

행환경 내에 포함되며, 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스터브(device stub), 장치 스터브에서 장치접속요청을 수신하여 공유 메모리에 복사를 요청하고, 제1가상실행환경에서 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터(Virtual-monitor)에 전송하는 제1가상 브릿지, 스위칭 코드를 수신하여 실행환경을 전환하고, IPI(inter processor interrupt) 스터브를 호출하여 IPI 스터브가 제2가상실행환경 내 제2가상 브릿지로 IPI를 전송하도록 하는 가상 모니터, 및 저장된 장치접속요청의 처리를 위해 장치 드라이버를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지를 포함한다.

이 발명을 지원한 국가연구개발사업

과제고유번호	2013066030
부처명	미래창조과학부
연구관리전문기관	한국연구재단
연구사업명	중견연구자지원사업_도약연구(전략)
연구과제명	미래 인터넷 환경을 위한 네트워크 가상화 기술 개발
기 여 율	1/1
주관기관	고려대학교 산학협력단
연구기간	2013.09.01 ~ 2014.08.31

명세서

청구범위

청구항 1

단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템으로서,

제1가상실행환경과 제2가상실행환경에 의해 공유되는 공유 메모리;

상기 제1가상실행환경 내에 포함되며, 상기 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스터브(device stub);

상기 장치 스터브에서 상기 장치접속요청을 수신하여 상기 공유 메모리에 복사를 요청하고, 상기 제1가상실행환경에서 상기 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터(Virtual-monitor)에 전송하는 제1가상 브릿지;

상기 스위칭 코드를 수신하여 실행환경을 전환하는 가상 모니터(Virtual-monitor); 및

저장된 상기 장치접속요청의 처리를 위해 장치 드라이버를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지를 포함하는 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 시스템.

청구항 2

멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템으로서,

멀티 코어 중에서 선택된 적어도 어느 하나의 코어를 포함하는 제1코어군 내의 제1가상실행환경과 상기 제1코어군을 포함하지 않는 멀티 코어 중에서 선택된 적어도 어느 하나의 코어를 포함하는 제2코어군 내의 제2가상실행환경에 의해 공유되는 공유 메모리;

상기 제1가상실행환경 내에 포함되며, 상기 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스터브;

상기 장치 스터브에서 상기 장치접속요청을 수신하여 상기 공유 메모리에 복사를 요청하고, 상기 제1가상실행환경에서 상기 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터에 전송하는 제1가상 브릿지;

상기 스위칭 코드를 수신하여 실행환경을 전환하고, IPI(inter processor interrupt) 스터브를 호출하여 상기 IPI 스터브가 상기 제2가상실행환경 내 제2가상 브릿지로 IPI를 전송하도록 하는 가상 모니터; 및

저장된 상기 장치접속요청의 처리를 위해 장치 드라이버를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지를 포함하는 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 시스템.

청구항 3

제1항 또는 제2항에 있어서,

상기 가상 모니터는 상기 공유 메모리에 저장된 상기 장치 접속요청을 제2가상 브릿지로 저장하도록 하며, 접속하려는 장치의 접근제어를 제공하는 것을 특징으로 하는 장치 공유 시스템.

청구항 4

제1항 또는 제2항에 있어서,

상기 장치 스터브는 제2가상실행환경 내 장치접속요청에 대한 응답을 수신하는 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 시스템.

청구항 5

제1항 또는 제2항에 있어서,

상기 스위칭 코드는 SMC(Secure Monitor Call)명령어 인 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 시스템.

청구항 6

제1항 또는 제2항에 있어서,

상기 제1가상실행환경은 비보안영역(normal world)이고, 제2가상실행환경은 보안영역(secure world)인 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 시스템.

청구항 7

단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법으로서,

비보안영역에 포함된 장치 스티브가 장치 접속 요청을 수신하여 제1가상 브릿지로 전송하는 단계;

공유 메모리에 상기 장치 접속 요청을 저장하는 단계;

상기 제1가상 브릿지가 스위칭 코드를 가상 모니터로 전송하는 단계;

상기 비보안영역을 보안영역(secure world)으로 전환하는 단계;

상기 보안영역에 포함된 제2가상 브릿지를 통해 상기 공유 메모리에 저장된 상기 장치 접속 요청을 수신하는 단계; 및

상기 제2가상 브릿지는 상기 장치 접속 요청에 따라 상기 보안영역에 포함된 장치 드라이버를 통해 상기 보안영역과 연결된 장치를 동작시키는 단계를 포함하는 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 방법.

청구항 8

멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법으로서,

제1코어의 비보안영역에 포함된 장치 스티브가 장치 접속 요청을 수신하여 제1가상 브릿지로 전송하는 단계;

공유 메모리에 상기 장치 접속 요청을 저장하는 단계;

상기 제1가상 브릿지가 스위칭 코드를 가상 모니터로 전송하는 단계;

상기 스위칭 코드를 전송받고 IPI스티브로 전송하는 단계;

제2코어의 보안영역에 포함된 제2가상 브릿지로 IPI를 전송하는 단계;

상기 제2가상 브릿지를 통해 상기 공유 메모리로부터 상기 장치 접속 요청을 수신하는 단계; 및

상기 장치 접속 요청에 따라 상기 보안영역에 포함된 장치 드라이버를 통해 상기 보안영역과 연결된 장치를 동작시키는 단계를 포함하는 것을 특징으로 하는 트러스트 존 가상환경에서의 장치 공유 방법.

발명의 설명

기술 분야

[0001]

본 발명은 트러스트 존 가상환경에서의 장치 공유 시스템 및 그 방법에 관한 발명이다.

배경 기술

[0002]

자동차 소프트웨어는 ECU(Electronic Control Unit)을 제어하는 제어 소프트웨어와 AVN(Audio, Video 및 Navigation)과 같은 인포테인먼트 장치를 제어하는 확장 소프트웨어로 구성되어 있다. 현재 자동차 시장의 트렌드는 제어 소프트웨어로 좀더 많은 확장 소프트웨어를 통합하는 추세로 진행되고 있다. 그러나 확장 소프트웨어는 몇몇 안전상의 문제를 안고 있다. 즉 확장 소프트웨어는 헤드업 디스플레이 및 터치 인터페이스와 같은 사용자 편의사항을 제공하기 위해 다양한 기능을 수행해야 하고 그만큼 복잡해지고 있다. 이러한 확장 소프트웨어의 복잡성은 소프트웨어 실행의 정확도를 담보하기 어렵고, 확장 소프트웨어의 오류는 제어 소프트웨어에 영향을

주어 안전상에 문제를 발생시키기도 한다. 따라서 제어 소프트웨어로부터 확장 소프트웨어를 고립시킬 필요성이 대두되고 있다.

[0003] 확장 소프트웨어의 고립을 위해서 트러스트 존이 사용된다. 트러스트 존은 두 개의 가상실행환경인 보안영역(secure world)과 비보안영역(normal world)으로 구별된다. 두 영역은 서로 다른 영역과 독립적인 CPU, 레지스터의 상태를 갖는다.

[0004] 일반적으로 제어 소프트웨어는 실시간 요청을 만족시키기 위해서 리얼타임OS(real-time operating system)에서 구동되고, 확장 소프트웨어는 인포테인먼트 서비스를 제공하기 위해 범용OS(general-purpose operating system)에서 구동된다.

[0005] 트러스트 존을 적용하면 제어 소프트웨어와 확장 소프트웨어는 각각 보안영역에 있는 리얼타임OS와 비보안영역에 있는 범용OS에 의해서 서로 고립된다. 그러나 비록 제어 소프트웨어와 확장 소프트웨어가 안전을 위해서 고립되어 있다 하더라도 CAN(control area network) 및 센서들과 같은 장치를 공유할 필요성이 있다. 예를 들어 디지털 클러스터는 CAN으로부터 주기적으로 엔진 RPM과 스피드 정보를 수집하고 수집된 데이터로 3차원 디지털 게이지를 제공한다.

[0006] 도1은 종래기술인 자동차 전장 시스템에서의 오류발생 상황을 가정한 모식도이다.

[0007] 도1에 도시된 바와 같이 디지털 클러스터가 일정하게 CAN으로 대량의 요청(request)을 보낸다면 다른 ECU들은 자동차를 제어하는 정보를 수집할 수 없고, 결과적으로 자동차는 제어의 범위를 벗어나게 된다. 결국 장치가 제어 소프트웨어와 확장 소프트웨어 사이에서 공유될 때, 트러스트 존을 사용한 소프트웨어 실행영역의 고립만으로는 확장 소프트웨어가 제어 소프트웨어에 영향을 주지 않도록 하기에 충분하지 않다. 결국 범용OS에서 확장 소프트웨어가 직접적으로 공유 장치에 접근하게 될 때 확장 소프트웨어의 오류는 제어 소프트웨어에 전파될 수 있다. 그러므로 확장 소프트웨어의 고립과 더불어 보안영역에서의 장치접근 매커니즘이 필연적으로 필요하다.

발명의 내용

해결하려는 과제

[0008] 본 발명은 위와 같은 문제를 해결하기 위해 안출된 것으로서 제어 소프트웨어와 확장 소프트웨어가 장치를 공유할 경우 오류가 발생하거나 제어 소프트웨어에 영향을 주지 않도록 하는 트러스트 존 가상환경에서의 장치 공유 시스템 및 그 방법을 제공함을 목적으로 한다.

과제의 해결 수단

[0009] 본 발명의 실시예에 따른 단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템은 제1가상실행환경과 제2가상실행환경에 의해 공유되는 공유 메모리, 제1가상실행환경 내에 포함되며, 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스터브(device stub), 장치 스터브에서 장치접속요청을 수신하여 공유 메모리에 복사를 요청하고, 제1가상실행환경에서 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터(Virtual-monitor)에 전송하는 제1가상 브릿지, 스위칭 코드를 수신하여 실행환경을 전환하는 가상 모니터(Virtual-monitor) 및 저장된 장치접속요청의 처리를 위해 장치 드라이버를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지를 포함한다.

[0010] 본 발명의 또 다른 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템은 멀티 코어 중에서 선택된 적어도 어느 하나의 코어를 포함하는 제1코어군 내의 제1가상실행환경과 제1코어군을 포함하지 않는 멀티 코어 중에서 선택된 적어도 어느 하나의 코어를 포함하는 제2코어군 내의 제2가상실행환경에 의해 공유되는 공유 메모리, 제1가상실행환경 내에 포함되며, 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스터브(device stub), 장치 스터브에서 장치접속요청을 수신하여 공유 메모리에 복사를 요청하고, 제1가상실행환경에서 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터(Virtual-monitor)에 전송하는 제1가상 브릿지, 스위칭 코드를 수신하여 실행환경을 전환하고, IPI(inter processor interrupt) 스터브를 호출하여 IPI 스터브가 제2가상실행환경 내 제2가상 브릿지로 IPI(inter processor interrupt)를 전송하도록 하는 가상 모니터(Virtual-monitor), 저장된 장치접속요청의 처리를 위해 장치 드라이버를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지를 포함한다.

발명의 효과

[0011] 본 발명의 실시예에 따른 트러스트 존 가상환경에서의 장치 공유 시스템 및 그 방법에 의하면 안전한 장치 공유 소프트웨어 프레임 워크가 제공되고, 멀티코어프로세서에서 공유장치 접근제어를 수행할 수 있다.

도면의 간단한 설명

[0012] 도1은 종래기술인 자동차 전장 시스템에서의 오류발생 상황을 가정한 모식도이다.
 도2는 본 발명의 실시예에 따른 단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템 블록도이다.
 도3은 본 발명의 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템 블록도이다.
 도4는 본 발명의 실시예에 따른 단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법의 순서도이다.
 도5는 본 발명의 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법의 순서도이다.

발명을 실시하기 위한 구체적인 내용

[0013] 본 명세서에 개시되어 있는 본 발명의 개념에 따른 실시 예들에 대해서 특정한 구조적 또는 기능적 설명은 단지 본 발명의 개념에 따른 실시 예들을 설명하기 위한 목적으로 예시된 것으로서, 본 발명의 개념에 따른 실시 예들은 다양한 형태들로 실시될 수 있으며 본 명세서에 설명된 실시 예들에 한정되지 않는다.

[0014] 본 발명의 개념에 따른 실시 예들은 다양한 변경들을 가할 수 있고 여러 가지 형태들을 가질 수 있으므로 실시 예들을 도면에 예시하고 본 명세서에서 상세하게 설명하고자 한다. 그러나, 이는 본 발명의 개념에 따른 실시 예들을 특정한 개시 형태들에 대해 한정하려는 것이 아니며, 본 발명의 사상 및 기술 범위에 포함되는 모든 변경, 균등물, 또는 대체물을 포함한다.

[0015] 제1 또는 제2 등의 용어는 다양한 구성 요소들을 설명하는데 사용될 수 있지만, 상기 구성 요소들은 상기 용어들에 의해 한정되어서는 안 된다. 상기 용어들은 하나의 구성 요소를 다른 구성 요소로부터 구별하는 목적으로만, 예컨대 본 발명의 개념에 따른 권리 범위로부터 벗어나지 않은 채, 제1 구성 요소는 제2 구성 요소로 명명될 수 있고 유사하게 제2 구성 요소는 제1 구성 요소로도 명명될 수 있다.

[0016] 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. 구성 요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0017] 본 명세서에서 사용한 용어는 단지 특정한 실시 예를 설명하기 위해 사용된 것으로서, 본 발명을 한정하려는 의도가 아니다. 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한, 복수의 표현을 포함한다. 본 명세서에서, "포함하다" 또는 "가지다" 등의 용어는 본 명세서에 기재된 특징, 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이지, 하나 또는 그 이상의 다른 특징들이나 숫자, 단계, 동작, 구성 요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0018] 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 가진다. 일반적으로 사용되는 사전에 정의되어 있는 것과 같은 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본 명세서에서 명백하게 정의하지 않는 한, 이상적이거나 과도하게 형식적인 의미로 해석되지 않는다.

[0019] 이하, 본 명세서에 첨부된 도면들을 참조하여 본 발명의 실시 예들을 상세히 설명한다.

[0020] 본 발명의 실시예에 따른 트러스트 존 가상환경에서의 장치 공유 시스템은 하나의 프로세서내에 단일 코어를 포함하는 트러스트 존 가상환경에서의 시스템과 다중 코어를 포함하는 트러스트 존 가상환경에서의 시스템으로 구

분될 수 있다.

- [0021] 먼저 본 발명의 실시예에 따른 단일 코어를 포함하는 트러스트 존 가상환경에서의 시스템을 설명한다.
- [0022] 도2는 본 발명의 실시예에 따른 단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템 블록도이다.
- [0023] 도2에 도시된 바와 같이 본 발명의 실시예에 따른 시스템은 제1가상실행환경과 제2가상실행환경에 의해 공유되는 공유 메모리(100), 제1가상실행환경 내에 포함되며, 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스터브(device stub)(210), 장치 스터브(210)에서 장치접속요청을 수신하여 공유 메모리(100)에 복사를 요청하고, 제1가상실행환경에서 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터(Virtual-monitor)(400)에 전송하는 제1가상 브릿지(220), 스위칭 코드를 수신하여 실행환경을 전환하는 가상 모니터(400) 및 저장된 장치접속요청의 처리를 위해 장치 드라이버(310)를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지(320)를 포함한다.
- [0024] 제1가상실행환경은 비보안영역(normal world)(200)일 수 있으며, 제2가상실행환경(secure world)(300)은 보안영역일 수 있다.
- [0025] 설명의 편의상 도2에 도시된 바와 같이 제1가상실행환경을 비보안영역(200)으로, 제2가상실행환경을 보안영역(300)으로 간주하고 설명한다.
- [0026] 공유 메모리(100)는 프로세서의 동작 및 기능의 실행을 위해 이용되는 인스트럭션 및 데이터를 저장할 수 있다. 공유 메모리(100)는 RAM(Random Access Memory), 정적 RAM(SRAM), 동적 RAM(DRAM), 동기식 DRAM(SDRAM), 듀얼 데이터 레이트 RAM(DDRAM) 등을 포함할 수 있으나 이에 제한되는 것은 아니다. 공유 메모리(100)는 내부에 저장되는 하나 이상의 코드 또는 데이터 시퀀스를 포함할 수 있으며, 코드 시퀀스는 하나 이상의 기능호출, 하나 이상의 서브루틴 또는 하나 이상의 동작을 나타내는 머신 인스트럭션 세트 또는 그룹일 수 있다. 본 명세서에서 프로그램은 이들 중 하나를 단독으로 지칭하거나 이들 중 하나 이상의 결합을 지칭할 수 있다.
- [0027] 공유 메모리(100)는 프로세서 내에 구별되는 비보안영역(200), 보안영역(300)과 연결되어 있으며, 보다 상세하게는 도2에 도시된 바와 같이 제1가상 브릿지(220), 제2가상 브릿지(320)에 연결되어 있을 수 있다.
- [0028] 장치 스터브(210)는 장치(500)에 접근을 시작하기 위한 기능요소로서 비보안영역(200) 및/또는 보안영역(300)에 포함될 수 있으며 각 가상실행환경 내에 포함되어 있는 가상 브릿지(220, 320)와 연결된다. 장치 스터브(210)는 장치(500) 공유를 위해 각 영역의 운영체제에 추가된 모듈로서 공유장치를 사용하고자 하는 영역의 운영체제에 포함되며, 의사(pseudo) 드라이버로서 운영체제로부터 장치 사용에 대한 요청(request)을 받고 상대 영역으로부터 장치 요청에 대한 응답(response)을 전달하기 위한 인터페이스 역할을 담당한다.
- [0029] 가상 브릿지(220, 320)는 장치 공유를 위한 모듈로서 비보안영역(200), 보안영역(300) 모두에 존재하여 상대영역과 통신하기 위한 인터페이스 역할을 한다. 본 발명의 가상 브릿지는 제1가상 브릿지(220), 제2가상 브릿지(320)로 구별된다.
- [0030] 제1가상 브릿지(220)는 비보안영역(200)에 포함되어 있으며, 장치스터브(210)로부터 장치접속요청을 수신한다. 제1가상 브릿지(220)는 장치접속요청을 수신하여 공유 메모리(100)에 복사를 요청한다. 또한 비보안영역(200)에서 보안영역(300)으로 실행환경전환을 명하는 스위칭 코드를 전송한다.
- [0031] 제2가상 브릿지(320)는 보안영역(300)에 포함되어 있으며, 저장된 장치접속요청의 처리를 위해 장치 드라이버(310)를 호출하고 접속하려는 장치(500)의 동작제어를 제공한다.
- [0032] 가상 모니터(400)는 제1가상 브릿지(220)로부터 스위칭 코드를 수신하여 실행환경을 전환하는 모듈로서 공유 메모리(100)에 저장된 장치접속요청을 제2가상 브릿지(320)에 저장하도록 하고 접속하려는 장치(500)의 접근제어를 담당한다. 특히 가상 모니터(400)에 포함되어 있는 IPC(Inter Process Communication)모듈(미도시)은 각 영역간의 통신을 위한 모듈로서 상대 영역과의 통신을 위한 스위칭 코드를 해석하고 처리하는 기능을 수행한다.
- [0033] 여기서 스위칭 코드는 SMC(Secure Monitor Call)명령어로서 비보안영역(200)에 포함되는 제1가상 브릿지(220)를 통해 가상 모니터(400)로 전달되어 실행환경을 전환하는 인스트럭션(instruction)이다.
- [0034] 이하 본 발명의 실시예에 따른 트러스트 존 가상환경에서의 장치 공유 시스템 작동상태를 도2를 참고하여 설명한다.

- [0035] 비보안영역(200)에서 제1가상 브릿지(220)는 장치 스테브(210)로부터 장치접속요청을 수신한다. 장치접속요청을 수신한 제1가상 브릿지(220)는 스위칭 코드인 SMC(Secure Monitor Call)를 가상 모니터(400)로 전송한다. 가상 모니터(400)는 트러스트 존에서 서로 다른 영역을 스위칭하기 위한 모니터 소프트웨어일 수 있고, 두 영역이 스위칭 될 때 캐시(cache), 변환색인버퍼(TLB, Translation look a side buffer), MMU(Memory Management Unit)를 포함하는 하드웨어 콘텍스트(hardware context)가 자동으로 변환되어 두 영역을 완전히 분리시킨다. 다만 범용 레지스터(general-purpose register)는 두 영역 사이에 공유되어진다. 제1가상 브릿지(220)로부터 SMC가 수신되면 트러스트 존이 모니터 모드로 진입하고, 가상 모니터(400)의 모니터 소프트웨어는 두 영역 사이의 범용 레지스터를 스위칭하게 된다.
- [0036] 가상 모니터(400)에 의해서 실행환경이 전환(switching)되면 제2가상 브릿지(320)는 공유 메모리(100)로부터 기 저장되어 있는 장치접속요청(request)을 전달받아 장치 드라이버(310)에 전송한다. 결국 장치 드라이버(310)는 보안영역(300)과 연결되어 있는 장치(500)를 제어함으로써 트러스트 존 가상환경에서의 장치 공유가 이루어지게 된다.
- [0037] 지금까지 살펴본 트러스트 존 가상환경에서의 장치 공유 시스템은 단일 코어에서의 장치 공유시스템에 대한 것으로서 이하 본 발명의 다른 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템에 대해서 살펴본다.
- [0038] 본 발명의 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템은 멀티 코어 중에서 선택된 적어도 어느 하나의 코어를 포함하는 제1코어군 내의 제1가상실행환경과 제1코어군을 포함하지 않는 멀티 코어 중에서 선택된 적어도 어느 하나의 코어를 포함하는 제2코어군 내의 제2가상실행환경에 의해 공유되는 공유 메모리(100), 제1가상실행환경 내에 포함되며, 제2가상실행환경 내 장치에 대한 장치접속요청을 수신하는 장치 스테브(210), 장치 스테브(210)에서 장치접속요청을 수신하여 공유 메모리(100)에 복사를 요청하고, 제1가상실행환경에서 제2가상실행환경으로 실행환경전환을 명하는 스위칭 코드를 가상 모니터(400)에 전송하는 제1가상 브릿지(220), 스위칭 코드를 수신하여 실행환경을 전환하고, IPI(inter processor interrupt) 스테브(230)를 호출하여 IPI 스테브(230)가 제2가상실행환경 내 제2가상 브릿지(320)로 IPI(inter processor interrupt)를 전송하도록 하는 가상 모니터(400), 저장된 장치접속요청의 처리를 위해 장치 드라이버(310)를 호출하고, 접속하려는 장치의 동작제어를 제공하는 제2가상 브릿지(320)를 포함한다.
- [0039] 도3은 본 발명의 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템 블록도이다.
- [0040] 앞서 설명한 단일 코어에서의 장치 공유 시스템의 구성요소와 중복되는 구성에 대한 설명은 생략한다. 또한 설명의 편의상 두 개의 코어 각각에 포함되어 있는 비보안영역(200)과 보안영역(300)으로 이루어진 시스템을 가정하고 설명한다.
- [0041] 멀티 코어 프로세서에서 리얼타임OS와 범용OS는 서로 분리된 코어에서 작동할 수 있다.
- [0042] 도3에 도시된 바와 같이 만약 범용OS가 제1코어(20)에서 작동하고, 리얼타임OS가 제2코어(30)에서 작동하는 상태에 있다면 앞서 설명한 스위칭 코드인 SMC명령어는 같은 코어에서만 영역을 스위칭하도록 되어 있기 때문에 범용OS가 실행되는 비보안영역(200)과 리얼타임OS가 실행되는 보안영역(300)을 스위칭할 수 없다. 이와 같은 한계를 극복하기 위해서 멀티 코어 프로세서에서 코어들 사이의 통신방법인 IPI를 사용한다.
- [0043] 즉 비보안영역(200)을 포함하는 제1코어(20)는 IPI스테브(230)를 포함한다. IPI는 트러스트 존에서 제약이 존재한다. 보안영역(300)은 IPI를 다른 코어에 있는 모든 영역으로 전송할 수 있으나 비보안영역(200)은 다른 코어에 있는 비보안영역으로만 IPI를 전송할 수 있다. 결국 이와 같은 IPI의 제약을 극복하고 도3에 도시된 바와 같이 IPI를 보안영역(300)이 존재하는 제2코어(30)로 전송하기 위해 IPI스테브(230)가 제1코어(20)에 존재한다. 가상 모니터(400)는 SMC를 수신하여 실행환경을 전환하기 위해 IPI 스테브(230)를 호출한다. IPI스테브(230)는 보안영역(300) 내 제2가상 브릿지(320)로 IPI를 전송하고, IPI를 입력받은 제2가상 브릿지(320)는 공유 메모리(100)로부터 장치접속요청을 전달받아 저장하고 장치 드라이버(310)에 전달하여 보안영역(300)과 연결되어 있는 장치(500)를 제어한다. IPI는 멀티 코어 프로세서에서 코어 간 통신을 위해 사용되는 인터럽트의 한 종류이다. IPI는 예약된 행동을 핸들러(미도시)로 등록하여 다른 코어로부터 IPI를 받으면 등록된 핸들러(미도시)가 동작하여 핸들러(미도시)를 통해 상대 코어로 하여금 원하는 코드를 수행하도록 할 수 있다.
- [0044] 이와 같은 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템은 도3에 도시된 구조 이외에 비보안영역(200), 보안영역(300)이 다수의 코어로 이루어진 코어 군에서 실행되는 장치 공유 시스템에도 적용이

가능하다. 즉 IPI의 전송은 IPI스터브(230)를 통해 이루어지고, 서로 다른 코어에 포함되어 있는 보안영역(300)간의 통신은 IPI를 통해 가능하기 때문에 코어 군에서 이루어지는 장치 공유 시스템이 가능해진다.

[0045] 이상 본 발명의 실시예에 따른 단일 코어, 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 시스템에 대해서 살펴보았다.

[0046] 이하 본 발명의 또 다른 양태에 따른 단일, 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법에 대해서 도4와 도5를 참고하여 살펴본다.

[0047] 도4는 본 발명의 실시예에 따른 단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법의 순서도이다.

[0048] 도4에 도시된 바와 같이 단일 코어에서의 장치 공유 방법은 비보안영역에 포함된 장치 스테브가 장치 접속 요청을 수신하여 제1가상 브릿지로 전송하는 단계(S100), 공유 메모리에 장치 접속 요청을 저장하는 단계(S200), 제1가상 브릿지가 스위칭 코드를 가상 모니터로 전송하는 단계(S300), 비보안영역을 보안영역으로 전환하는 단계(S400), 보안영역에 포함된 제2가상 브릿지를 통해 공유 메모리에 저장된 장치 접속 요청을 수신하는 단계(S500) 및 제2가상 브릿지가 장치 접속 요청에 따라 보안영역에 포함된 장치 드라이버를 통해 보안영역과 연결된 장치를 동작시키는 단계(S600)를 포함한다.

[0049] 제1가상 브릿지로 장치접속요청을 전송하는 단계(S100)는 가상 모니터가 스위칭 코드인 SMC명령어를 전송받아 영역의 전환을 수행하도록 한다. 또한 제1가상 브릿지를 통해 장치접속요청이 공유 메모리에 저장된다(S200). SMC명령어의 실행으로 인해서 비보안영역에서 보안영역으로 전환된 후 제2가상 브릿지를 통해 공유 메모리에 기 저장되어 있는 장치접속요청을 수신하여 장치 드라이버에 전달함으로써 장치를 동작시키고 제어한다.

[0050] 도5는 본 발명의 실시예에 따른 멀티 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유 방법의 순서도이다.

[0051] 도5에 도시된 바와 같이 멀티 코어에서의 장치 공유 방법은 제1코어의 비보안영역(normal world)에 포함된 장치 스테브가 장치 접속 요청을 수신하여 제1가상 브릿지로 전송하는 단계(S10), 공유 메모리에 장치 접속 요청을 저장하는 단계(S20), 제1가상 브릿지가 스위칭 코드를 가상 모니터로 전송하는 단계(S30), 스위칭 코드를 전송받고 IPI스터브로 전송하는 단계(S40), 제2코어의 보안영역(secure world)에 포함된 제2가상 브릿지로 IPI를 전송하는 단계(S50), 제2가상 브릿지를 통해 공유 메모리로부터 장치 접속 요청을 수신하는 단계(S60), 장치 접속 요청에 따라 보안영역에 포함된 장치 드라이버를 통해 보안영역과 연결된 장치를 동작시키는 단계(S70)를 포함한다.

[0052] 앞서 설명한 단일 코어를 포함하는 트러스트 존 가상환경에서의 장치 공유방법과 차이점은 스위칭 코드인 SMC명령어는 단일 코어내의 영역간의 전송은 가능하나 코어 간의 전송이 불가능하기 때문에 스위칭 코드를 전송받은 가상 모니터가 IPI스터브로 스위칭 코드를 전송하고, IPI스터브는 IPI를 제2가상 브릿지로 전송한다. 제2가상 브릿지는 공유 메모리로부터 장치 접속 요청을 수신하고 이를 장치 드라이버로 전송하여 장치를 동작시키도록 제어하는 과정을 거친다.

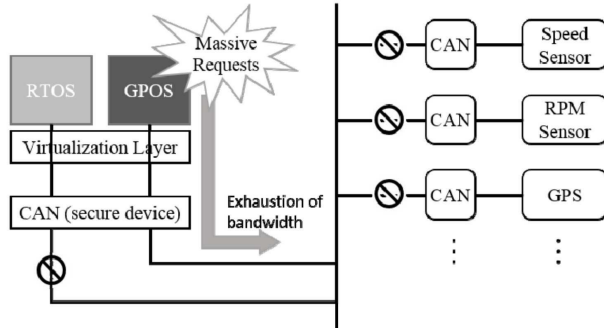
[0053] 결국 본 발명의 실시예에 따른 트러스트 존 가상환경에서의 장치 공유 시스템 및 그 방법에 의하면 안전한 장치 공유 소프트웨어 프레임 워크가 제공되고, 멀티코어프로세스에서 안전하게 공유장치 접근제어를 수행할 수 있다.

부호의 설명

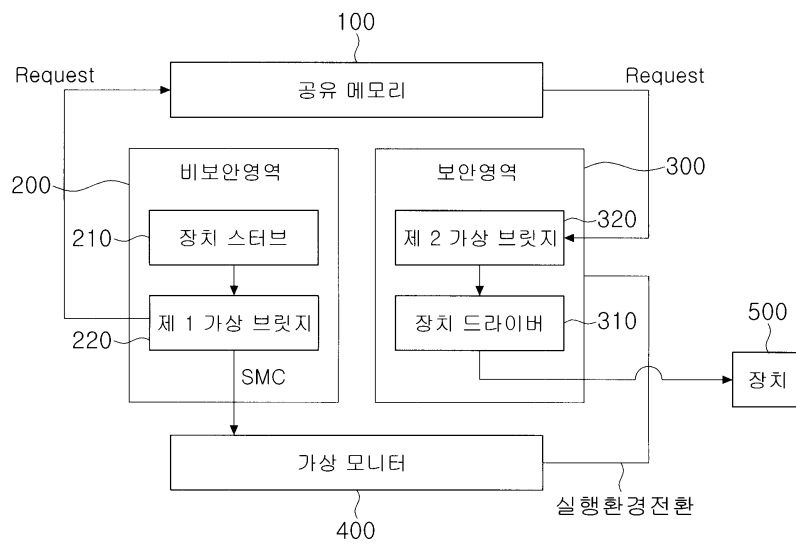
[0054]	20 : 제1코어	30 : 제2코어
	100 : 공유 메모리	200 : 비보안영역
	210 : 장치 스테브	220 : 제1가상 브릿지
	230 : IPI스터브	300 : 보안영역
	310 : 장치 드라이버	320 : 제2가상 브릿지
	400 : 가상 모니터	500 : 장치

도면

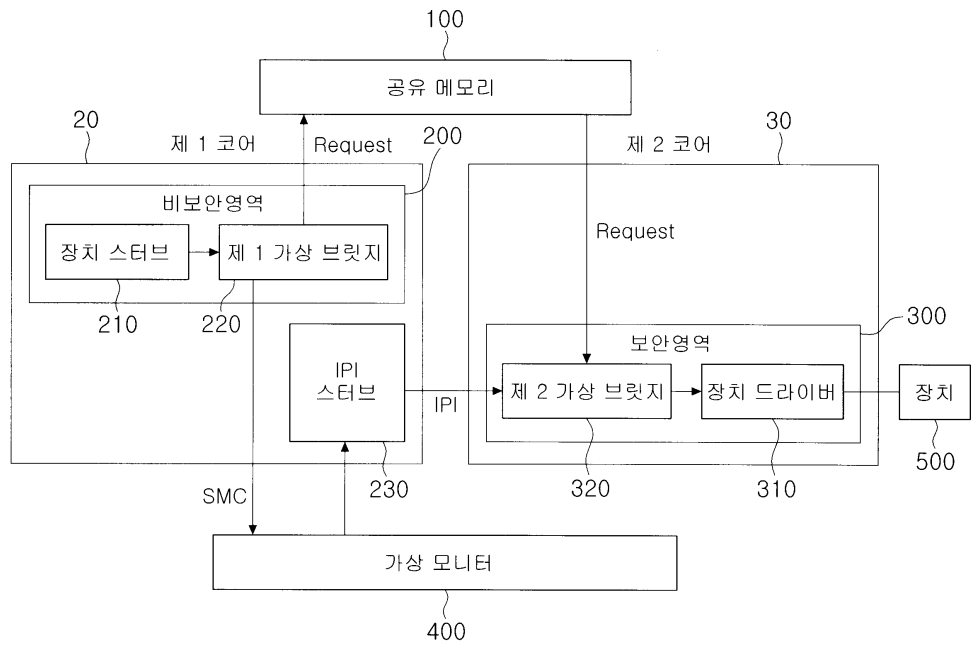
도면1



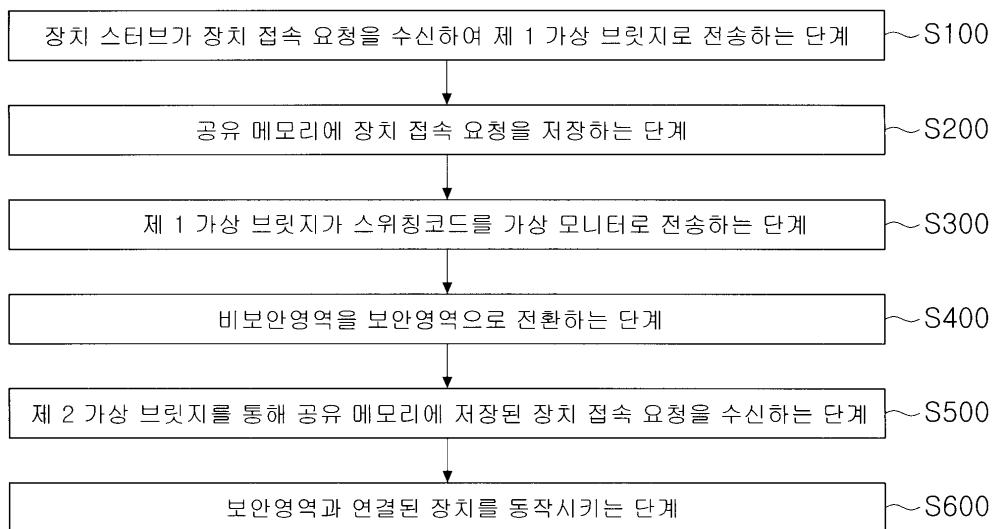
도면2



도면3



도면4



도면5

