



(19) 대한민국특허청(KR)  
(12) 공개특허공보(A)

(11) 공개번호 10-2013-0018235  
(43) 공개일자 2013년02월20일

(51) 국제특허분류(Int. Cl.) *H04L 1/00* (2006.01) *H04L 29/02* (2006.01)

(21) 출원번호 10-2012-7023180

(22) 출원일자(국제) 2011년03월07일  
심사청구일자 없음

(85) 번역문제출일자 2012년09월04일

(86) 국제출원번호 PCT/KR2011/001543

(87) 국제공개번호 WO 2011/108904  
국제공개일자 2011년09월09일

(30) 우선권주장  
13/038,938 2011년03월02일 미국(US)  
(뒷면에 계속)

(71) 출원인  
**삼성전자주식회사**  
경기도 수원시 영통구 삼성로 129 (매탄동)

(72) 발명자  
**아부수라 샤디**  
미국 텍사스 75080 리차드슨 콜린 컴퓨터,  
샵4223 더블유 레너 알디 280

**피섹 에란**  
미국 텍사스 75025 플라노 콜린 컴퓨터, 스톤 마운틴 코트 2601

**쿱 피칸**  
미국 텍사스 75002 알렌 콜린 컴퓨터, 새드브룩

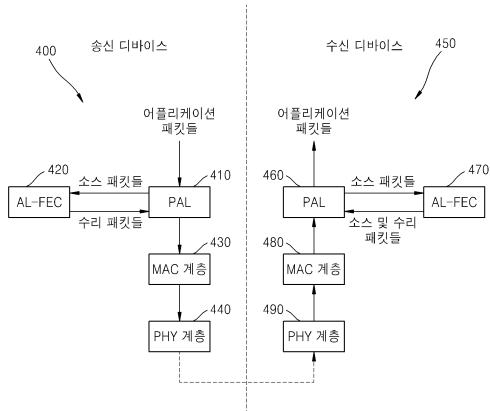
전체 청구항 수 : 총 20 항

(54) 발명의 명칭 와이 기그를 위한 응용 계층 순방향 오류정정 프레임워크

(57) 요약

무선 통신 네트워크에서 무선 통신 디바이스의 순방향 오류 정정(Forward Error Correction: FEC)을 수행하는 방법 및 장치가 제공된다. 무선 통신 네트워크 내의 무선 통신 디바이스의 순방향 오류 정정을 수행하는 방법은 성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하는 단계를 포함한다. 소스 패킷(Source Packet)들 및 적어도 하나의 패리티 패킷을 위해 시스터매틱(Systematic) 패킷들을 인코딩하기 위하여  $k$  개의 상기 소스 패킷들의 집합에 단일 패리티 검사(Single Parity Check: SPC) AL-FEC 코드(Code)를 적용한다. 각 패리티 패킷의 헤더는 패리티 패킷(Parity Packet) 지시자(indicator)를 포함한다. 미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 패킷을 처리한다.

## 대표도 - 도4



(30) 우선권주장

61/311,133 2010년03월05일 미국(US)

61/332,581 2010년05월07일 미국(US)

---

## 특허청구의 범위

### 청구항 1

무선 통신 네트워크(Network) 내의 무선 통신 디바이스(Device)의 순방향 오류 정정(Forward Error Correction: FEC)을 수행하는 방법에 있어서,

성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하는 단계;

각 패리티 패킷의 헤더는 패리티 패킷(Parity Packet) 지시자(indicator)를 포함하고, 소스 패킷(Source Packet)들 및 적어도 하나의 패리티 패킷을 위해 시스터매틱(Systematic) 패킷들을 인코딩하기 위해서  $k$  개의 상기 소스 패킷들의 집합에 단일 패리티 검사(Single Parity Check: SPC) AL-FEC 코드(Code)를 적용하는 단계; 및

미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 패킷을 처리하는 단계를 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행 방법.

### 청구항 2

제 1항에 있어서,

상기 SPC AL-FEC 코드는,

비-인터리브된(Non-Interleaved) 모드(Mode) 및 인터리브된(Interleaved) 모드 중 하나를 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

### 청구항 3

제 2항에 있어서,

상기 비-인터리브된 모드에서 상기 SPC AL-FEC 코드를 적용할 때, 상기 시스터매틱 패킷들 및 적어도 하나의 패리티 패킷을 인코딩(Encoding)하는 것은,

하나의 패리티 패킷을 위해 시퀀스(Sequence) 번호  $i$ 를 생성하는 단계;

상기  $k$  개의 소스 패킷들을 위해 시퀀스 번호들  $i+1, i+2, \dots, i+k$ 를 생성하는 단계; 및

상기 시퀀스 번호  $i$ 인 상기 패리티 패킷의 페이로드(Payload)를 계산하기 위해, 짧은 소스 패킷들에는 임시적으로 송신되지 않을 가상의 0들(Virtual Zeros)을 덧붙이고(Padded), 상기  $k$  소스 패킷들에 비트단위 베타적 논리합(A Bit-wise XOR)를 수행하는 단계를 포함하며,

상기 패리티 패킷을 위한 상기 시퀀스 번호는 상기 패리티 패킷의 상기 헤더 내에 포함되고, 상기  $k$  소스 패킷들을 위한 상기 시퀀스 번호들은 대응하는 시스터매틱 패킷들의 상기 헤더들 내에 포함되는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

### 청구항 4

제 2항에 있어서,

소스 패킷들의  $x$  개의 부분집합(Subset)을 가진 상기 인터리브된 모드에서 상기 SPC AL-FEC 코드를 적용할 때, 상기 시스터매틱 패킷들 및 상기 적어도 하나의 패리티 패킷을 인코딩하는 것은,

각 패리티 패킷은 소스 패킷들의  $x$  개의 부분집합들 중 하나에 대응하고,  $x$  개의 패리티 패킷들을 위해 시퀀스 번호들  $i, i+1, i+2, \dots, i+(x-1)$ 을 생성하는 단계;

상기  $k$  개의 소스 패킷들을 위해 시퀀스 번호들  $i+1, i+2, \dots, i+k$ 를 생성하는 단계;

연속적인 시퀀스 번호들을 가진 소스 패킷들이 다른 부분집합들에 할당되도록 상기  $x$  개의 부분집합들에 걸쳐 상기  $k$  개의 소스 패킷들을 인터리빙(Interleaving) 하는 단계; 및

짧은 소스 패킷들에는 임시적으로 송신되지 않을 가상의 0들(Virtual Zeros)을 덧붙이고(Padded), 상기 대응하는 소스 패킷들의 부분집합에 비트단위 배타적 논리합을 수행함으로써 각 패리티 패킷의 페이로드를 계산하는 단계를 포함하고,

상기 패리티 패킷들에 대한 시퀀스 넘버는 상기 대응하는 패리티 패킷들에 대한 상기 헤더들에 포함되고, 상기 k 개의 소스 패킷들을 위한 상기 시퀀스 번호들은 대응하는 시스터매틱 패킷들을 위한 헤더들 내에 포함되어 있는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

## 청구항 5

제 1 항에 있어서,

상기 소스 패킷들 및 적어도 하나의 패리티 패킷을 위해 시스터매틱 패킷들을 인코딩 하는 단계는,

상기 SPC AL-FEC 코드와 결합하여 해밍(Hamming) AL-FEC 코드를 수행하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

## 청구항 6

제 1항에 있어서,

시스터매틱 패킷들 및 수리(Repair) 패킷들 중 적어도 하나를 포함하는 패킷 블록을 수신하는 단계;

상기 수신된 패킷 블록 내에 적어도 하나의 패킷이 없어졌는지(Missing) 여부를 판단하는 단계;

적어도 하나의 패킷이 없어진 경우, 각 수신된 패킷 내의 상기 패리티 패킷 지시자(Indicator)를 검사함으로써, 상기 수신된 패킷들이 AL-FEC 인코딩되었는지 여부를 판단하는 단계;

상기 AL-FEC 코드가 감지된 경우, 상기 적어도 하나의 없어진 패킷이 복구될 수 있는지 여부를 판단하는 단계; 및

상기 적어도 하나의 없어진 패킷이 복구될 수 있는지의 판단에 대한 대응으로 상기 수신된 시스터매틱 및 수리 패킷들을 사용하여 AL-FEC 디코딩(Decoding)을 수행하는 단계를 더 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

## 청구항 7

제 6항에 있어서,

상기 수신된 패킷들이 AL-FEC 인코딩되었는지 여부를 판단하는 단계는,

수신된 패리티 패킷들의 숫자에 기초하여 상기 수신된 패킷들이 인터리브 되었는지 여부를 결정하는 단계를 더 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

## 청구항 8

무선 통신 네트워크(Network) 내의 무선 통신 디바이스(Device)의 순방향 오류 정정(Forward Error Correction: FEC)을 수행하기 위한 장치에 있어서,

다른 통신 디바이스들(Devices)과 통신하기 위해 설정된 송신기;

성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하고,

소스 패킷(Source Packet)들 및 적어도 하나의 패리티 패킷을 위해 시스터매틱(Systematic) 패킷들을 인코딩하기 위해서 k 개의 상기 소스 패킷들의 집합에 단일 패리티 검사(Single Parity Check: SPC) AL-FEC 코드(Code)를 적용하고, 각 인코딩된 패킷의 헤더는 패리티 패킷(Parity Packet) 지시자(indicator)를 포함하며,

미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 패킷을 처리하도록 설정되고, 상기 송신기와 결합된 컨트롤러(Controller)를 포함하는 것을 특징으로 하는 FEC를 수행하기 위한 장치.

**청구항 9**

제 8항에 있어서,

상기 SPC AL-FEC 코드는 비-인터리브된(Non-Interleaved) 모드(Mode) 및 인터리브된(Interleave) 모드 중 하나를 포함하는 것을 특징으로 하는 무선 통신 장치에서의 FEC를 수행하기 위한 장치.

**청구항 10**

제 9항에 있어서,

상기 비-인터리브된 모드에서 상기 SPC AL-FEC 코드를 적용할 때, 상기 컨트롤러는,

하나의 패리티 패킷을 위해 시퀀스(Sequence) 번호  $i$ 를 생성하고,

상기  $k$  개의 소스 패킷들을 위해 시퀀스 번호들  $i+1, i+2, \dots, i+k$ 를 생성하고,

상기 시퀀스 번호  $i$ 인 상기 패리티 패킷의 페이로드(Payload)를 계산하기 위해, 짧은 소스 패킷들에는 임시적으로 송신되지 않을 가상의 0들(Virtual Zeros)을 덧붙이고(Padded), 상기  $k$  소스 패킷들에 비트단위 배타적 논리합(A Bit-wise XOR)를 수행하고,

상기 패리티 패킷을 위한 상기 시퀀스 번호는 상기 패리티 패킷의 상기 헤더 내에 포함되고, 상기  $k$  소스 패킷들을 위한 상기 시퀀스 번호들은 상기 대응하는 시스터매틱 패킷들의 상기 헤더들 내에 포함되도록 더 설정된 것을 특징으로 하는 FEC를 수행하기 위한 장치.

**청구항 11**

제 9항에 있어서,

소스 패킷들의  $x$  개의 부분집합(Subset)을 가진 상기 인터리브된 모드에서 상기 SPC AL-FEC 코드를 적용할 때, 상기 컨트롤러는

각 패리티 패킷은 소스 패킷들의  $x$  개의 부분 집합들 중 하나에 대응하고,  $x$  개의 패리티 패킷들을 위해 시퀀스 번호들  $i, i+1, i+2, \dots, i+(x-1)$ 을 생성하고,

상기  $k$  개의 소스 패킷들을 위해 시퀀스 번호들  $i+1, i+2, \dots, i+k$ 를 생성하고,

연속적인 시퀀스 번호들을 가진 소스 패킷들이 다른 부분집합들에 할당되도록  $x$  개의 부분집합들에 걸쳐 상기  $k$  개의 소스 패킷들을 인터리브(Interleave)하며,

짧은 소스 패킷들에는 임시적으로 송신되지 않을 가상의 0들(Virtual Zeros)을 덧붙이고(Padded), 상기 대응하는 소스 패킷들의 부분집합에 비트단위 배타적 논리합을 수행함으로써 각 패리티 패킷의 페이로드를 계산하고,

상기 패리티 패킷들을 위한 상기 시퀀스 번호들은 상기 대응하는 패리티 패킷들의 상기 헤더들 내에 포함되고, 상기  $k$  소스 패킷들을 위한 상기 시퀀스 번호들은 상기 대응하는 시스터매틱 패킷들의 상기 헤더들 내에 포함되도록 더 설정되는 것을 특징으로 하는 FEC를 수행하기 위한 장치.

**청구항 12**

제 8항에 있어서,

상기 컨트롤러는,

상기 소스 패킷들 및 상기 적어도 하나의 패리티 패킷을 위해 상기 시스터매틱 패킷들을 인코딩 할 때, 상기 SPC AL-FEC 코드와 결합하여 해밍(Hamming) AL-FEC 코드를 수행하도록 더 설정되는 것을 특징으로 하는 FEC를 수행하기 위한 장치.

**청구항 13**

제 8항에 있어서,

상기 컨트롤러는,

시스터매티 패킷들 및 수리(Repair) 패킷들 중 적어도 하나를 포함하는 패킷 블록을 수신하고, 수신된 패킷 블록 내에 적어도 하나의 패킷이 없어졌는지(Missing) 여부를 판단하고, 어떤(Any) 패킷이 없어진 경우, 각 수신된 패킷 내의 상기 패리티 패킷 지시자를 검사함으로써, 상기 수신된 패킷들이 AL-FEC 인코딩되었는지 여부를 판단하고, 상기 AL-FEC 코드가 감지된 경우, 상기 없어진 패킷들이 복구될 수 있는지 여부를 판단하고, 상기 없어진 패킷들이 복구될 수 있는지의 판단에 대한 대응으로 상기 수신된 시스터매티 및 수리 패킷들을 사용하여 AL-FEC 디코딩(Decoding)을 수행하도록 더 설정된 것을 특징으로 하는 FEC를 수행하기 위한 장치.

#### 청구항 14

제 13항에 있어서,

상기 컨트롤러는,

상기 수신된 패킷들이 AL-FEC 인코딩되었는지 여부를 판단할 때 수신된 패리티 패킷들의 숫자에 기초하여 상기 수신된 패킷들이 인터리브 되었는지(Interleaved) 여부를 결정하도록 더 설정되는 것을 특징으로 하는 FEC를 수행하기 위한 장치.

#### 청구항 15

무선 통신 네트워크(Network) 내의 무선 통신 디바이스(Device)의 순방향 오류 정정(Forward Error Correction: FEC)을 수행하는 방법에 있어서,

성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하는 단계;

각 패리티 패킷의 헤더는 패리티 패킷(Parity Packet) 지시자(indicator)를 포함하고, 소스 패킷(Source Packet)들 및 적어도 하나의 수리 패킷을 위해 시스터매티(Systematic) 패킷들을 인코딩하기 위해서  $k$  개의 상기 소스 패킷들의 집합에 단일 패리티 검사(Single Parity Check: SPC) AL-FEC 코드를 적용하는 단계; 및

미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 패킷을 처리하는 단계를 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행 방법.

#### 청구항 16

제 15항에 있어서,

상기 AL-FEC 코드는

SPC, 컨볼루션 코드(Convolutional Code) 및 해밍 코드(Hamming Code) 중 적어도 하나를 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행 방법.

#### 청구항 17

제 15항에 있어서,

상기 SPC AL-FEC 코드는

상기 소스 패킷들을 인터리브(Interleave) 하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행 방법.

#### 청구항 18

제 15 항에 있어서,

상기 AL-FEC 코드는,

단일 패리티 검사와 해밍 코드를 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행 방법.

#### 청구항 19

제 15항에 있어서,

시스터매틱 패킷들 및 수리(Repair) 패킷들 중 적어도 하나를 포함하는 패킷 블록을 수신하는 단계;

상기 수신된 패킷 블록 내에 적어도 하나의 패킷이 없어졌는지(Missing) 여부를 판단하는 단계;

적어도 하나의 패킷이 없어진 경우, 각 수신된 패킷 내의 상기 패리티 패킷 지시자를 검사함으로써, 상기 수신된 패킷들이 AL-FEC 인코딩되었는지 여부를 판단하는 단계;

상기 AL-FEC 코드가 검색된 경우, 상기 적어도 하나의 없어진 패킷이 복구될 수 있는지 여부를 판단하는 단계; 및

상기 적어도 하나의 없어진 패킷을 복구될 수 있는지의 판단에 대한 대응으로 상기 수신된 시스터매틱 및 수리 패킷들을 사용하여 AL-FEC 디코딩(Decoding)을 수행하는 단계를 더 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

## 청구항 20

제 19 항에 있어서,

상기 수신된 패킷들이 AL-FEC 인코딩되었는지 여부를 판단하는 단계는,

수신된 패리티 패킷들의 숫자에 기초하여 상기 수신된 패킷들이 인터리브 되었는지(Interleaved) 여부를 결정하는 단계를 더 포함하는 것을 특징으로 하는 무선 통신 디바이스의 FEC 수행방법.

## 명세서

### 기술 분야

[0001]

본 발명은 일반적인 무선 연결들을 통한 신뢰할 수 있는 데이터 (Reliable data)를 전송하는 것에 관한 것으로, 특히 무선 기가 비트 얼라이언트(Wireless Gigabit Alliance: WiGig) 링크(link)를 통해 통신하기 위한 응용 계층(Application Layer)에서의 순방향 에러 정정(Forward Error Correction: FEC) 프레임워크를 구현하기 위한 방법 및 시스템에 관한 것이다.

### 배경 기술

[0002]

무선 통신들을 통한 신뢰할 수 있는 데이터 전송을 위해 많은 요구가 엔지니어들에게 큰 도전을 제시하고 있다. 일반적으로, 비디오 미디어는 지연(delay)이 허용되지 않는 높은 전송 레이트(Rate)를 요구한다. 게다가, 미디어는 스트리밍 TV 채널들과 같은 브로드캐스트(Broadcast) 채널(Channel) 또는 많은 사용자들을 위해 대용량 파일(Large File)로 압축된 비디오를 송신하는 것과 같은 멀티캐스트 채널(Multi cast)을 통해 전송된다. 이러한 채널의 성격은 딜레이가 허용되지 않을 뿐만 아니라, 전송 성공을 알리기 위한 피드백 채널(Feedback Channel)을 가지는 옵션이 제외된다. 결과적으로, 이러한 시나리오에서, 전송의 신뢰성을 보정하기 위해 대안적인 기술이 요구된다.

[0003]

무선 기가비트 얼라이언스(Wireless Gigabit Alliance: WiGig) 규격(Specification)은 멀티-기가비트(Multi-Gigabit) 속도 무선 통신 기술을 대상으로 한다. 이처럼, WiGig는 오늘날의 무선 로컬 영역 네트워크(Local Area Network: LAN) 디바이스들(Devices)의 성능들을 보충하는 고성능 무선 데이터, 디스플레이 및 오디오 어플리케이션들을 가능하게 한다. WiGig의 기술 규격 TWG-2010-0716-00-WGA-D102 및 PWG-2011-0019-00-AV PAL spec D0.9r0: "WiGig WGA Specification"에 개시되어 있고, 이는 참조로서 본 명세서에 포함된다.

[0004]

그러나, WiGig 규격은 브로드 캐스트, 멀티 캐스트 동안 자동적 반복 요청(Automatic Repeat Request: ARQ) 방식의 사용을 허용하지 않는다. 게다가, 시간에 민감한 어플리케이션들(예를 들면, 멀티미디어(Multimedia), 게임(Gaming), 등등)내에서, 특히, 채널이 긴 중단들(Outages) 및 장애(Blockage) 및 상대적으로 느린 빔포밍 알고리즘에 의한 높은 패킷 손실율(Packet Loss Rate)을 겪는 때는 ARQ는 가장 효과적인 오류 제어 방식이 아니다. ARQ 피드백(feedback)의 부재로 물리 계층 순방향 오류 정정(Physical Layer Forward Error Correcting: PHY FEC) 코드는 낮은 패킷 손실율(대략  $10^{-5}$ )을 성취하기 위해 충분한 보호를 제공받지 못할 수 있다. 이처럼, 패킷 손실율을 줄이기 위해 제 2의 FEC 방식이 필요하다.

## 발명의 내용

### 해결하려는 과제

[0005] 본 발명의 일 실시예에서는, 무선 통신 네트워크 내의 무선 통신 디바이스에서 순방향 오류 정정을 수행하는 방법을 제공하므로써, 낮은 페킷 손실율(대략  $10^{-5}$ )을 성취하면서도 충분한 보호를 받도록 한다.

### 과제의 해결 수단

[0006] 일 실시예로서, 무선 통신 네트워크(Network) 내의 무선 통신 디바이스(Device)의 순방향 오류 정정(Forward Error Correction: FEC)을 수행하는 방법이 제공된다. 무선 통신 네트워크 내의 무선 통신 디바이스의 순방향 오류 정정을 수행하는 방법은 성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하는 단계를 포함한다. 소스 페킷(Source Packet)들 및 적어도 하나의 페리티 페킷을 위해 시스터매틱(Systematic) 페킷들을 인코딩하기 위하여 k 개의 상기 소스 페킷들의 집합에 단일 페리티 검사(Single Parity Check: SPC) AL-FEC 코드(Code)를 적용한다. 각 페리티 페킷의 헤더는 페리티 페킷(Parity Packet) 지시자(indicator)를 포함한다. 미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 페킷을 처리한다.

[0007] 다른 실시예로서, 무선 통신 네트워크(Network) 내의 무선 통신 디바이스의 순방향 오류 정정을 수행하기 위한 장치가 제공된다. 무선 통신 네트워크내의 무선 통신 디바이스의 순방향 오류 정정을 수행하기 위한 장치는 송신기 및 송신기와 결합된 컨트롤러(Controller)를 포함한다. 송신기는 다른 통신 디바이스들과 통신한다. 컨트롤러는 성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하고, 소스 페킷(Source Packet)들 및 적어도 하나의 페리티 페킷을 위해 시스터매틱(Systematic) 페킷들을 인코딩하기 위하여 k 개의 상기 소스 페킷들의 집합에 단일 페리티 검사(Single Parity Check: SPC) AL-FEC 코드(Code)를 적용하고, 인코딩된 페킷의 헤더는 페리티 페킷(Parity Packet) 지시자(indicator)를 포함하고, 미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 페킷을 처리한다.

[0008] 또 다른 실시예로서, 무선 통신 네트워크(Network) 내의 무선 통신 디바이스(Device)의 순방향 오류 정정(Forward Error Correction: FEC)을 수행하는 방법이 제공된다. 무선 통신 네트워크 내의 무선 통신 디바이스의 순방향 오류 정정을 수행하는 방법은 성능들(Capabilities) 교환 동안 응용 계층 순방향 오류 정정(Application Forward Error Correction: AL-FEC) 성능 정보를 전송하는 것을 포함한다. 소스 페킷(Source Packet)들 및 적어도 하나의 페리티 페킷을 위해 시스터매틱(Systematic) 페킷들을 인코딩하기 위하여 k 개의 상기 소스 페킷들의 집합에 단일 페리티 검사(Single Parity Check: SPC) AL-FEC 코드를 적용한다. 각 페리티 페킷의 헤더는 페리티 페킷(Parity Packet) 지시자(indicator)를 포함한다. 그리고 미디어 접속 제어(Media Access Control: MAC) 계층 및 물리(Physical: PHY) 계층에서 송신을 위해 상기 인코딩된 페킷을 처리한다.

[0009] 아래의 발명의 상세한 설명을 읽기 전에 본 특허 명세서 전체에서 사용되는 특정 단어들 및 구문들의 정의들을 말하는 것이 도움이 될 것이다. 용어 "포함하다(Include, Comprise)" 및 파생되는 단어들은 제한없이 포함된다는 뜻이며, 용어 "또는(Or)"은 및/또는(And/Or)의 뜻을 포함한다. "관련된(Associated with)" 및 "그와 관련된(Associated therewith)" 및 파생되는 단어들은 포함하기 위해(To include), 포함된(Be included within), 관련된(Interconnect with), 들어있다(Contain), 들어 있는(be contained within), 이어진(Connect to or with), 연결된(Coupled to or with), 전달되는(Be communicable with), 협력하는(Cooperate with), 인터리브(interlave), 나란히 놓다(Juxtapose), 근접된(Be Proximate to), 결합된(Be bound to or with), 관계가 있다(Have), 특성을 보유하다(Have a Property of)와 같은 뜻일 수 있다. 또한 용어 "컨트롤러(Controller)"는 어떤 디바이스, 시스템 또는 그것들의 부분들의 적어도 하나의 동작을 제어하는 것을 뜻하며, 이러한 디바이스는 하드웨어(Hardware), 펌웨어(Firmware) 및 소프트웨어(Software) 또는 위와 같은 것들 중 적어도 두 개의 조합으로 구현될 수 있다. 특정한 컨트롤러와 관련된 기능들은 로컬(Locally)이든지, 원격이든지, 집중적 또는 분산적일 수 있음을 유의하여야 한다. 당해 기술분야에서 통상의 지식을 가진 자라면, 많은 또는 대부분의 경우에서, 본 특허 명세서 전체에서 제공되는 특정 단어들 및 문구들을 위한 정의들을 이해할 수 있으며, 이러한 정의들은 이전뿐만 아니라 이후의 사용에서, 정의된 단어들 및 문구들로 적용될 수 있다.

## 발명의 효과

[0010] 본 발명의 일 실시예에 따르면, 외부(Outer) FEC 방식은 허용 수준까지 패킷 손실율을 대폭 줄일 수 있다.

### 도면의 간단한 설명

[0011] 본 발명에 개시된 내용 및 효과에 대한 더 완전한 이해를 위해, 이하의 상세한 설명은 이하의 동반되는 도면들과 함께 참조되어야 하며, 동일한 참조 번호는 동일한 구성요소를 나타낸다.

도 1은 본 명세서의 원칙들에 따라 WiGig를 지원하는 무선 네트워크(100)을 도시한다.

도 2A는 본 명세서의 일 실시예에 따라 직교 주파수 분할 다중 접속 송신 경로(Orthogonal Frequency Multiple Access Transmit Path)의 상위 레벨(High-Level)의 다이어그램(Diagram)을 도시한다.

도 2B는 본 명세서의 일 실시예에 따라 직교 주파수 분할 다중 접속 수신 경로(Orthogonal Frequency Multiple Access Receive Path)의 상위 레벨(High-Level)의 다이어그램(Diagram)을 도시한다.

도 3은 본 명세서의 일 실시예에 따라 응용 계층 순방향 오류 정정(Application Layer-Forward Error Correction: AL-FEC) 코딩(Coding) 개념(Concept)을 도시한다.

도 4는 본 명세서의 일 실시예에 따라 AL-FEC 방식을 도시한다.

도 5A는 본 발명의 일 실시예에 따라 비-인터리브된 단일 패리티 검사 (Non-interleaved Single Parity Check) 방식을 도시한다.

도 5B는 본 발명의 일 실시예에 따라 인터리브된 단일 패리티 검사 (Interleaved Single Parity Check) 방식을 도시한다.

도 6은 본 명세서의 일 실시예에 따라 삭제된 패킷들을 복구하기 위한 컨볼루션 코드(Convolutional Code)를 도시한다.

도 7은 본 명세서의 일 실시예에 따라, AL-FEC를 사용하여 인코딩된 패킷의 헤더의 내용을 도시한다.

도 8은 본 명세서의 일 실시예에 따라, 무선 통신 시스템 내에서 AL-FEC의 사용을 위한 일반적은 프로세스 (Process)를 도시한다.

도 9는 본 명세서의 일 실시예에 따라 AL-FEC를 사용하여 송신하기 위해 인코딩 패킷들을 위한 프로세스를 도시한다.

도 10은 본 명세서의 일 실시예에 따라 AL-FEC를 사용하는 디코딩 패킷들을 위한 프로세스를 도시한다.

### 발명을 실시하기 위한 구체적인 내용

[0012] 이하의 도 1부터 도 10 및 본 특허 명세서 내의 본 명세서의 원칙들을 서술하기 위해 사용되는 다양한 실시예들은 단지 예시의 방식일 뿐이며, 본 명세서의 범위를 제한하는 어떠한 방식으로도 해석될 수 없다. 비록 기술된 본 발명의 실시예들이 WiGig를 지원하는 장치들을 나타낼 수 있으나, 본 명세서의 원칙들이 무선 통신 시스템의 적합한 배열에 의해 구현될 수 있음을 당해 기술분야에서 이해될 수 있다.

도 1은 본 명세서의 원칙에 따라 와이 기그(Wireless Gigabit Alliance: WiGig)를 지원하는 무선 네트워크 (100)를 도시한다. 도시된 예에서, 무선 네트워크(100)은 액세스 노드(Access Node: AN)(102), 모바일 노드 (Mobile Node)(104), 디스플레이 디바이스(106), 퍼스널 컴퓨터(Personal Computer: PC)(108) 및 카메라(11)을 포함한다. AN(102)는 라우터(Router), 베이스 스테이션(Base Station), 케이블 박스(Cable Box), 또는 데이터 서비스를 수신하는 어떤 장치일 수 있다. 모바일 디바이스(104)는 셀룰러 폰(Cell Phone), 개인용 휴대 단말기 (Personal Digital Assistants), 태블릿 디바이스(Tablet Device), 전자 리더(Electronic Reader) 등과 같은 어떤 휴대용(Portable) 디바이스 일 수 있다. 디스플레이 디바이스(106)는 프로젝터(Projector), 텔레비전 (Television), 컴퓨터 디스플레이(Computer Display), 스테레오 수신기(Stereo Receiver) 등과 같은 출력 디바이스일 수 있다. 카메라(110)는 프린터(Printer), 캠코더(Camcorder), 웹 캠(Webcam), 스캐너(Scanning Device), 의료용 영상장치(Medical Imaging Apparatus)등과 같은 주변(Peripheral) 디바이스일 수 있다. 모든 디바이스들(102-110)이 WiGig를 지원한다고 가정하면, 각각의 디바이스들(102-110)은 적어도 하나의 다른 디바이스들과 WiGig 연결을 형성할 수 있다. 예를 들면, PC(108)는 오디오/비디오(Audio/Video: A/V) 데이터를 디스플레이 디바이스(106)에게 송신할 수 있고, 카메라(또는 다른 주변 디바이스)(110)으로부터 데이터를 수신할 수 있으며, 모바일 디바이스(104)와 동기화 작업(Sync Operation)을 수행하고, AN(102)을 통해 인터넷에 접속할 수

있다.

[0014] 도 2A는 본 명세서의 일 실시예에 따라 직교 주파수 분할 다중 접속 (Orthogonal Frequency Multiple Access: OFDMA) 송신 경로의 상위 레벨(High-Level)의 다이어그램(Diagram)을 도시한다. 도 2A 및 도 2B에서 OFDMA 송신 경로(200) 및 OFDMA 수신 경로(250)은 무선 통신 디바이스의 송신기 및 수신기 내에 각각 구현된다. 일부의 실시예들에서 송신 경로(200) 및 수신 경로(250)의 요소들은 하나의 트랜시버(transceiver) 내에서 결합될 수 있다.

[0015] 송신 경로(200)는 채널 코딩(Channel Coding) 및 변조(Modulation) 블록(205), 직렬-병렬 변환(Serial to Parallel)(S-to-P) 블록(210), N 크기의 역 고속 푸리에 변환(Inverse Fast Fourier Transform: IFFT) 블록(215), 병렬-직렬 변환(P-to-S) 블록(220), 주기적 전치부호(Cyclic Prefix) 삽입 블록(225), 업-컨버터(Up-Converter: UC)(230)을 포함한다. 수신 경로(250)은 다운-컨버터(Down Converter: DC)(255), 주기적 전치부호 제거 블록(260), 직렬-병렬 변환(S-to-P) 블록(265), N 크기의 고속 푸리에 변환(Fast Fourier Transform: FFT)(270), 병렬-직렬 변환(P-to-S) 블록(275), 채널 디코딩 및 복조 블록(280)을 포함한다.

[0016] 도 2A 및 도 2B 내의 구성요소들 중 최소한의 일부는 소프트웨어(Software)로 구성되는 반면에, 다른 구성요소들은 설정 가능한 하드웨어(Configurable Hardware) 또는 소프트웨어 및 설정 가능한 하드웨어의 혼합에 의해 구현된다. 특히, 본 명세서에 기재된 FFT 블록들 및 IFFT 블록들은 설정 가능한 소프트웨어 알고리즘들에 의해 구현될 수 있고, 크기 N은 구현에 따라 수정될 수 있다. 또한, 무선 통신 디바이스는 도 2A 및 도 2B 내의 경로를 구현하도록 설정된 적어도 하나의 컨트롤러들(Controller)을 포함할 수 있다.

[0017] 추가적으로, 비록 본 명세서가 FFT 및 IFFT를 구현하는 실시예들을 대상으로 하고 있으나, 단지 예시의 방식일 뿐이며. 본 명세서의 범위를 제한하는 어떠한 방식으로도 해석될 수 없다. 본 명세서의 대안적인 실시예에서, FFT 함수 및 IFFT 함수는 각각 이산 푸리에 변환(Discrete Fourier Transform: DFT) 및 역 이산 푸리에 변환 (Inverse Discrete Fourier Transform: IDFT)로 쉽게 대체될 수 있음을 확인할 수 있을 것이다. DFT 및 IDFT 함수들을 위해 N 변수의 값은 정수(예를 들면, 1, 2, 3, 4, 등등)일 수 있고, 반면에 FFT 및 IFFT 함수들을 위해서 N 변수의 값은 2의 지수(예를 들면 1, 2, 4, 8, 16, 등등)가 될 수 있음이 확인할 수 있을 것이다.

[0018] 더욱이, 다른 실시예들에서, 송신 경로 및 수신 경로는 공통 요소들(프리엠블(Preamble) 및 채널과 같은)을 공유하는 변조 및 코딩 방식들의 다른 타입들을 사용할 수 있다. 예를 들면, 일 실시예는 작고, 저전력 휴대용 기기를 위한 저전력 소모를 유발하는 단일 캐리어를 지원할 수 있다.

[0019] 송신 경로(200)에서, 채널 코딩 및 변조 블록(205)은 주파수-도메인 변조 심볼들의 시퀀스(Sequence)를 생산하기 위한 입력 비트들인 코딩(즉, 저밀도 패리티 검사(Low Density Parity Check: LDPC) 코딩) 및 변조(즉, QPSK, QAM)가 적용된 정보 비트들의 집합을 수신한다. 직렬-병렬 변환 블록(210)은 변조된 직렬 심볼들을 N개의 병렬 심볼 스트림들을 생성하기 위해 병렬 데이터로 변환(즉, 역다중화(De-multiplexing))하며, N은 무선 통신 디바이스들에서 사용되는 IFFT/FFT의 크기이다. 그 후 N 크기의 IFFT 블록(215)은 시간-도메인 출력 신호들을 생성하기 위해 N개의 병렬 심볼 스트림들에 IFFT 연산을 수행한다. 병렬-직렬 변환 블록(220)은 N 크기의 IFFT 블록(215)로부터 직렬 시간-도메인 신호를 생성하기 위해 병렬 시간-도메인 출력 심볼들을 변환(즉, 다중화(Multiplexing))한다. 그 후 주기적 전치 부호(Cyclic Prefix) 삽입 블록(225)은 시간-도메인 신호에 주기적 전치 부호를 입한다. 마지막으로 무선 채널을 통해 전송하기 위해 업-컨버터(Up-Converter)(230)은 주기적 전치 부호 삽입 블록(225)의 출력을 RF 주파수로 변조(즉, 업-컨버트(Up-Converts))한다. 신호는 RF 주파수로 변환되기 전 베이스밴드(Baseband)에서 필터링(Filtered) 될 수 있다.

[0020] 전송된 RF 신호는 무선 채널을 지난 후 수신하는 디바이스의 수신 경로(250)에 도달한다. 다운-컨버터(Down-Converter)(255)는 수신된 신호를 베이스밴드 주파수로 다운-컨버트(Down-Converts)하고, 주기적 전치 부호 제거 블록(260)은 직렬 시간-도메인 베이스밴드 신호를 생성하기 위해 주기적 전치 부호를 제거한다. 직렬-병렬 변환 블록(265)은 시간-도메인 베이스밴드 신호를 병렬 시간 도메인 신호들로 변환한다. 그 후, N 크기의 FFT 블록(270)은 N개의 병렬 주파수-도메인 신호들을 생성하기 위해 FFT 알고리즘(Algorithm)을 수행한다. 병렬-직렬 변환 블록(275)은 병렬 주파수-도메인 신호들을 변조된 데이터 심볼들의 시퀀스(Sequence)로 변환한다. 채널 디코딩 및 복조 블록(280)은 변조된 심볼들을 원본의 입력 데이터 스트림을 복구하기 위해 복조한 후 디코딩 한다.

[0021] 오디오 및 비디오의 전송은 WiGig의 어플리케이션들에서 중요하다. 그러나, 좋은 비디오 품질을 위해서 패킷 손실율은  $10^{-5}$  보다 작아야 한다. 시간에 민감한 어플리케이션들(예를 들면, 멀티미디어(Multimedia), 게임 및

등등)에서, 지연 제약들(Delay Constraints)은 시스템의 성능을 제약한다. 시간에 민감한 어플리케이션에서, 재전송(Retransmission)을 위해 필요한 동적 버퍼(Dynamic Buffer)는 구현에 상당한 복잡도를 추가한다. 그리고 전술한 바와 같이, WiGig의 미디어 접속 제어(Media Access Control: MAC) 계층은 브로드캐스트/멀티캐스트 패킷들을 위한 확인들(Acknowledgements: ACKs)을 허용하지 않는다. 그러므로, 적은 손실들, 고정된 버퍼와 간단한 방식, 재전송 없음(No Retransmission) 및  $10^{-5}$ 보다 작은 패킷 손실율을 용인할 수 있는 어플리케이션들이 바람직하다.

[0022] 순방향 오류 제어(Forward Error Control: FEC) 방식은 이러한 시나리오들에서 적합한 후보들에 해당한다. 실제로, FEC는 채널의 통신들에서 사용되고 있다. WiGig 60GHz 통신들에서, 저밀도 패리티 검사(Low-Density Parity Check: LDPC) 코드(Codes) FEC 방식은 신뢰할만한 통신을 성취하기 위해 적용될 수 있다. LDPC 코드는 시스템의 물리 계층(Physical Layer)의 부분이고, 1/2, 5/8, 3/4 및 13/16의 코딩 레이트(Coding Rate)들을 가진다. 이러한 코드는 비트 레벨(Bit level)에서의 보호의 좋은 레벨을 달성할 수 있고, 일반적으로  $10^{-6}$ 보다 낮은 비트 오류율(Bit Error Rate: BER)을 획득할 수 있다. 그러나, 이러한 FEC 방식을 사용하여 성취가능한 패킷 손실율은 약 0.01이다. 결과적으로, 외부(Outer) FEC 방식은 패킷 손실율을 허용수준(대략  $10^{-5}$ )으로 상당히 줄일 수 있다. 특히, 응용 계층(FEC(Application Layer: AL-FEC) 코드는 그들의 동적 속도, 낮은 오버헤드(Low-Overhead) 및 구현 비용 측면에서 가장 적합한 후보이다. 본 명세서에서, AL-FEC는 MAC 계층 위의 모든 기능적 계층 위에서 수행되는 FEC를 참조할 수 있다.

[0023] 도 3은 본 명세서의 일 실시예에 따라 AL-FEC 코딩(Coding) 개념(Concept)을 도시한다. 송신기는 송신될  $n$  개의 패킷들의 블록을 생성하기 위해  $k$  개의 소스 패킷들에 AL-FEC 인코딩을 수행한다.  $n$  개의 패킷들은 AL-FEC 인코딩 연산을 통해서 생성된  $k$  개의 소스 패킷들 및  $r$  개의 수리 패킷(Repair Packet)들을 포함한다.  $n$  개의 패킷들이 전송된 후, 수신기는  $n$  패킷들 중  $m$  개를 수신하고,  $m$ 은 대략  $k$ 와 동일하다. 수신된  $m$  개의 패킷들은 없어진(Missing) 소스 패킷들을 복구하기 위해 디코딩된다. 오버헤드는  $m-k = 0$ 이고, 생성된 패킷들의 숫자는 식 1에 의해 계산된다.

[0024] [식 1]

$$n = m / (1 - \text{심볼 손실율}(\text{symbol loss rate}))$$

[0026] 이처럼, 심볼 삭제 채널 캐패시티(Symbol Erasure Channel Capacity)에 근접하게 시스템이 운영되기 때문에 오버헤드가 작다.

[0027] 이상적으로, AL-FEC 코드는, 다음과 같은 특징을 갖는 코드다.

[0028] 1) 레이트리스 코드(Rateless Codes): 원본 데이터로부터 무한한 숫자의 인코딩된 패킷들(심볼은 일반적으로 패킷이다)을 생성할 수 있다.

[0029] 2) 오버헤드를 넣지 않음(Introduce no Overhead): 수신기는  $k$  개의 인코딩된 패킷들이 수신된 후  $k$  패킷들을 포함하는 메시지를 재구성할 수 있다.

[0030] 3) 단순함(Simple): 인코딩/디코딩은 매우 빨라야하며, 시간에 선형(Linear in Time)인 것이 선호된다.

[0031] 이러한 코드의 근사적인 실현들은 매트릭스 코드(Matrix Codes), 리드-솔로몬 코드(Reed Solomon Codes), 루비-트랜스폼(Luby-Transform: LT) 코드 및 랩터 코드(Raptor Codes)이다. 랩터 코드는 AL-FEC 코드의 최고의 종합적인 근사치가 될 수 있다.

[0032] AL-FEC 프레임 워크(Framework)의 실시예들은 WiGig 내의 프로토콜 어댑션 계층(Protocol Adaption Layer: PAL)의 임의적인 흐름(Arbitrary Flows)(예를 들면, 오디오, 압축된 비디오, 비압축된 비디오 등등)에 패킷-레벨(Packet-Level) 보호를 제공한다. AL-FEC 프레임워크는 어떤 FEC 코드(삭제 코드(Erasure Code))와 함께 동작할 수 있으나, 삭제 코드는 시스템마티(Systematic)으로 가정한다. 일 실시예에서, AL-FEC 프레임워크는 랩터 코드를 지원하기 위해 설정된다.

[0033] 도 4는 본 명세서의 일 실시예에 따라 AL-FEC 방식을 도시한다. 하이픈으로 연결된 수직선이 송신 디바이스(400) 및 수신 디바이스(450)의 기능적 계층들을 분리한다. 접선은 송신 디바이스(400)로부터 수신 디바이스(450)으로 전송되는 데이터를 표시한다. AL-FEC 프레임워크와 관련하여, 송신 디바이스(400)는 물리(Physical: PHY) 계층(440), 미디어 접속 제어(Media Access Control: MAC) 계층(430), 프로토콜 어댑션 계층(Protocol Adaption Layer: PAL)(410) 및 AL-FEC 컴포넌트(Component)(420)를 포함한다. 유사하게도, 수신 디바이스(450)는

0)은 PHY 계층(490), MAC 계층(480), PAL(460) 및 AL-FEC 컴포넌트(Component)(470)를 포함한다. 송신 디바이스(400) 및 수신 디바이스(450) 각각은 WiGig가 가능한 디바이스들(102-110)과 같은 어떤 무선 통신 디바이스들 중 하나일 수 있다.

[0034] 본 명세서의 일 실시예에 따른, AL-FEC 방식에서 송신 디바이스(400)의 일반적인 연산은 이하의 내용과 같다. 송신 디바이스(400)에서, 함께 보호되어야 할 소스 패킷(Source Packet)들의 집합이 특정된다. AL-FEC 코드는 수리 패킷(Repair Packet)들의 집합을 생성하기 위해 소스 패킷에 적용된다. 수리 패킷이 생성된 이후, 송신 디바이스(400)은 소스 패킷들과 수리 패킷들을 수신 디바이스(450)에 송신한다.

[0035] 본 명세서의 일 실시예에 따른, AL-FEC 방식에서 수신 디바이스(450)의 일반적인 연산은 이하의 내용과 같다. 만약 모든 소스 패킷들이 수신 디바이스(450)에서 성공적으로 수신되면, 수신된 소스 패킷들은 AL-FEC 복구 없이 처리되고, 수신된 복구 패킷들은 폐기된다. 반대로, 만약 없어진 패킷들이 있다면, 없어진 패킷들을 복구하기 위해 성공적으로 수신된 소스 및 복구 패킷들에 AL-FEC 방식이 적용될 것이다.

[0036] 송신 디바이스(400)의 프로토콜 어댑션 계층(Protocol Adaption Layer: PAL)(410)은 상위 계층(Upper Layer)으로부터(미도시) 소스 데이터(즉, 응용 패킷들)을 수신하고, 송신될 패킷들을 준비하고, MAC 계층(430)으로 패킷들을 송신한다. 이를 위해, PAL(410)은 수신 디바이스(450)(T x m 바이트의 버퍼(Buffer) 크기를 갖는) PAL(460)이 소스 블록을 처리할 수 있도록 적합한 소스 블록 크기(k 개의 패킷들)을 선택함으로서, 보호될 소스 패킷들의 집합을 첫번째로 특정할 수 있다.

[0037] PAL(410) 소스 블록 숫자(Source Block Number: SBN)을 생성하고, 소스 블록, SBN, k, 및 T를 AL-FEC 컴포넌트(420)로 송신한다. PAL(410)은 또한 소스 패킷의 페이로드(Payload)에 헤더(header) 정보를 첨부함으로써 소스 패킷들을 구성하고, AL-FEC 컴포넌트(420)으로부터 수신된 소스 패킷들 및 수리 패킷들을 MAC 계층(430)에 송신한다. PAL(410) 및 MAC 계층(430)의 기능들(Functions)은 무선 통신 디바이스의 프로세서(Processor) 또는 컨트롤러(Controller)에 의해 수행될 수 있다.

[0038] 수신 디바이스(450)에서, PAL(460)은 MAC 계층으로부터 패킷들(소스 패킷들 및 수리 패킷들)을 수신하고, 소스 데이터를 복구하고, 소스 데이터를 상위 기능성 계층(Upper Functional Layer)(미도시)에 송신한다. 이를 위하여, PAL(460)은 모든 소스 패킷들이 성공적으로 수신되었는지 아닌지를 판단한다. 만약 모든 소스 패킷들이 성공적으로 수신되었으면, 소스 패킷들은 AL-FEC 복구 없이 처리되고, 수리 패킷들은 폐기된다. 반대로, 만약 없어진 패킷들이 있다면, PAL(460)은 AL-FEC 컴포넌트(470)에게 수신된 소스 및 수신 패킷들을 송신한다. PAL(460)은 AL-FEC 컴포넌트(470)으로부터 복수된 소스 패킷들을 획득한다.

[0039] 일 실시예로서, 본 명세서에서 서술된 PAL(410 및 460)의 모든 기능들은 MAC 계층들(430 및 480) 위의 다른 계층들에서 각각 수행될 수 있다. PAL들(410 및 460)은 WiGig를 위해 정의될 수 있는 데이터 및 디스플레이 기준들(Standards)과 같은 특정한 기준들(Standards)을 지원한다. 예를 들면, PAL들(410 및 460)은 고화질 멀티미디어 인터페이스(High-Definition Multimedia Interface: HDMI) 및 디스플레이 포트(DisplayProt)를 지원을 정의할 수 있다. 다른 실시예로서, PAL들(410 및 460)은 입/출력(Input/Output: I/O) 데이터가 될 수 있고 및 유니버설 시리얼 버스(Universal Serial Bus: USB) 지원할 수 있고, PCI 익스프레스(Peripheral Component Interconnect Express: PCIe)가 될 수 있다.

[0040] 송신 디바이스(400)의 AL-FEC 컴포넌트(420)는 수리 패킷들을 생성하기 위해 소스 데이터에 AL-FEC 코드를 적용한다. 다른 대안적인 실시예에서, 수리 패킷들은 PAL(410)에 의해 생성될 수 있다. 수신 디바이스(450)에서 AL-FEC 컴포넌트(470)은 없어진 패킷들을 복구하기 위해 AL-FEC 방식을 사용하여 PAL로부터 수신된 소스 및 복구 패킷들을 디코딩한다. 일부 실시예들에서, AL-FEC 컴포넌트들(420 및 470) 또는 AL-FEC 컴포넌트들(420 및 470)의 기능들은 각각 PAL들(410 및 460) 내에 통합될 수 있다. 또한 일부 실시예들에서, 송신 디바이스(400) 및 수신 디바이스(450)의 각 PAL들(410 및 460), AL-FEC 컴포넌트들(420 및 470), MAC 계층들(430 및 480), 및 PHY 계층들(440 및 490) 각각은 송신 및 수신과 관련된 기능들을 수행하기 위해 설정될 수 있다.

[0041] PHY 계층(440 및 490)은 하드웨어 통신 기능들을 참조한다. 다시 말하면, PHY 계층들(440 및 490)은 데이터의 로우(Raw) 비트들의 송신 및 수신의 수단을 정의한다. 일 실시예에서, FEC 방식은 송신 디바이스(400)의 PHY 계층(440)에 의해, 전송 중의 수신 디바이스(450)의 PHY 계층(460)에 의해 수행된다. PHY 계층의 FEC는 저밀도 패리티 검사(Low-Density Parity Check: LDPC) 코드 FEC 방식일 수 있다. MAC 계층들(430 및 480)은 당해 기술분야에서 잘 알려진 어드레싱(addressing) 및 채널 접속 제어 기능들을 수행한다.

[0042] 아래의 실시예들은 랙터 코드 파라미터(Parameter)들을 설명하며, 기가비트(Gigabit) 무선 통신을 위한 실시예

들이다. 랙터 코드 파라미터들은 "k = 1024"의 소스 블록 크기 및 심볼 길이 T = "48" 바이트를 포함할 수 있다. 이처럼, 소스 블록 수신을 위한 버퍼의 필요조건은 대략 "49"킬로 바이트(KiloByte:KB)이다. AL-FEC 이후의 패킷 손실율은 아래의 식 2을 사용하여 계산될 수 있다.

[0043] [식 2]

[0044] AL-FEC 이후의 패킷 손실율 = (AL-FEC 이전의 패킷 손실율) x (블록-복구 실패율)

[0045]  $10^{-3}$ 의 블록-복구 실패율 및  $10^{-3}$ 의 AL-FEC 이전의 패킷 손실율에서 "12" 패킷들의 오버헤드(Overhead)를 갖춘 경우, AL-FEC 이후의 패킷 손실율은  $10^{-5}$ 로 계산된다. 실패의 경우에도 정확하게 수신된 시스터매틱 패킷들은 여전히 사용할 수 있다.

[0046] 일부 실시예들에 따라 패킷 삭제율(Erasure Rate)을 줄이기 위해 사용되는 다른 AL-FEC 코딩 기술은 단일 패리티 검사(Single Parity Check: SPC), 컨볼루션 코딩(Convolution Coding), 해밍 코드(Hamming Code)를 포함할 수 있다. 이러한 방법들은 검색하거나, 더 나아가 패킷을 정정하거나. 또는 최소한 성능이 악화되지 않게 하기 위해 어떠한 패킷들이 삭제되어야 하는지의 선형적인 지식(a Priori Knowledge)을 사용하는 방법을 취할 수 있다. 추가적으로, 상기의 방법과 함께 부분적으로 정정된 패킷들(부분적인 비트들이 정정되었으나, 전체는 아닌-삭제된 패킷으로 식별된)을 사용하면 추가적으로 정정된 패킷들 내의 남은 비트들을 수정하는 것을 더 도울 수 있으며, 순환 중복 검사(Cyclic Redundancy Check: CRC) 또는 다른 정확성 검사를 통과하는데 도움이 될 수 있다.

[0047] 단일 패리티 검사 기술에서, 추가적인 패리티 패킷(즉, 수리 패킷)이 패킷들의 집합에 추가되고, 패킷 전체는  $n = k + '1'$ 로 만들어진다. 추가된 패킷은 모든 k 개의 패킷들의 패리티 검사 일 수 있다. 다시 말해서, 패리티 패킷은 k 개의 패킷들에 비트단위 배타적 논리합(A bit-wise Exclusive-or: Bit-wiser XOR)을 수행함으로써 생성된다. k 개의 패킷들은 원본 시퀀스 또는 실시예에 따른 다른 방식에 따라서 선택될 수 있다. 이 방법은 n 개의 패킷들 내의 단일 패킷 삭제(Single Packet Erasure)의 정정들을 보장한다. 일부 실시예에서, 추가된 패킷들의 높은 순서의 다른 조합들은 어떤 패킷이 삭제되었는지의 선형적인 지식을 사용함으로써 n 개의 패킷들 내의 더 삭제된 패킷들을 수정하기 위해 사용될 수 있다. n 개의 패킷들 내에 하나 이상의 삭제된(또는 없어진) 패킷들이 검색되는 경우 프로세스는 중단되거나 시작되지 않을 수 있고, 가능한 재전송을 위해 계층 구조 내의 상위 계층에 삭제된 패킷들을 위한 알림이 송신될 수 있다. SPC를 사용하면, 비트 레이트(Bit Rate)은 증가할 수 있다. 게다가, 간단한 SPC에서 다른 코딩 레이트들을 사용하면 쉽게 요구되는 성능을 회복시킬 수 있다.

[0048] 일부 실시예들에서, 단일 패리티 검사 방식은 인터리브된(Interleaved) 또는 비-인터리브된(Non-Interleaved) 방식 일 수 있다. 도 5A는 본 발명의 일부 실시예에 따라 비-인터리브된 단일 패리티 검사 (Non-interleaved Single Parity Check) 방식을 도시한다. 이 실시예에서  $k = '24'$  소스 패킷들이다. 추가된 패킷(패킷 0)은 모든 k 개의 데이터 패킷들(패킷들 1, ..., 24)의 패리티 검사일 수 있다. 일부 실시예에서 패킷 0은 PAL 또는 MAC 계층 위의 다른 계층 내의 24개의 소스 패킷들에 비트간 비트단위 배타적 논리합을 수행함으로써 생성될 수 있다. 이 방법은  $n = k + '1' = 25$  패킷들 내의 단일 패킷 삭제의 정정들을 보장한다.

[0049] 비-인터리브된 FEC에서, A/V PAL은 시퀀스 번호들이  $i+1, \dots, i+24$ (즉, '1', ..., '24')인 패킷들을 보호하기 위해 시퀀스 번호가  $i$ (도 5A 내의 '0')인 패리티 패킷 헤더를 생성한다. 패리티 패킷 길이 필드는  $i+1, \dots, i+24$  패킷들 중 가장 큰 패킷의 길이로 설정된다. FEC 인코더는 패킷들  $i+1$ 부터  $i+24$ 까지의 비트단위 배타적 논리합에 의해 패리티 패킷을 위한 페이로드를 계산하고, 더 짧은 패킷들은 가상의(송신되지 않은) 제로 비트들(Zero-bits)을 덧붙이는 것(Padded)으로 가정된다.

[0050] 일부 실시예에서 패킷 인터리버(Interleaver)는 성능을 증가시키기 위해 추가될 수 있다. 페이딩 채널(Fading Channel)의 특성 때문에, 삭제된 패킷들은 순차적일 수 있고, 이에 따라 상당히 SPC가 저하될 수 있다. 패킷 추가 인터리버는 삭제된 패킷들을 분산함으로서 SPC가 성능을 회복할 수 있도록 크게 도울 수 있다. 도 5B는 본 발명의 일부 실시예에 따라 인터리브된 단일 패리티 검사 (Interleaved Single Parity Check) 방식을 도시한다.

[0051] 도시된 실시예에서,  $n = '25'$  및  $k = '24'$ (하나의 SPC) 패킷과 같은 패킷들의 각 집합들을 위해 패킷들의 4개의 집합을 연결시키는 동안 사각형 인터리버가 사용된다. 즉, k 데이터 패킷들 및 하나의 패리티 패킷을 포함하는 단일 블록을 송신하는 비-인터리브된 SPC 방식과는 달리, 인터리브된 SPC 방식은 k 데이터 패킷들의 4개의 블록들 및 4개의 패리티 패킷들을 송신한다.

[0052] 인터리브된 FEC에서, A/V PAL은 시퀀스 넘버들이  $i+4, \dots, i+99$ 인 패킷들을 보호하기 위해 시퀀스 넘버가  $i$ ,

$i+1, i+2$  및  $i+3$ (도 5B의 '0', '1', '2' 및 '3')인 4개의 패리티 패킷 헤더들을 생성한다. FEC 인코더는 패리티 패킷의 페이로드들을 아래와 같이 계산한다. 패리티 패킷의 페이로드  $i$ 는 패킷들  $i+4, i+8, i+12, \dots, i+96$ 의 비트단위 배타적 논리합이고, 패킷  $i+1$ 의 페이로드는 패킷들  $i+5, i+9, i+13, \dots, i+97$ 의 비트단위 배타적 논리합이며, 패리티 패킷  $i+2$ 의 페이로드는 패킷들  $i+6, i+10, i+14, \dots, i+98$ 의 비트단위 배타적 논리합이며, 패리티 패킷  $i+3$ 의 페이로드는 패킷들  $i+7, i+11, i+15, \dots, i+99$ 의 비트단위 배타적 논리합이다.

[0053] 블록 당 하나의 패킷 이상을 복구하지 못하는 비-인터리브된 SPC 방식과는 다르게, 인터리브된 SPC는 2개 내지 4개의 연속적인 삭제된 패킷들을 복구할 수 있다. 패리티 패킷들 및 시스터매티 패킷들의 순서는 변경될 수 있다. 예를 들면, 시스터매티 패킷들(예를 들면 데이터 패킷들)은 패리티 패킷들 이전에 송신될 수 있다.

[0054] 도 6은 본 명세서의 일 실시예에 따라 삭제된 패킷들을 복구하기 위한 컨볼루션 코드(Convolutional Code)를 도시한다.

[0055] 컨볼루션 코드(Convolutional Code: CC)는 이웃해있는 비트들의 배타적 논리합(Exclusive OR: XOR) 하는 것의 기초한다. 컨볼루션 코드의 높은 구속장(Constraint Length: CL)은 좋은 수리 성능을 가진다. 더 높은 CL은 더 높은 디코딩 복잡도(비터비(Viterbi)/터보(Turbo) 디코딩)이다. 컨볼루션 코드의 단점은, 만약 삭제된 패킷이 존재하는 경우 패킷 내의 모든 비트들을 알 수 없다. 동일한 패킷으로부터의 연속적인 삭제된 비트들(CL 보다 더 많은)에서, 비터비 디코딩은 패킷을 수리할 수 없을 수 있다. 이는 패킷들이 인코딩 되기 전 또는 송신되기 전에 인터리브 되어야 하는 것을 뜻한다. 인터리버의 길이는 트레이스백(Traceback)(예를 들면  $5 \times CL$ ) 또는 최소한 CL 보다 큰 크기일 수 있다. 삭제된 비트들이 인터리브 되었다면, 그들은 복구될 가능성이 증가한다. 도 6은 '39'의 시스터매티 패킷들(즉, 포함된(Contain) 데이터) 및 하나의 패리티 패킷(CC는 수리 성능을 증가시키기 위해 모든 코딩 레이트를 지원함)이 존재하도록 '1000' 바이트들의  $n = 40$ 인 패킷들의 인터리버 크기 및  $CL = 7$  및 '39/40' ( $n-1/n$ )의 CC 코딩레이트(Coding Rate)를 가진 인터리브된 컨볼루션 코드에 따라 AL-FEC 인코딩된 패킷들의 예를 도시한다.

[0056]  $P_{x,y}$  표기법은 비트 패킷과 비트 위치를 나타낸다. 예를 들면  $P_{0,1}$ 은 패킷 '0' 및 비트 위치 1을 나타내고, ' $P_{1,0}$ '은 패킷 '1' 및 비트 위치 '0'을 표시한다. 이처럼, 각 데이터의 비트들 및 패리티 패킷은 도 6에 수직적으로 도시된 것과 같이, 컨볼루션 코드는 패리티 패킷 내의 대응하는 패리티 비트를 계산하기 위해 동일한 비트 위치에 걸쳐 블록 내의  $n-1$  패킷들까지 작동한다. 예를 들면, 패리티 패킷  $n-1$ 의 비트 '0'을 계산하기 위해(즉, ' $P_{n-1,0}$ '), 컨볼루션 코드는 ' $P_{0,0}$ ', ' $P_{1,0}$ ', ' $P_{2,0}$ ' ..., ' $P_{n-2,0}$ '에 걸쳐 동작한다. 다음으로 패리티 패킷의 비트 '0'을 계산하기 위해(즉, ' $P_{n-1,1}$ ')이 계산되며, 이런 식으로 패리티 패킷의 마지막 비트까지 계산된다. 만약 데이터 패킷들이 다른 길이들을 가지고 있다면, 패리티 패킷은 가장 큰 패킷의 길이를 가지고, 더 짧은 패킷들은 컨볼루션 코드의 실행의 목적을 위해 그들 각각의 꼬리(tail)들에 가상의 제로들이 덧붙여(Padded)진다. SPC와 마찬가지로 패리티 패킷과 시스터매티 패킷의 순서는 변경될 수 있다.

[0057] 또한 이 케이스에서, 삭제된 패킷들의 선형적 지식(A-Priori Knowledge)은 CC AL-FEC 가 사용되어야 하는지(예를 들면, 매우 낮은 SNR에서 패킷 삭제율이 SNR의 감소처럼 높은 수리 확률로 감소하는 때)를 결정할 수 있다.

[0058] 다른 실시예에서 AL-FEC를 위해 해밍 코딩(Hamming Coding)이 사용될 수 있다. 해밍 코딩(Hamming Coding: HC)을 사용하여 해밍 코딩의 최소한의 해밍 거리(Hamming Distance:  $d_{min}$ )로부터 수리된 패킷들의 정도(Level)(복구될 수 있는 삭제된 비트들의 숫자 또는 패킷)이 유도될 수 있다. 예를 들면,  $d_{min} = 3$ 을 가지는 H 매트릭스(Matrix)에서 수리된 패킷들의 숫자는  $d_{min} - 1 = 2$ '의 삭제된 패킷들 일 수 있다. 해밍 코드를 사용할 때는, 아마도 패킷들의 각 집합들을 위해 하나 이상의 패리티 패킷이 있을 것이다(코딩 레이트를 조절함으로써). H 매트릭스는  $d_{min}$  및 코딩레이트를 결정한다. 패킷들의 각 집합 내의 데이터 패킷들의 최대 숫자는  $m$  패리티 비트들(또는 패킷들)을 위해,  $2 - m - 1$  데이터 비트들(또는 패킷들)까지 인코딩될 수 있도록 해밍코드 패리티 패킷들의 숫자에 근거하여 결정된다. 예를 들면, 해밍(7,4) 코드는 모든 4개의 데이터 비트들을 7개의 비트들로 인코딩하고, 3개의 패리티 비트들을 추가한다.

[0059] 일부 실시예들에서, 인터리브된, 또는 인터리브되지 않은 단일 패리티 검사는 해밍코드와 함께 결합될 수 있다. 예를 들면, 단일 패리티 검사 인코딩(인터리브 방식, 또는 비-인터리브 방식)은 추가적인 패리티 패킷을 형성하기 위해 해밍 인코딩 패킷들에 수행될 수 있다.

[0060] 상기 실시예들은 랩터 코드들에 비해 AL-FEC를 구현하기 위한 더 간단한 방식을 설명한다. 설명된 모든 AL-FEC 방식들은 삭제된 패킷들을 수리하기 위해 저전력 디바이스에서 사용되기에 적합하다.

[0061] 도 7은 본 명세서의 일 실시예에 따라, AL-FEC를 사용하여 인코딩된 패킷의 헤더의 내용을 도시한다. 일부 실시

예들에서 패킷 헤더(Packet Header)(700)은 PAL 내의 AL-FEC 방식이 수행되는 특정한 기능적 계층에서 생성되고, 첨부될 수 있고, 패킷 헤더(700)은 PAL로부터 나가는 패킷들에 생성되거나 첨부될 수 있다.

[0062] 패킷 헤더(700) 내의 각 열은 적어도 하나의 필드와 대응된다. 상단 행의 각 블록 내의 텍스트는 패킷 헤더 필드(Packet Header Field) 내에 포함된 정보를 설명하고, 하단 행 내의 숫자들은 각각의 헤더 필드들의 옥텟(Octet) 길이(바이트들의 숫자)를 표시한다. 1 옥텟의 길이를 가지는 패킷타입 헤더 필드(Packet Type Header Field)는 패킷이 시스터매틱(Systematic)인지(예를 들면, 데이터를 포함하고 있는지) 또는 패리티인지 여부를 표시한다.

[0063] 일 실시예에서, 단일 비트(예를 들면 첫 번째 비트)는 시스터매틱(Systematic)(예를 들면, 데이터) 패킷으로부터 패리티 패킷을 구분하기 위해 패킷타입 필드 내에 할당될 수 있다. 패리티 패킷을 표시하기 위해 단일 비트를 사용할 때, '0'은 패킷이 시스터매틱(data) 패킷임을 표시하기 위해 사용될 수 있고, '1'은 패킷이 패리티(또는 그 반대)임을 표시하기 위해 사용될 수 있다. 예를 들면, '1xxxxxxxx'의 값을 가지는 패킷타입 헤더 필드는 패리티 패킷을 표시하는 것일 수 있고 반대로 '0xxxxxxxx'는 시스터매틱 패킷을 표시할 수 있다. 일 실시예에서 단일 비트는 다른 패킷 헤더 필드들에 할당될 수도 있다.

[0064] 다른 실시예로서, WiGig 표준에서 패킷타입 필드 내의 특정한 8-비트 값은 패리티 패킷을 표시하기 위해 정의될 수 있다. 예를 들면 '0x80'(또는 '10000000')의 값은 패리티 패킷을 표시할 수 있다. 다른 방식으로, 특정한 값이 시스터매틱 패킷들로부터 패리티 패킷을 구별하기 위해 다른 헤더 필드에 사용될 수 있다.

[0065] 스트림(Stream) 식별자(Identifier: ID)은 패킷이 속하는 데이터 스트림을 식별한다. 일 실시예로서, 스트림 ID는 송신 블록을 참조할 수 있다. 시퀀스 번호(Sequence Number: SeqNum)은 블록 내의 패킷의 순서를 식별할 수 있다. 일 실시예에 따라, 길이 필드(Length Field)는 페이로드의 크기 또는 블록 내의 패킷들의 총 숫자를 표시할 수 있다.

[0066] 송신 또는 수신 디바이스가 AL-FEC 성능(Capability)들을 가지고 있지 않은 경우가 있을 수 있다. 따라서, 본 명세서의 실시예에 따라, 디바이스들은 페이로드들을 전송하기 전에 성능 교환 기간 동안 AL-FEC 성능 정보를 교환할 수 있다.

[0067] 도 8은 본 명세서의 일 실시예에 따라, 무선 통신 시스템 내에서 AL-FEC의 사용을 위한 일반적은 프로세스를 도시한다. 데이터 통신 전에, 송신 디바이스(400)은 및 수신 디바이스(450)은 성능 정보(810)을 교환한다. 일부 실시예들에서, 성능 교환 정보(810)은 AL-FEC 성능 정보를 포함한다. 블록 820에서 성능 교환 프로세스가 완료된 후, 송신 디바이스(400)은 송신을 위해 AL-FEC 코드를 이용하여 패킷들을 인코딩한다. 이 프로세스는 도 9를 참조하여 더 설명될 것이다. 일부 실시예들에서, 송신 디바이스(400)은 AL-FEC 인코딩 이후 PHY 계층 내에서 추가적인 FEC 코딩을 수행할 수 있다. 송신 디바이스(400)은 이후 인코딩된 시스터매틱 패킷들(예를 들면, 데이터) 및 수리 패킷들(830)을 수신 디바이스(450)에게 송신한다. 블록 840에서, 수신 디바이스(450)은 AL-FEC 코드를 이용하여 없어진 패킷들(또는 오류가 발생한 패킷들)을 복구할 수 있다. 일부 실시예들에서 수신 디바이스는 AL-FEC 디코딩 프로세스를 사용하여 패킷 복구를 수행할지 여부를 먼저 결정할 수 있다. 이 프로세스는 도 10을 참조하여 더 설명될 것이다. 일부 실시예들에서, 수신 디바이스(450)은 AL-FEC 디코딩을 위해 수신된 패킷을 보내기 전에 PHY 계층 내에서 FEC 디코딩을 수행할 수 있다.

[0068] 도 9는 본 명세서의 일 실시예에 따라 AL-FEC를 사용하여 송신하기 위해 인코딩 패킷들을 위한 프로세스를 도시한다. 블록 910에서 손신 디바이스(예를 들면 송신 디바이스(400))은 적어도 하나 이상의 수신 디바이스(예를 들면, 수신 디바이스(450))에게 AL-FEC 성능 정보를 송신한다.

[0069] 블록 930에서, 송신 디바이스는 소스 패킷들에 AL-FEC 코드를 사용함으로써, 시스터매틱 패킷들 및 적어도 하나의 수리 패킷(AL-FEC 코딩 방식에 기초하여)을 생성한다. 일 실시예에서, PAL은 소스 블록(함께 보호되어야 할 소스 패킷들의 집합)에 소스 블록 번호(Source Block Number: SBN)을 할당하고, 소스 블록, SBN, T, 및 k 를 AL-FEC 인코더에 보낸다. AL-FEC 인코더(하드웨어 또는 소프트웨어로 구현될 수 있는)는 수리 패킷들의 집합을 생성하기 위해 소스 패킷들에 AL-FEC 코드를 적용한다.

[0070] 상기에서 설명한 AL-FEC 코딩 방식들 중 하나를 사용함에 따라 적어도 하나의 수리 패킷들이 생성된다. 예를 들면, 비-인터리브된 SPC 코딩 방식이 사용될 때, 소스 블록 당 하나의 수리(즉, 패리티) 패킷이 생성된다. 즉, n ='25' 및 k ='24'라 가정하면, PAL은 시퀀스 번호 i인 패리티 패킷 헤더를 생성하고, i+1, ..., i+24 패킷들을 FEC 인코더로 보낸다. FEC 인코더는 i+1, ..., i+24 패킷들의 비트단위 배타적 논리합에 의해 패리티 패킷의 페이로드를 생성한다. 4-인터리브된(Four-Interleaved) SPC 코딩 방식이 사용되는 때, 데이터 패킷들의 4개의 집

합들의 각각을 위해 인터리브된 SPC 코딩 방식이 사용되는 때 하나의 수리 패킷이 생성된다. 즉, 각 집합에서,  $n = '25'$  및  $k = '24'$ 로 가정하면, PAL은 시퀀스 번호  $i, i+1, i+2, i+3$ 인 4개의 패리티 패킷 헤더를 생성하고,  $i+4, \dots, i+99$  패킷들을 FEC 인코더에 보낸다. FEC 인코더는  $i+4, i+8, \dots, i+96$  패킷들의 비트단위 배타적 논리합에 의해 패리티 패킷 페이로드를 생성한다. FEC 인코더는 패리티 패킷들의 페이로드들을 다음과 같이 계산한다. 패리티 패킷  $i$ 의 페이로드는  $i+4, i+8, i+12, \dots, i+96$ 의 비트단위 배타적 논리합이고, 패리티 패킷  $i+1$ 의 페이로드는  $i+5, i+9, i+13, \dots, i+97$ 의 비트단위 배타적 논리합이고, 패리티 패킷  $i+2$ 의 페이로드는  $i+6, i+10, i+14, \dots, i+98$ 의 비트단위 배타적 논리합이고, 패리티 패킷  $i+3$ 의 페이로드는  $i+7, i+11, i+15, \dots, i+99$ 의 비트단위 배타적 논리합이다. 일 실시예에서, AL-FEC 방식이 사용될 수 있으며, 블럭 단위(Block-by-Block) 기반으로 변경될 수 있다.

[0071] 소스 블록 내의 적어도 하나의 소스 패킷들은 시스터매티 패킷들을 포함한다. 일 실시예에서, 각 시스터매티 패킷은 하나의 소스 패킷에 대응된다. 시스터매티 패킷들을 위해 수리 패킷들이 생성되고, 수리 패킷들의 길이는 패킷들 가운데 가장 긴 시스터매티 패킷의 길이와 동일하다. PAL은 시스터매티/수리 패킷들의 각각을 위한 패킷 헤더(시스터매티/패리티 패킷들, 스트림 ID, 시퀀스 번호, 등을 위한 지시자(indicator)를 포함하는)를 생성하고, 패킷 헤더를 대응하는 시스터매티/수리 패킷들에 첨부하고, 소스 패킷들을 MAC 계층에 보낸다.

[0072] 일 실시예로서, 블록 930은 PAL 및 AL-FEC 인코더에서 수행된다. 앞서 논의되었던 바와 같이, PAL 및 AL-FEC 인코더는 분리된 컴포넌트들일 수 있고, PAL 내에 병합될 수도 있다. 다른 실시예로서, 블록 930은 MAC 계층 위와 같은 다른 상위 기능적 계층에서 수행될 수 있다.

[0073] 블록 940에서, 블록 내의 인코딩된 시스터매티/수리 패킷들은 이후 송신을 위해 MAC 및 PHY 계층들에서 처리된다. 일 실시예로서, PHY 계층은 요구되는 성능을 달성하기 위해 송신 이전에 AL-FEC 인코딩된 패킷들에게 FEC 코딩 방식을 적용한다.

[0074] 도 10은 본 명세서의 일 실시예에 따라 AL-FEC를 사용하는 디코딩 패킷들을 위한 프로세스를 도시한다. 프로세스 840은 1) 수리 패킷이 없으면 AL-FEC를 사용하지 않는 옵션; 2) 각 블록에 25개의 패킷들(24개의 시스터매티 패킷들 및 한 개의 수리 패킷)이 포함되어 있는 비-인터리브된(25, 24)-SPC AL-FEC 옵션; 3) 각 블록에 25개의 패킷들의 4개의 집합들(각 집합 당 24개의 시스터매티 패킷들 및 한 개의 수리 패킷)이 포함되어 있는 4-인터리브된(25, 24)-SPC AL-FEC 옵션과 같은 3가지 옵션 중 하나로 동작할 수 있는 시스템의 일 실시예를 나타낸다.

[0075] 블록 1010에서, 수신 디바이스(예를 들면, 수신 디바이스(450))는 송신 디바이스(예를 들면, 송신 디바이스(400))으로부터 송신 블록의 패킷들을 수신한다. 블록 1020에서, 수신디바이스는 수리 패킷들이 수신되었는지 여부를 판단한다. 즉, 수신 디바이스 내의 PAL(또는 AL-FEC 디코딩을 수행하는 책임이 있는 기능적 계층)은 AL-FEC 지시자(예를 들면, 도 7과 관련하여 설명되었던 패킷 타입 필드)를 찾기 위해 각 패킷의 헤더를 검사한다.

[0076] 만약 수리 패킷이 수신된 바 없으면, 프로세스는 블록 1060으로 진행되고, AL-FEC 복구는 수행되지 않는다. 만약 AL-FEC를 사용하지 않는 상태로 패킷들이 전송되는 경우 이런 프로세스가 발생할 수 있다. 이처럼, PAL은 수리 패킷들로 표시되는 수신된 패킷들이 없는지, 로드(Load) 내의 수신된 모든 패킷들을 판단할 수 있다. 만약 AL-FEC를 사용하지 않는 패킷들이 없어진 경우, 수신 디바이스는 없어진 패킷들 또는 전체 블록의 재전송을 요청할 수 있다. 일 실시예로서, 수리(예를 들면, 패리티) 패킷(들)이 없어진 경우에만, 수신 디바이스는 없어진 수리 패킷들의 재전송을 요청할 수 있다.

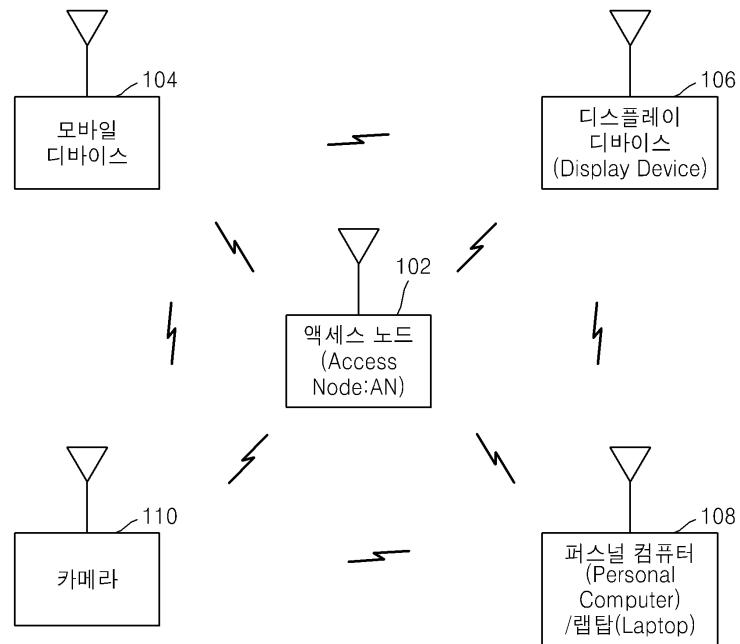
[0077] 만약 수리 패킷들이 수신되면, 블록 1030에서, 수신 디바이스는 패킷들이 인터리브 되었는지를 판단한다. 예를 들면, 수리 패킷들이 먼저 순차화(Sequenced) 되어 있다고 가정하면, 로드의 첫번째 패킷(SeqNum  $i$ 인)이 수리 패킷인 경우에만, 수신 디바이스는 패킷들이 인터리브되지 않은 패킷들이란 것을 판단할 수 있다. 블록 내의 첫 번째 패킷이 패킷 헤더 내의 AL-FEC 비트(예를 들면, 패리티 패킷 지시자)에 의해 표시되는 수리 패킷이어야 하기 때문이다. 패킷  $i+1$ ( $i+1$ 이 사라진 경우 패킷  $i+2$  또는  $i+3$ )이 수신된 이후, 만약 패리티 패킷 지시자가 0으로 설정되어 있으면, 패킷들  $i, i+1, \dots, i+24$ 는 AL-FEC 블록을 형성한다. 만약 소스 패킷들  $i+1, \dots, i+24$  중 하나가 사라진 경우, AL-FEC 블록 내의 다른 패킷들과 비트단위 배타적 논리합 동작을 통해 복구될 수 있다.

[0078] 예를 통해 계속하면, 만약 처음의 4개의 패킷들(SeqNum  $i, i+1, i+2$  및  $i+3$ 인) 중 적어도 두개가 AL-FEC를 사용한다고 표시되어(예를 들면, 패리티 패킷 지시자가 1로 설정) 수신되었다면, 수신 디바이스는 인터리브된 SPC AL-FEC 코딩 방식을 사용하여 패킷들이 송신되었다는 것을 판단할 수 있다. 즉, 패킷들  $i, i+4, i+8, \dots, i+96$ 은 AL-FEC 집합(예를 들면, 블록)을 형성하고, 패킷들  $i+1, i+5, i+9, \dots, i+97$ 이 집합을 형성하고, 패킷들  $i+2, i+6, i+10, \dots, i+98$ 이 집합을 형성하고, 패킷들  $i+3, i+7, i+11, \dots, i+99$ 가 집합을 형성한다.

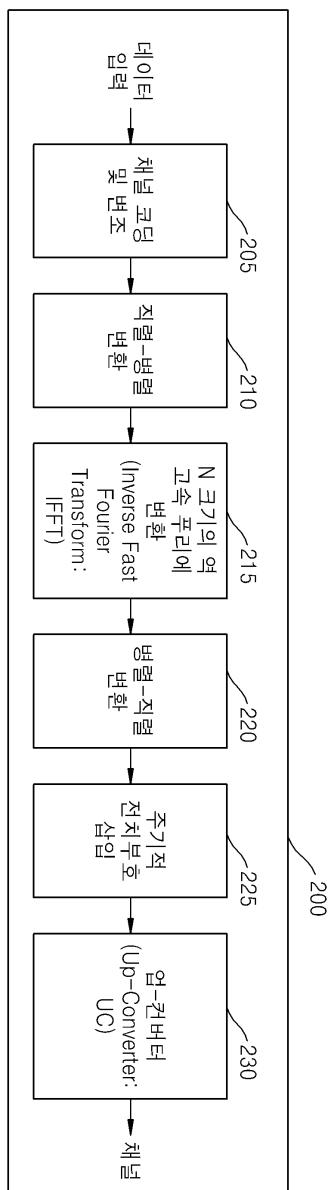
- [0079] 상술한 모든 경우들 및 모드들에서 삭제된(페리티/시스터메틱) 패킷들은 복구되거나 복구되지 못하던지 간에, 송신 디바이스의 PAL은 스트림 ID 및/또는 SeqNum을 알고 있는 점을 사용하여 패킷들을 어떻게 수리할지를 항상 결정할 수 있다는 것은 보장된다.
- [0080] 블록 1040에서, 수신 디바이스가 어떤 패킷들이 사라졌는지 여부를 결정한다. 만약 없어진 패킷들이 없으면,(즉 모든 소스 패킷이 성공적으로 수신됨), 그 후 소스 패킷은 AL-FEC 복구 없이 처리되고, 수리 패킷들은 폐기된다. 프로세스는 블록 1060으로 진행된다.
- [0081] 반대로 만약 소스 패킷들이 사라진 경우, 블록 1050에서, 수신 디바이스는 AL-FEC 코딩 방식의 타입에 기초하여 사라진 패킷들의 숫자가 한계값(Threshold) 보다 큰지 여부에 대해 판단한다. 도 5A 및 5B와 관련하여 설명한 바와 같이 비-인터리브된 SPC 방식 하에서는 오로지 하나의 없어진 패킷만이 복구될 수 있고, 4-인터리브된 SPC 방식 하에서는 집합 당 연속적인 4개의 없어진 패킷들까지 또는 하나의 없어진 패킷이 복구될 수 있다.
- [0082] 이처럼, 블록 1030에서, 만약 PAL이 패킷들이 인터리브 되지 않았다고 판단한 경우, 한계값은 하나의 없어진 패킷이다. 그렇지 않고, 블록 1030에서, 만약 PAL이 패킷들이 인터리브 되었다고 판단한 경우, 한계값은 4개의 연속적인 사라진 패킷들 또는 하나의 없어진 패킷이다.
- [0083] 만약 사라진 패킷들의 숫자가 대응하는 한계값보다 큰 경우, 프로세스는 블록 1060으로 진행되고 AL-FEC 복구를 수행하지 않는다. 일 실시예로서, 수신 디바이스는 없어진 패킷들 또는 전체 블록의 재전송을 요청할 수 있다.
- [0084] 반대로, 만약 사라진 패킷들의 숫자가 대응하는 한계값 보다 크지 않은 경우, 블록 1070에서, 수신 디바이스는 사라진 패킷(들)을 복구하기 위해 AL-FEC 디코딩을 수행한다. 즉, PAL은 사라진 패킷(들)을 복구하기 위해 블록들을 FEC 디코더로 보낸다.
- [0085] 도 10에 도시된 프로세스는 설명만을 목적으로 하는 것이며, 본 명세서의 범위를 제한하는 의미는 아니다. 다른 실시예들로서, 본 명세서의 범위를 벗어나지 않는 범위내에서 블록들의 순서는 재배열 될 수 있다. 예를 들면, 어떤 패킷들이 업성겼는지 여부를 판단하기 위한 블록 1040은 블록 1010에서 MAC 계층으로부터 패킷을 수신받은 이후 즉시 수행될 수 있다. 블록 1020은 그적어도 하나의 패킷이 없어졌는지를 결정한 이후에 수행될 수 있다.
- [0086] AL-FEC 블록 내의 각 AL-FEC 패킷의 길이는 최대 길이를 가진 패킷과 동일한 것에 유의해야한다. 이는 동일한 블록 내의 PAL 패킷들이 다른 길이를 가지는 것을 가능하게 한다. 예를 들면 PAL 패킷 길이가 AL-FEC 패킷 길이 보다 작을 때, AL-FEC 인코더/디코더는 '가상의 ' 0들을 덧붙임(Padding)으로서 이러한 차이를 보상할 수 있다.
- [0087] 비록 본 명세서가 전형적인 실시예로 설명되었으나, 다양한 변경들 및 수정들이 당업자에게 시사될 수 있다. 본 명세서는 상술한 변경들 및 수정들을 포함할 것이며, 이는 첨부된 청구항들의 범위 내에 포함될 것이다.

도면

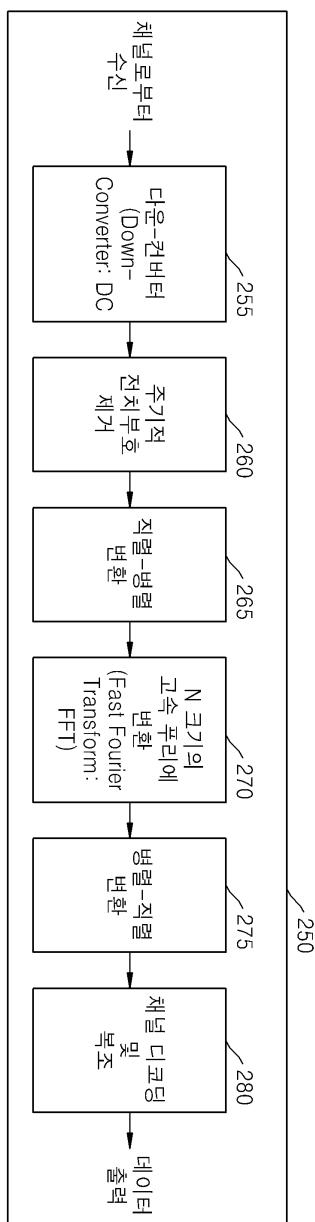
도면1



도면2a

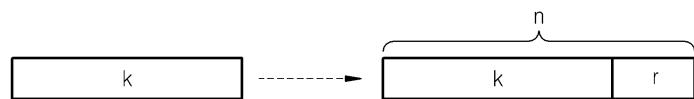


도면2b

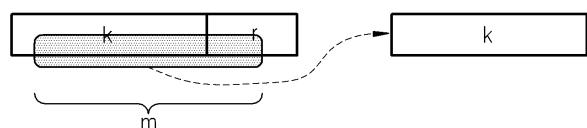


## 도면3

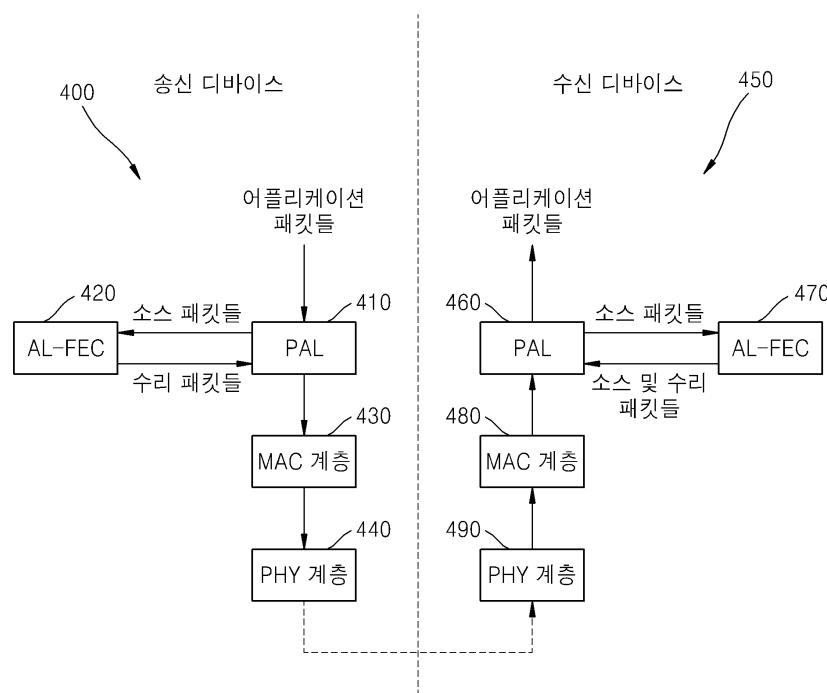
AL-FEC 인코딩



AL-FEC 디코딩

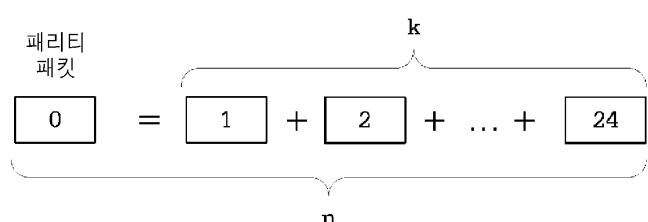


## 도면4



## 도면5a

비-인터리브된 단일 패리티 검사

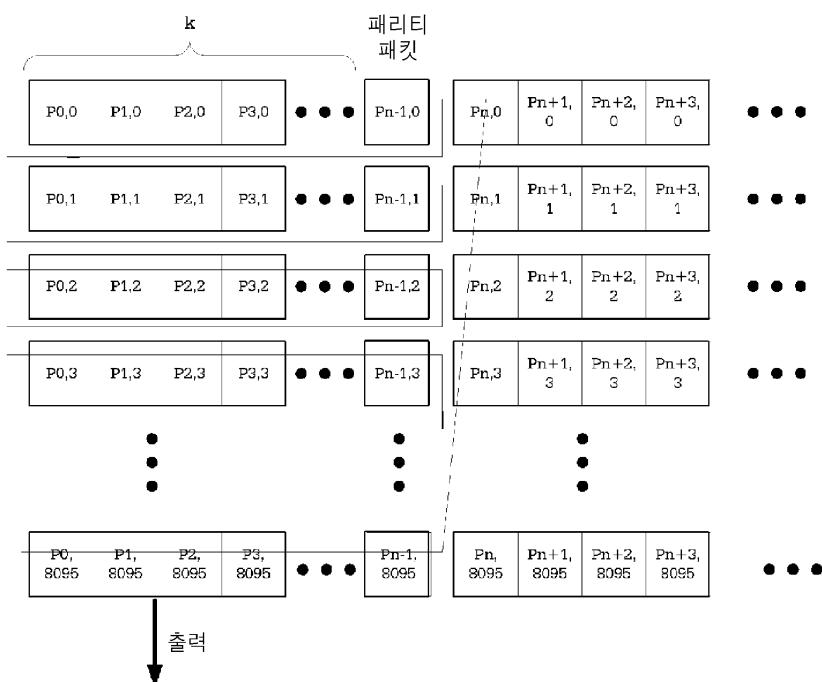


## 도면5b

인터리브된 단일 패리티 검사

$$\begin{array}{rcl}
 0 & = & 4 + 8 + \dots + 96 \\
 1 & = & 5 + 9 + \dots + 97 \\
 2 & = & 6 + 10 + \dots + 98 \\
 3 & = & 7 + 11 + \dots + 99
 \end{array}$$

## 도면6

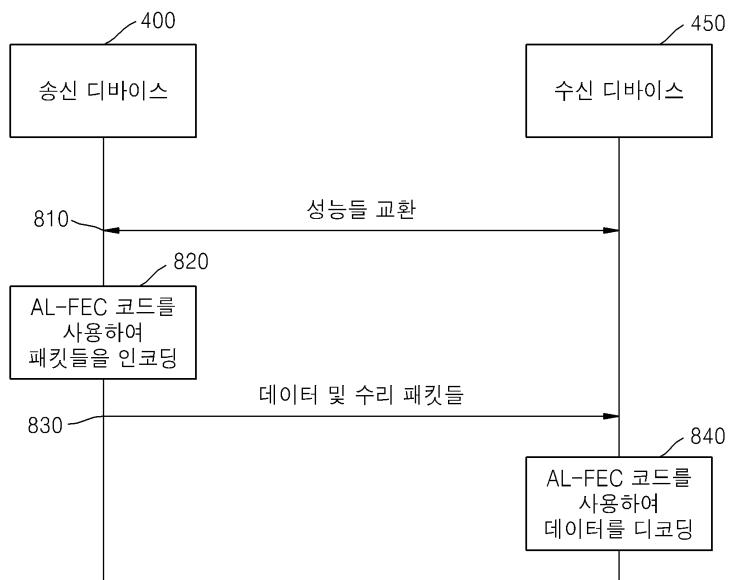
예: 컨볼루션 코드  
(CL=7, n=40, k=39)

## 도면7

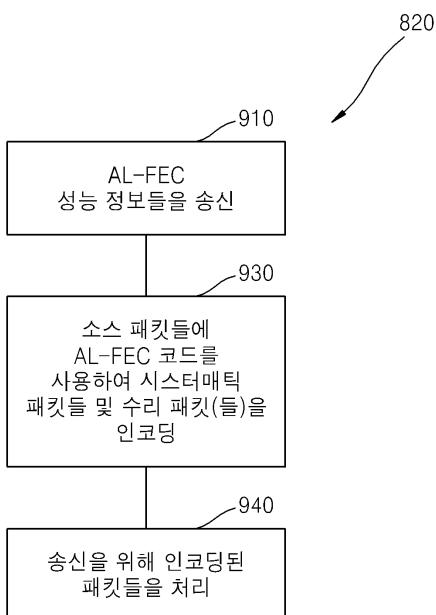
700

패킷 타입 (Packet Type)	스트림 ID (Stream ID)	시퀀스 번호 (SeqNum)	길이 (Length)	다른 헤더들 (Other Headers)	페이지 (Payload)
1	1	2	2	가변적	가변적

## 도면8



## 도면9



## 도면10

