(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2010년11월05일
(11) 등록번호 10-0992282
(24) 등록일자 2010년10월29일

(51) Int. Cl.
G06F 15/173 (2006.01) *H04L 12/56* (2006.01)
H04L 29/06 (2006.01) *G06F 15/16* (2006.01)
(21) 출원번호 10-2007-7001454
(22) 출원일자(국제출원일자) 2005년06월23일
심사청구일자 2008년03월31일
(85) 번역문제출일자 2007년01월19일
(65) 공개번호 10-2007-0042152
(43) 공개일자 2007년04월20일
(86) 국제출원번호 PCT/US2005/022348
(87) 국제공개번호 WO 2006/019512
국제공개일자 2006년02월23일
(30) 우선권주장
10/890,710 2004년07월14일 미국(US)
(56) 선행기술조사문헌
US20030046330 A1*
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
인터내셔널 비지네스 머신즈 코포레이션
미국 10504 뉴욕주 아몬크 뉴오차드 로드
(72) 발명자
프레이무트 더글라스 엠
미국 10025 뉴욕주 뉴욕 아파트먼트 8와이 웨스트 센트럴 파크400
후 엘버트 씨
미국 11373-2982 뉴욕주 엘머스트 애비뉴 76-26 47
(뒷면에 계속)
(74) 대리인
김창세, 장성구, 김원준

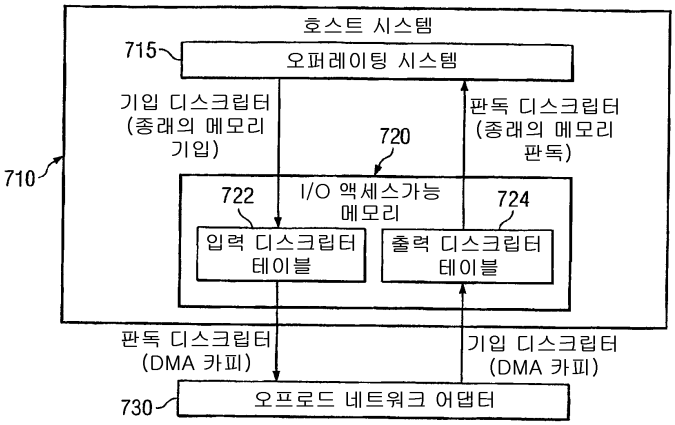
전체 청구항 수 : 총 10 항 심사관 : 안철용

(54) 통신 접속 수립 방법과 시스템, 데이터 전송 방법과 시스템, 및 컴퓨터 판독 가능한 저장 매체

(57) 요 약

호스트 프로세서로부터의 프로토콜 처리를 오프로딩하는 네트워크 어댑터에서의 다수의 개선이 제공된다. 구체적으로, 오프로드 네트워크 어댑터(730)를 활용하는 시스템 내에서 메모리 관리 및 최적화를 핸들링하는 메커니즘이 제공된다. 이 접속 수립 메커니즘은 접속 수립(1030)을 오프로드하는 기능과, 오프로드 네트워크 어댑터(730)에 대한 접속 상태 정보를 관리하는 기능을 제공한다. 접속 수립(1030)의 오프로드와 상태 정보 관리의 결과로서, 호스트 시스템(710)과 오프로드 네트워크 어댑터(730) 간에 요구되는 통신의 수가 감소될 수 있다. 또한, 오프로드 어댑터(730)에 대한 이들 기능의 오프로드는 공지된 컴퓨팅 시스템의 단편적인 통보가 아닌, 수립된 접속과 호스트 시스템(710)에 대한 상태 정보의 대량의(벌크) 통보를 허용한다.

대 표 도 - 도7



(72) 발명자

므라즈 로날드

미국 10590 뉴욕주 사우스 살렘 스미스 릿지 로드 169

나홈 에리츠 엠

미국 10024 뉴욕주 뉴욕 #406 웨스트 84번 스트리트 215

프라단 프래샨트

미국 10805 뉴욕주 뉴 로첼 아파트먼트 엠디 다벤포트 애비뉴 43

사후 삼비트

미국 10541 뉴욕주 마호팍 켄니컷 힐 로드 551

트레이시 존 엠

미국 10583 뉴욕주 스칼스데일 팔모스 로드 181

특허청구의 범위

청구항 1

데이터 처리 시스템들 간의 데이터 통신 접속을 수립하는 방법으로서,

제 1 데이터 처리 시스템의 호스트 시스템에서, 제 2 데이터 처리 시스템과의 접속을 수립하라는 요청을 수신하는 단계와,

상기 제 1 데이터 처리 시스템과 관련된 오프로드 네트워크 어댑터(offload network adapter)로 상기 요청을 전송하는 단계와,

상기 오프로드 네트워크 어댑터 내에서, 상기 제 2 데이터 처리 시스템과의 접속을 수립하되, 상기 접속을 기술하는 데이터 구조가 상기 오프로드 네트워크 어댑터 내에서 생성되는 단계와,

상기 오프로드 네트워크 어댑터 내에서 접속이 수립된 후, 수립된 상기 접속을 상기 호스트 시스템에 통보하는 단계를 포함하되,

상기 오프로드 네트워크 어댑터로 상기 요청을 전송하는 단계는,

상기 수신된 요청에 기초하여 접속 수립 요청 디스크립터(connection establishment request descriptor)를 생성하는 단계와,

상기 호스트 시스템의 입력/출력 액세스 가능한 메모리의 입력 디스크립터 테이블에 상기 접속 수립 요청 디스크립터를 기입하는 단계

를 포함하고,

상기 입력 디스크립터 테이블은 상기 오프로드 네트워크 어댑터에 의해 판독되고, 제어 및 데이터 인터페이스 요청을 제출하도록 상기 호스트 시스템에 의해 기입되며,

상기 오프로드 네트워크 어댑터 내에서 접속이 수립된 후, 수립된 상기 접속을 상기 호스트 시스템에 통보하는 단계는,

지연 기준이 충족되었는지의 여부를 판정하는 단계와,

상기 지연 기준이 충족된 것으로 판정된 것에 대한 응답으로, 상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 출력 디스크립터 테이블에 접속 완료 응답 디스크립터를 기입하는 단계

를 포함하고,

상기 지연 기준은 최종 접속 완료 응답 디스크립터가 상기 출력 디스크립터 테이블에 기록된 이래로 수립된 접속의 사전결정된 수이며,

상기 출력 디스크립터 테이블은 상기 호스트 시스템에 의해 판독되고, 이전 요청의 결과를 표시하며 상기 호스트로 데이터 도달을 통지하도록 상기 오프로드 네트워크 어댑터에 의해 기입되는

통신 접속 수립 방법.

청구항 2

데이터 처리 시스템들간의 데이터 통신 접속을 수립할 것을 컴퓨터 시스템에 지시하는 컴퓨터 판독가능 인스트럭션이 내장된 컴퓨터 프로그램 제품을 저장하는 컴퓨터 판독가능 매체로서,

상기 컴퓨터 프로그램 제품은,

제 1 데이터 처리 시스템의 호스트 시스템에서, 제 2 데이터 처리 시스템과의 접속을 수립하라는 요청을 수신하는 제 1 인스트럭션과,

상기 제 1 데이터 처리 시스템과 관련된 오프로드 네트워크 어댑터로 상기 요청을 전송하는 제 2 인스트럭션과,

상기 오프로드 네트워크 어댑터 내에서, 상기 제 2 데이터 처리 시스템과의 접속을 수립하되, 상기 접속을 기술하는 데이터 구조가 상기 오프로드 네트워크 어댑터 내에서 생성되는 제 3 인스트럭션과,

상기 오프로드 네트워크 어댑터 내에서 상기 접속이 수립된 후, 수립된 접속을 상기 호스트 시스템에 통보하는 제 4 인스트럭션을 포함하되,

상기 오프로드 네트워크 어댑터로 상기 요청을 전송하는 상기 제 2 인스트럭션은,

상기 수신된 요청에 기초하여 접속 수립 요청 디스크립터를 생성하는 인스트럭션과,

상기 호스트 시스템의 입력/출력 액세스 가능한 메모리의 입력 디스크립터 테이블에 상기 접속 수립 요청 디스크립터를 기입하는 인스트럭션

을 포함하고,

상기 입력 디스크립터 테이블은 상기 오프로드 네트워크 어댑터에 의해 판독되고, 제어 및 데이터 인터페이스 요청을 제출하도록 상기 호스트 시스템에 의해 기입되며,

상기 오프로드 네트워크 어댑터 내에서 접속이 수립된 후, 수립된 상기 접속을 상기 호스트 시스템에 통보하는 상기 제 4 인스트럭션은,

지연 기준이 충족되었는지의 여부를 판정하는 인스트럭션과,

상기 지연 기준이 충족된 것으로 판정된 것에 대한 응답으로, 상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 출력 디스크립터 테이블에 접속 완료 응답 디스크립터를 기입하는 인스트럭션

을 포함하고,

상기 지연 기준은 최종 접속 완료 응답 디스크립터가 상기 출력 디스크립터 테이블에 기록되었으므로 수립된 접속의 사전결정된 수이며,

상기 출력 디스크립터 테이블은 상기 호스트 시스템에 의해 판독되고, 이전 요청의 결과를 표시하며 상기 호스트로 데이터 도달을 통지하도록 상기 오프로드 네트워크 어댑터에 의해 기입되는

컴퓨터 판독 가능한 저장 매체.

청구항 3

데이터 처리 시스템들 간의 데이터 통신 접속 수립 시스템으로서,

제 1 데이터 처리 시스템의 호스트 시스템에서, 제 2 데이터 처리 시스템과의 접속을 수립하라는 요청을 수신하는 수단과,

상기 제 1 데이터 처리 시스템과 관련된 오프로드 네트워크 어댑터로 상기 요청을 전송하는 수단과,

상기 오프로드 네트워크 어댑터 내에서, 상기 제 2 데이터 처리 시스템과의 접속을 수립하되, 상기 접속을 기술하는 데이터 구조가 상기 오프로드 네트워크 어댑터 내에서 생성되는 수단과,

상기 오프로드 네트워크 어댑터 내에 상기 접속이 수립된 후 상기 수립된 접속을 상기 호스트 시스템에 통보하는 수단을 포함하되,

상기 오프로드 네트워크 어댑터로 상기 요청을 전송하는 수단은,

상기 수신된 요청에 기초하여 접속 수립 요청 디스크립터(connection establishment request descriptor)를 생성하는 수단과,

상기 호스트 시스템의 입력/출력 액세스 가능한 메모리의 입력 디스크립터 테이블에 상기 접속 수립 요청 디스크립터를 기입하는 수단

을 포함하고,

상기 입력 디스크립터 테이블은 상기 오프로드 네트워크 어댑터에 의해 판독되고, 제어 및 데이터 인터페이스 요청을 제출하도록 상기 호스트 시스템에 의해 기입되며,

상기 오프로드 네트워크 어댑터 내에서 접속이 수립된 후, 수립된 상기 접속을 상기 호스트 시스템에 통보하는 수단은,

지연 기준이 충족되었는지의 여부를 판정하는 수단과,

상기 지연 기준이 충족된 것으로 판정된 것에 대한 응답으로, 상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 출력 디스크립터 테이블에 접속 완료 응답 디스크립터를 기입하는 수단

을 포함하며,

상기 지연 기준은 최종 접속 완료 응답 디스크립터가 상기 출력 디스크립터 테이블에 기록되었으므로 수립된 접속의 사전결정된 수이고,

상기 출력 디스크립터 테이블은 상기 호스트 시스템에 의해 판독되며, 이전 요청의 결과를 표시하고 상기 호스트로 데이터 도달을 통지하도록 상기 오프로드 네트워크 어댑터에 의해 기입되는

통신 접속 수립 시스템.

청구항 4

제 1 항에 있어서,

상기 제 2 데이터 처리 시스템과의 접속을 수립하는 단계는 상기 데이터 구조에 상기 접속에 대한 상태 정보를 저장하는 단계를 포함하는

통신 접속 수립 방법.

청구항 5

제 1 항에 있어서,

상기 수립된 상기 접속을 상기 호스트 시스템에 통보하는 단계는,

접속 완료 응답 디스크립터를 생성하는 단계와,

상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 상기 출력 디스크립터 테이블에 상기 접속 완료 응답 디스크립터를 기입하는 단계를 포함하는

통신 접속 수립 방법.

청구항 6

제 1 항에 있어서,

상기 요청은 다수의 접속을 수립하라는 요청이고,

상기 제 1 데이터 처리 시스템과 관련된 오프로드 네트워크 어댑터로 상기 요청을 전송하는 단계는,

수립될 상기 다수의 접속에 상기 접속 각각을 식별하는 접속 요청 디스크립터를 생성하는 단계와,

상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 상기 입력 디스크립터 테이블에 상기 접속 요청 디스크립터를 기입하는 단계

를 포함하는

통신 접속 수립 방법.

청구항 7

제 1 항에 있어서,

상기 지연 기준은 상기 접속을 통해 도달하는 사전결정된 데이터 양, 수신된 데이터에서 검색되는 사전결정된 데이터 패턴, 및 최종 접속 완료 응답 디스크립터가 상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 상기 출력 디스크립터 테이블에 기록된 이래로의 사정결정된 시간 량을 더 포함하는

통신 접속 수립 방법.

청구항 8

제 1 항에 있어서,

상기 오프로드 네트워크 어댑터로부터 상기 호스트 시스템으로 상기 접속에 대한 상태 정보를 이주시키는 단계를 더 포함하는

통신 접속 수립 방법.

청구항 9

제 8 항에 있어서,

상기 오프로드 네트워크 어댑터로부터 상기 호스트 시스템으로 상기 접속에 대한 상태 정보를 이주시키는 단계는,

상기 접속의 상태를 식별하는 접속 속성 응답 디스크립터를 생성하는 단계와,

상기 호스트 시스템의 상기 입력/출력 액세스 가능한 메모리의 상기 출력 디스크립터 테이블에 상기 접속 속성 응답 디스크립터를 기록하는 단계

를 포함하는

통신 접속 수립 방법.

청구항 10

제 1 항에 있어서,

상기 제 2 데이터 처리 시스템과의 접속을 수립하라는 요청은 청취 요청(a listen request)이고,

상기 제 2 데이터 처리 시스템과의 접속을 수립하는 단계는 지정된 포트 번호에 대해 수신된 접속 요청을 식별하는 단계를 포함하는

통신 접속 수립 방법.

명 세 서

기 술 분 야

[0001] 본 발명은 일반적으로 개선된 데이터 처리 시스템에 관한 것이다. 더 상세하게, 본 발명은 오프로드 네트워크 어댑터에서 메모리 관리 동작을 지원하는 방법 및 장치에 관한 것이다.

배 경 기 술

[0002] 공지된 시스템에서, 오퍼레이팅 시스템은, 버퍼의 2개의 큐를 네트워크 인터페이스에 제공함으로써 오직 데이터 전송의 관점에서 종래의 네트워크 인터페이스와 통신한다. 버퍼의 제1큐는, 송신용으로 판독되는 호스트 메모리에서 판독-실시(read-made) 데이터 패킷에 포인팅하는 디스크립터로 이루어진다. 버퍼의 제2큐는, 처리용으로 수신된 호스트 메모리에서 미처리된 데이터 패킷으로 충진된 버퍼에 포인팅하는 디스크립터를 포함한다. 네트워크 인터페이스는, 큐가 물리적 메모리에 있는 네트워크 인터페이스를 통지하기 위해 메모리-매핑 입력/출력(I/O) 인터페이스를 제공하고, 데이터 패킷이 도달할 경우에 무슨 인터럽트가 발생하는지와 같은 일부 제어 정보에 대한 인터페이스를 제공한다.

[0003] 종래의 네트워크 인터페이스에 대한 네트워크 프로토콜 처리는 송신용으로 네트워크 어댑터에 제공되는 오직 데이터 패킷만을 갖는 호스트 내에서 전부 수행된다. 하지만, 네트워크 라인 속도는 마이크로프로세서 성능의 성장보다 더 신속하게 증가되어 왔다. 결과적으로, 호스트 프로세서는 다량의 TCP/IP 프로토콜 처리, 고장난(out-of-order) 데이터 패킷의 재조합, 리소스-인텐시브 메모리 카피(copy), 및 인터럽트로 부담이 되고 있다. 일부 고속 네트워크에서, 호스트 프로세서는, 구동하고 있는 애플리케이션에서보다 네트워크 트래픽을 핸들링할 더 많은 처리를 수행해야 한다. 따라서, 데이터 패킷은 네트워크 속도보다 더 낮은 레이트로 호스트에서 처리된다.

[0004] 이러한 문제를 해결하기 위하여, 최근에는, 호스트 프로세서로부터 네트워크 어댑터 상의 하드웨어로의 TCP/IP

프로토콜의 처리를 오프로딩(offloading)하는데 집중한다. 종종, 인텔리전트 네트워크 어댑터 또는 TCP/IP 오프로드 엔진(TOE)으로서도 지칭되는 그러한 네트워크 어댑터는 네트워크 프로세서 및 펌웨어, 특수 ASIC, 또는 이들 양자의 조합으로 구현될 수 있다. 이들 네트워크 어댑터는 호스트 프로세서 처리를 오프로딩하여 애플리케이션 성능을 증가시킬 뿐아니라, iSCSI 저장 영역 네트워크(SAN) 및 고성능 NAS(network attached storage) 애플리케이션과 같은 신규한 타입의 네트워크 및 디바이스와 통신하게 할 수 있다.

[0005] 이들 네트워크 어댑터는 데이터 패킷의 TCP/IP 프로토콜 처리를 오프로딩하지만, 네트워크를 통한 통신에 필요한 대부분의 처리는 여전히 호스트 시스템 내에서 유지된다. 예를 들어, 호스트 시스템은 여전히 접속의 수립, 수립된 접속 각각에 대한 상태 정보의 유지, 메모리 관리의 핸들링 등을 책임진다. 따라서, 호스트 시스템은 그 호스트 시스템에서 수행되어야 하는 이들 동작으로 인해, 또한, 호스트 시스템에서 이들 동작을 수행하기 위해 호스트 시스템과 네트워크 어댑터 사이에서 필요한 통신의 양으로 인해 여전히 프로세서 부하를 경험한다. 따라서, 호스트 시스템 상의 처리 부하가 최소화되고 네트워크 어댑터에서 더 많은 처리가 수행되도록 네트워크 어댑터의 동작을 개선시키는 장치 및 방법을 갖는 것이 바람직하다.

발명의 상세한 설명

[0006] 본 발명은, 이후 오프로드 네트워크 어댑터로서 지칭되는 호스트 프로세서로부터 프로토콜 처리를 오프로딩하는 네트워크 어댑터에 있어서 다수의 개선을 제공한다. 구체적으로, 본 발명은 오프로드 네트워크 어댑터를 활용하는 시스템 내에서 메모리 관리 및 최적화를 핸들링하는 메커니즘을 제공한다. 또한, 본 발명은 오프로드 네트워크 어댑터를 활용하는 시스템에 있어서 접속 수립을 개선하는 메커니즘을 제공한다. 또한, 본 발명은 오프로드 네트워크 어댑터를 활용하는 시스템에 있어서 데이터 패킷의 수신을 핸들링하는 메커니즘을 제공한다.

[0007] 본 발명의 일 양태는 오프로드 네트워크 어댑터에 대한 접속 수립 및 접속 상태 정보의 유지를 오프로딩하는 능력이다. 접속 수립 및 상태 정보 유지의 이러한 오프로딩의 결과로서, 호스트 시스템과 오프로드 네트워크 어댑터 사이에 필요한 통신의 수는 감소될 수도 있다. 또한, 오프로드 네트워크 어댑터에 대한 이들 기능의 오프로딩은, 공지의 컴퓨팅 시스템에서 존재하는 단편적 통지(piecemeal notification)보다는 호스트 시스템에 대한 수립 접속 및 상태 정보의 벌크(bulk) 통지를 허용한다.

[0008] 접속 수립에 더하여, 본 발명은 오프로드 네트워크 어댑터를 활용하는 데이터 처리 시스템에 있어서 메모리 관리를 더 개선시킨다. 본 발명에 따른 메모리 관리는 데이터의 버퍼링된 송수신뿐 아니라 데이터의 제로-카피(zero-copy) 송수신 양자를 허용한다. 또한, 본 발명은, 임의의 개수의 속성에 기초하여 특정 접속체 중에서 공유될 수 있는 DMA 버퍼의 그룹화를 허용한다. 또한, 본 발명은 호스트 시스템에 벌크로 전달될 수 있도록 DMA 요청을 지연시키는 부분 송수신 버퍼 동작, 및 호스트 시스템으로의 데이터의 신속한 전송을 위한 메커니즘을 허용한다.

[0009] 접속 수립 및 메모리 관리에 더하여, 본 발명은 오프로드 네트워크 어댑터를 활용하는 데이터 처리 시스템에 있어서 수신 데이터의 핸들링을 더 개선시킨다. 본 발명의 오프로드 네트워크 어댑터는 그 오프로드 네트워크 어댑터로 하여금 호스트 시스템에 대한 데이터 수신의 통지를 상이한 방식으로 지연시키게 하는 로직을 포함할 수도 있다. 호스트 시스템에 대한 데이터 패킷 수신의 통지를 지연시키는 이점은, 예를 들어, 단일 통지에 있어서 제1통지 직후에 도달할 수 있는 수개의 데이터 패킷의 집합에 대한 잠재성이다. 연속적인 데이터 패킷 도달을 갖는 스트림이 주어지면, 통지 지연에 대한 값이 설정될 수도 있으며, 이 값은 통신 소켓당 호스트 시스템에 대해 구성가능할 수도 있다.

[0010] 본 발명의 이들 특성 및 이점 그리고 다른 특성 및 이점은 바람직한 실시예에 대한 다음의 상세한 설명에서 설명될 것이고, 또는, 바람직한 실시예에 대한 다음의 상세한 설명의 관점에서 당업자에게 명백하게 될 것이다.

실 시 예

[0030] 본 발명은 오프로드 네트워크 어댑터, 즉, 일부 또는 모든 네트워크 프로토콜 처리를 수행하고 이에 따라 호스트로부터의 처리를 오프로딩하는 네트워크 어댑터의 동작을 개선시키는 장치 및 방법에 관한 것이다. 본 발명이 오프로드 네트워크 어댑터에 관한 것이기 때문에, 본 발명은 하나 이상의 네트워크를 갖는 분산 데이터 처리 시스템과 함께 사용하기에 특히 잘 적합하다. 도 1 내지 도 3은 본 발명의 양태가 구현될 수도 있는 그러한 분

산 데이터 처리 환경의 일예로서 제공된다. 도 1 내지 도 3은 단지 예시적이며 이들 예시적인 실시예에 대한 다수의 변경이 본 발명의 취지 및 범위를 벗어나지 않고 행해질 수도 있음을 이해해야 한다.

[0031] 다음으로, 도면을 참조하면, 도 1은 본 발명이 구현될 수도 있는 데이터 처리 시스템의 네트워크의 도식 표현을 나타낸 것이다. 네트워크 데이터 처리 시스템(100)은 본 발명이 구현될 수도 있는 컴퓨터의 네트워크이다. 네트워크 데이터 처리 시스템(100)은, 네트워크 데이터 처리 시스템(100) 내에 함께 접속된 다양한 디바이스와 컴퓨터 간의 통신 링크를 제공하기 위해 사용되는 매체인 네트워크(102)를 포함한다. 네트워크(102)는 유선, 무선 통신 링크, 또는 광섬유 케이블과 같은 접속체를 포함할 수도 있다.

[0032] 도시된 예에서, 서버(104)는 저장유닛(106)과 함께 네트워크(102)에 접속된다. 또한, 클라이언트(108, 110 및 112)가 네트워크(102)에 접속된다. 이들 클라이언트(108, 110 및 112)는, 예를 들어, 퍼스널 컴퓨터 또는 네트워크 컴퓨터일 수도 있다. 도시된 예에서, 서버(104)는 부트 파일, 오퍼레이팅 시스템 이미지, 및 애플리케이션과 같은 데이터를 클라이언트(108~112)에 제공한다. 클라이언트(108, 110 및 112)는 서버(104)에 대한 클라이언트이다. 네트워크 데이터 처리 시스템(100)은 추가적인 서버, 클라이언트, 및 도시되지 않은 다른 디바이스를 포함할 수도 있다. 도시된 예에서, 네트워크 데이터 처리 시스템(100)은, 프로토콜의 전송 제어 프로토콜/인터넷 프로토콜(TCP/IP) 슈트(suit)를 사용하여 서로 통신하는 네트워크와 게이트웨이의 세계적인 컬렉션을 나타내는 네트워크(102)를 갖는 인터넷이다. 데이터 및 메시지를 라우팅하는 수천개의 상업 시스템, 정부 시스템, 교육 시스템 및 다른 컴퓨터 시스템으로 이루어진 메이저 노드 또는 호스트 컴퓨터 간의 고속 데이터 통신 라인의 백본이 인터넷의 중심에 있다. 물론, 네트워크 데이터 처리 시스템(100)은 또한 예를 들어, 인트라넷, LAN(local area network), 또는 WAN(wide area network)과 같은 다수의 상이한 타입의 네트워크로서 구현될 수도 있다. 도 1은 본 발명에 대한 구조적 제한으로서가 아니라 일예로서 의도된다.

[0033] 도 2를 참조하면, 도 1의 서버(104)와 같은 서버로서 구현될 수도 있는 데이터 처리 시스템의 블록 다이어그램이 본 발명의 바람직한 실시예에 따라 도시된다. 데이터 처리 시스템(200)은, 시스템 버스(206)에 접속된 복수의 프로세서(202 및 204)를 포함하는 대칭형 멀티프로세서(SMP) 시스템일 수도 있다. 다른 방법으로, 단일 프로세서 시스템이 채용될 수도 있다. 또한, 로컬 메모리(209)로의 인터페이스를 제공하는 메모리 제어기/캐시(208)가 시스템 버스(206)에 접속된다. I/O 버스 브리지(210)가 시스템 버스(206)에 접속되고 I/O 버스(212)로의 인터페이스를 제공한다. 메모리 제어기/캐시(208) 및 I/O 버스 브리지(210)는 도시된 바와 같이 통합될 수도 있다.

[0034] I/O 버스(212)에 접속된 주변 컴포넌트 배선(PCI) 버스 브리지(214)는 PCI 로컬 버스(216)로의 인터페이스를 제공한다. 다수의 모뎀이 PCI 로컬 버스(216)에 접속될 수도 있다. 통상적인 PCI 버스 구현은 4개의 PCI 확장 슬롯 또는 애드-인 커넥터(add-in connector)를 지원할 것이다. 도 1의 클라이언트(108 내지 112)로의 통신 링크는 애드-인 커넥터를 통해 PCI 로컬 버스(216)에 접속된 모뎀(218) 및 네트워크 어댑터(220)를 통해 제공될 수도 있다.

[0035] 추가적인 PCI 버스 브리지(222 및 224)는 추가적인 PCI 로컬 버스(226 및 228)에 대한 인터페이스를 제공하며, 이로부터 추가적인 모뎀 또는 네트워크 어댑터가 지원될 수도 있다. 이러한 방식으로, 데이터 처리 시스템(200)은 다중의 네트워크 컴퓨터로의 접속을 허용한다. 또한, 메모리-매핑 그래픽 어댑터(230) 및 하드 디스크(232)는 도시된 바와 같이 I/O 버스(212)에 직접 또는 간접적으로 접속될 수도 있다.

[0036] 당업자는 도 2에 도시된 하드웨어가 변경될 수도 있음을 이해할 것이다. 또한, 예를 들어, 광학 디스크 드라이브 등과 같은 다른 주변 디바이스가 도시된 하드웨어에 추가하여 또는 그 대신에 사용될 수도 있다. 도시된 예는 본 발명에 대한 구조적 제한을 내포하도록 의미되지 않는다.

[0037] 도 2에 도시된 데이터 처리 시스템은, 예를 들어, AIX(Advanced Interactive Executive) 오퍼레이팅 시스템 또는 LINUX 오퍼레이팅 시스템을 구동시키는, 뉴욕소재 아르몽크의 IBM(International Business Machines)사의 제품인 IBM e서버 p시리즈 시스템일 수도 있다.

[0038] 도 3을 참조하면, 본 발명이 구현될 수도 있는 데이터 처리 시스템을 나타낸 블록 다이어그램이 도시되어 있다. 데이터 처리 시스템(300)은 클라이언트 컴퓨터의 일예이다. 데이터 처리 시스템(300)은 주변 컴포넌트 배선(PCI) 로컬 버스 구조를 채용한다. 비록 도시된 예가 PCI 버스를 채용하지만, AGP(Accelerated Graphics Port) 및 ISA(Industry Standard Architecture)와 같은 다른 버스 구조가 사용될 수도 있다. 프로세서(302) 및 메인 메모리(304)는 PCI 브리지(308)를 통하여 PCI 로컬 버스(306)에 접속된다. 또한, PCI 브리지(308)는 프로세서(302)에 대한 통합된 메모리 제어기 및 캐시 메모리를 포함할 수도 있다. PCI 로컬 버스(306)로의 추

가적인 접속이 집적 컴포넌트 배선 또는 애드-인 보드를 통해 행해질 수도 있다. 도시된 예에서, 로컬 영역 네트워크(LAN) 어댑터(310), SCSI 호스트 버스 어댑터(312), 및 확장 버스 인터페이스(314)가 집적 컴포넌트 접속에 의해 PCI 로컬 버스(306)에 접속된다. 이에 반하여, 오디오 어댑터(316), 그래픽 어댑터(318), 및 오디오/비디오 어댑터(319)는 확장 슬롯에 삽입된 애드-인 보드에 의해 PCI 로컬 버스(306)에 접속된다. 확장 버스 인터페이스(314)는 키보드 및 마우스 어댑터(320), 모뎀(322), 및 추가적인 메모리(324)에 대한 접속을 제공한다. 소형 컴퓨터 시스템 인터페이스(SCSI) 호스트 버스 어댑터(312)는 하드 디스크 드라이브(326), 테이프 드라이브(328), 및 CD-ROM 드라이브(330)에 대한 접속을 제공한다. 통상적인 PCI 로컬 버스 구현은 3 또는 4개의 PCI 확장 슬롯 또는 애드-인 커넥터를 지원할 것이다.

[0039] 오퍼레이팅 시스템은 프로세서(302)에서 구동하고, 도 3의 데이터 처리 시스템(300) 내의 다양한 컴포넌트의 제어를 조정 및 제공하기 위해 사용된다. 오퍼레이팅 시스템은, 마이크로소프트사로부터 입수가능한 윈도우즈 XP와 같은 상업적으로 이용가능한 오퍼레이팅 시스템일 수도 있다. 자바(Java)와 같은 객체 지향 프로그래밍 시스템이 오퍼레이팅 시스템과 함께 구동할 수도 있으며, 데이터 처리 시스템(300)에서 실행하는 자바 프로그램 또는 애플리케이션으로부터 오퍼레이팅 시스템에 대한 호출(call)을 제공한다. "자바"는 선 마이크로시스템즈사의 상표이다. 오퍼레이팅 시스템에 대한 명령, 객체-지향 프로그래밍 시스템, 및 애플리케이션 또는 프로그램은 하드 디스크 드라이브(326)와 같은 저장 디바이스에 위치되며, 프로세서(302)에 의한 실행을 위해 메인 메모리(304)에 로딩될 수도 있다.

[0040] 당업자는 도 3의 하드웨어가 구현에 의존하여 변경될 수도 있음을 이해할 것이다. 플래시 판독전용 메모리(ROM), 등가 비휘발성 메모리, 또는 광학 디스크 드라이브 등과 같은 다른 내부 하드웨어 또는 주변 디바이스가 도 3에 도시된 하드웨어에 추가하여 또는 그 대신에 사용될 수도 있다. 또한, 본 발명의 프로세서는 멀티프로세서 데이터 처리 시스템에 적용될 수도 있다.

[0041] 다른 예로서, 데이터 처리 시스템(300)은 일부 타입의 네트워크 통신 인터페이스에 의존하지 않고도 부팅가능하도록 구성된 자립형(stand-alone) 시스템일 수도 있다. 또다른 예로서, 데이터 처리 시스템(300)은 개인휴대 정보단말기(PDA) 디바이스일 수도 있으며, 이는 오퍼레이팅 시스템 파일 및/또는 사용자-생성 데이터를 저장하기 위한 비-휘발성 메모리를 제공하기 위하여 ROM 및/또는 플래시 ROM으로 구성된다.

[0042] 도 3에 도시된 예 및 상술된 예들은 구조적 제한을 내포하도록 의미되지 않는다. 예를 들어, 데이터 처리 시스템(300)은 또한 PDA의 형태를 취하는 것에 더하여 노트북 컴퓨터 또는 핸드-헬드(hand-held) 컴퓨터일 수도 있다. 또한, 데이터 처리 시스템(300)은 키오스크(kiosk) 또는 웹 어플라이언스일 수도 있다.

[0043] 다음으로 도 4를 참조하면, 네트워크 어댑터의 다이어그램이 본 발명의 바람직한 실시예에 따라 도시되어 있다. 네트워크 어댑터(400)는 도 2의 네트워크 어댑터(220), 도 3의 LAN 어댑터(310) 등과 같이 구현될 수도 있다. 도시된 바와 같이, 네트워크 어댑터(400)는 이더넷 인터페이스(402), 데이터 버퍼(404), 및 PCI 버스 인터페이스(406)를 포함한다. 이들 3개의 컴포넌트는 데이터 처리 시스템의 네트워크와 버스 간의 경로를 제공한다. 이더넷 인터페이스(402)는 데이터 처리 시스템에 접속된 네트워크로의 인터페이스를 제공한다. PCI 버스 인터페이스(406)는 PCI 버스(216 또는 306)와 같은 버스로의 인터페이스를 제공한다. 데이터 버퍼(404)는 네트워크 어댑터(400)를 통해 송신 및 수신되는 데이터를 저장하기 위해 사용된다. 또한, 이러한 데이터 버퍼는 추가 저장용으로 제공하기 위해 SRAM 인터페이스로의 접속을 포함한다.

[0044] 또한, 네트워크 어댑터(400)는 EEPROM(electrically erasable programmable read-only memory) 인터페이스(408), 레지스터/구성/상태/제어 유닛(410), 오실레이터(412), 및 제어 유닛(414)을 포함한다. EEPROM 인터페이스(408)는, 네트워크 어댑터(400)에 대한 명령 및 다른 구성 정보를 포함할 수도 있는 EEPROM 칩으로의 인터페이스를 제공한다. 상이한 파라미터 및 세팅이 EEPROM 인터페이스(408)를 통해 EEPROM 칩에 저장될 수도 있다. 레지스터/구성/상태/제어 유닛(410)은 네트워크 어댑터(400)에 대한 프로세스를 구성 및 구동하기 위해 사용되는 정보를 저장할 장소를 제공한다. 예를 들어, 타이머에 대한 타이머값이 이들 레지스터 내에 저장될 수도 있다. 부가적으로, 상이한 프로세스에 대한 상태 정보 또한 이러한 유닛 내에 저장될 수도 있다. 오실레이터(412)는 네트워크 어댑터(400)에 대한 프로세스를 실행하기 위한 클록 신호를 제공한다.

[0045] 제어 유닛(414)은 네트워크 어댑터(400)에 의해 수행되는 상이한 프로세스 및 기능을 제어한다. 제어 유닛(414)은 다양한 형태를 취할 수도 있다. 예를 들어, 제어 유닛(414)은 프로세서 또는 주문형 집적 칩(ASIC)일 수도 있다. 이들 예에서, 데이터의 플로우 제어를 관리하기 위해 사용되는 본 발명의 프로세스는 제어 유닛(414)에 의해 실행된다. 만약 프로세서로서 구현되면, 이들 프로세스에 대한 명령은 EEPROM 인터페이스(408)를 통해 액세스되는 칩에 저장될 수도 있다.

[0046] 데이터는 이더넷 인터페이스(402)를 통해 수신 동작 시에 수신된다. 이러한 데이터는 PCI 버스 인터페이스(406)를 통한 데이터 처리 시스템으로의 전송을 위해 데이터 버퍼(404)에 저장된다. 역으로, 데이터는 PCI 버스 인터페이스(406)를 통한 송신을 위해 호스트 시스템으로부터 수신되고, 데이터 버퍼(404)에 저장된다.

[0047] 종래의 데이터 처리 시스템에서, 네트워크 어댑터를 통해 호스트 시스템으로/으로부터 송신되는 데이터의 처리는 호스트 시스템 내에서 수행된다. 도 5는 TCP/IP 프로토콜 스택에 있어서 데이터 패킷의 종래의 처리가 수행되는 방식을 도시한 것이다. 도 5에 도시된 바와 같이, 애플리케이션 소프트웨어(510)는 오퍼레이팅 시스템(520) 및 네트워크 어댑터(530)를 통해 데이터를 송신 및 수신한다. TCP/IP 프로토콜 스택을 통한 데이터의 처리는 TCP/IP 프로토콜 처리를 수행하는 오퍼레이팅 시스템(520)으로 수행되어, 송신용의 포맷팅된 데이터 패킷을 생성하거나 또는 데이터 패킷 내의 데이터를 적절한 애플리케이션(510)으로 추출 및 라우팅되게 한다. 이들 동작은 호스트 시스템 상의 소프트웨어로 수행된다.

[0048] 포맷팅된 데이터 패킷은 네트워크 어댑터(530)를 통해 하드웨어로 송/수신된다. 네트워크 어댑터(530)는 매체 액세스 제어(MAC) 및 물리 레이어로부터의 데이터 패킷에 대해 동작한다. 매체 액세스 제어 레이어는 네트워크 상의 물리적 송신 매체로의 액세스를 제어하는 서비스이다. MAC 레이어 기능은 네트워크 어댑터 내에 형성되고, 각각의 네트워크 어댑터를 식별하는 고유의 식별번호를 포함한다. 물리 레이어는 네트워크 매체를 통한 비트의 송신용 서비스를 제공하는 레이어이다.

[0049] 도 5에 도시된 바와 같이, 종래의 네트워크 인터페이스에서, 데이터가 호스트 시스템으로부터 네트워크를 통해 송신될 경우, 그 데이터는 사용자 공간 내의 애플리케이션 버퍼(540)로부터 고정된 커널(pinned kernel) 버퍼(550)로 카피되고, 네트워크 어댑터 큐(560) 내의 엔트리는 송신을 위해 데이터를 네트워크 어댑터(530)에 큐잉하기 위해 생성된다. 데이터가 호스트 시스템 상의 애플리케이션(510)에 대한 네트워크로부터 수신될 경우, 데이터 패킷은 직접 메모리 액세스(DMA) 동작을 이용하여 호스트 커널 버퍼(540)에 기입된다. 그 후, 그 데이터는, 애플리케이션이 receive()를 호출할 경우, 호스트에 의해 사용자 공간 내의 애플리케이션의 버퍼(540)에 카피된다.

[0050] 도 6은 오프로드 네트워크 어댑터가 TCP/IP 프로토콜 스택에 있어서 데이터 패킷을 처리하는 방식을 도시한 것이다. 도 6에 도시된 바와 같이, 호스트 시스템의 오퍼레이팅 시스템(620)에서 종래 수행되는 TCP 및 IP 처리는 이동하여, 오프로드 네트워크 어댑터(630) 내에서 수행하게 된다. 결과적으로, 호스트 시스템에 의해 수행되는 처리는, 애플리케이션(610)이 더 효율적으로 실행될 수 있도록 감소된다.

[0051] 공지의 오프로드 네트워크 어댑터에 있어서, 비록 TCP/IP 스택의 처리가 네트워크 어댑터(630)로 시프트하였지만, 도 5에 대해 상술한 버퍼링된 송신 및 수신은 여전히 필요하다. 즉, 도 6에 도시된 바와 같이, 호스트 시스템으로부터의 데이터 패킷의 전송을 위해, 데이터는 먼저 사용자 공간 내의 애플리케이션 버퍼(640)로부터 커널 버퍼(650)로 카피되며, 여기서, 그 데이터는 네트워크 어댑터에 의한 처리를 위해 네트워크 어댑터 큐(660)에 큐잉된다. 유사하게, 수신된 데이터 패킷에 있어서, 데이터는 커널 버퍼(650)에 DMA되고, 추후에, 사용자 공간 내의 애플리케이션 버퍼(640)에 카피된다.

[0052] 따라서, 상기 종래의 경우에 있어서와 같이, 공지의 오프로드 네트워크 어댑터에서, 사용자 공간 애플리케이션 버퍼(640)와 커널 공간 커널 버퍼(650) 사이에서 데이터의 카피가 여전히 필요하다. 그러한 카피 동작은, 송신 또는 수신되는 모든 데이터 패킷에 대해 호스트 시스템에서 수행되어야 한다. 그러한 카피 동작과 관련된 오버헤드는 애플리케이션을 구동하기 위한 호스트 프로세서의 가용도를 감소시킨다.

[0053] 또한, 데이터 패킷의 TCP/IP 프로토콜 처리는 오프로드 네트워크 어댑터(630)에 오프로딩될 수도 있으며, 실제 접속 수립 및 각각의 수립된 접속에 대한 상태 정보의 유지는, 여전히, 예를 들어, 오퍼레이팅 시스템(620)인 호스트 시스템의 책임이다. 즉, 호스트는 아웃바운드(outbound) 및 인바운드 접속을 수립하기 위해 필요한 동작을 여전히 제공해야 한다. 또한, 호스트는 각각의 접속 변경의 상태로서 네트워크 어댑터와 메시지를 교환하여, 각각의 접속에 대해 호스트 시스템에 저장된 상태 정보가 유지될 수 있게 해야 한다.

[0054] 결과적으로, 호스트 시스템으로부터 네트워크 어댑터로의 TCP/IP 프로토콜 처리의 오프로딩이 컴퓨팅 시스템의 스루풋이 개선되었지만, 메모리가 그러한 오프로드 네트워크 어댑터 시스템에서 관리되는 방식을 개선함으로써, 또한 접속 수립이 오프로딩되고 호스트와 네트워크 어댑터 간의 메시징이 최소화되도록 접속이 수립되는 방식을 개선함으로써 추가적인 개선이 획득될 수도 있다. 또한, 네트워크 어댑터의 동작의 개선은, 네트워크 어댑터와 호스트 시스템 간의 상호작용이 최소화되도록 데이터가 오프로드 네트워크 어댑터에서 수신되는 방식을 개선함으로써 획득될 수도 있다.

[0055] 본 발명은, 호스트 시스템과 네트워크 어댑터 간의 상호작용이 최소화되도록 오프로드 네트워크 어댑터의 동작을 개선하는 메커니즘을 제공한다. 본 발명은 호스트 시스템의 오퍼레이팅 시스템과 오프로드 네트워크 어댑터 간의 개선된 인터페이스를 제공한다. 이러한 인터페이스는 제어부 및 데이터부를 포함한다. 그 인터페이스는, 그 인터페이스의 제어부 및 데이터부 양자를 나타내는 명시적인 데이터 구조로 사용되는 버퍼의 큐를 이용한다. 그 인터페이스의 제어부는 호스트 시스템으로 하여금 오프로드 네트워크 어댑터를 명령하게 하고, 오프로드 네트워크 어댑터로 하여금 호스트 시스템을 명령하게 한다. 예를 들어, 호스트 시스템은 어떠한 포트 번호가 청취되는지에 대해 네트워크 인터페이스를 명령할 수도 있으며, 오프로드 네트워크 어댑터는 신규한 접속의 수립, 데이터의 수신 등에 대해 호스트 시스템을 명령할 수도 있다. 그 인터페이스의 데이터부는 송신 및 수신 양자에 대해 수립된 접속체 상의 데이터의 전송을 위한 메커니즘을 제공한다. 그 인터페이스의 제어부는, 예를 들어, socket(), bind(), listen(), connect(), accept(), setsockopt() 등과 같이, 접속을 제어하는 종래의 소켓 애플리케이션 프로그래밍 인터페이스(API)를 이용함으로써 발생될 수도 있다. 그 인터페이스의 데이터부는, 예를 들어, send(), sendto(), write(), writev(), read(), readv() 등과 같이, 데이터를 송신 또는 수신하기 위해 소켓 API에 의해 발생될 수도 있다.

[0056] 도 7은 본 발명의 오프로드 네트워크 어댑터 프로그래밍 인터페이스를 사용하여 호스트 시스템과 오프로드 네트워크 어댑터 간의 통신을 나타내는 예시적인 다이어그램이다. 오프로드 네트워크 어댑터 프로그래밍 인터페이스는, 호스트 시스템 상의 I/O 액세스가능 메모리의 예비부에서 요청 및 응답 디스크립터를 기입 및 판독하기 위해 직접 메모리 액세스(DMA) 동작, 즉, DMA에 주로 기초하는 오프로드 네트워크 어댑터와 호스트 시스템 간의 통신 인터페이스를 제공한다.

[0057] 도 7에 도시된 바와 같이, 호스트 시스템(710)은 오프로드 네트워크 어댑터(730)에 또는 오프로드 네트워크 어댑터로부터 데이터 전송을 위한 요청을 제출하고, 오프로드 네트워크 어댑터(730)는 그 요청의 성공 또는 실패의 통지로 응답한다. 요청 및 응답은, 요청 디스크립터 및 응답 디스크립터로 지칭되는 데이터 구조로 패키징된다. 그 디스크립터들은 호스트 시스템(710) 상의 I/O 액세스가능 메모리(720) 내의 2개의 물리 영역에 기입 및 그 영역으로부터 판독된다. 이들 영역은 입력 디스크립터 테이블(722) 및 출력 디스크립터 테이블(724)로 지칭되며, 생산자-소비자 방식으로 사용된다.

[0058] 입력 디스크립터 테이블(722)은 오프로드 네트워크 어댑터(730)에 의해 판독되고 호스트 시스템(710)에 의해 기입되어 제어 및 데이터 인터페이스 요청을 제출한다. 출력 디스크립터 테이블(724)은 호스트 시스템(710)에 의해 판독되고, 출력 디스크립터 테이블(724)을 사용하여 이전 요청의 결과를 나타내고 호스트 시스템(710)에게 데이터 도달을 통지하는 오프로드 네트워크 어댑터(730)에 의해 기입된다.

[0059] 호스트 시스템(710) 및 오프로드 네트워크 어댑터(730) 모두 이들 디스크립터 테이블(722 및 724)로부터 판독하고 이들 테이블에 기입하지만, 그 디스크립터에 동일한 방식으로 액세스하지 않아야 한다. 호스트 시스템(710)은 종래의 메모리 판독 및 기입을 사용하여 디스크립터 테이블(722 및 724)에 액세스한다. 하지만, 오프로드 네트워크 어댑터는 DMA 동작을 사용하여 디스크립터 테이블(722 및 724)에 및 그 테이블로부터 임의의 세트의 디스크립터를 카피한다.

[0060] 종래의 네트워크 어댑터에서와 같이, 호스트 시스템(710)은, 예를 들어, 인터럽트를 폴링 또는 수신함으로써 오프로드 네트워크 어댑터(730)로부터 출력 디스크립터 테이블(724) 내의 신규한 응답 디스크립터를 통지받을 수도 있다. 즉, 데이터 패킷이 오프로드 네트워크 어댑터에 수신되고, 호스트 시스템(710)으로의 데이터 패킷의 도달의 통지에 대해 일정한 기준이 충족될 경우, 이후, 더 상세히 설명되는 바와 같이, 응답 디스크립터는 오프로드 네트워크 어댑터(730)에 의해 생성되고 출력 디스크립터 테이블(724)에 기입될 수도 있다. 그 후, 인터럽트는, 출력 디스크립터 테이블(724) 내에서 신규한 디스크립터를 나타내는 오퍼레이팅 시스템(715)에 의해 수신될 수도 있다. 다른 방법으로, 호스트 시스템(710)은 신규한 디스크립터에 대해 출력 디스크립터 테이블(724)을 주기적으로 폴링할 수도 있다. 만약 출력 디스크립터 테이블(724)이 오버플로우의 위험에 있으면, 오프로드 네트워크 어댑터(730)는 호스트 시스템(710)에 인터럽트를 발생시켜 그 상황을 통지할 수도 있다.

[0061] 본 발명의 일 예시적인 실시예에서, 디스크립터 테이블(722 및 724)에 기입되는 디스크립터는 256비트/32바이트이고, 디스크립터 소유자(1비트), 디스크립터 타입(5비트), 디스크립터 콘텐츠(250비트)와 같이 구조화된다. 소유자 비트는 디스크립터 테이블(722 및 724)에서 디스크립터의 생산자/소비자 관계용으로 사용된다. 즉, 통신하는 2개의 컴포넌트, 예를 들어, 호스트 오퍼레이팅 시스템 및 오프로드 네트워크 어댑터가 존재하기 때문에, 생산자/소비자 관계가 존재한다. 단일 비트가 디스크립터의 소유를 나타내기 위해 사용될 수 있다. 예를 들어, "1"은 호스트 생성 디스크립터를 나타낼 수도 있고, "제로"는 오프로드 네트워크 어댑터 생성 디스

크립터를 나타낼 수도 있으며, 그 역도 성립한다.

[0062] 디스크립터 타입은, 디스크립터와 관련된 동작 및/또는 요청을 식별한다. 예를 들어, 요청 디스크립터는 다음의 타입, 즉, 버퍼 송신, 버퍼 가용, 접속 요청, 종료 요청, 청취 요청, 삭제 요청, 접속 속성 제어 및 네트워크 어댑터 속성 제어 중 하나로 이루어질 수도 있다.

[0063] 버퍼 송신 디스크립터 타입은 송신될 데이터를 저장하는 버퍼를 할당하기 위한 요청과 관련되며, 이하 설명되는 버퍼, 사용할 접속 식별자, 및 ASAP 비트의 값을 식별한다. 버퍼 가용 디스크립터 타입은 수신 데이터를 저장하는 버퍼를 할당하기 위한 버퍼와 관련되며, 수신 데이터를 저장하는 버퍼 및 그 데이터가 수신되는 접속 식별자를 식별한다. 접속 요청 디스크립터 타입은 특정 로컬 포트 및 프로토콜에 대한 접속을 개시하기 위한 요청과 관련된다. 종료 요청 디스크립터 타입은 특정 접속을 중단하기 위한 요청과 관련된다. 청취 요청 디스크립터 타입은 포트 및 프로토콜에 대한 접속을 수신하기 위한 의지를 나타내는 요청과 관련된다. 삭제 요청 디스크립터 타입은 이전에 제출된 송신, 접속 또는 청취 요청을 삭제하기 위한 요청과 관련된다. 접속 속성 제어 디스크립터 타입은 접속 속성을 획득 또는 설정하기 위한 요청과 관련된다. 네트워크 어댑터 속성 제어 디스크립터 타입은 네트워크 어댑터-전체 속성을 획득 및 설정하기 위한 요청과 관련된다.

[0064] 응답 디스크립터 또한 다양한 타입을 가질 수도 있다. 예를 들어, 응답 디스크립터는 다음의 타입, 즉, 버퍼 수신, 버퍼 가용, 접속 도달, 접속 완료, 청취 응답, 종료 응답, 삭제 응답, 접속 속성, 및 네트워크 어댑터 속성 중 하나일 수도 있다. 버퍼 수신 디스크립터 타입은, 가용 데이터를 갖는 버퍼를 식별하고 데이터가 어는 접속을 향하는지를 식별한다. 버퍼 가용 디스크립터 타입은, DMA가 완전하고 송신 버퍼가 이용가능한지를 식별한다. 접속 도달 디스크립터 타입은 호스트에게 신규한 접속이 도달하였음을 통지하고 접속 식별자를 포함한다. 접속 완료 디스크립터 타입은 호스트에게 접속 요청이 성공하였는지 또는 실패하였는지를 통지한다. 청취 응답 디스크립터 타입은 제출된 청취 요청의 성공/실패를 나타낸다. 종료 응답 디스크립터 타입은 제출된 폐쇄 요청의 성공/실패를 나타낸다. 삭제 응답 디스크립터 타입은 제출된 삭제 요청의 성공/실패를 나타낸다. 접속 속성 디스크립터 타입은 이전 접속 속성값 또는 신규 값 성공/실패를 나타낸다. 네트워크 어댑터 속성 디스크립터 타입은 이전 네트워크 어댑터 속성값 또는 신규 네트워크 어댑터 속성값 성공/실패를 나타낸다.

[0065] 본 발명의 예시적인 실시예에서, 버퍼 송신 요청, 버퍼 가용 요청, 버퍼 수신 응답, 및 버퍼 가용 응답 디스크립터에 대한 디스크립터 콘텐츠 파일은 다음의 필드로 모두 포맷팅된다.

[0066] 베이스(base) 64비트 버퍼의 기초 물리 어드레스

[0067] Len 32비트 바이트 단위의 버퍼의 길이

[0068] Conn ID 64비트 네트워크 어댑터에 의해 제공된 고유의 접속 식별자

[0069] ASAP 1비트 가능한 신속하게 DMA에 요청(후술됨)

[0070] 변형(Modify) 1비트 이 버퍼가 변형되었는지 여부를 나타냄(후술됨)

[0071] 접속 ID(Conn ID)는 접속을 고유하게 식별하기 위한 값이고, 접속 요청에 응답하여 그리고 접속 도달에 대한 응답으로서 오프로드 네트워크 어댑터에 의해 제공된다. 접속 ID 0(제로)은 "접속 없음"을 의미하기 위해 예비된다. 이것은, 예를 들어, 버퍼가 임의의 접속용으로(예를 들어, 아직 ID를 갖지 않는 수동적으로 수용된 접속에 대한 데이터용으로) 사용될 수도 있다. 임의의 특정 접속과 관련되지 않은 버퍼는 "벌크 버퍼"로서 지칭된다.

[0072] ASAP 및 변형 필드는 버퍼 송신 요청 디스크립터용으로만 사용된다. ASAP 비트는 가능하면 신속하게 DMA된 이러한 버퍼를 갖기 위한 소망을 나타낸다. 변형 비트는, 오프로드 네트워크 어댑터에 제공된 마지막 시간 이래로 이러한 특정 버퍼가 변경되었는지 여부를 오프로드 네트워크 어댑터에게 통지하기 위한 것이다. 이것은 오프로드 네트워크 어댑터로 하여금 로컬 메모리에 이 버퍼의 카피를 이미 갖고 있는지 여부를 판정하고, 따라서, DMA 전송이 가능하게 한다.

[0073] 제어 디스크립터는 제어 버퍼를 기술하며, 이는 차례로 가변 개수의 임의-길이 속성 튜플(tuples)을 포함한다. 제어 디스크립터, 접속 요청, 종료 요청, 청취 요청, 삭제 요청 및 그 각각의 응답에 대한 디스크립터 콘텐츠 필드는 다음의 필드로 모두 포맷팅된다.

[0074] 개수 8비트 제어 버퍼 내의 속성 튜플의 수

[0075] 베이스 64비트 제어 버퍼의 기초 물리 어드레스

[0076] Len 32비트 바이트 단위의 제어 버퍼의 길이

[0077] Conn ID 64비트 고유의 접속 식별자

[0078] 접속 속성 요청, 오프로드 네트워크 어댑터 속성 요청, 및 그 각각의 응답에 대한 제어 버퍼 및 디스크립터 콘텐츠 필드는 다음의 필드로 모두 포맷팅된다.

[0079] 획득/설정 1비트 속성이 검색 또는 업데이트되어야 하는지를 나타냄

[0080] 속성 15비트 판독/기입용 속성을 식별

[0081] 길이 32비트 속성 데이터의 길이

[0082] 값 N/A 길이가 이전 필드에 의해 특정되는 실제 속성값

[0083] 전술한 제어 디스크립터는 가능한 일반적인 것으로 의도된다. 제어 디스크립터에 의해 특정될 수도 있는 속성의 볼륨으로 인해, 그 디스크립터 모두가 여기에 예시될 수는 없다. 네트워크 인터페이스 제어 속성의 예는 IP 어드레스, 도메인 네임, 및 라우팅 정보를 포함한다. 접속당(per-connection) 제어 속성의 예는 수신 윈도우 사이즈, 네이글(Nagle) 알고리즘 설정, 및 SACK 지원을 포함한다.

[0084] 본 발명에 있어서, 오프로드 네트워크 어댑터(730)는 본 발명의 오프로드 네트워크 어댑터 프로그래밍 인터페이스를 사용하기 위해 오프로드 네트워크 어댑터(730)의 펌웨어, ASIC 등에서와 같은 로직을 가진다. 즉, 오프로드 네트워크 어댑터(730)는 요청 디스크립터를 인식하고 요청 디스크립터 및 대응하는 데이터를 처리하는 로직, 및 출력 디스크립터 테이블(724)에 기입될 응답 디스크립터를 생성하는 로직을 가진다. 유사하게, 호스트 시스템의 오퍼레이팅 시스템(715), 오퍼레이팅 시스템(715)에 의해 로딩된 디바이스 드라이버 등은 입력 디스크립터 테이블(722)에 기입될 요청 디스크립터를 생성하며 출력 디스크립터 테이블(724)로부터 판독된 응답 디스크립터를 인식하는 로직, 및 응답 디스크립터 및 대응하는 데이터를 처리하는 로직을 가진다.

[0085] 본 발명의 오프로드 네트워크 어댑터 프로그래밍 인터페이스의 디스크립터를 이용하여 호스트 시스템과 네트워크 어댑터 간의 상호작용의 일반 개요가 제공되면, 다음 설명은 이러한 인터페이스가 오프로드 네트워크 어댑터를 이용하여 개선된 접속 수립, 메모리 관리, 및 데이터의 수신을 얼마나 용이하게 하는지를 설명할 것이다.

[0086] **<u>접속 수립</u>**

[0087] 본 발명의 일 양태는 오프로드 네트워크 어댑터에 대한 접속 상태 정보의 유지 및 접속 수립을 오프로딩하는 능력이다. 접속 수립 및 상태 정보 유지의 이러한 오프로딩의 결과로서, 호스트 시스템과 오프로드 네트워크 어댑터 사이에 필요한 통신의 수는 감소될 수도 있다. 또한, 이후 설명되는 바와 같이, 오프로드 네트워크 어댑터로의 이들 함수의 오프로딩은, 공지의 컴퓨팅 시스템에서 존재하는 단편적 통지보다는 호스트 시스템에 대한 수립 접속 및 상태 정보의 벌크 통지를 허용한다.

[0088] 도 8은 본 발명의 일 예시적인 실시예에 따라 통신 접속을 수립할 경우에 호스트 시스템과 오프로드 네트워크 어댑터 간의 통신의 예시적인 다이어그램이다. 도 8에 도시된 바와 같이, 아웃바운드 접속의 수립은, 접속이 수립될 것을 요청하는 오퍼레이팅 시스템(815)에 의해 애플리케이션(805)로부터의 요청의 수신에 의해 개시된다. 결과적으로, 오퍼레이팅 시스템(815)은 접속 요청 디스크립터를 생성하여, 입력 디스크립터 테이블(822)에 기입한다. 접속 요청 디스크립터 및 관련 제어 버퍼는 요청된 접속을 수립하기 위해 요구된 모든 정보를 포함한다. 예를 들어, 제어 버퍼 및 접속 요청 디스크립터는 원격 및 로컬 접속을 참조하기 위해 AF_INET, SOCK_STREAM, IP VERSION 정보 및 접속 식별자를 포함할 수도 있다.

[0089] 오프로드 네트워크 어댑터(830)는 입력 디스크립터 테이블(822)로부터 접속 요청 디스크립터를 판독하고, 그 후, 오프로드 네트워크 어댑터(830) 내의 접속 수립 로직(832)은 접속 요청 디스크립터에 수신된 정보에 기초하여 접속의 수립을 시도한다. 접속 요청 디스크립터에 기초한 접속의 수립은 접속을 위한 소켓 디스크립터, 즉, 호스트 시스템 및 원격 컴퓨팅 디바이스의 소켓을 기술하는 데이터 구조를 수립하는 단계, 접속 식별자를 접속체와 관련시키는 단계, 및 접속을 위해 오프로드 네트워크 어댑터(830)에 버퍼를 할당하는 단계를 포함한다. 즉, 오프로드 네트워크 어댑터는 종래의 시스템 호출 connect(), setsockopt(), bind(), accept() 등과 관련된 동작을 수행할 수도 있다. 접속이 수립되거나 지속기간 타임-아웃 조건과 같은 에러 조건이 충족될 경우에만, 호스트 시스템(810)은 접속 수립 동작의 결과적인 상태를 통지받는다.

[0090] 이러한 응답은 출력 디스크립터 테이블(824)에 대한 하나 이상의 응답 디스크립터의 기입일 수도 있다. 예를 들어, 접속 완료 디스크립터가 오프로드 네트워크 어댑터(830)에 의해 생성되어 출력 디스크립터 테이블(824)에 기입될 수도 있어서, 접속이 수립되었다는 것은 호스트 시스템(810)에 통보한다.

[0091] 인바운드 접속의 수립은 약간 상이한 방식으로 수행된다. 애플리케이션이 특정 포트상의 접속에 대해 "청취(listen)"하는 능력을 요청하는 경우에, 오퍼레이팅 시스템(815)은 입력 디스크립터 테이블(822)에 청취 요청 디스크립터를 기입할 수도 있다. 청취 요청 디스크립터는 청취하려는 포트 및 접속이 청취될 프로토콜을 식별한다. 그 후, 오프로드 네트워크 어댑터(820)의 접속 수립 로직(832)은 입력 디스크립터 테이블(822)로부터 청취 요청 디스크립터를 판독하고 적절한 인커밍 소켓상의 접속을 수립하는데 필요한 동작은 수행한다. 이것은 예를 들어, 종래의 accept() 및 bind() 시스템호출과 유사한 동작을 수행하지만, 오프로드 네트워크 어댑터(830)내에서 이들을 수행하는 것을 포함할 수도 있다. 접속이 수립되거나 에러 조건(타임-아웃 조건의 지속기간과 같은)이 충족될 때만, 접속 상태의 결과가 호스트 시스템(810)에 통지된다. 공지된 "오프로드" 구현에서, 호스트 시스템은 접속 수립의 각 스테이지에서 상호작용한다. 본 발명은 접속을 접속하거나 청취하기 위해 하이 레벨 커맨드를 이슈하고 접속이 수립되거나 타임-아웃 또는 에러 조건이 충족될 때만 응답한다.

[0092] 접속이 수립될 때, 접속에 관한 정보는 오프로드 어댑터 메모리(834)의 접속 상태 데이터 구조에 유지된다. 이러한 상태 정보는 수립된 접속을 통해 데이터를 전송 및 수신하기 위해 이용된다. 이러한 상태 정보는 또한 이하 논의하는 바와 같이, 호스트 시스템(810)에 의해 유지되는 접속 상태 정보를 업데이트하기 위해 이용될 수도 있다.

[0093] 전술한 설명으로부터 알 수 있는 바와 같이, 오프로드 네트워크 어댑터내에서 접속 수립 동작을 수행하고 본 발명의 오프로드 네트워크 어댑터 수행 인터페이스를 이용한 중요한 결과중 하나는 호스트 시스템과 네트워크 어댑터 사이의 통신이 접속의 수립 동안 최소화된다는 것이다. 그 결과, 처리할 호스트 시스템에 대한 메시지가 적다. 이것은 시스템이 다수의 접속이 수립되고 분해되는 서버 컴퓨팅 시스템일 때 특히 중요하다.

[0094] 전술한 바와 같이, 본 발명의 일 실시예에서, 호스트 시스템에는 접속이 수립되거나 에러 조건이 인카운터된 이후에 접속의 상태가 통지된다. 따라서, 그 결과, 접속 완료 응답 디스크립터가 접속이 수립되거나 접속을 수립하기 위해 시도가 실패하는 각 시간에 출력 디스크립터 테이블(824)에 기입된다. 출력 디스크립터 테이블(824)로의 각 접속 완료 응답 디스크립터의 기입으로, 신규한 응답 디스크립터가 처리하는 출력 디스크립터 테이블(824)에 존재한다는 것을 호스트 시스템(810)에 통지하기 위해 인터럽트가 오퍼레이팅 시스템(815)에 생성되고 전송될 수도 있다.

[0095] 출력 디스크립터 테이블(824)에 기입되는 접속 완료 응답 디스크립터의 횟수를 최소화하고, 따라서 호스트 시스템(810)에 생성되고 전송되는 인터럽트의 수를 최소화하기 위해, 본 발명은 다수의 상이한 방식에서 출력 디스크립터 테이블(824)로의 접속 완료 응답 디스크립터의 기입을 지연시킬 수도 있다. 호스트로의 접속 수립 상태 상태의 통지를 지연시키는 이점은 단일 통지에서 여러 접속의 집합에 대한 잠재성이다. 이러한 방식에서, 동일하거나 상이한 접속에 대한 복수의 완료 응답 디스크립터가 함께 "배치(batched)"될 수도 있고 오프로드 네트워크와 호스트 시스템 사이의 하나의 트랜잭션에서 호스트 시스템에 제공될 수도 있다.

[0096] 예를 들어, 구성 가능한 지연 값이 수립된 소켓 접속의 레이트에 기초하여 설정될 수도 있고, 이 레이트는 접속 요청이 수신되는 레이트이다. 이러한 지연 값은 집합내에서 각 접속의 상태를 지정하는 접속 완료 응답 디스크립터를 생성하기 이전에 오프로드 네트워크 어댑터(830) 메모리에 누적될 수도 있는 접속 수립 정보의 집합의 양을 식별할 수도 있다. 이러한 값은 오프로드 네트워크 어댑터(830)상의 메모리에 저장될 수도 있다.

[0097] 이 지연 값은 정적으로 또는 동적으로 결정될 수도 있고 접속 완료 응답 디스크립터를 이용하는 호스트 시스템으로의 통지와 접속의 수립 사이의 시간의 소정량, 수신된 접속 수립 상태 업데이트의 수, 즉, 접속 수립 동작의 성공/실패 등의 형태를 취할 수도 있다. 이 지연 값이 동적으로 결정되는 경우에, 예를 들어, 시간 주기 동안 수신된 접속의 레이트 또는 양, 소켓 접속 타이밍의 이력(historical) 관측 등에 기초하여 결정될 수도 있다. 예를 들어, 특정 소켓 수신 접속이 10 밀리초 동안 10개 접속 요청의 버스트를 가진 후 10초 동안 완전한 경우에, 호스트 시스템으로의 전체 통지를 감소시키기 위해 10개 접속이 이루어질 때까지 호스트 시스템으로의 모든 통지를 지연시키는 것이 현명할 수도 있다. 1초의 타임아웃 피처가 추가 소켓 접속을 대기하기 위해 이용될 수도 있다.

[0098] 출력 디스크립터 테이블(824)에 접속 완료 응답 디스크립터를 기입할 때를 결정하는 또 다른 옵션은 수립된 접속의 데이터 도달 이전에 대기하기 위한 오프로드 네트워크 어댑터(830)에 대한 것이다. 이러한 방식에서, 오

프로드 네트워크 어댑터(830)는 호스트 시스템(810)에 의해 처리하는 데이터가 수신되기 이전에 메모리에 수립된 접속에 관한 정보를 유지한다. 동시에, 접속 완료 응답 디스크립터는 호스트 시스템(810)에 접속의 수립을 통지하는 출력 디스크립터 테이블(824)에 기입될 수도 있고 버퍼 수신 응답 디스크립터는 수립된 접속을 통한 데이터의 수신을 나타내는 출력 디스크립터 테이블(824)에 기입될 수도 있다.

[0099] 본 발명의 또 다른 실시예에서, 출력 디스크립터 테이블(824)을 통한 호스트 시스템으로의 통지는 특정 데이터 패턴이 접속을 통해 수신되기 이전에 지연될 수도 있다. 이들 특정 데이터 패턴은 예를 들어, 특정 HTTP GET 요청, 단일 유닛으로서 처리될 수 있는 데이터의 시퀀스의 종단을 나타내기 위해 미리 결정된 특정 메타 태그 등일 수도 있다.

[0100] 이러한 데이터 패턴이 수립된 접속을 통해 수신되면, 오프로드 네트워크 어댑터(830)는 데이터 패턴이 수신되기 이전의 시간 주기 동안 성공적으로 수립되거나 실패한 모든 접속을 식별하는 출력 디스크립터 테이블(824)에 접속 완료 응답 디스크립터를 기입할 수도 있다. 이러한 방식에서, 호스트 시스템(810)에는 호스트 시스템(810)이 처리할 특정 데이터를 갖기 이전에 신규한 접속의 수립이 통지된다. 다시 말해, 호스트 시스템은 호스트 시스템이 수행할 특정한 것이 없는 경우에 디스크립터를 건드리지 않는다. 이러한 "무엇인가"는 검색할 데이터 패턴에 의해 정의된다.

[0101] 따라서, 본 발명은 수립된 접속 또는 접속 수립에서의 실패의 통지의 집합을 허용하여, 호스트 시스템으로 전송되는 통지의 수가 최소화된다. 이것은 호스트 시스템에 의해 수행되어야 하는 처리량을 감소시키고 호스트 시스템이 호스트 시스템상에서 구동하는 애플리케이션을 처리하기 위해 그 자원을 이용할 수 있게 한다.

[0102] 본 발명을 이용하면, 접속 수립이 오프로드 네트워크 어댑터(830)에 의해 수행되기 때문에, 수립된 접속의 상태가 오프로드 네트워크 어댑터(830)의 메모리에 유지된다. 그러나, 호스트 시스템(810)은 실패극복, 네트워크 에러의 경우에 이러한 상태 정보를 갖거나, 라우팅 결정을 할 필요가 있을 수도 있다. 따라서, 본 발명은 오프로드 네트워크 어댑터(830)에 유지된 수립된 접속에 대한 상태 정보를 호스트 시스템(810)으로 이동시키는 메커니즘을 제공한다.

[0103] 본 발명의 하나의 예시적인 실시예에서, 접속 속성 응답 디스크립터가 주기적으로 생성될 수도 있고 출력 디스크립터 테이블(824)에 기입될 수도 있다. 이러한 접속 속성 응답 디스크립터는 접속 각각의 현재 상태를 식별한다. 호스트 시스템(810)에는 오퍼레이팅 시스템(815)으로 인터럽트를 전송함으로써 출력 디스크립터 테이블(824)로의 접속 속성 응답 디스크립터의 추가가 통지된다. 그 후, 호스트 시스템(810)은 접속 속성 응답 디스크립터를 판독하고 호스트 시스템의 접속 상태 정보가 업데이트되도록 이것을 처리한다. 따라서, 호스트 시스템(810)에는 호스트 시스템(810)이 라우팅 결정을 수행하고 네트워크 에러 실패극복의 경우에 적절한 동작을 수행할 수도 있는 업데이트 정보가 제공된다.

[0104] 따라서, 본 발명은 접속 수립 동안 호스트 시스템과 오프로드 네트워크 어댑터 사이의 통신이 최소화되도록 오프로드 네트워크 어댑터에 접속 수립을 오프로딩하는 메커니즘을 제공한다. 이것은 호스트 시스템이 단일 접속 요청 디스크립터에서 오프로드 네트워크 어댑터로 벌크 접속 수립 요청을 전송할 수 있게 하고, 그 후, 호스트 시스템과의 또 다른 통신은 예를 들어, 소정의 수의 접속이 수립되고, 소정의 양의 데이터가 접속에 도달하고, 소정량의 시간이 경과하고, 소정의 데이터 패턴이 수신되는 등의 특정 기준이 충족되기 이전에 오프로드 네트워크 어댑터에 의해 필요로 하지 않는다. 유사하게, 호스트 시스템은 오프로드 네트워크 어댑터에 특정 포트에 대한 접속을 청취할 것을 지시한 후 이들 접속을 수용 및 바인드할 수도 있다. 그 결과, 호스트 시스템은 하나의 청취 요청 디스크립터를 전송할 수도 있고 소정의 기준이 청취될 포트에 대한 접속의 수립에 관하여 충족되기 이전에 다시 통신되지 않을 수도 있다. 또한, 본 발명은 오프로드 네트워크 어댑터에서 접속 상태 정보를 분류한 후 라우팅 결정 및 네트워크 에러 또는 실패극복의 경우에 이용하기 위해 호스트에 이러한 상태 정보를 이동시키는 메커니즘을 제공한다.

[0105] 도 9 및 10은 본 발명의 하나의 예시적인 실시예에 따른 본 발명의 엘리먼트의 동작을 나타내는 플로우차트이다. 이들 플로우차트 예시, 및 후술되는 다른 플로우차트 예시의 각 블록, 및 플로우차트 예시에서의 블록의 조합이 컴퓨터 프로그램 명령에 의해 구현될 수도 있다. 이들 컴퓨터 프로그램 명령은 머신을 제조하도록 프로세서 또는 다른 프로그램 가능한 데이터 처리 장치에 제공될 수도 있어서, 프로세서 또는 다른 프로그램 가능한 데이터 처리 장치상에서 실행하는 명령은 플로우차트 블록 또는 블록들에 특정된 기능을 구현하는 수단을 생성한다. 이들 컴퓨터 프로그램 명령은 또한 프로세서 또는 다른 프로그램 가능한 데이터 처리 장치로 하여금 특정 방식으로 기능하도록 지시할 수 있는 컴퓨터 판독가능 메모리 또는 저장 매체에 저장될 수도 있어서, 컴퓨터 판독가능 메모리 또는 저장 매체에 저장된 명령은 플로우차트 블록 또는 블록들에 특정된 기능

들을 구현하는 명령 수단을 포함하는 제품을 제조한다.

[0106] 따라서, 플로우차트 예시의 블록들은 특정 기능을 수행하는 수단의 조합, 특정 기능을 수행하는 단계의 조합 및 특정 기능을 수행하는 프로그램 명령 수단을 지원한다. 또한, 플로우차트 예시의 각 블록, 및 플로우차트 예시의 블록들의 조합은 특정 기능 또는 단계를 수행하는 특정 목적 하드웨어 기반 컴퓨터 시스템, 또는 특정 목적 하드웨어 및 컴퓨터 명령의 조합에 의해 구현될 수도 있다.

[0107] 도 9는 오프로드 네트워크 어댑터를 이용하여 접속을 수립할 때 본 발명의 호스트 시스템의 예시적인 동작을 나타낸 플로우차트이다. 도 9에 도시되어 있는 바와 같이, 동작은 애플리케이션으로부터 접속 수립 요청을 수신함으로써 시작한다(단계 910). 이러한 접속 수립 요청은 예를 들어, 특정 요청을 수립하기 위한 요청 또는 특정 포트에서의 접속에 대해 청취하기 위한 요청일 수도 있다. 접속 수립 요청 디스크립터가 입력 디스크립터 테이블에 기입된다(단계 920). 이러한 접속 수립 요청 디스크립터는 예를 들어, 접속 요청 디스크립터 또는 청취 요청 디스크립터일 수도 있다.

[0108] 그 후, 동작은 오프로드 네트워크 어댑터로부터의 접속 수립 동작의 완료에 관한 응답을 대기한다(단계 930). "대기"한다는 것은 응답이 수신되기 이전에는 이러한 접속에 관하여 호스트 시스템에 의해 다른 동작이 수행되지 않는다는 것을 의미한다. 명백하게, 호스트 시스템은 이러한 "대기"가 발생하는 동안 다른 동작을 수행한다.

[0109] 응답이 수신되었는지에 관한 결정이 이루어진다(단계 940). 그렇지 않은 경우, 접속 수립 요청이 타임-아웃되었는지에 관한 결정이 이루어진다(단계 950). 그렇지 않은 경우에, 동작은 단계 930으로 복귀하고 계속 대기한다. 접속 수립 요청이 타임-아웃된 경우에, 삭제 요청 디스크립터가 입력 디스크립터 테이블에 기입되고(단계 960) 동작은 종료한다.

[0110] 응답이 수신된 경우에, 접속 완료 응답 디스크립터가 출력 디스크립터 테이블로부터 판독된다(단계 970). 그 후, 접속 완료 응답 디스크립터가 호스트 시스템에 의해 처리되고(단계 980) 동작은 종료한다.

[0111] 단계 920에서 입력 디스크립터 테이블에 기입된 원래의 접속 수립 요청 디스크립터는 수립될 복수의 접속, 즉, 벌크 접속 수립 요청을 지정할 수도 있다는 것을 유의해야 한다. 따라서, 본 발명에 따르면, 호스트는 오프로드 네트워크 어댑터로 오프로딩되는 이들 접속을 수립하는데 필요한 모든 처리과 이러한 벌크 접속 수립을 수행하기 위해 입력 디스크립터 테이블과 하나의 트랜잭션만을 수행할 필요가 있다. 유사하게, 원래의 접속 수립 요청 디스크립터가 "청취" 요청 디스크립터인 경우에, 오프로드 네트워크 어댑터가 포트를 청취하면서 다수의 접속이 수립될 수도 있지만, 하나의 트랜잭션만이 이들 접속의 수립을 개시하기 위해 호스트 시스템에 의해 수행된다.

[0112] 도 10은 본 발명의 하나의 예시적인 실시예에 따라 접속을 수립할 때 오프로드 네트워크 어댑터의 예시적인 동작을 나타낸 플로우차트이다. 도 10에 도시되어 있는 바와 같이, 동작은 입력 디스크립터 테이블로부터 접속 수립 요청 디스크립터를 판독함으로써 시작한다(단계 1010). 접속 수립 동작은 소켓 디스크립터, 접속 식별자 등을 생성하고, 접속 수립 요청 디스크립터에서 식별된 접속(들)을 수립하기 위해 수행된다(단계 1020). 수립된 접속 각각에 관한 상태 정보는 어떤 접속이 수립되었는지 및 접속이 이전의 호스트 시스템에서의 통지로 인해 실패하였는지를 식별하는 정보에 따라 메모리에 저장된다(단계 1030).

[0113] 접속 완료 응답 디스크립터를 대기하는 지연 기준이 충족되었는지에 관한 결정이 이루어진다(단계 1040). 상기 언급한 바와 같이, 지연 기준은 다수의 상이한 여태를 취할 수도 있다. 예를 들어, 지연 기준은 호스트 시스템으로의 최종 통지, 접속중의 하나 동안 도달하는 소정의 데이터 양, 수신되는 특정 데이터 패턴, 호스트 시스템으로의 최종 통지로 인한 소정의 시간량 등으로 인해 수립되는 다수의 접속일 수도 있다.

[0114] 지연 기준이 충족되지 않은 경우에, 동작은 단계 1020으로 복귀하고 메모리에 유지된 접속 수립 정보 및 상태 정보와의 접속 수립을 계속한다. 지연 기준이 충족된 경우에, 접속 완료 응답 디스크립터가 생성되며 호스트 시스템으로의 최종 통지로 인해 수립이 실패된 접속 및 수립된 접속을 식별하는 출력 디스크립터 테이블에 기입된다(단계 1050). 그 후, 동작은 종료한다.

[0115] 따라서, 본 발명은 오프로드 네트워크 어댑터를 이용하여 접속을 수립하는 개선된 메커니즘을 제공한다. 본 발명의 이러한 양태는 호스트 시스템과 오프로드 네트워크 어댑터 사이의 통신이 최소화되는 벌크 접속 수립에 특히 적합하여서, 다수의 접속이 호스트 시스템과 오프로드 네트워크 어댑터 사이의 최소 상호작용 양에 의해서만 수립될 수도 있다. 이것은 애플리케이션을 구동하고 다른 유용 작업을 수행하는데 그것의 자원을 집중하는 호

스트 시스템에 자유롭다.

[0116] **메모리 관리**

[0117] 접속 수립에 추가하여, 본 발명은 오프로드 네트워크 어댑터를 이용하는 데이터 처리 시스템에서 메모리 관리를 개선시킨다. 본 발명에 따른 메모리 관리는 데이터의 버퍼링된 송신 및 수신 뿐만 아니라 데이터의 제로-카피 전송 및 수신 모두를 허용한다. 또한, 본 발명은 임의의 수의 속성에 기초하여 특정 조건 중에서 공유될 수 있는 DMA 버퍼의 그룹화를 허용한다. 본 발명은 또한 벌크로 호스트 시스템에 통신될 수도 있도록 DMA 요청을 지연시키는 버퍼 동작의 부분 송신 및 수신, 및 호스트 시스템으로의 데이터의 신속한 전송을 위한 메커니즘을 제공한다.

[0118] 오프로드 네트워크 어댑터 프로그래밍 인터페이스는 사용자 메모리로의 더 많은 직접 액세스를 허용하는 더 신규한 API 뿐만 아니라 소켓 인터페이스와 같은 사용자-레벨 애플리케이션 프로그램 인터페이스(API)를 지원한다. 본 발명의 오프로드 구조는 데이터의 버퍼링된 송신 및 수신 뿐만 아니라 데이터의 제로-카피 전송 및 수신 모두를 허용한다. 오프로드 네트워크 어댑터의 관점으로부터, 버퍼링된 제로-카피 송신은 거의 동일하게 처리된다. 이들 2개 타입의 데이터 전송이 구별되는 방식은 호스트 시스템이 오프로드 네트워크 어댑터를 어떻게 이용하는지에 기초한다.

[0119] 도 11은 데이터의 버퍼링된 송신 및 수신이 이용되는 본 발명에 따른 메모리 관리 메커니즘을 도시하는 예시적인 다이어그램이다. 설명을 위해, 호스트 시스템(1110)과 또 다른 컴퓨팅 디바이스(도시 생략) 사이의 접속이 상술한 메커니즘을 통해 수립되었다고 가정한다. read()호출이 이러한 접속을 참조하여 이루어질 때, 애플리케이션 버퍼(1130)가 이러한 접속을 위해 수립될 수도 있다. 오퍼레이팅 시스템(1150)은 또한 네트워크 어댑터 또는 특정 접속 버퍼, 예를 들어, 애플리케이션 버퍼(1130)로 전송하기 이전에 데이터가 기입되는, 다양한 접속에 대한 데이터를 수신하는 벌크 버퍼로서 칭할 수도 있는 고정된 커널 버퍼(1140)를 포함할 수도 있다. 커널 버퍼(1140)는 접속 이슈 시간에서 생성되고 접속을 위한 애플리케이션 버퍼(1130)가 데이터가 접속상에 전송되기 이전에 포스트되지 않을 때 이용된다. 애플리케이션 버퍼(1130)가 데이터 전송 이전에 포스트되지 않은 경우에, 애플리케이션 버퍼는 데이터를 수신하기 위해 이용된다. 또 다른 방법으로는, 후술하는 바와 같이, 애플리케이션 버퍼(1130) 및 커널 버퍼(1140) 모두가 어떤 버퍼링된 송신 실시예에서 이용될 수도 있다.

[0120] 도 11에 도시되어 있는 바와 같이, 호스트 시스템(1110)이 오프로드 네트워크 어댑터(1120)를 통해 또 다른 컴퓨팅 디바이스로 데이터 전송을 소망하는 경우에, 호스트 시스템(1110)은 사용자 공간의 애플리케이션 버퍼(1130)로부터 오퍼레이팅 시스템 커널 공간의 오퍼레이팅 시스템(1150)의 고정된 커널 버퍼(1140)로 데이터를 카피한다. 이러한 고정된 커널 버퍼(1140)는 하나 이상의 수립된 접속에 대해 애플리케이션 버퍼(1130) 및 오프로드 네트워크 어댑터(1120)로부터 데이터를 수신하는 벌크 버퍼이다. 따라서, 호스트 시스템(1110)은 복수의 접속이 현재 오픈된 경우에 복수의 애플리케이션 버퍼(1130)를 가질 수도 있고, 이들 접속에 대한 데이터는 고정된 커널 버퍼(1140)를 통해 송/수신될 수도 있다.

[0121] 이러한 방식에서, 데이터는 오프로드 네트워크 어댑터(1120)에 의해 송신을 위해 요청된다. 그 후, 호스트 시스템(1110)은 전송하는 데이터를 가질 때 고정된 커널 버퍼(1140)를 식별하는 입력 디스크립터 테이블상에 버퍼 전송 디스크립터를 포스트할 수도 있다. 그 후, 오프로드 네트워크 어댑터(1120)는 입력 디스크립터 테이블로부터의 입력 전송 요청 디스크립터 판독에 응답하여, 고정된 커널 버퍼(1140)로부터 데이터를 판독할 수도 있고 네트워크(도시 생략)를 통해 수신지 컴퓨팅 디바이스로 데이터를 송신할 수도 있다. 그 후, 오프로드 네트워크 어댑터(1120)는 데이터 송신이 완료되었다는 것을 나타내는 출력 디스크립터 테이블상에 버퍼 이용가능 응답 디스크립터를 포스트할 수도 있다. 따라서, 버퍼링된 송신 메커니즘을 이용하는 데이터 전송에 의하면, 본 발명은 송신을 위해 애플리케이션 버퍼(1130)로부터 고정된 커널 버퍼(1140)로 데이터를 카피한다.

[0122] 버퍼링된 수신은 유사한 방식으로 작용한다. 버퍼링된 수신 동작에 의하면, 오프로드 네트워크 어댑터(1120)는 오프로드 네트워크 어댑터(1120)로부터 고정된 커널 버퍼(1140)로 데이터를 송신하기 위해 직접 메모리 액세스(DMA) 동작을 수행한다. 입력 디스크립터 테이블상에 호스트 시스템(1110)에 의해 호스트된 버퍼 이용가능 요청 디스크립터에 응답하여, 오프로드 네트워크 어댑터(1120)는 출력 디스크립터 테이블상에 버퍼 수신 응답 디스크립터를 포스트할 수도 있다. 그 후, 호스트 시스템(1110)은 출력 디스크립터 테이블로부터 버퍼 수신 응답 디스크립터를 판독할 수도 있고 사용자 공간에서 고정된 커널 버퍼(1140)로부터 애플리케이션 버퍼(1130)로 데이터를 카피하기 위해 read() 소켓 호출을 호출할 수도 있다.

[0123] 버퍼링된 송신은 애플리케이션 버퍼(1130)로부터 고정된 커널 버퍼(1140)로 또는 그 반대로 데이터를 전달하기 위해 수행되어야 하는 데이터 카피 동작의 수로 인해 최적 보다 느려지는 경향이 있다. 그러나, 버퍼링된 송신은 2개의 이점을 제공한다. 데이터가 호스트 커널 메모리, 즉, 고정된 커널 메모리(1140)에 유지되기 때문에, 메모리 압력은 버퍼가 전송되기 이전에 버퍼가 오프로드 네트워크 어댑터(1120)로 DMA될 필요가 없어서 오프로드 네트워크 어댑터(1120)상에서 감소된다. 또한, 오프로드 네트워크 어댑터(1120)가 실패하는 경우에, 데이터가 또 다른 네트워크 어댑터를 통해 전송되도록 호스트 시스템의 고정된 커널 버퍼에서 여전히 이용가능하기 때문에 실패-극복 달성이 더 용이하다.

[0124] 본 발명의 구성은 또한 오프로드 네트워크 어댑터와 호스트 시스템 사이의 데이터의 제로-카피 송신을 위한 메커니즘을 제공한다. 용어 "제로-카피"는 호스트 시스템에 의한 메모리 대 메모리 카피의 제거를 칭한다. 도 12는 본 발명의 하나의 예시적인 실시예에 따른 제로-카피 동작을 예시하는 예시적인 다이어그램이다. 데이터를 호스트 시스템(1210)으로/으로부터 송신하기 위해, 호스트 시스템은 사용자 애플리케이션을 차단하고 그것의 애플리케이션 버퍼(1230)를 고정시킨다. 그 후, 호스트 시스템(1210)은 애플리케이션 버퍼(1230)로/로부터 오프로드 네트워크 어댑터(1220)로 데이터를 DMA하도록 오프로드 네트워크 어댑터(1220)를 발생시킬 수도 있다.

[0125] 현재 시스템에서, 수립된 접속으로부터 판독하기 위해, 애플리케이션은 3개의 독립변수로 read() 소켓 호출을 호출한다. 제 1 독립변수는 사용하기 위한 소켓 디스크립터를 특정하고, 제 2 독립변수는 애플리케이션 버퍼(1230)의 어드레스를 특정하며, 제 3 독립변수는 버퍼의 길이를 특정한다. 판독은 소켓에 도달된 데이터 바이트를 추출하고 이들을 사용자의 버퍼 영역, 예를 들어, 애플리케이션 버퍼(1230)에 카피한다. 사용자의 버퍼 영역에 피트(fit)하는 것 보다 적은 데이터가 도달하는 경우에, read()는 모든 데이터를 추출하고 이것이 발견된 바이트의 수를 복귀시킨다.

[0126] 본 발명에 따른 시스템에서의 제로-카피에 의하면, 애플리케이션 버퍼(1230), 즉, DMA 버퍼의 생성은 디스크립터 통신 패킷이 생성되게 하고 호스트 시스템(1210)으로부터 오프로드 네트워크 어댑터(1220)로 전송되게 하며, 예를 들어, 버퍼 이용가능 요청 디스크립터 통신 패킷이 입력 디스크립터 테이블에 생성 및 포스트될 수도 있다. 디스크립터는 애플리케이션 버퍼(1230), 그것의 속성을 설명하며, 애플리케이션 버퍼(1230)를 수립된 접속에 대한 접속 정보와 관련시킨다. 애플리케이션 버퍼가 오프로드 네트워크 어댑터(1220)에 이용가능할 때, 및 read() 소켓호출이 수행될 때, DMA 동작이 오프로드 네트워크 어댑터(1220)로부터 애플리케이션 버퍼(1230)로 데이터를 전달하기 위해 수행된다. 그 후, read()호출 완료 통지에 대해 요청되는 DMA 데이터 속성을 설명하는 오프로드 네트워크 어댑터(1220)로부터의 응답 디스크립터가 생성되고, 예를 들어, 버퍼 이용가능 응답 디스크립터가 호스트 시스템의 입력 디스크립터 테이블에 생성 및 포스트될 수도 있다.

[0127] 오프로드 네트워크 어댑터(1220)는 그것의 기능을 수행하는데 사용하기 위해 메모리에 각 오픈 접속에 대한 정보를 유지한다는 것을 유의해야 한다. 이러한 정보는 오픈 접속과 관련된 애플리케이션 버퍼의 식별 뿐만 아니라 다른 접속 특정 정보를 포함할 수도 있다. 그 후, 이러한 정보는 오프로드 네트워크 어댑터(1220)가 그 자체와 호스트 시스템(1210)상의 애플리케이션 사이에서 데이터 통신을 필요로 할 때 이용된다.

[0128] 따라서, 본 발명에 의하면, 오프로드 네트워크 어댑터는 직접 메모리 액세스 동작을 이용하여 사용자 공간에서 애플리케이션 버퍼로 데이터를 직접 전송할 수도 있다. 그렇게 하는 경우에, 고정된 커널 버퍼로부터 애플리케이션 버퍼로의 데이터의 카핑이 회피된다. 물론, 본 발명은 모드, 즉, 버퍼링된 송신/수신 또는 제로-카피 전송/수신에서 동작할 수도 있거나, 교환가능하게 모든 모드 또는 대략 동시에 이용할 수도 있다. 즉, 어떤 데이터는 버퍼링된 송신/수신을 이용하여 호스트 시스템과 오프로드 네트워크 어댑터 사이에서 전달될 수도 있고 다른 데이터는 제로-카피 전송/수신을 이용하여 전달될 수도 있다. 예를 들어, 제로-카피 전송/수신은 애플리케이션 read()호출이 소켓상에서 각각의 데이터의 수신을 진행할 때 마다 이용될 수도 있다. 이러한 방식에서, 애플리케이션 버퍼는 수립된 접속상에 데이터를 수신하기 위해 사전-포스트될 것이다. read()호출이 소켓상에서 데이터의 수신을 진행하지 않는 경우에, 버퍼링된 송신/수신이 이용될 수도 있다.

[0129] 바람직한 실시예에서, 제로 카피 전송/수신은 데이터를 호스트 시스템으로/으로부터 전송/수신하는 바람직한 방식이다. 그러나, 제로 카피 전송/수신이 불가능한 상황이 발생할 수도 있다. 예를 들어, 애플리케이션 버퍼의 이용가능 메모리가 초과되거나 애플리케이션 버퍼가 이용가능하지 않는 경우에, 오프로드 네트워크 어댑터는 직접 메모리 액세스 동작을 이용하여 애플리케이션 버퍼로 데이터를 직접 전송할 수 있다. 그 결과, 공유된 버퍼로의 데이터의 버퍼링된 송신이 요청될 수도 있다.

[0130] 본 발명의 오프로드 네트워크 어댑터는 임의의 수의 속성에 기초하여 특정 접속중에 공유될 수 있는 애플리케이션 버퍼를 그룹화하는 능력을 갖는다. 바람직한 실시예에서, 애플리케이션 버퍼의 그룹화는 접속 포트 수에 기

초한다. 즉, 동일한 포트 수를 모두 이용하는 애플리케이션 버퍼는 애플리케이션 버퍼를 공유할 수도 있다. 예를 들어, 웹 서빙 시나리오에서, 포트 마다 다중 접속이 존재할 수도 있다. 하나의 예가 웹 서버의 TCP/IP 포트 80이다. 포트 80을 통해 정보를 요청하는 수천의 클라이언트 HTTP 접속이 존재할 수도 있다. 포트 80에 할당된 버퍼는 그룹화될 수도 있고, 즉, 할당된 버퍼의 풀(pool)이 포트 80상에 들어오는 이들 정보 요청을 처리하기 위해 수립될 수도 있다.

[0131] 전송 동작에 대한 애플리케이션 버퍼 공유는 호스트 시스템 기반 브로드캐스트 또는 멀티캐스트 타입 접속에 대한 데이터 재이용을 허용한다. 즉, 데이터가 공유된 애플리케이션 버퍼에 1회 기입만 될 필요가 있지만, 이들 애플리케이션 버퍼를 공유하는 복수의 접속을 통해 송신될 수도 있다. 수신된 데이터에 대한 애플리케이션 버퍼 공유는 낮은 대역폭 요건 또는 트래픽의 일시적인 버스트를 갖는 활성 접속을 위해 메모리의 더욱 효율적인 이용을 허용한다. 즉, 다중 접속이 버퍼에 대한 다수의 메모리가 낮은 대역폭 또는 일시적인 버스트 접속과 이용되지 않을 수도 있는 자체 전용 개별 애플리케이션 버퍼를 가져야 하는 것 보다 작은 공유된 애플리케이션 버퍼를 공유할 수도 있다. 또한, 애플리케이션 버퍼 공유는 개별 애플리케이션 및 프로세스가 수신되는 데이터를 공유하는 것을 허용한다.

[0132] 도 13은 본 발명의 하나의 예시적인 실시예에 따른 공유된 버퍼 배치를 예시하는 예시적인 다이어그램이다. 도시된 예에서, 3개의 프로세스(X, Y 및 Z)가 호스트 시스템(1310)상에서 현재 구동하고 있다. 5개의 접속(A, B, C, D 및 E)이 수립되었고 대응하는 애플리케이션 버퍼(1350-1370)가 이들 접속을 위해 호스트 시스템(1310) 메모리에 수립되었다. 애플리케이션 버퍼(1350 및 1360)는 데이터가 DMA 동작을 이용하여 직접 전송될 수도 있는 개별 애플리케이션 버퍼이다. 또 다른 방법으로, 데이터는 상술한 바와 같이, 버퍼링된 송신/수신 동작의 일부로서 고정된 커널 버퍼(1330)를 이용하여 이들 애플리케이션 버퍼(1350-1360)로 카피될 수도 있다.

[0133] 애플리케이션 버퍼(1370)는 접속 C, D 및 E 사이에 공유되는 공유된 애플리케이션 버퍼이다. 예를 들어, 접속 C, D 및 E는 소켓 접속을 위해 동일한 포트 수를 모두 이용할 수도 있고, 낮은 대역폭 접속일 수도 있으며, 따라서, 버퍼 공간을 공유할 수도 있다. 또 다른 방법으로는, 접속 C, D 및 E는 데이터의 멀티캐스팅 또는 브로드캐스팅을 위해 버퍼(1370)를 공유하려는 멀티캐스트 또는 브로드캐스트 그룹의 일부일 수도 있다.

[0134] 도 13에 도시되어 있는 바와 같이, 데이터의 버퍼링된 송신/수신 전달이 이용될 때, 데이터는 호스트 시스템(1310)의 오퍼레이팅 시스템(1340)에서 오프로드 네트워크 어댑터(1320)로부터 고정된 커널 버퍼(1330)로, DMA 동작을 이용하여 먼저 전송된다. 출력 버퍼 테이블에서 버퍼 이용가능 요청 디스크립터를 포스트하는 호스트 시스템(1310)에 응답하여, 오프로드 네트워크 어댑터(1320)는 입력 디스크립터 테이블에 버퍼 수신 응답 디스크립터를 포스트한다. 그 후, 호스트 시스템(1310)은 접속 C, D 및 E를 위해 고정된 커널 버퍼(1330)로부터 공유된 애플리케이션 버퍼(1370)로 데이터를 카피하기 위해 read()를호출할 수도 있다. 이들 공유된 애플리케이션 버퍼(1370)로부터, 데이터는 공유된 애플리케이션 버퍼(1370)를 공유하는 하나 이상의 프로세스에 의해 판독될 수도 있다. 예를 들어, 프로세스 Z는 공유된 버퍼(1370)로부터 데이터를 판독할 수도 있다. 접속 C, D 또는 E상에서 데이터를 청취하는 임의의 프로세스가 고정된 커널 버퍼(1330)로부터 공유된 버퍼(1370)로 그것의 접속상에서 데이터를 판독하기 위해 이들 동작을 수행할 수도 있다.

[0135] 이와 달리, 개별 애플리케이션 버퍼(1350 및 1360)와 같이, 접속(C, D 및 E)에 대한 데이터가 오프로드 네트워크 어댑터(1320)로부터 직접 공유된 버퍼(1370)로 DMA될 수도 있다. 이러한 방식에서, 본 발명의 제로 카피 구현은 복수의 접속으로부터 전송/수신하기 위해 데이터를 홀딩하는데 공유된 버퍼(1370)를 이용할 수도 있다.

[0136] 공유된 버퍼(1370)가 특히 유용한 하나의 경우는 애플리케이션이 데이터를 수신하는 애플리케이션 버퍼를 수립하기 이전에 호스트 시스템(1310) 메모리로 데이터를 DMA할 필요가 있을 때이다. 예를 들어, 이것은 오프로드 어댑터(1320)에 연속 수신되는 데이터가 소정의 임계값은 초과하고 오프로드 네트워크 어댑터가 메모리를 다 소모한 위험에 있을 수 있을 때 발생할 수도 있다. 이러한 시나리오가 존재한다고 가정하는 경우에, 호스트 메모리에서 공유된 시스템 버퍼(1370)로의 데이터의 중간 카피는 이러한 상황을 경감시키기 위한 목적이다. 즉, 데이터는 버퍼(1350)와 같은 전용 접속 애플리케이션 버퍼 보다는 오픈 접속 모두에 대한 공유된 버퍼(1370)로 카피될 수도 있다.

[0137] 따라서, 호스트 시스템과 오프로드 네트워크 어댑터 사이의 제로 카피 데이터 전송과 관련된 이점에 부가하여, 본 발명은 또한 접속이 접속 버퍼에 의해 사용된 호스트 시스템 메모리의 양을 최소화하기 위해 버퍼를 공유할 수도 있는 메커니즘을 제공하고, 오프로드 네트워크 어댑터 메모리가 초과하는 경우에 데이터를 처리하는 메커니즘을 제공하며, 전용 접속 버퍼에 할당된 이용되지 않은 호스트 시스템 메모리를 회피하는 메커니즘을 제공한다.

[0138] 전술한 메모리 관리 메커니즘에 부가하여, 본 발명은 또한 수립된 접속에 대한 부분 수신 및 전송 버퍼를 제공한다. 본 발명의 "부분 수신 및 전송 버퍼"는 수신 데이터는 애플리케이션에 대해 이미 수신/전송된 데이터를 갖는 버퍼에 수신 데이터를 첨부하기 위한 본 발명의 능력을 칭한다. 이 버퍼는 할당된 2개의 개별 버퍼 보다는 애플리케이션 데이터 전송을 위해 재이용된다.

[0139] 도 14는 본 발명의 하나의 예시적인 실시예에 따라 부분 수신/전송 버퍼가 동작하는 방식을 예시한다. 부분 수신/전송 버퍼를 이용하여, 호스트 시스템(1410)은 오프로드 네트워크 어댑터(1420)에 특정 접속을 위해 할당되는 애플리케이션 버퍼를 통지한다. 예를 들어, 버퍼 이용가능 요청 디스크립터는 입력 디스크립터 테이블에 포스트될 수도 있다. 이러한 방식에서, 호스트 시스템(1410)은 애플리케이션 버퍼(1430)의 소유권을 오프로드 네트워크 어댑터(1420)로 핸드오버한다.

[0140] 그 후, 오프로드 네트워크 어댑터(1420)는 접속을 통해 데이터를 수신하고 호스트 시스템(1410)상의 애플리케이션 버퍼(1430)로 데이터를 DMA한다. 그 후, 오프로드 네트워크 어댑터(1420)는 출력 디스크립터 테이블에 버퍼 수신 응답 디스크립터를 포스트할 수도 있다. 도시된 예에서, 애플리케이션 버퍼(1430)로 DMA된 데이터는 애플리케이션 버퍼(1430)를 부분적으로 채우는데 충분하다.

[0141] 호스트 시스템(1410)이 애플리케이션 버퍼(1430)에서의 데이터의 도달을 통지할 때, 네트워크 인터페이스는 호스트 시스템(1420)으로 이러한 "부분" 애플리케이션 버퍼(1430)의 제어를 핸드오버한다. 초기 버퍼의 임의의 나머지 부분은 여전히 오프로드 네트워크 어댑터(1420)의 제어하에 있다. Read()호출의 의미는 응답에서 "바이트 오프셋"의 추가를 요구한다. 호스트 시스템(1410)에서의 애플리케이션은 복귀된 데이터의 오프셋+길이가 원래 애플리케이션 버퍼(1430)의 전체 길이와 동일할 때 호스트 시스템(1410)으로 복귀되는 애플리케이션 버퍼(1430)의 풀 제어를 알 것이다. 데이터의 오프셋+길이가 원래 애플리케이션 버퍼(1430)의 전체 길이와 동일하지 않은 경우에, 오프로드 네트워크 어댑터(1420)는 버퍼의 부분 제어를 여전히 유지한다. 또 다른 방법으로는, 애플리케이션 버퍼(1430)에 대한 데이터의 최종 전달을 나타내는 추가 필드가 제공될 수 있다. 이것이 애플리케이션 버퍼(1430)에 대한 데이터의 최종 전달인 경우에, 제어는 호스트 시스템(1410)으로 복귀되고 오프로드 네트워크 어댑터(1430)는 애플리케이션 버퍼(1430)의 부분 제어를 유지하지 않는다.

[0142] 그 후, 추가 데이터가 접속을 통해 수신되는 경우에, 오프로드 네트워크 어댑터(1420)는 이러한 추가 데이터를 호스트 시스템(1410)상의 동일한 애플리케이션 버퍼(1430)로 이러한 추가 데이터를 DMA할 수도 있어서, 데이터가 애플리케이션 버퍼(1430)에 첨부된다. 그 후, 호스트 시스템(1410)은 출력 디스크립터 테이블에서의 또 다른 버퍼 수신 응답 디스크립터의 포스팅을 통하는 것과 같이, 오프로드 네트워크 어댑터(1420)에 의해 추가 데이터가 접속에 도달하였다는 것이 통지된다.

[0143] 전술한 바와 같은 이러한 메커니즘에 의하면, 네트워크 패킷 사이즈가 호스트 메모리 버퍼 사이즈와 동일하지 않는 경우에 단편화(fragmentation)가 문제일 수도 있다. 그러나, 대형 연속 가상 버퍼가 애플리케이션 이용을 위해 제공되는 경우에, 버퍼 단편화는 가상 연속 공간 선호도를 보존하기 위해 이용될 수도 있다. 이것은 가상 메모리상에 연쇄 버퍼의 추가된 코어(chore)로부터 애플리케이션을 세이브한다.

[0144] 예를 들어, 송신될 데이터에 대해 4 메가바이트 애플리케이션 버퍼를 제공하는 애플리케이션 Read()호출을 고려한다. 이것은 예를 들어, 디스플레이를 위한 대형 데이터 파일 또는 멀티미디어 스트림 수신을 기대할 수도 있다. 오프로드 네트워크 어댑터는 이러한 데이터의 1500 바이트 부분을 이들이 네트워크로부터 수신될 때 애플리케이션 버퍼로 직접 복귀시킬 수 있다. 이러한 배치는 이러한 데이터가 애플리케이션 사이트상에서 데이터의 재조합의 추가 복잡성을 세이브하는 연속 가상(애플리케이션) 공간에 수신되는 것을 허용한다.

[0145] 한편, 오프로드 네트워크 어댑터(1420)는 애플리케이션 버퍼가 수신된 데이터의 위치를 최적화하기 위한 대형 연속 가상 버퍼의 일부가 아닐 때 단편화를 허용하도록 선택할 수도 있다. 단편화 허용은 오프로드 네트워크 어댑터(1430)로부터 호스트 시스템(1410)으로 및 그 반대로 전달되는 버퍼의 수를 감소시키는데 도움을 줄 수도 있다. 따라서, 데이터의 제로 카피 전송, 데이터의 버퍼링된 전송, 및 공유된 버퍼의 허용에 추가하여, 본 발명은 또한 접속에 의한 사용을 위해 할당된 버퍼의 수를 최소화하도록 부분적으로 채워진 버퍼의 재사용을 위한 메커니즘을 제공한다.

[0146] 전술한 바와 같이, 오프로드 네트워크 어댑터가 자신과 호스트 시스템 사이에서 데이터를 통신 및 전달하는 방식은 DMA 동작을 통한다. 접속의 수립과 같이, 오프로드 네트워크 어댑터는 오프로드 네트워크 어댑터 및 호스트 시스템으로/으로부터 데이터를 전달할 때 이들 DMA 동작을 지연시킬 수도 있어서, 데이터의 벌크 전달이 달성될 수도 있다. 즉, 오프로드 네트워크 어댑터는 호스트 시스템이 데이터 전송을 요청하자 마자 DMA 요청을

반드시 개시하지 않는다. 오프로드 네트워크 어댑터가 이것이 적절하다고 여길 때, 오프로드 네트워크 어댑터는 DMA 동작이 송신된 데이터에 대해 개시될 때를 결정할 수도 있다.

[0147] 예를 들어, 오프로드 네트워크 어댑터는 접속을 통해 전송하기 위해 오프로드 네트워크 어댑터의 메모리에 데이터를 이미 충분히 가진 경우에 접속을 통해 데이터를 전달하는 DMA 동작을 지연시킬 수도 있다. 오프로드 네트워크 어댑터는 다양한 기준, 예를 들어, 대역폭 및 지연의 프로덕트의 현재 추정, 밀집 윈도우, 오프로드 네트워크 어댑터에 대해 이용가능한 메모리 등에 기초하여 무엇이 데이터의 "충분한" 양을 구성하는지를 결정할 수도 있다. 오프로드 네트워크 어댑터는 또한 페어 큐잉(fair queuing), 접속과 관련된 애플리케이션과 관련된 서비스 품질, 서비스의 상이함 등과 같은 다른 가능한 기준에 기초하여 결정할 수도 있다.

[0148] 예를 들어, 애플리케이션 Read()호출이 전달될 데이터에 대해 4 메가바이트 버퍼를 제공하는 경우를 고려한다. 오프로드 네트워크 어댑터는 네트워크로부터 수신될 때 이러한 데이터의 1500 바이트 부분을 버퍼로 직접 복귀시킬 수 있다. 오프로드 네트워크 어댑터는 애플리케이션이 벌크 데이터 전송을 기대하는 매우 큰 버퍼를 제공하고 그 후, 추가 패킷 수신을 기대하는 네트워크로부터 수신된 다중 1500 바이트 패킷을 배치(batch)할 수도 있다는 것을 인식할 수 있다. 벌크 전달에서의 1500 바이트 패킷의 수는 호스트 시스템과 오프로드 네트워크 어댑터 사이의 접속의 특성의 함수일 수 있다. 예로서, PCI-Express와 같은 더 신규한 기술은 더 이른 PCI 2.1 버스가 상호접속하는 더 큰 데이터 블록, 즉, 64K를 더욱 효율적으로 이동시킬 수 있다.

[0149] 전술한 바와 같이, 데이터가 전송을 위해 애플리케이션 버퍼에 위치될 때, 버퍼 전송 요청 디스크립터가 입력 디스크립터 테이블에 포스트될 수도 있다. 이러한 버퍼 전송 요청 디스크립터는 데이터 전송이 추진될지 여부를 나타내는 가능한 한 빠른(as soon as possible; ASAP) 비트를 나타낼 수도 있다. ASAP 비트의 전송은 또한 얼마나 많은 DMA 동작이 지연될지를 결정하는데 있어서 오프로드 네트워크 어댑터에 의해 이용되는 기준일 수도 있다. 물론, 가능할 때 마다, 오프로드 네트워크 어댑터는 이러한 ASAP 비트의 세팅을 통해 데이터의 추진된 송신에 대한 호스트 시스템의 요청을 신용하는 시도를 해야 한다.

[0150] DMA 동작은 프로세서 사이클, 요청된 메모리 자원 등과 관련하여, 고정된 셋업 비용 뿐만 아니라 바이트당 전달 비용을 갖는 경향이 있다. I/O 버스를 더 양하게 사용하고 바이트당 비용에 대한 셋업 비용을 감소시키기 위해, 오프로드 네트워크 어댑터는 DMA 전달에 대한 2개의 요청이 물리적 메모리의 인접 영역이다는 것을 인식함으로써 DMA 전달을 종합할 수도 있다. 호스트 시스템은 예를 들어, 접속 마다 대형 애플리케이션 버퍼를 할당하고, 애플리케이션 버퍼의 서브세트에 증분적으로 채우며, 메모리의 인접 서브세트에 대한 요청을 그에 따라 생성함으로써 이러한 프로세스 촉진을 시도할 수도 있다. 오프로드 네트워크 어댑터는 이 서브세트를 인접한 것으로 인식할 수도 있고 DMA 전달을 종합할 수도 있다.

[0151] 예로서, 디스크립터 큐는 DMA 전달에 대한 어드레스 및 길이의 상세한 정보를 포함한다. DMA 동작을 수행하기 이전에 인접 디스크립터의 검사는 후속하는 DMA 요청이 현재 요청의 연속을 단순화시키는지, 즉, 메모리의 인접 부분으로 향하는지를 나타낼 수도 있다. 이러한 경우에서, 모든 DMA 전달은 수행될 필요가 있는 모든 DMA 동작을 참조하는 단일, 조합된 요청으로 충족될 수 있다. 이것은 이들 DMA 전달의 벌크 통지를 제공함으로써 호스트 시스템과 오프로드 네트워크 어댑터 사이에서 DMA 전달 요청을 처리하는 오버헤드를 감시시킨다.

[0152] 본 발명은 DMA 데이터 전송의 충분한 수가 제공되기 이전에 DMA 데이터 전송을 "저장(store up)"할 수도 있다. "충분함"을 결정하는 기준은 상술한 바와 같이 변화할 수도 있다. DMA 데이터 전송의 충분한 수가 실행을 위해 준비되면, 본 발명은 이들 DMA 데이터 전송이 발생할 순서를 결정하는 우선순위 메커니즘을 이용한다. 따라서, 본 발명의 하나의 예시적인 실시예에서, DMA 동작은 우선순위 메커니즘에 기초하여 오프로드 네트워크 어댑터에 의해 재정렬되어, 선호도가 소망하는 접속 및 높은 우선순위 접속에 제공될 수도 있다.

[0153] 도 15는 본 발명의 하나의 예시적인 실시예에 따른 예시적인 DMA 전송 순서 판정 프로세스를 도시한다. 도 15에 도시되어 있는 바와 같이, 3개의 접속, 접속 A, B 및 C가 수립되어 있다. 이들 접속에는 A, B 및 C의 의미적 우선순위 순서가 제공되었고, A가 가장 높고 바람직한 접속이다. 이러한 우선순위 순서는 예를 들어, 사용자 또는 호스트 시스템에 의해 애플리케이션 또는 애플리케이션 접속에 할당된 우선순위에 기초하여 결정될 수도 있다. 상기 언급한 바와 같이, 오프로드 네트워크 어댑터는 수립된 접속에 관한 정보를 저장할 수도 있다. 이러한 우선순위 정보는 오프로드 네트워크 어댑터에 접속 정보의 일부로서 저장될 수도 있고 호스트 시스템상에서 접속 정보의 나머지에 따라 복제될 수도 있다. 이러한 방식에서, 우선순위 정보는 DMA 동작의 순서를 결정하는데 이용하기 위해 오프로드 네트워크 어댑터 및 호스트 시스템 모두에 이용가능하게 이루어진다.

[0154] 표시된 시간에서, 모든 접속은 접속 A, B 및 C를 통해 전송하기 위해 오프로드 네트워크 어댑터(1520)상에서 충

분한 데이터를 갖는다. 이루어질 필요가 있는 결정은 데이터가 송신을 위해 애플리케이션 버퍼(1530, 1540 및 1550)로부터 오프로드 네트워크 어댑터 버퍼(1560, 1570 및 1580)로 DMA되어야 하는 순서이다.

[0155] 본 발명에 의하면, 데이터의 벌크 전달이 이용가능한 애플리케이션 버퍼(1530-1550)가 데이터를 전송할 전송 동작 및 어드레스를 설명하는 입력 디스크립터 테이블(1590)에 디스크립터의 그룹을 저장함으로써 용이해진다. 오프로드 네트워크 어댑터는 지정된 접속의 우선순위에 기초하여 입력 디스크립터 테이블(1590)에서 디스크립터의 리스트를 재정렬한다.

[0156] 디스크립터 리스트의 재정렬은, 하나의 예시적인 실시예에서, 현재의 데이터 소망 접속에 기초하여 초기 수행된다. 즉, 접속이 소망되는 데이터인 경우에, 즉, 데이터가 소정의 시간 주기 동안 접속을 통해 송신되지 않은 경우에, 이러한 접속을 통한 송신을 위한 데이터와 관련된 디스크립터가 디스크립터 리스트에서 먼저 순서화된다. 그 후, 디스크립터는 접속과 관련된 우선순위에 기초하여 재정렬된다.

[0157] 따라서, 도시된 예에 의하면, 입력 디스크립터 테이블 엔트리(1590), 즉, 접속 A, B 및 C에 대한 버퍼 송신 요청 디스크립터는 오프로드 네트워크 어댑터(1520)에 의해 판독 및 재정렬되어, 디스크립터의 재정열된 리스트가 다음 순서, 즉, A1, A2, A3, B1, B2, B3, C1, C2, C3의 순서를 갖게 할 것이다. 그 후, 그 데이터는 이러한 순서로 애플리케이션 버퍼(1530-1550)로부터 판독되고, 우선순위가 접속 A에 주어지도록 오프로드 네트워크 어댑터 버퍼(1560-1580)에 저장될 것이다.

[0158] 따라서, 본 발명은 애플리케이션 버퍼, 버퍼 송신 요청 디스크립터, 입력 디스크립터 테이블, 및 호스트 시스템과 오프로드 네트워크 어댑터 간의 DMA 동작을 이용하여 벌크 전송을 위한 메커니즘을 더 제공한다. 이러한 방식으로, DMA 동작은, 호스트 시스템 상에서 구동하는 애플리케이션의 단편적인 인터럽트 보다는 벌크로 수행될 수 있도록 지연될 수도 있다.

[0159] 도 16은 본 발명의 일 예시적인 실시예의 양태에 따른 호스트 시스템과 오프로드 네트워크 어댑터를 사용하여 데이터를 전송할 경우의 예시적인 동작을 나타내는 플로우차트이다. 도 16에 도시된 바와 같이, 동작은 애플리케이션에 의해 오퍼레이팅 시스템으로 전송되는 데이터를 송신하기 위한 요청으로 시작한다(단계 1610). 그 후, 데이터는 애플리케이션 버퍼로부터 고정된 커널 버퍼로 카피된다(단계 1620). 그 후, 버퍼 송신 디스크립터가 입력 디스크립터 테이블에 포스팅된다(단계 1630).

[0160] 그 후, 오프로드 네트워크 어댑터는, DMA 동작을 통해, 입력 디스크립터 테이블의 다음 엔트리를 판독한다(단계 1640). 본 설명을 위하여, 다음 엔트리는 버퍼 송신 디스크립터인 것으로 가정한다. 입력 디스크립터 테이블은 벌크 전송 리스트에 저장되며(단계 1650), 지연 기준이 충족되었는지에 대해 판정한다(단계 1660). 그렇지 않으면, 동작은 단계 1640으로 복귀하여 입력 디스크립터 테이블의 다음 엔트리를 판독한다. 하지만, 만약 지연 기준이 충족되었으면, 벌크 전송 리스트는, 임의의 접속이 중단되었는지에 대한 판정 및 접속 우선순위에 기초하여 재구성된다(단계 1670).

[0161] 전술된 바와 같이, 이러한 판정의 일부로서, 버퍼 송신 디스크립터는 ASAP 비트가 설정되었음을 나타내는지를 판정할 수도 있다. 그렇다면, 지연 기준이 충족되었는지가 판정되고, 가능하면, 데이터의 송신이 즉시 수행된다.

[0162] 이후, 데이터는 DMA 동작을 통해 고정된 커널 버퍼로부터 판독되며, 벌크 전송 리스트의 재구성으로부터 결정된 순서로 오프로드 네트워크 어댑터에 의해 송신된다(단계 1680). 그 후, 버퍼 가용 응답 디스크립터는, 오프로드 네트워크 어댑터에 의한 데이터의 전송을 확인응답하기 위해 호스트 시스템에 의해 판독되는 출력 디스크립터 테이블에 포스팅될 수도 있다(단계 1690). 그 후, 동작은 종료한다.

[0163] 도 17은 본 발명의 일 예시적인 실시예의 양태에 따른 호스트 시스템과 오프로드 네트워크 어댑터 간의 데이터의 제로 카피 전송을 수행하는 경우의 예시적인 동작을 나타낸 플로우차트이다. 도 17에 도시된 바와 같이, 동작은 수립된 접속을 통해 오프로드 네트워크 어댑터에서 데이터를 수신함으로써 시작한다(단계 1710). 그 후, 오프로드 네트워크 어댑터는 버퍼 수신 응답 디스크립터를 출력 디스크립터 테이블에 포스팅한다(단계 1720). 호스트 시스템은 출력 디스크립터 테이블의 다음 엔트리를 판독한다(단계 1730). 본 설명을 위하여, 출력 디스크립터 테이블의 다음 엔트리는 버퍼 수신 응답 디스크립터인 것으로 가정한다. 출력 디스크립터 테이블 엔트리는 벌크 전송 리스트에 저장될 수도 있다(단계 1740).

[0164] 지연 기준이 충족되었는지에 대해 판정한다(단계 1750). 그렇지 않으면, 동작은 단계 1730으로 복귀한다. 만약 지연 기준이 충족되었으면, 벌크 전송 리스트는, 임의의 접속이 중단되었는지 여부 및 접속 우선순위에 기초하여 재구성된다(단계 1760). 그 후, 데이터는 DMA 동작을 이용하여, 벌크 전송 리스트의 재정렬로부터 결정된

순서로, 데이터가 존재하는 각각의 접속체와 관련된 애플리케이션 버퍼에 직접 전송된다(단계 1770). 그 후, 호스트 시스템은, 완료된 각각의 DMA 동작에 대한 입력 디스크립터 테이블에 버퍼 가용 응답 디스크립터를 포스팅할 수도 있다(단계 1780). 그 후, 동작은 종료한다.

[0165] 데이터가 DMA 동작을 이용하여 전송된 애플리케이션 버퍼는 하나 이상의 공유된 애플리케이션 버퍼를 포함할 수도 있음을 이해해야 한다. 따라서, 하나 이상의 공유된 애플리케이션 버퍼를 공유하는 다양한 접속을 위해 수신된 데이터는 공유된 애플리케이션 버퍼에 DMA될 수도 있으며, 그 애플리케이션은 공유된 애플리케이션 버퍼로부터 데이터를 검색할 수도 있다. 이것은 또한 도 16에서 설명된 데이터 송신 동작에 대해 참이며, 즉, 데이터가 전송된 애플리케이션 버퍼가 공유된 애플리케이션 버퍼일 수도 있다.

[0166] 따라서, 본 발명은, 데이터의 벌크 전송을 달성하도록 호스트 시스템과 오프로드 네트워크 어댑터 간의 통신을 지연시키는 애플리케이션 버퍼를 공유하는 메커니즘, 및 호스트 시스템과 오프로드 네트워크 어댑터 간의 데이터의 제로 카피 전송을 위한 메커니즘을 제공한다. 또한, 본 발명은, 데이터가 이미 송신된 동일한 애플리케이션 버퍼에 그 데이터가 전송될 수도 있도록 부분 버퍼 데이터 전송을 위한 메커니즘을 제공한다.

[0167] **수신 데이터의 핸들링**

[0168] 접속 수립 및 메모리 관리에 더하여, 본 발명은 오프로드 네트워크 어댑터를 활용하는 데이터 처리 시스템에 있어서 수신 데이터의 핸들링을 개선시킨다. 상술된 바와 같이, 본 발명의 오프로드 네트워크 어댑터는, 그 오프로드 네트워크 어댑터로 하여금 호스트 시스템으로의 데이터 수신의 통지를 상이한 방식으로 지연시키게 하는 로직을 포함한다. 호스트 시스템으로의 데이터 패킷 수신의 통지를 지연시키는 이점은, 예를 들어, 단일 통지에 있어서 제1통지 직후에 도달할 수 있는 수개의 데이터 패킷의 집합에 대한 잠재성이다. 연속적인 데이터 패킷 도달을 갖는 스트림이 주어지면, 통지 지연에 대한 값이 설정될 수도 있으며, 이 값은 통신 소켓당 호스트 시스템에 대해 구성가능할 수도 있다.

[0169] 지연값은 정적 또는 동적으로 설정될 수도 있다. 예를 들어, 지연값은 소켓 접속체에 수신된 데이터의 이력 관측을 통하여 일 시간주기 동안 수신된 데이터의 양 또는 레이트에 기초하여 설정될 수도 있다. 일예는, 특정 수신 접속이 10밀리초 동안 10개의 데이터 패킷을 버스트(burst)로 동작한 후 10초 동안 완전한 경우, 호스트 시스템에 대한 전체 통지를 감소시키기 위해 10밀리초 동안의 패킷 도달의 모든 통지를 지연시키는 것이 현명할 수도 있다.

[0170] 다른 방법으로, 호스트 시스템이 애플리케이션 버퍼를 접속체에 포스팅하는 레이트가 모니터링되어 이러한 지연값을 동적으로 설정하는 기초로서 이용될 수도 있다. 만약 호스트가 특정 레이트, 예를 들어, 매 10밀리초 당 1회의 레이트로 애플리케이션 버퍼를 포스팅하면, 버퍼가 오프로드 네트워크 어댑터로부터 호스트 시스템으로의 데이터의 제로 카피 전송용으로 이용가능함을 보장하기 위하여, 데이터 도달 통지를 10밀리초 만큼 지연시키는 것이 이해될 것이다.

[0171] 또 다른 대안으로서, 데이터 도달 통지가 호스트 시스템을 전송된 이후 호스트 시스템이 접속을 위해 신규한 버퍼를 포스팅하는 레이트가 모니터링되어 지연값을 설정하는 기초로서 이용될 수도 있다. 이것은, 호스트 시스템이 특정 접속체로부터 데이터를 소비하는 레이트를 나타낸다. 예를 들어, 호스트 시스템이 버퍼 내의 데이터를 소비하고 사용을 위해 오프로드 네트워크 어댑터에 버퍼를 포스팅하는데 10밀리초가 소요될 수도 있다. 따라서, 10밀리초의 통지 지연은 오프로드 네트워크 어댑터와 호스트 시스템 간의 데이터의 제로 카피 전송을 위해 데이터 버퍼의 대체를 보장하는데 현명할 수도 있다.

[0172] 또 다른 실시예에서, 데이터의 양이 버퍼 수신 포스팅 지연에 대한 시간 메트릭 대신에 사용될 수도 있다. 이 경우, 지연값은 데이터 패킷의 수신을 호스트 시스템에게 통지하기 전에 수신될 일정량의 데이터를 대기하도록 설정된다. 데이터의 양은 접속의 셋업에 있어서의 옵션으로서 호스트 시스템에 의해 정적으로 설정되거나, 이력 관측에 기초하여 오프로드 네트워크 어댑터에 의해 동적으로 설정될 수 있다. 지연값의 설정을 결정하는 다른 방법 및 메커니즘이 본 발명의 취지 및 범위를 벗어나지 않고 사용될 수도 있다.

[0173] 또 다른 실시예가 지연의 양을 결정하기 위해 선택된다는 것과 무관하게, 최대 지연값은 제1데이터 도달과 호스트 시스템으로의 데이터 도달의 최종 통지 간의 최대 지연을 식별하기 위해 오프로드 네트워크 어댑터에 유지될 수도 있다. 이것은, 데이터의 도달과 호스트 시스템으로의 데이터의 도달의 통지 사이의 과도한 지연이 없음을 보장한다. 지연값, 최대 지연값, 및 그 지연값을 결정하기 위해 필요한 다른 정보는, 지연값을 설정함에 있어서 사용하기 위해 그리고 오프로드 네트워크 어댑터로부터 호스트 시스템으로의 통지를 얼마나 지연시킬 것인지

를 결정하기 위해 오프로드 네트워크 어댑터 상의 메모리에 저장될 수도 있다.

[0174] 본 발명의 동작의 이전 설명에서, 상술된 대안 중 하나 이상에 따라 결정된 지연값과 최대 지연값은 지연 기준을 충족하는지 여부를 결정함에 있어서 활용된다. 예를 들어, 지연 기준이 충족되는지를 판정할 때, 제1데이터 패킷의 수신으로부터 타이밍 지연의 비교가 지연값에 대해 비교될 수도 있다. 일단 타이밍 지연이 지연값을 충족하거나 초과하면, 데이터 패킷의 벌크 전송은 오프로드 네트워크 어댑터로부터 호스트 시스템으로 행해지거나 그 역이 성립할 수도 있다. 유사하게, 만약 지연값이 데이터의 양의 관점에서 제공되면, 수신된 제1데이터 패킷으로부터 접속체를 통해 수신된 데이터의 양은 지연값에 대해 비교되어, 데이터의 양이 지연값에서 설정된 데이터의 양을 충족하거나 초과하는지 판정할 수도 있다. 그렇다면, 오프로드 네트워크 어댑터로부터 호스트 시스템으로 또는 그 역으로의 데이터의 벌크 전송은 호스트 시스템 또는 오프로드 네트워크 어댑터로 송신된 벌크 데이터 수신 통지, 예를 들어, 입력 또는 출력 디스크립터 테이블에 포스팅된 버퍼 수신 응답 디스크립터를 통해 개시될 수도 있다.

[0175] 현재의 비-인텔리전트 호스트-네트워크 어댑터 시스템에서, 모든 데이터는 호스트의 오퍼레이팅 시스템 레이어 내의 비-접속 특정 애플리케이션 버퍼의 풀을 통과한다. 접속 특정 애플리케이션 버퍼로의 데이터의 제로 카피 전송이 본 발명의 메커니즘을 이용하여 가능하다고 주어지면, 본 발명은 어떠한 접속 특정 애플리케이션 버퍼 또는 공유된 애플리케이션 버퍼도 데이터를 수신하기 위해 그 애플리케이션에 의해 현재 포스팅되어 있지 않을 경우에 대한 판정 프로세스를 제공한다. 디폴트로서, 만약 접속 특정 애플리케이션 버퍼 또는 공유된 애플리케이션 버퍼가 접속체에 할당되어 있지 않으면, 본 발명의 판정 프로세스는 비-접속 특정 애플리케이션 버퍼의 풀로부터의 버퍼를 이용하여 오프로드 네트워크 어댑터로부터 애플리케이션으로 데이터를 전송한다.

[0176] 하지만, 본 발명에 있어서, 구성 파라미터가 제공된 호스트 시스템은, 어떠한 접속 특정 버퍼도 존재하지 않으면, 접속 특정 애플리케이션 버퍼가 비-접속 특정 애플리케이션 버퍼를 사용하는 것 대신에 할당될 때까지 오프로드 네트워크 어댑터가 대기할 수 있도록 제공될 수도 있다. 이러한 파라미터는 오프로드 네트워크 어댑터의 메모리에 저장될 수도 있으며, 시스템의 디폴트 행위를 무효로 하여, 데이터가 호스트 시스템에 DMA되기 전에 접속 특정 애플리케이션 버퍼가 접속을 위해 할당될 때까지 오프로드 네트워크 어댑터가 대기하도록 이용될 수도 있다. 이러한 대기는, 접속 특정 애플리케이션 버퍼가 할당되거나 최대 대기 시간이 충족되거나 초과될 때까지 수행될 수도 있다. 최대 대기 시간이 충족되거나 초과되면, 접속을 위해 오프로드 네트워크 어댑터에 저장된 데이터는 비-접속 특정 애플리케이션 버퍼에 DMA될 수도 있다.

[0177] 비-접속 특정 애플리케이션 버퍼의 디폴트 행위를 무효로 하기 위해 미리 정의된 호스트 제공 구성 파라미터를 설정하는 것 대신, 오프로드 네트워크 어댑터 자체는 호스트 시스템 공급 접속 특정 애플리케이션 버퍼의 이력 데이터에 기초하여, 접속 특정 애플리케이션 버퍼를 대기해야 하는지, 얼마나 오래 접속 특정 버퍼를 대기해야 하는지, 또는 접속 특정 애플리케이션 버퍼를 대기하지 말아야 하는지를 결정하도록 하는 로직을 제공받을 수도 있다.

[0178] 예를 들어, 호스트 시스템은 이력 데이터에서 관측된 시간 프레임 내의 시간의 100%를 제로 카피 동작을 위해 접속 특정 애플리케이션 버퍼에 제공될 수도 있다. 즉, 데이터 전송의 마지막 x 번호에서, 접속 특정 애플리케이션 버퍼는 이들 데이터 전송을 용이하게 하는 시간의 100%가 활용되었다. 결과적으로, 접속 특정 애플리케이션 버퍼를 대기하는 상기 동작이 수행될 수도 있다.

[0179] 하지만, 데이터 전송이 접속 특정 애플리케이션 버퍼를 이용하여 시간의 100%가 수행되지 않았음을 이력 데이터가 나타내면, 접속 특정 애플리케이션 버퍼가 활용된 시간의 퍼센티지가 소정의 임계량보다 작은지를 판정한다. 그렇다면, 오프로드 네트워크 어댑터는 할당될 접속 특정 애플리케이션 버퍼를 대기하지 않으며, 비-접속 특정 애플리케이션 버퍼를 이용할 수도 있다. 다른 방법으로, 오프로드 네트워크 어댑터가 접속 특정 애플리케이션 버퍼를 대기하는 시간의 양은, 퍼센티지 값이 소정의 임계값 이하가 되는지에 따라 감소될 수도 있다. 데이터 전송이 계속됨에 따라, 오프로드 네트워크 어댑터 내에 유지되는 이력 데이터는 각각의 데이터 전송과 함께 이동하는 시간 윈도우일 수도 있다. 따라서, 더 많은 데이터 전송이 접속 특정 애플리케이션 버퍼를 이용하여 수행됨에 따라, 퍼센티지 값은 소정의 임계값 이상으로 증가할 수도 있고, 시스템은 할당될 접속 특정 애플리케이션 버퍼를 대기하거나 접속 특정 애플리케이션 버퍼에 대한 원래의 대기 시간으로 복귀할 수도 있다.

[0180] 본 발명의 예시적인 실시예의 다른 양태에서, 만약 비-접속 특정 애플리케이션 버퍼가 오프로드 네트워크 어댑터로부터 호스트 시스템으로 데이터를 DMA함에 있어서 사용하기 위해 풀로부터 선택되어야 하면, 본 발명은, 데이터를 전송하는 비-접속 특정 애플리케이션 버퍼를 선택하기 위해 오프로드 네트워크 어댑터 내에 로직을 제공한다. 이 로직은 버퍼 풀 내의 다양한 비-접속 특정 애플리케이션 버퍼의 각 특징을 주시하고, 오프로드 네트

워크 어댑터로부터 호스트 시스템으로 전송될 데이터에 대한 최상의 매치(match)를 제공하는 것을 선택한다. 버퍼에 대한 정보는 호스트 시스템 및/또는 오프로드 네트워크 어댑터에 유지된 접속 정보로부터 획득될 수도 있다.

[0181] 예를 들어, 오프로드 네트워크 어댑터가 버퍼 풀로부터 비-접속 특정 애플리케이션 버퍼를 이용해야 한다고 판정할 경우, 오프로드 네트워크 어댑터는 호스트 시스템으로부터의 풀 내의 버퍼에 대한 특징 정보를 판독한다. 이러한 특징 정보는, 예를 들어, 버퍼의 사이즈, 버퍼의 속도, 호스트 프로세서 구조에서의 버퍼의 배치 등일 수도 있다. 이들 특징에 기초하여, 오프로드 네트워크 어댑터는, 그 오프로드 네트워크 어댑터로부터 호스트 시스템으로 데이터를 전송함에 있어서 사용하기 위한 최상의 후보인 풀로부터 버퍼를 선택한다.

[0182] 일예로서, 선택 프로세스가 키잉(key)된 특징으로서 버퍼 사이즈를 취하면, 상이한 사이즈를 갖는 버퍼 풀에서 이용가능한 수개의 비-접속 특정 애플리케이션 버퍼가 존재할 수도 있다. 일정량의 데이터가 호스트 시스템에 전송된다고 주어지면, 오프로드 네트워크 어댑터는, 복수의 버퍼를 통해 데이터를 확산하기 보다는 데이터를 전부 포함하기에 충분한 사이즈를 갖는 버퍼 풀로부터 비-접속 특정 애플리케이션 버퍼를 선택한다. 상술한 다른 특징은, 특정 데이터 전송용으로 사용하기 위한 최상의 버퍼를 판정하기 위해 유사한 방식으로 사용될 수도 있다.

[0183] 도 18은 본 발명의 일 예시적인 실시예의 양태에 따라 데이터를 전송하기 위해 애플리케이션 버퍼를 판정하는 예시적인 동작을 나타낸 플로우차트이다. 도 18에 도시된 바와 같이, 동작은 호스트 시스템으로의 전송을 위해 오프로드 네트워크 어댑터에서 데이터를 수신함으로써 시작한다(단계 1810). 그 후, 수신 데이터가 지향하는 접속(들)에 대해 접속 특정 애플리케이션 버퍼가 할당되는지를 판정한다(단계 1820). 그렇다면, 그 후, 데이터는 DMA 동작을 이용하여 할당된 접속 특정 애플리케이션 버퍼(들)에게 송신되고(단계 1830), 동작은 종료한다.

[0184] 만약 데이터가 지향하는 접속에 대해 접속 특정 애플리케이션 버퍼가 할당되지 않으면(단계 1820), 대기 파라미터가 설정되었는지에 대해 판정한다(단계 1840). 만약 그렇다면, 대기 임계값이 초과하였는지에 대해 판정한다(단계 1850). 만약 그렇지 않다면, 동작은 단계 1820으로 루프-백하고, 대기 임계값이 초과할 때까지 또는 접속 특정 애플리케이션 버퍼가 할당될 때까지 루프를 계속한다.

[0185] 만약 대기 임계값이 초과되었거나(단계 1850) 대기 파라미터가 설정되지 않았다면(단계 1840), 버퍼 풀에서 비-접속 특정 애플리케이션 버퍼에 대한 특징 정보가 호스트 시스템으로부터 검색된다(단계 1860). 그 후, 비-접속 특정 애플리케이션 버퍼가 검색된 특징 정보에 기초하여 이 풀로부터 선택된다(단계 1870). 그 후, 데이터는 DMA 를 이용하여 선택된 비-접속 특정 애플리케이션 버퍼에 직접 전송되고(단계 1880), 동작을 종료한다.

[0186] 추가적인 설계가 DMA 배치에 대한 옵션으로서 L3 캐시 구조에 직접 데이터 배치를 허용할 수도 있다. 즉, 데이터는 호스트 시스템에 의해 제공된 가상 어드레스 및 캐시 주입 메커니즘을 사용하여 L3 캐시로 푸쉬될 수도 있다. 애플리케이션 버퍼 내의 데이터의 DMA 배치에 추가하여 또는 그 대신에, 신속하게 처리될 필요가 있는 데이터는 즉각적인 처리를 위해 L3 캐시에 제공될 수도 있다.

[0187] 특정 데이터가 L3 캐시에 주입되어야 되는지 여부를 판정할 수 있는 다양한 방법이 존재한다. 예를 들어, 데이터가 L3 캐시에 주입되어야 하는지의 판정은 접속당 호스트 시스템에 의해 수립된 명시적인 구성 정보에 기초할 수도 있다. 다른 방법으로, 이러한 판정은 최근 얼마나 많은 데이터가 L3 캐시에 주입되었는지의 모니터링에 기초하여, 캐시 오버플로우 상황이 가능한지를 판정할 수도 있다. L3 캐시로의 데이터의 주입이 임의의 이점을 획득하는지 또는 캐시 오버플로우를 발생하는지 여부를 방해하는 다른 메커니즘이 또한 사용될 수도 있다.

[0188] 전술한 바와 같이, 이러한 타입의 메모리 관리 메커니즘은, 웹 요청/응답 트래픽과 같은 즉각적인 CPU 주의를 요구하는 일정한 트래픽에 대해 바람직할 수도 있다. 파일 시스템에 대해 미리 패치되는 ISCSI 데이터와 같은 다른 타입의 데이터는, 일정 시간 동안에 요구되지 않을 수도 있기 때문에 DMA로서 더 양호할 수도 있다. 이러한 파라미터는 네트워크 판독 또는 구성 파라미터에 대한 요청의 원천에 기초하여 식별될 수 있다.

[0189] 비록 상술된 대체 실시예가 L3 캐시로의 데이터의 주입을 참조하여 행해졌지만, 이 실시예는 L3 캐시와 함께 사용하는 것에 제한되지 않음을 이해해야 한다. L3는 다수의 공지된 구조에 있어서 물리 어드레스 매핑을 갖기 때문에 예시적인 실시예에서 선호된다. 이것은 입력/출력 디바이스로부터 직접 데이터를 이동시키는 설계에 있어서 복잡도를 감소시킨다. 하지만, InfiniBand와 같은 시스템 영역 네트워크의 RDMA 네트워크 어댑터와 같은 새로운 네트워크 어댑터에 있어서, 사용자 어드레스는, 메모리 계층에서 가상 어드레스가능한 L3 캐시뿐 아니라 임의의 다른 캐시로의 데이터 주입을 허용하도록 제공될 수도 있다. 또한, 어드레스 변환은 실제로부터 가상으로 행해질 수 있으며, 이에 의해, 필요한 어드레스를 임의의 타입의 캐시에 제공할 수 있다. 따라서, 예시적인

대체 실시예의 메커니즘은 시스템의 특정 구조에 의존하는 임의의 레벨 캐시에 적용될 수도 있다.

[0190] 본 발명의 또 다른 양태에서, 오프로드 네트워크 어댑터는 데이터 버퍼의 별개지만 순서적인 세그먼트를 재조합하는 로직을 포함할 수도 있다. 오프로드 네트워크 어댑터에 의해 생성됨에 있어서의 디스크립터는, 그 디스크립터를 출력 디스크립터 테이블에 포스팅하기 전에, 이동될 데이터가 연속적인 물리 어드레스 공간인지를 확인하기 위해 검사될 수도 있다. 다중의 디스크립터가 메모리에서 연속적인 물리 어드레스를 식별하도록 생성되면, 복수의 디스크립터를 출력 디스크립터 테이블에 포스팅하는 대신, 전송될 데이터는 오프로드 네트워크 어댑터에서 조합될 수도 있으며, 단일의 조합된 디스크립터는 각각의 데이터 전송을 식별하기 위해 사용될 수도 있다. 예를 들어, TCP/IP 세그먼트는 적절히 사이징된 버퍼(예를 들어, 4K 페이지 정렬된 데이터)로 재조합되고 호스트 시스템에 벌크로 전달될 수도 있다. 이것은 호스트 시스템에 대한 더 용이한 데이터 버퍼 관리를 위해 그리고 더 큰 효율성을 위해 제공된다. 이것은, 이들 다중의 접속에 요구된 버퍼의 양을 잠재적으로 감소시킬 수 있다.

[0191] 본 발명의 예시적인 실시예의 또 다른 양태에서, 오프로드 네트워크 어댑터는 수신 패킷 내에서 데이터를 검사하는 로직을 제공받지만 그 데이터를 소비하지 않는다. 수신 호출은, 호스트 애플리케이션에 수신된 데이터 패킷의 부분, 예를 들어, 헤더의 카피를 제공할 수도 있는 "피크(peek)" 옵션을 특정할 수도 있다. 이것은 호스트 애플리케이션으로 하여금 헤더 데이터를 검사하게 하고, 페이로드가 어떻게 소비되는지 판정하게 할 수도 있다. 일예로서, 애플리케이션은 헤더 식별자에 의해 태그된 상이한 타입의 데이터를 수신할 것을 기대하고 있을 수도 있다. 이것은, 헤더 및 페이로드 데이터가 가변 길이인 경우에 특히 유용하다. 프로그램은 임의의 헤더의 최대 길이에 대해 간단히 "피크"하여 헤더 정보를 검사할 수 있다. 헤더의 피킹은 프로그램으로 하여금 데이터 패킷의 페이로드를 전송하기 위한 애플리케이션 버퍼가 의도한 프로그램 스트림에 기초하는지를 판정하게 할 수도 있다.

[0192] 따라서, "피크" 옵션이 오프로드 네트워크 어댑터에서의 접속을 위해 설정될 경우, 수신 데이터 패킷의 헤더의 카피는, 어떤 타입의 데이터가 수신되는지 그리고 어떤 소켓, 즉, 접속체가 데이터 패킷 페이로드를 송신해야 하는지를 판정할 경우에 호스트 애플리케이션에 제공된다. 예를 들어, 애플리케이션은 비디오 데이터 및 오디오 데이터에 대한 개별 접속체를 가질 수도 있다. 헤더로부터, 애플리케이션은 데이터 패킷의 페이로드에서 데이터의 타입을 판정할 수 있다. 만약 데이터가 비디오 데이터라면, 피크 동작은 호스트 애플리케이션으로 하여금 데이터 패킷 페이로드가 제1접속체와 관련된 애플리케이션 버퍼에 DMA되어야 함을 지정하게 한다. 데이터가 오디오 데이터라면, 피크 동작은 호스트 애플리케이션으로 하여금 데이터 패킷 페이로드가 제2접속체와 관련된 애플리케이션 버퍼에 DMA되어야 함을 지정하게 한다.

[0193] 이러한 피크 동작을 제공하기 위하여, 오프셋을 갖는 데이터를 판독하기 위해 옵션이 제공된다. 이러한 방식으로, 데이터 패킷의 페이로드는, 피크된 헤더로부터 용이하게 분리될 수도 있다. 즉, 호스트 애플리케이션이 헤더의 실제 사이즈를 알기 때문에, 오프셋은, 데이터 패킷을 처리할 때에 헤더를 스키핑(skipping)하는데 사용하기 위해 생성 및 저장될 수도 있다. 이것은, 헤더가 피크 동작에 특정된 바이트 수보다 작을 경우에 가장 유용하다.

[0194] 본 발명이 완전히 기능하는 데이터 처리 시스템의 맥락에서 설명되었지만, 당업자는 본 발명의 프로세스가 명령의 컴퓨터 판독가능 매체의 형태 및 다양한 형태로 분배될 수 있고, 본 발명이 그 분배를 실행하는데 실제로 사용되는 특정 타입의 신호 보유 매체에 관계없이 동일하게 적용됨을 이해할 것이다. 컴퓨터 판독가능 매체의 예는 플로피 디스크, 하드 디스크 드라이브, RAM, CD-ROM, DVD-ROM과 같은 레코드가능형 매체, 및 예를 들어 무선 주파수 및 광파 전송과 같은 전송 형태를 사용하는 디지털 및 아날로그 통신 링크, 유선 또는 무선 통신 링크와 같은 전송형 매체를 포함한다. 컴퓨터 판독가능 매체는, 특정 데이터 처리 시스템에서의 실제 사용을 위해 디코딩되는 코딩 포맷의 형태를 취할 수도 있다.

[0195] 본 발명의 설명은 예시 및 설명의 목적으로 제시되었으며, 배타적으로 의도되거나 개시된 형태로 본 발명에 제한되도록 의도되지 않는다. 다수의 변경예 및 변형예는 당업자에게 자명할 것이다. 실시예는, 본 발명의 원리, 실제 애플리케이션을 가장 잘 설명하기 위하여 선택 및 기술되었으며, 다른 당업자로 하여금 고려되는 특정 사용예에 적합한 다양한 변경예를 갖는 다양한 실시예에 대해 본 발명을 이해할 수 있게 하도록 선택 및 기술되었다.

도면의 간단한 설명

[0011] 본 발명의 특징으로 간주되는 신규한 특징이 첨부된 특허청구범위에 기재되어 있다. 하지만, 본 발명 자체뿐

아니라 바람직한 실시형태, 본 발명의 또 다른 목적 및 이점은, 첨부도면과 관련하여 판독될 경우, 예시적인 실시예에 대한 다음의 상세한 설명을 참조하여 가장 잘 이해될 것이다.

[0012] 도 1은 본 발명의 양태가 구현될 수도 있는 분산 데이터 처리 시스템의 예시적인 다이어그램이다.

[0013] 도 2는 본 발명의 양태가 구현될 수도 있는 서버 컴퓨팅 디바이스의 예시적인 다이어그램이다.

[0014] 도 3은 본 발명의 양태가 구현될 수도 있는 클라이언트 컴퓨팅 디바이스의 예시적인 다이어그램이다.

[0015] 도 4는 본 발명의 일 예시적인 실시예에 따른 네트워크 어댑터의 예시적인 다이어그램이다.

[0016] 도 5는 종래의 네트워크 인터페이스 카드를 활용하는 시스템에 있어서 TCP/IP 처리를 예시하는 다이어그램이다.

[0017] 도 6은 TCP/IP 오프로드 엔진 또는 오프로드 네트워크 어댑터를 활용하는 시스템에 있어서 TCP/IP 처리를 나타내는 다이어그램이다.

[0018] 도 7은 본 발명의 오프로드 네트워크 어댑터 프로그래밍 인터페이스에 대한 본 발명의 일 예시적인 실시예의 양태를 나타내는 예시적인 다이어그램이다.

[0019] 도 8은 오프로드 네트워크 어댑터 및 오프로드 네트워크 어댑터 프로그래밍 인터페이스를 사용한 접속의 수립에 대한 본 발명의 일 예시적인 실시예의 양태를 나타내는 예시적인 다이어그램이다.

[0020] 도 9는 오프로드 네트워크 어댑터를 사용하여 접속을 수립한 경우에 본 발명의 호스트 시스템의 예시적인 동작을 나타낸 플로우차트이다.

[0021] 도 10은 본 발명의 일 예시적인 실시예에 따라 접속을 수립한 경우에 오프로드 네트워크 어댑터의 예시적인 동작을 나타낸 플로우차트이다.

[0022] 도 11은 데이터의 버퍼링된 송수신이 활용되는 본 발명에 따른 메모리 관리 메커니즘을 나타낸 예시적인 다이어그램이다.

[0023] 도 12는 본 발명의 일 예시적인 실시예에 따른 제로-카피 동작을 나타낸 예시적인 다이어그램이다.

[0024] 도 13은 본 발명의 일 예시적인 실시예에 따른 공유 버퍼 배열을 나타낸 예시적인 다이어그램이다.

[0025] 도 14는 본 발명의 일 예시적인 실시예에 따라 부분 송수신 버퍼가 동작하는 방식을 나타낸 것이다.

[0026] 도 15는 본 발명의 일 예시적인 실시예에 따른 예시적인 DMA 전송 순서 판정 프로세스를 나타낸 것이다.

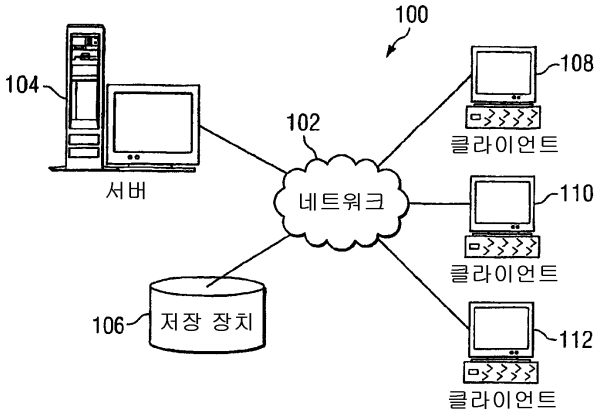
[0027] 도 16은 본 발명의 일 예시적인 실시예의 양태에 따른 호스트 시스템 및 오프로드 네트워크 어댑터를 사용하여 데이터를 송신한 경우에 예시적인 동작을 나타낸 플로우차트이다.

[0028] 도 17은 본 발명의 일 예시적인 실시예의 양태에 따른 호스트 시스템과 오프로드 네트워크 어댑터 간의 데이터의 제로-카피 전송을 수행할 경우에 예시적인 동작을 나타낸 플로우차트이다.

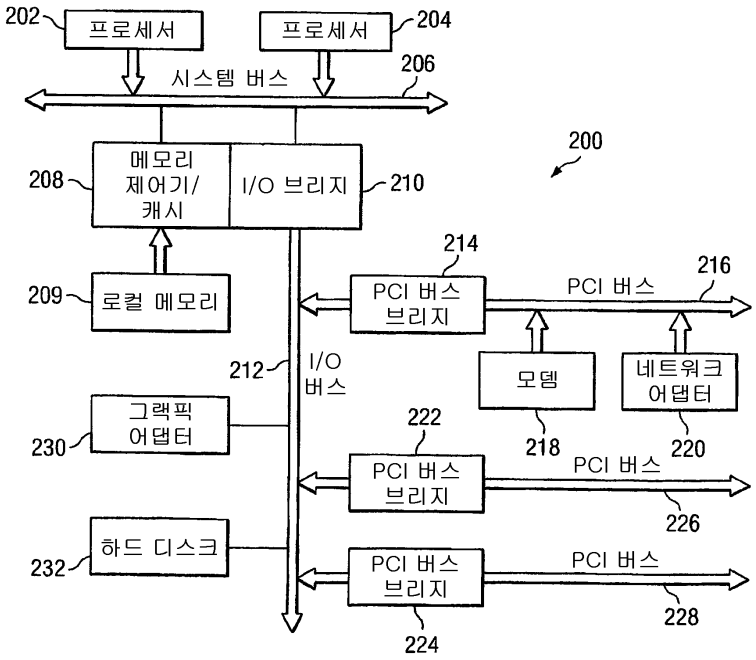
[0029] 도 18은 본 발명의 일 예시적인 실시예의 양태에 따라 데이터를 송신하기 위해 애플리케이션 버퍼를 결정하는 예시적인 동작을 나타낸 플로우차트이다.

도면

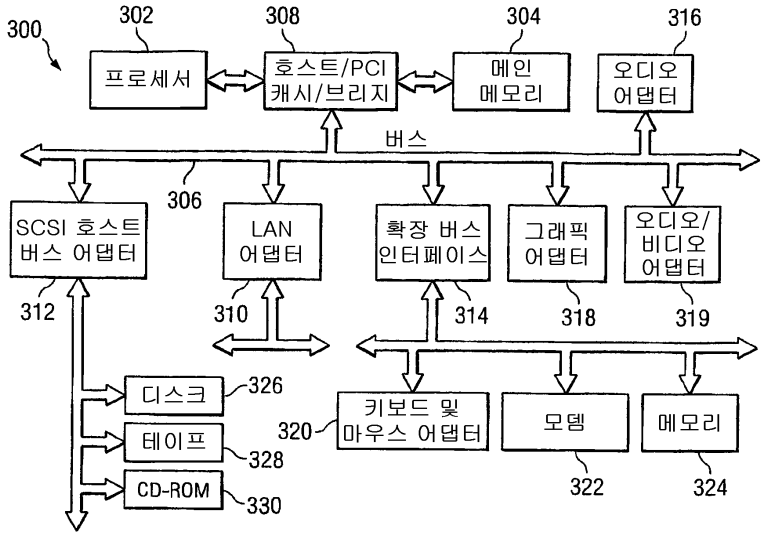
도면1


100
104
서버
102
네트워크
108
클라이언트
110
클라이언트
112
클라이언트
106
저장 장치

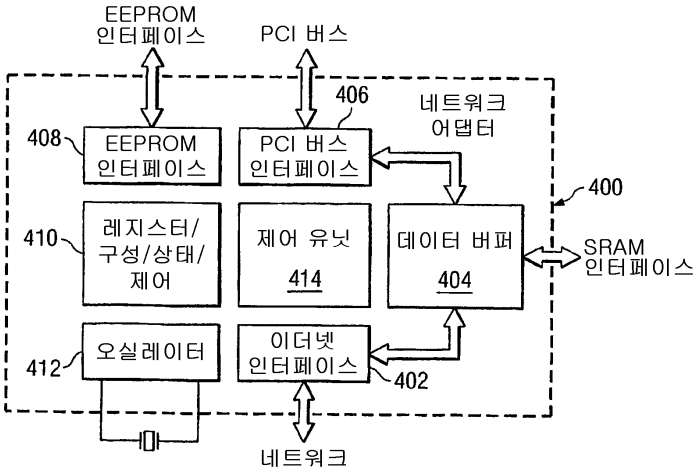
도면2


202
프로세서
204
프로세서
206
시스템 버스
200
208
메모리 제어기/ 캐시
I/O 브리지
210
209
로컬 메모리
214
PCI 버스 브리지
216
PCI 버스
212
I/O 버스
모뎀
218
네트워크 어댑터
220
그래픽 어댑터
230
222
PCI 버스 브리지
PCI 버스
226
하드 디스크
232
PCI 버스 브리지
PCI 버스
228
224

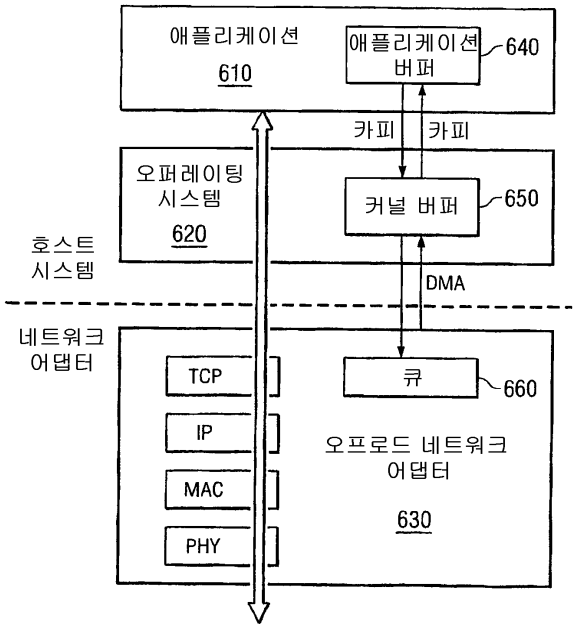
도면3


300
302
308
304
316
프로세서
호스트/PCI 캐시/브리지
메인 메모리
오디오 어댑터
버스
306
SCSI 호스트 버스 어댑터
LAN 어댑터
확장 버스 인터페이스
그래픽 어댑터
오디오/비디오 어댑터
312
310
314
318
319
디스크
326
테이프
328
CD-ROM
330
320
키보드 및 마우스 어댑터
모뎀
메모리
322
324

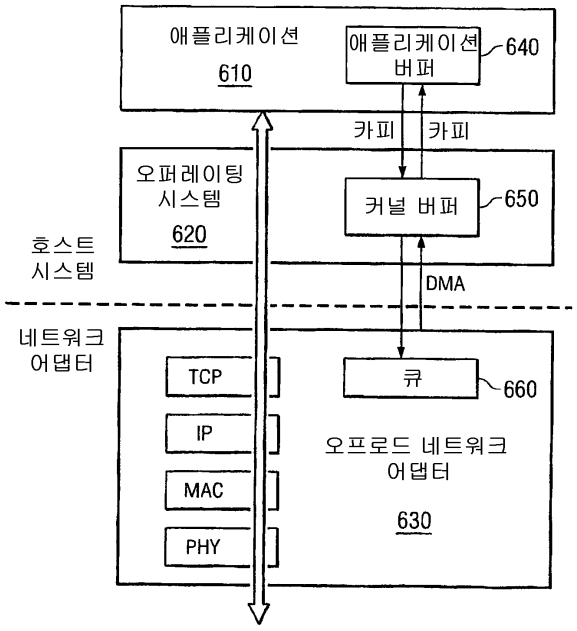
도면4


EEPROM 인터페이스
PCI 버스
406
네트워크 어댑터
408
EEPROM 인터페이스
PCI 버스 인터페이스
400
410
레지스터/구성/상태/제어
제어 유닛
414
데이터 버퍼
404
SRAM 인터페이스
412
오실레이터
이더넷 인터페이스
402
네트워크

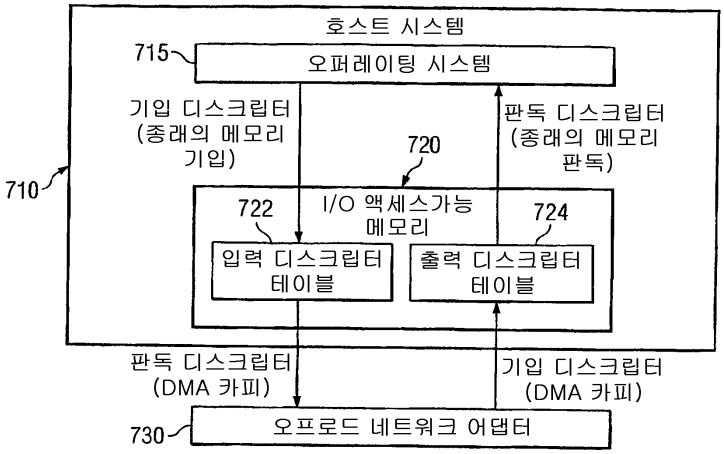
도면5


애플리케이션
610
애플리케이션 버퍼
640
카피
카피
오퍼레이팅 시스템
620
커널 버퍼
650
호스트 시스템
DMA
네트워크 어댑터
TCP
IP
MAC
PHY
큐
660
오프로드 네트워크 어댑터
630

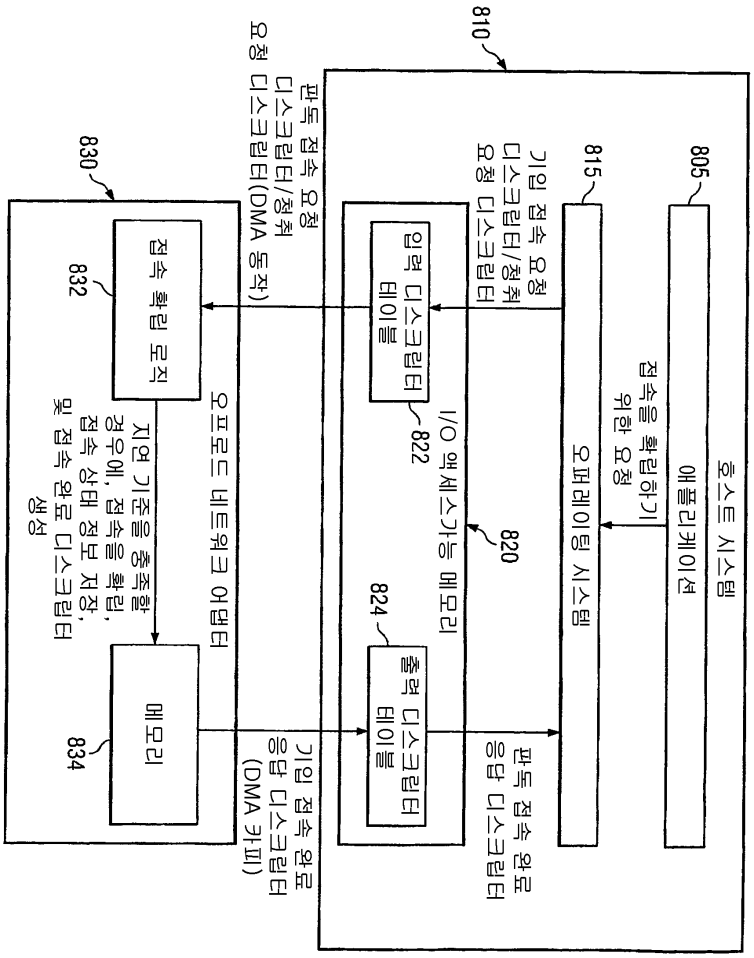
도면6


애플리케이션
610
애플리케이션 버퍼
640
카피
카피
오퍼레이팅 시스템
620
커널 버퍼
650
호스트 시스템
DMA
네트워크 어댑터
TCP
IP
MAC
PHY
큐
660
오프로드 네트워크 어댑터
630

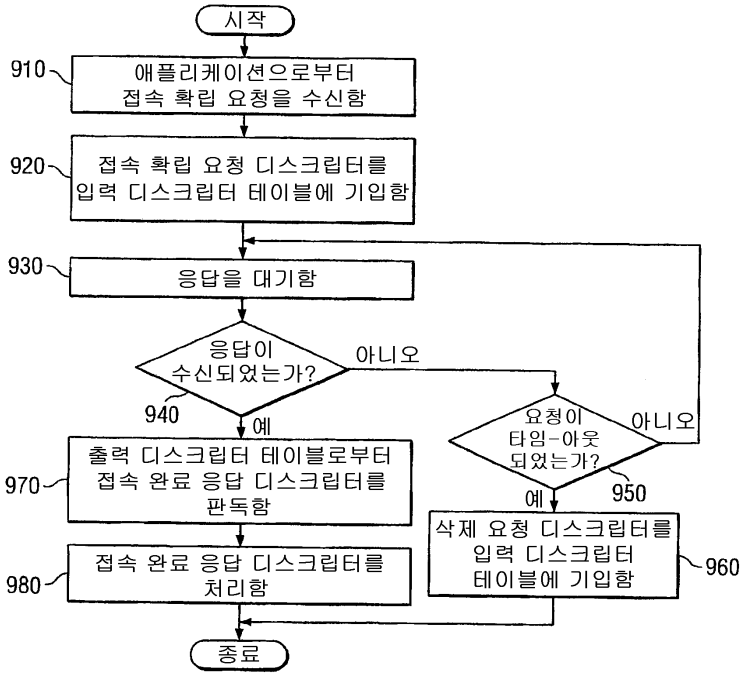
도면7


호스트 시스템
715
오퍼레이팅 시스템
기입 디스크립터
(종래의 메모리
기입)
판독 디스크립터
(종래의 메모리
판독)
720
710
722
I/O 액세스가능
메모리
724
입력 디스크립터
테이블
출력 디스크립터
테이블
판독 디스크립터
(DMA 카피)
기입 디스크립터
(DMA 카피)
730
오프로드 네트워크 어댑터

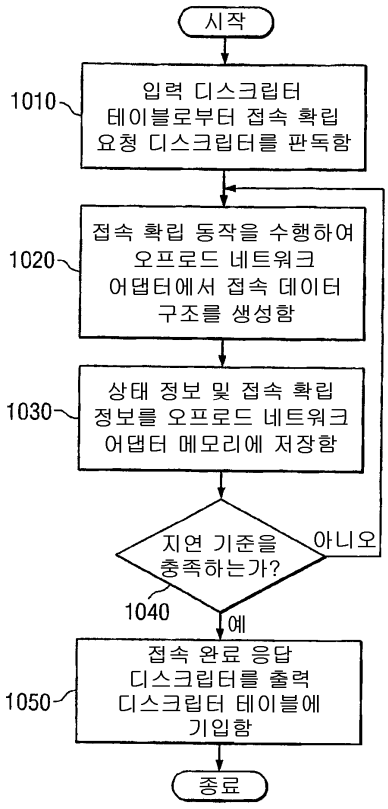
도면8


810
805
815
820
830
832
834
822
824
호스트 시스템
애플리케이션
접속을 확립하기
위한 요청
오퍼레이팅 시스템
기입 접속 요청
디스크립터/청취
요청 디스크립터
판독 접속 요청
디스크립터/청취
요청 디스크립터(DMA 동작)
입력 디스크립터
테이블
I/O 액세스가능 메모리
출력 디스크립터
테이블
판독 접속 완료
응답 디스크립터
기입 접속 완료
응답 디스크립터
(DMA 카피)
오프로드 네트워크 어댑터
접속 확립 로직
지연 기준을 충족할
경우에, 접속을 확립,
접속 상태 정보 저장,
및 접속 완료 디스크립터
생성
메모리

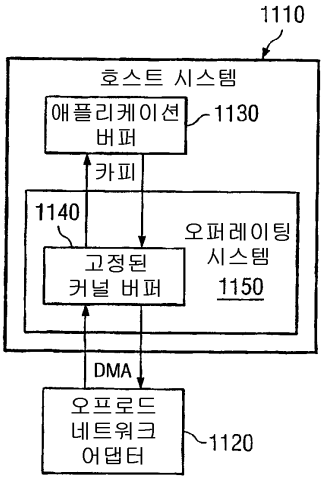
도면9


시작
910
애플리케이션으로부터 접속 확립 요청을 수신함
920
접속 확립 요청 디스크립터를 입력 디스크립터 테이블에 기입함
930
응답을 대기함
응답이 수신되었는가?
아니오
940
예
요청이 타임-아웃 되었는가?
아니오
950
예
970
출력 디스크립터 테이블로부터 접속 완료 응답 디스크립터를 판독함
삭제 요청 디스크립터를 입력 디스크립터 테이블에 기입함
960
980
접속 완료 응답 디스크립터를 처리함
종료

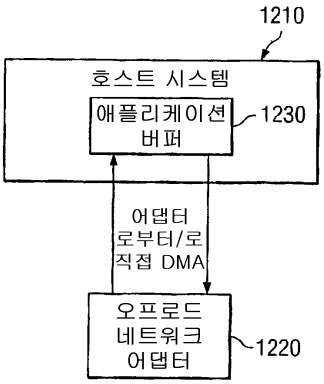
도면10


시작
1010
입력 디스크립터 테이블로부터 접속 확립 요청 디스크립터를 판독함
1020
접속 확립 동작을 수행하여 오프로드 네트워크 어댑터에서 접속 데이터 구조를 생성함
1030
상태 정보 및 접속 확립 정보를 오프로드 네트워크 어댑터 메모리에 저장함
지연 기준을 충족하는가?
아니오
1040
예
1050
접속 완료 응답 디스크립터를 출력 디스크립터 테이블에 기입함
종료

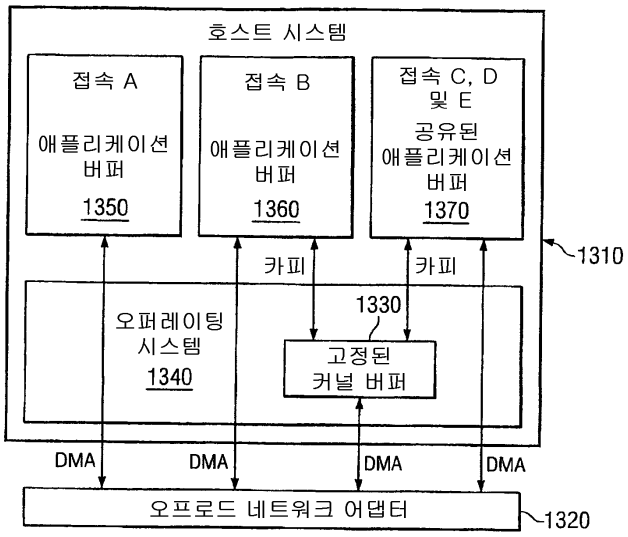
도면11



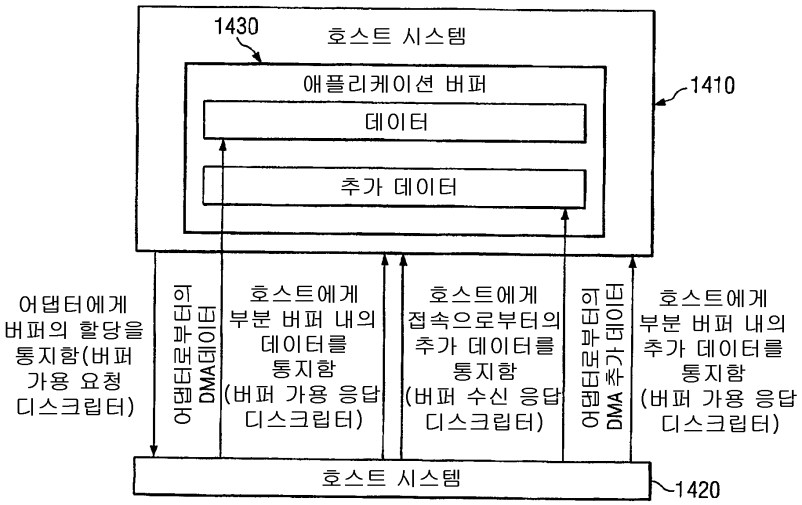
도면12



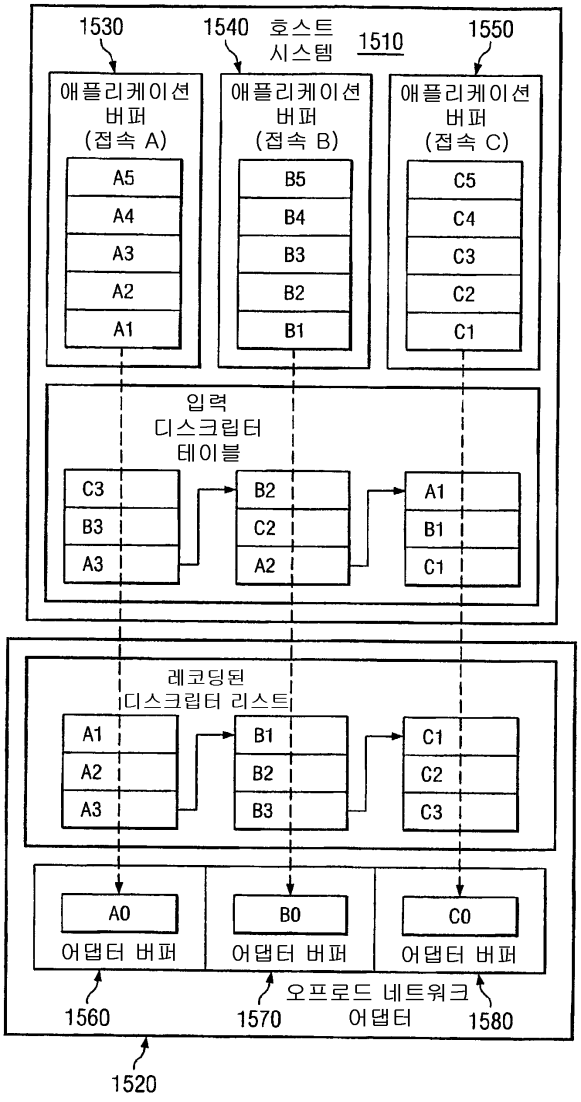
도면13



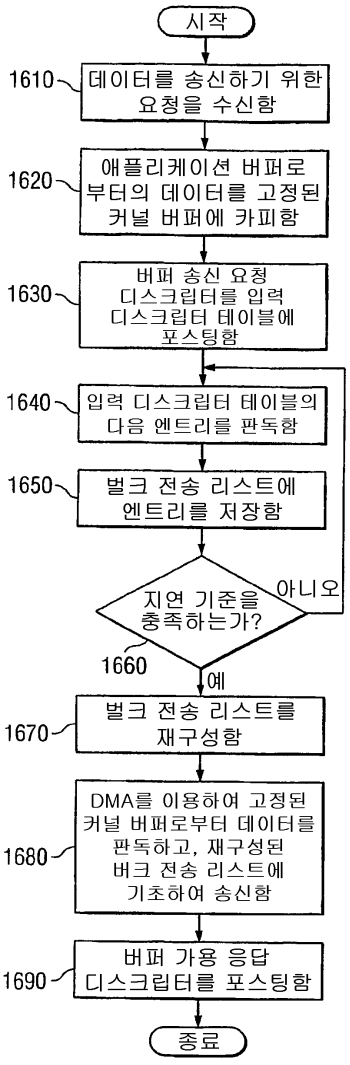
도면14


1430
호스트 시스템
애플리케이션 버퍼
데이터
추가 데이터
1410
어댑터에게 버퍼의 할당을 통지함(버퍼 가용 요청 디스크립터)
어댑터로부터의 DMA데이터
호스트에게 부분 버퍼 내의 데이터를 통지함 (버퍼 가용 응답 디스크립터)
호스트에게 접속으로부터의 추가 데이터를 통지함 (버퍼 수신 응답 디스크립터)
어댑터로부터의 DMA 추가데이터
호스트에게 부분 버퍼 내의 추가 데이터를 통지함 (버퍼 가용 응답 디스크립터)
호스트 시스템
1420

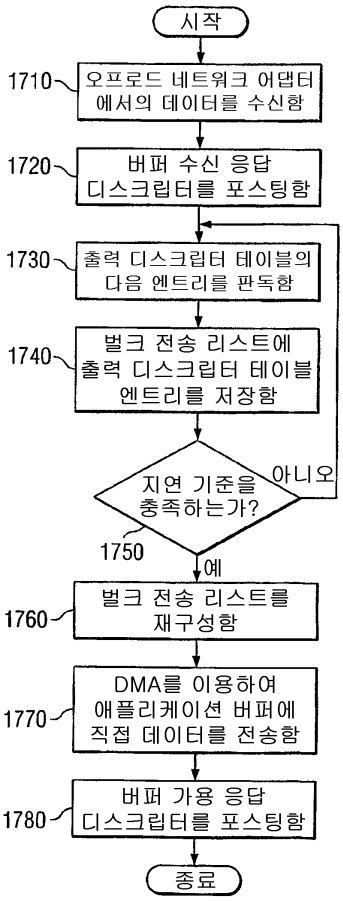
도면15


1530
1540
호스트 시스템
1510
1550
애플리케이션 버퍼 (접속 A)
A5
A4
A3
A2
A1
애플리케이션 버퍼 (접속 B)
B5
B4
B3
B2
B1
애플리케이션 버퍼 (접속 C)
C5
C4
C3
C2
C1
입력 디스크립터 테이블
C3
B3
A3
B2
C2
A2
A1
B1
C1
레코딩된 디스크립터 리스트
A1
A2
A3
B1
B2
B3
C1
C2
C3
A0
어댑터 버퍼
B0
어댑터 버퍼
C0
어댑터 버퍼
1560
1570
오프로드 네트워크 어댑터
1580
1520

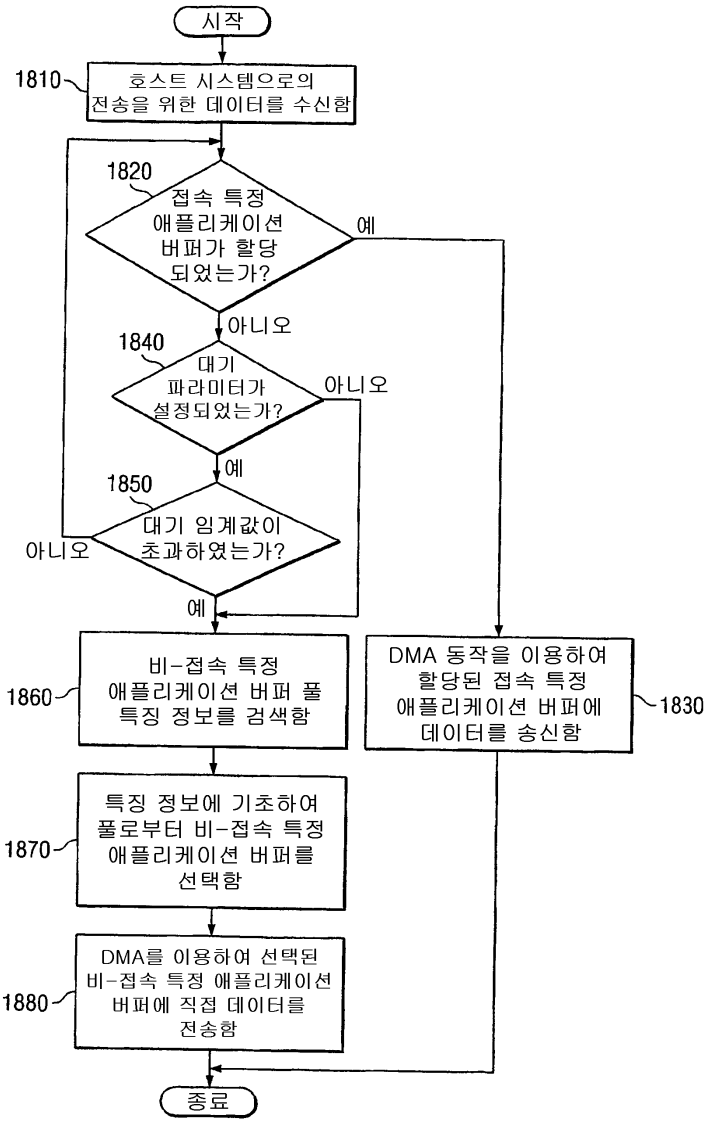
도면16


시작
1610
데이터를 송신하기 위한 요청을 수신함
1620
애플리케이션 버퍼로부터의 데이터를 고정된 커널 버퍼에 카피함
1630
버퍼 송신 요청 디스크립터를 입력 디스크립터 테이블에 포스팅함
1640
입력 디스크립터 테이블의 다음 엔트리를 판독함
1650
벌크 전송 리스트에 엔트리를 저장함
지연 기준을 충족하는가?
아니오
1660
예
1670
벌크 전송 리스트를 재구성함
1680
DMA를 이용하여 고정된 커널 버퍼로부터 데이터를 판독하고, 재구성된 버크 전송 리스트에 기초하여 송신함
1690
버퍼 가용 응답 디스크립터를 포스팅함
종료

도면17


시작
1710
오프로드 네트워크 어댑터에서의 데이터를 수신함
1720
버퍼 수신 응답 디스크립터를 포스팅함
1730
출력 디스크립터 테이블의 다음 엔트리를 판독함
1740
벌크 전송 리스트에 출력 디스크립터 테이블 엔트리를 저장함
지연 기준을 충족하는가?
아니오
1750
예
1760
벌크 전송 리스트를 재구성함
1770
DMA를 이용하여 애플리케이션 버퍼에 직접 데이터를 전송함
1780
버퍼 가용 응답 디스크립터를 포스팅함
종료

도면18


시작
1810
호스트 시스템으로의 전송을 위한 데이터를 수신함
1820
접속 특정 애플리케이션 버퍼가 할당 되었는가?
예
아니오
1840
대기 파라미터가 설정되었는가?
아니오
예
1850
대기 임계값이 초과하였는가?
아니오
예
1860
비-접속 특정 애플리케이션 버퍼 풀 특징 정보를 검색함
1870
특징 정보에 기초하여 풀로부터 비-접속 특정 애플리케이션 버퍼를 선택함
1880
DMA를 이용하여 선택된 비-접속 특정 애플리케이션 버퍼에 직접 데이터를 전송함
DMA 동작을 이용하여 할당된 접속 특정 애플리케이션 버퍼에 데이터를 송신함
1830
종료