

(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(51) Int. Cl. ⁶ H04L 29/02	(11) 공개번호 특 1999-0072861
	(43) 공개일자 1999년 09월 27일
(21) 출원번호	10-1999-0005988
(22) 출원일자	1999년 02월 23일
(30) 우선권주장	1998-042656 1998년 02월 24일 일본(JP) 1998-097917 1998년 04월 09일 일본(JP) 1998-112355 1998년 04월 22일 일본(JP)
(71) 출원인	캐논 가부시끼가이샤 미따라이 하지메
(72) 발명자	일본 도쿄도 오오따꾸 시모마루꼬 3쵸메 30방 2고 니이다미쯔오 일본도쿄도오오따꾸시모마루꼬3쵸메30방2고캐논가부시끼가이샤내 고바야시다까시 일본도쿄도오오따꾸시모마루꼬3쵸메30방2고캐논가부시끼가이샤내 하따에시니찌 일본도쿄도오오따꾸시모마루꼬3쵸메30방2고캐논가부시끼가이샤내 오니시신지
(74) 대리인	일본도쿄도오오따꾸시모마루꼬3쵸메30방2고캐논가부시끼가이샤내 구영창, 장수길, 주성민

심사청구 : 있음

(54) 데이터통신시스템, 데이터통신방법, 데이터통신장치및디지털인터페이스

요약

소스 노드와 하나이상의 수신지 노드가 논리적으로 접속되고 그 논리적 접속 관계를 식별하는 접속 ID가 노드사이의 데이터 통신을 제어하기 위해 사용되는 통신 시스템 및 통신 프로토콜에 관한 것이다. 또한, i번째(i는 임의의 정수)데이터를 송신하기 위한 시간과 (i+1)번째 데이터를 송신하기 위한 시간 사이의 최적의 시간 간격을 설정할 수 있는 효율적인 통신 시스템 및 통신 프로토콜에 관한다. 또한, i번째 데이터가 정상적으로 수신되지 않을 때, 수신 처리가 늦은 수신지 노드와 송신 처리가 빠른 소스 노드사이에서 재시도가 불필요하게 발생하는 것을 방지하기 위해 소정의 시간동안만 재시도가 금지되는 통신 시스템 및 통신 프로토콜에 관한다.

대표도

도2

색인어

데이터 통신 시스템, 디지털 인터페이스, 컴퓨터, MPU, 통신 프로토콜

명세서

도면의 간단한 설명

- 도 1은 종래 시스템의 설명도.
- 도 2는 실시예의 통신 시스템 구성의 일예를 도시한 블록도.
- 도 3은 실시예의 통신 프로토콜의 기본 구성을 도시한 개략도.
- 도 4a, 도 4b 및 도 4c는 제1실시예의 통신 프로토콜이 기본 통신 프로시저를 도시한 시퀀스도.
- 도 5는 제1실시예의 비동기 방송 패킷의 구성을 도시한 도면.
- 도 6의 (a) 및 도 6의 (b)는 각 노드의 어드레스 공간을 도시한 설명도.
- 도 7은 오브젝트 데이터의 전송 모델을 도시한 설명도.

- 도 8은 실시예의 1394 인터페이스의 구성을 도시한 설명도.
- 도 9는 제1실시예에서 규정된 응답 기간을 도시한 시퀀스도.
- 도 10은 제2실시예에서 규정된 단일 단계 재시도 동작의 일예를 도시한 상태 천이도.
- 도 11은 제2실시예에서 규정된 이중 단계 재시도 동작의 일예를 도시한 상태 천이도.
- 도 12는 제2실시예의 통신 프로토콜의 기본 통신 프로시저를 도시한 시퀀스도.
- 도 13은 제2실시예의 비동기 방송 패킷을 도시한 도면.
- 도 14는 제2실시예에서 규정된 확인 응답 패킷의 구성을 도시한 도면.
- 도 15는 실시예의 최소 재시도 기간을 설정하는 동작을 도시한 도면.
- 도 16은 실시예의 재시도 코드의 종류를 도시한 도면.
- 도 17은 실시예의 확인 응답 코드의 종류를 도시한 도면.

<도면의 주요 부분에 대한 부호의 설명>

- 10 : 컴퓨터
- 12 : MPU
- 14, 44, 62 : 1394 인터페이스
- 16 : 조작부
- 18 : 디코더
- 20 : 디스플레이
- 22 : 하드 디스크
- 24 : 내부 메모리
- 26 : PCI 버스
- 28 : 디지털 카메라 레코더
- 30 : 화상 픽업부(opt)
- 32 : 아날로그-디지털 변환기
- 34 : 비디오 처리부
- 36 : 압축/신장부
- 38, 40 : 메모리
- 42 : 데이터 셀렉터
- 46, 48 : 메모리 제어부
- 50 : 시스템 컨트롤러
- 52 : 조작부
- 54 : 전자 뷰 파인더(EVF)
- 56 : D/A 변환기
- 58 : 기록/재생기
- 60 : 프린터
- 64 : 데이터 셀렉터
- 66 : 조작부
- 68 : 프린터 컨트롤러
- 70 : 디코더
- 72 : 내부 메모리
- 74 : 화상 처리부
- 76 : 드라이버
- 78 : 프린터 헤드
- 300 : 컨트롤러
- 302 : 소스 노드

304 : 수신지 노드

308 : 오브젝트 데이터

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

본 발명은 데이터 통신 시스템, 데이터 통신 방법, 데이터 통신 장치 및 디지털 인터페이스에 관한 것으로, 특히 정보 데이터(화상 데이터를 포함함) 및 커맨드 데이터가 혼합되어 고속으로 통신이 실행되는 네트워크 및 상기 네트워크에 적합한 통신 프로토콜에 관한 것이다.

지금까지 하드 디스크 및 프린터는 퍼스널 컴퓨터(이하, PC라 함)의 주변 장치중에서 가장 높은 사용 빈도를 갖고 있다. 이들 주변 장치는 전용의 입력/출력 인터페이스, SCSI(small computer system interfaces) 또는 다른 범용 디지털 인터페이스를 거쳐 PC에 접속된다.

한편, 최근에는 디지털 카메라, 디지털 비디오 카메라 및 다른 AV(Audio/Visual) 장치가 PC 주변 장치로서 대중의 관심을 얻고 있다. AV 장치도 전용의 인터페이스를 거쳐 PC에 접속된다.

도 1은 PC 및 AV 장치로 구성된 종래 통신 시스템을 도시한 도면이다.

도 1에서, (101)은 AV 장치 또는 디지털 카메라, (102)는 PC, (103)은 프린터이다.

디지털 카메라(101)는 촬영된 화상을 압축하여 기록하는 메모리(104), 메모리(104)내에 기록되어 있는 압축된 화상 데이터를 신장하여 디코딩하는 디코딩부(105), 화상 처리부(106), D/A 변환기(107), EVF를 포함하는 디스플레이(108), 및 디지털 카메라(101)와 PC(102)를 접속하는 전용의 디지털 I/O부(109)를 포함한다.

PC(102)는 PC(102)와 디지털 카메라(101)를 접속하는 전용의 디지털 I/O부(110), 키보드, 마우스 등을 포함하는 조작부(111), 압축된 화상 데이터를 신장하여 디코딩하는 디코딩부(112), 디스플레이(113), 하드 디스크(114), RAM 또는 다른 메모리(115), MPU(116), PCI 버스(117), 및 PC(102)와 프린터(103)를 접속하는 SCSI 인터페이스(118)를 포함한다.

프린터(103)는 프린터(103)와 PC(102)를 접속하는 SCSI 인터페이스(119), 메모리(120), 프린터 헤드(121), 프린터(103)의 동작을 제어하는 프린터 컨트롤러(122), 및 드라이버(123)를 포함한다.

종래 컴퓨터 시스템에서는 디지털 카메라(101)의 디지털 인터페이스 또는 디지털 I/O부(109)가 프린터(103)의 디지털 인터페이스 또는 SCSI 인터페이스(110)와 호환되지 않으므로, 그들을 직접 상호접속할 수 없었다. 예를 들면, 정지 화상은 반드시 PC를 거쳐 디지털 카메라(101)에서 프린터(103)로 전송될 필요가 있다.

또한, 종래 전용의 인터페이스 또는 SCSI 인터페이스에서는 AV 장치가 유지하는 정지 화상 또는 움직임 화상 등의 대용량 데이터를 취급할 때, 데이터 전송 속도가 낮고, 파라렐 통신을 위한 통신 케이블이 두 겹고, 접속가능한 주변 장치의 종류가 몇 개 없고, 접속 시스템이 제한되는 수 많은 문제가 발생되어, 실시간 데이터 전송을 실행할 수 없다.

상기 문제를 해소하기 위한 차세대 고속 고성능 디지털 인터페이스중의 하나로서 IEEE(The institute of Electrical and Electronics Engineers, Inc.) 1394-1995 규격이 알려져 있다.

IEEE 1394-1995 규격에 따르는 디지털 인터페이스(이하, 1394 인터페이스라 함)는 다음의 특징을 갖는다.

- (1) 데이터 전송 속도가 높고,
- (2) 실시간 데이터 전송 시스템(즉, 등시성 전송 시스템(Isochronous transfer system)) 및 비동기 전송 시스템(Asynchronous transfer system)이 지원되고,
- (3) 자유도가 높은 접속 구성(토폴로지)을 구축할 수 있고,
- (4) 플러그-앤드-플레이(plug-and-play) 기능 및 핫라인 플러그/언플러그(hot-line plug/unplug) 기능이 지원됨.

IEEE 1394-1995 규격에서는 커넥터의 물리적 전기적 구조, 두가지 기본적인 데이터 전송 시스템 등은 규정되어 있지만, 어떤 종류의 데이터가 어떤 데이터 포맷으로 어떤 통신 프로토콜에 따라 송신/수신되는가는 규정되어 있지 않다.

또한, IEEE 1394-1995 규격의 등시성 전송 시스템에서는 송출 패킷에 대한 응답이 규정되어 있지 않으므로, 각 등시성 패킷이 확실히 수신되는 것이 보장되지 않는다. 따라서, 다수의 연속 데이터를 확실히 전송해야 할 때 또는 하나의 파일 데이터를 다수의 데이터로 분할하여 확실히 전송해야 할 때에는 등시성 전송 시스템을 사용할 수 없다.

또한, IEEE 1394-1995 규격의 등시성 전송 시스템에서는 전송 대역에 빈곳이 있더라도 통신 총수는 64로 제한된다. 따라서, 다수의 통신이 몇개의 통신 대역에서 실행될 때, 등시성 전송 시스템을 사용할 수 없다.

또한, IEEE 1394-1995 규격에서는 노드 전원의 ON/OFF, 노드의 접속/비접속 등에 응답하여 버스 리셋이

발생하면, 데이터 전송을 인터럽트해야 한다. 그러나, IEEE 1394-1995 규격에서는 데이터 전송이 버스 리셋 또는 전송시 에러에 의해 인터럽트될 때, 데이터의 어떤 내용이 상실되었는가를 알 수 없다. 또한, 인터럽트된 전송으로부터 일단 복귀하기 위해서는 매우 복잡한 통신 프로시저를 실행할 필요가 있다.

여기서, 버스 리셋은 새로운 토폴로지의 인식 및 각 노드에 할당된 어드레스(노드 ID)의 설정을 자동으로 실행하는 기능을 나타낸다. 따라서, 플러그-앤드-플레이 기능 및 핫라인 플러그/언플러그 기능은 IEEE 1394-1995 규격에 마련될 수 있다.

또한, IEEE 1394-1995 규격에 따르는 통신 시스템에서는 하나 이상의 세그먼트 데이터로 분할하고, 실시간 특성을 갖는 것이 아니고 신뢰성을 갖는 것이 요구되는 비교적 대량의 오브젝트 데이터(예를 들면, 정지 화상 데이터, 그래픽 데이터, 텍스트 데이터, 파일 데이터, 프로그램 데이터 등)를 연속해서 전송하는 것에 대하여 통신 프로토콜이 명확히 제안되어 있지 않다.

또한, IEEE 1394-1995 규격에 따르는 통신 시스템에서는 데이터가 비동기 방송(broadcast)되는 통신 시스템을 사용하는 다수의 장치사이의 데이터 통신을 실현하는 것에 대해서도 통신 프로토콜이 명확히 제안되어 있지 않다.

발명이 이루고자하는 기술적 과제

본 발명의 목적은 상술한 문제를 해소하는 것이다.

본 발명의 다른 목적은 데이터 통신 시스템, 데이터 통신 방법, 데이터 통신 장치 및 디지털 인터페이스에 있어서 실시간 특성이 요구되지 않는 오브젝트 데이터를 연속해서 확실하게 전송할 수 있는 기술을 제공하는 것이다.

본 발명의 또 다른 목적은 데이터 통신 시스템, 데이터 통신 방법, 데이터 통신 장치 및 디지털 인터페이스에 있어서 연속해서 전송된 데이터 사이의 시간 간격을 최적화할 수 있어, 일련의 데이터 전송에서의 불필요한 인터럽트를 용이하고 확실하며 효율적으로 방지할 수 있는 기술을 제공하는 것이다.

본 발명의 또 다른 목적은 데이터 통신 시스템, 데이터 통신 방법, 데이터 통신 장치 및 디지털 인터페이스에 있어서 불필요한 재시도(retry) 발생을 용이하고 확실하게 방지하여, 효율적인 데이터 통신을 실행할 수 있는 기술을 제공하는 것이다.

발명의 구성 및 작용

그러한 목적을 위한 바람직한 실시예로서, 본 발명은 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 비동기 통신을 실행하는 소스 노드, 소스 노드에서 전송된 데이터를 수신하는 하나 이상의 수신지 노드, 및 소스 노드와 하나 이상의 수신지 노드사이의 논리적 접속 관계를 설정하는 컨트롤러를 포함하며, 소스 노드와 상기 컨트롤러의 적어도 하나가 비동기 통신을 실행하는 타이밍을 제어하는 데이터 통신 시스템을 기재한다.

다른 바람직한 실시예로서, 논리적 접속 관계에 따라 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 방송 통신을 실행하는 소스 노드, 및 논리적 접속 관계에 따라 소스 노드에서 전송된 데이터를 수신하는 하나 이상의 수신지 노드를 포함하며, 소스 노드가 방송 통신을 실행하는 타이밍을 제어하는 데이터 통신 시스템을 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와 하나 이상의 수신지 노드사이의 논리적 접속 관계를 설정하는 스텝, 하나 이상의 수신지 노드로 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 비동기 통신을 실행하는 스텝, 비동기 통신을 실행하는 타이밍을 제어하는 스텝, 및 비동기 통신을 사용하여 전송된 데이터를 수신하기 위해 논리적 접속 관계를 사용하는 스텝을 포함하는 데이터 통신 방법을 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 논리적 접속 관계에 따라 하나 이상의 수신지 노드로 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 방송 통신을 실행하는 스텝, 방송 통신을 실행하는 타이밍을 제어하는 스텝, 및 논리적 접속 관계에 따라 소스 노드에서 전송된 데이터를 수신하는 스텝을 포함하는 데이터 통신 방법을 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 하나 이상의 세그먼트를 포함하는 데이터를 다수의 통신 패킷으로 패킷화하는 스텝, 및 하나 이상의 수신지 노드와의 논리적 접속 관계 세트에 따라 통신 패킷을 연속해서 전송하는 스텝을 포함하며, 통신 패킷이 소정의 시간 경과후 비동기로 전송되는 데이터 통신 방법을 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와의 논리적 접속 관계 세트에 따라 소스 노드에서 연속해서 전송되고, 소정의 시간 경과후 비동기로 전송되는 통신 패킷을 수신하는 스텝, 및 통신 패킷에 포함된 데이터를 다른 장치와 공통인 메모리 공간에 기록하는 스텝을 포함하는 데이터 통신 방법을 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와 하나 이상의 수신지 노드 사이의 논리적 접속 관계를 설정하는 스텝, 소스 노드 및 하나 이상의 수신지 노드에게 상기 논리적 접속 관계를 식별하는 접속 ID를 통지하는 스텝, 및 논리적 접속 관계에 따라 연속해서 전송된 통신 패킷의 시간 간격을 상기 소스 노드에 있어서 설정하는 스텝을 포함하는 데이터 통신 방법을 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 하나 이상의 세그먼트를 포함하는 데이터를 다수의 통신 패킷으로 패킷화하는 유닛, 및 하나 이상의 수신지 노드와의 논리적 접속 관계 세트에 따라 통신 패킷을 연속해서 전송하는 유닛을 포함하며, 통신 패킷이 소정의 시간 경과후 비동기로 전송되는 데이터 통신 장치를

기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와의 논리적 접속 관계 세트에 따라 소스 노드에서 연속해서 전송된 통신 패킷을 수신하는 유닛, 및 통신 패킷에 포함된 데이터를 다른 장치와 공동인 메모리 공간에 기록하는 유닛을 포함하며, 통신 패킷이 소정의 시간 경과후 비동기로 전송되는 데이터 통신 장치를 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와 하나이상의 수신지 노드 사이의 논리적 접속 관계를 설정하고, 논리적 접속 관계에 따라 연속해서 전송된 통신 패킷의 시간 간격을 소스 노드에 있어서 설정하는 유닛, 및 소스 노드 및 하나이상의 수신지 노드에게 논리적 접속 관계를 식별하는 접속 ID를 통지하는 유닛을 포함하는 데이터 통신 장치를 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 하나이상의 세그먼트를 포함하는 데이터를 다수의 통신 패킷으로 패킷화하는 유닛, 및 하나이상의 수신지 노드와의 논리적 접속 관계 세트에 따라 통신 패킷을 연속해서 전송하는 유닛을 포함하며, 통신 패킷이 소정의 시간 경과후 비동기로 전송되는 디지털 인터페이스를 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와의 논리적 접속 관계 세트에 따라 소스 노드에서 연속해서 전송된 통신 패킷을 수신하는 유닛, 및 통신 패킷에 포함된 데이터를 다른 장치와 공동인 메모리 공간에 기록하는 유닛을 포함하며, 통신 패킷이 소정의 시간 경과후 비동기로 전송되는 디지털 인터페이스를 기재한다.

또 다른 바람직한 실시예로서, 본 발명은 소스 노드와 하나이상의 수신지 노드 사이의 논리적 접속 관계를 설정하고, 논리적 접속 관계에 따라 연속해서 전송된 통신 패킷의 시간 간격을 소스 노드에 있어서 설정하는 유닛, 및 소스 노드 및 하나이상의 수신지 노드에게 논리적 접속 관계를 식별하는 접속 ID를 통지하는 유닛을 포함하는 디지털 인터페이스를 기재한다.

본 발명의 또 다른 목적 및 그의 효과는 실시예의 다음의 상세한 설명에서 충분히 명확하게 될 것이다.

이하, 본 발명의 바람직한 실시예를 첨부 도면을 참조하여 상세히 설명한다.

도 2는 실시예의 데이터 통신 시스템 구성의 일예를 도시한 것이다. 도 2에 도시한 바와 같이, 실시예의 데이터 통신 시스템은 컴퓨터(10), 디지털 비디오 카메라 레코더(28) 및 프린터(60)로 구성된다.

컴퓨터(10)의 구성을 먼저 설명한다. (12)는 컴퓨터(10)의 동작을 제어하는 마이크로프로세서부(MPU)이다. (14)는 IEEE 1394-1995 규격에 따른 기능 및 실시예에서 규정되는 통신 프로토콜에 관한 기능을 갖는 1394 인터페이스이다. (16)은 키보드, 마우스 등으로 구성된 조작부이다. (18)은 압축/인코딩된 디지털 데이터(움직임 화상 데이터, 정지 화상 데이터, 오디오 데이터 등)를 디코딩하는 디코더이다. (20)은 CRT 디스플레이, 액정 패널 또는 다른 디스플레이 장치로 구성된 디스플레이이다. (22)는 여러 가지 디지털 데이터(움직임 화상 데이터, 정지 화상 데이터, 오디오 데이터, 그래픽 데이터, 텍스트 데이터, 프로그램 데이터 등)를 기록하기 위한 하드 디스크(HD)이다. (24)는 내부 메모리이다. (26)은 컴퓨터(10) 내의 처리부를 상호접속하는 PCI 버스 또는 내부 버스이다.

이하, 디지털 비디오 카메라 레코더(이하, DVCR이라 함)(28)의 구성을 설명한다. (30)은 피사체의 광학적 화상을 전기적 신호로 변환하는 화상 픽업부(opt)이다. (32)는 아날로그-디지털(A/D) 변환기이다. (34)는 디지털화된 움직임 화상 또는 정지 화상을 소정의 포맷의 디지털 화상 데이터로 변환하는 비디오 처리부이다. (36)은 압축/인코딩된 디지털 데이터(움직임 화상 데이터, 정지 화상 데이터, 오디오 데이터 등)를 디코딩하는 기능 및 고효율로 디지털 화상 데이터를 인코딩하는 기능을 갖는 압축/신장부이다(예를 들면, MPEG 또는 DV 시스템에서, 데이터는 소정의 화상 단위로 직교 변환되고, 양자화되며, 가변 길이 등으로 인코딩됨). (38)은 고효율로 인코딩된 디지털 화상 데이터를 일시 저장하는 메모리이다. (40)은 고효율 인코딩이 실시되지 않은 디지털 화상 데이터를 일시 저장하는 메모리이다. (42)는 셀렉터이다. (44)는 IEEE 1394-1995 규격에 따른 기능 및 실시예에서 규정되는 통신 프로토콜에 관한 기능을 갖는 1394 인터페이스이다. (46), (48)은 메모리(38), (40)의 기록 및 판독을 제어하는 메모리 제어부이다. (50)은 마이크로컴퓨터를 갖고 DVCR(28)의 동작을 제어하는 시스템 컨트롤러이다. (52)는 리모트 컨트롤러, 조작 패널 등을 포함하는 조작부이다. (54)는 전자 뷰 파인더(EVF)이다. (56)은 D/A 컨버터이다. (58)은 여러 가지 디지털 데이터(움직임 화상 데이터, 정지 화상 데이터, 오디오 데이터 등)를 기록/재생하기 위한 자기 테이프, 자기 디스크, 자기 광학 디스크 또는 다른 기록 매체가 마련된 기록/재생기이다.

이하, 프린터(60)의 구성을 설명한다. 프린터(60)는 IEEE 1394-1995 규격에 따른 기능 및 실시예에서 규정되는 통신 프로토콜에 관한 기능을 갖는 1394 인터페이스(62), 데이터 셀렉터(64), 조작 버튼, 터치 패널 등으로 구성된 조작부(66), 프린터(60)의 동작을 제어하는 프린터 컨트롤러(68), 디코더(70), 내부 메모리(72), 1394 인터페이스를 거쳐 수신되는 정지 화상 데이터, 텍스트 데이터, 그래픽 데이터 등을 처리하는 화상 처리부(74), 드라이버(76) 및 프린터(78)를 포함한다.

도 2에 도시한 바와 같이, 각 통신 장치(이하, 노드라 함)에 대하여, 컴퓨터(10), DVCR(28) 및 프린터(60)는 1394 인터페이스(14, 44, 62)를 거쳐 상호접속된다(이하, 1394 인터페이스를 포함하는 네트워크를 1394 시리얼 버스와 함). 각 노드에 있어서, 여러 가지 오브젝트 데이터(예를 들면, 움직임 화상 데이터, 정지 화상 데이터, 오디오 데이터, 그래픽 데이터, 텍스트 데이터, 프로그램 데이터 등)는 송신/수신될 수 있고, 소정의 통신 프로토콜을 규정하는 것에 의해 커맨드 데이터를 사용하여 원격 조작이 실현될 수 있다. 실시예에서는 비동기 전송 시스템을 사용하는 통신 프로토콜을 규정한다.

이하, 실시예의 통신 시스템을 구성하는 노드의 동작을 도 2를 참조하여 설명한다.

먼저, 컴퓨터(10)를 구성하는 처리부의 기능 및 동작을 설명한다.

실시예에서, 컴퓨터(10)는 DVCR(28)과 프린터(60) 사이에서 화상 데이터의 송신/수신을 제어하는 컨트롤러 또는 DVCR(28) 및 프린터(60)를 원격 조작하는 컨트롤러로서 기능한다.

MPU(12)는 하드 디스크(22)에 기록된 소프트웨어를 실행하고, 여러 가지 데이터를 내부 메모리(24)로 이동시킨다. 또한, MPU(12)는 내부 버스(26)를 거쳐 접속된 처리부를 조절하는 동작을 실행한다.

1394 인터페이스(14)는 1394 시리얼 버스상으로 전송된 화상 데이터를 수신할 수 있고 또한 하드 디스크(22) 및 내부 메모리(24)에 기록된 화상 데이터를 1394 시리얼 버스로 송신할 수 있다. 또한, 1394 인터페이스(14)는 1394 시리얼 버스상의 다른 노드를 원격 조작하는 커맨드 데이터를 송신할 수 있다. 또한, 1394 인터페이스(14)는 1394 시리얼 버스를 거쳐 전송된 신호를 다른 노드로 전송하는 기능도 갖고 있다.

사용자는 조작부(16)를 거쳐 소망 소프트웨어를 선택하여, MPU(12)가 하드 디스크(22)에 기록된 소프트웨어를 조작하게 한다. 여기서, 소프트웨어에 관한 정보는 디스플레이(20)에 의해 사용자에게 제출된다. 디코더(18)는 1394 시리얼 버스상에서 수신된 화상 데이터를 소프트웨어에 따라 디코드한다. 디코드된 화상 데이터는 디스플레이(20)에 의해 사용자에게 제출된다.

이하, DVCR(28)을 구성하는 처리부의 기능 및 동작을 설명한다.

실시예에서, DVCR(28)은 예를 들면, 실시예의 통신 프로토콜에 따라 화상 데이터의 비동기 전송을 실행하는 화상 전송 장치(소스 노드)로서 기능한다.

화상 픽업부(30)는 피사체의 광학적 화상을 휘도 신호 Y와 색차 신호C로 구성된 전기적 신호로 변환하고, 그 전기적 신호를 A/D 변환기(32)에 공급한다.

비디오 처리부(34)는 디지털화된 휘도 신호 및 색차 신호에 소정의 화상 처리를 적용하고, 그 신호를 멀티플렉스한다. 압축/신장부(36)는 디지털화된 휘도 신호 및 색차 신호의 데이터량을 압축한다. 여기서, 압축/신장부(36)는 독립적인 압축 처리 회로를 사용하여 휘도 신호 및 색차 신호를 병렬로 처리한다. 또한, 이것은 공통의 압축 처리 회로를 사용하여 시분할로 신호를 처리할 수도 있다.

또한, 전송 경로에서의 에러에 저항하기 위해, 압축/신장부(36)는 압축된 화상 데이터에 셔플링 처리(shuffling processing)를 적용한다. 이것에 의해, 연속 코드 에러(즉, 버스트 에러)는 용이하게 수리 또는 보간될 수 있는 불연속 에러(discrete error)(즉, 랜덤 에러)로 변환될 수 있다. 여기서, 화면내 거친 화상에 기인하는 정보량의 편차가 균일할 때, 이 처리는 압축 처리에 앞서 실행되는 것이 바람직할 수 있고, 이것은 가변 런 길이(variable run length) 또는 다른 길이에 의한 인코딩에 편리하게 된다.

압축/신장부(36)에서는 셔플링을 복구하기 위해 압축된 화상 데이터에 데이터 식별 정보 ID가 부가된다. 압축/신장부(36)는 기록/재생시 에러를 저감하기 위해 압축된 화상 데이터에 에러 정정 코드 ECC를 부가한다.

압축/신장부(36)에서 압축된 화상 데이터는 메모리(38) 및 기록/재생기(58)로 공급된다. 기록/재생기(58)는 ID 및 ECC가 부가되어 있는 압축된 화상 데이터를 자기 테이프 또는 다른 기록 매체에 기록한다. 여기서, 압축된 화상 데이터는 오디오 데이터용 영역과 다른 또는 그것과 독립인 기록 영역에 기록된다.

한편, 비디오 처리부(34)에서 D/A 변환기(56)로 공급된 화상 데이터는 D/A 변환된다. EVF(54)는 D/A 변환기(56)에서 공급된 아날로그 화상 신호를 나타낸다. 또한, 비디오 처리부(34)에서 처리된 화상 데이터도 메모리(40)로 공급된다. 여기서, 압축되지 않은 화상 데이터는 메모리(40)에 저장된다.

데이터 셀렉터(42)는 사용자의 명령에 따라 메모리(38) 또는 (40)를 선택하고, 압축된 화상 데이터 또는 압축되지 않은 화상 데이터를 1394 인터페이스(44)로 공급한다. 또한, 데이터 셀렉터(42)는 1394 인터페이스(44)에서 공급된 화상 데이터를 메모리(38) 또는 (40)에 공급한다.

1394 인터페이스(44)는 다음에 설명하는 바와 같은 실시예의 통신 프로토콜에 따라 압축된 화상 데이터 또는 압축되지 않은 화상 데이터의 비동기 전송을 실행한다. 또한, 1394 인터페이스(44)는 1394 시리얼 버스를 거쳐 DVCR(28)을 제어하는 제어 커맨드를 수신한다. 수신된 제어 커맨드는 데이터 셀렉터(42)를 거쳐 시스템 컨트롤러(50)에 공급된다. 1394 인터페이스(44)는 제어 커맨드에 대한 응답을 반환한다.

이하, 프린터(60)를 구성하는 각 처리부의 기능 및 동작을 설명한다.

실시예에서 프린터(60)는 예를 들면, 실시예의 통신 프로토콜에 따라 비동기 전송된 화상 데이터를 수신하여 프린트하는 화상 수신 장치(수신지 노드)로서 기능한다.

1394 인터페이스(62)는 1394 시리얼 버스를 거쳐 비동기 전송된 제어 커맨드 또는 화상 데이터를 수신한다. 또한, 1394 인터페이스(62)는 제어 커맨드에 대한 응답을 송출한다.

수신된 화상 데이터는 데이터 셀렉터(64)를 거쳐 디코더(70)에 공급된다. 디코더(70)는 화상 데이터를 디코드하고, 그 결과를 화상 처리부(74)로 송신한다. 화상 처리부(74)는 메모리(72)에 디코드된 화상 데이터를 일시적으로 저장한다.

또한, 화상 처리부(74)는 메모리(72)에 일시 저장되어 있는 화상 데이터를 프린트될 데이터로 변환하고, 그 데이터를 프린터 헤드(78)에 공급한다. 프린터 헤드(78)는 프린터 컨트롤러(68)의 제어하에 프린팅을 실행한다.

한편, 수신된 제어 커맨드는 데이터 셀렉터(64)를 거쳐 프린터 컨트롤러(68)로 송신된다. 프린터 컨트롤러(68)는 제어 데이터에 따라 프린팅에 관한 여러 가지 제어를 실행한다. 예를 들면, 드라이버(76)에

의한 용지 공급, 프린터 헤드(78)의 위치 등이 제어된다.

이하, 실시예의 1394 인터페이스(14, 44, 62)의 구성을 도 8을 참조하여 상세히 설명한다.

1394 인터페이스는 다수의 계층으로 기능적으로 구성된다. 도 8에서, 1394 인터페이스는 IEEE 1394-1995 규격에 따른 통신 케이블(801)을 거쳐 다른 노드의 1394 인터페이스에 접속된다. 또한, 1394 인터페이스는 적어도 하나의 통신 포트(802)를 갖고, 각각의 통신 포트(802)는 하드웨어에 포함된 물리적 계층(803)에 접속된다.

도 8에서, 하드웨어는 물리적 계층(803) 및 링크 계층(804)으로 구성된다. 물리적 계층(803)은 다른 노드와의 물리적 전기적 인터페이스, 버스 리셋의 검출 및 입력/출력 신호의 처리, 인코딩/디코딩, 권리를 사용하는 버스의 조정(reconciliation) 등을 실행한다. 또한, 링크 계층(804)은 통신 패킷의 발생, 여러 가지 통신 패킷의 송신/수신, 사이클 타이머의 제어 등을 실행한다. 또한, 링크 계층(804)은 다음에 설명하는 바와 같은 비동기 방송 패킷을 발생하고 송신/수신하는 기능을 제공한다.

또한, 도 8에서, 펌웨어는 트랜잭션 계층(805) 및 시리얼 버스 매니지먼트(806)를 구비한다. 트랜잭션 계층(805)은 비동기 전송 시스템을 제어하고, 여러가지 트랜잭션(판독, 기록, 로크)을 제공한다. 또한, 트랜잭션 계층(805)은 다음에 설명하는 바와 같은 비동기 방송 트랜잭션의 기능을 제공한다. 시리얼 버스 매니지먼트(806)는 자기 노드를 제어하고, 다음에 설명하는 IEEE 1212 CSR 규격을 기초로 하여 자기 노드의 접속 상태, 자기 노드의 ID 정보 및 시리얼 버스 네트워크의 소스를 관리하는 기능을 제공한다.

도 8에 도시한 하드웨어 및 펌웨어는 실질적으로 1394 인터페이스를 구성하고, 기본 구성은 IEEE 1394-1995 규격에 의해 규정된다.

또한, 소프트웨어에 포함된 애플리케이션 계층(807)은 사용 애플리케이션 소프트웨어에 따라 변하고, 또한 이것은 어떤 오브젝트 데이터가 전송되고 어떻게 전송되는 가를 제어한다.

다음에 설명하는 실시예의 통신 프로토콜은 1394 인터페이스를 구성하는 하드웨어 및 펌웨어의 기능을 확장하여, 소프트웨어에게 새로운 전송 프로시저를 제공한다.

이하, 실시예에서 규정되는 통신 프로토콜의 기본 구성을 도 3을 참조하여 설명한다.

도 3은 컨트롤러(300), 소스 노드(302), $n(n \geq 1)$ 개의 수신지 노드(304), 소스 노드의 서브유닛(306) 및 정지 화상 데이터, 그래픽 데이터, 텍스트 데이터, 파일 데이터, 프로그램 데이터 등의 오브젝트(308)를 도시하고 있다.

수신지 노드(304)의 내부에 마련된 제1메모리 공간(310)은 소정의 수신지 오프셋(destination_offset #0)에 의해 지정된다. 제1접속(312)은 소스 노드(302)와 수신지 노드(304) 사이의 논리적 접속 관계(즉, 접속)를 나타낸다. 여기서, 수신지 오프셋은 n 개의 수신지 노드(304)의 메모리 공간을 공통으로 지정하는 어드레스를 의미한다.

수신지 노드(304)의 내부에 마련된 N 번째 메모리 공간(314)은 소정의 수신지 오프셋(destination_offset # n)에 의해 지정된다. n 번째 접속(316)은 소스 노드(302)와 수신지 노드(304) 사이의 논리적 접속 관계(즉, 접속)를 나타낸다.

실시예에서, 각 노드는 IEEE 1212 CSR(Control and Status Register Architecture) 규격 (또는 ISO/IEC 13213 : 1994 규격)에 따르는 64비트의 어드레스 공간에 의해 제1메모리 공간(310)에서 n 번째 메모리 공간(314)까지를 제어한다. IEEE 1212 CSR 규격은 시리얼 버스의 제어, 관리, 또는 어드레스 할당을 규정한다.

도 6의 (a) 및 도 6의 (b)는 각 노드의 어드레스 공간의 설명도이다. 도 6의 (a)는 64 비트의 어드레스에 의해 표현된 논리적 메모리 공간을 도시하고 있다. 또한, 도 6의 (b)는 도 6의 (a)에 도시한 어드레스 공간의 일부, 예를 들면 상위 16 비트가 $FFFF_{16}$ 을 형성하는 어드레스 공간을 도시하고 있다. 도 3에 도시한 첫 번째 메모리 공간(310)에서 n 번째 메모리 공간(314)까지로서, 도 6의 (b)에 도시한 메모리 공간의 일부가 사용된다. 메모리 공간(310) 내지 (314) 각각은 어드레스의 하위 48비트를 나타내는 수신지 오프셋에 의해 규정된다.

도 6의 (b)에서, 예를 들면, 000000000000_{16} 내지 $000000003FF_{16}$ 은 예약된 영역이고, 오브젝트 데이터(308)가 실제로 기록되는 영역은 하위 48 비트를 나타내는 $FFFFF000400_{16}$ 부터의 영역이다.

도 3에서, 소스 노드(302)는 다음에 설명하는 통신 프로토콜에 따라 오브젝트 데이터(308)를 전송하는 기능을 갖고, 수신지 노드(304)는 소스 노드(302)에서 전송된 오브젝트 데이터(308)를 수신하는 기능을 갖는다. 또한, 컨트롤러(300)는 다음에 설명하는 통신 프로토콜에 따라 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이의 논리적 접속 관계(즉, 접속)를 확립하고, 그 접속을 제어한다.

여기서, 컨트롤러(300), 소스 노드(302) 및 수신지 노드(304)는 따로따로 기능할 수도 있다. 또한, 컨트롤러(300) 및 소스 노드(302)는 동일한 노드에서 기능할 수도 있다. 또한, 컨트롤러(300) 및 수신지 노드(304)는 동일한 노드에서 기능할 수도 있다. 이 경우, 컨트롤러(300)와 소스 노드(302) 또는 수신지 노드(304) 사이에서는 트랜잭션이 필요없게 되고, 이것은 통신 프로시저를 단순화한다.

실시예에서, 컨트롤러(300), 소스 노드(302) 및 수신지 노드(304)가 독립 노드에서 따로따로 기능하는 경우를 설명한다. 예를 들면, 1394 인터페이스(14)가 마련된 컴퓨터(10)는 컨트롤러(300)로서 기능한다. 또한, 1394 인터페이스(44)가 마련된 DVCR(28)은 소스 노드(302)로서 기능하고, 1394 인터페이스(62)가 마련된 프린터(60)는 수신지 노드(304)로서 기능한다.

실시예에서, 도 3에 도시한 바와 같이, 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이에는 적어도 하나의 접속이 설정될 수 있다. 어떤 오브젝트 데이터의 전송 요구가 있을 때, 이들 접속은 다음

에 설명하는 통신 프로토콜에 따라 적어도 하나의 컨트롤러(300)에 의해 설정된다.

실시예에서, 하나의 접속에서 이용가능한 하나 이상의 수신지 오프셋이 설정될 수 있다. 수신지 오프셋의 값은 컨트롤러(300) 또는 소스 노드(302)에 의해 프리셋 또는 일정하지 않게 설정될 수 있다. 또한, 접속과 수신지 오프셋 사이의 관계는 다음에 설명하는 통신 프로토콜에 따라 설정된다.

다수의 수신지 오프셋이 하나의 접속에 있어서 설정될 때, 다수의 데이터 통신 모드가 하나의 접속에 의해 동시에 실현될 수 있다. 예를 들면, 상이한 수신지 오프셋을 그 데이터 통신 모드에 할당하는 것에 의해 하나의 접속으로 1대1, 1대N, N대N 데이터 통신이 동시에 실현될 수 있다.

또한, 실시예에서, 컨트롤러(300)로서의 컴퓨터(10)는 수신지 노드(304)로서 동작할 수도 있다. 이 경우, 하나의 소스 노드(302)와 2개의 수신지 노드(304)사이에서 접속이 설정되고, 오브젝트 데이터(308)가 전송된다.

또한, 실시예에서, 컴퓨터(10)가 컨트롤러(300)로서 기능하는 경우를 설명하였지만, 컴퓨터(10)는 반드시 컨트롤러(300)로서의 기능을 가질 필요가 없다. DVCR(28) 또는 프린터(60)가 컨트롤러(300)로서 동작할 수도 있다.

제1실시예

이하, 제1실시예에서 규정되는 통신 프로토콜의 기본 전송 프로시저를 설명한다.

도 4a 및 도 4c는 하나의 오브젝트 데이터가 전송될 때까지 실행되는 프로시저를 도시한 시퀀스도이다. 도 4b는 오브젝트 데이터의 전송중 버스 리셋 또는 전송 에러가 발생하는 경우의 전송 프로시저를 도시한 시퀀스도이다.

이 실시예의 통신 프로토콜에서는 상술한 접속이 컨트롤러(300)에 의해 설정된 후, 하나의 오브젝트 데이터가 적어도 하나의 비동기 방송 트랜잭션에서 전송된다. 이하, 비동기 방송 트랜잭션의 상세한 통신 프로시저를 도 4를 참조하여 설명한다. 또한, 비동기 방송 트랜잭션에서 사용되는 패킷(이하, 비동기 방송 패킷(Asynchronous broadcast packet)이라 함)을 도 5를 참조하여 설명한다.

또한, 비동기 방송 트랜잭션 및 비동기 방송 패킷은 이 실시예의 통신 프로토콜에서 규정되는 완전히 새로운 통신 프로시저 및 패킷 포맷이다.

이 실시예의 통신 프로토콜에 따른 기본 전송 프로시저를 도 4a 및 도 4c를 참조하여 설명한다. 여기서, 도 4a는 데이터 통신이 하나의 접속에서 하나의 수신지 노드(304)에 의해 실행되는 경우를 도시한 시퀀스도이다. 또한, 도 4c는 데이터 통신이 하나의 접속에서 3개의 수신지 노드(304)에 의해 실행되는 경우를 도시한 시퀀스도이다.

컨트롤러(300)는 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이의 논리적 접속 관계(접속)를 식별하기 위한 접속 ID를 설정한다. 이어서, 컨트롤러(300)는 각 노드에게 접속 ID를 통지하고, 하나의 접속을 설정한다(도 4a, 도 4c의 (401, 402)).

접속 ID의 통지후, 컨트롤러(300)는 소스 노드(302)에게 오브젝트 데이터(308)의 전송 개시를 명령한다(도 4a, 도 4c의 (403)).

트랜잭션 커맨드를 수신한 후, 소스 노드(302)는 적어도 하나의 수신지 노드(304)와 교섭을 실행하여, 비동기 방송 트랜잭션의 초기화를 실행한다(도 4a, 도 4c의 (404, 405)).

초기화 설정이 완료된 후, 소스 노드(302)는 하나이상의 세그먼트 데이터로 구성된 오브젝트 데이터를 연속해서 방송하기 위해 비동기 방송 트랜잭션을 실행한다(도 4a, 도 4c의 (406) 내지 (409)).

여기서, 이 실시예의 오브젝트 데이터의 전송 모델을 도 7을 참조하여 설명한다. 도 7에서, 오브젝트 데이터는 예를 들면, 128 킬로바이트의 데이터 크기를 갖는 정지 화상 데이터이다.

소스 노드(302)는 초기화 설정에 있어서 인식된 각 수신지 노드(304)의 수용 능력에 따라 오브젝트 데이터(308)를 예를 들면 500 조각의 세그먼트 데이터(한 조각의 세그먼트 데이터는 256 바이트에 대응함)로 분할한다. 여기서, 하나의 세그먼트 데이터의 크기는 각 수신지 노드(304)의 내부 버퍼의 크기에 따라 소스 노드(302)에 의해 여러 가지로 설정된다. 도 7은 크기가 오브젝트 데이터(308)와 동일한 내부 버퍼가 고정된 경우를 도시하고 있다.

또한, 소스 노드(302)는 적어도 하나의 비동기 방송 트랜잭션을 사용하여 하나이상의 세그먼트 데이터를 전송한다. 도 7에서, 하나의 세그먼트 데이터는 비동기 방송 트랜잭션을 사용하여 전송된다.

모든 세그먼트 데이터의 전송후, 소스 노드(302)는 적어도 하나의 수신지 노드(304)와의 데이터 통신을 완료한다(도 4a, 도 4c의 (410, 411)).

컨트롤러(300)의 동작을 도 4a 및 도 4c를 참조하여 설명한다.

컨트롤러(300)는 사용자에 의해 선택된 적어도 하나의 수신지 노드(304)와 소스 노드(302) 사이의 접속을 설정하기 위해 교섭을 실행한다. 이어서, 컨트롤러(300)는 노드 사이의 접속을 설정하기 위해 패킷(이하, 접속 설정 패킷이라 함)의 비동기 전송을 실행한다(도 4a, 도 4c의 (401, 402)).

이 경우, 각 수신지 노드(304)는 자신의 허용가능한 간격 시간(도 9에 도시한 데이터 전송 지연)을 컨트롤러(300)에게 통지한다. 컨트롤러(300)는 각 수신지 노드(304)의 간격 시간에 따라 소스 노드(302)가 각 비동기 방송 트랜잭션시 대기중인 시간의 최적 기간(도 9에 도시한 응답 기간)을 동적으로 결정한다. 예를 들면, 각 수신지 노드(304)의 간격 시간의 최대값이 결정된다. 이 기간은 접속 설정 패킷과 함께 소스 노드(302)에게 통지된다.

또한, 이 기간은 간격 시간보다 크게 설정된다. 또한, 간격 시간은 수신지 노드(304)의 수신 능력 및 성능에 의해 동적으로 변경한다. 따라서, 간격 시간은 수신 능력 및 성능이 높으면 단축되고, 낮으면 길어진다.

소스 노드(302)와 수신지 노드(304) 사이의 접속을 나타내는 접속 ID는 접속 설정 패킷의 페이로드(payload)에 저장된다. 각 노드는 접속 ID에 의해 그의 설정된 접속을 식별한다. 또한, 접속은 소스 노드(302)에 대하여 이미 설정된 접속 ID 및 각 수신지 노드(304)에 대하여 이미 설정된 접속 ID에 따라 컨트롤러(300)에 의해 설정된다.

이어서, 컨트롤러(300)는 소스 노드(302)로의 전송 커맨드 패킷(트랜잭션 커맨드 패킷)의 비동기 전송을 실행한다(도 4a, 도 4c의 (403)).

전송 커맨드 패킷 수신시, 소스 노드(302)는 비동기 방송 트랜잭션을 실행하기 위해 컨트롤러(300)에서 통지된 접속 ID를 사용하여 초기화 설정을 실행한다(도 4a, 도 4c의 (404) 내지 (409)). 비동기 방송 트랜잭션을 통하여, 소스 노드(302)는 하나 이상의 세그먼트 데이터로 구성된 오브젝트 데이터(308)를 연속해서 전송할 수 있다.

또한, 이 실시예의 통신 프로토콜에서, 컨트롤러(300)는 접속/비접속을 제어하는 기능을 제공한다. 따라서, 접속이 설정된 후, 오브젝트 데이터(308)는 소스 노드(302)와 수신지 노드(304) 사이의 교섭에 의해 전송된다.

일련의 비동기 방송 트랜잭션이 완료된 후, 소스 노드(302)는 세그먼트 끝을 나타내는 비동기 방송 패킷(이하, 세그먼트 종료 패킷이라 함)을 방송한다(도 4a, 도 4c의 (410)).

소스 노드(302)에서 세그먼트 종료 패킷을 수신한 후, 접속은 해제되고 데이터 전송은 완료된다(도 4a, 도 4c의 (411)).

여기서, 세그먼트 종료 패킷이 방송되므로, 그 패킷의 내용은 수신지 노드(304)에서도 검출될 수 있다. 따라서, 컨트롤러(300) 대신에, 수신지 노드(304)가 소스 노드(302)에서의 접속을 해제할 수도 있다.

이하, 소스 노드(302)의 동작을 도 4a 및 도 4c를 참조하여 상세히 설명한다.

컨트롤러(300)로부터 접속 설정 패킷 및 트랜잭션 커맨드 패킷을 수신한 소스 노드(302)는 데이터 전송을 요구하는 각 수신지 노드(304)에게 비동기 방송 패킷(이하, 송출 요구 패킷이라 함)을 송신한다(도 4a, 도 4c의 (404)).

여기서, 송출 요구 패킷은 오브젝트 데이터(308)의 비동기 방송 트랜잭션을 실행하기 전에 필요한 초기 정보를 얻는 요구 패킷을 의미한다. 컨트롤러(300)에 의해 지정된 접속 ID는 이 패킷에 기록된다.

수신지 노드(304)는 송출 요구 패킷에 대응하는 응답을 나타내는 비동기 방송 패킷(이하, 확인 응답 패킷(ack response packet)이라 함)을 방송한다(도 4a, 도 4c의 (405)). 여기서, 송출 요구 패킷에서와 동일한 접속 ID가 확인 응답 패킷에 저장된다. 따라서, 소스 노드(302)는 수신된 패킷의 접속 ID를 확인하는 것에 의해, 어느 접속을 거쳐 확인 응답 패킷이 전송되는가를 식별할 수 있다.

여기서, 각 수신지 노드(304)가 고정될 수 있는 내부 버퍼의 크기 및 소정의 메모리 공간을 지정하는 오프셋 어드레스는 확인 요구 패킷에 저장된다. 확인 요구 패킷을 수신한 후, 소스 노드(302)는 수신지 노드(304)에 대한 공통의 메모리 공간을 지정하는 수신지 오프셋을 설정하고, 비동기 방송 트랜잭션을 시작한다. 여기서, 수신지 오프셋은 각 수신지 노드(304)의 확인 요구 패킷에 포함된 오프셋 어드레스를 사용하여 설정된다.

또한, 이 실시예에서, 비동기 방송 트랜잭션에 사용되는 수신지 오프셋은 확인 요구 패킷에 포함된 오프셋 어드레스를 사용하여 설정되고, 이것은 제한되지 않는다. 예를 들면, 컨트롤러(300)에는 각 접속에 의해 사용되는 수신지 오프셋을 제어하는 기능이 마련될 수 있으므로, 수신지 오프셋은 접속 ID와 함께 설정된다. 이 경우, 각 접속에 대응하는 수신지 오프셋은 컨트롤러(300)에서 소스 노드(302)로 통지된다.

또한, 각 수신지 노드(304)는 확인 요구 패킷을 사용하여 간격 시간을 소스 노드에게 직접 통보할 수 있다. 이 경우, 컨트롤러(300)대신에, 소스 노드(302)가 각 비동기 방송 트랜잭션에서 대기하는 시간의 최적 기간을 소스 노드(302)가 동적으로 결정한다.

이어서, 소스 노드(302)는 수신지 오프셋에 의해 지시된 메모리 공간에 제1비동기 방송 패킷을 기록한다(도 4a, 도 4c의 (406)). 접속 ID 및 세그먼트 데이터의 시퀀스 번호가 이 패킷에 저장된다.

제1비동기 방송 패킷을 송신한 후, 소스 노드(302)는 수신지 노드(304)로 부터의 응답 패킷을 기다린다. 수신지 노드(304)는 접속 ID 및 시퀀스 번호가 저장된 응답 패킷을 비동기 방송 패킷의 형태로 송신한다. 응답 패킷을 수신한 후, 소스 노드(302)는 시퀀스 번호를 증가시키고, 다음 세그먼트 데이터를 포함하는 비동기 방송 패킷을 전송한다(도 4a, 도 4c의 (407)).

소스 노드(302)는 이 프로시저를 반복하여 비동기 방송 트랜잭션을 연속해서 실행한다(도 4a, 도 4c의 (408, 409)). 수신지 노드(304)로 부터의 응답을 기다리는 시간 기간은 간격 시간에 의해 결정된다. 이 시간 기간을 이 실시예에서 응답 기간이라고 한다.

예를 들면, i번째 세그먼트 데이터의 비동기 방송 트랜잭션 후 응답 기간이 경과한 후에도, 응답 패킷을 수신할 수 없다. 이 경우, 소스 노드(302)는 i번째 세그먼트 데이터와 동일한 비동기 방송 패킷을 재송출한다.

또한, 응답 패킷이 재송출을 요구하는 수신지 노드(304)에서 전송될 때, 소스 노드(302)는 지정된 시퀀

스 번호의 데이터를 다시 방송할 수 있다.

모든 오브젝트 데이터(308)의 비동기 방송 패킷 트랜잭션이 실행된 후, 소스 노드(302)는 세그먼트 종료 패킷을 방송하고, 데이터 전송을 완료한다(도 4a, 도 4c의 (410, 411)).

여기서, 상술한 바와 같이, 소스 노드(302)는 오브젝트 데이터(308)를 필요에 따라 하나 이상의 세그먼트 데이터로 분할한다. 각 세그먼트 데이터의 비동기 방송 트랜잭션에 있어서, 상술한 응답 패킷이 발생된다. 비동기 방송 트랜잭션을 한번 실행하는 것에 의해 하나의 세그먼트 데이터가 전송된다. 수신지 노드(304)는 버퍼 크기에 의해 지시된 부피의 버퍼를 갖는다.

또한, 이 실시예에서, 응답 패킷은 하나의 세그먼트 데이터의 비동기 트랜잭션에서 반드시 송출되고, 이것은 제한되지 않는다. 수신지 노드(304)의 데이터 버퍼가 다수의 연속 세그먼트 데이터로 채워진 후, 수신지 노드(304)는 응답 패킷을 송신할 수 있다. 구성에 있어서, 수신지 노드(304)에 의해 실행되는 응답 동작의 회수가 저감될 수 있으므로, 수신지 노드(304)의 구성이 단순화될 수 있고, 처리 속도가 향상될 수 있다.

이하, 수신지 노드(304)의 동작을 도 4a 및 도 4c를 참조하여 상세히 설명한다.

컨트롤러(300)에서 접속 설정 패킷을 수신한 수신지 노드(304)는 소스 노드(302)에서의 송출 요구 패킷을 기다린다(도 4a, 도 4c의 (404)).

송출 요구 패킷을 수신한 수신지 노드(304)는 그 패킷에 기록된 접속 ID 및 컨트롤러로부터 통지된 접속 ID를 확인하고, 패킷을 소스 노드(302)로부터 전송할지의 여부를 결정한다.

소스 노드(302)에서 송출 요구 패킷을 수신한 후, 각 수신지 노드(304)는 접속 ID, 고정될 수 있는 내부 버퍼의 크기 및 소정의 메모리 공간을 지정하는 오프셋 어드레스가 라이트된 확인 응답 패킷을 방송한다(도 4a, 도 4c의 (405)). 또한, 각 수신지 노드(304)는 송출 요구 패킷을 사용하는 것에 의해 간격 시간을 소스 노드(302)에게 직접 통지할 수도 있다.

소스 노드(302)로부터 전송된 비동기 방송 패킷이 메모리 공간에 기록된 후, 수신지 노드(304)는 패킷의 접속 ID를 확인한다. 패킷에 포함된 접속 ID가 수신지 노드(304) 자신의 접속 ID와 일치할 때, 접속 ID 및 시퀀스 번호가 저장된 응답 패킷이 방송된다(도 4a, 도 4c의 (406) 내지 (409)). 이 경우, 수신된 패킷에 포함된 세그먼트 데이터는 내부 버퍼에 저장된다. 여기서, 수신된 패킷에 포함된 접속 ID가 그의 접속 ID와 다를 때, 수신지 노드(304)는 수신한 패킷을 버린다.

또한, 수신지 노드(304)가 수신된 패킷의 시퀀스 번호의 불일치를 검출할 때, 응답 패킷이 재송출 요구를 위해 송출될 수도 있다. 이 경우, 수신지 노드(304)는 재송출 요구에 대한 시퀀스 번호를 지정하고, 소스 노드(302)에게 그 번호를 통지한다.

모든 비동기 방송 트랜잭션이 완료될 때, 세그먼트 종료 패킷이 소스 노드(302)에서 방송된다. 그 패킷을 수신할 때, 수신지 노드(304)는 데이터 전송 처리를 완료한다(도 4a, 도 4c의 (410)).

세그먼트 종료 패킷을 수신한 후, 수신지 노드(304)는 세그먼트 종료 패킷이 정상적으로 수신된 것을 나타내는 응답 패킷을 방송한다(도 4a, 도 4c의 (411)).

상술한 바와 같이, 이 실시예의 통신 시스템은 종래 통신 시스템의 불편을 해소할 수 있다. 또한, 실시간 특성을 요구하지 않는 데이터 전송에 있어서도, 데이터를 고속으로 용이하게 전송할 수 있다.

또한, 이 실시예에서는 컨트롤러(300)가 접속을 설정한 후, 오브젝트 데이터를 전송하는 처리가 컨트롤러(300)에 의한 제어없이 소스 노드(302)와 각 수신지 노드(304) 사이에서 실행된다. 따라서, 컨트롤러(300)의 부하를 저감하고 복잡한 통신 프로시저가 필요없는 단순한 통신 프로토콜이 마련될 수 있다.

또한, 이 실시예에서는 수신지 노드(304)가 각 비동기 방송 트랜잭션에 대한 응답을 확실히 송출한다. 따라서, 실시간 특성을 요구하지 않는 데이터가 확실히 전송될 수 있는 통신 프로토콜이 마련될 수 있다.

더욱 안전한 데이터 전송을 실현하기 위해, 데이터 전송이 버스 리셋 또는 어떤 전송 에러의 발생에 의해 인터럽트될 때, 어떠한 데이터의 빠짐도 없이 데이터 전송을 즉시 재개할 필요가 있다. 이하, 이 실시예의 통신 프로토콜에 규정된 재개 프로시저를 도 4b를 참조하여 설명한다.

예를 들면, 세그먼트 번호 1을 갖는 비동기 방송 패킷이 수신된 후 버스 리셋이 발생할 때, 각 노드는 전송 처리를 중단하고, IEEE 1394-1995 규격에 규정된 프로시저에 따라 버스 초기화, 접속 구성의 인식, 노드 ID의 설정 등을 실행한다(도 4b의 (420, 421)).

버스 재구성이 완료된 후, 각 수신지 노드(304)는 접속 ID 및 시퀀스 번호 i 가 저장된 재송출 요구 패킷을 방송한다(도 4b의 (422)).

비동기 방송 트랜잭션을 재개할 수 있을 때, 소스 노드(302)는 수신된 재송출 요구 패킷의 접속 ID를 확인하여 접속 ID가 저장된 확인 응답 패킷을 방송한다(도 4b의 (423)).

이어서, 소스 노드(302)는 수신된 재송출 요구 패킷에 의해 요구된 시퀀스 번호와 그 다음 번호의 세그먼트 데이터, 즉 시퀀스 번호 $i+1$ 로 시작하는 시퀀스 데이터를 연속해서 방송한다(도 4b의 (424)).

상술한 프로시저에서, 데이터 전송이 인터럽트되더라도, 컨트롤러(300), 소스 노드(302) 및 수신지 노드(304)는 각 노드 ID를 고려하지 않고 다음 데이터 전송을 용이하고 확실하게 재개할 수 있다.

또한, 상술한 바와 같이, 이 실시예에서는 데이터 전송이 인터럽트될 때에도, 컨트롤러(300)의 제어 프

로시저가 효과적으로 단순화될 수 있다.

이하, 이 실시예에서 규정된 비동기 방송 패킷의 구성을 도 5를 참조하여 설명한다. 비동기 방송 패킷은 예를 들면 1 쿼드렛(Quadlet)(4 바이트 = 32 비트)의 단위를 갖는 데이터 패킷이다.

먼저, 패킷 헤더(521)의 구성을 설명한다.

도 5에서, 필드(501)(16 비트)는 destination_ID 및 수신지의 노드 ID(즉, 수신지 노드304)를 나타낸다. 이 실시예의 통신 프로토콜에서, 오브젝트 데이터(308)의 비동기 방송 트랜잭션을 실현하기 위해, 이 필드의 값은 방송 ID(즉, FFFF₁₆)로서 설정된다.

필드(502)(6 비트)는 트랜잭션 라벨(t1) 필드, 또는 각 트랜잭션 고유의 태그를 나타낸다.

필드(503)(2 비트)는 재시도(rt) 코드를 나타내고, 패킷이 재시도되는 가의 여부를 지정한다.

필드(504)(4 비트)는 실행해야 할 패킷 포맷 또는 트랜잭션 종류를 지정하는 트랜잭션 코드(tc)를 나타낸다. 이 실시예에서, 이 필드의 값은 예를 들면 0001₂로 설정되어 destination_offset 필드(507)에 의해 지시된 메모리 공간에 패킷의 데이터 블록(522)을 기록하는 처리가 요구된다(즉, 기록 트랜잭션).

필드(505)(4 비트)는 우선권(pri)을 나타내고, 우선권의 순위를 지정한다. 이 실시예에서, 이 필드의 값은 0000₂로 설정된다.

필드(506)(16 비트)는 source_ID 또는 송신측(즉, 소스 노드(302))의 노드 ID를 나타낸다.

필드(507)(48 비트)는 destination_offset를 나타내고, 각 수신지 노드(304)의 어드레스 공간의 하위 48비트를 공통으로 지정한다. 여기서, destination_offset에 대하여, 동일한 값이 모든 접속에 있어서 설정될 수도 있거나 또는 그 접속에 있어서 상이한 값이 설정될 수도 있다. 그러나, 상이한 값이 설정될 때, 다수의 접속으로 부터의 비동기 방송 패킷은 병렬로 효율적으로 처리될 수 있다.

필드(508)(16 비트)는 data_length를 나타내고, 바이트 단위로 다음에 설명하는 데이터 필드의 길이를 나타낸다.

필드(509)(16 비트)는 extended_tc)를 나타낸다. 이 실시예에서 이 필드의 값은 0000₁₆으로 설정된다.

필드(510)(32 비트)는 필드(501) 내지 (509)에 대한 에러 검출 코드가 저장된 header_CRC를 나타낸다.

이하, 데이터 블록(522)의 구성을 설명한다. 이 실시예에서 데이터 블록(522)은 헤더 정보(523) 및 데이터 필드(524)로 구성된다.

노드사이의 논리적 접속 관계를 식별하는 접속 ID 등은 헤더 정보(523)에 저장된다.

또한, 데이터 필드(524)는 세그먼트 데이터가 저장되는 가변 길이를 갖는다. 여기서, 데이터 필드(524)에 저장된 세그먼트 데이터가 쿼드렛의 배수가 아닐 때, 쿼드렛을 만족하지 않는 부분은 제로로 채워진다.

필드(511)(16 비트)는 connection_ID를 나타내고, 이 실시예의 접속 ID를 저장한다. 이 실시예의 1394 인터페이스는 이 필드에 저장된 접속 ID에 따라 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이의 접속 세트를 식별한다. 이 실시예에서는 2^{16} × 노드수의 접속이 확립될 수 있다. 따라서, 각 접속에 의해 사용되는 통신 대역의 총량이 전송 경로의 용량에 도달할 때까지 다수의 접속이 설정될 수 있다.

필드(512)(8 비트)는 protocol_type를 나타내고, 헤더 정보(523)(즉, 통신 프로토콜 종류)에 따른 통신 프로시저가 지시된다. 이 실시예의 통신 프로토콜이 지시될 때, 이 필드의 값은 예를 들면, 01₁₆이다.

필드(513)(8 비트)는 control_flags를 나타내고, 이 실시예의 통신 프로토콜의 통신 프로시저 등을 제어하는 소정의 제어 데이터가 설정된다. 이 실시예에서, 이 필드의 최상위 비트는 예를 들면, resend_request 플래그로서 설정된다. 따라서, 이 필드의 최상위 비트가 1의 값을 가질 때, 이 실시예의 통신 프로토콜에 따른 재송출 요구가 발생된 것을 나타낸다.

필드(514)(16 비트)는 sequence_number를 나타내고, 연속하는 값(즉, 시퀀스 번호)이 지정된 접속 ID(필드(511)에 의해 지정된 접속 ID)에 따라 전송된 패킷에 대하여 설정된다. 수신지 노드(304)는 시퀀스 번호에 의해 비동기 방송 트랜잭션이 연속해서 가해진 세그먼트 데이터의 연속성을 감시할 수 있다. 고르지 못함(inequality)이 발생할 때, 수신지 노드(304)는 시퀀스 번호에 따라 재송출을 요구할 수 있다.

필드(515)(16 비트)는 reconfirmation_number를 나타낸다. 이 실시예에서, 이 필드는 재송출 요구 플래그가 1의 값을 가질때만 의미를 갖는다. 예를 들면, 재송출 요구 플래그의 값이 1일 때, 재송출을 요구하는 패킷의 시퀀스 번호는 이 필드에 설정된다.

필드(516)(16 비트)는 buffer_size를 나타낸다. 수신지 노드(304)의 버퍼 크기가 이 필드에 설정된다.

필드(517)(48 비트)는 offset_address를 나타낸다. 수신지 노드(304)의 어드레스 공간의 하위 48비트가 이 필드에 저장된다. 따라서, 도 3에 도시한 첫 번째 메모리 공간(310) 내지 n번째 메모리 공간(314)중 어느 하나가 지정된다.

필드(518)(32 비트)는 destination_interval을 나타낸다. 간격 시간이 이 필드에 저장된다. 각 수신지 노드(304)는 소스 노드(302) 및 컨트롤러(300)에게 이 필드에 의한 간격 시간을 통지한다.

필드(519)(32 비트)는 data_CRC를 나타내고, (헤더 정보(523) 및 데이터 필드(524)를 포함하여) 필드(511) 내지 (518)에 대한 에러 검출 코드가 header_CRC와 동일한 방식으로 저장된다.

이하, 제1실시예의 통신 프로토콜을 도 9를 참조하여 상세히 설명한다.

도 9에서, 특히, i 번째(i 는 임의의 정수) 비동기 방송 트랜잭션이 실행될때와 $(i+1)$ 번째 비동기 방송 트랜잭션이 실행될 때 사이의 시간 간격을 상세히 설명한다. 또한, 도 9에서는 설명을 간략화하기 위해, 하나의 소스 노드(302)와 하나의 수신지 노드(304)사이에서 전송을 도시하고 있지만, 하나 이상의 수신지 노드(304)가 마련될 때에도 동일한 처리가 실행될 수 있다.

도 9는 수신지 노드(304)의 내부 버퍼(252), 내부 버퍼(252)의 데이터를 처리하는 다음단 회로(254), i 번째 비동기 방송 패킷(256), i 번째 비동기 방송 패킷에 대응하는 응답 패킷(258), $(i+1)$ 번째 비동기 방송 패킷(260), $(i+1)$ 번째 비동기 방송 패킷에 대응하는 응답 패킷(262), 응답 기간(267), 수신지 노드(304) 내부에서의 (내부 버퍼(252)에서 다음단 회로(254)로의) 데이터 이동(266), 및 수신지 노드(304)내부의 데이터 이동의 지연 시간(268)을 도시한다.

i 번째 비동기 방송 트랜잭션이 시작될 때, i 번째 비동기 방송 패킷(256)이 소스 노드(302)로부터 전송된다. 수신지 노드(304)가 그 패킷에 포함된 세그먼트 데이터를 내부 버퍼(252)에 소정의 메모리 공간을 거쳐 일시적으로 저장하고 나서, 세그먼트 데이터는 다음단 회로(254)로 이동한다.

세그먼트 데이터의 이동이 완료될 때, 수신지 노드(304)는 완료를 나타내는 응답 패킷(258)을 준비하고, 그 패킷을 소스 노드(302)로 전송한다. 이 경우, 수신지 노드(304)의 성능에 의존하는 지연 시간(268)은 수신지 노드(304) 내부의 데이터 이동(266)과 함께 응답 패킷(258)의 전송에 있어서 발생된다.

수신지 노드(304)의 예상 지연 시간(268)은 사전에 컨트롤러(300)에서 소스 노드(302)로 간격 시간으로서 통지된다. 소스 노드(302)는 그 간격 시간에 따라 응답 기간(264)을 결정한다. 이 시간동안, 소스 노드(302)는 수신지 노드(304)로부터의 응답 패킷을 기다리고, 다음 비동기 방송 트랜잭션을 실행하지 않는다. 여기서, 응답 기간(264)은 통상 간격 시간보다 길게 설정된다.

$(i+1)$ 번째 비동기 방송 트랜잭션에 대하여, i 번째 비동기 방송 트랜잭션과 동일한 프로시저로 처리가 실행된다. 이어서, 각 비동기 방송 트랜잭션이 동일한 방식으로 처리되므로, 모든 비동기 방송 패킷이 재송출없이 확실히 전송된다.

상술한 구성에 있어서, 접속 설정시, 제1실시예의 컨트롤러(300)(또는 소스 노드(302))는 각 수신지 노드(304)의 수신 능력 및 성능에 따라 최적의 응답 기간을 동적으로 설정할 수 있다. 예를 들면, 수신지 노드(304)의 수신 능력 및 성능이 높을때는 그 기간이 짧게 설정되고, 그들이 낮을 때에는 그 기간이 길게 설정된다.

따라서, 각 수신지 노드(304)의 성능이 매우 높지 않을 때에도, 소스 노드(302)는 재송출 처리를 빈번히 발생하는 일없이 각각의 비동기 방송 트랜잭션을 확실히 실행할 수 있다. 또한, 네트워크의 전송 효율이 저하되는 것이 방지된다. 또한, 수신지 노드(304)가 고속 고기능 수신 능력을 요구하지 않으므로, 수신지 노드(304)의 기능을 실현하기 위해 필요한 비용을 저감할 수 있다.

제2실시예

이하, 제2실시예의 통신 프로토콜을 도 10 내지 도 16을 참조하여 설명한다. 제2실시예에서는 특히 재시도가 불필요하게 발생하는 것을 방지하는 프로시저를 상세히 설명한다. 재시도는 i 번째 데이터가 정상적으로 수신되지 않을 때 소정의 시간동안만 금지된다.

도 10은 이 실시예에서 규정된 단일 단계 재시도 기능(single phase retry function)이 마련된 소스 노드(302)의 재시도 프로시저를 도시한 상태 전이도이다.

소스 노드(302)는 그의 노드 컨트롤러로 부터의 초기화 또는 리셋에 대한 트랜잭션 제어 요구에 응답하여 다른 상태에서 OSR0 상태로 이동한다.(도 10의 (1001)) 이 상태에서, 소스 노드(302)는 각 수신지 노드(304)로 소정의 패킷을 송신할 준비를 한다. 따라서, 이 상태에서, ack_busy_A, ack_busy_X이외의 확인 응답 코드를 나타내는 응답 패킷이 수신지 노드(304)에서 수신될 때, 소스 노드(302)는 다음 패킷을 전송할 수 있어, 재시도를 실행할 필요가 없다(도 10의 (1006)).

OSR0 상태에서, 소스 노드(302)가 ack_busy_A, ack_busy_B 또는 ack_busy_X를 나타내는 응답 패킷을 수신할 때, 소스 노드(302)는 수신지 노드(304)가 사용중인 것을 인식한다. 이 경우, 소스 노드(302)는 OSR0에서 OSR1로 상태를 변경한다(도 10의 (1002)).

OSR1 상태에서, 처리해야할 계류중인 재시도가 있을 때, 소스 노드(302)는 다른 임의의 패킷을 전송하기 전에 재시도를 처리한다. 이 경우, 소스 노드(302)는 재시도를 실행하기 위해 재시도 코드를 retry_X로 지정한다.

OSR1 상태에서, 소스 노드(302)가 재시도 한계를 초과하지 않고, 계류중인 재시도 패킷이 재시도 큐에 없을 때, 소스 노드(302)는 다시 재시도를 반복한다(도 10의 (1007)). 여기서, 재시도 큐는 재시도 패킷의 큐를 의미한다.

또한, OSR1 상태에서, 최소 재시도 기간이 소스 노드(302)에 설정된다. 소스 노드(302)는 최소 재시도 기간이 경과할 때까지 재시도를 실행하지 않고 OSR1 상태에 머무른다(도 10의 (1008)).

OSR1 상태에서, 재시도 한계가 초과되지 않고, 재시도 패킷이 재시도 큐에 있을 때, 소스 노드(302)는 OSR0 상태에서 OSR1 상태로 이동한다(도 10의 (1005)).

또한, OSR1 상태에서, 응답 패킷이 ack_busy_A, ack_busy_B 및 ack_busy_X이외의 확인 응답 코드를 나타낼 때, 재시도가 처리되었다고 판정하고, 소스 노드(302)는 OSR1 상태에서 OSR0 상태로 변한다(도 10의 (1003)).

또한, OSR1 상태에서, 소스 노드(302)가 재시도 한계가 초과될때까지 어떤 패킷의 재시도를 실행할 때,

재시도가 실패했다고 판정되고, 소스 노드(302)는 OSR1 상태에서 OSR0 상태로 변한다(도 10의 (1004)).

도 11은 이 실시예에서 규정된 2중 단계 재시도 기능이 마련된 소스 노드(302)의 재시도 프로시저를 도시한 상태 천이도이다. 또한, 2중 단계 재시도 기능을 지원하는 소스 노드(302)는 단일 단계 재시도 기능도 지원한다.

소스 노드(302)는 그의 노드 컨트롤러로 부터의 초기화 또는 리셋에 대한 트랜잭션 제어 요구에 응답하여 다른 상태에서 ODR0 상태로 변한다(도 1의 (1101)). 이 상태에서, 소스 노드(302)는 각 수신지 노드(304)로 소정의 패킷을 전송할 준비를 한다. 소스 노드(302)는 재시도 코드를 `retry_1`로 설정하고 대기한다.

ODR0 상태에서, `ack_busy_A`, `ack_busy_B` 및 `ack_busy_X`이외의 응답 패킷이 각 수신지 노드(304)에서 수신될 때, 소스 노드(302)는 다음 패킷을 전송할 수 있고 재시도를 실행할 필요가 없다(도 11의 (1107)).

ODR0 상태에서, `ack_busy_A` 또는 `ack_busy_B`의 응답 패킷이 어떤 수신지 노드(304)에서 수신될 때, 소스 노드(302)는 수신지 노드(304)가 2중 단계 재시도 기능을 지원하는 노드이고 사용중이라고 판정한다. 이 경우, 소스 노드(302)는 2중 단계 재시도 기능에 따라 재시도를 실행한다.

`ack_busy_A`를 나타내는 응답 패킷이 수신될 때, 소스 노드(302)는 재시도 단계 A를 실행하고, ODR0 상태에서 ODR1 상태로 변한다(도 11의 (1102)). 또한, `ack_busy_B`를 나타내는 응답 패킷이 수신될 때, 소스 노드(302)는 재시도 단계 B를 실행하고, ODR0 상태에서 ODR2 상태로 변한다(도 11의 (1108)).

또한, ODR0 상태에서, `ack_busy_X`의 응답 패킷이 어떤 수신지 노드(304)에서 수신될 때, 소스 노드(302)는 수신지 노드(304)가 단일 단계 재시도 기능을 지원하는 노드이고 사용중이라고 판정한다. 이 경우, 소스 노드(302)는 단일 단계 재시도 기능에 따라 재시도를 실행한다. `ack_busy_X`를 나타내는 응답 패킷이 수신될 때, 소스 노드(302)는 ODR0 상태에서 ODR3 상태로 변한다(도 11의 (1113)).

ODR1 상태에서, 소스 노드(302)는 재시도 단계 A를 실행하고 있어, 해소해야 할 계류중인 재시도를 갖는다. 이 상태에서, 최소 재시도 기간이 소스 노드(302)에 설정된다. 소스 노드(302)는 최소 재시도 기간이 경과할때까지 재시도를 실행하지 않고 ODR1 상태에 머문다(도 11의 (1106)).

ODR1 상태에서, 소스 노드(302)는 재시도 코드를 `retry_A`로 설정하고, 재시도를 실행한다. 이 경우, 수신지 노드(304)로 부터의 응답 패킷이 `ack_busy_A`를 나타내고, 4개의 공평한 간격 타임아웃 기간(fairness interval timeout period)이 경과하지 않을 때, 소스 노드(302)는 다시 재시도를 반복한다(도 11의 (1105)). 여기서, 공평한 간격 타임아웃 기간은 비동기 전송이 가능한 기간으로 설정된 기간을 의미하고, 이것은 네트워크를 사용할 노드에서 공정한 액세스 권리를 준다.

또한, ODR1 상태에서, `ack_busy_A` 및 `ack_busy_B`이외의 확인 응답 코드를 포함하는 응답 재시도 코드가 수신될 때, 재시도가 처리되었다고 판정하고, 소스 노드(302)는 ODR1 상태에서 ODR0 상태로 변한다(도 11의 (1103)).

또한, ODR1 상태에서, 4개의 공평한 간격 타임아웃 기간이 경과할 때, 재시도가 실패했다고 판정하고, 소스 노드(302)는 ODR1 상태에서 ODR0 상태로 변한다(도 11의 (1104)).

ODR2 상태에서는 소스 노드(302)가 재시도 단계 B를 실행하고 있어, 해소해야 할 계류중인 재시도를 갖는다. 이 상태에서, 최소 재시도 기간이 소스 노드(302)에 설정된다. 소스 노드(302)는 최소 재시도 기간이 경과할때까지 재시도를 실행하지 않고 ODR2 상태에 머문다(도 11의 (1112)).

ODR2 상태에서, 소스 노드(302)는 재시도 코드를 `retry_B`로 설정하고, 재시도를 실행한다. 이 경우, 수신지 노드(304)로 부터의 응답 패킷이 `ack_busy_B`를 나타내고 4개의 공평한 간격 타임아웃 기간이 경과하지 않을 때, 소스 노드(302)는 다시 재시도를 반복한다(도 11의 (1111)).

또한, ODR2 상태에서, `ack_busy_A` 및 `ack_busy_B`이외의 확인 응답 코드를 포함하는 응답 재시도 코드가 수신될 때, 재시도가 처리되었다고 판정하고, 소스 노드(302)는 ODR2 상태에서 ODR1 상태로 변한다(도 11의 (1109)).

또한, ODR2 상태에서, 4개의 공평한 간격 타임아웃 기간이 경과할 때, 재시도가 실패했다고 판정하고, 소스 노드(302)는 ODR2 상태에서 ODR0 상태로 변한다(도 11의 (1110)).

ODR3 상태에서, 소스 노드(302)는 단일 단계 재시도를 실행하고 있어, 해소해야 할 계류중인 재시도를 갖는다. 이 상태에서, 최소 재시도 기간이 소스 노드(302)에 설정된다. 소스 노드(302)는 최소 재시도 기간이 경과할 때까지 재시도를 실행하지 않고 ODR3 상태에 머문다(도 11의 (1117)).

ODR3 상태에서, 소스 노드(302)는 재시도 코드를 `retry_X`로 설정하고, 재시도를 실행한다. 이 경우, 수신지 노드(304)로 부터의 응답 패킷이 `ack_busy_X`를 나타내고 재시도 한계가 초과되지 않을 때, 소스 노드(302)는 다시 재시도를 반복한다(도 11의 (1116)).

또한, ODR3 상태에서, `ack_busy_X`이외의 확인 응답 코드를 포함하는 응답 재시도 코드가 수신될 때, 재시도가 처리되었다고 판정하고, 소스 노드(302)는 ODR3 상태에서 ODR0 상태로 변한다(도 11의 (1114)).

또한, ODR3 상태에서, 재시도 계수가 초과될 때, 재시도가 실패했다고 판정하고 소스 노드(302)는 ODR3 상태에서 ODR0 상태로 변한다(도 11의 (1115)).

상술한 바와 같이, 이 구성은 소정의 재시도 기간을 가지므로, 이 실시예의 소스 노드(302)는 더욱 확실한 통신을 보장할 수 있다. 또한, 버스가 혼잡할 때에도, 수신지 노드(304)의 사용중 상태가 빈번히 발생하는 것이 방지되어, 대드락(deadlock)을 방지할 수 있다.

이하, 제2실시예의 통신 프로토콜에 따른 전송 프로시저를 도 12를 참조하여 설명한다. 또한, 제2실시

예의 통신 프로토콜은 기본적으로 제1실시예의 통신 프로토콜과 동일한 방식으로 처리된다. 따라서, 도 4에서와 동일한 처리를 실행하는 프로시저는 동일한 코드로 나타내고, 그의 반복 설명은 생략한다. 또한, 도 12에서, 설명을 간략화하기 위해, 하나의 소스 노드(302)와 하나의 수신지 노드(304) 사이의 전송을 도시하지만, 하나이상의 수신지 노드(304)가 마련될 때에도 동일한 처리를 실행할 수 있다.

컨트롤러(300)는 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이의 논리적 접속 관계를 식별하는 접속 ID를 설정한다. 이어서, 컨트롤러(300)는 접속 ID를 각 노드에게 통지하여, 하나의 접속을 설정한다(도 12의 (401, 402)).

접속 ID를 통지한 후, 컨트롤러(300)는 소스 노드(302)에게 오브젝트 데이터(308)의 전송 개시를 명령한다(도 12의 (403)).

트랜잭션 커맨드를 수신한 후, 소스 노드(302)는 초기 비동기 방송 트랜잭션을 설정하기 위해 적어도 하나의 수신지 노드(304)와의 교섭을 실행한다(도 12의 (404, 405)).

초기화를 완료한 후, 소스 노드(302)는 하나이상의 세그먼트 데이터로 구성된 오브젝트 데이터(308)를 연속해서 방송하기 위해 비동기 방송 트랜잭션을 실행한다(도 12의 (406) 내지 (409)).

여기서, 소스 노드(302)는 제1실시예와 동일한 방식으로 적어도 한번 비동기 방송 트랜잭션을 실행하는 것에 의해 하나이상의 세그먼트 데이터를 전송한다. 오브젝트 데이터(308)는 예를 들면 도 7에 도시한 바와 같은 다수의 세그먼트로 분할되고, 하나의 세그먼트 데이터는 비동기 방송 트랜잭션을 한번 사용하는 것에 의해 전송된다.

모든 세그먼트 데이터가 전송된 후, 소스 노드(302)는 하나이상의 수신지 노드(304)와의 데이터 통신을 완료한다(도 12의 (410, 411)).

이하, 컨트롤러(300)의 동작을 도 12를 참조하여 상세히 설명한다.

컨트롤러(300)는 사용자에 의해 선택된 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이의 접속을 설정하기 위해 교섭을 실행한다. 이어서, 컨트롤러(300)는 노드 사이의 접속을 설정하기 위해 패킷(이하, 접속 설정 패킷이라 함)의 비동기 전송을 실행한다(도 12의 (401, 402)).

여기서, 컨트롤러(300)는 각 접속에 의해 사용되는 수신지 오프셋을 제어하는 기능을 갖는다. 컨트롤러(300)는 각 수신지 노드(304)에서 통지된 오프셋 어드레스를 사용하여 각 수신지 노드(304)의 메모리 공간을 공통으로 지정하기 위한 수신지 오프셋을 설정한다. 또한, 어떤 접속에 대응하는 수신지 오프셋은 소정의 프로시저로 설정된다. 설정후, 수신지 오프셋은 컨트롤러(300)에서 소스 노드(302)로 통지된다.

소스 노드(302)와 수신지 노드(304) 사이의 접속을 나타내는 접속 ID는 접속 설정 패킷의 페이로드에 저장된다. 각 노드는 접속 ID에 의해 그 자신에 설정된 접속을 식별한다. 또한, 접속 ID는 소스 노드(302)에 이미 설정된 접속 ID 및 각 수신지 노드(304)에 이미 설정된 접속 ID에 따라 컨트롤러(300)에 의해 설정된다.

이어서, 컨트롤러(300)는 소스 노드(302)에 대한 트랜잭션 커맨드 패킷의 비동기 전송을 실행한다(도 12의 (403)).

트랜잭션 커맨드 패킷을 수신할 때, 소스 노드(302)는 컨트롤러(300)로부터 통지된 접속 ID를 사용하여 초기화를 실행하는 것에 의해 비동기 방송 트랜잭션을 실행한다(도 12의 (404) 내지 (409)). 비동기 방송 트랜잭션을 거쳐, 소스 노드(302)는 하나이상의 세그먼트 데이터로 구성된 오브젝트 데이터(308)를 연속해서 전송할 수 있다.

또한, 이 실시예의 통신 프로토콜에 있어서, 컨트롤러(300)는 접속/비접속을 제어하는 기능을 제공한다. 따라서, 접속이 설정된 후, 오브젝트 데이터(308)는 소스 노드(302)와 수신지 노드(304) 사이의 교섭에 의해 전송된다.

일련의 비동기 방송 트랜잭션이 완료된 후, 소스 노드(302)는 세그먼트 종료를 나타내는 비동기 방송 패킷(이하, 세그먼트 종료 패킷이라 함)을 방송한다(도 12의 (410)).

소스 노드(302)로부터 세그먼트 종료 패킷을 수신한 후, 컨트롤러(300)는 접속을 해제하여 데이터 전송을 완료한다(도 12의 (411)).

여기서, 세그먼트 종료 패킷이 방송되므로, 수신지 노드(304)에서도 패킷의 내용을 검출할 수 있다. 따라서, 컨트롤러(300) 대신에, 수신지 노드(304) 자신이 소스 노드(302)로부터의 접속을 해제할 수도 있다.

이하, 소스 노드(302)의 동작을 도 12를 참조하여 상세히 설명한다.

컨트롤러(300)로부터 접속 설정 패킷 및 트랜잭션 커맨드 패킷을 수신한 소스 노드(302)는 데이터 전송을 요구하는 각 수신지 노드(304)로 비동기 방송 패킷(이하, 송출 요구 패킷이라 함)을 송신한다.

여기서, 송출 요구 패킷은 오브젝트 데이터(308)의 비동기 방송 트랜잭션을 실행하기 전에 필요한 초기 정보를 얻는 요구 패킷을 의미한다. 컨트롤러(300)에 의해 지정된 접속 ID는 이 패킷에 기록된다.

수신지 노드(304)는 송출 요구 패킷에 대응하는 응답을 나타내는 비동기 방송 패킷(이하, 수락 응답 패킷(accepted response packet)이라 함)을 방송한다(도 12의 (405)). 여기서, 송출 요구 패킷에서와 동일한 접속 ID가 수락 응답 패킷에 저장된다. 따라서, 소스 노드(302)는 수신된 패킷의 접속 ID를 확인하는 것에 의해, 어느 접속을 거쳐 수락 응답 패킷이 전송되는가를 식별할 수 있다.

여기서, 각 수신지 노드(304)가 확보될 수 있는 내부 버퍼의 크기는 수락 응답 패킷에 저장된다. 수락

응답 패킷이 수신된 후, 소스 노드(302)는 수신지 노드(304)를 위한 공통의 메모리 공간을 지정하기 위한 수신지 오프셋을 설정하고, 비동기 방송 트랜잭션을 시작한다. 여기서, 수신지 오프셋은 컨트롤러(300)에 의해 통지된다.

이어서, 소스 노드(302)는 수신지 오프셋에 의해 지시된 메모리 공간에 제1비동기 방송 패킷을 기록한다(도 12의 (406)). 접속 ID 및 세그먼트 데이터의 시퀀스 번호는 이 패킷에 저장된다.

제1비동기 방송 패킷을 송신한 후, 소스 노드(302)는 각 수신지 노드(304)로 부터의 확인 응답 패킷을 기다린다(도 12의 (1201)). 각 수신지 노드(304)로부터 전송되는 확인 응답 패킷은 도 13에 도시한 바와 같이 구성된다. 여기서, 수신지 노드(304)가 사용중일 때, 소스 노드(302)는 2종류의 재시도를 실행하므로, 더욱 확실한 데이터 전송을 보장할 수 있고, 데드락을 방지할 수 있다.

확인 응답 패킷을 수신한 후, 소스 노드(302)는 시퀀스 번호를 증가시키고, 다음 세그먼트 데이터를 구비하는 비동기 방송 패킷을 전송한다(도 12의 (407)).

소스 노드(302)는 이 프로시저를 반복하여 비동기 방송 트랜잭션을 연속해서 실행한다(도 12의 (408, 409)). 수신지 노드(304)로 부터의 확인 응답 패킷을 기다리는 시간의 최대 기간은 사전에 결정된다. 이 기간은 응답 기간이라고 한다. 응답 기간은 제1실시예와 동일한 방식으로 설정된다. i번째 세그먼트 데이터의 비동기 방송 트랜잭션후 응답 기간이 경과한 후에도, 확인 응답 패킷이 반환되지 않는다. 이 경우, 소스 노드(302)는 그 세그먼트 데이터와 동일한 데이터를 재송출한다.

또한, 응답 패킷이 재송출을 요구하는 수신지 노드(304)로부터 전송될 때, 소스 노드(302)는 지정된 시퀀스 번호의 데이터를 다시 방송할 수 있다.

모든 오브젝트 데이터(308)의 비동기 방송 패킷 트랜잭션이 실행된 후, 소스 노드(302)는 세그먼트 종료 패킷을 방송하고, 데이터 전송을 완료한다(도 12의 (410, 411)).

여기서, 상술한 바와 같이, 소스 노드(302)는 오브젝트 데이터(308)를 필요에 따라 하나 이상의 세그먼트 데이터로 분할한다. 각 세그먼트 데이터의 비동기 방송 트랜잭션에 있어서, 상술한 응답 패킷이 발생된다. 비동기 방송 트랜잭션을 한번 실행하는 것에 의해 하나의 세그먼트 데이터가 전송된다. 수신지 노드(304)는 버퍼 크기에 의해 지시된 버퍼의 버퍼를 갖는다.

또한, 이 실시예에서, 확인 응답 패킷은 이것에 제한되지 않지만 하나의 세그먼트 데이터의 비동기 방송 트랜잭션에 있어서 반드시 송출된다. 수신지 노드(304)의 내부 버퍼가 다수의 연속하는 세그먼트 데이터로 채워진 후, 수신지 노드(304)는 확인 응답 패킷을 송신할 수도 있다. 이러한 구성에서는 수신지 노드(304)에 의해 실행되는 응답 동작의 회수를 저감할 수 있으므로, 수신지 노드(304)의 구성이 단순화될 수 있고, 처리 속도가 향상될 수 있다.

이하, 수신지 노드(304)의 동작을 도 12를 참조하여 상세히 설명한다.

컨트롤러(300)로부터 접속 설정 패킷을 수신한 수신지 노드(304)는 소스 노드(302)로 부터의 송출 요구 패킷을 기다린다(도 12의 (404)).

송출 요구 패킷을 수신한 수신지 노드(304)는 그 패킷에 기록된 접속 ID 및 컨트롤러로부터 통지된 접속 ID를 확인하고, 그 패킷이 소스 노드(302)로부터 전송된 것인지 여부를 결정한다.

소스 노드(302)로부터 송출 요구 패킷을 수신한 후, 각 수신지 노드(304)는 접속 ID 및 내부 버퍼의 확보 사이즈가 기록된 수락 응답 패킷을 방송한다(도 12의 (405)).

소스 노드(302)로 부터 전송된 비동기 방송 패킷이 메모리 공간에 기록된 후, 수신지 노드(304)는 그 패킷의 접속 ID를 확인한다. 그 패킷에 포함된 접속 ID가 수신지 노드(304) 자신의 접속 ID와 일치할 때, 확인 응답 패킷이 방송된다(도 12의 (406) 내지 (409)). 이 경우, 수신된 패킷에 포함된 세그먼트 데이터는 내부 버퍼에 저장된다. 여기서, 수신된 패킷에 포함된 접속 ID가 그의 접속 ID와 다를 때, 수신지 노드(304)는 수신된 패킷을 버린다.

또한, 수신지 노드(304)는 수신된 패킷의 시퀀스 번호의 불일치를 검출하고, 재송출을 요구하기 위해 패킷을 송출할 수도 있다. 이 경우, 수신지 노드(304)는 재송출 요구에 대한 시퀀스 번호를 지정하고, 그 번호를 소스 노드(302)에게 통지한다.

모든 비동기 방송 트랜잭션이 완료될 때, 세그먼트 종료 패킷이 소스 노드(302)에서 방송된다. 그 패킷을 수신할 때, 수신지 노드(304)는 데이터 전송 처리를 완료한다(도 12의 (410)).

세그먼트 종료 패킷을 수신한 후, 수신지 노드(304)는 세그먼트 종료 패킷이 정상적으로 수신된 것을 나타내는 패킷을 방송한다(도 12의 (411)).

상술한 바와 같이, 이 실시예의 통신 시스템은 종래 통신 시스템의 불편함을 해소할 수 있다. 또한, 실시간 특성을 요구하지 않는 데이터 전송에 있어서도, 데이터를 고속으로 용이하게 전송할 수 있다.

또한, 이 실시예에서, 컨트롤러(300)가 접속을 설정한 후, 오브젝트 데이터를 전송하는 처리는 컨트롤러(300)에 의한 제어없이 소스 노드(302)와 수신지 노드(304) 사이에서 실행된다. 따라서, 컨트롤러(300)의 부하가 줄어들고 복잡한 통신 프로시저가 필요하지 않은 단순한 통신 프로토콜이 마련될 수 있다.

또한, 이 실시예에서, 수신지 노드(304)는 각각의 비동기 방송 트랜잭션에 대한 응답을 확실히 송출한다. 따라서, 실시간 특성을 요구하지 않는 데이터가 확실히 전송될 수 있는 통신 프로토콜이 마련될 수 있다.

더욱 확실한 데이터 전송을 실현하기 위해, 버스 리셋 또는 어떤 전송 에러의 발생에 의해 데이터 전송이

인터럽트될 때, 데이터 전송은 어떠한 데이터도 빠뜨리지 않고 즉시 재개될 필요가 있다. 이하, 이 실시예의 통신 프로토콜에 규정된 재개 프로시저를 도 4b를 참조하여 설명한다.

예를 들면, 세그먼트 번호 i 를 갖는 비동기 방송 패킷이 수신된 후 버스 리셋이 발생할 때, 각 노드는 전송 처리를 중단하고, IEEE 1394-1995 규격에 규정된 프로시저에 따라 버스 초기화, 접속 구성의 재인식, 노드 ID의 설정 등을 실행한다(도 4b의 (420, 421)).

버스 재구성이 완료된 후, 각각의 수신지 노드(304)는 접속 ID 및 시퀀스 번호 i 가 저장된 재송출 요구 패킷을 방송한다(도 4b의 (422)).

비동기 방송 트랜잭션을 재개할 수 있을 때, 소스 노드(302)는 수신된 재송출 요구 패킷의 접속 ID를 확인하여 접속 ID가 저장된 확인 응답 패킷을 방송한다(도 4b의 (423)).

이어서, 소스 노드(302)는 수신된 재송출 요구 패킷에 의해 요구된 시퀀스 번호 및 그다음 번호의 세그먼트 데이터, 즉 시퀀스 번호 $i+1$ 로 시작하는 시퀀스 데이터를 연속해서 방송한다(도 4b의 (424)).

상술한 프로시저에서, 데이터 전송이 인터럽트되더라도, 컨트롤러(300), 소스 노드(302) 및 수신지 노드(304)는 각각의 노드 ID를 고려하지 않고 용이하고 확실하게 다음 데이터 전송을 재개할 수 있다.

또한, 상술한 바와 같이, 이 실시예에서는 데이터 전송이 인터럽트될때에도, 컨트롤러(300)의 제어 프로시저가 효과적으로 단순화될 수 있다.

이하, 제2실시예에서 규정되는 비동기 방송 패킷의 구성을 도 13을 참조하여 설명한다. 또한, 도 13에서, 제1실시예의 비동기 방송 패킷과 동일한 동일한 기능을 갖는 필드는 도 5에서와 같은 코드로 나타낸다.

먼저, 패킷 헤더(521)의 구성을 설명한다.

도 13에서, 필드(501)(16 비트)는 destination_ID를 나타내고, 수신지(즉, 수신지 노드304)의 노드 ID가 지시된다. 이 실시예의 통신 프로토콜에서, 오브젝트 데이터(308)의 비동기 방송 트랜잭션을 실현하기 위해, 이 필드의 값은 방송 ID(즉, $FFFF_{16}$)로서 설정된다.

필드(502)(6 비트)는 트랜잭션 라벨(t1) 필드, 또는 각 트랜잭션 고유의 태그를 나타낸다.

필드(503)(2 비트)는 재시도(rt) 코드를 나타내고, 패킷이 재시도되는가의 여부를 지정한다.

필드(504)(4 비트)는 실행해야 할 패킷 포맷 또는 트랜잭션 종류를 지정하는 트랜잭션 코드(tcode)를 나타낸다. 이 실시예에서, 이 필드의 값은 예를 들면 0001_2 로 설정되어 destination_offset 필드(507)에 의해 지시된 메모리 공간에 패킷의 데이터 블록(522)을 기록하는 처리가 요구된다(즉, 기록 트랜잭션).

필드(505)(4 비트)는 우선권(pri)을 나타내고, 우선권의 순위를 지정한다. 이 실시예에서, 이 필드의 값은 0000_2 로 설정된다.

필드(506)(16 비트)는 source_ID 또는 송신측(즉, 소스 노드(302))의 노드 ID를 나타낸다.

필드(507)(48 비트)는 destination_offset 및 각 수신지 노드(304)의 어드레스 공간의 하위 48 비트를 공통으로 지정한다. 여기서, destination_offset에 대하여, 동일한 값이 모든 접속에 있어서 설정될 수도 있거나 또는 그 접속에 있어서 상이한 값이 설정될 수도 있다. 그러나, 상이한 값이 설정될 때, 다수의 접속으로 부터의 비동기 방송 패킷은 병렬로 효율적으로 처리될 수 있다.

필드(508)(16 비트)는 data_length를 나타내고, 다음에 설명하는 데이터 필드의 길이가 바이트 단위로 지시된다.

필드(509)(16 비트)는 extended_tcode를 나타낸다. 이 실시예에서 이 필드의 값은 0000_{16} 으로 설정된다.

필드(510)(32 비트)는 필드(501) 내지 (509)에 대한 에러 검출 코드가 저장된 header_CRC를 나타낸다.

이하, 데이터 블록(522)의 구성을 설명한다. 이 실시예에서 데이터 블록(522)은 헤더 정보(1301) 및 데이터 필드(524)로 구성된다.

노드사이의 논리적 접속 관계를 식별하는 접속 ID 등은 헤더 정보(1301)에 저장된다.

또한, 데이터 필드(524)는 세그먼트 데이터가 저장되는 가변 길이를 갖는다. 여기서, 데이터 필드(524)에 저장된 세그먼트 데이터가 퀴드렛의 배수가 아닐 때, 퀴드렛을 만족하지 않는 부분은 제로로 채워진다.

필드(511)(16 비트)는 connection_ID를 나타내고, 이 실시예의 접속 ID를 저장한다. 이 실시예의 1394 인터페이스는 이 필드에 저장된 접속 ID에 따라 소스 노드(302)와 적어도 하나의 수신지 노드(304) 사이에 설정된 접속을 식별한다. 이 실시예에서는 2^{16} × 노드수의 접속이 확립될 수 있다. 따라서, 각 접속에 의해 사용되는 통신 대역의 총량이 전송 경로의 용량에 도달할 때까지 다수의 접속이 설정될 수 있다.

필드(512)(8 비트)는 protocol_type를 나타내고, 헤더 정보(1301)에 따른 통신 프로시저(즉, 통신 프로토콜 종류)가 지시된다. 이 실시예의 통신 프로토콜이 지시될 때, 이 필드의 값은 예를 들면, 01_{16} 이다.

필드(513)(8 비트)는 control_flags를 나타내고, 이 실시예의 통신 프로토콜의 통신 프로시저 등을 제어하는 소정의 제어 데이터가 설정된다. 이 실시예에서, 이 필드의 최상위 비트는 예를 들면, resend_request 플래그로서 설정된다. 따라서, 이 필드의 최상위 비트가 1의 값을 가질 때, 이 실시예

의 통신 프로토콜에 따른 재송출 요구가 발생된 것을 나타낸다.

필드(514)(16 비트)는 sequence_number를 나타내고, 연속하는 값(즉, 시퀀스 번호)이 지정된 접속 ID(필드(511)에 의해 지정된 접속 ID)에 따라 전송된 패킷에 대하여 설정된다. 수신지 노드(304)는 시퀀스 번호에 의해 비동기 방송 트랜잭션이 연속해서 가해진 세그먼트 데이터의 연속성을 감시할 수 있다. 고르지 못한 발생할 때, 수신지 노드(304)는 시퀀스 번호에 따라 재송출을 요구할 수 있다.

필드(515)(16 비트)는 reconfirmation_number를 나타낸다. 이 실시예에서, 이 필드는 재송출 요구 플래그가 1의 값을 가질때만 의미를 갖는다. 예를 들면, 재송출 요구 플래그의 값이 1일 때, 재송출을 요구하는 패킷의 시퀀스 번호가 이 필드에 설정된다.

필드(516)(16 비트)는 buffer_size를 나타낸다. 수신지 노드(304)의 내부 버퍼 크기가 이 필드에 설정된다.

필드(1302)(16 비트)는 예약을 나타내고, 장래 확장 규격을 위해 예약된다.

필드(519)(32 비트)는 data_CRC를 나타내고, (헤더 정보(1301) 및 데이터 필드(524)를 포함하여) 필드(511) 내지 (518)에 대한 에러 검출 코드가 header_CRC와 동일한 방식으로 저장된다.

이하, 확인 응답 패킷의 구성을 설명한다. 도 13에 도시한 비동기 방송 패킷에 의해 세그먼트 데이터를 수신한 수신지 노드(304)는 도 14에 도시한 확인 응답 패킷을 사용하여 응답을 반환한다.

도 14에서, 필드(1401)(4 비트)는 ack_code가 저장된 필드이다. 상술한 ack_busy_A, ack_busy_B, ack_busy_X 또는 다른 코드가 이 필드에 의해 소스 노드(302)로 전송된다.

필드(1402)(4 비트)는 ack_parity가 저장된 필드이다. 이 필드에는 확인 응답 패킷의 패리티 체크 코드가 저장된다. 예를 들면, 제2실시예에서 ack_code의 값 1의 보수가 저장된다. 이 필드는 필드(1401)의 에러 검출에 사용된다.

다음 필드(1403)(8 비트)는 min_retry_period가 저장된 필드이다. 이 필드에는 도 10 및 도 11에 도시한 최소 재시도 기간의 값이 예를 들면 1 밀리초의 단위로 저장된다. 예를 들면, 이 필드의 값이 01₁₆일 때, 확인 응답 패킷을 수신한 소스 노드(302)는 최소 재시도 기간을 1 밀리초로 설정하고, 이 기간동안 재시도를 실행하지 않도록 동작한다. 또한, 필드(1403)에 설정될 수 있는 시간 단위는 1 밀리초에 한정되지 않고, 다른 시간 단위를 사용할 수도 있다.

여기서, 각 수신지 노드(304)는 그의 수신 능력 또는 부하 상태에 따라 최적의 값을 가변적으로 설정하여 이 필드(1403)에 저장할 수 있다. 예를 들면, 각각의 수신지 노드(304)는 그의 부하 상태를 감시하여, 그 부하가 클 때 최소 재시도 기간의 값을 크게 설정하고, 부하가 작을 때는 그 값을 작게 설정한다.

노드 부하 상태에 대하여, 각 노드에서 실행되는 재시도 회수, 노드로 전송되는 패킷수, 노드 버퍼의 점유 상태, 노드의 접속수 및 다른 여러 인디케이터를 사용할 수 있다. 또한, 노드 부하 상태를 검출하는 인디케이터는 노드 부하 상태를 검출할 수 있는한 제한되지 않는다.

또한, 이 실시예에서, 필드(1403)에 저장되는 값에는 특별한 의미가 마련될 수 있다. 예를 들면, 표준(디폴트)값은 00₁₆으로서 설정되고, 이 값에 의해 지시된 최소 재시도 기간은 100 밀리초로 설정될 수 있다. 또한, 표준(디폴트)값에 의해 지시되는 최소 재시도 기간은 100 밀리초에 한정되지 않고, 다른 값을 사용할 수도 있다. 또한, 예를 들면, 소위 즉시 재시도를 지시할 수 있고, 10₁₆을 가능한한 빨리 재시도를 실행하기 위한 표준(디폴트)값으로서 설정한다.

도 15는 최소 재시도 기간을 설정하는 동작을 도시한 도면이다. 도 15의 (a)는 비동기 방송 트랜잭션을 실행하는 소스 노드(302)와 그 트랜잭션에 대하여 도 14에 도시한 확인 응답 패킷을 반환하는 수신지 노드(304) 사이에서 패킷이 송신/수신되는 동작 흐름을 도시하고 있다. 또한, 도 15의 (b)는 시간에 따른 수신지 노드(304)의 부하 상태의 변화를 도시한 그래프이다. 도 15의 (b)에서, 수신지 노드(304)의 부하 상태는 횡방향으로 도시되고, 경과 시간은 종방향으로 도시되어 있다.

도 15에서, 소스 노드(302)가 제1비동기 방송 트랜잭션(기록 요구 #1)를 실행하는 시간 t₁에서, 수신지 노드(304)의 부하는 비교적 작은 L₁이고, 사용중이다(도 15의 (1501)). 이 경우, 예를 들면, 수신지 노드(304)는 min_retry_period의 값을 32₁₆으로 설정하고, ack_busy_X의 확인 응답 패킷을 반환한다(도 15의 (1502)).

소스 노드(302)는 확인 응답 패킷의 min_retry_period의 값에서 50 밀리초로 최소 재시도 기간을 설정한다(도 15의 (1503)). 소스 노드(302)는 최소 재시도 기간동안은 재시도를 실행하지 않고, 적어도 50 밀리초후 재시도를 실행한다(도 15의 (1504), 재시도 #1). 도 15에서, 재시도가 성공적이고, 기록 트랜잭션이 완료된다(도 15의 (1505)).

이어서, 소스 노드(302)가 제2 비동기 방송 트랜잭션(기록 요구 #2)를 실행하는 시간 t₂에서, 수신지 노드(304)의 부하는 비교적 큰 L₂이고, 사용중이다(도 15의 (1506)). 예를 들면, 수신지 노드(304)는 min_retry_period의 값을 B4₁₆으로 설정하고, ack_busy_X의 확인 응답 패킷을 반환한다(도 15의 (1506)).

소스 노드(302)는 확인 응답 패킷의 min_retry_period의 값에서 180 밀리초로 최소 재시도 기간을 설정한다(도 15의 (1507)). 소스 노드(302)는 이 최소 재시도 기간 동안 재시도를 실행하지 않고, 적어도 180 밀리초후 재시도를 실행한다(도 15의 (1508), 재시도 #2).

상술한 동작에 있어서, 이 실시예에서는 확인 응답 패킷을 송출해야할 수신지 노드가 최소 재시도 기간

을 동적으로 설정하므로, 더욱 확실한 데이터 통신이 보장되는 통신 시스템 및 통신 프로토콜이 마련될 수 있다. 또한, 버스가 혼잡하더라도, 사용중 상태가 빈번히 발생하는 것이 방지되어, 대드락이 발생하는 것을 방지할 수 있다. 또한, 상술한 동작에 있어서, 이 실시예에서는 최소 재시도 기간이 동적으로 설정될 수 있으므로, 통신 리소스가 적절히 분산될 수 있어, 통신 경로가 재시도를 실행하기 위해 소스 노드에 의해 점유되지 않으면서 통신 효율을 향상될 수 있다.

또한, 상술한 동작에서는 송신 속도가 빠른 소스 노드와 수신 속도가 낮은 수신지 노드 사이에서 전송이 실행될 때에도, 소스 노드가 빈번히 재시도를 실행하는 것을 방지할 수 있고, 수신지 노드의 수신 버퍼가 풀로 되는 것이 항상 방지될 수 있다.

또한, 도 15에서는 설명을 간략화하기 위해, 하나의 소스 노드(302)와 하나의 수신지 노드(304) 사이의 전송을 도시하고 있지만, 하나 이상의 수신지 노드(304)가 마련될 때에도 동일한 처리를 실행할 수 있다. 이 경우, 소스 노드(302)는 각각의 수신지 노드(304)에서 전송된 확인 응답 패킷에 따라 최적의 최소 재시도 기간을 결정한다.

상술한 바와 같이, 이 실시예에서는 물리적 접속 모드에 독립인 논리적 접속 관계가 IEEE 1394-1995 규격과 같은 버스형 네트워크에서 구성될 수 있다.

또한, 이 실시예에 따르면, IEEE 1394-1995 규격에 따르는 통신 시스템에서, 실시간 특성은 요구하지 않지만 신뢰성은 요구하는 비교적 대량의 오브젝트 데이터(예를 들면, 정지 화상 데이터, 그래픽 데이터, 텍스트 데이터, 파일 데이터, 프로그램 데이터 등)가 하나 이상의 세그먼트 데이터로 분할되어 연속적으로 전송되는 완전히 새로운 통신 프로토콜이 마련될 수 있다.

또한, 이 실시예에 따르면, IEEE 1395-1995 규격에 따르는 통신 시스템에서, 데이터를 비동기로 방송하기 위해 통신 시스템을 사용하는 다수의 장치사이에서 데이터 통신을 실현하는 완전히 새로운 통신 프로토콜이 마련될 수 있다.

또한, 이 실시예에서는 다수의 연속 데이터가 IEEE 1394-1995 규격의 동시성 전송 시스템을 사용하지 않고 확실히 전송될 수 있다. 또한, 하나의 오브젝트 데이터가 다수의 데이터로 분할되어 확실히 전송될 수 있다.

또한, 이 실시예에서는 다수의 장치사이의 통신이 하나의 접속으로 제어되므로, 많은 통신 대역을 사용하지 않고 다수의 통신을 동시에 실행할 수 있다.

또한, 이 실시예에서는 데이터 전송이 버스 리셋 또는 전송 에러에 의해 인터럽트될 때, 어느 세그먼트 데이터가 상실되었는가를 알 수 있고, 다음의 매우 복잡한 통신 프로시저없이 전송을 재개할 수 있다.

다른 실시예

상기 실시예에서 설명된 통신 프로토콜을 실현하기 위한 통신 프로토콜 및 여러 가지 필요한 처리 동작은 소프트웨어에 의해 실현될 수 있다.

예를 들면, 상술한 실시예 기능을 실현하는 프로그램 코드가 저장된 기록 매체는 실시예의 통신 시스템을 구성하는 장치 컨트롤러(예를 들면, 도 2의 MPU(12), 시스템 컨트롤러(50), 프린터 컨트롤러(68))에 공급된다. 이어서, 컨트롤러는 그 기록 매체에 저장된 프로그램 코드를 판독하고, 그 프로그램에 코드에 따라 실시예 기능을 실현하기 위해 통신 시스템 또는 장치 동작을 제어한다. 따라서, 상술한 실시예가 실현될 수 있다.

또한, 상술한 실시예 기능을 실현하기 위한 프로그램 코드가 저장된 기록 매체는 각 장치에 탑재된 1394 인터페이스(14, 44, 62)에 제공되고, 1394 인터페이스(14, 44, 62)의 동작을 제어하는 컨트롤러(예를 들면, 도 8의 시리얼 버스 매니지먼트(806))는 그 기록 매체에 저장된 프로그램 코드에 따라 실시예 기능을 실현하는 처리 동작을 제어한다.

이 경우, 기록 매체에서 판독된 프로그램 코드는 실시예 기능을 실현하고, 프로그램 코드 및 프로그램 코드를 컨트롤러(예를 들면, 기록 매체 자신)에 공급하는 수단은 본 발명을 구성한다.

프로그램 코드가 저장된 기록 매체에 대하여, 예를 들면, 플로피 디스크, 하드 디스크, 광디스크, 자기 광학 디스크, CD-ROM, 자기 테이프, 비휘발성 메모리 카, ROM 등을 사용할 수 있다.

또한, 기록 매체에서 판독된 프로그램 코드가 컨트롤러상에서 조작되는 OS(operating system), 여러 가지 애플리케이션 소프트웨어 등과 협력하여 실시예 기능을 실현하는 경우도 본 발명이 포함하는 것은 물론이다.

또한, 본 발명은 기록 매체에서 판독된 프로그램 코드가 컨트롤러에 접속된 기능 확장 유닛에 탑재된 메모리에 저장된 후, 기능 확장 유닛에 마련된 컨트롤러가 메모리에 저장된 프로그램 코드에 따라 실제 처리의 일부 또는 부분을 실행하여 실시예 기능을 실현하는 경우도 포함한다.

본 발명은 그의 정신 또는 본질적 특성을 벗어나지 않고 다른 특정 형태로 구현될 수 있다.

예를 들면, 제1실시예의 통신 프로토콜은 제2실시예의 것과 결합될 수 있다. 따라서, i 번째와 $(i+1)$ 번째 비동기 방송 트랜잭션 사이의 최적 시간 간격을 결정할 수 있다. 또한, i 번째 비동기 방송 트랜잭션이 정상적으로 수신되지 않더라도, 재시도를 소정의 시간동안 금지할 수 있다.

또한, 실시예에서는 IEEE 1394-1995 규격에 따르는 네트워크에 적용가능한 통신 프로토콜을 설명하였지만, 본 발명은 그것에 한정되지 않는다. 실시예의 통신 프로토콜은 IEEE 1394-1995 규격과 같은 버스형 네트워크 또는 버스형 네트워크를 실질적으로 구성할 수 있는 네트워크에 적용될 수 있다.

따라서, 상술한 실시예는 모든 점에 있어서 단지 예시이고, 본 발명을 제한하는 것으로 해석되어서는 아

니된다.

발명의 효과

본 발명의 범위는 첨부 특허청구범위의 범위에 의해 규정되고, 본 명세서의 특정 설명에 의해 전혀 제한되지 않는다. 또한, 특허청구범위에 속하는 모든 수정 및 변경은 본 발명의 범위내에 든다고 고려된다.

(57) 청구의 범위

청구항 1

하나이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 비동기 통신을 실행하는 소스 노드;

상기 소스 노드에서 전송된 데이터를 수신하는 하나이상의 수신지 노드; 및

상기 소스 노드와 상기 하나이상의 수신지 노드사이의 논리적 접속 관계를 설정하는 컨트롤러를 포함하며,

상기 소스 노드와 상기 컨트롤러의 적어도 하나는 상기 비동기 통신을 실행하는 타이밍을 제어하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 2

제 1 항에 있어서,

상기 소스 노드는 상기 하나이상의 수신지 노드와의 논리적 접속 관계에 따라 상기 데이터를 전송하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 3

제 1 항에 있어서,

상기 소스 노드는 상기 비동기 통신을 적어도 한번 연속해서 실행하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 4

제 1 항에 있어서,

상기 하나이상의 수신지 노드는 상기 소스 노드와의 논리적 접속 관계에 따라 상기 데이터를 수신하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 5

제 1 항에 있어서,

상기 하나이상의 수신지 노드는 전송된 데이터에 대한 응답을 반환하기 위해 상기 비동기 통신을 사용하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 6

제 1 항에 있어서,

상기 컨트롤러는 상기 소스 노드와 상기 하나이상의 수신지 노드 사이의 하나이상의 논리적 접속 관계를 설정할 수 있는 것을 특징으로 하는 데이터 통신 시스템.

청구항 7

제 1 항에 있어서,

상기 논리적 접속 관계는 상기 데이터가 전송된후 상기 컨트롤러 또는 수신지 노드에 의해 해제되는 것을 특징으로 하는 데이터 통신 시스템.

청구항 8

제 1 항에 있어서,

상기 소스 노드와 상기 컨트롤러의 적어도 하나는 다음 비동기 통신이 실행될 때까지의 시간의 기간을 제어하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 9

제 1 항에 있어서,

상기 소스 노드와 상기 컨트롤러의 적어도 하나는 상기 비동기 통신에 대응하는 응답이 수신될 때까지의 시간을 제어하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 10

제 1 항에 있어서,

상기 하나이상의 수신지 노드는 상기 소스 노드와 상기 컨트롤러의 적어도 하나에게 상기 비동기 통신을 수신하기 위해 필요한 시간 간격을 통지하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 11

제 1 항에 있어서,

상기 소스 노드는 상기 비동기 통신을 재시도하는 시간을 제어하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 12

제 1 항에 있어서,

상기 하나이상의 수신지 노드는 상기 소스 노드에게 상기 비동기 통신을 재시도하기 위해 필요한 시간 간격을 통지하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 13

제 1 항에 있어서,

상기 데이터 통신 시스템은 상기 소스 노드와 상기 하나이상의 수신지 노드 사이에서 상기 데이터를 전송하기 위해 필요한 초기화를 실행하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 14

제 13 항에 있어서,

상기 컨트롤러는 상기 초기화에 있어서 초기 정보 세트의 일부를 설정할 수 있는 것을 특징으로 하는 데이터 통신 시스템.

청구항 15

제 13 항에 있어서,

상기 하나이상의 수신지 노드는 상기 소스 노드에게 상기 초기화에 필요한 초기 정보를 통지하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 16

제 13 항에 있어서,

상기 소스 노드는 상기 초기화를 실행하기 위해 상기 하나이상의 수신지 노드에서 통지된 초기 정보를 사용하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 17

제 13 항에 있어서,

상기 하나이상의 수신지 노드의 메모리 공간을 공통으로 지정하는 적어도 하나의 수신지 어드레스 및 수신 버퍼 크기는 상기 초기화에 있어서 설정되는 것을 특징으로 하는 데이터 통신 시스템.

청구항 18

제 1 항에 있어서,

상기 소스 노드는 상기 데이터를 방송하기 위해 상기 비동기 통신을 사용하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 19

제 1 항에 있어서,

상기 소스 노드는 상기 하나이상의 수신지 노드의 공통 메모리 공간에 상기 데이터를 기록하기 위해 상기 비동기 통신을 사용하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 20

제 1 항에 있어서,

상기 하나이상의 수신지 노드는 수신지 노드의 공통 메모리 공간에 상기 데이터를 저장하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 21

제 1 항에 있어서,

상기 비동기 통신은 IEEE 1394-1995 규격의 비동기 전송 시스템에 따르는 것을 특징으로 하는 데이터 통신 시스템.

청구항 22

제 1 항에 있어서,

상기 데이터 통신 시스템은 버스형 네트워크인 것을 특징으로 하는 데이터 통신 시스템.

청구항 23

제 1 항에 있어서,

상기 데이터 통신 시스템은 IEEE 1394-1995 규격에 따르는 네트워크인 것을 특징으로 하는 데이터 통신 시스템.

청구항 24

제 1 항에 있어서,

상기 하나 이상의 세그먼트를 포함하는 데이터는 정지 화상 데이터, 그래픽 데이터, 텍스트 데이터, 파일 데이터 및 프로그램 데이터중 적어도 하나인 것을 특징으로 하는 데이터 통신 시스템.

청구항 25

소스 노드와 하나 이상의 수신지 노드사이의 논리적 접속 관계를 설정하는 스텝;

상기 하나 이상의 수신지 노드로 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 비동기 통신을 실행하는 스텝;

상기 비동기 통신을 실행하는 타이밍을 제어하는 스텝; 및

상기 비동기 통신을 사용하여 전송된 데이터를 수신하기 위해 상기 논리적 접속 관계를 사용하는 스텝을 포함하는 것을 특징으로 하는 데이터 통신 방법.

청구항 26

논리적 접속 관계에 따라 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 방송 통신을 실행하는 소스 노드; 및

상기 논리적 접속 관계에 따라 상기 소스 노드에서 전송된 데이터를 수신하는 하나 이상의 수신지 노드를 포함하며,

상기 소스 노드는 상기 방송 통신을 실행하는 타이밍을 제어하는 것을 특징으로 하는 데이터 통신 시스템.

청구항 27

논리적 접속 관계에 따라 하나 이상의 수신지 노드로 하나 이상의 세그먼트를 포함하는 데이터를 전송하기 위해 적어도 한번 방송 통신을 실행하는 스텝;

상기 방송 통신을 실행하는 타이밍을 제어하는 스텝; 및

상기 논리적 접속 관계에 따라 상기 소스 노드에서 전송된 데이터를 수신하는 스텝을 포함하는 것을 특징으로 하는 데이터 통신 방법.

청구항 28

하나 이상의 세그먼트를 포함하는 데이터를 다수의 통신 패킷으로 패킷화하는 수단; 및

하나 이상의 수신지 노드와의 논리적 접속 관계 세트에 따라 상기 통신 패킷을 연속해서 전송하는 수단을 포함하며,

상기 통신 패킷은 소정의 시간 경과후 비동기로 전송되는 것을 특징으로 하는 데이터 통신 장치.

청구항 29

제 28 항에 있어서,

상기 통신 패킷은 소정의 시간 경과후 재시도되는 것을 특징으로 하는 데이터 통신 장치.

청구항 30

하나 이상의 세그먼트를 포함하는 데이터를 다수의 통신 패킷으로 패킷화하는 스텝; 및

하나 이상의 수신지 노드와의 논리적 접속 관계 세트에 따라 상기 통신 패킷을 연속해서 전송하는 스텝을 포함하며,

상기 통신 패킷은 소정의 시간 경과후 비동기로 전송되는 것을 특징으로 하는 데이터 통신 방법.

청구항 31

제 30 항에 있어서,

소정의 시간 경과후 상기 통신 패킷을 재시도하는 스텝을 더 포함하는 것을 특징으로 하는 데이터 통신 방법.

청구항 32

소스 노드와의 논리적 접속 관계 세트에 따라 상기 소스 노드에서 연속해서 전송된 통신 패킷을 수신하는 수단; 및

상기 통신 패킷에 포함된 데이터를 다른 장치와 공통인 메모리 공간에 기록하는 수단을 포함하며,

상기 통신 패킷은 소정의 시간 경과후 비동기로 전송되는 것을 특징으로 하는 데이터 통신 장치.

청구항 33

소스 노드와의 논리적 접속 관계 세트에 따라 상기 소스 노드에서 연속해서 전송되고, 소정의 시간 경과후 비동기로 전송되는 통신 패킷을 수신하는 스텝; 및

상기 통신 패킷에 포함된 데이터를 다른 장치와의 공통인 메모리 공간에 기록하는 스텝을 포함하는 것을 특징으로 하는 데이터 통신 방법.

청구항 34

소스 노드와 하나이상의 수신지 노드 사이의 논리적 접속 관계를 설정하고, 상기 논리적 접속 관계에 따라 연속해서 전송된 통신 패킷의 시간 간격을 상기 소스 노드에 있어서 설정하는 수단; 및

상기 소스 노드 및 상기 하나이상의 수신지 노드에게 상기 논리적 접속 관계를 식별하는 접속 ID를 통지하는 수단

을 포함하는 것을 특징으로 하는 데이터 통신 장치.

청구항 35

소스 노드와 하나이상의 수신지 노드 사이의 논리적 접속 관계를 설정하는 스텝;

상기 소스 노드 및 상기 하나이상의 수신지 노드에게 상기 논리적 접속 관계를 식별하는 접속 ID를 통지하는 스텝; 및

상기 논리적 접속 관계에 따라 연속해서 전송된 통신 패킷의 시간 간격을 상기 소스 노드에 있어서 설정하는 스텝

을 포함하는 것을 특징으로 하는 데이터 통신 방법.

청구항 36

하나이상의 세그먼트를 포함하는 데이터를 다수의 통신 패킷으로 패킷화하는 수단; 및

하나이상의 수신지 노드와의 논리적 접속 관계 세트에 따라 상기 통신 패킷을 연속해서 전송하는 수단을 포함하며,

상기 통신 패킷은 소정의 시간 경과후 비동기로 전송되는 것을 특징으로 하는 디지털 인터페이스.

청구항 37

소스 노드와의 논리적 접속 관계 세트에 따라 상기 소스 노드에서 연속해서 전송된 통신 패킷을 수신하는 수단; 및

상기 통신 패킷에 포함된 데이터를 다른 장치와 공통인 메모리 공간에 기록하는 수단을 포함하며,

상기 통신 패킷은 소정의 시간 경과후 비동기로 전송되는 것을 특징으로 하는 디지털 인터페이스

청구항 38

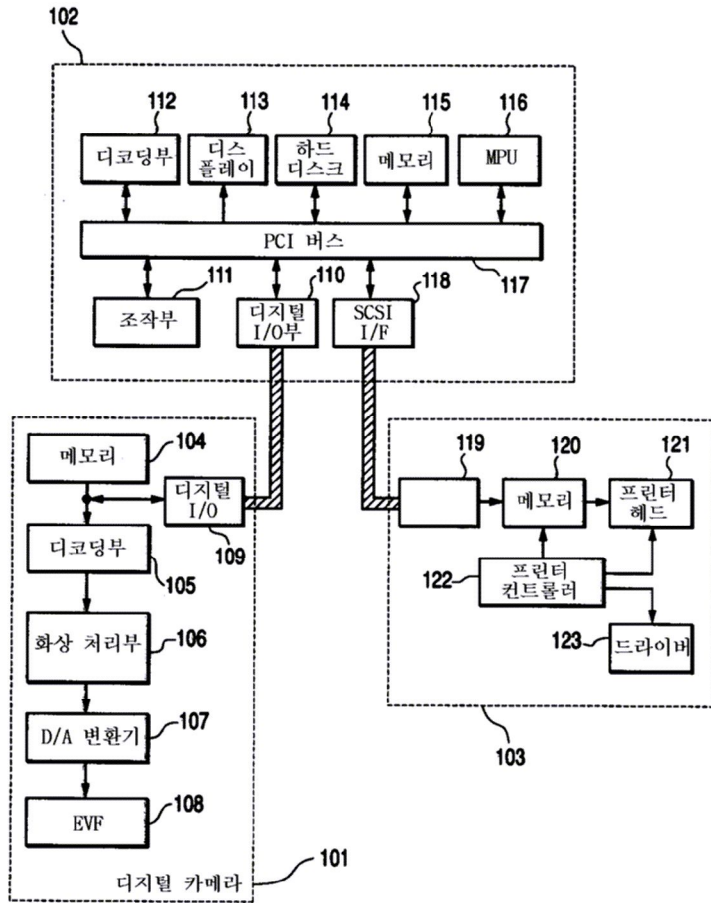
소스 노드와 하나이상의 수신지 노드 사이의 논리적 접속 관계를 설정하고, 상기 논리적 접속 관계에 따라 연속해서 전송된 통신 패킷의 시간 간격을 상기 소스 노드에 있어서 설정하는 수단; 및

상기 소스 노드 및 상기 하나이상의 수신지 노드에게 상기 논리적 접속 관계를 식별하는 접속 ID를 통지하는 수단

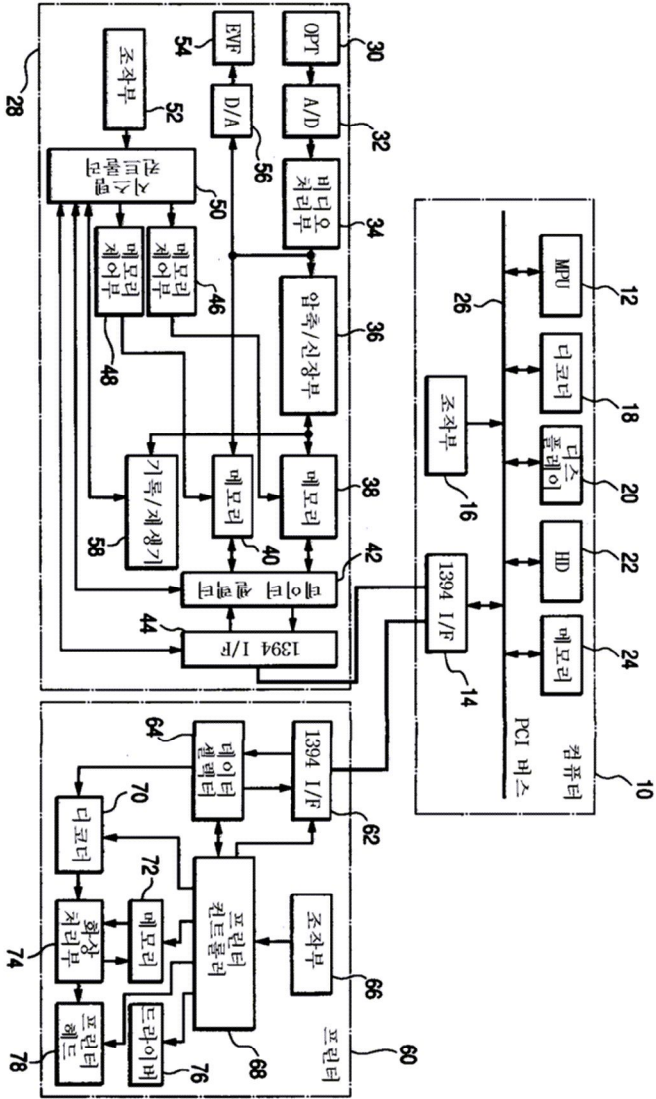
을 포함하는 것을 특징으로 하는 디지털 인터페이스.

도면

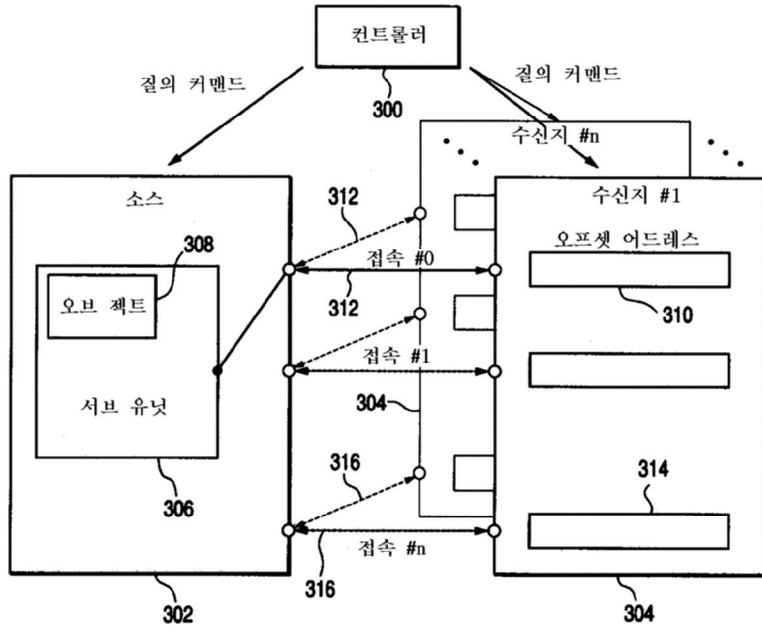
도면1



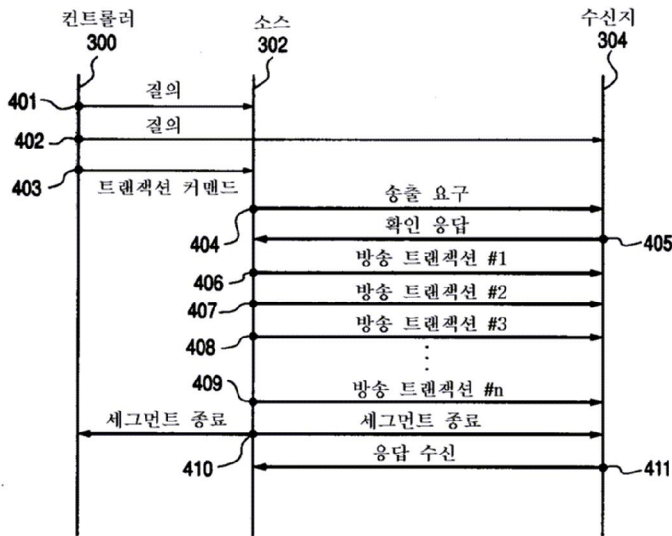
도면2



도면3



도면4a



도면4b

