

**발명의 내용**

**해결하려는 과제**

- [0006] 실시예는 디스크 자원 절약이 가능한 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 제공할 수 있다.
- [0007] 또한, 구동 속도가 개선된 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 제공할 수 있다.
- [0008] 또한, 차량 내 사용자 맞춤형 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 제공할 수 있다.
- [0009] 또한, 차량 내 사용자 별 웹 어플리케이션을 용이하게 사용하는 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 제공할 수 있다.
- [0010] 또한, 주행 방해가 제거된 차량 내 음향 처리가 이루어지는 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 제공할 수 있다.
- [0011] 실시예에서 해결하고자 하는 과제는 이에 한정되는 것은 아니며, 아래에서 설명하는 과제의 해결수단이나 실시 형태로부터 파악될 수 있는 목적이나 효과도 포함될 수 있다고 할 것이다.

**과제의 해결 수단**

- [0012] 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 방법은 제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신하는 단계; 수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교하는 단계; 및 비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어로 송신하는 단계;를 포함하고, 상기 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재된다.
- [0013] 상기 제1 사운드 클라이언트는 클러스터 운영체제에 탑재되고, 상기 제2 사운드 클라이언트는 AVN(Audio Video Navigation) 운영체제, 코-드라이버(Co-Driver) 운영체제, RSE(Rear Seat Entertainment) 운영체제에 탑재될 수 있다.
- [0014] 상기 제1 사운드 출력정보의 우선순위는 상기 제2 사운드 출력정보의 우선순위보다 높을 수 있다.
- [0015] 상기 우선순위를 비교하는 단계 이후에, 상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중인지 여부 확인하는 단계;를 더 포함할 수 있다.
- [0016] 상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중이 아니라면 상기 제1 사운드 출력정보를 출력한 후 상기 제2 사운드 출력정보를 출력하는 단계;를 더 포함할 수 있다.
- [0017] 상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중이라면 상기 제1 사운드 출력정보의 레벨에 따라 상기 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력 여부 및 출력 시간을 조절하여 출력하는 단계;를 포함할 수 있다.
- [0018] 상기 출력정보의 레벨은 제1 레벨 및 상기 제1 레벨보다 낮은 제2 레벨을 포함하고, 상기 제1 사운드 출력정보의 레벨이 제1 레벨이면 상기 제1 사운드 출력정보만 출력하고, 상기 제2 사운드 출력정보의 출력을 차단하는

단계;를 더 포함할 수 있다.

- [0019] 상기 제1 사운드 출력정보의 레벨이 제2 레벨이면, 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보를 제1 시간 동안 출력하는 단계; 및 상기 제2 사운드 출력정보를 제1 시간에 연속하는 제2 시간 동안 출력하는 단계;를 포함하고, 상기 제1 시간 동안 상기 제1 사운드 출력정보의 크기는 상기 제2 사운드 출력정보의 크기보다 클 수 있다.
- [0020] 상기 제2 사운드 출력정보만을 수신하면 상기 송신하는 단계에서 상기 제2 사운드 출력정보를 상기 하드웨어로 송신할 수 있다.
- [0021] 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 장치는 제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신하는 수신부; 수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교하는 비교부; 및 비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어로 송신부;를 포함하고, 상기 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재된다.

**발명의 효과**

- [0022] 실시예에 따르면, 디스크 자원 절약이 가능한 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 구현할 수 있다.
- [0023] 또한, 구동 속도가 개선된 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 구현할 수 있다.
- [0024] 또한, 차량 내 사용자 맞춤형 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 구현할 수 있다.
- [0025] 또한, 차량 내 사용자 별 웹 어플리케이션을 용이하게 사용하는 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 구현할 수 있다.
- [0026] 또한, 주행 방향이 제거된 차량 내 음향 처리가 이루어지는 차량 가상화 구조 기반의 시스템 제어 방법 및 장치를 구현할 수 있다.
- [0027] 본 발명의 다양하면서도 유익한 장점과 효과는 상술한 내용에 한정되지 않으며, 본 발명의 구체적인 실시형태를 설명하는 과정에서 보다 쉽게 이해될 수 있을 것이다.

**도면의 간단한 설명**

- [0028] 도 1은 실시예에 따른 차량 가상화 구조 기반의 시스템의 개념도이고,
- 도 2는 실시예에 따른 차량 가상화 구조 기반의 시스템의 구체적인 블록도이고,
- 도 3은 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법에 대한 순서도이고,
- 도 4는 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법에서 업그레이드의 필요성 확인에 대한 구체적인 순서도이고,
- 도 5는 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법에 대한 구체적인 순서도이고,
- 도 6은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너의 구조에 대한 개념도이고,
- 도 7 및 도 8은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너를 구동을 설명하는 도면이고,
- 도 9는 도 2에서 K부분의 확대도이고,
- 도 10은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 웹 플랫폼 제어 방법에 대한 순서도이고,
- 도 11은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 웹 플랫폼 제어 방법에서 웹 어플리케이션의 구동을 도시한 도면이고,
- 도 12는 다른 실시예에 따른 차량 가상화 구조 기반의 시스템의 개념도이고,
- 도 13은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 디스플레이 장치를 설명하는 도면이고,
- 도 14는 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 장치의 블록도이고,

- 도 15는 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 방법의 순서도이고,
- 도 16은 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 방법의 구체화된 순서도이고,
- 도 17은 일예에 대한 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력을 설명하는 도면이고,
- 도 18은 다른예에 대한 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력을 설명하는 도면이고,
- 도 19는 또 다른예에 대한 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력을 설명하는 도면이고,
- 도 20은 변형예에 따른 차량 가상화 구조 기반의 시스템의 개념도이다.

**발명을 실시하기 위한 구체적인 내용**

- [0029] 본 발명은 다양한 변경을 가할 수 있고 여러 가지 실시예를 가질 수 있는 바, 특정 실시예들을 도면에 예시하고 설명하고자 한다. 그러나, 이는 본 발명을 특정한 실시 형태에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물 내지 대체물을 포함하는 것으로 이해되어야 한다.
- [0030] 제2, 제1 등과 같이 서수를 포함하는 용어는 다양한 구성요소들을 설명하는데 사용될 수 있지만, 구성요소들은 용어들에 의해 한정되지는 않는다. 용어들은 하나의 구성요소를 다른 구성요소로부터 구별하는 목적으로만 사용된다. 예를 들어, 본 발명의 권리 범위를 벗어나지 않으면서 제2 구성요소는 제1 구성요소로 명명될 수 있고, 유사하게 제1 구성요소도 제2 구성요소로 명명될 수 있다. 및/또는 이라는 용어는 복수의 관련된 기재된 항목들의 조합 또는 복수의 관련된 기재된 항목들 중의 어느 항목을 포함한다.
- [0031] 어떤 구성요소가 다른 구성요소에 "연결되어" 있다거나 "접속되어" 있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결되어 있거나 또는 접속되어 있을 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어" 있다거나 "직접 접속되어" 있다고 언급된 때에는, 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다.
- [0032] 본 출원에서 사용한 용어는 단지 특정한 실시예를 설명하기 위해 사용된 것으로, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 출원에서, "포함하다" 또는 "가지다" 등의 용어는 명세서상에 기재된 특징, 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성요소, 부품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.
- [0033] 다르게 정의되지 않는 한, 기술적이거나 과학적인 용어를 포함해서 여기서 사용되는 모든 용어들은 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가지고 있다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥 상 가지는 의미와 일치하는 의미를 가지는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.
- [0034] 이하, 첨부된 도면을 참조하여 실시예를 상세히 설명하되, 도면 부호에 관계없이 동일하거나 대응하는 구성 요소는 동일한 참조 번호를 부여하고 이에 대한 중복되는 설명은 생략하기로 한다.
- [0035] 본 개시의 실시예는 차량, 차량 인포테인먼트 등을 위한 차량 스트리밍 제어 장치 및 제어 방법에 관한 것으로, 애플리케이션(동일한 의미로 '앱', '어플리케이션', application) 등을 차량에서 이용함에 있어서, 차량 인포테인먼트와 접목한 시스템 환경을 제공하는 것을 내용으로 한다. 다만, 이러한 내용이 본 특허의 권리를 한정하는 것은 아니다.
- [0036] 보다 상세히 서술하면, 차량용 시스템의 경우 차량용 인포테인먼트 기능을 제공하는 것과 동시에 차량의 제어에 관련된 중요한 기능을 수행하고 있다. 여기서 차량의 제어에 관련된 기능이라 함은 사용자의 차량 조작에 영향을 주는 네 비게이션 시스템 등을 포함하는 애플리케이션부터 차량의 속도나 RPM 등을 표시하는 계기 판을 뜻하는 클러스터 시스템, 자율제어 차량의 자율제어 시스템과 같이 탑승자의 생명에 관련된 중요한 기능까지도 포함한다. 그리고 이러한 애플리케이션의 디스플레이는 자유롭게 이루어지되, 차량 제어 등 주행에 방해가 발생하면 안된다. 즉, 차량 주행에 대한 보호는 필수적으로 지켜져야 한다. 이러한 보호가 이루어지지 않을 경우, 사고가 발생할 수도 있다. 따라서, 본 발명의 바람직한 실시예에서는 주행을 방해하지 않는 범위 내에서 차량 내의 스트리밍이 수행되어, 오디오 재생, 비디오 재생, 게임 실행 등과 같이 엔터테인먼트에 관련된 서비스, 애플리케이션이 운전자를 포함한 차량의 탑승자에 대한 접근 및 조작이 용이하게 이루어질 수 있다.

- [0037] 도 1은 실시예에 따른 차량 가상화 구조 기반의 시스템의 개념도이고, 도 2는 실시예에 따른 차량 가상화 구조 기반의 시스템의 구체적인 블록도이다.
- [0038] 도 1 및 도 2를 참조하면, 실시예에 따른 차량 가상화 구조 기반의 시스템은 하드웨어(미도시됨), 하드웨어(미도시됨) 상에 탑재된 베이스 운영 체제(10), 베이스 운영 체제(10) 상에 탑재된 제1 가상엔진(20a)과 제2 가상엔진(20b), 각 가상엔진 상에 탑재되는 적어도 하나의 운영체제(30a 내지 30f), 적어도 하나의 운영체제(30a 내지 30f) 상에 탑재되고 미들웨어 및 중요 애플리케이션이 포함된 소프트웨어 집합으로 복수의 운영체제 플랫폼(40a 내지 40f), 운영체제 플랫폼(40a 내지 40f)에서 구동되는 복수의 애플리케이션(50a 내지 50d), 복수의 운영체제 플랫폼(40a 내지 40f) 또는 복수의 애플리케이션 등을 출력하는 복수의 디스플레이(60a 내지 60f) 및 가상엔진(20a 내지 20b) 상에서 복수의 운영체제(30a 내지 30d)에 대한 환경을 관리하는 컨테이너 관리부(CM)를 포함할 수 있다. 적어도 하나의 운영체제는 이하 '컨테이너(container)'로 설명한다.
- [0039] 여기서, 하드웨어(미도시됨)는 프로세서, 표시부, 저장부, 메모리부, 컨트롤부, I/O 장치를 포함하는 개념일 수 있다.
- [0040] 실시예에서, 주행부(CP)는 제1 가상엔진(20a) 상에서 탑재 또는 구동되는 시스템 영역이고, 인포테인먼트부(IP, Infotainment Platform)는 제1 가상엔진(20a)과 이종의 가상엔진인 제2 가상엔진(20b) 상에서 탑재 또는 구동되는 시스템 영역일 수 있다. 이러한 구성에 의하여, 인포테인먼트부(IP) 내에서 업그레이드 또는 기타 오작동이 발생하더라도, 주행부(CP)는 이에 대한 영향을 받지 않을 수 있다. 즉, 주행 중 영향이 최소화될 수 있다. 나아가, 인포테인먼트부(IP) 이외에 웹 플랫폼부(Web Platform)이 베이스 운영 체제(10) 및 추가 가상 엔진(20) 상에서 탑재 또는 구동될 수 있다. 인포테인먼트부(IP)와 마찬가지로 플랫폼부(Web Platform)는 컨테이너 관리자를 인포테인먼트부(IP)와 공유할 수 있다. 또는 플랫폼부(Web Platform)는 인포테인먼트부(IP)의 일부로 구성될 수 있다. 이하에서는 개별적인 구성으로 설명한다.
- [0041] 플랫폼부(Web Platform)는 인포테인먼트부(IP)와 같이 컨테이너 관리부에 탑재된 하나의 운영체제(예로, 컨테이너, 30), 운영체제 상에 탑재되고 미들웨어 및 중요 애플리케이션이 포함된 소프트웨어 집합으로 복수의 운영체제 플랫폼(예로, AGL, 40), 운영체제 플랫폼(40)에서 구동되는 복수의 애플리케이션(예로, 웹어플리케이션, 50)을 포함할 수 있다.
- [0042] 먼저, 베이스 운영 체제(10)는 예를 들어 다양한 운영체제일 수 있다. 예컨대, 베이스 운영 체제(10)는 리눅스(linux), 하이퍼바이저(hypervisor), QNX, GENIVI 등을 포함할 수 있다.
- [0043] 예를 들어, 제1 가상엔진(20a)과 제2 가상엔진(20b)은 미들웨어 솔루션 또는 다양한 플랫폼으로서, 모바일 C 언어로 개발될 수 있다. 제1 가상엔진(20a)과 제2 가상엔진(20b)은 다수의 내장 라이브러리를 제공할 수 있으며, 다양한 이동 단말기 등에서 동일한 동작을 수행할 수도 있다. 예컨대, 제1 가상엔진(20a)과 제2 가상엔진(20b)은 리눅스 기반의 안드로이드 커널일 수 있으며, 메모리 보호, 가상 메모리 모듈 및 스케줄 캐싱을 초기화할 수도 있다.
- [0044] 또한, 적어도 하나의 컨테이너(30a 내지 30d)는 제2 가상엔진(20a) 상에 탑재되고 제1 가상엔진(10a) 상에 탑재되지 않을 수 있다. 이에, 주행부(CP)에 대한 보안성이 향상되므로, 주행 안정성도 개선될 수 있다.
- [0045] 적어도 하나의 컨테이너(30a 내지 30d)는 실시예로 가상화 방식의 일 형태로서 프로세스 가상화의 일정일 수 있다. 예컨대, 컨테이너를 이용한 가상화 기술은 호스트 OS(operating system) 내부를 물리적 자원을 관리하는 커널 공간과 사용자 프로세스, 즉 응용 프로그램(어플리케이션, application, APP)을 실행하는 사용자 공간으로 구분하고 사용자 공간을 여러 개로 나누어 각각의 사용자 프로세스에서 사용되는 하드웨어 자원을 할당하고 공유하는 기술을 의미할 수 있다. 예컨대, 컨테이너(30a 내지 30d)는 호출된 라이브러리를 시스템 라이브러리와 인터페이스되도록 변환하여, 베이스 운영체제와 복수의 운영체제 플랫폼 간의 연결 또는 호환을 수행할 수 있다.
- [0046] 복수의 운영체제 플랫폼(40a 내지 40f)은 컨테이너 상에 탑재될 수 있다. 복수의 운영체제 플랫폼(40a 내지 40f)은 안드로이드, AGL(Automotive Grade Linux), 웹(web)플랫폼, 클러스터(cluster) 플랫폼, 헤드업 디스플레이(HUD) 플랫폼 등을 포함할 수 있다.
- [0047] 적어도 하나의 애플리케이션(50a 내지 50d)은 시스템 프로그램을 제외한 응용 프로그램으로써, 차량 내에서는 AVN(Audio Video Navigation), 코-드라이버(Co-Driver), RSE(Rear Seat Entertainment) 등을 포함할 수 있다.
- [0048] 복수의 디스플레이(60a 내지 60f)는 운영체제 플랫폼(40a 내지 40f) 또는 플랫폼 상의 응용 프로그램 등을 사용

자(예로, 탑승자)에게 표시할 수 있다. 예컨대, 복수의 디스플레이(60a 내지 60f)는 다양한 표시 장치(예로, OLED)를 포함할 수 있다.

- [0049] 컨테이너 관리부(CM)는 가상엔진(20a 내지 20b) 상에서 복수의 운영체제(30a 내지 30d)에 대한 환경을 관리할 수 있다. 즉, 컨테이너 관리부(CM)는 복수의 운영체제(30a 내지 30d) 예로, 컨테이너에 대한 자원 할당, 스트리밍 연결, 업데이트, 디스플레이 출력 담당, 출력부와 연결 등을 수행할 수 있다. 이에 대한 자세한 설명은 후술한다.
- [0050] 구체적으로, 컨테이너 관리부(CM)는 컨테이너 제어부(CM1), 컨테이너 업그레이드부(CM2), 컨테이너 상태관리부(CM3), 웹 플랫폼 컨테이너 관리부(CM4), 컨테이너 브리지부(CM5)를 포함할 수 있다.
- [0051] 컨테이너 제어부(CM1)는 컨테이너에 할당되는 시스템 자원, 자원 정책, 스트리밍 제어 관련, 디스플레이로 출력 관련, 파일 시스템 관리 등을 수행할 수 있다.
- [0052] 이러한 컨테이너 제어부(CM1)는 컨테이너 리소스 관리부(M1-1), 디스플레이 제어부(M1-2), 컨테이너 정책 관리부(M1-3), 컨테이너 파일 시스템 관리부(M1-4), 컨테이너 스트리밍 관리부(M1-5) 및 오디오 정책 및 제어 관리부(M1-6)를 포함할 수 있다.
- [0053] 컨테이너 리소스 관리부(M1-1)는 컨테이너에 할당되는 시스템 자원을 동적으로 할당할 수 있다. 예컨대, 컨테이너 리소스 관리부(M1-1)는 자원 사용량을 수집하여 자원 부족으로 인한 QoS 성능저하를 예방할 수 있다. 예를 들어, 컨테이너 리소스 관리부(M1-1)는 기설정된 컨테이너의 우선순위, 자원사용량 임계값 및 모니터링 감시 주기를 이용하여 전용코어에 할당되는 컨테이너를 설정할 수 있다.
- [0054] 컨테이너 정책 관리부(M1-3)는 상술한 컨테이너 중 전용코어에 할당되는 우선순위에 대한 정보를 포함할 수 있다.
- [0055] 디스플레이 제어부(M1-2)는 복수의 디스플레이 각각의 동작을 담당하는 컨테이너 들이 차량 내의 각 디스플레이에 화면을 출력하도록 관리할 수 있다.
- [0056] 컨테이너 파일 시스템 관리부(M1-4)는 복수의 컨테이너 환경에서 각 컨테이너의 동작에 필요한 파일들을 체계적이고 효율적으로 관리할 수 있다. 예컨대, 컨테이너 파일 시스템 관리부(M1-4)는 후술하는 바와 같이 어느 하나의 파일 시스템 예로 컨테이너의 파일 시스템을 다른 하나의 컨테이너 파일 시스템으로 오버레이할 수 있다. 이에, 사용자(복수의 탑승자)는 컨테이너와 같은 파일 시스템을 다른 사용자 또는 다른 위치(예로, 좌석)에서 용이하게 공유할 수 있다.
- [0057] 즉, 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너는 시스템 초기화 및 관리를 공유할 수 있다. 예를 들어, 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너는 rootfs를 공유할 수 있다. 그리고 각 운영체제 즉 컨테이너는 별도 변경 디렉토리를 저장 또는 복사(카피)하여 디스크 등의 자원이 효율적으로 이용될 수 있다. 나아가, 각 컨테이너 별 업그레이드도 변경 디렉토리만을 이용하여 수행될 수 있다. 예를 들어, 각 디스플레이 장치가 효과적으로 업그레이드될 수 있다.
- [0058] 컨테이너 스트리밍 관리부(M1-5)는 각 디스플레이에서 출력되는 영상의 이동 등을 관리할 수 있다. 즉, 컨테이너 스트리밍 관리부(M1-5)는 주행부와 인포테인먼트부 나아가, 디스플레이 각각의 정책에 따라 스트리밍 가능 여부를 판단할 수 있다.
- [0059] 오디오 정책 및 제어 관리부(M1-6)는 적어도 하나의 컨테이너 간의 오디오 출력에 대한 정책을 반영하여 오디오 출력을 제어할 수 있다. 이에 대한 자세한 설명은 후술한다.
- [0060] 컨테이너 업그레이드부(CM2)는 복수의 운영체제로 이루어진 인포테인먼트부의 각 운영체제 별로 업그레이드를 수행할 수 있다. 예컨대, 컨테이너 업그레이드부(CM2)는 컨테이너로 구성된 인포테인먼트부의 각 컨테이너를 용이하게 업그레이드할 수 있다.
- [0061] 컨테이너 업그레이드부(CM2)는 업데이트 및 확인 관리부(M2-1) 및 인증 관리부(M2-2)를 포함할 수 있다.
- [0062] 업데이트 및 확인 관리부(M2-1)는 복수의 컨테이너 중 업데이트가 필요한 컨테이너를 위한 컨테이너 별 식별자 및 업데이트 후 정상 동작 확인을 수행할 수 있다.
- [0063] 인증 관리부(M2-2)는 업데이트를 수행하기 위해 차량 및 컨테이너 별 인증, 버전 비교를 수행할 수 있다.
- [0064] 컨테이너 상태관리부(CM3)는 컨테이너의 정상 동작 여부에 대한 확인 등 복수의 컨테이너 상태를 관리할 수 있

다.

- [0065] 컨테이너 상태관리부(CM3)는 상태 관리부(M3-1), 로그 관리부(M3-2)를 포함할 수 있다.
- [0066] 상태 관리부(M3-1)는 주기적인 모니터링을 통해 컨테이너 및 컨테이너 관리부의 기능이 정상 동작하는지 확인할 수 있다.
- [0067] 로그 관리부(M3-2)는 컨테이너 상태관리부를 통해 확인된 내용을 로그를 보관할 수 있다. 나아가, 로그 관리부(M3-2)는 소정의 시간 후에 로그를 삭제하거나 백업 처리할 수 있다. 나아가, 컨테이너가 비정상 동작하는 경우 로그 관리부(M3-2)를 통해 상태 관리부(M3-1)가 해당 문제를 용이하게 해결할 수 있다.
- [0068] 웹 플랫폼 컨테이너 관리부(CM4)는 웹 플랫폼 컨테이너에 대한 관리를 수행할 수 있으며, 웹 플랫폼 관리부(M4-1), 웹 계정 관리부(M4-2)를 포함할 수 있다.
- [0069] 웹 플랫폼 관리부(M4-1)는 웹 플랫폼의 다양한 기능이 운영체제 즉, 컨테이너 상에서 정상 동작하는지 관리할 수 있다. 그리고 웹 플랫폼 관리부(M4-1)는 복수의 웹 플랫폼 컨테이너가 동작하는 경우 각 컨테이너 별로 로그인 계정을 관리하여 개인화된 웹 이용 환경을 관리할 수 있다. 이에 대한 자세한 설명은 후술한다.
- [0070] 나아가, 컨테이너 브리지부(CM5)는 인터페이스, 하드웨어 등과 연결되어 디스플레이, 오디오, 어플리케이션, 네트워크 등에 대한 각 운영체제 간의 상호 연결을 수행할 수 있다.
- [0071] 그리고 도면에 기재된 각 요소는 예시이며, 이하에서는 이를 기준으로 설명한다.
- [0072] 또한, 사용자(예로, 탑승자)는 사용자 인터페이스 등을 통해 차량 가상화 구조 기반의 시스템을 조작할 수 있다. 그리고 조작 과정 또는 결과가 디스플레이, 오디오 등의 출력 장치를 통해 사용자에게 제공될 수 있다. 또한, 사용자가 디스플레이 터치를 조작하는 등의 입력을 통해, 이에 대한 피드백이 이루어질 수 있다.
- [0073] 도 3은 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법에 대한 순서도이고, 도 4는 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법에서 업그레이드의 필요성 확인에 대한 구체적인 순서도이고, 도 5는 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법에 대한 구체적인 순서도이고, 도 6은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너의 구조에 대한 개념도이고, 도 7 및 도 8은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너를 구동을 설명하는 도면이다.
- [0074] 실시예에 따른 차량 가상화 구조 기반의 시스템 제어 방법은 복수의 컨테이너를 부팅하는 단계(S310), 복수의 컨테이너의 업그레이드의 필요성을 확인하는 단계(S320), 및 업그레이드의 필요성에 대응하여 복수의 컨테이너에 대한 업그레이드를 수행하는 단계(S330)를 포함할 수 있다.
- [0075] 이 때, 실시예에 따른 차량 가상화 구조 기반의 시스템에서 복수의 컨테이너는 상술한 바와 같이 시스템 초기화 및 관리를 공유할 수 있다.
- [0076] 예컨대, 복수의 컨테이너(Container1 내지 Container 3)는 공유된 시스템 초기화 및 관리인 rootfs를 공유할 수 있다. 즉, 복수의 컨테이너(Container1 내지 Container 3)는 커널(kernel), 디스트로(distro) 및 라이브러리(library)를 서로 공유할 수 있다.
- [0077] 이 때, 차량 가상화 구조 기반의 시스템 제어 방법은 상술한 컨테이너 관리부(CM)에서 수행될 수 있다. 즉, 차량 가상화 구조 기반의 시스템 제어 장치는 컨테이너 관리부(CM)를 포함할 수 있다. 나아가, 차량 가상화 구조 기반의 시스템 제어 장치는 컨테이너 제어부(CM1), 컨테이너 업그레이드부(CM2)를 포함할 수 있으며, 후술하는 각 제어 단계를 수행할 수 있다. 이하에서 각 단계가 수행되는 장치에 대해서 간략하게 기재한다.
- [0078] 실시예로, 컨테이너 제어부는 복수의 컨테이너를 부팅할 수 있다(S310). 그리고 컨테이너 업그레이드부(CM2)는 운영체제 즉 부팅된 복수의 컨테이너에 대한 업그레이드의 필요성을 판단할 수 있다(S320)
- [0079] 구체적으로, 복수의 컨테이너의 업그레이드의 필요성을 확인하는 단계는 업그레이드 정보를 수신하는 단계(S321), 업그레이드 정보와 복수의 컨테이너의 업그레이드를 비교하는 단계(S322), 업그레이드의 필요성을 결정하는 단계(S323)을 포함할 수 있다.
- [0080] 컨테이너 업그레이드부(CM2)는 서버 등으로부터 업데이트 정보를 수신할 수 있다. 업데이트 정보는 해당 컨테이너의 버전 정보를 포함할 수 있다.
- [0081] 그리고 컨테이너 관리부(CM) 또는 컨테이너 업그레이드부(CM2)는 수신한 업그레이드와 복수의 컨테이너의 업그레이드 즉, 업그레이드 정보와 비교할 수 있다. 이에, 수신한 업그레이드 정보와 대응하는 컨테이너의 업그레이

드 정보를 비교하여, 컨테이너의 업그레이드 정보가 수신한 업그레이드 정보보다 최신이 아닌지 판단할 수 있다.

- [0082] 예컨대, 컨테이너 관리부(CM) 또는 컨테이너 업그레이드부(CM2)는 수신한 업그레이드 정보와 컨테이너의 업그레이드 정보가 동일한 버전이거나, 컨테이너의 업그레이드 정보가 최신인 경우에 업그레이드의 필요성이 없다고 판단할 수 있다. 또한, 수신한 업그레이드의 정보가 컨테이너의 업그레이드 정보보다 낮은 버전인 경우에 업그레이드의 필요성이 있다고 판단할 수 있다. 즉, 복수의 컨테이너의 업그레이드의 필요성 존재 유무를 판단할 수 있다(S334).
- [0083] 그리고 컨테이너 관리부(CM)는 업그레이드의 필요성에 대응하여 복수의 컨테이너에 대한 업그레이드를 수행할 수 있다(S330). 업그레이드의 필요성이 존재하는 경우에만, 컨테이너 관리부(CM)는 복수의 컨테이너에 대한 업그레이드를 수행할 수 있다. 그리고 업그레이드의 필요성이 존재하지 않는 경우에, 컨테이너 제어부(CM1)는 복수의 컨테이너를 구동할 수 있다(S340). 즉 각 디스플레이 별 사용자 데이터(personal Data)를 로딩하여 컨테이너를 구동할 수 있다. 이 때, 상술한 바와 같이, 컨테이너 관리부(CM) 또는 컨테이너 제어부(CM1)는 차량 내 각 디스플레이별 컨테이너에서 시스템 초기화 및 관리인 rootfs를 공유하며, 각 컨테이너(예로, 디스플레이 별)가 사용자 데이터를 공유된 rootfs에 오버레이하므로 디스크 자원을 절약할 수 있다. 이에, 효율적인 컨테이너 운영 또는 구동이 이루어질 수 있다. 나아가, 컨테이너는 사용자 데이터를 복사하여(copied) 구동될 수 있다. 실시예로, 사용자 데이터 중 변경 디렉토리 카피하여 복수의 컨테이너를 구동 또는 수행할 수 있다. 즉, 최신 버전의 사용자 데이터(delta0/)가 새로운 사용자 데이터(delta1/)로 복사(copied)될 수 있다. 이에, 기존 컨테이너(original)의 구동 이후에 새로운 컨테이너(snap clone)의 구동이 용이하게 이루어질 수 있다. 예컨대, 복수개의 RSE 디스플레이에 대한 컨테이너는 구동이 보다 용이하게 이루어질 수 있다. 또한, 기존 최신 버전의 사용자 데이터 또는 구동된 사용자 데이터(new files)가 상술한 바와 같이 공유된 rootfs(root files)에 오버레이될 수 있다(mount). 이로써, 컨테이너의 구동 및 업데이트(시스템 파일 및 사용자 데이터)가 모두 용이하게 이루어질 수 있다.
- [0084] 그리고 컨테이너 관리부(CM) 또는 컨테이너 제어부(CM1)는 컨테이너와 연결된 디스플레이로 어플리케이션을 실행할 수 있다(S350). 이에, 차량 내 각 탑승자에 위치한 디스플레이 별 각 서비스(예로, 어플리케이션 실행)가 탑승자에게 제공될 수 있다.
- [0085] 그리고 컨테이너 관리부(CM) 또는 컨테이너 제어부(CM1)는 업그레이드의 필요성이 존재하는 경우에 시스템 디렉터리에 대한 업그레이드 또는 사용자 데이터(personal data)에 대한 업그레이드인지 판단할 수 있다(S360). 이는 상술한 업그레이드의 필요성에 대응하여 복수의 컨테이너에 대한 업그레이드를 수행하는 단계(S330)가 구체화되는 단계일 수 있다. 즉, 컨테이너에 대한 업그레이드의 수행이 이하와 같이 구체화되어 이루어질 수 있다.
- [0086] 시스템 디렉터리에 대한 업그레이드인 경우, 컨테이너 관리부(CM)는 업그레이드인지 판단한(S360) 이후에, 사용자 인증을 수행하고(S370), 시스템 디렉터리를 수신할 수 있다(S380).
- [0087] 컨테이너 관리부(CM) 또는 컨테이너 업그레이드부(CM2)는 사용자 인증을 수행할 수 있다. 사용자 인증은 차량 및 컨테이너에 대한 인증을 포함할 수 있다. 즉, 차량의 종류와 컨테이너의 종류를 판단한 뒤, 해당 차량에 대응하는 컨테이너에 대한 인증을 수행할 수 있다(S370).
- [0088] 그리고 컨테이너 관리부(CM) 또는 컨테이너 업그레이드부(CM2)는 서버로부터 최신 버전의 파일 시스템(file system)을 수신할 수 있다. 예컨대, 파일 시스템으로 커널, OS, 라이브러리가 존재한다. 수신된 파일 시스템이 공유된 rootfs에 적용될 수 있다. 즉, 실시예에 따른 차량 가상화 구조 기반의 제어 방법 및 장치는 컨테이너 관리부에 의해(또는 컨테이너 관리부를 포함하여) 동일 가상 엔진 상에 동일 컨테이너 종류에 대한 시스템 디렉터리 업그레이드를 용이하게 수행할 수 있다. 즉, 1번의 시스템 디렉터리 수신으로 복수의 컨테이너에 대한 시스템 디렉터리 업그레이드가 이루어질 수 있다. 이후에, 복수의 컨테이너를 재부팅할 수 있다(S390).
- [0089] 이후에, 업그레이드의 필요성을 확인하는 단계(S320)으로 돌아가 상술한 단계 또는 기능을 수행할 수 있다.
- [0090] 사용자 데이터(personal data)에 대한 업그레이드인 경우 업그레이드인지 판단하는 단계(S360) 이후에 복수의 컨테이너를 구동할 수 있다(S340). 다만 이전에 최신의 사용자 데이터를 수신(S365)할 수 있다. 여기서, 최신은 보전이 상위이거나 수정된 날짜가 가장 가까운 경우를 의미한다.
- [0091] 최신의 사용자 데이터를 각 컨테이너에 적용할 수 있다. 이 때, 상술한 바와 같이, 공유된 rootfs에 사용자 데이터를 오버레이하여 사용자 데이터를 로딩하고 개별 컨테이너를 구동할 수 있다. 이에, 디스크 자원에 대한 사

용을 최소화할 수 있다.

- [0092] 나아가, 컨테이너 관리부(CM)는 컨테이너와 연결된 디스플레이로 어플리케이션을 실행할 수 있다(S350). 즉, 어플리케이션이 각 컨테이너 상에 구동되고 각 디스플레이를 통해 출력될 수 있다. 이에, 차량 내 각 탑승자에 위치한 디스플레이 별 각 서비스(예로, 어플리케이션 실행)가 탑승자에게 제공될 수 있다.
- [0093] 도 9는 도 2에서 K부분의 확대도이고, 도 10은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 웹 플랫폼 제어 방법에 대한 순서도이고, 도 11은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 웹 플랫폼 제어 방법에서 웹 어플리케이션의 구동을 도시한 도면이다.
- [0094] 도 9 내지 도 11을 참조하면, 상술한 바와 같이 실시예에 따른 차량 가상화 구조 기반의 시스템은 베이스 운영체제(10) 및 추가 가상 엔진(20) 상에서 탑재 또는 구동되는 웹 플랫폼부(Web Platform)을 포함할 수 있다. 인포테인먼트부(IP)와 마찬가지로 플랫폼부(Web Platform)는 컨테이너 관리자를 인포테인먼트부(IP)와 공유할 수 있다. 또는 플랫폼부(Web Platform)는 인포테인먼트부(IP)의 일부로 구성될 수 있다. 이에 대해서는 언급한 바와 같이 개별적인 구성으로 설명한다.
- [0095] 또한, 플랫폼부(Web Platform)는 인포테인먼트부(IP)와 같이 컨테이너 관리부에 탑재된 하나의 운영체제(예로, 컨테이너, 30), 운영체제 상에 탑재되고 미들웨어 및 중요 애플리케이션이 포함된 소프트웨어 집합으로 복수의 운영체제 플랫폼(예로, AGL, 40), 운영체제 플랫폼(40)에서 구동되는 복수의 애플리케이션(예로, 웹어플리케이션, 50)을 포함할 수 있다.
- [0096] 나아가, 가상 엔진(20)에는 차량 정보 서비스 서버(Vehicle Information Service Sever, VIS server)를 포함할 수 있다. 차량 정보 서비스 서버(Vehicle Information Service Sever, VIS server)는 차량 구동부 정보(속도, 연료량 등) 및 차량 설정 정보(온도, 시트위치 등)를 저장하고 각 웹 플랫폼 컨테이너 즉 운영체제 플랫폼(40)에 전송할 수 있다. 즉, 차량 정보 서비스 서버(Vehicle Information Service Sever, VIS server)는 각 웹 운영체제 플랫폼(40) 내의 차량 정보 서비스 클라이언트(Vehicle Information Service Sever, VIS Client)로 요청된 또는 탑승자에 대응한 차량 구동부 정보 및 차량 설정 정보를 제공할 수 있다.
- [0097] 그리고 컨테이너 관리자(CM)는 상술한 바와 같이 웹 플랫폼 컨테이너 관리부(CM4)를 포함할 수 있다. 또한, 웹 플랫폼 컨테이너 관리부(CM4)는 웹 플랫폼 컨테이너에 대한 관리를 수행할 수 있으며, 웹 플랫폼 관리부, 웹 계정 관리부를 포함할 수 있다.
- [0098] 웹 플랫폼 관리부는 웹 플랫폼의 다양한 기능이 운영체제 즉, 컨테이너 상에서 정상 동작하는지 관리할 수 있다. 그리고 웹 플랫폼 관리부는 복수의 웹 플랫폼 컨테이너가 동작하는 경우 각 컨테이너 별로 로그인 계정을 관리하여 개인화된 웹 이용 환경을 관리할 수 있다.
- [0099] 운영체제 플랫폼(40)은 AGL로, 요청된 또는 탑승자에 대응한 차량 구동부 정보 및 차량 설정 정보를 제공받아 차량 환경을 설정하는 서비스 클라이언트(Vehicle Information Service Sever, VIS Client)를 포함할 수 있다.
- [0100] 나아가, 운영체제 플랫폼(40)은 사용자 인식 및 등록을 위한 API 서비스부(Device API service), 운영체제 플랫폼에 포함된 웹 어플리케이션을 관리하는 기능을 확장하기 위해 복수 개의 웹 플랫폼 컨테이너(운영체제, 30)가 구동 시 GPS 등의 하드웨어 자원을 복수의 웹 어플리케이션이 이용할 수 있는지 여부를 모니터링하고 관리하는 애드온부(WAM add on), 및 웹 어플리케이션 이용 시 텍스트 입력이 필요한 경우를 위한 가상키보드부(Virtual Keyboard)를 포함할 수 있다.
- [0101] 보다 구체적으로, 실시예에 따른 차량 가상화 구조 기반의 제어 방법은 웹 플랫폼 제어와 관련하여 사용자를 인식하는 단계(S405), 인식된 사용자가 기등록된 사용자인지 확인하는 단계(S410)를 포함할 수 있다.
- [0102] 즉, 운영체제 플랫폼(40) 또는 API 서비스부(Device API service)는 사용자를 인식할 수 있다(S405). 다시 말해, 차량 내 각 탑승자에 대한 개별적인 인식이 이루어질 수 있다. 이는 다양한 장치(센서, 모바일, 어플리케이션 등)를 통해 사용자 인식이 수행될 수 있다. 또한, 사용자를 인식하는 단계(S405)는 전술한 업데이트 및 컨테이너 구동 이후에 수행될 수 있다. 예를 들어, 사용자를 인식하는 단계(S405)는 어플리케이션이 각 컨테이너 상에서 구동되고 각 디스플레이로 출력된(S350) 이후에 수행될 수 있다. 이 때, 차량 내 탑승자의 사용자가 웹 플랫폼 컨테이너를 구동하고 사용자 인식이 수행될 수 있다.
- [0103] 운영체제 플랫폼(40) 또는 API 서비스부(Device API service)는 인식된 사용자가 기등록된 사용자인지 확인할 수 있다(S410). 운영체제 플랫폼(40) 또는 API 서비스부(Device API service)는 기존에 등록된 사용자가 아닌 경우에 사용자 등록을 진행하고(S415), 기존에 등록된 사용자인 경우에 차량 환경을 설정할 수 있다(S420). 환

경은 시트 포지션, 선호 온도, 조명광 상태 등을 포함할 수 있다.

- [0104] 특히, 사용자 인식은 컨테이너 관리부(CM) 중 웹 플랫폼 컨테이너 관리부(CM4)과 연계하여 이루어질 수 있다. 웹 플랫폼 컨테이너 관리부(CM4)는 웹 계정 등록, 변경, 삭제 등을 관리하고, 웹 계정이 등록된 계정인 확인할 수 있으며, 웹 계정 별 어플리케이션 리스트 실행 및 관리를 수행하거나, 웹 계정 로그인이 중복으로 일어나지 않도록 로그인 상태를 관리할 수 있다. 즉, 운영체제 플랫폼(40) 또는 API 서비스부(Device API service)으로부터 제공된 사용자 정보가 컨테이너 관리부(CM)에서 확인되고 계정 별 어플리케이션 리스트 실행 및 관리나 중복 로그인 관리가 이루어질 수 있다.
- [0105] 그리고 컨테이너 관리부(CM) 중 웹 플랫폼 컨테이너 관리부(CM4)는 인식된 사용자 즉, 사용자 정보에 대한 웹 계정 등록 여부를 확인할 수 있다(S425).
- [0106] 이에, 컨테이너 관리부(CM) 중 웹 플랫폼 컨테이너 관리부(CM4)는 해당 사용자에게 웹 계정이 등록되지 않았거나 없는 경우 웹 계정을 등록하고(S430) 저장하며, 웹 계정이 등록된 경우에 웹 계정이 다른 웹 플랫폼 컨테이너에 로그인되었는지 확인할 수 있다(S435).
- [0107] 이에, 컨테이너 관리부(CM) 중 웹 플랫폼 컨테이너 관리부(CM4)는 웹 계정이 다른 웹 플랫폼 컨테이너에 로그인된 경우 종료 또는 재로그인을 출력하나(S440), 해당 웹 계정이 로그인된 상태가 아니라면 웹 계정 별 어플리케이션을 추출(S445)할 수 있다. 그리고 추출된 웹 계정 별 선호 어플리케이션을 사용자에게 제공할 수 있다(도 11 참조). 이에, 사용자는 웹 계정 별 선호하는 또는 실시간 웹 어플리케이션의 구동에 대해 용이하게 인식할 수 있다.
- [0108] 그리고 컨테이너 관리부(CM) 중 웹 플랫폼 컨테이너 관리부(CM4)는 네비게이션 어플리케이션을 구동하고 목적지를 디스플레이에 표시할 수 있다(S450). 그리고 사용자는 가상키보드부(Virtual Keyboard)를 통해 목적지 입력할 수 있다(S455). 이와 같이, 실시예에 따른 차량 가상화 구조 기반의 제어 방법 및 장치에서 웹 플랫폼 제어로(방법 또는 장치), 사용자 인식에 따라 웹 플랫폼 어플리케이션을 용이하게 사용하며, 각 운영체제(웹 플랫폼 컨테이너)에 동일 어플리케이션을 상이한 사용자 또는 탑승자 웹 계정으로 구동할 수 있다. 이로써, 차량 내 탑승자 간의 개별 계정을 이용하여 동일 어플리케이션을 구동 및 실행할 수 있다.
- [0109] 도 12는 다른 실시예에 따른 차량 가상화 구조 기반의 시스템의 개념도이고, 도 13은 실시예에 따른 차량 가상화 구조 기반의 시스템에서 디스플레이 장치를 설명하는 도면이다.
- [0110] 도 12 및 도 13을 참조하면, 상술한 바와 같이 실시예에 따른 차량 가상화 구조 기반의 시스템은 하드웨어(미도시됨), 하드웨어(미도시됨) 상에 탑재된 베이스 운영 체제(10), 베이스 운영 체제(10) 상에 탑재된 제1 가상엔진(20a)과 제2 가상엔진(20b), 각 가상엔진 상에 탑재되는 적어도 하나의 운영체제(30a 내지 30f), 적어도 하나의 운영체제(30a 내지 30f) 상에 탑재되고 미들웨어 및 중요 애플리케이션이 포함된 소프트웨어 집합으로 복수의 운영체제 플랫폼(40a 내지 40f), 운영체제 플랫폼(40a 내지 40f)에서 구동되는 복수의 애플리케이션(50a 내지 50d), 복수의 운영체제 플랫폼(40a 내지 40f) 또는 복수의 애플리케이션 등을 출력하는 복수의 디스플레이(60a 내지 60f) 및 가상엔진(20a 내지 20b) 상에서 복수의 운영체제(30a 내지 30d)에 대한 환경을 관리하는 컨테이너 관리부(CM)를 포함할 수 있다. 적어도 하나의 운영체제는 이하 '컨테이너(container)'로 설명한다.
- [0111] 여기서, 하드웨어(미도시됨)는 프로세서, 표시부, 저장부, 메모리부, 컨트롤부, I/O 장치를 포함하는 개념일 수 있다.
- [0112] 실시예에서, 주행부(CP)는 제1 가상엔진(20a) 상에서 탑재 또는 구동되는 시스템 영역이고, 인포테인먼트부(IP, Infotainment Platform)는 제1 가상엔진(20a)과 이종의 가상엔진인 제2 가상엔진(20b) 상에서 탑재 또는 구동되는 시스템 영역일 수 있다. 이러한 구성에 의하여, 인포테인먼트부(IP) 내에서 업그레이드 또는 기타 오작동이 발생하더라도, 주행부(CP)는 이에 대한 영향을 받지 않을 수 있다. 즉, 주행 중 영향이 최소화될 수 있다. 나아가, 인포테인먼트부(IP) 이외에 웹 플랫폼부(Web Platform)이 베이스 운영 체제(10) 및 추가 가상 엔진(20) 상에서 탑재 또는 구동될 수 있다. 인포테인먼트부(IP)와 마찬가지로 플랫폼부(Web Platform)는 컨테이너 관리자를 인포테인먼트부(IP)와 공유할 수 있다. 또는 플랫폼부(Web Platform)는 인포테인먼트부(IP)의 일부로 구성될 수 있다. 이하에서는 개별적인 구성으로 설명한다.
- [0113] 플랫폼부(Web Platform)는 인포테인먼트부(IP)와 같이 컨테이너 관리부에 탑재된 하나의 운영체제(예로, 컨테이너, 30), 운영체제 상에 탑재되고 미들웨어 및 중요 애플리케이션이 포함된 소프트웨어 집합으로 복수의 운영체제 플랫폼(예로, AGL, 40), 운영체제 플랫폼(40)에서 구동되는 복수의 애플리케이션(예로, 웹어플리케이션, 5

0)을 포함할 수 있다.

- [0114] 먼저, 베이스 운영 체제(10)는 예를 들어 다양한 운영체제일 수 있다. 예컨대, 베이스 운영 체제(10)는 리눅스(Linux), 하이퍼바이저(hypervisor), QNX, GENIVI 등을 포함할 수 있다.
- [0115] 예를 들어, 제1 가상엔진(20a)과 제2 가상엔진(20b)은 미들웨어 솔루션 또는 다양한 플랫폼으로서, 모바일 C언어로 개발될 수 있다. 제1 가상엔진(20a)과 제2 가상엔진(20b)은 다수의 내장 라이브러리를 제공할 수 있으며, 다양한 이동 단말기 등에서 동일한 동작을 수행할 수도 있다. 예컨대, 제1 가상엔진(20a)과 제2 가상엔진(20b)은 리눅스 기반의 안드로이드 커널일 수 있으며, 메모리 보호, 가상 메모리 모듈 및 스케줄 캐싱을 초기화할 수도 있다.
- [0116] 또한, 적어도 하나의 컨테이너(30a 내지 30d)는 제2 가상엔진(20a) 상에 탑재되고 제1 가상엔진(10a) 상에 탑재되지 않을 수 있다. 이에, 주행부(CP)에 대한 보안성이 향상되므로, 주행 안정성도 개선될 수 있다.
- [0117] 적어도 하나의 컨테이너(30a 내지 30d)는 실시예로 가상화 방식의 일 형태로서 프로세스 가상화의 일정일 수 있다. 예컨대, 컨테이너를 이용한 가상화 기술은 호스트 OS(operating system) 내부를 물리적 자원을 관리하는 커널 공간과 사용자 프로세스, 즉 응용 프로그램(어플리케이션, application, APP)을 실행하는 사용자 공간으로 구분하고 사용자 공간을 여러 개로 나누어 각각의 사용자 프로세스에서 사용되는 하드웨어 자원을 할당하고 공유하는 기술을 의미할 수 있다. 예컨대, 컨테이너(30a 내지 30d)는 호출된 라이브러리를 시스템 라이브러리와 인터페이스되도록 변환하여, 베이스 운영체제와 복수의 운영체제 플랫폼 간의 연결 또는 호환을 수행할 수 있다.
- [0118] 복수의 운영체제 플랫폼(40a 내지 40f)은 컨테이너 상에 탑재될 수 있다. 복수의 운영체제 플랫폼(40a 내지 40f)은 안드로이드, AGL(Automotive Grade Linux), 웹(web)플랫폼, 클러스터(cluster) 플랫폼, 헤드업 디스플레이(HUD) 플랫폼 등을 포함할 수 있다. 나아가, 운영체제 플랫폼(40) 내에 복수 개의 사운드 클라이언트가 배치될 수 있다. 복수 개의 사운드 클라이언트는 제1 사운드 클라이언트 및 제2 사운드 클라이언트를 포함할 수 있다. 이 때, 제1 사운드 클라이언트는 제1 가상엔진(20a) 상의 운영체제 플랫폼(40a, 40b)에 탑재될 수 있다. 제2 사운드 클라이언트는 제2 가상엔진(20b) 상의 운영체제 플랫폼(40c 등)에 탑재될 수 있다. 즉, 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재될 수 있다. 그리고 제1 사운드 클라이언트는 클러스터 운영체제에 탑재(40a) 또는 HUD 운영체제(40b)에 탑재될 수 있다. 그리고 제2 사운드 클라이언트는 AVN(Audio Video Navigation) 운영체제, 코-드라이버(Co-Driver) 운영체제, RSE(Rear Seat Entertainment) 운영체제에 탑재될 수 있다.
- [0119] 나아가, 사운드 서버는 제1 사운드 클라이언트(SC1) 및 제2 사운드 클라이언트(SC2)로부터 각각 제1 사운드 출력정보 및 제2 사운드 출력정보를 수신할 수 있다. 사운드 서버는 가상엔진(20a, 20b)에 또는 베이스 운영체제(10) 내에 탑재될 수 있다. 이하에서는 사운드 서버는 제1 사운드 서버(SS1), 제2 사운드 서버(SS2)를 포함하고, 제1 사운드 서버(SS1) 및 제2 사운드 서버(SS2)는 이종의 가상엔진에 탑재되는 것을 기준으로 설명한다.
- [0120] 적어도 하나의 애플리케이션(50a 내지 50d)은 시스템 프로그램을 제외한 응용 프로그램으로써, 차량 내에서는 AVN(Audio Video Navigation), 코-드라이버(Co-Driver), RSE(Rear Seat Entertainment) 등을 포함할 수 있다.
- [0121] 복수의 디스플레이(60a 내지 60f)는 운영체제 플랫폼(40a 내지 40f) 또는 플랫폼 상의 응용 프로그램 등을 사용자(예로, 탑승자)에게 표시할 수 있다. 예컨대, 복수의 디스플레이(60a 내지 60f)는 다양한 표시 장치(예로, OLED)를 포함할 수 있다.
- [0122] 이하에 대한 설명은 상술한 오디오 정책 및 제어 관리부에 대한 내용일 수 있다.
- [0123] 또한, 사용자(예로, 탑승자)는 사용자 인터페이스 등을 통해 차량 가상화 구조 기반의 시스템을 조작할 수 있다. 그리고 조작 과정 또는 결과가 디스플레이, 오디오 등의 출력 장치를 통해 사용자에게 제공될 수 있다.
- [0124] 도 14는 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 장치의 블록도이고, 도 15는 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 방법의 순서도이고, 도 16은 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 방법의 구체화된 순서도이고, 도 17은 일예에 대한 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력을 설명하는 도면이고, 도 18은 다른예에 대한 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력을 설명하는 도면이고, 도 19는 또 다른예에 대한 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력을 설명하는 도면이다.
- [0125] 도 14를 참조하면, 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 장치(100)는 수신부(110), 비교부

(120), 판단부(130) 및 송신부(140)를 포함할 수 있다. 이 때, 차량 가상화 구조 기반의 사운드 제어 장치(100)는 오디오 정책 및 제어 관리부를 포함하며, 오디오 정책 및 제어 관리부는 수신부(110), 비교부(120), 판단부(130) 및 송신부(140)에 대응할 수 있다.

- [0126] 먼저, 수신부(110)는 각 사운드 클라이언트로부터 사운드 출력정보를 수신할 수 있다. 예컨대, 수신부(110)는 제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신할 수 있다. 그리고 사운드 출력정보는 사용자의 입력 등에 의해 발생할 수 있다. 즉, 사운드 출력정보는 사용자 입력에 의해 어플리케이션이 수행되면서 출력되는 사운드 정보를 포함할 수 있다.
- [0127] 비교부(120)는 수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교할 수 있다. 제1 사운드 출력정보의 우선순위는 상기 제2 사운드 출력정보의 우선순위보다 높을 수 있다. 이에, 주행부로부터 수신한 사운드 출력 정보가 인포테인먼트부로부터 수신한 사운드 출력 정보보다 우선순위가 높으므로, 주행에 대한 안정성을 향상시킬 수 있다.
- [0128] 판단부(130)는 상기 제1 사운드 출력정보의 발생 시점이 차량의 주행중인지 여부를 판단할 수 있다. 이 때, 판단부(130)는 제1 사운드 출력정보를 출력한 후 제2 사운드 출력정보를 출력하도록 판단할 수 있다.
- [0129] 그리고 판단부(130)는 제1 사운드 출력정보의 발생 시점이 차량의 주행중이라면 제1 사운드 출력정보의 레벨에 따라 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력 여부 및 출력 시간을 조절하여 출력하도록 판단할 수 있다.
- [0130] 구체적으로, 판단부(130)는 제1 사운드 출력정보의 레벨이 제1 레벨이면 상기 제1 사운드 출력정보만 출력하고, 상기 제2 사운드 출력정보의 출력을 차단하도록 판단할 수 있다. 이 때, 출력정보의 레벨은 제1 레벨 및 상기 제1 레벨보다 낮은 제2 레벨을 포함할 수 있다.
- [0131] 그리고 판단부(130)는 제1 사운드 출력정보의 레벨이 제2 레벨이면, 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보를 제1 시간 동안 출력할 수 있다. 또한, 판단부(130)는 제2 사운드 출력정보를 제1 시간에 연속하는 제2 시간 동안 출력하도록 판단할 수 있다. 이 때, 제1 시간 동안 상기 제1 사운드 출력정보의 크기는 상기 제2 사운드 출력정보의 크기보다 클 수 있다. 이러한 구성에 의하여, 차량 주행 중의 안정성 향상이 이루어질 수 있다.
- [0132] 또한, 가상 엔진(가상화 환경) 또는 호스트 운영체제에 탑재된(또는 임베디드된) 운영체제(예로, 게스트 운영체제 등) 내의 사운드 클라이언트는 각 운영체제 등과 연결된 디스플레이 장치 등으로 사운드 출력정보를 수신 또는 요청할 수 있다. 사운드 출력정보의 수신 및 요청은 출력이 요구되는 사운드 데이터를 의미할 수 있다.
- [0133] 그리고 가상 엔진 또는 컨테이너 또는 베이스 운영 체제 내에 탑재된 사운드 서버는 데이터 즉 사운드 출력정보를 수신하여 출력 인터페이스인 스피커 등을 통해 사운드를 출력하도록 하드웨어로 사운드 출력 정보를 수신할 수 있다. 이 때, 사운드 출력정보 즉 데이터는 가상 엔진 또는 호스트 운영체제에 탑재된(또는 임베디드된) 운영체제(예로, 게스트 운영체제 등) 내의 사운드 클라이언트를 통해 디스플레이 장치로 전달되어 사용자에게 제공될 수 있다.
- [0134] 또한, 송신부(140)는 최종 판단된 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어 등으로 송신할 수 있다. 이를 통해, 차량의 전면 또는 후면 사운드 출력장치(예로, 스피커)를 통해 사운드가 출력될 수 있다.
- [0135] 도 15 내지 도 19를 참조하면, 실시예에 따른 차량 가상화 구조 기반의 사운드 제어 방법은 제1 사운드 클라이언트로부터 제1 사운드 출력정보 및 제2 사운드 클라이언트로부터 제2 사운드 출력정보 중 적어도 하나를 수신하는 단계(S510), 수신된 상기 제1 사운드 출력정보 및 상기 제2 사운드 출력정보에 대한 우선순위를 비교하는 단계(S520) 및 비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보 중 적어도 하나를 하드웨어로 송신하는 단계(S530)을 포함할 수 있다. 이하에서, 각 단계는 상술한 차량 가상화 구조 기반의 사운드 제어 장치에 의해 수행될 수 있다.
- [0136] 그리고 제1 사운드 클라이언트 및 상기 제2 사운드 클라이언트는 서로 다른 가상엔진 상에 탑재되며, 제1 사운드 클라이언트는 클러스터 운영체제에 탑재되고, 상기 제2 사운드 클라이언트는 AVN(Audio Video Navigation) 운영체제, 코-드라이버(Co-Driver) 운영체제, RSE(Rear Seat Entertainment) 운영체제에 탑재될 수 있다.
- [0137] 또한, 제1 사운드 출력정보의 우선순위는 상기 제2 사운드 출력정보의 우선순위보다 높을 수 있다.

- [0138] 그리고 우선순위를 비교한(S520) 이후에, 제1 사운드 출력정보의 발생 시점이 차량의 주행중인지 여부를 확인할 수 있다(S540).
- [0139] 실시예로, 제1 사운드 출력정보의 발생 시점이 차량의 주행중이 아니라면, 제1 사운드 출력정보를 출력한 후 제2 사운드 출력정보를 출력할 수 있다(S550). 예컨대, 주행이 시작되기 전에 인포테인먼트부의 오디오 재생 전 구동부의 알람음의 재생이 필요할 수 있다. 이 경우, 도 17과 같이 구동부의 알람(제1 사운드 출력정보)을 특정 시간( $t_a$ )까지 먼저 출력하고, 인포테인먼트부의 오디오(제2 사운드 출력 정보)가 특정 시간( $t_a$ )이후에 출력될 수 있다. 즉, 시동을 켜 후, AVN의 라디오 등 미디어 재생 전에 연료 부족 알람, 소모품 교체 알람 등의 알람음이 우선적으로 출력(재생)될 수 있다. 다시 말해, 비교 결과에 대응하여 제1 사운드 출력정보 및 제2 사운드 출력정보(차량 주행중에는 제1 사운드 출력정보 송신 후 제2 사운드 출력 정보 송신) 중 적어도 하나를 송신할 수 있다.
- [0140] 그리고 제1 사운드 출력정보의 발생 시점이 차량의 주행중이라면 상기 제1 사운드 출력정보의 레벨에 따라 상기 제1 사운드 출력정보 및 제2 사운드 출력정보의 출력 여부 및 출력 시간을 조절하여 출력할 수 있다(S570). 여기서, 출력정보의 레벨은 제1 레벨 및 상기 제1 레벨보다 낮은 제2 레벨을 포함할 수 있다.
- [0141] 그리고 제1 사운드 출력정보의 레벨이 제1 레벨인지 판단할 수 있다. 제1 사운드 출력정보의 레벨이 제1 레벨이면 제1 사운드 출력정보만 출력하고, 제2 사운드 출력정보의 출력을 차단할 수 있다(S580).
- [0142] 예컨대, 차량의 주행중에 구동부의 오디오(또는 사운드)를 출력하고 인포테인먼트부의 오디오를 음소거(mute)할 수 있다. 이를 통해, 운전자가 위험 상태를 완전하게 인지하게 되어 사고 발생이 억제될 수 있다. 즉, 도 18과 같이, 제1 사운드 출력정보만이 출력되고 제2 사운드 출력정보의 출력이 차단될 수 있다.
- [0143] 예컨대, 주행 중 차량 구동부 고장, 졸음 방지 경고음 등(제1 사운드 출력 정보)이 인포테인먼트부의 오디오(제2 사운드 출력정보)가 차단된 상태에서 출력되어야 하므로, 제1 사운드 출력정보만이 하드웨어로 송신될 수 있다.
- [0144] 그리고 제1 사운드 출력정보의 레벨이 제2 레벨이면, 제1 사운드 출력정보 및 상기 제2 사운드 출력정보를 제1 시간 동안 출력하고(S590), 제2 사운드 출력정보를 제1 시간에 연속하는 제2 시간 동안 출력할 수 있다(S595).
- [0145] 도 19를 참조하면, 제1 시간( $t_1$ )까지 제1 사운드 출력정보 및 제2 사운드 출력 정보를 출력할 수 있다. 이 때, 제1 시간( $t_1$ ) 동안 상기 제1 사운드 출력정보의 크기(AM1)는 상기 제2 사운드 출력정보의 크기(AM2)보다 클 수 있다. 이에, 운전자는 제1 사운드 출력정보의 오디오를 용이하게 인지할 수 있다.
- [0146] 이로써, 주행 중 차선 이탈 경보음, 소모품 교체 알람 등의 순간적 주의를 요구하는 제1 사운드 출력정보가 운전자에게 용이하게 제공될 수 있다.
- [0147] 나아가, 제2 사운드 출력정보만을 수신하면 상기 송신하는 단계에서 상기 제2 사운드 출력정보를 상기 하드웨어로 송신할 수 있다.
- [0148] 이 때, 디스플레이 장치들이 오디오 데이터를 출력하는 경우, 이들 장치 간의 우선순위가 존재할 수 있다. 예컨대, AVN, RSE, Co-drive, 순으로 우선순위가 설정될 수 있다.
- [0149] 예를 들어, AVN만 오디오 출력이 이루어지는 경우, 차량의 모든 사운드 장치(이하, 스피커)가 해당 오디오를 출력할 수 있다. 그리고 AVN과 RSE가 동시에 오디오 출력을 수행하는 경우, 차량의 전면부 상에 오디오가 출력되는 경우, 차량의 전면부 스피커는 AVN의 오디오 출력이 출력되고, 차량의 후면부 스피커는 RSE의 오디오 출력이 출력될 수 있다. 또한, AVN과 Co-drive가 동시에 오디오 출력을 수행하는 경우, AVN과 Co-drive 중 최근에 출력된 오디오 출력이 차량의 모든 스피커로 출력될 수 있다.
- [0150] 도 20은 변형예에 따른 차량 가상화 구조 기반의 시스템의 개념도이다.
- [0151] 상술한 바와 같이 실시예에 따른 차량 가상화 구조 기반의 시스템은 하드웨어(미도시됨), 하드웨어(미도시됨) 상에 탑재된 베이스 운영 체제(10), 베이스 운영 체제(10) 상에 탑재된 적어도 하나의 운영체제(30a 내지 30f), 적어도 하나의 운영체제(30a 내지 30f) 상에 탑재되고 미들웨어 및 중요 애플리케이션이 포함된 소프트웨어 집합으로 복수의 운영체제 플랫폼(40a 내지 40f), 운영체제 플랫폼(40a 내지 40f)에서 구동되는 복수의 애플리케이션(50a 내지 50d), 복수의 운영체제 플랫폼(40a 내지 40f) 또는 복수의 애플리케이션 등을 출력하는 복수의 디스플레이(60a 내지 60f) 및 가상엔진(20a 내지 20b) 상에서 복수의 운영체제(30a 내지 30d)에 대한 환경을 관리하는 컨테이너 관리부(CM)를 포함할 수 있다. 적어도 하나의 운영체제는 이하 '컨테이너(container)'로 설명

한다. 즉, 상술한 가상엔진 없이 도 20에 도시된 차량 가상화 구조 기반의 시스템으로 상술한 사운드 제어가 수행될 수 있다.

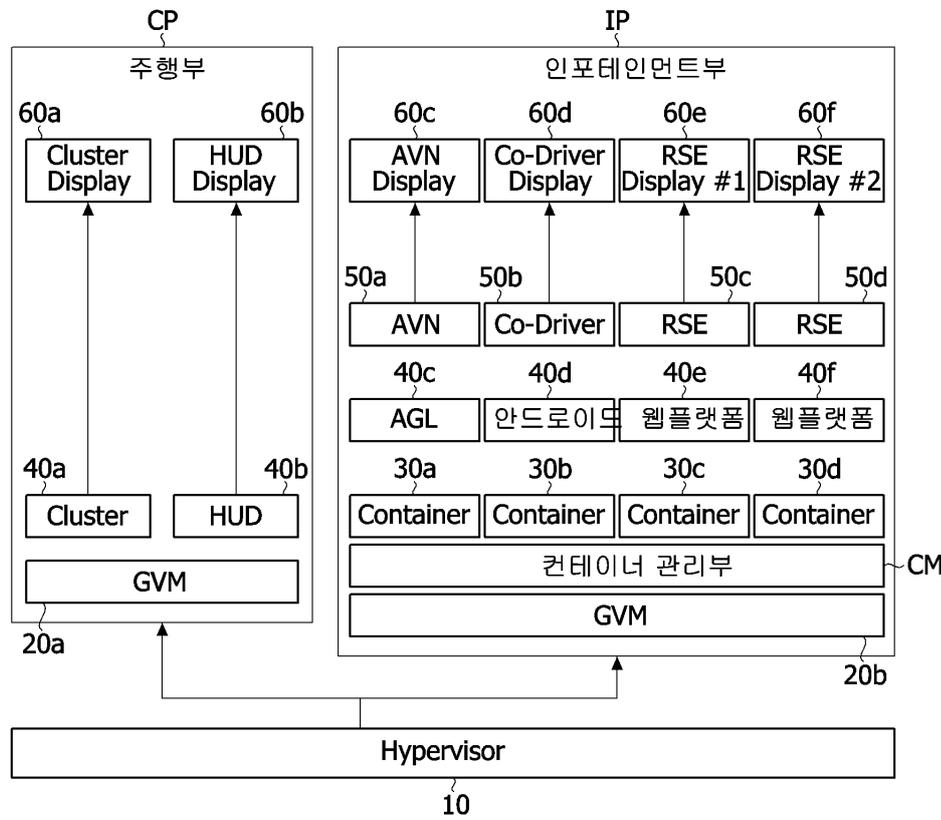
[0152] 상술한, 본 발명의 실시예는 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터에 의해 실행 가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체 일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독가능 매체는 컴퓨터 저장 매체를 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분 리형 및 비분리형 매체를 모두 포함할 수 있다.

[0153] 또한, 본 실시예에서 설명한 차량용 소프트웨어 제어 장치는 컴퓨터 구현 가능한 저장매체에 저장된 컴퓨터 프로그램으로 구현될 수 있다. 또한, 본 실시예에서 사용되는 '~부'라는 용어는 소프트웨어 또는 FPGA(field-programmable gate array) 또는 ASIC과 같은 하드웨어 구성요소를 의미하며, '~부'는 어떤 역할들을 수행할 수 있다. 그렇지만 '~부'는 소프트웨어 또는 하드웨어에 한정되는 의미는 아니다. '~부'는 어드레싱할 수 있는 저장 매체에 있도록 구성될 수도 있고 하나 또는 그 이상의 프로세서들을 재생시키도록 구성될 수도 있다. 따라서, 일 예로서 '~부'는 소프트웨어 구성요소들, 객체지향 소프트웨어 구성요소들, 클래스 구성요소들 및 태스크 구성요소들과 같은 구성요소들과, 프로세스들, 함수들, 속성들, 프로시저들, 서브루틴들, 프로그램 코드의 세그먼트들, 드라이버들, 펌웨어, 마이크로코드, 회로, 데이터, 데이터베이스, 데이터 구조들, 테이블들, 어레이들, 및 변수들을 포함할 수 있다. 구성요소들과 '~부'들 안에서 제공되는 기능은 더 작은 수의 구성요소들 및 '~부'들로 결합되거나 추가적인 구성요소들과 '~부'들로 더 분리될 수 있다. 뿐만 아니라, 구성요소들 및 '~부'들은 디바이스 또는 보안 멀티미디어카드 내의 하나 또는 그 이상의 CPU들을 재생시키도록 구현될 수도 있다.

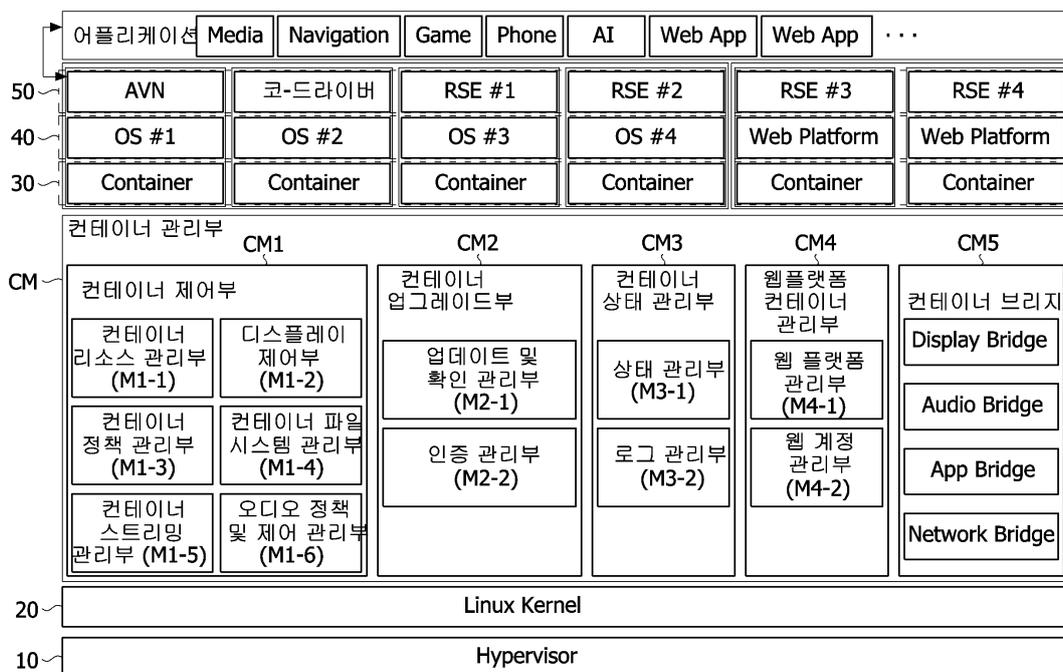
[0154] 이상에서 실시예를 중심으로 설명하였으나 이는 단지 예시일 뿐 본 발명을 한정하는 것이 아니며, 본 발명이 속하는 분야의 통상의 지식을 가진 자라면 본 실시예의 본질적인 특성을 벗어나지 않는 범위에서 이상에 예시되지 않은 여러 가지의 변형과 응용이 가능함을 알 수 있을 것이다. 예를 들어, 실시예에 구체적으로 나타난 각 구성 요소는 변형하여 실시할 수 있는 것이다. 그리고 이러한 변형과 응용에 관계된 차이점들은 첨부된 청구 범위에서 규정하는 본 발명의 범위에 포함되는 것으로 해석되어야 할 것이다.

도면

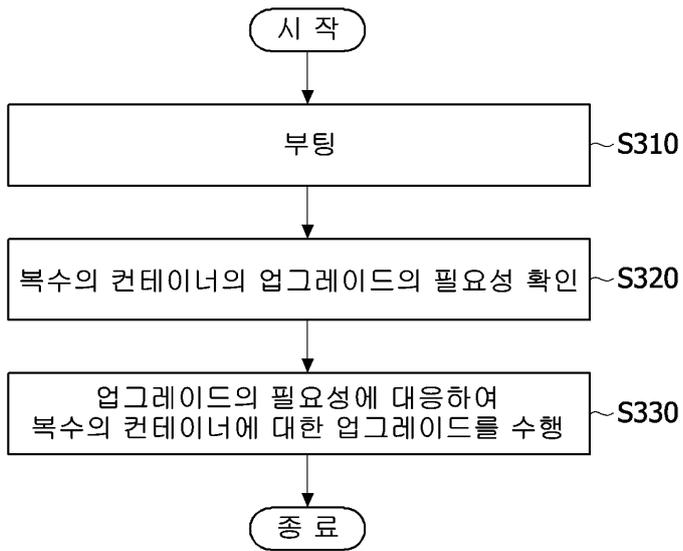
도면1



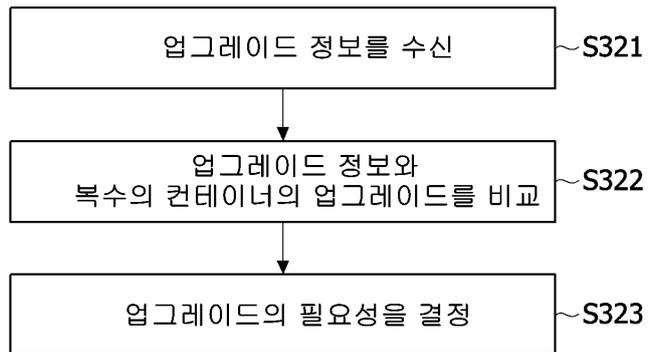
도면2



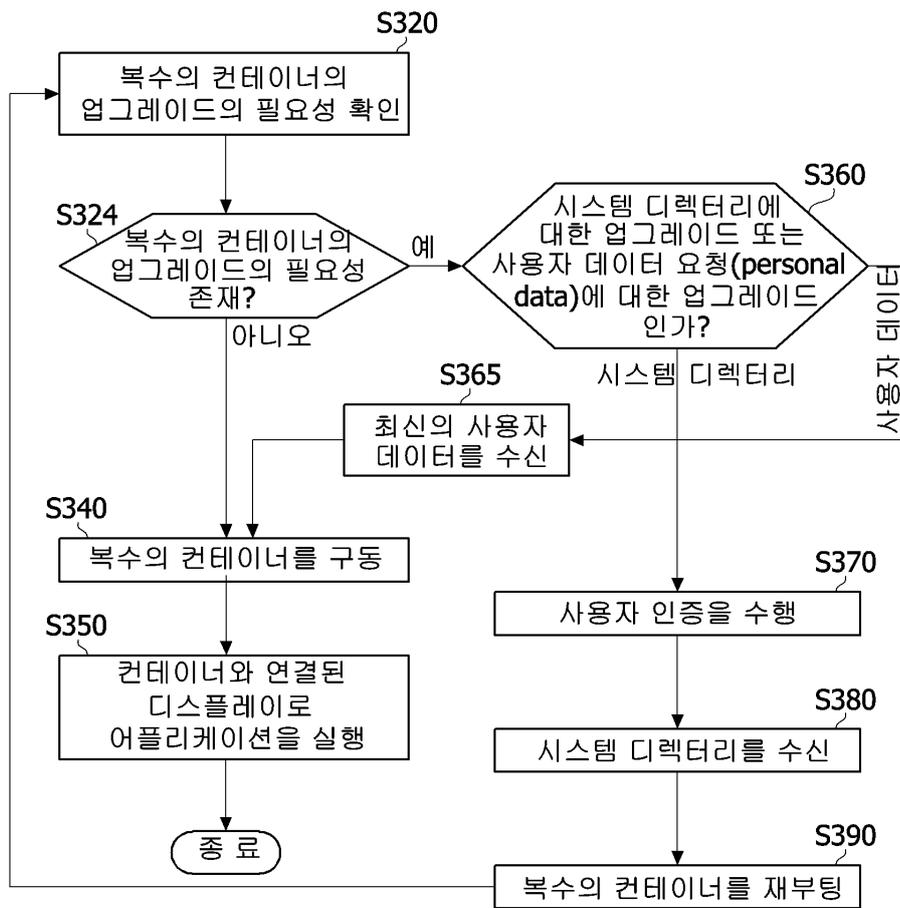
도면3



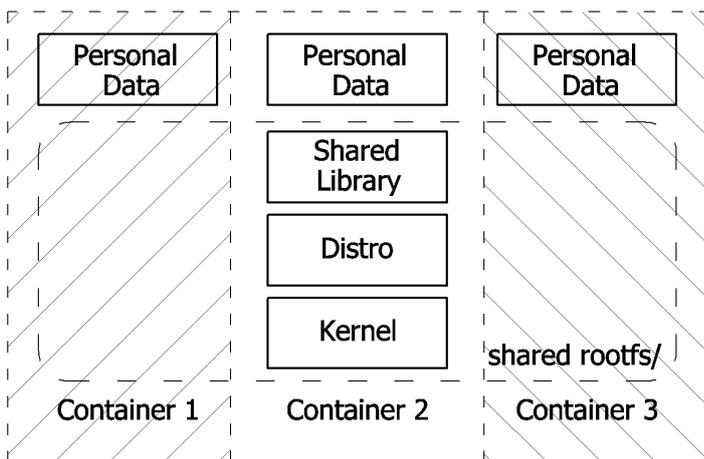
도면4



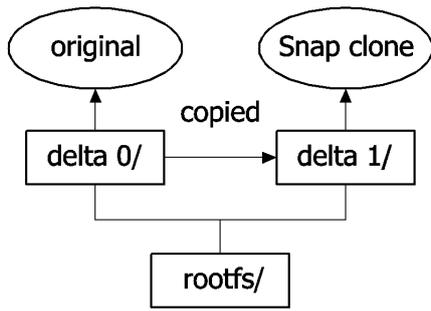
도면5



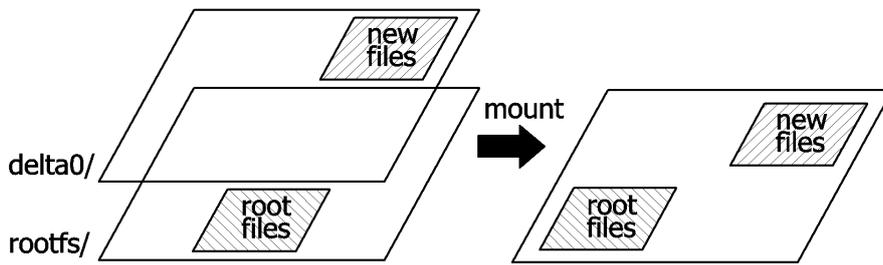
도면6



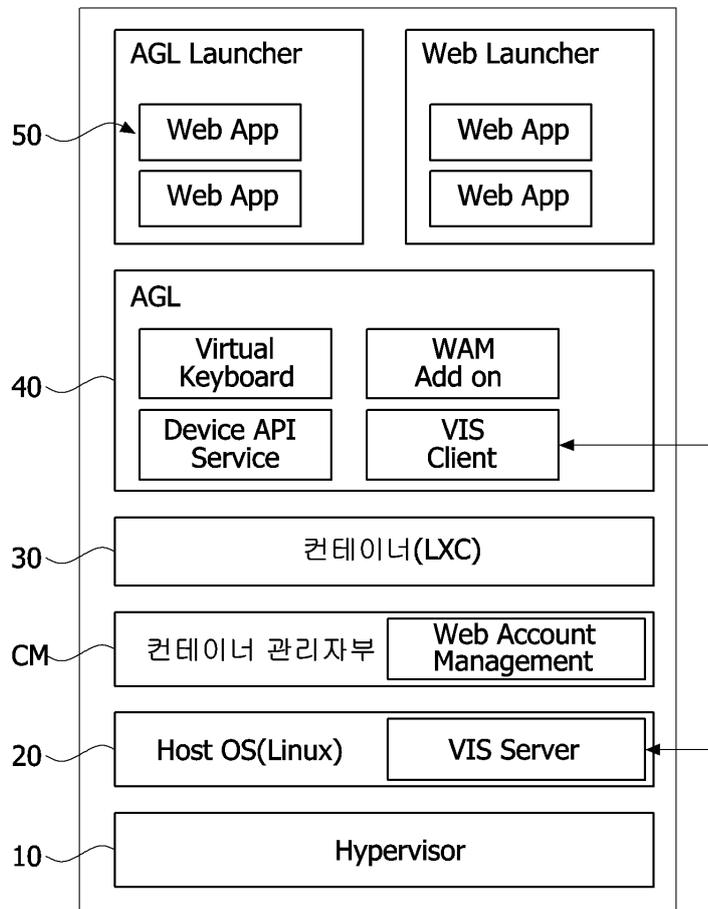
도면7



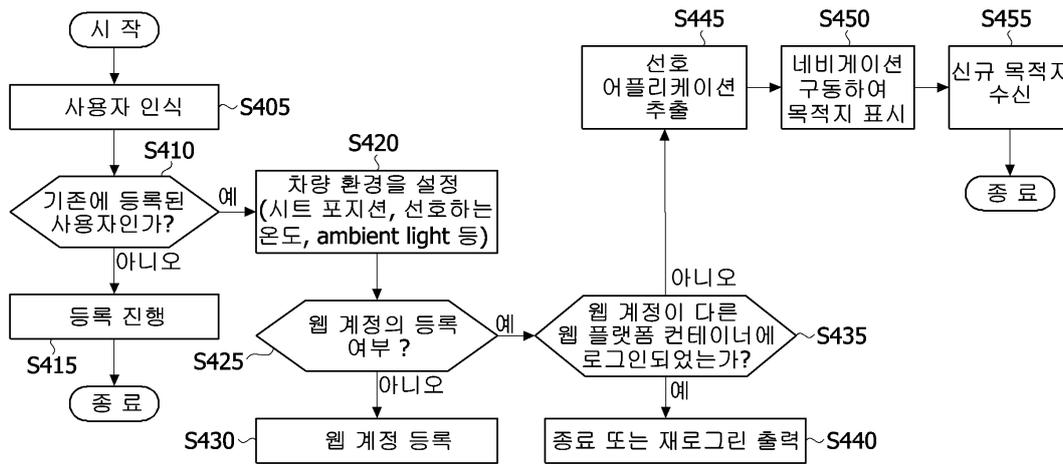
도면8



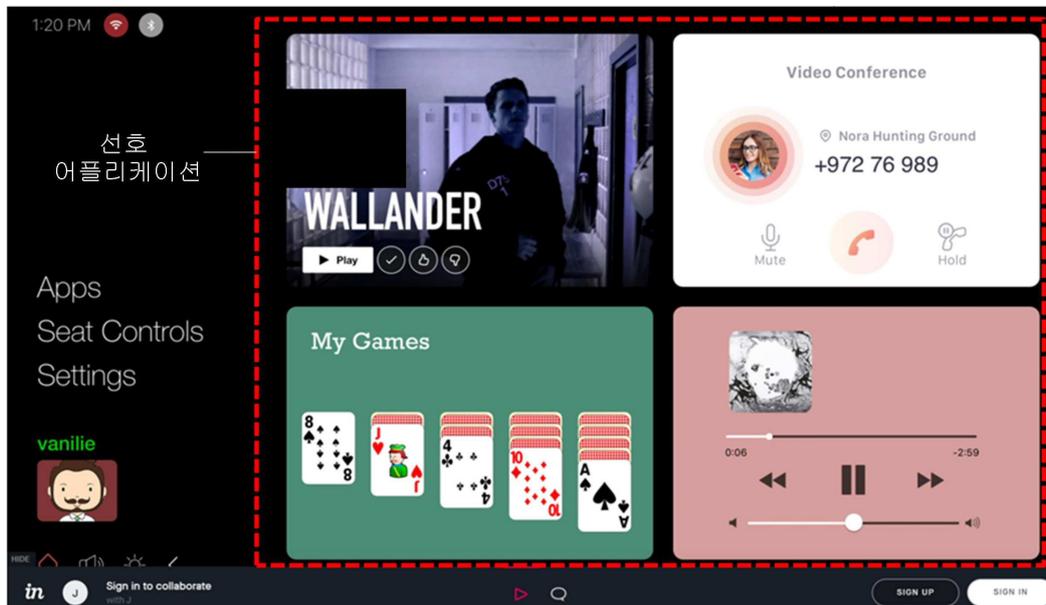
도면9



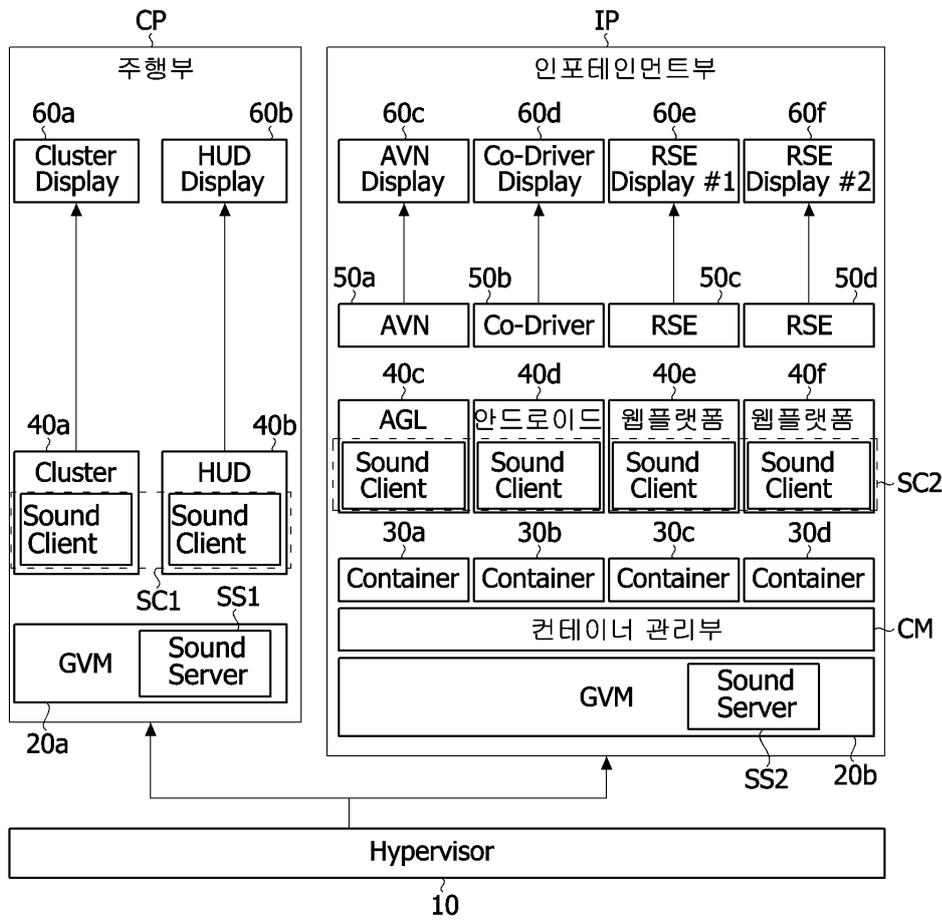
도면10



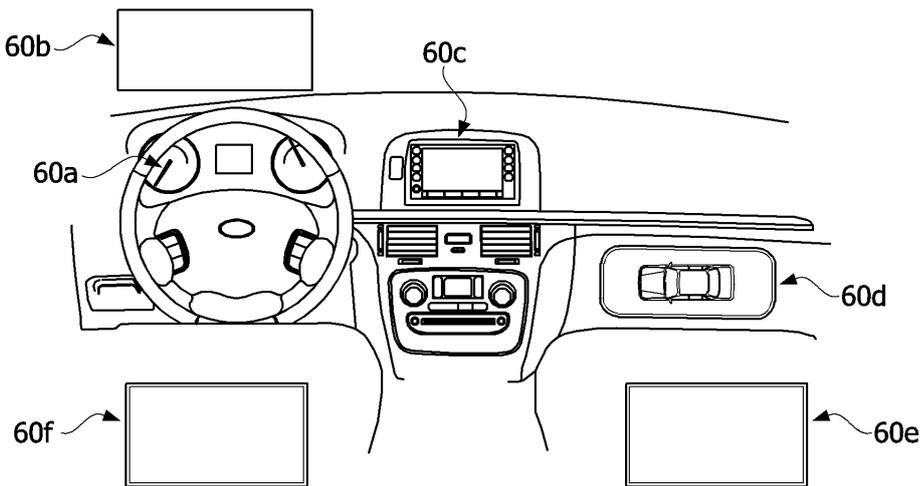
도면11



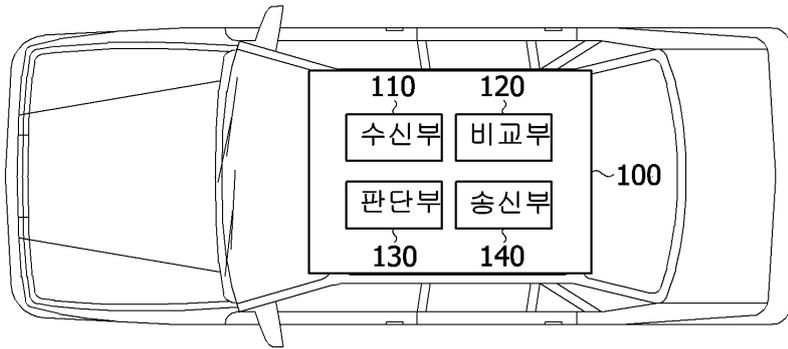
도면12



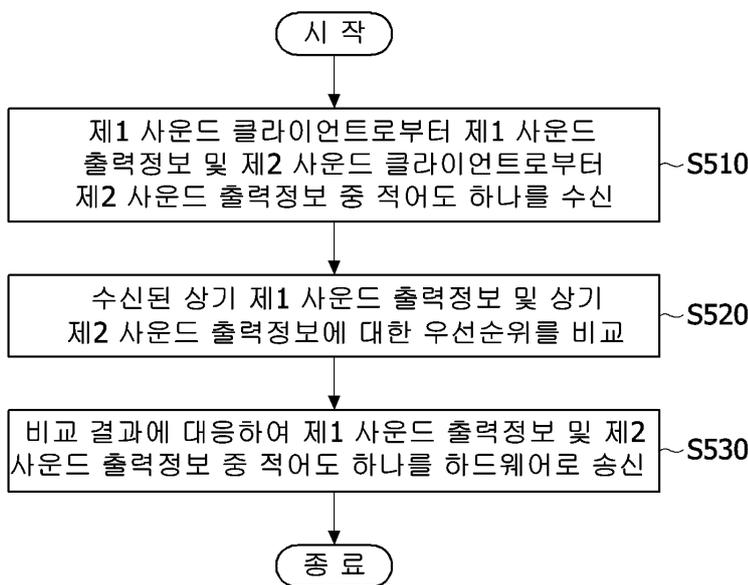
도면13



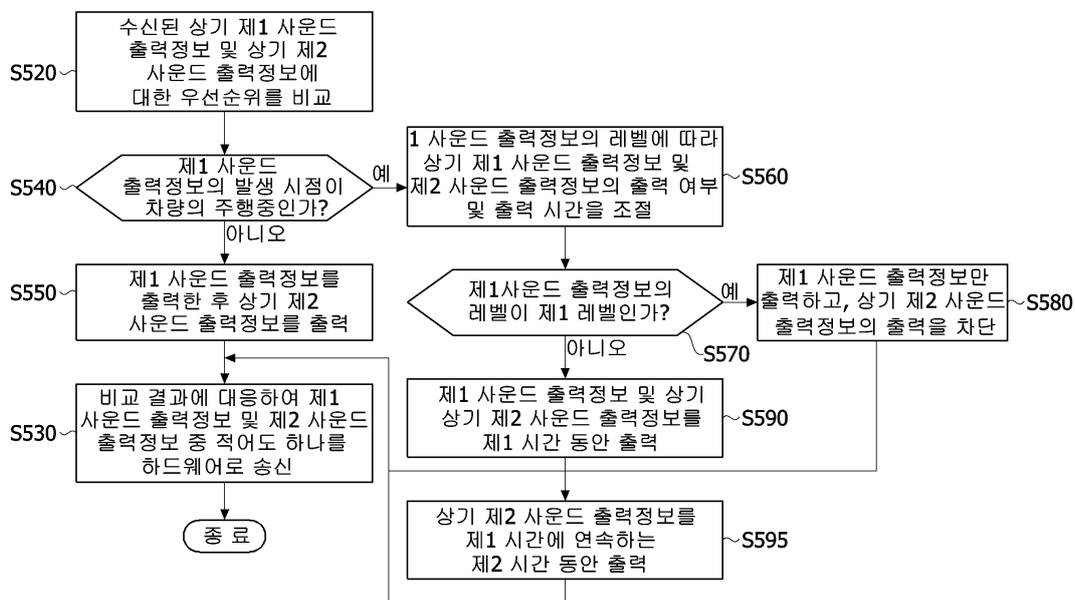
도면14



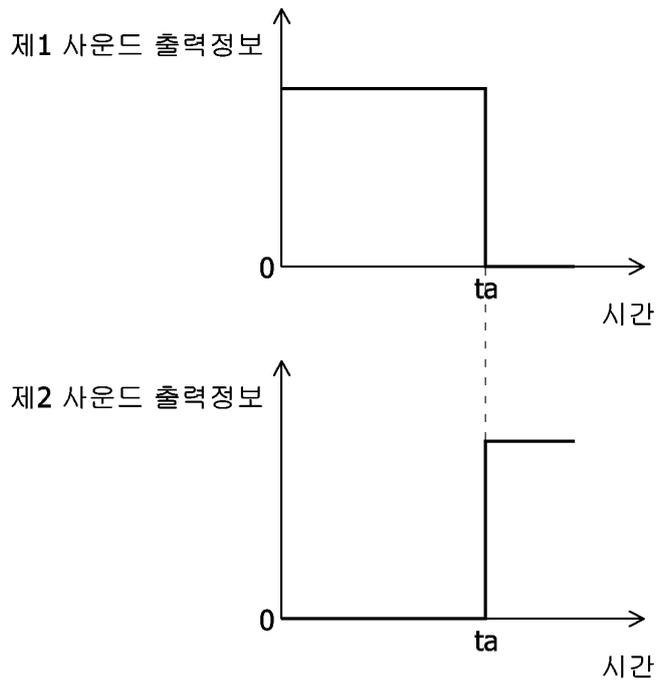
도면15



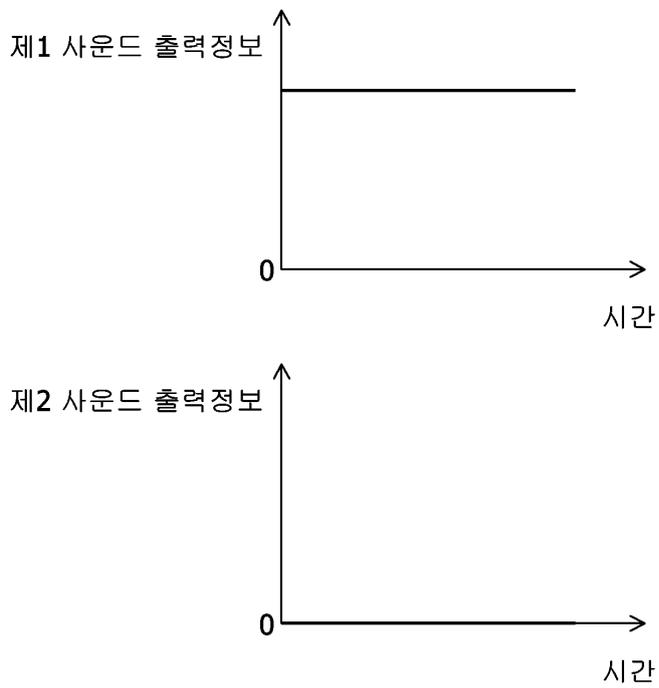
도면16



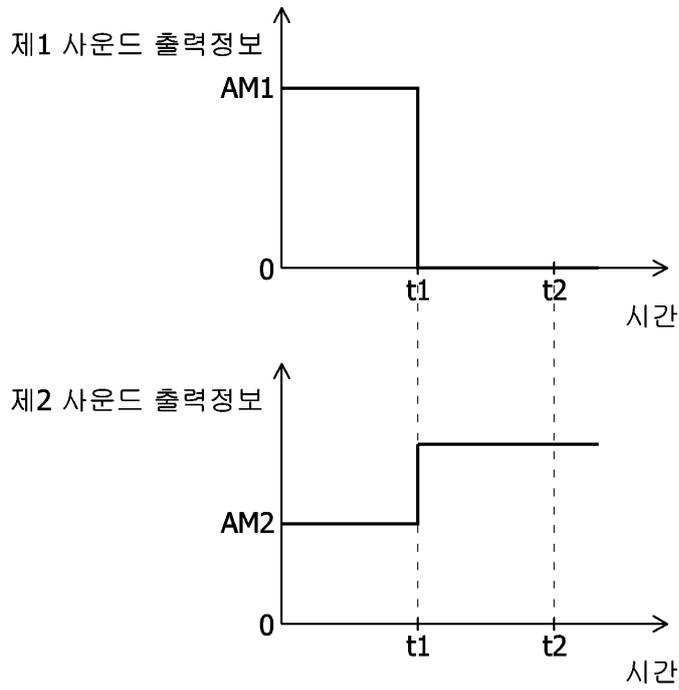
도면17



도면18



도면19



도면20

