



(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ

(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN)  
CỤC SỞ HỮU TRÍ TUỆ

(11)



1-0030438

(51)<sup>7</sup> H04L 1/00; H04L 27/26

(13) B

(21) 1-2017-02072

(22) 05/03/2015

(86) PCT/KR2015/002127 05/03/2015

(87) WO 2016/072566 A1 12/05/2016

(30) 62/075,898 06/11/2014 US; 62/080,382 16/11/2014 US

(45) 25/12/2021 405

(43) 25/08/2017 353A

(73) LG ELECTRONICS INC. (KR)

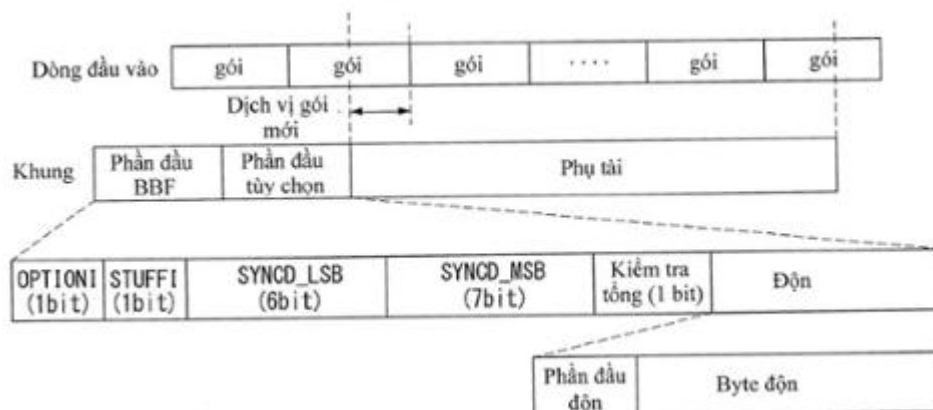
128, Yeoui-daero, Yeongdeungpo-gu, Seoul 150-721, Republic of Korea

(72) HWANG, Jaeho (KR); KO, Woosuk (KR); HONG, Sungryong (KR).

(74) Công ty TNHH một thành viên Sở hữu trí tuệ VCCI (VCCI-IP CO.,LTD)

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ THU TÍN HIỆU PHÁT RỘNG

(57) Sáng chế đề cập đến phương pháp thu tín hiệu phát rộng. Phương pháp này bao gồm bước đưa ra dòng dữ liệu bao gồm các bit được giải mã, và bước đưa ra bao gồm bước phát hiện thông tin được bao gồm trong phần đầu của khung dải gốc. Sáng chế còn đề cập đến thiết bị thu tín hiệu phát rộng.



## **Lĩnh vực kỹ thuật được đề cập**

Sáng chế đề cập đến thiết bị truyền tín hiệu phát rộng, thiết bị thu tín hiệu phát rộng, và các phương pháp truyền và thu tín hiệu phát rộng.

## **Tình trạng kỹ thuật của sáng chế**

Khi việc truyền tín hiệu phát rộng tương tự đã đến hồi kết, các công nghệ khác nhau để truyền và thu tín hiệu phát rộng số đã được phát triển. Tín hiệu phát rộng số có thể bao gồm nhiều dữ liệu video/audio hơn so với tín hiệu phát rộng tương tự và còn bao gồm các loại dữ liệu bổ sung khác nhau cũng như dữ liệu video/audio.

Vấn đề cần được giải quyết bởi sáng chế

Nghĩa là, hệ thống phát rộng số có thể cung cấp ảnh độ phân giải cao (High Definition, viết tắt là HD), audio đa kênh, và các dịch vụ bổ sung khác nhau. Tuy nhiên, đối với việc phát rộng số, hiệu suất truyền dữ liệu đối với việc truyền nhiều dữ liệu, độ mạnh của mạng truyền và thu, tính linh hoạt của mạng xét về thiết bị thu di động cần được cải thiện.

## **Bản chất kỹ thuật của sáng chế**

Cách thức giải quyết vấn đề

Hơn nữa, mục đích của sáng chế là đề xuất phương pháp truyền tín hiệu sự có mặt của trường độn trong BBF.

Ngoài ra, mục đích khác của sáng chế là đề xuất phương pháp định rõ trường độn mà có trong BBF, nghĩa là, loại sử dụng của dữ liệu độn.

Ngoài ra, mục đích khác nữa của sáng chế là đề xuất việc sử dụng hiệu quả trường loại độn bằng cách chia và sử dụng trường loại độn thành hai trường khác nhau.

Các mục đích kỹ thuật cần đạt được bởi sáng chế không giới hạn ở các mục đích kỹ thuật nêu trên và các mục đích kỹ thuật không được đề cập khác sẽ được hiểu rõ bởi những người có hiểu biết trung bình trong lĩnh vực kỹ thuật từ phần mô tả dưới đây.

Theo một phương án của sáng chế, phương pháp thu tín hiệu phát rộng, phương pháp này bao gồm các bước: thu tín hiệu phát rộng bao gồm ít nhất một khung tín hiệu; giải điều biến OFDM trên tín hiệu phát rộng thu được; phân tách ít nhất một khung tín hiệu của tín hiệu phát rộng thu được để trích dữ liệu dịch vụ hoặc dữ liệu thành phần dịch vụ; chuyển đổi dữ liệu dịch vụ hoặc dữ liệu thành phần dịch vụ thành các bit; giải mã các bit; và đưa ra dòng dữ liệu bao gồm các bit được giải mã, trong đó bước đưa ra bao gồm bước phát hiện (hoặc thu nhận) thông tin được bao gồm trong phần đầu của khung dải gốc, trong đó phần đầu bao gồm thông tin điều khiển chỉ báo xem liệu trường độ dài có mặt trong phần đầu hay không.

Phần đầu có thể bao gồm ít nhất một trong số thông tin chỉ báo chỉ báo xem liệu phần bit trọng số lớn nhất (most significant bit, viết tắt là MSB) của phần độ dài độ dài có mặt trong trường độ dài hay trường loại độ dài (STUFF\_TYPE) chỉ báo loại dữ liệu độ dài.

Trường độ dài có thể bao gồm phần đầu độ dài và dữ liệu độ dài, và thông tin chỉ báo và trường loại độ dài có thể được bao gồm trong phần đầu độ dài.

Kích thước của thông tin chỉ báo có thể là 1 bit, và kích thước của trường loại độ dài (STUFF\_TYPE) có thể là 2 bit.

Phần đầu độ dài có thể còn bao gồm trường độ dài độ dài (STUFF\_LEN) chỉ báo độ dài của trường độ dài, và trường độ dài độ dài (STUFF\_LEN) có thể được chia thành phần độ dài độ dài MSB ((STUFF\_LEN\_MSB) và phần độ dài độ dài LSB (STUFF\_LEN\_LSB).

Thông tin điều khiển có thể là trường ký hiệu chỉ báo mở rộng (EXT\_I), và thông tin chỉ báo có thể là trường (ký hiệu chỉ báo) MSB\_I.

Trường độ dài có thể được bao gồm trong khung dải gốc khi phụ tải không được điền đầy với gói dữ liệu hoặc việc truyền tín hiệu trong dải được sử dụng.

Dữ liệu độ dài có thể chỉ báo ít nhất một trong số đệm hoặc truyền tín hiệu trong dải.

Khi độ dài của trường độ dài là 32 byte hoặc nhỏ hơn, phần MSB của độ dài độ dài có thể không được bao gồm trong trường độ dài.

Theo phương án khác của sáng chế, thiết bị dùng để thu tín hiệu phát rộng, thiết bị này bao gồm: bộ thu dùng để thu tín hiệu phát rộng bao gồm ít nhất một khung tín hiệu; bộ giải điều biến dùng để giải điều biến tín hiệu phát rộng thu được bởi sơ đồ OFDM (Orthogonal Frequency Division Multiplexing - đa hợp phân chia theo tần số trực giao); bộ phân tách khung để phân tách ít nhất một khung tín hiệu của tín hiệu phát rộng thu được để trích dữ liệu dịch vụ hoặc dữ liệu thành phần dịch vụ; bộ chuyển đổi để chuyển đổi dữ liệu dịch vụ hoặc dữ liệu thành phần dịch vụ thành các bit; bộ giải mã để giải mã các bit; và bộ xử lý đầu ra để đưa ra dòng dữ liệu bao gồm các bit được giải mã, trong đó bộ xử lý đầu ra bao gồm bộ xử lý khung dải gốc để phát hiện (hoặc thu nhận) thông tin được bao gồm trong phần đầu của khung dải gốc, trong đó phần đầu bao gồm thông tin điều khiển chỉ báo xem liệu trường độộn có mặt trong phần đầu hay không.

#### Hiệu quả của sáng chế

Sáng chế có thể đề xuất các dịch vụ phát rộng khác nhau bằng cách điều khiển chất lượng dịch vụ (Quality Of Service, viết tắt là QoS) đối với mỗi dịch vụ hoặc thành phần dịch vụ bằng cách xử lý dữ liệu theo đặc trưng dịch vụ.

Ngoài ra theo sáng chế, tính linh hoạt truyền có thể đạt được bằng cách truyền các dịch vụ phát rộng khác nhau qua cùng độ rộng dải tín hiệu tần số radio (radio frequency, viết tắt là RF).

Ngoài ra, theo sáng chế, hiệu suất truyền dữ liệu và độ mạnh truyền và thu tín hiệu phát rộng có thể được nâng cao nhờ sử dụng hệ thống nhiều đầu vào nhiều đầu ra (Multiple-Input Multiple-Output, viết tắt là MIMO).

Bên cạnh đó, theo sáng chế, các thiết bị và các phương pháp truyền và thu tín hiệu phát rộng có thể được đề xuất, mà có thể thu tín hiệu phát rộng số nhờ sử dụng thiết bị thu di động mà không có lỗi bất kể môi trường trong nhà.

Theo sáng chế, việc liệu trường độộn có trong BBF hay không có thể được biết đến nhanh chóng và chính xác bằng cách xác định trường mới chỉ báo xem liệu trường độộn có trong BBF hay không.

Thông tin khác có thể được sử dụng trong trường độộn ngoài phần độộn bằng cách xác định trường độộn mà có trong BBF, nghĩa là, sử dụng loại dữ liệu độộn.

Ngoài ra, trường loại độn có thể được thao tác hiệu quả bằng cách chia và sử dụng một trường loại độn thành hai trường khác nhau.

Các hiệu quả thu được bởi sáng chế không giới hạn ở các hiệu quả nêu trên và các hiệu quả không được đề cập khác sẽ được hiểu rõ bởi những người có hiểu biết trung bình trong lĩnh vực kỹ thuật từ phần mô tả dưới đây.

### **Mô tả vắn tắt các hình vẽ**

Các hình vẽ kèm theo được bao gồm để giúp hiểu rõ hơn sáng chế và được bao gồm theo sáng chế, và cấu thành một phần của sáng chế, minh họa các phương án của sáng chế cùng với phần mô tả chi tiết để mô tả nguyên lý sáng chế.

Fig.1 minh họa cấu trúc của thiết bị truyền tín hiệu phát rộng dùng cho dịch vụ phát rộng thể hệ tiếp theo theo một phương án ví dụ của sáng chế.

Fig.2 minh họa khối định dạng đầu vào theo một phương án ví dụ của sáng chế.

Fig.3 minh họa khối định dạng đầu vào theo phương án ví dụ khác của sáng chế.

Fig.4 minh họa khối định dạng đầu vào theo phương án ví dụ khác nữa của sáng chế.

Fig.5 minh họa khối điều biến và mã hóa đan xen bit (bit interleaved coding & modulation, viết tắt là BICM) theo một phương án ví dụ của sáng chế.

Fig.6 minh họa khối BICM theo phương án ví dụ khác của sáng chế.

Fig.7 minh họa khối xây dựng khung theo một phương án ví dụ của sáng chế.

Fig.8 minh họa khối tạo đa hợp phân chia theo tần số trực giao (Orthogonal Frequency Division Multiplexing, viết tắt là OFDM) theo một phương án ví dụ của sáng chế.

Fig.9 minh họa cấu trúc của thiết bị thu tín hiệu phát rộng dùng cho dịch vụ phát rộng thể hệ tiếp theo theo một phương án ví dụ của sáng chế.

Fig.10 minh họa cấu trúc khung theo một phương án ví dụ của sáng chế.

Fig.11 minh họa cấu trúc lớp truyền tín hiệu của cấu trúc khung theo một phương án ví dụ của sáng chế.

Fig.12 minh họa dữ liệu truyền tín hiệu đoạn đầu theo một phương án ví dụ của sáng chế.

Fig.13 minh họa dữ liệu PLS1 theo một phương án ví dụ của sáng chế.

Fig.14 minh họa dữ liệu PLS2 theo một phương án ví dụ của sáng chế.

Fig.15 minh họa dữ liệu PLS2 theo phương án ví dụ khác của sáng chế.

Fig.16 minh họa cấu trúc logic của khung theo một phương án ví dụ của sáng chế.

Fig.17 minh họa việc ánh xạ truyền tín hiệu lớp vật lý (Physical Layer Signaling, viết tắt là PLS) theo một phương án ví dụ của sáng chế.

Fig.18 minh họa việc ánh xạ kênh cảnh báo khẩn cấp (Emergency Alert Channel, viết tắt là EAC) theo một phương án ví dụ của sáng chế.

Fig.19 minh họa việc ánh xạ kênh thông tin nhanh (Fast Information Channel, viết tắt là FIC) theo một phương án ví dụ của sáng chế.

Fig.20 minh họa loại ống dữ liệu (data pipe, viết tắt là DP) theo một phương án ví dụ của sáng chế.

Fig.21 minh họa loại ánh xạ ống dữ liệu (DP) theo một phương án ví dụ của sáng chế.

Fig.22 minh họa cấu trúc hiệu chỉnh lỗi phía trước (Forward Error Correction, viết tắt là FEC) theo một phương án ví dụ của sáng chế.

Fig.23 minh họa sự đan xen bit theo một phương án ví dụ của sáng chế.

Fig.24 minh họa sự giải đa hợp ô-từ theo một phương án ví dụ của sáng chế.

Fig.25 minh họa sự đan xen thời gian theo một phương án ví dụ của sáng chế.

Fig.26 minh họa hoạt động cơ bản của bộ đan xen khối hàng-cột xoắn theo một phương án ví dụ của sáng chế.

Fig.27 minh họa hoạt động của bộ đan xen khối hàng-cột xoắn theo phương

án ví dụ khác của sáng chế.

Fig.28 minh họa mẫu đọc chéo của bộ đan xen khối hàng-cột xoắn theo một phương án ví dụ của sáng chế.

Fig.29 minh họa XFECBLOCK được đan xen từ mỗi mảng đan xen theo một phương án ví dụ của sáng chế.

Fig.30 là sơ đồ minh họa một ví dụ về môđun đồng bộ hóa và giải điều biến trên Fig.9.

Fig.31 là sơ đồ minh họa một ví dụ về môđun tách khung trên Fig.9.

Fig.32 là sơ đồ minh họa một ví dụ về môđun giải ánh xạ và giải mã trên Fig.9.

Fig.33 là sơ đồ minh họa một ví dụ về bộ xử lý đầu ra trên Fig.9.

Fig.34 là sơ đồ minh họa ví dụ khác về bộ xử lý đầu ra trên Fig.9.

Fig.35 minh họa môđun mã hóa và điều biến theo phương án ví dụ khác của sáng chế.

Fig.36 là sơ đồ minh họa môđun giải ánh xạ và giải mã theo phương án ví dụ khác của sáng chế.

Fig.37 là sơ đồ minh họa một ví dụ về môđun thích ứng chế độ được đề xuất theo sáng chế.

Fig.38 là sơ đồ minh họa một ví dụ về bộ xử lý đầu ra được đề xuất theo sáng chế.

Fig.39 là sơ đồ minh họa một ví dụ về cấu trúc khung BB theo kỹ thuật đã biết.

Fig.40 là sơ đồ minh họa ví dụ khác về cấu trúc khung BB theo kỹ thuật đã biết.

Fig.41 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB theo kỹ thuật đã biết.

Fig.42 minh họa một ví dụ về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.43 là sơ đồ minh họa ví dụ khác về cấu trúc khung BB được đề xuất theo

sáng chế.

Fig.44 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.45 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.46 là sơ đồ minh họa sự so sánh kết quả tính toán chi phí truyền khung BB trong các cấu trúc khung BB khác nhau.

Fig.47 minh họa một ví dụ về cấu trúc khung BB theo kỹ thuật đã biết.

Fig.48 là sơ đồ minh họa một ví dụ về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.49 là sơ đồ minh họa ví dụ khác về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.50 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.51 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.52 là lưu đồ minh họa một ví dụ về phương pháp truyền tín hiệu phát rộng được đề xuất theo sáng chế.

Fig.53 là lưu đồ minh họa một ví dụ về phương pháp thu tín hiệu phát rộng được đề xuất theo sáng chế.

### **Mô tả chi tiết sáng chế**

Sự tham chiếu sẽ được thực hiện chi tiết cho các phương án ưu tiên của sáng chế, các ví dụ của các phương án này được minh họa trên các hình vẽ kèm theo. Phần mô tả chi tiết, mà sẽ được đưa ra dưới đây có dựa vào các hình vẽ kèm theo, nhằm giải thích các phương án ví dụ của sáng chế, hơn là để thể hiện chỉ các phương án mà có thể được thực hiện theo sáng chế. Phần mô tả chi tiết dưới đây bao gồm các phần chi tiết cụ thể để nhằm hiểu sâu sáng chế. Tuy nhiên, nó sẽ trở nên rõ ràng đối với những người có hiểu biết trung bình trong lĩnh vực kỹ thuật mà sáng chế có

thể được thực hiện mà không cần đến các phân chi tiết cụ thể như vậy.

Mặc dù hầu hết các thuật ngữ được sử dụng theo sáng chế đã được lựa chọn từ các thuật ngữ chung được sử dụng rộng rãi trong lĩnh vực kỹ thuật, một số thuật ngữ đã được lựa chọn theo cách tùy ý bởi người nộp đơn và nghĩa của chúng được giải thích chi tiết trong phần mô tả dưới đây nếu cần. Do đó, sáng chế cần được hiểu là được dựa trên các nghĩa dự định của các thuật ngữ hơn là theo các tên gọi hoặc nghĩa đơn giản của chúng.

Sáng chế đề xuất các thiết bị và các phương pháp truyền và thu các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai. Các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế bao gồm dịch vụ phát rộng mặt đất, dịch vụ phát rộng di động, dịch vụ UHDTV, v.v.. Sáng chế có thể xử lý các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai qua không phải là MIMO (non-Multiple Input Multiple Output – không phải là nhiều đầu vào nhiều đầu ra) hoặc MIMO theo một phương án. Sơ đồ không phải là MIMO theo một phương án của sáng chế có thể bao gồm sơ đồ MISO (Multiple Input Single Output – nhiều đầu vào một đầu ra), sơ đồ SISO (Single Input Single Output – một đầu vào một đầu ra), v.v..

Trong khi MISO hoặc MIMO sử dụng hai anten như dưới đây để thuận tiện cho việc mô tả, sáng chế có thể ứng dụng cho các hệ thống sử dụng hai hoặc nhiều hơn hai anten.

Sáng chế có thể xác định ba biên dạng lớp vật lý (PL) – các biên dạng cơ sở, cầm tay và nâng cao – mỗi trong số chúng được tối ưu để giảm thiểu tính phức tạp của bộ thu trong khi đạt được hiệu suất được yêu cầu đối với trường hợp sử dụng cụ thể. Các biên dạng lớp vật lý (PHY) này là các tập hợp con của tất cả các cấu hình mà bộ thu tương ứng phải thực hiện.

Ba biên dạng PHY chia sẻ hầu hết các khối chức năng nhưng hơi khác ở các khối và/hoặc các thông số cụ thể. Các biên dạng PHY bổ sung có thể được xác định trong tương lai. Đối với sự phát triển của hệ thống, các biên dạng tương lai có thể cũng được đa hợp với các biên dạng hiện tại trong kênh RF đơn qua khung mở rộng tương lai (Future Extension Frame, viết tắt là FEF). Các phân chi tiết của mỗi biên dạng PHY được mô tả dưới đây.

## 1. Biên dạng cơ sở

Biên dạng cơ sở thể hiện trường hợp sử dụng chính cho các thiết bị thu cố định mà thường được kết nối với anten trên mái nhà. Biên dạng cơ sở cũng bao gồm các thiết bị cầm tay mà có thể được vận chuyển tới một nơi nhưng thuộc kiểu thu tương đối cố định. Việc sử dụng biên dạng cơ sở có thể được mở rộng đến các thiết bị cầm tay hoặc thậm chí trong xe cộ trong một số cách thực hiện được cải tiến, nhưng các trường hợp sử dụng này là không mong muốn đối với hoạt động của bộ thu biên dạng cơ sở.

Khoảng thu SNR đích là từ khoảng 10 đến 20 dB, mà bao gồm dung lượng thu SNR 15 dB của hệ thống phát rộng hiện có (ví dụ ATSC A/53). Tính phức tạp của bộ thu và sự tiêu thụ điện không mang tính quyết định như trong các thiết bị cầm tay hoạt động bằng ắc quy, mà sẽ sử dụng biên dạng cầm tay. Các thông số hệ thống quan trọng đối với biên dạng cơ sở được thể hiện trong bảng 1 dưới đây.

Bảng 1

|                                          |                                                  |
|------------------------------------------|--------------------------------------------------|
| Độ dài từ mã LDPC                        | 16K, 64K bit                                     |
| Kích thước chòm điểm                     | 4~10 bpcu (các bit cho mỗi lần sử dụng kênh)     |
| Kích thước bộ nhớ giải đan xen thời gian | $\leq 2^{19}$ ô dữ liệu                          |
| Các mẫu tín hiệu dẫn đường               | Mẫu tín hiệu dẫn đường dùng cho việc thu cố định |
| Kích thước FFT                           | 16K, 32K điểm                                    |

## 2. Biên dạng cầm tay

Biên dạng cầm tay được định rõ để sử dụng trong các thiết bị xe và cầm mà hoạt động với nguồn điện ắc quy. Các thiết bị có thể dịch chuyển với tốc độ đi bộ hoặc tốc độ di chuyển của xe. Sự tiêu thụ điện cũng như tính phức tạp của bộ thu là rất quan trọng đối với sự thực hiện của các thiết bị của biên dạng cầm tay. Khoảng SNR đích của biên dạng cầm tay là khoảng từ 0 đến 10 dB, nhưng có thể được tạo cấu hình để đạt được dưới 0 dB khi được dự định để thu sâu ở môi trường trong nhà.

Ngoài dung lượng SNR thấp, biên dạng đàn hồi với hiệu ứng Doppler gây ra bởi tính di động của bộ thu là thuộc tính hiệu suất quan trọng nhất của biên dạng cầm tay. Các thông số hệ thống quan trọng đối với biên dạng cầm tay được thể hiện trong bảng 2 dưới đây.

Bảng 2

|                                          |                                                                |
|------------------------------------------|----------------------------------------------------------------|
| Độ dài từ mã LDPC                        | 16 Kbit                                                        |
| Kích thước chòm điểm                     | 2~8 bpcu                                                       |
| Kích thước bộ nhớ giải đan xen thời gian | $\leq 2^{18}$ ô dữ liệu                                        |
| Các mẫu tín hiệu dẫn đường               | Các mẫu tín hiệu dẫn đường đối với sự thu di động và trong nhà |
| Kích thước FFT                           | 8K, 16K điểm                                                   |

### 3. Biên dạng nâng cao

Biên dạng nâng cao cung cấp dung lượng kênh cao nhất đánh đổi bằng tính phức tạp cao hơn khi thực hiện. Biên dạng này yêu cầu sử dụng việc truyền và thu MIMO, và dịch vụ UHD TV là trường hợp sử dụng đích mà biên dạng này được thiết kế dành riêng. Dung lượng gia tăng này cũng có thể được sử dụng để cho phép số lượng các dịch vụ được gia tăng trong độ rộng dải tần được quy định, ví dụ, nhiều dịch vụ SDTV hoặc HDTV.

Khoảng SNR đích của biên dạng nâng cao là từ khoảng 20 đến 30 dB. Việc truyền MIMO có thể ban đầu sử dụng thiết bị truyền phân cực elip hiện có, với sự mở rộng đến việc truyền phân cực ngang toàn năng lượng trong tương lai. Các thông số hệ thống quan trọng đối với biên dạng nâng cao được thể hiện trong bảng 3 dưới đây.

Bảng 3

|                                          |                         |
|------------------------------------------|-------------------------|
| Độ dài từ mã LDPC                        | 16K, 64K bit            |
| Kích thước chòm điểm                     | 8~12 bpcu               |
| Kích thước bộ nhớ giải đan xen thời gian | $\leq 2^{19}$ ô dữ liệu |

|                            |                                               |
|----------------------------|-----------------------------------------------|
| Các mẫu tín hiệu dẫn đường | Mẫu tín hiệu dẫn đường đối với sự thu cố định |
| Kích thước FFT             | 16K, 32K điểm                                 |

Trong trường hợp này, biên dạng cơ sở có thể được sử dụng làm biên dạng cho cả dịch vụ phát rộng mặt đất và dịch vụ phát rộng di động. Nghĩa là, biên dạng cơ sở có thể được sử dụng để xác định khái niệm biên dạng mà bao gồm biên dạng di động. Ngoài ra, biên dạng nâng cao có thể được chia thành biên dạng nâng cao đối với biên dạng cơ sở với MIMO và biên dạng nâng cao đối với biên dạng cầm tay với MIMO. Ngoài ra, ba biên dạng có thể được thay đổi theo mục đích của nhà thiết kế.

Các thuật ngữ và các định nghĩa dưới đây có thể được áp dụng cho sáng chế. Các thuật ngữ và các định nghĩa dưới đây có thể được thay đổi theo thiết kế.

Dòng phụ trợ: dãy các ô mang dữ liệu của việc điều biến chưa được xác định và việc mã hóa, mà có thể được sử dụng để mở rộng trong tương lai hoặc khi được yêu cầu bởi các nhà quảng bá hoặc các nhà điều hành mạng.

Ổng dữ liệu cơ sở: ổng dữ liệu mà mang dữ liệu truyền tín hiệu dịch vụ.

Khung dải gốc (hoặc BBFRAME): tập hợp các bit K<sub>bch</sub> mà tạo nên đầu vào cho một quy trình mã hóa FEC (mã hóa BCH và LDPC).

Ô: giá trị điều biến mà được mang bởi một sóng mang của việc truyền OFDM.

Khối được mã hóa: khối LDPC được mã hóa của dữ liệu PLS1 hoặc một trong số các khối LDPC được mã hóa của dữ liệu PLS2.

Ổng dữ liệu: kênh logic trong lớp vật lý mà mang dữ liệu dịch vụ hoặc siêu dữ liệu liên quan, mà có thể mang một hoặc nhiều dịch vụ hoặc thành phần dịch vụ.

Đơn vị ổng dữ liệu: đơn vị cơ sở để cấp phát các ô dữ liệu đến DP trong khung.

Ký hiệu dữ liệu: ký hiệu OFDM trong khung mà không phải là ký hiệu đoạn đầu (ký hiệu truyền tín hiệu khung và ký hiệu biên khung được bao gồm trong ký hiệu dữ liệu).

DP\_ID: trường 8 bit này nhận dạng duy nhất một DP trong hệ thống được nhận dạng bởi SYSTEM\_ID.

Ô giả: ô mang giá trị giả ngẫu nhiên được sử dụng để điền vào dung lượng còn lại không được sử dụng để truyền tín hiệu PLS, các DP hoặc các dòng phụ trợ.

Kênh cảnh báo khẩn cấp: phân khung mà mang dữ liệu thông tin EAS.

Khung: khe thời gian lớp vật lý mà bắt đầu bằng đoạn đầu và kết thúc với ký hiệu biên khung.

Đơn vị lặp lại khung: tập hợp các khung thuộc biên dạng lớp vật lý giống hoặc khác nhau bao gồm FEF, mà được lặp lại tám lần trong một siêu khung.

Kênh thông tin nhanh: kênh logic trong khung mà mang thông tin ánh xạ giữa dịch vụ và DP cơ sở tương ứng.

FECBLOCK: tập hợp các bit LDPC được mã hóa của dữ liệu DP.

Kích thước FFT: kích thước FFT danh định được sử dụng cho chế độ cụ thể, bằng với khoảng thời gian ký hiệu hoạt động  $T_s$  được thể hiện trong các chu kỳ thời gian cơ bản  $T$ .

Ký hiệu truyền tín hiệu khung: ký hiệu OFDM với mật độ tín hiệu dẫn đường cao được sử dụng ở thời điểm bắt đầu của khung theo cách kết hợp nhất định của kích thước FFT, khoảng bảo vệ và mẫu tín hiệu dẫn đường rời rạc, mà mang một phần của dữ liệu PLS.

Ký hiệu biên khung: ký hiệu OFDM với mật độ tín hiệu dẫn đường cao được sử dụng ở thời điểm kết thúc của khung theo cách kết hợp nhất định của kích thước FFT, khoảng bảo vệ và mẫu tín hiệu dẫn đường rời rạc.

Nhóm khung: tập hợp tất cả các khung có cùng loại biên dạng PHY trong một siêu khung.

Khung mở rộng tương lai: khe thời gian lớp vật lý nằm trong siêu khung mà có thể được sử dụng để mở rộng trong tương lai, mà bắt đầu bằng đoạn đầu.

Hệ thống Futurecast UTB: hệ thống phát rộng lớp vật lý được đề xuất, trong đó đầu vào là một hoặc nhiều MPEG2-TS hoặc IP hoặc (các) dòng thông thường và trong đó đầu ra là tín hiệu RF.

Dòng đầu vào: Dòng dữ liệu cho tập hợp các dịch vụ được phân phối tới những

người dùng cuối cùng bởi hệ thống.

Ký hiệu dữ liệu thông thường: ký hiệu dữ liệu ngoại trừ ký hiệu truyền tín hiệu khung và ký hiệu biên khung.

Biên dạng PHY: tập hợp con của tất cả các cấu hình mà bộ thu tương ứng phải thực hiện.

PLS: truyền tín hiệu lớp vật lý dữ liệu bao gồm PLS1 và PLS2.

PLS1: tập hợp dữ liệu PLS thứ nhất được mang trong các ký hiệu FSS có kích thước cố định, mã hóa và điều biến, mà mang thông tin cơ bản về hệ thống cũng như các thông số cần để giải mã PLS2.

Lưu ý: dữ liệu PLS1 giữ nguyên không đổi trong khoảng thời gian của nhóm khung.

PLS2: tập hợp dữ liệu PLS thứ hai được truyền trong ký hiệu FSS, mà mang dữ liệu PLS chi tiết hơn về hệ thống và các DP.

Dữ liệu động PLS2: dữ liệu PLS2 mà có thể thay đổi động theo từng khung.

Dữ liệu tĩnh PLS2: dữ liệu PLS2 mà duy trì tĩnh trong khoảng thời gian của nhóm khung.

Dữ liệu truyền tín hiệu đoạn đầu: dữ liệu truyền tín hiệu được mang bởi ký hiệu đoạn đầu và được sử dụng để nhận dạng chế độ cơ bản của hệ thống.

Ký hiệu đoạn đầu: ký hiệu tín hiệu dẫn đường độ dài cố định mà mang dữ liệu PLS cơ sở và được đặt ở phần bắt đầu của khung

Lưu ý: ký hiệu đoạn đầu chủ yếu được sử dụng để quét nhanh dải ban đầu để phát hiện tín hiệu hệ thống, sự định giờ, dịch vị thời gian, và kích thước FFT của nó.

Dự trữ để sử dụng trong tương lai: không được xác định bởi sáng chế này nhưng có thể được xác định trong tương lai.

Siêu khung: tập hợp các đơn vị lặp lại tám khung.

Khối đan xen thời gian (khối TI): tập hợp các ô trong đó sự đan xen thời gian được thực hiện, tương ứng với một lần sử dụng bộ nhớ đan xen thời gian.

Nhóm TI: đơn vị mà trên đó sự cấp phát dung lượng động cho DP cụ thể được thực hiện, tạo thành từ số nguyên, thay đổi động của các XFECBLOCK.

Lưu ý: Nhóm TI có thể được ánh xạ trực tiếp tới một khung hoặc có thể được ánh xạ đến nhiều khung. Nó có thể chứa một hoặc nhiều khối TI.

DP loại 1: DP của khung mà ở đó tất cả các DP được ánh xạ vào trong khung theo kiểu TDM.

DP loại 2: DP của khung mà ở đó tất cả các DP được ánh xạ vào trong khung kiểu FDM.

XFECBLOCK: tập hợp các ô Ncells mang tất cả các bit của một LDPC FECBLOCK.

Fig.1 minh họa cấu trúc của thiết bị để truyền các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế.

Thiết bị để truyền các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế có thể bao gồm khối định dạng đầu vào 1000, khối BICM (Bit Interleaved Coding & Modulation – điều biến và mã hóa đan xen bit) 1010, khối cấu trúc khung 1020, khối tạo OFDM (Orthogonal Frequency Division Multiplexing - đa hợp phân chia theo tần số trực giao) 1030 và khối tạo truyền tín hiệu 1040. Hoạt động của mỗi môđun của thiết bị để truyền các tín hiệu phát rộng sẽ được mô tả.

Khối định dạng đầu vào 1000 có thể được biểu diễn thành bộ định dạng đầu vào.

Khối BICM (điều biến và mã hóa đan xen bit) 1010 có thể được thể hiện thành bộ mã hóa.

Khối cấu trúc khung 1020 có thể được biểu thị thành bộ xây dựng khung.

Khối tạo OFDM (đa hợp phân chia theo tần số trực giao) 1030 có thể được biểu diễn thành bộ điều biến.

Các gói/dòng IP và MPEG2-TS là các định dạng đầu vào chính, các loại dòng khác được xử lý như các dòng thông thường. Ngoài các dữ liệu đầu vào này, thông tin quản lý được nhập vào để điều khiển việc lập lịch và cấp phát của dải tần tương

ứng đối với mỗi dòng đầu vào. Một hoặc nhiều dòng TS đầu vào, dòng IP và/hoặc (các) dòng thông thường được cho phép đồng thời.

Khối định dạng đầu vào 1000 có thể giải đa hợp mỗi dòng đầu vào thành một hoặc nhiều ống dữ liệu, mà mỗi trong số chúng việc mã hóa và điều biến độc lập được áp dụng. Ống dữ liệu (DP) là đơn vị cơ sở để điều khiển độ mạnh, do đó ảnh hưởng tới chất lượng dịch vụ (QoS). Một hoặc nhiều dịch vụ hoặc thành phần dịch vụ có thể được mang bởi một DP duy nhất. Chi tiết về các thao tác của khối định dạng đầu vào 1000 sẽ được mô tả sau đây.

Ống dữ liệu là kênh logic trong lớp vật lý mà mang dữ liệu dịch vụ hoặc siêu dữ liệu liên quan, mà có thể mang một hoặc nhiều dịch vụ hoặc thành phần dịch vụ.

Ống dữ liệu có thể được biểu diễn thành kênh truyền dữ liệu.

Ngoài ra, đơn vị ống dữ liệu: đơn vị cơ sở để cấp phát các ô dữ liệu đến DP trong khung.

Trong khối BICM 1010, dữ liệu chẵn lẻ được bổ sung để hiệu chỉnh lỗi và các dòng bit được mã hóa được ánh xạ đến các ký hiệu chùm điểm giá trị phức tạp. Các ký hiệu được đan xen ngang độ sâu đan xen cụ thể mà được sử dụng cho DP tương ứng. Đối với biên dạng nâng cao, mã hóa MIMO được thực hiện trong khối BICM 1010 và đường dẫn dữ liệu bổ sung được bổ sung ở đầu ra để truyền MIMO. Chi tiết về các thao tác của khối BICM 1010 sẽ được mô tả sau đây.

Khối định dạng đầu vào trên Fig.1 thực hiện các chức năng, các quy trình, và/hoặc các phương pháp được đề xuất trên Fig.50, Fig.51, và Fig.52 được mô tả dưới đây.

Khối xây dựng khung 1020 có thể ánh xạ các ô dữ liệu của các DP đầu vào thành các ký hiệu OFDM nằm trong khung. Sau khi ánh xạ, sự đan xen tần số được sử dụng để phân tập miền tần số, cụ thể là để chống lại các kênh mờ dần lựa chọn tần số. Chi tiết về các thao tác của khối xây dựng khung 1020 sẽ được mô tả sau đây.

Sau khi chèn phần đầu tại điểm đầu của mỗi khung, khối tạo OFDM 1030 có thể áp dụng sự điều biến OFDM thông thường có tiền tố tuần hoàn là khoảng bảo vệ. Đối với sự phân tập không gian anten, sơ đồ MISO được phân bố được áp dụng trên các bộ truyền. Ngoài ra, sơ đồ giảm công suất đỉnh trên trung bình (Peak-to-

Average Power Reduction, viết tắt là PAPR)) được thực hiện trong miền thời gian. Đối với quy hoạch mạng linh hoạt, đề xuất này cung cấp tập hợp các kích thước FFT, các khoảng bảo vệ độ dài và các mẫu tín hiệu dẫn đường tương ứng khác nhau. Chi tiết về các thao tác của khối tạo OFDM 1030 sẽ được mô tả sau đây.

Khối tạo truyền tín hiệu 1040 có thể tạo ra thông tin truyền tín hiệu lớp vật lý được sử dụng cho thao tác của mỗi khối chức năng. Thông tin truyền tín hiệu này cũng được truyền sao cho các dịch vụ theo sở thích được khôi phục hoàn toàn ở phía bộ thu. Chi tiết về các thao tác của khối tạo truyền tín hiệu 1040 sẽ được mô tả sau đây.

Fig.2, Fig.3 và Fig.4 minh họa khối định dạng đầu vào 1000 theo các phương án của sáng chế. Mỗi hình vẽ sẽ được mô tả.

Khối định dạng đầu vào trên các hình vẽ từ Fig.2 đến Fig.4 thực hiện các chức năng, các quy trình, và/hoặc các phương pháp được đề xuất trên Fig.50, Fig.51, và Fig.52 được mô tả dưới đây.

Fig.2 minh họa khối định dạng đầu vào theo một phương án của sáng chế. Fig.2 thể hiện mô đun định dạng đầu vào khi tín hiệu đầu vào là dòng đầu vào đơn.

Khối định dạng đầu vào được minh họa trên Fig.2 tương ứng với một phương án của khối định dạng đầu vào 1000 được mô tả dựa vào Fig.1.

Đầu vào tới lớp vật lý có thể bao gồm một hoặc nhiều dòng dữ liệu. Mỗi dòng dữ liệu được mang bởi một DP. Các mô đun thích ứng chế độ cắt dòng dữ liệu đến thành các trường dữ liệu của khung dải gốc (BBF). Hệ thống hỗ trợ ba loại dòng dữ liệu đầu vào: MPEG2-TS, giao thức internet (IP) và dòng chung (GS). MPEG2-TS được tạo đặc tính bởi các gói độ dài cố định (188 byte) với byte thứ nhất là byte đồng bộ (0x47). Dòng IP bao gồm các gói dữ liệu IP độ dài khác nhau, như được truyền tín hiệu trong các phần đầu gói IP. Hệ thống hỗ trợ cả IPv4 và IPv6 đối với dòng IP. GS có thể bao gồm các gói độ dài biến đổi hoặc các gói độ dài không đổi, được truyền tín hiệu trong các phần đầu gói đóng kín.

(a) thể hiện khối thích ứng chế độ 2000 và thích ứng dòng 2010 đối với tín hiệu DP và (b) thể hiện khối tạo PLS 2020 và bộ xáo trộn PLS 2030 để tạo và xử lý dữ liệu PLS. Thao tác của mỗi khối sẽ được mô tả dưới đây.

Bộ tách dòng đầu vào tách các dòng TS, IP, GS đầu vào thành nhiều dòng dịch vụ hoặc dòng thành phần dịch vụ (audio, video, v.v.). Môđun thích ứng chế độ 2010 được cấu thành từ bộ mã hóa CRC, bộ cắt khung (dải tần) BB, và khối chèn phần đầu khung BB.

Bộ mã hóa CRC cung cấp ba loại mã hóa CRC để phát hiện lỗi ở mức gói người dùng (UP), nghĩa là, CRC-8, CRC-16, và CRC-32. Các byte CRC được tính được thêm vào sau UP. CRC-8 được sử dụng cho dòng TS và CRC-32 cho dòng IP. Nếu dòng GS không cung cấp mã hóa CRC, mã hóa CRC được đề xuất nên được áp dụng.

Bộ cắt khung BB ánh xạ đầu vào thành định dạng bit logic bên trong. Bit thu được thứ nhất được xác định là MSB. Bộ cắt khung BB cấp phát số lượng các bit đầu vào bằng với dung lượng trường dữ liệu khả dụng. Để phân phối số lượng các bit đầu vào bằng với phụ tải BBF, dòng gói UP được cắt để vừa với trường dữ liệu của BBF.

Khối chèn phần đầu khung BB có thể chèn phần đầu BBF độ dài cố định là 2 byte được chèn vào phía trước khung BB. Phần đầu BBF bao gồm STUFFI (1 bit), SYNCD (13 bit), và RFU (2 bit). Ngoài phần đầu BBF 2 byte cố định, BBF có thể có trường mở rộng (1 hoặc 3 byte) ở thời điểm kết thúc của phần đầu BBF 2 byte.

Thích ứng dòng 2010 bao gồm khối chèn độn và bộ xáo trộn BB.

Khối chèn độn có thể chèn trường độn vào phụ tải của khung BB. Nếu dữ liệu đầu vào dùng cho sự thích ứng dòng là đủ để điền vào khung BB, STUFFI được thiết đặt là '0' và BBF không có trường độn. Nếu không thì STUFFI được thiết đặt là '1' và trường độn được chèn ngay sau phần đầu BBF. Trường độn bao gồm hai byte của phần đầu trường độn và dữ liệu độn có kích thước biến đổi.

Bộ xáo trộn BB xáo trộn hoàn toàn BBF để phân tán năng lượng. Tần số xáo trộn là đồng bộ với BBF. Tần số xáo trộn được tạo ra bởi bộ đăng ký dịch chuyển phản hồi.

Khối tạo PLS 2020 có thể tạo ra dữ liệu truyền tín hiệu lớp vật lý (PLS). PLS cung cấp bộ thu nhằm truy cập các DP lớp vật lý. Dữ liệu PLS bao gồm dữ liệu PLS1 và dữ liệu PLS2.

Dữ liệu PLS1 là tập hợp dữ liệu PLS thứ nhất được mang trong các ký hiệu FSS trong khung có kích thước cố định, mã hóa và điều biến, mà mang thông tin cơ bản về hệ thống cũng như các thông số cần để giải mã dữ liệu PLS2. Dữ liệu PLS1 cung cấp các thông số truyền cơ bản bao gồm các thông số được yêu cầu để cho phép thu và giải mã của dữ liệu PLS2. Ngoài ra, dữ liệu PLS1 giữ nguyên không đổi trong khoảng thời gian của nhóm khung.

Dữ liệu PLS2 là tập hợp dữ liệu PLS thứ hai được truyền trong ký hiệu FSS, mà mang dữ liệu PLS chi tiết hơn về hệ thống và các DP. PLS2 chứa các thông số mà cung cấp thông tin đủ cho bộ thu để giải mã DP mong muốn. Truyền tín hiệu PLS2 còn bao gồm hai loại thông số, dữ liệu tĩnh PLS2 (dữ liệu PLS2-STAT) và dữ liệu động PLS2 (dữ liệu PLS2-DYN). Dữ liệu tĩnh PLS2 là dữ liệu PLS2 mà duy trì tĩnh trong khoảng thời gian của nhóm khung và dữ liệu động PLS2 là dữ liệu PLS2 mà có thể thay đổi động theo từng khung.

Các chi tiết về dữ liệu PLS sẽ được mô tả sau đây.

Bộ xáo trộn PLS 2030 có thể xáo trộn dữ liệu PLS được tạo ra để phân tán năng lượng.

Các khối nêu trên có thể được bỏ qua hoặc được thay thế bởi các khối có các chức năng giống hoặc tương tự nhau.

Fig.3 minh họa khối định dạng đầu vào theo phương án khác của sáng chế.

Khối định dạng đầu vào được minh họa trên Fig.3 tương ứng với một phương án của khối định dạng đầu vào 1000 được mô tả dựa vào Fig.1.

Fig.3 thể hiện khối thích ứng chế độ của khối định dạng đầu vào khi tín hiệu đầu vào tương ứng với nhiều dòng đầu vào.

Khối thích ứng chế độ của khối định dạng đầu vào để xử lý nhiều dòng đầu vào có thể xử lý độc lập nhiều dòng đầu vào.

Dựa vào Fig.3, khối thích ứng chế độ để xử lý tương ứng nhiều dòng đầu vào có thể bao gồm bộ tách dòng đầu vào 3000, bộ đồng bộ dòng đầu vào 3010, khối trễ bù 3020, khối xóa gói trống 3030, khối nén phần đầu 3040, bộ mã hóa CRC 3050, bộ cắt khung BB 3060 và khối chèn phần đầu BB 3070. Mỗi khối của khối thích ứng

chế độ sẽ được mô tả.

Các thao tác của bộ mã hóa CRC 3050, bộ cắt khung BB 3060 và khối chèn phần đầu BB 3070 tương ứng với các thao tác của bộ mã hóa CRC, bộ cắt khung BB và khối chèn phần đầu BB được mô tả dựa vào Fig.2 và do đó phần mô tả của nó được bỏ qua.

Bộ tách dòng đầu vào 3000 có thể tách các dòng TS, IP, GS đầu vào thành nhiều dòng dịch vụ hoặc dòng thành phần dịch vụ (audio, video, v.v.).

Bộ đồng bộ dòng đầu vào 3010 có thể được gọi là ISSY. ISSY có thể cung cấp cách thức thích hợp để đảm bảo tốc độ bit không đổi (Constant Bit Rate, viết tắt là CBR) và độ trễ truyền đầu đến đầu không đổi cho định dạng dữ liệu đầu vào bất kỳ. ISSY thường được sử dụng cho trường hợp nhiều DP mang TS, và được sử dụng tùy ý cho nhiều DP mang các dòng GS.

Khối trễ bù 3020 có thể làm trễ dòng gói TS tách sau khi chèn thông tin ISSY để cho phép cơ chế tổ hợp gói TS mà không cần bộ nhớ bổ sung trong bộ thu.

Khối xóa gói trống 3030, được sử dụng chỉ đối với trường hợp dòng đầu vào TS. Một số dòng đầu vào TS hoặc các dòng TS tách có thể có số lượng lớn các gói trống có mặt để điều tiết các dịch vụ VBR (tỷ lệ bit thay đổi) trong dòng CBR TS. Trong trường hợp này, để tránh chi phí truyền không cần thiết, các gói trống có thể được xác định và không được truyền. Trong bộ thu, các gói trống được loại bỏ có thể được chèn lại vào chính vị trí cũ bằng cách tham chiếu đến bộ đếm gói trống được xóa (DNP) mà được chèn trong khi truyền, do đó đảm bảo tỷ lệ bit không đổi và không cần cập nhật tem thời gian (PCR).

Khối nén phần đầu 3040 có thể cung cấp nén phần đầu gói để tăng hiệu suất truyền đối với các dòng đầu vào IP hoặc TS. Bởi vì bộ thu có thể có thông tin tiên nghiệm trong các phần nhất định của phần đầu, thông tin đã biết này có thể được xóa trong bộ truyền.

Đối với dòng vận chuyển, bộ thu có thông tin tiên nghiệm về cấu hình đồng bộ byte (0x47) và độ dài gói (188 Byte). Nếu dòng TS đầu vào mang nội dung mà chỉ có một PID, nghĩa là, đối với chỉ một thành phần dịch vụ (video, audio, v.v.) hoặc thành phần con dịch vụ (lớp cơ bản SVC, lớp tăng cường SVC, dạng cơ bản MVC

hoặc các dạng phụ thuộc MVC), việc nén phần đầu gói TS có thể được áp dụng (tùy ý) cho dòng vận chuyển. Việc nén phần đầu gói IP được sử dụng tùy ý nếu dòng đầu vào là dòng IP.

Các khối nêu trên có thể được bỏ qua hoặc được thay thế bởi các khối có các chức năng giống hoặc tương tự nhau.

Fig.4 minh họa khối định dạng đầu vào theo phương án khác của sáng chế.

Khối định dạng đầu vào được minh họa trên Fig.4 tương ứng với một phương án của khối định dạng đầu vào 1000 được mô tả dựa vào Fig.1.

Fig.4 minh họa khối thích ứng dòng của môđun định dạng đầu vào khi tín hiệu đầu vào tương ứng với nhiều dòng đầu vào.

Dựa vào Fig.4, khối thích ứng chế độ để xử lý tương ứng nhiều dòng đầu vào có thể bao gồm bộ lập lịch 4000, khối trễ 1 khung 4010, khối chèn độn 4020, truyền tín hiệu trong dải 4030, bộ xáo trộn khung BB 4040, khối tạo PLS 4050 và bộ xáo trộn PLS 4060. Mỗi khối của khối thích ứng dòng sẽ được mô tả sau đây.

Các thao tác của khối chèn độn 4020, bộ xáo trộn khung BB 4040, khối tạo PLS 4050 và bộ xáo trộn PLS 4060 tương ứng với các thao tác của khối chèn độn, bộ xáo trộn BB, khối tạo PLS và bộ xáo trộn PLS được mô tả dựa vào Fig.2 và do đó phần mô tả của nó được bỏ qua.

Bộ lập lịch 4000 có thể xác định việc cấp phát ô tổng quan trên toàn bộ khung từ lượng của các FECFRAME của mỗi DP. Bao gồm sự cấp phát đối với PLS, EAC và FIC, bộ lập lịch tạo ra các giá trị của dữ liệu PLS2-DYN, mà được truyền như truyền tín hiệu trong dải hoặc ô PLS trong FSS của khung. Các chi tiết về FECBLOCK, EAC và FIC sẽ được mô tả sau đây.

Khối trễ 1 khung 4010 có thể làm trễ dữ liệu đầu vào bởi một khung truyền sao cho thông tin lập lịch về khung tiếp theo có thể được truyền qua khung hiện tại đối với thông tin truyền tín hiệu trong dải cần được chèn vào các DP.

Truyền tín hiệu trong dải 4030 có thể chèn phần không được làm trễ của dữ liệu PLS2 vào DP của khung.

Các khối nêu trên có thể được bỏ qua hoặc được thay thế bởi các khối có các

chức năng giống hoặc tương tự nhau.

Fig.5 minh họa khối BICM theo một phương án của sáng chế.

Khối BICM được minh họa trên Fig.5 tương ứng với một phương án của khối BICM 1010 được mô tả dựa vào Fig.1.

Như được nêu trên, thiết bị truyền các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế có thể cung cấp dịch vụ phát rộng mặt đất, dịch vụ phát rộng di động, dịch vụ UHDTV, v.v..

Vì QoS (chất lượng dịch vụ) phụ thuộc vào các đặc tính của dịch vụ được cung cấp bởi thiết bị để truyền các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế, dữ liệu tương ứng với các dịch vụ tương ứng cần được xử lý qua các sơ đồ khác nhau. Do đó, khối BICM theo một phương án của sáng chế có thể xử lý độc lập các DP được đưa vào đó bằng cách áp dụng độc lập các sơ đồ SISO, MISO và MIMO cho các ống dữ liệu theo thứ tự tương ứng với các đường dữ liệu. Kết quả là, thiết bị để truyền các tín hiệu phát rộng dùng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế có thể điều khiển QoS đối với mỗi dịch vụ hoặc thành phần dịch vụ được truyền qua mỗi DP.

(a) thể hiện khối BICM được chia sẻ bởi biên dạng cơ sở và biên dạng cầm tay và (b) thể hiện khối BICM của biên dạng nâng cao.

Khối BICM được chia sẻ bởi biên dạng cơ sở và biên dạng cầm tay và khối BICM của biên dạng nâng cao có thể bao gồm nhiều khối xử lý để xử lý mỗi DP.

Mỗi khối xử lý của khối BICM đối với biên dạng cơ sở và biên dạng cầm tay và khối BICM đối với biên dạng nâng cao sẽ được mô tả sau đây.

Khối xử lý 5000 của khối BICM đối với biên dạng cơ sở và biên dạng cầm tay có thể bao gồm bộ mã hóa dữ liệu FEC 5010, bộ đan xen bit 5020, bộ ánh xạ chòm điểm 5030, khối mã hóa SSD (Signal Space Diversity – phân tập không gian tín hiệu) 5040 và bộ đan xen thời gian 5050.

Bộ mã hóa dữ liệu FEC 5010 có thể thực hiện việc mã hóa FEC trên BBF đầu vào để tạo ra thủ tục FECBLOCK sử dụng mã hóa ngoài (BCH), và mã hóa trong

(LDPC). Mã hóa ngoài (BCH) là phương pháp mã hóa tùy chọn. Chi tiết về các thao tác của bộ mã hóa dữ liệu FEC 5010 sẽ được mô tả sau đây.

Bộ đan xen bit 5020 có thể đan xen các đầu ra của bộ mã hóa dữ liệu FEC 5010 để đạt được chất lượng được tối ưu hóa kết hợp với các mã LDPC và sơ đồ điều biến trong khi cung cấp cấu trúc có thể thực hiện được một cách hiệu quả. Chi tiết về các thao tác của bộ đan xen bit 5020 sẽ được mô tả sau đây.

Bộ ánh xạ chòm điểm 5030 có thể điều biến mỗi từ-ô (cell word) từ bộ đan xen bit 5020 trong các biên dạng cơ sở và biên dạng cầm tay, hoặc từ-ô từ bộ giải đa hợp từ-ô 5010-1 trong biên dạng nâng cao sử dụng cả QPSK, QAM-16, QAM không đều (NUQ-64, NUQ-256, NUQ-1024) hoặc chòm điểm không đều (NUC-16, NUC-64, NUC-256, NUC-1024) để đưa ra điểm chòm điểm được thông thường hóa năng lượng. Việc ánh xạ chòm điểm này được áp dụng chỉ đối với các DP. Quan sát được rằng QAM-16 và các NUQ có hình dạng vuông, trong khi các NUC có hình dạng bất kỳ. Khi mỗi chòm điểm được quay bởi các góc 90 độ bất kỳ, chòm điểm được quay chồng lặp một cách chính xác lên chòm điểm ban đầu của nó. Đặc tính đối xứng “cảm quay” này khiến cho các dung lượng và các công suất trung bình của các thành phần thật và tưởng tượng bằng với nhau. Cả các NUQ và các NUC được xác định riêng đối với mỗi tỷ lệ mã và phân cụ thể được sử dụng được truyền tín hiệu bởi thông số DP\_MOD được điền trong dữ liệu PLS2.

Khối mã hóa SSD 5040 có thể tiền mã hóa các ô trong hai (2D), ba (3D), và bốn (4D) chiều để cải thiện độ mạnh thu nhận dưới các điều kiện tắt dần khó khăn.

Bộ đan xen thời gian 5050 có thể thao tác ở mức DP. Các thông số của sự đan xen thời gian (TI) có thể được thiết đặt riêng cho mỗi DP. Chi tiết về các thao tác của bộ đan xen thời gian 5050 sẽ được mô tả sau đây.

Khối xử lý 5000-1 của khối BICM đối với biên dạng nâng cao có thể bao gồm bộ mã hóa dữ liệu FEC, bộ đan xen bit, bộ ánh xạ chòm điểm, và bộ đan xen thời gian. Tuy nhiên, khối xử lý 5000-1 được phân biệt với khối xử lý 5000 còn bao gồm bộ giải đa hợp từ-ô 5010-1 và khối mã hóa MIMO 5020-1.

Ngoài ra, các thao tác của bộ mã hóa dữ liệu FEC, bộ đan xen bit, bộ ánh xạ chòm điểm, và bộ đan xen thời gian trong khối xử lý 5000-1 tương ứng với các thao tác của bộ mã hóa dữ liệu FEC 5010, bộ đan xen bit 5020, bộ ánh xạ chòm điểm

5030, và bộ đan xen thời gian 5050 đã được mô tả và do đó phần mô tả của nó được bỏ qua.

Bộ giải đa hợp từ-ô 5010-1 được sử dụng cho DP của biên dạng nâng cao để chia dòng từ-ô đơn thành các dòng từ-ô đôi cho quy trình xử lý MIMO. Chi tiết về các thao tác của bộ giải đa hợp từ-ô 5010-1 sẽ được mô tả sau đây.

Khối mã hóa MIMO 5020-1 có thể xử lý đầu ra của bộ giải đa hợp từ-ô 5010-1 sử dụng sơ đồ mã hóa MIMO. Sơ đồ mã hóa MIMO được tối ưu để phát rộng truyền tín hiệu. Kỹ thuật MIMO là cách hứa hẹn để đạt được sự tăng dung lượng nhưng nó phụ thuộc vào các đặc tính kênh. Đặc biệt đối với phát rộng, thành phần LOS mạnh của kênh hoặc sự khác biệt trong năng lượng tín hiệu thu được giữa hai anten gây ra bởi các đặc tính lan truyền tín hiệu khác nhau khiến nó khó đạt được dung lượng có được từ MIMO. Sơ đồ mã hóa MIMO được đề xuất vượt qua vấn đề này nhờ sử dụng tiền mã hóa trên cơ sở quay và ngẫu nhiên hóa pha của một trong số các tín hiệu đầu ra MIMO.

Mã hóa MIMO là nhằm cho hệ thống 2x2 MIMO cần ít nhất hai anten ở cả bộ truyền và bộ thu. Hai chế độ mã hóa MIMO được xác định trong đề xuất này; đa hợp không gian tỷ lệ đủ (full-rate spatial multiplexing, viết tắt là FR-SM) và đa hợp không gian phân tập đủ tỷ lệ đủ (full-rate full-diversity spatial multiplexing, viết tắt là FRFD-SM). Mã hóa FR-SM cung cấp sự tăng dung lượng với sự tăng tương đối ít tính phức tạp ở phía bộ thu trong khi mã hóa FRFD-SM cung cấp sự tăng dung lượng và sự phân tập bổ sung với sự tăng nhiều tính phức tạp ở phía bộ thu. Sơ đồ mã hóa MIMO được đề xuất không có hạn chế về cấu hình phân cực anten.

Quy trình xử lý MIMO là cần thiết đối với khung biên dạng nâng cao, có nghĩa là tất cả các DP trong khung biên dạng nâng cao được xử lý bởi bộ mã hóa MIMO. Quy trình xử lý MIMO được áp dụng ở mức DP. Các cặp đầu ra bộ ánh xạ chòm điểm NUQ ( $e_{1,i}$  và  $e_{2,i}$ ) được đưa vào đầu vào của bộ mã hóa MIMO. Đầu ra bộ mã hóa MIMO được ghép đôi ( $g_{1,i}$  và  $g_{2,i}$ ) được truyền bởi cùng sóng mang  $k$  và ký hiệu OFDM  $l$  của anten TX tương ứng của chúng.

Các khối được nêu trên có thể được bỏ qua hoặc được thay thế bởi các khối có các chức năng giống hoặc tương tự nhau.

Fig.6 minh họa khối BICM theo phương án khác của sáng chế.

Khối BICM được minh họa trên Fig.6 tương ứng với một phương án của khối BICM 1010 được mô tả dựa vào Fig.1.

Fig.6 minh họa khối BICM để bảo vệ truyền tín hiệu lớp vật lý (PLS), kênh báo hiệu dự phòng (EAC) và kênh thông tin nhanh (FIC). EAC là một phần của khung mà mang dữ liệu thông tin EAS và FIC là kênh logic trong khung mà mang thông tin ánh xạ giữa dịch vụ và DP cơ sở tương ứng. Chi tiết về EAC và FIC sẽ được mô tả sau đây.

Dựa vào Fig.6, khối BICM để bảo vệ PLS, EAC và FIC có thể bao gồm bộ mã hóa PLS FEC 6000, bộ đan xen bit 6010, bộ ánh xạ chòm điểm 6020 và bộ đan xen thời gian 6030.

Ngoài ra, bộ mã hóa PLS FEC 6000 có thể bao gồm bộ xáo trộn, khối mã hóa BCH/chèn số không (zero), khối mã hóa LDPC và khối chấm thùng chắn lẻ LDPC. Mỗi khối của khối BICM sẽ được mô tả dưới đây.

Bộ mã hóa PLS FEC 6000 có thể mã hóa dữ liệu PLS 1/2 được xáo trộn, phần EAC và FIC.

Bộ xáo trộn có thể xáo trộn dữ liệu PLS1 và dữ liệu PLS2 trước khi mã hóa BCH và mã hóa LDPC được chấm thùng và được rút ngắn.

Khối mã hóa BCH/chèn số không có thể thực hiện mã hóa ngoài trên dữ liệu 1/2 PLS được chấm thùng bằng cách sử dụng mã BCH được rút ngắn để bảo vệ PLS và chèn các bit số không sau khi mã hóa BCH. Chỉ đối với dữ liệu PLS1, các bit đầu ra của việc chèn số không có thể được hoán vị trước khi mã hóa LDPC.

Khối mã hóa LDPC có thể mã hóa đầu ra của khối mã hóa BCH/chèn số không bằng cách sử dụng mã LDPC. Nhằm tạo ra khối được mã hóa hoàn thiện,  $C_{ldpc}$ , các bit chắn lẻ,  $P_{ldpc}$  được mã hóa một cách hệ thống từ mỗi khối thông tin PLZ được chèn số không,  $I_{ldpc}$  và được cộng thêm phía sau nó.

Công thức toán học 1

$$C_{ldpc} = [I_{ldpc} \ P_{ldpc}] = [i_0, i_1, \dots, i_{K_{ldpc}-1}, p_0, p_1, \dots, p_{N_{ldpc}-K_{ldpc}-1}]$$

Các thông số mã hóa LDPC cho PLS1 và PLS2 như trong bảng 4 dưới đây.

Bảng 4

| Loại truyền tín hiệu | Ksig  | Kbch | Nbch_parity | Kldpc (=Nbch) | Nldpc | Nldpc_parity | Tỷ lệ mã hóa | Qldpc |
|----------------------|-------|------|-------------|---------------|-------|--------------|--------------|-------|
| PLS1                 | 342   | 1020 | 60          | 1080          | 4320  | 3240         | 1/4          | 36    |
| PLS2                 | <1021 |      |             |               |       |              |              |       |
|                      | >1020 | 2100 |             | 2160          | 7200  | 5040         | 3/10         | 56    |

Khối chấm thùng chẵn lẻ LDPC có thể thực hiện việc chấm thùng trên dữ liệu PLS1 và dữ liệu PLS2.

Khi việc rút ngắn được áp dụng để bảo vệ dữ liệu PLS1, một số bit chẵn lẻ LDPC được chấm thùng sau khi mã hóa LDPC. Ngoài ra, để bảo vệ dữ liệu PLS2, các bit chẵn lẻ LDPC của PLS2 được chấm thùng sau khi mã hóa LDPC. Các bit được chấm thùng này không được truyền.

Bộ đan xen bit 6010 có thể đan xen mỗi dữ liệu PLS1 và dữ liệu PLS2 được rút ngắn và được chấm thùng.

Bộ ánh xạ chòm điểm 6020 có thể ánh xạ dữ liệu PLS1 và dữ liệu PLS2 đã được đan xen bit lên các chòm điểm.

Bộ đan xen thời gian 6030 có thể xếp đan xen dữ liệu PLS1 và dữ liệu PLS2 được ánh xạ.

Các khối được nêu trên có thể được bỏ qua hoặc được thay thế bằng các khối có các chức năng giống hoặc tương tự nhau.

Fig.7 minh họa khối xây dựng khung theo một phương án của sáng chế.

Khối xây dựng khung được minh họa trên Fig.7 tương ứng với một phương án của khối xây dựng khung 1020 được mô tả dựa vào Fig.1.

Theo Fig.7, khối xây dựng khung có thể bao gồm khối bù trễ 7000, bộ ánh xạ

ô 7010 và bộ đan xen tần số 7020. Mỗi khối của khối xây dựng khung sẽ được mô tả.

Khối bù trễ 7000 có thể điều chỉnh sự định thời giữa các ống dữ liệu và dữ liệu PLS tương ứng để đảm bảo rằng chúng đồng thời gian tại phía bộ truyền. Dữ liệu PLS được làm trễ với lượng giống như các ống dữ liệu bằng cách định địa chỉ các độ trễ của các ống dữ liệu gây ra bởi khối định dạng đầu vào và khối BICM. Độ trễ của khối BICM chủ yếu là do bộ đan xen thời gian 5050. Dữ liệu truyền tín hiệu trong dải mang thông tin của nhóm TI tiếp theo sao cho chúng được mang một khung phía trước của các DP cần được truyền tín hiệu. Khối bù trễ làm trễ dữ liệu truyền tín hiệu trong dải tương ứng.

Bộ ánh xạ ô 7010 có thể ánh xạ PLS, EAC, FIC, các DP, các dòng phụ trợ và các ô giả vào các sóng mang hoạt động của các ký hiệu OFDM trong khung. Chức năng cơ bản của bộ ánh xạ ô 7010 là để ánh xạ các ô dữ liệu được tạo ra bởi các TI đối với mỗi trong số các DP, các ô PLS, và các ô EAC/FIC, nếu có, vào trong dãy các ô OFDM hoạt động tương ứng với mỗi ký hiệu OFDM nằm trong khung. Dữ liệu truyền tín hiệu dịch vụ (ví dụ như thông tin đặc tả chương trình (PSI (Program Specific Information)/SI) có thể được tập hợp riêng biệt và được truyền bởi ống dữ liệu. Bộ ánh xạ ô thao tác theo thông tin động được tạo bởi bộ lập lịch và cấu hình của cấu trúc khung. Các chi tiết về khung sẽ được mô tả dưới đây.

Bộ đan xen tần số 7020 có thể xếp đan xen các ô dữ liệu được thu từ bộ ánh xạ ô 7010 để cung cấp sự phân tập tần số. Ngoài ra, bộ đan xen tần số 7020 có thể thao tác trên cặp ký hiệu OFDM bao gồm hai ký hiệu OFDM liên tiếp bằng cách sử dụng thứ tự đan xen khác để đạt được sự đan xen lớn nhất có được trong khung đơn. Chi tiết về các thao tác của bộ đan xen tần số 7020 sẽ được mô tả dưới đây.

Các khối được nêu trên có thể được bỏ qua hoặc được thay thế bởi các khối có các chức năng giống hoặc tương tự nhau.

Fig.8 minh họa khối tạo OFMD theo một phương án của sáng chế.

Khối tạo OFDM điều biến các sóng mang OFDM bởi các ô được tạo ra bởi khối xây dựng khung, chèn các tín hiệu dẫn đường, và tạo ra tín hiệu miền thời gian để truyền. Ngoài ra, khối này chèn tuần tự các khoảng bảo vệ, và áp dụng quy trình xử lý giảm năng lượng vô tuyến đỉnh tới trung bình (PAPR(Peak-to-Average Power

Radio)) để tạo tín hiệu RF cuối cùng.

Dựa vào Fig.8, khối xây dựng khung có thể bao gồm tín hiệu dẫn đường và khối chèn âm (tone) dự trữ 8000, khối mã hóa 2D-eSFN 8010, khối biến đổi Fourier nhanh nghịch đảo (Inverse Fast Fourier Transform, viết tắt là IFFT) 8020, khối giảm PAPR 8030, khối chèn khoảng bảo vệ 8040, khối chèn đoạn đầu 8050, khối chèn hệ thống khác 8060 và khối DAC 8070. Mỗi khối của khối xây dựng khung sẽ được mô tả dưới đây.

Tín hiệu dẫn đường và khối chèn âm dự trữ 8000 có thể chèn các tín hiệu dẫn đường và âm dự trữ.

Các ô khác nhau nằm trong ký hiệu OFDM được điều biến với thông tin tham chiếu, được biết đến như các tín hiệu dẫn đường, mà có các tín hiệu truyền được biết trước trong bộ thu. Thông tin của các ô tín hiệu dẫn đường được tạo bởi các tín hiệu dẫn đường rời rạc, các tín hiệu dẫn đường liên tục, các tín hiệu dẫn đường biên, các tín hiệu dẫn đường FSS (ký hiệu truyền tín hiệu khung) và các tín hiệu dẫn đường FES (ký hiệu biên khung). Mỗi tín hiệu dẫn đường được truyền ở mức công suất tăng cường cụ thể theo loại tín hiệu dẫn đường và mẫu tín hiệu dẫn đường. Giá trị của thông tin dẫn đường được bắt nguồn từ tần số tham chiếu, mà là một dãy các giá trị, một giá trị dùng cho mỗi sóng mang được truyền trên ký hiệu được cho bất kỳ. Các tín hiệu dẫn đường có thể được sử dụng để đồng bộ hóa khung, đồng bộ hóa tần số, đồng bộ hóa thời gian, đánh giá kênh, và xác định chế độ truyền, và cũng có thể được sử dụng theo tiếng ồn pha.

Thông tin tham chiếu, được lấy từ chuỗi tham chiếu, được truyền trong các ô dẫn đường rời rạc trong mỗi ký hiệu ngoại trừ đoạn đầu, FSS và FES của khung. Các tín hiệu dẫn đường liên tục được chèn vào mỗi ký hiệu của khung. Số lượng và vị trí của các tín hiệu dẫn đường liên tục phụ thuộc vào cả kích thước FFT và mẫu tín hiệu dẫn đường rời rạc. Các sóng mang biên là các tín hiệu dẫn đường biên trong mỗi ký hiệu ngoại trừ ký hiệu đoạn đầu. Chúng được chèn nhằm cho phép nội suy tần số tới biên của phổ. Các tín hiệu dẫn đường FSS được chèn trong (các) FSS và các tín hiệu dẫn đường FES được chèn trong FES. Chúng được chèn nhằm cho phép nội suy tần số tới biên của khung.

Hệ thống theo một phương án của sáng chế hỗ trợ mạng SFN, trong đó sơ đồ

MISO được phân phối được sử dụng tùy ý để hỗ trợ chế độ truyền rất mạnh. 2D-eSFN là sơ đồ MISO được phân phối mà sử dụng nhiều anten TX, mỗi trong số chúng được định vị trong các địa điểm kênh truyền khác nhau trong mạng SFN.

Khối mã hóa 2D-eSFN 8010 có thể xử lý quy trình 2D-eSFN làm biến dạng pha của các tín hiệu được truyền từ nhiều bộ truyền, nhằm tạo sự phân tập cả thời gian và tần số trong cấu hình SFN. Do đó, các lỗi nhóm do sự tắt dần phẳng thấp hoặc sự tắt dần mạnh trong thời gian dài có thể được giảm nhẹ.

Khối IFFT 8020 có thể điều biến đầu ra từ khối mã hóa 2D-eSFN 8010 bằng cách sử dụng sơ đồ điều biến OFDM. Ô bất kỳ trong các ký hiệu dữ liệu mà chưa được định rõ làm tín hiệu dẫn đường (hoặc làm âm dự trữ) mang một trong số các ô dữ liệu từ bộ đan xen tần số. Các ô được ánh xạ tới các sóng mang OFDM.

Khối giảm PAPR 8030 có thể thực hiện giảm PAPR lên tín hiệu đầu vào bằng cách sử dụng các hàm giảm PAPR khác nhau trong miền thời gian.

Khối chèn khoảng bảo vệ 8040 có thể chèn các khoảng bảo vệ và khối chèn đoạn đầu 8050 có thể chèn đoạn đầu vào phía trước tín hiệu. Các chi tiết về cấu trúc của đoạn đầu sẽ được mô tả dưới đây. Khối chèn hệ thống khác 8060 có thể đa hợp các tín hiệu của nhiều hệ thống thu/truyền phát rộng trong miền thời gian sao cho dữ liệu của hai hoặc nhiều hệ thống thu/truyền phát rộng khác nhau cung cấp các dịch vụ phát rộng có thể được truyền đồng thời trong cùng một dải tần tín hiệu RF. Trong trường hợp này, hai hoặc nhiều hệ thống thu/truyền phát rộng khác nhau là các hệ thống cung cấp các dịch vụ phát rộng khác nhau. Các dịch vụ phát rộng khác nhau có thể là dịch vụ phát rộng trên mặt đất, dịch vụ phát rộng di động, v.v... Dữ liệu liên quan đến các dịch vụ phát rộng tương ứng có thể được truyền qua các khung khác nhau.

Khối DAC 8070 có thể chuyển đổi tín hiệu số đầu vào thành tín hiệu tương tự và đưa ra tín hiệu tương tự. Đầu ra tín hiệu từ khối DAC 8070 có thể được truyền qua nhiều anten đầu ra theo các biên dạng lớp vật lý. Anten Tx theo một phương án của sáng chế có thể có tương quan cực theo chiều dọc hoặc chiều ngang.

Các khối được nêu trên có thể được bỏ qua hoặc được thay thế bởi các khối có các chức năng giống hoặc tương tự theo thiết kế.

Fig.9 minh họa cấu trúc của thiết bị để thu các tín hiệu phát rộng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế.

Thiết bị để thu các tín hiệu phát rộng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế có thể tương ứng với thiết bị để truyền các tín hiệu phát rộng cho các dịch vụ phát rộng trong tương lai, được mô tả dựa vào Fig.1.

Thiết bị để thu các tín hiệu phát rộng cho các dịch vụ phát rộng trong tương lai theo một phương án của sáng chế có thể bao gồm môđun đồng bộ hóa và giải điều biến 9000, môđun phân tách khung 9010, môđun giải ánh xạ và giải mã 9020, bộ xử lý đầu ra 9030 và môđun giải mã truyền tín hiệu 9040. Thao tác của mỗi môđun của thiết bị để thu các tín hiệu phát rộng sẽ được mô tả dưới đây.

Môđun đồng bộ hóa và giải điều biến 9000 có thể được biểu diễn thành bộ thu và bộ giải điều biến OFDM.

Môđun phân tách khung 9010 có thể được biểu diễn thành bộ phân tách khung.

Môđun giải ánh xạ và giải mã 9020 có thể được biểu diễn thành bộ chuyển đổi và bộ giải mã.

Môđun đồng bộ hóa và giải điều biến 9000 có thể thu các tín hiệu đầu vào qua các anten m Rx, thực hiện việc phát hiện tín hiệu và đồng bộ theo với hệ thống tương ứng với thiết bị để thu các tín hiệu phát rộng và thực hiện việc giải điều biến tương ứng với quy trình xử lý nghịch đảo của quy trình xử lý được thực hiện bởi thiết bị để truyền các tín hiệu phát rộng.

Môđun phân tách khung 9100 có thể phân tách các khung tín hiệu đầu vào và trích dữ liệu mà qua đó dịch vụ được lựa chọn bởi người dùng được truyền. Nếu thiết bị để truyền các tín hiệu phát rộng thực hiện đan xen, môđun phân tách khung 9100 có thể thực hiện việc giải đan xen tương ứng với quy trình xử lý nghịch đảo của đan xen. Trong trường hợp này, các vị trí của tín hiệu và dữ liệu mà cần để trích có thể thu được bằng cách giải mã dữ liệu được đưa ra từ môđun giải mã truyền tín hiệu 9400 để khôi phục thông tin lập lịch được tạo ra bởi thiết bị để truyền các tín hiệu phát rộng.

Môđun giải ánh xạ và giải mã 9200 có thể chuyển đổi các tín hiệu đầu vào thành dữ liệu miền bit và sau đó giải đan xen các tín hiệu này nếu cần thiết. Môđun

giải ánh xạ và giải mã 9200 có thể thực hiện việc giải ánh xạ đối với ánh xạ được áp dụng cho hiệu suất truyền và hiệu chỉnh lỗi được tạo ra trên kênh truyền qua việc giải mã. Trong trường hợp này, môđun giải ánh xạ và giải mã 9200 có thể thu được các thông số truyền cần thiết để giải ánh xạ và giải mã bằng cách giải mã dữ liệu được đưa ra từ môđun giải mã truyền tín hiệu 9400.

Bộ xử lý đầu ra 9300 có thể thực hiện các quy trình xử lý nghịch đảo của các quy trình xử lý tín hiệu/nén khác nhau mà được áp dụng bởi thiết bị để truyền các tín hiệu phát rộng để cải thiện hiệu suất truyền. Trong trường hợp này, bộ xử lý đầu ra 9300 có thể thu thập thông tin điều khiển cần thiết từ đầu ra dữ liệu từ môđun giải mã truyền tín hiệu 9400. Đầu ra của bộ xử lý đầu ra 8300 tương ứng với tín hiệu được đưa vào thiết bị để truyền các tín hiệu phát rộng và có thể là các MPEG-TS, các dòng IP (v4 hoặc v6) và các dòng chung.

Môđun giải mã truyền tín hiệu 9400 có thể thu được thông tin PLS từ tín hiệu được giải điều biến bởi môđun đồng bộ hóa và giải điều biến 9000. Như mô tả ở trên, môđun phân tách khung 9100, môđun giải điều biến và giải mã 9200 và bộ xử lý đầu ra 9300 có thể thực hiện các chức năng của chúng bằng cách sử dụng đầu ra dữ liệu từ môđun giải mã truyền tín hiệu 9400.

Bộ xử lý đầu ra trên Fig.9 thực hiện các chức năng, các quy trình, và/hoặc các phương pháp được đề xuất trên Fig.50, Fig.51, và Fig.53 được mô tả dưới đây.

Fig.10 minh họa cấu trúc khung theo một phương án của sáng chế.

Fig.10 thể hiện cấu hình ví dụ của các kiểu khung và các FRU trong siêu khung. (a) thể hiện siêu khung theo một phương án của sáng chế, (b) thể hiện đơn vị lặp lại khung (FRU (Frame Repetition Unit)) theo một phương án của sáng chế, (c) thể hiện các khung của các biên dạng PHY khác nhau trong FRU và (d) thể hiện cấu trúc của khung.

Siêu khung có thể được bao gồm bởi tám FRU. FRU là đơn vị đa hợp cơ bản cho TDM của các khung, và được lặp lại tám lần trong siêu khung.

Mỗi khung trong FRU thuộc về một trong các biên dạng PHY, (cơ bản, cầm tay, nâng cao) hoặc FEF. Số lượng khung lớn nhất cho phép trong FRU là bốn và biên dạng PHY đã cho có thể có mặt số lần thời gian bất kì từ không lần đến bốn lần

trong FRU (ví dụ, cơ bản, cầm tay, nâng cao). Các định nghĩa biên dạng PHY có thể được mở rộng bằng cách sử dụng các giá trị dự trữ của PHY\_PROFILE trong đoạn đầu, nếu cần thiết.

Phần FEF được chèn vào điểm cuối của FRU, nếu được bao gồm. Khi FEF được bao gồm trong FRU, số lượng tối thiểu của FEF là 8 trong siêu khung. Các phần FEF không nên nằm kề nhau.

Một khung ngoài ra còn được chia thành một số các ký hiệu OFDM và đoạn đầu. Như được thể hiện trên (d), khung bao gồm đoạn đầu, một hoặc nhiều ký hiệu truyền tín hiệu khung (FSS), các ký hiệu dữ liệu thông thường và ký hiệu biên khung (FES).

Đoạn đầu là ký hiệu đặc biệt mà có khả năng phát hiện nhanh tín hiệu hệ thống Futurecast UTB và cung cấp tập hợp các thông số truyền cơ bản đối với hiệu suất truyền và thu nhận của tín hiệu. Mô tả chi tiết về đoạn đầu sẽ được mô tả sau đây.

Mục đích chính của (các) FSS là để mang dữ liệu PLS. Để đồng bộ hóa và đánh giá kênh nhanh, và bởi vậy giải mã nhanh dữ liệu PLS, FSS có mẫu tín hiệu dẫn đường dày hơn so với ký hiệu dữ liệu thông thường. FES có các tín hiệu dẫn đường giống hệt như các tín hiệu dẫn đường của FSS, mà có khả năng nội suy chỉ tần số nằm trong FES và nội suy tạm thời, mà không có khả năng ngoại suy, đối với các ký hiệu ngay trước FES.

Fig.11 minh họa cấu trúc phân cấp truyền tín hiệu của khung theo một phương án của sáng chế.

Fig.11 minh họa cấu trúc phân cấp truyền tín hiệu, mà được chia thành ba phần chính: dữ liệu truyền tín hiệu đoạn đầu 11000, dữ liệu PLS1 11010 và dữ liệu PLS2 11020. Mục đích của đoạn đầu, mà được mang bởi ký hiệu đoạn đầu trong mỗi khung, là để chỉ báo loại truyền và các thông số truyền cơ bản của khung đó. PLS1 cho phép bộ thu truy cập và giải mã dữ liệu PLS2, mà chứa các thông số để truy cập DP mong muốn. PLS2 được mang bởi mỗi khung và được chia thành hai phần chính: dữ liệu PLS2-STAT và dữ liệu PLS2-DYN. Phần tĩnh và động của dữ liệu PLS2 được theo sau bởi đệm, nếu cần thiết.

Fig.12 minh họa dữ liệu truyền tín hiệu đoạn đầu theo một phương án của

sáng chế.

Dữ liệu truyền tín hiệu đoạn đầu mang 21 bit thông tin mà cần thiết để cho phép bộ thu truy cập dữ liệu PLS và tìm vết các DP nằm trong cấu trúc khung. Các chi tiết về dữ liệu truyền tín hiệu đoạn đầu như sau đây:

PHY\_PROFILE: Trường 3-bit này chỉ báo loại biên dạng PHY của khung hiện thời. Việc ánh xạ các loại biên dạng PHY khác nhau được thể hiện trong bảng 5 dưới đây.

Bảng 5

| Giá trị   | Biên dạng PHY      |
|-----------|--------------------|
| 000       | Biên dạng cơ sở    |
| 001       | Biên dạng cầm tay  |
| 010       | Biên dạng nâng cao |
| 011 ~ 110 | Dự trữ             |
| 111       | FEF                |

FFT\_SIZE: Trường 2 bit này chỉ báo kích thước FFT của khung hiện thời nằm trong nhóm khung, như được thể hiện trong bảng 6 dưới đây.

Bảng 6

| Giá trị | Kích thước FFT |
|---------|----------------|
| 00      | FFT 8K         |
| 01      | FFT 16K        |
| 10      | FFT 32K        |

|    |        |
|----|--------|
| 11 | Dự trữ |
|----|--------|

GI\_FRACTION: Trường 3 bit này chỉ báo giá trị phần khoảng bảo vệ trong siêu khung hiện thời, như được thể hiện trong bảng 7 dưới đây.

Bảng 7

| Giá trị   | GI_FRACTION |
|-----------|-------------|
| 000       | 1/5         |
| 001       | 1/10        |
| 010       | 1/20        |
| 011       | 1/40        |
| 100       | 1/80        |
| 101       | 1/160       |
| 110 ~ 111 | Dự trữ      |

EAC\_FLAG: Trường 1 bit này chỉ báo xem liệu EAC có được bố trí trong khung hiện thời hay không. Nếu trường này được thiết đặt là '1', dịch vụ cảnh báo dự phòng (EAS) được bố trí trong khung hiện thời. Nếu trường này được thiết đặt là '0', EAS không được mang trong khung hiện thời. Trường này có thể được chuyển mạch tĩnh trong siêu khung.

PILOT\_MODE: Trường 1-bit này chỉ báo xem liệu chế độ tín hiệu dẫn đường là chế độ di động hay chế độ cố định đối với khung hiện thời trong nhóm khung hiện thời. Nếu trường này được thiết đặt là '0', chế độ tín hiệu dẫn đường di động được sử dụng. Nếu trường này được thiết đặt là '1', chế độ tín hiệu dẫn đường cố định được sử dụng.

PAPR\_FLAG: Trường 1-bit này chỉ báo xem liệu việc giảm PAPR có được sử

dụng đối với khung hiện thời trong nhóm khung hiện thời hay không. Nếu trường này được thiết đặt giá trị ‘1’, dự trữ âm được sử dụng để giảm PAPR. Nếu trường này được thiết đặt là ‘0’, sự giảm PAPR không được sử dụng.

FRU\_CONFIGURE: Trường 3-bit này chỉ báo các cấu hình loại biên dạng PHY của các đơn vị lặp lại khung (FRU) mà có mặt trong siêu khung hiện thời. Tất cả các loại biên dạng được vận chuyển trong siêu khung hiện thời được xác định trong trường này trong tất cả các đoạn đầu của siêu khung hiện thời. Trường 3-bit có định nghĩa khác nhau cho mỗi biên dạng, như được thể hiện trong bảng 8 dưới đây.

Bảng 8

|                     | PHY_PROFI<br>LE hiện thời<br>= ‘000’ (cơ<br>bản) | PHY_PROFI<br>LE hiện thời<br>= ‘001’ (cầm<br>tay) | PHY_PROFI<br>LE hiện thời<br>= ‘010’<br>(nâng cao) | PHY_PROFI<br>LE hiện thời<br>= ‘111’ (FEF) |
|---------------------|--------------------------------------------------|---------------------------------------------------|----------------------------------------------------|--------------------------------------------|
| FRU_CONFIGURE = 000 | Chỉ biên dạng cơ sở có mặt                       | Chỉ biên dạng cầm tay có mặt                      | Chỉ biên dạng nâng cao có mặt                      | Chỉ FEF có mặt                             |
| FRU_CONFIGURE = 1XX | Biên dạng cầm tay có mặt                         | Biên dạng cơ sở có mặt                            | Biên dạng cơ sở có mặt                             | Biên dạng cơ sở có mặt                     |
| FRU_CONFIGURE = X1X | Biên dạng nâng cao có mặt                        | Biên dạng nâng cao có mặt                         | Biên dạng cầm tay có mặt                           | Biên dạng cầm tay có mặt                   |
| FRU_CONFIGURE = XX1 | FEF có mặt                                       | FEF có mặt                                        | FEF có mặt                                         | Biên dạng nâng cao có mặt                  |

RESERVED (dự trữ): Trường 7-bit này được dự trữ để sử dụng trong tương

lai.

Fig.13 minh họa dữ liệu PLS1 theo một phương án của sáng chế.

Dữ liệu PLS1 cung cấp các thông số truyền cơ bản bao gồm các thông số cần thiết để cho phép thu và giải mã PLS2. Như được nêu trên, dữ liệu PLS1 giữ nguyên không đổi trong toàn bộ khoảng thời gian của một nhóm khung. Định nghĩa chi tiết của các trường truyền tín hiệu của dữ liệu PLS1 là như dưới đây:

**PREAMBLE\_DATA:** Trường 20-bit này là bản sao của tín hiệu truyền dữ liệu đoạn đầu ngoại trừ EAC\_FLAG.

**NUM\_FRAME\_FRU:** Trường 2-bit này chỉ báo số lượng của các khung mỗi FRU.

**PAYLOAD\_TYPE:** Trường 3-bit này chỉ báo định dạng của dữ liệu phụ tải được mang trong nhóm khung. PAYLOAD\_TYPE được truyền tín hiệu như được thể hiện trong bảng 9.

Bảng 9

| Giá trị | Loại phụ tải        |
|---------|---------------------|
| 1XX     | Dòng TS được truyền |
| X1X     | Dòng IP được truyền |
| XX1     | Dòng GS được truyền |

**NUM\_FSS:** Trường 2-bit này chỉ báo số lượng các ký hiệu FSS trong khung hiện thời.

**SYSTEM\_VERSION:** Trường 8-bit này chỉ báo phiên bản của định dạng tín hiệu được truyền. SYSTEM\_VERSION được phân chia thành hai trường 4-bit, là phiên bản chính và phiên bản phụ.

Phiên bản chính: bốn bit MSB của trường SYSTEM\_VERSION chỉ báo thông tin phiên bản chính. Thay đổi trong trường phiên bản chính chỉ báo thay đổi không thể tương thích lại được. Giá trị mặc định bằng '0000'. Đối với phiên bản được mô

tả trong tiêu chuẩn này, giá trị được thiết đặt bằng '0000'.

Phiên bản phụ: bốn bit LSB của trường SYSTEM\_VERSION chỉ báo thông tin phiên bản phụ. Thay đổi trong trường phiên bản phụ là có thể tương thích ngược.

CELL\_ID: Đây là trường 16-bit mà xác định duy nhất ô địa lý trong mạng ATSC. Khu vực bao phủ ô ATSC có thể bao gồm một hoặc nhiều tần số, phụ thuộc vào số lượng các tần số được sử dụng trong mỗi hệ thống Futurecast UTB. Nếu giá trị của CELL\_ID là không biết hoặc không được cụ thể hóa, trường này được thiết đặt bằng '0'.

NETWORK\_ID: Đây là trường 16-bit mà xác định duy nhất mạng ATSC hiện thời.

SYSTEM\_ID: Trường 16-bit này xác định duy nhất hệ thống Futurecast UTB nằm trong mạng ATSC. Hệ thống Futurecast UTB là hệ thống phát rộng mặt đất mà đầu vào của nó là một hoặc nhiều dòng đầu vào (TS, IP, GS) và đầu ra của nó là tín hiệu RF. Hệ thống Futurecast UTB mang một hoặc nhiều biên dạng PHY và FEF, nếu có. Hệ thống Futurecast UTB tương tự có thể mang các dòng đầu vào khác nhau và sử dụng các tần số RF khác nhau trong các khu vực địa lý khác nhau, cho phép chèn dịch vụ cục bộ. Cấu trúc khung và việc lập lịch được điều khiển trong một vị trí và là giống nhau cho mọi quá trình truyền nằm trong hệ thống Futurecast UTB. Một hoặc nhiều hệ thống Futurecast UTB có thể có SYSTEM\_ID giống nhau nghĩa là chúng đều có cấu trúc và cấu hình lớp vật lý giống nhau.

Vòng lặp sau đây bao gồm FRU\_PHY\_PROFILE, FRU\_FRAME\_LENGTH, FRU\_GI\_FRACTION, và RESERVED mà được sử dụng để chỉ báo cấu hình FRU và độ dài của mỗi loại khung. Kích thước vòng lặp được cố định sao cho bốn biên dạng PHY (bao gồm FEF) được phát tín hiệu trong FRU. Nếu NUM\_FRAME\_FRU là nhỏ hơn 4, các trường không được sử dụng được điền đầy bởi các số không (zero).

FRU\_PHY\_PROFILE: Trường 3-bit này chỉ báo loại biên dạng PHY của khung thứ (i+1) (i là chỉ số vòng lặp) của FRU liên quan. Trường này sử dụng định dạng truyền tín hiệu giống như định dạng truyền tín hiệu được thể hiện trong bảng 8.

FRU\_FRAME\_LENGTH: Trường 2-bit này chỉ báo độ dài của khung thứ (i+1)

của FRU liên quan. Sử dụng FRU\_FRAME\_LENGTH cùng với FRU\_GI\_FRACTION, giá trị chính xác của khoảng thời gian khung có thể được thu.

FRU\_GI\_FRACTION: Trường 3-bit này chỉ báo giá trị phần khoảng bảo vệ của khung thứ (i+1) của FRU liên quan. FRU\_GI\_FRACTION được truyền tín hiệu như bảng 7.

RESERVED: Trường 4-bit này được dự trữ để sử dụng trong tương lai.

Các trường dưới đây cung cấp các thông số để giải mã dữ liệu PLS2.

PLS2\_FEC\_TYPE: Trường 2-bit này chỉ báo loại FEC được sử dụng để bảo vệ PLS2. Loại FEC được truyền tín hiệu như bảng 10. Các chi tiết về các mã LDPC sẽ được mô tả sau đây.

Bảng 10

| Nội dung | Loại PEC PLS2                 |
|----------|-------------------------------|
| 00       | Các mã LDPC 4K-1/4 và 7K-3/10 |
| 01 ~ 11  | Dự trữ                        |

PLS2\_MOD: Trường 3-bit này chỉ báo loại điều biến được sử dụng bởi PLS2. Loại điều biến được truyền tín hiệu như bảng 11.

Bảng 11

| Giá trị | PLS2_MODE |
|---------|-----------|
| 000     | BPSK      |
| 001     | QPSK      |
| 010     | QAM-16    |
| 011     | NUQ-64    |

|           |        |
|-----------|--------|
| 100 ~ 111 | Dự trữ |
|-----------|--------|

PLS2\_SIZE\_CELL: Trường 15-bit này chỉ báo  $C_{\text{total\_partial\_block}}$ , kích thước (cụ thể là số lượng các ô QAM) của các khối được mã hóa đầy đủ đối với PLS2 mà được mang trong nhóm khung hiện thời. Giá trị này là không đổi trong toàn bộ quá trình của nhóm khung hiện thời.

PLS2\_STAT\_SIZE\_BIT: Trường 14-bit này chỉ báo kích thước, với đơn vị bit, của PLS2-STAT đối với nhóm khung hiện thời. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

PLS2\_DYN\_SIZE\_BIT: Trường 14-bit này chỉ báo kích thước, với đơn vị bit, của PLS2-DYN đối với nhóm khung hiện thời. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

PLS2\_REP\_FLAG: Cờ hiệu 1-bit này chỉ báo xem liệu chế độ lặp PLS2 có được sử dụng trong nhóm khung hiện thời hay không. Khi trường này được thiết đặt với giá trị '1', chế độ lặp PLS2 được kích hoạt. Khi trường này được thiết đặt với giá trị '0', chế độ lặp PLS2 được vô hiệu hóa.

PLS2\_REP\_SIZE\_CELL: Trường 15-bit này chỉ báo  $C_{\text{total\_partial\_block}}$ , kích thước (cụ thể là số lượng các ô QAM) của các khối được mã hóa một phần đối với PLS2 được mang trong mỗi khung của nhóm khung hiện thời, khi việc lặp PLS2 được sử dụng. Nếu việc lặp không được sử dụng, giá trị của trường này là bằng 0. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

PLS2\_NEXT\_FEC\_TYPE: Trường 2-bit này chỉ báo loại FEC được sử dụng đối với PLS2 mà được mang trong mỗi khung của nhóm khung tiếp theo. Loại FEC được truyền tín hiệu như bảng 10.

PLS2\_NEXT\_MOD: Trường 3-bit này chỉ báo loại điều biến được sử dụng đối với PLS2 mà được mang trong mỗi khung của nhóm khung tiếp theo. Loại điều biến được truyền tín hiệu như bảng 11.

PLS2\_NEXT\_REP\_FLAG: Cờ 1-bit này chỉ báo xem liệu chế độ lặp PLS2 có được sử dụng trong nhóm khung tiếp theo hay không. Khi trường này được thiết đặt giá trị '1', chế độ lặp PLS2 được kích hoạt. Khi trường này được thiết đặt giá trị

‘0’, chế độ lặp PLS2 được ngắt kích hoạt.

PLS2\_NEXT\_REP\_SIZE\_CELL: Trường 15-bit này chỉ báo  $C_{total\_full\_block}$ , kích thước (cụ thể là số lượng các ô QAM) của các khối được mã hóa đầy đủ đối với PLS2 mà được mang trong mỗi sóng mang của nhóm khung tiếp theo, khi việc lặp PLS2 được sử dụng. Nếu việc lặp không được sử dụng trong nhóm khung tiếp theo, giá trị của trường này là bằng 0. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

PLS2\_NEXT\_REP\_STAT\_SIZE\_BIT: Trường 14-bit này chỉ báo kích thước, với đơn vị bit, của PLS2-STAT cho nhóm khung tiếp theo. Giá trị này là không đổi trong nhóm khung hiện thời.

PLS2\_NEXT\_REP\_DYN\_SIZE\_BIT: Trường 14 bit này chỉ báo kích thước, với đơn vị bit, của PLS2-DYN cho nhóm khung tiếp theo. Giá trị này là không đổi trong nhóm khung hiện thời.

PLS2\_AP\_MODE: Trường 2 bit này chỉ báo xem liệu chặn lẻ bổ sung có được cung cấp cho PLS2 trong nhóm khung hiện thời hay không. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời. Bảng 12 dưới đây đưa ra các giá trị của trường này. Khi trường này được thiết đặt bằng ‘00’, tính chặn lẻ bổ sung không được sử dụng đối với PLS2 trong nhóm khung hiện thời.

Bảng 12

| Giá trị | Chế độ PLS2-AP         |
|---------|------------------------|
| 00      | AP không được cung cấp |
| 01      | Chế độ AP1             |
| 10 ~ 11 | Dự trữ                 |

PLS2\_AP\_SIZE\_CELL: Trường 15 bit này chỉ báo kích thước (cụ thể là số lượng các ô QAM) của các bit chặn lẻ bổ sung của PLS2. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

PLS2\_NEXT\_AP\_MODE: Trường 2 bit này chỉ báo xem liệu chẵn lẻ bổ sung có được cung cấp để truyền tín hiệu PLS2 trong mỗi khung của nhóm khung tiếp theo hay không. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời. Bảng 12 định nghĩa các giá trị của trường này.

PLS2\_NEXT\_AP\_SIZE\_CELL: Trường 15 bit này chỉ báo kích thước (cụ thể là số lượng của các ô QAM) của các bit chẵn lẻ bổ sung của PLS2 trong mỗi khung của nhóm khung tiếp theo. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

RESERVED: Trường 32 bit này được dự trữ để sử dụng trong tương lai.

CRC\_32: Mã phát hiện lỗi 32 bit, mà được áp dụng cho toàn bộ việc truyền tín hiệu PLS1.

Fig.14 minh họa dữ liệu PLS2 theo một phương án của sáng chế.

Fig.14 minh họa dữ liệu PLS2-STAT của dữ liệu PLS2. Dữ liệu PLS2-STAT là giống nhau trong một nhóm khung, trong đó dữ liệu PLS2-DYN cung cấp thông tin là riêng cho khung hiện thời.

Chi tiết về các trường của dữ liệu PLS2-STAT là như sau đây:

FIC\_FLAG: Trường 1 bit này chỉ báo xem liệu FIC có được sử dụng trong nhóm khung hiện thời hay không. Nếu trường này được thiết đặt là '1', FIC được bố trí trong khung hiện thời. Nếu trường này được thiết đặt là '0', FIC không được mang trong khung hiện thời. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

AUX\_FLAG: Trường 1 bit này chỉ báo xem liệu (các) dòng phụ trợ có được sử dụng trong nhóm khung hiện thời hay không. Nếu trường này được thiết đặt là '1', dòng phụ trợ được bố trí trong khung hiện thời. Nếu trường này được thiết đặt là '0', dòng phụ trợ không được mang trong khung hiện thời. Giá trị này là không đổi trong suốt khoảng thời gian của nhóm khung hiện thời.

NUM\_DP: Trường 6 bit này chỉ báo số lượng các DP được mang nằm trong khung hiện thời. Giá trị của trường này nằm trong khoảng từ 1 tới 64, và số lượng các DP bằng NUM\_DP+1.

DP\_ID: Trường 6 bit này xác định duy nhất DP nằm trong biên dạng PHY.

DP\_TYPE: Trường 3 bit này chỉ báo loại của DP. Nó được truyền tín hiệu như bảng 13 sau đây.

Bảng 13

| Giá trị   | Loại DP   |
|-----------|-----------|
| 000       | DP loại 1 |
| 001       | DP loại 2 |
| 010 ~ 111 | Dự trữ    |

DP\_GROUP\_ID: Trường 8 bit này nhận dạng nhóm DP mà với nó DP hiện thời được kết hợp. Nó có thể được sử dụng bởi bộ thu để truy cập các DP của các thành phần dịch vụ được kết hợp với dịch vụ cụ thể, mà có cùng DP\_GROUP\_ID.

BASE\_DP\_ID: Trường 6 bit này chỉ báo DP mang dữ liệu truyền tín hiệu dịch vụ (ví dụ như PSI/SI) được sử dụng trong lớp quản lý. DP được chỉ báo bởi BASE\_DP\_ID có thể là DP thông thường mang dữ liệu truyền tín hiệu dịch vụ cùng với dữ liệu dịch vụ hoặc DP dành riêng chỉ mang dữ liệu truyền tín hiệu dịch vụ.

DP\_FEC\_TYPE: Trường 2 bit này chỉ báo loại FEC được sử dụng bởi DP được kết hợp. Loại FEC được truyền tín hiệu như bảng 14 sau đây.

Bảng 14

| Giá trị | FEC_TYPE |
|---------|----------|
| 00      | LDPC 16K |
| 01      | LDPC 64K |
| 10 ~ 11 | Dự trữ   |

DP\_COD: Trường 4 bit này chỉ báo tỷ lệ mã được sử dụng bởi DP kết hợp.

Tỷ lệ mã được truyền tín hiệu như bảng 15 sau đây.

Bảng 15

| Giá trị     | Tỷ lệ mã |
|-------------|----------|
| 0000        | 5/15     |
| 0001        | 6/15     |
| 0010        | 7/15     |
| 0011        | 8/15     |
| 0100        | 9/15     |
| 0101        | 10/15    |
| 0110        | 11/15    |
| 0111        | 12/15    |
| 1000        | 13/15    |
| 1001 ~ 1111 | Dự trữ   |

DP\_MOD: Trường 4 bit này chỉ báo sự điều biến được sử dụng bởi DP kết hợp. Sự điều biến được truyền tín hiệu như bảng 16 sau đây.

Bảng 16

| Giá trị | Tỷ lệ mã |
|---------|----------|
| 0000    | QPSK     |
| 0001    | QAM-16   |

|             |          |
|-------------|----------|
| 0010        | NUQ-64   |
| 0011        | NUQ-256  |
| 0100        | NUQ-1024 |
| 0101        | NUC-16   |
| 0110        | NUC-64   |
| 0111        | NUC-256  |
| 1000        | NUC-1024 |
| 1001 ~ 1111 | Dự trữ   |

DP\_SSD\_FLAG: Trường 1 bit này chỉ báo xem liệu chế độ SSD có được sử dụng trong DP kết hợp hay không. Nếu trường này được thiết đặt giá trị '1', SSD được sử dụng. Nếu trường này được thiết đặt giá trị '0', SSD không được sử dụng.

Trường sau đây chỉ có mặt nếu PHY\_PROFILE bằng '010', mà là chỉ báo biên dạng nâng cao:

DP\_MIMO: Trường 3 bit này chỉ báo loại quy trình xử lý mã hóa MIMO nào được áp dụng cho DP kết hợp. Loại quy trình xử lý mã hóa MIMO được truyền tín hiệu như bảng 17.

Bảng 17

| Giá trị   | Mã hóa MIMO |
|-----------|-------------|
| 000       | FR-SM       |
| 001       | FRFD-SM     |
| 010 ~ 111 | Dự trữ      |

DP\_TI\_TYPE: Trường 1 bit này chỉ báo loại đan xen thời gian. Giá trị ‘0’ chỉ báo rằng một nhóm TI tương ứng với một khung và chứa một hoặc nhiều khối TI. Giá trị ‘1’ chỉ báo rằng một nhóm TI được mang trong nhiều hơn một khung và chứa chỉ một khối TI.

DP\_TI\_LENGTH: Việc sử dụng trường 2 bit này (các giá trị cho phép chỉ là 1, 2, 4, 8) được xác định bởi tập hợp các giá trị nằm trong trường DP\_TI\_TYPE như sau đây:

Nếu DP\_TI\_TYPE được thiết đặt với giá trị ‘1’, trường này chỉ báo PI, số lượng của các khung mà nhóm TI được ánh xạ vào đó, và có một khối TI cho mỗi nhóm TI (NTI=1). Các giá trị PI cho phép với trường 2 bit được xác định trong bảng 18 sau đây.

Nếu DP\_TI\_TYPE được thiết đặt với giá trị ‘0’, trường này chỉ báo số lượng các khối TI NTI mỗi nhóm TI, và có một nhóm TI cho mỗi khung (PI=1). Các giá trị PI cho phép với trường 2 bit được xác định trong bảng 18 sau đây.

Bảng 18

| Trường 2 bit | PI | NTI |
|--------------|----|-----|
| 00           | 1  | 1   |
| 01           | 2  | 2   |
| 10           | 4  | 3   |
| 11           | 8  | 4   |

DP\_FRAME\_INTERVAL: Trường 2 bit này chỉ báo khoảng khung ( $I_{JUMP}$ ) nằm trong nhóm khung đối với DP kết hợp và các giá trị cho phép là 1, 2, 4, 8 (trường 2 bit tương ứng là ‘00’, ‘01’, ‘10’, hoặc ‘11’, tương ứng). Đối với các DP mà không có mặt trong mỗi khung của nhóm khung, giá trị của trường này bằng với khoảng giữa các khung kế tiếp. Ví dụ, nếu DP có mặt trên các khung 1, 5, 9, 13, v.v., trường này được thiết đặt bằng ‘4’. Đối với các DP mà có mặt trên mỗi khung, trường này

được thiết đặt bằng '1'.

DP\_TI\_BYPASS: Trường 1 bit này xác định sự khả dụng của bộ đan xen thời gian 5050. Nếu đan xen thời gian không được sử dụng đối với DP, nó được thiết đặt bằng '1'. Trong khi nếu đan xen thời gian được sử dụng nó được thiết đặt bằng '0'.

DP\_FIRST\_FRAME\_IDX: Trường 5 bit này chỉ báo chỉ số của khung thứ nhất của siêu khung trong đó DP hiện thời xảy ra. Giá trị của DP\_FIRST\_FRAME\_IDX nằm trong khoảng từ 0 đến 31.

DP\_NUM\_BLOCK\_MAX: Trường 10 bit này chỉ báo giá trị lớn nhất của DP\_NUM\_BLOCKS đối với DP này. Giá trị của trường này có cùng khoảng như DP\_NUM\_BLOCKS.

DP\_PAYLOAD\_TYPE: Trường 2 bit này chỉ báo loại dữ liệu phụ tải được mang bởi DP đã cho. DP\_PAYLOAD\_TYPE được truyền tín hiệu theo bảng 19 sau đây.

Bảng 19

| Giá trị | Loại phụ tải |
|---------|--------------|
| 00      | TS.          |
| 01      | IP           |
| 10      | GS           |
| 11      | Dự trữ       |

DP\_INBAND\_MODE: Trường 2 bit này chỉ báo xem liệu DP hiện thời có thông tin truyền tín hiệu trong dải hay không. Loại truyền tín hiệu trong dải được truyền tín hiệu theo bảng 20 sau đây.

Bảng 20

| Giá trị | Chế độ trong dải                          |
|---------|-------------------------------------------|
| 00      | Truyền tín hiệu trong dải không được mang |
| 01      | Chỉ mang INBAND-PLS                       |
| 10      | Chỉ mang INBAND-ISSY                      |
| 11      | INBAND-PLS và INBAND-ISSY được mang       |

DP\_PROTOCOL\_TYPE: Trường 2 bit này chỉ báo loại giao thức của phụ tải được mang bởi DP đã cho. Nó được truyền tín hiệu theo bảng 21 sau đây khi loại phụ tải đầu vào được lựa chọn.

Bảng 21

| Giá trị | Nếu<br>DP_PAYLOAD_TYPE<br>là TS | Nếu<br>DP_PAYLOAD_TYPE<br>là IP | Nếu<br>DP_PAYLOAD_TYPE<br>là GS |
|---------|---------------------------------|---------------------------------|---------------------------------|
| 00      | MPEG2-TS                        | IPv4                            | (Chú thích)                     |
| 01      | Dự trữ                          | IPv6                            | Dự trữ                          |
| 10      | Dự trữ                          | Dự trữ                          | Dự trữ                          |
| 11      | Dự trữ                          | Dự trữ                          | Dự trữ                          |

DP\_CRC\_MODE: Trường 2 bit này chỉ báo xem liệu mã hóa CRC có được sử dụng trong khối định dạng đầu vào hay không. Chế độ CRC được truyền tín hiệu theo bảng 22 sau đây.

Bảng 22

| Giá trị | Chế độ CRC         |
|---------|--------------------|
| 00      | Không được sử dụng |
| 01      | CRC-8              |
| 10      | CRC-16             |
| 11      | CRC-32             |

DNP\_MODE: Trường 2 bit này chỉ báo chế độ xóa gói trống được sử dụng bởi DP kết hợp khi DP\_PAYLOAD\_TYPE được thiết đặt là TS ('00'). DNP\_MODE được truyền ký hiệu theo bảng 23 sau đây. Nếu DP\_PAYLOAD\_TYPE không phải là TS ('00'), DNP\_MODE được thiết đặt với giá trị '00'.

Bảng 23

| Giá trị | Chế độ phát hiện gói trống |
|---------|----------------------------|
| 00      | Không được sử dụng         |
| 01      | DNP-NORMAL                 |
| 10      | DNP-OFFSET                 |
| 11      | Dự trữ                     |

ISSY\_MODE: Trường 2 bit này chỉ báo chế độ ISSY được sử dụng bởi DP kết hợp khi DP\_PAYLOAD\_TYPE được thiết đặt là TS ('00'). ISSY\_MODE được truyền tín hiệu theo bảng 24 sau đây. Nếu DP\_PAYLOAD\_TYPE không phải là TS ('00'), ISSY\_MODE được thiết đặt với giá trị '00'.

Bảng 24

| Giá trị | Chế độ ISSY        |
|---------|--------------------|
| 00      | Không được sử dụng |
| 01      | ISSY-UP            |
| 10      | ISSY-BBF           |
| 11      | Dự trữ             |

HC\_MODE\_TS: Trường 2 bit này chỉ báo chế độ nén phần đầu TS được sử dụng bởi DP kết hợp khi DP\_PAYLOAD\_TYPE được thiết đặt là TS ('00'). HC\_MODE\_TS được truyền tín hiệu theo bảng 25 sau đây.

Bảng 25

| Giá trị | Chế độ nén phần đầu |
|---------|---------------------|
| 00      | HC_MODE_TS 1        |
| 01      | HC_MODE_TS 2        |
| 10      | HC_MODE_TS 3        |
| 11      | HC_MODE_TS 4        |

HC\_MODE\_IP: Trường 2 bit này chỉ báo chế độ nén phần đầu IP khi DP\_PAYLOAD\_TYPE được thiết đặt là IP ('01'). HC\_MODE\_IP được truyền tín hiệu theo bảng 26 sau đây.

Bảng 26

| Giá trị | Chế độ nén phần đầu |
|---------|---------------------|
| 00      | Không nén           |

|         |              |
|---------|--------------|
| 01      | HC_MODE_IP 1 |
| 10 ~ 11 | Dự trữ       |

PID : Trường 13 bit này chỉ báo số lượng PID của nén phân đầu TS khi DP\_PAYLOAD\_TYPE được thiết đặt là TS ('00') và HC\_MODE\_TS được thiết đặt là '01' hoặc '10'.

RESERVED: Trường 8 bit này được dự trữ để sử dụng trong tương lai.

Trường sau đây chỉ có mặt nếu FIC\_FLAG bằng '1':

FIC\_VERSION: Trường 8 bit này chỉ báo số lượng phiên bản của FIC.

FIC\_LENGTH\_BYTE: Trường 13 bit này chỉ báo độ dài, với đơn vị byte, của FIC.

RESERVED: Trường 8 bit này được dự trữ để sử dụng trong tương lai.

Trường sau đây chỉ có mặt nếu AUX\_FLAG bằng '1':

NUM\_AUX: Trường 4 bit này chỉ báo số lượng các dòng phụ trợ. Zero (0) có nghĩa không có dòng phụ trợ nào được sử dụng.

AUX\_CONFIG\_RFU: Trường 8 bit này được dự trữ để sử dụng trong tương lai.

AUX\_STREAM\_TYPE: Trường 4 bit này được dự trữ để sử dụng trong tương lai để chỉ báo loại dòng phụ trợ hiện thời.

AUX\_PRIVATE\_CONFIG: Trường 28 bit này được dự trữ để sử dụng trong tương lai để truyền tín hiệu các dòng phụ trợ.

Fig.15 minh họa dữ liệu PLS2 theo một phương án khác của sáng chế.

Fig.15 minh họa dữ liệu PLS2-DYN của dữ liệu PLS2. Các giá trị của dữ liệu PLS2-DYN có thể thay đổi trong suốt khoảng thời gian của một nhóm khung, trong đó kích thước các trường giữ nguyên không đổi.

Chi tiết về các trường của dữ liệu PLS2-DYN là như sau đây:

FRAME\_INDEX: Trường 5 bit này chỉ báo chỉ số khung của khung hiện thời nằm trong siêu khung. Chỉ số khung của khung thứ nhất của siêu khung được thiết đặt bằng '0'.

PLS\_CHANGE\_COUNTER: Trường 4 bit này chỉ báo số lượng các siêu khung trên trong đó cấu hình sẽ thay đổi. Siêu khung tiếp theo sẽ thay đổi về cấu hình được chỉ báo bởi giá trị được truyền tín hiệu nằm trong trường này. Nếu trường này được thiết đặt với giá trị '0000', có nghĩa là không có thay đổi được lập lịch nào được thấy trước: ví dụ, giá trị '1' chỉ báo rằng có thay đổi trong siêu khung tiếp theo.

FIC\_CHANGE\_COUNTER: Trường 4 bit này chỉ báo số lượng các siêu khung trên ở đó cấu hình (nghĩa là, các nội dung của FIC) sẽ thay đổi. Siêu khung tiếp theo với các thay đổi về cấu hình được chỉ báo bởi giá trị được truyền tín hiệu nằm trong trường này. Nếu trường này được thiết đặt với giá trị '0000', có nghĩa là không có thay đổi được lập lịch nào được thấy trước: ví dụ, giá trị '0001' chỉ báo rằng có thay đổi trong siêu khung tiếp theo.

RESERVED: Trường 16 bit này được dự trữ để sử dụng trong tương lai.

Các trường sau đây có mặt trong chu kì trên NUM\_DP, mà mô tả các thông số được kết hợp với DP được mang trong khung hiện thời.

DP\_ID: Trường 6 bit này chỉ báo duy nhất DP nằm trong biên dạng PHY.

DP\_START: Trường 15-bit (hoặc 13-bit) này chỉ báo vị trí bắt đầu của DP thứ nhất trong số các DP sử dụng sơ đồ địa chỉ DPU. Trường DP\_START có kích thước khác theo biên dạng PHY và kích thước FFT như được thể hiện trong bảng 27 sau đây.

Bảng 27

| Biên dạng PHY | Kích thước trường DP_START |        |
|---------------|----------------------------|--------|
|               | 64K                        | 16K    |
| Cơ bản        | 13 bit                     | 15 bit |

|          |        |        |
|----------|--------|--------|
| Cầm tay  | -      | 13 bit |
| Nâng cao | 13 bit | 15 bit |

DP\_NUM\_BLOCK: Trường 10 bit này chỉ báo số lượng các khối FEC trong nhóm TI hiện thời đối với DP hiện thời. Giá trị của DP\_NUM\_BLOCK nằm trong khoảng từ 0 đến 1023.

RESERVED: Trường 8 bit này được dự trữ để sử dụng trong tương lai.

Các trường sau chỉ báo các thông số FIC liên quan đến EAC.

EAC\_FLAG: Trường 1 bit này chỉ báo sự tồn tại của EAC trong khung hiện thời. Bit này có giá trị giống như EAC\_FLAG trong đoạn đầu.

EAS\_WAKE\_UP\_VERSION\_NUM: Trường 8 bit này chỉ báo số lượng phiên bản của chỉ báo đánh thức.

Nếu trường EAC\_FLAG là bằng '1', 12 bit sau đây được phân phối cho trường EAC\_LENGTH\_BYTE. Nếu trường EAC\_FLAG là bằng '0', 12 bit sau đây được phân phối cho EAC\_COUNTER.

EAC\_LENGTH\_BYTE: Trường 12 bit này chỉ báo độ dài, với đơn vị byte, của EAC.

EAC\_COUNTER: Trường 12 bit này chỉ báo số lượng các khung trước khung mà ở đó EAC đến nơi.

Trường sau đây chỉ có mặt nếu trường AUX\_FLAG là bằng '1':

AUX\_PRIVATE\_DYN: Trường 48 bit này được dự trữ để sử dụng trong tương lai cho các dòng phụ trợ truyền tín hiệu. Ý nghĩa của trường này phụ thuộc vào giá trị của AUX\_STREAM\_TYPE trong PLS2-STAT có thể cấu hình được.

CRC\_32: Mã phát hiện lỗi 32 bit, mà được áp dụng cho toàn bộ PLS2.

Fig.16 minh họa cấu trúc logic của khung theo một phương án của sáng chế.

Như được đề cập ở trên, PLS, EAC, FIC, các DP, các dòng phụ trợ và các ô giả được ánh xạ vào trong các sóng mang hoạt động của các ký hiệu OFDM trong

khung. PLS1 và PLS2 đầu tiên được ánh xạ vào một hoặc nhiều FSS. Sau đó, các ô EAC, nếu có, được ánh xạ ngay sau trường PLS, được theo sau bởi các ô FIC, nếu có. Các DP được ánh xạ tiếp sau PLS hoặc EAC, FIC, nếu có. Các DP loại 1 theo sau đầu tiên, và sau đó là các DP loại 2. Các chi tiết về loại của DP sẽ được mô tả sau đây. Trong một số trường hợp, các DP có thể mang một số dữ liệu đặc biệt đối với EAS hoặc dữ liệu truyền tín hiệu dịch vụ. Dòng hoặc các dòng phụ trợ, nếu có, theo sau các DP, DP này lại được theo sau bởi các ô giả. Ánh xạ tất cả chúng theo thứ tự được đề cập ở trên, nghĩa là PLS, EAC, FIC, các DP, các dòng phụ trợ và các ô dữ liệu giả điền đầy chính xác dung lượng ô trong khung.

Fig.17 minh họa ánh xạ PLS theo một phương án của sáng chế.

Các ô PLS được ánh xạ tới các sóng mang hoạt động của (các) FSS. Phụ thuộc vào số lượng các ô được chiếm giữ bởi PLS, một hoặc nhiều ký hiệu được thiết kế làm (các) FSS, và số lượng (các) FSS NFSS được truyền tín hiệu bởi NUM\_FSS trong PLS1. FSS là ký hiệu đặc biệt để mang các ô PLS. Do sức chịu đựng và thời gian chờ là các vấn đề mang tính quyết định trong PLS, (các) FSS có mật độ cao hơn của các tín hiệu dẫn đường cho phép đồng bộ hóa nhanh và nội suy chỉ tần số nằm trong FSS.

Các ô PLS được ánh xạ để kích hoạt các sóng mang của (các) NFSS FSS theo cách thức từ trên xuống như được thể hiện trong ví dụ trên Fig.17. Các ô PLS1 được ánh xạ đầu tiên từ ô thứ nhất của FSS thứ nhất theo thứ tự tăng dần chỉ số ô. Các ô PLS2 theo sau ngay ô cuối cùng của PLS1 và ánh xạ tiếp tục cho đến chỉ số ô cuối cùng của FSS thứ nhất. Nếu tổng số lượng của các ô PLS cần thiết vượt quá số lượng các sóng mang hoạt động của một FSS, việc ánh xạ thực hiện với FSS tiếp theo và tiếp tục theo cách giống hệt như FSS thứ nhất.

Sau khi ánh xạ PLS được hoàn tất, các DP được mang tiếp theo. Nếu EAC, FIC hoặc cả hai có mặt trong khung hiện thời, chúng được đặt giữa PLS và các DP “thông thường”.

Fig.18 minh họa ánh xạ EAC theo một phương án của sáng chế.

EAC là kênh dành riêng để mang các tin nhắn EAS và các liên kết tới các DP cho EAS. Hỗ trợ EAS được cung cấp nhưng bản thân EAC có thể có mặt hoặc không có mặt trong mỗi khung. EAC, nếu có, được ánh xạ ngay sau các ô PLS2. EAC

không bị đứng trước bởi bất kỳ FIC, các DP, các dòng phụ trợ hoặc các ô giả nào ngoài các ô PLS. Quy trình ánh xạ các ô EAC là giống hệt như quy trình ánh xạ các ô PLS.

Các ô EAC được ánh xạ từ ô tiếp theo của PLS2 theo thứ tự tăng dần chỉ số ô được thể hiện trong ví dụ trên Fig.18. Dựa vào kích thước tin nhắn EAS, các ô EAC có thể chiếm giữ một số ký hiệu, như được thể hiện trên Fig.18.

Các ô EAC theo ngay sau ô cuối cùng của PLS2, và ánh xạ tiếp tục cho đến chỉ số ô cuối cùng của FSS cuối cùng. Nếu tổng số lượng các ô EAC được yêu cầu vượt quá số lượng các sóng mang hoạt động đang duy trì của FSS cuối cùng, ánh xạ thực hiện tới ký hiệu tiếp theo và tiếp tục theo cách giống hệt như với (các) FSS. Ký hiệu tiếp theo để ánh xạ trong trường hợp này là ký hiệu dữ liệu thông thường, mà có nhiều sóng mang hoạt động hơn FSS.

Sau khi ánh xạ EAC hoàn tất, FIC được mang tiếp theo, nếu có tồn tại. Nếu FIC không được truyền (như được truyền tín hiệu trong trường PLS2), các DP theo ngay sau ô cuối cùng của EAC.

Fig.19 minh họa ánh xạ FIC theo một phương án của sáng chế.

(a) thể hiện ánh xạ ví dụ của ô FIC không có EAC và (b) thể hiện ánh xạ ví dụ của ô FIC có EAC.

FIC là ô dành riêng để mang thông tin cốt lõi để cho phép việc dò sóng dịch vụ và quét kênh nhanh. Thông tin này căn bản bao gồm thông tin liên kết kênh giữa các DP và các dịch vụ cho mỗi nhà quảng bá. Đối với quét nhanh, bộ thu có thể giải mã FIC và thu thập thông tin ví dụ như ID nhà quảng bá, số lượng các dịch vụ, và BASE\_DP\_ID chẳng hạn. Đối với dò sóng dịch vụ nhanh, bên cạnh FIC, DP cơ bản có thể được giải mã bằng cách sử dụng BASE\_DP\_ID. Ngoài thông tin mà nó mang, DP cơ bản được mã hóa và được ánh xạ tới khung theo cách giống hệt như DP thông thường. Do đó, không cần thiết phải có thêm mô tả đối với DP cơ bản. Dữ liệu FIC được tạo ra và được tiêu thụ trong lớp quản lý (Management Layer). Nội dung của dữ liệu FIC là như được mô tả trong phần mô tả lớp quản lý.

Dữ liệu FIC là tùy chọn và việc sử dụng FIC được truyền tín hiệu bởi thông số FIC\_FLAG trong phần tính của PLS2. Nếu FIC được sử dụng, FIC\_FLAG được

thiết đặt là '1' và trường truyền tín hiệu đối với FIC được định nghĩa trong phần tĩnh của PLS2. Được truyền tín hiệu trong trường này là FIC\_VERSION, và FIC\_LENGTH\_BYTE. FIC sử dụng cùng các thông số điều biến, mã hóa và đan xen thời gian như PLS2. FIC chia sẻ cùng các thông số truyền tín hiệu như PLS2\_MOD và PLS2\_FEC. Dữ liệu FIC, nếu có, được ánh xạ ngay sau PLS2 hoặc EAC nếu có. FIC không bị đứng trước bởi bất kì DP, dòng phụ trợ hoặc ô giả nào. Phương pháp ánh xạ các ô FIC là giống hệt như phương pháp ánh xạ của EAC mà lại là giống như phương pháp ánh xạ của PLS.

Không có EAC sau PLS, các ô FIC được ánh xạ từ ô tiếp theo của PLS2 theo thứ tự tăng dần của chỉ số ô như được thể hiện trong ví dụ trong (a). Dựa vào kích thước dữ liệu FIC, các ô FIC có thể được ánh xạ trên vài ký hiệu, như được thể hiện trên (b).

Các ô FIC theo ngay sau ô cuối cùng của PLS2, và ánh xạ tiếp tục cho đến chỉ số ô cuối cùng của FSSB cuối cùng. Nếu tổng số lượng các ô FIC được yêu cầu vượt quá số lượng các sóng mang hoạt động đang duy trì của FSS cuối cùng, việc ánh xạ thực hiện tới ký hiệu tiếp theo và tiếp tục theo cách giống hệt như (các) FSS. Ký hiệu tiếp theo để ánh xạ trong trường hợp này là ký hiệu dữ liệu thông thường mà có nhiều sóng mang hoạt động hơn một FSS.

Nếu các tín hiệu EAS được truyền trong khung hiện thời, EAC đứng trước FIC, và các ô FIC được ánh xạ từ ô tiếp theo của EAC theo thứ tự tăng dần chỉ số ô như được thể hiện trên (b).

Sau khi ánh xạ FIC được hoàn tất, một nhiều DP được ánh xạ, được theo sau bởi các dòng phụ trợ, nếu có, và các ô giả.

Fig.20 minh họa loại của DP theo một phương án của sáng chế.

Fig.20 (a) thể hiện DP loại 1 và Fig.20 (b) thể hiện DP loại 2.

Sau khi các kênh đứng trước, nghĩa là, PLS, EAC và FIC, được ánh xạ, các ô của các DP được ánh xạ. DP được phân loại vào một hoặc hai loại theo phương pháp ánh xạ:

DP loại 1: DP được ánh xạ bởi TDM

## DP loại 2: DP được ánh xạ bởi FDM

Loại của DP được chỉ báo bởi trường DP\_TYPE trong phần tĩnh của PLS2. Fig.20 minh họa thứ tự ánh xạ của các DP loại 1 và các DP loại 2. Các DP loại 1 đầu tiên được ánh xạ theo thứ tự tăng dần chỉ số ô, và sau đó sau khi đến chỉ số ô cuối cùng, chỉ số ký hiệu được tăng thêm 1. Nằm trong ký hiệu tiếp sau, DP tiếp tục được ánh xạ theo thứ tự tăng dần chỉ số ô bắt đầu từ  $p = 0$ . Với số lượng các DP được ánh xạ cùng nhau trong một khung, mỗi DP loại 1 được nhóm trong cùng thời gian, tương tự với việc đa hợp TDM của các DP.

Các DP loại 2 được ánh xạ đầu tiên theo thứ tự tăng dần của chỉ số ký hiệu, và sau đó sau khi tiến đến ký hiệu OFDM cuối cùng của khung, chỉ số ô tăng lên một và chỉ số ký hiệu quay trở lại ký hiệu khả dụng đầu tiên và sau đó tăng từ chỉ số ký hiệu. Sau khi ánh xạ số lượng của các DP trong cùng một khung, mỗi DP loại 2 được nhóm trong tần số cùng với nhau, tương tự như việc đa hợp FDM của các DP.

Các DP loại 1 và các DP loại 2 có thể cùng tồn tại trong khung nếu cần thiết với một hạn chế; các DP loại 1 luôn luôn đứng trước các DP loại 2. Tổng số lượng các ô OFDM mang các DP loại 1 và các DP loại 2 không thể vượt quá tổng số lượng các ô OFDM có sẵn để truyền các DP:

Công thức toán học 2

$$D_{DP1} + D_{DP2} \leq D_{DP}$$

trong đó  $D_{DP1}$  là số lượng của các ô OFDM được chiếm giữ bởi các DP loại 1,  $D_{DP2}$  là số lượng các ô được chiếm giữ bởi các DP loại 2. Do PLS, EAC, FIC đều được ánh xạ với cùng cách như DP loại 1, chúng đều tuân theo “Quy tắc ánh xạ loại 1”. Do đó, về tổng thể, ánh xạ loại 1 luôn luôn đứng trước ánh xạ loại 2.

Fig.21 minh họa ánh xạ DP theo một phương án của sáng chế.

(a) thể hiện địa chỉ của các ô OFDM cho ánh xạ các DP loại 1 và (b) thể hiện địa chỉ của các ô OFDM cho ánh xạ các DP loại 2.

Địa chỉ của các ô OFDM cho ánh xạ các DP loại 1 ( $0, \dots, D_{DP1}-1$ ) được định nghĩa đối với các ô dữ liệu hoạt động của các DP loại 1. Sơ đồ địa định rõ nghĩa thứ

tự trong đó các ô từ các TI đối với mỗi DP loại 1 được phân phối tới các ô dữ liệu hoạt động. Nó cũng được sử dụng để báo hiệu các vị trí của các DP trong phần động của PLS2.

Không có EAC và FIC, địa chỉ 0 là để chỉ ô ngay sau ô cuối cùng mang PLS trong FSS cuối cùng. Nếu EAC được truyền và FIC không ở trong khung tương ứng, địa chỉ 0 là để chỉ ô ngay sau ô cuối cùng mang EAC. Nếu FIC được truyền trong khung tương ứng, địa chỉ 0 là để chỉ ô ngay sau ô cuối cùng mang FIC. Địa chỉ 0 đối với các DP loại 1 có thể được tính toán bằng cách xem xét hai trường hợp khác nhau như được thể hiện trong phần (a). Theo ví dụ trong phần (a), PLS, EAC và FIC được coi là đều được truyền. Mở rộng các trường hợp trong đó EAC và FIC hoặc cả hai đều được bỏ qua là hiển nhiên. Nếu có ô còn lại trong FSS sau khi ánh xạ tất cả các ô tới FIC như được thể hiện trong phía trái của phần (a).

Địa chỉ của các ô OFDM để ánh xạ các DP loại 2 ( $0, \dots, D_{DP2}-1$ ) được xác định cho các ô dữ liệu hoạt động của các DP loại 2. Sơ đồ địa chỉ xác định thứ tự trong đó các ô từ các TI cho mỗi DP loại 2 được phân phối tới các ô dữ liệu hoạt động. Nó cũng được sử dụng để truyền tín hiệu các vị trí của các DP trong phần động của PLS2.

Ba trường hợp với một chút khác biệt như được thể hiện trong phần (b). Đối với trường hợp thứ nhất được thể hiện trong phía trái của phần (b), các ô trong FSS cuối cùng là khả dụng đối với ánh xạ DP loại 2. Đối với trường hợp thứ hai được thể hiện ở giữa, FIC chiếm giữ các ô với ký hiệu thông thường, nhưng số lượng của các ô FIC trên ký hiệu này là không lớn hơn  $C_{FSS}$ . Trường hợp thứ ba, được thể hiện ở bên phải trong phần (b), là giống như trường hợp thứ hai ngoại trừ việc số lượng các ô FIC được ánh xạ trên ký hiệu này vượt quá  $C_{FSS}$ .

Mở rộng trường hợp trong đó (các) DP loại 1 đứng trước các DP loại 2 là hiển nhiên do PLS, EAC và FIC tuân theo “quy tắc ánh xạ loại 1” giống như (các) DP loại 1.

Đơn vị ống dữ liệu (DPU) là đơn vị cơ bản để phân phối các ô dữ liệu tới DP trong khung.

DPU được xác định là các đơn vị truyền tín hiệu để định vị các DP trong khung. Bộ ánh xạ ô 7010 có thể ánh xạ các ô được tạo ra bởi các TI cho mỗi DP. Bộ đơn

xen thời gian 5050 đưa ra loạt các khối TI và mỗi khối TI bao gồm một loạt số lượng các XFECBLOCK mà trong đó các XFEXBLOCK này lại bao gồm tập hợp các ô. Số lượng các ô trong XFECBLOCK,  $N_{cells}$ , phụ thuộc vào kích thước FECBLOCK,  $N_{ldpc}$ , và số lượng các bit được truyền trên mỗi ký hiệu chùm điểm. DPU được xác định là ước số chung lớn nhất của tất cả các giá trị có thể của các ô trong XFECBLOCK,  $N_{cells}$ , được hỗ trợ trong biên dạng PHY đã cho. Độ dài của DPU trong các ô được xác định là  $L_{DPU}$ . Do mỗi biên dạng PHY hỗ trợ các kết hợp khác nhau của kích thước FECBLOCK và số lượng các bit khác nhau với mỗi ký hiệu tương quan,  $L_{DPU}$  được xác định trên cơ sở biên dạng PHY.

Fig.22 minh họa cấu trúc FEC theo một phương án của sáng chế.

Fig.22 minh họa cấu trúc FEC theo một phương án của sáng chế trước khi đan xen bit. Như được đề cập ở trên, bộ mã hóa FEC dữ liệu có thể thực hiện việc mã hóa FEC trên BBF đầu vào để tạo ra quy trình xử lý FECBLOCK sử dụng mã hóa ngoài (BCH), và mã hóa trong (LDPC). Cấu trúc FEC được minh họa tương ứng với FECBLOCK. Ngoài ra, FECBLOCK và cấu trúc FEC có cùng giá trị tương ứng với độ dài từ mã LDPC.

Mã hóa BCH được áp dụng với mỗi BBF (các bit  $K_{bch}$ ), và sau đó mã hóa LDPC được áp dụng cho BCH được mã hóa BBF (các bit  $K_{ldpc}$  = các bit  $N_{bch}$ ) như được minh họa trên Fig.22.

Giá trị của  $N_{ldpc}$  là 64800 bit (FECBLOCK dài) hoặc 16200 bit (FECBLOCK ngắn).

Bảng 28 và bảng 29 dưới đây lần lượt thể hiện các thông số mã hóa FEC cho FECBLOCK dài và FECBLOCK ngắn.

Bảng 28

| Tỷ lệ LDPC | $N_{ldpc}$ | $K_{ldpc}$ | $K_{bch}$ | Dung lượng hiệu đỉnh lỗi BCH | $N_{bch} - K_{bch}$ |
|------------|------------|------------|-----------|------------------------------|---------------------|
| 5/15       |            | 21600      | 21408     |                              |                     |

|       |       |       |       |    |     |
|-------|-------|-------|-------|----|-----|
| 6/15  | 64800 | 25920 | 25728 | 12 | 192 |
| 7/15  |       | 30240 | 30048 |    |     |
| 8/15  |       | 34560 | 34368 |    |     |
| 9/15  |       | 38880 | 38688 |    |     |
| 10/15 |       | 43200 | 43008 |    |     |
| 11/15 |       | 47520 | 47328 |    |     |
| 12/15 |       | 51840 | 51648 |    |     |
| 13/15 |       | 56160 | 55968 |    |     |

Bảng 29

| Tỷ lệ LDPC | $N_{ldpc}$ | $K_{ldpc}$ | $K_{bch}$ | Dung lượng hiệu chỉnh lỗi BCH | $N_{bch} - K_{bch}$ |
|------------|------------|------------|-----------|-------------------------------|---------------------|
| 5/15       | 16200      | 5400       | 5232      | 12                            | 168                 |
| 6/15       |            | 6480       | 6312      |                               |                     |
| 7/15       |            | 7560       | 7392      |                               |                     |
| 8/15       |            | 8640       | 8472      |                               |                     |
| 9/15       |            | 9720       | 9552      |                               |                     |
| 10/15      |            | 10800      | 10632     |                               |                     |
| 11/15      |            | 11880      | 11712     |                               |                     |
| 12/15      |            | 12960      | 12792     |                               |                     |
| 13/15      |            | 14040      | 13872     |                               |                     |

Chi tiết về các thao tác mã hóa BCH và mã hóa LDPC là như dưới đây:

Mã BCH hiệu chỉnh lỗi 12 được sử dụng để mã hóa ngoài của BBF. Đa thức tạo BCH đối với FECBLOCK ngắn và FECBLOCK dài được thu bằng cách nhân tất cả các đa thức với nhau.

Mã LDPC được sử dụng để mã hóa đầu ra của mã hóa BCH đầu ra. Nhằm tạo ra  $B_{ldpc}$  (FECBLOCK) hoàn thiện,  $P_{ldpc}$  (các bit chẵn lẻ) được mã hóa một cách hệ thống từ mỗi  $I_{ldpc}$  (BBF được mã hóa BCH), và cộng thêm với  $I_{ldpc}$ .  $B_{ldpc}$  hoàn thiện (FECBLOCK) được thể hiện trong công thức toán học sau đây.

Công thức toán học 3

$$B_{ldpc} = [ \mathbf{I}_{ldpc} \quad \mathbf{P}_{ldpc} ] = [ i_0, i_1, \dots, i_{K_{ldpc}-1}, p_0, p_1, \dots, p_{N_{ldpc}-K_{ldpc}-1} ]$$

Các thông số cho FECBLOCK dài và FECBLOCK ngắn được nêu trong bảng 28 và 29 ở trên, theo thứ tự.

Quy trình cụ thể để tính toán các bit chẵn lẻ  $N_{ldpc} - K_{ldpc}$  đối với FECBLOCK dài, là như dưới đây:

1) Khởi đầu các bit chẵn lẻ,

Công thức toán học 4

$$p_0 = p_1 = p_2 = \dots = p_{N_{ldpc}-K_{ldpc}-1} = 0$$

2) Tích lũy bit thông tin thứ nhất -  $i_0$ , ở các địa chỉ bit chẵn lẻ được cụ thể hóa trong dòng thứ nhất của các địa chỉ của ma trận kiểm tra chẵn lẻ. Các chi tiết của các địa chỉ của ma trận kiểm tra chẵn lẻ sẽ được mô tả sau đây. Ví dụ, đối với tỷ lệ 13/15:

Công thức toán học 5

$$p_{983} = p_{983} \oplus i_0 \quad p_{2815} = p_{2815} \oplus i_0$$

$$p_{4837} = p_{4837} \oplus i_0 \quad p_{4989} = p_{4989} \oplus i_0$$

$$p_{6138} = p_{6138} \oplus i_0 \quad p_{6458} = p_{6458} \oplus i_0$$

$$p_{6921} = p_{6921} \oplus i_0 \quad p_{6974} = p_{6974} \oplus i_0$$

$$p_{7572} = p_{7572} \oplus i_0 \quad p_{8260} = p_{8260} \oplus i_0$$

$$p_{8496} = p_{8496} \oplus i_0$$

3) Đối với 359 bit thông tin tiếp theo,  $i_s, s=1, 2, \dots, 359$  tích lũy là ở các địa chỉ bit chẵn lẻ bằng cách sử dụng công thức toán học sau đây.

Công thức toán học 6

$$\{x + (s \bmod 360) \times Q_{ldpc}\} \bmod (N_{ldpc} - K_{ldpc})$$

trong đó  $x$  biểu thị địa chỉ của bộ tích lũy bit chẵn lẻ tương ứng với bit  $i_0$  thứ nhất, và  $Q_{ldpc}$  hằng số phụ thuộc tỷ lệ mã được cụ thể hóa trong các địa chỉ của ma trận kiểm tra chẵn lẻ. Tiếp tục với ví dụ,  $Q_{ldpc} = 24$  đối với tỷ lệ 13/15, do đó đối với bit thông tin  $i_1$ , các thao tác dưới đây được thực hiện:

Công thức toán học 7

$$p_{1007} = p_{1007} \oplus i_1 \quad p_{2839} = p_{2839} \oplus i_1$$

$$p_{4861} = p_{4861} \oplus i_1 \quad p_{5013} = p_{5013} \oplus i_1$$

$$p_{6162} = p_{6162} \oplus i_1 \quad p_{6482} = p_{6482} \oplus i_1$$

$$p_{6945} = p_{6945} \oplus i_1 \quad p_{6998} = p_{6998} \oplus i_1$$

$$p_{7596} = p_{7596} \oplus i_1 \quad p_{8284} = p_{8284} \oplus i_1$$

$$p_{8520} = p_{8520} \oplus i_1$$

4) Đối với bit thông tin thứ 361 bit thông tin  $i_{360}$ , các địa chỉ của bộ tích lũy bit chẵn lẻ được đưa ra ở dòng thứ hai của các địa chỉ của ma trận kiểm tra chẵn lẻ. Với cách tương tự các địa chỉ của các bộ tích lũy bit chẵn lẻ của 359 bit thông tin phía sau  $i_s$ ,  $s = 361, 362, \dots, 719$  được thu bằng cách sử dụng công thức toán học 6, trong đó  $x$  biểu thị địa chỉ của bộ tích lũy bit chẵn lẻ tương ứng với bit thông tin  $i_{360}$ , nghĩa là, các mục nhập trong dòng thứ hai của các địa chỉ của ma trận kiểm tra chẵn lẻ.

5) Theo cách tương tự, đối với mỗi bit trong số 360 bit thông tin mới, dòng mới từ các địa chỉ của các ma trận kiểm tra chẵn lẻ được sử dụng để tìm các địa chỉ của các bộ tích lũy bit chẵn lẻ.

Sau khi tất cả các bit thông tin cạn kiệt, các bit chẵn lẻ cuối cùng được thu như dưới đây:

6) Thực hiện tuần tự các thao tác dưới đây bắt đầu với  $i=1$

Công thức toán học 8

$$p_i = p_i \oplus p_{i-1}, \quad i = 1, 2, \dots, N_{ldpc} - K_{ldpc} - 1$$

Trong đó nội dung cuối cùng của  $p_i$ ,  $i=0,1,\dots,N_{ldpc} - K_{ldpc} - 1$  là bằng với bit chẵn lẻ  $p_i$ .

Bảng 30

| Tỷ lệ mã hóa | $Q_{ldpc}$ |
|--------------|------------|
| 5/15         | 120        |
| 6/15         | 108        |
| 7/15         | 96         |
| 8/15         | 84         |
| 9/15         | 72         |
| 10/15        | 60         |
| 11/15        | 48         |
| 12/15        | 36         |
| 13/15        | 24         |

Quy trình mã hóa LDPC này đối với FECBLOCK ngắn là tương tự như quy trình mã hóa LDPC đối với FECBLOCK dài, ngoại trừ việc thay bảng 30 bằng bảng 31, và thay thế các địa chỉ của ma trận kiểm tra chẵn lẻ đối với FECBLOCK dài bằng các địa chỉ của ma trận kiểm tra chẵn lẻ đối với FECBLOCK ngắn.

Bảng 31

| Tỷ lệ mã hóa | Qldpc |
|--------------|-------|
| 5/15         | 30    |
| 6/15         | 27    |
| 7/15         | 24    |
| 8/15         | 21    |
| 9/15         | 18    |
| 10/15        | 15    |
| 11/15        | 12    |
| 12/15        | 9     |
| 13/15        | 6     |

Fig.23 minh họa sự đan xen bit theo một phương án của sáng chế.

Các đầu ra của bộ mã hóa LDPC được đan xen bit, mà bao gồm việc đan xen lẫn lộn cùng với việc đan xen khối tựa như vòng lặp (Quasi-Cyclic Block, viết tắt là QCB) và việc đan xen nhóm trong.

(a) thể hiện việc đan xen khối tựa như vòng lặp (QCB) và (b) thể hiện việc đan xen nhóm trong.

FECBLOCK có thể được đan xen lẫn lộn. Tại đầu ra của việc đan xen lẫn lộn, từ mã LDPC bao gồm 180 khối QC liên tiếp trong FECBLOCK dài và 45 khối QC liên tiếp trong FECBLOCK ngắn. Mỗi khối QC của cả FECBLOCK ngắn và dài đều bao gồm 360 bit. Từ mã LDPC đan xen lẫn lộn được đan xen bằng việc đan xen QCB. Đơn vị đan xen QCB là khối QC. Các khối QC tại đầu ra của việc đan xen lẫn lộn được hoán vị bởi việc đan xen QCB như được minh họa trên Fig.23, trong đó  $N_{\text{cells}} = 64800/\eta_{\text{mod}}$  hoặc  $16200/\eta_{\text{mod}}$  theo độ dài FECBLOCK. Mẫu đan xen

QCB là duy nhất đối với mỗi kết hợp của loại điều biến và tỷ lệ mã hóa LDPC.

Sau quá trình đan xen QCB, quá trình đan xen nhóm trong được thực hiện theo loại điều biến và thứ tự ( $\eta_{\text{mod}}$ ) mà được xác định trong bảng 32 dưới đây. Số lượng các khối QC cho một nhóm trong,  $N_{\text{QCB\_IG}}$ , cũng được xác định.

Bảng 32

| Loại điều biến | $\eta_{\text{mod}}$ | $N_{\text{QCB\_IG}}$ |
|----------------|---------------------|----------------------|
| QAM-16         | 4                   | 2                    |
| NUC-16         | 4                   | 4                    |
| NUQ-64         | 6                   | 3                    |
| NUC-64         | 6                   | 6                    |
| NUQ-256        | 8                   | 4                    |
| NUC-256        | 8                   | 8                    |
| NUQ-1024       | 10                  | 5                    |
| NUC-1024       | 10                  | 10                   |

Quy trình đan xen nhóm trong được thực hiện với các khối QC  $N_{\text{QCB\_IG}}$  của đầu ra đan xen QCB. Việc đan xen nhóm trong có quy trình ghi và đọc các bit của nhóm trong bằng cách sử dụng 360 cột và các dòng  $N_{\text{QCB\_IG}}$ . Trong thao tác viết, các bit từ đầu ra đan xen QCB được viết theo dòng. Thao tác đọc được thực hiện theo cột để đọc ra  $m$  bit từ mỗi dòng, trong đó  $m$  bằng 1 đối với NUC và 2 đối với NUQ.

Fig.24 minh họa việc giải đa hợp ô-từ theo một phương án của sáng chế.

(a) thể hiện việc giải đa hợp ô-từ cho MIMO 8 và 12 bpcu và (b) thể hiện việc giải đa hợp ô-từ cho MIMO 10 bpcu.

Mỗi ô từ ( $c_{0,l}, c_{1,l}, \dots, c_{\eta \bmod -1,l}$ ) của đầu ra đan xen bit được giải đa hợp thành ( $d_{1,0,m}, d_{1,1,m}, \dots, d_{1,\eta \bmod -1,m}$ ) và ( $d_{2,0,m}, d_{2,1,m}, \dots, d_{2,\eta \bmod -1,m}$ ) như được thể hiện trên (a), mà mô tả quy trình giải đa hợp từ-ô đối với một XFECBLOCK.

Đối với trường hợp MIMO 10 bpcu sử dụng các loại NUQ khác nhau cho mã hóa MIMO, bộ đan xen bit đối với NUQ-1024 được tái sử dụng. Mỗi ô từ ( $c_{0,l}, c_{1,l}, \dots, c_{9,l}$ ) của đầu ra bộ đan xen bit được giải đa hợp thành ( $d_{1,0,m}, d_{1,1,m}, \dots, d_{1,3,m}$ ) và ( $d_{2,0,m}, d_{2,1,m}, \dots, d_{2,5,m}$ ), như được thể hiện trên (b).

Fig.25 minh họa việc đan xen thời gian theo một phương án của sáng chế.

(a) đến (c) thể hiện các ví dụ về chế độ TI.

Bộ đan xen thời gian thao tác ở mức DP. Các thông số của việc đan xen thời gian (TI) có thể được thiết đặt khác nhau cho mỗi DP.

Các thông số dưới đây, mà có mặt trong phần dữ liệu PLS2-STAT, tạo cấu hình TI:

DP\_TI\_TYPE (các giá trị được phép: 0 hoặc 1): thể hiện chế độ TI; ‘0’ chỉ báo chế độ với nhiều khối TI (nhiều hơn một khối TI) mỗi nhóm TI. Trong trường hợp này, một nhóm TI được ánh xạ trực tiếp lên một khung (không đan xen liên khung). ‘1’ chỉ báo chế độ với chỉ một khối TI mỗi nhóm TI. Trong trường hợp này, khối TI có thể lan rộng ra nhiều hơn một khung (đan xen liên khung).

DP\_TI\_LENGTH: Nếu DP\_TI\_TYPE = ‘0’, thông số này là số lượng của các khối TI NTI mỗi nhóm TI. Đối với DP\_TI\_TYPE = ‘1’, thông số này là số lượng của các khung PI lan rộng ra từ một nhóm TI.

DP\_NUM\_BLOCK\_MAX (các giá trị cho phép: 0 đến 1023): thể hiện số lượng lớn nhất của các XFECBLOCK trong mỗi nhóm TI.

DP\_FRAME\_INTERVAL (các giá trị cho phép: 1, 2, 4, 8): thể hiện số lượng các khung IJUMP giữa hai khung kế tiếp mang cùng một DP của biên dạng PHY đã cho.

DP\_TI\_BYPASS (các giá trị cho phép: 0 hoặc 1): nếu xen kẽ thời gian không được sử dụng đối với DP, thông số này được thiết đặt là ‘1’. Nó được thiết đặt là ‘0’ nếu xen kẽ thời gian được sử dụng.

Ngoài ra, thông số DP\_NUM\_BLOCK từ dữ liệu PLS2-DYN được sử dụng để thể hiện số lượng các XFECBLOCK được mang bởi một nhóm TI của DP.

Khi xen kẽ thời gian không được sử dụng đối với DP, nhóm TI, thao tác xen kẽ thời gian, và chế độ TI sau đây không được xét đến. Tuy nhiên, khối bù trễ đối với thông tin cấu hình động từ bộ lập lịch vẫn sẽ được yêu cầu. Trong mỗi DP, các XFECBLOCK được thu từ việc mã hóa SSD/MIMO được nhóm thành các nhóm TI. Nghĩa là, mỗi nhóm TI là tập hợp số nguyên số lượng các XFECBLOCK và sẽ bao gồm số lượng động khác nhau của các XFECBLOCK. Số lượng XFECBLOCK trong nhóm TI của chỉ số n được biểu diễn bởi NxBLOCK\_Group(n) và được truyền tín hiệu là DP\_NUM\_BLOCK trong dữ liệu PLS2-DYN. Lưu ý rằng NxBLOCK\_Group(n) có thể thay đổi từ giá trị tối thiểu bằng 0 tới giá trị lớn nhất của NxBLOCK\_Group\_MAX (tương ứng với DP\_NUM\_BLOCK\_MAX) trong đó giá trị lớn nhất là 1023.

Mỗi nhóm TI đều được ánh xạ trực tiếp lên một khung hoặc được lan rộng trên các khung PI. Mỗi nhóm TI cũng được phân chia thành nhiều hơn một khối TI (NTI), trong đó mỗi khối TI tương ứng với một lần sử dụng bộ nhớ bộ đệm xen thời gian. Các khối TI nằm trong nhóm TI có thể bao gồm các số lượng XFECBLOCK khác biệt một chút. Nếu nhóm TI được phân chia thành nhiều khối TI, nó được ánh xạ trực tiếp chỉ lên một khung. Có ba khả năng đối với đan xen thời gian (ngoại trừ khả năng đặc biệt là bỏ qua đan xen thời gian) như được thể hiện trong bảng 33 dưới đây.

Bảng 33

| Chế độ     | Mô tả                                                                                                                                                                                                        |
|------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Lựa chọn 1 | Mỗi nhóm TI chứa một khối TI và được ánh xạ trực tiếp tới một khung như được thể hiện trên (a). Phương án này được truyền tín hiệu trên PLS2-STAT bởi DP_TI_TYPE = '0' và DP_TI_LENGTH = '1' ( $N_{TI}=1$ ). |
| Lựa chọn 2 | Mỗi nhóm TI chứa một khối TI và được ánh xạ tới nhiều hơn một khung. (b) thể hiện ví dụ, ở đó một nhóm TI                                                                                                    |

|            |                                                                                                                                                                                                                                                                                                                                       |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|            | được ánh xạ tới hai khung, nghĩa là, DP_TI_LENGTH = '2' (Pt = 2) và DP_FRAME_INTERVAL (IJUMP = 2). Việc này khiến phân tập thời gian lớn hơn đối với các dịch vụ tỷ lệ dữ liệu thấp. Phương án này được truyền tín hiệu trong PLS2-STAT bởi DP_TI_TYPE = '1'                                                                          |
| Lựa chọn 3 | Mỗi nhóm TI được chia thành nhiều khối TI và được ánh xạ trực tiếp tới một khung như được thể hiện trên (c). Mỗi khối TI có thể sử dụng bộ nhớ TI đầy đủ, để cung cấp tỷ lệ bit lớn nhất đối với DP. Phương án này được truyền tín hiệu trong truyền tín hiệu PLS2-STAT bởi DP_TI_TYPE = '0' và DP_TY_LENGTH = NTI, trong khi PI = 1. |

Trong mỗi DP, bộ nhớ TI lưu trữ các XFECBLOCK đầu vào (các XFECBLOCK đầu ra từ khối mã hóa SSD/MIMO). Coi rằng các XFECBLOCK đầu vào được định nghĩa là

$$(d_{n,s,0,0}, d_{n,s,0,1}, \dots, d_{n,s,0,N_{\text{cells}}-1}, d_{n,s,1,0}, \dots, d_{n,s,1,N_{\text{cells}}-1}, \dots, d_{n,s,N_{\text{BLOCK\_TI}}(n,s)-1,0}, \dots, d_{n,s,N_{\text{BLOCK\_TI}}(n,s)-1,N_{\text{cells}}-1}),$$

trong đó  $d_{n,s,r,q}$  là ô thứ  $q$  của XFECBLOCK thứ  $r$  trong khối TI thứ  $s$  của nhóm TI thứ  $n$  và thể hiện các đầu ra của các mã hóa SSD và MIMO như dưới đây

$$d_{n,s,r,q} = \begin{cases} f_{n,s,r,q}, & \text{đầu ra của mã hóa SSD} \\ g_{n,s,r,q}, & \text{đầu ra của mã hóa MIMO} \end{cases}$$

Ngoài ra, giả định rằng các XFECBLOCK đầu ra từ bộ đan xen thời gian 5050 được định nghĩa bằng

$$(h_{n,s,0}, h_{n,s,1}, \dots, h_{n,s,i}, \dots, h_{n,s,N_{\text{BLOCK\_TI}}(n,s) \times N_{\text{cells}}-1}),$$

trong đó  $h_{n,s,i}$  là ô đầu ra thứ  $i$  (đối với  $i = 0, \dots, N_{\text{BLOCK\_TI}}(n,s) \times N_{\text{cells}} - 1$ ) trong khối TI thứ  $s$  của nhóm TI thứ  $n$ .

Thông thường, bộ đan xen thời gian cũng đóng vai trò là bộ đệm cho dữ liệu DP trước quy trình xử lý xây dựng khung. Điều này đạt được nhờ hai ngân hàng bộ

nhớ cho mỗi DP. Khối TI thứ nhất được ghi trong ngân hàng thứ nhất. Khối TI thứ hai được ghi trong ngân hàng thứ hai trong khi ngân hàng thứ nhất đang được đọc từ và tiếp tục như vậy.

TI là bộ đan xen khối cột dòng xoắn. Đối với khối TI thứ  $s$  của nhóm TI thứ  $n$ , số lượng các dòng  $N_r$  của bộ nhớ TI bằng với số lượng các ô  $N_{\text{cell}}$ , nghĩa là,  $N_r = N_{\text{cell}}$  trong khi số lượng các cột  $N_c$  bằng với số lượng  $N_{\text{XBLOCK\_TI}}(n, s)$ .

Fig.26 minh họa hoạt động cơ bản của bộ đan xen khối hàng-cột xoắn theo một phương án ví dụ của sáng chế.

Fig.26A minh họa thao tác ghi trong bộ đan xen thời gian và Fig.26B minh họa thao tác đọc trong bộ đan xen thời gian. Như được minh họa trên Fig.26A, XFECBLOCK thứ nhất được ghi trong cột thứ nhất của bộ nhớ đan xen theo thời gian theo hướng cột và XFECBLOCK thứ hai được ghi trong cột tiếp theo, và thao tác như vậy được tiếp tục. Ngoài ra, trong mảng đan xen, ô được đọc theo chiều đường chéo. Như được minh họa trên Fig.26B, trong khi việc đọc theo đường chéo được tiến hành từ hàng thứ nhất (về phía bên phải dọc theo hàng này bắt đầu từ cột ngoài cùng bên trái) tới hàng cuối cùng, các ô  $N_r$  được đọc. Cụ thể hơn, khi giả định rằng  $z_{n,s,i} (i = 0, \dots, N_r N_c)$  là vị trí ô bộ nhớ đan xen thời gian cần được đọc tiếp theo, thao tác đọc trong mảng đan xen được thi hành bằng cách tính chỉ số hàng  $R_{n,s,i}$ , chỉ số cột  $C_{n,s,i}$ , và thông số xoắn kết hợp  $T_{n,s,i}$  như được thể hiện ở Công thức toán học dưới đây.

Công thức toán học 9

$$\begin{aligned} \text{GENERATE}(R_{n,s,i}, C_{n,s,i}) = & \\ \{ & \\ R_{n,s,i} = \text{mod}(i, N_r), & \\ T_{n,s,i} = \text{mod}(S_{\text{shift}} \times R_{n,s,i}, N_c), & \\ C_{n,s,i} = \text{mod}(T_{n,s,i} + \left\lfloor \frac{i}{N_r} \right\rfloor, N_c) & \\ \} & \end{aligned}$$

trong đó,  $S_{\text{shift}}$  là giá trị chuyển dịch chung cho quá trình đọc đường chéo bất kể  $N_{\text{XBLOCK\_TI}}(n, s)$  và giá trị chuyển dịch này được quyết định bởi  $N_{\text{XBLOCK\_TI\_MAX}}$  được đưa ra trong PLS2-STAT như được thể hiện ở công thức toán học dưới đây.

## Công thức toán học 10

$$for \begin{cases} N'_{xBLOCK\_TI\_MAX} = N_{xBLOCK\_TI\_MAX} + 1, & \text{if } N_{xBLOCK\_TI\_MAX} \bmod 2 = 0 \\ N'_{xBLOCK\_TI\_MAX} = N_{xBLOCK\_TI\_MAX}, & \text{if } N_{xBLOCK\_TI\_MAX} \bmod 2 = 1 \end{cases}$$

$$S_{shift} = \frac{N'_{xBLOCK\_TI\_MAX} - 1}{2}$$

Do đó, vị trí ô cần được đọc được tính bởi tọa độ  $z_{n,s,i} = N_r C_{n,s,i} + R_{n,s,i}$ .

Fig.27 minh họa hoạt động của bộ đan xen khối hàng-cột xoắn theo phương án ví dụ khác của sáng chế.

Cụ thể hơn, Fig.27 minh họa mảng đan xen trong bộ nhớ đan xen thời gian cho các nhóm đan xen thời gian tương ứng bao gồm XFECBLOCK ảo khi  $N_{xBLOCK\_TI}(0,0) = 3$ ,  $N_{xBLOCK\_TI}(1,0) = 6$ , và  $N_{xBLOCK\_TI}(2,0) = 5$ .

$N_{xBLOCK\_TI}(n,s) = N_r$  thay đổi sẽ bằng hoặc nhỏ hơn  $N'_{xBLOCK\_TI\_MAX}$ . Do đó để cho bộ thu có được sự đan xen bộ nhớ đơn lẻ bất kể  $N_{xBLOCK\_TI}(n,s)$ , kích thước của mảng đan xen đối với bộ đan xen khối hàng-cột xoắn được thiết lập là kích thước của  $N_r \times N_c = N_{cells} \times N'_{xBLOCK\_TI\_MAX}$  bằng cách chèn XFECBLOCK ảo vào trong bộ nhớ đan xen thời gian và quá trình đọc đạt được như được thể hiện ở công thức toán học dưới đây.

## Công thức toán học 11

$$p = 0;$$

$$for i = 0; i < N_{cells} N'_{xBLOCK\_TI\_MAX}; i = i + 1$$

$$\{ GENERATE(R_{n,s,i}, C_{n,s,i});$$

$$V_i = N_r C_{n,s,i} + R_{n,s,i}$$

$$if V_i < N_{cells} N_{xBLOCK\_TI}(n,s)$$

$$\{$$

$$Z_{n,s,p} = V_i; p = p + 1;$$

$$\}$$

$$\}$$

Số lượng các nhóm đan xen thời gian được thiết lập là 3. Tùy chọn của bộ đan xen thời gian được truyền tín hiệu trong PLS2-STAT bởi DP\_TI\_TYPE = '0', DP\_FRAME\_INTERVAL = '1', và DP\_TI\_LENGTH = '1', tức là, NTI = 1, IJUMP = 1, và PI = 1. Số lượng các XFECBLOCK tương ứng cho mỗi nhóm đan xen thời

gian, mà các  $N_{\text{cell}}$  của nó = 30 được truyền tín hiệu trong dữ liệu PLS2-DYN bởi  $N_{\text{xBLOCK\_TI}}(0,0) = 3$ ,  $N_{\text{xBLOCK\_TI}}(1,0) = 6$ , và  $N_{\text{xBLOCK\_TI}}(2,0) = 5$  của các XFECBLOCK tương ứng. Số lượng lớn nhất của các XFECBLOCK được truyền tín hiệu trong dữ liệu PLS2-STAT bởi  $N_{\text{xBLOCK\_Group\_MAX}}$  và điều này được tiếp tục với  $\lfloor N_{\text{xBLOCK\_Group\_MAX}} / N_{\text{TI}} \rfloor = N_{\text{xBLOCK\_TI\_MAX}} = 6$ .

Fig.28 minh họa mẫu đọc chéo của bộ đan xen khối hàng-cột xoắn theo phương án ví dụ của sáng chế.

Cụ thể hơn, Fig.28 minh họa mẫu đọc đường chéo từ các mảng đan xen tương ứng có các thông số  $N'_{\text{xBLOCK\_TI\_MAX}} = 7$  và  $S_{\text{shift}} = (7-1)/2 = 3$ . Trong trường hợp này, trong quá trình đọc được biểu thị bởi mã giả được đưa ra nêu trên, khi  $V_i \geq N_{\text{cells}} N_{\text{xBLOCK\_TI}}(n, s)$ , giá trị của  $V_i$  được bỏ qua và giá trị tính tiếp theo của  $V_i$  được sử dụng.

Fig.29 minh họa XFECBLOCK được đan xen từ mỗi mảng đan xen theo một phương án ví dụ của sáng chế.

Fig.29 minh họa XFECBLOCK được đan xen từ mỗi mảng đan xen có các thông số  $N'_{\text{xBLOCK\_TI\_MAX}} = 7$  và  $S_{\text{shift}} = 3$  theo một phương án ví dụ của sáng chế.

Fig.30 minh họa môđun đồng bộ hóa và giải điều biến theo một phương án của sáng chế.

Môđun đồng bộ hóa và giải điều biến được minh họa trên Fig.30 tương ứng với phương án của môđun đồng bộ hóa và giải điều biến được mô tả trên Fig.9. Ngoài ra môđun đồng bộ hóa và giải điều biến được minh họa trên Fig.30 có thể thực hiện thao tác ngược với môđun tạo dạng sóng được mô tả trên Fig.9.

Như được minh họa trên Fig.30, môđun đồng bộ hóa và giải điều biến theo phương án của sáng chế như một phương án của môđun đồng bộ hóa và giải điều biến của thiết bị thu sử dụng  $m$  anten Rx có thể bao gồm  $m$  khối xử lý để giải điều biến và đưa ra đầu vào tín hiệu miễn là  $m$  đường. Tất cả  $m$  khối xử lý có thể thực hiện thủ tục xử lý giống nhau. Dưới đây, hoạt động của khối xử lý thứ nhất 30000 trong số  $m$  khối xử lý sẽ được mô tả trước.

Khối xử lý thứ nhất 30000 có thể bao gồm bộ điều hướng 30100, khối ADC 30200, bộ phát hiện đoạn đầu 30300, bộ phát hiện chuỗi bảo vệ 30400, khối biến đổi

dạng sóng 30500, khối đồng bộ thời gian/tần số 30600, bộ phát hiện tín hiệu tham chiếu 30700, bộ cân bằng kênh 30800, và khối biến đổi dạng sóng nghịch đảo 30900.

Bộ điều hướng 30100 lựa chọn dải tần số mong muốn và bù độ lớn của tín hiệu thu được để đưa ra tín hiệu đến khối ADC 30200.

Khối ADC 30200 có thể chuyển đổi tín hiệu được đưa ra từ bộ điều hướng 30100 thành tín hiệu số.

Bộ phát hiện đoạn đầu 30300 có thể phát hiện phần đầu (theo cách khác, tín hiệu phần đầu hoặc ký hiệu đoạn đầu) để kiểm tra xem liệu tín hiệu số là tín hiệu của hệ thống tương ứng với thiết bị thu hay không. Trong trường hợp này, bộ phát hiện đoạn đầu 30300 có thể giải mã các thông số truyền cơ bản thu được qua đoạn đầu.

Bộ phát hiện chuỗi bảo vệ 30400 có thể phát hiện chuỗi bảo vệ theo tín hiệu số. Khối đồng bộ thời gian/tần số 30600 có thể thực hiện sự đồng bộ thời gian/tần số nhờ sử dụng chuỗi bảo vệ được phát hiện và bộ cân bằng kênh 30800 có thể đánh giá kênh qua chuỗi thu được/được khôi phục nhờ sử dụng chuỗi bảo vệ được phát hiện.

Khi việc chuyển đổi dạng sóng ngược được thực hiện ở phía truyền, khối biến đổi dạng sóng 30500 có thể thực hiện thủ tục chuyển đổi ngược để chuyển đổi dạng sóng ngược. Khi hệ thống truyền/thu phát rộng theo phương án của sáng chế là hệ thống đa sóng mang, khối biến đổi dạng sóng 30500 có thể thực hiện thủ tục chuyển đổi FFT. Ngoài ra trong trường hợp mà ở đó hệ thống truyền/thu phát rộng theo phương án của sáng chế là hệ thống sóng mang đơn, khi các tín hiệu thu được trong miền thời gian được sử dụng để được xử lý trong miền tần số hoặc tất cả của các tín hiệu thu được được xử lý trong miền thời gian, khối biến đổi dạng sóng 30500 có thể không được sử dụng.

Khối đồng bộ thời gian/tần số 30600 có thể thu dữ liệu đầu ra của bộ phát hiện đoạn đầu 30300, bộ phát hiện chuỗi bảo vệ 30400, và bộ phát hiện tín hiệu tham chiếu 30700 và thực hiện sự đồng bộ thời gian và sự đồng bộ tần số sóng mang bao gồm việc phát hiện chuỗi bảo vệ và sự định vị của số khối đối với tín hiệu được phát hiện. Trong trường hợp này, khối đồng bộ thời gian/tần số 30600 có thể phản hồi và sử dụng tín hiệu đầu ra của khối biến đổi dạng sóng 30500 để đồng bộ tần số.

Bộ phát hiện tín hiệu tham chiếu 30700 có thể phát hiện tín hiệu chuẩn thu được. Do đó, thiết bị thu theo phương án của sáng chế có thể thực hiện sự đồng bộ hoặc sự đánh giá kênh.

Bộ cân bằng kênh 30800 có thể đánh giá kênh truyền đến mỗi thiết bị thu từ mỗi anten truyền từ chuỗi bảo vệ hoặc tín hiệu chuẩn và thực hiện việc làm cân bằng kênh đối với mỗi dữ liệu thu được nhờ sử dụng kênh được đánh giá.

Khi khối biến đổi dạng sóng 30500 thực hiện chuyển đổi dạng sóng để thực hiện hiệu quả sự đồng bộ và sự đánh giá kênh/sự làm cân bằng, khối biến đổi dạng sóng nghịch đảo 30900 có thể dùng để khôi phục mỗi dữ liệu thu được đến vùng dữ liệu thu được ban đầu một lần nữa. Trong trường hợp mà ở đó hệ thống truyền/thu phát rộng theo phương án của sáng chế là hệ thống sóng mang đơn, khối biến đổi dạng sóng 30500 có thể thực hiện FFT để thực hiện sự đồng bộ/sự đánh giá kênh/sự làm cân bằng trong miền tần số và khối biến đổi dạng sóng nghịch đảo 30900 thực hiện IFFT đối với tín hiệu trong đó việc làm cân bằng kênh được hoàn thành để khôi phục ký hiệu dữ liệu được truyền. Khi hệ thống truyền/thu phát rộng theo phương án của sáng chế là hệ thống đa sóng mang, khối biến đổi dạng sóng nghịch đảo 30900 có thể không được sử dụng.

Ngoài ra các khối nêu trên có thể được loại bỏ theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Fig.31 minh họa môđun tách khung theo một phương án của sáng chế.

Môđun tách khung được minh họa trên Fig.31 tương ứng với phương án của môđun tách khung được mô tả trên Fig.9.

Như được minh họa trên Fig.31, môđun tách khung theo phương án của sáng chế có thể bao gồm ít nhất một hoặc nhiều bộ giải đan xen khối 31000 và ít nhất một hoặc nhiều bộ giải ánh xạ ô 31100.

Bộ giải đan xen khối 31000 có thể thực hiện việc giải đan xen cho dữ liệu trên mỗi tín hiệu khối đối với dữ liệu được nhập vào các đường dữ liệu tương ứng của m anten thu và được xử lý trong môđun đồng bộ hóa và giải điều biến. Trong trường hợp này, như được mô tả trên Fig.8, khi sự đan xen từng cặp được thực hiện ở phía truyền, bộ giải đan xen khối 31000 có thể xử lý hai dữ liệu liên tiếp cho mỗi đường

đầu vào làm một cặp. Do đó, bộ giải đan xen khối 31000 có thể đưa ra hai dữ liệu liên tiếp đầu ra ngay cả khi giải đan xen dữ liệu. Ngoài ra bộ giải đan xen khối 31000 thực hiện thủ tục ngược với thủ tục đan xen được thực hiện ở phía truyền để đưa ra dữ liệu theo thứ tự dữ liệu ban đầu.

Bộ giải ánh xạ ô 31100 có thể trích các ô tương ứng với dữ liệu thông thường từ khung tín hiệu thu được, các ô tương ứng với ống dữ liệu, và các ô tương ứng với dữ liệu PLS. Trong trường hợp cần thiết, bộ giải ánh xạ ô 31100 hợp nhất dữ liệu được phân phối và được truyền đến các phần để đưa ra dữ liệu hợp nhất làm một dòng. Ngoài ra như được mô tả trên Fig.7, khi hai dữ liệu đầu vào ô liên tiếp được xử lý như một cặp để được ánh xạ, bộ giải ánh xạ ô 31100 có thể thực hiện việc giải ánh xạ từng cặp ô để xử lý hai ô đầu vào liên tiếp làm một bộ như thủ tục ngược tương ứng với nó.

Ngoài ra bộ giải ánh xạ ô 31100 có thể trích và đưa ra tương ứng tất cả dữ liệu truyền tín hiệu PLS thu được qua khung hiện tại như dữ liệu PLS trước và dữ liệu PLS sau.

Các khối nêu trên có thể được bỏ qua theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Fig.32 minh họa môđun giải ánh xạ và giải mã theo một phương án của sáng chế.

Môđun giải ánh xạ và giải mã được minh họa trên Fig.32 tương ứng với phương án của môđun giải ánh xạ và giải mã được mô tả trên Fig.9.

Như được nêu trên, môđun mã hóa và điều biến của thiết bị truyền theo phương án của sáng chế có thể áp dụng độc lập và xử lý các sơ đồ SISO, MISO, và MIMO để nhập vào các ống dữ liệu đối với các đường tương ứng. Do đó, môđun giải ánh xạ và giải mã được minh họa trên Fig.32 có thể cũng bao gồm các khối cho dữ liệu xử lý SISO, MISO, và MIMO được đưa ra từ bộ phân tách khung tương ứng với thiết bị truyền.

Như được minh họa trên Fig.32, môđun giải ánh xạ và giải mã theo phương án của sáng chế có thể bao gồm khối thứ nhất 32000 đối với sơ đồ SISO, khối thứ hai 32100 đối với sơ đồ MISO, và khối thứ ba 32200 đối với sơ đồ MIMO, và khối

thứ tư 32300 xử lý thông tin PLS trước/sau. Môđun giải ánh xạ và giải mã được minh họa trên Fig.32 chỉ là một phương án và môđun giải ánh xạ và giải mã có thể bao gồm chỉ khối thứ nhất 32000 và khối thứ tư 32300, chỉ khối thứ hai 32100 và khối thứ tư 32300, và chỉ khối thứ ba 32200 và khối thứ tư 32300 theo mục đích của nhà thiết kế. Nghĩa là, môđun giải ánh xạ và giải mã có thể bao gồm các khối để xử lý các ống dữ liệu tương ứng giống hoặc khác nhau theo mục đích của nhà thiết kế.

Dưới đây, mỗi khối sẽ được mô tả.

Khối thứ nhất 32000 là khối để xử lý SISO ống dữ liệu đầu vào có thể bao gồm khối bộ giải đan xen thời gian 32010, khối bộ giải đan xen ô 32020, khối bộ giải ánh xạ chòm điểm 32030, khối đa hợp ô với bit 32040, khối bộ giải đan xen bit 32050, và khối bộ giải mã FEC 32060.

Khối bộ giải đan xen thời gian 32010 có thể thực hiện thủ tục ngược với khối bộ đan xen thời gian. Nghĩa là, khối bộ giải đan xen thời gian 32010 có thể giải đan xen ký hiệu đầu vào được đan xen trong miền thời gian đến vị trí ban đầu.

Khối bộ giải đan xen ô 32020 có thể thực hiện thủ tục ngược với khối bộ đan xen ô. Nghĩa là, khối bộ giải đan xen ô 32020 có thể giải đan xen các vị trí của các ô mở rộng trong một khối FEC đến các vị trí ban đầu.

Khối bộ giải ánh xạ chòm điểm 32030 có thể thực hiện thủ tục ngược với khối bộ ánh xạ chòm điểm. Nghĩa là, khối bộ giải ánh xạ chòm điểm 32030 có thể giải ánh xạ tín hiệu đầu vào của miền ký hiệu đến dữ liệu của miền bit. Ngoài ra khối bộ giải ánh xạ chòm điểm 32030 có thể đưa ra dữ liệu bit được quyết định bằng cách thực hiện sự quyết định rõ ràng và đưa ra tỷ số logarit hợp lý (Log-Likelihood Ratio, viết tắt là LLR) của mỗi bit tương ứng với giá trị quyết định mềm hoặc giá trị xác suất. Khi phía truyền ứng dụng chòm điểm được quay để thu độ lợi phân tập bổ sung, khối bộ giải ánh xạ chòm điểm 32030 có thể thực hiện giải ánh xạ LLR 2 chiều tương ứng với chòm điểm được quay. Trong trường hợp này, bộ giải ánh xạ chòm điểm 32030 có thể thực hiện sự tính toán sao cho thiết bị truyền bù giá trị trễ được thực hiện đối với thành phần I hoặc Q ở thời điểm tính toán LLR.

Khối đa hợp ô với bit 32040 có thể thực hiện thủ tục ngược với khối tách bit với ô. Nghĩa là, khối đa hợp ô với bit 32040 có thể khôi phục dữ liệu bit được ánh xạ trong khối tách bit với ô để tạo nên dòng bit ban đầu.

Khối bộ giải đan xen bit 32050 có thể thực hiện thủ tục ngược với bộ đan xen bit khối. Nghĩa là, khối bộ giải đan xen bit 32050 có thể giải đan xen dòng bit được đưa ra từ khối đa hợp ô với bit 32040 theo thứ tự ban đầu.

Khối bộ giải mã FEC 32060 có thể thực hiện thủ tục ngược với khối bộ mã hóa FEC. Nghĩa là, khối bộ giải mã FEC 32060 có thể sửa lỗi mà có mặt trên kênh truyền bằng cách thực hiện việc giải mã LDPC và việc giải mã BCH.

Khối thứ hai 32100 là khối để xử lý theo MISO ống dữ liệu đầu vào có thể bao gồm khối bộ giải đan xen thời gian, khối bộ giải đan xen ô, khối bộ giải ánh xạ chòm điểm, khối đa hợp ô với bit, khối bộ giải đan xen bit, và khối bộ giải mã FEC giống như khối thứ nhất 32000 như được minh họa trên Fig.32, nhưng khối thứ hai 32100 khác với khối thứ nhất 32000 trong đó khối thứ hai 32100 còn bao gồm khối giải mã MISO 32110. Vì khối thứ hai 32100 thực hiện thủ tục đóng vai trò giống nhau từ bộ giải đan xen thời gian đến đầu ra giống như khối thứ nhất 32000, phần mô tả của các khối giống nhau sẽ được bỏ qua.

Khối giải mã MISO 32110 có thể thực hiện thủ tục ngược với khối xử lý MISO. Khi hệ thống truyền/thu phát rộng theo phương án của sáng chế là hệ thống sử dụng STBC, khối giải mã MISO 32110 có thể thực hiện giải mã Alamouti.

Khối thứ ba 32200 là khối xử lý theo MIMO ống dữ liệu đầu vào có thể bao gồm khối bộ giải đan xen thời gian, khối bộ giải đan xen ô, khối bộ giải ánh xạ chòm điểm, khối đa hợp ô với bit, khối bộ giải đan xen bit, và khối bộ giải mã FEC giống như khối thứ hai 32100 như được minh họa trên Fig.32, nhưng khối thứ ba 32200 khác với khối thứ hai 32100 ở chỗ khối thứ ba 32200 còn bao gồm khối giải mã MIMO 32210. Các hoạt động của các khối bộ giải đan xen thời gian, khối bộ giải đan xen ô, khối bộ giải ánh xạ chòm điểm, khối đa hợp ô với bit, và khối bộ giải đan xen bit được bao gồm trong khối thứ ba 32200 có thể khác với các hoạt động và các chức năng chi tiết của các khối tương ứng được bao gồm trong các khối thứ nhất và thứ hai 32000 và 32100, nhưng các khối được bao gồm trong khối thứ ba 32200 là giống như các khối được bao gồm trong các khối thứ nhất và thứ hai về các vai trò cơ bản.

Khối giải mã MIMO 32210 có thể thu dữ liệu đầu ra của bộ giải đan xen ô là đầu vào đối với tín hiệu đầu vào của m anten thu và thực hiện giải mã MIMO như

thủ tục ngược với khối xử lý MIMO. Khối giải mã MIMO 32210 có thể thực hiện việc giải mã hợp lệ lớn nhất để thực hiện hiệu suất giải mã lớn nhất hoặc giải mã cầu để làm giảm độ phức tạp. Theo cách khác, khối giải mã MIMO 32210 thực hiện sự phát hiện MMSE hoặc thực hiện giải mã lặp kết hợp với sự phát hiện MMSE để đạt được hiệu suất giải mã cải thiện.

Khối thứ tư 32300 là khối để xử lý thông tin PLS trước/sau có thể thực hiện giải mã SISO hoặc MISO. Khối thứ tư 32300 có thể thực hiện thủ tục ngược với khối thứ tư.

Các thao tác của bộ giải đan xen thời gian, bộ giải đan xen ô, bộ giải ánh xạ chòm điểm, đa hợp ô với bit, và các khối bộ giải đan xen bit được bao gồm trong khối thứ tư 32300 có thể khác với các thao tác và các chức năng chi tiết của các khối tương ứng được bao gồm trong các khối từ thứ nhất đến thứ ba từ 32000 đến 32200, nhưng các khối được bao gồm trong khối thứ tư 32300 là giống như các khối được bao gồm trong các khối từ thứ nhất đến thứ ba về các vai trò cơ bản.

Bộ giải mã FEC được rút ngắn/được chấm thủng 32310 có thể thực hiện thủ tục ngược với khối bộ mã hóa FEC được rút ngắn/được chấm thủng. Nghĩa là, bộ giải mã FEC được rút ngắn/được chấm thủng 32310 có thể thực hiện giải rút ngắn và giải chấm thủng, và sau đó, FEC giải mã dữ liệu thu được trong khi được rút ngắn/được chấm thủng theo độ dài của dữ liệu PLS. Trong trường hợp này, vì bộ giải mã FEC được sử dụng trong ống dữ liệu có thể cũng được sử dụng ngay cả trong PLS, phân tách phần cứng bộ giải mã FEC đối với chỉ PLS là không cần thiết, và kết quả là, việc thiết kế hệ thống là đơn giản và việc mã hóa hiệu quả là khả dụng.

Các khối nêu trên có thể được bỏ qua theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Kết quả là, như được minh họa trên Fig.32, môđun giải ánh xạ và giải mã theo phương án của sáng chế có thể xuất đến bộ xử lý đầu ra ống dữ liệu và PLS thông tin được xử lý cho mỗi đường dẫn.

Fig.33 và Fig.34 minh họa bộ xử lý đầu ra theo một phương án của sáng chế.

Fig.33 minh họa bộ xử lý đầu ra theo một phương án của sáng chế.

Bộ xử lý đầu ra được minh họa trên Fig.33 tương ứng với phương án của bộ

xử lý đầu ra được mô tả trên Fig.9. Ngoài ra bộ xử lý đầu ra được minh họa trên Fig.33 được sử dụng để thu ống dữ liệu đơn được đưa ra từ môđun giải ánh xạ và giải mã và đưa ra dòng đầu ra đơn có thể thực hiện thao tác ngược với môđun định dạng đầu vào.

Bộ xử lý đầu ra trên Fig.33 thực hiện các chức năng, các quy trình, và/hoặc các phương pháp được đề xuất trên Fig.50, Fig.51, và Fig.53 được mô tả dưới đây.

Bộ xử lý đầu ra được minh họa trên Fig.33 có thể bao gồm khối bộ giải xáo trộn BB 33000, khối loại bỏ phần đệm 33100, khối bộ giải mã CRC-8 33200, và khối bộ xử lý khung BB 33300.

Khối bộ giải xáo trộn BB 33000 tạo ra PRBS giống như được sử dụng ở phía truyền đối với dòng bit đầu vào và thao tác XOR đối với PRBS và dòng bit để thực hiện giải xáo trộn.

Khối loại bỏ phần đệm 33100 có thể loại bỏ bit đệm được chèn bởi phía truyền khi cần thiết.

Khối bộ giải mã CRC-8 33200 thực hiện giải mã CRC dòng bit thu được từ khối loại bỏ phần đệm 33100 để kiểm tra lỗi khối.

Khối bộ xử lý khung BB 33300 có thể giải mã thông tin được truyền đến phần đầu khung BB và khôi phục MP3G-TS, dòng IP (v4 hoặc v6), hoặc dòng chung.

Các khối nêu trên có thể được bỏ qua theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Fig.34 minh họa bộ xử lý đầu ra theo phương án khác của sáng chế.

Bộ xử lý đầu ra được minh họa trên Fig.34 tương ứng với phương án của bộ xử lý đầu ra được mô tả trên Fig.9. Ngoài ra bộ xử lý đầu ra được minh họa trên Fig.34 tương ứng với trường hợp thu nhiều ống dữ liệu được đưa ra từ môđun giải ánh xạ và giải mã. Giải mã nhiều ống dữ liệu có thể bao gồm trường hợp gộp dữ liệu thông thường mà có thể được áp dụng chung cho các ống dữ liệu và ống dữ liệu được kết hợp với dữ liệu thông thường và giải mã dữ liệu thông thường và ống dữ liệu được gộp hoặc trường hợp trong đó thiết bị thu giải mã đồng thời một số các dịch vụ hoặc các thành phần dịch vụ (bao gồm dịch vụ video mở rộng được).

Bộ xử lý đầu ra được minh họa trên Fig.34 có thể bao gồm khối bộ giải xáo trộn BB, khối loại bỏ phần đệm, khối bộ giải mã CRC-8, và khối bộ xử lý khung BB 33300 giống như bộ xử lý đầu ra.

Bộ xử lý đầu ra trên Fig.34 thực hiện các chức năng, các quy trình, và/hoặc các phương pháp được đề xuất trên Fig.50, Fig.51 và Fig.53 được mô tả dưới đây.

Các khối tương ứng có thể khác với các khối được mô tả trên Fig.33 xét về các thao tác và các thao tác chi tiết, nhưng các khối tương ứng là giống như các khối trên Fig.33 xét về vai trò cơ bản.

Khối bộ đệm loại bỏ sự chập chờn 34000 được bao gồm trong bộ xử lý đầu ra được minh họa trên Fig.34 có thể bù độ trễ được chèn ở phía truyền thông số thời gian được đưa ra (Time To Output, viết tắt là TTO) đã khôi phục để đồng bộ hóa nhiều ống dữ liệu.

Ngoài ra khối chèn gói trống 34100 có thể khôi phục gói trống được loại bỏ trong dòng dựa vào thông tin gói trống được xóa (Deleted Null Package, viết tắt là DNP) được khôi phục và đưa ra dữ liệu thông thường.

Khối tạo xung nhịp TS 34200 có thể khôi phục sự đồng bộ thời gian chi tiết của gói đầu ra dựa vào ISCR – thông tin tham chiếu thời gian dòng đầu vào.

Khối tái kết hợp TS 34300 tái kết hợp dữ liệu thông thường được đưa ra từ khối chèn gói trống 34100 và các ống dữ liệu được kết hợp với dữ liệu thông thường để khôi phục dữ liệu thông thường và các ống dữ liệu được tái kết hợp thành MPEG-TS ban đầu, dòng IP (v4 hoặc v6), hoặc dòng chung và đưa ra MPEG-TS được phục hồi, dòng IP (v4 hoặc v6), hoặc dòng chung. Thông tin TTO, DNP, và ISCR đều có thể thu được qua phần đầu khung BB.

Khối bộ giải mã truyền tín hiệu trong dải 34400 có thể khôi phục và đưa ra thông tin truyền tín hiệu lớp vật lý trong dải được truyền qua trường bit đệm trong mỗi khung FEC của ống dữ liệu.

Bộ xử lý đầu ra được minh họa trên Fig.34 thực hiện giải xáo trộn BB thông tin PLS trước và thông tin PLS sau được nhập vào theo đường dẫn PLS trước và đường dẫn PLS sau, và giải mã dữ liệu được xáo trộn để khôi phục dữ liệu PLS ban đầu. Dữ liệu PLS được khôi phục có thể được chuyển đến bộ điều khiển hệ thống

trong thiết bị thu và bộ điều khiển hệ thống có thể cung cấp thông số được yêu cầu đến môđun đồng bộ hóa và giải điều biến, môđun tách khung, môđun giải ánh xạ và giải mã, và môđun bộ xử lý đầu ra trong thiết bị thu.

Các khối nêu trên có thể được bỏ qua theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Fig.35 minh họa môđun mã hóa và điều biến theo phương án khác của sáng chế.

Môđun mã hóa và điều biến được minh họa trên Fig.35 có thể bao gồm khối thứ nhất 35000 đối với sơ đồ SISO, khối thứ hai 35100 đối với sơ đồ MISO, và khối thứ ba 35200 đối với sơ đồ MIMO, và khối thứ tư 35300 để xử lý thông tin PLS trước/sau để điều khiển QoS đối với mỗi dịch vụ hoặc thành phần dịch vụ được truyền qua mỗi ống dữ liệu. Ngoài ra môđun mã hóa và điều biến theo phương án của sáng chế có thể bao gồm các khối để xử lý giống hoặc khác nhau các ống dữ liệu tương ứng theo mục đích của nhà thiết kế như được nêu trên. Các khối từ thứ nhất đến thứ tư 35000 đến 35300 được minh họa trên Fig.35 có thể bao gồm các khối về cơ bản giống nhau làm các khối từ thứ nhất đến thứ tư.

Tuy nhiên, các khối từ thứ nhất đến thứ tư 35000 đến 35300 khác với các khối từ thứ nhất đến thứ tư nêu trên trong đó chức năng của khối bộ ánh xạ chòm điểm 35010 được bao gồm trong các khối từ thứ nhất đến thứ ba 35000 đến 35200 khác với chức năng của khối bộ ánh xạ chòm điểm được bao gồm trong các khối từ thứ nhất đến thứ ba, và khối bộ đan xen I/O và quay 35020 được bao gồm giữa bộ đan xen ô và bộ đan xen thời gian của các khối từ thứ nhất đến thứ tư 35000 đến 35300, và cấu hình của khối thứ ba 35200 đối với sơ đồ MIMO khác với cấu hình của khối thứ ba đối với sơ đồ MIMO.

Khối bộ giải ánh xạ chòm điểm 35010 được minh họa trên Fig.35 có thể ánh xạ từ-bit đầu vào đến ký hiệu phức tạp.

Khối bộ ánh xạ chòm điểm 35010 được minh họa trên Fig.35 có thể được áp dụng chung cho các khối từ thứ nhất đến thứ ba 35000 đến 35200 như được nêu trên.

Khối bộ đan xen I/O và quay 35020 đan xen độc lập các thành phần theo pha và pha vuông góc của các ký hiệu phức tạp tương ứng của dữ liệu đan xen ô được

đưa ra từ bộ đan xen ô để đưa ra các thành phần được đan xen bởi bộ ký hiệu. Số lượng dữ liệu đầu vào và các ký hiệu đầu ra của khối bộ đan xen I/O và quay 35020 là hai hoặc nhiều hơn và có thể được thay đổi theo mục đích của nhà thiết kế. Ngoài ra khối bộ đan xen I/O và quay 35020 có thể không đan xen các thành phần theo pha.

Khối bộ đan xen I/O và quay 35020 có thể được áp dụng chung cho các khối từ thứ nhất đến thứ tư 35000 đến 35300 như được nêu trên. Trong trường hợp này, xem liệu khối bộ đan xen I/O và quay 35020 có được áp dụng cho khối thứ tư 35300 để xử lý thông tin PLS trước/sau có thể được truyền tín hiệu qua phân đầu nêu trên hay không.

Khối thứ ba 35200 đối với sơ đồ MIMO có thể bao gồm khối bộ đan xen khối Q 35210 và khối bộ tạo ký hiệu phức tạp 35220 như được minh họa trên Fig.35.

Khối bộ đan xen khối Q 35210 có thể thực hiện sự hoán vị phần chẵn lẻ của khối FEC đã mã hóa FEC thu được từ bộ mã hóa FEC. Do đó, phần chẵn lẻ của ma trận LDPC H có thể được thực hiện theo cấu trúc vòng giống như phần thông tin. Khối bộ đan xen khối Q 35210 hoán vị các thứ tự của các khối bit có kích thước Q trong ma trận LDPC H và sau đó, thực hiện sự đan xen khối hàng-cột của các khối bit để tạo ra và đưa ra dòng bit cuối cùng.

Khối bộ tạo ký hiệu phức tạp 35220 có thể thu các dòng bit được đưa ra từ khối bộ đan xen khối Q 35210 và ánh xạ các dòng bit thu được với ký hiệu phức tạp và đưa ra các dòng bit và ký hiệu phức tạp được ánh xạ. Trong trường hợp này, khối bộ tạo ký hiệu phức tạp 35220 có thể đưa ra các ký hiệu qua ít nhất hai đường dẫn. Điều này có thể được thay đổi theo mục đích của nhà thiết kế.

Các khối nêu trên có thể được bỏ qua theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Kết quả là, như được minh họa trên Fig.35, mã hóa và điều biến theo phương án khác của sáng chế có thể đưa ra ống dữ liệu, thông tin PLS trước, và thông tin PLS sau được xử lý cho mỗi đường dẫn đến môđun cấu trúc khung.

Fig.36 minh họa môđun giải ánh xạ và giải mã theo phương án khác của sáng chế.

Môđun giải ánh xạ và giải mã được minh họa trên Fig.36 tương ứng với

phương án khác của môđun giải ánh xạ và giải mã được mô tả trên Fig.9 và Fig.32. Ngoài ra môđun giải ánh xạ và giải mã được minh họa trên Fig.36 có thể thực hiện thao tác ngược với môđun mã hóa và điều biến được mô tả trên Fig.35.

Như được minh họa trên Fig.36, môđun giải ánh xạ và giải mã theo phương án khác của sáng chế có thể bao gồm khối thứ nhất 36000 đối với sơ đồ SISO, khối thứ hai 36100 đối với sơ đồ MISO, khối thứ ba 36200 đối với sơ đồ MIMO, và khối thứ tư 36300 để xử lý thông tin PLS trước/sau. Ngoài ra môđun giải ánh xạ và giải mã theo phương án của sáng chế có thể bao gồm các khối để xử lý giống hoặc khác nhau các ống dữ liệu tương ứng theo mục đích của nhà thiết kế như được nêu trên. Các khối từ thứ nhất đến thứ tư từ 36000 đến 36300 được minh họa trên Fig.36 có thể bao gồm các khối về cơ bản giống nhau làm các khối từ thứ nhất đến thứ tư từ 32000 đến 32300 được mô tả trên Fig.32.

Tuy nhiên, các khối từ thứ nhất đến thứ tư từ 36000 đến 36300 khác với các khối từ thứ nhất đến thứ tư nêu trên trong đó khối bộ giải đan xen I/Q và giải quay 36010 được bao gồm giữa bộ giải đan xen thời gian và bộ giải đan xen ô, chức năng khối bộ giải ánh xạ chòm điểm 36020 được bao gồm trong các khối từ thứ nhất đến thứ ba từ 36000 đến 36200 khác với chức năng của bộ ánh xạ chòm điểm 42030 được bao gồm trong các khối từ thứ nhất đến thứ ba từ 32000 đến 32200 trên Fig.32, và cấu hình của khối thứ ba 36200 đối với sơ đồ MIMO khác với cấu hình của khối thứ ba 36200 đối với sơ đồ MIMO được minh họa trên Fig.36. Dưới đây, các khối giống nhau như Fig.36 sẽ không được mô tả và các sự khác biệt nêu trên sẽ được mô tả trước.

Khối bộ giải đan xen I/Q và giải quay 36010 có thể thực hiện thủ tục ngược với khối bộ đan xen IQ và quay 35020 được mô tả trên Fig.35. Nghĩa là, khối bộ giải đan xen I/Q và giải quay 36010 có thể giải đan xen các thành phần I và Q được đan xen I/Q và được truyền ở phía truyền và giải quay và đưa ra ký hiệu phức tạp có thành phần I/Q được khôi phục một lần nữa.

Khối bộ giải đan xen I/Q và giải quay 36010 có thể được áp dụng chung cho các khối từ thứ nhất đến thứ tư từ 36000 đến 36300 như được nêu trên. Trong trường hợp này, xem liệu khối bộ giải đan xen I/Q và giải quay 36010 được áp dụng cho khối thứ tư 36300 để xử lý thông tin PLS trước/sau có thể được truyền tín hiệu qua

phần đầu nêu trên hay không.

Khối bộ giải ánh xạ chòm điểm 36020 có thể thực hiện thủ tục ngược với khối bộ ánh xạ chòm điểm 35010 được mô tả trên Fig.35. Nghĩa là, khối bộ giải ánh xạ chòm điểm 36020 có thể không thực hiện việc giải quay, nhưng giải ánh xạ dữ liệu đã giải đan xen ô.

Khối thứ ba 36200 đối với sơ đồ MIMO có thể bao gồm khối bộ tạo ký hiệu phức tạp 36210 và khối bộ giải đan xen khối Q 36220 như được minh họa trên Fig.36.

Khối tách ký hiệu phức tạp 36210 có thể thực hiện thủ tục ngược với khối bộ tạo ký hiệu phức tạp 35220 được mô tả trên Fig.35. Nghĩa là, khối tách ký hiệu phức tạp 36210 có thể tách ký hiệu dữ liệu phức tạp, và giải ánh xạ ký hiệu dữ liệu phức tạp được tách đến dữ liệu bit và dữ liệu đầu ra. Trong trường hợp này, khối tách ký hiệu phức tạp 36210 có thể thu các ký hiệu dữ liệu phức tạp qua ít nhất hai đường dẫn.

Khối bộ giải đan xen khối Q 36220 có thể thực hiện thủ tục ngược với khối bộ đan xen khối Q 35210 được mô tả trên Fig.35. Nghĩa là, khối bộ giải đan xen khối Q 36220 có thể khôi phục các khối kích thước Q nhờ việc giải đan xen hàng-cột, khôi phục các thứ tự đã hoán vị của các khối tương ứng về các thứ tự ban đầu, và sau đó, khôi phục các vị trí của các bit chẵn lẻ về các vị trí ban đầu qua việc giải đan xen chẵn lẻ và đưa ra các bit chẵn lẻ.

Các khối nêu trên có thể được bỏ qua theo mục đích của nhà thiết kế hoặc được thay thế bởi các khối khác có chức năng giống hoặc tương tự nhau.

Kết quả là, như được minh họa trên Fig.36, môđun giải ánh xạ và giải mã theo phương án khác của sáng chế có thể đưa ra ông dữ liệu và PLS thông tin được xử lý cho mỗi đường dẫn đến bộ xử lý đầu ra.

Dưới đây, cấu trúc phần đầu BBF mới để làm giảm chi phí của việc truyền BBF và bổ sung các chức năng khác nhau sử dụng trường đệm được đề xuất theo sáng chế sẽ được mô tả chi tiết.

Fig.37 minh họa một ví dụ về môđun thích ứng chế độ được đề xuất theo sáng chế.

Như được nêu trên, môđun định dạng đầu vào bao gồm môđun thích ứng chế độ.

Cấu hình của môđun thích ứng chế độ trên Fig.37 có thể khác một phần so với cấu hình của môđun thích ứng chế độ được nêu trên.

Như được minh họa trên Fig.37, môđun thích ứng chế độ có thể được tạo cấu hình để bao gồm ít nhất một trong số khối xử lý hoặc tách trước 3710, khối giao diện đầu vào 3720, khối bộ đồng bộ dòng đầu vào 3730, khối bù trễ 3740, khối nén phần đầu 3750, khối tái sử dụng dữ liệu trống 3760, khối phát hiện gói trống 3770, và khối chèn phần đầu khung BB 3780.

Khối xử lý trước có thể tách hoặc giải đa hợp các dòng đầu vào đến các ống dữ liệu. Ở đây, ống dữ liệu có thể được gọi là ống lớp vật lý (PLP). Ở đây, dòng đầu vào có thể là MPEG2-TS, giao thức internet (IP), và/hoặc dòng chung (GS).

Theo một số phương án, dòng đầu vào có dạng khác nhau có thể cũng khả dụng.

Khối nén phần đầu có thể nén phần đầu gói. Điều này có thể được sử dụng để cải thiện hiệu suất truyền của TS hoặc IP dòng đầu vào. Vì bộ thu có thông tin ưu tiên của phần đầu, dữ liệu đã biết có thể được loại bỏ ở phía truyền. Chẳng hạn, thông tin chẳng hạn như PID, hoặc tương tự có thể được nén và thông tin có các dạng khác nhau có thể được loại bỏ hoặc được thay thế. Theo một số phương án, khối nén phần đầu có thể được đặt sau khối xóa gói trống.

Khối tái sử dụng dữ liệu trống có thể thực hiện thao tác chèn dữ liệu trống vào gói sau khi nén phần đầu. Khối này có thể được bỏ qua theo một số phương án.

Khối chèn phần đầu khung BB có thể thao tác theo chế độ khác với khối chèn phần đầu khung BB nêu trên.

Sáng chế đề xuất phương pháp để làm giảm việc truyền tín hiệu của độ dài trường dữ liệu của khung (phương pháp làm giảm tín hiệu độ dài trường dữ liệu).

Ngoài ra sáng chế đề xuất phương pháp để làm giảm chi phí của việc truyền của khung BB đến khối FEC.

Nghĩa là, phương pháp cấu hình khung BB mới được đề xuất theo sáng chế

có thể được thực hiện trong khối chèn phần đầu khung BB.

Nhờ phương pháp được đề xuất theo sáng chế, khung BB và phần đầu khung BB có thể được tạo cấu hình. Sáng chế có thể đề cập đến thủ tục trong đó khung BB được tạo ra để truyền dòng đầu vào đến khối FEC qua đầu vào xử lý.

Ngoài ra, sáng chế có thể đề cập đến phương pháp làm tăng hiệu suất truyền bằng cách làm giảm kích thước của phần đầu khung BB. Các nội dung chi tiết được kết hợp với khối chèn phần đầu khung BB sẽ được mô tả dưới đây.

Theo kỹ thuật đã biết, trong khung BB, độ dài trường dữ liệu (DFL) được phân phối đến mỗi phần đầu khung BB để thông báo độ dài của trường dữ liệu đến thiết bị thu. DFL có thể là 16 bit hoặc 11 bit. Kết quả là, kỹ thuật đã biết cần chi phí lớn để truyền BBF.

Khi độ dài trường dữ liệu được thay đổi trong khung BB có cùng kích thước liên tục, khung BB có thể không được điền đầy hoàn toàn với dữ liệu hoặc khung BB có thể bao gồm thông tin truyền tín hiệu trong dải.

Trong kỹ thuật đã biết khác, khung BB được truyền chỉ ký hiệu chỉ báo thay vì thông báo trực tiếp độ dài của trường dữ liệu. Ngoài ra, khung BB được truyền tín hiệu độ dài của phần đệm của khung BB trong phần đệm. Tuy nhiên, trong trường hợp này, vì truyền tín hiệu trong dải không được xét đến, khi việc truyền tín hiệu trong dải được thao tác, có thể có hạn chế.

Phương pháp được đề xuất theo sáng chế có thể là phương pháp tạo cấu hình phần đầu khung BB mà có thể làm giảm DFL và chèn trường bổ sung. Ở đây, trường bổ sung có thể chỉ báo loại truyền tín hiệu trong dải, hoặc tương tự hoặc có thể được sử dụng cho mục đích khác.

Qua phương pháp được đề xuất theo sáng chế, chi phí để truyền BBF có thể được tối thiểu hóa và các chức năng khác nhau có thể được bổ sung vào trường phần đệm (theo cách khác, độ).

Fig.38 minh họa một ví dụ về bộ xử lý đầu ra được đề xuất theo sáng chế.

Như được nêu trên, bộ xử lý đầu ra có thể bao gồm khối bộ tách phần đầu khung BB. Các thành phần của bộ xử lý đầu ra trên Fig.38 có thể khác một phần so

với các thành phần của bộ xử lý đầu ra được nêu trên.

Bộ xử lý đầu ra trên Fig.38 thực hiện các chức năng, các quy trình, và/hoặc các phương pháp được đề xuất trên Fig.50, Fig.51 và Fig.53 được mô tả dưới đây.

Bộ xử lý đầu ra có thể được tạo cấu hình để bao gồm ít nhất một trong số khối bộ tách phần đầu khung BB 3810, khối chèn gói trống 3820, khối tái tạo dữ liệu trống 3830, khối giải nén phần đầu 3840, khối tạo xung nhịp TS 3850, khối bộ đệm loại bỏ sự chập chờn 3860, và khối tái kết hợp TS 3870.

Ở đây, khối chèn gói trống, khối tạo xung nhịp TS, khối bộ đệm loại bỏ sự chập chờn, và khối tái kết hợp TS có thể thực hiện các thao tác giống như các khối của bộ xử lý đầu ra.

Phần đầu khung BB tạo cấu hình theo phương pháp được đề xuất theo sáng chế có thể tương ứng với khối bộ tách phần đầu khung BB ở phía thu (theo cách khác, thiết bị thu hoặc bộ thu).

Khối bộ tách phần đầu khung BB 3810 có thể thao tác khác với khối bộ tách phần đầu khung BB. Khối bộ tách phần đầu khung BB 3810 có thể thực hiện thao tác tách phần đầu khung BB theo phương pháp được đề xuất theo sáng chế.

Khung BB và phần đầu khung BB tạo cấu hình theo phương pháp được đề xuất theo sáng chế sẽ được mô tả dưới đây.

Khối tái tạo dữ liệu trống có thể tương ứng với khối tái sử dụng dữ liệu trống ở phía thu. Khối tái tạo dữ liệu trống có thể đưa ra đầu ra đến khối giải nén nghe. Khối này có thể được bỏ qua theo một số phương án.

Khối giải nén phần đầu có thể tương ứng với khối nén phần đầu ở phía thu. Khối giải nén phần đầu có thể khôi phục việc nén của phần đầu gói được nén. Như được nêu trên, phần đầu gói có thể được nén để cải thiện hiệu suất truyền của TS hoặc IP dòng đầu vào. Theo một số phương án, khối giải nén phần đầu có thể được đặt trước khối chèn gói trống.

Fig.39 minh họa một ví dụ về cấu trúc khung BB theo kỹ thuật đã biết.

Các dòng dữ liệu được nhập vào mô đun định dạng đầu vào, cụ thể là, mô đun thích ứng chế độ có thể được cắt với độ dài thích hợp sao cho mô đun BICM có thể

thực hiện FEC. Do đó, khung BB có thể được tạo ra.

Độ dài trường dữ liệu của khung BB tương ứng với giá trị thu được bằng cách trừ đi độ dài của phần đầu khung BB từ độ dài tổng của khung BB.

Gói người dùng (User Package, viết tắt là UP) thực tế có thể được chèn vào phần trường dữ liệu của BBF.

Độ dài của trường dữ liệu có thể được thông báo trong trường độ dài trường dữ liệu (Data Field Length, viết tắt là DFL) của phần đầu khung BB. Trường DFL có thể được thể hiện như DFL.

Khung BB được tạo ra qua định dạng đầu vào có thể được mã hóa trong khối định trước FEC.

Ở đây, độ dài tổng của khung BB có thể được cố định.

Ngoài ra khi độ dài của trường dữ liệu của BBF được thay đổi, khung BB có thể không được điền đầy hoàn toàn bằng UP bởi vì UP không đầy đủ hoặc thông tin truyền tín hiệu trong dải có thể được bao gồm một cách cố ý.

Khi khung BB có thể không được điền đầy hoàn toàn, khoảng trống tương ứng có thể được điền đầy với phần đệm. Phần đệm có thể được thể hiện như phần đệm.

Fig.40 minh họa ví dụ khác nữa về cấu trúc khung BB theo kỹ thuật đã biết.

Như được minh họa trên Fig.40b, khi trường dữ liệu (theo cách khác, phụ tải) của khung BB không được điền đầy hoàn toàn với dữ liệu cần được truyền, các byte đệm có thể được chèn.

Trường STUFFI có thể được chèn vào phần đầu BBF để truyền tín hiệu các byte đệm. Phần đầu BBF là phần đầu TS.

Trường STUFFI thể hiện ký hiệu chỉ báo là 1 bit chỉ báo xem liệu các byte đệm có mặt trong khung BB hay không.

Khi phụ tải của khung BB được điền đầy hoàn toàn với UP, các byte đệm không có mặt. Trong trường hợp này, STUFFI có thể được thiết đặt là '0'.

Khi phụ tải của khung BB không được điền đầy hoàn toàn với UP, các byte

độn có thể có mặt. Trong trường hợp này, STUFFI có thể được thiết đặt là '1'.

Khi các byte độn được bao gồm trong khung BB, độ dài của byte độn có thể được kiểm tra qua byte thứ nhất của phụ tải khung BB.

Ví dụ, khi byte thứ nhất giá trị của phụ tải khung BB là 0xFF, một byte độn (byte độn là 1 byte) có thể được bao gồm trong phụ tải khung BB.

Khi các giá trị của byte thứ nhất và byte thứ hai của phụ tải khung BB là 0xFE và 0xFF, một cách tương ứng, hai byte độn có thể được bao gồm trong phụ tải khung BB.

Ở đây, khi các byte độn là hai hoặc lớn hơn (kích thước của byte độn là 2 byte hoặc lớn hơn), giá trị byte thứ nhất và thứ hai được thiết đặt là MSB và LSB, tương ứng để truyền tín hiệu độ dài của byte độn.

Trong bảng trên Fig.40a, 'N' thể hiện độ dài tổng của byte độn.

Khi giá trị của 'N' là 1 byte, độ dài của trường chỉ báo độ dài tổng của byte độn có thể là 1 byte. Trong trường hợp này, giá trị trường có thể được thiết đặt là 0xFF.

Ở đây, trường chỉ báo độ dài tổng của byte độn có thể được thể hiện như trường độ dài byte độn.

Khi giá trị của 'N' là 2 byte, độ dài của trường độ dài của byte độn có thể là 2 byte.

Trong trường hợp này, giá trị trường độ dài byte độn có thể được thiết đặt là 0xFE và 0xFF.

Khi giá trị của 'N' là '3 hoặc lớn hơn', ví dụ, ngay cả khi N có giá trị giữa 3 và 65278, độ dài của trường độ dài byte độn có thể là 2 byte.

Trong trường hợp này, trường độ dài byte độn có thể được cấu thành bởi MSB và LSB.

Nghĩa là, trường độ dài byte độn 2 byte có thể truyền tín hiệu độ dài tổng của byte độn.

Như được minh họa trên Fig.40, các byte độn bổ sung có thể có mặt sau MSB

và LSB. Nghĩa là, vì độ dài byte độn tổng là  $N$  và các độ dài của MSB và LSB là 2 byte, độ dài của byte độn sau là  $N - 2$  byte.

Fig.41 minh họa ví dụ khác nữa về cấu trúc khung BB theo kỹ thuật đã biết.

Như được minh họa trên Fig.41, ký hiệu chỉ báo 2 bit có thể được sử dụng để chỉ báo trạng thái của byte độn. Ký hiệu chỉ báo có thể được thể hiện như ký hiệu chỉ báo đệm (Padding Indicator, viết tắt là PADI).

Khi byte độn, nghĩa là, phần đệm không được bao gồm trong phụ tải BBF (theo cách khác, trường dữ liệu hoặc khung FEC), PADI có thể được thiết đặt là '00'.

Trong khung BB thứ nhất được minh họa trên Fig.41b, PADI có thể được thiết đặt là '00' và có thể được kiểm tra rằng không có phần đệm trong phụ tải BBF.

Khi PADI là '01', có thể được thể hiện rằng độ dài của phần đệm được bao gồm trong phụ tải BBF là 1 byte.

Trong khung BB thứ hai được minh họa trên Fig.41b, PADI có thể được thiết đặt là '01' và có thể được kiểm tra rằng độ dài của phần đệm là 1 byte. 'P' được thể hiện thể hiện byte đệm.

Khi PADI là '10', có thể được thể hiện rằng các byte đệm là hai hoặc nhiều hơn.

Trong trường hợp này, trường đệm có thể truyền tín hiệu độ dài của phần đệm nhờ sử dụng MSB và LSB.

Trong khung BB thứ ba được minh họa trên Fig.41b, có thể nhận thấy rằng giá trị PADI được thiết đặt là '10' và các byte thứ nhất và thứ hai của trường đệm được phân phối đến MSB và LSB, một cách tương ứng.

Phần đệm bổ sung được đánh dấu 'P' có thể có mặt sau MSB và LSB.

Fig.42 minh họa một ví dụ về cấu trúc khung BB được đề xuất theo sáng chế.

Sáng chế đề xuất sơ đồ dưới đây đối với khung BB và cấu hình của phần đầu khung BB.

Khung BB có thể được tạo cấu hình để bao gồm ít nhất một trong số phần đầu khung BB, trường độn, và phụ tải.

Fig.42 minh họa một ví dụ về cấu trúc khung BB trong đó trường độn được đặt trước phụ tải.

Trường độn có thể được đặt sau phụ tải theo một số phương án và điều sẽ được mô tả chi tiết trên Fig.44 và Fig.45.

Trường độn và phụ tải được kết hợp cần được gọi là phụ tải khung BB (theo cách khác, trường dữ liệu khung BB hoặc khung FEC).

Phần đầu khung BB có thể mô tả định dạng của phụ tải, nghĩa là, trường dữ liệu.

Ngoài ra thông tin được kết hợp với gói trống được xóa (DNP) hoặc bộ đồng bộ dòng đầu vào (ISSY) có thể được chèn bổ sung trước trường độn.

Như được nêu trên, phụ tải có thể có nghĩa là trường dữ liệu.

Phần đầu khung BB có thể bao gồm trường STUFFI.

Trường STUFFI có thể dùng làm ký hiệu chỉ báo chỉ báo xem liệu các byte độn có mặt trong khung BB hay không.

Trường STUFFI có thể là 1 bit. Theo một số phương án, vị trí của STUFF1 có thể được thay đổi.

Ví dụ, khi giá trị STUFFI là '0', khung BB không bao gồm trường độn và có thể không bao gồm ngay cả trong trường truyền tín hiệu trong dải.

Khi giá trị trường STUFFI là '1', khung BB có thể bao gồm trường độn hoặc trường truyền tín hiệu trong dải. Nghĩa là, thông tin khác với UP, nghĩa là, phần đệm hoặc trường trong dải có thể có mặt bổ sung trong phụ tải.

Theo một số phương án của sáng chế, các giá trị được thể hiện là '0' và '1' của giá trị STUFFI có thể được chuyển đổi cho nhau.

Trường độn có thể bao gồm ít nhất một trong số phần đầu trường độn và vùng dữ liệu độn.

Vùng dữ liệu độn có thể bao gồm ít nhất một trong số dữ liệu độn và thông tin truyền tín hiệu trong dải.

Phần đầu trường độn có thể là 2 byte theo một số phương án.

Ngoài ra phần đầu trường độn có thể bao gồm ít nhất một trong số STUFF\_ONE (theo cách khác, PAD\_ONE), STUFF\_TYPE (PAD\_TYPE), và STUFF\_LEN (theo cách khác, PAD\_LEN).

Byte thứ nhất được minh họa trên Fig.42 thể hiện byte thứ nhất của trường độn.

Byte thứ hai có thể cũng được bao gồm trong trường độn. Theo một số phương án, hai byte thứ nhất (byte thứ nhất và byte thứ hai) có thể tương ứng với phần đầu trường độn.

Theo một số phương án, byte thứ ba (byte thứ ba) hoặc cao hơn có thể được bao gồm trong vùng dữ liệu độn hoặc phụ tải.

Trường PAD\_ONE có thể được thể hiện như trường STUFF\_ONE theo một số phương án.

Khi STUFFI là '1', STUFF\_ONE có thể được kiểm tra. STUFF\_ONE có thể thể hiện xem liệu độ dài của byte độn có là 1 byte hay không. STUFF\_ONE có thể là MSB 1 bit. Khi STUFF\_ONE là 1, độ dài của byte độn có thể là 1 byte. Trong trường hợp này, STUFF\_LEN\_LSB thể hiện độ dài của byte độn có thể không được sử dụng.

Ngoài ra tất cả các giá trị của STUFF\_LEN\_MSB có thể được thiết đặt là 0. Trong trường hợp này, tất cả các giá trị của STUFF\_LEN\_MSB có thể được thiết đặt là 1. Nghĩa là, theo một số phương án, byte độn 1 byte có thể có giá trị trong số 00000000, 11111111, 10000000, hoặc 01111111.

Khi STUFF\_ONE là 0, độ dài của byte độn có thể lớn hơn 1 byte.

Trong trường hợp này, phần đầu trường độn 2 byte có thể được sử dụng để thể hiện độ dài và loại vùng dữ liệu độn.

Các giá trị của STUFF\_ONE có thể được chuyển đổi giá trị cho nhau phụ thuộc vào nhà thiết kế. Nghĩa là, các giá trị được thể hiện là 1 và 0 có thể được chuyển đổi cho nhau.

STUFF\_ONE (PAD\_ONE) được minh họa có thể được đặt ở bit thứ nhất của

byte thứ nhất. Vị trí này có thể được thay đổi theo một số phương án. STUFF\_ONE có thể được đặt ở phần đầu khung BB theo một số phương án.

Theo một số phương án, một trường là 2 bit, mà dùng như STUFFI và STUFF\_ONE có thể được tạo cấu hình theo một số phương án. Vì mỗi trong số STUFFI và STUFF\_ONE là 1 bit, một trường là 2 bit được tạo cấu hình và các vai trò của STUFFI và STUFF\_ONE có thể được thay thế. Trường có thể được đặt ở phần đầu khung BB hoặc trong trường đệm.

PAD\_LEN có thể được gọi là STUFF\_LEN theo một số phương án. STUFF\_LEN có thể bao gồm ít nhất một trong số STUFF\_LEN\_MSB và STUFF\_LEN\_LSB.

STUFF\_LEN\_MSB và STUFF\_LEN\_LSB có thể lần lượt là các trường 5 và 8 bit.

Các trường STUFF\_LEN\_MSB và STUFF\_LEN\_LSB có thể được sử dụng để thể hiện độ dài tổng của trường đệm. Theo một số phương án, các độ dài của STUFF\_LEN\_MSB và STUFF\_LEN\_LSB được chuyển đổi cho nhau để là 8 bit và 5 bit, một cách tương ứng. Ngoài ra theo một số phương án, các vị trí của cả hai phía có thể cũng được chuyển đổi cho nhau. Theo một số phương án, trường chỉ báo độ dài của phần đệm có thể được đặt trong vùng dữ liệu đệm.

Theo kỹ thuật đã biết, độ dài của phần đệm được thể hiện nhờ sử dụng 2 byte thứ nhất. Tuy nhiên, khi 64K LDPC được sử dụng, độ dài của phần đệm có giá trị lớn nhất 6370 byte (64k, tỷ lệ mã 5/6, mã BCH). Do đó, độ dài của phần đệm có thể được thể hiện đầy đủ bởi 13 bit ( $2^{13} = 8192$  byte).

Do đó, PAD\_LEN được đề xuất theo sáng chế có thể có 13 (5 + 8) bit.

Khi độ dài của phần đệm được thể hiện bởi 13 bit, 2 bit dự trữ trong 2 byte thứ nhất có thể vẫn còn.

Theo sáng chế, phương pháp được đề xuất, mà cấp phát 2 bit dự trữ đến PAD\_TYPE để truyền tín hiệu loại khi vùng đệm được sử dụng cho mục đích khác (chẳng hạn, truyền tín hiệu trong dải).

STUFF\_TYPE có thể được gọi là PAD\_TYPE theo một số phương án.

STUFF\_TYPE là trường 2 bit có thể thể hiện loại dữ liệu độn (theo cách khác, vùng dữ liệu độn) như được nêu trên.

Như được minh họa trên Fig.42, khi giá trị STUFF\_TYPE là '00', vùng dữ liệu độn có thể bao gồm chỉ dữ liệu độn.

Khi giá trị STUFF\_TYPE là '01', thông tin truyền tín hiệu trong dải loại cụ thể có thể được bao gồm trong vùng dữ liệu độn cùng với dữ liệu độn.

Khi giá trị STUFF\_TYPE là '10', thông tin truyền tín hiệu trong dải loại khác có thể được bao gồm trong vùng dữ liệu độn cùng với dữ liệu độn.

Khi giá trị STUFF\_TYPE là '11', cả thông tin truyền tín hiệu trong dải loại cụ thể và loại khác có thể được bao gồm trong vùng dữ liệu độn cùng với dữ liệu độn.

Ở đây, thông tin truyền tín hiệu trong dải loại cụ thể có thể có nghĩa là 'A trong dải' và thông tin truyền tín hiệu trong dải loại khác có thể có nghĩa là 'B trong dải'.

Đó chỉ là một phương án và loại được chỉ báo bởi giá trị STUFF\_TYPE có thể được thay đổi bởi các sơ đồ khác.

Ngoài ra STUFF\_TYPE có thể chỉ báo phụ tải khung BB và cấu hình của phụ tải. Chẳng hạn, STUFF\_TYPE có thể chỉ báo vị trí của gói thứ nhất thông thường mà không bị cắt trong phụ tải.

Như được đề xuất theo sáng chế, khi việc truyền tín hiệu được thực hiện trong trường độn, việc truyền tín hiệu trong dải có thể được chèn vào các khung khác. Ngoài ra trường hợp này có thể được phân biệt với trường hợp trong đó chỉ phần độn được bao gồm mà không truyền tín hiệu trong dải.

STUFF\_TYPE có thể được đặt ở phần đầu khung BB theo một số phương án.

Theo cách khác, như được mô tả trong phương án, STUFF\_TYPE có thể được đặt trong trường độn. Theo một số phương án, độ dài của STUFF\_TYPE có thể được thay đổi.

Các giá trị của STUFF\_TYPE có thể được chuyển đổi giá trị cho nhau phụ thuộc vào nhà thiết kế.

Chẳng hạn, giá trị được biểu diễn là 00 và giá trị được biểu diễn là 11 có thể

được chuyển đổi cho nhau. Ngoài ra giá trị được biểu diễn là 10 và giá trị được biểu diễn là 01 có thể được chuyển đổi cho nhau.

Tất cả dữ liệu độn có thể có giá trị là 0 hoặc 1 theo một số phương án.

Dưới đây, trường hợp #1 đến trường hợp #6 được minh họa trên Fig.42 sẽ được mô tả chi tiết.

(1) Trường hợp #1 minh họa trường hợp mà ở đó dữ liệu độn và truyền tín hiệu trong dải không được bao gồm trong khung BB.

Trong trường hợp này, trường STUFFI có thể được thiết đặt là '0'. Do đó, trong cấu trúc của khung BB, vùng dữ liệu, nghĩa là, phụ tải có thể được đặt tiếp theo phần đầu khung BB.

(2) Trường hợp #2 minh họa trường hợp mà ở đó trường độn là 1 byte có trong khung BB và truyền tín hiệu trong dải không tồn tại.

Trong trường hợp này, trường STUFFI có thể được thiết đặt là '1'. Nghĩa là, khung BB bao gồm trường độn và trường độn có thể có kích thước là 1 byte.

Ở đây, bit thứ nhất của trường độn thể hiện trường STUFF\_ONE, và có giá trị là '1' bởi vì kích thước của trường độn là 1 byte.

7 bit còn lại của trường độn có thể có giá trị là 1111111.

Do đó, trường độn là 1 byte có thể được thể hiện là 11111111.

(3) Trường hợp #3 minh họa trường hợp mà ở đó trường độn lớn hơn 1 byte có trong khung BB và truyền tín hiệu trong dải không tồn tại.

Nghĩa là, trường độn có thể là 2 byte hoặc lớn hơn 2 byte.

Vì trường độn tồn tại, trường STUFFI có thể được thiết đặt là '1'.

Trường độn có thể có phần đầu trường độn là 2 byte. Bit thứ nhất của byte thứ nhất của phần đầu trường độn tương ứng với trường STUFF\_ONE.

Trường STUFF\_ONE có thể được thiết đặt là giá trị của '0' bởi vì kích thước của trường độn lớn hơn 1 byte.

Bit thứ hai và thứ ba của byte thứ nhất của phần đầu trường độn tương ứng

với trường STUFF\_TYPE.

Vì chỉ dữ liệu độn có trong vùng dữ liệu độn của khung BB, như được nêu trên, STUFF\_TYPE có thể có giá trị là 00.

Trên hình vẽ, như phương án ví dụ khác, trường hợp mà ở đó STUFF\_TYPE có giá trị là 11 được minh họa.

Nghĩa là, trường hợp này là trường hợp mà ở đó chỉ dữ liệu độn có trong vùng dữ liệu độn của khung BB, và trường STUF\_TYPE có thể được chỉ báo với giá trị là 11.

Sau đó, STUFF\_LEN\_MSB và STUFF\_LEN\_LSB của phần đầu trường độn có thể có thông tin độ dài của trường độn. Như được nêu trên, độ dài của trường độn có thể được thể hiện nhờ sử dụng tổng số 13 bit. Sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB, vùng dữ liệu độn có thể được đặt vào. Trong trường hợp này, chỉ dữ liệu độn có thể được đặt trong vùng dữ liệu độn.

(4) Trường hợp #4 minh họa trường hợp mà ở đó trường độn lớn hơn 1 byte có trong khung BB và truyền tín hiệu trong dải tồn tại.

Trong trường hợp này, dữ liệu độn và truyền tín hiệu A trong dải có thể tồn tại trong vùng dữ liệu độn của khung BB.

Truyền tín hiệu A trong dải có thể có nghĩa là loại truyền tín hiệu trong dải cụ thể được nêu trên. Trong trường hợp này, bởi vì trường độn tồn tại, STUFFI có thể có giá trị là 1.

Bit thứ nhất của byte thứ nhất của phần đầu trường độn là trường STUFF\_ONE và có thể có giá trị là '0' bởi vì kích thước của trường độn lớn hơn 1 byte.

Các bit thứ hai và thứ ba của byte thứ nhất của phần đầu trường độn có thể là trường STUFF\_TYPE nêu trên.

Vì chỉ truyền tín hiệu A trong dải có trong vùng dữ liệu độn của khung BB, như được nêu trên, STUFF\_TYPE có thể có giá trị là 10. Theo một phương án ví dụ, giá trị có thể cũng là 01.

Tiếp theo, STUFF\_LEN\_MSB và STUFF\_LEN\_LSB của phần đầu trường

độn có thể có thông tin độ dài của trường độn. Như được nêu trên, độ dài của trường độn có thể được thể hiện nhờ sử dụng tổng số 13 bit. Sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB, vùng dữ liệu độn có thể được đặt. Trong trường hợp này, truyền tín hiệu A trong dải ngoài dữ liệu độn có thể tồn tại trong vùng dữ liệu độn của khung BB.

(5) Trường hợp #5 minh họa trường hợp mà ở đó trường độn lớn hơn 1 byte có trong khung BB và truyền tín hiệu B trong dải tồn tại.

Trong trường hợp này, dữ liệu độn và truyền tín hiệu B trong dải có thể tồn tại trong vùng dữ liệu độn của khung BB.

Truyền tín hiệu B trong dải có thể có nghĩa là loại truyền tín hiệu trong dải khác được nêu trên. Trong trường hợp này, bởi vì trường độn tồn tại, STUFFI có thể có giá trị là 1.

Bit thứ nhất của byte thứ nhất của phần đầu trường độn là trường STUFF\_ONE và có thể có giá trị là '0' bởi vì kích thước của trường độn lớn hơn 1 byte.

Các bit thứ hai và thứ ba của byte thứ nhất của phần đầu trường độn có thể là trường STUFF\_TYPE nêu trên. Vì chỉ truyền tín hiệu B trong dải có trong vùng dữ liệu độn của khung BB, như được nêu trên, STUFF\_TYPE có thể có giá trị là 01. Theo một phương án ví dụ, giá trị có thể cũng là 10.

Tiếp theo, STUFF\_LEN\_MSB và STUFF\_LEN\_LSB của phần đầu trường độn có thể có thông tin độ dài của trường độn. Như được nêu trên, độ dài của trường độn có thể được thể hiện nhờ sử dụng tổng số 13 bit. Sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB, vùng dữ liệu độn có thể được đặt. Trong trường hợp này, truyền tín hiệu B trong dải ngoài dữ liệu độn có thể tồn tại trong vùng dữ liệu độn của khung BB.

(6) Trường hợp #6 minh họa trường hợp mà ở đó trường độn lớn hơn 1 byte có trong khung BB và truyền tín hiệu A và B trong dải tồn tại.

Trong trường hợp này, tất cả của dữ liệu độn và truyền tín hiệu A và B trong dải có thể tồn tại trong vùng dữ liệu độn của khung BB.

Trong trường hợp này, STUFFI có thể có giá trị là '1'. Bit thứ nhất của byte thứ nhất của phần đầu trường độn là trường STUFF\_ONE và có thể có giá trị là '0' bởi vì kích thước của trường độn lớn hơn 1 byte. Các bit thứ hai và thứ ba của byte thứ nhất của phần đầu trường độn có thể là trường STUFF\_TYPE nêu trên. Vì truyền tín hiệu A và B trong dải tồn tại trong vùng dữ liệu độn của khung BB, như được nêu trên, STUFF\_TYPE có thể có giá trị là 11.

Trên hình vẽ, như phương án ví dụ khác, trường hợp mà ở đó STUFF\_TYPE có giá trị là 00 được minh họa. Nghĩa là, trường hợp này là trường hợp mà ở đó tất cả truyền tín hiệu A và B trong dải tồn tại trong vùng dữ liệu độn của khung BB, và trường STUFF\_TYPE có thể được chỉ báo với giá trị là 00.

Tiếp theo, STUFF\_LEN\_MSB và STUFF\_LEN\_LSB của phần đầu trường độn có thể có thông tin độ dài của trường độn. Như được nêu trên, độ dài của trường độn có thể được thể hiện nhờ sử dụng tổng số 13 bit.

Sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB, vùng dữ liệu độn có thể được đặt. Trong trường hợp này, truyền tín hiệu A và B trong dải ngoài dữ liệu độn có thể tồn tại trong vùng dữ liệu độn.

Fig.43 là sơ đồ minh họa ví dụ khác về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.43A có thể minh họa khung BB trong trường hợp mà ở đó chỉ dữ liệu tồn tại mà không có phần đệm, nghĩa là, không có dữ liệu độn.

STUFFI của phần đầu khung BB có thể có giá trị là 0. Phụ tải có thể được đặt ngay sau phần đầu khung BB mà không có trường độn. Trường hợp này có thể tương ứng với trường hợp #1 trên Fig.42.

Fig.43B có thể là trường hợp có phần đệm là 1 byte.

Trong trường hợp này, STUFFI của phần đầu khung BB có thể có giá trị là 1. Bit thứ nhất của byte thứ nhất có thể có giá trị là 1 như STUFF\_ONE. Điều này có thể có nghĩa là phần đệm là 1 byte. Trên Fig.43, mỗi bit của phần đệm có thể có giá trị là 11111111 (0xFF). Theo cách khác, theo một phương án ví dụ, mỗi bit có thể có giá trị là 10000000. Trường hợp có thể tương ứng với trường hợp #2 trên Fig.42.

Fig.43C có thể là trường hợp có phần đệm của n byte.

Trong trường hợp này, STUFFI của phần đầu khung BB có thể có giá trị là 1. Ngoài ra, STUFF\_ONE có thể có giá trị là 0. STUFF\_TYPE có thể chỉ báo rằng chỉ dữ liệu đệm được sử dụng mà không truyền tín hiệu trong dải.

Nghĩa là, theo một phương án ví dụ, STUFF\_TYPE có thể có giá trị là 00.

Tiếp theo, 13 bit còn lại có thể chỉ báo rằng độ dài của trường đệm là n byte. 13 bit có thể là STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Dữ liệu đệm có thể được đặt sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Trường hợp có thể tương ứng với trường hợp mà ở đó trường đệm là 3 byte hoặc lớn hơn trong trường hợp #3 trên Fig.42.

Fig.43D có thể là trường hợp có đệm của n byte ngoài truyền tín hiệu A trong dải.

Trong trường hợp này, STUFFI của phần đầu khung BB có thể có giá trị là 1. Ngoài ra STUFF\_ONE có thể có giá trị là 0. STUFF\_TYPE có thể chỉ báo rằng truyền tín hiệu A trong dải được sử dụng.

Nghĩa là, theo một phương án ví dụ, STUFF\_TYPE có thể có giá trị là 01. Giá trị của STUFF\_TYPE có thể được thay đổi như được nêu trên. Tiếp theo, 13 bit còn lại có thể chỉ báo rằng độ dài của trường đệm là n byte. 13 bit có thể là STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Truyền tín hiệu A trong dải có thể được đặt sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Trường hợp có thể tương ứng với trường hợp #4 trên Fig.42.

Fig.43E có thể là trường hợp có phần đệm của n byte ngoài truyền tín hiệu B trong dải.

Trong trường hợp này, STUFFI của phần đầu khung BB có thể có giá trị là 1. Ngoài ra STUFF\_ONE có thể có giá trị là 0. STUFF\_TYPE có thể chỉ báo rằng việc truyền tín hiệu B trong dải được sử dụng.

Nghĩa là, theo một phương án ví dụ, STUFF\_TYPE có thể có giá trị là 10. Giá trị của STUFF\_TYPE có thể được thay đổi như được nêu trên.

Tiếp theo, 13 bit còn lại có thể chỉ báo rằng độ dài của trường đệm là n byte.

13 bit có thể là STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Truyền tín hiệu B trong dải có thể được đặt sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Trường hợp có thể tương ứng với trường hợp #5 trên Fig.42.

Fig.43F có thể là trường hợp có phần đệm của n byte ngoài truyền tín hiệu A và B trong dải.

Trong trường hợp này, STUFFI của phần đầu khung BB có thể có giá trị là 1. Ngoài ra STUFF\_ONE có thể có giá trị là 0. STUFF\_TYPE có thể chỉ báo rằng các truyền tín hiệu A và B trong dải được sử dụng.

Nghĩa là, theo một phương án ví dụ, STUFF\_TYPE có thể có giá trị là 11. Giá trị của STUFF\_TYPE có thể được thay đổi như được nêu trên. Tiếp theo, 13 bit còn lại có thể chỉ báo rằng độ dài của trường độn là n byte. 13 bit có thể là STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Các truyền tín hiệu A và B trong dải có thể được đặt sau STUFF\_LEN\_MSB và STUFF\_LEN\_LSB. Trường hợp có thể tương ứng với trường hợp #6 trên Fig.42.

Fig.44 là sơ đồ minh họa ví dụ khác về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.44 minh họa một ví dụ về cấu trúc khung BB trong trường hợp mà ở đó trường độn được đặt ở cuối của khung BB (tiếp theo phụ tải).

Khung BB bao gồm phần đầu BBF và phụ tải khung BB.

Phần đầu BBF được chèn vào trước phụ tải khung BB để thể hiện định dạng của trường dữ liệu BBF.

Phần đầu BBF có thể có độ dài cố định là 2 byte.

Phần đầu BBF bao gồm trường STUFFI tương ứng với ký hiệu chỉ báo thể hiện xem liệu độn và đệm có trong khung BB hay không. Trường STUFFI có thể được thể hiện bởi 1 bit.

Phụ tải khung BB có thể bao gồm trường độn và phụ tải.

Trường độn được bao gồm trong trường hợp mà ở đó tất cả các gói người dùng (các UP) không được điền đầy trong phụ tải khung BB.

Như một ví dụ, khi trường STUFFI được thiết đặt là '1', phụ tải khung BB có thể bao gồm trường độn.

Phụ tải có nghĩa là vùng mà ở đó UP được bao gồm.

Trường độn có thể được tạo cấu hình bởi trường phần đầu độn (theo cách khác, phần đầu trường độn) và (vùng) dữ liệu độn.

Vùng dữ liệu độn có thể được thể hiện bởi trường dữ liệu độn hoặc dữ liệu độn.

Dữ liệu độn, thông tin truyền tín hiệu trong dải, và tương tự có thể được bao gồm trong vùng dữ liệu độn.

Trường phần đầu độn có thể bao gồm trường STUFF\_ONE, trường STUFF\_TYPE, và trường STUFF\_LEN.

Trường STUFF\_LEN thể hiện độ dài của toàn bộ trường độn bao gồm trường phần đầu độn và có thể bao gồm trường STUFF\_LEN\_MSB và trường STUFF\_LEN\_LSB. Trường STUFF\_LEN được thể hiện bởi 13 bit.

Trường STUFF\_ONE có nghĩa là trường là 1 bit thể hiện xem liệu độ dài của trường độn có là 1 byte hay không.

Như một ví dụ, khi trường STUFF\_ONE được thiết đặt là '1', độ dài của trường độn là 1 byte. Trong trường hợp này, trường STUFF\_LEN\_LSB không được bao gồm trong trường độn, nghĩa là, trường STUFF\_LEN.

Khi trường STUFF\_ONE được thiết đặt là '0', độ dài của trường độn lớn hơn 1 byte. Trong trường hợp này, 2 byte của phần đầu độn được sử dụng để thể hiện loại và độ dài của dữ liệu độn.

Nghĩa là, trường STUFF\_TYPE thể hiện loại dữ liệu độn và có thể được thể hiện bởi 2 bit.

Bảng 34 dưới đây minh họa một ví dụ về trường STUFF\_TYPE trên Fig.44.

Bảng 34

|            |                  |
|------------|------------------|
| STUFF_TYPE | Loại dữ liệu độn |
|------------|------------------|

|    |                                                            |
|----|------------------------------------------------------------|
| 00 | Chỉ dữ liệu độn                                            |
| 01 | A trong dải được sử dụng với dữ liệu độn                   |
| 10 | B trong dải được sử dụng với dữ liệu độn                   |
| 11 | Cả A trong dải và B trong dải được sử dụng với dữ liệu độn |

Dựa vào bảng 34 và Fig.44, khi trường STUFF\_TYPE được thiết đặt là (1) ‘00’, vùng dữ liệu độn có thể được sử dụng làm chỉ dữ liệu độn, khi trường STUFF\_TYPE được thiết đặt là (2) ‘01’, vùng dữ liệu độn có thể được sử dụng làm thông tin truyền tín hiệu A trong dải và dữ liệu độn, khi trường STUFF\_TYPE được thiết đặt là (3) ‘10’, vùng dữ liệu độn có thể được sử dụng làm truyền tín hiệu B trong dải thông tin và dữ liệu độn, và khi trường STUFF\_TYPE được thiết đặt là (4) ‘11’, vùng dữ liệu độn có thể được sử dụng làm truyền tín hiệu B trong dải thông tin và dữ liệu độn.

Trong bảng 34, A trong dải có thể là ISSY trong dải, và B trong dải có thể là PLS trong dải.

Trường STUFF\_LEN\_MSB thể hiện giá trị bit trọng số lớn nhất (Most Significant Bit, viết tắt là MSB) của toàn bộ độ dài trường độn bao gồm trường phần đầu độn và được thể hiện bởi 5 bit.

Như một ví dụ, khi trường STUFF\_ONE được thiết đặt là ‘1’, trường STUFF\_LEN\_MSB có thể được thể hiện là ‘11111’. Theo cách khác, trường STUFF\_LEN\_MSB có thể được thể hiện là ‘00000’.

Trường STUFF\_LEN\_LSB thể hiện bit trọng số thấp nhất (Least Significant Bit, viết tắt là LSB) giá trị của toàn bộ độ dài trường độn và được thể hiện bởi 8 bit.

Trường dữ liệu độn có thể bao gồm (các) trường truyền tín hiệu trong dải và/hoặc độn.

Ở đây, ‘truyền tín hiệu trong dải và/hoặc độn’ có nghĩa là độn, truyền tín hiệu trong dải hoặc độn, và truyền tín hiệu trong dải.

Nghĩa là, cách viết ‘A và/hoặc B’ có thể có ý nghĩa giống như ít nhất một trong số A hoặc B.

Dựa vào Fig.44, bit thứ tám của byte thứ N của trường độn thể hiện trường

STUFF\_ONE, các bit thứ sáu và thứ bảy của byte thứ N thể hiện trường STUFF\_TYPE, các bit từ thứ nhất đến thứ năm của byte thứ N của trường độ dài thể hiện trường STUFF\_LEN\_LSB, và byte thứ N-1 của trường độ dài thể hiện trường STUFF\_LEN\_MSB.

Ngoài ra dữ liệu UP, dữ liệu độ dài, dữ liệu A trong dải, dữ liệu B trong dải, hoặc dữ liệu A trong dải và B dữ liệu từ byte thứ N-2 của trường độ dài có thể được thể hiện.

Phần mô tả chi tiết hơn đối với trường hợp #1 đến trường hợp # 6 trên Fig.44 sẽ dựa vào phần mô tả của trường hợp #1 đến trường hợp # 6 trên Fig.42 mà tương ứng với mỗi trường hợp trên Fig.44.

Cấu trúc khung trên Fig.44 có thể thực hiện chức năng giống như cấu trúc khung trên Fig.42.

Cấu trúc khung BB tương tự được minh họa trên Fig.44, khi trường độ dài được đặt ở thời điểm kết thúc của khung BB, thiết bị thu có thể thu trực tiếp gói người dùng (UP) mà không kiểm tra phần độ dài, và kết quả là, thời gian truy cập UP ngắn hơn so với thời gian truy cập trong cấu trúc khung BB được minh họa trên Fig.42.

Fig.45 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Fig.45 minh họa các cấu trúc khung BB khác nhau trong trường hợp mà ở đó trường độ dài được đặt ở cuối của khung BB (theo cách khác, được đặt tiếp theo phụ tải và khung FEC).

Vì cấu trúc khung trên Fig.45 khác với cấu trúc khung trên Fig.43 trong chỉ vị trí của trường độ dài nhưng giống như cấu trúc khung trên Fig.43 trong tất cả các phần khác, phần mô tả chi tiết trên Fig.41 dựa vào Fig.43.

Fig.46 là sơ đồ minh họa sự so sánh của kết quả tính toán chi phí truyền khung BB trong các cấu trúc khung BB khác nhau.

Đồ thị được thể hiện bởi DVB-T2 có thể là đồ thị chi phí theo kỹ thuật đã biết được nêu trên. DVB-T2 có thể có nghĩa là tiêu chuẩn liên quan đến hệ thống phát rộng truyền hình mặt đất để phát rộng video số (DVB). DVB-T2 có thể có nghĩa là tiêu chuẩn liên quan đến hệ thống phát rộng mặt đất thế hệ tiếp theo ở châu Âu. Đồ

thị được thể hiện bởi DVB-T2 có thể là đồ thị thu được bằng cách tính toán chi phí trong khung BB theo kỹ thuật tiêu chuẩn này.

Đồ thị được thể hiện bởi MH có thể là đồ thị chi phí khác theo kỹ thuật đã biết được nêu trên. MH có thể có nghĩa là tiêu chuẩn liên quan đến hệ thống DTV cầm tay/di động của hiệp hội tiêu dùng điện tử (Consumer Electronics Association, viết tắt là CEA). MH có thể có nghĩa là tiêu chuẩn liên quan đến thiết bị cầm tay/di động ở Bắc Mỹ. Đồ thị được thể hiện bởi MH có thể là đồ thị thu được bằng cách tính toán chi phí trong khung BB theo kỹ thuật tiêu chuẩn này.

Đồ thị được thể hiện bởi SS&SN có thể là đồ thị chi phí khác nữa theo kỹ thuật đã biết được nêu trên. SS&SN có thể có nghĩa là một trong số kỹ thuật đã biết. Đồ thị thu được bằng cách tính toán chi phí khi khung BB và phần đầu khung BB được tạo cấu hình bởi phương pháp được đề xuất theo kỹ thuật đã biết được minh họa như đồ thị được thể hiện bởi SS&SN.

Bảng 35 dưới đây là bảng minh họa kết quả tính toán chi phí khi truyền của mỗi khung BB.

Bảng 35

|                | FE | 64k        |            |            |            |            |            | 16k        |            |            |            |            |            |
|----------------|----|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
|                | CR | 5/6        | 4/5        | 3/4        | 2/3        | 3/5        | 1/2        | 5/6        | 4/5        | 3/4        | 2/3        | 3/5        | 1/2        |
|                | Kb | 538        | 516        | 484        | 430        | 386        | 322        | 131        | 126        | 118        | 108        | 972        | 720        |
|                | ch | 40         | 48         | 08         | 40         | 88         | 08         | 52         | 00         | 80         | 00         | 0          | 0          |
| DV<br>B-<br>T2 |    | 0,02<br>97 | 0,0<br>310 | 0,03<br>31 | 0,0<br>372 | 0,04<br>14 | 0,04<br>97 | 0,12<br>17 | 0,12<br>70 | 0,13<br>47 | 0,14<br>81 | 0,16<br>46 | 0,22<br>22 |
| MH             |    | 0,00<br>19 | 0,0<br>019 | 0,00<br>21 | 0,0<br>023 | 0,00<br>26 | 0,00<br>31 | 0,00<br>76 | 0,00<br>79 | 0,00<br>84 | 0,00<br>93 | 0,01<br>03 | 0,01<br>39 |
| SS&<br>SN      |    | 0,00<br>37 | 0,0<br>039 | 0,00<br>41 | 0,0<br>046 | 0,00<br>52 | 0,00<br>62 | 0,01<br>52 | 0,01<br>59 | 0,01<br>68 | 0,01<br>85 | 0,02<br>06 | 0,02<br>78 |
| LG             |    | 0,00<br>19 | 0,0<br>019 | 0,00<br>21 | 0,0<br>023 | 0,00<br>26 | 0,00<br>31 | 0,00<br>76 | 0,00<br>79 | 0,00<br>84 | 0,00<br>93 | 0,01<br>03 | 0,01<br>39 |

Chi phí có thể có nghĩa là chi phí của trường thể hiện độ dài của trường dữ

liệu.

Theo kỹ thuật đã biết, vì trường là 2 byte được sử dụng cho mỗi khung BB, chi phí có thể lớn nhất là 0,22%.

Trong kỹ thuật đã biết khác, vì chỉ trường là 1 bit được sử dụng, chi phí có thể có giá trị lớn nhất 0,0139 %. Chi phí này có thể là thấp nhất.

Theo kỹ thuật đã biết khác nữa, trường là 2 bit có thể được sử dụng. Trong trường hợp này, chi phí có thể lớn hơn hai lần so với chi phí của kỹ thuật đã biết khác.

Đồ thị được thể hiện bởi LG có thể là đồ thị chi phí theo sáng chế. Theo sáng chế, chỉ trường gồm 1 bit có thể được sử dụng để truyền tín hiệu của trường đơn. Do đó, chi phí có thể là nhỏ nhất. Ngoài ra có lợi thế trong đó trường dư là 2 bit được chuẩn bị bổ sung để được sử dụng để chỉ báo loại truyền tín hiệu trong dải và tương tự. Sáng chế có thể hỗ trợ cấu trúc mà có thể sử dụng được vào việc sử dụng khác, chẳng hạn như thể hiện cấu hình của khung BB, nhờ sử dụng trường dư.

Fig.47 minh họa một ví dụ về cấu trúc khung BB theo kỹ thuật đã biết.

Như được minh họa trên Fig.47, khung BB bao gồm phần đầu, phần đầu tùy chọn, và dữ liệu phụ tải.

Phần đầu bao gồm trường ký hiệu chỉ báo chế độ con trỏ bắt đầu gói (Package Start Pointer Mode Indicator, viết tắt là PSPMI), trường ký hiệu chỉ báo đệm (PADI), và trường các bit trọng số thấp con trỏ bắt đầu gói (Package Start Pointer Low Significant Bit, viết tắt là PKTSPTR\_LSB).

Trường PSPMI có nghĩa là trường cờ có kích thước là 1 bit biểu diễn xem liệu trường con trỏ bắt đầu gói (PKTSPTR) là chế độ ngắn hay chế độ dài.

Trường PKTSPTR có thể là khái niệm giống như trường SYNCD.

Nghĩa là, trường PSPMI có nghĩa là cờ thể hiện xem liệu độ dài của trường PKTSPTR là nhỏ hay lớn.

Trường PKTSPTR\_LSB thể hiện 5 bit LSB của trường PKTSPTR là 13 bit.

Phần đầu tùy chọn có thể bao gồm trường các bit trọng số lớn nhất con trỏ bắt

đầu gói (Package Start Pointer Most Significant Bit, viết tắt là PKTSPTR\_MSB) và trường đệm.

Trường PKTSPTR\_MSB thể hiện 8 bit MSB của trường PKTSPTR là 13 bit.

Ngoài ra trường đệm có thể bao gồm trường độ dài dữ liệu đệm (PADL) và trường PADDING\_DATA.

Trường PADL thể hiện độ dài của trường dữ liệu đệm và có kích thước là 15 bit.

Trường PADDING\_DATA có độ dài thay đổi và thể hiện thông tin đệm.

Như được minh họa trên Fig.47, cấu trúc khung BB tính toán độ dài của trường dữ liệu trong thiết bị thu bằng cách truyền độ dài của trường PADDING\_DATA mà không sử dụng thông tin (chẳng hạn, DFL) biểu diễn độ dài trường dữ liệu, để biểu diễn độ dài của (phụ tải) trường dữ liệu lớn nhất là 13 byte.

Ở đây, độ dài của trường đệm tương ứng với kích thước dữ liệu phụ tải của độ dài khung BB của trường dữ liệu.

Khi trường đệm không tồn tại trong khung BB, độ dài trường dữ liệu (DFL) được tính toán nhờ sử dụng kích thước khung BB.

Khi trường đệm có trong khung BB, PADI gồm 2 bit được bao gồm trong phần đầu khung BB để chỉ báo độ dài phần đệm.

Hiệu quả hơn là, để truyền khung dài gốc (BBF) đến khối FEC, nghĩa là, để làm giảm chi phí của việc truyền của phần đầu khung BB, trường PKTSPRT được chia thành PKTSPTR\_LSB và PKTSPTR\_MSB để thao tác.

Nghĩa là, trường PKTSPTR có thể hỗ trợ lên tới kích thước là 2 byte, nhưng khi độ dài của trường PKTSPTR nhỏ ( $\leq 31$  byte), chỉ PKTSPTR\_LSB có thể được sử dụng, và kết quả là, kích thước truyền của trường PKTSPTR có thể được rút ngắn xuống 1 byte.

Tuy nhiên, vì độ dài của PKTSPTR\_LSB nhỏ đến 5 bit, chỉ khi kích thước của trường PKTSPTR là 31 byte hoặc nhỏ hơn, có bất lợi là trong đó phần đầu BBF là 1 byte có thể được tạo cấu hình.

Như được mô tả trên Fig.39, khung BB hiện thời được sử dụng bằng cách cấp phát DFL đến mỗi phần đầu khung BB để chỉ báo độ dài của trường dữ liệu của khung BB đến thiết bị thu (theo cách khác, thiết bị đầu cuối thu, và kết quả là, chi phí khi truyền khung BB đến khối FEC là lớn).

Do đó, cấu trúc khung BB mới để nâng cao hiệu suất truyền của phần đầu khung BB và bổ sung chức năng mới để kiểm tra lỗi sẽ được mô tả chi tiết.

Nghĩa là, sáng chế này đề xuất phương pháp làm giảm toàn bộ kích thước của phần đầu khung BB bằng cách điều khiển kích thước của trường SYNCED được bao gồm trong phần đầu khung BB, phương pháp thực hiện kiểm tra lỗi nhờ sử dụng 1 bit dư trong phần đầu khung BB, và tương tự.

Dưới đây, các phương pháp và cấu trúc khung BB được đề xuất theo sáng chế này hoạt động trong khối chèn phần đầu khung BB của thiết bị truyền và khối tách phần không hoạt động trong thiết bị thu.

Fig.48 là sơ đồ minh họa một ví dụ về cấu trúc khung BB được đề xuất theo sáng chế.

Dòng đầu vào trên Fig.48A tạo nên cấu trúc khung BB trên Fig.48B qua môđun thích ứng chế độ của môđun định dạng đầu vào.

Như được minh họa trên Fig.48, dòng đầu vào bao gồm các gói được cắt hoặc được ánh xạ đến phụ tải qua môđun thích ứng chế độ, và phần đầu bao gồm thông tin trên phụ tải được bổ sung sau phụ tải.

Phụ tải có thể được thể hiện bởi trường dữ liệu khung BB.

Phần đầu có thể bao gồm ít nhất một trong số trường OPTIONI, trường STUFFI, trường SYNCED\_LSB, trường SYNCED\_MSB, trường Checksum (trường kiểm tra tổng), và trường độn.

Như được nêu trên, trường độn có thể bao gồm trường phần đầu độn và trường byte độn.

Trường byte độn có thể được thể hiện bởi trường dữ liệu độn hoặc vùng dữ liệu độn.

Phần đầu khung BB bao gồm trường OPTIONI, trường STUFFI, và trường

SYNCD\_LSB có thể được xác định, và phần đầu tùy chọn bao gồm trường SYNCD\_MSB và trường Checksum có thể được xác định.

Fig.48 minh họa rằng phần đầu khung BB và phần đầu tùy chọn được xác định.

Ngoài ra trường độn có thể được bao gồm trong phần đầu hoặc có thể không được bao gồm trong phần đầu.

Khi trường độn không được bao gồm trong phần đầu, phụ tải khung BB có thể được tạo cấu hình cùng với phụ tải.

Trường độn có thể được đặt trước phụ tải (Fig.48) hoặc sau phụ tải.

Trường SYNCD có thể thể hiện khoảng cách từ điểm bắt đầu của trường dữ liệu đến điểm bắt đầu của UP được truyền thứ nhất bắt đầu trong trường dữ liệu.

Ở đây, trường SYNCD có thể được chia thành trường SYNCD\_LSB và trường SYNCD\_MSB và có kích thước là 13 bit.

Trường SYNCD\_LSB có kích thước là 6 bit làm giá trị thể hiện LSB của SYNCD và có thể thể hiện SYNCD lớn nhất là 63 byte.

Như được minh họa trên Fig.48, khi phần đầu được chia thành phần đầu khung BB và phần đầu tùy chọn, trường SYNCD\_LSB có thể được bao gồm trong phần đầu khung BB.

Ngoài ra trường SYNCD\_MSB có kích thước là 7 bit làm giá trị thể hiện MSB của SYNCD.

Như được minh họa trên Fig.48, khi phần đầu được chia thành phần đầu khung BB và phần đầu tùy chọn, trường SYNCD\_MSB có thể được bao gồm trong phần đầu tùy chọn.

Việc sử dụng trường SYNCD\_MSB được xác định bởi trường OPTIONI.

Trường OPTIONI thể hiện xem liệu vị trí của gói mà bắt đầu mới trong số các gói được truyền qua phụ tải được thể hiện bởi SYNCD\_LSB là 6 bit.

Như một ví dụ, khi trường OPTIONI được thiết đặt là '0', trường OPTIONI thể hiện rằng vị trí của gói mà bắt đầu mới trong số các gói được truyền qua phụ tải

có thể được thể hiện bởi SYNCD\_LSB là 6 bit.

Khi trường OPTIONI được thiết đặt là '1', trường OPTIONI thể hiện rằng vị trí của gói mà bắt đầu mới trong số các gói được truyền qua phụ tải có thể không được thể hiện bởi SYNCD\_LSB là 6 bit.

Do đó, khi trường OPTIONI được thiết đặt là '1', trường OPTIONI cần thể hiện vị trí của gói mà bắt đầu mới trong phụ tải nhờ sử dụng trường SYNCD\_LSB là 6 bit của trường SYNCD\_MSB là 7 bit.

Ở đây, khi trường SYNCD\_MSB được bao gồm trong phần đầu tùy chọn, phần đầu tùy chọn được bao gồm trong khung BB.

Trường STUFFI có kích thước là 1 bit và có nghĩa là ký hiệu chỉ báo thể hiện xem liệu trường độn (theo cách khác, byte độn) hoặc trường truyền tín hiệu trong dải có trong khung BB hay không.

Trường Checksum có thể được sử dụng để kiểm tra lỗi của phần đầu khung BB hoặc trường OPTIONI, với kích thước là 1 bit.

Trường Checksum có thể được bao gồm trong phần đầu tùy chọn khi phần đầu được chia thành phần đầu khung BB và phần đầu tùy chọn.

Như được nêu trên, trường độn bao gồm phần đầu độn và byte độn.

Trường SYNCD\_LSB trên Fig.48 và trường PKTSPTR\_LSB trên Fig.47 có thể được sử dụng với cùng ý nghĩa.

Ở đây, kích thước của trường SYNCD\_LSB trên Fig.48 được đề xuất theo sáng chế này được tăng lên 6 bit bởi thêm 1 bit, trong khi kích thước của trường PKTSPTR\_LSB là 5 bit.

Nghĩa là, độ dài của SYNCD mà có thể được thể hiện bởi trường SYNCD\_LSB gồm 6 bit đến khoảng 63 (26-1) byte, trong khi độ dài của PKTSPTR mà có thể được thể hiện bởi trường PKTSPTR\_LSB 5 bit lớn nhất là 31 (25-1) byte.

Nghĩa là, trường hợp mà ở đó trường SYNCD\_MSB được bổ sung vào phần đầu hoặc phần đầu khung BB hoặc phần đầu tùy chọn bằng cách kiểm soát kích thước của trường SYNCD\_LSB được đề xuất theo sáng chế này được làm giảm, và kết quả là, chi phí của việc truyền của khung BB có thể được làm giảm.

Chẳng hạn, giả định rằng dòng MPEG2-TS là 188 byte được truyền.

(1) Trong trường hợp cấu trúc khung BB trên Fig.43, trường hợp mà ở đó chỉ trường PKTSPTR\_LSB được bao gồm trong phần đầu khung BB để truyền gói TS là 188 byte được bao gồm (nghĩa là, trường hợp mà ở đó độ dài PKTSPTR có giá trị là 31 byte hoặc nhỏ hơn) tương ứng với khoảng 16,49% (31 byte/188 byte).

Nghĩa là, khung BB tương ứng với 16,49% bao gồm phần đầu có kích thước là 1 byte, và khung BB tương ứng với phần còn lại 83,51% bao gồm phần đầu có kích thước là 2 byte.

Ở đây, phần đầu thể hiện định dạng có liên quan tới phụ tải, và có thể có nghĩa là phần đầu khung BB hoặc có nghĩa là bao gồm phần đầu khung BB và phần đầu tùy chọn.

Do đó, khung BB bao gồm trung bình phần đầu có kích thước là 1,83 byte.

(2) Mặt khác, trong trường hợp cấu trúc khung BB trên Fig.48, trường hợp mà ở đó chỉ trường SYNCD\_LSB được bao gồm trong phần đầu khung BB để truyền gói TS là 188 byte được bao gồm (nghĩa là, trường hợp mà ở đó độ dài SYNCD có giá trị là 63 byte hoặc nhỏ hơn) tương ứng với khoảng 33,51% (63 byte/188 byte).

Nghĩa là, khung BB tương ứng với 33,51% bao gồm phần đầu có kích thước là 1 byte, và khung BB tương ứng với phần còn lại 66,49% bao gồm phần đầu có kích thước là 2 byte.

Do đó, khung BB bao gồm trung bình phần đầu khung BB có kích thước là 1,66 byte, và kết quả là, có thể nhận thấy rằng chi phí của việc truyền của khung BB có thể được làm giảm nhiều so với trường hợp có cấu trúc khung BB trên Fig.47.

Ngoài ra cấu trúc khung BB trên Fig.48 có thể thực hiện chức năng bổ sung mà có thể phát hiện lỗi đối với phần đầu, nhờ sử dụng 1 bit được bao gồm trong phần đầu tùy chọn làm 1 bit kiểm tra tổng của phần đầu hoặc 1 bit kiểm tra tổng của trường OPTIONI được bao gồm trong phần đầu.

Fig.49 là sơ đồ minh họa ví dụ khác về cấu trúc khung BB được đề xuất theo sáng chế.

Bộ xử lý đầu ra trên Fig.49 thực hiện các chức năng, các quy trình, và/hoặc

các phương pháp được đề xuất trên Fig.50, Fig.51 và Fig.53 được mô tả dưới đây.

Cấu trúc khung BB trên Fig.49 khác với cấu trúc khung BB trên Fig.48 về các kích thước của trường SYNCD\_LSB /trường SYNCD\_MSB và vị trí của trường STUFFI, nhưng các phần khác của nó là giống nhau.

Dưới đây, phần mô tả cho các phần giống nhau như cấu trúc khung BB trên Fig.48 được bỏ qua, và các phần khác nhau sẽ được mô tả chính.

Trường OPTIONI và trường SYNCD\_LSB được kết hợp để được xác định như phần đầu khung BB, và trường SYNCD\_MSB, trường STUFFI, và trường Checksum được kết hợp để được xác định như phần đầu tùy chọn.

Ngoài ra trường OPTIONI, trường SYNCD\_LSB, trường SYNCD\_MSB, trường STUFFI, và trường Checksum được kết hợp để được xác định như một phần đầu.

Trong trường hợp này, phần đầu có thể cũng được thể hiện bởi phần đầu khung BB.

Như ví dụ khác nữa, trường STUFFI và trường Checksum có thể được kết hợp thành một trường cụ thể. Điều này sẽ được mô tả chi tiết trên Fig.50 và Fig.51 được mô tả dưới đây. Như được minh họa trên Fig.49, kích thước của trường SYNCD\_LSB là 7 bit, và kích thước của trường SYNCD\_MSB là 6 bit.

Giống như Fig.49, khi kích thước của trường SYNCD\_LSB là 7 bit, độ dài của số lượng lớn các SYNCD có thể được thể hiện.

Nghĩa là, khi kích thước của trường SYNCD\_LSB là 7 bit, độ dài thể hiện được của SYCND là 127 (27-1) byte và trở nên lớn hơn khoảng bốn lần so với trường hợp (31 byte) trên Fig.48.

Tương tự, giả định rằng dòng MPEG2-TS là 188 byte được truyền. Như được minh họa trên Fig.50, kích thước của trường SYNCD\_LSB là 7 bit, và kích thước của trường SYNCD\_MSB là 6 bit.

Trong trường hợp của cấu trúc khung BB trên Fig.49, trường hợp mà ở đó chỉ trường SYNCD\_LSB được bao gồm trong phần đầu khung BB để truyền gói TS là 188 byte được bao gồm (nghĩa là, trường hợp mà ở đó độ dài SYNCD có giá trị là

127 byte hoặc nhỏ hơn) tương ứng với khoảng 67,55% (127 byte/188 byte).

Nghĩa là, khung BB tương ứng với 67,55% bao gồm phần đầu có kích thước là 1 byte, và khung BB tương ứng với phần còn lại 32,45% bao gồm phần đầu có kích thước là 2 byte.

Do đó, khung BB bao gồm trung bình phần đầu có kích thước là 1,32 byte, và kết quả là, chi phí của việc truyền của khung BB có thể được làm giảm nhiều so với trường hợp có các cấu trúc khung BB trên Fig.47 và Fig.48.

Tương tự, ngay cả trong cấu trúc khung BB trên Fig.49, việc kiểm tra lỗi đối với phần đầu có thể được thực hiện bổ sung nhờ sử dụng 1 bit dư có trong phần đầu như kiểm tra tổng (là 1 bit kiểm tra tổng của phần đầu hoặc như kiểm tra tổng của trường OPTIONI).

Fig.50 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Như được minh họa trên Fig.50, trường STUFFI và trường Checksum nêu trên có thể được kết hợp thành một trường cụ thể 5010.

Trường cụ thể 5010 có thể được sử dụng làm giá trị chỉ báo xem liệu trường độn có mặt trong khung BB hay không.

Trường cụ thể 5010 có thể được thể hiện như trường ký hiệu chỉ báo mở rộng (EXT\_I) và có thể có kích thước là 2 bit.

Ngoài ra trường cụ thể 5010 có thể được thể hiện như trường ký hiệu chỉ báo phần đầu tùy chọn (Optional Header Indicator, viết tắt là OPTI).

Trường OPTI có thể có nghĩa là trường chỉ báo xem liệu phần đầu có bao gồm phần độn hay không.

Khung BB có thể được cấu thành bởi phần đầu và phụ tải và phần nghe có thể được cấu thành bởi một hoặc nhiều phần đầu con.

Nghĩa là, một hoặc nhiều phần đầu con có thể được thể hiện như phần đầu thứ nhất, phần đầu thứ hai, phần đầu thứ ba, và tương tự.

Ví dụ, phần đầu thứ nhất có thể được thể hiện như phần đầu BBF, phần đầu

cơ sở, hoặc tương tự và phần đầu thứ hai có thể được thể hiện như phần đầu tùy chọn, phần đầu không bắt buộc, hoặc tương tự.

Trường cụ thể 5010 có thể được bao gồm trong đoạn đầu tùy chọn của khung BB.

Bảng 36 được đưa ra dưới đây minh họa một ví dụ về định dạng trường cụ thể (trường EXT\_I).

Bảng 36

| EXT_I | Lưu ý        |
|-------|--------------|
| 00    | Không độn    |
| 01    | 1 byte độn   |
| 10    | 2 byte độn   |
| 11    | 3 ~ byte độn |

Trong bảng 36, giá trị trường cụ thể là (1) ‘00’ chỉ báo trường hợp mà ở đó không có phần độn trong khung BB, (2) ‘01’ chỉ báo trường hợp mà ở đó phần độn là 1 byte có mặt trong khung BB, (3) ‘10’ chỉ báo trường hợp mà ở đó phần độn là 1 byte có mặt trong khung BB, và (4) ‘11’ chỉ báo trường hợp mà ở đó phần độn là 3 byte hoặc lớn hơn có mặt trong khung BB.

Như được nêu trên, trường độn có thể bao gồm trường phần đầu độn và trường byte độn.

Ngoài ra trường phần đầu độn có thể bao gồm trường STUFF\_TYPE, trường STUFF\_LEN\_LSB, và trường STUFF\_LEN\_MSB.

Trường STUFF\_TYPE có thể được thể hiện trường loại mở rộng (EXT\_TYPE). Trong trường hợp này, trường EXT\_TYPE có thể chỉ báo loại trường độn.

Ngoài ra trường độn có thể được thể hiện trường mở rộng.

Trường STUFF\_TYPE có thể được bao gồm trong khung BB hoặc phần đầu khung BB khi trường EXT\_I có '01', '10', hoặc '11'. Các nội dung chi tiết của nó sẽ được mô tả dựa vào bảng 37 được đưa ra dưới đây.

Trường STUFF\_TYPE có thể là 3 bit, trường STUFF\_LEN\_LSB có thể là 5 bit, và trường STUFF\_LEN\_MSB có thể là 8 bit.

Trường STUFF\_LEN\_LSB có thể được thể hiện như trường EXT\_LEN\_LSB và trường STUFF\_LEN\_MSB có thể được thể hiện như trường EXT\_LEN\_MSB.

Dưới đây, trường STUFF\_TYPE và trường STUFF\_LEN mà có thể được xác định theo giá trị trường cụ thể, và các ý nghĩa của nó sẽ được mô tả dựa vào bảng 37 như một ví dụ.

Bảng 37

| EXT_I | STUFF_TYPE    | STUFF_LEN     | Mô tả                    |
|-------|---------------|---------------|--------------------------|
| 00    | Không tồn tại | Không tồn tại | Không độn                |
| 01    | 000           | 00000         | 1 byte độn               |
| 10    | 000           | 00000         | 2 byte độn               |
| 11    | 000           | stuff_len     | 3 ~ byte độn             |
| 11    | 001           | stuff_len     | Độn+chế độ 1(ISSY)       |
| 11    | 010           | stuff_len     | Độn+chế độ 2(INBAND_SIG) |
| ...   | ...           | ...           | ...                      |
| 11    | 111           | stuff_len     | Độn+chế độ 7(dự trữ)     |

Trong bảng 37, khi giá trị trường cụ thể (ví dụ, trường EXT\_I) là '00', vì không độn có mặt, trường STUFF\_TYPE và trường STUFF\_LEN không có mặt

trong trường độn.

Khi giá trị trường cụ thể ‘01’, giá trị trường STUFF\_TYPE là ‘000’, và giá trị trường STUFF\_LEN là ‘00000’, phần độn là 1 byte được bao gồm trong khung BB (theo cách khác, trường độn).

Khi giá trị trường cụ thể là ‘10’, giá trị trường STUFF\_TYPE là ‘000’, và giá trị trường STUFF\_LEN là ‘00000’, phần độn là 2 byte được bao gồm trong khung BB (theo cách khác, trường độn). Khi giá trị trường cụ thể ‘110’, giá trị trường STUFF\_TYPE là ‘000’, và giá trị trường STUFF\_LEN là ‘stuff\_len’, phần độn là 3 byte hoặc lớn hơn được bao gồm trong khung BB (theo cách khác, trường độn).

Khi giá trị trường cụ thể là ‘11’, giá trị trường STUFF\_TYPE là ‘001’, và giá trị trường STUFF\_LEN là ‘stuff\_len’, phần độn và truyền tín hiệu A trong dải được bao gồm trong khung BB (theo cách khác, trường độn).

A trong dải có thể là INBAND\_ISSY.

Khi giá trị trường cụ thể là ‘11’, giá trị trường STUFF\_TYPE là ‘010’, và giá trị trường STUFF\_LEN là ‘stuff\_len’, phần độn và truyền tín hiệu B trong dải được bao gồm trong khung BB (theo cách khác, trường độn).

B trong dải có thể là INBAND\_SIG.

Khi giá trị trường cụ thể là ‘11’, giá trị trường STUFF\_TYPE là ‘111’, và giá trị trường STUFF\_LEN là ‘stuff\_len’, phần độn và thông tin khác được bao gồm trong khung BB (theo cách khác, trường độn).

Ngoài ra giá trị trường STUFF\_LEN có thể được chia thành giá trị trường STUFF\_LEN\_LSB (5 bit) và giá trị trường STUFF\_LEN\_MSB (8 bit).

Điều này sẽ được mô tả dựa vào bảng 38.

Bảng 38

| EXT_I | STUFF_TYPE    | STUFF_LEN_LSB | STUFF_LEN_MSB | Mô tả     |
|-------|---------------|---------------|---------------|-----------|
| 00    | Không tồn tại | Không tồn tại | Không tồn tại | Không độn |

|     |     |               |               |                      |
|-----|-----|---------------|---------------|----------------------|
| 01  | 000 | 00000         | Không tồn tại | 1 byte độn           |
| 10  | 000 | 00000         | 00000000      | 2 byte độn           |
| 11  | 000 | stuff_len_lsb | stuff_len_msb | 3 ~ byte độn         |
| 11  | 001 | stuff_len_lsb | Không tồn tại | Chỉ chế độ 1 (ISSY)  |
| 11  | 010 | stuff_len_lsb | stuff_len_msb | Độn+chế độ 1(ISSY)   |
| ... | ... | ...           | ...           | ...                  |
| 11  | 111 | stuff_len_lsb | stuff_len_msb | Độn+chế độ 8(dự trữ) |

Trong bảng 38, khi giá trị trường cụ thể là ‘01’, giá trị trường STUFF\_TYPE là ‘000’, giá trị trường STUFF\_LEN\_LSB là ‘00000’, và giá trị trường STUFF\_LEN\_MSB là ‘không tồn tại’, phần độn là 1 byte được bao gồm trong khung BB (theo cách khác, trường độn).

Khi giá trị trường cụ thể là ‘10’, giá trị trường STUFF\_TYPE là ‘000’, giá trị trường STUFF\_LEN\_LSB là ‘00000’, và giá trị trường STUFF\_LEN\_MSB là ‘00000000’, phần độn là 2 byte được bao gồm trong khung BB (theo cách khác, trường độn).

Khi giá trị trường cụ thể là ‘11’, giá trị trường STUFF\_TYPE là ‘000’, giá trị trường STUFF\_LEN\_LSB là ‘stuff\_len\_lsb’, và giá trị trường STUFF\_LEN\_MSB là ‘stuff\_len\_msb’, phần độn là 3 byte hoặc lớn hơn được bao gồm trong khung BB (theo cách khác, trường độn).

Khi giá trị trường cụ thể là ‘11’, giá trị trường STUFF\_TYPE là ‘001’, giá trị trường STUFF\_LEN\_LSB là ‘stuff\_len\_lsb’, và giá trị trường STUFF\_LEN\_MSB là ‘không tồn tại’, chỉ truyền tín hiệu A trong dải được bao gồm trong khung BB (theo cách khác, trường độn). Tốt hơn là, chỉ truyền tín hiệu A trong dải được bao

gồm trong trường độ chỉ khi truyền tín hiệu A trong dải có thể được thể hiện bởi 32 byte. A trong dải có thể là INBAND\_ISSY.

Khi giá trị trường cụ thể là '11', giá trị trường STUFF\_TYPE là '010', giá trị trường STUFF\_LEN\_LSB là 'stuff\_len\_lsb', và giá trị trường STUFF\_LEN\_MSB là 'stuff\_len\_msb', phần độ và truyền tín hiệu A trong dải được bao gồm trong khung BB (theo cách khác, trường độ).

Khi giá trị trường cụ thể '11', giá trị trường STUFF\_TYPE là '111', giá trị trường STUFF\_LEN\_LSB là 'stuff\_len\_lsb', và giá trị trường STUFF\_LEN\_MSB là 'stuff\_len\_msb', phần độ và thông tin khác được bao gồm trong khung BB (theo cách khác, trường độ).

Fig.51 là sơ đồ minh họa ví dụ khác nữa về cấu trúc khung BB được đề xuất theo sáng chế.

Trên Fig.51, như phương pháp được kết hợp với phương pháp sử dụng trường STUFF\_TYPE, phương pháp này được đề xuất, mà chia và sử dụng trường STUFF\_TYPE 3 bit thành trường ký hiệu chỉ báo MSB 1 bit (MSB\_I) 5111 và trường STUFF\_TYPE 2 bit 5112 để sử dụng hiệu quả.

Trường MSB\_I 5111 thể hiện ký hiệu chỉ báo mà chỉ báo xem liệu trường STUFF\_LEN\_MSB có mặt hay không.

Ví dụ, khi giá trị trường MSB\_I là '0', giá trị này có thể chỉ báo rằng chỉ STUFF\_LEN\_LSB (5 bit) được sử dụng trong phần đầu độ và khi giá trị trường MSB\_I là '1', giá trị có thể chỉ báo rằng trường STUFF\_LEN\_LSB (5 bit) và STUFF\_LEN\_MSB (8 bit) được sử dụng trong phần đầu độ 5110.

Ví dụ, khi giá trị trường MSB\_I là '0', chỉ trường STUFF\_LEN\_LSB (5 bit) được sử dụng (theo cách khác, được bao gồm) trong phần đầu độ và kích thước của trường độ mà có thể được thể hiện là 32 byte.

Khi giá trị trường MSB\_I là '1', trường STUFF\_LEN\_LSB (5 bit) và trường STUFF\_LEN\_MSB (8 bit) được sử dụng (theo cách khác, được bao gồm) trong phần đầu độ và kích thước của STUFF\_LEN có thể được thể hiện là 13 bit.

Tiếp theo, trường STUFF\_TYPE (2 bit, 4712) chỉ báo loại trường độ sử dụng.

Trong ví dụ về trường STUFF\_TYPE có thể bao gồm ISSY, truyền tín hiệu trong dải, và tương tự.

Trường STUFF\_TYPE chỉ báo loại độn được thiết kế sao cho khoảng trường độn được sử dụng cho các mục đích khác (ví dụ, truyền tín hiệu trong dải và trong dải ISSY).

Các ý nghĩa của trường MSB\_I và trường STUFF\_TYPE được xác định mới với kích thước là 2 bit sẽ được mô tả cụ thể hơn dựa vào bảng 39 được đưa ra dưới đây.

Bảng 39

| EXT_I | MSB_I | STUFF_TYPE | STUFF_LEN_LSB | STUFF_LEN_MSB | Mô tả                                                 |
|-------|-------|------------|---------------|---------------|-------------------------------------------------------|
| 11    | 0     | 00         | stuff_len_lsb | Không tồn tại | Dự trữ                                                |
| 11    | 1     | 00         | stuff_len_lsb | stuff_len_msb | 3 ~ byte độn (stuff_len)                              |
| 11    | 0     | 01         | stuff_len_lsb | Không tồn tại | Bảng ISSY trong dải + độn (kích thước $\leq 32$ byte) |
| 11    | 1     | 01         | stuff_len_lsb | stuff_len_msb | Bảng ISSY trong dải + độn (kích thước $> 32$ byte)    |
| 11    | 0     | 10         | stuff_len_lsb | Không tồn tại | Bảng PLS trong dải + độn (kích thước $\leq 32$ byte)  |
| 11    | 1     | 10         | stuff_len_lsb | stuff_len_msb | Bảng PLS trong dải + độn (kích                        |

|    |   |    |               |               |                                                       |
|----|---|----|---------------|---------------|-------------------------------------------------------|
|    |   |    |               |               | thước > 32 byte)                                      |
| 11 | 0 | 11 | stuff_len_lsb | Không tồn tại | Chế độ dự trữ +<br>độn (kích thước<br>$\leq 32$ byte) |
| 11 | 1 | 11 | stuff_len_lsb | stuff_len_msb | Chế độ dự trữ +<br>độn (kích thước<br>> 32 byte)      |

Trong bảng 39, khi giá trị trường STUFF\_TYPE là '00', giá trị chỉ báo trường hợp trong đó trường độn được cấu thành bởi chỉ các byte độn và khi giá trị trường STUFF\_TYPE là 01, 10, và 11, các giá trị thể hiện các chế độ tương ứng được sử dụng cho các mục đích khác nhau trong trường độn.

Như được minh họa trên Fig.51 và bảng 39, trong trường hợp mà ở đó cả trường MSB\_I và trường STUFF\_TYPE được sử dụng trong phần đầu độn, khi kích thước của tổng ISSY và độn trong trường độn là 32 byte hoặc nhỏ hơn, vì phần đầu độn không bao gồm trường STUFF\_LEN\_MSB, 1 byte chi phí có thể được làm giảm.

Khi kích thước của tổng ISSY và độn trong trường độn lớn hơn 32 byte, trường MSI\_I có thể được thiết đặt là (theo cách khác, được đánh dấu) '1' và trường STUFF\_LEN\_MSB có thể được sử dụng trong phần đầu độn.

Fig.52 là lưu đồ minh họa một ví dụ về phương pháp để truyền tín hiệu phát rộng được đề xuất theo sáng chế.

Dựa vào Fig.52, thiết bị truyền tín hiệu phát rộng được đề xuất theo sáng chế xử lý các dòng đầu vào qua môđun định dạng đầu vào (S5210). Nghĩa là, thiết bị truyền tín hiệu phát rộng định dạng các dòng đầu vào với nhiều ống dữ liệu (các DP) trong môđun định dạng đầu vào.

Chi tiết, thiết bị truyền tín hiệu phát rộng phân phối các gói dữ liệu đến phụ tải của khung dải gốc (BBF) và bổ sung phần đầu chỉ báo định dạng cho phụ tải của khung dải gốc, để định dạng đầu vào trong S5210.

Các ống dữ liệu (các DP) có thể được biểu diễn thành các kênh truyền dữ liệu.

Phần đầu có thể bao gồm trường EXT\_I và trường độn như được minh họa trên Fig.50 và Fig.51.

Trường EXT\_I có nghĩa là trường chỉ báo xem liệu trường độn có mặt trong khung BB hay không và có thể có kích thước là 2 bit.

Trường độn bao gồm phần đầu độn và phần đầu độn bao gồm trường MSB\_I và trường STUFF\_TYPE.

Trường MSB\_I thể hiện ký hiệu chỉ báo mà chỉ báo xem liệu trường STUFF\_LEN\_MSB có mặt hay không và có thể có kích thước là 1 bit.

Ngoài ra trường STUFF\_TYPE có nghĩa là trường chỉ báo loại trường độn sử dụng và có thể có kích thước là 2 bit.

Trường EXT\_I, trường MSB\_I, và trường STUFF\_TYPE có thể được sử dụng để thể hiện thông tin điều khiển để được áp dụng cho các phương án ví dụ khác.

Sau đó, thiết bị truyền tín hiệu phát rộng mã hóa dữ liệu của nhiều (được định dạng) DP đối với mỗi DP qua môđun điều biến và mã hóa đan xen bit (BICM) (S5220).

Môđun điều biến và mã hóa đan xen bit (BICM) có thể được biểu diễn thành bộ mã hóa.

Sau đó, thiết bị truyền tín hiệu phát rộng ánh xạ dữ liệu DP được mã hóa qua môđun xây dựng khung để tạo ra ít nhất một khung tín hiệu (S5230).

Môđun xây dựng khung có thể được biểu diễn thành bộ xây dựng khung.

Sau đó, thiết bị truyền tín hiệu phát rộng điều biến dữ liệu của khung tín hiệu được tạo ra của môđun tạo đa hợp phân chia theo tần số trực giao (OFDM) bởi sơ đồ đa hợp phân chia theo tần số trực giao (OFDM) và truyền tín hiệu phát rộng bao gồm dữ liệu được điều biến của khung tín hiệu (S5240).

Fig.53 là lưu đồ minh họa một ví dụ về phương pháp thu tín hiệu phát rộng được đề xuất theo sáng chế.

Dựa vào Fig.53, thiết bị thu tín hiệu phát rộng thu tín hiệu phát rộng từ bên ngoài qua môđun đồng bộ hóa và giải điều biến và giải điều biến dữ liệu nhờ sơ đồ

OFDM đối với tín hiệu phát rộng thu được (S5310).

Môđun đồng bộ hóa và giải điều biến có thể được biểu diễn thành bộ thu và bộ giải điều biến.

Sau đó, thiết bị thu tín hiệu phát rộng tách dữ liệu được giải điều biến thành ít nhất một khung tín hiệu qua môđun tách (S5320).

Môđun tách có thể được biểu diễn thành bộ phân tách khung.

Sau đó, thiết bị thu tín hiệu phát rộng giải mã ít nhất một khung tín hiệu được tách thành nhiều DP qua môđun giải ánh xạ và giải mã (S5330).

Môđun giải ánh xạ và giải mã có thể được biểu diễn thành bộ chuyển đổi và bộ giải mã.

Sau đó, thiết bị thu tín hiệu phát rộng khôi phục nhiều ống dữ liệu được đưa ra từ môđun giải ánh xạ và giải mã đến các dòng đầu vào qua môđun bộ xử lý đầu ra.

Thiết bị thu tín hiệu phát rộng giải mã thông tin được truyền đến phần đầu của khung dải gốc qua bộ xử lý khung dải gốc khối và khôi phục các dòng đầu vào nhờ sử dụng thông tin được giải mã, để xử lý đầu ra trong S5340.

Phần đầu có thể bao gồm trường EXT\_I và trường độn như được minh họa trên các hình vẽ từ Fig.50 đến Fig.52.

Trường EXT\_I có nghĩa là trường chỉ báo xem liệu trường độn có mặt trong khung BB hay không và có thể có kích thước là 2 bit.

Trường độn bao gồm phần đầu độn và phần đầu độn bao gồm trường MSB\_I và trường STUFF\_TYPE.

Trường MSB\_I thể hiện ký hiệu chỉ báo mà chỉ báo xem liệu trường STUFF\_LEN\_MSB có mặt hay không và có thể có kích thước là 1 bit.

Ngoài ra trường STUFF\_TYPE có nghĩa là trường mà chỉ báo loại trường độn sử dụng và có thể có kích thước là 2 bit.

Trường EXT\_I, trường MSB\_I, và trường STUFF\_TYPE có thể được sử dụng để biểu thị thông tin điều khiển để được áp dụng cho các phương án ví dụ khác.

Sáng chế có thể được hiểu rõ bởi những người có hiểu biết trung bình trong lĩnh vực kỹ thuật rằng các thay đổi và các sửa đổi khác nhau của sáng chế có thể được thực hiện mà không trệch khỏi nguyên lý và phạm vi của sáng chế. Do đó, chế này nhằm bao gồm các thay đổi và các sửa đổi được đề xuất trong các điểm yêu cầu bảo hộ kèm theo và phạm vi tương ứng của chúng.

Theo sáng chế, các thiết bị và phương pháp được đề cập và các phần mô tả về thiết bị và phương pháp có thể được áp dụng để bổ sung cho nhau.

Theo sáng chế, các phương pháp và các thiết bị để thu và truyền tín hiệu phát rộng được sử dụng.

Các phương án khác nhau đã được mô tả theo phương thức tốt nhất để thực hiện sáng chế.

Sáng chế là khả dụng trong một loạt các lĩnh vực sử dụng tín hiệu phát rộng.

Sáng sẽ trở nên rõ ràng đối với những người có hiểu biết trung bình trong lĩnh vực kỹ thuật mà các thay đổi và các sửa đổi khác nhau có thể được thực hiện theo sáng chế mà không trệch khỏi nguyên lý và phạm vi của sáng chế. Do đó, được dự định rằng sáng chế bao gồm các thay đổi và các sửa đổi khác nhau miễn là chúng nằm trong phạm vi của các điểm yêu cầu bảo hộ kèm theo và các phần tương đương của chúng.

## YÊU CẦU BẢO HỘ

1. Phương pháp thu tín hiệu phát rộng, phương pháp này bao gồm các bước:

thu tín hiệu phát rộng bao gồm khung tín hiệu;

thực hiện việc giải điều biến đa hợp phân chia theo tần số trực giao (Orthogonal Frequency Division Multiplexing, viết tắt là OFDM) trên tín hiệu phát rộng thu được;

phân tách khung tín hiệu của tín hiệu phát rộng được giải điều biến;

giải ánh xạ dữ liệu của ống lớp vật lý (Physical Layer Pipe, viết tắt là PLP) trong khung tín hiệu;

giải mã dữ liệu của PLP; và

xử lý ít nhất một khung dải gốc trong dữ liệu của PLP để đưa ra dòng dữ liệu,

trong đó khung dải gốc bao gồm phần đầu và phụ tải,

trong đó phần đầu bao gồm thông tin điều khiển để chỉ báo xem liệu phần thứ nhất có mặt trong phần đầu hay không và độ dài của phần thứ nhất khi phần thứ nhất có mặt, và

trong đó phần thứ nhất bao gồm thông tin loại để chỉ báo loại của phần thứ hai và thông tin độ dài để chỉ báo độ dài của phần thứ hai.

2. Phương pháp theo điểm 1, trong đó:

khi thông tin điều khiển có giá trị thứ nhất, giá trị thứ nhất của thông tin điều khiển chỉ báo rằng phần thứ nhất không có mặt,

khi thông tin điều khiển có giá trị thứ hai, giá trị thứ hai của thông tin điều khiển chỉ báo rằng phần thứ nhất có mặt và độ dài của phần thứ nhất là 1 byte, và

khi thông tin điều khiển có giá trị thứ ba, giá trị thứ ba của thông tin điều khiển chỉ báo rằng phần thứ nhất có mặt và độ dài của phần thứ nhất là 2 byte.

3. Phương pháp theo điểm 2, trong đó:

khi độ dài của phần thứ nhất là 2 byte, byte thứ nhất của phần thứ nhất bao

gồm thông tin loại và phần bit trọng số thấp nhất (Least Significant Bit, viết tắt là LSB) của thông tin độ dài và byte thứ hai của phần thứ nhất bao gồm phần bit trọng số lớn nhất (Most Significant Bit, viết tắt là MSB) của thông tin độ dài.

4. Phương pháp theo điểm 1, trong đó:

độ dài của thông tin loại là 3 bit,

độ dài của phần LSB của thông tin độ dài là 5 bit, và

độ dài của phần MSB của thông tin độ dài là 8 bit.

5. Phương pháp theo điểm 1, trong đó:

khi thông tin loại có giá trị thứ nhất, giá trị thứ nhất của thông tin loại chỉ báo rằng phần thứ hai bao gồm dữ liệu đệm, hoặc

khi thông tin loại có giá trị thứ hai, giá trị thứ hai của thông tin loại chỉ báo rằng phần thứ hai bao gồm dữ liệu chỉ báo thông tin truyền tín hiệu bổ sung.

6. Phương pháp theo điểm 1, trong đó:

thông tin điều khiển là trường ký hiệu chỉ báo mở rộng (EXT\_I).

7. Phương pháp theo điểm 1, trong đó độ dài của thông tin điều khiển là 2 bit.

8. Thiết bị thu tín hiệu phát rộng, thiết bị này bao gồm:

bộ thu để thu tín hiệu phát rộng bao gồm khung tín hiệu;

bộ giải điều biến để thực hiện việc giải điều biến đa hợp phân chia theo tần số trực giao (Orthogonal Frequency Division Multiplexing, viết tắt là OFDM) trên tín hiệu phát rộng thu được;

bộ phân tách khung dùng để phân tách khung tín hiệu của tín hiệu phát rộng được giải điều biến;

bộ giải ánh xạ dùng để giải ánh xạ dữ liệu của ống lớp vật lý (PLP) trong khung tín hiệu;

bộ giải mã dùng để giải mã dữ liệu của PLP; và

bộ xử lý đầu ra dùng để xử lý ít nhất một khung dải gốc trong dữ liệu của PLP

để đưa ra dòng dữ liệu,

trong đó khung dải gốc bao gồm phần đầu và phụ tải,

trong đó phần đầu bao gồm thông tin điều khiển để chỉ báo xem phần thứ nhất có mặt trong phần đầu hay không và độ dài của phần thứ nhất khi phần thứ nhất có mặt,

trong đó phần thứ nhất bao gồm thông tin loại để chỉ báo loại của phần thứ hai và thông tin độ dài để chỉ báo độ dài của phần thứ hai.

9. Thiết bị theo điểm 8, trong đó:

khi thông tin điều khiển có giá trị thứ nhất, giá trị thứ nhất của thông tin điều khiển chỉ báo rằng phần thứ nhất không có mặt,

khi thông tin điều khiển có giá trị thứ hai, giá trị thứ hai của thông tin điều khiển chỉ báo rằng phần thứ nhất có mặt và độ dài của phần thứ nhất là 1 byte, và

khi thông tin điều khiển có giá trị thứ ba, giá trị thứ ba của thông tin điều khiển chỉ báo rằng phần thứ nhất có mặt và độ dài của phần thứ nhất là 2 byte.

10. Thiết bị theo điểm 9, trong đó:

khi độ dài của phần thứ nhất là 2 byte, byte thứ nhất của phần thứ nhất bao gồm thông tin loại và phần bit trọng số nhỏ nhất (LSB) của thông tin độ dài và byte thứ hai của phần thứ nhất bao gồm phần bit trọng số lớn nhất (MSB) của thông tin độ dài.

11. Thiết bị theo điểm 8, trong đó:

khi thông tin loại có giá trị thứ nhất, giá trị thứ nhất của thông tin loại chỉ báo rằng phần thứ hai bao gồm dữ liệu đệm, hoặc

khi thông tin loại có giá trị thứ hai, giá trị thứ hai của thông tin loại chỉ báo rằng phần thứ hai bao gồm dữ liệu chỉ báo thông tin truyền tín hiệu bổ sung.

12. Thiết bị theo điểm 8, trong đó:

độ dài của thông tin loại là 3 bit,

độ dài của phần LSB của thông tin độ dài là 5 bit, và

độ dài của phần MSB của thông tin độ dài là 8 bit.

13. Thiết bị theo điểm 8, trong đó độ dài của thông tin điều khiển là 2 bit.

14. Thiết bị theo điểm 8, trong đó:

thông tin điều khiển là trường ký hiệu chỉ báo mở rộng (EXT\_I).

FIG. 1

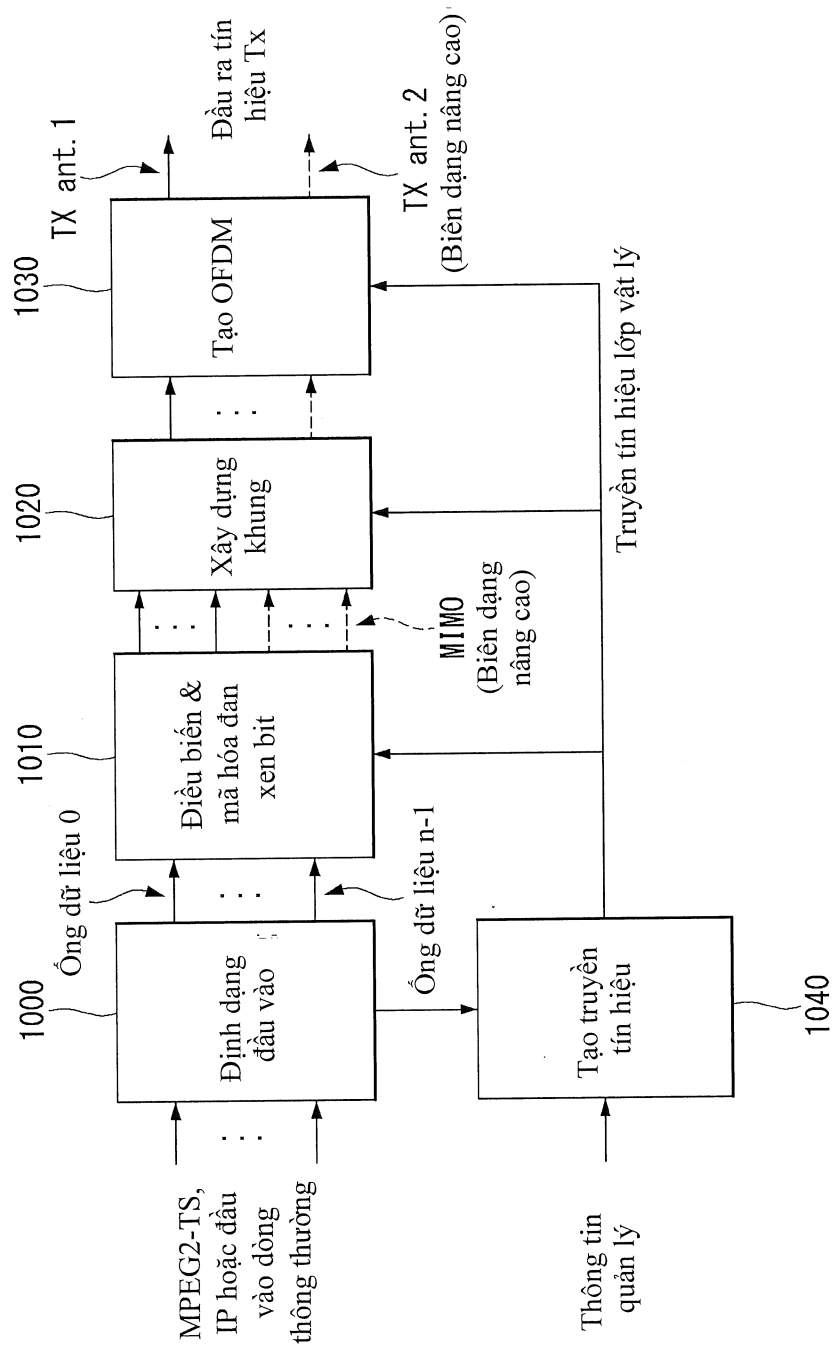


FIG. 2

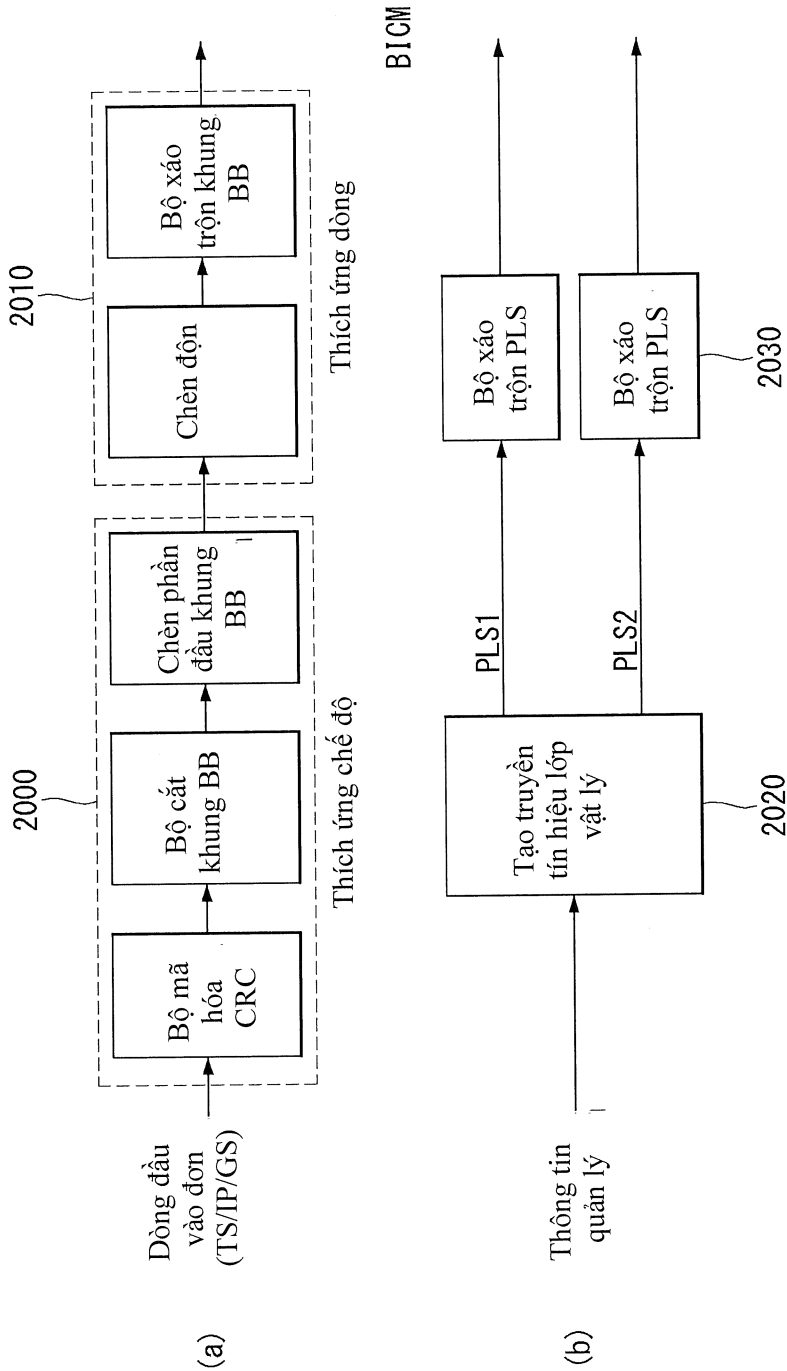


FIG. 3

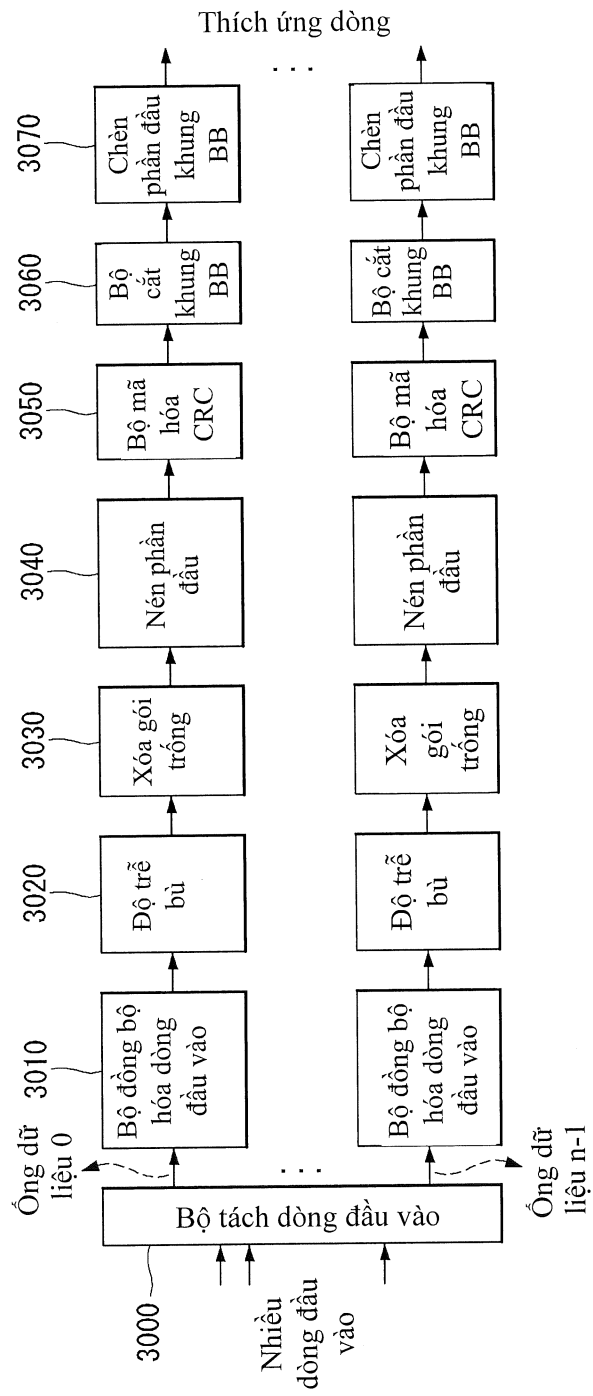


FIG. 4

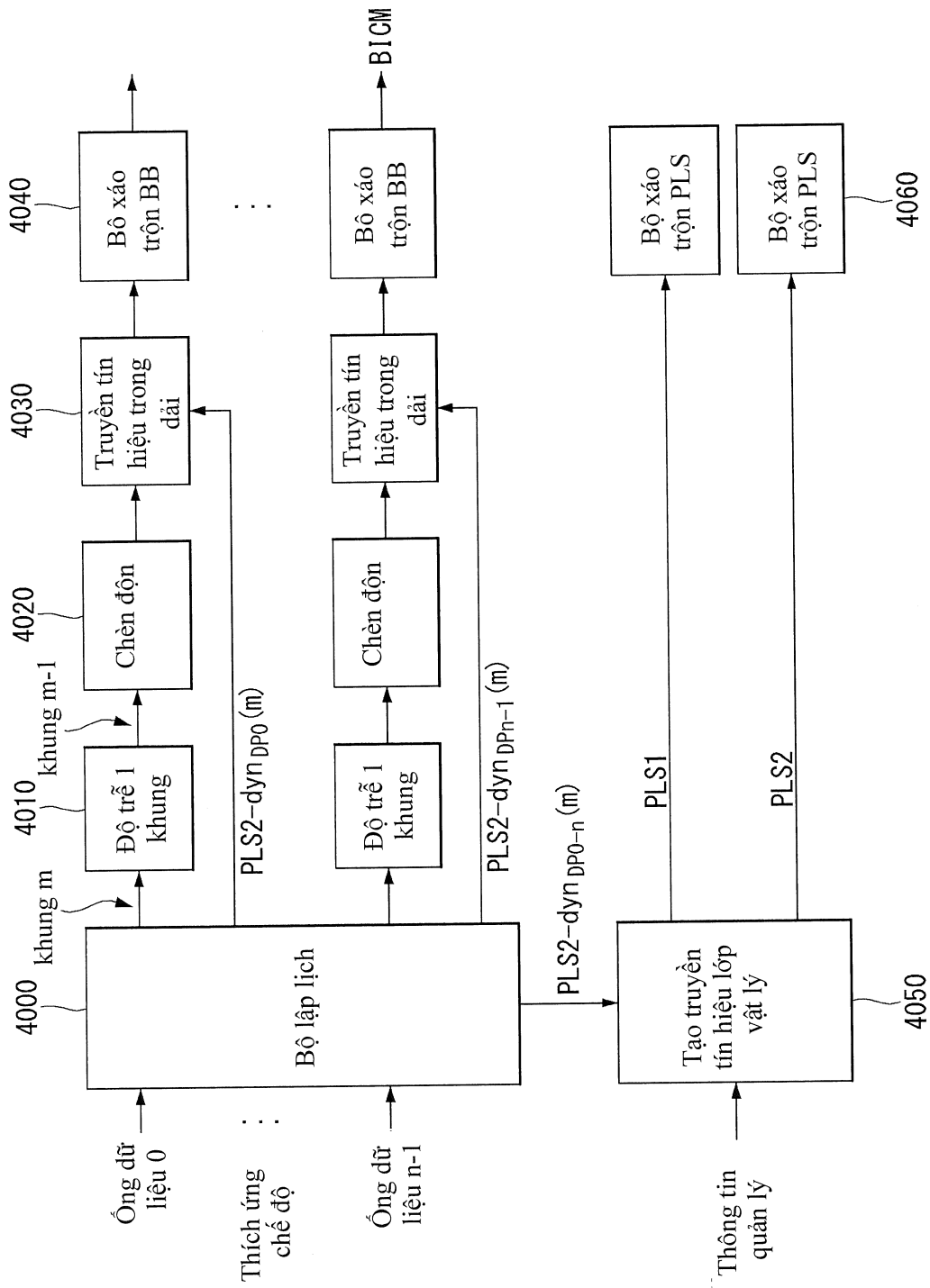


FIG. 5

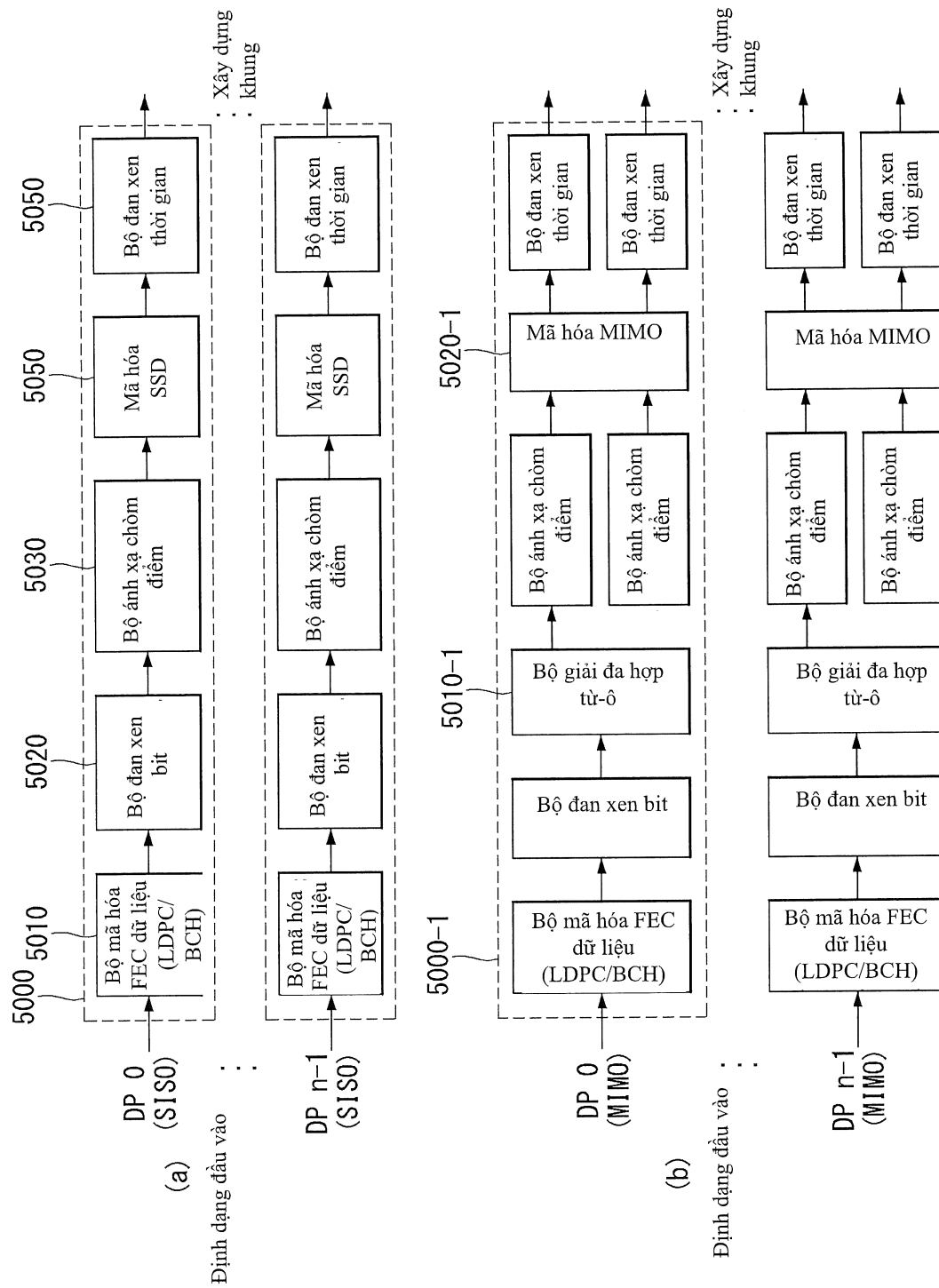


FIG. 6

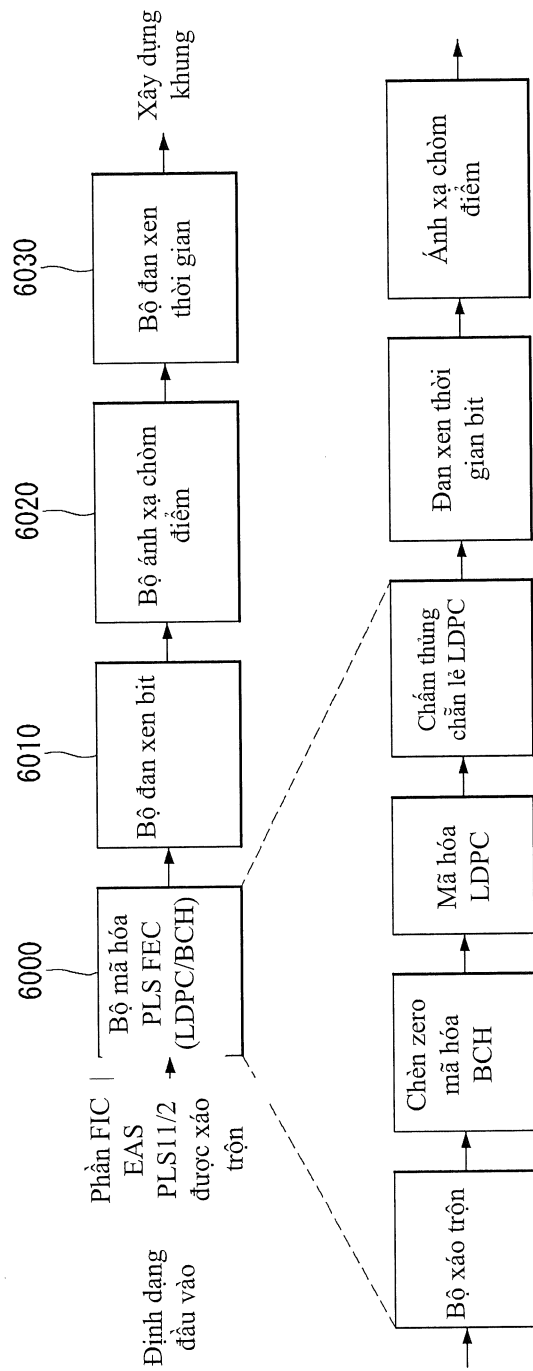


FIG. 7

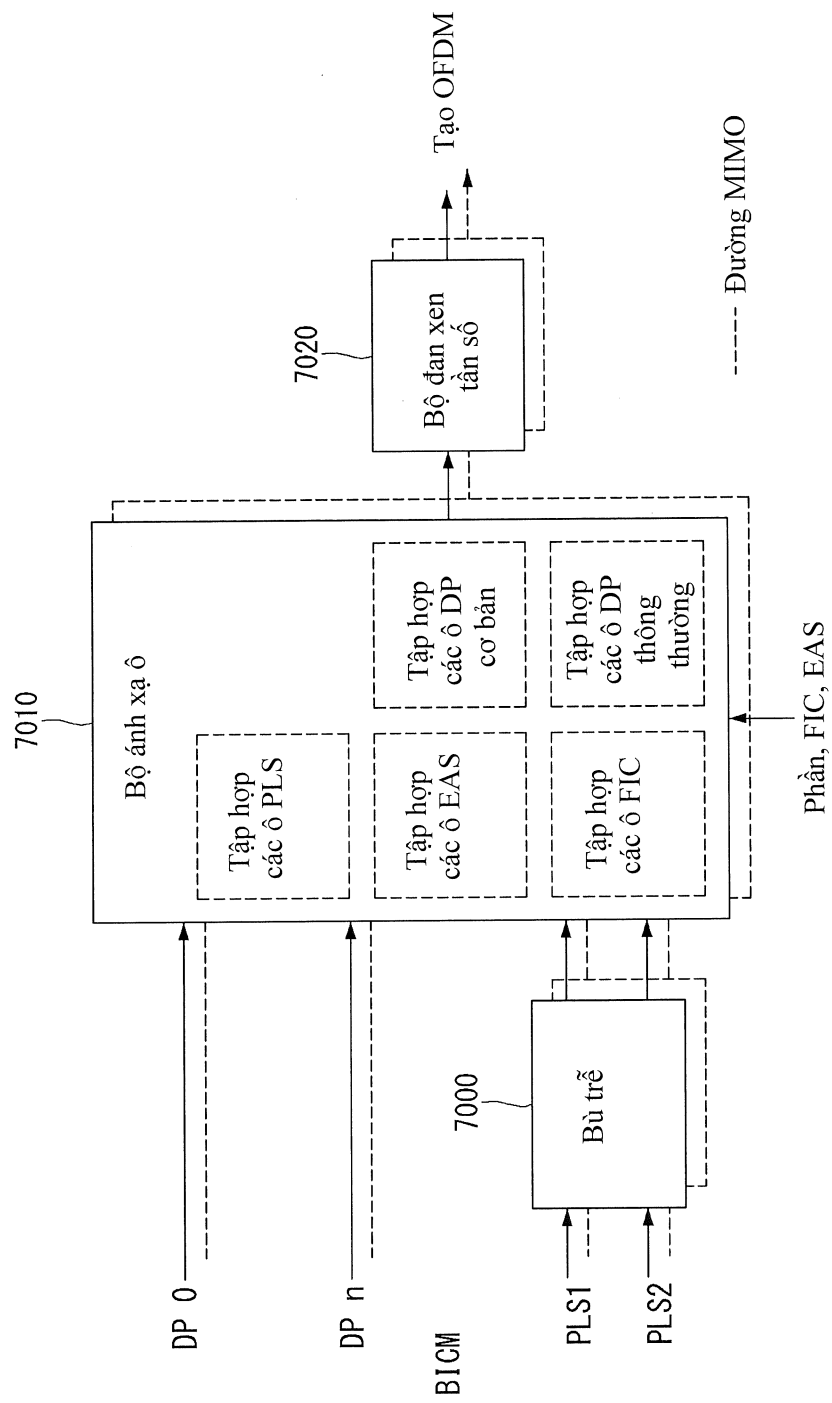


FIG. 8

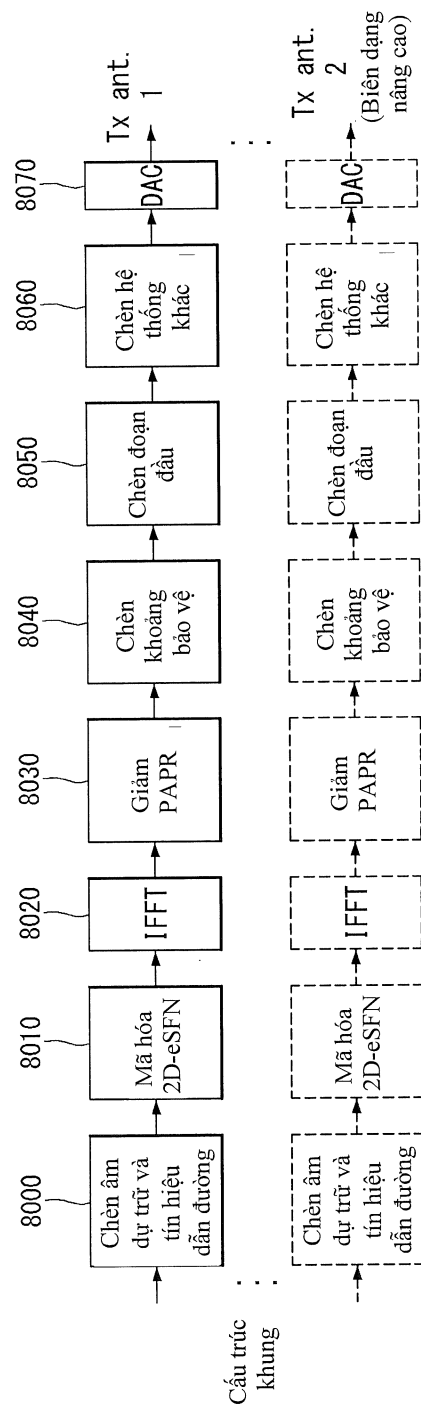


FIG. 9

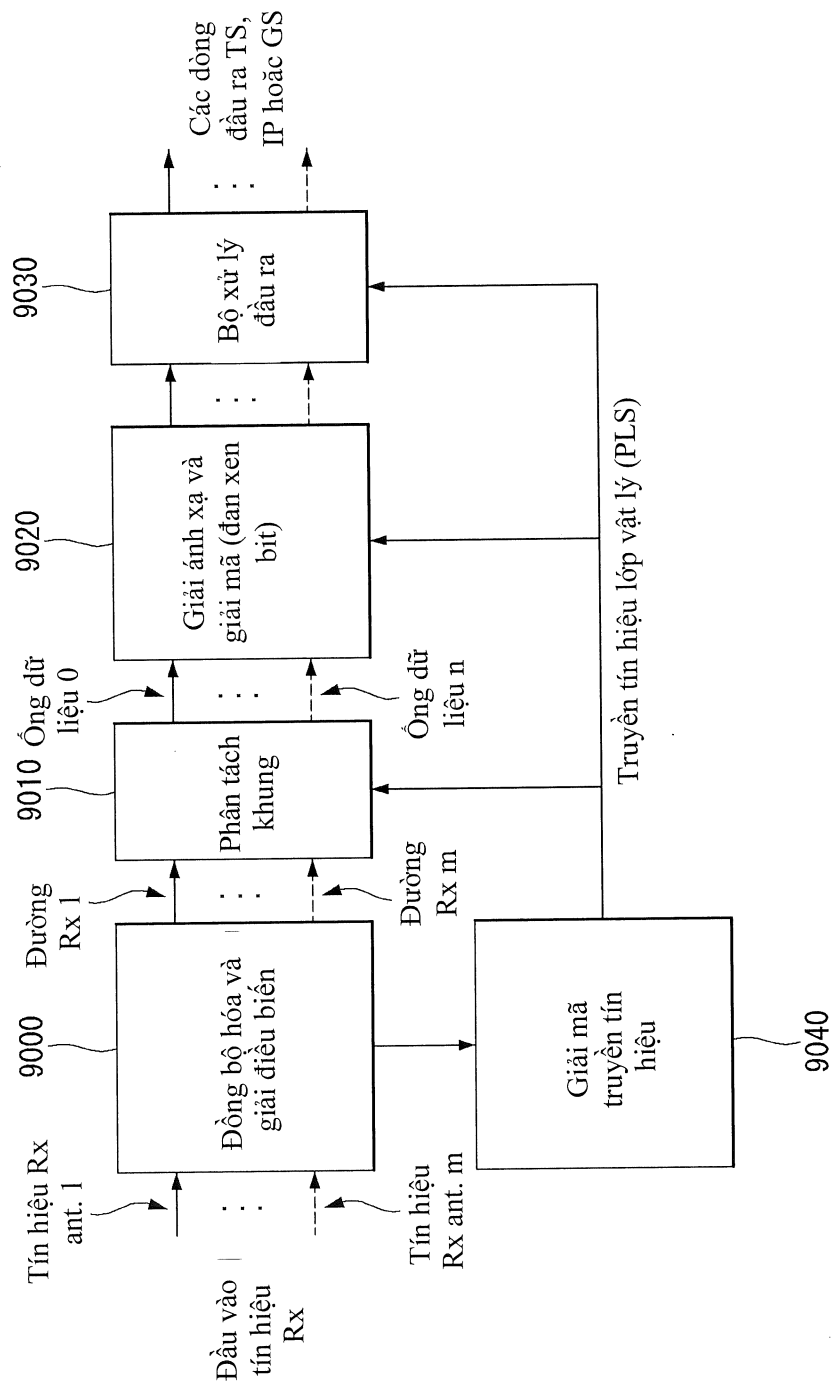


FIG. 10

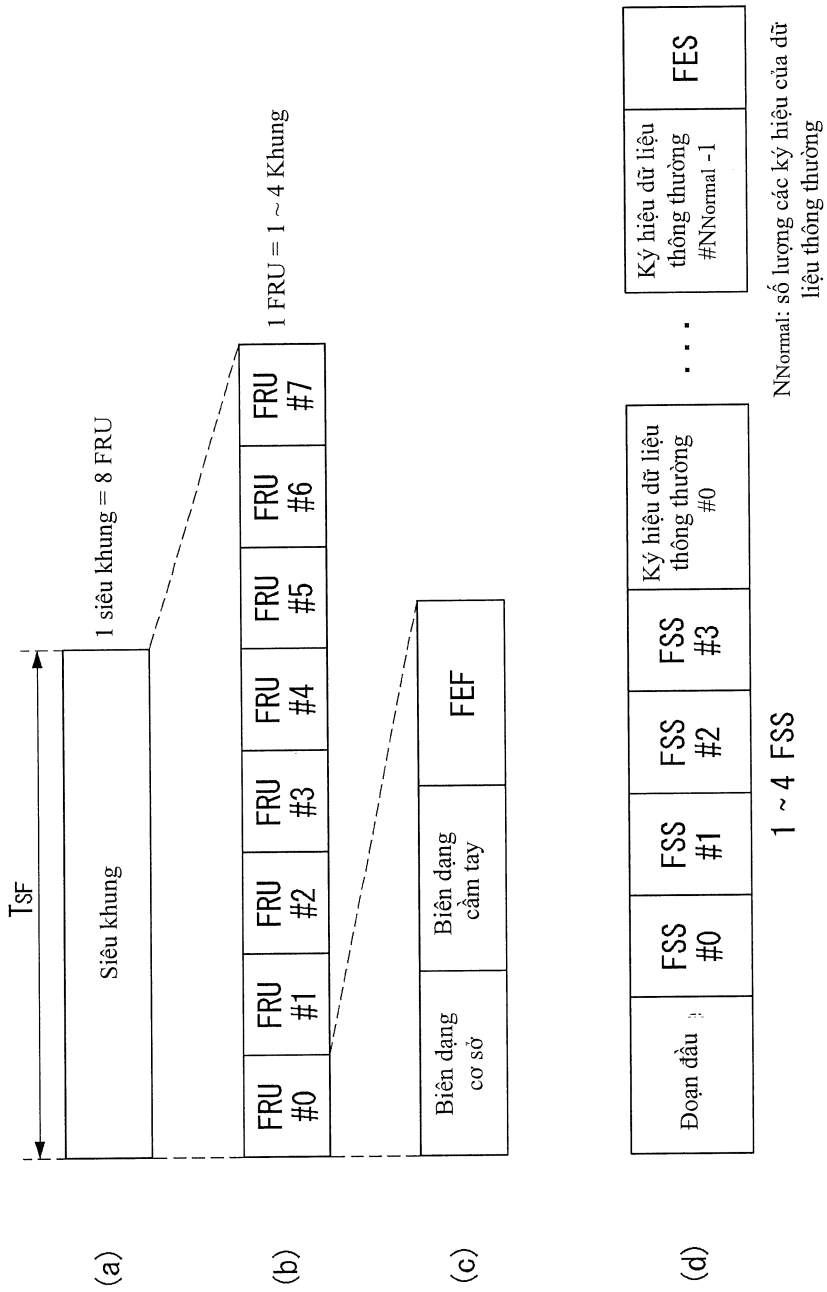


FIG. 11

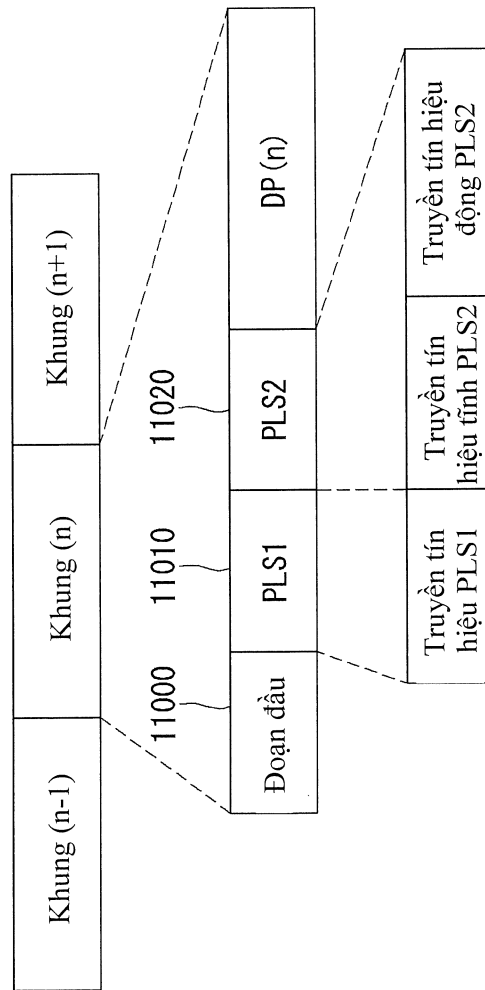


FIG.12

| Nội dung      | Số bit |
|---------------|--------|
| PHY_PROFILE   | 3      |
| FFT_SIZE      | 2      |
| GI_FRACTION   | 3      |
| EAC_FLAG      | 1      |
| POLOT_MODE    | 1      |
| PAPR_FLAG     | 1      |
| FRU_CONFIGURE | 3      |
| RESERVED      | 7      |

FIG.13

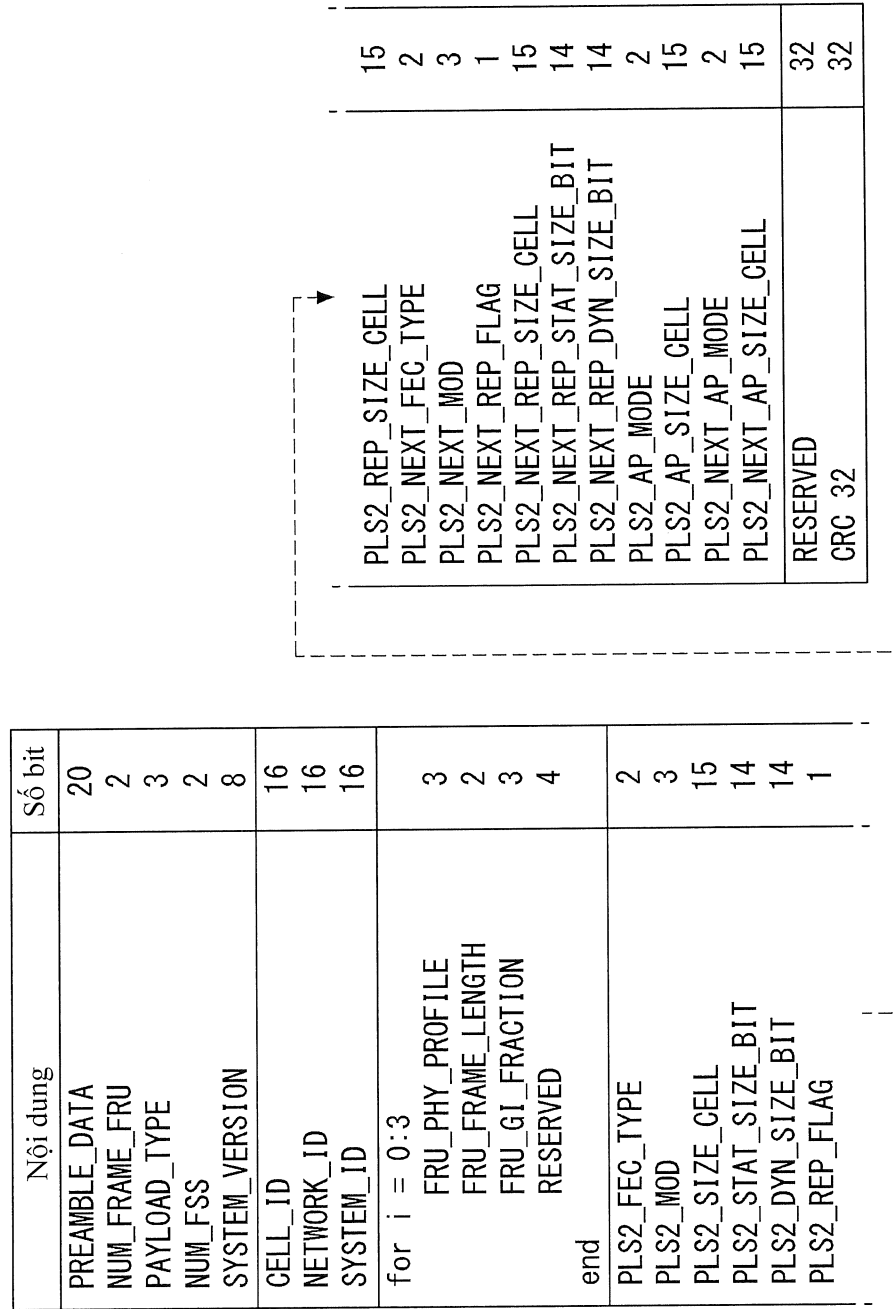


FIG. 14

| Nội dung                     | Bit |
|------------------------------|-----|
| FIC_FLAG                     | 1   |
| AUX_FLAG                     | 1   |
| NUM_DP                       | 6   |
| for i = NUM_DP               |     |
| DP_ID                        | 6   |
| DP_TYPE                      | 3   |
| DP_GROUP_ID                  | 8   |
| BASE_DP_ID                   | 6   |
| DP_FEC_TYPE                  | 2   |
| DP_COD                       | 4   |
| DP_MOD                       | 4   |
| DP_SSD_FLAG                  | 1   |
| if PHY_PROFILE = '010'       |     |
| DP_MIMO                      | 3   |
| end                          |     |
| DP_TI_TYPE                   | 1   |
| DP_TI_LENGTH                 | 2   |
| DP_TI_BYPASS                 | 1   |
| DP_FRAME_INTERVAL            | 2   |
| DP_FIRST_FRAME_IDX           | 5   |
| DP_NUM_BLOCK_MAX             | 10  |
| DP_PAYLOAD_TYPE              | 2   |
| DP_INBAND_MODE               | 2   |
| DP_PROTOCOL_TYPE             | 2   |
| DP_CRC_MODE                  | 2   |
|                              |     |
| if DP_PAYLOAD_TYPE==TS('00') |     |
| DNP_MODE                     | 2   |
| ISSY_MODE                    | 2   |
| HC_MODE_TS                   | 2   |
| if HC_MODE_TS=='01' or '10'  |     |
| PID                          | 13  |
| end                          |     |
| if DP_PAYLOAD_TYPE==IP('01') |     |
| HC_MODE_IP                   | 2   |
| end                          |     |
| RESERVED                     | 8   |
| end                          |     |
| if FIC_FLAG == 1             |     |
| FIC_VERSION                  | 8   |
| FIC_LENGTH_BYTE              | 13  |
| RESERVED                     | 8   |
| end                          |     |
| if AUX_FLAG == 1             |     |
| NUM_AUX                      | 4   |
| AUX_CONFIG_RFU               | 8   |
| for i=1:NUM_AUX              |     |
| AUX_STREAM_TYPE              | 4   |
| AUX_PRIVATE_CONF             | 28  |
| end                          |     |
| end                          |     |

FIG. 15

| Nội dung                     | Bit          |
|------------------------------|--------------|
| FRAME_INDEX                  | 5            |
| PLS_CHANGE_COUNTER           | 4            |
| FIC_CHANGE_COUNTER           | 4            |
| RESERVED<br>for i = 1:NUM_DP | 16           |
| DP_ID                        | 6            |
| DP_START                     | 15 (hoặc 13) |
| DP_NUM_BLOCK                 | 10           |
| RESERVED                     | 8            |
| end                          |              |
| EAC_FLAG                     | 1            |
| EAS_WAKE_UP_VERSION_NUM      | 8            |
| if EAC_FLAG == 1             |              |
| EAC_LENGTH_BYTE              | 12           |
| else                         |              |
| EAC_COUNTER                  | 12           |
| end                          |              |
| for i = 1:NUM_AUX            |              |
| AUX_PRIVATE_DYN              | 48           |
| end                          |              |
| CRC 32                       | 32           |

FIG. 16

|          |      |      |     |     |        |                     |     |
|----------|------|------|-----|-----|--------|---------------------|-----|
| Đoạn đầu | PLS1 | PLS2 | EAC | FIC | Các DP | Các dòng<br>phụ trợ | Giả |
|----------|------|------|-----|-----|--------|---------------------|-----|

FIG. 17

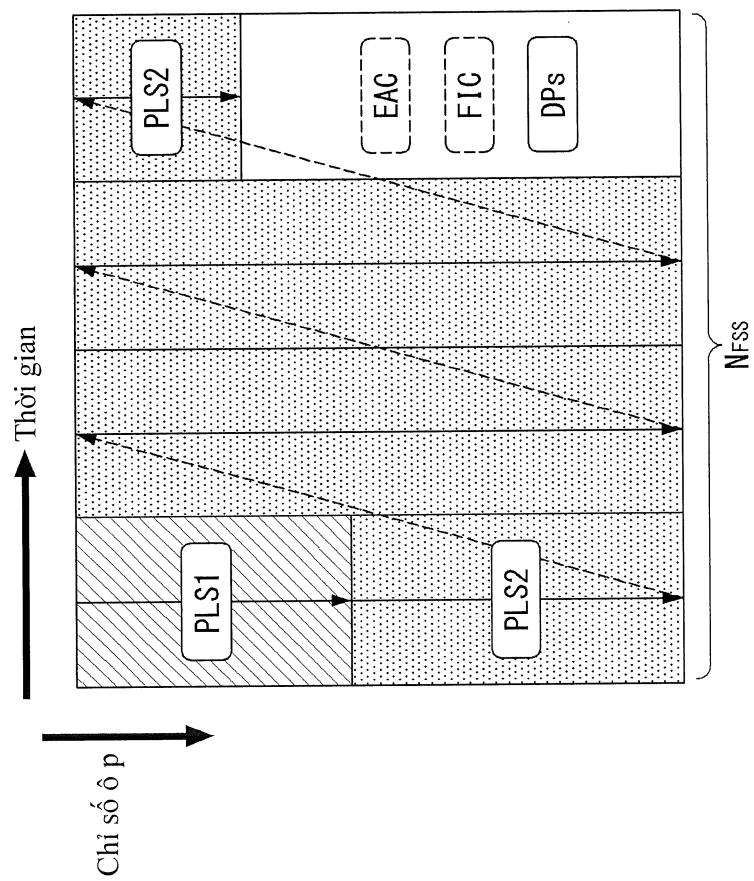


FIG. 18

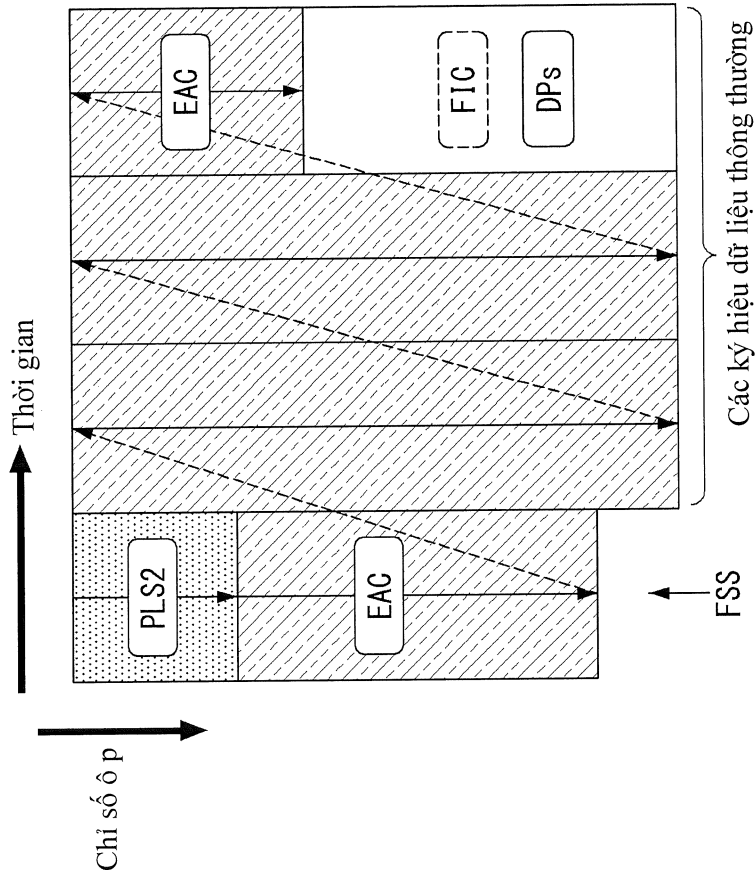
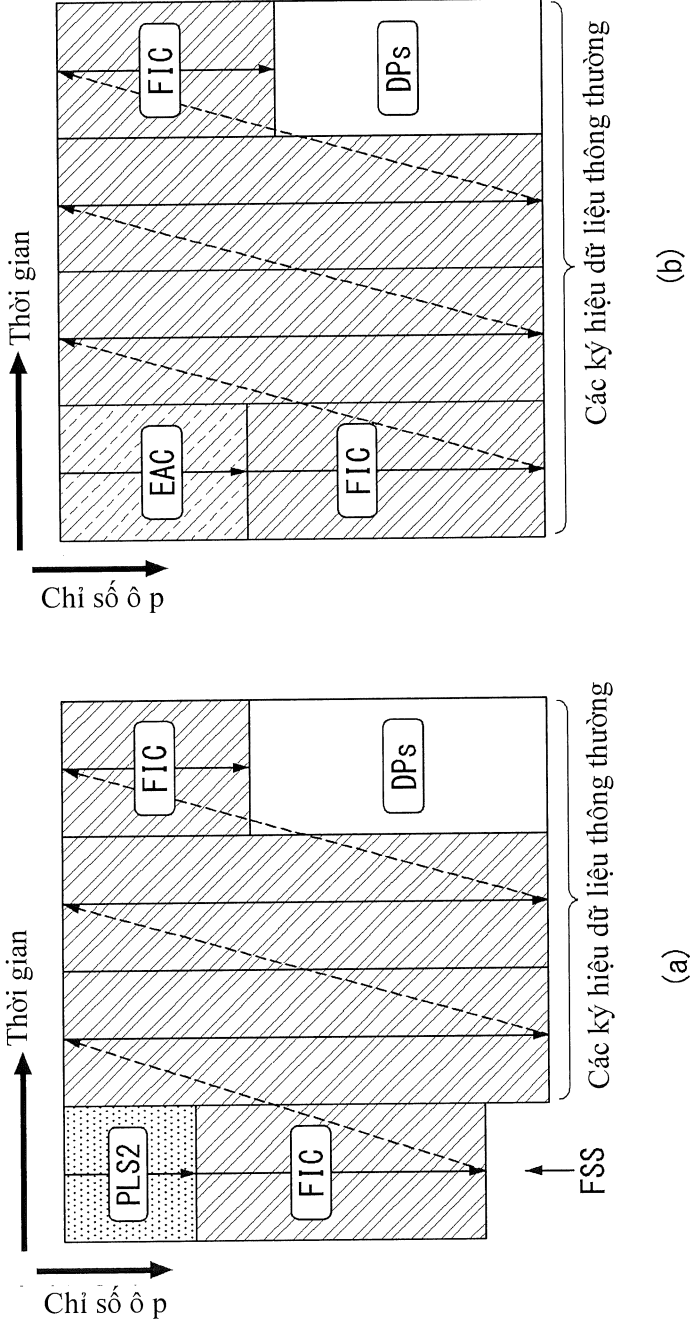


FIG. 19



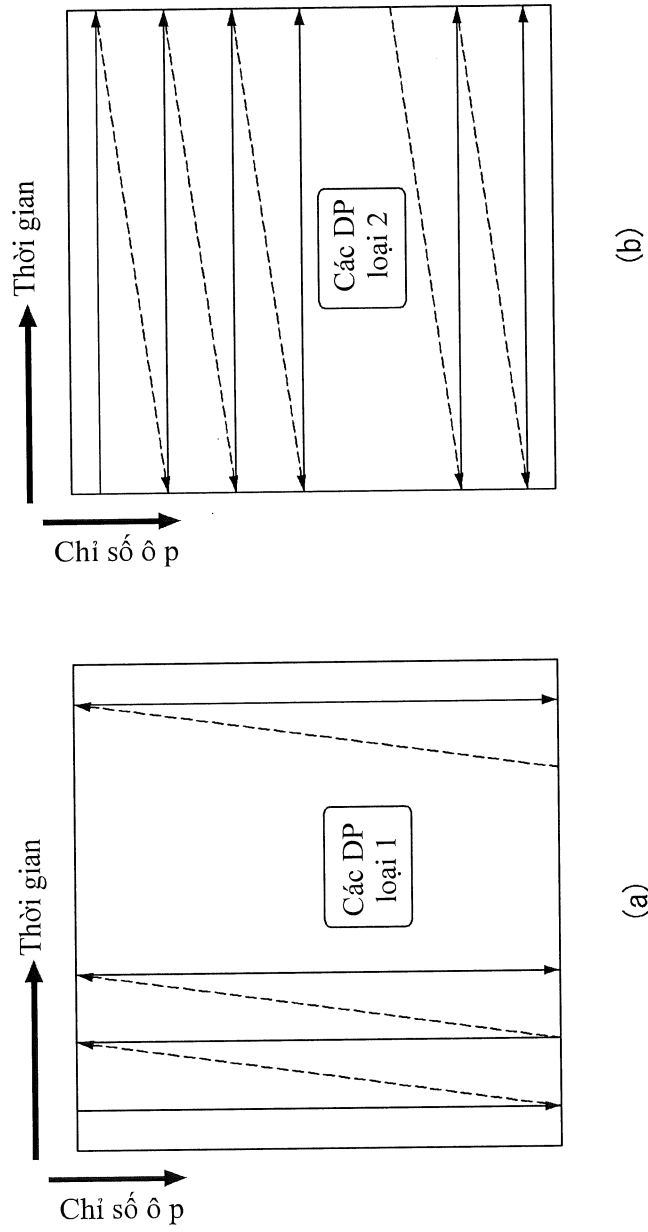


FIG. 20

FIG. 21

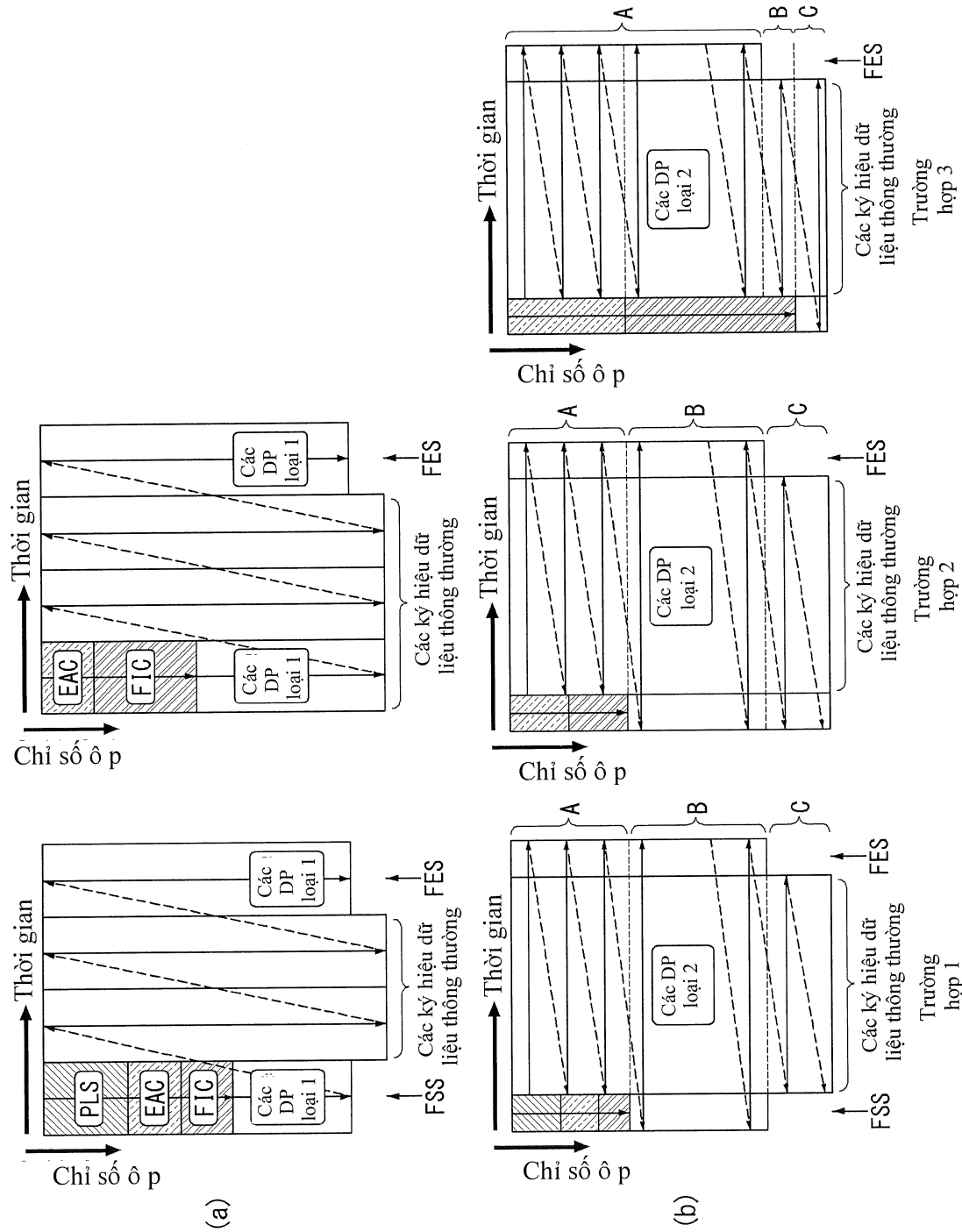


FIG. 22

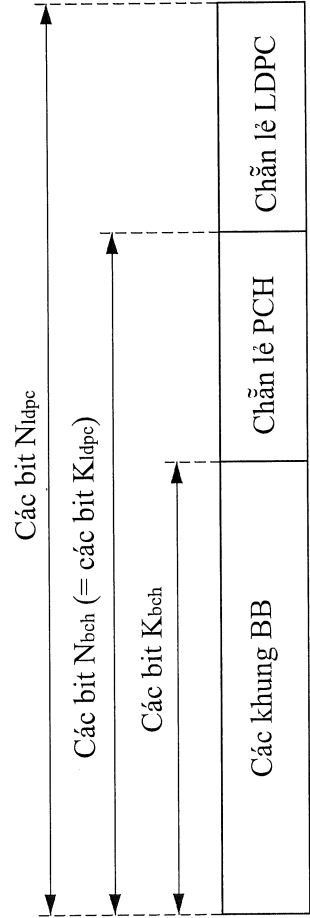


FIG. 23

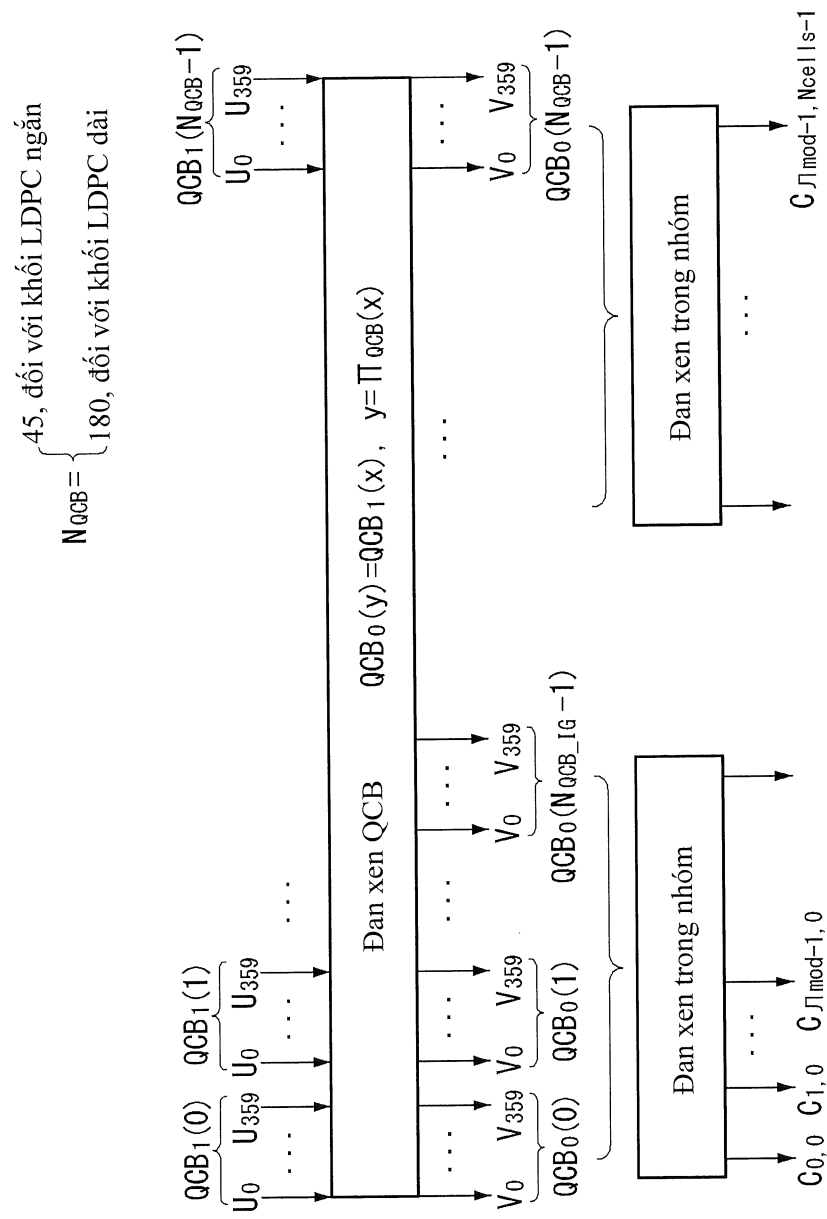


FIG. 24

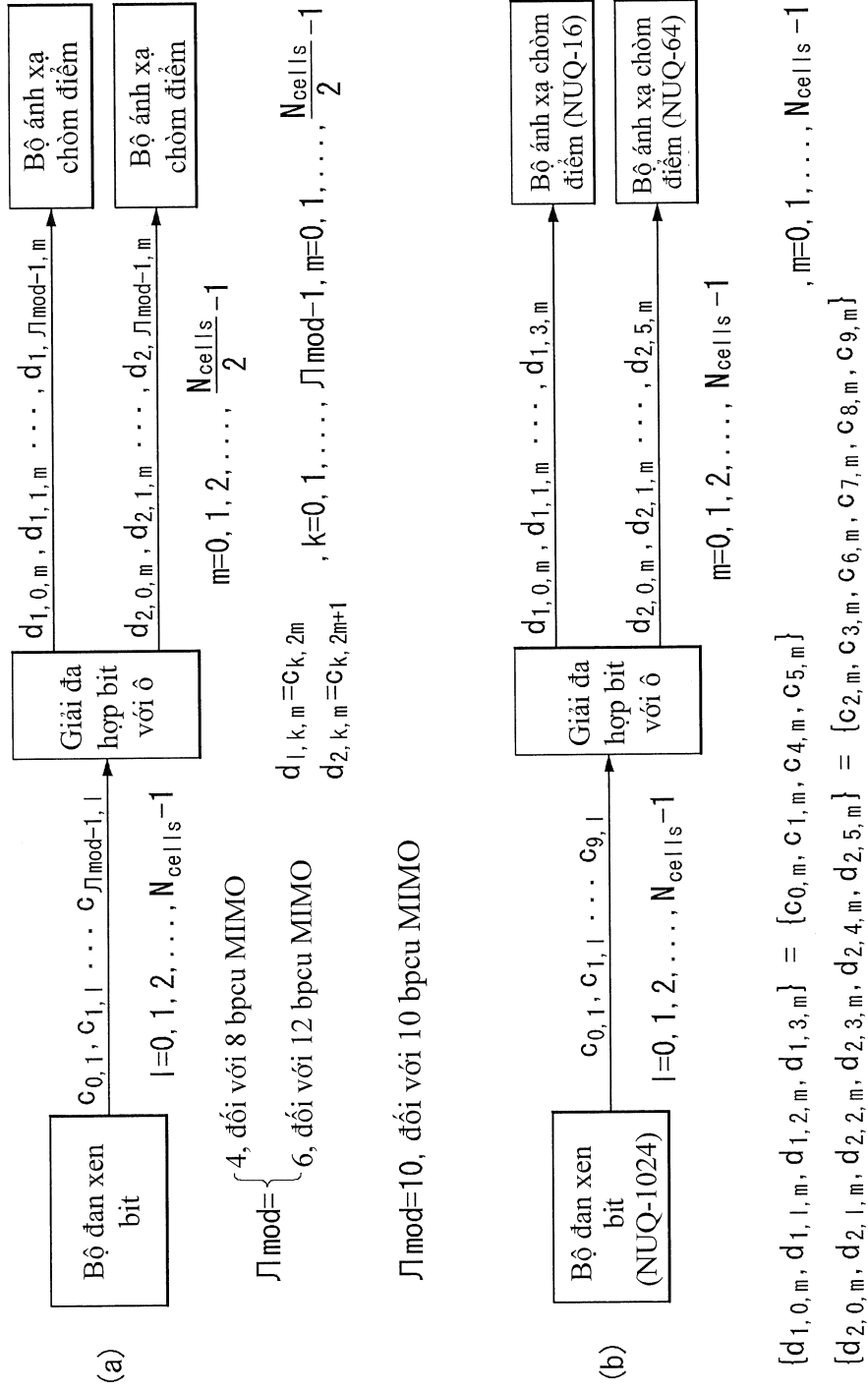


FIG. 25

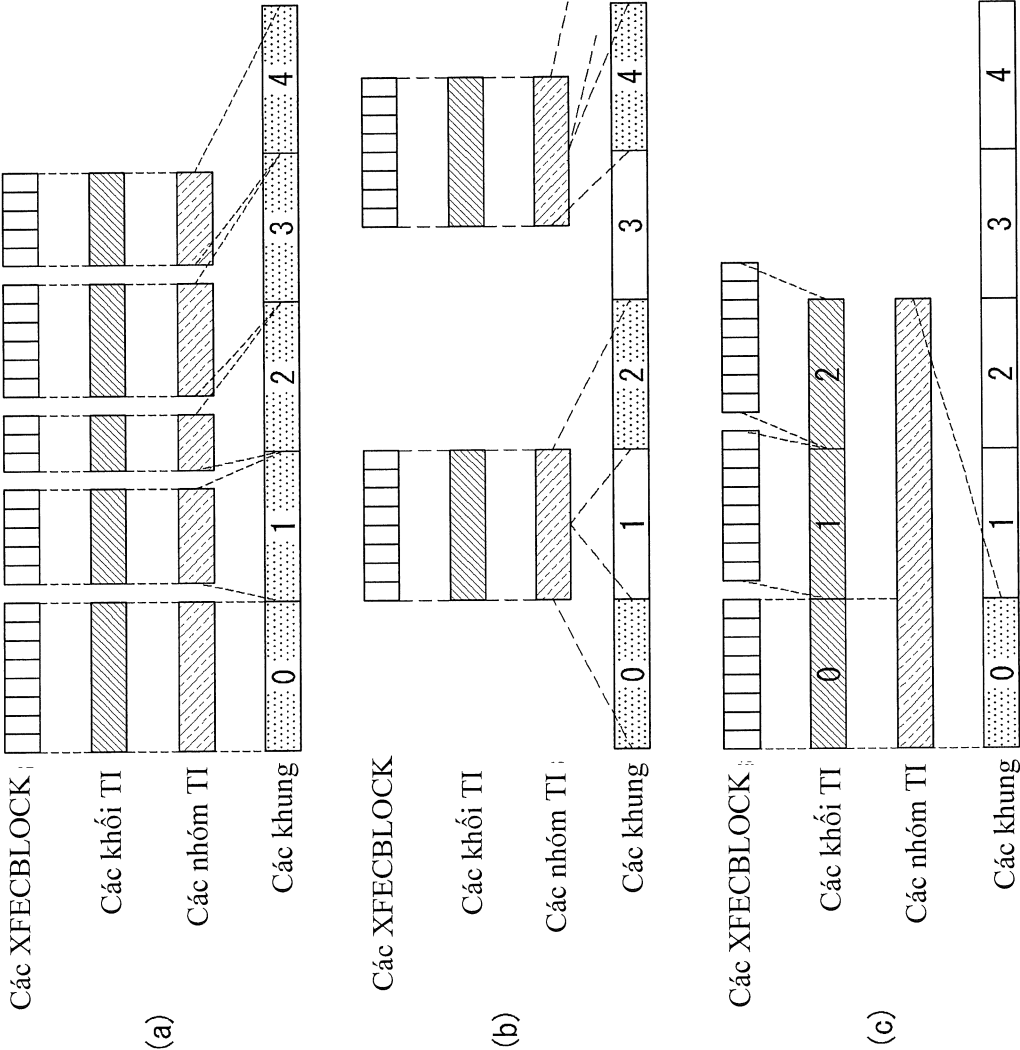


FIG. 26

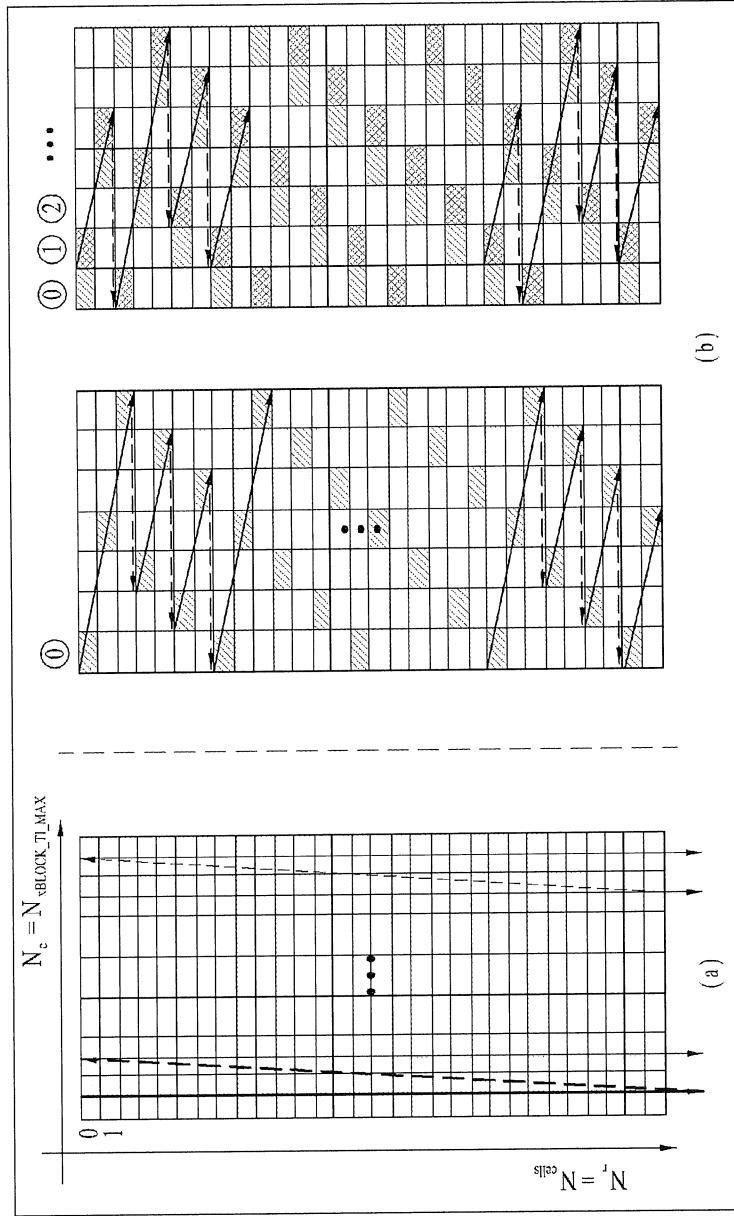


FIG. 27

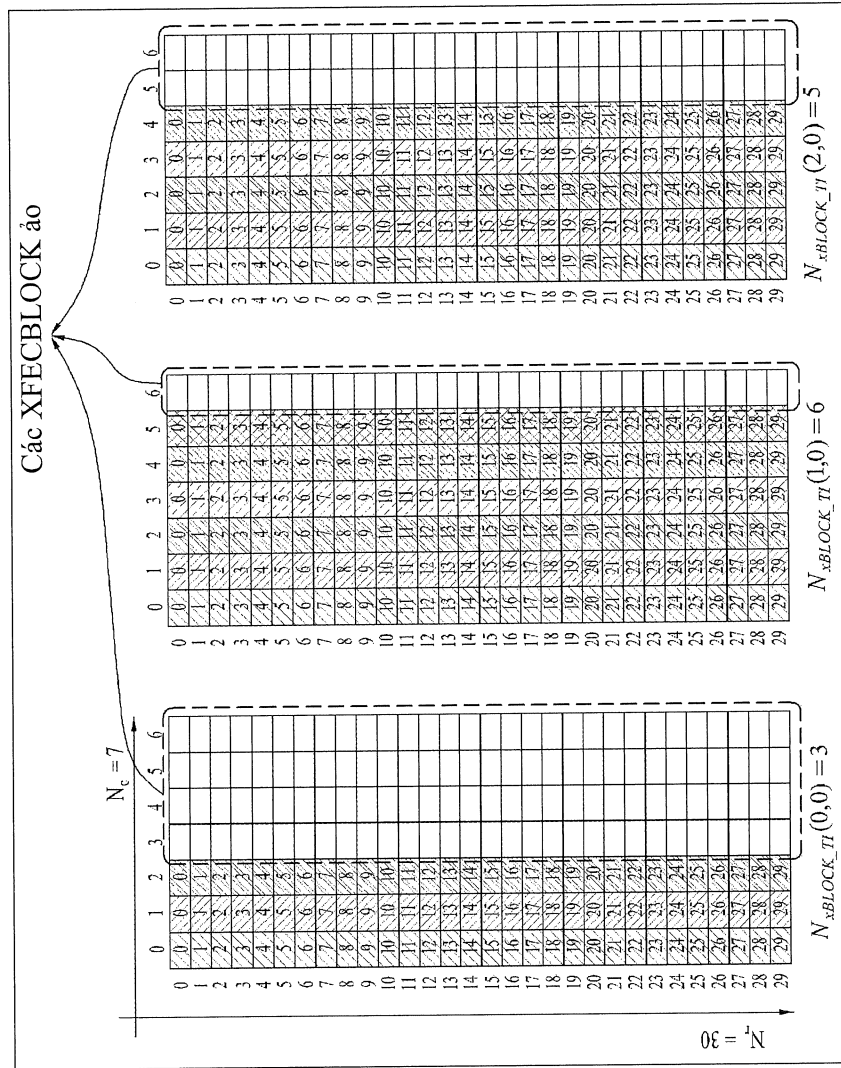


FIG. 28

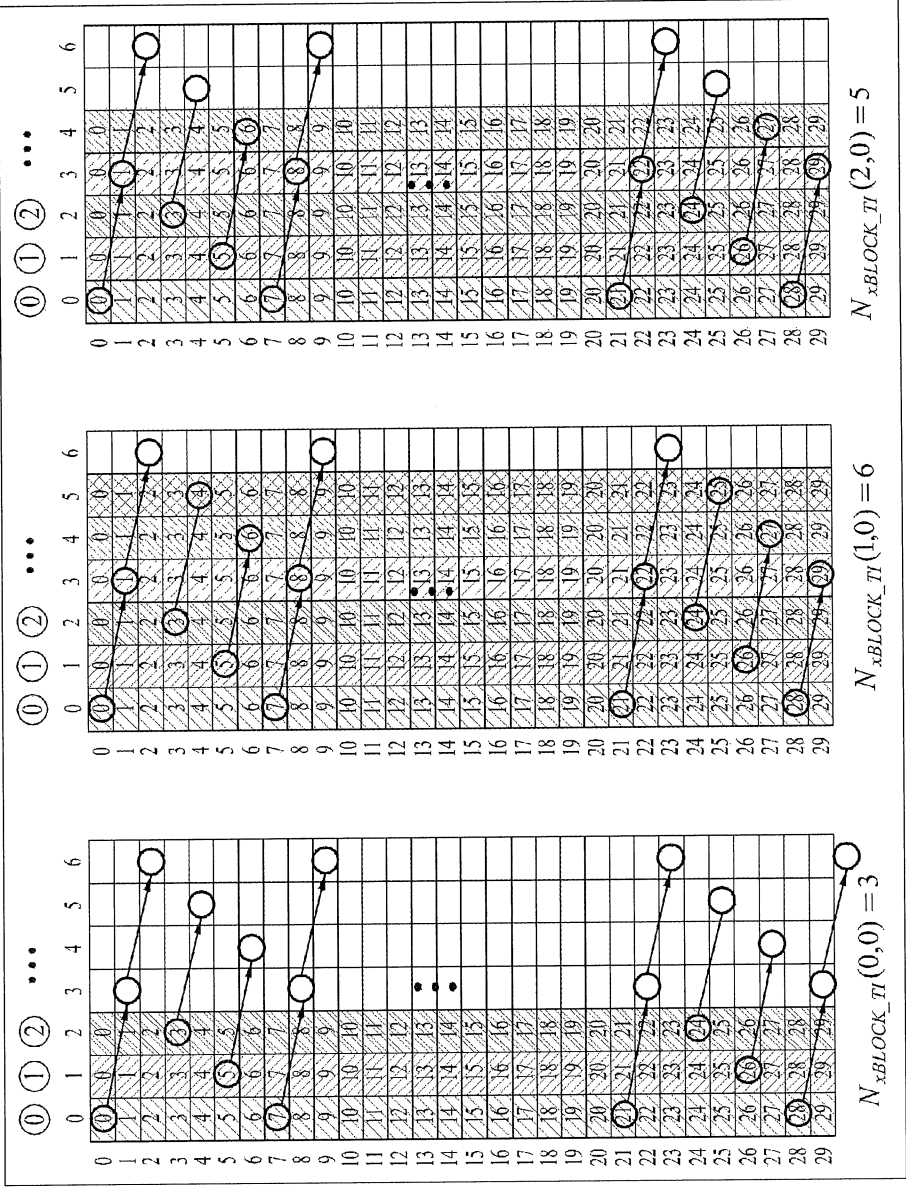


FIG. 29

| $N_{xBLOCK\_TI}(0,0)=3$ |    |    | $N_{xBLOCK\_TI}(1,0)=6$ |    |    |    |    | $N_{xBLOCK\_TI}(2,0)=5$ |    |    |    |    |    |    |
|-------------------------|----|----|-------------------------|----|----|----|----|-------------------------|----|----|----|----|----|----|
| 0                       | 1  | 2  | 0                       | 1  | 2  | 3  | 4  | 5                       | 0  | 1  | 2  | 3  | 4  |    |
| 0                       | 0  | 22 | 0                       | 5  | 9  | 14 | 20 | 24                      | 0  | 0  | 10 | 23 | 6  | 17 |
| 1                       | 3  | 1  | 1                       | 6  | 10 | 16 | 21 | 25                      | 1  | 1  | 12 | 24 | 7  | 18 |
| 2                       | 5  | 4  | 3                       | 7  | 11 | 17 | 22 | 27                      | 2  | 3  | 4  | 25 | 8  | 20 |
| 3                       | 7  | 16 | 2                       | 8  | 12 | 18 | 23 | 28                      | 3  | 5  | 15 | 26 | 9  | 22 |
| 4                       | 10 | 18 | 4                       | 9  | 14 | 19 | 25 | 29                      | 4  | 6  | 16 | 28 | 11 | 23 |
| 5                       | 12 | 21 | 6                       | 10 | 15 | 20 | 26 | 1                       | 5  | 7  | 17 | 0  | 13 | 24 |
| 6                       | 14 | 33 | 7                       | 12 | 16 | 21 | 27 | 2                       | 6  | 8  | 19 | 2  | 14 | 25 |
| 7                       | 17 | 25 | 8                       | 13 | 17 | 23 | 28 | 3                       | 7  | 10 | 21 | 4  | 15 | 27 |
| 8                       | 19 | 28 | 10                      | 14 | 18 | 24 | 29 | 4                       | 8  | 12 | 22 | 5  | 16 | 29 |
| 9                       | 21 | 1  | 13                      | 15 | 19 | 25 | 0  | 5                       | 9  | 13 | 23 | 6  | 18 | 1  |
| 10                      | 24 | 4  | 15                      | 16 | 21 | 26 | 1  | 6                       | 10 | 14 | 24 | 7  | 20 | 3  |
| 11                      | 36 | 5  | 17                      | 17 | 23 | 27 | 2  | 8                       | 11 | 15 | 26 | 9  | 21 | 4  |
| 12                      | 28 | 9  | 20                      | 18 | 23 | 28 | 3  | 9                       | 13 | 17 | 28 | 11 | 22 | 5  |
| 13                      | 0  | 11 | 22                      | 19 | 24 | 0  | 4  | 10                      | 15 | 19 | 29 | 12 | 23 | 6  |
| 14                      | 7  | 13 | 24                      | 20 | 24 | 5  | 9  | 15                      | 17 | 21 | 0  | 13 | 25 | 8  |
| 15                      | 5  | 16 | 27                      | 21 | 25 | 1  | 6  | 11                      | 18 | 22 | 2  | 14 | 27 | 10 |
| 16                      | 1  | 18 | 29                      | 22 | 26 | 2  | 7  | 12                      | 19 | 23 | 3  | 16 | 28 | 11 |
| 17                      | 9  | 20 | 1                       | 23 | 28 | 4  | 8  | 13                      | 20 | 24 | 4  | 18 | 29 | 12 |
| 18                      | 12 | 23 | 3                       | 24 | 29 | 5  | 9  | 15                      | 21 | 26 | 5  | 19 | 1  | 13 |
| 19                      | 14 | 35 | 5                       | 26 | 0  | 6  | 10 | 16                      | 22 | 27 | 7  | 20 | 2  | 15 |
| 20                      | 16 | 27 | 8                       | 27 | 2  | 7  | 11 | 17                      | 24 | 28 | 9  | 21 | 3  | 17 |
| 21                      | 19 | 1  | 10                      | 28 | 3  | 8  | 13 | 18                      | 25 | 29 | 10 | 23 | 4  | 18 |
| 22                      | 21 | 4  | 12                      | 29 | 4  | 9  | 14 | 19                      | 26 | 0  | 11 | 25 | 6  | 19 |
| 23                      | 23 | 6  | 15                      | 0  | 5  | 11 | 15 | 20                      | 27 | 1  | 12 | 26 | 8  | 20 |
| 24                      | 24 | 8  | 17                      | 6  | 12 | 16 | 22 | 23                      | 28 | 2  | 14 | 27 | 9  | 22 |
| 25                      | 28 | 11 | 19                      | 7  | 13 | 17 | 23 | 24                      | 29 | 3  | 16 | 28 | 10 | 24 |
| 26                      | 0  | 3  | 22                      | 9  | 14 | 18 | 24 | 25                      | 0  | 4  | 10 | 15 | 20 | 25 |
| 27                      | 1  | 15 | 24                      | 10 | 15 | 20 | 25 | 26                      | 1  | 5  | 11 | 16 | 21 | 26 |
| 28                      | 4  | 18 | 26                      | 11 | 16 | 21 | 26 | 27                      | 2  | 7  | 12 | 18 | 22 | 27 |
| 29                      | 7  | 20 | 29                      | 12 | 18 | 22 | 27 | 29                      | 3  | 8  | 13 | 19 | 23 | 29 |

FIG. 30

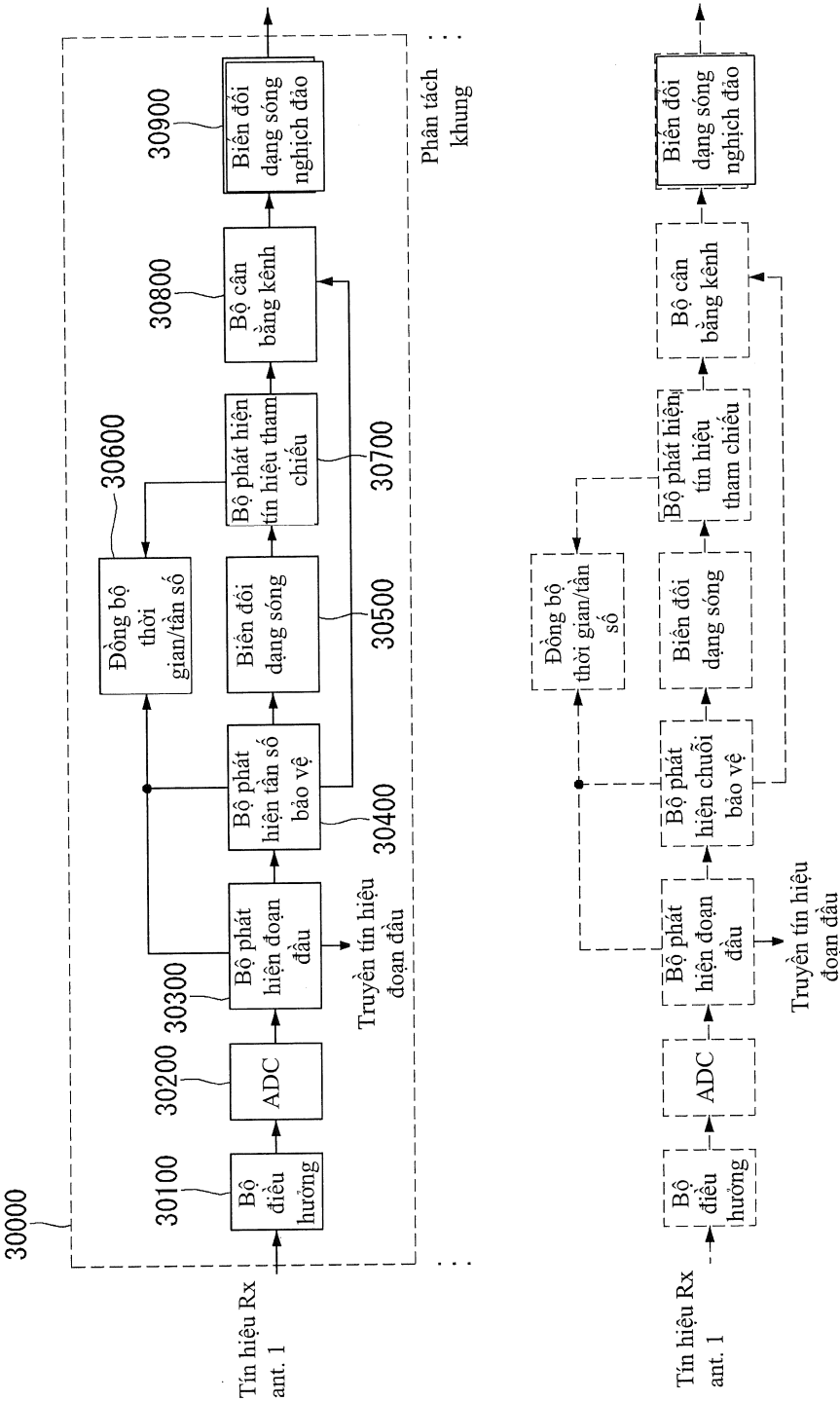


FIG. 31

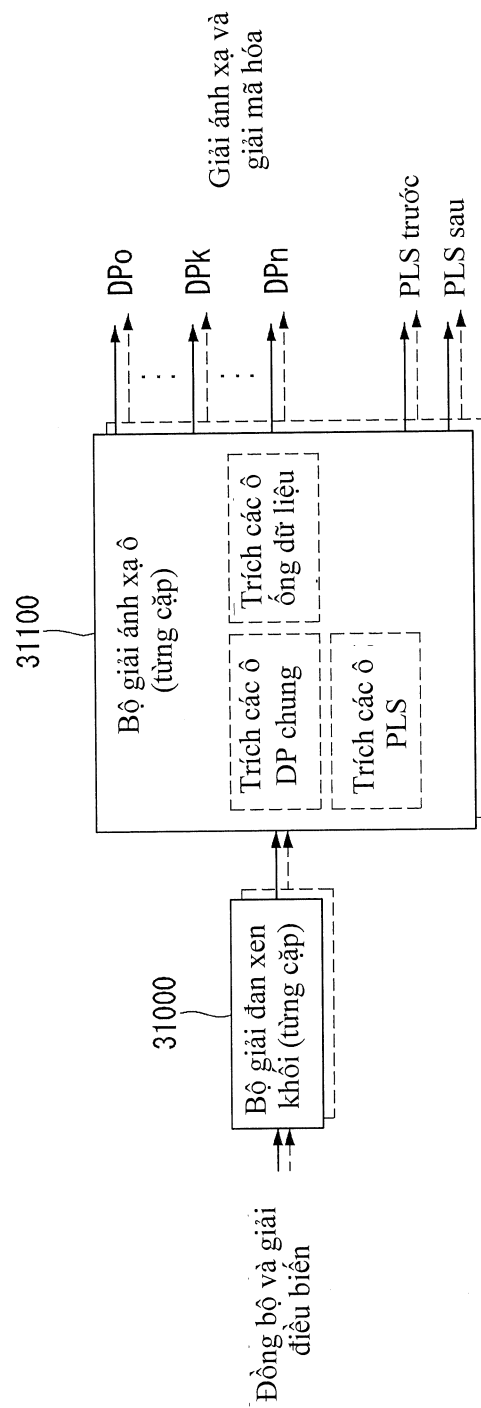


FIG. 32A

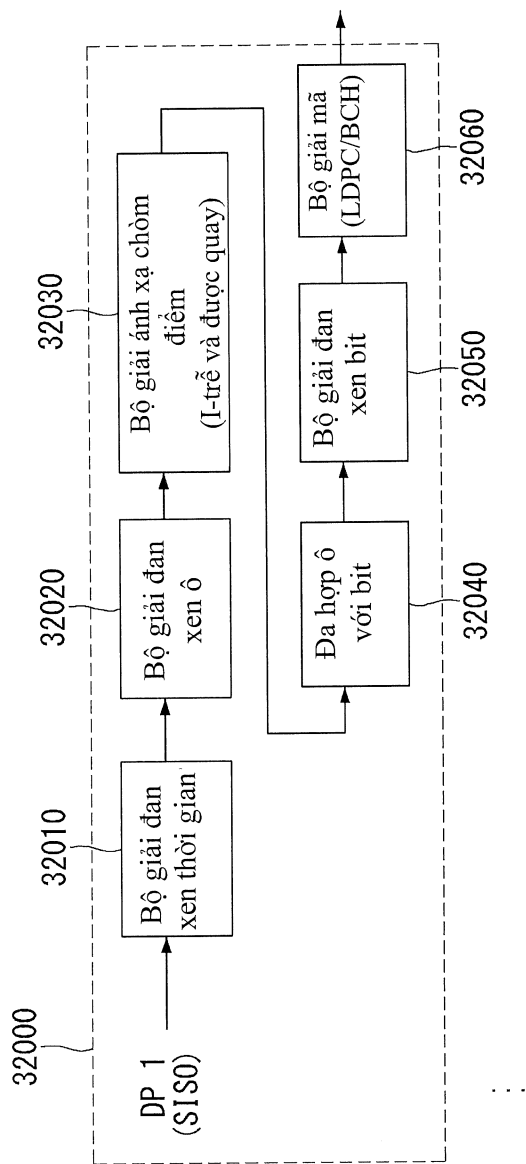


FIG. 32B

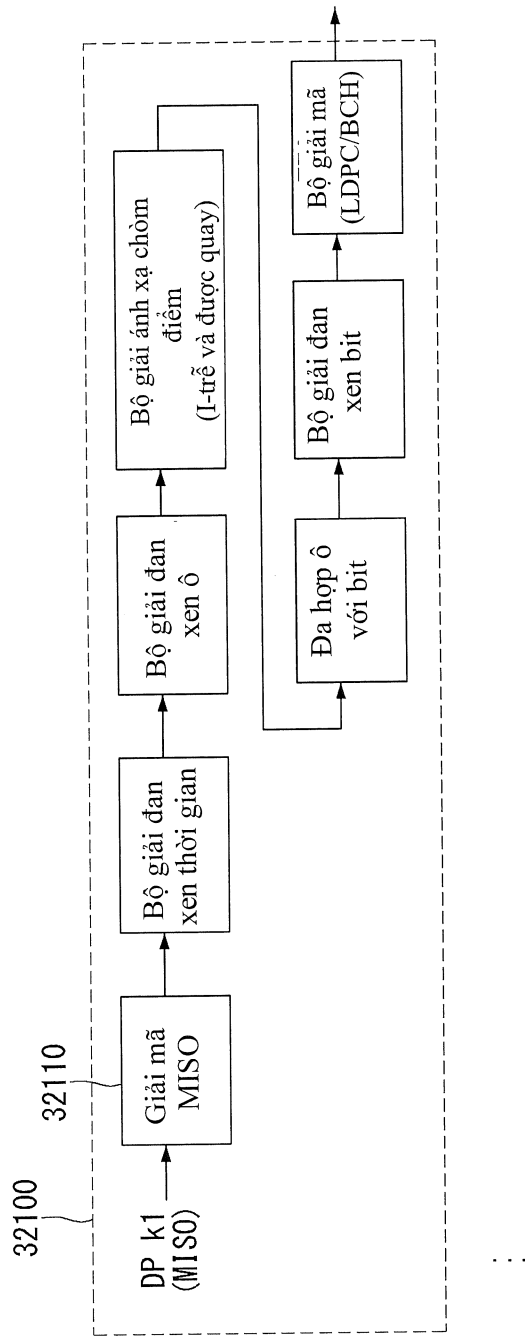


FIG. 32C

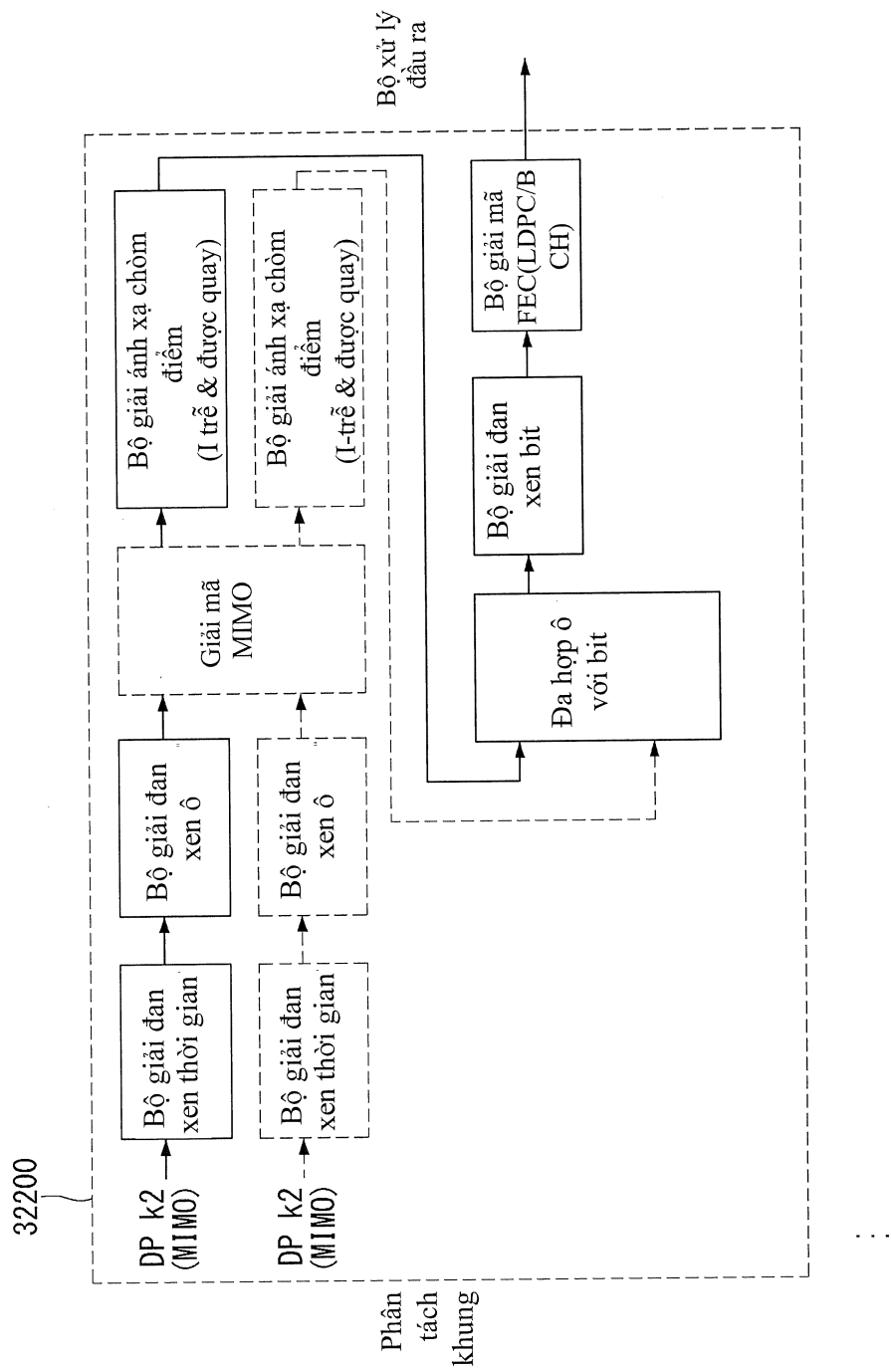


FIG. 32D

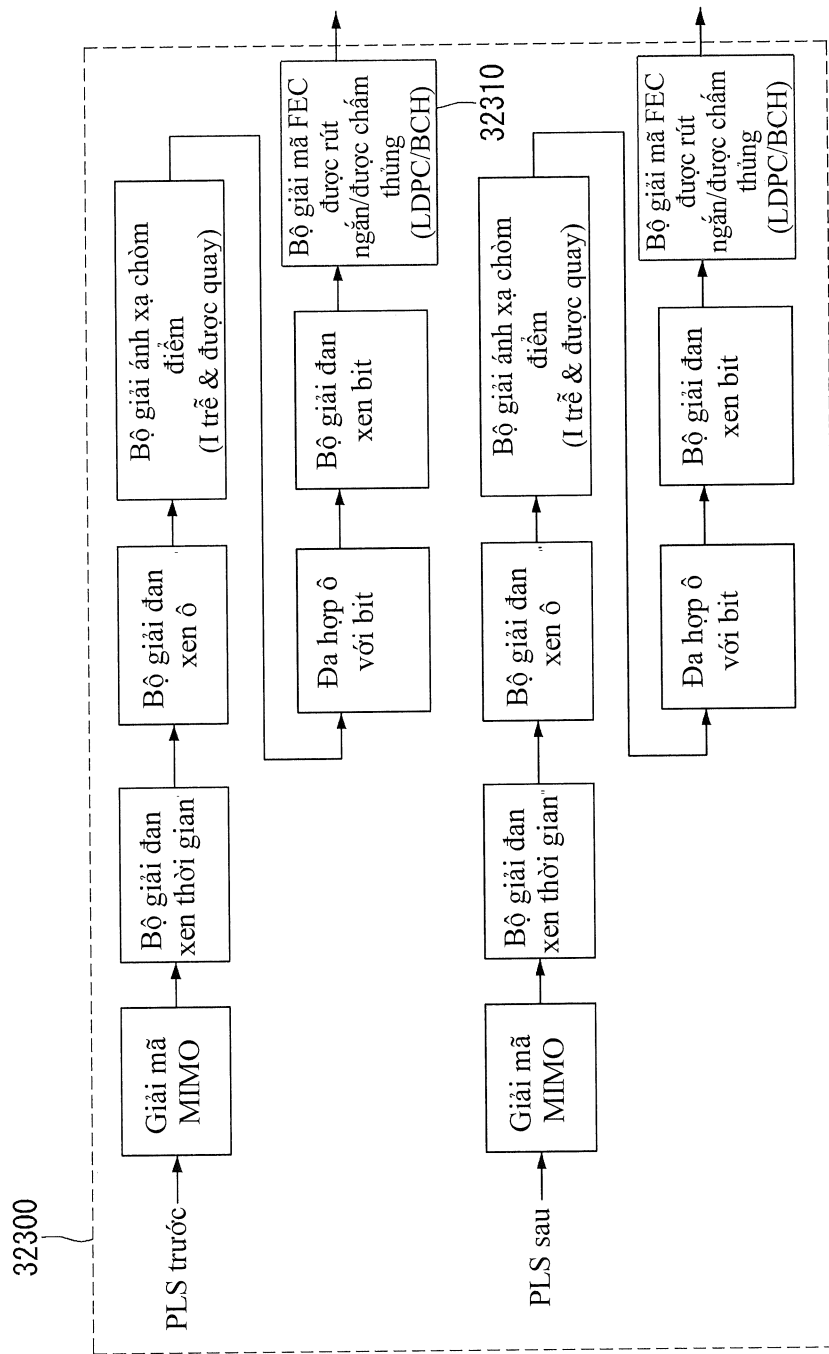


FIG. 33

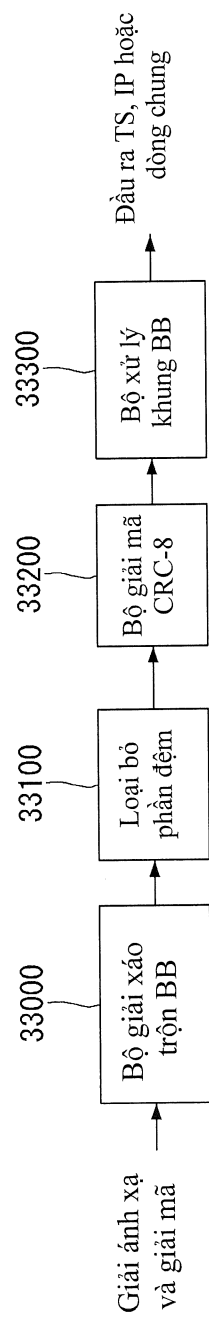


FIG. 34a

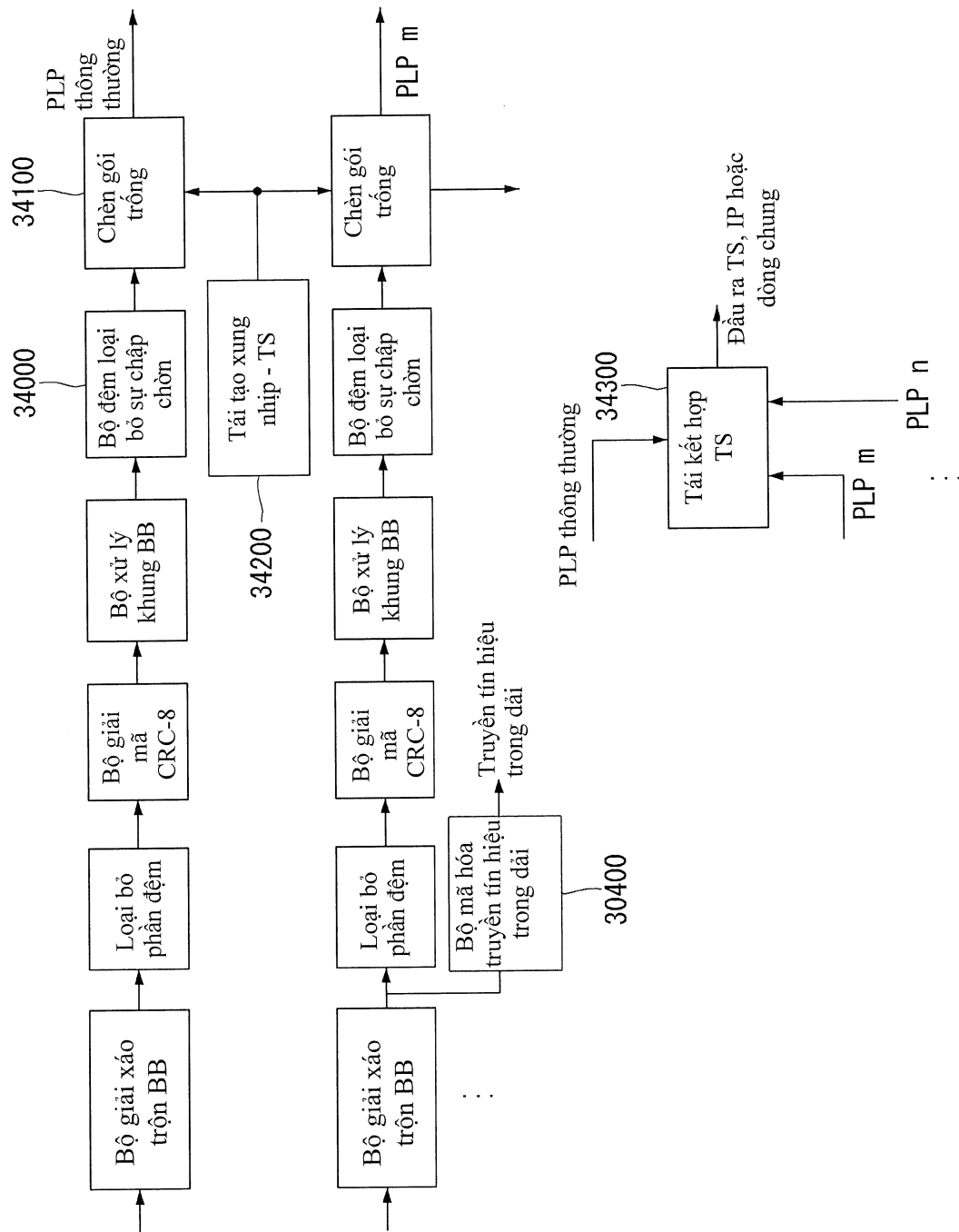


FIG. 34b

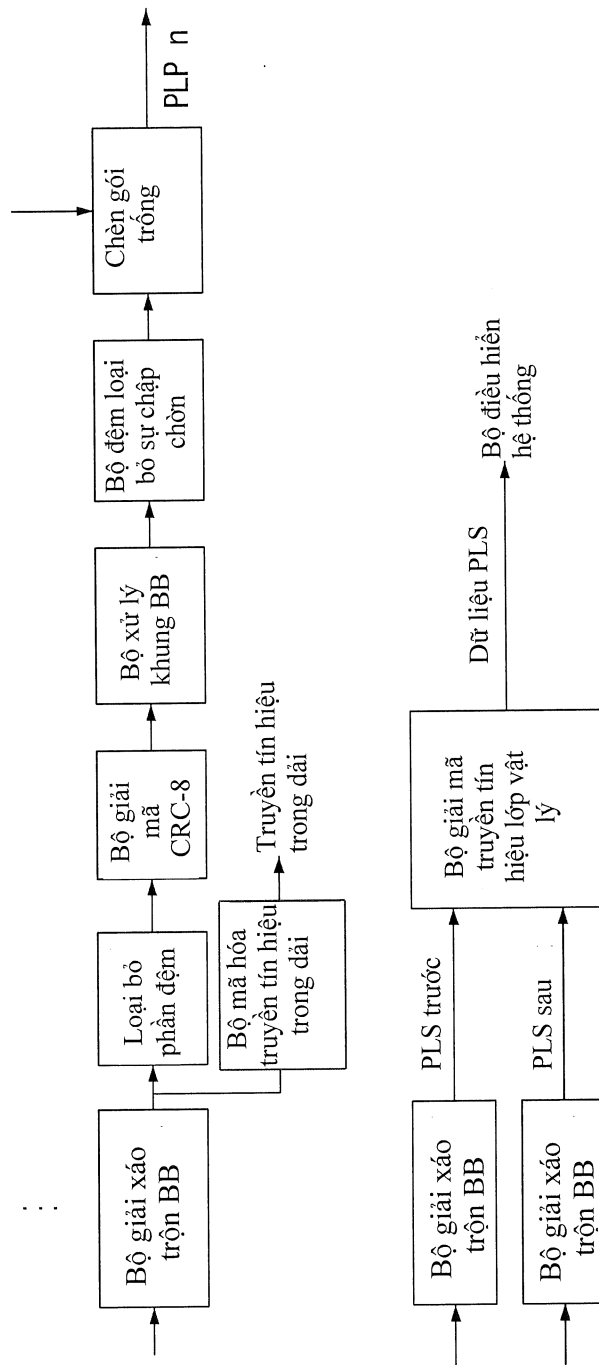


FIG. 35a

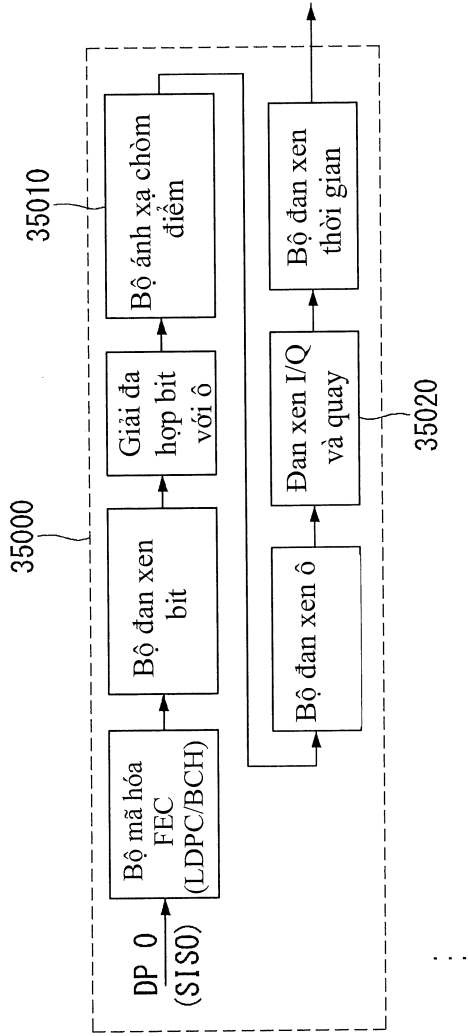


FIG. 35b

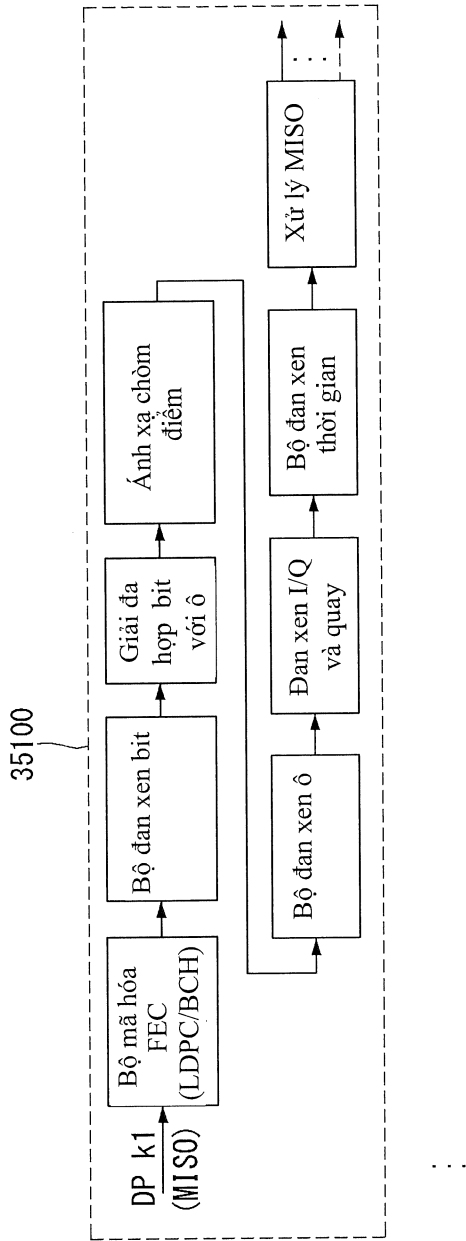


FIG. 35c

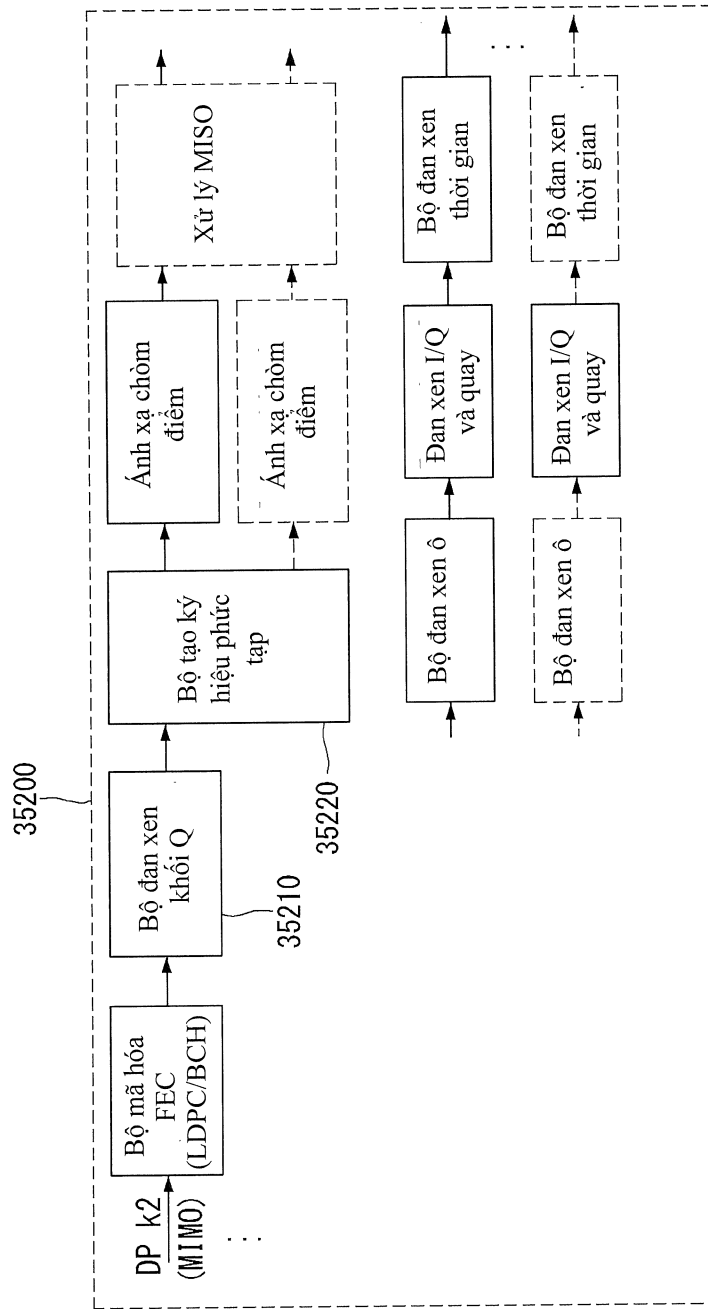


FIG. 35d

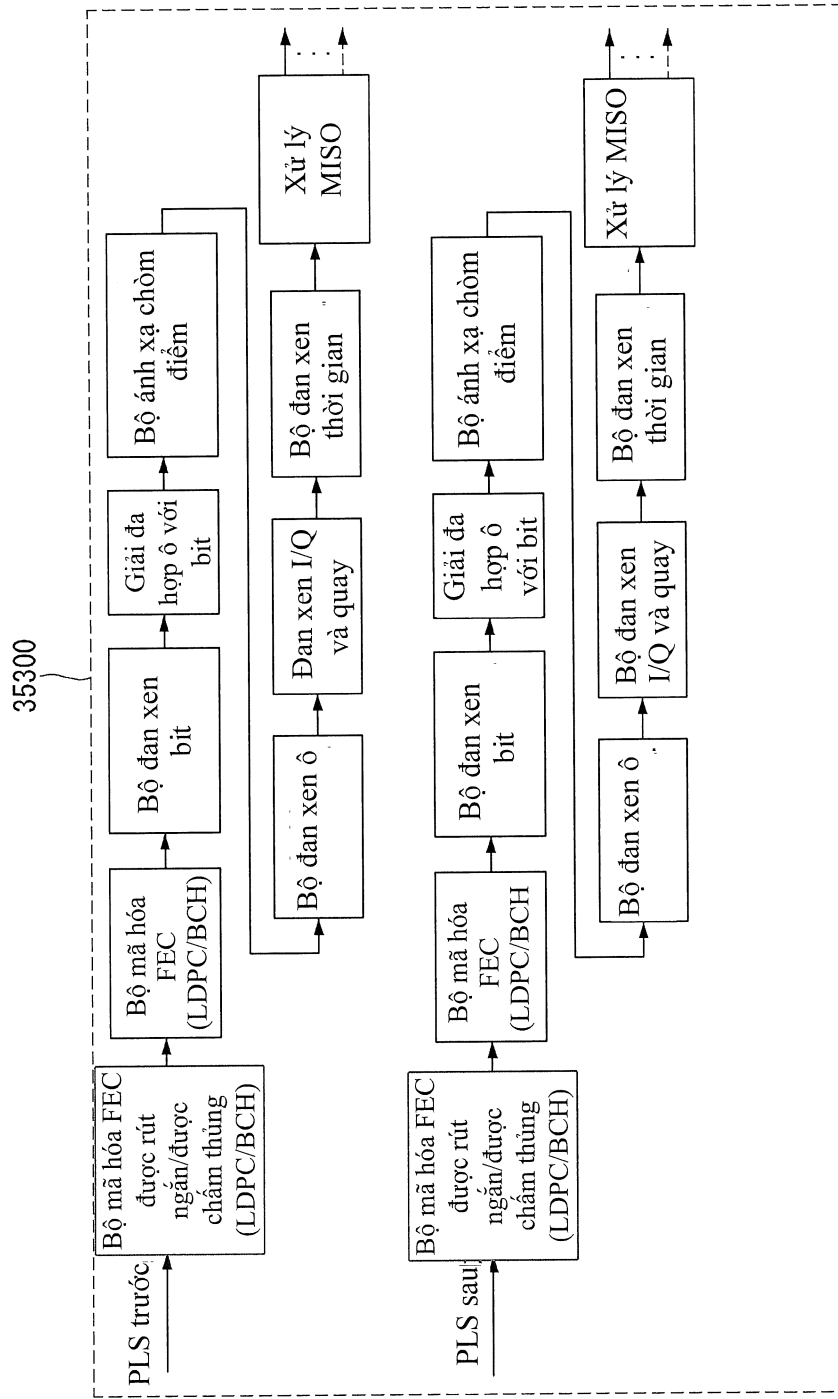


FIG. 36a

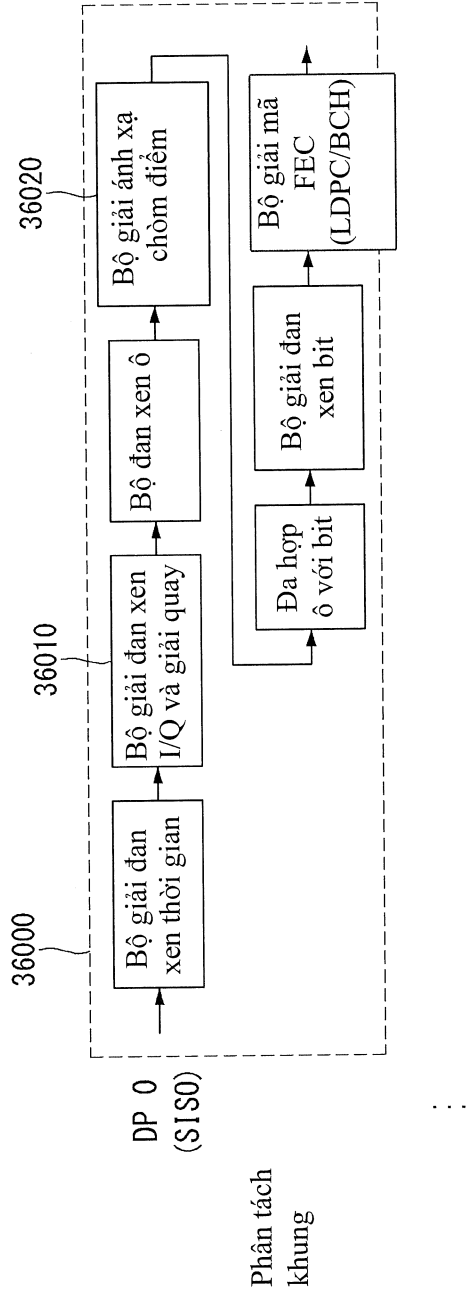


FIG. 36b

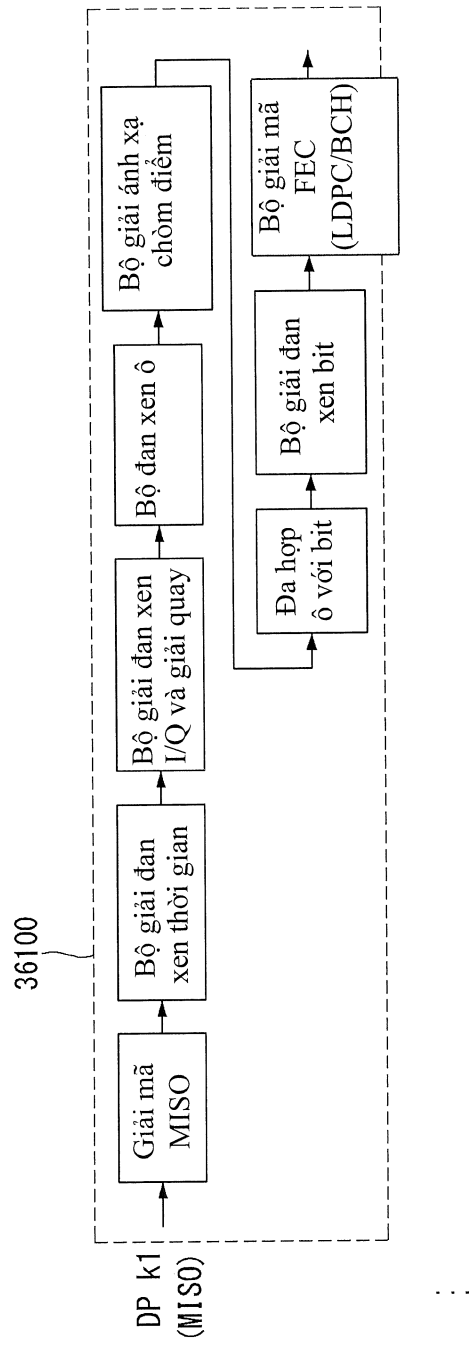


FIG. 36c

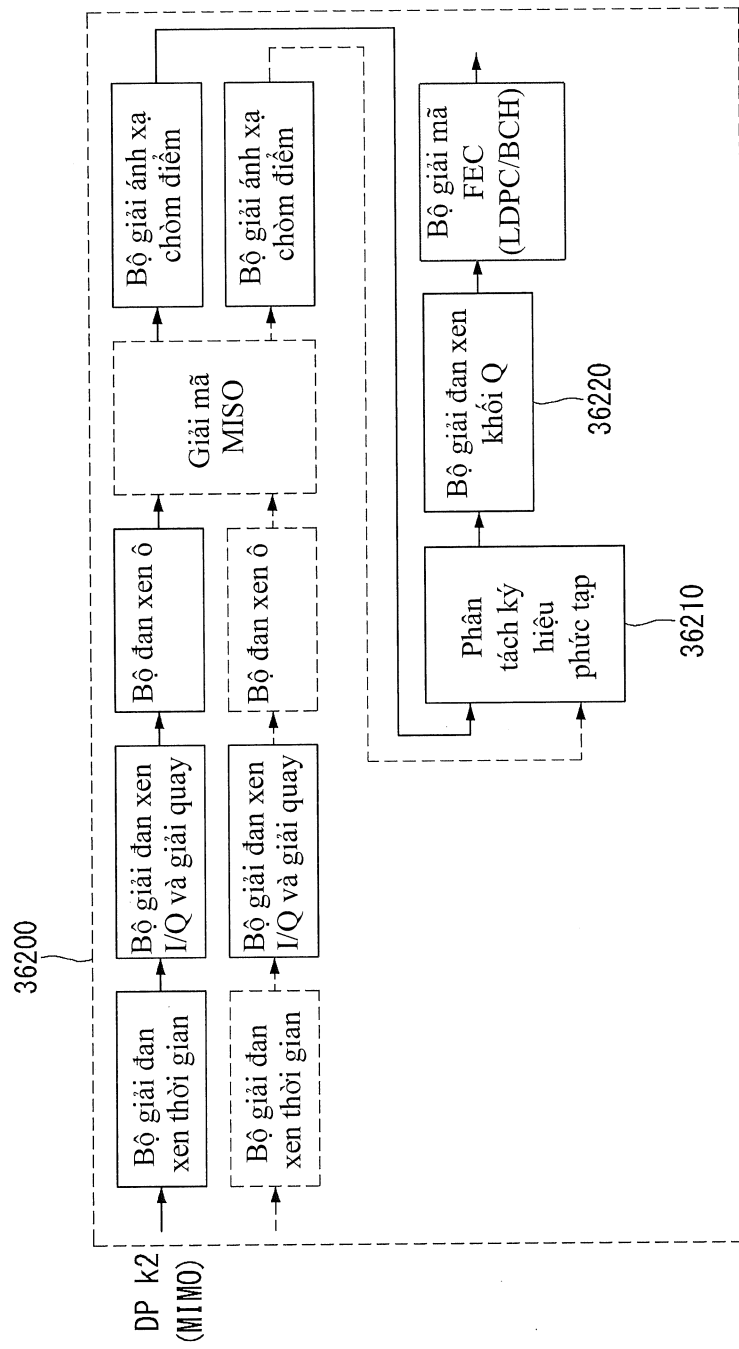


FIG. 36d

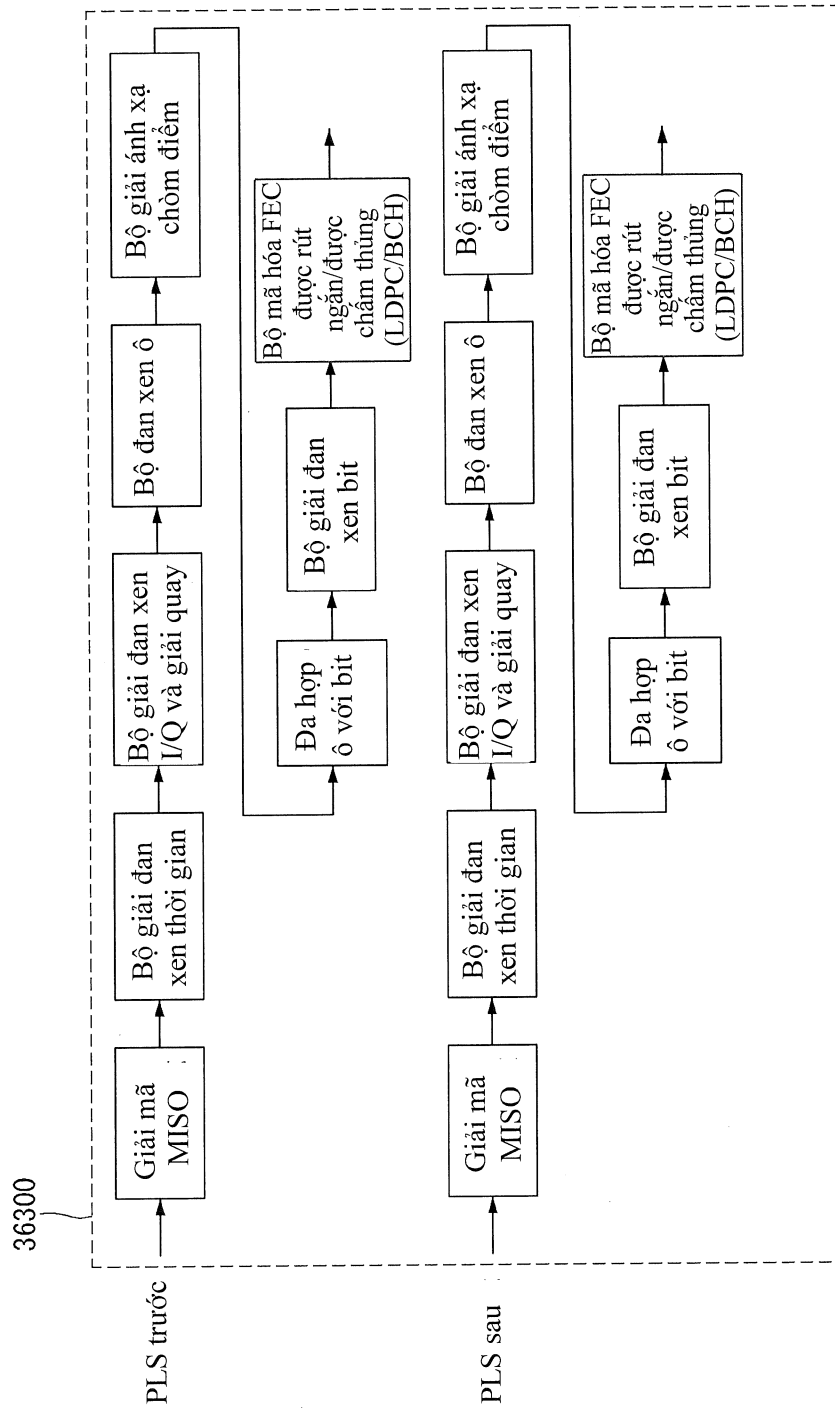


FIG. 37

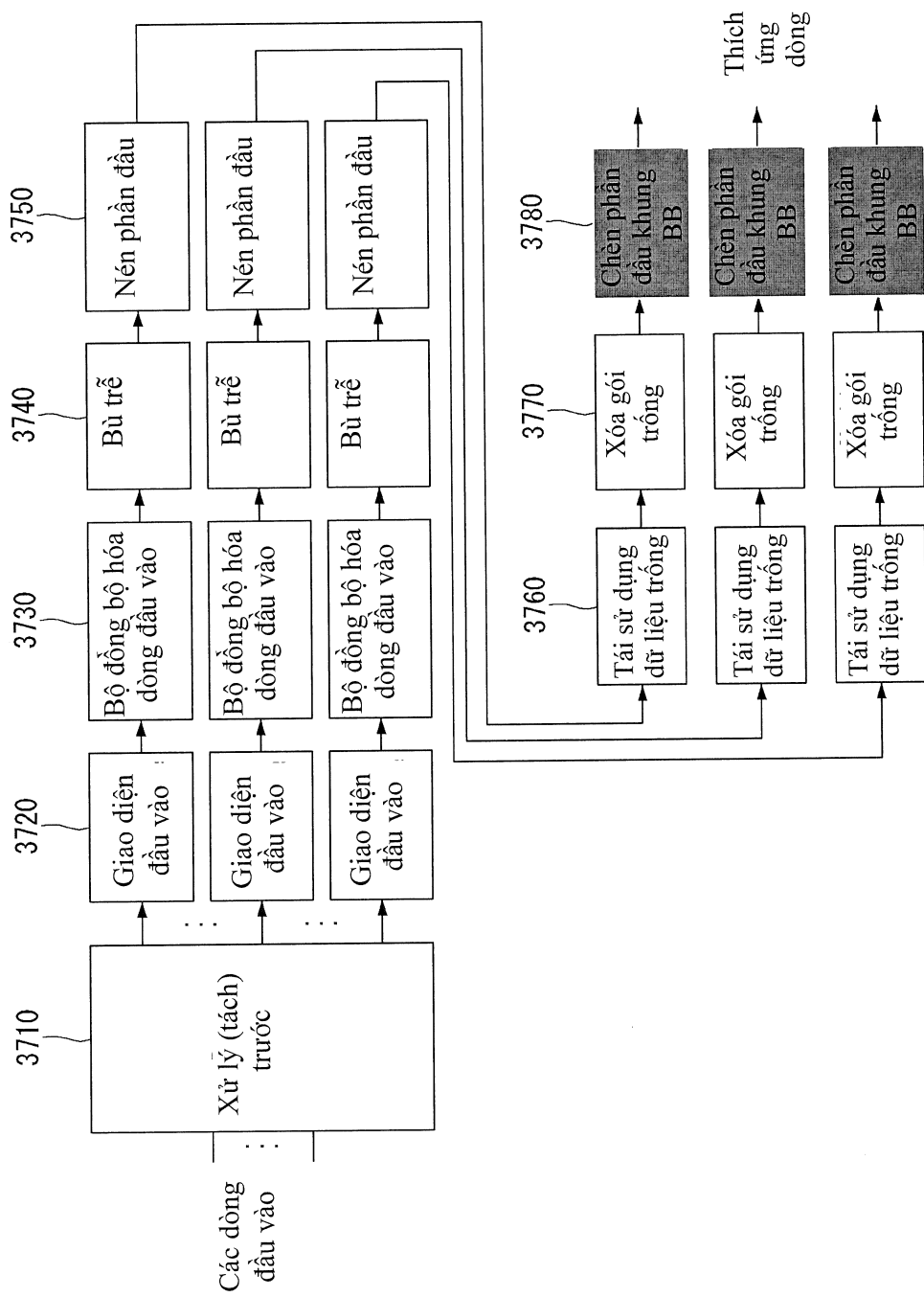


FIG. 38

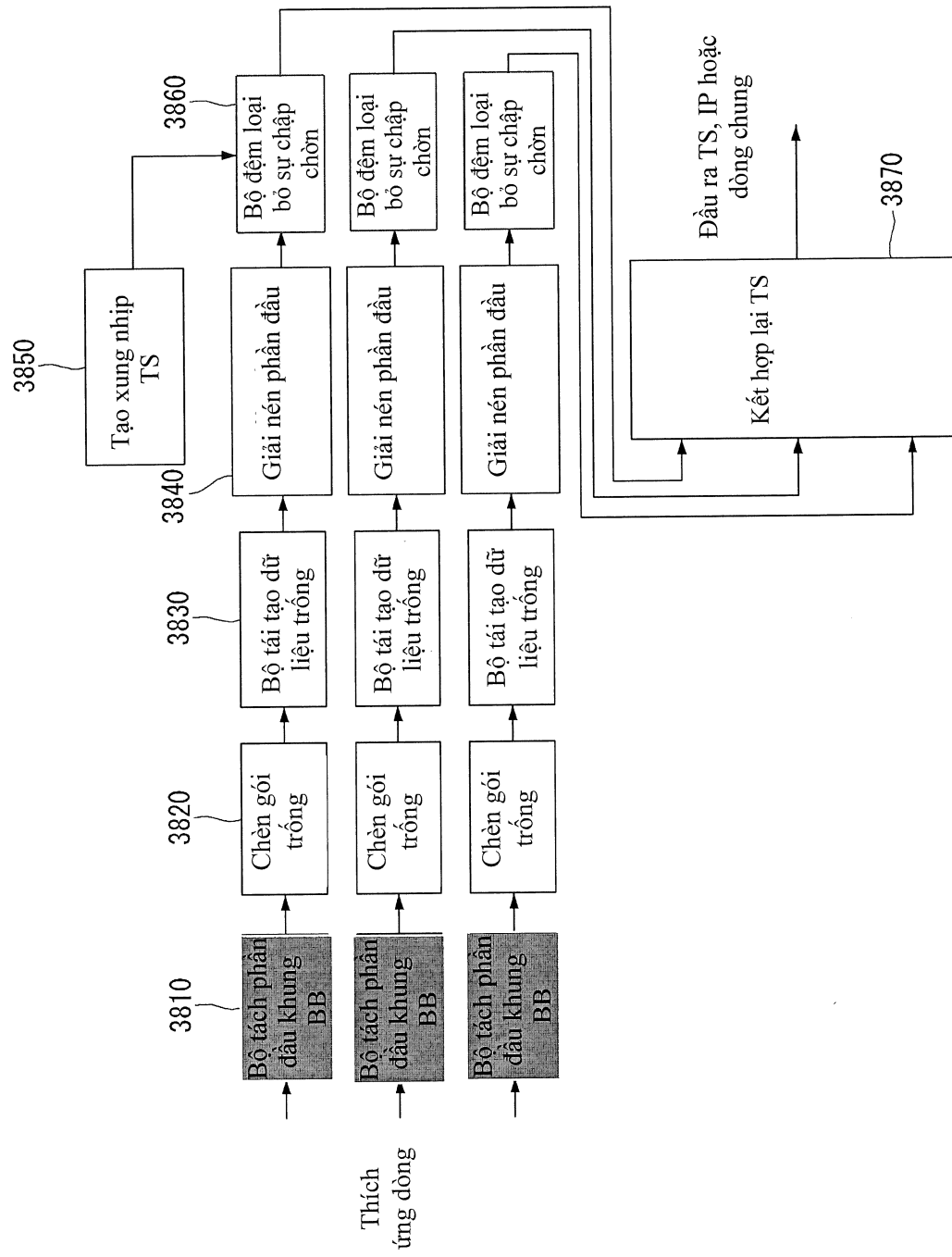
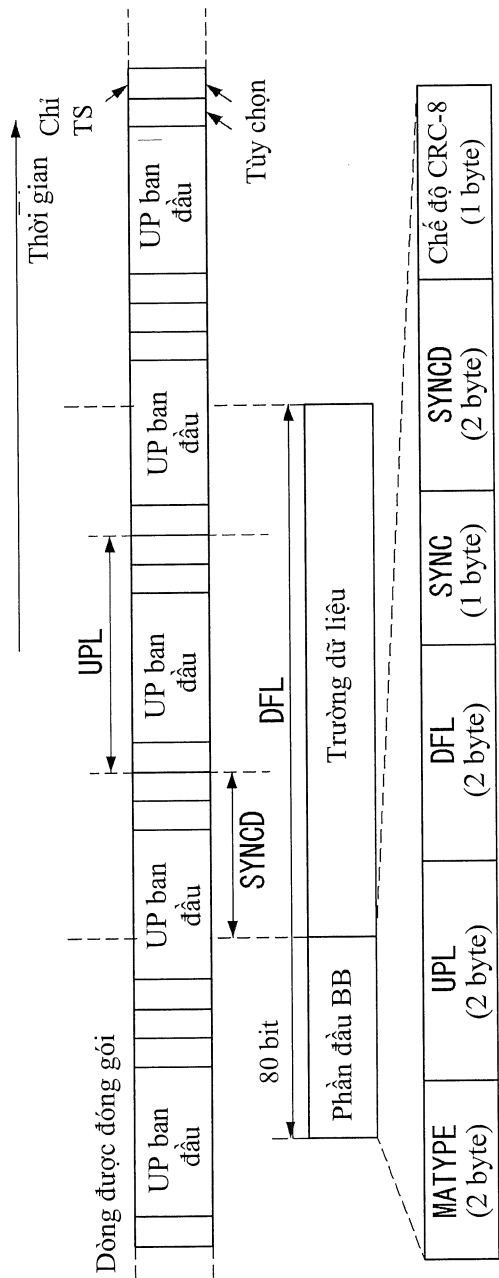


FIG. 39



Định dạng dòng ở đầu ra của bộ thích ứng chế độ, chế độ thông thường, GFPS và TS

FIG. 40

Các giá trị byte độn

| Độ dài tổng N của trường byte độn | Kích thước của trường con độ dài trong các byte | Giá trị của trường con độ dài              |
|-----------------------------------|-------------------------------------------------|--------------------------------------------|
| 1                                 | 1                                               | 0xFF (và không có các byte độn theo sau)   |
| 2                                 | 2                                               | 0xFEFF (và không có các byte độn theo sau) |
| 3-65278                           | 2                                               | N (và N-2 byte độn theo sau)               |

|                |     |                              |                              |                              |     |
|----------------|-----|------------------------------|------------------------------|------------------------------|-----|
| Stuff 1<br>(0) | ... | Phụ tải khung FEC (khung BB) |                              |                              |     |
| Stuff 1<br>(1) | ... | 0xFF                         | Phụ tải khung FEC (khung BB) |                              |     |
| Stuff 1<br>(1) | ... | 0xFE                         | 0xFF                         | Phụ tải khung FEC (khung BB) |     |
| Stuff 1<br>(1) | ... | MSB                          | LSB                          | P                            | ... |
|                |     |                              |                              | P                            |     |

FIG. 41

Trường 2 bit này chỉ báo trạng thái của vùng con đếm

|    |                                                        |
|----|--------------------------------------------------------|
| 00 | Không đếm                                              |
| 01 | 1 byte đếm (không có trường PADL)                      |
| 10 | Phần đếm lớn hơn hoặc bằng 2 byte (2 byte trường PADL) |
| 11 | Dự trữ                                                 |

|              |     |                      |                      |   |                      |
|--------------|-----|----------------------|----------------------|---|----------------------|
| PADI<br>(00) | ... | Khung FEC (khung BB) |                      |   |                      |
| PADI<br>(01) | ... | P                    | Khung FEC (khung BB) |   |                      |
| PADI<br>(10) | ... | MSB                  | LSB                  | P | ...                  |
|              |     |                      |                      |   | Khung FEC (khung BB) |

FIG. 42

| STUFF1 | Loại độn        | Kích thước độn | Trong dài                      | Byte thứ nhất   | Byte thứ hai | Byte thứ ba            | ... |
|--------|-----------------|----------------|--------------------------------|-----------------|--------------|------------------------|-----|
| 0      | Dữ liệu         | .              | .                              | Dữ liệu         |              |                        |     |
| 1      | Độn             | 1byte          | .                              | 1 1 1 1 1 1 1 1 | Dữ liệu      | Dữ liệu                | ... |
| 1      | Độn             | > 1byte        | .                              | 0 0 0 MSB       | LSB          | Độn                    | ... |
| 1      | Độn + trong dài | > 2byte        | Truyền tín hiệu A trong dài    | 0 0 1 MSB       | LSB          | Dữ liệu A trong dài    | ... |
| 1      | Độn + trong dài | > 2byte        | Truyền tín hiệu B trong dài    | 0 1 0 MSB       | LSB          | Dữ liệu B trong dài    | ... |
| 1      | Độn + trong dài | > 2byte        | Truyền tín hiệu A, B trong dài | 0 1 1 MSB       | LSB          | Dữ liệu A, B trong dài | ... |

[STUFF\_ONE] Ký hiệu chỉ báo đếm 1 byte  
 [STUFF\_TYPE] Truyền tín hiệu loại độn  
 [STUFF\_LEN] Độ dài độn

FIG. 43

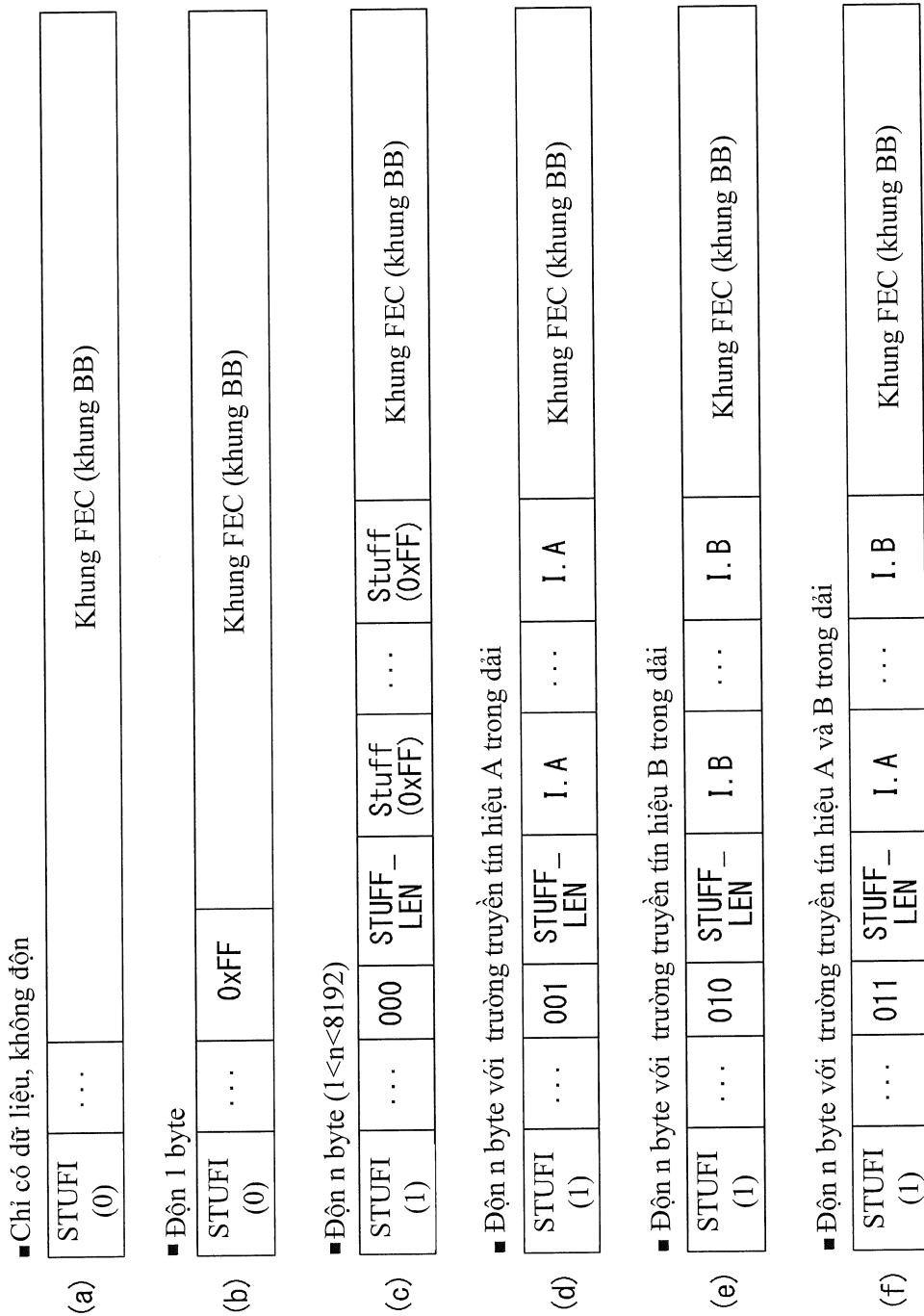


FIG. 44

| STUFF1 | Loại độn        | Kích thước độn | Trong dài                      | ... | Byte thứ n-2           | Byte thứ n-1    | Byte thứ n      |
|--------|-----------------|----------------|--------------------------------|-----|------------------------|-----------------|-----------------|
| 0      | Dữ liệu         | .              | .                              | ... | Dữ liệu                | Dữ liệu         | Dữ liệu         |
| 1      | Độn             | 1byte          | .                              | ... | Dữ liệu                | 1 1 1 1 1 1 1 1 | 1 1 1 1 1 1 1 1 |
| 1      | Độn             | > 1byte        | .                              | ... | Độn                    | MSB             | LSB 0 0 0       |
| 1      | Độn + trong dài | > 2byte        | Truyền tín hiệu A trong dài    | ... | Dữ liệu A trong dài    | MSB             | LSB 0 1 0       |
| 1      | Độn + trong dài | > 2byte        | Truyền tín hiệu B trong dài    | ... | Dữ liệu B trong dài    | MSB             | LSB 1 0 0       |
| 1      | Độn + trong dài | > 2byte        | Truyền tín hiệu A, B trong dài | ... | Dữ liệu A, B trong dài | MSB             | LSB 1 1 0       |

[STUFF\_LEN]

Độ dài độn

[STUFF\_TYPE]

Truyền tín hiệu loại độn

[STUFF\_ONE]

Ký hiệu chỉ báo đếm 1 byte

FIG. 45

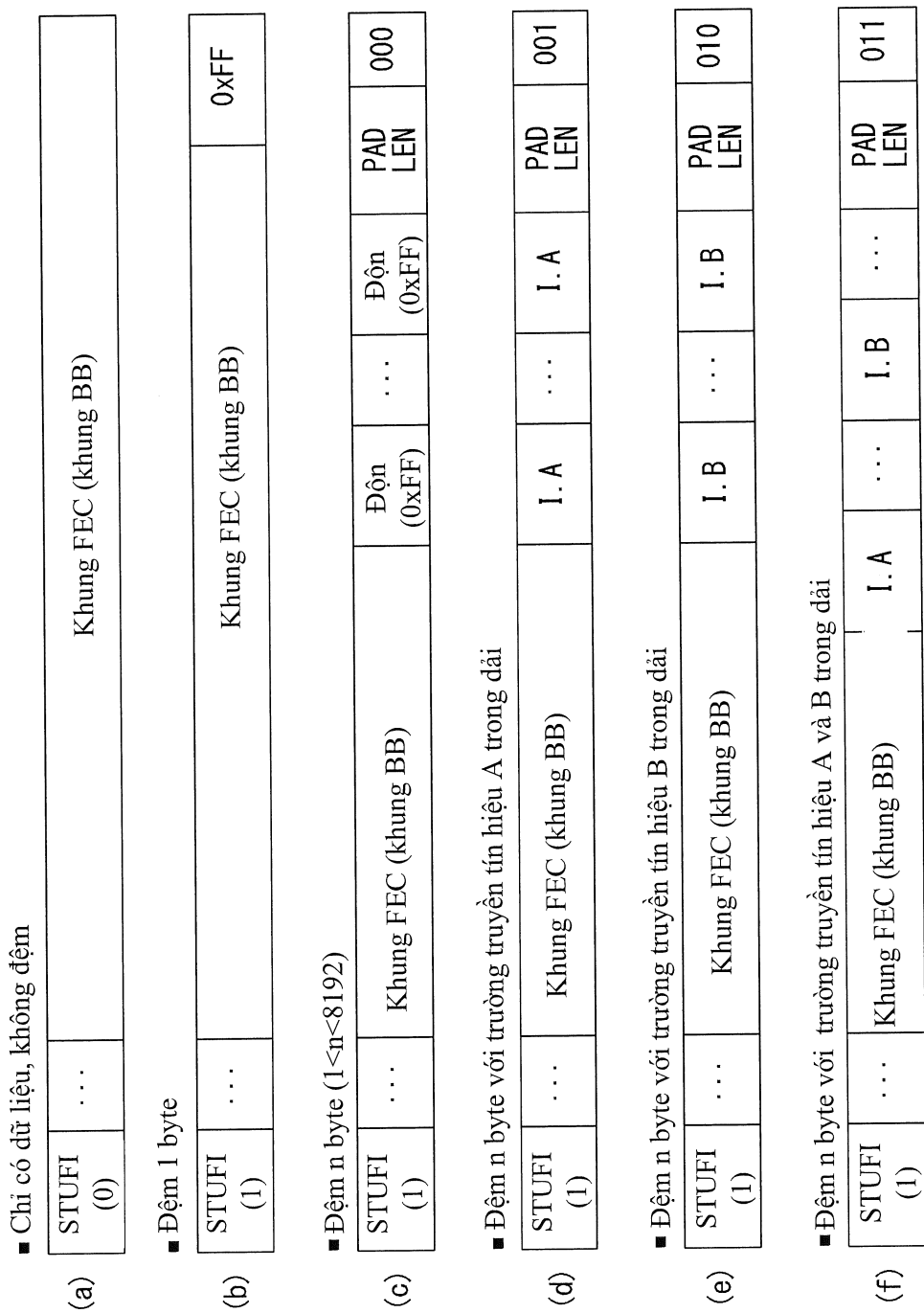


FIG. 46

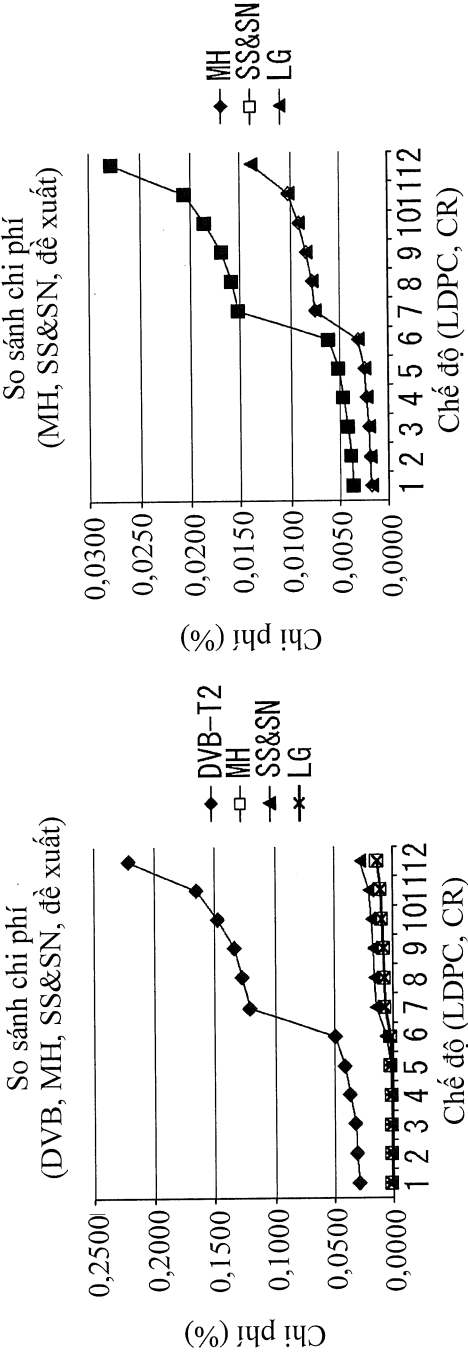


FIG. 47

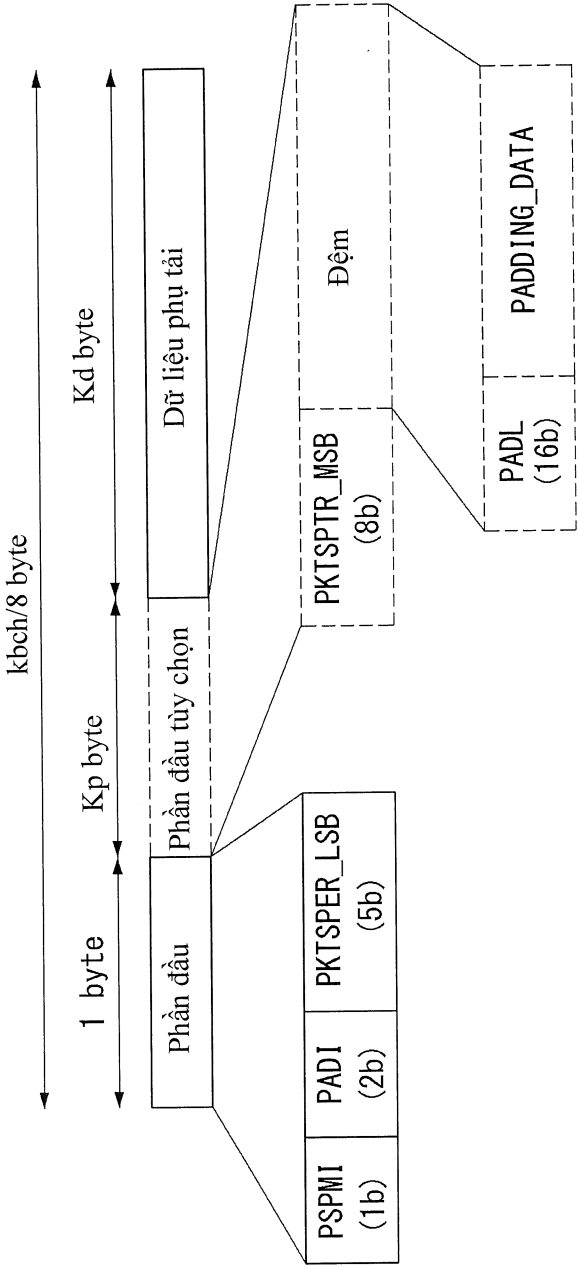




FIG. 49

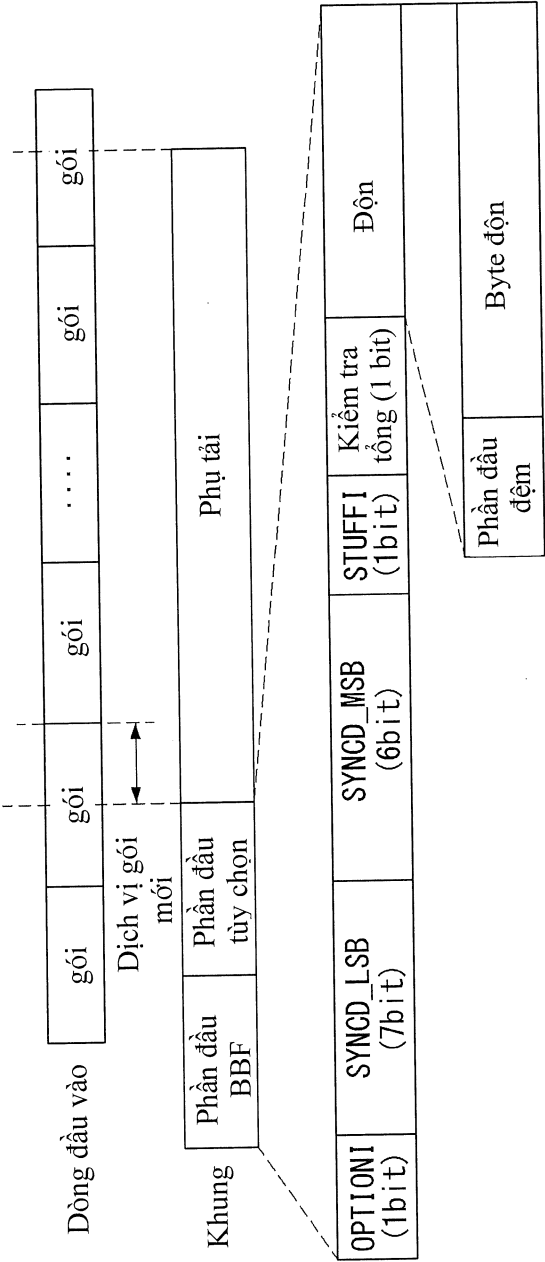


FIG. 50

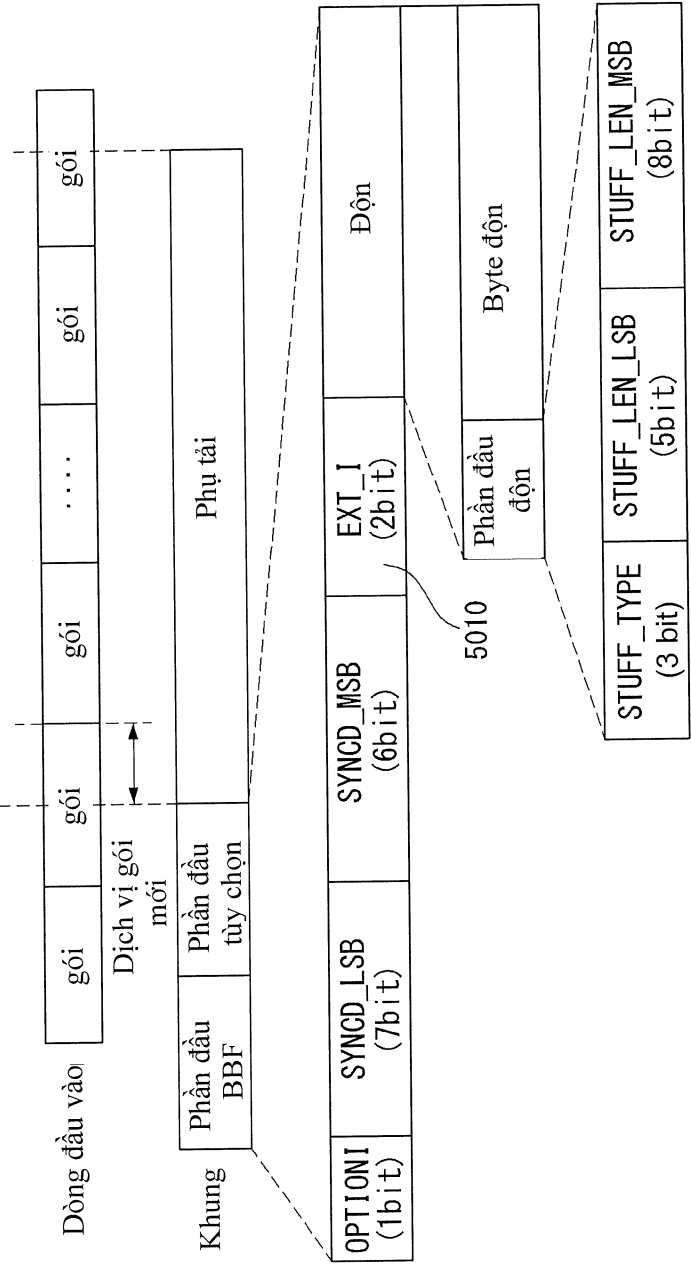


FIG. 51

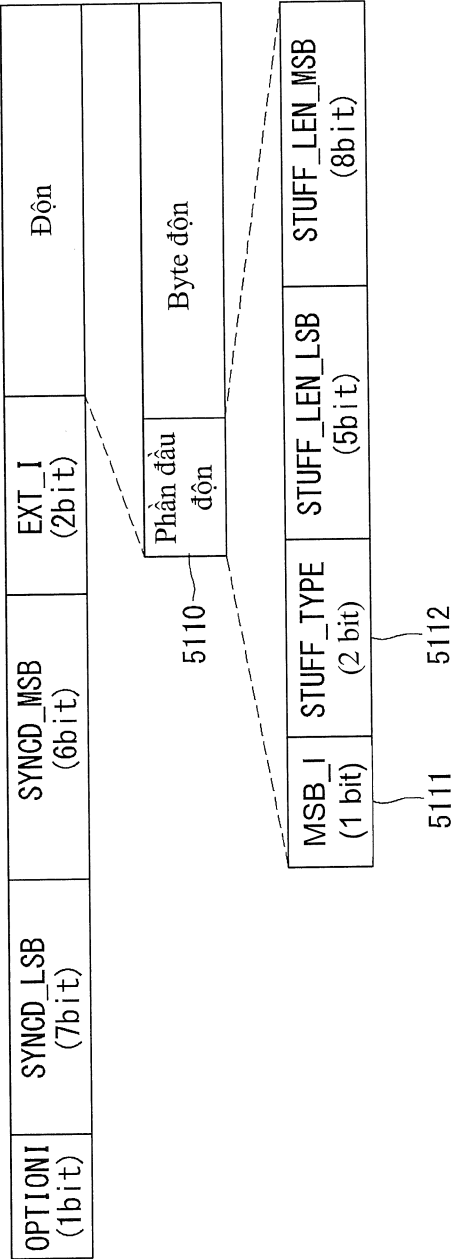


FIG. 52

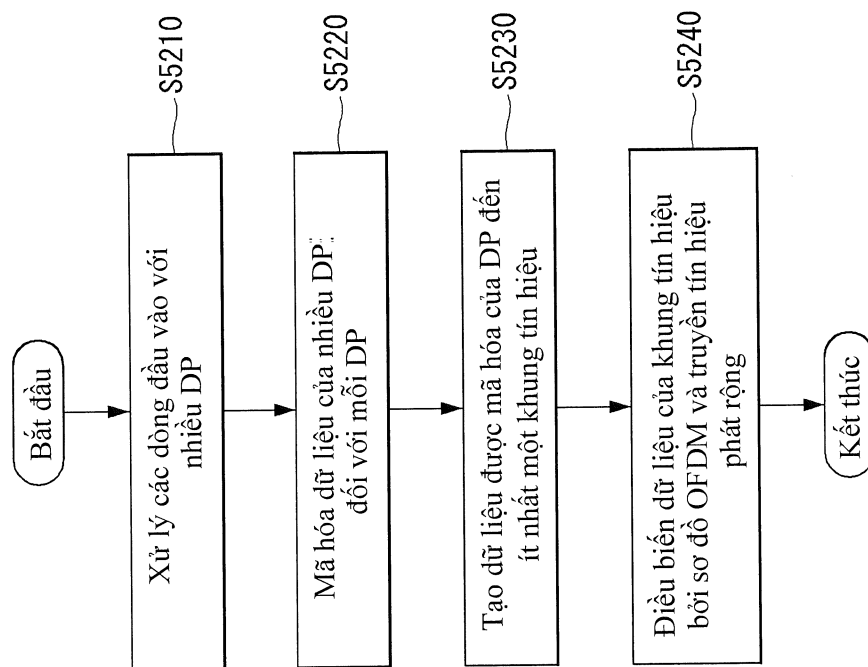


FIG. 53

