



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2016년07월08일
 (11) 등록번호 10-1634672
 (24) 등록일자 2016년06월23일

(51) 국제특허분류(Int. Cl.)
 H04L 12/24 (2006.01) H04L 29/10 (2006.01)
 (21) 출원번호 10-2014-0182713
 (22) 출원일자 2014년12월17일
 심사청구일자 2014년12월17일
 (65) 공개번호 10-2016-0073867
 (43) 공개일자 2016년06월27일
 (56) 선행기술조사문헌
 KR1020130067906 A*
 KR1020110046895 A*
 *는 심사관에 의하여 인용된 문헌

(73) 특허권자
 (주)구름네트웍스
 서울특별시 광진구 능동로 120, 212호(화양동, 건국대학교 창의관)
 (72) 발명자
 김성민
 경기도 용인시 기흥구 강남서로 52, 302동401호(구갈동, 강남마을주공아파트)
 (74) 대리인
 유철현

전체 청구항 수 : 총 10 항

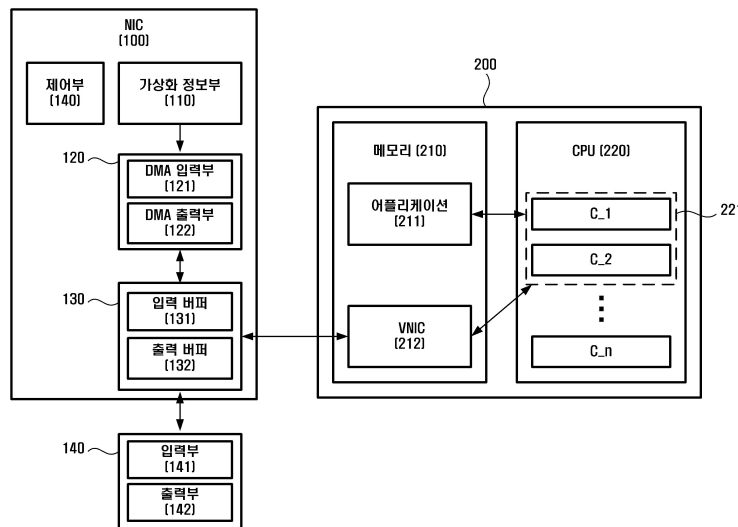
심사관 : 박보미

(54) 발명의 명칭 **네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램**

(57) 요약

네트워크 인터페이스 가상화 장치가 제공된다. 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 장치는, 컴퓨팅 디바이스의 메모리 상에 설정된 복수의 가상 네트워크 인터페이스 카드(VNIC)의 정보를 관리하는 가상화 정보부를 포함하되, 상기 가상화 정보부는, 상기 복수의 가상 네트워크 인터페이스 카드가, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션에서 점유하는 복수의 CPU코어와 일대일로 대응되도록 설정한다.

대표도 - 도1



명세서

청구범위

청구항 1

컴퓨팅 디바이스의 메모리 상에 설정된 복수의 가상 네트워크 인터페이스 카드(VNIC)의 정보를 관리하는 가상화 정보부; 및

상기 복수의 가상 네트워크 인터페이스 카드로 입력되는 복수의 패킷 데이터의 입력 설정을 관리하는 패킷 관리부를 포함하고,

상기 가상화 정보부는, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션을 상기 복수의 가상 네트워크 인터페이스 카드와 매핑하되,

상기 하나의 네트워크 어플리케이션은 복수의 CPU코어를 점유하고, 상기 복수의 CPU코어는 상기 복수의 가상 네트워크 인터페이스 카드와 일대일로 대응하고, 상기 복수의 가상 네트워크 인터페이스 카드는 모두 동일한 MAC 주소 또는 VLAN ID를 가지고,

상기 패킷 관리부는, 패킷 입력 정책에 따라 상기 입력되는 복수의 패킷 데이터를 상기 복수의 가상 네트워크 인터페이스 카드에 분배하는, 네트워크 인터페이스 가상화 장치.

청구항 2

제1항에 있어서,

상기 하나의 네트워크 어플리케이션에 의해 점유되는 상기 복수의 CPU코어와 일대일로 대응하는 상기 복수의 가상 네트워크 인터페이스 카드의 목록을 관리하는 VNIC리스트관리부를 더 포함하는, 네트워크 인터페이스 가상화 장치.

청구항 3

제1항에 있어서,

상기 패킷 관리부는, 상기 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하고, 상기 일련번호에 따라 상기 복수의 패킷 데이터가 상기 복수의 가상 네트워크 인터페이스 카드로 입력되도록 제어하는, 네트워크 인터페이스 가상화 장치.

청구항 4

제1항에 있어서,

상기 복수의 CPU코어로부터 수신된 정보에 따라 네트워크로 출력되는 복수의 패킷 데이터에 관한 패킷 출력 정책을 변경하는 정책변경관리부를 더 포함하는, 네트워크 인터페이스 가상화 장치.

청구항 5

제1항에 있어서,

상기 패킷 관리부는, 네트워크로 출력되는 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하고,

상기 네트워크와 연결되어, 상기 일련번호에 따라 상기 복수의 패킷 데이터를 상기 네트워크로 순차적으로 출력하는 출력부를 더 포함하는, 네트워크 인터페이스 가상화 장치.

청구항 6

삭제

청구항 7

컴퓨팅 디바이스의 메모리 상에 복수의 가상 네트워크 인터페이스 카드(VNIC)를 생성하는 단계;

상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션을 상기 복수의 가상 네트워크 인터페이스 카드와 매핑하는 단계로서,

상기 하나의 네트워크 어플리케이션은 복수의 CPU코어를 점유하고, 상기 복수의 CPU코어는 상기 복수의 가상 네트워크 인터페이스 카드와 일대일로 대응하고, 상기 복수의 가상 네트워크 인터페이스 카드는 모두 동일한 MAC 주소 또는 VLAN ID를 가지는, 단계; 및

패킷 입력 정책에 따라 입력되는 복수의 패킷 데이터를 상기 복수의 가상 네트워크 인터페이스 카드에 분배하는 단계를 포함하는, 네트워크 인터페이스 가상화 방법.

청구항 8

삭제

청구항 9

제7항에 있어서,

상기 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하는 단계; 및

상기 일련번호에 따라 상기 복수의 패킷 데이터가 상기 복수의 가상 네트워크 인터페이스 카드로 입력되도록 제어하는 단계를 더 포함하는, 네트워크 인터페이스 가상화 방법.

청구항 10

제7항에 있어서,

상기 복수의 CPU코어로부터 수신된 정보에 따라 네트워크로 출력되는 복수의 패킷 데이터에 관한 패킷 출력 정책을 변경하는 단계를 더 포함하는, 네트워크 인터페이스 가상화 방법.

청구항 11

제7항에 있어서,

네트워크로 출력되는 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하는 단계; 및

상기 네트워크와 연결되어, 상기 일련번호에 따라 상기 복수의 패킷 데이터를 상기 네트워크로 순차적으로 출력하는 단계를 더 포함하는, 네트워크 인터페이스 가상화 방법.

청구항 12

삭제

청구항 13

하드웨어와 결합되어, 제7항, 제9항 내지 제11항 중 어느 한 항에 따른 방법을 실행하기 위하여 컴퓨터 판독가능 매체에 저장된 컴퓨터 프로그램.

발명의 설명

기술 분야

[0001] 본 발명은 네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램에 관한 것으로, 더욱 상세하게는 복수의 가상 네트워크 인터페이스 카드를 이용하여 패킷을 고속으로 처리할 수 있는 네트워크 인터페이스 가상화 장치 및 방법에 관한 것이다.

배경 기술

[0002] 가상화 기술분야에 있어서, 호스트 가상화 기술은 수년간 급격한 발전을 이루어 기술적으로 거의 완성 단계에 이르렀다. 그러나 네트워크 가상화에 있어서, 네트워크 장치는 대부분의 기능이 H/W에 의존적이기 때문에 가상

화 하기 매우 어려우며, 패킷 데이터의 전송은 고속으로 이루어지는 특징상 S/W로 가상화 기능을 구현할 경우 가상화된 네트워크 장치의 성능이 매우 떨어진다.

[0003] 대한민국 등록특허 제10-1232211호 및 대한민국 등록특허 제10-1275293호와, 대한민국 공개특허 제10-2011-0046240호에서도 네트워크 장치의 가상화를 구현하고 있으나 가상 네트워크를 지원하는 장치의 구조에 초점을 맞추고 있다.

선행기술문헌

특허문헌

[0004] (특허문헌 0001) 대한민국 등록특허 제10-1232211호
 (특허문헌 0002) 대한민국 등록특허 제10-1275293호
 (특허문헌 0003) 대한민국 공개특허 제10-2011-0046240호

발명의 내용

해결하려는 과제

[0005] 본 발명의 목적은, 하나의 네트워크 어플리케이션이 점유하는 복수의 CPU코어가 하나의 가상 네트워크 인터페이스 카드로 접근할 때 발생하는 병목현상을 해결하기 위한 네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램을 제공하는 데 있다.

[0006] 본 발명의 다른 목적은, 복수의 가상 네트워크 인터페이스 카드의 패킷 데이터 입출력을 효율적으로 제어할 수 있는, 네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램을 제공하는 데 있다.

과제의 해결 수단

[0007] 상기 과제를 해결하기 위한 본 발명의 일 실시예에 따른, 네트워크 인터페이스 가상화 장치는, 컴퓨팅 디바이스의 메모리 상에 설정된 복수의 가상 네트워크 인터페이스 카드(VNIC)의 정보를 관리하는 가상화 정보부를 포함하되, 상기 가상화 정보부는, 상기 복수의 가상 네트워크 인터페이스 카드가, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션에서 점유하는 복수의 CPU코어와 일대일로 대응되도록 설정한다.

[0008] 상기 가상 네트워크 인터페이스 카드로 전송되는 복수의 패킷 데이터의 입출력 설정값을 관리하는 패킷 관리부를 더 포함할 수 있다.

[0009] 상기 패킷 관리부는, 패킷 입출력 정책에 따라 상기 복수의 패킷 데이터 각각에 고유한 일련번호를 부여할 수 있다.

[0010] 상기 패킷 관리부는, 상기 복수의 패킷 데이터가 상기 일련번호에 따라 순차적으로 상기 복수의 가상 네트워크 인터페이스 카드로 입력되도록 제어할 수 있다.

[0011] 네트워크와 연결되어 상기 복수의 패킷 데이터를 송수신하는 입출력부를 더 포함하되, 상기 입출력부는 상기 일련번호에 따라 상기 복수의 패킷 데이터를 순차적으로 상기 네트워크로 출력할 수 있다.

[0012] 상기 복수의 가상 네트워크 인터페이스 카드는 동일한 ID를 가질 수 있다.

[0013] 상기 과제를 해결하기 위한 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 방법은, 컴퓨팅 디바이스의 메모리 상에 복수의 가상 네트워크 인터페이스 카드(VNIC)를 생성하는 단계, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션에서 점유하는 복수의 CPU코어와, 상기 복수의 가상 네트워크 인터페이스 카드를 일대일로 대응시키는 단계를 포함한다.

[0014] 상기 복수의 가상 네트워크 인터페이스 카드로 복수의 패킷 데이터를 제공하는 단계를 더 포함할 수 있다.

[0015] 패킷 입출력 정책에 따라 상기 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하는 단계를 더 포함할 수 있다.

[0016] 상기 일련번호에 따라 상기 복수의 가상 네트워크 인터페이스 카드로 상기 복수의 패킷 데이터를 제공하는 단계

를 더 포함할 수 있다.

- [0017] 상기 복수의 가상 네트워크 인터페이스 카드로부터 상기 복수의 CPU코어에서 처리가 완료된 상기 복수의 패킷 데이터를 상기 일련번호에 따라 물리 네트워크 인터페이스 카드(NIC)로 제공하는 단계를 더 포함할 수 있다.
- [0018] 상기 복수의 CPU코어로부터 패킷 변경 요청을 수신하는 단계, 및 상기 패킷 변경 요청을 기초로 상기 복수의 가상 네트워크 인터페이스 카드 상에서의 상기 복수의 패킷 데이터의 입출력 순서를 변경하는 단계를 더 포함할 수 있다.
- [0019] 상기 과제를 해결하기 위한 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 방법을 실행하기 위한 컴퓨터 프로그램은, 컴퓨팅 디바이스의 메모리 상에 복수의 가상 네트워크 인터페이스 카드(VNIC)를 생성하는 단계, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션에서 점유하는 복수의 CPU코어와, 상기 복수의 가상 네트워크 인터페이스 카드를 일대일로 대응시키는 단계를 포함한다.

발명의 효과

- [0020] 본 발명의 실시예들에 따른 네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램은 하나의 네트워크 어플리케이션 단위로 점유되는 복수의 CPU코어별로 가상 네트워크 인터페이스 카드를 생성하여, 패킷 처리 과정에서 발생할 수 있는 병목 현상을 방지할 수 있다.
- [0021] 또한, 본 발명의 실시예들에 따른 네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램은 복수의 CPU 코어별로 가상 네트워크 인터페이스 카드를 생성함으로써, 패킷 입출력을 고속으로 처리할 수 있다.
- [0022] 뿐만 아니라, 본 발명의 실시예들에 따른 네트워크 인터페이스 가상화 장치, 방법 및 이를 실행하기 위한 컴퓨터 프로그램은 각각의 패킷에 소정의 일련번호를 부여함으로써, 복수의 가상 네트워크 인터페이스 카드를 통해 패킷을 분산처리하더라도 패킷 간의 입출력 순서를 그대로 유지할 수 있다.

도면의 간단한 설명

- [0023] 도 1은 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 장치의 블록도이다.
 도 2 내지 도 4는 도 1의 네트워크 인터페이스 가상화 장치의 병목 구조를 설명하기 위한 도면이다.
 도 5 및 도 6은 본 발명의 다른 실시예에 따른 네트워크 인터페이스 가상화 장치의 블록도이다.
 도 7은 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 장치를 통해 패킷 데이터가 분배되는 과정을 설명하기 위한 도면이다.
 도 8은 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 방법을 나타내는 순서도이다.

발명을 실시하기 위한 구체적인 내용

- [0024] 이하의 실시예들은 본 발명의 구성 요소들과 특징들을 소정 형태로 결합한 것들이다. 각 구성 요소 또는 특징은 별도의 명시적 언급이 없는 한 선택적인 것으로 고려될 수 있다. 각 구성 요소 또는 특징은 다른 구성 요소나 특징과 결합되지 않은 형태로 실시될 수 있다. 또한, 일부 구성 요소들 및/또는 특징들을 결합하여 본 발명의 실시예를 구성할 수도 있다. 본 발명의 실시예들에서 설명되는 동작들의 순서는 변경될 수 있다. 어느 실시예의 일부 구성이나 특징은 다른 실시예에 포함될 수 있고, 또는 다른 실시예의 대응하는 구성 또는 특징과 교체될 수 있다.
- [0025] 본 발명의 실시예들은 다양한 수단을 통해 구현될 수 있다. 예를 들어, 본 발명의 실시예들은 하드웨어, 펌웨어 (firmware), 소프트웨어 또는 그것들의 결합 등에 의해 구현될 수 있다.
- [0026] 하드웨어에 의한 구현의 경우, 본 발명의 실시예들에 따른 방법은 하나 또는 그 이상의 ASICs(application specific integrated circuits), DSPs(digital signal processors), DSPDs(digital signal processing devices), PLDs(programmable logic devices), FPGAs(field programmable gate arrays), 프로세서, 컨트롤러, 마이크로 컨트롤러, 마이크로 프로세서 등에 의해 구현될 수 있다.
- [0027] 펌웨어나 소프트웨어에 의한 구현의 경우, 본 발명의 실시예들에 따른 방법은 이상에서 설명된 기능 또는 동작들을 수행하는 모듈, 절차 또는 함수 등의 형태로 구현될 수 있다. 소프트웨어 코드는 메모리 유닛에 저장되어

프로세서에 의해 구동될 수 있다. 상기 메모리 유닛은 상기 프로세서 내부 또는 외부에 위치하여, 이미 공지된 다양한 수단에 의해 상기 프로세서와 데이터를 주고 받을 수 있다.

- [0028] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 직접적으로 연결되어 있는 경우뿐만 아니라, 그 중간에 다른 소자를 사이에 두고 전기적으로 연결되어 있는 경우도 포함한다. 또한, 어떤 부분이 어떤 구성 요소를 포함한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성 요소를 제외하는 것이 아니라 다른 구성 요소를 더 포함할 수 있다는 것을 의미한다.
- [0029] 또한, 본 명세서에서 기재한 모듈(module)이란 용어는 특정한 기능이나 동작을 처리하는 하나의 단위를 의미하며, 이는 하드웨어나 소프트웨어 또는 하드웨어 및 소프트웨어의 결합으로 구현할 수 있다.
- [0030] 이하의 설명에서 사용되는 특정(特定) 용어들은 본 발명의 이해를 돕기 위해서 제공된 것이며, 이러한 특정 용어의 사용은 본 발명의 기술적 사상을 벗어나지 않는 범위에서 다른 형태로 변경될 수 있다.
- [0031] 도 1 내지 도 4를 참조하면, 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 장치는, 컴퓨터 메모리(210) 상에 특별한 구조를 갖는 메모리 공간을 할당하여 그 공간과 구조를 VNIC(212)로 활용할 수 있으며, VNIC(212)는 복수로 생성될 수 있다. 복수로 생성된 VNIC(212)는 CPU(220)의 코어들(221)과 매칭되어, 코어(C_n, n은 정수)와 매칭될 수 있다. 본 명세서에서 메모리(210)와 CPU(220)는 물리적인 실제 리소스를 의미할 수 있으나, 이에 한정되는 것은 아니며 물리머신 상에 생성된 가상머신을 구성하는 가상 메모리와 가상 CPU를 의미할 수도 있다.
- [0032] 또한, 몇몇 다른 실시예에서 하나의 네트워크 어플리케이션(211)이 접근하는 CPU(220) 내 코어들(221)을 하나의 VNIC(212)와 매핑할 수 있다.
- [0033] 즉, 도 3에 도시된 바와 같이, 하나의 네트워크 어플리케이션(211)이 각각 하나의 VNIC(212)와 매핑될 수 있다. 예를 들어, 메모리(210) 상에 로드된 어플리케이션_1(211_1)은 3개의 코어(C₁, C₂, C₃)로 구성된 제1 CPU 그룹(221)을 점유하여 사용하고, 어플리케이션_2(211_2)와 어플리케이션 3(211_3)은 각각 2개의 코어(C₄, C₅)로 구성된 제2 CPU 그룹(222)와, 1개의 코어(C₆)로 구성된 제3 CPU 그룹(223)을 점유하여 사용한다고 할 때, 네트워크 어플리케이션(211) 단위로 독립적인 패킷 데이터 처리가 가능해지며, 독립적인 싱글코어의 경우에는 1개의 코어(C₆)가 하나의 VNIC(212)에 접근하게 되므로 순차적으로 패킷 데이터의 처리가 가능하다.
- [0034] 다만, 도 2 및 도 4에서와 같이, 멀티코어를 사용하는 네트워크 어플리케이션의 경우, 즉, 특정한 네트워크 어플리케이션(211)이 복수의 코어들(221)을 점유하여 동시에 사용하는 경우 일시에 하나의 VNIC(212)에 패킷 데이터 처리를 위해 접근하게 되면, 교착 내지 병목 현상이 발생할 수도 있다.
- [0035] 도 5 및 도 6를 참조하여, 본 발명의 다른 실시예에 따른 네트워크 인터페이스 가상화 장치에 대해 설명한다. 도 5를 참조하면, 본 발명의 다른 실시예에 따른 네트워크 인터페이스 가상화 장치는, 컴퓨팅 디바이스(200)의 메모리(210) 상에 설정된 복수의 가상 네트워크 인터페이스 카드(VNIC)가 네트워크 어플리케이션(211) 단위로 생성 및 관리되는 것이 아니라, 네트워크 어플리케이션(211)이 점유하는 복수의 CPU코어와 일대일로 대응된다. 예를 들어, 메모리(210) 상에 로드된 어플리케이션_1(211_1)은 3개의 코어(C₁, C₂, C₃)로 구성된 제1 CPU 그룹(221)을 점유하여 사용하는데, 3개의 코어(C₁, C₂, C₃)는 각각 VNIC₁(212₁), VNIC₂(212₂), VNIC₃(212₃)와 연결되도록, NIC(100) 내의 가상화 정보부(110)에서 VNIC정보를 관리할 수 있다. 마찬가지로 어플리케이션_2(211_2)는 2개의 코어(C₄, C₅)로 구성된 제2 CPU 그룹(222)을 점유하는데, 2개의 코어(C₄, C₅)가 각각 VNIC₄(212₄), VNIC₅(212₅)와 연결되도록, 가상화 정보부(110)에서 VNIC 정보를 관리할 수 있다.
- [0036] 몇몇 다른 실시예에서, 네트워크 어플리케이션(211)이 점유하는 복수의 CPU코어 1개당, VNIC가 2개 이상 배치될 수도 있다.
- [0037] 또한, 몇몇 다른 실시예에서, 하나의 네트워크 어플리케이션(211)이 점유하는 복수의 코어와 대응되는 복수의 가상 네트워크 인터페이스 카드는 서로 동일한 ID 예를 들어 동일한 MAC주소나 동일한 VLAN ID를 가질 수 있다.
- [0038] 도 6을 참조하면, 본 실시예에 따른 네트워크 인터페이스 가상화 장치는, 제어부(140), 가상화 정보부(110), DMA 입력부(121)와 DMA 출력부(122)를 구비한 DMA 입출력부(120), 입력 버퍼부(131)와 출력 버퍼부(132)를 구비한 입출력 버퍼부(130), 및 입력부(141)와 출력부(142)를 구비한 입출력부(140)를 포함한다. 여기서, 입출력 버퍼부(130)는, 각각 하나씩의 입력 버퍼부(131)와, 출력 버퍼부(132)를 구비할 수 있으며, 컴퓨팅 디바이스(200)의 메모리(210) 상에 생성 및 관리되는 복수의 가상 네트워크 인터페이스 카드(VNIC)(212)와 관련된 메타

데이터를 생성, 저장 및 관리하는 메타데이터 관리부(150)를 더 포함할 수 있다.

- [0039] 우선, 상기 가상화 장치가 장착되어 구현되거나 연동할 컴퓨터(200)는 적어도 CPU(220)와 메모리(210)를 구비하고, 복수 개의 가상 네트워크 인터페이스 카드(Network Interface Card, NIC)(212)가 영역이 메모리(210) 상에 미리 설정된다.
- [0040] 가상화 정보부(110)는 가상 NIC의 MAC 주소 또는 VLAN 정보, 입력 ring 버퍼의 주소 목록, 입출력 대역폭, 입출력 가능한 MAC 주소 목록 및 NIC의 상태 정보를 포함하는 NIC 가상화 정보가 저장된다. 여기서, 가상화 정보부는 네트워크 인터페이스 카드에 탑재된 메모리로 구현될 수 있다.
- [0041] 입력부(141)는 네트워크로부터 패킷 데이터를 전송받아 입력 버퍼부(131)에 상기 패킷 데이터를 쓰고, 저장 알림 신호를 생성하여 제어부(140)에 출력한다. 입력 버퍼부(131)는 입력부(141)에서 전송된 여러 개의 패킷 데이터를 임시로 보관한다. DMA 입력부(121)는 가상화 정보부(110)의 정보를 기반으로 입력 버퍼부(131)의 패킷 데이터를 컴퓨터의 메모리(210)로 전송한다.
- [0042] DMA 출력부(122)는 가상화 정보부(110)의 정보를 기반으로 컴퓨터 메모리(210) 상의 패킷 데이터를 출력 버퍼부(132)로 복사하며, 출력 버퍼부(132)는 출력부(142)가 패킷 데이터를 출력하도록 상기 패킷 데이터를 임시로 보관한다. 출력부(142)는 출력 버퍼부(132)의 패킷 데이터를 네트워크로 전송한다.
- [0043] 제어부(140)는 상기 저장 알림 신호와 상기 NIC 가상화 정보에 기반하여 가상화 정보부(110), DMA 입출력부(121, 122), 입출력 버퍼부(131, 132) 및 입출력부(141, 142)를 제어하기 위한 제어신호를 출력한다.
- [0044] 메타데이터 관리부(150)는, VNIC(212)로 입출력되는 복수의 패킷 데이터의 입출력 설정값을 관리하는 패킷 관리부(160)를 더 포함할 수 있으며, 복수의 VNIC(212) 중에서 하나의 네트워크 어플리케이션(211)에서 점유하는 코어들과 대응되는 VNIC(212)에 대한 목록을 관리하는 VNIC리스트관리부(170)를 더 포함할 수 있다. 즉, 이전 실시예에서는 네트워크 어플리케이션(211) 당 하나의 VNIC(212)만 존재하기 때문에 별도의 리스트를 관리할 필요가 없었으나, 본 실시예에서는 코어별로 1개 이상의 VNIC(212)가 할당될 수 있기 때문에 이에 대한 리스트를 관리할 수 있다.
- [0045] 패킷관리부(160)는, 패킷 입출력 정책에 따라 복수의 패킷 데이터 각각에 고유한 일련번호를 부여할 수 있으며, 복수의 패킷 데이터가 일련번호에 따라 순차적으로 복수의 VNIC(212)로 입력되도록 제어할 수 있다.
- [0046] 입력정책정보부(161)는, 패킷 데이터가 입력되었을 때 어떤 정책에 따라 여러개의 VNIC에 분배하는지에 대한 정보를 저장할 수 있다. 입력순서관리부(162)는 패킷 데이터 입력시 패킷 데이터마다 부여하는 일련번호를 저장할 수 있다.
- [0047] 패킷 데이터 입력 정책은 예를 들어, CPU코어를 순서대로 돌아가며 패킷 데이터를 입력하는 라운드 로빈(Round-robin) 방식, CPU 코어 각각에 비중을 두어서 해당 비중에 따라 순서대로 돌아가며 패킷 데이터를 입력하는 비중 라운드 로빈(Weighted Round-robin) 방식, CPU 코어의 입력버퍼나 출력버퍼 또는 이를 조합한 패킷의 개수 중 가장 적은 입력버퍼에 패킷 데이터를 집어넣는 버퍼 최소 패킷(Buffer Least Packet) 방식, 버퍼 최소 패킷 방식에 비중을 고려하는 비중 버퍼 최소 패킷(Weighted Buffer Least Packet)방식, 및 소스 MAC주소를 해싱(Hashing)된 값에 따라 CPU코어의 입력 버퍼에 패킷을 집어넣는 소스 해싱(Source Hashing)방식이 있을 수 있으나, 이에 한정되는 것은 아니며 패킷 데이터를 적절히 각각의 VNIC에 분산하는 다른 방법이 사용될 수도 있다.
- [0048] 출력정책정보부(163)는 CPU 코어에서 처리된 패킷 데이터를 VNIC(212)에서 NIC(100)로 출력할 때 어떤 정책에 따라 패킷 데이터를 출력하는지에 대한 정보를 저장할 수 있다. 출력순서관리부(164)는 패킷 데이터의 출력 정책이 순차출력 방식인 경우, 어떤 일련번호에 해당하는 패킷 데이터를 출력할 것인지에 대한 정보를 관리할 수 있다. 즉, 네트워크와 연결되어 복수의 패킷 데이터를 송수신하는 입출력부(140)는 출력 정책 및 일련번호에 따라 복수의 패킷 데이터를 순차적으로 네트워크로 출력할 수도 있다.
- [0049] 패킷 데이터 출력 정책은 앞서 언급한 패킷 데이터 입력 정책과 유사하며, 이에 추가로 패킷 입출력을 담당하는 NIC장치에서 부여한 출력 시퀀스 번호(Output Sequence Number)에 해당하는 패킷을 찾아서 이를 NIC(100)에 출력하고, 시퀀스 번호를 1만큼 증가시켜 출력 시퀀스 번호에 맞도록 패킷을 출력하는 시퀀스(Sequence) 방식이 있을 수 있다.
- [0050] 정책변경관리부(165)는 처리된 패킷 데이터를 VNIC(212)에서 NIC(100)로 출력할 때 정책이 변경될 경우 이에 대한 정보를 저장할 수 있다. 예를 들어, CPU코어가 패킷의 출력 시퀀스 번호(Output Sequence Number), 패킷 출력 정책(Packet Output Policy), 및 차기 출력 시퀀스 번호(Next Output Sequence Number)를 VNIC(212)에게 보

내면, VNIC(212)는 해당 정보를 정책변경관리부(165)에 저장한다. 이어서, VNIC(212)는 출력 시퀀스 번호와 동일한 번호를 가진 패킷 데이터를 NIC(100)로 출력하고, 출력 이후 VNIC(212)의 정책을 해당 패킷 출력 정책으로 변경하고, 출력 시퀀스 번호를 수신한 차기 출력 시퀀스 번호로 변경할 수 있다. 이와 같은 정책변경관리부(165)는 순차적이지 않도록 패킷을 출력하는 도중에 순차적으로 새로운 무리의 패킷을 출력하는데 사용될 수 있다.

[0051] 도 7을 참조하면, 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 장치를 통해 패킷 데이터가 분배되는 과정이 도시된다.

[0052] 도 7을 참조하면, 먼저, 입력부(141)를 통해 외부 네트워크로부터 입력된 패킷 데이터(P)는 입출력버퍼(130)에 입력되고, 상기한 패킷 데이터 입력 정책에 따라 패킷관리부(160)가 해당 패킷 데이터(P)에 입력 시퀀스 번호(P6)를 할당하고, 패킷 데이터 입력 정책에 따라 해당 어플리케이션(211_2)의 코어들(222)과 적어도 1대1로 매핑되는 복수의 VNIC(212_2, 212_3, 212_4)로 패킷 데이터를 입력할 수 있다. 도시된 예에서는, 라운드 로빈 방식이 패킷 입력 정책인 경우를 예로 들어, 입력 시퀀스 번호 6번인 패킷(P6)이 비어있는 VNIC_4(212_4)로 제공될 수 있다.

[0053] 도시되지는 않았으나, 반대로 VNIC(212)로부터 NIC(100)로 출력되는 패킷 데이터도 패킷 출력 정책에 따라 패킷의 출력 시퀀스 번호 등을 기초로 출력 순서를 결정할 수도 있다.

[0054] 이하에서는, 도 8을 참고하여, 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 방법에 대해 설명한다.

[0055] 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 방법은, 컴퓨팅 디바이스의 메모리 상에 복수의 가상 네트워크 인터페이스 카드(VNIC)를 생성하는 단계, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션에서 점유하는 복수의 CPU코어와, 상기 복수의 가상 네트워크 인터페이스 카드를 일대일로 대응시키는 단계를 포함한다.

[0056] **NIC 가상화 정보 생성 단계(S100)**

[0057] 앞서 언급한 바와 같이, 본 실시예에 따른 네트워크 인터페이스 가상화 방법은, 컴퓨팅 디바이스(200)의 메모리(210) 상에 설정된 복수의 가상 네트워크 인터페이스 카드(VNIC)가 네트워크 어플리케이션(211)이 점유하는 복수의 CPU코어(C_n)와 일대일로 대응될 수 있다. 즉, 메모리(210) 상에 복수 개의 VNIC(212)이 구현될 수 있으며, 이러한 VNIC의 개수는 컴퓨터에 구비된 CPU 코어(C_n)의 개수와 같을 수 있다. 즉, 컴퓨팅 디바이스(200)에 구현된 복수의 CPU 코어(221) 각각에 대응하여 메모리(210)상에 복수의 VNIC(212)이 구현될 수 있으며, 하나의 VNIC(212)에는 하나의 CPU 코어(C_n)가 매칭될 수 있다. 이러한 VNIC(212)은 NIC(110)로부터 수신된 데이터 패킷을 임시 저장하기 위한 입력버퍼부 및 해당 VNIC과 연동되는 CPU 코어(C_n)에 의해 처리된 데이터 패킷을 임시 저장하는 출력 버퍼부를 포함할 수 있다. NIC(110)는 예를 들어, CPU 코어의 0번이거나 VNIC를 지원하는 네트워크 인터페이스 카드일 수 있다.

[0058] CPU(220)는 제어부(140)를 통하여 가상화 정보부(11)에 위와 같이 생성된 복수의 VNIC(212)의 MAC 주소 또는 VLAN 정보, 입력 ring 버퍼의 주소 목록, 입출력 대역폭, 입출력 가능한 MAC 주소 목록 및 NIC의 상태 정보를 입력할 수 있다(S100).

[0059] **패킷 데이터 입력 단계(S200)**

[0060] NIC(100)에 구비된 입력부(131)는 네트워크로부터 패킷 데이터를 읽어 입력 버퍼부(121)에 쓰고 제어부(140) 알린다. 제어부(140)는 입력 버퍼부(121) 안에 있는 패킷 데이터의 Destination MAC 또는 VLAN 정보와 가상화 정보부(110)의 정보를 참고해 가상 NIC를 선택하고, 상기 선택된 가상 NIC의 입력 ring 버퍼 주소를 얻는다. 제어부(140)는 Destination MAC이 멀티캐스트(multicast)이면 기설정된 MAC 주소 또는 기설정된 VLAN ID를 갖는 가상 NIC를 선택한다. 여기서, 일예로 기설정된 MAC 주소는 ff:ff:ff:ff:ff:ff 이고, 기설정된 VLAN ID는 4095가 될 수 있다.

[0061] 제어부(140)는 여분의 패킷 ring 버퍼 주소가 없을 경우 패킷 데이터를 버린다. 또한, 제어부(140)는 Destination MAC 또는 VLAN 정보에 해당되는 가상 NIC이 없을 경우에도 패킷 데이터를 버린다. 또한, 제어부(140)는 가상 NIC의 입력 대역폭을 참고해 패킷 데이터의 대역폭이 할당된 가상 NIC의 입력 대역폭을 초과하는 경우에도 패킷 데이터를 버린다. 제어부(140)는 패킷 데이터에 포함된 MAC 주소가 가상 NIC의 입출력 가능한 MAC 주소 목록에 없는 경우에도 패킷 데이터를 버린다.

[0062] 제어부(140)는 DMA 출력부(122)를 통해 가상 NIC의 입력 ring 버퍼 주소에 패킷 데이터를 쓰고, 입력 버퍼부에서 해당 패킷 데이터를 삭제하고, VNIC의 입력 대역폭 정보를 갱신한다.

[0063] 이때, 메타데이터 관리부(150)에 포함된 패킷관리부(160)의 패킷 입출력 정책에 따라 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하고, 복수의 패킷 데이터가 일련번호에 따라 순차적으로 또는 특정한 순서에 따라 복수의 VNIC(212)로 입력될 수 있다.

[0064] 예를 들어, 입력정책정보부(161)에서 설정한 입력정책이 라운드로빈(round robin) 방식일 경우, 하나의 네트워크 어플리케이션(211)에서 점유하는 CPU 코어들(221) 각각에 대응되는 복수의 VNIC(212)에 번갈아가면서 순차적으로 패킷을 입력할 수 있다. 이때, 입력순서관리부(162)에서 패킷 데이터마다 부여되는 일련번호를 저장하여 관리할 수 있으며, 서로 다른 네트워크 어플리케이션(211)에서 점유하는 다른 그룹의 CPU 코어(C_n) 사이에서는 위와 같은 입력정책이 적용되지 않을 수 있다. 예를 들어, 4개의 CPU 코어(C₁, C₂, C₃, C₄)가 라운드 로빈 방식으로 패킷 데이터를 입력받는 경우, 1번 패킷부터 4번 패킷까지 순서대로 입력되고, 그 이후에 다시 4개의 코어에 5번부터 8번 패킷까지 순서대로 입력될 수 있다.

[0065] 위와 같이, 복수의 가상 네트워크 인터페이스 카드(VNIC)로 복수의 패킷 데이터를 제공하고 중첩 할당되지 않도록 설정되어 각각에 할당된 VNIC(212)로 CPU 코어(C_n)가 접근하여 신속하게 소정의 작업을 처리할 수 있다. 또한, 본 과정에서 앞에서 상술한 바와 같이, 패킷 입출력 정책에 따라 복수의 패킷 데이터 각각에 고유한 일련번호를 부여할 수 있으며, 소정의 일련번호에 따라 복수의 가상 네트워크 인터페이스 카드(VNIC)로 복수의 패킷 데이터를 제공할 수 있다.

[0066] **패킷 데이터 출력 단계(S300)**

[0067] 제어부(140)는 모든 VNIC(212)에 대해 아래의 과정을 수행하며, VNIC(212)의 대역폭에 여유가 있을 경우 아래의 과정을 실행한다.

[0068] 제어부(140)는 DMA 출력부(122)를 통해 출력 ring 버퍼에서 패킷 데이터를 출력 버퍼(132)로 복사하고, CPU(220)에 패킷 데이터를 가져갔다라는 정보를 전달한다. VNIC(212)로 제공된 패킷 데이터는 각각의 VNIC(212)와 일대일로 대응되는 CPU 코어(C_n)에서 작업을 완료하면, 출력정책정보부(163)에서 설정한 패킷 데이터 출력 정책에 따라 출력순서관리부(164)의 순서를 기초로 처리가 완료된 패킷 데이터에 대응하는 패킷을 각각의 VNIC(212)에서 NIC(100)로 전송할 수 있다. 예를 들어, 출력정책이 시퀀스(sequence) 방식인 경우, NIC(100)에서 부여한 출력 시퀀스 번호에 해당하는 패킷을 순차적으로 출력하고, 출력 시퀀스 번호를 하나씩 가산하여 출력 패킷 순서를 관리할 수 있다.

[0069] 제어부(140)는 패킷 데이터의 Destination MAC이 상기 입출력 가능한 MAC 주소 목록에 없으면 패킷 데이터를 버린다. 제어부(140)는 출력부(142)에 패킷 데이터를 출력하라고 명령하고, 가상 NIC의 출력 대역폭 정보를 갱신한다. 출력부(142)는 출력 버퍼부(132)의 패킷 데이터를 네트워크로 출력한 후 출력 버퍼부(132)에서 해당 패킷 데이터를 삭제한다.

[0070] 또한 몇몇 다른 실시예에서, 복수의 VNIC로부터 복수의 CPU코어에서 처리가 완료된 복수의 패킷 데이터를 앞서 설정한 일련번호에 따라 물리 네트워크 인터페이스 카드(NIC)로 제공하는 단계를 더 포함할 수 있다.

[0071] 또한, 패킷 정책을 변경하여 출력하고자 하는 경우, 복수의 CPU코어로부터 패킷 변경 요청을 수신하고, 패킷 변경 요청을 기초로 복수의 VNIC 상에서의 상기 복수의 패킷 데이터의 입출력 순서를 변경하는 단계를 더 포함할 수 있다. 구체적으로, CPU 코어(C_n)가 패킷의 출력 시퀀스 번호(output sequence number), 수정 패킷 출력 정책(next packet output policy) 및 수정 출력 시퀀스 번호(next packet sequence number)를 VNIC(212)에 보내면, VNIC(212)는 해당 정보를 정책 변경 큐(Policy Change Queue)에 추가할 수 있다. VNIC(212)는 정책 변경 큐의 첫번째 명령에 해당하는 패킷의 출력 시퀀스 번호와 동일한 패킷을 출력한 이후, VNIC(212)의 정책을 CPU 코어(C_n)로부터 수신한 수정 패킷 출력 정책으로 변경하여, VNIC(212)와 NIC(100) 사이의 패킷 입출력 순서를

설정할 수 있다. 이어서, VNIC(212)의 출력 시퀀스 번호를 수정 출력 시퀀스 번호로 변경하여 다음번 출력 패킷의 순서를 수정할 수 있다.

[0072] **NIC 가상화 정보 삭제 단계(S400)**

[0073] CPU(220)는 가상화 정보부(110)에 생성한 NIC 가상화 정보를 삭제한다.

[0074] 본 발명의 일 실시예에 따른 네트워크 인터페이스 가상화 방법을 실행하기 위한 컴퓨터 프로그램은, 컴퓨팅 디바이스의 메모리 상에 복수의 가상 네트워크 인터페이스 카드(VNIC)를 생성하는 단계, 상기 컴퓨팅 디바이스에서 실행되는 하나의 네트워크 어플리케이션에서 점유하는 복수의 CPU코어와, 상기 복수의 가상 네트워크 인터페이스 카드를 일대일로 대응시키는 단계를 포함한다.

[0075] 또한, 컴퓨터 프로그램은, 상기 복수의 가상 네트워크 인터페이스 카드로 복수의 패킷 데이터를 제공하는 단계를 더 포함할 수 있다.

[0076] 또한, 컴퓨터 프로그램은, 패킷 입출력 정책에 따라 상기 복수의 패킷 데이터 각각에 고유한 일련번호를 부여하는 단계를 더 포함할 수 있다.

[0077] 또한, 컴퓨터 프로그램은, 상기 일련번호에 따라 상기 복수의 가상 네트워크 인터페이스 카드로 상기 복수의 패킷 데이터를 제공하는 단계를 더 포함할 수 있다.

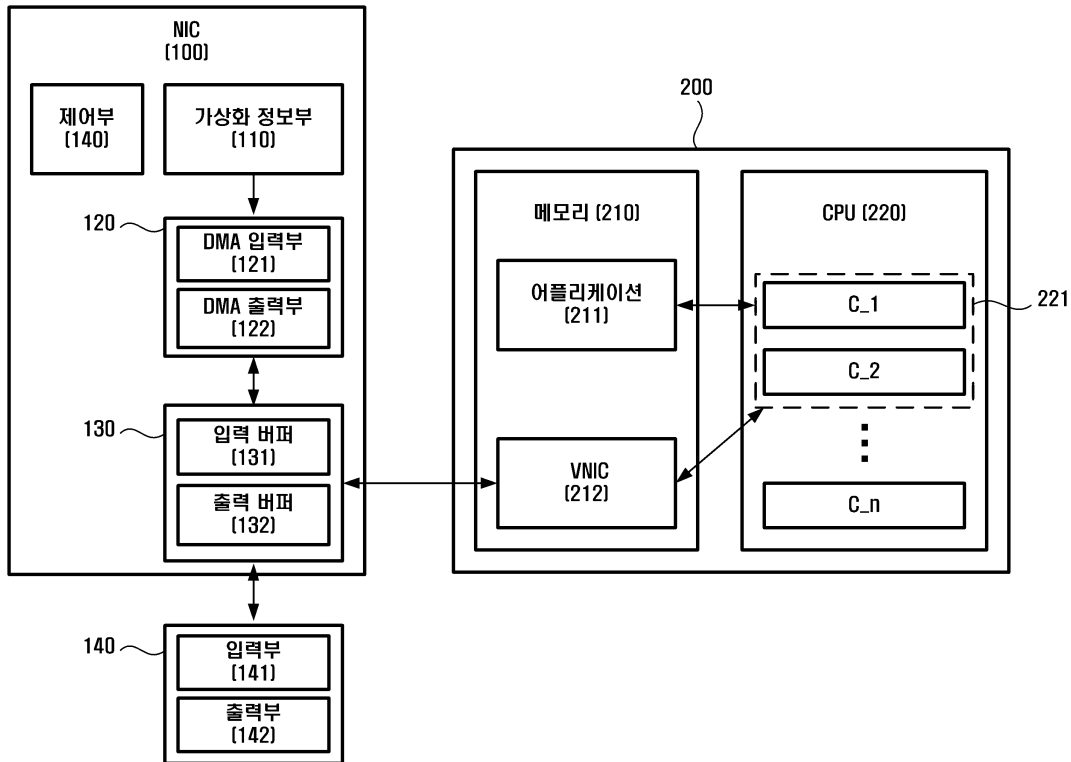
[0078] 또한, 컴퓨터 프로그램은, 상기 복수의 가상 네트워크 인터페이스 카드로부터 상기 복수의 CPU코어에서 처리가 완료된 상기 복수의 패킷 데이터를 상기 일련번호에 따라 물리 네트워크 인터페이스 카드(NIC)로 제공하는 단계를 더 포함할 수 있다.

[0079] 또한, 컴퓨터 프로그램은, 상기 복수의 CPU코어로부터 패킷 변경 요청을 수신하는 단계, 및 상기 패킷 변경 요청을 기초로 상기 복수의 가상 네트워크 인터페이스 카드 상에서의 상기 복수의 패킷 데이터의 입출력 순서를 변경하는 단계를 더 포함할 수 있다.

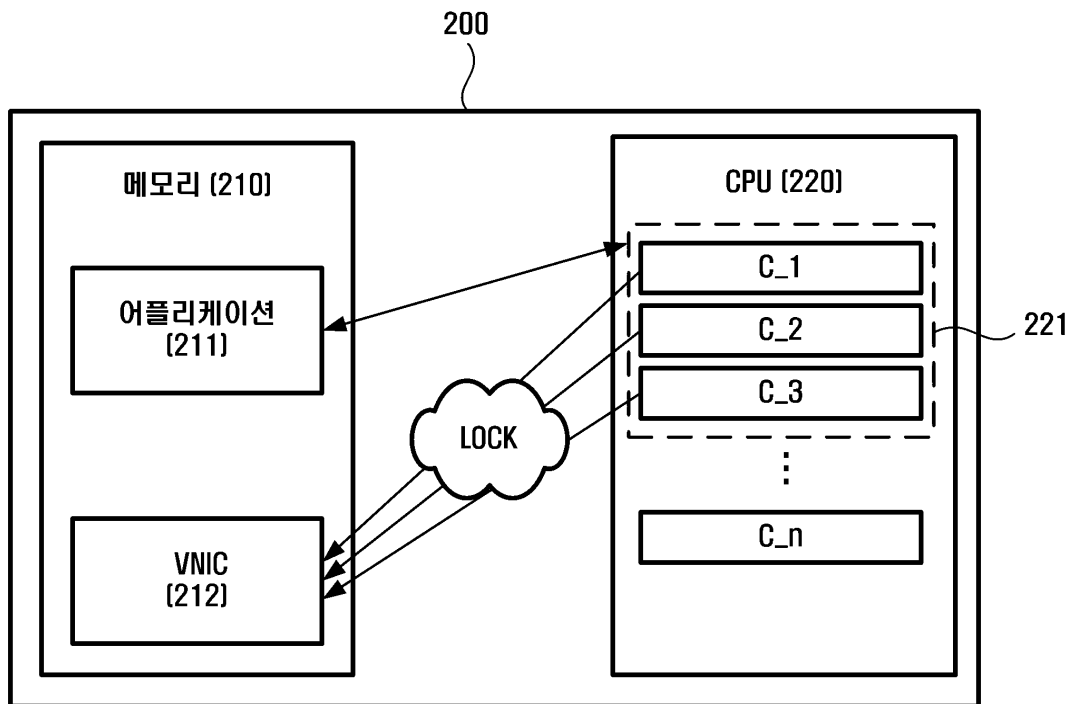
[0080] 이상의 설명은 본 발명의 기술 사상을 예시적으로 설명한 것에 불과한 것으로서, 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자라면 본 발명의 본질적인 특성에서 벗어나지 않는 범위 내에서 다양한 수정, 변경 및 치환이 가능할 것이다. 따라서, 본 발명에 개시된 실시예 및 첨부된 도면들은 본 발명의 기술 사상을 한정하기 위한 것이 아니라 설명하기 위한 것이고, 이러한 실시예 및 첨부된 도면에 의하여 본 발명의 기술 사상의 범위가 한정되는 것은 아니다. 본 발명의 보호 범위는 아래의 청구범위에 의하여 해석되어야 하며, 그와 동등한 범위 내에 있는 모든 기술 사상은 본 발명의 권리범위에 포함되는 것으로 해석되어야 할 것이다. 또한, 특허청구 범위에서 명시적인 인용 관계가 있지 않은 청구항들을 결합하여 실시예를 구성하거나 출원 후의 보정에 의해 새로운 청구항으로 포함시킬 수 있다.

도면

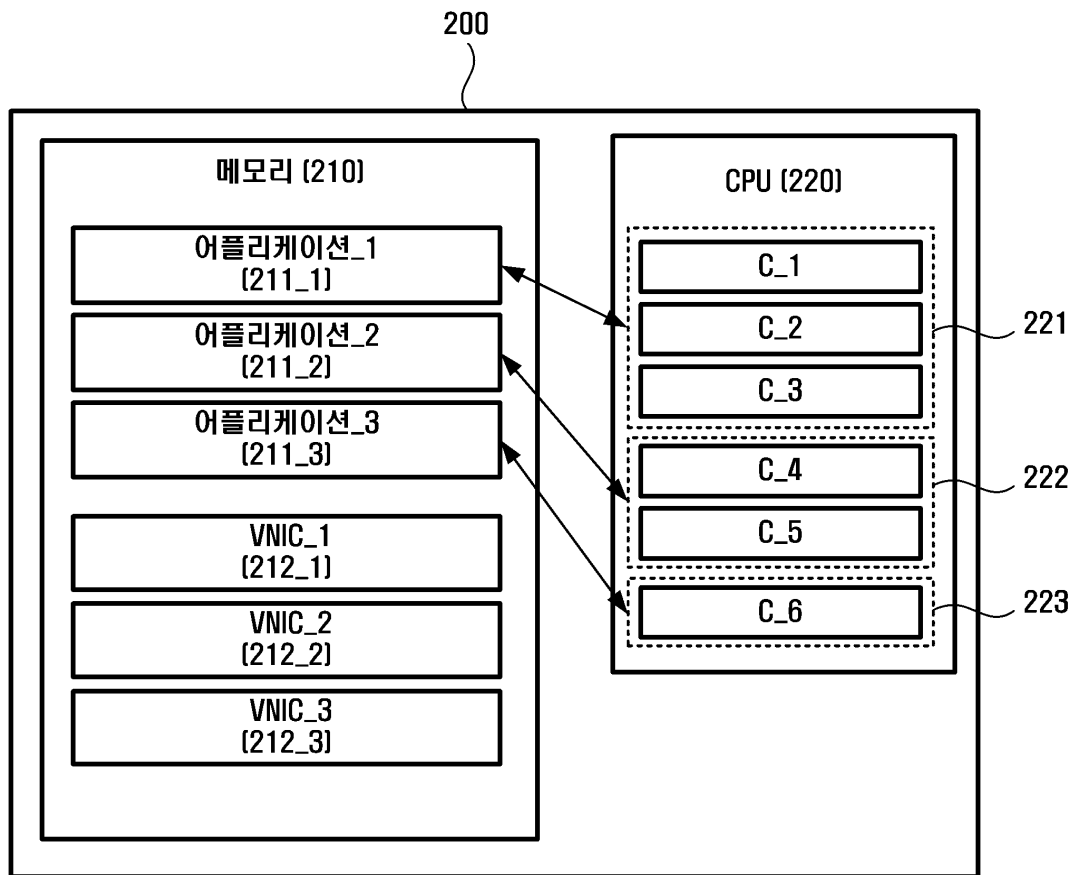
도면1



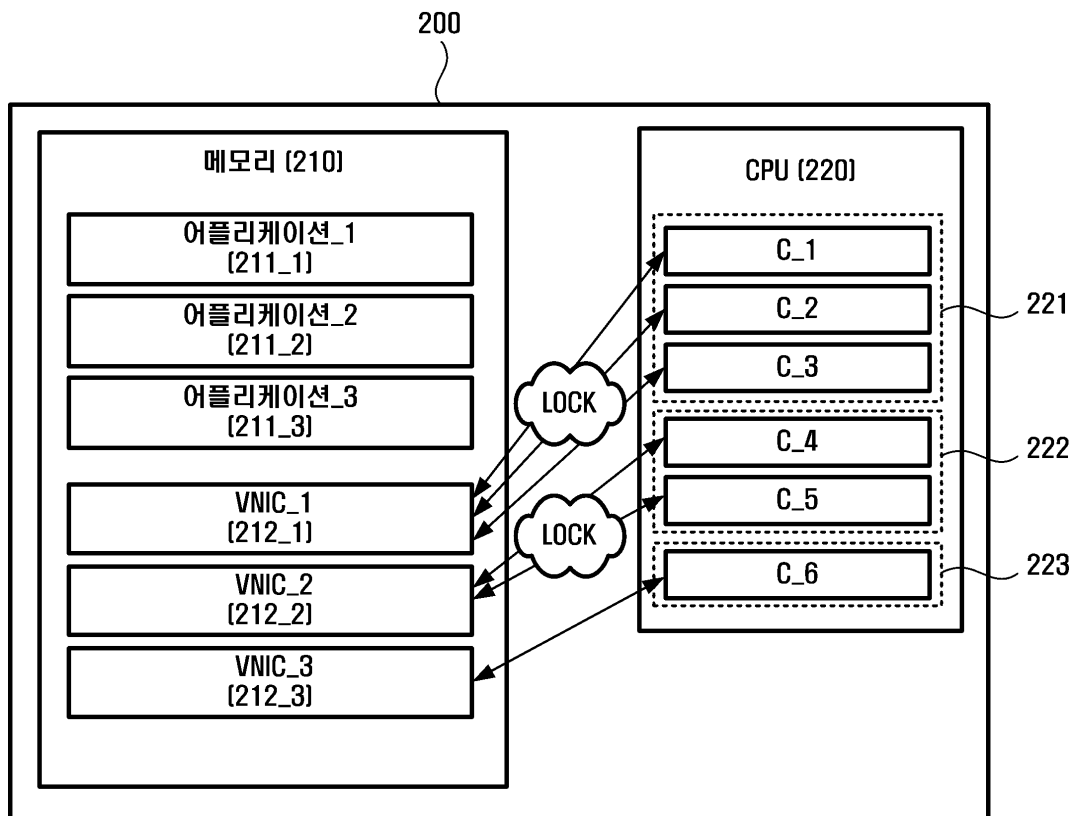
도면2



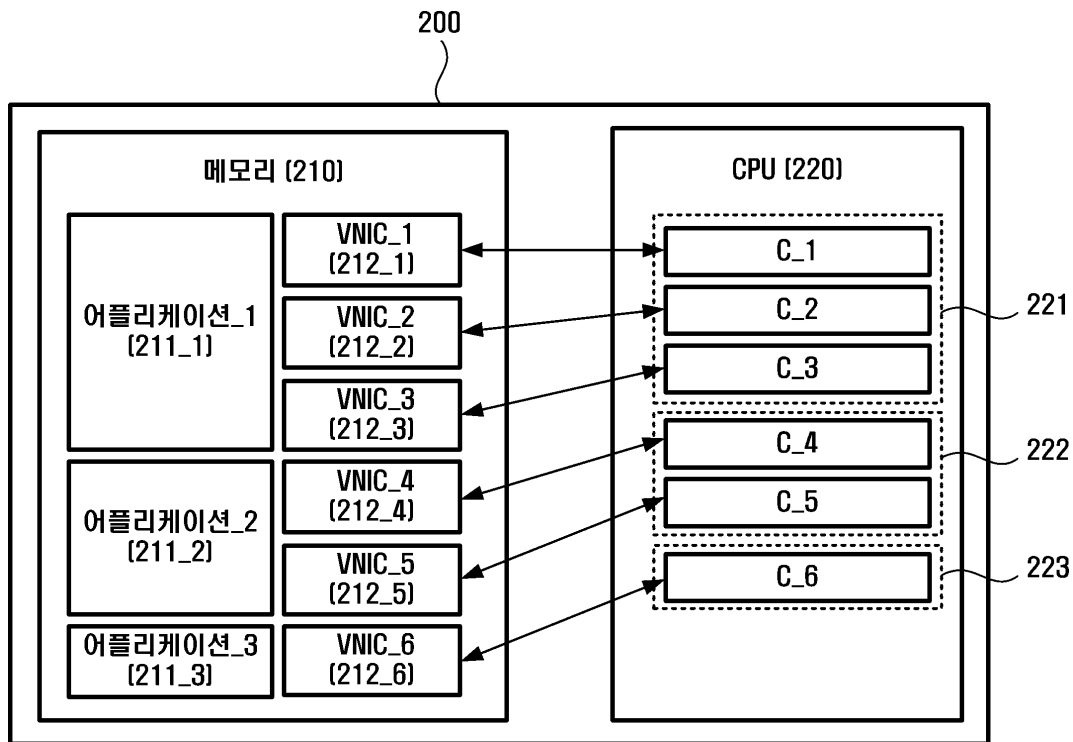
도면3



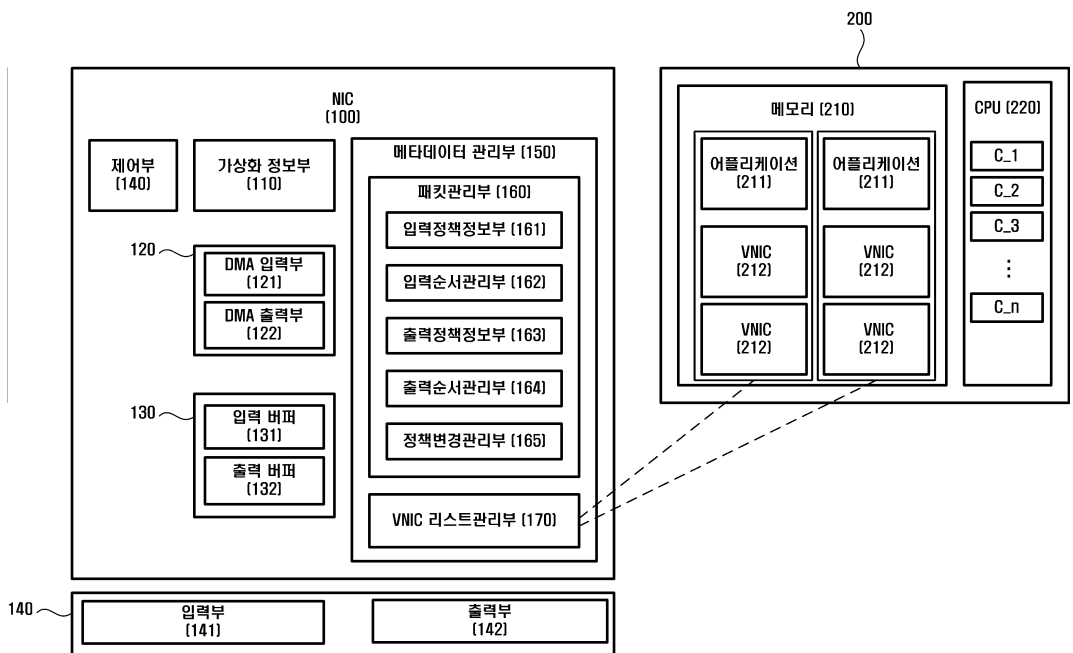
도면4



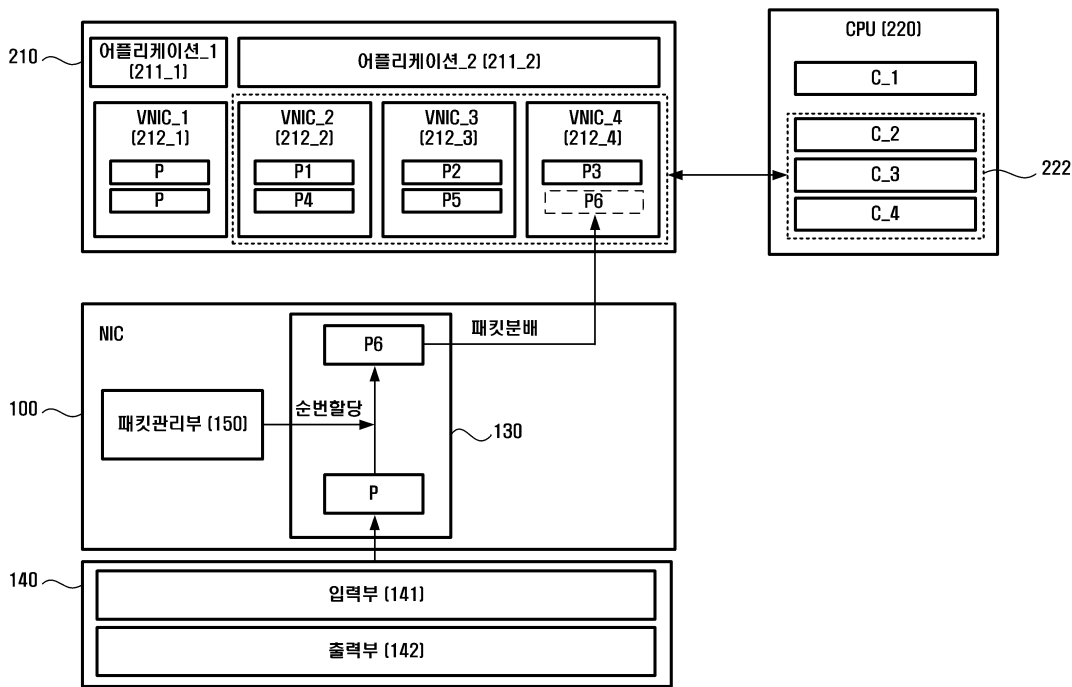
도면5



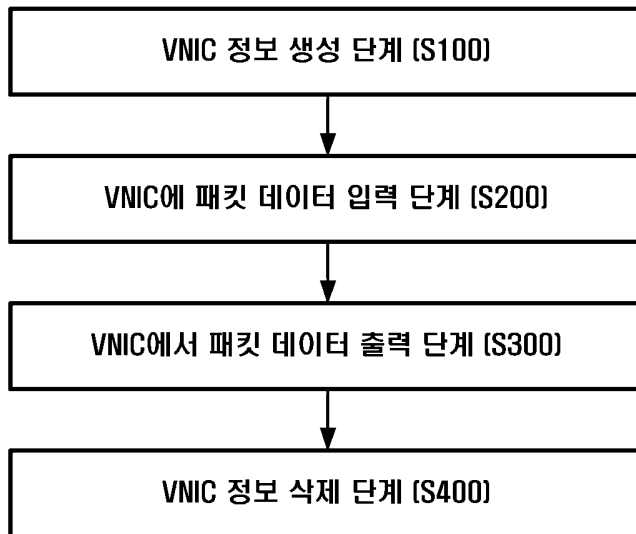
도면6



도면7



도면8



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 7

【변경전】

상기 입력되는

【변경후】

입력되는

【직권보정 2】

【보정항목】 청구범위

【보정세부항목】 청구항 13

【변경전】

제7항 내지 제12항 중

【변경후】

제7항, 제9항 내지 제11항 중