



(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)



1-0052335

(51)⁸ H03M 13/11; H03M 13/09

(13) B

(21) 1-2017-03404

(22) 15/02/2016

(86) PCT/KR2016/001506 15/02/2016

(87) WO 2016/129975 18/08/2016

(30) 62/115,810 13/02/2015 US; 62/120,543 25/02/2015 US; 62/202,304 07/08/2015 US;
10-2015-0137191 27/09/2015 KR

(45) 27/10/2025 451

(43) 25/10/2017 355A

(73) SAMSUNG ELECTRONICS CO., LTD. (KR)

129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do, 16677, Republic of Korea

(72) JEONG, Hong-sil (KR); KIM, Kyung-joong (KR); MYUNG, Se-ho (KR).

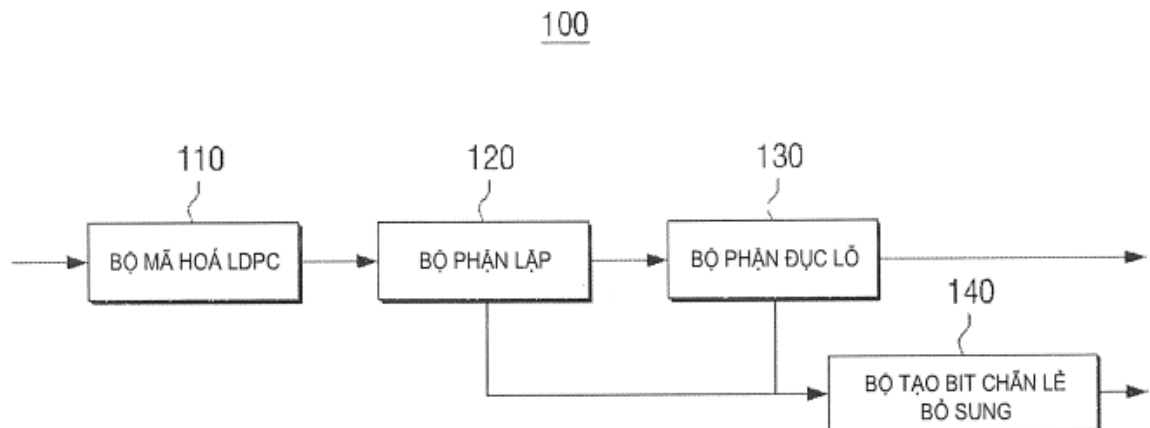
(74) Công ty Luật TNHH WINCO (WINCO LAW FIRM)

(54) THIẾT BỊ TRUYỀN VÀ THIẾT BỊ THU TÍN HIỆU PHÁT RỘNG TRUYỀN HÌNH

(21) 1-2017-03404

(57) Sáng chế đề cập đến thiết bị truyền tín hiệu và phương pháp tạo ra bit chắn lẻ bổ sung trong thiết bị truyền tín hiệu. Thiết bị truyền tín hiệu này bao gồm: bộ mã hoá kiểm tra chắn lẻ mật độ thấp (Low Density Parity Check, LDPC) để mã hoá các bit đầu vào chứa các bit mã hoá ngoài để tạo ra từ mã LDPC gồm có các bit đầu vào và các bit chắn lẻ được truyền đến thiết bị thu tín hiệu trong khung hiện thời; bộ phân đục lỗ để đục lỗ một phần của các bit chắn lẻ không được truyền trong khung hiện thời; và bộ tạo bit chắn lẻ bổ sung để chọn ít nhất một phần của các bit chắn lẻ để tạo ra các bit chắn lẻ bổ sung được truyền đến thiết bị thu tín hiệu trong khung đứng trước khung hiện thời, trong đó số lượng bit chắn lẻ bổ sung được xác định dựa vào số lượng bit mã hoá ngoài và số lượng bit chắn lẻ còn lại sau khi đục lỗ.

Fig. 1



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến thiết bị truyền tín hiệu và phương pháp tạo ra bit chắn lẻ bổ sung để truyền tín hiệu.

Tình trạng kỹ thuật của sáng chế

Các dịch vụ truyền thông phát rộng trong xã hội công nghệ thông tin ở thế kỷ thứ 21 đang chuyển sang kỷ nguyên số hoá, phân phối qua nhiều kênh, mở rộng dải tần số và cung cấp các dịch vụ chất lượng cao. Cụ thể, khi thiết bị thu tín hiệu truyền hình (Television, TV) kỹ thuật số chất lượng cao và các thiết bị thu tín hiệu phát rộng cầm tay được sử dụng rộng rãi, thì nhu cầu về các dịch vụ phát rộng kỹ thuật số có hỗ trợ nhiều sơ đồ thu tín hiệu khác nhau ngày càng tăng lên.

Theo nhu cầu đó, các nhóm tiêu chuẩn hoá đã thiết lập các tiêu chuẩn truyền thông phát rộng để cung cấp nhiều dịch vụ truyền và thu tín hiệu khác nhau nhằm thoả mãn nhu cầu của người dùng. Tuy nhiên, vẫn cần phải có phương pháp cung cấp dịch vụ tốt hơn cho người dùng với hiệu quả cao hơn nữa.

Bản chất kỹ thuật của sáng chế

Các phương án làm ví dụ thực hiện sáng chế có thể khắc phục các nhược điểm của thiết bị truyền tín hiệu, thiết bị thu tín hiệu và phương pháp được thực hiện trong các thiết bị này theo các giải pháp đã biết. Tuy nhiên, các phương án đó không nhất thiết phải khắc phục hoặc có thể không khắc phục được các nhược điểm nêu trên. Các phương án làm ví dụ thực hiện sáng chế đề xuất thiết bị truyền tín hiệu và phương pháp tạo ra bit chắn lẻ bổ sung sử dụng các bit chắn lẻ được tạo ra bằng cách mã hoá.

Theo một khía cạnh của phương án làm ví dụ thực hiện sáng chế, sáng chế đề xuất thiết bị truyền tín hiệu có thể bao gồm: bộ mã hoá kiểm tra chẵn lẻ mật độ thấp (Low Density Parity Check, LDPC) để mã hoá các bit đầu vào chứa các bit mã hoá ngoài để tạo ra từ mã LDPC gồm có các bit đầu vào và các bit chẵn lẻ được truyền đến thiết bị thu tín hiệu trong khung hiện thời; bộ phận đọc lỗi để đọc lỗi một phần của các bit chẵn lẻ không được truyền trong khung hiện thời; và bộ tạo bit chẵn lẻ bổ sung để chọn ít nhất một phần của các bit chẵn lẻ để tạo ra các bit chẵn lẻ bổ sung được truyền đến thiết bị thu tín hiệu trong khung đứng trước khung hiện thời, trong đó số lượng bit chẵn lẻ bổ sung được xác định dựa vào số lượng bit mã hoá ngoài và số lượng bit chẵn lẻ còn lại sau khi đọc lỗi.

Số lượng bit chẵn lẻ bổ sung có thể được xác định dựa thêm vào số lượng bit chẵn lẻ được đọc lỗi không được truyền đến thiết bị thu tín hiệu trong khung hiện thời.

Thiết bị truyền tín hiệu này có thể còn bao gồm bộ phận lặp để lặp lại, trong từ mã LDPC, ít nhất một phần của các bit trong từ mã LDPC sao cho các bit được lặp lại của từ mã LDPC sẽ được truyền cùng với từ mã LDPC trước khi lặp lại trong khung hiện thời.

Số lượng bit chẵn lẻ bổ sung được xác định dựa thêm vào số lượng bit được lặp lại của từ mã LDPC.

Số lượng bit chẵn lẻ bổ sung có thể được tính dựa vào số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung được tính dựa vào biểu thức 8.

Số lượng bit chẵn lẻ bổ sung có thể được tính dựa vào biểu thức 10.

Theo một khía cạnh của phương án làm ví dụ khác để thực hiện sáng chế, sáng chế đề xuất phương pháp tạo ra bit chẵn lẻ bổ sung trong thiết bị truyền tín hiệu.

Phương pháp này có thể bao gồm các bước: mã hoá các bit đầu vào chứa các bit mã hoá ngoài để tạo ra từ mã LDPC gồm có các bit đầu vào và các bit chẵn lẻ được truyền đến thiết bị thu tín hiệu trong khung hiện thời; đọc lỗi một phần của các bit chẵn lẻ không được truyền trong khung hiện thời; và tạo ra các bit chẵn lẻ bổ sung được truyền đến thiết bị thu tín hiệu trong khung đứng trước khung hiện thời bằng cách chọn ít nhất một phần của các bit chẵn lẻ, trong đó số lượng bit chẵn lẻ bổ sung được xác định dựa vào số lượng bit mã hoá ngoài và số lượng bit chẵn lẻ còn lại sau khi đọc lỗi.

Phương pháp này có thể còn bao gồm bước lặp lại, trong từ mã LDPC, ít nhất một phần của các bit trong từ mã LDPC sao cho các bit được lặp lại của từ mã LDPC sẽ được truyền cùng với từ mã LDPC trước khi lặp lại trong khung hiện thời.

Số lượng bit chẵn lẻ bổ sung được xác định dựa thêm vào số lượng bit được lặp lại của từ mã LDPC.

Các bit được lặp lại của từ mã LDPC có thể là ít nhất một phần của các bit chẵn lẻ. Khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ được đọc lỗi và nhỏ hơn tổng số của các bit chẵn lẻ được đọc lỗi và các bit chẵn lẻ lặp lại, các bit được đọc lỗi và các bit kể từ bit đầu tiên trong số các bit chẵn lẻ lặp lại có thể được chọn để tạo ra các bit chẵn lẻ bổ sung. Khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng bộ tạo bit chẵn lẻ bổ sung lớn hơn tổng số của số lượng bit chẵn lẻ được đọc lỗi và số lượng bit chẵn lẻ, các bit có thể được chọn theo thứ tự lần lượt là các bit được đọc lỗi, các bit chẵn lẻ lặp lại và các bit chẵn lẻ, để tạo ra các bit chẵn lẻ bổ sung. Khi số lượng bit chẵn lẻ bổ sung được tạo ra nhỏ hơn số lượng bit chẵn lẻ được đọc lỗi, các bit có thể được chọn kể từ bit được đọc lỗi cuối cùng trong số các bit chẵn

lẻ được đục lỗ để tạo ra các bit chẵn lẻ bổ sung.

Số lượng bit chẵn lẻ bổ sung có thể được đặt bằng một bội số nguyên lần của bậc điều biến dùng để điều biến các bit đầu vào và các bit chẵn lẻ còn lại để truyền đến thiết bị thu tín hiệu.

Như đã nêu trên, theo các phương án làm ví dụ thực hiện sáng chế, một số bit trong số các bit chẵn lẻ có thể được truyền thêm để đạt được độ tăng ích do mã hoá và độ tăng ích do phân tập.

Mô tả vắn tắt các hình vẽ

Các khía cạnh nêu trên và/hoặc các khía cạnh khác của các phương án làm ví dụ thực hiện sáng chế sẽ được mô tả dưới đây dựa vào các hình vẽ kèm theo, trong đó:

Fig.1 là sơ đồ khối thể hiện cấu hình của thiết bị truyền tín hiệu theo phương án làm ví dụ thực hiện sáng chế;

Fig.2 và Fig.3 là các sơ đồ thể hiện ma trận kiểm tra chẵn lẻ theo các phương án làm ví dụ thực hiện sáng chế;

Fig.4 đến Fig.7 là các sơ đồ khối thể hiện phương pháp lập theo các phương án làm ví dụ thực hiện sáng chế;

Fig.8 đến Fig.11 là các sơ đồ khối thể hiện phương pháp đục lỗ theo các phương án làm ví dụ thực hiện sáng chế;

Fig.12 đến Fig.40 là các sơ đồ thể hiện phương pháp tạo ra bit chẵn lẻ bổ sung theo các phương án làm ví dụ thực hiện sáng chế;

Fig.41 là sơ đồ thể hiện cấu trúc khung theo phương án làm ví dụ thực hiện sáng chế;

Fig.42 và Fig.43 là các sơ đồ khối thể hiện cấu hình chi tiết của thiết bị truyền

tín hiệu theo các phương án làm ví dụ thực hiện sáng chế;

Fig.44 đến Fig.57 là các sơ đồ thể hiện phương pháp xử lý tín hiệu theo các phương án làm ví dụ thực hiện sáng chế;

Fig.58 và Fig.59 là các sơ đồ khối thể hiện cấu hình của thiết bị thu tín hiệu theo các phương án làm ví dụ thực hiện sáng chế;

Fig.60 và Fig.61 là các sơ đồ thể hiện ví dụ về phương pháp kết hợp các giá trị tỷ số hợp lý dạng loga (Log-Likelihood Ratio, LLR) của thiết bị thu tín hiệu theo các phương án làm ví dụ thực hiện sáng chế;

Fig.62 là sơ đồ thể hiện ví dụ về phương pháp cung cấp thông tin về độ dài của thông tin báo hiệu cho tầng L1 theo phương án làm ví dụ thực hiện sáng chế;

Fig.63 là lưu đồ thể hiện phương pháp tạo ra bit chắn lẻ bổ sung theo phương án làm ví dụ thực hiện sáng chế; và

Fig.64 là sơ đồ thể hiện độ tăng ích do mã hoá và độ tăng ích do phân tập có thể đạt được khi sử dụng các bit chắn lẻ bổ sung theo phương án làm ví dụ thực hiện sáng chế.

Mô tả chi tiết sáng chế

Các phương án làm ví dụ thực hiện sáng chế sẽ được mô tả chi tiết dưới đây dựa vào các hình vẽ kèm theo.

Fig.1 là sơ đồ khối thể hiện cấu hình của thiết bị truyền tín hiệu theo phương án làm ví dụ thực hiện sáng chế. Dựa vào Fig.1, bộ truyền tín hiệu 100 bao gồm bộ mã hoá kiểm tra chẵn lẻ mật độ thấp (LDPC) 110, bộ phận lặp 120, bộ phận đục lỗ 130 và bộ tạo bit chắn lẻ bổ sung 140.

Bộ mã hoá LDPC 110 có thể mã hoá các bit đầu vào của dịch vụ có thể có nhiều

loại dữ liệu khác nhau. Nói cách khác, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC trên các bit đầu vào để tạo ra các bit chẵn lẻ, tức là các bit chẵn lẻ LDPC.

Cụ thể, các bit đầu vào là các bit thông tin LDPC để mã hoá LDPC và có thể chứa các bit mã hoá ngoài và các bit không (tức là các bit có giá trị 0). Trong đó, các bit mã hoá ngoài gồm có các bit thông tin và các bit chẵn lẻ (hay các bit kiểm tra chẵn lẻ) được tạo ra bằng cách mã hoá ngoài trên các bit thông tin.

Dưới đây, các bit thông tin có thể là các bit thông tin báo hiệu (gọi theo cách khác là “các bit báo hiệu” hoặc “thông tin báo hiệu”). Các bit thông tin báo hiệu này có thể là thông tin cần thiết cho thiết bị thu tín hiệu 200 (như được thể hiện trên Fig.58 hoặc Fig.59) để xử lý dữ liệu dịch vụ (ví dụ dữ liệu phát rộng) được truyền từ thiết bị truyền tín hiệu 100.

Mã hoá ngoài là thao tác mã hoá được thực hiện trước khi mã hoá trong theo sơ đồ mã hoá ghép nối, và có thể sử dụng nhiều sơ đồ mã hoá khác nhau như sơ đồ mã hoá Bose, Chaudhuri, Hocquenghem (BCH) và/hoặc sơ đồ mã hoá kiểm dư vòng (Cyclic Redundancy Check, CRC). Trong trường hợp như vậy, thao tác mã hoá trong có thể là mã hoá LDPC.

Để mã hoá LDPC, cần phải có một số lượng định trước của các bit thông tin LDPC được xác định theo tỷ lệ mã và độ dài mã. Vì vậy, khi số lượng bit mã hoá ngoài được tạo ra bằng cách mã hoá ngoài trên các bit thông tin nhỏ hơn số lượng bit thông tin LDPC cần thiết, thì một số lượng thích hợp của các bit không sẽ được đệm vào để đáp ứng số lượng bit thông tin LDPC cần thiết để mã hoá LDPC. Vì vậy, các bit mã hoá ngoài và các bit không được đệm vào có thể tạo nên các bit thông tin

LDPC với số lượng đúng bằng số lượng bit cần thiết để mã hoá LDPC.

Vì các bit không được đệm vào là các bit cần thiết chỉ nhằm đạt được số lượng bit định trước để mã hoá LDPC, nên các bit không được đệm vào sẽ được mã hoá LDPC và sau đó không được truyền đến thiết bị thu tín hiệu 200. Như vậy, thủ tục đệm bit không, và sau đó, không truyền các bit không được đệm vào đến thiết bị thu tín hiệu 200 có thể được gọi là thủ tục rút gọn. Trong trường hợp như vậy, các bit không được đệm vào có thể được gọi là các bit rút gọn (hay các bit được rút gọn).

Ví dụ, giả sử rằng số lượng bit thông tin bằng K_{sig} và số lượng bit khi M_{outer} bit chẵn lẻ được gắn vào các bit thông tin bằng cách mã hoá ngoài, tức là số lượng bit mã hoá ngoài gồm có các bit thông tin và các bit chẵn lẻ, bằng $N_{\text{outer}} (= K_{\text{sig}} + M_{\text{outer}})$.

Trong trường hợp như vậy, khi số lượng N_{outer} bit mã hoá ngoài nhỏ hơn số lượng K_{ldpc} bit thông tin LDPC, thì số lượng $K_{\text{ldpc}} - N_{\text{outer}}$ bit không sẽ được đệm vào sao cho các bit mã hoá ngoài cùng với các bit không được đệm vào có thể tạo nên các bit thông tin LDPC.

Ví dụ nêu trên mô tả rằng các bit không được đệm vào, nhưng đó chỉ là một ví dụ.

Khi các bit thông tin là thông tin báo hiệu cho dữ liệu hoặc dữ liệu dịch vụ, thì độ dài của các bit thông tin có thể thay đổi tùy theo lượng dữ liệu. Vì vậy, khi số lượng bit thông tin lớn hơn số lượng bit thông tin LDPC cần thiết để mã hoá LDPC, thì các bit thông tin có thể được phân đoạn theo một giá trị định trước.

Vì vậy, khi số lượng bit thông tin hoặc số lượng bit thông tin đã phân đoạn nhỏ hơn số lượng thu được sau khi lấy số lượng bit thông tin LDPC trừ đi số lượng bit chẵn lẻ (tức là M_{outer}) được tạo ra bằng cách mã hoá ngoài, các bit không được đệm

vào với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit thông tin LDPC trừ đi số lượng bit mã hoá ngoài, thì các bit thông tin LDPC có thể được tạo nên bởi các bit mã hoá ngoài và các bit không được đệm vào.

Tuy nhiên, khi số lượng bit thông tin hoặc số lượng bit thông tin đã phân đoạn bằng số lượng thu được sau khi lấy số lượng bit thông tin LDPC trừ đi số lượng bit chẵn lẻ được tạo ra bằng cách mã hoá ngoài, thì các bit thông tin LDPC có thể được tạo nên bởi các bit mã hoá ngoài mà không có các bit không được đệm vào.

Ví dụ nêu trên mô tả rằng các bit thông tin được mã hoá ngoài, nhưng đó chỉ là một ví dụ. Tuy nhiên, các bit thông tin có thể không được mã hoá ngoài và tạo nên các bit thông tin LDPC cùng với các bit không được đệm vào tùy theo số lượng bit thông tin hoặc chỉ có các bit thông tin có thể tạo nên các bit thông tin LDPC mà không có các bit không được đệm vào.

Để cho dễ hiểu, thao tác mã hoá ngoài sẽ được mô tả dưới đây với giả sử rằng thao tác đó được thực hiện bằng cách mã hoá BCH.

Cụ thể, các bit đầu vào sẽ được mô tả với giả sử rằng các bit đầu vào đó gồm có các bit mã hoá BCH và các bit không, các bit mã hoá BCH gồm có các bit thông tin và các bit kiểm tra chẵn lẻ BCH (hay các bit chẵn lẻ BCH) được tạo ra bằng cách mã hoá BCH trên các bit thông tin.

Điều này có nghĩa là, giả sử rằng số lượng bit thông tin bằng K_{sig} và số lượng bit khi M_{outer} bit kiểm tra chẵn lẻ BCH được tạo ra bằng cách mã hoá BCH trên các bit thông tin, tức là số lượng bit mã hoá BCH gồm có các bit thông tin và các bit kiểm tra chẵn lẻ BCH, bằng $N_{\text{outer}} (= K_{\text{sig}} + M_{\text{outer}})$. Theo sáng chế, $M_{\text{outer}} = 168$.

Ví dụ nêu trên mô tả rằng các bit không, sẽ được rút gọn, được đệm vào, nhưng

đó chỉ là một ví dụ. Điều này có nghĩa là, vì các bit không là các bit có giá trị được định trước bởi thiết bị truyền tín hiệu 100 và thiết bị thu tín hiệu 200 và được đem vào chỉ để tạo ra các bit thông tin LDPC cùng với các bit thông tin chứa thông tin cần bản được truyền đến thiết bị thu tín hiệu 200, cho nên các bit có giá trị khác (ví dụ 1) được định trước bởi thiết bị truyền tín hiệu 100 và thiết bị thu tín hiệu 200 thay cho các bit không có thể được đem vào để rút gọn.

Bộ mã hoá LDPC 110 có thể mã hoá có hệ thống trên các bit thông tin LDPC để tạo ra các bit chẵn lẻ LDPC, và xuất ra từ mã LDPC (hay các bit mã hoá LDPC) gồm có các bit thông tin LDPC và các bit chẵn lẻ LDPC. Điều này có nghĩa là, mã LDPC là mã có hệ thống, và vì vậy, từ mã LDPC có thể gồm có các bit thông tin LDPC trước khi mã hoá LDPC và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Ví dụ, bộ mã hoá LDPC 110 có thể mã hoá LDPC trên K_{ldpc} bit thông tin LDPC $i = (i_0, i_1, \dots, i_{K_{ldpc}-1})$ để tạo ra N_{ldpc_parity} bit chẵn lẻ LDPC $(p_0, p_1, \dots, p_{N_{ldpc}-K_{ldpc}-1})$ và xuất ra từ mã LDPC $\Lambda = (c_0, c_1, \dots, c_{N_{inner}-1}) = (i_0, i_1, \dots, i_{K_{ldpc}-1}, p_0, p_1, \dots, p_{N_{inner}-K_{ldpc}-1})$ gồm có $N_{inner} (= K_{ldpc} + N_{ldpc_parity})$ bit.

Trong trường hợp như vậy, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC trên các bit đầu vào (tức là các bit thông tin LDPC) ở các tỷ lệ mã khác nhau để tạo ra từ mã LDPC có độ dài định trước.

Ví dụ, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC trên 3240 bit đầu vào ở tỷ lệ mã bằng 3/15 để tạo ra từ mã LDPC gồm có 16200 bit. Ví dụ khác, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC trên 6480 bit đầu vào ở tỷ lệ mã bằng 6/15 để tạo ra từ mã LDPC gồm có 16200 bit.

Quy trình thực hiện bước mã hoá LDPC là quy trình tạo ra từ mã LDPC sao cho thoả mãn hệ thức $H \cdot C^T = 0$, và do đó, bộ mã hoá LDPC 110 có thể sử dụng ma trận kiểm tra chẵn lẻ để thực hiện bước mã hoá LDPC. Trong đó, H là ma trận kiểm tra chẵn lẻ và C là từ mã LDPC.

Cấu trúc của ma trận kiểm tra chẵn lẻ theo các phương án làm ví dụ thực hiện sáng chế sẽ được mô tả dưới đây dựa vào các hình vẽ kèm theo. Trong ma trận kiểm tra chẵn lẻ, các phần tử nếu không phải là có giá trị bằng 1 thì đều có giá trị bằng 0.

Ví dụ, ma trận kiểm tra chẵn lẻ theo phương án làm ví dụ thực hiện sáng chế có thể có cấu trúc như được thể hiện trên Fig.2.

Dựa vào Fig.2, ma trận kiểm tra chẵn lẻ 20 có thể gồm có năm ma trận con A, B, C, Z và D. Để mô tả cấu trúc của ma trận kiểm tra chẵn lẻ 20, cấu trúc của mỗi ma trận con sẽ được mô tả dưới đây.

Ma trận con A gồm có K cột và g hàng, và ma trận con C gồm có $K+g$ cột và $N-K-g$ hàng. Trong đó, K (hoặc K_{ldpc}) là độ dài của các bit thông tin LDPC, và N (hoặc N_{inner}) là độ dài của từ mã LDPC.

Ngoài ra, trong các ma trận con A và C, chỉ số của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i có thể được xác định dựa vào bảng 1 khi độ dài của từ mã LDPC bằng 16200 và tỷ lệ mã bằng 3/15. Trong đó, số lượng cột thuộc cùng một nhóm cột có thể bằng 360.

Bảng 1

8	372	841	4522	5253	7430	8542	9822	10550	11896	11988
80	255	667	1511	3549	5239	5422	5497	7157	7854	11267
257	406	792	2916	3072	3214	3638	4090	8175	8892	9003
80	150	346	1883	6838	7818	9482	10366	10514	11468	12341
32	100	978	3493	6751	7787	8496	10170	10318	10451	12561
504	803	856	2048	6775	7631	8110	8221	8371	9443	10990
152	283	696	1164	4514	4649	7260	7370	11925	11986	12092
127	1034	1044	1842	3184	3397	5931	7577	11898	12339	12689
107	513	979	3934	4374	4658	7286	7809	8830	10804	10893
2045	2499	7197	8887	9420	9922	10132	10540	10816	11876	
2932	6241	7136	7835	8541	9403	9817	11679	12377	12810	
2211	2288	3937	4310	5952	6597	9692	10445	11064	11272	

Vị trí (gọi theo cách khác là “chỉ số” hoặc “giá trị chỉ số”) của các hàng có chứa giá trị 1 trong các ma trận con A và C sẽ được mô tả chi tiết dưới đây, ví dụ, dựa vào bảng 1.

Khi độ dài của từ mã LDPC bằng 16200 và tỷ lệ mã bằng 3/15, thì các thông số mã hoá M_1 , M_2 , Q_1 và Q_2 dựa trên ma trận kiểm tra chẵn lẻ 200 lần lượt bằng 1080, 11880, 3 và 33.

Trong đó, Q_1 là khoảng cách dịch chuyển tuần hoàn của các cột thuộc cùng một nhóm cột trong ma trận con A, và Q_2 là khoảng cách dịch chuyển tuần hoàn của các cột thuộc cùng một nhóm cột trong ma trận con C.

Ngoài ra, $Q_1 = M_1/L$, $Q_2 = M_2/L$, $M_1 = g$, $M_2 = N-K-g$, và L là khoảng lặp lại của mẫu cột trong các ma trận con A và C tương ứng, tức là số lượng cột (ví dụ 360 cột) thuộc cùng một nhóm cột.

Chỉ số của các hàng có chứa giá trị 1 trong các ma trận con A và C tương ứng, có thể được xác định dựa vào giá trị M_1 .

Ví dụ, trong bảng 1 nêu trên, vì $M_1 = 1080$, nên vị trí của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i trong ma trận con A có thể được xác định dựa

vào các giá trị nhỏ hơn 1080 trong số các giá trị chỉ số trong bảng 1 nêu trên, và vị trí của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i trong ma trận con C có thể được xác định dựa vào các giá trị bằng hoặc lớn hơn 1080 trong số các giá trị chỉ số trong bảng 1 nêu trên.

Cụ thể, dãy tương ứng với nhóm cột thứ 0 trong bảng 1 nêu trên là “8 372 841 4522 5253 7430 8542 9822 10550 11896 11988”. Vì vậy, trong cột thứ 0 của nhóm cột thứ 0 trong ma trận con A , giá trị 1 có thể lần lượt nằm ở hàng thứ tám, hàng thứ 372 và hàng thứ 841, và trong cột thứ 0 của nhóm cột thứ 0 trong ma trận con C , giá trị 1 có thể lần lượt nằm ở hàng thứ 4522, hàng thứ 5253, hàng thứ 7430, hàng thứ 8542, hàng thứ 9822, hàng thứ 10550, hàng thứ 11896 và hàng thứ 11988.

Trong ma trận con A , khi đã xác định được vị trí của giá trị 1 trong cột thứ 0 của mỗi nhóm cột, thì có thể cho vị trí đó dịch chuyển tuần hoàn với khoảng cách Q_1 để xác định được vị trí của các hàng có chứa giá trị 1 trong các cột khác của mỗi nhóm cột, và trong ma trận con C , khi đã xác định được vị trí của giá trị 1 trong cột thứ 0 của mỗi nhóm cột, thì có thể cho vị trí đó dịch chuyển tuần hoàn với khoảng cách Q_2 để xác định được vị trí của các hàng có chứa giá trị 1 trong các cột khác của mỗi nhóm cột.

Trong ví dụ nêu trên, trong cột thứ 0 của nhóm cột thứ 0 trong ma trận con A , giá trị 1 nằm ở hàng thứ tám, hàng thứ 372 và hàng thứ 841. Trong trường hợp như vậy, vì $Q_1 = 3$, nên chỉ số của các hàng có chứa giá trị 1 trong cột thứ nhất của nhóm cột thứ 0 có thể là 11 ($= 8+3$), 375 ($= 372+3$) và 844 ($= 841+3$), và chỉ số của các hàng có chứa giá trị 1 trong cột thứ hai của nhóm cột thứ 0 có thể là 14 ($= 11+3$), 378 ($= 375+3$) và 847 ($= 844+3$).

Trong cột thứ 0 của nhóm cột thứ 0 trong ma trận con C, giá trị 1 nằm ở hàng thứ 4522, hàng thứ 5253, hàng thứ 7430, hàng thứ 8542, hàng thứ 9822, hàng thứ 10550, hàng thứ 11896 và hàng thứ 11988. Trong trường hợp như vậy, vì $Q_2 = 33$, nên chỉ số của các hàng có chứa giá trị 1 trong cột thứ nhất của nhóm cột thứ 0 có thể là 4555 ($= 4522+33$), 5286 ($= 5253+33$), 7463 ($= 7430+33$), 8575 ($= 8542+33$), 9855 ($= 9822+33$), 10583 ($= 10550+33$), 11929 ($= 11896+33$) và 12021 ($= 11988+33$), và chỉ số của các hàng có chứa giá trị 1 trong cột thứ hai của nhóm cột thứ 0 có thể là 4588 ($= 4555+33$), 5319 ($= 5286+33$), 7496 ($= 7463+33$), 8608 ($= 8575+33$), 9888 ($= 9855+33$), 10616 ($= 10583+33$), 11962 ($= 11929+33$) và 12054 ($= 12021+33$).

Theo sơ đồ này, có thể xác định được vị trí của các hàng có chứa giá trị 1 trong tất cả các nhóm cột trong các ma trận con A và C.

Ma trận con B là ma trận hai đường chéo, ma trận con D là ma trận đơn vị, và ma trận con Z là ma trận không.

Do đó, cấu trúc của ma trận kiểm tra chẵn lẻ 20 như được thể hiện trên Fig.2 có thể được xác định dựa vào các ma trận con A, B, C, D và Z có cấu trúc nêu trên.

Quy trình thực hiện bước mã hoá LDPC, bằng bộ mã hoá LDPC 110, dựa trên ma trận kiểm tra chẵn lẻ 20 như được thể hiện trên Fig.2 sẽ được mô tả dưới đây.

Mã LDPC có thể được sử dụng để mã hoá khối thông tin $S = (s_0, s_1, \dots, s_{K-1})$. Trong trường hợp như vậy, để tạo ra từ mã LDPC $\Lambda = (\lambda_0, \lambda_1, \dots, \lambda_{N-1})$ có độ dài $N = K + M_1 + M_2$, thì các khối chẵn lẻ $P = (p_0, p_1, \dots, p_{M_1+M_2-1})$ từ khối thông tin S có thể được mã hoá có hệ thống.

Vì vậy, từ mã LDPC có thể là $\Lambda = (s_0, s_1, \dots, s_{K-1}, p_0, p_1, \dots, p_{M_1+M_2-1})$.

Theo sáng chế, M_1 và M_2 lần lượt là kích thước của các ma trận con chẵn lẻ tương ứng với ma trận con hai đường chéo B và ma trận con đơn vị D, trong đó $M_1 = g$ và $M_2 = N-K-g$.

Quy trình tính các bit chẵn lẻ có thể được trình bày dưới đây. Để cho dễ hiểu, một ví dụ về trường hợp trong đó ma trận kiểm tra chẵn lẻ 20 được xác định như thể hiện trong bảng 1 nêu trên sẽ được mô tả dưới đây.

Bước 1) Thiết lập giá trị ban đầu cho λ_i bằng s_i ($i = 0, 1, \dots, K-1$) và thiết lập giá trị ban đầu cho p_j bằng 0 ($j = 0, 1, \dots, M_1+M_2-1$).

Bước 2) Cộng bit thông tin thứ nhất λ_0 vào địa chỉ chứa bit chẵn lẻ được xác định ở hàng thứ nhất trong bảng 1 nêu trên.

Bước 3) Với $L-1$ bit thông tin kế tiếp λ_m ($m = 1, 2, \dots, L-1$), cộng bit thông tin λ_m vào địa chỉ chứa bit chẵn lẻ được tính dựa vào biểu thức 1 dưới đây:

$$\begin{aligned} (x+m \times Q_1) \bmod M_1 & \quad (\text{nếu } x < M_1) \\ M_1 + \{(x-M_1+m \times Q_2) \bmod M_2\} & \quad (\text{nếu } x \geq M_1) \end{aligned} \quad (1)$$

Trong biểu thức 1 nêu trên, x là địa chỉ của vùng chứa bit chẵn lẻ tương ứng với bit thông tin thứ nhất λ_0 . Ngoài ra, $Q_1 = M_1/L$ và $Q_2 = M_2/L$.

Trong trường hợp như vậy, vì độ dài của từ mã LDPC bằng 16200 và tỷ lệ mã bằng 3/15, nên $M_1 = 1080$, $M_2 = 11880$, $Q_1 = 3$, $Q_2 = 33$, $L = 360$.

Bước 4) Vì địa chỉ chứa bit chẵn lẻ ở hàng thứ hai trong bảng 1 nêu trên được sử dụng cho bit thông tin thứ L λ_L , giống như sơ đồ nêu trên, nên tính địa chỉ chứa bit chẵn lẻ cho $L-1$ bit thông tin kế tiếp λ_m ($m = L+1, L+2, \dots, 2L-1$) theo sơ đồ được mô tả ở bước 3) nêu trên. Trong trường hợp như vậy, x là địa chỉ của vùng chứa bit chẵn lẻ tương ứng với bit thông tin λ_L và giá trị của nó có thể lấy từ hàng thứ hai

trong bảng 1 nêu trên.

Bước 5) Với L bit thông tin mới trong mỗi nhóm bit, lấy các hàng mới trong bảng 1 nêu trên để làm địa chỉ của vùng chứa bit chẵn lẻ, và từ đó, thực hiện lặp lại quy trình nêu trên.

Bước 6) Sau khi thực hiện lặp lại quy trình nêu trên với các bit từ mã từ λ_0 đến λ_{K-1} , lần lượt tính các giá trị trong biểu thức 2 dưới đây bắt đầu từ $i = 1$:

$$p_i = p_i \oplus p_{i-1} \quad (i = 1, 2, \dots, M_1-1) \quad (2)$$

Bước 7) Tính các bit chẵn lẻ từ λ_K đến λ_{K+M_1-1} tương ứng với ma trận con hai đường chéo B dựa vào biểu thức 3 dưới đây:

$$\lambda_{K+L \times t+s} = p_{Q_1 \times s+t} \quad (0 \leq s < L, 0 \leq t < Q_1) \quad (3)$$

Bước 8) Tính địa chỉ của vùng chứa bit chẵn lẻ cho L bit từ mã mới từ λ_K đến λ_{K+M_1-1} của mỗi nhóm bit dựa vào hàng mới trong bảng 1 nêu trên và biểu thức 1 nêu trên.

Bước 9) Sau khi đã tính địa chỉ cho các bit từ mã từ λ_K đến λ_{K+M_1-1} , tính các bit chẵn lẻ từ λ_{K+M_1} đến $\lambda_{K+M_1+M_2-1}$ tương ứng với ma trận con D dựa vào biểu thức 4 dưới đây:

$$\lambda_{K+M_1+L \times t+s} = p_{M_1+Q_2 \times s+t} \quad (0 \leq s < L, 0 \leq t < Q_2) \quad (4)$$

Nhờ đó, có thể tính được các bit chẵn lẻ theo sơ đồ nêu trên. Tuy nhiên, đó chỉ là một ví dụ, và vì vậy, sơ đồ để tính các bit chẵn lẻ dựa trên ma trận kiểm tra chẵn lẻ như được thể hiện trên Fig.2 có thể được xác định theo nhiều cách khác nhau.

Như vậy, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC dựa vào bảng 1 nêu trên để tạo ra từ mã LDPC.

Cụ thể, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC trên 3240 bit

đầu vào, tức là các bit thông tin LDPC, ở tỷ lệ mã bằng 3/15 dựa vào bảng 1 nêu trên để tạo ra 12960 bit chẵn lẻ LDPC, và xuất ra các bit chẵn lẻ LDPC và từ mã LDPC có các bit chẵn lẻ LDPC đó. Trong trường hợp như vậy, từ mã LDPC có thể gồm có 16200 bit.

Ví dụ khác, ma trận kiểm tra chẵn lẻ theo phương án làm ví dụ thực hiện sáng chế có thể có cấu trúc như được thể hiện trên Fig.3.

Dựa vào Fig.3, ma trận kiểm tra chẵn lẻ 30 gồm có ma trận con thông tin 31 là ma trận con tương ứng với các bit thông tin (tức là các bit thông tin LDPC), và ma trận con chẵn lẻ 32 là ma trận con tương ứng với các bit chẵn lẻ (tức là các bit chẵn lẻ LDPC).

Ma trận con thông tin 31 có K_{ldpc} cột và ma trận con chẵn lẻ 32 có $N_{ldpc_parity} = N_{inner} - K_{ldpc}$ cột. Số lượng hàng của ma trận kiểm tra chẵn lẻ 30 bằng số lượng $N_{ldpc_parity} = N_{inner} - K_{ldpc}$ cột của ma trận con chẵn lẻ 32.

Ngoài ra, trong ma trận kiểm tra chẵn lẻ 30, N_{inner} là độ dài của từ mã LDPC, K_{ldpc} là độ dài của các bit thông tin, và $N_{ldpc_parity} = N_{inner} - K_{ldpc}$ là độ dài của các bit chẵn lẻ.

Cấu trúc của ma trận con thông tin 31 và ma trận con chẵn lẻ 32 sẽ được mô tả dưới đây.

Ma trận con thông tin 31 là ma trận có K_{ldpc} cột (tức là từ cột thứ 0 đến cột thứ $(K_{ldpc}-1)$) và tuân thủ các quy tắc sau đây:

Thứ nhất, K_{ldpc} cột tạo nên ma trận con thông tin 31 được phân ra cứ M cột thì đưa vào cùng một nhóm và được phân chia ra thành tổng số K_{ldpc}/M nhóm cột. Các cột thuộc cùng một nhóm cột có mối quan hệ sao cho cột này so với cột kia có

khoảng cách dịch chuyển tuần hoàn bằng Q_{ldpc} . Điều này có nghĩa là, Q_{ldpc} có thể được coi là giá trị thông số khoảng cách dịch chuyển tuần hoàn đối với các cột trong một nhóm cột của ma trận con thông tin tạo nên ma trận kiểm tra chẵn lẻ 30.

Trong đó, M là khoảng lặp lại của mẫu cột trong ma trận con thông tin 31 (ví dụ $M = 360$), và Q_{ldpc} là khoảng cách dịch chuyển tuần hoàn của mỗi cột trong ma trận con thông tin 31. M là ước số chung của N_{inner} và K_{ldpc} , và được xác định sao cho thỏa mãn hệ thức $Q_{\text{ldpc}} = (N_{\text{inner}} - K_{\text{ldpc}})/M$. Trong đó, M và Q_{ldpc} là các số nguyên, và K_{ldpc}/M cũng là số nguyên. M và Q_{ldpc} có thể có các giá trị khác nhau tùy theo độ dài của từ mã LDPC và tỷ lệ mã.

Ví dụ, khi $M = 360$, độ dài N_{inner} của từ mã LDPC bằng 16200, và tỷ lệ mã bằng 6/15, thì Q_{ldpc} có thể bằng 27.

Thứ hai, nếu bậc (trong đó, bậc là số lượng giá trị 1 có mặt trong cột và tất cả các cột thuộc cùng một nhóm cột đều có cùng một bậc) của cột thứ 0 trong nhóm cột thứ i ($i = 0, 1, \dots, K_{\text{ldpc}}/M - 1$) được gọi là D_i , và vị trí (hoặc chỉ số) của mỗi hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i được gọi là $R_{i,0}^{(0)}, R_{i,0}^{(1)}, \dots, R_{i,0}^{(D_i-1)}$, thì chỉ số $R_{i,j}^{(k)}$ của hàng có chứa giá trị 1 thứ k trong cột thứ j của nhóm cột thứ i được xác định theo biểu thức 5 dưới đây:

$$R_{i,j}^{(k)} = R_{i,(j-1)}^{(k)} + Q_{\text{ldpc}} \bmod (N_{\text{inner}} - K_{\text{ldpc}}) \quad (5)$$

Trong biểu thức 5 nêu trên, $k = 0, 1, 2, \dots, D_i - 1$; $i = 0, 1, \dots, K_{\text{ldpc}}/M - 1$; và $j = 1, 2, \dots, M - 1$.

Biểu thức 5 nêu trên có thể được biểu diễn dưới dạng biểu thức 6 dưới đây:

$$R_{i,j}^{(k)} = (R_{i,0}^{(k)} + (j \bmod M) \times Q_{\text{ldpc}}) \bmod (N_{\text{inner}} - K_{\text{ldpc}}) \quad (6)$$

Trong biểu thức 6 nêu trên, $k = 0, 1, 2, \dots, D_i - 1$; $i = 0, 1, \dots, K_{\text{ldpc}}/M - 1$; và $j = 1, 2, \dots, M - 1$.

2, ..., M-1. Trong biểu thức 6 nêu trên, vì $j = 1, 2, \dots, M-1$, nên $(j \bmod M)$ có thể được coi là bằng j .

Trong các biểu thức nêu trên, $R_{i,j}^{(k)}$ là chỉ số của hàng có chứa giá trị 1 thứ k trong cột thứ j của nhóm cột thứ i , N_{inner} là độ dài của từ mã LDPC, K_{ldpc} là độ dài của các bit thông tin, D_i là bậc của các cột thuộc nhóm cột thứ i , M là số lượng cột thuộc một nhóm cột, và Q_{ldpc} là khoảng cách dịch chuyển tuần hoàn của mỗi cột trong nhóm cột.

Từ đó, dựa vào các biểu thức nêu trên, nếu biết giá trị $R_{i,0}^{(k)}$, thì có thể tìm được chỉ số $R_{i,j}^{(k)}$ của hàng có chứa giá trị 1 thứ k trong cột thứ j của nhóm cột thứ i . Vì vậy, khi đã lưu trữ giá trị chỉ số của hàng có chứa giá trị 1 thứ k trong cột thứ 0 của mỗi nhóm cột, thì có thể biết được vị trí của cột và hàng có chứa giá trị 1 trong ma trận kiểm tra chẵn lẻ 30 (tức là ma trận con thông tin 31 của ma trận kiểm tra chẵn lẻ 30) với cấu trúc được thể hiện trên Fig.3.

Theo các quy tắc nêu trên, tất cả các bậc của các cột thuộc nhóm cột thứ i đều là D_i . Vì vậy, theo các quy tắc nêu trên, mã LDPC để lưu trữ thông tin về ma trận kiểm tra chẵn lẻ có thể được biểu diễn ngắn gọn như sau.

Ví dụ, khi N_{inner} bằng 30, K_{ldpc} bằng 15, và Q_{ldpc} bằng 3, thì thông tin vị trí của hàng có chứa giá trị 1 trong cột thứ 0 của các nhóm có ba cột có thể được biểu diễn bằng dãy trong biểu thức 7 dưới đây, dãy này có thể được gọi là ‘dãy vị trí của giá trị 1’.

$$\begin{aligned} R_{1,0}^{(1)} &= 1, R_{1,0}^{(2)} = 2, R_{1,0}^{(3)} = 8, R_{1,0}^{(4)} = 10, \\ R_{2,0}^{(1)} &= 0, R_{2,0}^{(2)} = 9, R_{2,0}^{(3)} = 13, \\ R_{3,0}^{(1)} &= 0, R_{3,0}^{(2)} = 14. \end{aligned} \tag{7}$$

Trong biểu thức 7 nêu trên, $R_{i,j}^{(k)}$ là chỉ số của hàng có chứa giá trị 1 thứ k trong cột thứ j của nhóm cột thứ i .

Dãy vị trí của giá trị 1 theo biểu thức 7 nêu trên thể hiện chỉ số của hàng có chứa giá trị 1 trong cột thứ 0 của mỗi nhóm cột có thể được biểu diễn ngắn gọn trong bảng 2 dưới đây:

Bảng 2

1	2	8	10
0	9	13	
0	14		

Bảng 2 nêu trên thể hiện vị trí của các phần tử có giá trị 1 trong ma trận kiểm tra chẵn lẻ, và dãy vị trí của giá trị 1 thứ i được biểu diễn bằng chỉ số của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i .

Ma trận con thông tin 31 của ma trận kiểm tra chẵn lẻ theo phương án làm ví dụ thực hiện sáng chế đã mô tả trên đây có thể được xác định dựa vào bảng 3 dưới đây.

Theo sáng chế, bảng 3 dưới đây thể hiện chỉ số của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i trong ma trận con thông tin 31. Điều này có nghĩa là, ma trận con thông tin 31 có nhiều nhóm cột, mỗi nhóm cột có M cột, và vị trí của các giá trị 1 ở cột thứ 0 của mỗi nhóm cột trong số các nhóm cột này có thể được xác định như được thể hiện trong bảng 3 dưới đây.

Ví dụ, khi độ dài N_{inner} của từ mã LDPC bằng 16200, tỷ lệ mã bằng 6/15, và M bằng 360, thì chỉ số của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i trong ma trận con thông tin 31 được thể hiện trong bảng 3 dưới đây.

Bảng 3

27 430 519 828 1897 1943 2513 2600 2640 3310 3415 4266 5044 5100 5328
5483 5928 6204 6392 6416 6602 7019 7415 7623 8112 8485 8724 8994 9445
9667
27 174 188 631 1172 1427 1779 2217 2270 2601 2813 3196 3582 3895 3908
3948 4463 4955 5120 5809 5988 6478 6604 7096 7673 7735 7795 8925 9613
9670
27 370 617 852 910 1030 1326 1521 1606 2118 2248 2909 3214 3413 3623 3742
3752 4317 4694 5300 5687 6039 6100 6232 6491 6621 6860 7304 8542 8634
990 1753 7635 8540
933 1415 5666 8745
27 6567 8707 9216
2341 8692 9580 9615
260 1092 5839 6080
352 3750 4847 7726
4610 6580 9506 9597
2512 2974 4814 9348
1461 4021 5060 7009
1796 2883 5553 8306
1249 5422 7057
3965 6968 9422
1498 2931 5092
27 1090 6215
26 4232 6354

Theo phương án làm ví dụ khác để thực hiện sáng chế, ma trận kiểm tra chẵn lẻ trong đó có thay đổi thứ tự của các chỉ số trong mỗi dãy tương ứng với mỗi nhóm cột trong bảng 3 nêu trên cũng được coi là ma trận kiểm tra chẵn lẻ cho mã LDPC giống như ma trận kiểm tra chẵn lẻ nêu trên.

Theo phương án làm ví dụ khác nữa để thực hiện sáng chế, ma trận kiểm tra chẵn lẻ trong đó có thay đổi thứ tự sắp xếp của các dãy tương ứng với các nhóm cột trong bảng 3 nêu trên cũng được coi là ma trận kiểm tra chẵn lẻ giống như ma trận kiểm tra chẵn lẻ nêu trên là vì các ma trận này có các đặc trưng đại số như đặc trưng tuần hoàn và sơ đồ phân bố bậc giống nhau trên đồ thị của mã.

Theo phương án làm ví dụ khác nữa để thực hiện sáng chế, ma trận kiểm tra

chẵn lẻ trong đó một bội số của Q_{ldpc} được cộng vào tất cả các chỉ số của dãy tương ứng với nhóm cột trong bảng 3 nêu trên cũng được coi là ma trận kiểm tra chẵn lẻ giống như ma trận kiểm tra chẵn lẻ nêu trên là vì các ma trận này có các đặc trưng tuần hoàn và sơ đồ phân bố bậc giống nhau trên đồ thị của mã. Trong đó, cần lưu ý rằng nếu giá trị thu được sau khi cộng bội số của Q_{ldpc} vào dãy cho trước bằng hoặc lớn hơn $N_{inner} - K_{ldpc}$, thì cần phải thay thế giá trị đó bằng giá trị thu được sau khi lấy giá trị đó để thực hiện phép toán môđulô đối với $N_{inner} - K_{ldpc}$ và sau đó mới được dùng.

Khi đã xác định được vị trí của các hàng có chứa giá trị 1 trong cột thứ 0 của nhóm cột thứ i trong ma trận con thông tin 31 như được thể hiện trong bảng 3 nêu trên, thì có thể cho vị trí đó dịch chuyển tuần hoàn với khoảng cách Q_{ldpc} , và do đó, có thể xác định được vị trí của các hàng có chứa giá trị 1 trong các cột khác của mỗi nhóm cột.

Ví dụ, như được thể hiện trong bảng 3 nêu trên, vì dãy tương ứng với cột thứ 0 của nhóm cột thứ 0 trong ma trận con thông tin 31 là “27 430 519 828 1897 1943 2513 2600 2640 3310 3415 4266 5044 5100 5328 5483 5928 6204 6392 6416 6602 7019 7415 7623 8112 8485 8724 8994 9445 9667”, nên trong cột thứ 0 của nhóm cột thứ 0 trong ma trận con thông tin 31, giá trị 1 nằm ở hàng thứ 27, hàng thứ 430, hàng thứ 519, ...

Trong trường hợp như vậy, vì $Q_{ldpc} = (N_{inner} - K_{ldpc})/M = (16200 - 6480)/360 = 27$, nên chỉ số của các hàng có chứa giá trị 1 trong cột thứ nhất của nhóm cột thứ 0 có thể là 54 ($= 27+27$), 457 ($= 430+27$), 546 ($= 519+27$), ..., 81 ($= 54+27$), 484 ($= 457+27$), 573 ($= 546+27$), ...

Theo sơ đồ nêu trên, có thể xác định được chỉ số của các hàng có chứa giá trị 1 trong tất cả các hàng của mỗi nhóm cột.

Quy trình thực hiện bước mã hoá LDPC dựa trên ma trận kiểm tra chẵn lẻ 30 như được thể hiện trên Fig.3 sẽ được mô tả dưới đây.

Trước hết, gọi các bit thông tin cần mã hoá là $i_0, i_1, \dots, i_{K_{ldpc}-1}$, và gọi các bit mã được xuất ra sau khi mã hoá LDPC là $c_0, c_1, \dots, c_{N_{ldpc}-1}$.

Hơn nữa, vì mã LDPC là mã có hệ thống, cho nên với k ($0 \leq k < K_{ldpc}-1$), c_k được đặt bằng i_k . Các bit mã còn lại được đặt bằng $p_k := c_{k+K_{ldpc}}$.

Quy trình tính các bit chẵn lẻ p_k sẽ được mô tả dưới đây.

Dưới đây, $q(i,j,0)$ là mục thứ j của hàng thứ i trong danh sách chỉ số như ở trong bảng 3 nêu trên, và $q(i,j,l)$ được đặt bằng $q(i,j,0) + Q_{ldpc} \times l \bmod (N_{inner} - K_{ldpc})$ với $0 < i < 360$. Tất cả các phép cộng có thể được thực hiện bằng cách cộng trong trường Galois (Galois Field, GF) (2). Ngoài ra, trong bảng 3 nêu trên, vì độ dài của từ mã LDPC bằng 16200 và tỷ lệ mã bằng 6/15, nên Q_{ldpc} bằng 27.

Khi đã xác định được $q(i,j,0)$ và $q(i,j,l)$ như nêu trên, quy trình tính bit chẵn lẻ được thực hiện như sau.

Bước 1) Thiết lập giá trị ban đầu cho các bit chẵn lẻ bằng '0'. Điều này có nghĩa là, $p_k = 0$ với $0 \leq k < N_{inner} - K_{ldpc}$.

Bước 2) Với mọi k có giá trị $0 \leq k < K_{ldpc}$, i và l được đặt bằng $i := \lfloor k/360 \rfloor$ và $l := k \bmod 360$. Trong đó, $\lfloor x \rfloor$ là số nguyên lớn nhất không lớn hơn x .

Tiếp theo, với mọi i , cộng i_k vào $p_{q(i,j,l)}$. Nghĩa là, tính $p_{q(i,0,l)} = p_{q(i,0,l)} + i_k$, $p_{q(i,1,l)} = p_{q(i,1,l)} + i_k$, $p_{q(i,2,l)} = p_{q(i,2,l)} + i_k$, ..., $p_{q(i,w(i)-1,l)} = p_{q(i,w(i)-1,l)} + i_k$.

Theo sáng chế, $w(i)$ là số lượng giá trị (phần tử) của hàng thứ i trong danh sách

chỉ số như ở trong bảng 3 nêu trên và là số lượng giá trị 1 trong cột tương ứng với i_k trong ma trận kiểm tra chẵn lẻ. Ngoài ra, trong bảng 3 nêu trên, $q(i,j,0)$ là mục thứ j của hàng thứ i , là chỉ số của bit chẵn lẻ và là vị trí của các hàng có chứa giá trị 1 trong cột tương ứng với i_k trong ma trận kiểm tra chẵn lẻ.

Cụ thể, trong bảng 3 nêu trên, $q(i,j,0)$ là mục thứ j của hàng thứ i , là vị trí của các hàng có chứa giá trị 1 trong cột thứ nhất (tức là cột 0) của nhóm cột thứ i trong ma trận kiểm tra chẵn lẻ của mã LDPC.

$q(i,j,0)$ cũng có thể được coi là chỉ số của bit chẵn lẻ được tạo ra bằng cách mã hoá LDPC theo phương pháp cho phép thiết bị thực tế áp dụng sơ đồ để cộng i_k vào $p_{q(i,j,l)}$ với mọi i , và cũng có thể được coi là chỉ số ở dạng khác khi áp dụng sơ đồ mã hoá khác. Tuy nhiên, đó chỉ là một ví dụ, và vì vậy, rõ ràng là để thu được kết quả tương đương với kết quả mã hoá LDPC có thể thu được từ ma trận kiểm tra chẵn lẻ của mã LDPC về cơ bản được tạo ra dựa vào các giá trị $q(i,j,0)$ trong bảng 3 nêu trên thì có thể áp dụng sơ đồ mã hoá bất kỳ.

Bước 3) Tính bit chẵn lẻ p_k bằng cách tính $p_k = p_k + p_{k-1}$ với mọi k thoả mãn điều kiện $0 < k < N_{\text{inner}} - K_{\text{ldpc}}$.

Do đó, có thể thu được tất cả các bit mã $c_0, c_1, \dots, c_{N_{\text{ldpc}}-1}$.

Nhờ đó, có thể tính được các bit chẵn lẻ theo sơ đồ nêu trên. Tuy nhiên, đó chỉ là một ví dụ, và vì vậy, sơ đồ để tính các bit chẵn lẻ dựa trên ma trận kiểm tra chẵn lẻ như được thể hiện trên Fig.3 có thể được xác định theo nhiều cách khác nhau.

Như vậy, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC dựa vào bảng 3 nêu trên để tạo ra từ mã LDPC.

Cụ thể, bộ mã hoá LDPC 110 có thể thực hiện bước mã hoá LDPC trên 6480 bit

đầu vào, tức là các bit thông tin LDPC, ở tỷ lệ mã bằng 6/15 dựa vào bảng 3 nêu trên để tạo ra 9720 bit chẵn lẻ LDPC và xuất ra các bit chẵn lẻ LDPC và từ mã LDPC có các bit chẵn lẻ LDPC đó. Trong trường hợp như vậy, từ mã LDPC có thể gồm có 16200 bit.

Như đã nêu trên, bộ mã hoá LDPC 110 có thể mã hoá các bit đầu vào ở các tỷ lệ mã khác nhau để tạo ra từ mã LDPC gồm có các bit đầu vào và các bit chẵn lẻ LDPC.

Bộ phận lặp 120 lặp lại ít nhất một số bit của từ mã LDPC ở trong từ mã LDPC sao cho các bit được lặp lại của từ mã LDPC sẽ được truyền trong khung hiện thời. Trong đó, các bit được lặp lại ở trong từ mã LDPC được gọi là các bit lặp hoặc các bit được lặp lại. Ngoài ra, bộ phận lặp 120 có thể xuất ra từ mã LDPC lặp lại, tức là từ mã LDPC có các bit lặp lại, gọi là từ mã LDPC có bit lặp, để cung cấp cho bộ phận đọc lỗi 130. Ngoài ra, bộ phận lặp 120 có thể xuất ra từ mã LDPC có bit lặp để cung cấp cho bộ tạo bit chẵn lẻ bổ sung 140 và cung cấp thông tin lặp (ví dụ số lượng, vị trí, v.v., của các bit lặp) cho bộ tạo bit chẵn lẻ bổ sung 140.

Cụ thể, bộ phận lặp 120 có thể lặp lại một số lượng bit định trước trong số các bit từ mã LDPC (ví dụ N_{repeat} bit chẵn lẻ LDPC) ở vị trí định trước trong từ mã LDPC. Trong trường hợp như vậy, số lượng bit lặp có thể có nhiều giá trị khác nhau tùy theo từng hệ thống có thiết bị truyền tín hiệu 100 và/hoặc thiết bị thu tín hiệu 200.

Ví dụ, bộ phận lặp 120 có thể gán một số lượng bit định trước trong số các bit chẵn lẻ LDPC vào phía sau các bit thông tin LDPC trong từ mã LDPC gồm có các bit thông tin LDPC và các bit chẵn lẻ LDPC. Điều này có nghĩa là, bộ phận lặp 120 có thể gán ít nhất một số bit trong số các bit chẵn lẻ vào phía sau các bit đầu vào, tức là các bit thông tin LDPC.

Ngoài ra, bộ phận lặp 120 có thể gắn một số lượng bit định trước trong số các bit chẵn lẻ LDPC vào phía sau các bit chẵn lẻ LDPC, gắn một số lượng bit định trước trong số các bit chẵn lẻ LDPC vào vị trí định trước ở giữa các bit thông tin LDPC, hoặc gắn một số lượng bit định trước trong số các bit chẵn lẻ LDPC vào vị trí định trước ở giữa các bit chẵn lẻ LDPC.

Như vậy, vì một số lượng bit định trước trong số các bit chẵn lẻ LDPC trong từ mã LDPC có bit lặp có thể được lặp lại và được truyền thêm đến thiết bị thu tín hiệu 200, cho nên phương pháp nêu trên có thể được gọi là phương pháp lặp.

Ví dụ trong đó các bit được lặp lại theo các phương án làm ví dụ thực hiện sáng chế sẽ được mô tả dưới đây dựa vào các hình vẽ kèm theo.

Khi số lượng N_{repeat} bit được lặp lại bằng hoặc nhỏ hơn số lượng bit chẵn lẻ LDPC trong từ mã LDPC, bộ phận lặp 120 có thể gắn N_{repeat} bit trong số các bit chẵn lẻ LDPC, kể từ bit chẵn lẻ LDPC đầu tiên, vào phía sau các bit thông tin LDPC.

Ví dụ, khi N_{repeat} bằng hoặc nhỏ hơn $N_{\text{ldpc_parity}}$, tức là khi $N_{\text{repeat}} \leq N_{\text{ldpc_parity}}$, như được thể hiện trên Fig.4, bộ phận lặp 120 có thể gắn N_{repeat} bit đầu tiên ($p_0, p_1, \dots, p_{N_{\text{repeat}}-1}$) trong số các bit chẵn lẻ LDPC ($p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$) vào phía sau các bit thông tin LDPC ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}$).

Vì vậy, từ bit thứ nhất đến bit thứ N_{repeat} trong số các bit chẵn lẻ LDPC được gắn vào phía sau các bit thông tin LDPC, và N_{repeat} bit này được đặt vào giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC dưới dạng ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{repeat}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$).

Khi số lượng N_{repeat} bit được lặp lại lớn hơn số lượng bit chẵn lẻ LDPC, bộ phận lặp 120 có thể gắn tất cả các bit chẵn lẻ LDPC, dưới dạng là một phần của các bit lặp,

vào phía sau các bit thông tin LDPC, gắn các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit lập trừ đi số lượng bit chẵn lẻ LDPC, kể từ bit chẵn lẻ LDPC đầu tiên, vào phía sau các bit chẵn lẻ LDPC đã gắn lần trước.

Trong trường hợp như vậy, bộ phận lập 120 có thể gắn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit lập trừ đi số lượng bit chẵn lẻ LDPC, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC hiện có, tức là các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, chứ không phải là trong số các bit chẵn lẻ LDPC được lập lại, vào phía sau các bit chẵn lẻ LDPC đã gắn lần đầu.

Ví dụ, khi N_{repeat} lớn hơn $N_{\text{ldpc_parity}}$, tức là khi $N_{\text{repeat}} > N_{\text{ldpc_parity}}$, như được thể hiện trên Fig.5, bộ phận lập 120 gắn $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC ($p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$), dưới dạng là một phần của các bit lập, vào phía sau các bit thông tin LDPC ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}$). Ngoài ra, bộ phận lập 120 có thể gắn thêm $N_{\text{repeat}} - N_{\text{ldpc_parity}}$ bit ($p_0, p_1, \dots, p_{N_{\text{repeat}}-N_{\text{ldpc_parity}}-1}$) trong số các bit chẵn lẻ LDPC vào phía sau $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã gắn lần trước.

Vì vậy, $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC có thể được gắn vào phía sau các bit thông tin LDPC và $N_{\text{repeat}} - N_{\text{ldpc_parity}}$ bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC có thể được gắn thêm vào phía sau $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã gắn lần trước.

Vì vậy, N_{repeat} bit được đặt ở giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC, dưới dạng ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{repeat}}-N_{\text{ldpc_parity}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$).

Ví dụ nêu trên mô tả rằng các bit lập được gắn vào phía sau các bit thông tin LDPC, nhưng đó chỉ là một ví dụ. Theo phương án làm ví dụ khác để thực hiện sáng chế, bộ phận lập 120 có thể gắn các bit lập vào phía sau các bit chẵn lẻ LDPC.

Ví dụ, khi N_{repeat} bằng hoặc nhỏ hơn $N_{\text{ldpc_parity}}$, tức là khi $N_{\text{repeat}} \leq N_{\text{ldpc_parity}}$, như được thể hiện trên Fig.6, bộ phận lặp 120 có thể gán N_{repeat} bit ($p_0, p_1, \dots, p_{N_{\text{repeat}}-1}$) trong số các bit chẵn lẻ LDPC vào phía sau các bit chẵn lẻ LDPC ($p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$).

Vì vậy, từ bit thứ nhất đến bit thứ N_{repeat} trong số các bit chẵn lẻ LDPC được gán vào phía sau các bit chẵn lẻ LDPC, và N_{repeat} bit này được đặt vào phía sau các bit chẵn lẻ LDPC dưới dạng ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{repeat}}-1}$).

Ngoài ra, khi N_{repeat} lớn hơn $N_{\text{ldpc_parity}}$, tức là khi $N_{\text{repeat}} > N_{\text{ldpc_parity}}$, như được thể hiện trên Fig.7, bộ phận lặp 120 có thể gán $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC ($p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}$) vào phía sau các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Ngoài ra, bộ phận lặp 120 có thể gán thêm $N_{\text{repeat}} - N_{\text{ldpc_parity}}$ bit đầu tiên ($p_0, p_1, \dots, p_{N_{\text{repeat}}-N_{\text{ldpc_parity}}-1}$) trong số các bit chẵn lẻ LDPC vào phía sau $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã gán lần trước.

Vì vậy, $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC có thể được gán vào phía sau các bit chẵn lẻ LDPC dưới dạng là một phần của các bit lặp, và $N_{\text{repeat}} - N_{\text{ldpc_parity}}$ bit trong số các bit chẵn lẻ LDPC có thể được gán thêm, dưới dạng là một phần khác của các bit lặp, vào phía sau $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã gán lần trước.

Vì vậy, N_{repeat} bit $i_{K_{\text{ldpc}}-1}$ dưới dạng ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{repeat}}-N_{\text{ldpc_parity}}-1}$) được đặt vào phía sau các bit chẵn lẻ LDPC.

Ví dụ nêu trên mô tả rằng các bit chẵn lẻ LDPC ở phần đầu được lặp lại, nhưng đó chỉ là một ví dụ. Theo phương án làm ví dụ khác để thực hiện sáng chế, bộ phận lặp 120 có thể lặp lại các bit chẵn lẻ LDPC nằm ở các vị trí khác nhau chẳng hạn như

ở phần cuối và phần giữa của các bit chẵn lẻ LDPC.

Ví dụ nêu trên mô tả rằng chỉ có các bit chẵn lẻ LDPC trong từ mã LDPC được lặp lại, nhưng đó chỉ là một ví dụ. Theo phương án làm ví dụ khác để thực hiện sáng chế, các bit thông tin LDPC hoặc một số bit thông tin LDPC và một số bit chẵn lẻ LDPC cũng có thể được lặp lại.

Ví dụ nêu trên mô tả rằng phương pháp lặp được thực hiện, nhưng đó chỉ là một ví dụ. Trong một số trường hợp, phương pháp lặp cũng có thể không được thực hiện. Trong trường hợp như vậy, một số bit chẵn lẻ LDPC ở trong từ mã LDPC có thể được đọc lỗi bằng bộ phận đọc lỗi 130 như sẽ được mô tả dưới đây. Việc có thực hiện phương pháp lặp hay không có thể được xác định trước tùy theo từng hệ thống.

Bộ phận đọc lỗi 130 có thể đọc lỗi một số bit trong số các bit chẵn lẻ LDPC trong từ mã LDPC. Ngoài ra, bộ phận đọc lỗi 130 xuất ra từ mã LDPC đã được đọc lỗi, đó là phần còn lại của các bit từ mã LDPC, ngoại trừ các bit được đọc lỗi, tức là từ mã LDPC có đọc lỗi. Ngoài ra, bộ phận đọc lỗi 130 có thể cung cấp thông tin (ví dụ số lượng và vị trí của các bit được đọc lỗi, v.v.) liên quan đến các bit chẵn lẻ LDPC được đọc lỗi cho bộ tạo bit chẵn lẻ bổ sung 140.

Theo sáng chế, đọc lỗi có nghĩa là một số bit chẵn lẻ LDPC không được truyền đến thiết bị thu tín hiệu 200. Trong trường hợp như vậy, bộ phận đọc lỗi 130 có thể loại bỏ các bit chẵn lẻ LDPC được đọc lỗi hoặc chỉ xuất ra các bit còn lại, ngoại trừ các bit chẵn lẻ LDPC được đọc lỗi trong từ mã LDPC.

Cụ thể, bộ phận đọc lỗi 130 có thể đọc lỗi một số lượng bit định trước (ví dụ N_{punc} bit) trong số các bit chẵn lẻ LDPC. Theo sáng chế, số lượng N_{punc} bit được đọc lỗi bằng 0 hoặc là số nguyên dương, và có thể có nhiều giá trị khác nhau tùy theo từng

hệ thống. $N_{\text{punc}} = 0$ có nghĩa là phương pháp đục lỗ không được thực hiện.

Trong trường hợp như vậy, bộ phận đục lỗ 130 có thể đục lỗ số lượng bit định trước ở phần cuối của các bit chẵn lẻ LDPC. Ví dụ, bộ phận đục lỗ 130 có thể tuần tự đục lỗ N_{punc} bit chẵn lẻ LDPC kể từ bit chẵn lẻ LDPC cuối cùng.

Tuy nhiên, đó chỉ là một ví dụ và vị trí mà ở đó các bit được đục lỗ trong các bit chẵn lẻ LDPC có thể thay đổi theo nhiều cách khác nhau. Ví dụ, bộ phận đục lỗ 130 có thể đục lỗ N_{punc} bit chẵn lẻ LDPC nằm ở phần đầu hoặc phần giữa của các bit chẵn lẻ LDPC hoặc đục lỗ N_{punc} bit chẵn lẻ LDPC nằm ở các vị trí định trước trong các bit chẵn lẻ LDPC.

Ngoài ra, khi phương pháp lặp được thực hiện, bộ phận đục lỗ 130 có thể không đục lỗ các bit lặp, mà có thể đục lỗ một số lượng bit định trước trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Ví dụ, giả sử rằng phương pháp lặp được thực hiện để gắn N_{repeat} bit chẵn lẻ LDPC vào phía sau các bit thông tin LDPC.

Trong trường hợp như vậy, từ mã LDPC có bit lặp gồm có các bit lặp và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Trong trường hợp như vậy, các bit lặp nằm ở giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, và do đó, bộ phận đục lỗ 130 có thể đục lỗ N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Phương pháp đục lỗ theo các phương án làm ví dụ thực hiện sáng chế sẽ được mô tả dưới đây dựa vào các hình vẽ kèm theo từ Fig.8 đến Fig.11. Fig.8 đến Fig.11 thể hiện ví dụ để mô tả phương pháp đục lỗ khi phương pháp lặp được thực hiện như

được thể hiện trên các hình vẽ từ Fig.4 đến Fig.7.

Trước hết, như được thể hiện trên Fig.4, giả sử rằng N_{repeat} bit chẵn lẻ LDPC được gắn bằng phương pháp lặp vào phía sau các bit thông tin LDPC và trước các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, như được thể hiện trên Fig.8, bộ phận đọc lỗi 130 có thể đọc lỗi N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng trong số $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC.

Vì vậy, số lượng bit chẵn lẻ LDPC trong từ mã LDPC đã được lặp lại và đọc lỗi bằng $N_{\text{ldpc_parity}} + N_{\text{repeat}} - N_{\text{punc}}$ và có thể được biểu diễn dưới dạng $(p_0, p_1, \dots, p_{N_{\text{repeat}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-N_{\text{punc}}-1})$.

Ví dụ khác, như được thể hiện trên Fig.5, giả sử rằng N_{repeat} bit chẵn lẻ LDPC được gắn bằng phương pháp lặp vào phía sau các bit thông tin LDPC và trước các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, như được thể hiện trên Fig.9, bộ phận đọc lỗi 130 có thể đọc lỗi N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng trong số $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC.

Vì vậy, số lượng bit chẵn lẻ LDPC trong từ mã LDPC đã được lặp lại và đọc lỗi bằng $N_{\text{ldpc_parity}} + N_{\text{repeat}} - N_{\text{punc}}$ và có thể được biểu diễn dưới dạng $(p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-1}, p_0, p_1, \dots, p_{N_{\text{repeat}}-N_{\text{ldpc_parity}}-1}, p_0, p_1, \dots, p_{N_{\text{ldpc}}-K_{\text{ldpc}}-N_{\text{punc}}-1})$.

Ví dụ khác, như được thể hiện trên Fig.6, giả sử rằng N_{repeat} bit chẵn lẻ LDPC được gắn bằng phương pháp lặp vào phía sau các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, như được thể hiện trên Fig.10, bộ phận đọc lỗi 130 có

thể đọc lỗi N_{punc} bit kể từ bit cuối cùng trong số $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC.

Vì vậy, số lượng bit chẵn lẻ LDPC trong từ mã LDPC đã được lặp lại và đọc lỗi bằng $N_{\text{ldpc_parity}} + N_{\text{repeat}} - N_{\text{punc}}$ và có thể được biểu diễn dưới dạng $(p_0, p_1, \dots, p_{N_{\text{ldpc}} - K_{\text{ldpc}} - N_{\text{punc}} - 1}, p_0, p_1, \dots, p_{N_{\text{repeat}} - 1})$.

Ví dụ khác, như được thể hiện trên Fig.7, giả sử rằng N_{repeat} bit chẵn lẻ LDPC được gắn bằng phương pháp lặp vào phía sau các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, như được thể hiện trên Fig.11, bộ phận đọc lỗi 130 có thể đọc lỗi N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng trong số $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC.

Vì vậy, số lượng bit chẵn lẻ LDPC trong từ mã LDPC đã được lặp lại và đọc lỗi bằng $N_{\text{ldpc_parity}} + N_{\text{repeat}} - N_{\text{punc}}$ và có thể được biểu diễn dưới dạng $(p_0, p_1, \dots, p_{N_{\text{ldpc}} - K_{\text{ldpc}} - N_{\text{punc}} - 1}, p_0, p_1, \dots, p_{N_{\text{ldpc}} - K_{\text{ldpc}} - 1}, p_0, p_1, \dots, p_{N_{\text{repeat}} - N_{\text{ldpc_parity}} - 1})$.

Bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ để tạo ra các bit chẵn lẻ bổ sung được truyền trong khung đứng trước.

Trong trường hợp như vậy, các bit chẵn lẻ bổ sung có thể được chọn trong số các bit chẵn lẻ LDPC được tạo ra dựa vào các bit thông tin được truyền trong khung hiện thời sẽ được truyền đến thiết bị thu tín hiệu 200 trong một khung đứng trước khung hiện thời, tức là khung trước đó.

Cụ thể, các bit đầu vào chứa các bit thông tin được mã hoá LDPC, và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được gắn vào các bit đầu vào để tạo nên từ mã LDPC.

Ngoài ra, phương pháp lặp, đọc lỗi và rút gọn được thực hiện trên từ mã LDPC,

và từ mã LDPC đã được lập lại, đục lỗ và rút gọn, tức là các bit từ mã LDPC có các bit lặp, ngoại trừ các bit được đục lỗ và các bit được rút gọn, có thể được ánh xạ lên một khung sẽ được truyền đến thiết bị thu tín hiệu 200. Tuy nhiên, khi phương pháp lập không được thực hiện, từ mã LDPC đã được đục lỗ và rút gọn có thể được ánh xạ lên khung sẽ được truyền đến thiết bị thu tín hiệu 200.

Trong trường hợp như vậy, các bit thông tin tương ứng với mỗi khung có thể được truyền đến thiết bị thu tín hiệu 200 trong mỗi khung, cùng với các bit chắn lẻ LDPC. Ví dụ, từ mã LDPC đã được lập lại, đục lỗ và rút gọn chứa các bit thông tin tương ứng với khung thứ $(i-1)$ có thể được ánh xạ lên khung thứ $(i-1)$ sẽ được truyền đến thiết bị thu tín hiệu 200, và từ mã LDPC đã được lập lại, đục lỗ và rút gọn chứa các bit thông tin tương ứng với khung thứ i có thể được ánh xạ lên khung thứ i sẽ được truyền đến thiết bị thu tín hiệu 200.

Bộ tạo bit chắn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit chắn lẻ LDPC được tạo ra dựa vào các bit thông tin được truyền trong khung thứ i để tạo ra các bit chắn lẻ bổ sung.

Cụ thể, một số bit chắn lẻ LDPC được tạo ra bằng cách mã hoá LDPC trên các bit thông tin được đục lỗ và sau đó không được truyền đến thiết bị thu tín hiệu 200. Trong trường hợp như vậy, bộ tạo bit chắn lẻ bổ sung 140 có thể chọn một số hoặc tất cả các bit chắn lẻ LDPC được đục lỗ trong số các bit chắn lẻ LDPC được tạo ra bằng cách mã hoá LDPC trên các bit thông tin được truyền trong khung thứ i , từ đó tạo ra các bit chắn lẻ bổ sung.

Ngoài ra, bộ tạo bit chắn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit chắn lẻ LDPC được truyền đến thiết bị thu tín hiệu 200 trong khung thứ i để

tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, các bit chẵn lẻ LDPC có trong từ mã LDPC đã được lặp lại, đọc lỗi và rút gọn được ánh xạ lên khung thứ i có thể gồm có các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và các bit chẵn lẻ LDPC được lặp lại.

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC có trong từ mã LDPC đã được lặp lại, đọc lỗi và rút gọn sẽ được ánh xạ lên khung thứ i để tạo ra các bit chẵn lẻ bổ sung. Tuy nhiên, khi phương pháp lặp không được thực hiện, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC có trong từ mã LDPC đã được rút gọn và đọc lỗi sẽ được ánh xạ lên khung thứ i để tạo ra các bit chẵn lẻ bổ sung.

Các bit chẵn lẻ bổ sung có thể được truyền đến thiết bị thu tín hiệu 200 trong khung đứng trước khung thứ i , tức là khung thứ $(i-1)$.

Điều này có nghĩa là, thiết bị truyền tín hiệu 100 có thể không chỉ truyền từ mã LDPC đã được lặp lại, đọc lỗi và rút gọn chứa các bit thông tin tương ứng với khung thứ $(i-1)$ mà còn truyền các bit chẵn lẻ bổ sung đã tạo ra được chọn trong số các bit chẵn lẻ LDPC được tạo ra dựa vào các bit thông tin được truyền trong khung thứ i đến thiết bị thu tín hiệu 200 trong khung thứ $(i-1)$.

Ví dụ nêu trên mô tả rằng các bit chẵn lẻ bổ sung được truyền đến thiết bị thu tín hiệu 200 trong khung thứ $(i-1)$, nhưng đó chỉ là một ví dụ. Vì vậy, các bit chẵn lẻ bổ sung có thể được truyền đến thiết bị thu tín hiệu 200 trong một khung được truyền trước khung thứ i xét về mặt thời gian.

Phương pháp tạo ra bit chẵn lẻ bổ sung bằng cách chọn các bit trong số các bit

chẵn lẻ LDPC sẽ được mô tả dưới đây.

Bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra trong số các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi số lượng bit chẵn lẻ LDPC được đọc lỗi bằng hoặc lớn hơn số lượng bit chẵn lẻ bổ sung được tạo ra, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Khi số lượng bit chẵn lẻ LDPC được đọc lỗi nhỏ hơn số lượng bit chẵn lẻ bổ sung được tạo ra, thì bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và sau đó chọn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC có trong từ mã LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi phương pháp lặp không được thực hiện, thì các bit chẵn lẻ LDPC có trong từ mã LDPC là các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và sau đó chọn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit LDPC đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Theo sáng chế, các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được

phân chia ra thành các bit chẵn lẻ LDPC không được đọc lỗi và các bit chẵn lẻ LDPC được đọc lỗi. Vì vậy, khi phương pháp đọc lỗi được thực hiện kể từ bit cuối cùng trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, nếu các bit được chọn kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC để tạo ra các bit chẵn lẻ bổ sung, thì các bit đó có thể được chọn theo thứ tự lần lượt là các bit chẵn lẻ LDPC không được đọc lỗi và các bit chẵn lẻ LDPC được đọc lỗi.

Khi phương pháp lập được thực hiện, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit của từ mã LDPC có bit lập để tạo ra các bit chẵn lẻ bổ sung.

Như đã nêu trên, các bit chẵn lẻ LDPC của từ mã LDPC có bit lập gồm có các bit lập và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và sau đó chọn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit lập và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Vì vậy, khi các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi đã chọn thêm, thì các bit lập được chọn trước tiên, và khi số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi lớn hơn số lượng bit lập, thì các bit có thể được chọn thêm trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Trong trường hợp như vậy, khi các bit được chọn thêm trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, thì có thể bắt đầu chọn bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra

bằng cách mã hoá LDPC.

Như đã nêu trên, các bit lặp có thể được đặt ở các vị trí khác nhau trong từ mã LDPC có bit lặp.

Phương pháp tạo ra bit chẵn lẻ bổ sung khi phương pháp lặp được thực hiện sẽ được mô tả chi tiết dưới đây dựa vào, ví dụ, trường hợp trong đó các bit chẵn lẻ LDPC được lặp lại nằm ở giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, giả sử rằng bộ phận lặp 120 chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC và gắn các bit chẵn lẻ đã chọn vào phía sau các bit thông tin LDPC, và bộ phận đọc lỗi 130 thực hiện phương pháp đọc lỗi kể từ bit cuối cùng trong số các bit chẵn lẻ LDPC gồm có các bit chẵn lẻ LDPC được lặp lại và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá.

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit lặp được gắn vào phía sau các bit đầu vào, tức là các bit thông tin LDPC, dựa vào số lượng bit chẵn lẻ bổ sung được tạo ra và số lượng bit chẵn lẻ LDPC được đọc lỗi, để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, thì bộ tạo bit chẵn lẻ bổ sung 140 chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn các bit tương ứng với số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit lặp, để tạo ra các bit chẵn lẻ bổ sung.

Theo sáng chế, để tạo ra các bit chẵn lẻ bổ sung, nếu các bit được chọn kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC, thì các bit đó có thể được chọn theo thứ tự lần

lượt là các bit lẻ và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Ngoài ra, trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, các bit bổ sung có thể được chọn theo thứ tự lần lượt là các bit chẵn lẻ LDPC không được đọc lỗi và các bit chẵn lẻ LDPC được đọc lỗi.

Như vậy, khi các bit chẵn lẻ bổ sung được tạo ra với số lượng đúng bằng số lượng định trước, thì các bit được đọc lỗi sẽ được ưu tiên chọn trước nhất, tuy nhiên không nhất thiết phải làm đúng như vậy. Ngoài ra, khi chọn các bit với số lượng đúng bằng số lượng vượt quá các bit được đọc lỗi, thì sẽ ưu tiên chọn các bit chẵn lẻ LDPC được lặp lại trong số các bit chẵn lẻ LDPC tùy thuộc vào việc phương pháp lặp có được thực hiện hay không.

Như vậy, có thể đạt được độ tăng ích do mã hoá bởi vì các bit được đọc lỗi không được truyền trong khung hiện thời được chọn, mà được truyền dưới dạng các bit chẵn lẻ bổ sung. Ngoài ra, sau khi chọn các bit được đọc lỗi, thì sẽ chọn các bit chẵn lẻ LDPC được lặp lại, đó là các bit tương đối quan trọng hơn trong số các bit chẵn lẻ LDPC, để tạo ra các bit chẵn lẻ bổ sung. Ngoài ra, các bit chẵn lẻ LDPC được sắp xếp theo thứ tự đọc lỗi, và do đó, có thể coi là các bit chẵn lẻ LDPC được sắp xếp theo thứ tự ưu tiên của các bit chẵn lẻ. Các vấn đề có liên quan đến thứ tự đọc lỗi sẽ được mô tả chi tiết dưới đây.

Khi phương pháp đọc lỗi không được thực hiện, tức là khi số lượng bit được đọc lỗi bằng 0, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit của từ mã LDPC hoặc từ mã LDPC lặp lại để tạo ra các bit chẵn lẻ bổ sung.

Trước hết, khi phương pháp lặp không được thực hiện, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung

được tạo ra, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC, để tạo ra các bit chẵn lẻ bổ sung. Điều này có nghĩa là, khi số lượng bit được đọc lỗi bằng 0 và số lượng bit lặp bằng 0, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Khi phương pháp lặp được thực hiện, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được lặp lại, để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi số lượng bit được đọc lỗi bằng 0 và số lượng bit lặp bằng hoặc lớn hơn 1, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra, kể từ bit đầu tiên trong số các bit được lặp lại của các bit lặp và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Vì vậy, khi các bit lặp được chọn trước tiên và số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit lặp lớn hơn số lượng bit lặp, thì các bit có thể được chọn thêm trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Trong trường hợp như vậy, khi các bit được chọn thêm trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, thì có thể bắt đầu chọn bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Các bit được đọc lỗi có nghĩa là các bit được đọc lỗi dựa vào từ mã LDPC đã được đọc lỗi sẽ được truyền trong khung mà các bit thông tin được truyền ở trong đó.

Trong ví dụ nêu trên trong đó số lượng bit chẵn lẻ bổ sung được tạo ra (dưới đây

gọi là N_{AP}) lớn hơn số lượng bit chẵn lẻ LDPC được đục lỗ, sau khi chọn tất cả các bit chẵn lẻ LDPC được đục lỗ, để dùng làm các bit bổ sung lần đầu, thì sẽ chọn các bit bổ sung còn lại, tức là các bit tương ứng với số lượng $(N_{AP} - N_{punc})$ thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đục lỗ, kể từ bit đầu tiên trong số các bit lặp, nhưng đó chỉ là một ví dụ. Điều này có nghĩa là, bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể chọn thêm $N_{AP} - N_{punc}$ bit kể từ bit thông tin đầu tiên hoặc bit mã hoá ngoài đầu tiên.

Ngoài ra, khi các bit chẵn lẻ LDPC không được đục lỗ, thì bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit lặp để tạo ra các bit chẵn lẻ bổ sung.

Quy trình tính số lượng bit chẵn lẻ bổ sung được tạo ra sẽ được mô tả dưới đây.

Trước hết, bộ tạo bit chẵn lẻ bổ sung 140 tính số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung được tạo ra dựa vào biểu thức 8 dưới đây:

$$N_{AP_temp} = \min \left\{ \begin{array}{l} 0,5 \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat}) \\ (N_{ldpc_parity} + N_{punc} + N_{repeat}) \end{array} \right\}, K = 0, 1, 2 \quad (8)$$

$$\text{Trong biểu thức 8 nêu trên, } \min(a, b) = \begin{cases} a, & \text{khi } a \leq b \\ b, & \text{khi } b < a \end{cases}.$$

Ngoài ra, N_{ldpc_parity} là số lượng bit chẵn lẻ LDPC, và N_{punc} là số lượng bit chẵn lẻ LDPC được đục lỗ. Ngoài ra, N_{outer} là số lượng bit mã hoá ngoài. Trong đó, khi thao tác mã hoá ngoài được thực hiện bằng cách mã hoá BCH, thì N_{outer} là số lượng bit mã hoá BCH. Ngoài ra, N_{repeat} là số lượng bit lặp và khi phương pháp lặp không được thực hiện, thì $N_{repeat} = 0$.

Vì vậy, $N_{ldpc_parity} - N_{punc} + N_{repeat}$ là tổng số (tức là tổng số bit chẵn lẻ LDPC có trong từ mã LDPC đã được lặp lại, đục lỗ và rút gọn) của các bit chẵn lẻ LDPC sẽ

được truyền trong khung hiện thời mà các bit thông tin được truyền ở trong đó, và $N_{\text{outer}} + N_{\text{ldpc_parity}} - N_{\text{punc}} + N_{\text{repeat}}$ là tổng số của các bit từ mã LDPC (tức là tổng số của các bit từ mã LDPC đã được lặp lại, đục lỗ và rút gọn) sẽ được truyền trong khung hiện thời.

Ngoài ra, K là tỷ số giữa số lượng bit chẵn lẻ bổ sung và một nửa của tổng số bit tạo nên từ mã LDPC đã được lặp lại, đục lỗ và rút gọn. Trong đó, khi $K = 2$, thì số lượng bit chẵn lẻ bổ sung bằng tổng số bit của từ mã LDPC được truyền trong khung hiện thời.

Như vậy, số lượng bit chẵn lẻ bổ sung được tạo ra có thể được xác định dựa vào tổng số bit sẽ được truyền trong khung hiện thời.

Dựa vào Fig.12, theo phương án làm ví dụ thực hiện sáng chế, khi độ dài của các bit chẵn lẻ bổ sung được tính, thì tất cả các bit được đục lỗ, các bit lặp và các bit chẵn lẻ LDPC được chọn có xét đến hiệu quả và độ phức tạp, và sau đó, các bit sẽ không được chọn thêm nữa, để tạo ra các bit chẵn lẻ bổ sung. Điều này có nghĩa là, như được thể hiện trên Fig.12, độ dài của các bit chẵn lẻ bổ sung bằng hoặc nhỏ hơn $N_{\text{AP_max}} (= N_{\text{ldpc_parity}} + N_{\text{punc}} + N_{\text{repeat}})$, tức là độ dài của các bit chẵn lẻ bổ sung không lớn hơn $N_{\text{AP_max}} (= N_{\text{ldpc_parity}} + N_{\text{punc}} + N_{\text{repeat}})$.

Ví dụ, khi số lượng bit chẵn lẻ LDPC được đục lỗ bằng 3200 và $K = 2$, thì trường hợp này sẽ được coi như là số lượng bit chẵn lẻ bổ sung được tạo ra bằng 13000 ($= N_{\text{outer}} + N_{\text{ldpc_parity}} - N_{\text{punc}} = 6480 + 9720 - 3200$).

Trong trường hợp như vậy, vì số lượng bit chẵn lẻ LDPC được đục lỗ bằng 3200, cho nên khi tất cả các bit chẵn lẻ LDPC được đục lỗ để tạo ra các bit chẵn lẻ bổ sung được chọn và tất cả các bit chẵn lẻ LDPC được chọn, thì số lượng bit được chọn

bằng 12920 ($= 3200 + 9720$). Vì vậy, khi không có giới hạn cụ thể, thì có thể chọn thêm 80 bit. Tuy nhiên, theo biểu thức 12 dưới đây, khi độ dài tối đa của các bit chẵn lẻ bổ sung bị giới hạn ở mức $N_{ldpc_parity} + N_{punc} + N_{repeat}$, thì số lượng bit chẵn lẻ bổ sung bị giới hạn ở 12920 và sẽ không cần chọn thêm 80 bit.

Tuy nhiên, sự giới hạn độ dài tối đa của các bit chẵn lẻ bổ sung như vậy chỉ là một ví dụ, và khi độ dài của các bit chẵn lẻ bổ sung không bị giới hạn, thì số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung được tạo ra có thể được tính dựa vào biểu thức 9 dưới đây:

$$N_{AP_temp} = 0,5 \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat}), K = 0, 1, 2 \quad (9)$$

Khi độ dài của các bit chẵn lẻ bổ sung không bị giới hạn, thì bộ tạo bit chẵn lẻ bổ sung 140 có thể tính số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung dựa vào biểu thức 9 nêu trên.

Bộ tạo bit chẵn lẻ bổ sung 140 có thể tính số lượng N_{AP} bit chẵn lẻ bổ sung được tạo ra dựa vào số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung được tạo ra được tính dựa vào biểu thức 8 hoặc biểu thức 9 nêu trên.

Cụ thể, bộ tạo bit chẵn lẻ bổ sung 140 có thể tính số lượng N_{AP} bit chẵn lẻ bổ sung dựa vào biểu thức 10 dưới đây.

Như vậy, số lượng N_{AP} bit chẵn lẻ bổ sung được tạo ra có thể được tính dựa vào số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung được tạo ra được tính dựa vào biểu thức 8 hoặc biểu thức 9 nêu trên, và, cụ thể, có thể được tính dựa vào biểu thức 10 dưới đây:

$$N_{AP} = \left\lfloor \frac{N_{AP_temp}}{\eta_{MOD}} \right\rfloor \times \eta_{MOD} \quad (10)$$

Trong biểu thức 10 nêu trên, η_{MOD} là bậc điều biến. Ví dụ, trong sơ đồ điều biến

địch pha vuông góc (Quadrature Phase Shift Keying, QPSK), sơ đồ điều biến biên độ vuông góc có 16 điểm (16-Quadrature Amplitude Modulation, 16-QAM), sơ đồ điều biến biên độ vuông góc có 64 điểm (64-Quadrature Amplitude Modulation, 64-QAM) và sơ đồ điều biến biên độ vuông góc có 256 điểm (256-Quadrature Amplitude Modulation, 256-QAM), η_{MOD} có thể lần lượt bằng 2, 4, 6 hoặc 8.

Vì vậy, số lượng bit chẵn lẻ bổ sung có thể là một bội số nguyên lần của bậc điều biến. Điều này có nghĩa là, vì các bit chẵn lẻ bổ sung được điều biến riêng biệt với các bit thông tin được ánh xạ lên các ký hiệu chòm điểm, nên số lượng bit chẵn lẻ bổ sung có thể được xác định là một bội số nguyên lần của bậc điều biến, giống như trong biểu thức 10 nêu trên.

Trong trường hợp như vậy, biểu thức 8 nêu trên có thể được biểu diễn dưới dạng biểu thức 11 dưới đây, và biểu thức 9 nêu trên có thể được biểu diễn dưới dạng biểu thức 12 dưới đây:

$$N_{AP} = \left\lfloor \frac{\min \left\{ \alpha \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat}), (N_{ldpc_parity} + N_{punc} + N_{repeat}) \right\}}{\eta_{MOD}} \right\rfloor \times \eta_{MOD} \quad (11)$$

$$N_{AP} = \left\lfloor \frac{\alpha \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat})}{\eta_{MOD}} \right\rfloor \times \eta_{MOD} \quad (12)$$

Trong biểu thức 11 và biểu thức 12 nêu trên, α có thể bằng 0,5.

Như vậy, số lượng bit chẵn lẻ bổ sung được tạo ra có thể được xác định dựa vào số lượng bit mã hoá ngoài được truyền trong khung hiện thời và số lượng bit chẵn lẻ còn lại sau khi đọc lỗi.

Theo sáng chế, khi phương pháp lập được thực hiện, số lượng bit chẵn lẻ bổ

sung được tạo ra có thể được xác định dựa vào số lượng bit mã hoá ngoài được truyền trong khung hiện thời, số lượng bit chẵn lẻ còn lại sau khi đục lỗ và số lượng bit được lấp lại trong khung hiện thời.

Tỷ lệ mã có thay đổi tùy thuộc vào việc sử dụng các bit chẵn lẻ bổ sung sẽ được mô tả dưới đây.

Nếu tỷ lệ mã R khi các bit chẵn lẻ bổ sung không được truyền bằng k/n , thì tỷ lệ mã R_{ap} khi các bit chẵn lẻ bổ sung được truyền bằng $k/(n + N_{AP})$ và N_{AP} có giá trị bằng $1/2 \times n$ hoặc n tùy thuộc vào giá trị của K . Vì vậy, tỷ lệ mã R_{ap} trong trường hợp các bit chẵn lẻ bổ sung được truyền bằng $k/(3/2 \times n) = 2/3R$ hoặc bằng $k/(2 \times n) = 1/2R$, và như vậy, nếu so sánh với trường hợp trong đó các bit chẵn lẻ bổ sung không được truyền, thì tỷ lệ mã giảm xuống còn $2/3$ hoặc $1/2$, do đó có thể đạt được độ tăng ích do mã hoá. Ngoài ra, các bit không phải là các bit chẵn lẻ bổ sung và các bit chẵn lẻ bổ sung được truyền trong các khung khác nhau, và, do đó, có thể đạt được độ tăng ích do phân tập. Điều này có thể cho phép giữ nguyên các đặc trưng khi có sự thay đổi tỷ lệ mã phụ thuộc vào độ dài của tín hiệu đầu vào sao cho tương ứng với sự thay đổi tỷ lệ mã như đã mô tả trên đây bất kể độ dài của tín hiệu đầu vào, tức là độ dài của các bit thông tin đầu vào.

Phương pháp tạo ra bit chẵn lẻ bổ sung bằng cách chọn các bit trong số các bit chẵn lẻ LDPC sẽ được mô tả chi tiết dưới đây dựa vào các hình vẽ.

Bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng đã tính trong số các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng hoặc nhỏ hơn số lượng bit chẵn lẻ LDPC được đục lỗ, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với

số lượng đúng bằng số lượng đã tính kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗ để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ, các bit chẵn lẻ LDPC được gắn vào phía sau các bit thông tin LDPC bằng cách lặp lại và do đó từ mã LDPC có bit lặp được tạo nên gồm có các bit theo thứ tự lần lượt là các bit thông tin LDPC, các bit chẵn lẻ LDPC được lặp lại và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, từ mã LDPC lặp lại $V = (v_0, v_1, \dots, v_{N_{\text{inner}}+N_{\text{repeat}}-1})$ có thể được biểu diễn như được thể hiện trên Fig.13.

Cụ thể, khi N_{AP} bằng hoặc nhỏ hơn N_{punc} , tức là $N_{\text{AP}} \leq N_{\text{punc}}$, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗ như được thể hiện trên Fig.14 và Fig.15 để tạo ra các bit chẵn lẻ bổ sung.

Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}}, v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}+1}, \dots, v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}+N_{\text{AP}}-1})$.

Khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ LDPC được đọc lỗ, bộ tạo bit chẵn lẻ bổ sung 140 chọn tất cả các bit chẵn lẻ LDPC được đọc lỗ, và chọn các bit tương ứng với số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗ, kể từ bit đầu tiên trong số các bit lặp, để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ, khi N_{AP} lớn hơn N_{punc} , tức là $N_{\text{AP}} > N_{\text{punc}}$, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗ như được thể hiện trên Fig.16 và Fig.17. Vì vậy, trước tiên có thể chọn các bit $(v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}}, v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}+1}, \dots, v_{N_{\text{repeat}}+N_{\text{inner}}-1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn thêm $N_{\text{AP}} - N_{\text{punc}}$ bit kể từ

bit đầu tiên trong số các bit chẵn lẻ LDPC gồm có các bit chẵn lẻ LDPC được lặp lại và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, các bit chẵn lẻ LDPC được gắn vào phía sau các bit thông tin LDPC bằng phương pháp lặp, và do đó, các bit chẵn lẻ LDPC được lặp lại và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được sắp xếp tuần tự để tạo nên các bit chẵn lẻ LDPC trong từ mã LDPC.

Vì vậy, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn thêm các bit (tức là $N_{AP} - N_{punc}$ bit) với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được lặp lại. Trong trường hợp như vậy, vì các bit chẵn lẻ bổ sung được chọn kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được lặp lại, cho nên khi $N_{AP} - N_{punc}$ lớn hơn số lượng N_{repeat} bit chẵn lẻ LDPC được lặp lại, thì ít nhất một số bit trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC cũng có thể được chọn để dùng làm các bit chẵn lẻ bổ sung.

Vì vậy, có thể chọn thêm các bit ($v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1}$).

Do đó, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit ($v_{N_{repeat}+N_{inner}-N_{punc}}, v_{N_{repeat}+N_{inner}-N_{punc}+1}, \dots, v_{N_{repeat}+N_{inner}-1}$) và các bit ($v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1}$).

Bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể tạo ra các bit chẵn lẻ bổ sung bằng cách sử dụng nhiều phương pháp khác ngoài các phương pháp nêu trên.

Ví dụ, khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng hoặc nhỏ hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{AP} \leq N_{punc}$, như được thể hiện trên Fig.18, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung. Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{repeat}+N_{inner}-N_{punc}}, v_{N_{repeat}+N_{inner}-N_{punc}+1}, \dots, v_{N_{repeat}+N_{inner}-N_{punc}+N_{AP}+1})$.

Ngoài ra, khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, nhưng bằng hoặc nhỏ hơn tổng số của số lượng bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{punc} < N_{AP} \leq N_{ldpc_parity} + N_{punc}$, như được thể hiện trên Fig.19, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi. Vì vậy, trước tiên có thể chọn các bit $(v_{N_{repeat}+N_{inner}-N_{punc}}, v_{N_{repeat}+N_{inner}-N_{punc}+1}, \dots, v_{N_{repeat}+N_{inner}-1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn thêm $N_{AP} - N_{punc}$ bit kể từ bit đầu tiên trong số N_{ldpc_parity} bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Vì vậy, có thể chọn thêm các bit $(v_{N_{repeat}+K_{ldpc}}, v_{N_{repeat}+K_{ldpc}+1}, \dots, v_{N_{repeat}+K_{ldpc}-N_{AP}+N_{punc}-1})$.

Do đó, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{repeat}+N_{inner}-N_{punc}}, v_{N_{repeat}+N_{inner}-N_{punc}+1}, \dots, v_{N_{repeat}+N_{inner}-1})$ và các bit $(v_{N_{repeat}+K_{ldpc}}, v_{N_{repeat}+K_{ldpc}+1}, \dots, v_{N_{repeat}+K_{ldpc}-N_{AP}+N_{punc}-1})$.

Ngoài ra, khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn tổng của số lượng bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi và tất cả các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, và chọn các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được lặp lại, để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{\text{ldpc_parity}} + N_{\text{punc}} < N_{\text{AP}}$, như được thể hiện trên Fig.20, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi. Vì vậy, trước tiên có thể chọn các bit $(v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}}, v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}+1}, \dots, v_{N_{\text{repeat}}+N_{\text{inner}}-1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 chọn thêm tất cả $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Vì vậy, có thể chọn thêm các bit $(v_{N_{\text{repeat}}+K_{\text{ldpc}}}, v_{N_{\text{repeat}}+K_{\text{ldpc}}+1}, \dots, v_{N_{\text{repeat}}+N_{\text{inner}}-1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn thêm $N_{\text{AP}} - N_{\text{ldpc_parity}} - N_{\text{punc}}$ bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được lặp lại. Vì vậy, có thể chọn thêm các bit $(v_{K_{\text{ldpc}}}, v_{K_{\text{ldpc}}+1}, \dots, v_{K_{\text{ldpc}}+N_{\text{AP}}-N_{\text{punc}}-N_{\text{inner}}-1})$.

Do đó, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}}, v_{N_{\text{repeat}}+N_{\text{inner}}-N_{\text{punc}}+1}, \dots, v_{N_{\text{repeat}}+N_{\text{inner}}-1})$, các bit $(v_{N_{\text{repeat}}+K_{\text{ldpc}}}, v_{N_{\text{repeat}}+K_{\text{ldpc}}+1}, \dots, v_{N_{\text{repeat}}+N_{\text{inner}}-1})$, và các bit $(v_{K_{\text{ldpc}}}, v_{K_{\text{ldpc}}+1}, \dots, v_{K_{\text{ldpc}}+N_{\text{AP}}-N_{\text{punc}}-N_{\text{inner}}-1})$.

Trong ví dụ nêu trên, giả sử xét trường hợp trong đó các bit lặp được gán bằng phương pháp lặp vào phía sau các bit thông tin LDPC.

Ví dụ khác, khi các bit lặp được gán bằng phương pháp lặp vào phía sau các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, phương pháp tạo ra bit chẵn lẻ bổ sung sẽ được mô tả dưới đây. Trong trường hợp như vậy, các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và các bit chẵn lẻ LDPC được lặp lại có thể được sắp xếp tuần tự để tạo nên các bit chẵn lẻ LDPC.

Từ mã LDPC có bit lặp có thể được biểu diễn dưới dạng $V = (v_0, v_1, \dots, v_{N_{\text{inner}} + N_{\text{repeat}} - 1})$.

Trước hết, khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng hoặc nhỏ hơn tổng số của số lượng bit chẵn lẻ LDPC được đọc lỗi và số lượng bit chẵn lẻ LDPC được lặp lại, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, vì các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và các bit chẵn lẻ LDPC được lặp lại được sắp xếp tuần tự, cho nên khi chọn các bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi, thì cũng có thể chọn ít nhất một số bit chẵn lẻ LDPC được lặp lại, để dùng làm các bit chẵn lẻ bổ sung, tùy thuộc vào số lượng bit chẵn lẻ bổ sung được tạo ra.

Điều này có nghĩa là, khi $N_{\text{AP}} \leq N_{\text{punc}} + N_{\text{repeat}}$, như được thể hiện trên Fig.21, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi. Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{\text{inner}} - N_{\text{punc}}}, v_{N_{\text{inner}} - N_{\text{punc}} + 1}, \dots, v_{N_{\text{inner}} - N_{\text{punc}} + N_{\text{AP}} - 1})$.

Ngoài ra, khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn tổng của số lượng bit chẵn lẻ LDPC được đọc lỗi và số lượng bit chẵn lẻ LDPC được lặp lại, bộ tạo bit

chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi và tất cả các bit chẵn lẻ LDPC được lặp lại, và chọn các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi và số lượng bit chẵn lẻ LDPC được lặp lại, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{AP} > N_{punc} + N_{repeat}$, như được thể hiện trên Fig.22, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi và tất cả các bit chẵn lẻ LDPC được lặp lại. Vì vậy, trước tiên có thể chọn các bit $(v_{N_{inner}-N_{punc}}, v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-N_{repeat}-1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn thêm $N_{AP} - N_{punc}$ bit kể từ bit đầu tiên trong số N_{ldpc_parity} bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Vì vậy, có thể chọn thêm các bit $(v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1})$.

Do đó, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{inner}-N_{punc}}, v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-N_{repeat}-1})$ và các bit $(v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1})$.

Ví dụ khác, khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng hoặc nhỏ hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{AP} \leq N_{punc}$, như được thể hiện trên Fig.23, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung. Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{inner}-N_{punc}}, v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-N_{punc}+N_{AP}-1})$.

Ngoài ra, khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn các bit tương ứng với số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{AP} > N_{punc}$, như được thể hiện trên Fig.24, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi. Vì vậy, trước tiên có thể chọn các bit $(v_{N_{inner}-N_{punc}}, v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-N_{punc}+N_{AP}-1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn thêm $N_{AP} - N_{punc}$ bit kể từ bit đầu tiên trong số N_{ldpc_parity} bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Vì vậy, có thể chọn thêm các bit $(v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1})$.

Do đó, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{inner}-N_{punc}}, v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-N_{punc}+N_{AP}-1})$ và các bit $(v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1})$.

Theo phương án làm ví dụ khác để thực hiện sáng chế, một phương pháp khác để tạo ra các bit chẵn lẻ bổ sung sẽ được mô tả dưới đây.

Ví dụ, giả sử xét trường hợp trong đó $N_{punc} > 0$ và $N_{repeat} = 0$.

Cụ thể, khi số lượng bit chẵn lẻ LDPC được đọc lỗi bằng hoặc lớn hơn số lượng bit chẵn lẻ bổ sung được tạo ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung, và mặt khác, có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi và tuần tự chọn các bit chẵn lẻ bổ sung còn lại trong số từ mã LDPC để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ khi $N_{punc} > 0$ và $N_{repeat} = 0$, từ mã LDPC có thể được biểu diễn như được

thể hiện trên Fig.25. Trong đó, số lượng bit thông tin LDPC bằng N_{outer} , các bit không được đệm vào, và do đó, các bit thông tin LDPC chỉ có các bit mã hoá ngoài.

Trước hết, khi $N_{\text{AP}} \leq N_{\text{punc}}$, như được thể hiện trên Fig.26, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Ngoài ra, khi $N_{\text{AP}} > N_{\text{punc}}$, như được thể hiện trên Fig.27, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn thêm $N_{\text{AP}} - N_{\text{punc}}$ bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ nêu trên mô tả rằng $N_{\text{AP}} - N_{\text{punc}}$ bit được chọn thêm kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC, nhưng đó chỉ là một ví dụ. Điều này có nghĩa là, bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể chọn thêm $N_{\text{AP}} - N_{\text{punc}}$ bit kể từ bit đầu tiên trong số các bit thông tin LDPC.

Ví dụ khác, khi $N_{\text{AP}} > N_{\text{punc}}$, như được thể hiện trên Fig.28, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn thêm $N_{\text{AP}} - N_{\text{punc}}$ bit trong số các bit chẵn lẻ LDPC dựa vào mẫu định trước để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ khác, khi $N_{\text{punc}} = 0$, như được thể hiện trên Fig.29, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit chẵn lẻ LDPC đầu tiên để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ nêu trên mô tả rằng các bit được chọn trong từ mã LDPC có bit lặp để tạo ra các bit chẵn lẻ bổ sung.

Tuy nhiên, theo phương án làm ví dụ thực hiện sáng chế, các bit chẵn lẻ bổ sung

cũng có thể được chọn trong từ mã LDPC để tạo ra các bit chẵn lẻ bổ sung trước khi phương pháp lặp được thực hiện. Điều này có nghĩa là, phương pháp lặp có thể được thực hiện sau khi đọc lỗi và tạo ra các bit chẵn lẻ bổ sung. Nhằm mục đích này, bộ mã hoá LDPC 110 có thể xuất ra từ mã LDPC để cung cấp cho bộ phận đọc lỗi 130 đứng sau bộ tạo bit chẵn lẻ bổ sung 140.

Ví dụ, giả sử xét trường hợp trong đó $N_{\text{repeat}} > 0$. Dưới đây, giả sử rằng số lượng bit thông tin LDPC bằng K_{ldpc} .

Bộ tạo bit chẵn lẻ bổ sung 140 có thể ưu tiên chọn các bit chẵn lẻ LDPC được đọc lỗi, tuy nhiên không nhất thiết phải làm đúng như vậy, để tạo ra các bit chẵn lẻ bổ sung và chọn một số bit chẵn lẻ LDPC đúng bằng số lượng bit chẵn lẻ LDPC được đọc lỗi nhỏ hơn số lượng bit chẵn lẻ bổ sung được tạo ra, để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, các bit chẵn lẻ LDPC có thể chỉ có các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Cụ thể, bộ tạo bit chẵn lẻ bổ sung 140 có thể bắt đầu chọn bit chẵn lẻ LDPC đầu tiên để tạo ra các bit chẵn lẻ bổ sung.

Theo phương án làm ví dụ khác để thực hiện sáng chế, giả sử xét trường hợp trong đó $N_{\text{repeat}} < N_{\text{ldpc_parity}} - N_{\text{punc}}$.

Khi $N_{\text{AP}} \leq N_{\text{punc}}$, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ, như được thể hiện trên Fig.30, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung. Trong trường hợp như vậy, như được thể hiện trên Fig.30, bộ phận

lặp 120 có thể chọn N_{repeat} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC và gán N_{repeat} bit đã chọn vào phía sau các bit chẵn lẻ LDPC. Do đó, sau khi đọc lỗi, các bit lặp được đặt vào phía sau các bit chẵn lẻ LDPC không được đọc lỗi.

Khi $N_{\text{AP}} > N_{\text{punc}}$, như được thể hiện trên Fig.31, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn $N_{\text{AP}} - N_{\text{punc}}$ bit kể từ bit chẵn lẻ LDPC đầu tiên, để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, như được thể hiện trên Fig.30, bộ phận lặp 120 có thể chọn N_{repeat} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC và gán N_{repeat} bit đã chọn vào phía sau các bit chẵn lẻ LDPC, và do đó, sau khi đọc lỗi, các bit lặp được đặt vào phía sau các bit chẵn lẻ LDPC không được đọc lỗi. Vì vậy, ít nhất một số bit được chọn để dùng làm các bit chẵn lẻ bổ sung trong số các bit chẵn lẻ LDPC có thể được chọn làm các bit lặp.

Ví dụ khác, giả sử xét trường hợp trong đó $N_{\text{ldpc_parity}} \geq N_{\text{repeat}} \geq N_{\text{ldpc_parity}} - N_{\text{punc}}$.

Khi $N_{\text{AP}} \leq N_{\text{punc}}$, như được thể hiện trên Fig.32, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung. Trong trường hợp như vậy, như được thể hiện trên Fig.32, bộ phận lặp 120 có thể chọn N_{repeat} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC và gán N_{repeat} bit đã chọn vào phía sau các bit chẵn lẻ LDPC, và do đó, sau khi đọc lỗi, các bit lặp được đặt vào phía sau các bit chẵn lẻ LDPC không được đọc lỗi.

Khi $N_{\text{AP}} > N_{\text{punc}}$, như được thể hiện trên Fig.33, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn $N_{\text{AP}} - N_{\text{punc}}$ bit kể từ bit chẵn lẻ LDPC đầu tiên để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, như được thể hiện trên Fig.33, bộ phận lặp 120 có

thể chọn N_{repeat} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC và gán N_{repeat} bit đã chọn vào phía sau các bit chẵn lẻ LDPC, và do đó, sau khi đọc lỗi, các bit lặp được đặt vào phía sau các bit chẵn lẻ LDPC không được đọc lỗi. Vì vậy, ít nhất một số bit được chọn để dùng làm các bit chẵn lẻ bổ sung trong số các bit chẵn lẻ LDPC có thể được chọn làm các bit lặp.

Ví dụ khác, giả sử xét trường hợp trong đó $N_{\text{repeat}} > N_{\text{ldpc_parity}}$.

Khi $N_{\text{AP}} \leq N_{\text{punc}}$, như được thể hiện trên Fig.34, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Khi $N_{\text{AP}} > N_{\text{punc}}$ như được thể hiện trên Fig.33, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn $N_{\text{AP}} - N_{\text{punc}}$ bit kể từ bit chẵn lẻ LDPC đầu tiên, để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, như được thể hiện trên Fig.33 và Fig.34, bộ phận lặp 120 có thể chọn N_{repeat} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC và gán N_{repeat} bit đã chọn vào phía sau các bit chẵn lẻ LDPC, và do đó, sau khi đọc lỗi, các bit lặp được đặt vào phía sau các bit chẵn lẻ LDPC không được đọc lỗi.

Ngoài ra, khi N_{repeat} lớn hơn $N_{\text{ldpc_parity}}$, như được thể hiện trên Fig.34 và Fig.35, tất cả các bit chẵn lẻ LDPC có thể được lặp lại và còn có thêm ít nhất một số bit trong số các bit chẵn lẻ LDPC có thể được lặp lại.

Bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể chọn số lượng bit định trước ở trong phần còn lại của các bit chẵn lẻ LDPC, ngoại trừ các bit chẵn lẻ LDPC được lặp lại trong số các bit chẵn lẻ LDPC.

Ví dụ, khi $N_{\text{AP}} \leq N_{\text{punc}}$, như được thể hiện trên Fig.36, bộ tạo bit chẵn lẻ bổ sung

140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc để tạo ra các bit chẵn lẻ bổ sung.

Ngoài ra, khi $N_{AP} > N_{punc}$, như được thể hiện trên Fig.37, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc, và chọn $N_{AP} - N_{punc}$ bit kể từ bit đầu tiên ở trong phần còn lại của các bit chẵn lẻ LDPC, ngoại trừ các bit chẵn lẻ LDPC được lặp lại trong số các bit chẵn lẻ LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, như được thể hiện trên Fig.38, khi chọn toàn bộ phần còn lại của các bit chẵn lẻ LDPC, ngoại trừ các bit chẵn lẻ LDPC được lặp lại trong số các bit chẵn lẻ LDPC, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit bổ sung còn lại trong số các bit chẵn lẻ LDPC được lặp lại, để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ nêu trên mô tả rằng các bit được chọn trong số các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung, nhưng đó chỉ là một ví dụ. Vì vậy, khi số lượng bit chẵn lẻ LDPC nhỏ hơn số lượng bit chẵn lẻ bổ sung được tạo ra, bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể chọn các bit trong số các bit mã hoá ngoài và các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, khi các bit được chọn trong số các bit mã hoá ngoài và các bit chẵn lẻ LDPC, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit khác, ngoại trừ các bit đã chọn. Trong trường hợp như vậy, khi cần phải chọn thêm, thì các bit có tần suất chọn thấp nhất có thể được chọn trước tiên để tạo ra các bit chẵn lẻ bổ sung.

Ngoài ra, ví dụ nêu trên mô tả rằng phương pháp lặp được thực hiện cùng với

việc tạo ra các bit chẵn lẻ bổ sung có tính đến các bit lặp, nhưng đó chỉ là một ví dụ. Điều này có nghĩa là, trong một số trường hợp, phương pháp lặp cũng có thể không được thực hiện.

Trong trường hợp như vậy, khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng hoặc nhỏ hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng bit chẵn lẻ bổ sung được tạo ra kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Ngoài ra, khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ LDPC được đọc lỗi, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và chọn các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC, tức là các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Ngoài ra, ví dụ nêu trên mô tả rằng các bit chẵn lẻ LDPC được chọn theo đơn vị bit để tạo ra các bit chẵn lẻ bổ sung, nhưng đó chỉ là một ví dụ. Vì vậy, bộ tạo bit chẵn lẻ bổ sung 140 cũng có thể chọn các bit chẵn lẻ LDPC theo đơn vị nhóm bit để tạo ra các bit chẵn lẻ bổ sung.

Ví dụ, giả sử xét trường hợp trong đó từ mã LDPC ($v_0, v_1, \dots, v_{N_{\text{inner}}-1}$) được phân chia ra thành N_{group} nhóm bit, và do đó, được biểu diễn dưới dạng $V = (Y_0, Y_1, \dots, Y_{N_{\text{group}}-1})$.

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 140 có thể tính số lượng tạm thời $N_{\text{AP_temp}}$ bit chẵn lẻ bổ sung dựa vào biểu thức 13 dưới đây:

$$N_{AP_temp} = \alpha \times K \times (N_{ldpc_parity} - N_{punc}) \quad (13)$$

Trong biểu thức 13 nêu trên, N_{ldpc_parity} là số lượng bit chẵn lẻ LDPC và N_{punc} là số lượng bit chẵn lẻ LDPC được đục lỗ. Ngoài ra, $\alpha = 0,5$ và $K = 0, 1, 2$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể tính số lượng N_{AP} bit chẵn lẻ bổ sung dựa vào biểu thức 14 hoặc biểu thức 15 dưới đây:

$$N_{AP} = \left\lceil \frac{N_{AP_temp}}{\eta_{MOD}} \right\rceil \times \eta_{MOD} \quad (14)$$

$$N_{AP} = \left\lfloor \frac{N_{AP_temp}}{\eta_{MOD}} \right\rfloor \times \eta_{MOD} \quad (15)$$

Trong biểu thức 14 hoặc biểu thức 15 nêu trên, $\lceil x \rceil$ là số nguyên nhỏ nhất bằng hoặc lớn hơn x , và $\lfloor x \rfloor$ là số nguyên lớn nhất không lớn hơn x . Theo sáng chế, η_{MOD} là bậc điều biến. Ví dụ, trong các sơ đồ điều biến QPSK, 16-QAM, 64-QAM và 256-QAM, η_{MOD} có thể lần lượt bằng 2, 4, 6 và 8.

Tiếp theo, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng đã tính trong số các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi số lượng bit chẵn lẻ bổ sung được tạo ra bằng hoặc nhỏ hơn số lượng bit chẵn lẻ LDPC được đục lỗ, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng đã tính kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đục lỗ để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi $N_{AP} \leq N_{punc}$, như được thể hiện trên Fig.39, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đục lỗ để tạo ra các bit chẵn lẻ bổ sung. Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit $(v_{N_{inner}-N_{punc}}, v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-N_{punc}+N_{AP}-1})$.

Khi số lượng bit chẵn lẻ bổ sung được tạo ra lớn hơn số lượng bit chẵn lẻ LDPC

được đọc lỗi, tức là $N_{AP} > N_{punc}$, bộ tạo bit chẵn lẻ bổ sung 140 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi. Vì vậy, có thể chọn các bit $v_{N_{inner}-N_{punc}}$, $v_{N_{inner}-N_{punc}+1}, \dots, v_{N_{inner}-1}$ để dùng làm các bit chẵn lẻ bổ sung.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung được tạo ra trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi theo đơn vị nhóm bit.

Nhằm mục đích này, bộ tạo bit chẵn lẻ bổ sung 140 có thể tính số lượng nhóm bit mà tất cả các bit trong nhóm bit đó đều được chọn để dùng làm các bit chẵn lẻ bổ sung, dựa vào biểu thức 16 dưới đây:

$$N_{AP_groups} = \left\lfloor \frac{N_{AP} - N_{punc}}{360} \right\rfloor \quad (16)$$

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 140 có thể xác định ít nhất một nhóm bit được dùng để tạo ra các bit chẵn lẻ bổ sung trong số nhiều nhóm bit dựa vào mẫu bit chẵn lẻ bổ sung, và chọn tất cả các bit ở trong một số nhóm bit đã xác định và một số bit ở trong các nhóm bit còn lại để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, mẫu bit chẵn lẻ bổ sung là mẫu xác định thứ tự của các nhóm bit được chọn làm các bit chẵn lẻ bổ sung trong số các nhóm bit tạo nên các bit chẵn lẻ LDPC, và, ví dụ, mẫu này có thể được xác định dưới dạng $\pi_{AP}(0) = x_0$, $\pi_{AP}(1) = x_1, \dots, \pi_{AP}(N_{group}-1) = x_{N_{group}-1}$, với $\pi_{AP}(j)$ ($0 \leq j < N_{group}$).

Trong đó, $x_0, x_1, \dots, x_{N_{group}-1}$ là chỉ số của các nhóm bit tạo nên các bit chẵn lẻ LDPC, N_{group} là số lượng nhóm bit tạo nên các bit chẵn lẻ LDPC, và một nhóm bit có thể có 360 bit.

Cụ thể, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit của nhóm bit thứ $\pi_{AP}(0)$, nhóm bit thứ $\pi_{AP}(1)$, ..., nhóm bit thứ $\pi_{AP}(N_{AP_groups}-1)$ trong số các nhóm bit

để dùng làm các bit chẵn lẻ bổ sung, dựa vào mẫu bit chẵn lẻ bổ sung. Điều này có nghĩa là, có thể chọn tất cả các bit của nhóm bit thứ x_0 , nhóm bit thứ x_1 , ..., nhóm bit thứ $x_{N_{\text{group}}-1}$ để dùng làm các bit chẵn lẻ bổ sung.

Bộ tạo bit chẵn lẻ bổ sung 140 cần chọn $N_{\text{AP}} - N_{\text{punc}}$ bit, nhưng số lượng bit được chọn từ $N_{\text{AP_groups}}$ nhóm bit là $360 \times N_{\text{AP_groups}}$, và vì vậy, bộ tạo bit chẵn lẻ bổ sung 140 có thể cần chọn thêm $N_{\text{AP}} - N_{\text{punc}} - 360 \times N_{\text{AP_groups}}$ bit.

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 140 có thể xác định nhóm bit có các bit được chọn thêm dựa vào mẫu bit chẵn lẻ bổ sung, và chọn thêm $N_{\text{AP}} - N_{\text{punc}} - 360 \times N_{\text{AP_groups}}$ bit kể từ bit đầu tiên trong nhóm bit đã xác định.

Cụ thể, bộ tạo bit chẵn lẻ bổ sung 140 có thể xác định nhóm bit $\pi_{\text{AP}}(N_{\text{AP_groups}})$ dựa vào mẫu bit chẵn lẻ bổ sung, và chọn từ bit đầu tiên cho đến khi có đủ $N_{\text{AP}} - N_{\text{punc}} - 360 \times N_{\text{AP_groups}}$ bit trong nhóm bit thứ $\pi_{\text{AP}}(N_{\text{AP_groups}})$ để dùng làm các bit chẵn lẻ bổ sung. Điều này có nghĩa là, có thể chọn thêm từng bit trong nhóm bit thứ $x_{N_{\text{group}}}$ cho đến khi có đủ $N_{\text{AP}} - N_{\text{punc}} - 360 \times N_{\text{AP_groups}}$ bit, để dùng làm các bit chẵn lẻ bổ sung.

Do đó, khi $N_{\text{AP}} > N_{\text{punc}}$, như được thể hiện trên Fig.40, bộ tạo bit chẵn lẻ bổ sung 140 có thể chọn tất cả các bit chẵn lẻ LDPC được đục lỗ, và chọn tất cả các bit trong nhóm bit thứ $\pi_{\text{AP}}(0)$, nhóm bit thứ $\pi_{\text{AP}}(1)$, ..., nhóm bit thứ $\pi_{\text{AP}}(N_{\text{AP_groups}}-1)$ và từ bit đầu tiên cho đến khi có đủ $N_{\text{AP}} - N_{\text{punc}} - 360 \times N_{\text{AP_groups}}$ bit trong nhóm bit thứ $\pi_{\text{AP}}(N_{\text{AP_groups}})$ để tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp các bit lặp được chọn dựa vào mẫu định trước, mẫu lặp định trước có thể được sử dụng và coi như có mức ưu tiên cao hơn so với các bit chẵn lẻ bổ sung được chọn. Điều này có nghĩa là, sau khi chọn xong các bit lặp dựa vào mẫu

lắp, thì sẽ chọn các bit để dùng làm các bit chắn lẻ bổ sung. Khi số lượng bit, ngoại trừ số lượng bit lắp nằm trong số lượng tối đa của các bit lắp được xác định là mẫu lắp, lớn hơn số lượng bit chắn lẻ bổ sung được tạo ra, thì các bit chắn lẻ bổ sung có thể được tạo ra dựa vào cả mẫu lắp lẫn mẫu bit chắn lẻ bổ sung. Mẫu lắp có nghĩa là thứ tự sắp xếp của các nhóm bit có hiệu quả cao khi truyền/thu tín hiệu trong số các bit ở trong một từ mã LDPC nhất định khi truyền từ mã LDPC, và do đó, các bit chắn lẻ bổ sung được tạo ra dựa vào mẫu lắp, và khi cần có thêm các bit chắn lẻ bổ sung, thì các bit chắn lẻ bổ sung đó sẽ được xác định dựa vào mẫu bit chắn lẻ bổ sung.

Thiết bị truyền tín hiệu 100 có thể truyền các bit được xuất ra từ bộ phận đọc lỗi 130 và các bit được xuất ra từ bộ tạo bit chắn lẻ bổ sung 140 đến thiết bị thu tín hiệu 200.

Trong trường hợp như vậy, thiết bị truyền tín hiệu 100 có thể truyền các bit từ mã LDPC, tức là từ mã LDPC đã được lắp lại, đọc lỗi và rút gọn, ngoại trừ các bit không được đệm vào trong từ mã LDPC đã được lắp lại và đọc lỗi, được xuất ra từ bộ phận đọc lỗi 130, đến thiết bị thu tín hiệu 200.

Cụ thể, thiết bị truyền tín hiệu 100 có thể lần lượt điều biến các bit từ mã LDPC đã được lắp lại, đọc lỗi và rút gọn và các bit chắn lẻ bổ sung, ánh xạ các bit lên các ký hiệu chòm điểm, và ánh xạ các ký hiệu lên khung để truyền đến thiết bị thu tín hiệu 200.

Tuy nhiên, khi phương pháp lắp không được thực hiện, thiết bị truyền tín hiệu 100 có thể truyền các bit từ mã LDPC, tức là từ mã LDPC đã được rút gọn và đọc lỗi, ngoại trừ các bit không được đệm vào trong từ mã LDPC đã được đọc lỗi, đến thiết bị thu tín hiệu 200.

Trong trường hợp như vậy, thiết bị truyền tín hiệu 100 có thể thực hiện bước điều biến theo sơ đồ điều biến QPSK, 16-QAM, 64-QAM, 256-QAM, hoặc sơ đồ điều biến khác, và điều biến các bit từ mã LDPC đã được lặp lại, đục lỗ và rút gọn (hay từ mã LDPC đã được đục lỗ và rút gọn) và các bit chẵn lẻ bổ sung theo cùng một sơ đồ điều biến.

Thiết bị truyền tín hiệu 100 có thể ánh xạ các bit chẵn lẻ bổ sung được tạo ra dựa vào các bit thông tin được truyền trong khung hiện thời lên khung đứng trước khung hiện thời.

Điều này có nghĩa là, thiết bị truyền tín hiệu 100 có thể ánh xạ từ mã LDPC đã được rút gọn và đục lỗ có các bit thông tin tương ứng với khung thứ $(i-1)$ lên khung thứ $(i-1)$, và ánh xạ các bit chẵn lẻ bổ sung được tạo ra dựa vào các bit thông tin tương ứng với khung thứ i lên khung thứ $(i-1)$, và truyền các bit đã được ánh xạ đến thiết bị thu tín hiệu 200.

Vì vậy, các bit thông tin tương ứng với khung thứ $(i-1)$ và các bit chẵn lẻ được tạo ra dựa vào các bit thông tin này cũng như các bit chẵn lẻ bổ sung được tạo ra dựa vào các bit thông tin tương ứng với khung thứ i có thể được ánh xạ lên khung thứ $(i-1)$.

Như đã nêu trên, vì các bit thông tin là các bit thông tin báo hiệu chứa thông tin cần thiết cho thiết bị thu tín hiệu 200 để xử lý dữ liệu dịch vụ, cho nên thiết bị truyền tín hiệu 100 có thể ánh xạ dữ liệu lên khung cùng với thông tin báo hiệu để xử lý và truyền dữ liệu và thông tin báo hiệu đã được ánh xạ đến thiết bị thu tín hiệu 200.

Cụ thể, thiết bị truyền tín hiệu 100 có thể xử lý dữ liệu theo sơ đồ định trước để tạo ra các ký hiệu chòm điểm và ánh xạ các ký hiệu chòm điểm đã tạo ra lên các ký

hiệu dữ liệu của mỗi khung. Ngoài ra, thiết bị truyền tín hiệu 100 có thể ánh xạ thông tin báo hiệu, liên quan đến dữ liệu được ánh xạ lên mỗi khung, vào phần mở đầu của khung tương ứng. Ví dụ, thiết bị truyền tín hiệu 100 có thể ánh xạ thông tin báo hiệu liên quan đến dữ liệu được ánh xạ lên khung thứ i vào trong khung thứ i .

Do đó, thiết bị thu tín hiệu 200 có thể sử dụng thông tin báo hiệu thu được từ khung để thu nhận và xử lý dữ liệu từ khung tương ứng.

Theo phương án làm ví dụ thực hiện sáng chế, các bit thông tin nêu trên có thể tạo nên thông tin báo hiệu chi tiết cho tầng L1. Thiết bị truyền tín hiệu 100 có thể tạo ra các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 bằng cách sử dụng phương pháp nêu trên và truyền các bit chẵn lẻ bổ sung đã tạo ra đến thiết bị thu tín hiệu 200.

Theo sáng chế, thông tin báo hiệu chi tiết cho tầng L1 có thể là các bit thông tin báo hiệu được định nghĩa theo tiêu chuẩn Advanced Television System Committee (ATSC) 3.0.

Có bảy (7) chế độ xử lý thông tin báo hiệu chi tiết cho tầng L1. Thiết bị truyền tín hiệu 100 theo phương án làm ví dụ thực hiện sáng chế có thể tạo ra các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 dựa vào bảy chế độ này.

Tiêu chuẩn ATSC 3.0 quy định thông tin báo hiệu cơ bản cho tầng L1, bên cạnh thông tin báo hiệu chi tiết cho tầng L1. Thiết bị truyền tín hiệu 100 có thể xử lý thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 bằng cách sử dụng một sơ đồ cụ thể và truyền thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 đã xử lý đến thiết bị thu tín hiệu 200.

Phương pháp xử lý thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu

chi tiết cho tầng L1 sẽ được mô tả dưới đây.

Thiết bị truyền tín hiệu 100 có thể ánh xạ thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 vào trong phần mở đầu của khung và ánh xạ dữ liệu lên các ký hiệu dữ liệu của khung để truyền đến thiết bị thu tín hiệu 200.

Dựa vào Fig.41, khung có thể được tạo nên gồm có ba phần, đó là phần khởi động, phần mở đầu và phần dữ liệu.

Phần khởi động được sử dụng để đồng bộ hoá ban đầu và cung cấp thông số cơ bản cần thiết cho thiết bị thu tín hiệu 200 để giải mã thông tin báo hiệu cho tầng L1. Ngoài ra, phần khởi động có thể chứa thông tin về chế độ xử lý thông tin báo hiệu cơ bản cho tầng L1 ở thiết bị truyền tín hiệu 100, tức là thông tin về chế độ mà thiết bị truyền tín hiệu 100 sử dụng để xử lý thông tin báo hiệu cơ bản cho tầng L1.

Phần mở đầu chứa thông tin báo hiệu cho tầng L1, và có thể được tạo nên gồm có hai phần, đó là phần thông tin báo hiệu cơ bản cho tầng L1 và phần thông tin báo hiệu chi tiết cho tầng L1.

Theo sáng chế, thông tin báo hiệu cơ bản cho tầng L1 có thể chứa thông tin về thông tin báo hiệu chi tiết cho tầng L1, và thông tin báo hiệu chi tiết cho tầng L1 có thể chứa thông tin về dữ liệu. Theo sáng chế, dữ liệu là dữ liệu phát rộng để cung cấp các dịch vụ phát rộng và có thể được truyền qua ít nhất một kênh truyền tầng vật lý (Physical Layer Pipe, PLP).

Cụ thể, thông tin báo hiệu cơ bản cho tầng L1 là thông tin cần thiết cho thiết bị thu tín hiệu 200 để xử lý thông tin báo hiệu chi tiết cho tầng L1. Thông tin này chứa, ví dụ, thông tin về chế độ xử lý thông tin báo hiệu chi tiết cho tầng L1 ở thiết bị truyền tín hiệu 100, tức là thông tin về chế độ mà thiết bị truyền tín hiệu 100 sử dụng

để xử lý thông tin báo hiệu chi tiết cho tầng L1, thông tin về độ dài của thông tin báo hiệu chi tiết cho tầng L1, thông tin về chế độ chặn lẻ bổ sung, tức là thông tin về giá trị K dùng cho thiết bị truyền tín hiệu 100 để tạo ra các bit chặn lẻ bổ sung bằng cách sử dụng trường L1B_L1_Detail_additional_parity_mode (trong đó, khi trường L1B_L1_Detail_additional_parity_mode được đặt bằng '00', thì $K = 0$ và các bit chặn lẻ bổ sung không được sử dụng), và thông tin về độ dài của tổng số các ô. Ngoài ra, thông tin báo hiệu cơ bản cho tầng L1 có thể là thông tin báo hiệu cơ bản liên quan đến hệ thống có thiết bị truyền tín hiệu 100 như kích thước biến đổi Fourier nhanh (Fast Fourier Transform, FFT), khoảng bảo vệ và mẫu sóng chủ.

Ngoài ra, thông tin báo hiệu chi tiết cho tầng L1 là thông tin cần thiết cho thiết bị thu tín hiệu 200 để giải mã các kênh truyền PLP, ví dụ, vị trí đầu tiên của các ô được ánh xạ lên các ký hiệu dữ liệu trên mỗi kênh truyền PLP, thông tin nhận dạng (Identifier, ID) của kênh truyền PLP, kích thước của kênh truyền PLP, sơ đồ điều biến, tỷ lệ mã, v.v..

Vì vậy, thiết bị thu tín hiệu 200 có thể thu nhận tín hiệu đồng bộ hoá khung, thu nhận thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 từ phần mở đầu, và thu thông tin dịch vụ mà người dùng yêu cầu từ các ký hiệu dữ liệu bằng cách sử dụng thông tin báo hiệu chi tiết cho tầng L1.

Phương pháp xử lý thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 sẽ được mô tả chi tiết dưới đây dựa vào các hình vẽ kèm theo.

Fig.42 và Fig.43 là các sơ đồ khối thể hiện cấu hình chi tiết của thiết bị truyền tín hiệu 100 theo phương án làm ví dụ thực hiện sáng chế.

Cụ thể, như được thể hiện trên Fig.42, để xử lý thông tin báo hiệu cơ bản cho

tầng L1, thiết bị truyền tín hiệu 100 có thể bao gồm bộ phận xáo trộn 211, bộ mã hoá BCH 212, bộ đệm bit không 213, bộ mã hoá LDPC 214, bộ phận hoán vị chẵn lẻ 215, bộ phận lặp 216, bộ phận đục lỗ 217, bộ phận loại bỏ bit không 218, bộ phân kênh bit 219 và bộ ánh xạ chòm điểm 221.

Ngoài ra, như được thể hiện trên Fig.43, để xử lý thông tin báo hiệu chi tiết cho tầng L1, thiết bị truyền tín hiệu 100 có thể bao gồm bộ phận phân đoạn 311, bộ phận xáo trộn 312, bộ mã hoá BCH 313, bộ đệm bit không 314, bộ mã hoá LDPC 315, bộ phận hoán vị chẵn lẻ 316, bộ phận lặp 317, bộ phận đục lỗ 318, bộ tạo bit chẵn lẻ bổ sung 319, bộ phận loại bỏ bit không 321, các bộ phân kênh bit 322 và 323, và các bộ ánh xạ chòm điểm 324 và 325.

Theo sáng chế, các bộ phận được thể hiện trên Fig.42 và Fig.43 là các bộ phận để thực hiện quy trình mã hoá và điều biến trên thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1, nhưng đó chỉ là một ví dụ. Theo các phương án làm ví dụ khác để thực hiện sáng chế, một số bộ phận được thể hiện trên Fig.42 và Fig.43 có thể được loại bỏ hoặc thay đổi, và các bộ phận khác cũng có thể được bổ sung vào. Ngoài ra, vị trí của một số bộ phận có thể thay đổi. Ví dụ, vị trí của các bộ phận lặp 216 và 317 có thể lần lượt được đặt ở sau các bộ phận đục lỗ 217 và 318.

Bộ mã hoá LDPC 315, bộ phận lặp 317, bộ phận đục lỗ 318 và bộ tạo bit chẵn lẻ bổ sung 319 được thể hiện trên Fig.43 có thể lần lượt thực hiện các thao tác được thực hiện bằng bộ mã hoá LDPC 110, bộ phận lặp 120, bộ phận đục lỗ 130 và bộ tạo bit chẵn lẻ bổ sung 140 được thể hiện trên Fig.1.

Khi mô tả dựa vào Fig.42 và Fig.43, để cho thuận tiện, các bộ phận thực hiện

chức năng giống nhau sẽ được mô tả cùng nhau.

Thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được bảo vệ bằng dây ghép nối của mã ngoài BCH và mã trong LDPC. Tuy nhiên, đó chỉ là một ví dụ. Vì vậy, nếu thao tác mã hoá ngoài được thực hiện trước khi mã hoá trong theo sơ đồ mã hoá ghép nối, thì một sơ đồ mã hoá khác như sơ đồ mã hoá CRC, ngoài sơ đồ mã hoá BCH, có thể được áp dụng. Ngoài ra, thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được bảo vệ chỉ bằng mã trong LDPC mà không có mã ngoài.

Trước hết, thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được xáo trộn. Ngoài ra, thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được mã hoá BCH, và do đó, các bit kiểm tra chẵn lẻ BCH của thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được tạo ra bằng cách mã hoá BCH có thể lần lượt được gắn vào thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1. Ngoài ra, các bit thông tin báo hiệu ghép nối và các bit kiểm tra chẵn lẻ BCH có thể được bảo vệ thêm bằng mã 16K LDPC đã được rút gọn và đục lỗ.

Để tạo ra nhiều mức khả năng bảo vệ khác nhau phù hợp với khoảng giá trị tỷ số tín hiệu/tạp nhiễu (Signal to Noise Ratio, SNR) rộng, thì mức bảo vệ của thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được phân chia ra thành bảy (7) chế độ. Điều này có nghĩa là, mức bảo vệ của thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được phân chia ra thành bảy chế độ dựa vào mã LDPC, bậc điều biến, các thông số rút gọn/đục lỗ (tức là tỷ số giữa số lượng bit được đục lỗ và số lượng bit được rút gọn),

và số lượng bit được đục lỗ cơ bản (tức là số lượng bit được đục lỗ cơ bản khi số lượng bit được rút gọn bằng 0). Ở mỗi chế độ, ít nhất một dạng kết hợp khác nhau của mã LDPC, bậc điều biến, chòm điểm và mẫu rút gọn/đục lỗ có thể được sử dụng.

Chế độ mà thiết bị truyền tín hiệu 100 áp dụng để xử lý thông tin báo hiệu có thể được thiết lập trước tùy theo từng hệ thống. Vì vậy, thiết bị truyền tín hiệu 100 có thể xác định các thông số (ví dụ cấu trúc sơ đồ điều biến và tỷ lệ mã (cấu trúc ModCod) cho mỗi chế độ, thông số cho sơ đồ mã hoá BCH, thông số cho sơ đồ đệm bit không, mẫu rút gọn, tỷ lệ mã/độ dài mã của mã LDPC, mẫu đan xen theo đơn vị nhóm, thông số cho sơ đồ lặp, thông số cho mẫu đục lỗ, và sơ đồ điều biến, v.v.) để xử lý thông tin báo hiệu tùy theo chế độ đã thiết lập, và có thể xử lý thông tin báo hiệu dựa vào các thông số đã xác định và truyền thông tin báo hiệu đã xử lý đến thiết bị thu tín hiệu 200. Nhằm mục đích này, thiết bị truyền tín hiệu 100 có thể lưu trữ trước các thông số để xử lý thông tin báo hiệu tùy theo từng chế độ.

Các cấu trúc sơ đồ điều biến và tỷ lệ mã (cấu trúc ModCod) cho bảy chế độ để xử lý thông tin báo hiệu cơ bản cho tầng L1 và bảy chế độ để xử lý thông tin báo hiệu chi tiết cho tầng L1 được thể hiện trong bảng 4 dưới đây. Thiết bị truyền tín hiệu 100 có thể mã hoá và điều biến thông tin báo hiệu dựa vào các cấu trúc ModCod được xác định trong bảng 4 dưới đây theo chế độ tương ứng. Điều này có nghĩa là, thiết bị truyền tín hiệu 100 có thể xác định tỷ lệ mã và sơ đồ điều biến cho thông tin báo hiệu ở mỗi chế độ dựa vào bảng 4 dưới đây, và có thể mã hoá và điều biến thông tin báo hiệu theo sơ đồ đã xác định. Trong trường hợp như vậy, ngay cả khi điều biến thông tin báo hiệu cho tầng L1 theo cùng một sơ đồ điều biến, nhưng thiết bị truyền tín hiệu 100 vẫn có thể sử dụng các chòm điểm khác nhau.

Bảng 4

Kiểu mã hoá FEC cho thông tin báo hiệu		K _{sig}	Cấu trúc ModCod		Chòm điểm
			Độ dài mã	Tỷ lệ mã	
Thông tin báo hiệu cơ bản cho tầng L1	Chế độ 1	200	16200	3/15 (Loại A)	QPSK
	Chế độ 2				QPSK
	Chế độ 3				QPSK
	Chế độ 4				NUC_16-QAM
	Chế độ 5				NUC_64-QAM
	Chế độ 6				NUC_256-QAM
	Chế độ 7				NUC_256-QAM
Thông tin báo hiệu chi tiết cho tầng L1	Chế độ 1	400 ~ 2352		6/15 (Loại B)	QPSK
	Chế độ 2	400 ~ 3072			QPSK
	Chế độ 3	400 ~ 6312			QPSK
	Chế độ 4				NUC_16-QAM
	Chế độ 5				NUC_64-QAM
	Chế độ 6				NUC_256-QAM
	Chế độ 7				NUC_256-QAM

Trong bảng 4 nêu trên, K_{sig} là số lượng bit thông tin cho khối mã hoá. Điều này có nghĩa là, vì các bit thông tin báo hiệu cho tầng L1 có độ dài K_{sig} được mã hoá để tạo ra khối mã hoá, nên độ dài của thông tin báo hiệu cho tầng L1 trong một khối mã hoá sẽ là K_{sig} . Vì vậy, các bit thông tin báo hiệu cho tầng L1 có độ dài K_{sig} có thể được coi là tương ứng với một khối mã hoá LDPC.

Dựa vào bảng 4 nêu trên, giá trị K_{sig} của thông tin báo hiệu cơ bản cho tầng L1 được giữ cố định bằng 200. Tuy nhiên, vì số lượng bit thông tin báo hiệu chi tiết cho tầng L1 thay đổi, nên giá trị K_{sig} của thông tin báo hiệu chi tiết cho tầng L1 thay đổi.

Cụ thể, đối với thông tin báo hiệu chi tiết cho tầng L1, số lượng bit thông tin báo hiệu chi tiết cho tầng L1 thay đổi, và do đó, khi số lượng bit thông tin báo hiệu chi tiết cho tầng L1 lớn hơn một giá trị định trước, thì thông tin báo hiệu chi tiết cho tầng L1 có thể được phân đoạn sao cho có độ dài bằng hoặc nhỏ hơn giá trị định

trước.

Trong trường hợp như vậy, kích thước của mỗi khối thông tin báo hiệu chi tiết cho tầng L1 đã phân đoạn (tức là đoạn thông tin báo hiệu chi tiết cho tầng L1) có thể có giá trị K_{sig} như được xác định trong bảng 4 nêu trên. Ngoài ra, mỗi khối thông tin báo hiệu chi tiết cho tầng L1 đã phân đoạn có kích thước K_{sig} có thể tương ứng với một khối mã hoá LDPC.

Tuy nhiên, khi số lượng bit thông tin báo hiệu chi tiết cho tầng L1 bằng hoặc nhỏ hơn giá trị định trước, thì thông tin báo hiệu chi tiết cho tầng L1 không được phân đoạn. Trong trường hợp như vậy, độ dài của thông tin báo hiệu chi tiết cho tầng L1 có thể có giá trị K_{sig} như được xác định trong bảng 4 nêu trên. Ngoài ra, thông tin báo hiệu chi tiết cho tầng L1 có độ dài K_{sig} có thể tương ứng với một khối mã hoá LDPC.

Phương pháp phân đoạn thông tin báo hiệu chi tiết cho tầng L1 sẽ được mô tả dưới đây.

Bộ phận phân đoạn 311 phân đoạn thông tin báo hiệu chi tiết cho tầng L1. Cụ thể, vì độ dài của thông tin báo hiệu chi tiết cho tầng L1 thay đổi, nên khi độ dài của thông tin báo hiệu chi tiết cho tầng L1 lớn hơn giá trị định trước, thì bộ phận phân đoạn 311 có thể phân đoạn thông tin báo hiệu chi tiết cho tầng L1 sao cho có số lượng bit bằng hoặc nhỏ hơn giá trị định trước và xuất ra từng đoạn thông tin báo hiệu chi tiết cho tầng L1 để cung cấp cho bộ phận xáo trộn 312.

Tuy nhiên, khi độ dài của thông tin báo hiệu chi tiết cho tầng L1 bằng hoặc nhỏ hơn giá trị định trước, thì bộ phận phân đoạn 311 không thực hiện thao tác phân đoạn riêng biệt.

Phương pháp phân đoạn, bằng bộ phận phân đoạn 311, đối với thông tin báo hiệu chi tiết cho tầng L1 diễn ra như sau.

Số lượng bit thông tin báo hiệu chi tiết cho tầng L1 thay đổi và chủ yếu phụ thuộc vào số lượng kênh truyền PLP. Vì vậy, để truyền tất cả các bit của thông tin báo hiệu chi tiết cho tầng L1, cần có ít nhất một khung sửa lỗi trước (Forward Error Correction, FEC). Theo sáng chế, khung FEC có thể có dạng trong đó thông tin báo hiệu chi tiết cho tầng L1 được mã hoá, và do đó, các bit chẵn lẻ theo sơ đồ mã hoá được gắn vào thông tin báo hiệu chi tiết cho tầng L1.

Cụ thể, khi thông tin báo hiệu chi tiết cho tầng L1 không được phân đoạn, thì thông tin báo hiệu chi tiết cho tầng L1 được mã hoá BCH và mã hoá LDPC để tạo ra một khung FEC, và vì vậy, cần có một khung FEC để truyền thông tin báo hiệu chi tiết cho tầng L1. Mặt khác, khi thông tin báo hiệu chi tiết cho tầng L1 được phân đoạn ra thành ít nhất hai đoạn, thì mỗi đoạn trong số ít nhất hai đoạn thông tin báo hiệu chi tiết cho tầng L1 được mã hoá BCH và mã hoá LDPC để tạo ra ít nhất hai khung FEC, và vì vậy, cần có ít nhất hai khung FEC để truyền thông tin báo hiệu chi tiết cho tầng L1.

Do đó, bộ phận phân đoạn 311 có thể tính số lượng $N_{L1D_FECFRAME}$ khung FEC của thông tin báo hiệu chi tiết cho tầng L1 dựa vào biểu thức 17 dưới đây. Điều này có nghĩa là, số lượng $N_{L1D_FECFRAME}$ khung FEC của thông tin báo hiệu chi tiết cho tầng L1 có thể được xác định dựa vào biểu thức 17 dưới đây:

$$N_{L1D_FECFRAME} = \left\lceil \frac{K_{L1D_ex_pad}}{K_{seg}} \right\rceil \quad (17)$$

Trong biểu thức 17 nêu trên, $\lceil x \rceil$ là số nguyên nhỏ nhất bằng hoặc lớn hơn x .

Ngoài ra, trong biểu thức 17 nêu trên, $K_{L1D_ex_pad}$ là độ dài của thông tin báo hiệu

chi tiết cho tầng L1 không kể L1 bit đệm như được thể hiện trên Fig.44, và có thể được xác định theo giá trị của trường L1B_L1_Detail_size_bits có trong thông tin báo hiệu cơ bản cho tầng L1.

Ngoài ra, K_{seg} là giá trị ngưỡng để phân đoạn, được xác định dựa vào số lượng K_{ldpc} bit thông tin được nhập vào bộ mã hoá LDPC 315, tức là các bit thông tin LDPC. Ngoài ra, K_{seg} có thể được xác định dựa vào số lượng bit kiểm tra chẵn lẻ BCH của mã BCH và giá trị bội số của 360.

K_{seg} được xác định sao cho, sau khi thông tin báo hiệu chi tiết cho tầng L1 được phân đoạn, số lượng K_{sig} bit thông tin trong khối mã hoá được coi là bằng hoặc nhỏ hơn $K_{ldpc} - M_{outer}$. Cụ thể, khi thông tin báo hiệu chi tiết cho tầng L1 được phân đoạn dựa vào K_{seg} , vì độ dài của đoạn thông tin báo hiệu chi tiết cho tầng L1 không vượt quá K_{seg} , nên độ dài của đoạn thông tin báo hiệu chi tiết cho tầng L1 được coi là bằng hoặc nhỏ hơn $K_{ldpc} - M_{outer}$ khi K_{seg} được thiết lập như được thể hiện trong bảng 5 dưới đây.

Theo sáng chế, M_{outer} và K_{ldpc} được xác định trong bảng 6 và bảng 7 dưới đây. Với khả năng bảo vệ phù hợp, giá trị K_{seg} cho thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1 có thể được đặt bằng $K_{ldpc} - M_{outer} - 720$.

K_{seg} cho thông tin báo hiệu chi tiết cho tầng L1 theo mỗi chế độ có thể được xác định như trong bảng 5 dưới đây. Trong trường hợp như vậy, bộ phận phân đoạn 311 có thể xác định K_{seg} theo chế độ tương ứng như được thể hiện trong bảng 5 dưới đây.

Bảng 5

Thông tin báo hiệu chi tiết cho tầng L1	K_{seg}
Chế độ 1	2352
Chế độ 2	3072
Chế độ 3	6312

Chế độ 4	
Chế độ 5	
Chế độ 6	
Chế độ 7	

Như được thể hiện trên Fig.44, toàn bộ thông tin báo hiệu chi tiết cho tầng L1 có thể gồm có các bit thông tin báo hiệu chi tiết cho tầng L1 và các bit đệm cho tầng L1.

Trong trường hợp như vậy, bộ phận phân đoạn 311 có thể tính độ dài của trường L1_PADDING trong thông tin báo hiệu chi tiết cho tầng L1, tức là số lượng K_{L1D_PAD} bit đệm cho tầng L1 dựa vào biểu thức 18 dưới đây.

Tuy nhiên, phương pháp tính K_{L1D_PAD} dựa vào biểu thức 18 dưới đây chỉ là một ví dụ. Điều này có nghĩa là, bộ phận phân đoạn 311 có thể tính độ dài của trường L1_PADDING trong thông tin báo hiệu chi tiết cho tầng L1, tức là số lượng K_{L1D_PAD} bit đệm cho tầng L1 dựa vào các giá trị $K_{L1D_ex_pad}$ và $N_{L1D_FECFRAME}$. Ví dụ, giá trị K_{L1D_PAD} có thể thu được dựa vào biểu thức 18 dưới đây. Điều này có nghĩa là, biểu thức 18 dưới đây chỉ là một ví dụ về phương pháp thu được giá trị K_{L1D_PAD} , và do đó, có thể áp dụng phương pháp khác dựa vào các giá trị $K_{L1D_ex_pad}$ và $N_{L1D_FECFRAME}$ để đạt được kết quả tương đương.

$$K_{L1D_PAD} = \left\lceil \frac{K_{L1D_ex_pad}}{(N_{L1D_FECFRAME} \times 8)} \right\rceil \times 8 \times N_{L1D_FECFRAME} - K_{L1D_ex_pad} \quad (18)$$

Ngoài ra, bộ phận phân đoạn 311 có thể nạp K_{L1D_PAD} bit không (tức là các bit có giá trị 0) vào trường L1_PADDING. Vì vậy, như được thể hiện trên Fig.44, K_{L1D_PAD} bit không có thể được nạp vào trường L1_PADDING.

Như vậy, sau khi tính độ dài của trường L1_PADDING và các bit không được đệm vào theo độ dài đã tính của trường L1_PADDING, thông tin báo hiệu chi tiết cho tầng L1 có thể được phân đoạn ra thành nhiều khối có cùng một số lượng bit khi

thông tin báo hiệu chi tiết cho tầng L1 được phân đoạn.

Tiếp theo, bộ phận phân đoạn 311 có thể tính độ dài cuối cùng K_{L1D} của toàn bộ thông tin báo hiệu chi tiết cho tầng L1 có chứa các bit không được đệm vào dựa vào biểu thức 19 dưới đây:

$$K_{L1D} = K_{L1D_ex_pad} + K_{L1D_PAD} \quad (19)$$

Ngoài ra, bộ phận phân đoạn 311 có thể tính số lượng K_{sig} bit thông tin trong mỗi khối trong số $N_{L1D_FECFRAME}$ khối dựa vào biểu thức 20 dưới đây:

$$K_{sig} = \frac{K_{L1D}}{N_{L1D_FECFRAME}} \quad (20)$$

Tiếp theo, bộ phận phân đoạn 311 có thể phân đoạn thông tin báo hiệu chi tiết cho tầng L1 bằng cách lấy số lượng bit thông tin báo hiệu chi tiết cho tầng L1 chia cho số lượng K_{sig} bit.

Cụ thể, như được thể hiện trên Fig.44, khi $N_{L1D_FECFRAME}$ lớn hơn 1, thì bộ phận phân đoạn 311 có thể lấy số lượng bit thông tin báo hiệu chi tiết cho tầng L1 chia cho số lượng K_{sig} bit để phân đoạn thông tin báo hiệu chi tiết cho tầng L1 ra thành $N_{L1D_FECFRAME}$ khối.

Vì vậy, thông tin báo hiệu chi tiết cho tầng L1 có thể được phân đoạn ra thành $N_{L1D_FECFRAME}$ khối, và số lượng bit thông tin báo hiệu chi tiết cho tầng L1 trong mỗi khối trong số $N_{L1D_FECFRAME}$ khối có thể bằng K_{sig} . Ngoài ra, mỗi đoạn thông tin báo hiệu chi tiết cho tầng L1 được mã hoá. Theo kết quả mã hoá, khối mã hoá, tức là khung FEC, được tạo ra sao cho số lượng bit thông tin báo hiệu chi tiết cho tầng L1 trong mỗi khối trong số $N_{L1D_FECFRAME}$ khối mã hoá có thể bằng K_{sig} .

Tuy nhiên, khi thông tin báo hiệu chi tiết cho tầng L1 không được phân đoạn, thì $K_{sig} = K_{L1D_ex_pad}$.

Các khối thông tin báo hiệu chi tiết cho tầng L1 đã phân đoạn có thể được mã hoá theo quy trình như sau.

Cụ thể, tất cả các bit của mỗi khối trong số các khối thông tin báo hiệu chi tiết cho tầng L1 có độ dài K_{sig} có thể được xáo trộn. Tiếp theo, mỗi khối trong số các khối thông tin báo hiệu chi tiết cho tầng L1 có thể được mã hoá bằng dãy ghép nối của mã ngoài BCH và mã trong LDPC.

Cụ thể, mỗi khối trong số các khối thông tin báo hiệu chi tiết cho tầng L1 được mã hoá BCH, và do đó M_{outer} ($= 168$) bit kiểm tra chẵn lẻ BCH có thể được gắn vào K_{sig} bit thông tin báo hiệu chi tiết cho tầng L1 trong mỗi khối, và sau đó, dãy ghép nối gồm có các bit thông tin báo hiệu chi tiết cho tầng L1 và các bit kiểm tra chẵn lẻ BCH của mỗi khối có thể được mã hoá bằng mã 16K LDPC đã được rút gọn và đục lỗ. Trong đó, mã BCH và mã LDPC sẽ được mô tả chi tiết dưới đây. Tuy nhiên, các phương án làm ví dụ thực hiện sáng chế chỉ mô tả trường hợp trong đó $M_{\text{outer}} = 168$, nhưng cần phải hiểu rõ là M_{outer} có thể thay đổi với giá trị thích hợp tùy thuộc vào các yêu cầu của hệ thống.

Các bộ phận xáo trộn 211 và 312 lần lượt xáo trộn thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1. Cụ thể, các bộ phận xáo trộn 211 và 312 có thể lần lượt ngẫu nhiên hoá thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1, và xuất ra thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 đã được ngẫu nhiên hoá để cung cấp cho các bộ mã hoá BCH 212 và 313 tương ứng.

Trong trường hợp như vậy, các bộ phận xáo trộn 211 và 312 có thể xáo trộn các bit thông tin theo đơn vị là K_{sig} .

Điều này có nghĩa là, vì số lượng bit thông tin báo hiệu cơ bản cho tầng L1 được truyền đến thiết bị thu tín hiệu 200 trong mỗi khung bằng 200, nên bộ phận xáo trộn 211 có thể xáo trộn các bit thông tin báo hiệu cơ bản cho tầng L1 theo đơn vị là K_{sig} ($= 200$).

Vì số lượng bit thông tin báo hiệu cơ bản cho tầng L1 được truyền đến thiết bị thu tín hiệu 200 trong mỗi khung thay đổi, cho nên trong một số trường hợp, thông tin báo hiệu chi tiết cho tầng L1 có thể được phân đoạn bằng bộ phận phân đoạn 311. Ngoài ra, bộ phận phân đoạn 311 có thể xuất ra thông tin báo hiệu chi tiết cho tầng L1 gồm có K_{sig} bit hoặc các khối thông tin báo hiệu chi tiết cho tầng L1 đã phân đoạn để cung cấp cho bộ phận xáo trộn 312. Nhờ đó, bộ phận xáo trộn 312 có thể xáo trộn các bit thông tin báo hiệu chi tiết cho tầng L1 theo đơn vị là K_{sig} được xuất ra từ bộ phận phân đoạn 311.

Các bộ mã hoá BCH 212 và 313 thực hiện bước mã hoá BCH trên thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 để tạo ra các bit kiểm tra chẵn lẻ BCH.

Cụ thể, các bộ mã hoá BCH 212 và 313 có thể lần lượt thực hiện bước mã hoá BCH trên thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được xuất ra từ các bộ phận xáo trộn 211 và 313, để tạo ra các bit kiểm tra chẵn lẻ BCH, và xuất ra các bit mã hoá BCH trong đó các bit kiểm tra chẵn lẻ BCH được gắn vào mỗi thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 để cung cấp cho các bộ đệm bit không 213 và 314 tương ứng.

Ví dụ, các bộ mã hoá BCH 212 và 313 có thể lần lượt thực hiện bước mã hoá BCH trên K_{sig} bit đầu vào để tạo ra M_{outer} (tức là $K_{sig} = K_{payload}$) bit kiểm tra chẵn lẻ

BCH và xuất ra các bit mã hoá BCH gồm có $N_{\text{outer}} (= K_{\text{sig}} + M_{\text{outer}})$ bit để cung cấp cho các bộ đệm bit không 213 và 314 tương ứng.

Các thông số để mã hoá BCH có thể được xác định trong bảng 6 dưới đây.

Bảng 6

Kiểu mã hoá FEC cho thông tin báo hiệu		$K_{\text{sig}} = K_{\text{payload}}$	M_{outer}	$N_{\text{outer}} = K_{\text{sig}} + M_{\text{outer}}$
Thông tin báo hiệu cơ bản cho tầng L1	Chế độ 1	200	168	368
	Chế độ 2			
	Chế độ 3			
	Chế độ 4			
	Chế độ 5			
	Chế độ 6			
	Chế độ 7			
Thông tin báo hiệu chi tiết cho tầng L1	Chế độ 1	400 ~ 2352		568 ~ 2520
	Chế độ 2	400 ~ 3072		568 ~ 3240
	Chế độ 3	400 ~ 6312		568 ~ 6480
	Chế độ 4			
	Chế độ 5			
	Chế độ 6			
	Chế độ 7			

Dựa vào Fig.42 và Fig.43, cần phải hiểu rằng các bộ mã hoá LDPC 214 và 315 có thể lần lượt đặt ở sau các bộ mã hoá BCH 212 và 313.

Vì vậy, thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được bảo vệ bằng dãy ghép nối của mã ngoài BCH và mã trong LDPC.

Cụ thể, thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được mã hoá BCH, và do đó, các bit kiểm tra chẵn lẻ BCH của thông tin báo hiệu cơ bản cho tầng L1 được gắn vào thông tin báo hiệu cơ bản cho tầng L1 và các bit kiểm tra chẵn lẻ BCH của thông tin báo hiệu chi tiết cho tầng L1 được gắn vào thông tin báo hiệu chi tiết cho tầng L1. Ngoài ra, các bit thông tin báo hiệu cơ bản cho tầng L1 và các bit kiểm tra chẵn lẻ BCH đã được ghép nối được bảo vệ thêm

bằng mã LDPC, và các bit thông tin báo hiệu chi tiết cho tầng L1 và các bit kiểm tra chẵn lẻ BCH đã được ghép nối có thể được bảo vệ thêm bằng mã LDPC.

Theo sáng chế, giả sử rằng mã LDPC là mã 16K LDPC, và do đó, trong các bộ mã hoá BCH 212 và 213, mã BCH có hệ thống với $N_{\text{inner}} = 16200$ (tức là độ dài của mã 16K LDPC bằng 16200 và từ mã LDPC được tạo ra bằng cách mã hoá LDPC có thể gồm có 16200 bit) có thể được sử dụng để thực hiện bước mã hoá ngoài trên thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1.

Các bộ đệm bit không 213 và 314 đệm các bit không. Cụ thể, đối với mã LDPC, cần có một số lượng định trước của các bit thông tin LDPC được xác định theo tỷ lệ mã và độ dài mã, và do đó, các bộ đệm bit không 213 và 314 có thể lần lượt đệm các bit không để mã hoá LDPC nhằm tạo ra số lượng định trước của các bit thông tin LDPC gồm có các bit mã hoá BCH và các bit không, và xuất ra các bit đã tạo ra để cung cấp cho các bộ mã hoá LDPC 214 và 315 tương ứng, khi số lượng bit mã hoá BCH nhỏ hơn số lượng bit thông tin LDPC. Ngoài ra, khi số lượng bit mã hoá BCH bằng số lượng bit thông tin LDPC, thì các bit không không được đệm vào.

Theo sáng chế, các bit không được đệm vào bằng các bộ đệm bit không 213 và 314 được đệm vào để mã hoá LDPC, và vì vậy, các bit không được đệm vào không được truyền đến thiết bị thu tín hiệu 200 nhờ quy trình rút gọn.

Ví dụ, khi số lượng bit thông tin LDPC của mã 16K LDPC bằng K_{ldpc} , để tạo ra K_{ldpc} bit thông tin LDPC, thì các bit không sẽ được đệm vào.

Cụ thể, khi số lượng bit mã hoá BCH bằng N_{outer} , số lượng bit thông tin LDPC của mã 16K LDPC bằng K_{ldpc} , và $N_{\text{outer}} < K_{\text{ldpc}}$, thì các bộ đệm bit không 213 và 314 có thể đệm $K_{\text{ldpc}} - N_{\text{outer}}$ bit không và sử dụng N_{outer} bit mã hoá BCH, dưới dạng là

phần còn lại của các bit thông tin LDPC, để tạo ra các bit thông tin LDPC gồm có K_{ldpc} bit. Tuy nhiên, khi $N_{outer} = K_{ldpc}$, thì các bit không sẽ không được đệm vào.

Nhằm mục đích này, các bộ đệm bit không 213 và 314 có thể phân chia các bit thông tin LDPC ra thành nhiều nhóm bit.

Ví dụ, các bộ đệm bit không 213 và 314 có thể phân chia K_{ldpc} bit thông tin LDPC ($i_0, i_1, \dots, i_{K_{ldpc}-1}$) ra thành $N_{info_group} (= K_{ldpc}/360)$ nhóm bit dựa vào biểu thức 21 hoặc biểu thức 22 dưới đây. Điều này có nghĩa là, các bộ đệm bit không 213 và 314 có thể phân chia các bit thông tin LDPC ra thành các nhóm bit sao cho số lượng bit có trong mỗi nhóm bit bằng 360.

$$Z_j = \left\{ i_k \mid j = \left\lfloor \frac{k}{360} \right\rfloor, 0 \leq k < K_{ldpc} \right\} \text{ với } 0 \leq j < N_{info_group} \quad (21)$$

$$Z_j = \{ i_k \mid 360 \times j \leq k < 360 \times (j+1) \} \text{ với } 0 \leq j < N_{info_group} \quad (22)$$

Trong biểu thức 21 và biểu thức 22 nêu trên, Z_j là nhóm bit thứ j .

Các thông số N_{outer} , K_{ldpc} và N_{info_group} của các bit không được đệm vào trong thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được xác định như được thể hiện trong bảng 7 dưới đây. Trong trường hợp như vậy, các bộ đệm bit không 213 và 314 có thể xác định các thông số cho các bit không được đệm vào theo chế độ tương ứng như được thể hiện trong bảng 7 dưới đây.

Bảng 7

Kiểu mã hoá FEC cho thông tin báo hiệu	N_{outer}	K_{ldpc}	$N_{\text{info_group}}$
Thông tin báo hiệu cơ bản cho tầng L1 (tất cả các chế độ)	368	3240	9
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1	568 ~ 2520		
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2	568 ~ 3240		
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 3	568 ~ 6480	6480	18
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 4			
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 5			
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 6			
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 7			

Ngoài ra, với $0 \leq j < N_{\text{info_group}}$, mỗi nhóm bit Z_j như được thể hiện Fig.45 có thể có 360 bit.

Cụ thể, Fig.45 thể hiện định dạng dữ liệu sau khi mỗi thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được mã hoá LDPC. Trên Fig.45, mã LDPC FEC được gắn vào K_{ldpc} bit thông tin LDPC biểu thị các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Dựa vào Fig.45, K_{ldpc} bit thông tin LDPC được phân chia ra thành $N_{\text{info_group}}$ nhóm bit và mỗi nhóm bit có thể có 360 bit.

Khi số lượng $N_{\text{outer}} (= K_{\text{sig}} + M_{\text{outer}})$ bit mã hoá BCH của thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 nhỏ hơn K_{ldpc} , tức là $N_{\text{outer}} (= K_{\text{sig}} + M_{\text{outer}}) < K_{\text{ldpc}}$, để mã hoá LDPC, thì K_{ldpc} bit thông tin LDPC có thể được nạp N_{outer} bit mã hoá BCH và $K_{\text{ldpc}} - N_{\text{outer}}$ bit không sẽ được đệm vào. Trong trường hợp như vậy, các bit không được đệm vào sẽ không được truyền đến thiết bị

thu tín hiệu 200.

Quy trình rút gọn được thực hiện bằng các bộ đệm bit không 213 và 314 sẽ được mô tả chi tiết dưới đây.

Các bộ đệm bit không 213 và 314 có thể tính số lượng bit không được đệm vào. Điều này có nghĩa là, để có đủ số lượng bit cần thiết để mã hoá LDPC, thì các bộ đệm bit không 213 và 314 có thể tính số lượng bit không cần đệm vào.

Cụ thể, các bộ đệm bit không 213 và 314 có thể tính hiệu số giữa số lượng bit thông tin LDPC và số lượng bit mã hoá BCH để lấy kết quả đó làm số lượng bit không được đệm vào. Điều này có nghĩa là, với N_{outer} cho trước, các bộ đệm bit không 213 và 314 có thể tính số lượng bit không được đệm vào dưới dạng là $K_{\text{ldpc}} - N_{\text{outer}}$.

Ngoài ra, các bộ đệm bit không 213 và 314 có thể tính số lượng nhóm bit mà tất cả các bit ở trong đó đều là bit được đệm vào. Điều này có nghĩa là, các bộ đệm bit không 213 và 314 có thể tính số lượng nhóm bit mà tất cả các bit ở trong nhóm bit đó đều là các bit không được đệm vào.

Cụ thể, các bộ đệm bit không 213 và 314 có thể tính số lượng N_{pad} nhóm bit mà tất cả các bit ở trong đó đều là bit được đệm vào dựa vào biểu thức 23 hoặc biểu thức 24 dưới đây:

$$N_{\text{pad}} = \left\lfloor \frac{K_{\text{ldpc}} - N_{\text{outer}}}{360} \right\rfloor \quad (23)$$

$$N_{\text{pad}} = \left\lfloor \frac{(K_{\text{ldpc}} - M_{\text{outer}}) - K_{\text{sig}}}{360} \right\rfloor \quad (24)$$

Tiếp theo, các bộ đệm bit không 213 và 314 có thể xác định các nhóm bit mà các bit không được đệm vào trong đó trong số nhiều nhóm bit dựa vào mẫu rút gọn,

và có thể đệm các bit không vào tất cả các bit trong một số nhóm bit đã xác định và một số bit trong các nhóm bit còn lại.

Trong trường hợp như vậy, mẫu rút gọn của nhóm bit đệm có thể được xác định như được thể hiện trong bảng 8 dưới đây. Trong trường hợp như vậy, các bộ đệm bit không 213 và 314 có thể xác định mẫu rút gọn theo chế độ tương ứng như được thể hiện trong bảng 8 dưới đây.

Bảng 8

Kiểu mã hoá FEC cho thông tin báo hiệu	N _{info_group}	$\pi_s(j)$ ($0 \leq j < N_{\text{info_group}}$)									
		$\pi_s(0)$	$\pi_s(1)$	$\pi_s(2)$	$\pi_s(3)$	$\pi_s(4)$	$\pi_s(5)$	$\pi_s(6)$	$\pi_s(7)$	$\pi_s(8)$	
		$\pi_s(9)$	$\pi_s(10)$	$\pi_s(11)$	$\pi_s(12)$	$\pi_s(13)$	$\pi_s(14)$	$\pi_s(15)$	$\pi_s(16)$	$\pi_s(17)$	
Thông tin báo hiệu cơ bản cho tầng L1 (tất cả các chế độ)	9	4	1	5	2	8	6	0	7	3	
		-	-	-	-	-	-	-	-	-	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1		7	8	5	4	1	2	6	3	0	
		-	-	-	-	-	-	-	-	-	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2		6	1	7	8	0	2	4	3	5	
		-	-	-	-	-	-	-	-	-	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 3	18	0	12	15	13	2	5	7	9	8	
		6	16	10	14	1	17	11	4	3	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 4		0	15	5	16	17	1	6	13	11	
		4	7	12	8	14	2	3	9	10	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 5		2	4	5	17	9	7	1	6	15	
		8	10	14	16	0	11	13	12	3	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 6		0	15	5	16	17	1	6	13	11	
		4	7	12	8	14	2	3	9	10	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 7		15	7	8	11	5	10	16	4	12	
		3	0	6	9	1	14	17	2	13	

Theo sáng chế, $\pi_s(j)$ là chỉ số của nhóm bit được đệm vào thứ j. Điều này có nghĩa là, giá trị $\pi_s(j)$ biểu thị thứ tự mẫu rút gọn của nhóm bit thứ j. Ngoài ra,

$N_{\text{info_group}}$ là số lượng nhóm bit tạo nên các bit thông tin LDPC.

Cụ thể, các bộ đệm bit không 213 và 314 có thể xác định $Z_{\pi_s(0)}, Z_{\pi_s(1)}, \dots, Z_{\pi_s(N_{\text{pad}}-1)}$ là các nhóm bit mà tất cả các bit ở trong nhóm bit đó đều là các bit không được đệm vào dựa vào mẫu rút gọn, và đệm các bit không vào tất cả các bit ở trong các nhóm bit đó. Điều này có nghĩa là, các bộ đệm bit không 213 và 314 có thể đệm các bit không vào tất cả các bit ở trong nhóm bit thứ $\pi_s(0)$, nhóm bit thứ $\pi_s(1)$, ..., nhóm bit thứ $\pi_s(N_{\text{pad}}-1)$ trong số các nhóm bit dựa vào mẫu rút gọn.

Như vậy, khi N_{pad} khác 0, thì các bộ đệm bit không 213 và 314 có thể xác định danh sách gồm N_{pad} nhóm bit, tức là $Z_{\pi_s(0)}, Z_{\pi_s(1)}, \dots, Z_{\pi_s(N_{\text{pad}}-1)}$ dựa vào bảng 8 nêu trên, và đệm các bit không vào tất cả các bit trong nhóm bit đã xác định.

Tuy nhiên, khi N_{pad} bằng 0, thì quy trình nêu trên có thể không được thực hiện.

Vì số lượng bit của tất cả các bit không được đệm vào bằng $K_{\text{ldpc}} - N_{\text{outer}}$ và số lượng bit không được đệm vào N_{pad} nhóm bit bằng $360 \times N_{\text{pad}}$, cho nên các bộ đệm bit không 213 và 314 có thể đệm thêm các bit không vào $K_{\text{ldpc}} - N_{\text{outer}} - 360 \times N_{\text{pad}}$ bit thông tin LDPC.

Trong trường hợp như vậy, các bộ đệm bit không 213 và 314 có thể xác định nhóm bit mà các bit không được đệm thêm vào ở trong đó dựa vào mẫu rút gọn, và có thể đệm thêm các bit không từ phần đầu của nhóm bit đã xác định.

Cụ thể, các bộ đệm bit không 213 và 314 có thể xác định $Z_{\pi_s(N_{\text{pad}})}$ là nhóm bit mà các bit không được đệm thêm vào ở trong đó dựa vào mẫu rút gọn, và có thể đệm thêm các bit không vào $K_{\text{ldpc}} - N_{\text{outer}} - 360 \times N_{\text{pad}}$ bit nằm ở phần đầu của nhóm bit $Z_{\pi_s(N_{\text{pad}})}$. Vì vậy, $K_{\text{ldpc}} - N_{\text{outer}} - 360 \times N_{\text{pad}}$ bit không có thể được đệm vào kể từ bit đầu tiên của nhóm bit thứ $\pi_s(N_{\text{pad}})$.

Nhờ đó, với nhóm bit $Z_{\pi_s(N_{\text{pad}})}$, các bit không có thể được đệm thêm vào $K_{\text{ldpc}} - N_{\text{bch}} - 360 \times N_{\text{pad}}$ bit nằm ở phần đầu của nhóm bit $Z_{\pi_s(N_{\text{pad}})}$.

Ví dụ nêu trên mô tả rằng $K_{\text{ldpc}} - N_{\text{outer}} - 360 \times N_{\text{pad}}$ bit không được đệm vào kể từ bit đầu tiên của nhóm bit $Z_{\pi_s(N_{\text{pad}})}$, nhưng đó chỉ là một ví dụ. Vì vậy, vị trí mà các bit không được đệm vào ở đó trong nhóm bit $Z_{\pi_s(N_{\text{pad}})}$ có thể thay đổi. Ví dụ, $K_{\text{ldpc}} - N_{\text{outer}} - 360 \times N_{\text{pad}}$ bit không có thể được đệm vào phần giữa hoặc phần cuối của nhóm bit $Z_{\pi_s(N_{\text{pad}})}$ hoặc cũng có thể được đệm vào ở một vị trí bất kỳ trong nhóm bit $Z_{\pi_s(N_{\text{pad}})}$.

Tiếp theo, các bộ đệm bit không 213 và 314 có thể ánh xạ các bit mã hoá BCH lên các vị trí mà các bit không được đệm vào ở đó để tạo nên các bit thông tin LDPC.

Vì vậy, N_{outer} bit mã hoá BCH được ánh xạ tuần tự lên các vị trí bit mà các bit không trong số K_{ldpc} bit thông tin LDPC ($i_0, i_1, \dots, i_{K_{\text{ldpc}}-1}$) không được đệm vào ở đó, và vì vậy, K_{ldpc} bit thông tin LDPC có thể gồm có N_{outer} bit mã hoá BCH và $K_{\text{ldpc}} - N_{\text{outer}}$ bit thông tin.

Các bit không được đệm vào không được truyền đến thiết bị thu tín hiệu 200. Như vậy, thủ tục đệm bit không hay quy trình đệm bit không và sau đó không truyền các bit không được đệm vào đến thiết bị thu tín hiệu 200 có thể được gọi là thủ tục rút gọn.

Các bộ mã hoá LDPC 214 và 315 lần lượt thực hiện bước mã hoá LDPC trên thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1.

Cụ thể, các bộ mã hoá LDPC 214 và 315 có thể lần lượt thực hiện bước mã hoá LDPC trên các bit thông tin LDPC được xuất ra từ các bộ đệm bit không 213 và 314

để tạo ra các bit chẵn lẻ LDPC, và xuất ra từ mã LDPC gồm có các bit thông tin LDPC và các bit chẵn lẻ LDPC để cung cấp cho các bộ phận hoán vị chẵn lẻ 215 và 316 tương ứng.

Điều này có nghĩa là, K_{ldpc} bit được xuất ra từ bộ đệm bit không 213 có thể có K_{sig} bit thông tin báo hiệu cơ bản cho tầng L1, $M_{outer} (= N_{outer} - K_{sig})$ bit kiểm tra chẵn lẻ BCH, và $(K_{ldpc} - N_{outer})$ bit không được đệm, các bit này có thể tạo nên K_{ldpc} bit thông tin LDPC $i = (i_0, i_1, \dots, i_{K_{ldpc}-1})$ cho bộ mã hoá LDPC 214.

Ngoài ra, K_{ldpc} bit được xuất ra từ bộ đệm bit không 314 có thể có K_{sig} bit thông tin báo hiệu chi tiết cho tầng L1, $M_{outer} (= N_{outer} - K_{sig})$ bit kiểm tra chẵn lẻ BCH, và $(K_{ldpc} - N_{outer})$ bit không được đệm, các bit này có thể tạo nên K_{ldpc} bit thông tin LDPC $i = (i_0, i_1, \dots, i_{K_{ldpc}-1})$ cho bộ mã hoá LDPC 315.

Trong trường hợp như vậy, các bộ mã hoá LDPC 214 và 315 có thể thực hiện một cách có hệ thống bước mã hoá LDPC trên K_{ldpc} bit thông tin LDPC để tạo ra từ mã LDPC $\Lambda = (c_0, c_1, \dots, c_{N_{inner}-1}) = (i_0, i_1, \dots, i_{K_{ldpc}-1}, p_0, p_1, \dots, p_{N_{inner}-K_{ldpc}-1})$ gồm có N_{inner} bit.

Đối với thông tin báo hiệu cơ bản cho tầng L1 theo các chế độ và thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1 và chế độ 2, các bộ mã hoá LDPC 214 và 315 có thể mã hoá thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 ở tỷ lệ mã bằng 3/15 để tạo ra 16200 bit từ mã LDPC. Trong trường hợp như vậy, các bộ mã hoá LDPC 214 và 315 có thể thực hiện bước mã hoá LDPC dựa vào bảng 1 nêu trên.

Ngoài ra, đối với thông tin báo hiệu chi tiết cho tầng L1 theo các chế độ 3, 4, 5, 6 và 7, bộ mã hoá LDPC 315 có thể mã hoá thông tin báo hiệu chi tiết cho tầng L1 ở

tỷ lệ mã bằng 6/15 để tạo ra 16200 bit từ mã LDPC. Trong trường hợp như vậy, bộ mã hoá LDPC 315 có thể thực hiện bước mã hoá LDPC dựa vào bảng 3 nêu trên.

Tỷ lệ mã và độ dài mã của thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được xác định như được thể hiện trong bảng 4 nêu trên, và số lượng bit thông tin LDPC được xác định như được thể hiện trong bảng 7 nêu trên.

Các bộ phận hoán vị chẵn lẻ 215 và 316 thực hiện bước hoán vị chẵn lẻ. Điều này có nghĩa là, các bộ phận hoán vị chẵn lẻ 215 và 316 có thể thực hiện bước hoán vị chỉ trên các bit chẵn lẻ LDPC trong số các bit thông tin LDPC và các bit chẵn lẻ LDPC.

Cụ thể, các bộ phận hoán vị chẵn lẻ 215 và 316 lần lượt có thể thực hiện bước hoán vị chỉ trên các bit chẵn lẻ LDPC trong các từ mã LDPC được xuất ra từ các bộ mã hoá LDPC 214 và 315, và xuất ra các từ mã LDPC đã được hoán vị chẵn lẻ để cung cấp cho các bộ phận lặp 216 và 317 tương ứng. Trong đó, bộ phận hoán vị chẵn lẻ 316 có thể xuất ra từ mã LDPC đã được hoán vị chẵn lẻ để cung cấp cho bộ tạo bit chẵn lẻ bổ sung 319. Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 319 có thể sử dụng từ mã LDPC đã được hoán vị chẵn lẻ được xuất ra từ bộ phận hoán vị chẵn lẻ 316 để tạo ra các bit chẵn lẻ bổ sung.

Nhằm mục đích này, các bộ phận hoán vị chẵn lẻ 215 và 316 có thể có bộ phận đan xen chẵn lẻ (không được thể hiện trên hình vẽ) và bộ phận đan xen theo đơn vị nhóm (không được thể hiện trên hình vẽ).

Trước hết, bộ phận đan xen có thể chỉ đan xen các bit chẵn lẻ LDPC trong số các bit thông tin LDPC và các bit chẵn lẻ LDPC tạo nên từ mã LDPC. Tuy nhiên, bộ

phần đan xen có thể thực hiện bước đan xen chẵn lẻ chỉ trên thông tin báo hiệu chi tiết cho tầng L1 theo các chế độ 3, 4, 5, 6 và 7. Điều này có nghĩa là, vì thông tin báo hiệu cơ bản cho tầng L1 theo các chế độ và thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1 và chế độ 2 có bước đan xen chẵn lẻ như là một phần trong quy trình mã hoá LDPC, cho nên đối với thông tin báo hiệu cơ bản cho tầng L1 theo các chế độ và thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1 và chế độ 2, bộ phận đan xen có thể không thực hiện bước đan xen chẵn lẻ.

Ở chế độ thực hiện bước đan xen chẵn lẻ, bộ phận đan xen có thể đan xen các bit chẵn lẻ LDPC dựa vào biểu thức 25 dưới đây:

$$\begin{aligned} u_i &= c_i \text{ với } 0 \leq i < K_{\text{ldpc}} \text{ (các bit thông tin không được đan xen)} \\ u_{K_{\text{ldpc}}+360t+s} &= c_{K_{\text{ldpc}}+27s+t} \text{ với } 0 \leq s < 360, 0 \leq t < 27 \end{aligned} \quad (25)$$

Cụ thể, dựa vào biểu thức 25 nêu trên, từ mã LDPC $(c_0, c_1, \dots, c_{N_{\text{inner}}-1})$ được đan xen chẵn lẻ bằng bộ phận đan xen và tín hiệu đầu ra của bộ phận đan xen có thể được biểu diễn dưới dạng $U = (u_0, u_1, \dots, u_{N_{\text{inner}}-1})$.

Vì thông tin báo hiệu cơ bản cho tầng L1 theo các chế độ và thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1 và chế độ 2 không sử dụng bộ phận đan xen, nên tín hiệu đầu ra $U = (u_0, u_1, \dots, u_{N_{\text{inner}}-1})$ của bộ phận đan xen có thể được biểu diễn theo biểu thức 26 dưới đây:

$$u_i = c_i \text{ với } 0 \leq i < N_{\text{inner}} \quad (26)$$

Bộ phận đan xen theo đơn vị nhóm có thể thực hiện bước đan xen theo đơn vị nhóm trên tín hiệu đầu ra của bộ phận đan xen.

Theo sáng chế, như đã nêu trên, tín hiệu đầu ra của bộ phận đan xen có thể là từ mã LDPC được đan xen chẵn lẻ bằng bộ phận đan xen hoặc có thể là từ mã LDPC

không được đan xen chẵn lẻ bằng bộ phận đan xen.

Vì vậy, khi bước đan xen chẵn lẻ được thực hiện, bộ phận đan xen theo đơn vị nhóm có thể thực hiện bước đan xen theo đơn vị nhóm trên từ mã LDPC đã được đan xen chẵn lẻ, và khi bước đan xen chẵn lẻ không được thực hiện, bộ phận đan xen theo đơn vị nhóm có thể thực hiện bước đan xen theo đơn vị nhóm trên từ mã LDPC không được đan xen chẵn lẻ.

Cụ thể, bộ phận đan xen theo đơn vị nhóm có thể đan xen tín hiệu đầu ra của bộ phận đan xen theo đơn vị nhóm bit.

Nhằm mục đích này, bộ phận đan xen theo đơn vị nhóm có thể phân chia từ mã LDPC được xuất ra từ bộ phận đan xen ra thành nhiều nhóm bit. Nhờ đó, các bit chẵn lẻ LDPC được xuất ra từ bộ phận đan xen có thể được phân chia ra thành nhiều nhóm bit.

Cụ thể, bộ phận đan xen theo đơn vị nhóm có thể phân chia các bit mã hoá LDPC $(u_0, u_1, \dots, u_{N_{\text{inner}}-1})$ được xuất ra từ bộ phận đan xen ra thành $N_{\text{group}} (= N_{\text{inner}}/360)$ nhóm bit dựa vào biểu thức 27 dưới đây:

$$X_j = \{u_k | 360 \times j \leq k < 360 \times (j+1), 0 \leq k < N_{\text{inner}}\} \text{ với } 0 \leq j < N_{\text{group}} \quad (27)$$

Trong biểu thức 27 nêu trên, X_j là nhóm bit thứ j .

Fig.46 thể hiện ví dụ về phương pháp phân chia từ mã LDPC được xuất ra từ bộ phận đan xen ra thành nhiều nhóm bit.

Dựa vào Fig.46, từ mã LDPC được phân chia ra thành $N_{\text{group}} (= N_{\text{inner}}/360)$ nhóm bit, và mỗi nhóm bit X_j với $0 \leq j < N_{\text{group}}$ có 360 bit.

Nhờ đó, các bit thông tin LDPC gồm có K_{ldpc} bit có thể được phân chia ra thành $K_{\text{ldpc}}/360$ nhóm bit và các bit chẵn lẻ LDPC gồm có $N_{\text{inner}} - K_{\text{ldpc}}$ bit có thể được phân

chia ra thành $(N_{\text{inner}} - K_{\text{ldpc}})/360$ nhóm bit.

Ngoài ra, bộ phận đan xen theo đơn vị nhóm thực hiện bước đan xen theo đơn vị nhóm trên từ mã LDPC được xuất ra từ bộ phận đan xen.

Trong trường hợp như vậy, bộ phận đan xen theo đơn vị nhóm không thực hiện bước đan xen trên các bit thông tin LDPC, và có thể thực hiện bước đan xen chỉ trên các bit chẵn lẻ LDPC trong số các bit thông tin LDPC và các bit chẵn lẻ LDPC để thay đổi thứ tự của nhiều nhóm bit tạo nên các bit chẵn lẻ LDPC.

Nhờ đó, các bit thông tin LDPC trong số các bit LDPC có thể không được đan xen bằng bộ phận đan xen theo đơn vị nhóm, mà chỉ có các bit chẵn lẻ LDPC trong số các bit LDPC có thể được đan xen bằng bộ phận đan xen theo đơn vị nhóm. Trong trường hợp như vậy, các bit chẵn lẻ LDPC có thể được đan xen theo đơn vị nhóm.

Cụ thể, bộ phận đan xen theo đơn vị nhóm có thể thực hiện bước đan xen theo đơn vị nhóm trên từ mã LDPC được xuất ra từ bộ phận đan xen dựa vào biểu thức 28 dưới đây:

$$\begin{aligned} Y_j &= X_j, & 0 \leq j < K_{\text{ldpc}}/360 \\ Y_j &= X_{\pi_p(j)}, & K_{\text{ldpc}}/360 \leq j < N_{\text{group}} \end{aligned} \quad (28)$$

Trong đó, X_j là nhóm bit thứ j trong số các nhóm bit tạo nên từ mã LDPC, tức là nhóm bit thứ j không được đan xen theo đơn vị nhóm, và Y_j là nhóm bit thứ j được đan xen theo đơn vị nhóm. Ngoài ra, $\pi_p(j)$ là thứ tự hoán vị để đan xen theo đơn vị nhóm.

Trong đó, thứ tự hoán vị có thể được xác định dựa vào bảng 9 và bảng 10 dưới đây. Theo sáng chế, bảng 9 thể hiện mẫu đan xen theo đơn vị nhóm của phần chẵn lẻ trong thông tin báo hiệu cơ bản cho tầng L1 theo các chế độ và thông tin báo hiệu chi

tiết cho tầng L1 theo chế độ 1 và chế độ 2, và bảng 10 thể hiện mẫu đan xen theo đơn vị nhóm của phần chẵn lẻ trong thông tin báo hiệu chi tiết cho tầng L1 theo các chế độ 3, 4, 5, 6 và 7.

Trong trường hợp như vậy, bộ phận đan xen theo đơn vị nhóm có thể xác định mẫu đan xen theo đơn vị nhóm theo chế độ tương ứng được thể hiện trong bảng 9 và bảng 10 dưới đây.

Bảng 9

Kiểu mã hoá FEC cho thông tin báo hiệu	N _{group}	Thứ tự đan xen theo đơn vị nhóm π _p (j) (9 ≤ j < 45)											
		π _p (9)	π _p (10)	π _p (11)	π _p (12)	π _p (13)	π _p (14)	π _p (15)	π _p (16)	π _p (17)	π _p (18)	π _p (19)	π _p (20)
		π _p (21)	π _p (22)	π _p (23)	π _p (24)	π _p (25)	π _p (26)	π _p (27)	π _p (28)	π _p (29)	π _p (30)	π _p (31)	π _p (32)
		π _p (33)	π _p (34)	π _p (35)	π _p (36)	π _p (37)	π _p (38)	π _p (39)	π _p (40)	π _p (41)	π _p (42)	π _p (43)	π _p (44)
Thông tin báo hiệu cơ bản cho tầng L1 (tất cả các chế độ)	45	20	23	25	32	38	41	18	9	10	11	31	24
		14	15	26	40	33	19	28	34	16	39	27	30
		21	44	43	35	42	36	12	13	29	22	37	17
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1		16	22	27	30	37	44	20	23	25	32	38	41
		9	10	17	18	21	33	35	14	28	12	15	19
		11	24	29	34	36	13	40	43	31	26	39	42
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2		9	31	23	10	11	25	43	29	36	16	27	34
		26	18	37	15	13	17	35	21	20	24	44	12
		22	40	19	32	38	41	30	33	14	28	39	42

Bảng 10

Kiểu mã hoá FEC cho thông tin báo hiệu	N_{group}	Thứ tự đan xen theo đơn vị nhóm $\pi_p(j)$ ($18 \leq j < 45$)												
		$\pi_p(18)$	$\pi_p(19)$	$\pi_p(20)$	$\pi_p(21)$	$\pi_p(22)$	$\pi_p(23)$	$\pi_p(24)$	$\pi_p(25)$	$\pi_p(26)$	$\pi_p(27)$	$\pi_p(28)$	$\pi_p(29)$	$\pi_p(30)$
		$\pi_p(31)$	$\pi_p(32)$	$\pi_p(33)$	$\pi_p(34)$	$\pi_p(35)$	$\pi_p(36)$	$\pi_p(37)$	$\pi_p(38)$	$\pi_p(39)$	$\pi_p(40)$	$\pi_p(41)$	$\pi_p(42)$	$\pi_p(43)$
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 3	45	19	37	30	42	23	44	27	40	21	34	25	32	29
		26	35	39	20	18	43	31	36	38	22	33	28	41
Thông tin báo hiệu		20	35	42	39	26	23	30	18	28	37	32	27	44

chi tiết cho tầng L1 theo chế độ 4		41	40	38	36	34	33	31	29	25	24	22	21	19	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 5		19	37	33	26	40	43	22	29	24	35	44	31	27	20
		21	39	25	42	34	18	32	38	23	30	28	36	41	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 6		20	35	42	39	26	23	30	18	28	37	32	27	44	43
		41	40	38	36	34	33	31	29	25	24	22	21	19	
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 7		44	23	29	33	24	28	21	27	42	18	22	31	32	37
		43	30	25	35	20	34	39	36	19	41	40	26	38	

Ví dụ, đối với mẫu đan xen theo đơn vị nhóm trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, sự hoạt động của bộ phận đan xen theo đơn vị nhóm sẽ được mô tả dưới đây.

Trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, bộ mã hoá LDPC 315 thực hiện bước mã hoá LDPC trên 3240 bit thông tin LDPC ở tỷ lệ mã bằng 3/15 để tạo ra 12960 bit chẵn lẻ LDPC. Trong trường hợp như vậy, từ mã LDPC có thể gồm có 16200 bit.

Mỗi nhóm bit có 360 bit, và vì vậy từ mã LDPC gồm có 16200 bit được phân chia ra thành 45 nhóm bit.

Theo sáng chế, vì số lượng bit thông tin LDPC bằng 3240 và số lượng bit chẵn lẻ LDPC bằng 12960, nên từ nhóm bit thứ 0 đến nhóm bit thứ 8 tương ứng với các bit thông tin LDPC và từ nhóm bit thứ 9 đến nhóm bit thứ 44 tương ứng với các bit chẵn lẻ LDPC.

Trong trường hợp như vậy, bộ phận đan xen theo đơn vị nhóm không thực hiện bước đan xen trên các nhóm bit tạo nên các bit thông tin LDPC, tức là từ nhóm bit thứ 0 đến nhóm bit thứ 8 dựa vào biểu thức 28 và bảng 9 nêu trên, nhưng có thể thực hiện bước đan xen trên các nhóm bit tạo nên các bit chẵn lẻ LDPC, tức là từ nhóm bit thứ 9 đến nhóm bit thứ 44 theo đơn vị nhóm để thay đổi thứ tự của các nhóm bit từ

nhóm bit thứ 9 đến nhóm bit thứ 44.

Cụ thể, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2 trong bảng 9 nêu trên, biểu thức 28 nêu trên có thể được biểu diễn dưới dạng $Y_0 = X_0$, $Y_1 = X_1$, ..., $Y_7 = X_7$, $Y_8 = X_8$, $Y_9 = X_{\pi p(9)} = X_9$, $Y_{10} = X_{\pi p(10)} = X_{31}$, $Y_{11} = X_{\pi p(11)} = X_{23}$, ..., $Y_{42} = X_{\pi p(42)} = X_{28}$, $Y_{43} = X_{\pi p(43)} = X_{39}$, $Y_{44} = X_{\pi p(44)} = X_{42}$.

Vì vậy, bộ phận đan xen theo đơn vị nhóm không thay đổi thứ tự của các nhóm bit từ nhóm bit thứ 0 đến nhóm bit thứ 8 chứa các bit thông tin LDPC, nhưng có thể thay đổi thứ tự của các nhóm bit từ nhóm bit thứ 9 đến nhóm bit thứ 44 chứa các bit chẵn lẻ LDPC.

Cụ thể, bộ phận đan xen theo đơn vị nhóm có thể thay đổi thứ tự của các nhóm bit từ nhóm bit thứ 9 đến nhóm bit thứ 44 sao cho nhóm bit thứ 9 được đặt vào vị trí thứ 9, nhóm bit thứ 31 được đặt vào vị trí thứ 10, nhóm bit thứ 23 được đặt vào vị trí thứ 11, ..., nhóm bit thứ 28 được đặt vào vị trí thứ 42, nhóm bit thứ 39 được đặt vào vị trí thứ 43, nhóm bit thứ 42 được đặt vào vị trí thứ 44.

Như được mô tả dưới đây, vì các bộ phận đọc lỗ 217 và 318 thực hiện phương pháp đọc lỗ kể từ bit chẵn lẻ cuối cùng, cho nên các nhóm bit chẵn lẻ có thể được sắp xếp theo thứ tự ngược với mẫu đọc lỗ sau khi hoán vị chẵn lẻ. Điều này có nghĩa là, nhóm bit đầu tiên được đọc lỗ nằm ở vị trí là nhóm bit cuối cùng.

Ví dụ nêu trên mô tả rằng chỉ có các bit chẵn lẻ được đan xen, nhưng đó chỉ là một ví dụ. Điều này có nghĩa là, các bộ phận hoán vị chẵn lẻ 215 và 316 cũng có thể đan xen các bit thông tin LDPC. Trong trường hợp như vậy, các bộ phận hoán vị chẵn lẻ 215 và 316 có thể đan xen các bit thông tin LDPC theo từng đơn vị và xuất ra các bit thông tin LDPC có thứ tự giống như trước khi đan xen sao cho thứ tự của các

bit thông tin LDPC không thay đổi.

Các bộ phận lặp 216 và 317 có thể lặp lại ít nhất một số bit của từ mã LDPC đã được hoán vị chẵn lẻ ở vị trí nằm phía sau các bit thông tin LDPC, và xuất ra từ mã LDPC lặp lại, tức là các bit từ mã LDPC có các bit lặp, để cung cấp cho các bộ phận đọc lỗi 217 và 318. Trong đó, bộ phận lặp 317 cũng có thể xuất ra từ mã LDPC lặp lại để cung cấp cho bộ tạo bit chẵn lẻ bổ sung 319. Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 319 có thể sử dụng từ mã LDPC lặp lại này để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, các bộ phận lặp 216 và 317 có thể lặp lại một số lượng định trước của các bit chẵn lẻ LDPC ở phía sau các bit thông tin LDPC. Điều này có nghĩa là, các bộ phận lặp 216 và 317 có thể gán số lượng bit chẵn lẻ LDPC được lặp lại định trước vào phía sau các bit thông tin LDPC. Vì vậy, các bit chẵn lẻ LDPC được lặp lại nằm ở giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC trong từ mã LDPC.

Như vậy, vì số lượng bit định trước trong từ mã LDPC sau khi lặp lại có thể được lặp lại và được truyền thêm đến thiết bị thu tín hiệu 200, cho nên phương pháp nêu trên có thể được gọi là phương pháp lặp.

Thuật ngữ “gán” có nghĩa là đặt các bit lặp vào giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC sao cho các bit này được lặp lại.

Phương pháp lặp có thể được thực hiện chỉ đối với chế độ 1 cho thông tin báo hiệu cơ bản cho tầng L1 và chế độ 1 cho thông tin báo hiệu chi tiết cho tầng L1, và có thể không được thực hiện đối với các chế độ khác. Trong trường hợp như vậy, các bộ phận lặp 216 và 317 không thực hiện phương pháp lặp, và có thể xuất ra từ mã LDPC đã được hoán vị chẵn lẻ để cung cấp cho các bộ phận đọc lỗi 217 và 318.

Quy trình thực hiện phương pháp lặp sẽ được mô tả chi tiết dưới đây.

Các bộ phận lặp 216 và 317 có thể tính số lượng N_{repeat} bit được truyền thêm trong mỗi từ mã LDPC dựa vào biểu thức 29 dưới đây:

$$N_{\text{repeat}} = 2 \times \lfloor C \times N_{\text{outer}} \rfloor + D \quad (29)$$

Trong biểu thức 29 nêu trên, C là một số cố định và D có thể là số nguyên chẵn. Dựa vào biểu thức 29 nêu trên, cần phải hiểu rằng số lượng bit cần lặp lại có thể được tính bằng cách nhân C với N_{outer} cho trước và cộng với D .

Các thông số C và D cho phương pháp lặp có thể được chọn dựa vào bảng 11 dưới đây. Điều này có nghĩa là, các bộ phận lặp 216 và 317 có thể xác định các thông số C và D dựa vào chế độ tương ứng như được thể hiện trong bảng 11 dưới đây.

Bảng 11

	N_{outer}	K_{sig}	K_{ldpc}	C	D	$N_{\text{ldpc_parity}}$ ($= N_{\text{inner}} - K_{\text{ldpc}}$)	η_{MOD}
Thông tin báo hiệu cơ bản cho tầng L1 theo chế độ 1	368	200	3240	0	3672	12960	2
Thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 1	568 ~ 2520	400 ~ 2352	3240	61/16	-508	12960	2

Ngoài ra, các bộ phận lặp 216 và 317 có thể lặp lại N_{repeat} bit chẵn lẻ LDPC.

Cụ thể, khi $N_{\text{repeat}} \leq N_{\text{ldpc_parity}}$, các bộ phận lặp 216 và 317 có thể gán N_{repeat} bit đầu tiên trong số các bit chẵn lẻ LDPC đã được hoán vị chẵn lẻ vào các bit thông tin LDPC như được thể hiện trên Fig.47. Điều này có nghĩa là, các bộ phận lặp 216 và 317 có thể gán bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC đã được hoán vị chẵn lẻ để làm bit chẵn lẻ LDPC thứ N_{repeat} vào phía sau các bit thông tin LDPC.

Khi $N_{\text{repeat}} > N_{\text{ldpc_parity}}$, các bộ phận lặp 216 và 317 có thể gán $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã được hoán vị chẵn lẻ vào các bit thông tin LDPC như được thể hiện trên

Fig.48, và có thể gắn thêm số lượng $N_{\text{repeat}} - N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã được hoán vị chẵn lẻ vào $N_{\text{ldpc_parity}}$ bit chẵn lẻ LDPC đã gắn lần đầu. Điều này có nghĩa là, các bộ phận lặp 216 và 317 có thể gắn tất cả các bit chẵn lẻ LDPC đã được hoán vị chẵn lẻ vào phía sau các bit thông tin LDPC và gắn thêm bit chẵn lẻ LDPC đầu tiên vào để làm bit chẵn lẻ LDPC thứ $N_{\text{repeat}} - N_{\text{ldpc_parity}}$ trong số các bit chẵn lẻ LDPC đã được hoán vị chẵn lẻ vào phía sau các bit chẵn lẻ LDPC đã gắn lần đầu.

Vì vậy, ở chế độ 1 cho thông tin báo hiệu cơ bản cho tầng L1 và chế độ 1 cho thông tin báo hiệu chi tiết cho tầng L1, N_{repeat} bit gắn thêm có thể được chọn trong từ mã LDPC và được truyền.

Các bộ phận đọc lỗi 217 và 318 có thể đọc lỗi một số bit chẵn lẻ LDPC có trong từ mã LDPC được xuất ra từ các bộ phận lặp 216 và 317, và xuất ra từ mã LDPC đã được đọc lỗi (tức là các bit từ mã LDPC còn lại, ngoại trừ các bit được đọc lỗi và cũng được gọi là từ mã LDPC sau khi đọc lỗi) để cung cấp cho các bộ phận loại bỏ bit không 218 và 321. Ngoài ra, bộ phận đọc lỗi 318 có thể cung cấp thông tin (ví dụ số lượng và vị trí của các bit được đọc lỗi, v.v.) liên quan đến các bit chẵn lẻ LDPC được đọc lỗi cho bộ tạo bit chẵn lẻ bổ sung 319. Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 319 có thể tạo ra các bit chẵn lẻ bổ sung dựa vào các thông tin đó.

Như vậy, sau khi đi qua bộ phận hoán vị chẵn lẻ, một số bit chẵn lẻ LDPC có thể được đọc lỗi.

Trong trường hợp như vậy, các bit chẵn lẻ LDPC đã được đọc lỗi không được truyền trong khung mà các bit thông tin báo hiệu cho tầng L1 được truyền trong đó. Cụ thể, các bit chẵn lẻ LDPC đã được đọc lỗi không được truyền trong khung hiện thời mà các bit thông tin báo hiệu cho tầng L1 được truyền trong đó, và trong một số

trường hợp, các bit chẵn lẻ LDPC đã được đọc lỗi có thể được truyền trong khung đứng trước khung hiện thời, trường hợp này sẽ được mô tả liên quan đến bộ tạo bit chẵn lẻ bổ sung 319.

Nhằm mục đích này, các bộ phận đọc lỗi 217 và 318 có thể xác định số lượng bit chẵn lẻ LDPC cần đọc lỗi trong mỗi từ mã LDPC và kích thước của một khối mã hoá.

Cụ thể, các bộ phận đọc lỗi 217 và 318 có thể tính số lượng tạm thời $N_{\text{punc_temp}}$ bit chẵn lẻ LDPC cần đọc lỗi dựa vào biểu thức 30 dưới đây. Điều này có nghĩa là, với N_{outer} cho trước, các bộ phận đọc lỗi 217 và 318 có thể tính số lượng tạm thời $N_{\text{punc_temp}}$ bit chẵn lẻ LDPC cần đọc lỗi dựa vào biểu thức 30 dưới đây:

$$N_{\text{punc_temp}} = \lfloor A \times (K_{\text{ldpc}} - N_{\text{outer}}) \rfloor + B \quad (30)$$

Dựa vào biểu thức 30 nêu trên, độ dài tạm thời của các bit cần đọc lỗi có thể được tính bằng cách cộng số nguyên không đổi B với số nguyên thu được từ kết quả của phép nhân độ dài rút gọn (tức là $K_{\text{ldpc}} - N_{\text{outer}}$) với giá trị hằng số định trước A. Theo phương án làm ví dụ thực hiện sáng chế, rõ ràng là giá trị hằng số A được thiết lập theo tỷ số giữa số lượng bit được đọc lỗi và số lượng bit được rút gọn có thể được thiết lập khác nhau tùy thuộc vào các yêu cầu của hệ thống.

Theo sáng chế, giá trị B là giá trị biểu thị độ dài của các bit cần đọc lỗi ngay cả khi độ dài rút gọn bằng 0, và do đó, giá trị này biểu thị độ dài nhỏ nhất có thể có của các bit đọc lỗi. Ngoài ra, các giá trị A và B dùng để điều chỉnh tỷ lệ mã được truyền trên thực tế. Điều này có nghĩa là, để chuẩn bị cho trường hợp trong đó các bit thông tin, tức là các bit thông tin báo hiệu cho tầng L1 có độ dài nhỏ hoặc trường hợp trong đó các bit thông tin báo hiệu cho tầng L1 có độ dài lớn, thì các giá trị A và B dùng để điều chỉnh tỷ lệ mã được truyền trên thực tế cần được rút gọn.

Các giá trị K_{ldpc} , A và B nêu trên được thể hiện trong bảng 12 dưới đây, bảng này thể hiện các thông số để đọc lỗi. Vì vậy, các bộ phận đọc lỗi 217 và 318 có thể xác định các thông số để đọc lỗi theo chế độ tương ứng như được thể hiện trong bảng 12 dưới đây.

Bảng 12

Kiểu mã hoá FEC cho thông tin báo hiệu		N _{outer}	K _{ldpc}	A	B	N _{ldpc_parity}	η _{MOD}
Thông tin báo hiệu cơ bản cho tầng L1	Chế độ 1	368	3240	0	9360	12960	2
	Chế độ 2				11460		2
	Chế độ 3				12360		2
	Chế độ 4				12292		4
	Chế độ 5				12350		6
	Chế độ 6				12432		8
	Chế độ 7				12776		8
Thông tin báo hiệu chi tiết cho tầng L1	Chế độ 1	568 ~ 2520	6480	7/2	0	9720	2
	Chế độ 2	568 ~ 3240		2	6036		2
	Chế độ 3	568 ~ 6480		11/16	4653		2
	Chế độ 4			29/32	3200		4
	Chế độ 5			3/4	4284		6
	Chế độ 6			11/16	4900		8
	Chế độ 7			49/256	8246		8

Các bộ phận đọc lỗi 217 và 318 có thể tính kích thước tạm thời N_{FEC_temp} của một khối mã hoá như được biểu diễn bằng biểu thức 31 dưới đây. Theo sáng chế, số lượng N_{ldpc_parity} bit chẵn lẻ LDPC theo chế độ tương ứng được xác định như ở trong bảng 12 nêu trên.

$$N_{FEC_temp} = N_{outer} + N_{ldpc_parity} - N_{punc_temp} \quad (31)$$

Ngoài ra, các bộ phận đọc lỗi 217 và 318 có thể tính kích thước N_{FEC} của một khối mã hoá như được biểu diễn bằng biểu thức 32 dưới đây:

$$N_{FEC} = \left\lceil \frac{N_{FEC_temp}}{\eta_{MOD}} \right\rceil \times \eta_{MOD} \quad (32)$$

Trong biểu thức 32 nêu trên, η_{MOD} là bậc điều biến. Ví dụ, khi thông tin báo hiệu

cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được điều biến theo sơ đồ điều biến QPSK, 16-QAM, 64-QAM hoặc 256-QAM theo chế độ tương ứng, thì η_{MOD} có thể bằng 2, 4, 6 và 8 như được thể hiện trong bảng 12 nêu trên. Theo biểu thức 32 nêu trên, N_{FEC} có thể là một bội số nguyên lần của bậc điều biến.

Ngoài ra, các bộ phận đọc lỗi 217 và 318 có thể tính số lượng N_{punc} bit chẵn lẻ LDPC cần đọc lỗi dựa vào biểu thức 33 dưới đây:

$$N_{\text{punc}} = N_{\text{punc_temp}} - (N_{\text{FEC}} - N_{\text{FEC_temp}}) \quad (33)$$

Trong đó, N_{punc} bằng 0 hoặc là số nguyên dương. Ngoài ra, N_{FEC} là số lượng bit của khối thông tin thu được bằng cách lấy $N_{\text{outer}} + N_{\text{ldpc_parity}}$ bit thu được sau khi thực hiện bước mã hoá BCH và mã hoá LDPC trên K_{sig} bit thông tin trừ đi N_{punc} bit cần đọc lỗi. Điều này có nghĩa là, N_{FEC} là số lượng bit ngoại trừ các bit lặp trong số các bit được truyền trên thực tế, và có thể được gọi là các bit từ mã LDPC đã được rút gọn và đọc lỗi.

Dựa vào quy trình nêu trên, các bộ phận đọc lỗi 217 và 318 nhân A với số lượng bit không được đệm, tức là độ dài rút gọn, và cộng B với kết quả thu được để tính số lượng tạm thời $N_{\text{punc_temp}}$ bit chẵn lẻ LDPC cần đọc lỗi.

Ngoài ra, các bộ phận đọc lỗi 217 và 318 tính số lượng tạm thời $N_{\text{FEC_temp}}$ bit từ mã LDPC để tạo nên từ mã LDPC sau khi đọc lỗi và rút gọn dựa vào $N_{\text{punc_temp}}$.

Cụ thể, các bit thông tin LDPC được mã hoá LDPC, và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được gắn vào các bit thông tin LDPC để tạo nên từ mã LDPC. Theo sáng chế, các bit thông tin LDPC có các bit mã hoá BCH, trong đó thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được mã hoá BCH, và trong một số trường hợp, có thể còn có các bit không được

đệm vào.

Trong trường hợp như vậy, vì các bit không được đệm vào sẽ được mã hoá LDPC, và sau đó, không được truyền đến thiết bị thu tín hiệu 200, cho nên từ mã LDPC rút gọn, đó là từ mã LDPC (tức là từ mã LDPC được rút gọn) loại trừ các bit không được đệm vào, có thể gồm có các bit mã hoá BCH và các bit chẵn lẻ LDPC.

Vì vậy, các bộ phận đọc lỗi 217 và 318 lấy tổng số của số lượng bit mã hoá BCH và số lượng bit chẵn lẻ LDPC trừ đi số lượng tạm thời của các bit chẵn lẻ LDPC cần đọc lỗi để tính N_{FEC_temp} .

Từ mã LDPC đã được rút gọn và đọc lỗi (tức là các bit từ mã LDPC còn lại sau khi đọc lỗi và rút gọn) được ánh xạ lên các ký hiệu chòm điểm theo các sơ đồ điều biến khác nhau như QPSK, 16-QAM, 64-QAM hoặc 256-QAM theo chế độ tương ứng, và các ký hiệu chòm điểm có thể được truyền đến thiết bị thu tín hiệu 200 trong một khung.

Vì vậy, các bộ phận đọc lỗi 217 và 318 xác định số lượng N_{FEC} bit từ mã LDPC để tạo nên từ mã LDPC sau khi đọc lỗi và rút gọn dựa vào N_{FEC_temp} , N_{FEC} là một bội số nguyên lần của bậc điều biến, và xác định số lượng N_{punc} bit cần đọc lỗi dựa vào các bit từ mã LDPC sau khi rút gọn để thu được N_{FEC} .

Khi các bit không không được đệm vào, thì từ mã LDPC có thể gồm có các bit mã hoá BCH và các bit chẵn lẻ LDPC, và bước rút gọn có thể không được thực hiện.

Ngoài ra, ở chế độ 1 cho thông tin báo hiệu cơ bản cho tầng L1 và chế độ 1 cho thông tin báo hiệu chi tiết cho tầng L1, phương pháp lặp được thực hiện, và do đó, số lượng bit từ mã LDPC đã được rút gọn và đọc lỗi bằng $N_{FEC} + N_{repeat}$.

Các bộ phận đọc lỗi 217 và 318 có thể đọc lỗi các bit chẵn lẻ LDPC với số lượng

đúng bằng số lượng đã tính.

Trong trường hợp như vậy, các bộ phận đọc lỗ 217 và 318 có thể đọc lỗ N_{punc} bit cuối cùng của tất cả các từ mã LDPC. Điều này có nghĩa là, các bộ phận đọc lỗ 217 và 318 có thể đọc lỗ N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng.

Cụ thể, khi phương pháp lặp không được thực hiện, thì từ mã LDPC đã được hoán vị chẵn lẻ chỉ có các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, các bộ phận đọc lỗ 217 và 318 có thể đọc lỗ N_{punc} bit cuối cùng của tất cả các từ mã LDPC đã được hoán vị chẵn lẻ. Vì vậy, N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC có thể được đọc lỗ.

Khi phương pháp lặp được thực hiện, thì từ mã LDPC đã được hoán vị chẵn lẻ và lặp lại có các bit chẵn lẻ LDPC được lặp lại và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, các bộ phận đọc lỗ 217 và 318 có thể lần lượt đọc lỗ N_{punc} bit cuối cùng của tất cả các từ mã LDPC đã được hoán vị chẵn lẻ và lặp lại, như được thể hiện trên Fig.49 và Fig.50.

Cụ thể, các bit chẵn lẻ LDPC được lặp lại nằm ở giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, và do đó, các bộ phận đọc lỗ 217 và 318 có thể lần lượt đọc lỗ N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Như vậy, các bộ phận đọc lỗ 217 và 318 có thể lần lượt đọc lỗ N_{punc} bit kể từ bit chẵn lẻ LDPC cuối cùng.

N_{punc} bằng 0 hoặc là số nguyên dương và phương pháp lặp có thể được áp dụng

chỉ đối với chế độ 1 cho thông tin báo hiệu cơ bản cho tầng L1 và chế độ 1 cho thông tin báo hiệu chi tiết cho tầng L1.

Ví dụ nêu trên mô tả rằng phương pháp lập được thực hiện, và sau đó, phương pháp đọc lỗi được thực hiện, nhưng đó chỉ là một ví dụ. Trong một số trường hợp, sau khi phương pháp đọc lỗi được thực hiện, phương pháp lập có thể được thực hiện.

Bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn các bit trong số các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung (Additional Parity, AP).

Trong trường hợp như vậy, các bit chẵn lẻ bổ sung có thể được chọn trong số các bit chẵn lẻ LDPC được tạo ra dựa vào thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung hiện thời, và được truyền đến thiết bị thu tín hiệu 200 trong một khung đứng trước khung hiện thời, tức là khung trước đó.

Cụ thể, thông tin báo hiệu chi tiết cho tầng L1 được mã hoá LDPC, và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được gắn vào thông tin báo hiệu chi tiết cho tầng L1 để tạo nên từ mã LDPC.

Ngoài ra, quy trình đọc lỗi và rút gọn được thực hiện trên từ mã LDPC, và từ mã LDPC đã được rút gọn và đọc lỗi có thể được ánh xạ lên một khung sẽ được truyền đến thiết bị thu tín hiệu 200. Theo sáng chế, khi phương pháp lập được thực hiện theo chế độ tương ứng, thì từ mã LDPC đã được rút gọn và đọc lỗi có thể có các bit chẵn lẻ LDPC được lập lại.

Trong trường hợp như vậy, thông tin báo hiệu chi tiết cho tầng L1 tương ứng với mỗi khung có thể được truyền đến thiết bị thu tín hiệu 200 trong mỗi khung, cùng với các bit chẵn lẻ LDPC. Ví dụ, từ mã LDPC đã được rút gọn và đọc lỗi chứa thông tin báo hiệu chi tiết cho tầng L1 tương ứng với khung thứ $(i-1)$ có thể được ánh xạ lên

khung thứ $(i-1)$ sẽ được truyền đến thiết bị thu tín hiệu 200, và từ mã LDPC đã được rút gọn và đọc lỗi chứa thông tin báo hiệu chi tiết cho tầng L1 tương ứng với khung thứ i có thể được ánh xạ lên khung thứ i sẽ được truyền đến thiết bị thu tín hiệu 200.

Bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC được tạo ra dựa vào thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung thứ i để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, một số bit chẵn lẻ LDPC được tạo ra bằng cách thực hiện bước mã hoá LDPC trên thông tin báo hiệu chi tiết cho tầng L1 được đọc lỗi, và sau đó, không được truyền đến thiết bị thu tín hiệu 200. Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn ít nhất một số bit chẵn lẻ LDPC được đọc lỗi trong số các bit chẵn lẻ LDPC được tạo ra bằng cách thực hiện bước mã hoá LDPC trên thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung thứ i , từ đó tạo ra các bit chẵn lẻ bổ sung.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC sẽ được truyền đến thiết bị thu tín hiệu 200 trong khung thứ i để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, các bit chẵn lẻ LDPC có trong từ mã LDPC đã được rút gọn và đọc lỗi sẽ được ánh xạ lên khung thứ i có thể được tạo nên chỉ có các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC theo chế độ tương ứng hoặc các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC và các bit chẵn lẻ LDPC được lặp lại.

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn ít nhất một số bit trong số các bit chẵn lẻ LDPC có trong từ mã LDPC đã được rút gọn và đọc lỗi sẽ được ánh xạ lên khung thứ i để tạo ra các bit chẵn lẻ bổ sung.

Các bit chẵn lẻ bổ sung có thể được truyền đến thiết bị thu tín hiệu 200 trong khung đứng trước khung thứ i , tức là khung thứ $(i-1)$.

Điều này có nghĩa là, thiết bị truyền tín hiệu 100 có thể không chỉ truyền từ mã LDPC đã được rút gọn và đọc lỗi chứa thông tin báo hiệu chi tiết cho tầng L1 tương ứng với khung thứ $(i-1)$ mà còn truyền các bit chẵn lẻ bổ sung được tạo ra dựa vào thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung thứ i đến thiết bị thu tín hiệu 200 trong khung thứ $(i-1)$.

Trong trường hợp như vậy, khung trong đó các bit chẵn lẻ bổ sung được truyền có thể là khung trước nhất tạm thời trong số các khung đứng trước khung hiện thời.

Ví dụ, các bit chẵn lẻ bổ sung có phiên bản lớn/nhỏ của phần khởi động giống như khung hiện thời trong số các khung đứng trước khung hiện thời, và có thể được truyền trong khung trước nhất tạm thời.

Trong một số trường hợp, bộ tạo bit chẵn lẻ bổ sung 319 có thể không tạo ra các bit chẵn lẻ bổ sung.

Trong trường hợp như vậy, thiết bị truyền tín hiệu 100 có thể truyền thông tin về việc các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 trong khung kế tiếp được truyền trong khung hiện thời đến thiết bị thu tín hiệu 200 sử dụng thông tin báo hiệu cơ bản cho tầng L1 được truyền trong khung hiện thời.

Ví dụ, việc sử dụng các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 trong khung kế tiếp có phiên bản lớn/nhỏ của phần khởi động giống như khung hiện thời có thể được báo hiệu thông qua trường `L1B_L1_Detail_additional_parity_mode` trong thông tin báo hiệu cơ bản cho tầng L1 của khung hiện thời. Cụ thể, khi trường `L1B_L1_Detail_additional_parity_mode`

trong thông tin báo hiệu cơ bản cho tầng L1 của khung hiện thời được đặt bằng '00', thì các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 trong khung kế tiếp không được truyền trong khung hiện thời.

Như vậy, để tăng thêm khả năng của thông tin báo hiệu chi tiết cho tầng L1, các bit chẵn lẻ bổ sung có thể được truyền trong khung đứng trước khung hiện thời mà thông tin báo hiệu chi tiết cho tầng L1 trong khung hiện thời được truyền ở trong đó.

Fig.51 thể hiện ví dụ trong đó các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 trong khung thứ i được truyền trong phần mở đầu của khung thứ $(i-1)$.

Fig.51 thể hiện rằng thông tin báo hiệu chi tiết cho tầng L1 truyền trong khung thứ i được phân đoạn ra thành M khối bằng cách phân đoạn và mỗi khối đã phân đoạn được mã hoá FEC.

Vì vậy, số lượng M từ mã LDPC, tức là từ mã LDPC có các bit thông tin LDPC $L1-D(i)_1$ và các bit chẵn lẻ cho từ mã $L1-D(i)_1, \dots$, và từ mã LDPC có các bit thông tin LDPC $L1-D(i)_M$ và các bit chẵn lẻ cho từ mã $L1-D(i)_M$, được ánh xạ lên khung thứ i sẽ được truyền đến thiết bị thu tín hiệu 200.

Trong trường hợp như vậy, các bit chẵn lẻ bổ sung được tạo ra dựa vào thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung thứ i có thể được truyền đến thiết bị thu tín hiệu 200 trong khung thứ $(i-1)$.

Cụ thể, các bit chẵn lẻ bổ sung, tức là các bit AP cho từ mã $L1-D(i)_1, \dots$, các bit AP cho từ mã $L1-D(i)_M$, được tạo ra dựa vào thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung thứ i có thể được ánh xạ vào trong phần mở đầu của khung thứ $(i-1)$ sẽ được truyền đến thiết bị thu tín hiệu 200. Nhờ sử dụng các bit

chẵn lẻ bổ sung, có thể đạt được độ tăng ích do phân tập cho thông tin báo hiệu cho tầng L1.

Phương pháp tạo ra bit chẵn lẻ bổ sung sẽ được mô tả dưới đây.

Bộ tạo bit chẵn lẻ bổ sung 319 tính số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung dựa vào biểu thức 34 dưới đây:

$$N_{AP_temp} = \min \left\{ \begin{array}{l} 0,5 \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat}) \\ (N_{ldpc_parity} + N_{punc} + N_{repeat}) \end{array} \right\}, K = 0, 1, 2 \quad (34)$$

$$\text{Trong biểu thức 34 nêu trên, } \min(a, b) = \begin{cases} a, & \text{khi } a \leq b \\ b, & \text{khi } b < a \end{cases}.$$

Ngoài ra, K là tỷ số giữa số lượng bit chẵn lẻ bổ sung và tổng số bit của khối thông tin báo hiệu chi tiết cho tầng L1 được mã hoá và được truyền (tức là các bit tạo nên khối thông tin báo hiệu chi tiết cho tầng L1 được lặp lại, đực lỗ, và đã loại bỏ các bit không (tức là đã được rút gọn)).

Trong trường hợp như vậy, K tương ứng với trường L1B_L1_Detail_additional_parity_mode trong thông tin báo hiệu cơ bản cho tầng L1. Trong đó, giá trị của trường L1B_L1_Detail_additional_parity_mode liên quan đến thông tin báo hiệu chi tiết cho tầng L1 của khung thứ i (tức là khung (#i)) có thể được truyền trong khung thứ (i-1) (tức là khung (#i-1)).

Như đã nêu trên, đối với thông tin báo hiệu chi tiết cho tầng L1 theo các chế độ 2, 3, 4, 5, 6 và 7, vì phương pháp lặp không được thực hiện, cho nên trong biểu thức 34 nêu trên, N_{repeat} bằng 0.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 319 tính số lượng N_{AP} bit chẵn lẻ bổ sung dựa vào biểu thức 35 dưới đây. Vì vậy, số lượng N_{AP} bit chẵn lẻ bổ sung có thể là bội số của bậc điều biến.

$$N_{AP} = \left\lfloor \frac{N_{AP_temp}}{\eta_{MOD}} \right\rfloor \times \eta_{MOD} \quad (35)$$

Trong đó, $\lfloor x \rfloor$ là số nguyên lớn nhất không lớn hơn x . Trong đó, η_{MOD} là bậc điều biến. Ví dụ, khi thông tin báo hiệu chi tiết cho tầng L1 được điều biến theo sơ đồ điều biến QPSK, 16-QAM, 64-QAM hoặc 256-QAM theo chế độ tương ứng, thì η_{MOD} có thể lần lượt bằng 2, 4, 6 hoặc 8.

Như vậy, số lượng bit chẵn lẻ bổ sung được tạo ra có thể được xác định dựa vào tổng số bit được truyền trong khung hiện thời.

Tiếp theo, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn các bit với số lượng đúng bằng số lượng bit đã tính trong các bit chẵn lẻ LDPC để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi số lượng bit chẵn lẻ LDPC được đọc lỗi bằng hoặc lớn hơn số lượng bit chẵn lẻ bổ sung được tạo ra, thì bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn các bit với số lượng đúng bằng số lượng đã tính kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Khi số lượng bit chẵn lẻ LDPC được đọc lỗi nhỏ hơn số lượng bit chẵn lẻ bổ sung được tạo ra, thì bộ tạo bit chẵn lẻ bổ sung 319 trước tiên có thể chọn tất cả các bit chẵn lẻ LDPC được đọc lỗi, và sau đó chọn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chẵn lẻ bổ sung đã tính trừ đi số lượng bit chẵn lẻ LDPC được đọc lỗi, kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC có trong từ mã LDPC, để tạo ra các bit chẵn lẻ bổ sung.

Cụ thể, khi phương pháp lặp không được thực hiện, thì các bit chẵn lẻ LDPC có trong từ mã LDPC lặp lại là các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, bộ tạo bit chắn lẻ bổ sung 319 trước tiên có thể chọn tất cả các bit chắn lẻ LDPC được đọc lỗi và sau đó chọn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chắn lẻ bổ sung trừ đi số lượng bit chắn lẻ LDPC được đọc lỗi, kể từ bit chắn lẻ LDPC đầu tiên trong số các bit chắn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, để tạo ra các bit chắn lẻ bổ sung.

Theo sáng chế, các bit chắn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được phân chia ra thành các bit chắn lẻ LDPC không được đọc lỗi và các bit chắn lẻ LDPC được đọc lỗi. Nhờ đó, khi các bit được chọn kể từ bit đầu tiên trong số các bit chắn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, các bit đó có thể được chọn theo thứ tự lần lượt là các bit chắn lẻ LDPC không được đọc lỗi và các bit chắn lẻ LDPC được đọc lỗi.

Khi phương pháp lập được thực hiện, thì các bit chắn lẻ LDPC có trong từ mã LDPC lặp lại là các bit chắn lẻ LDPC được lặp lại và các bit chắn lẻ LDPC được tạo ra bằng cách mã hoá. Theo sáng chế, các bit chắn lẻ LDPC được lặp lại nằm ở giữa các bit thông tin LDPC và các bit chắn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Trong trường hợp như vậy, bộ tạo bit chắn lẻ bổ sung 319 trước tiên có thể chọn tất cả các bit chắn lẻ LDPC được đọc lỗi và sau đó chọn thêm các bit với số lượng đúng bằng số lượng thu được sau khi lấy số lượng bit chắn lẻ bổ sung trừ đi số lượng bit chắn lẻ LDPC được đọc lỗi, kể từ bit chắn lẻ LDPC đầu tiên trong số các bit chắn lẻ LDPC được lặp lại, để tạo ra các bit chắn lẻ bổ sung.

Theo sáng chế, khi các bit được chọn kể từ bit đầu tiên trong số các bit chắn lẻ LDPC được lặp lại, các bit đó có thể được chọn theo thứ tự lần lượt là các bit lặp và

các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Ngoài ra, các bit đó có thể được chọn theo thứ tự lần lượt là các bit chẵn lẻ LDPC không được đục lỗ và các bit chẵn lẻ LDPC được đục lỗ, trong số các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Phương pháp tạo ra bit chẵn lẻ bổ sung theo các phương án làm ví dụ thực hiện sáng chế sẽ được mô tả chi tiết dưới đây dựa vào các hình vẽ từ Fig.52 đến Fig.54.

Fig.52 đến Fig.54 là các sơ đồ thể hiện phương pháp tạo ra bit chẵn lẻ bổ sung khi phương pháp lặp được thực hiện, theo các phương án làm ví dụ thực hiện sáng chế. Trong trường hợp như vậy, từ mã LDPC lặp lại $V = (v_0, v_1, \dots, v_{N_{\text{inner}} + N_{\text{repeat}} - 1})$ có thể được biểu diễn như được thể hiện trên Fig.52.

Trước hết, khi $N_{\text{AP}} \leq N_{\text{punc}}$, như được thể hiện trên Fig.53, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn N_{AP} bit kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đục lỗ để tạo ra các bit chẵn lẻ bổ sung.

Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit chẵn lẻ LDPC được đục lỗ $(v_{N_{\text{repeat}} + N_{\text{inner}} - N_{\text{punc}}}, v_{N_{\text{repeat}} + N_{\text{inner}} - N_{\text{punc}} + 1}, \dots, v_{N_{\text{repeat}} + N_{\text{inner}} - N_{\text{punc}} - N_{\text{AP}} - 1})$. Điều này có nghĩa là, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn N_{AP} bit kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đục lỗ để tạo ra các bit chẵn lẻ bổ sung.

Khi $N_{\text{AP}} > N_{\text{punc}}$, như được thể hiện trên Fig.54, bộ tạo bit chẵn lẻ bổ sung 319 chọn tất cả các bit chẵn lẻ LDPC được đục lỗ.

Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn tất cả các bit chẵn lẻ LDPC được đục lỗ $(v_{N_{\text{repeat}} + N_{\text{inner}} - N_{\text{punc}}}, v_{N_{\text{repeat}} + N_{\text{inner}} - N_{\text{punc}} + 1}, \dots, v_{N_{\text{repeat}} + N_{\text{inner}} - 1})$.

Ngoài ra, bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn thêm $N_{\text{AP}} - N_{\text{punc}}$ bit đầu tiên trong số các bit chẵn lẻ LDPC gồm có các bit chẵn lẻ LDPC được lặp lại và các

bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC.

Điều này có nghĩa là, vì các bit chẵn lẻ LDPC được lặp lại và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC được sắp xếp tuần tự, cho nên bộ tạo bit chẵn lẻ bổ sung 319 có thể chọn thêm $N_{AP} - N_{punc}$ bit chẵn lẻ kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được lặp lại.

Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn thêm các bit chẵn lẻ LDPC ($v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1}$).

Trong trường hợp như vậy, bộ tạo bit chẵn lẻ bổ sung 319 có thể gán các bit đã chọn thêm vào các bit đã chọn trước đó để tạo ra các bit chẵn lẻ bổ sung. Điều này có nghĩa là, như được thể hiện trên Fig.54, bộ tạo bit chẵn lẻ bổ sung 319 có thể gán các bit chẵn lẻ LDPC đã chọn thêm vào các bit chẵn lẻ LDPC được đọc lỗi để tạo ra các bit chẵn lẻ bổ sung.

Vì vậy, để dùng làm các bit chẵn lẻ bổ sung, có thể chọn các bit ($v_{N_{repeat}+N_{inner}-N_{punc}}, v_{N_{repeat}+N_{inner}-N_{punc}+1}, \dots, v_{N_{repeat}+N_{inner}-1}, v_{K_{ldpc}}, v_{K_{ldpc}+1}, \dots, v_{K_{ldpc}+N_{AP}-N_{punc}-1}$).

Như vậy, khi số lượng bit được đọc lỗi bằng hoặc lớn hơn số lượng bit chẵn lẻ bổ sung, thì các bit chẵn lẻ bổ sung có thể được tạo ra bằng cách chọn các bit trong số các bit được đọc lỗi dựa vào thứ tự đọc lỗi. Mặt khác, trong những trường hợp khác, các bit chẵn lẻ bổ sung có thể được tạo ra bằng cách chọn tất cả các bit được đọc lỗi và $N_{AP} - N_{punc}$ bit chẵn lẻ.

Vì $N_{repeat} = 0$ khi phương pháp lặp không được thực hiện, cho nên phương pháp tạo ra bit chẵn lẻ bổ sung khi phương pháp lặp không được thực hiện giống như trong trường hợp $N_{repeat} = 0$ được thể hiện trên các hình vẽ từ Fig.52 đến Fig.54.

Các bit chẵn lẻ bổ sung có thể được đan xen bit, và có thể được ánh xạ lên chòm

điểm. Trong trường hợp như vậy, chòm điểm cho các bit chẵn lẻ bổ sung có thể được tạo ra bằng phương pháp giống như chòm điểm cho các bit thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung hiện thời, trong đó các bit thông tin báo hiệu chi tiết cho tầng L1 được lặp lại, đọc lỗi và đã loại bỏ các bit không. Ngoài ra, như được thể hiện trên Fig.51, sau khi được ánh xạ lên chòm điểm, các bit chẵn lẻ bổ sung có thể được gắn vào phía sau khối thông tin báo hiệu chi tiết cho tầng L1 trong khung đứng trước khung hiện thời mà thông tin báo hiệu chi tiết cho tầng L1 trong khung hiện thời được truyền ở trong đó.

Bộ tạo bit chẵn lẻ bổ sung 319 có thể xuất ra các bit chẵn lẻ bổ sung để cung cấp cho bộ phân kênh bit 323.

Như đã được mô tả trên đây dựa vào bảng 9 và bảng 10, mẫu đan xen theo đơn vị nhóm xác định thứ tự hoán vị có thể có hai mẫu: mẫu thứ nhất và mẫu thứ hai.

Cụ thể, vì giá trị B trong biểu thức 30 nêu trên biểu thị độ dài nhỏ nhất của các bit chẵn lẻ LDPC cần đọc lỗi, cho nên một số lượng bit định trước có thể luôn luôn được đọc lỗi phụ thuộc vào giá trị B, bất kể độ dài của thông tin báo hiệu đầu vào. Ví dụ, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, vì $B = 6036$ và nhóm bit gồm có 360 bit, cho nên ngay cả khi độ dài rút gọn bằng 0, vẫn có ít nhất $\left\lfloor \frac{6036}{360} \right\rfloor = 16$ nhóm bit luôn luôn được đọc lỗi.

Trong trường hợp như vậy, vì phương pháp đọc lỗi được thực hiện kể từ bit chẵn lẻ LDPC cuối cùng, cho nên số lượng nhóm bit định trước kể từ nhóm bit cuối cùng trong số các nhóm bit tạo nên các bit chẵn lẻ LDPC được đan xen theo đơn vị nhóm có thể luôn luôn được đọc lỗi bất kể độ dài rút gọn.

Ví dụ, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, 16 nhóm bit

cuối cùng trong số 36 nhóm bit tạo nên các bit chẵn lẻ LDPC được đan xen theo đơn vị nhóm có thể luôn luôn được đọc lỗi.

Do đó, một số mẫu đan xen theo đơn vị nhóm để xác định thứ tự hoán vị có các nhóm bit luôn luôn được đọc lỗi, và vì vậy, mẫu đan xen theo đơn vị nhóm có thể được phân chia ra thành hai mẫu. Cụ thể, một mẫu xác định các nhóm bit còn lại, ngoại trừ các nhóm bit luôn luôn được đọc lỗi, trong mẫu đan xen theo đơn vị nhóm được gọi là mẫu thứ nhất, và một mẫu xác định các nhóm bit luôn luôn được đọc lỗi được gọi là mẫu thứ hai.

Ví dụ, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, vì mẫu đan xen theo đơn vị nhóm được xác định như ở trong bảng 9 nêu trên, nên mẫu biểu thị chỉ số của các nhóm bit không được đan xen theo đơn vị nhóm và nằm ở trong các nhóm bit từ nhóm bit thứ 9 đến nhóm bit thứ 28 sau khi đan xen theo đơn vị nhóm, tức là $Y_9 = X_{\pi p(9)} = X_9$, $Y_{10} = X_{\pi p(10)} = X_{31}$, $Y_{11} = X_{\pi p(11)} = X_{23}$, ..., $Y_{26} = X_{\pi p(26)} = X_{17}$, $Y_{27} = X_{\pi p(27)} = X_{35}$, $Y_{28} = X_{\pi p(28)} = X_{21}$ có thể là mẫu thứ nhất, và mẫu biểu thị chỉ số của các nhóm bit không được đan xen theo đơn vị nhóm và nằm ở trong các nhóm bit từ nhóm bit thứ 29 đến nhóm bit thứ 44 sau khi đan xen theo đơn vị nhóm, tức là $Y_{29} = X_{\pi p(29)} = X_{20}$, $Y_{30} = X_{\pi p(30)} = X_{24}$, $Y_{31} = X_{\pi p(31)} = X_{44}$, ..., $Y_{42} = X_{\pi p(42)} = X_{28}$, $Y_{43} = X_{\pi p(43)} = X_{39}$, $Y_{44} = X_{\pi p(44)} = X_{42}$ có thể là mẫu thứ hai.

Như đã nêu trên, mẫu thứ hai xác định các nhóm bit luôn luôn được đọc lỗi trong khung hiện thời bất kể độ dài rút gọn, và mẫu thứ nhất xác định các nhóm bit được đọc lỗi thêm khi độ dài rút gọn lớn, cho nên mẫu thứ nhất có thể được sử dụng để xác định các bit chẵn lẻ LDPC được truyền trong khung hiện thời sau khi đọc lỗi.

Cụ thể, theo số lượng bit chẵn lẻ LDPC cần đọc lỗi, ngoài các bit chẵn lẻ LDPC

luôn luôn được đọc lỗi, còn có các bit chẵn lẻ LDPC nữa có thể được đọc lỗi thêm.

Ví dụ, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, khi số lượng bit chẵn lẻ LDPC cần đọc lỗi bằng 7200, 20 nhóm bit cần đọc lỗi, và do đó, có bốn (4) nhóm bit cần đọc lỗi thêm, ngoài 16 nhóm bit luôn luôn được đọc lỗi.

Trong trường hợp như vậy, bốn (4) nhóm bit được đọc lỗi thêm tương ứng với các nhóm bit nằm ở các vị trí từ vị trí thứ 25 đến vị trí thứ 28 sau khi đan xen theo đơn vị nhóm, và vì các nhóm bit này được xác định theo mẫu thứ nhất, tức là thuộc về mẫu thứ nhất, cho nên mẫu thứ nhất có thể được sử dụng để xác định các nhóm bit được đọc lỗi.

Điều này có nghĩa là, khi các bit chẵn lẻ LDPC được đọc lỗi nhiều hơn giá trị nhỏ nhất của số lượng bit chẵn lẻ LDPC cần đọc lỗi, thì các nhóm bit cần đọc lỗi thêm được xác định là các nhóm bit nằm ở phía sau các nhóm bit luôn luôn được đọc lỗi. Vì vậy, theo chiều đọc lỗi, mẫu thứ nhất xác định các nhóm bit nằm ở phía sau các nhóm bit luôn luôn được đọc lỗi có thể được coi là mẫu xác định các nhóm bit được đọc lỗi.

Điều này có nghĩa là, như trong ví dụ nêu trên, khi số lượng bit chẵn lẻ LDPC cần đọc lỗi bằng 7200, ngoài 16 nhóm bit luôn luôn được đọc lỗi, có bốn (4) nhóm bit, tức là các nhóm bit nằm ở vị trí thứ 28, vị trí thứ 27, vị trí thứ 26 và vị trí thứ 25, sau khi thực hiện bước đan xen theo đơn vị nhóm, được đọc lỗi thêm. Theo sáng chế, các nhóm bit nằm ở các vị trí từ vị trí thứ 25 đến vị trí thứ 28 sau khi đan xen theo đơn vị nhóm được xác định theo mẫu thứ nhất.

Do đó, mẫu thứ nhất có thể được coi là dùng để xác định các nhóm bit cần đọc lỗi. Ngoài ra, các bit chẵn lẻ LDPC còn lại, ngoại trừ các bit chẵn lẻ LDPC được đọc lỗi được truyền trong khung hiện thời, và vì vậy, mẫu thứ nhất có thể được coi là dùng

để xác định các nhóm bit được truyền trong khung hiện thời.

Mẫu thứ hai có thể được dùng chỉ để xác định các bit chẵn lẻ bổ sung được truyền trong khung trước đó.

Cụ thể, vì các nhóm bit được xác định là luôn luôn được đọc lỗi sẽ luôn luôn được đọc lỗi, và sau đó, sẽ không được truyền trong khung hiện thời, cho nên các nhóm bit này chỉ cần được đặt vào vị trí mà ở đó đặt các bit luôn luôn được đọc lỗi sau khi đan xen theo đơn vị nhóm. Vì vậy, không cần quan tâm đến vị trí mà ở đó đặt các nhóm bit này sau khi đan xen theo đơn vị nhóm.

Ví dụ, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, các nhóm bit được đặt ở vị trí thứ 20, vị trí thứ 24, vị trí thứ 44, ..., vị trí thứ 28, vị trí thứ 39 và vị trí thứ 42 trước khi đan xen theo đơn vị nhóm chỉ cần được đặt vào các vị trí từ vị trí thứ 29 đến vị trí thứ 44 sau khi đan xen theo đơn vị nhóm. Vì vậy, không cần quan tâm đến vị trí mà ở đó đặt các nhóm bit này.

Như vậy, mẫu thứ hai xác định các nhóm bit luôn luôn được đọc lỗi được dùng chỉ để xác định các nhóm bit cần đọc lỗi. Vì vậy, việc xác định thứ tự của các nhóm bit trong mẫu thứ hai không có ý nghĩa đối với phương pháp đọc lỗi, và do đó, mẫu thứ hai xác định các nhóm bit luôn luôn được đọc lỗi có thể được coi là không được dùng cho phương pháp đọc lỗi.

Tuy nhiên, để xác định các bit chẵn lẻ bổ sung, thì vị trí của các nhóm bit luôn luôn được đọc lỗi trong các nhóm bit này sẽ được xem xét.

Cụ thể, vì các bit chẵn lẻ bổ sung được tạo ra bằng cách chọn các bit với số lượng đúng bằng số lượng định trước kể từ bit đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi, cho nên có thể chọn các bit có trong ít nhất một số nhóm bit luôn luôn

được đọc lỗi, để dùng làm ít nhất một số bit trong số các bit chẵn lẻ bổ sung, tùy thuộc vào số lượng bit chẵn lẻ LDPC được đọc lỗi và số lượng bit chẵn lẻ bổ sung.

Điều này có nghĩa là, khi các bit chẵn lẻ bổ sung được chọn nhiều hơn nhóm bit được xác định theo mẫu thứ nhất, vì các bit chẵn lẻ bổ sung được chọn tuần tự kể từ phần đầu của mẫu thứ hai, cho nên thứ tự của các nhóm bit thuộc mẫu thứ hai có ý nghĩa đối với việc chọn các bit chẵn lẻ bổ sung. Do đó, mẫu thứ hai xác định nhóm bit luôn luôn được đọc lỗi có thể được coi là dùng để xác định các bit chẵn lẻ bổ sung.

Ví dụ, trong thông tin báo hiệu chi tiết cho tầng L1 theo chế độ 2, tổng số bit chẵn lẻ LDPC bằng 12960 và số lượng nhóm bit luôn luôn được đọc lỗi bằng 16.

Trong trường hợp như vậy, mẫu thứ hai có thể được dùng để tạo ra các bit chẵn lẻ bổ sung tùy thuộc vào việc giá trị thu được sau khi lấy số lượng bit của tất cả các bit chẵn lẻ LDPC trừ đi số lượng bit chẵn lẻ LDPC cần đọc lỗi và cộng kết quả thu được của phép trừ này với số lượng bit chẵn lẻ bổ sung có phải là lớn hơn 7200 hay không. Theo sáng chế, 7200 là số lượng bit chẵn lẻ LDPC có trong các nhóm bit còn lại, ngoại trừ các nhóm bit luôn luôn được đọc lỗi, trong số các nhóm bit tạo nên các bit chẵn lẻ LDPC. Điều này có nghĩa là, $7200 = (36-16) \times 360$.

Cụ thể, khi giá trị thu được sau phép trừ và phép cộng nêu trên bằng hoặc nhỏ hơn 7200, tức là $12960 - N_{\text{punc}} + N_{\text{AP}} \leq 7200$, thì các bit chẵn lẻ bổ sung có thể được tạo ra theo mẫu thứ nhất.

Tuy nhiên, khi giá trị thu được sau phép trừ và phép cộng nêu trên lớn hơn 7200, tức là $12960 - N_{\text{punc}} + N_{\text{AP}} > 7200$, thì các bit chẵn lẻ bổ sung có thể được tạo ra theo mẫu thứ nhất và mẫu thứ hai.

Cụ thể, khi $12960 - N_{\text{punc}} + N_{\text{AP}} > 7200$, thì có thể chọn các bit có trong nhóm

bit nằm ở vị trí thứ 28 kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi, và có thể chọn các bit có trong nhóm bit nằm ở vị trí định trước so với vị trí thứ 29, để dùng làm các bit chẵn lẻ bổ sung.

Theo sáng chế, nhóm bit mà trong đó có bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi và nhóm bit (tức là nhóm bit mà trong đó có các bit chẵn lẻ LDPC được chọn cuối cùng khi chọn tuần tự kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi) ở vị trí định trước có thể được xác định theo số lượng bit chẵn lẻ LDPC được đọc lỗi và số lượng bit chẵn lẻ bổ sung được tạo ra.

Trong trường hợp như vậy, nhóm bit nằm ở vị trí thứ 28 kể từ bit chẵn lẻ LDPC đầu tiên trong số các bit chẵn lẻ LDPC được đọc lỗi được xác định theo mẫu thứ nhất, và nhóm bit nằm ở vị trí định trước so với vị trí thứ 29 được xác định theo mẫu thứ hai.

Nhờ đó, các bit chẵn lẻ bổ sung được xác định theo mẫu thứ nhất và mẫu thứ hai.

Như vậy, mẫu thứ nhất có thể được dùng để xác định các bit chẵn lẻ bổ sung được tạo ra cũng như các bit chẵn lẻ LDPC được đọc lỗi, còn mẫu thứ hai có thể được dùng để xác định các bit chẵn lẻ bổ sung được tạo ra và các bit chẵn lẻ LDPC luôn luôn được đọc lỗi, bất kể số lượng bit chẵn lẻ được đọc lỗi bằng các bộ phận đọc lỗi 217 và 318.

Ví dụ nêu trên mô tả rằng mẫu đan xen theo đơn vị nhóm có mẫu thứ nhất và mẫu thứ hai, nhưng ví dụ đó chỉ là để cho dễ hiểu về phương pháp đọc lỗi và phương pháp tạo ra bit chẵn lẻ bổ sung. Điều này có nghĩa là, mẫu đan xen theo đơn vị nhóm

có thể được coi là một mẫu không có sự phân chia ra thành mẫu thứ nhất và mẫu thứ hai. Trong trường hợp như vậy, thao tác đan xen theo đơn vị nhóm có thể được coi là được thực hiện theo một mẫu đối với cả phương pháp đục lỗ lần phương pháp tạo ra bit chẵn lẻ bổ sung.

Các giá trị được sử dụng trong ví dụ nêu trên như số lượng bit chẵn lẻ LDPC được đục lỗ chỉ được coi là ví dụ.

Các bộ phận loại bỏ bit không 218 và 321 có thể loại bỏ các bit không được đệm vào bằng các bộ đệm bit không 213 và 314 ra khỏi các từ mã LDPC được xuất ra từ các bộ phận đục lỗ 217 và 318, và xuất ra các bit còn lại để cung cấp cho các bộ phân kênh bit 219 và 322.

Theo sáng chế, bước loại bỏ các bit không không chỉ có loại bỏ các bit không được đệm vào mà có thể còn có xuất ra các bit còn lại, ngoại trừ các bit không được đệm vào, trong các từ mã LDPC.

Cụ thể, các bộ phận loại bỏ bit không 218 và 321 có thể loại bỏ $K_{ldpc} - N_{outer}$ bit không được đệm vào bằng các bộ đệm bit không 213 và 314. Vì vậy, $K_{ldpc} - N_{outer}$ bit không được đệm sẽ được loại bỏ, và do đó, có thể không được truyền đến thiết bị thu tín hiệu 200.

Ví dụ, như được thể hiện trên Fig.55, giả sử rằng tất cả các bit trong nhóm bit thứ nhất, nhóm bit thứ tư, nhóm bit thứ năm, nhóm bit thứ bảy và nhóm bit thứ tám trong số nhiều nhóm bit tạo nên từ mã LDPC được đệm bằng các bit không, và một số bit trong nhóm bit thứ hai được đệm bằng các bit không.

Trong trường hợp như vậy, các bộ phận loại bỏ bit không 218 và 321 có thể loại bỏ các bit không đã được đệm vào nhóm bit thứ nhất, nhóm bit thứ hai, nhóm bit thứ

tư, nhóm bit thứ năm, nhóm bit thứ bảy và nhóm bit thứ tám.

Như vậy, khi các bit không được loại bỏ, như được thể hiện trên Fig.55, có thể còn lại từ mã LDPC gồm có K_{sig} bit thông tin (tức là K_{sig} bit thông tin báo hiệu cơ bản cho tầng L1 hoặc K_{sig} bit thông tin báo hiệu chi tiết cho tầng L1), 168 bit kiểm tra chẵn lẻ BCH (tức là BCH FEC), và $N_{inner} - K_{ldpc} - N_{punc}$ hoặc $N_{inner} - K_{ldpc} - N_{punc} + N_{repeat}$ bit chẵn lẻ.

Điều này có nghĩa là, khi phương pháp lập được thực hiện, thì độ dài của tất cả các từ mã LDPC là $N_{FEC} + N_{repeat}$. Trong đó, $N_{FEC} = N_{outer} + N_{ldpc_parity} - N_{punc}$. Tuy nhiên, ở chế độ mà trong đó phương pháp lập không được thực hiện, thì độ dài của tất cả các từ mã LDPC là N_{FEC} .

Các bộ phân kênh bit 219 và 322 có thể đan xen các bit được xuất ra từ các bộ phận loại bỏ bit không 218 và 321, phân kênh cho các bit đã đan xen, và sau đó xuất ra các bit này để cung cấp cho các bộ ánh xạ chòm điểm 221 và 324.

Nhằm mục đích này, các bộ phân kênh bit 219 và 322 có thể có bộ phận đan xen khối (không được thể hiện trên hình vẽ) và bộ phân kênh (không được thể hiện trên hình vẽ).

Trước hết, sơ đồ đan xen khối được áp dụng trong bộ phận đan xen khối được thể hiện trên Fig.56.

Cụ thể, các bit có độ dài bằng N_{FEC} hoặc $N_{FEC} + N_{repeat}$ ở phía sau các bit không được loại bỏ có thể được ghi tuần tự theo từng cột vào bộ phận đan xen khối. Theo sáng chế, số lượng cột của bộ phận đan xen khối tương đương với bậc điều biến và số lượng hàng bằng N_{FEC}/η_{MOD} hoặc $(N_{FEC} + N_{repeat})/\eta_{MOD}$.

Ngoài ra, ở thao tác đọc, các bit cho một ký hiệu chòm điểm có thể được đọc

tuần tự theo hướng hàng được nhập vào bộ phân kênh. Thao tác này có thể được tiếp tục thực hiện đến hàng cuối cùng của cột.

Điều này có nghĩa là, N_{FEC} hoặc $(N_{\text{FEC}} + N_{\text{repeat}})$ bit có thể được ghi vào nhiều cột theo hướng cột kể từ hàng thứ nhất của cột thứ nhất, và các bit đã ghi vào nhiều cột được đọc tuần tự từ hàng thứ nhất đến hàng cuối cùng của các cột theo hướng hàng. Trong trường hợp như vậy, các bit được đọc trong cùng một hàng có thể tạo nên một ký hiệu điều biến.

Bộ phân kênh có thể phân kênh cho các bit được xuất ra từ bộ phận đan xen khối.

Cụ thể, bộ phân kênh có thể phân kênh cho từng nhóm bit được đan xen khối, tức là các bit được xuất ra khi đọc cùng một hàng bằng bộ phận đan xen khối trong nhóm bit theo đơn vị bit, trước khi các bit được ánh xạ lên chòm điểm.

Trong trường hợp như vậy, có thể có hai quy tắc ánh xạ theo bậc điều biến.

Cụ thể, khi sơ đồ điều biến QPSK được áp dụng để điều biến, vì độ tin cậy của các bit trong một ký hiệu chòm điểm là như nhau, nên bộ phân kênh không thực hiện thao tác phân kênh trên một nhóm bit. Vì vậy, nhóm bit được đọc và xuất ra từ bộ phận đan xen khối có thể được ánh xạ lên ký hiệu QPSK mà không cần có thao tác phân kênh.

Tuy nhiên, khi sơ đồ điều biến bậc cao được áp dụng, thì bộ phân kênh có thể thực hiện thao tác phân kênh trên nhóm bit được đọc và xuất ra từ bộ phận đan xen khối dựa vào biểu thức 36 dưới đây. Điều này có nghĩa là, một nhóm bit có thể được ánh xạ lên ký hiệu điều biến QAM dựa vào biểu thức 36 dưới đây:

$$S_{\text{demux_in}(i)} = \{b_i(0), b_i(1), b_i(2), \dots, b_i(\eta_{\text{MOD}}-1)\},$$

$$S_{\text{demux_out}(i)} = \{c_i(0), c_i(1), c_i(2), \dots, c_i(\eta_{\text{MOD}}-1)\}, \quad (36)$$

$$c_i(0) = b_i(i\% \eta_{\text{MOD}}), c_i(1) = b_i((i+1)\% \eta_{\text{MOD}}), \dots, c_i(\eta_{\text{MOD}}-1) = b_i((i+\eta_{\text{MOD}}-1)\% \eta_{\text{MOD}})$$

Trong biểu thức 36 nêu trên, % là phép toán môđulô, và η_{MOD} là bậc điều biến.

Ngoài ra, i là chỉ số của nhóm bit tương ứng với chỉ số hàng của bộ phận đan xen khối. Điều này có nghĩa là, nhóm bit đầu ra $S_{\text{demux_out}(i)}$ được ánh xạ lên mỗi ký hiệu điều biến QAM có thể được dịch chuyển tuần hoàn với khoảng cách $S_{\text{demux_in}(i)}$ theo chỉ số của nhóm bit i .

Fig.57 thể hiện ví dụ về phương pháp thực hiện thao tác phân kênh bit trên chòm điểm không đồng đều điều biến bậc 16 (16-Non Uniform Constellation, 16-NUC), tức là chòm điểm NUC 16-QAM. Thao tác này có thể được tiếp tục thực hiện cho tới khi tất cả các nhóm bit được đọc trong bộ phận đan xen khối.

Bộ phân kênh bit 323 có thể thực hiện thao tác, giống như thao tác được thực hiện bằng các bộ phân kênh bit 219 và 322, trên các bit chẵn lẻ bổ sung được xuất ra từ bộ tạo bit chẵn lẻ bổ sung 319, và xuất ra các bit đã được đan xen khối và phân kênh để cung cấp cho bộ ánh xạ chòm điểm 325.

Các bộ ánh xạ chòm điểm 221, 324 và 325 có thể lần lượt ánh xạ các bit được xuất ra từ các bộ phân kênh bit 219, 322 và 323 lên các ký hiệu chòm điểm.

Điều này có nghĩa là, mỗi bộ ánh xạ chòm điểm 221, 324 và 325 có thể ánh xạ ký hiệu $S_{\text{demux_out}(i)}$ lên một từ ô bằng cách sử dụng chòm điểm theo chế độ tương ứng. Theo sáng chế, ký hiệu $S_{\text{demux_out}(i)}$ có thể được tạo ra gồm có các bit với số lượng đúng bằng bậc điều biến.

Cụ thể, các bộ ánh xạ chòm điểm 221, 324 và 325 có thể ánh xạ các bit được xuất ra từ các bộ phân kênh bit 219, 322 và 323 lên các ký hiệu chòm điểm bằng cách

áp dụng sơ đồ điều biến QPSK, 16-QAM, 64-QAM, 256-QAM, v.v., theo chế độ tương ứng.

Trong trường hợp như vậy, các bộ ánh xạ chòm điểm 221, 324 và 325 có thể sử dụng chòm điểm NUC. Điều này có nghĩa là, các bộ ánh xạ chòm điểm 221, 324 và 325 có thể sử dụng chòm điểm NUC 16-QAM, NUC 64-QAM hoặc NUC 256-QAM. Trong đó, sơ đồ điều biến để áp dụng cho thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 theo chế độ tương ứng được thể hiện trong bảng 4 nêu trên.

Thiết bị truyền tín hiệu 100 có thể ánh xạ các ký hiệu chòm điểm lên khung và truyền các ký hiệu đã được ánh xạ đến thiết bị thu tín hiệu 200.

Cụ thể, thiết bị truyền tín hiệu 100 có thể ánh xạ các ký hiệu chòm điểm tương ứng với mỗi thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 được xuất ra từ các bộ ánh xạ chòm điểm 221 và 324, và ánh xạ các ký hiệu chòm điểm tương ứng với các bit chẵn lẻ bổ sung được xuất ra từ bộ ánh xạ chòm điểm 325 lên ký hiệu trong phần mở đầu của khung.

Trong trường hợp như vậy, thiết bị truyền tín hiệu 100 có thể ánh xạ các bit chẵn lẻ bổ sung được tạo ra dựa vào thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung hiện thời lên khung đứng trước khung hiện thời.

Điều này có nghĩa là, thiết bị truyền tín hiệu 100 có thể ánh xạ các bit từ mã LDPC chứa thông tin báo hiệu cơ bản cho tầng L1 tương ứng với khung thứ $(i-1)$ lên khung thứ $(i-1)$, ánh xạ các bit từ mã LDPC chứa thông tin báo hiệu chi tiết cho tầng L1 tương ứng với khung thứ $(i-1)$ lên khung thứ $(i-1)$, và ánh xạ thêm các bit chẵn lẻ bổ sung đã tạo ra được chọn trong số các bit chẵn lẻ LDPC được tạo ra dựa vào thông

tin báo hiệu chi tiết cho tầng L1 tương ứng với khung thứ i lên khung thứ $(i-1)$ và có thể truyền các bit đã được ánh xạ đến thiết bị thu tín hiệu 200.

Ngoài ra, thiết bị truyền tín hiệu 100 có thể ánh xạ dữ liệu lên các ký hiệu dữ liệu của khung, ngoài thông tin báo hiệu cho tầng L1, và truyền khung chứa thông tin báo hiệu cho tầng L1 và dữ liệu đến thiết bị thu tín hiệu 200.

Trong trường hợp như vậy, vì thông tin báo hiệu cho tầng L1 chứa thông tin báo hiệu liên quan đến dữ liệu, nên thông tin báo hiệu liên quan đến dữ liệu tương ứng với từng dữ liệu có thể được ánh xạ vào trong phần mở đầu của khung tương ứng. Ví dụ, thiết bị truyền tín hiệu 100 có thể ánh xạ thông tin báo hiệu cho tầng L1 chứa thông tin báo hiệu liên quan đến dữ liệu được ánh xạ lên khung thứ i , vào trong khung thứ i .

Nhờ đó, thiết bị thu tín hiệu 200 có thể sử dụng thông tin báo hiệu thu được từ khung để thu dữ liệu từ khung tương ứng để xử lý.

Fig.58 và Fig.59 là các sơ đồ khối thể hiện cấu hình của thiết bị thu tín hiệu theo phương án làm ví dụ thực hiện sáng chế.

Cụ thể, như được thể hiện trên Fig.58, thiết bị thu tín hiệu 200 có thể bao gồm bộ khử ánh xạ chòm điểm 2510, bộ dòn kênh 2520, bộ phận chèn giá trị tỷ số hợp lý dạng loga (Log-Likelihood Ratio, LLR) 2530, bộ phận kết hợp LLR 2540, bộ phận khử hoán vị chẵn lẻ 2550, bộ giải mã LDPC 2560, bộ phận loại bỏ bit không 2570, bộ giải mã BCH 2580 và bộ phận khử xáo trộn 2590 để xử lý thông tin báo hiệu cơ bản cho tầng L1.

Ngoài ra, như được thể hiện trên Fig.59, thiết bị thu tín hiệu 200 có thể bao gồm các bộ khử ánh xạ chòm điểm 2611 và 2612, các bộ dòn kênh 2621 và 2622, bộ phận

chèn LLR 2630, bộ phận kết hợp LLR 2640, bộ phận khử hoán vị chẵn lẻ 2650, bộ giải mã LDPC 2660, bộ phận loại bỏ bit không 2670, bộ giải mã BCH 2680, bộ phận khử xáo trộn 2690 và bộ phận khử phân đoạn 2695 để xử lý thông tin báo hiệu chi tiết cho tầng L1.

Theo sáng chế, các bộ phận được thể hiện trên Fig.58 và Fig.59 lần lượt là các bộ phận thực hiện các chức năng tương ứng với chức năng của các bộ phận được thể hiện trên Fig.42 và Fig.43, nhưng đó chỉ là một ví dụ, và trong một số trường hợp, một số bộ phận có thể được loại bỏ hoặc thay đổi, và các bộ phận khác có thể được bổ sung vào.

Thiết bị thu tín hiệu 200 có thể thu nhận tín hiệu đồng bộ hoá khung từ phần khởi động của khung và thu thông tin báo hiệu cơ bản cho tầng L1 từ phần mở đầu của khung bằng cách sử dụng thông tin để xử lý thông tin báo hiệu cơ bản cho tầng L1 có trong phần khởi động.

Ngoài ra, thiết bị thu tín hiệu 200 có thể thu thông tin báo hiệu chi tiết cho tầng L1 từ phần mở đầu bằng cách sử dụng thông tin để xử lý thông tin báo hiệu chi tiết cho tầng L1 có trong thông tin báo hiệu cơ bản cho tầng L1, và thu dữ liệu phát rộng theo yêu cầu của người dùng từ các ký hiệu dữ liệu trong khung bằng cách sử dụng thông tin báo hiệu chi tiết cho tầng L1.

Vì vậy, thiết bị thu tín hiệu 200 có thể xác định chế độ đã được sử dụng ở thiết bị truyền tín hiệu 100 để xử lý thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1, và xử lý tín hiệu thu được từ thiết bị truyền tín hiệu 100 theo chế độ đã xác định để thu thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1. Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ

trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để xử lý thông tin báo hiệu theo các chế độ tương ứng.

Như vậy, thông tin báo hiệu cơ bản cho tầng L1 và thông tin báo hiệu chi tiết cho tầng L1 có thể được thu nhận tuần tự từ phần mở đầu. Khi mô tả dựa vào Fig.58 và Fig.59, các bộ phận thực hiện chức năng giống nhau sẽ được mô tả cùng nhau để cho thuận tiện.

Các bộ khử ánh xạ chòm điểm 2510, 2611 và 2612 giải điều biến tín hiệu thu được từ thiết bị truyền tín hiệu 100.

Cụ thể, các bộ khử ánh xạ chòm điểm 2510, 2611 và 2612 lần lượt là các bộ phận tương ứng với các bộ ánh xạ chòm điểm 221, 324 và 325 trong thiết bị truyền tín hiệu 100, và có thể giải điều biến tín hiệu thu được từ thiết bị truyền tín hiệu 100 và tạo ra các giá trị tương ứng với các bit được truyền từ thiết bị truyền tín hiệu 100.

Điều này có nghĩa là, như đã nêu trên, thiết bị truyền tín hiệu 100 ánh xạ từ mã LDPC chứa thông tin báo hiệu cơ bản cho tầng L1 và từ mã LDPC chứa thông tin báo hiệu chi tiết cho tầng L1 vào trong phần mở đầu của khung, và truyền từ mã LDPC đã ánh xạ đến thiết bị thu tín hiệu 200. Ngoài ra, trong một số trường hợp, thiết bị truyền tín hiệu 100 có thể ánh xạ các bit chẵn lẻ bổ sung vào trong phần mở đầu của khung và truyền các bit đã ánh xạ đến thiết bị thu tín hiệu 200.

Nhờ đó, các bộ khử ánh xạ chòm điểm 2510 và 2611 có thể tạo ra các giá trị tương ứng với các bit từ mã LDPC chứa thông tin báo hiệu cơ bản cho tầng L1 và các bit từ mã LDPC chứa thông tin báo hiệu chi tiết cho tầng L1. Ngoài ra, bộ khử ánh xạ chòm điểm 2612 có thể tạo ra các giá trị tương ứng với các bit chẵn lẻ bổ sung.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về sơ

điều biến đã được sử dụng ở thiết bị truyền tín hiệu 100 để điều biến thông tin báo hiệu cơ bản cho tầng L1, thông tin báo hiệu chi tiết cho tầng L1 và các bit chẵn lẻ bổ sung theo các chế độ tương ứng. Vì vậy, các bộ khử ánh xạ chòm điểm 2510, 2611 và 2612 có thể giải điều biến tín hiệu thu được từ thiết bị truyền tín hiệu 100 theo các chế độ tương ứng để tạo ra các giá trị tương ứng với các bit từ mã LDPC và các bit chẵn lẻ bổ sung.

Giá trị tương ứng với bit được truyền từ thiết bị truyền tín hiệu 100 là giá trị được tính dựa vào xác suất để cho bit thu được bằng 0 và bằng 1, và để thay thế, bản thân xác suất này cũng có thể được dùng làm giá trị tương ứng với mỗi bit. Ví dụ khác, giá trị này cũng có thể là giá trị tỷ số hợp lý (Likelihood Ratio, LR) hoặc giá trị LLR.

Cụ thể, giá trị LR có thể là tỷ số giữa xác suất để cho bit được truyền từ thiết bị truyền tín hiệu 100 bằng 0 và xác suất để cho bit đó bằng 1, và giá trị LLR có thể là giá trị thu được khi lấy logarit cho tỷ số giữa xác suất để cho bit được truyền từ thiết bị truyền tín hiệu 100 bằng 0 và xác suất để cho bit đó bằng 1.

Ví dụ nêu trên sử dụng giá trị LR hoặc giá trị LLR, nhưng đó chỉ là một ví dụ. Theo phương án làm ví dụ khác để thực hiện sáng chế, cũng có thể sử dụng bản thân tín hiệu thu được, chứ không phải là giá trị LR hoặc giá trị LLR.

Các bộ dồn kênh 2520, 2621 và 2622 thực hiện thao tác dồn kênh trên các giá trị LLR được xuất ra từ các bộ khử ánh xạ chòm điểm 2510, 2611 và 2612.

Cụ thể, các bộ dồn kênh 2520, 2621 và 2622 là các bộ phận tương ứng với các bộ phận kênh bit 219, 322 và 323 trong thiết bị truyền tín hiệu 100, và có thể lần lượt thực hiện các thao tác tương ứng với các thao tác của các bộ phận kênh bit 219, 322

và 323.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện thao tác phân kênh và đan xen khối. Vì vậy, các bộ dồn kênh 2520, 2621 và 2622 có thể thực hiện các thao tác đảo ngược lại các thao tác phân kênh và đan xen khối của các bộ phân kênh bit 219, 322 và 323 trên giá trị LLR tương ứng với một từ ô để dồn kênh giá trị LLR tương ứng với từ ô đó theo đơn vị bit.

Các bộ phận chèn LLR 2530 và 2630 có thể lần lượt chèn các giá trị LLR cho các bit đục lỗ và rút gọn vào trong các giá trị LLR được xuất ra từ các bộ dồn kênh 2520 và 2621. Trong trường hợp như vậy, các bộ phận chèn LLR 2530 và 2630 có thể chèn các giá trị LLR định trước vào giữa các giá trị LLR được xuất ra từ các bộ dồn kênh 2520 và 2621 hoặc vào phần đầu hoặc phần cuối của các giá trị này.

Cụ thể, các bộ phận chèn LLR 2530 và 2630 lần lượt là các bộ phận tương ứng với các bộ phận loại bỏ bit không 218 và 321 và các bộ phận đục lỗ 217 và 318 trong thiết bị truyền tín hiệu 100, và có thể lần lượt thực hiện các thao tác tương ứng với các thao tác của các bộ phận loại bỏ bit không 218 và 321 và các bộ phận đục lỗ 217 và 318.

Trước hết, các bộ phận chèn LLR 2530 và 2630 có thể chèn các giá trị LLR tương ứng với các bit không vào các vị trí mà ở đó các bit không được đệm vào trong từ mã LDPC. Trong trường hợp như vậy, các giá trị LLR tương ứng với các bit không được đệm vào, tức là các bit không được rút gọn, có thể bằng ∞ hoặc $-\infty$. Tuy nhiên, ∞ hoặc $-\infty$ là giá trị lý thuyết, còn trên thực tế đó có thể là giá trị cực đại hoặc giá trị cực tiểu của giá trị LLR được sử dụng ở thiết bị thu tín hiệu 200.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số và/hoặc các mẫu đã được sử dụng ở thiết bị truyền tín hiệu 100 để đếm các bit không theo các chế độ tương ứng. Vì vậy, các bộ phận chèn LLR 2530 và 2630 có thể xác định các vị trí mà ở đó các bit không được đếm vào trong các từ mã LDPC theo các chế độ tương ứng, và chèn các giá trị LLR tương ứng với các bit không được rút gọn vào các vị trí tương ứng.

Ngoài ra, các bộ phận chèn LLR 2530 và 2630 có thể chèn các giá trị LLR tương ứng với các bit được đọc lỗi vào vị trí của các bit được đọc lỗi trong từ mã LDPC. Trong trường hợp như vậy, các giá trị LLR tương ứng với các bit được đọc lỗi có thể bằng 0.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số và/hoặc các mẫu đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện phương pháp đọc lỗi theo các chế độ tương ứng. Vì vậy, các bộ phận chèn LLR 2530 và 2630 có thể xác định độ dài của các bit chẵn lẻ LDPC được đọc lỗi theo các chế độ tương ứng, và chèn các giá trị LLR tương ứng vào các vị trí mà ở đó các bit chẵn lẻ LDPC được đọc lỗi.

Trong trường hợp các bit chẵn lẻ bổ sung được chọn từ các bit được đọc lỗi trong số các bit chẵn lẻ bổ sung, bộ phận chèn LLR 2630 có thể chèn các giá trị LLR tương ứng với các bit chẵn lẻ bổ sung thu được, chứ không phải là giá trị LLR '0' cho bit được đọc lỗi, vào vị trí của các bit được đọc lỗi.

Các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp, tức là cộng, các giá trị LLR được xuất ra từ các bộ phận chèn LLR 2530 và 2630 với giá trị LLR được xuất ra từ bộ dòn kênh 2622. Tuy nhiên, các bộ phận kết hợp LLR 2540 và 2640 dùng để

cập nhật các giá trị LLR cho các bit xác định với những giá trị chính xác hơn. Tuy nhiên, các giá trị LLR cho các bit xác định cũng có thể được giải mã dựa trên các giá trị LLR thu được mà không có các bộ phận kết hợp LLR 2540 và 2640, và vì vậy, trong một số trường hợp, các bộ phận kết hợp LLR 2540 và 2640 có thể được loại bỏ.

Cụ thể, bộ phận kết hợp LLR 2540 là bộ phận tương ứng với bộ phận lặp 216 trong thiết bị truyền tín hiệu 100, và có thể thực hiện thao tác tương ứng với thao tác của bộ phận lặp 216. Theo cách khác, bộ phận kết hợp LLR 2640 là bộ phận tương ứng với bộ phận lặp 317 và bộ tạo bit chẵn lẻ bổ sung 319 trong thiết bị truyền tín hiệu 100, và có thể thực hiện các thao tác tương ứng với các thao tác của bộ phận lặp 317 và bộ tạo bit chẵn lẻ bổ sung 319.

Trước hết, các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR tương ứng với các bit lặp với các giá trị LLR khác. Theo sáng chế, các giá trị LLR khác có thể là các bit làm cơ sở để tạo ra các bit lặp bằng thiết bị truyền tín hiệu 100, tức là các giá trị LLR cho các bit chẵn lẻ LDPC, được chọn làm đối tượng lặp.

Điều này có nghĩa là, như đã nêu trên, thiết bị truyền tín hiệu 100 chọn các bit trong số các bit chẵn lẻ LDPC và lặp lại các bit đã chọn ở giữa các bit thông tin LDPC và các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, và truyền các bit lặp đến thiết bị thu tín hiệu 200.

Do đó, các giá trị LLR cho các bit chẵn lẻ LDPC có thể gồm có các giá trị LLR cho các bit chẵn lẻ LDPC được lặp lại và các giá trị LLR cho các bit chẵn lẻ LDPC không phải là được lặp lại, tức là các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC. Vì vậy, các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR cho cùng một bit chẵn lẻ LDPC.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện phương pháp lặp theo các chế độ tương ứng. Vì vậy, các bộ phận kết hợp LLR 2540 và 2640 có thể xác định độ dài của các bit chẵn lẻ LDPC được lặp lại, xác định vị trí của các bit làm cơ sở để lặp lại, và kết hợp các giá trị LLR cho các bit chẵn lẻ LDPC được lặp lại với các giá trị LLR cho các bit chẵn lẻ LDPC làm cơ sở để lặp lại và được tạo ra bằng cách mã hoá LDPC.

Ví dụ, như được thể hiện trên Fig.60 và Fig.61, các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR cho các bit chẵn lẻ LDPC được lặp lại với các giá trị LLR cho các bit chẵn lẻ LDPC làm cơ sở để lặp lại và được tạo ra bằng cách mã hoá LDPC.

Khi các bit chẵn lẻ LDPC được lặp lại n lần, thì các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR cho các bit ở cùng một vị trí với n lần hoặc với số lần ít hơn.

Ví dụ, Fig.60 thể hiện trường hợp trong đó một số bit chẵn lẻ LDPC, ngoại trừ các bit được đọc lỗi, được lặp lại một lần. Trong trường hợp như vậy, các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR cho các bit chẵn lẻ LDPC được lặp lại với các giá trị LLR cho các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC, và sau đó, xuất ra các giá trị LLR đã được kết hợp, hoặc xuất ra các giá trị LLR cho các bit chẵn lẻ LDPC được lặp lại thu được hoặc các giá trị LLR cho các bit chẵn lẻ LDPC được tạo ra bằng cách mã hoá LDPC thu được mà không cần kết hợp các giá trị đó.

Ví dụ khác, Fig.61 thể hiện trường hợp trong đó một số bit chẵn lẻ LDPC được

truyền, không được đọc lỗi, được lặp lại hai lần, phần còn lại được lặp lại một lần, và các bit chẵn lẻ LDPC được đọc lỗi được lặp lại một lần.

Trong trường hợp như vậy, các bộ phận kết hợp LLR 2540 và 2640 có thể xử lý phần còn lại và các bit được đọc lỗi được lặp lại một lần bằng cách sử dụng sơ đồ giống như sơ đồ đã được mô tả trên đây. Tuy nhiên, các bộ phận kết hợp LLR 2540 và 2640 có thể xử lý phần được lặp lại hai lần theo cách như sau. Trong trường hợp như vậy, để cho thuận tiện, một trong số hai phần được tạo ra bằng cách lặp lại hai lần một số bit chẵn lẻ LDPC được gọi là phần thứ nhất và phần kia được gọi là phần thứ hai.

Cụ thể, các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR cho mỗi phần trong số phần thứ nhất và phần thứ hai với các giá trị LLR cho các bit chẵn lẻ LDPC. Theo cách khác, các bộ phận kết hợp LLR 2540 và 2640 có thể kết hợp các giá trị LLR cho phần thứ nhất với các giá trị LLR cho các bit chẵn lẻ LDPC, kết hợp các giá trị LLR cho phần thứ hai với các giá trị LLR cho các bit chẵn lẻ LDPC, hoặc kết hợp các giá trị LLR cho phần thứ nhất với các giá trị LLR cho phần thứ hai. Theo cách khác, các bộ phận kết hợp LLR 2540 và 2640 có thể xuất ra các giá trị LLR cho phần thứ nhất, các giá trị LLR cho phần thứ hai, các giá trị LLR cho phần còn lại, và các bit được đọc lỗi, mà không cần kết hợp riêng.

Ngoài ra, bộ phận kết hợp LLR 2640 có thể kết hợp các giá trị LLR tương ứng với các bit chẵn lẻ bổ sung với các giá trị LLR khác. Theo sáng chế, các giá trị LLR khác có thể là các bit chẵn lẻ LDPC làm cơ sở để tạo ra các bit chẵn lẻ bổ sung bằng thiết bị truyền tín hiệu 100, tức là các giá trị LLR cho các bit chẵn lẻ LDPC được chọn để tạo ra các bit chẵn lẻ bổ sung.

Điều này có nghĩa là, như đã nêu trên, thiết bị truyền tín hiệu 100 có thể ánh xạ các bit chẵn lẻ bổ sung cho thông tin báo hiệu chi tiết cho tầng L1 được truyền trong khung hiện thời lên khung trước đó và truyền các bit đã ánh xạ đến thiết bị thu tín hiệu 200.

Trong trường hợp như vậy, các bit chẵn lẻ bổ sung có thể có các bit chẵn lẻ LDPC được đọc lỗi và không được truyền trong khung hiện thời, và trong một số trường hợp, có thể còn có các bit chẵn lẻ LDPC được truyền trong khung hiện thời.

Do đó, bộ phận kết hợp LLR 2640 có thể kết hợp các giá trị LLR cho các bit chẵn lẻ bổ sung thu được trong khung hiện thời với các giá trị LLR được chèn vào vị trí của các bit chẵn lẻ LDPC được đọc lỗi trong từ mã LDPC thu được từ khung kế tiếp và các giá trị LLR cho các bit chẵn lẻ LDPC thu được từ khung kế tiếp.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số và/hoặc các mẫu đã được sử dụng ở thiết bị truyền tín hiệu 100 để tạo ra các bit chẵn lẻ bổ sung theo các chế độ tương ứng. Vì vậy, bộ phận kết hợp LLR 2640 có thể xác định độ dài của các bit chẵn lẻ bổ sung, xác định vị trí của các bit chẵn lẻ LDPC làm cơ sở để tạo ra các bit chẵn lẻ bổ sung, và kết hợp các giá trị LLR cho các bit chẵn lẻ bổ sung với các giá trị LLR cho các bit chẵn lẻ LDPC làm cơ sở để tạo ra các bit chẵn lẻ bổ sung.

Các bộ phận khử hoán vị chẵn lẻ 2550 và 2650 có thể lần lượt khử hoán vị cho các giá trị LLR được xuất ra từ các bộ phận kết hợp LLR 2540 và 2640.

Cụ thể, các bộ phận khử hoán vị chẵn lẻ 2550 và 2650 là các bộ phận tương ứng với các bộ phận hoán vị chẵn lẻ 215 và 316 trong thiết bị truyền tín hiệu 100, và có thể lần lượt thực hiện các thao tác tương ứng với các thao tác của các bộ phận hoán vị

chẵn lẻ 215 và 316.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số và/hoặc các mẫu đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện các thao tác đan xen theo đơn vị nhóm và đan xen bit chẵn lẻ theo các chế độ tương ứng. Vì vậy, các bộ phận khử hoán vị chẵn lẻ 2550 và 2650 có thể lần lượt thực hiện các thao tác đảo ngược lại các thao tác đan xen theo đơn vị nhóm và đan xen bit chẵn lẻ của các bộ phận hoán vị chẵn lẻ 215 và 316 trên các giá trị LLR tương ứng với các bit từ mã LDPC, tức là thực hiện các thao tác khử đan xen theo đơn vị nhóm và khử đan xen bit chẵn lẻ để thực hiện chức năng khử hoán vị chẵn lẻ trên các giá trị LLR tương ứng với các bit từ mã LDPC.

Các bộ giải mã LDPC 2560 và 2660 có thể lần lượt thực hiện bước giải mã LDPC dựa vào các giá trị LLR được xuất ra từ các bộ phận khử hoán vị chẵn lẻ 2550 và 2650.

Cụ thể, các bộ giải mã LDPC 2560 và 2660 là các bộ phận tương ứng với các bộ mã hoá LDPC 214 và 315 trong thiết bị truyền tín hiệu 100 và có thể lần lượt thực hiện các thao tác tương ứng với các thao tác của các bộ mã hoá LDPC 214 và 315.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện bước mã hoá LDPC theo các chế độ tương ứng. Vì vậy, các bộ giải mã LDPC 2560 và có thể thực hiện bước giải mã LDPC dựa vào các giá trị LLR được xuất ra từ các bộ phận khử hoán vị chẵn lẻ 2550 và 2650 theo các chế độ tương ứng.

Ví dụ, các bộ giải mã LDPC 2560 và 2660 có thể thực hiện bước giải mã LDPC dựa vào các giá trị LLR được xuất ra từ các bộ phận khử hoán vị chẵn lẻ 2550 và

2650 bằng cách giải mã lặp dựa vào thuật toán tổng-tích và xuất ra các bit đã được sửa lỗi bằng cách giải mã LDPC.

Các bộ phận loại bỏ bit không 2570 và 2670 có thể lần lượt loại bỏ các bit không ra khỏi các bit được xuất ra từ các bộ giải mã LDPC 2560 và 2660.

Cụ thể, các bộ phận loại bỏ bit không 2570 và 2670 là các bộ phận tương ứng với các bộ đệm bit không 213 và 314 trong thiết bị truyền tín hiệu 100, và có thể lần lượt thực hiện các thao tác tương ứng với các thao tác của các bộ đệm bit không 213 và 314.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số và/hoặc các mẫu đã được sử dụng ở thiết bị truyền tín hiệu 100 để đệm các bit không theo các chế độ tương ứng. Vì vậy, các bộ phận loại bỏ bit không 2570 và 2670 có thể lần lượt loại bỏ các bit không được đệm vào bằng các bộ đệm bit không 213 và 314 ra khỏi các bit được xuất ra từ các bộ giải mã LDPC 2560 và 2660.

Các bộ giải mã BCH 2580 và 2680 có thể lần lượt thực hiện bước giải mã BCH trên các bit được xuất ra từ các bộ phận loại bỏ bit không 2570 và 2670.

Cụ thể, các bộ giải mã BCH 2580 và 2680 là các bộ phận tương ứng với các bộ mã hoá BCH 212 và 313 trong thiết bị truyền tín hiệu 100, và có thể lần lượt thực hiện các thao tác tương ứng với các thao tác của các bộ mã hoá BCH 212 và 313.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện bước mã hoá BCH. Vì vậy, các bộ giải mã BCH 2580 và 2680 có thể sửa lỗi bằng cách thực hiện bước giải mã BCH trên các bit được xuất ra từ các bộ phận loại bỏ bit không 2570 và 2670 và xuất ra các bit đã được sửa lỗi.

Các bộ phận khử xáo trộn 2590 và 2690 có thể lần lượt khử xáo trộn cho các bit được xuất ra từ các bộ giải mã BCH 2580 và 2680.

Cụ thể, các bộ phận khử xáo trộn 2590 và 2690 là các bộ phận tương ứng với các bộ phận xáo trộn 211 và 312 trong thiết bị truyền tín hiệu 100, và có thể thực hiện các thao tác tương ứng với các thao tác của các bộ phận xáo trộn 211 và 312.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện thao tác xáo trộn. Vì vậy, các bộ phận khử xáo trộn 2590 và 2690 có thể lần lượt khử xáo trộn cho các bit được xuất ra từ các bộ giải mã BCH 2580 và 2680 và xuất ra các bit đã được khử xáo trộn.

Nhờ đó, thông tin báo hiệu cơ bản cho tầng L1 được truyền từ thiết bị truyền tín hiệu 100 có thể được khôi phục. Ngoài ra, khi thiết bị truyền tín hiệu 100 không thực hiện thao tác phân đoạn trên thông tin báo hiệu chi tiết cho tầng L1, thì thông tin báo hiệu chi tiết cho tầng L1 được truyền từ thiết bị truyền tín hiệu 100 cũng có thể được khôi phục.

Tuy nhiên, khi thiết bị truyền tín hiệu 100 thực hiện thao tác phân đoạn trên thông tin báo hiệu chi tiết cho tầng L1, thì bộ phận khử phân đoạn 2695 có thể khử phân đoạn cho các bit được xuất ra từ bộ phận khử xáo trộn 2690.

Cụ thể, bộ phận khử phân đoạn 2695 là bộ phận tương ứng với bộ phận phân đoạn 311 trong thiết bị truyền tín hiệu 100, và có thể thực hiện thao tác tương ứng với thao tác của bộ phận phân đoạn 311.

Nhằm mục đích này, thiết bị thu tín hiệu 200 có thể lưu trữ trước thông tin về các thông số đã được sử dụng ở thiết bị truyền tín hiệu 100 để thực hiện thao tác phân

đoạn. Vì vậy, bộ phận khử phân đoạn 2695 có thể kết hợp các bit được xuất ra từ bộ phận khử xáo trộn 2690, tức là các đoạn thông tin báo hiệu chi tiết cho tầng L1 để khôi phục thông tin báo hiệu chi tiết cho tầng L1 trước khi phân đoạn.

Thông tin về độ dài của thông tin báo hiệu cho tầng L1 được cung cấp ở dạng như được thể hiện trên Fig.62. Vì vậy, thiết bị thu tín hiệu 200 có thể tính độ dài của thông tin báo hiệu chi tiết cho tầng L1 và độ dài của các bit chẵn lẻ bổ sung.

Dựa vào Fig.62, vì thông tin báo hiệu cơ bản cho tầng L1 cung cấp thông tin về tổng số các ô của thông tin báo hiệu chi tiết cho tầng L1, nên thiết bị thu tín hiệu 200 cần phải tính độ dài của thông tin báo hiệu chi tiết cho tầng L1 và độ dài của các bit chẵn lẻ bổ sung.

Cụ thể, khi trường L1B_L1_Detail_additional_parity_mode trong thông tin báo hiệu cơ bản cho tầng L1 có giá trị khác 0, vì đã biết thông tin của trường L1B_L1_Detail_total_cells biểu thị độ dài của tổng số các ô ($= N_{L1_detail_total_cells}$), cho nên thiết bị thu tín hiệu 200 có thể tính độ dài $N_{L1_detail_cells}$ của thông tin báo hiệu chi tiết cho tầng L1 và độ dài $N_{AP_total_cells}$ của các bit chẵn lẻ bổ sung dựa vào các biểu thức từ biểu thức 37 đến biểu thức 39 dưới đây:

$$N_{L1_FEC_cells} = \frac{N_{outer} + N_{repeat} + N_{ldpc_parity} - N_{punc}}{\eta_{MOD}} = \frac{N_{FEC}}{\eta_{MOD}} \quad (37)$$

$$N_{L1_detail_cells} = N_{L1D_FECFRAME} \times N_{L1_FEC_cells} \quad (38)$$

$$N_{AP_total_cells} = N_{L1_detail_total_cells} - N_{L1_detail_cells} \quad (39)$$

Trong trường hợp như vậy, dựa vào các biểu thức từ biểu thức 37 đến biểu thức 39 nêu trên, có thể thu được giá trị $N_{AP_total_cells}$ dựa vào giá trị $N_{L1_detail_total_cells}$ là giá trị có thể thu được từ thông tin của trường L1B_L1_Detail_total_cells trong thông tin báo hiệu cơ bản cho tầng L1, giá trị N_{FEC} , giá trị $N_{L1D_FECFRAME}$, và bậc điều biến

η_{MOD} . Ví dụ, $N_{AP_total_cells}$ có thể được tính dựa vào biểu thức 40 dưới đây:

$$N_{AP_total_cells} = N_{L1_detail_total_cells} - N_{L1D_FECFRAME} \times \frac{N_{FEC}}{\eta_{MOD}} \quad (40)$$

Cú pháp và ngữ nghĩa của các trường trong thông tin báo hiệu cơ bản cho tầng L1 được thể hiện trong bảng 13 dưới đây.

Bảng 13

Cú pháp	Số bit	Định dạng
L1_basic_signaling() {		
L1B_L1_Detail_size_bits	16	uimsbf
L1B_L1_Detail_fec_type	3	uimsbf
L1B_L1_Detail_additional_parity_mode	2	uimsbf
L1B_L1_Detail_total_cells	19	uimsbf
L1B_Reserved	?	uimsbf
L1B_crc	32	uimsbf
}		

Do đó, thiết bị thu tín hiệu 200 có thể thực hiện thao tác thu các bit chẵn lẻ bổ sung trong khung kế tiếp dựa vào các bit chẵn lẻ bổ sung được truyền đến $N_{AP_total_cells}$ ô trong số các ô thu được trong thông tin báo hiệu chi tiết cho tầng L1.

Fig.63 là lưu đồ thể hiện phương pháp tạo ra bit chẵn lẻ bổ sung theo phương án làm ví dụ thực hiện sáng chế.

Trước hết, các bit đầu vào chứa các bit mã hoá ngoài được mã hoá để tạo ra các bit chẵn lẻ (S6210) và một số bit trong số các bit chẵn lẻ được đọc lỗi (S6220).

Ngoài ra, ít nhất một số bit trong số các bit chẵn lẻ được chọn để tạo ra các bit chẵn lẻ bổ sung được truyền trong khung trước đó (S6230).

Trong trường hợp như vậy, số lượng bit chẵn lẻ bổ sung được tạo ra có thể được xác định dựa vào số lượng bit mã hoá ngoài được truyền trong khung hiện thời và số lượng bit chẵn lẻ còn lại sau khi đọc lỗi.

Phương pháp này có thể còn bao gồm bước lặp lại ít nhất một số bit của từ mã LDPC trong từ mã LDPC để lặp lại ít nhất một số bit của từ mã LDPC gồm có các bit đầu vào và các bit chẵn lẻ trong khung hiện thời và truyền các bit này.

Trong trường hợp như vậy, khi phương pháp lặp được thực hiện, số lượng bit chẵn lẻ bổ sung được tạo ra có thể được xác định dựa vào số lượng bit mã hoá ngoài được truyền trong khung hiện thời, số lượng bit chẵn lẻ còn lại sau khi đọc lỗi, và số lượng bit được lặp lại trong khung hiện thời.

Ngoài ra, số lượng bit chẵn lẻ bổ sung được tạo ra được tính dựa vào số lượng tạm thời N_{AP_temp} bit chẵn lẻ bổ sung được tạo ra được tính dựa vào biểu thức 8 nêu trên, và cụ thể, có thể được tính dựa vào biểu thức 10 nêu trên.

Phương pháp tạo ra bit chẵn lẻ bổ sung đã được mô tả chi tiết trên đây.

Fig.64 là sơ đồ thể hiện hiệu quả đạt được đối với trường hợp có sử dụng các bit chẵn lẻ bổ sung và đối với trường hợp không sử dụng các bit chẵn lẻ bổ sung theo phương án làm ví dụ thực hiện sáng chế.

Trên Fig.64, với độ dài của mỗi thông tin báo hiệu chi tiết cho tầng L1 bằng 2000, 3000 và 4000, trường hợp không sử dụng các bit chẵn lẻ bổ sung (đường nét đứt) và trường hợp có sử dụng các bit chẵn lẻ bổ sung (đường nét liền) được dùng để biểu diễn tỷ lệ lỗi khung (Frame Error Rate, FER), và có thể thấy rằng khi sử dụng các bit chẵn lẻ bổ sung thì có thể đạt được độ tăng ích do mã hoá và độ tăng ích do phân tập (độ dốc của đồ thị).

Sáng chế có thể đề cập đến vật ghi bất khả biến đọc được bằng máy tính trong đó lưu trữ chương trình để thực hiện các phương pháp được mô tả theo phương án làm ví dụ thực hiện sáng chế. Vật ghi bất khả biến đọc được bằng máy tính không

phải là vật ghi lưu trữ dữ liệu tạm thời trên đó, như thanh ghi, bộ nhớ tác động nhanh, bộ nhớ, hoặc các loại vật ghi khác, mà là vật ghi lưu trữ dữ liệu bán cố định trên đó và có thể đọc được bằng thiết bị như bộ vi xử lý. Cụ thể, nhiều ứng dụng hoặc chương trình nêu trên có thể được lưu trữ và cung cấp trên vật ghi bất khả biến đọc được bằng máy tính như đĩa compac (Compact Disk, CD), đĩa đa năng kỹ thuật số (Digital Versatile Disk, DVD), đĩa cứng, đĩa Blu-ray, bus nối tiếp đa năng (Universal Serial Bus, USB), thẻ nhớ, bộ nhớ chỉ đọc (Read Only Memory, ROM), hoặc các loại vật ghi khác.

Ít nhất một trong số các bộ phận, phần tử, môđun hoặc đơn vị được biểu thị ở dạng khối như được thể hiện trên các hình vẽ Fig.1, Fig.42, Fig.43, Fig.58 và Fig.59 có thể được thực hiện dưới dạng cấu trúc phần cứng, phần mềm và/hoặc phần sụn để thực hiện các chức năng tương ứng nêu trên, theo phương án làm ví dụ thực hiện sáng chế. Ví dụ, ít nhất một trong số các bộ phận, phần tử, môđun hoặc đơn vị có thể sử dụng cấu trúc mạch trực tiếp, như bộ nhớ, bộ xử lý, mạch logic, bảng tra cứu, v.v., để có thể thực hiện các chức năng tương ứng theo sự điều khiển của một hoặc nhiều bộ vi xử lý hoặc thiết bị điều khiển khác. Ngoài ra, ít nhất một trong số các bộ phận, phần tử, môđun hoặc đơn vị có thể được biểu thị ở dạng cụ thể bằng môđun, chương trình, hoặc đoạn mã, chứa một hoặc nhiều lệnh thi hành được để thực hiện các chức năng logic cụ thể, và được thi hành bằng một hoặc nhiều bộ vi xử lý hoặc thiết bị điều khiển khác. Hơn nữa, ít nhất một trong số các bộ phận, phần tử, môđun hoặc đơn vị có thể còn có hoặc được thực hiện dưới dạng bộ xử lý như bộ xử lý trung tâm (Central Processing Unit, CPU) để thực hiện các chức năng tương ứng, bộ vi xử lý, hoặc các bộ xử lý khác. Hai hoặc nhiều hơn hai bộ phận, phần tử, môđun hoặc đơn vị

nêu trên có thể được kết hợp lại thành một bộ phận, phần tử, môđun hoặc đơn vị để thực hiện tất cả các thao tác hoặc chức năng của hai hoặc nhiều hơn hai bộ phận, phần tử, môđun hoặc đơn vị được kết hợp. Ngoài ra, ít nhất một phần của các chức năng của ít nhất một trong số các bộ phận, phần tử, môđun hoặc đơn vị có thể được thực hiện bằng một bộ phận, phần tử, môđun hoặc đơn vị khác. Ngoài ra, mặc dù trong các sơ đồ khối nêu trên không thể hiện bus, nhưng việc truyền thông giữa các bộ phận, phần tử, môđun hoặc đơn vị có thể được thực hiện thông qua bus. Các khía cạnh liên quan đến chức năng trong các phương án làm ví dụ thực hiện sáng chế nêu trên có thể được thực hiện dưới dạng các thuật toán để thực hiện trong một hoặc nhiều bộ xử lý. Hơn nữa, các bộ phận, phần tử, môđun hoặc đơn vị được biểu thị ở dạng khối hoặc các bước xử lý có thể áp dụng các kỹ thuật bất kỳ trong lĩnh vực liên quan để tạo cấu hình cho thiết bị điện tử, xử lý tín hiệu và/hoặc thông tin điều khiển, xử lý dữ liệu và các chức năng khác.

Mặc dù các phương án làm ví dụ thực hiện sáng chế được thể hiện trên các hình vẽ và được mô tả trên đây, nhưng sáng chế không chỉ giới hạn ở các phương án làm ví dụ nêu trên, người có hiểu biết trung bình về lĩnh vực kỹ thuật tương ứng có thể tìm ra nhiều phương án cải biến khác nhau để thực hiện sáng chế mà vẫn không bị coi là nằm ngoài phạm vi của sáng chế, như được xác định dựa vào các điểm yêu cầu bảo hộ kèm theo. Ví dụ, các phương án làm ví dụ thực hiện sáng chế được mô tả liên quan đến phương pháp mã hoá và giải mã BCH và phương pháp mã hoá và giải mã LDPC. Tuy nhiên, các phương án đó không giới hạn phạm vi của sáng chế chỉ ở một phương pháp mã hoá và giải mã cụ thể, mà thay vì thế, sáng chế có thể được áp dụng cho nhiều loại phương pháp mã hoá và giải mã khác nhau với những thay đổi cần

thiết. Các phương án thay đổi như vậy cũng được coi là nằm trong phạm vi của sáng chế.

YÊU CẦU BẢO HỘ

1. Thiết bị truyền tín hiệu phát rộng truyền hình (Television, TV) để truyền thông tin báo hiệu cho tầng 1 (L1) và dữ liệu phát rộng TV, thiết bị truyền tín hiệu phát rộng TV này có thể hoạt động ở một chế độ trong số nhiều chế độ, thiết bị truyền tín hiệu này bao gồm:

bộ mã hoá được tạo cấu hình để mã hoá các bit thông tin gồm có các bit đầu vào để tạo ra các bit chẵn lẻ, trong đó các bit đầu vào này được xác định dựa vào thông tin báo hiệu cho tầng L1 đối với dữ liệu phát rộng TV;

bộ phân lập được tạo cấu hình để gắn một hoặc nhiều bit trong số các bit chẵn lẻ được tạo ra bằng cách lặp lại ở giữa các bit thông tin và các bit chẵn lẻ được tạo ra, theo một chế độ định trước trong số nhiều chế độ;

bộ phân đục lỗ được tạo cấu hình để đục lỗ một hoặc nhiều bit trong số các bit chẵn lẻ được tạo ra;

bộ tạo bit chẵn lẻ bổ sung được tạo cấu hình để tính số lượng bit chẵn lẻ bổ sung dựa vào số lượng bit của một hoặc nhiều bit được gắn vào và tạo ra các bit chẵn lẻ bổ sung dựa vào số lượng đã tính;

bộ tạo tín hiệu được tạo cấu hình để tạo ra tín hiệu phát rộng TV dựa vào khung thứ nhất và khung thứ hai, trong đó khung thứ nhất có các bit chẵn lẻ bổ sung, và khung thứ hai có các bit đầu vào, các bit chẵn lẻ còn lại trong số các bit chẵn lẻ được tạo ra sau khi đục lỗ và một hoặc nhiều bit chẵn lẻ được gắn vào; và

bộ truyền được tạo cấu hình để truyền tín hiệu phát rộng TV,

trong đó khung thứ nhất được truyền trước khung thứ hai.

2. Thiết bị truyền tín hiệu theo điểm 1, trong đó thiết bị truyền tín hiệu này còn bao

gồm bộ phận ánh xạ được tạo cấu hình để điều biến các bit chẵn lẻ bổ sung dựa vào bậc điều biến.

3. Thiết bị truyền tín hiệu theo điểm 1, trong đó số lượng bit chẵn lẻ bổ sung được tính dựa vào biểu thức dưới đây:

$$N_{AP} = \left\lfloor \frac{N_{AP_temp}}{\eta_{MOD}} \right\rfloor \times \eta_{MOD},$$

trong đó N_{AP} là số lượng bit chẵn lẻ bổ sung và N_{AP_temp} là số lượng tạm thời của các bit chẵn lẻ bổ sung, và η_{MOD} là bậc điều biến.

4. Thiết bị truyền tín hiệu theo điểm 3, trong đó số lượng tạm thời của các bit chẵn lẻ bổ sung được tính dựa vào biểu thức dưới đây:

$$N_{AP_temp} = \min \left\{ \begin{array}{l} 0,5 \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat}) \\ (N_{ldpc_parity} + N_{punc} + N_{repeat}) \end{array} \right\}, K = 0, 1, 2,$$

trong đó, N_{AP_temp} là số lượng tạm thời của các bit chẵn lẻ bổ sung, min là hàm chọn số nhỏ nhất, và N_{ldpc_parity} là số lượng bit chẵn lẻ, N_{punc} là số lượng bit của một hoặc nhiều bit được đục lỗ, N_{outer} là số lượng bit đầu vào, và N_{repeat} là số lượng bit của một hoặc nhiều bit được gán vào.

5. Thiết bị thu tín hiệu có thể hoạt động ở một chế độ trong số nhiều chế độ, thiết bị thu tín hiệu này bao gồm:

bộ thu được tạo cấu hình để thu tín hiệu phát rộng truyền hình (TV) có khung thứ nhất và khung thứ hai từ thiết bị truyền tín hiệu;

bộ giải điều biến được tạo cấu hình để giải điều biến dữ liệu của từ mã thứ nhất có trong khung thứ nhất để tạo ra các giá trị thứ nhất, và giải điều biến dữ liệu của từ mã thứ hai có trong khung thứ hai để tạo ra các giá trị thứ hai;

bộ phận chèn được tạo cấu hình để chèn các giá trị định trước;

bộ phận kết hợp được tạo cấu hình để, nếu chế độ hoạt động là chế độ định trước trong số nhiều chế độ, kết hợp các giá trị thứ ba được chọn trong số các giá trị thứ nhất và các giá trị định trước được chèn với các giá trị thứ tư được chọn trong số các giá trị thứ nhất và các giá trị định trước được chèn và kết hợp các giá trị thứ năm được chọn trong số các giá trị thứ hai với các giá trị thứ sáu được chọn trong số các giá trị thứ nhất và các giá trị định trước được chèn; và

bộ giải mã được tạo cấu hình để giải mã các giá trị đã được cộng và các giá trị còn lại trong số các giá trị thứ nhất và các giá trị định trước được chèn dựa vào mã kiểm tra chẵn lẻ mật độ thấp (Low Density Parity Check, LDPC) để tạo ra các bit đầu ra tương ứng với tín hiệu phát rộng TV,

trong đó nếu chế độ hoạt động của thiết bị truyền tín hiệu là chế độ định trước, từ mã thứ nhất được xác định dựa vào các bit chẵn lẻ còn lại sau khi đục lỗ một hoặc nhiều bit chẵn lẻ trong số các bit chẵn lẻ được tạo ra dựa vào các bit thông tin, và các bit chẵn lẻ lặp lại được chọn trong số các bit chẵn lẻ được tạo ra, và từ mã thứ hai được xác định dựa vào một hoặc nhiều bit chẵn lẻ được chọn trong số các bit chẵn lẻ lặp lại và các bit chẵn lẻ được tạo ra,

trong đó số lượng bit của một hoặc nhiều bit chẵn lẻ được chọn được tính dựa vào số lượng bit chẵn lẻ lặp lại,

trong đó các giá trị thứ ba tương ứng với các bit chẵn lẻ lặp lại,

trong đó các giá trị thứ năm tương ứng với một hoặc nhiều bit chẵn lẻ được chọn,

trong đó các giá trị định trước tương ứng với một hoặc nhiều bit chẵn lẻ được đục lỗ, và

trong đó khung thứ hai được thu trước khung thứ nhất.

6. Thiết bị thu tín hiệu theo điểm 5, trong đó một hoặc nhiều bit chẵn lẻ được chọn được tính dựa vào biểu thức dưới đây:

$$N_{AP} = \left\lfloor \frac{N_{AP_temp}}{\eta_{MOD}} \right\rfloor \times \eta_{MOD},$$

trong đó N_{AP} là số lượng bit của một hoặc nhiều bit chẵn lẻ được chọn và N_{AP_temp} là số lượng tạm thời của một hoặc nhiều bit chẵn lẻ được chọn, và η_{MOD} là bậc điều biến ở trong thiết bị truyền tín hiệu.

7. Thiết bị thu tín hiệu theo điểm 6, trong đó số lượng tạm thời của một hoặc nhiều bit chẵn lẻ được chọn được tính dựa vào biểu thức dưới đây:

$$N_{AP_temp} = \min \left\{ \begin{array}{l} 0,5 \times K \times (N_{outer} + N_{ldpc_parity} - N_{punc} + N_{repeat}) \\ (N_{ldpc_parity} + N_{punc} + N_{repeat}) \end{array} \right\}, K = 0, 1, 2,$$

trong đó, N_{AP_temp} là số lượng tạm thời của một hoặc nhiều bit chẵn lẻ được chọn, min là hàm chọn số nhỏ nhất, và N_{ldpc_parity} là số lượng bit chẵn lẻ được tạo ra, N_{punc} là số lượng bit của một hoặc nhiều bit chẵn lẻ được đục lỗ, N_{outer} là số lượng bit đầu vào có trong các bit thông tin, và N_{repeat} là số lượng bit chẵn lẻ lặp lại.

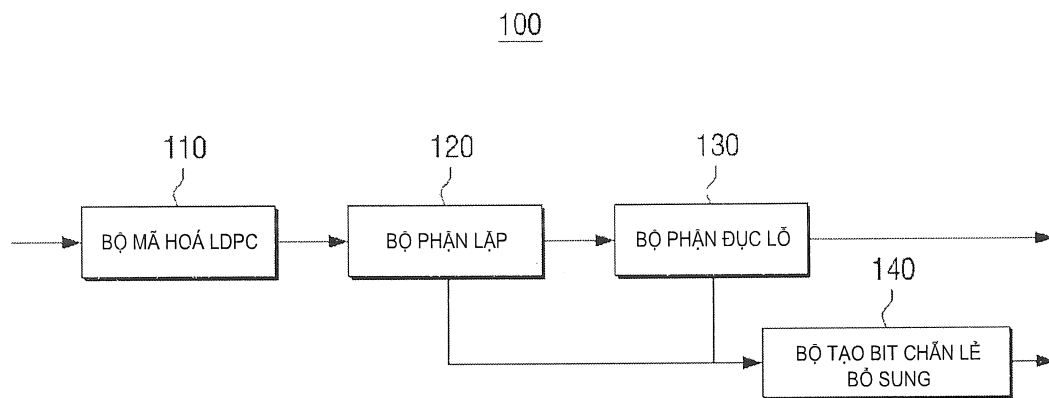
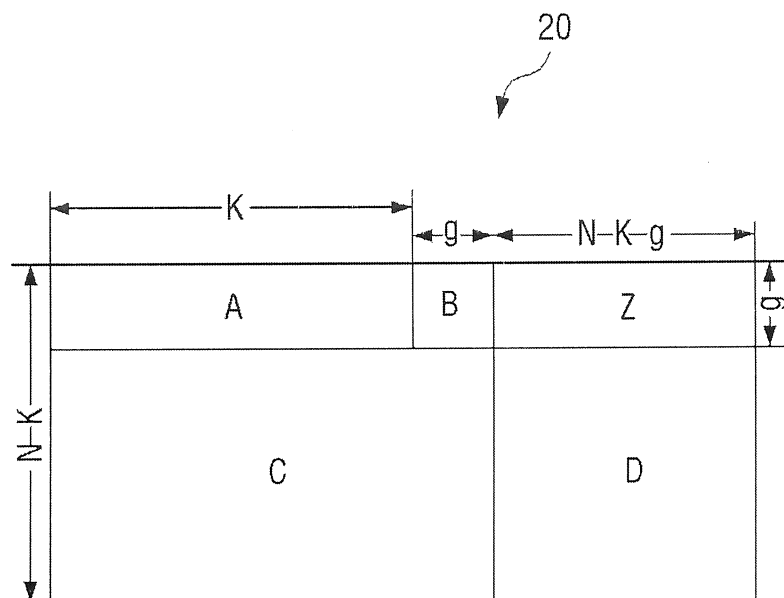
Fig. 1**Fig. 2**

Fig. 3

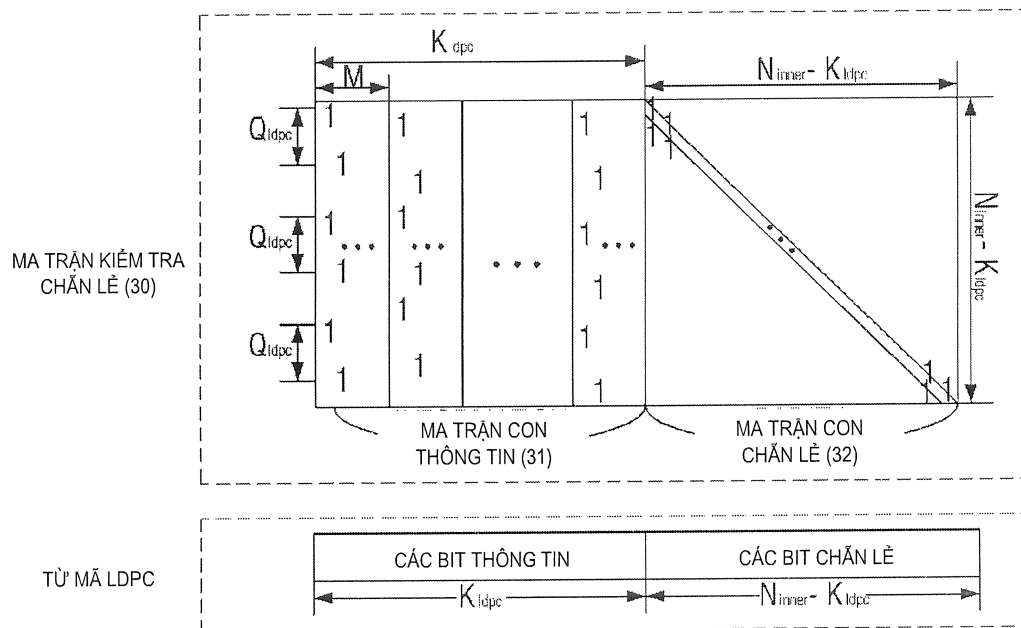


Fig. 4

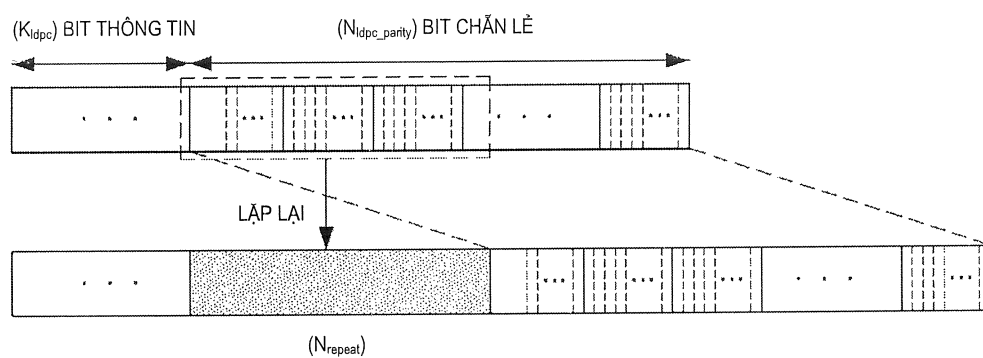


Fig. 5

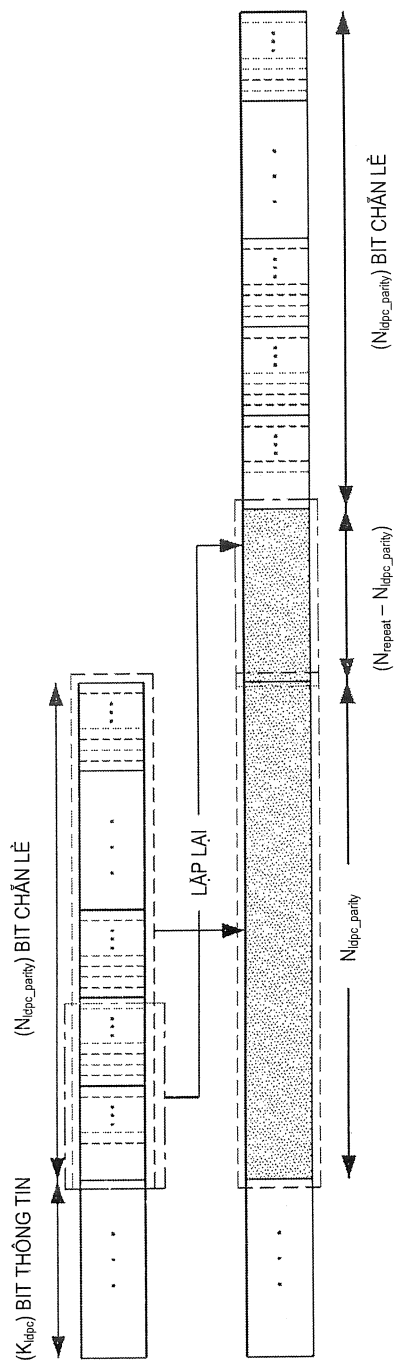


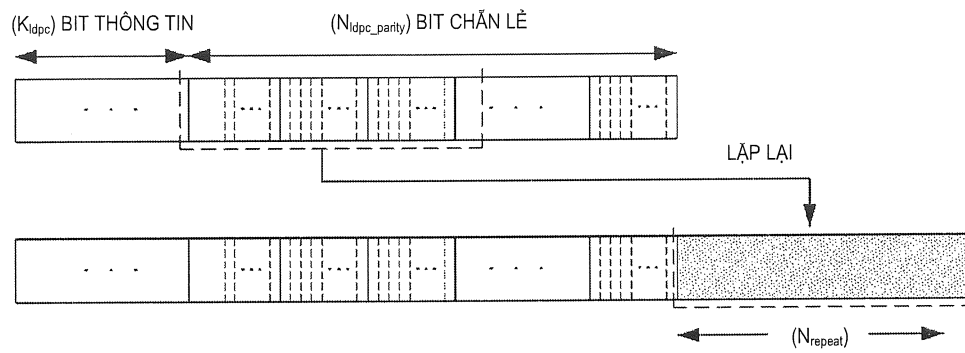
Fig. 6

Fig. 7

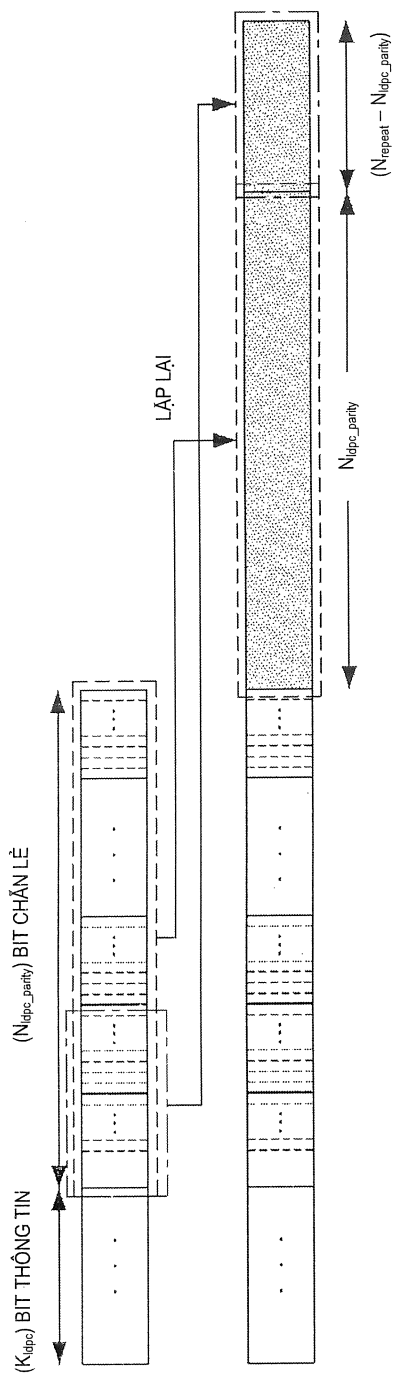


Fig. 8

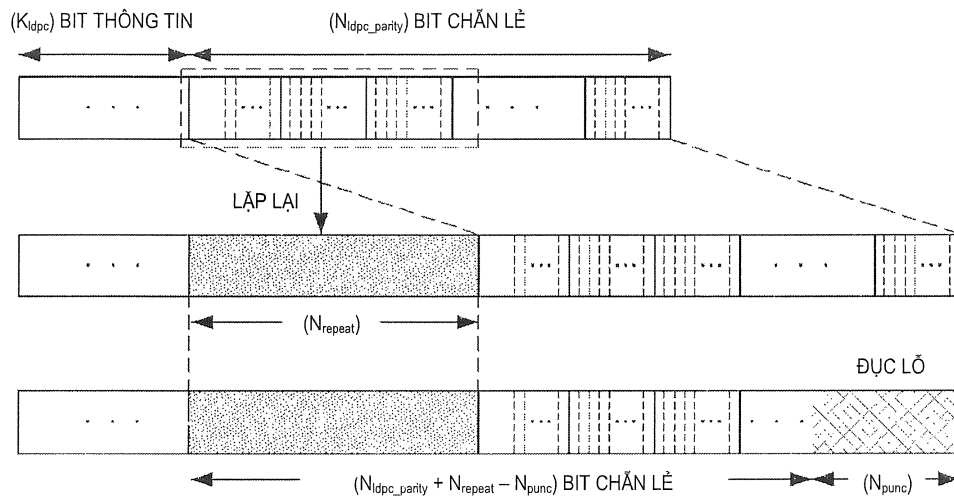


Fig. 9

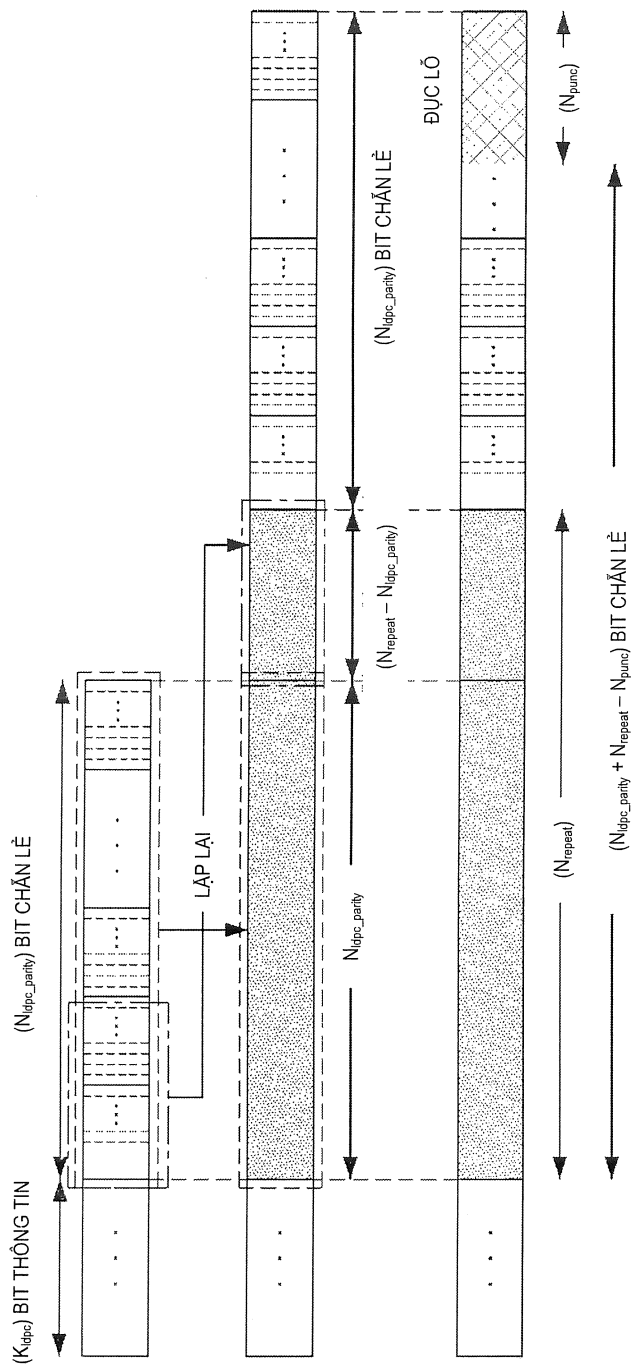


Fig. 10

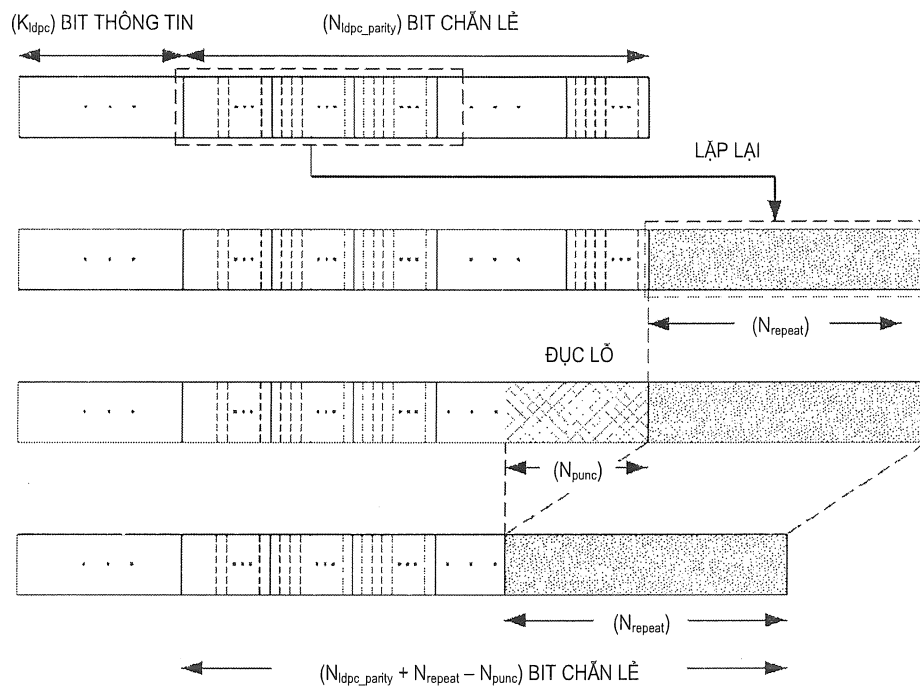


Fig. 11

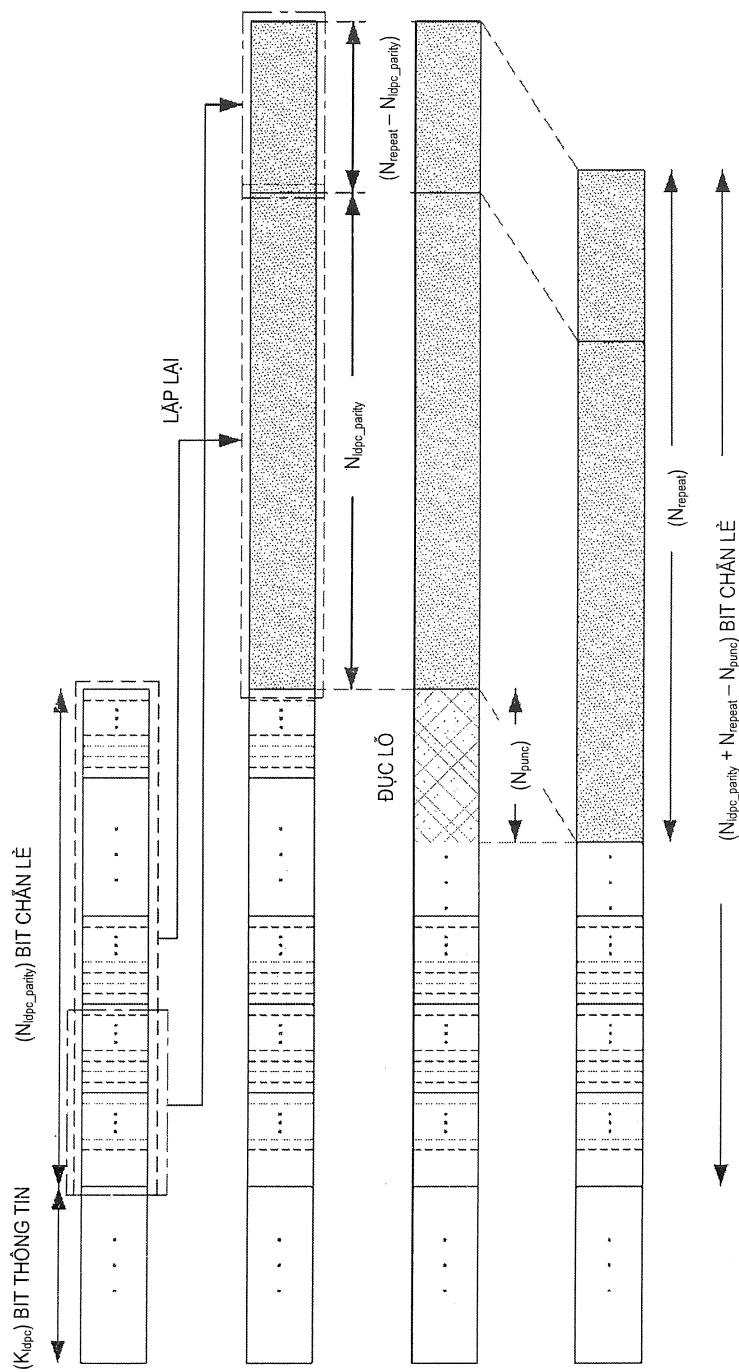


Fig. 12

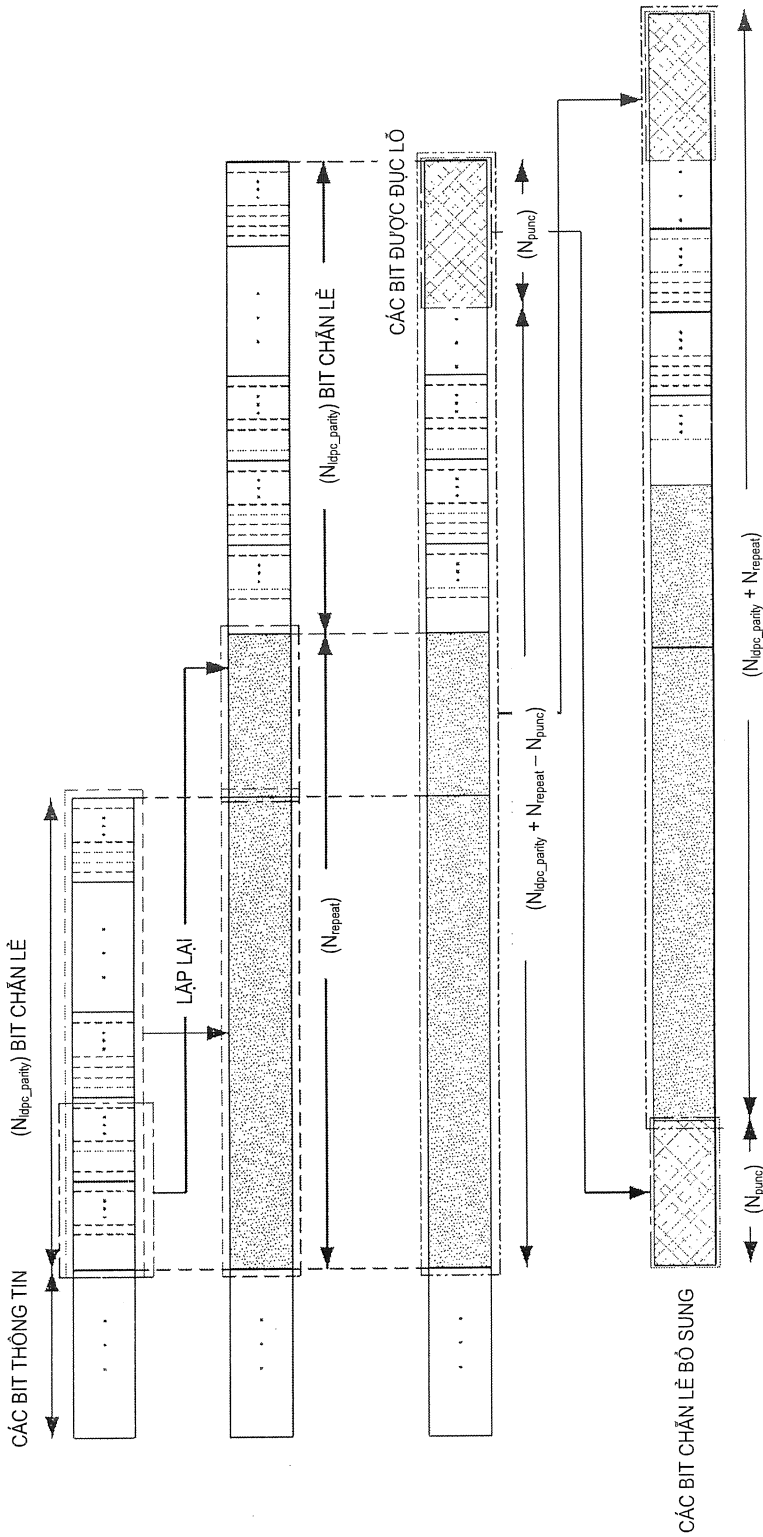


Fig. 13

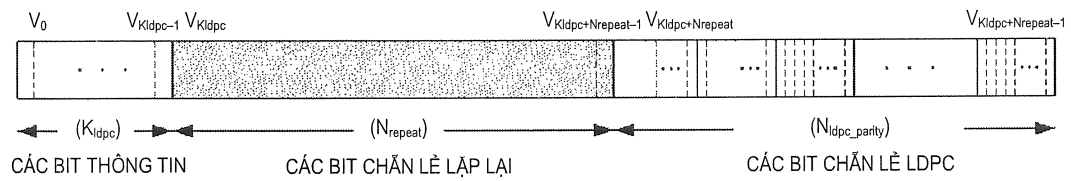
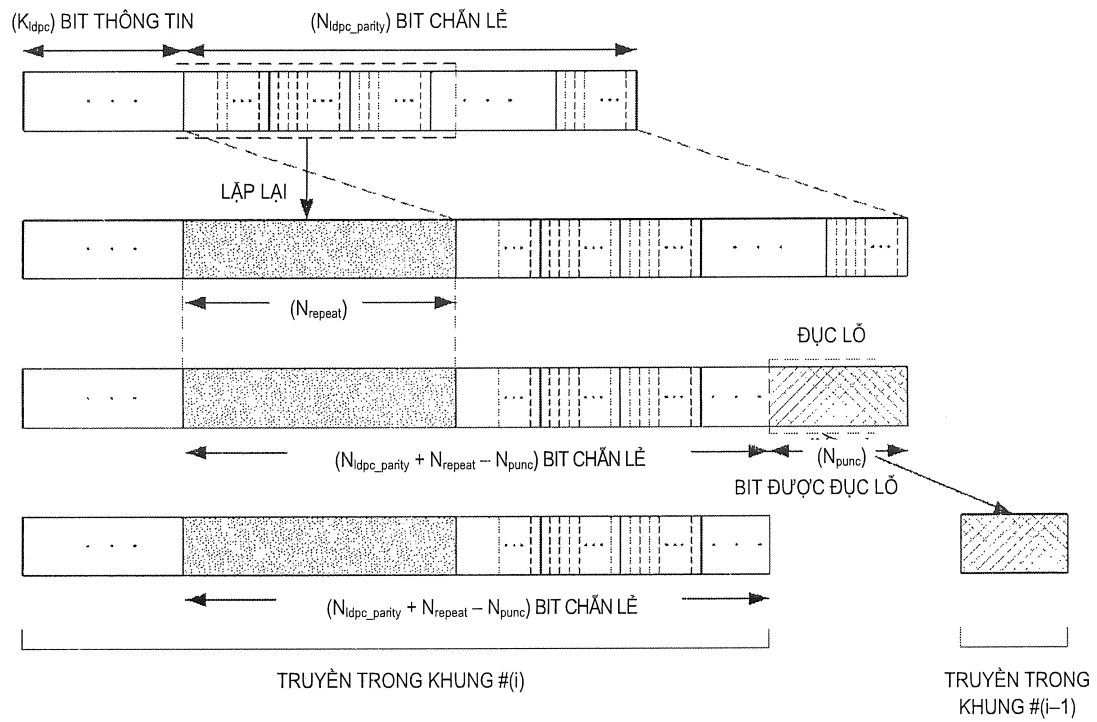


Fig. 14



The diagram illustrates the transmission of a frame in a TDMA system. It shows the structure of a frame and the transmission of multiple frames.

Frame Structure:

- Header:** Contains the frame number (K_{tdpc}) and the parity bit (N_{tdpc_parity}) .
- Data Field:** Contains the data bits (N_{data}) .
- Parity Field:** Contains the parity bits (N_{punc}) .
- Total Frame Size:** $N_{frame} = N_{data} + N_{tdpc_parity} + N_{punc}$.

Transmission:

- The frame is repeated N_{repeat} times.
- The total transmission size is $N_{repeat} \times N_{frame} + N_{punc}$.
- The diagram also shows the transmission of multiple frames, with the total transmission size being $N_{repeat} \times N_{frame} + N_{punc}$.

Fig. 16

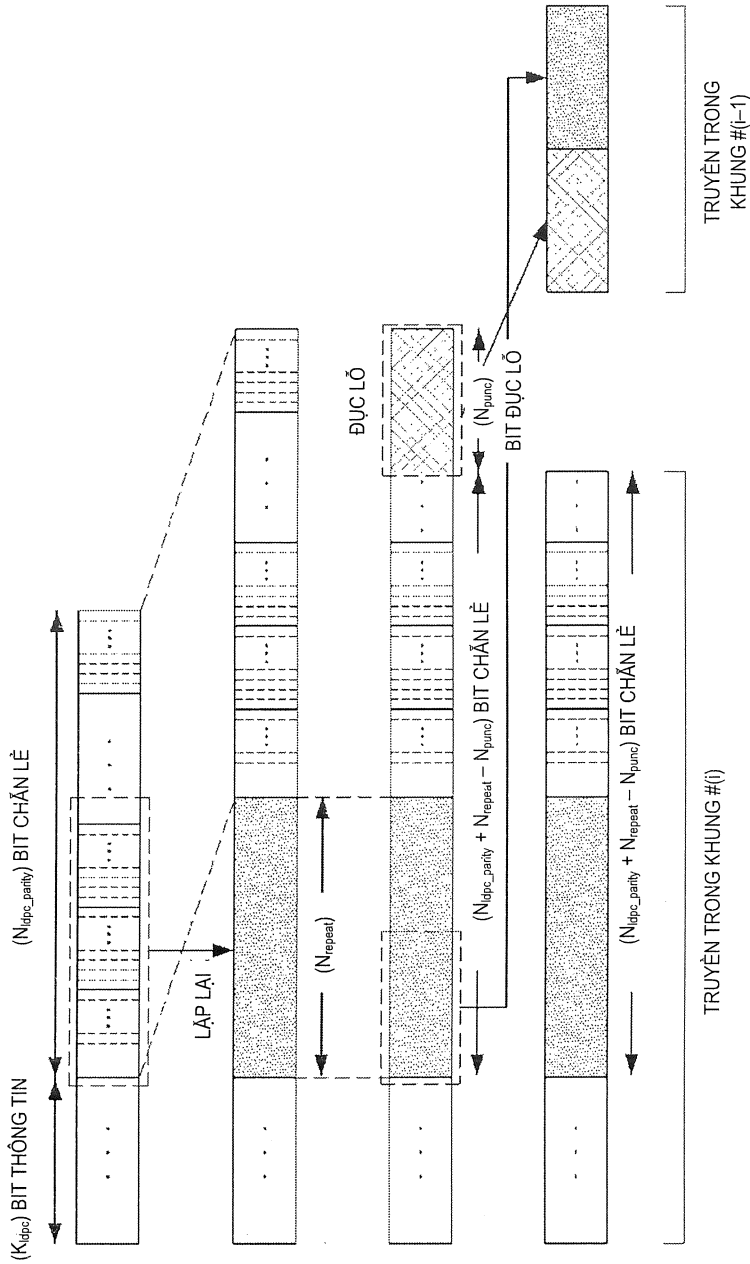


Fig. 17

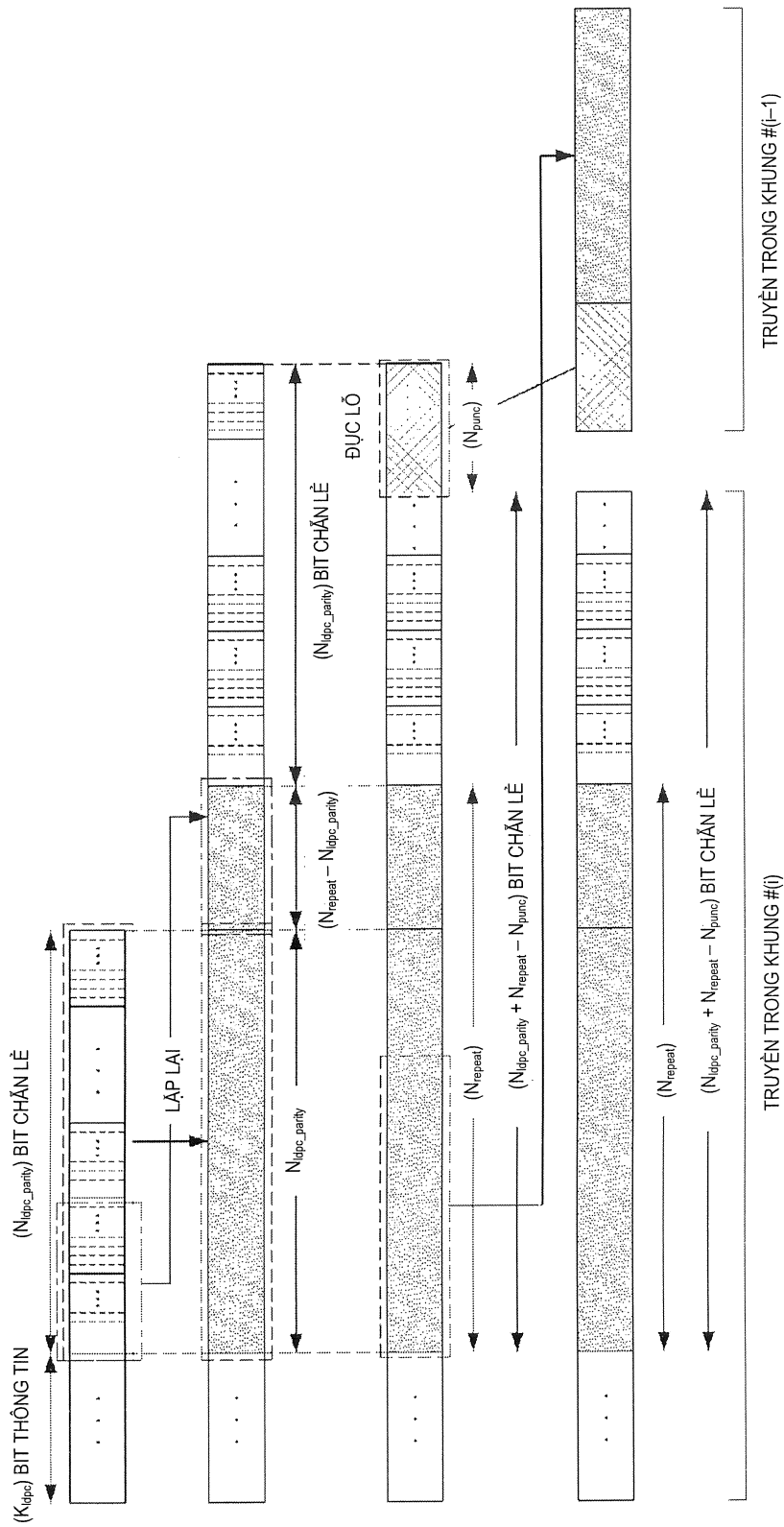


Fig. 18

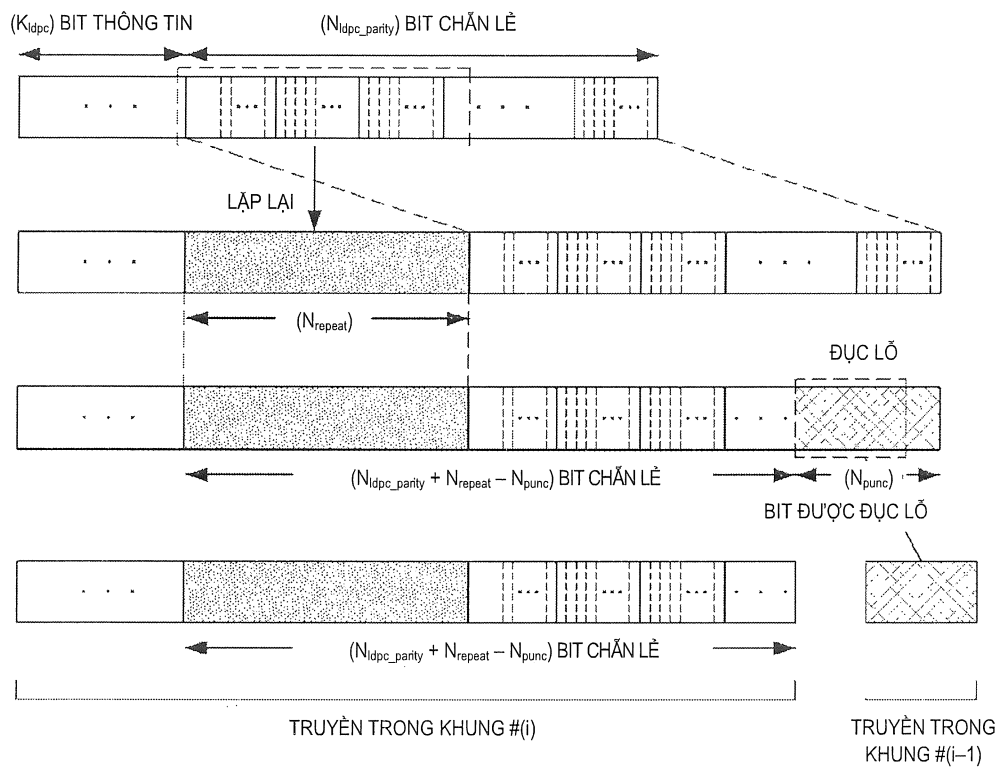


Fig. 19

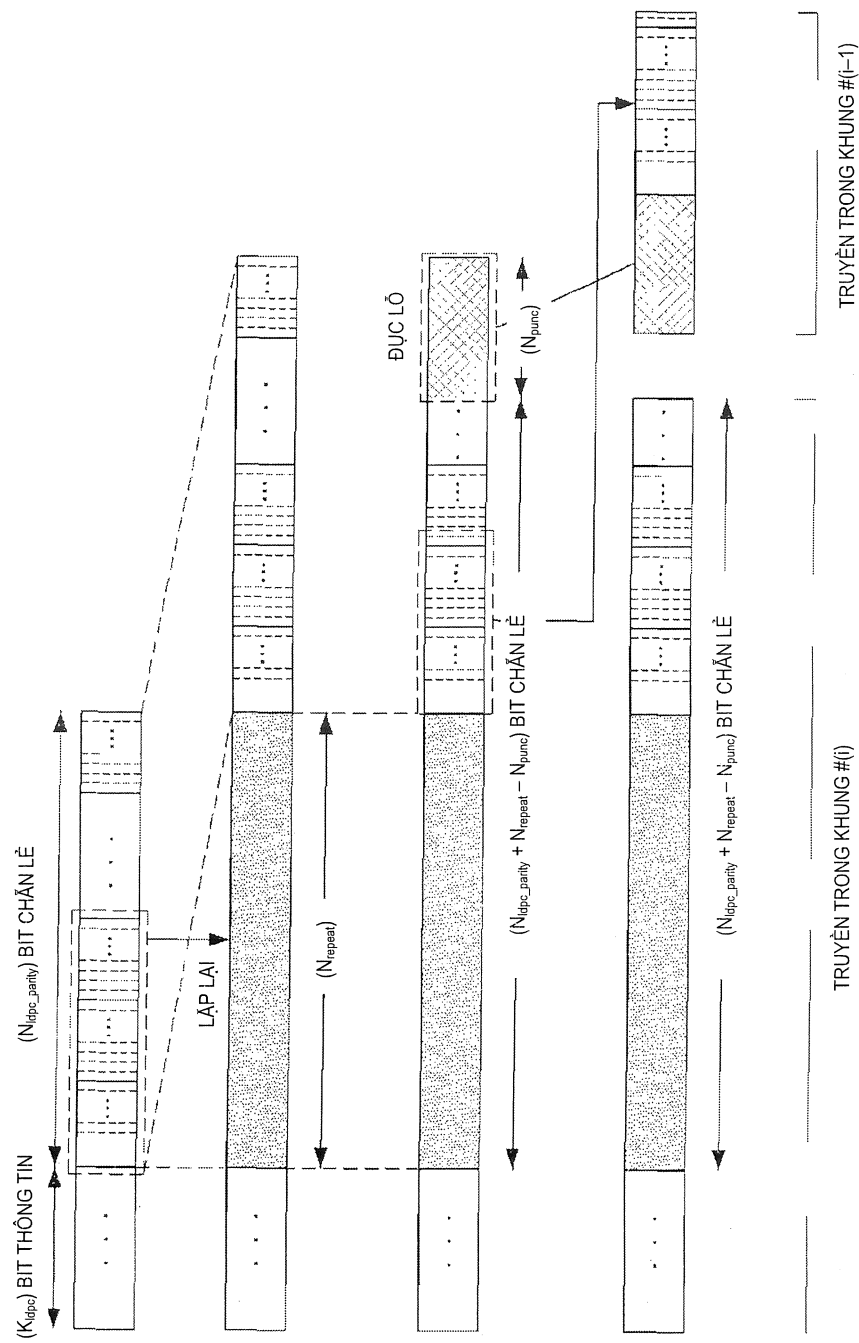


Fig. 20

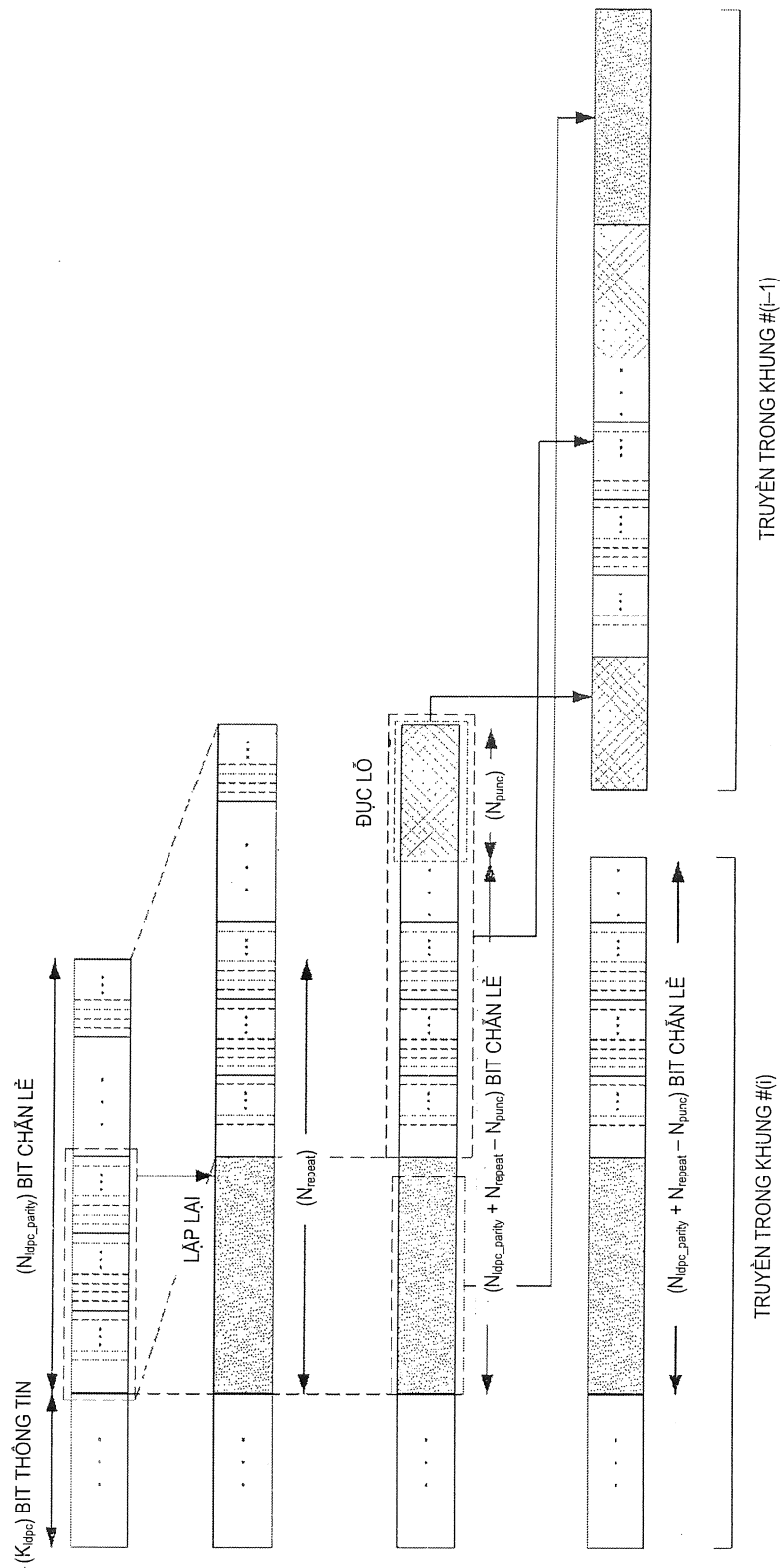


Fig. 21

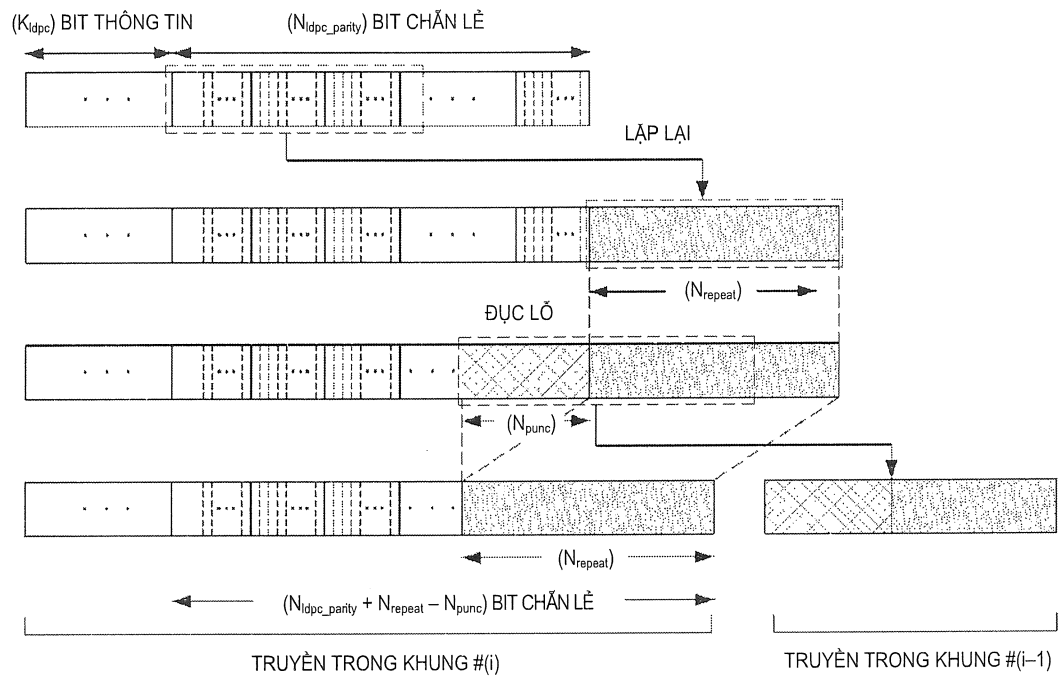


Fig. 22

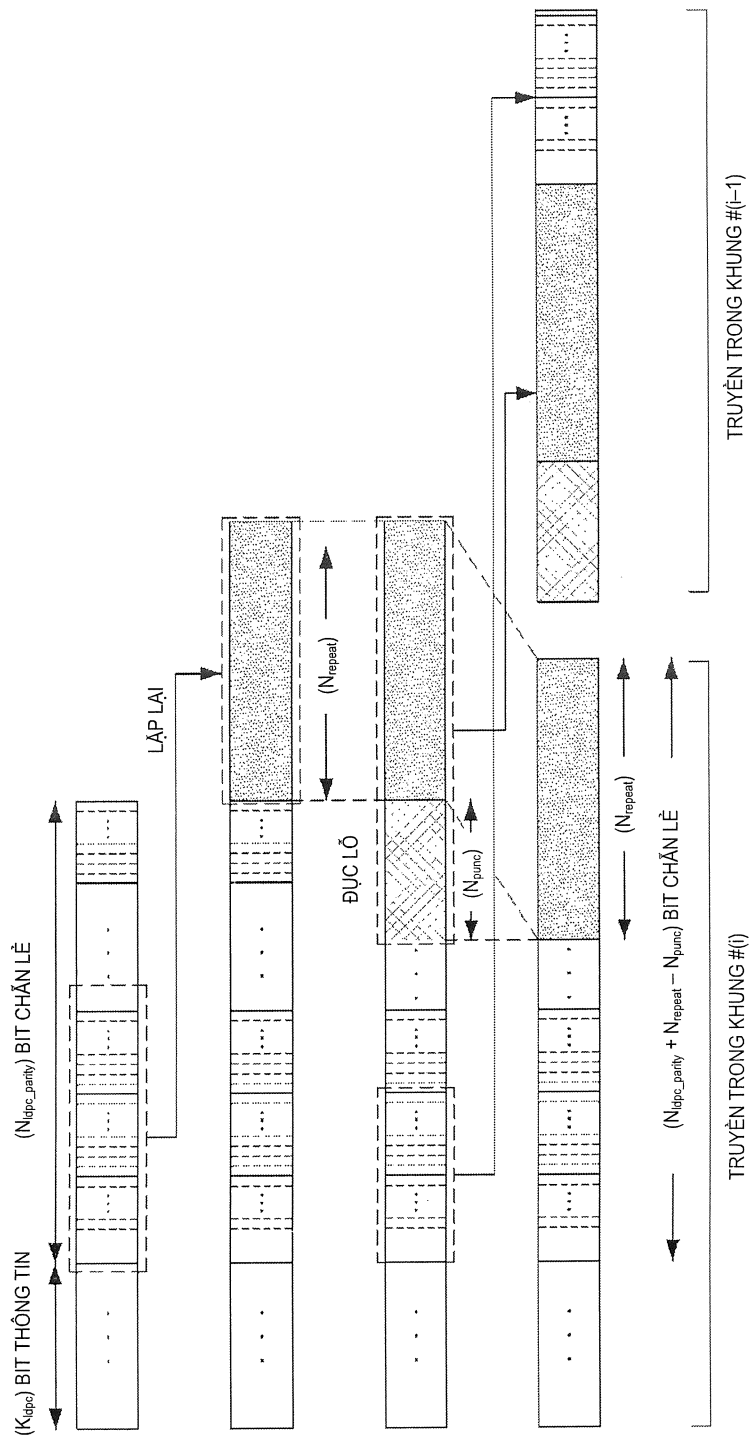


Fig. 23

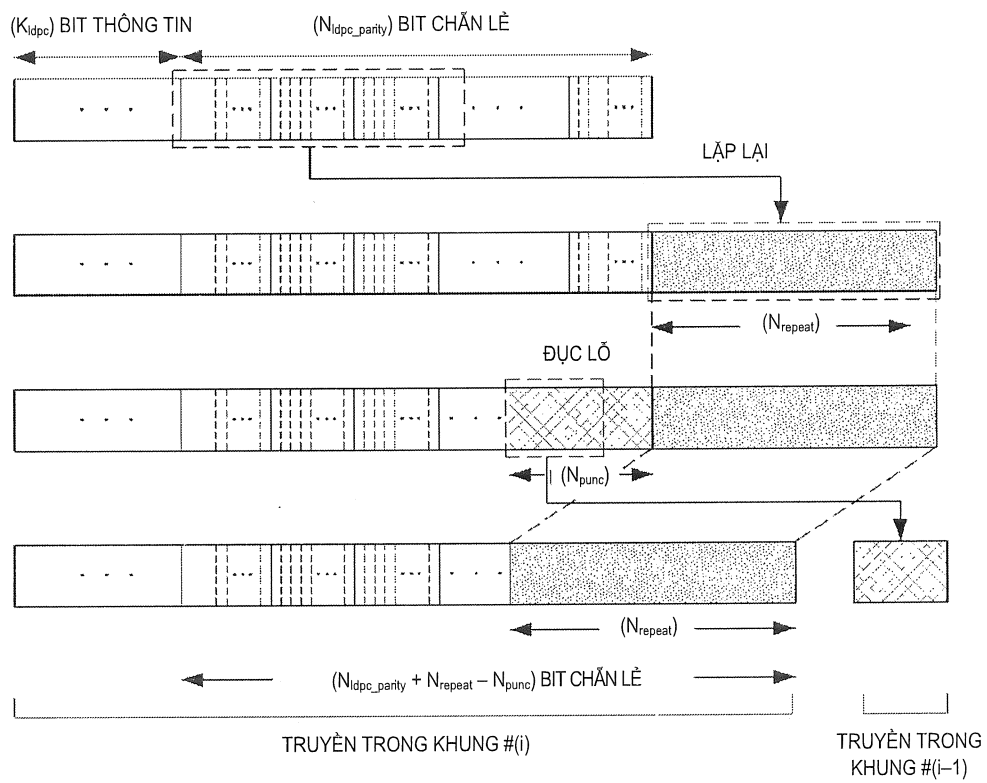


Fig. 24

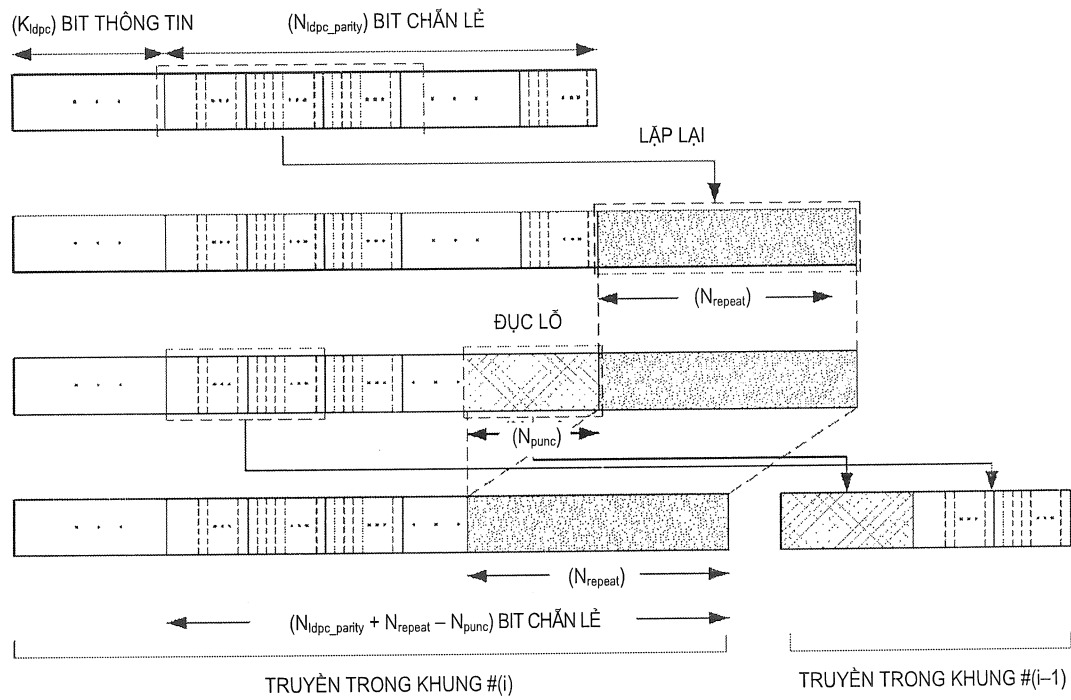


Fig. 25

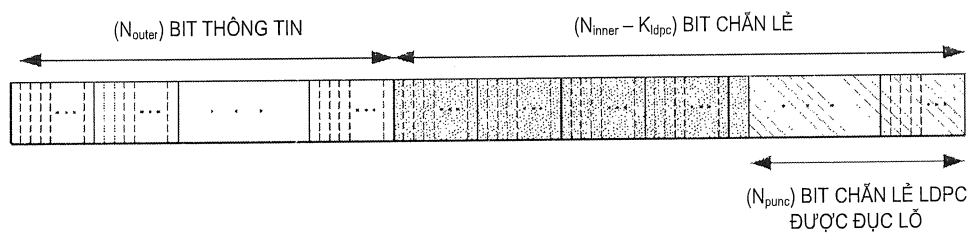


Fig. 26

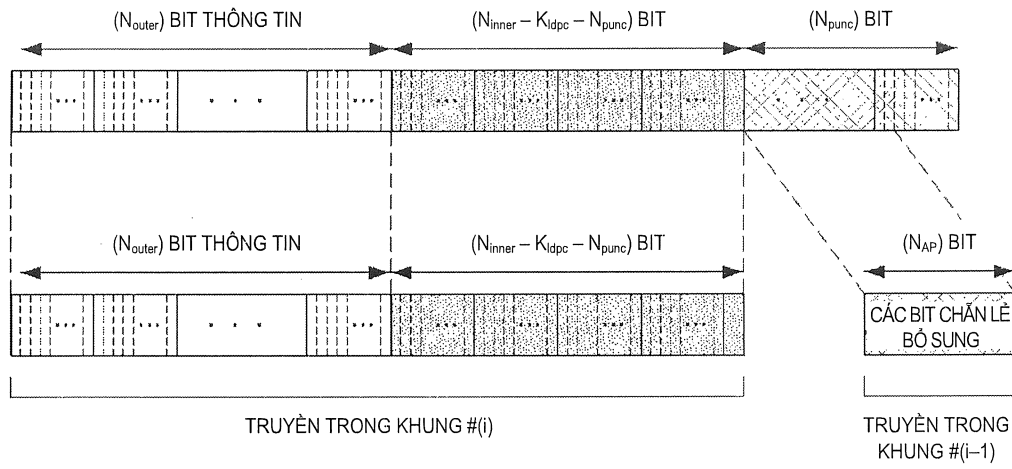


Fig. 27

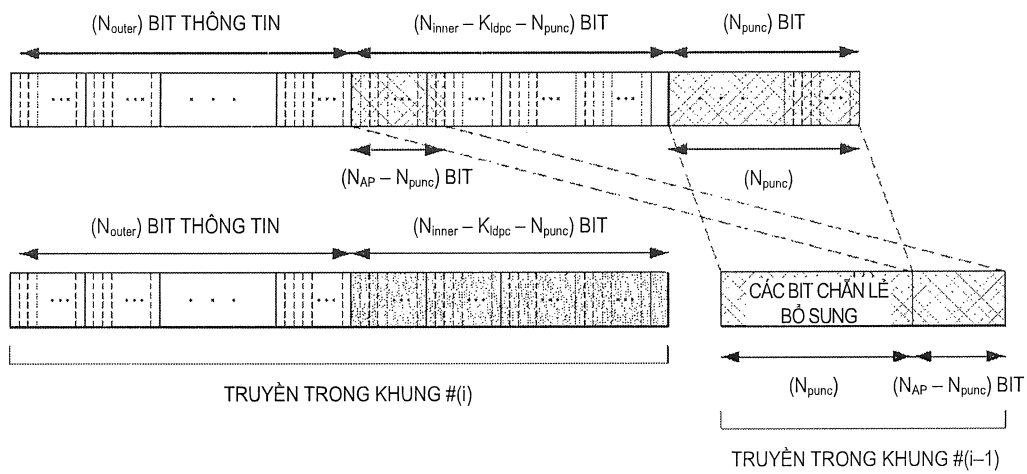


Fig. 28

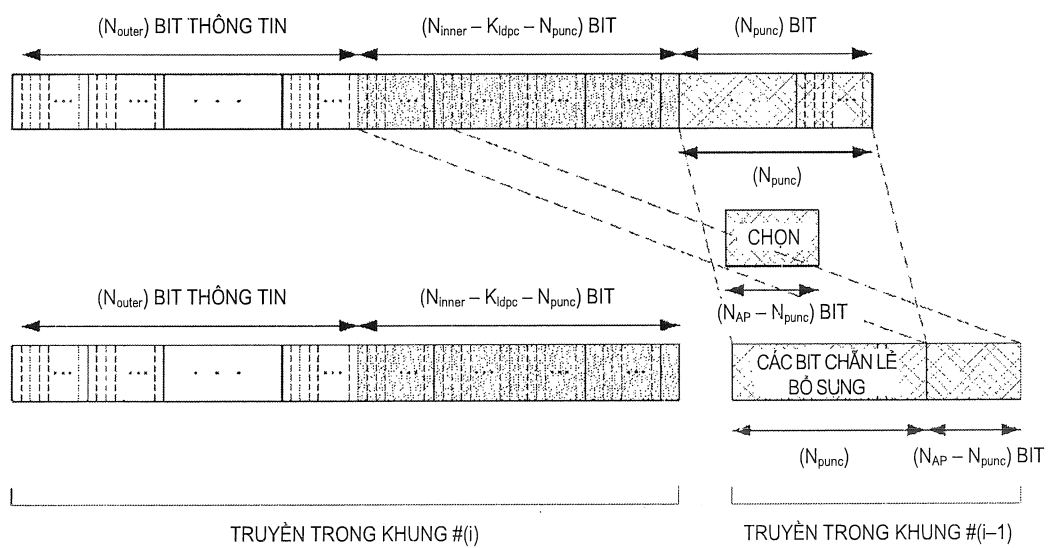


Fig. 29

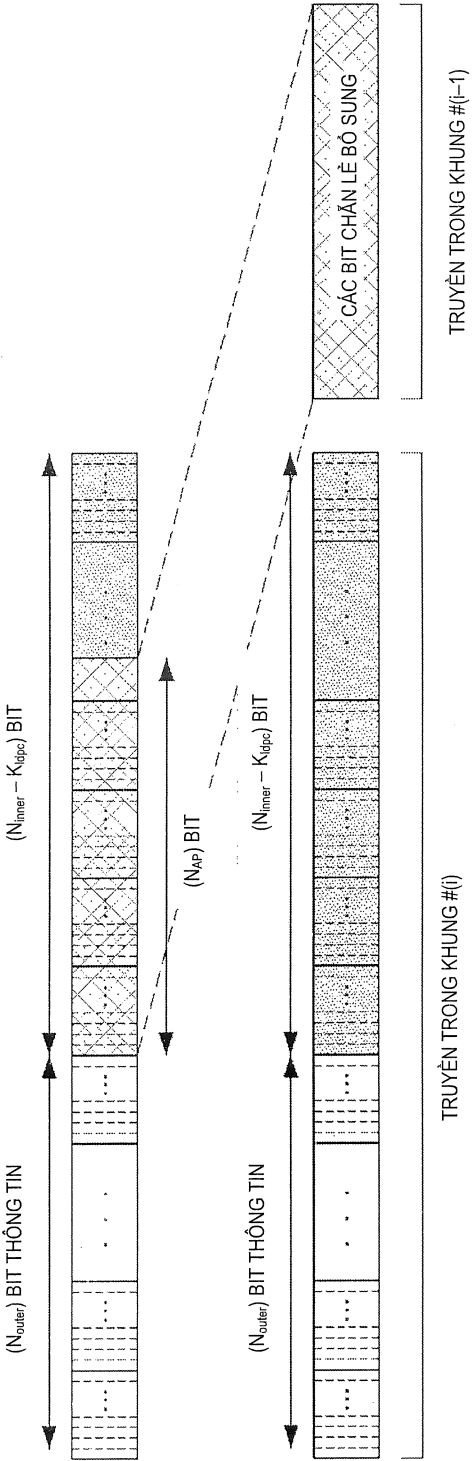


Fig. 30

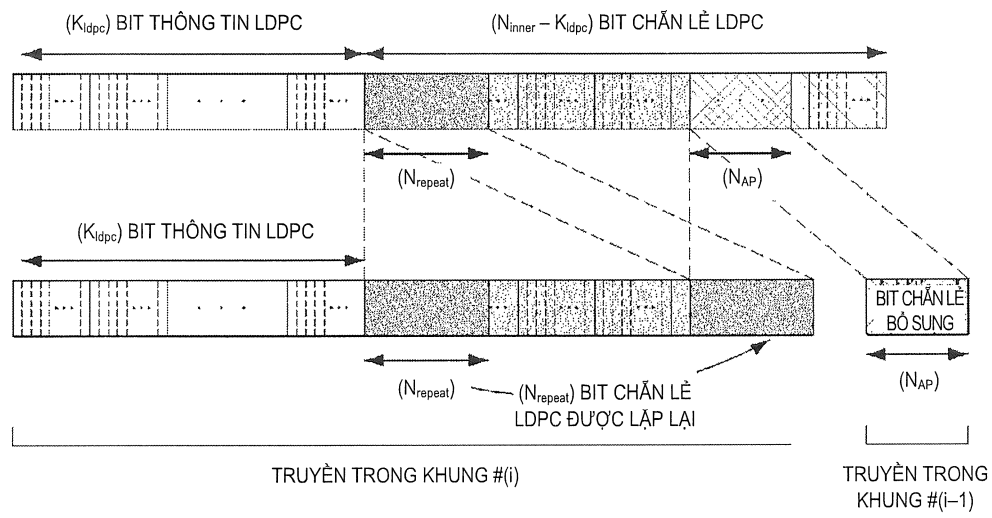


Fig. 31

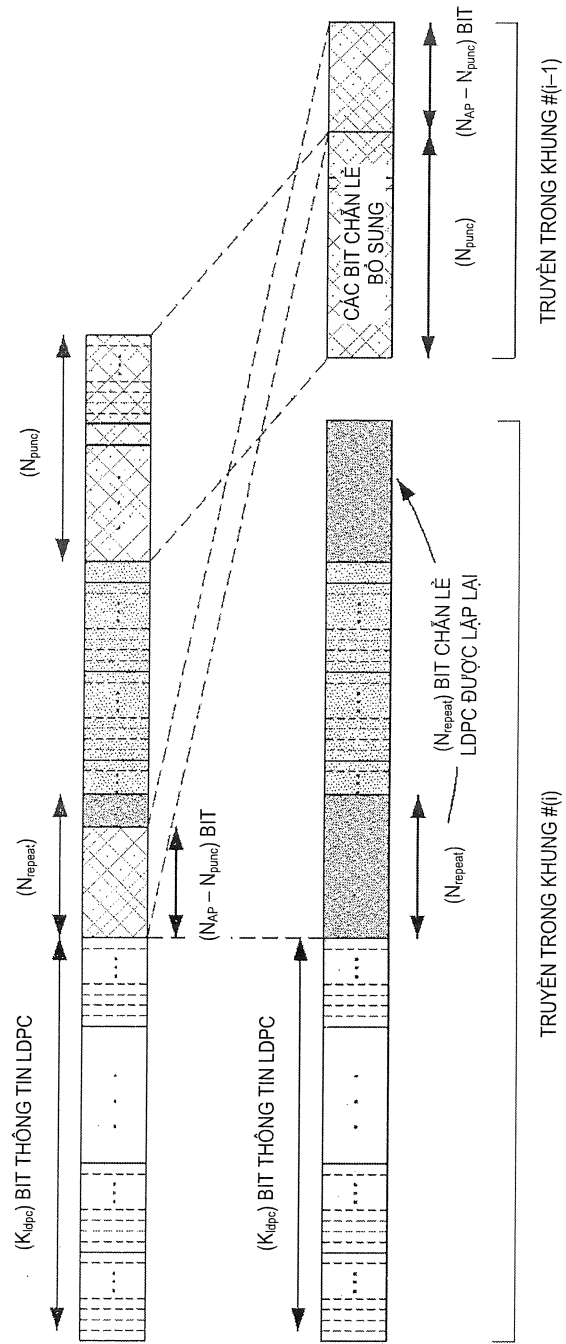


Fig. 32

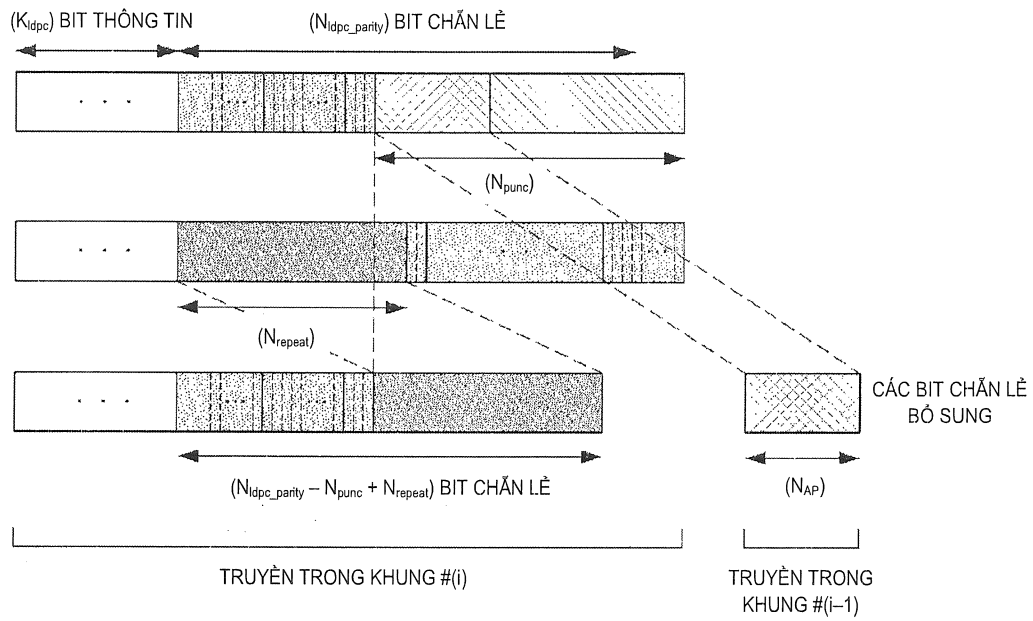


Fig. 33

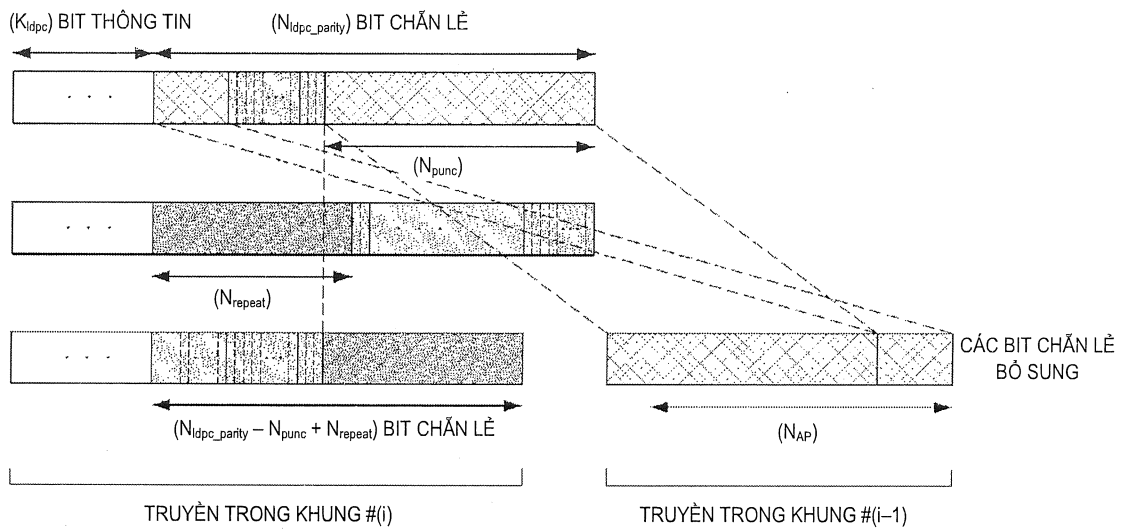


Fig. 34

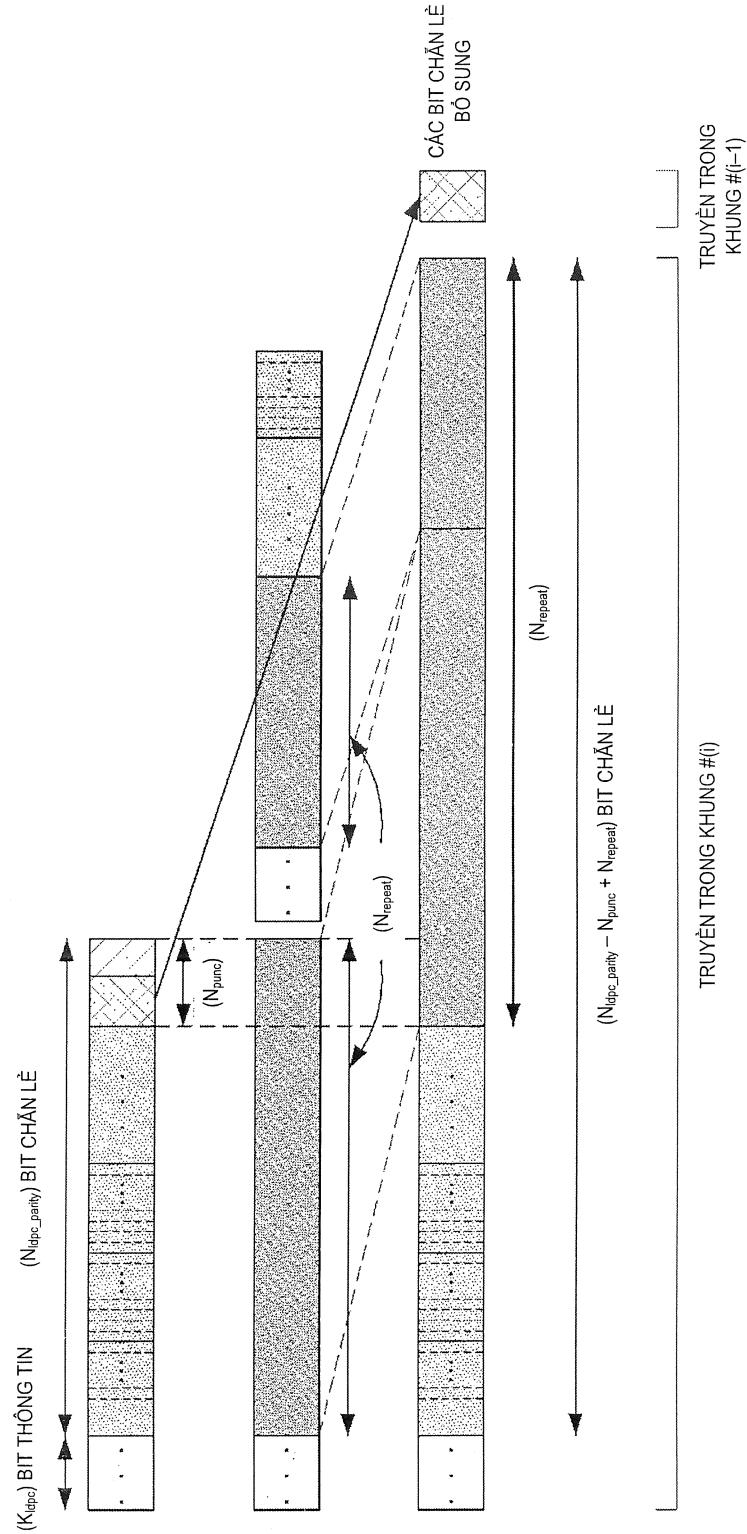


Fig. 35

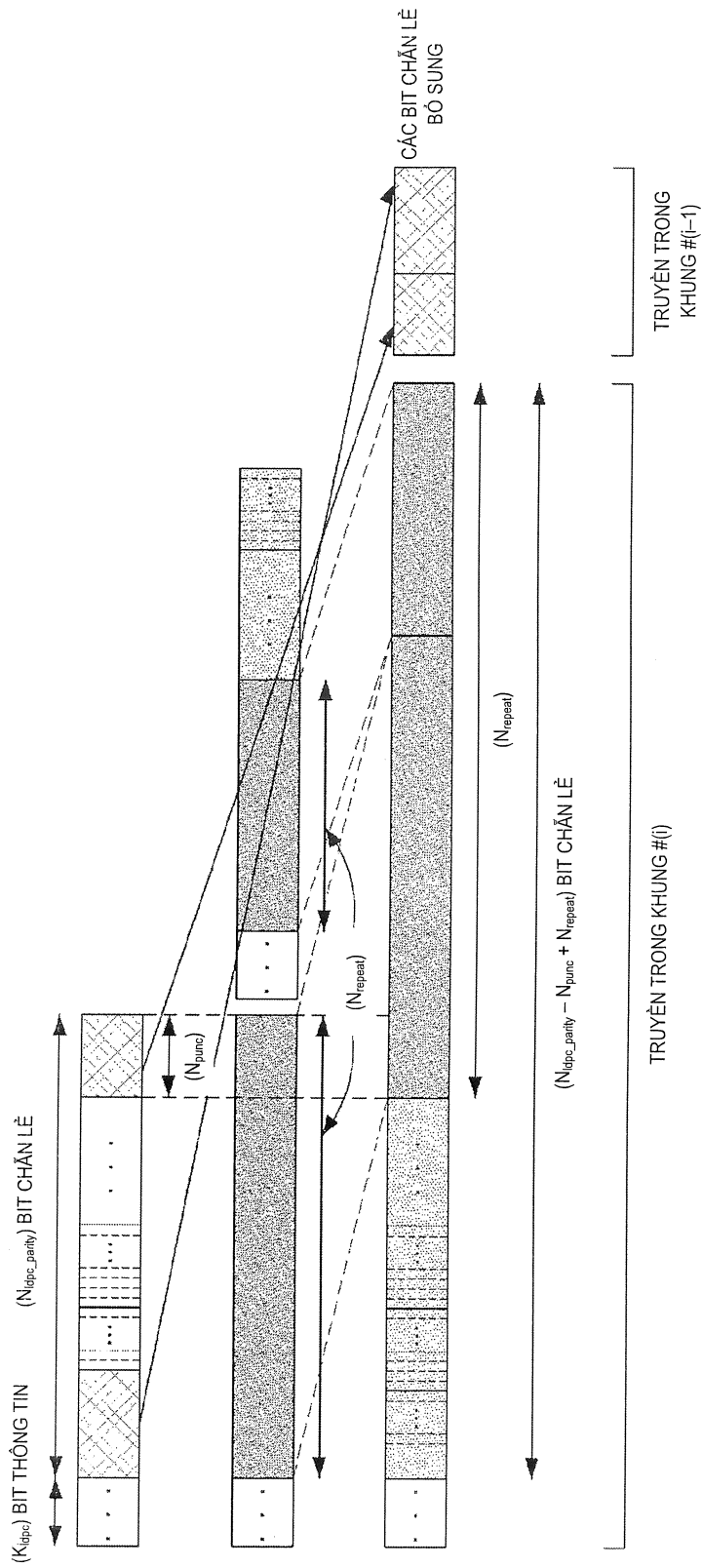


Fig. 36

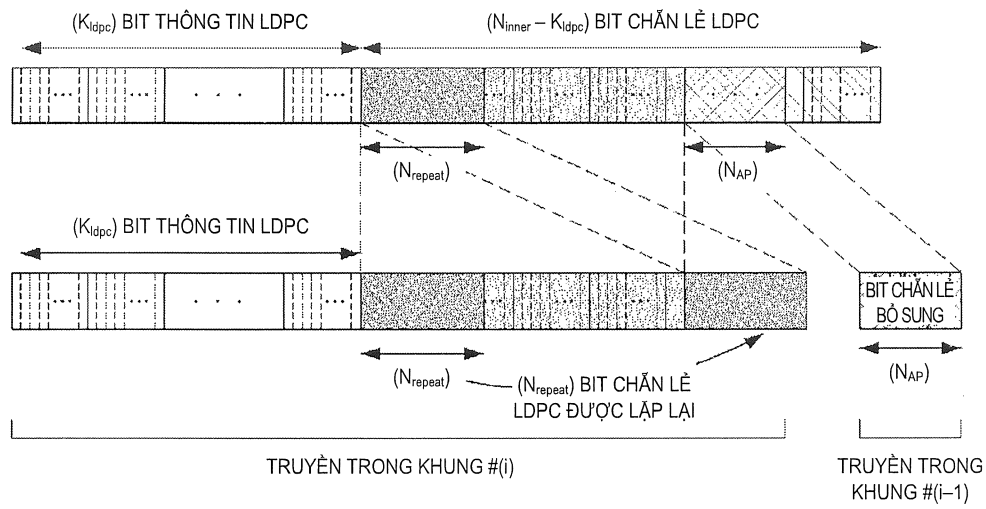


Fig. 37

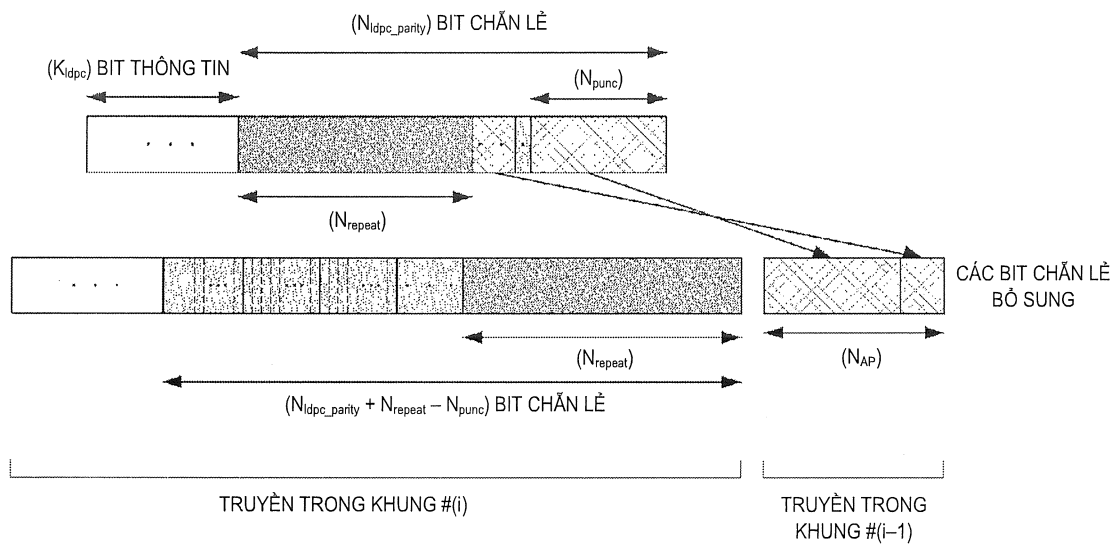


Fig. 38

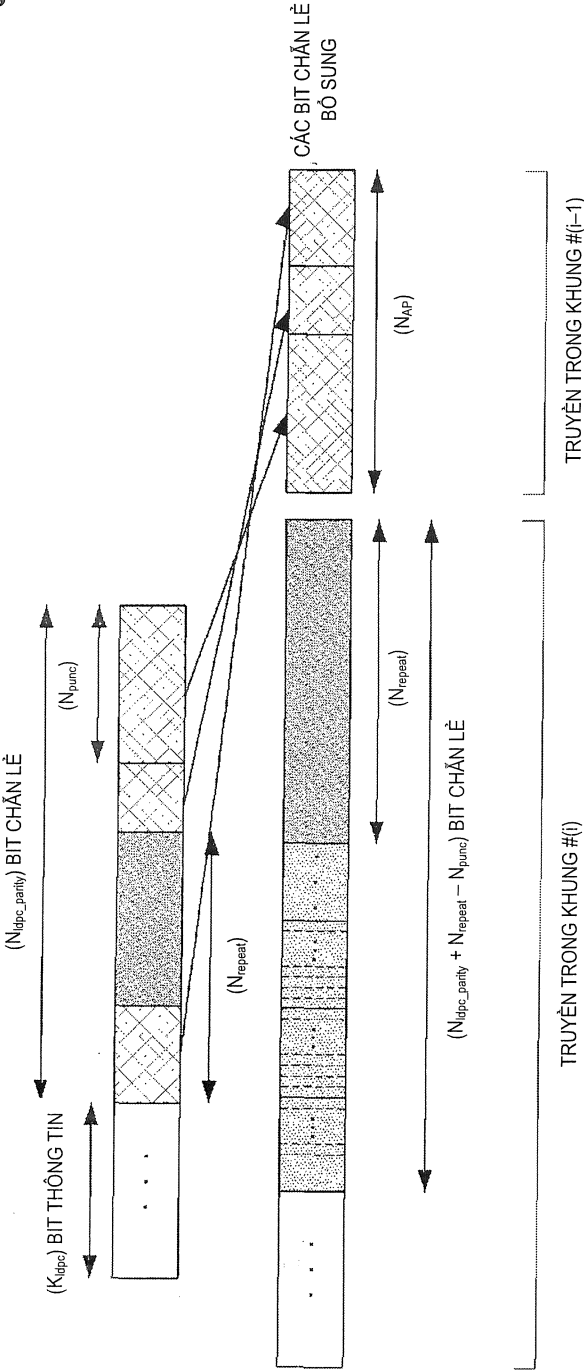


Fig. 39

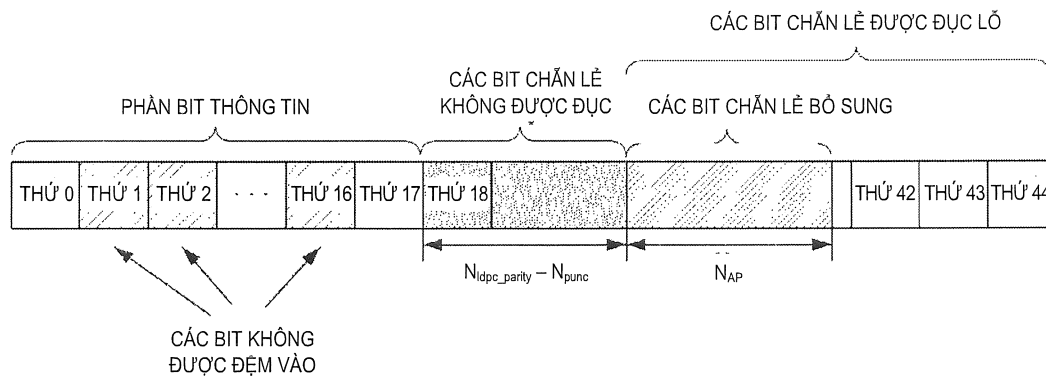


Fig. 40

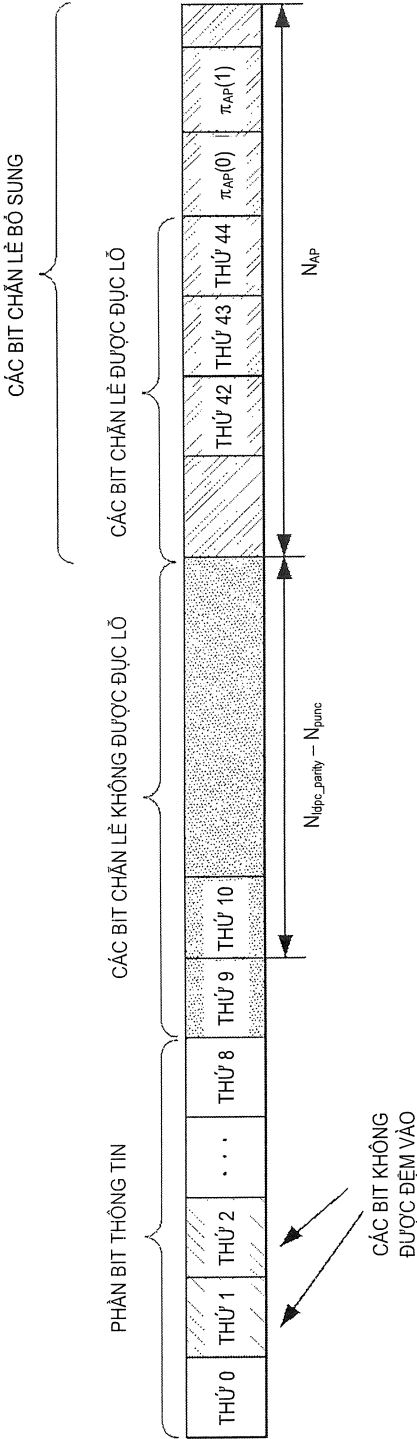


Fig. 41

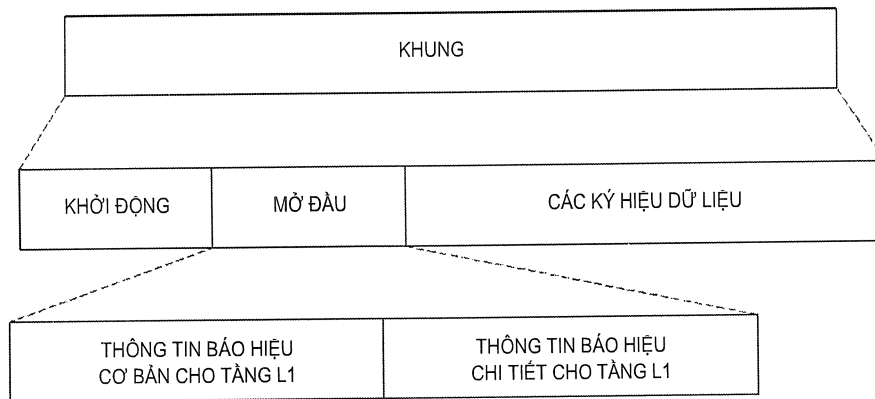


Fig. 42

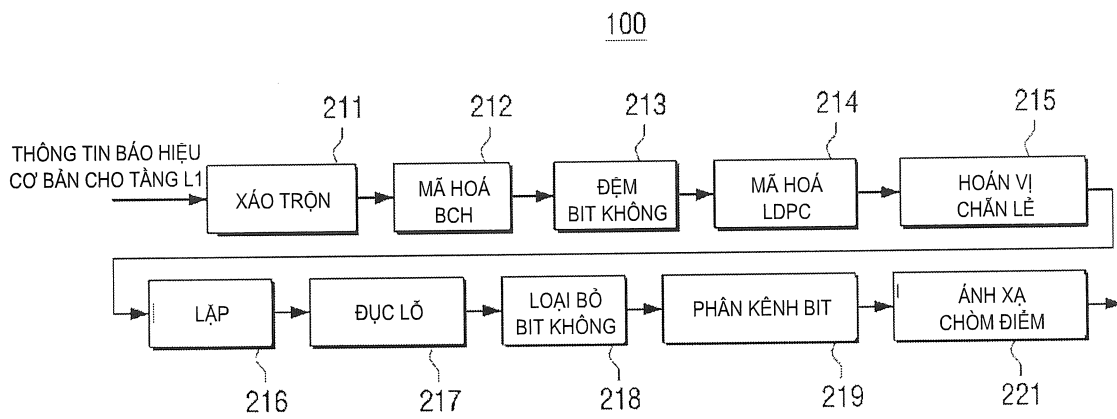


Fig. 43

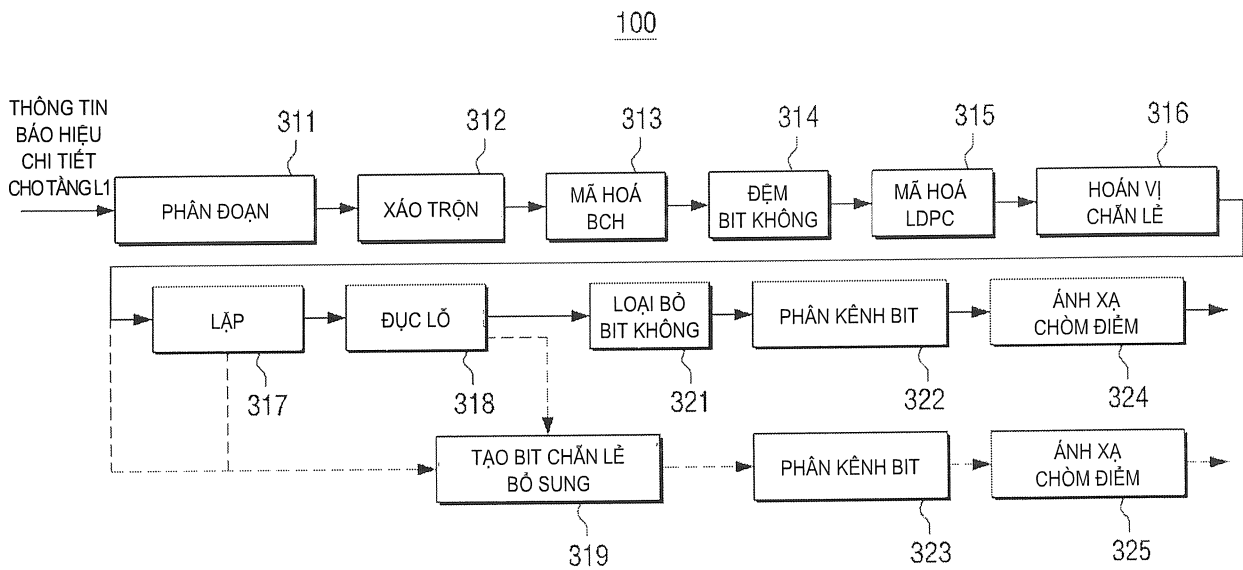


Fig. 44

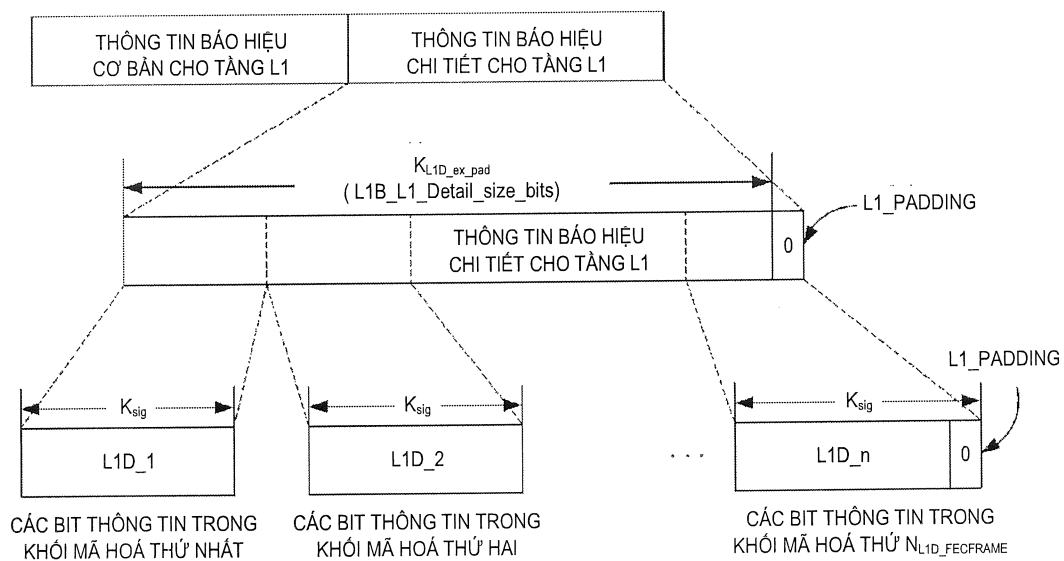


Fig. 45

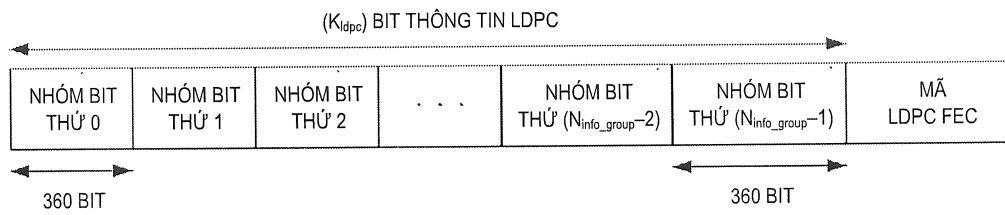


Fig. 46

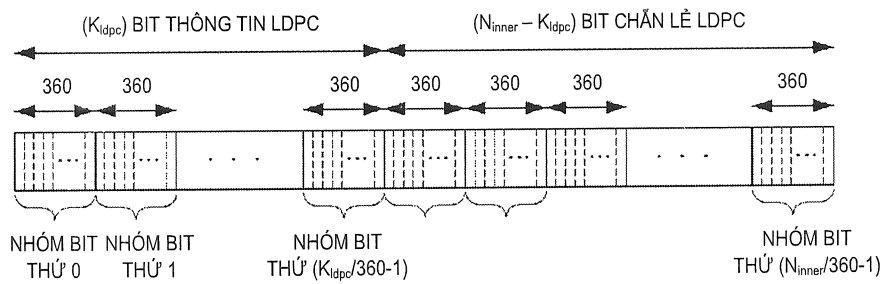


Fig. 47

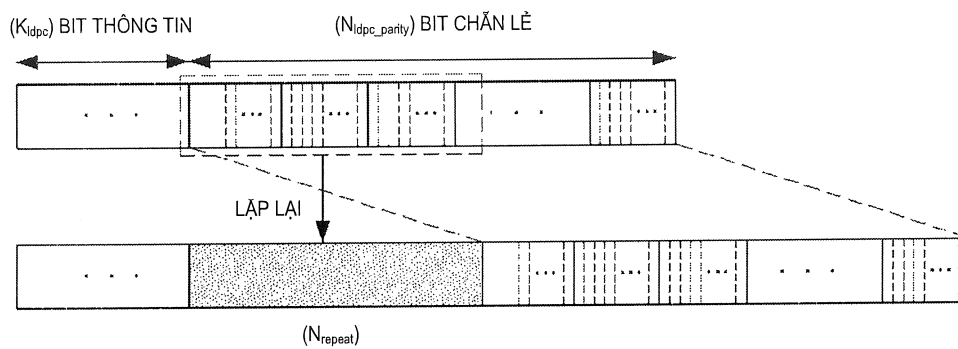


Fig. 48

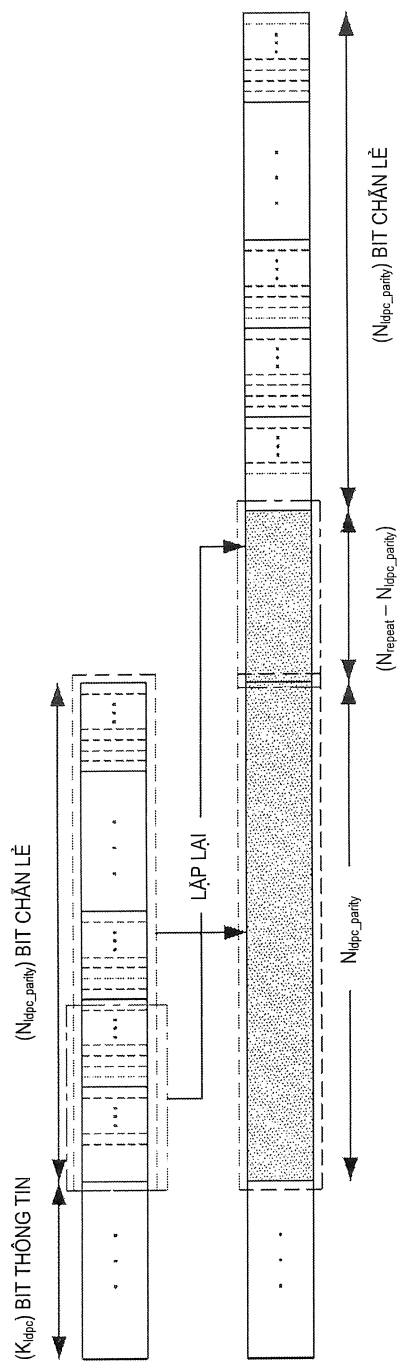


Fig. 49

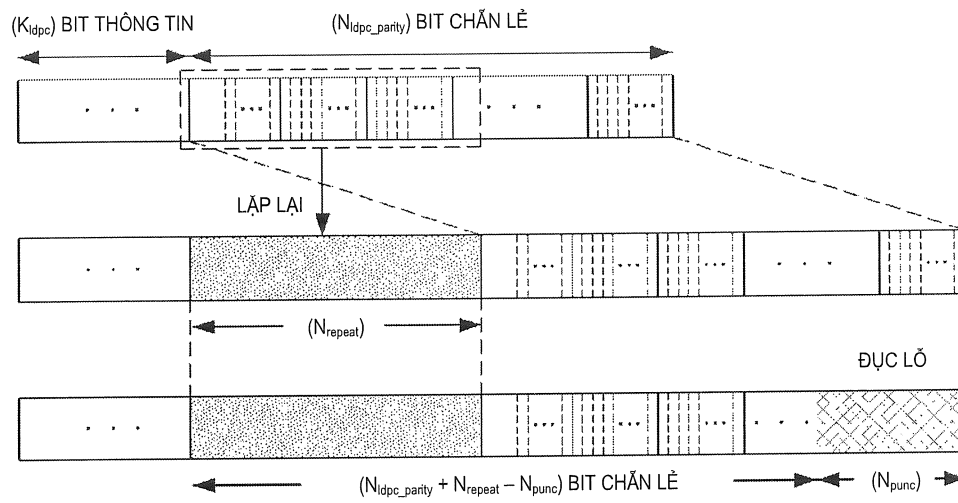


Fig. 50

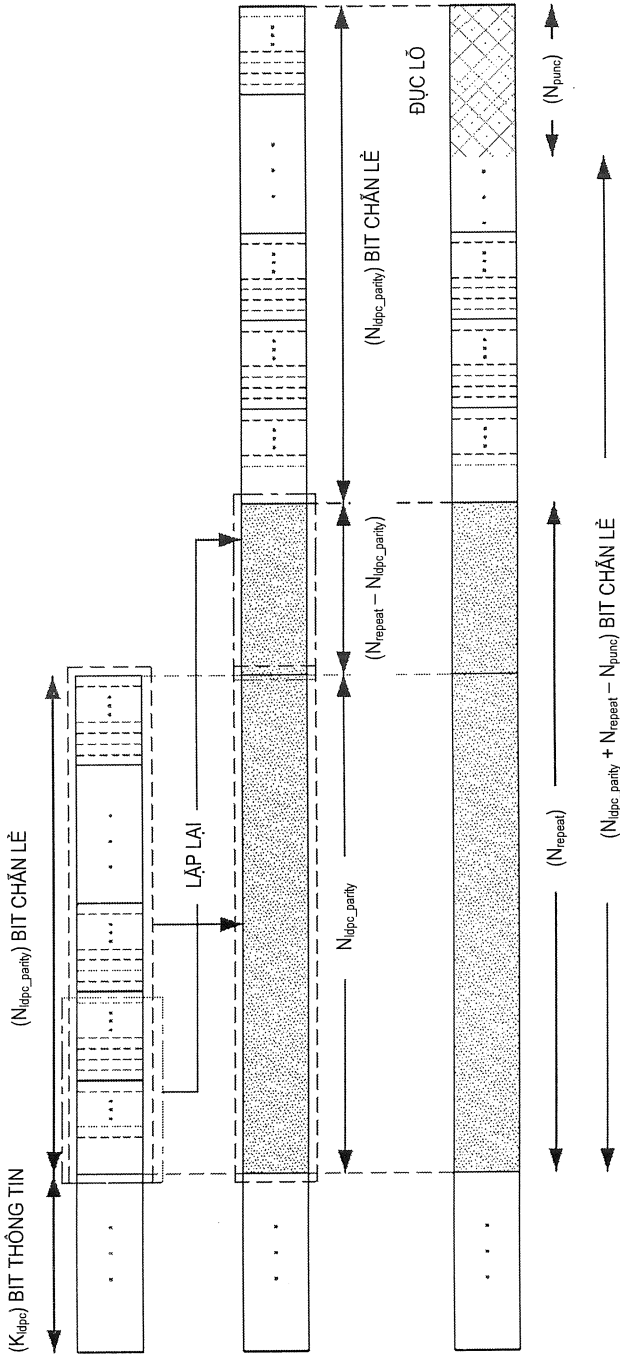


Fig. 51

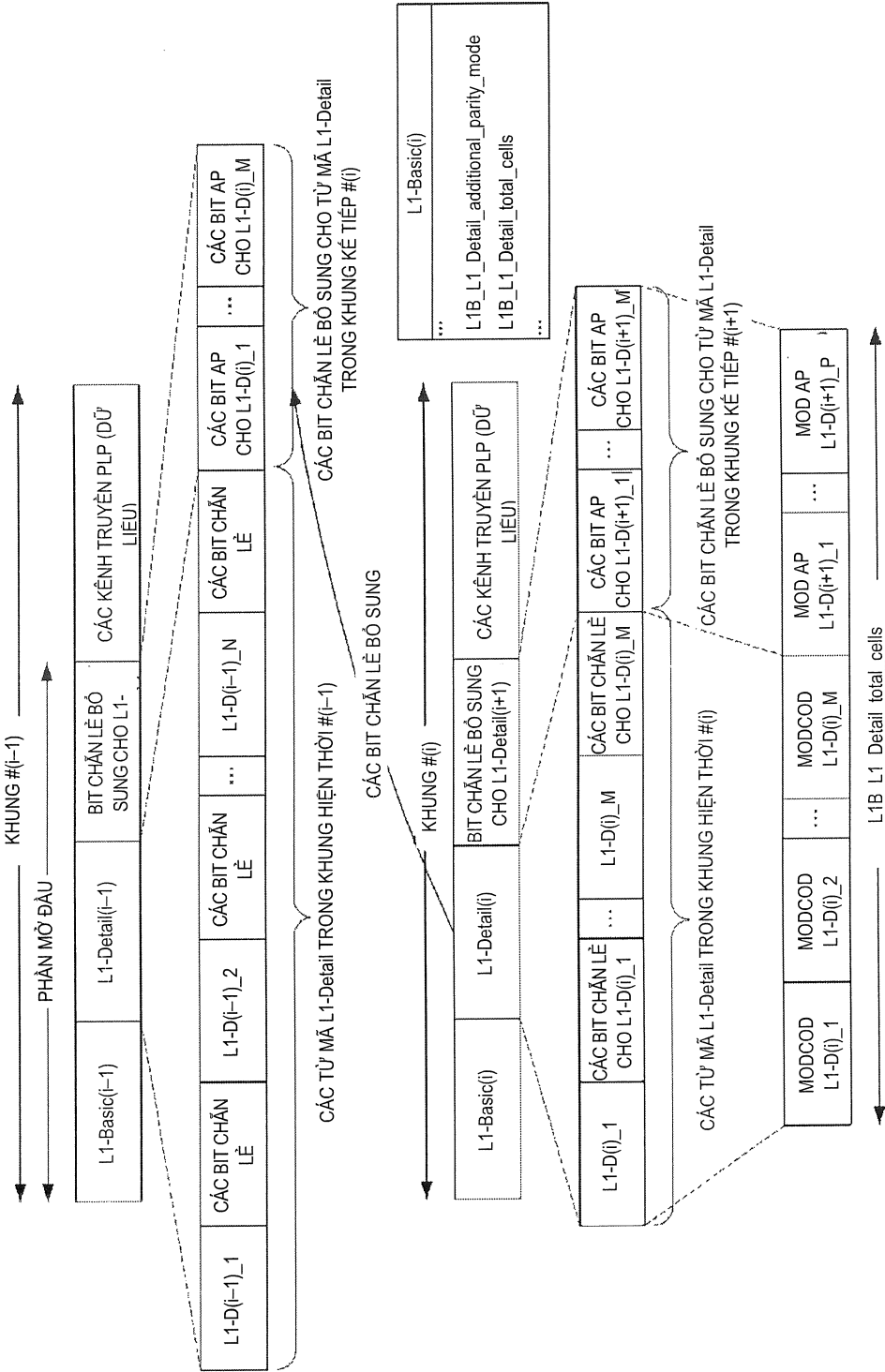


Fig. 52

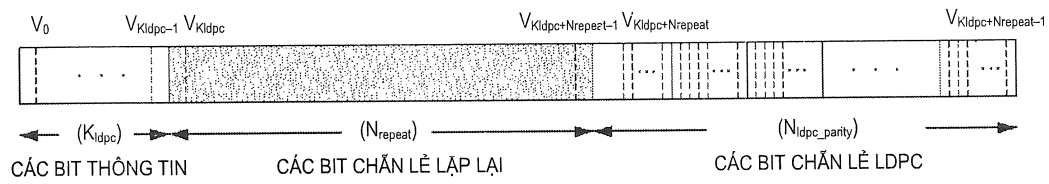


Fig. 53

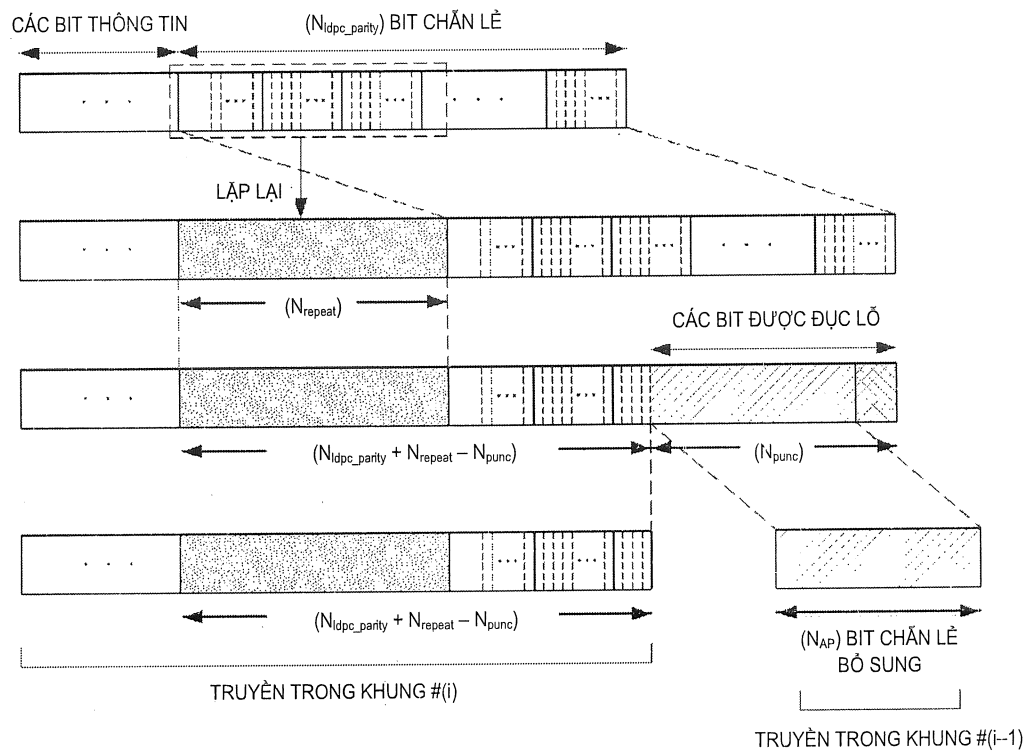


Fig. 54

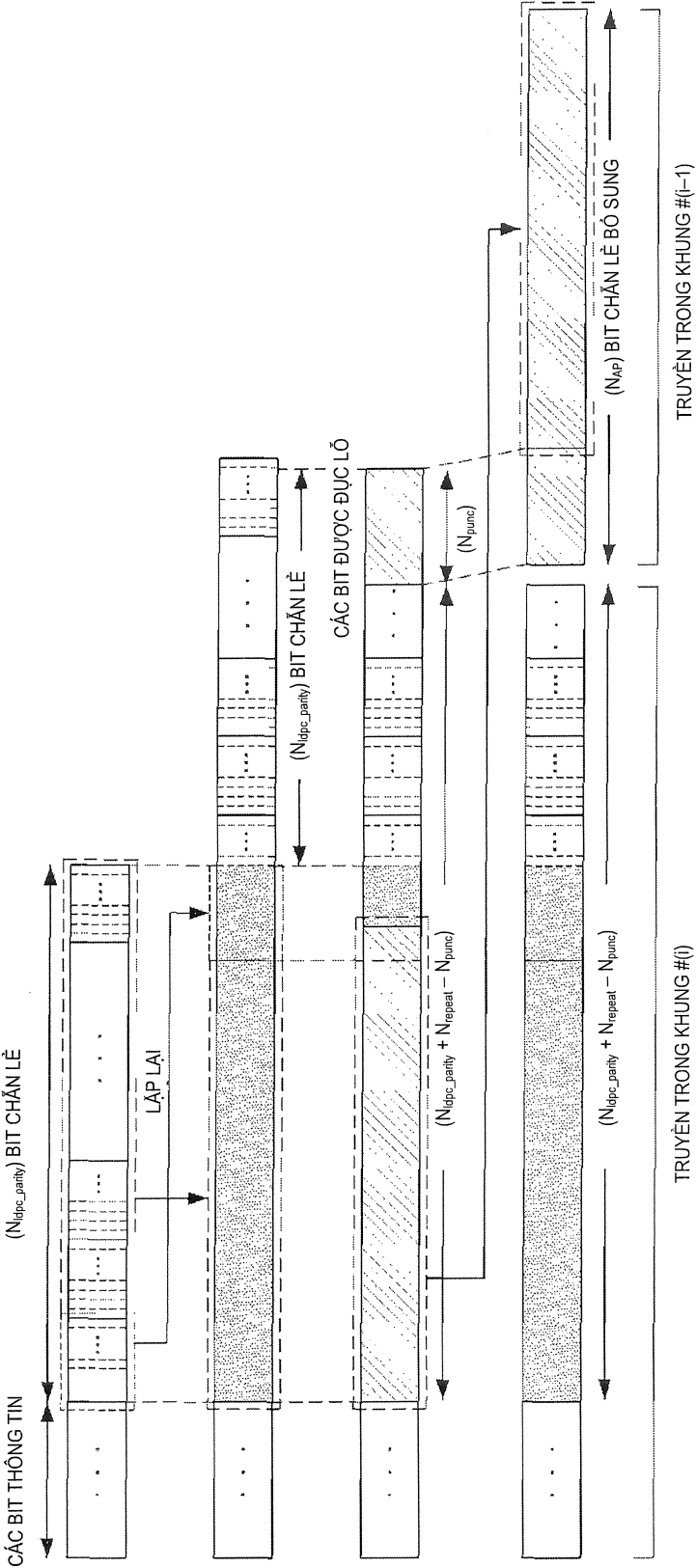


Fig. 55

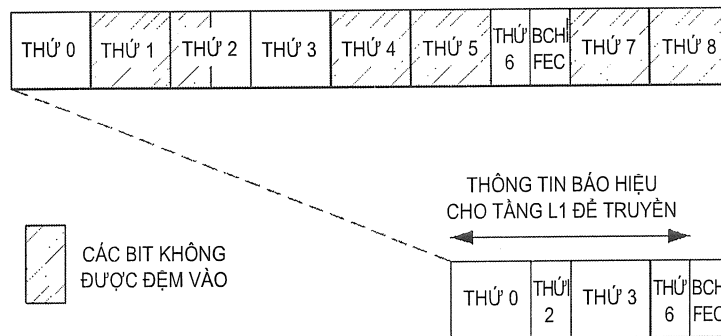


Fig. 56

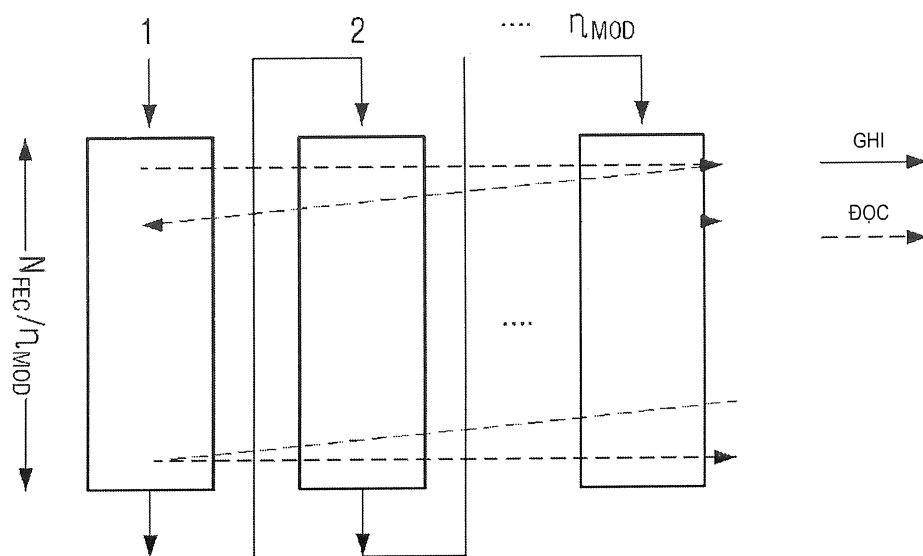


Fig. 57

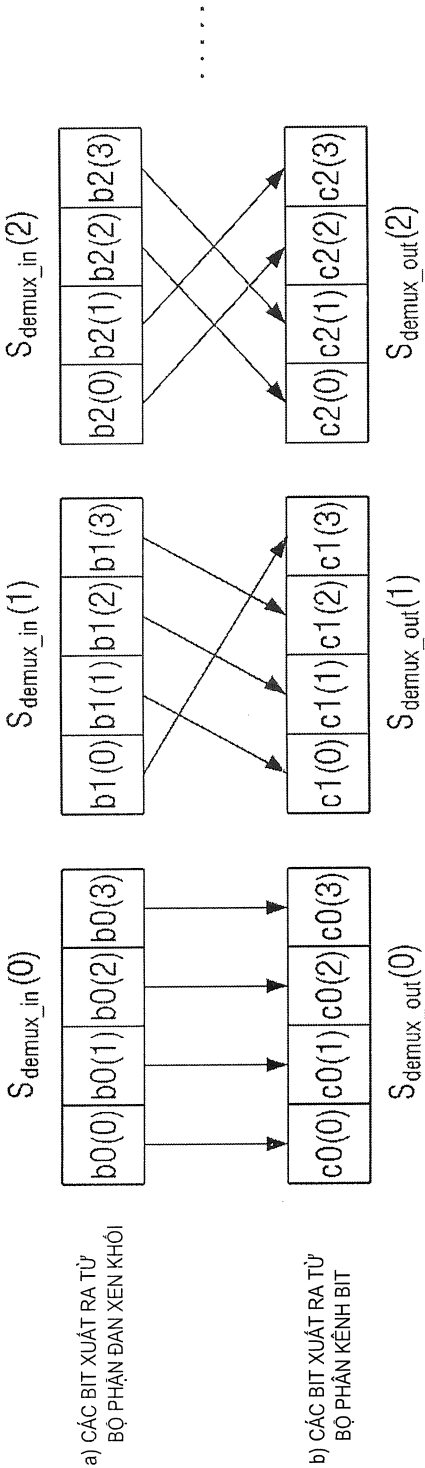


Fig. 58

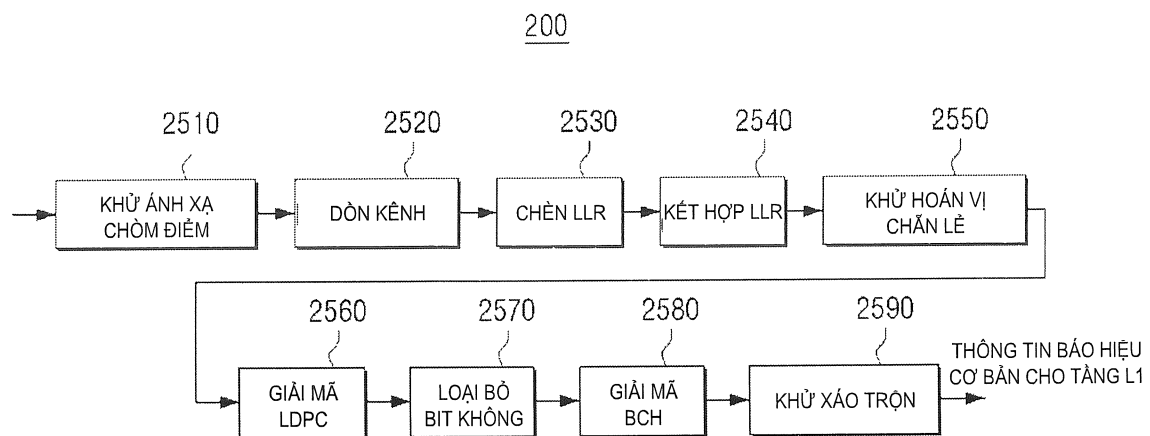


Fig. 59

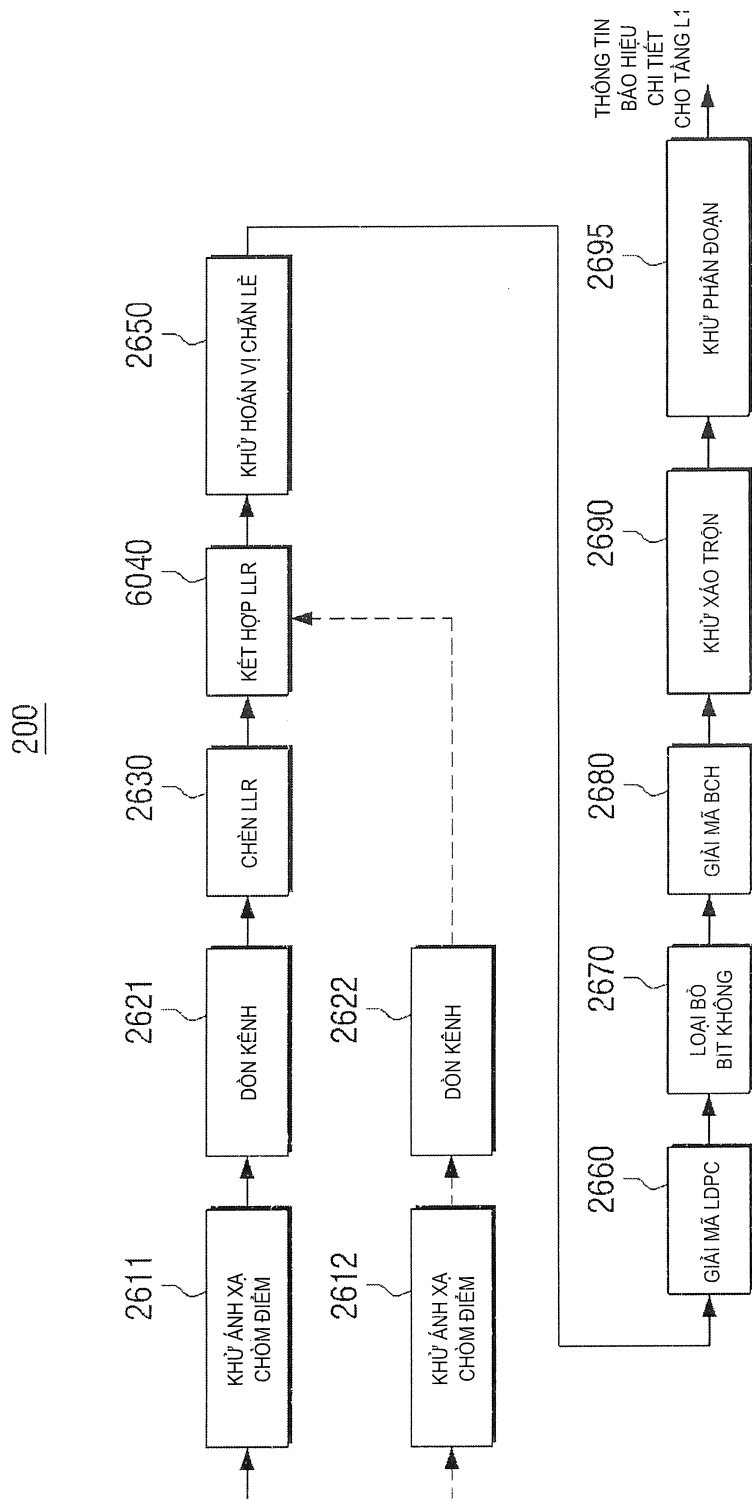


Fig. 60

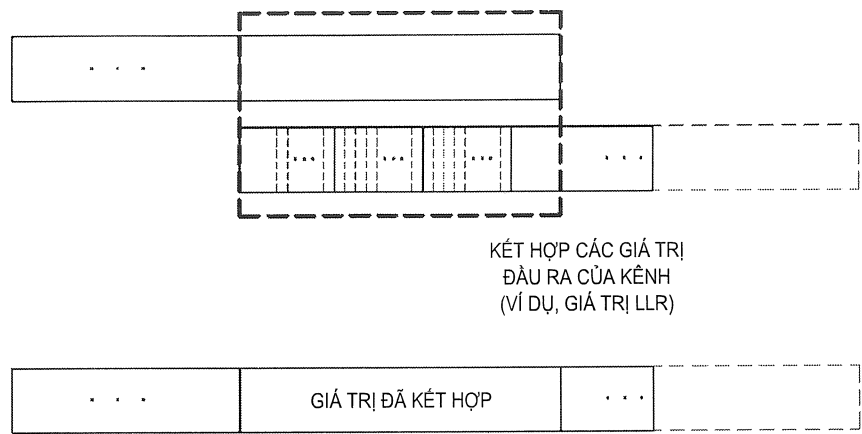


Fig. 61

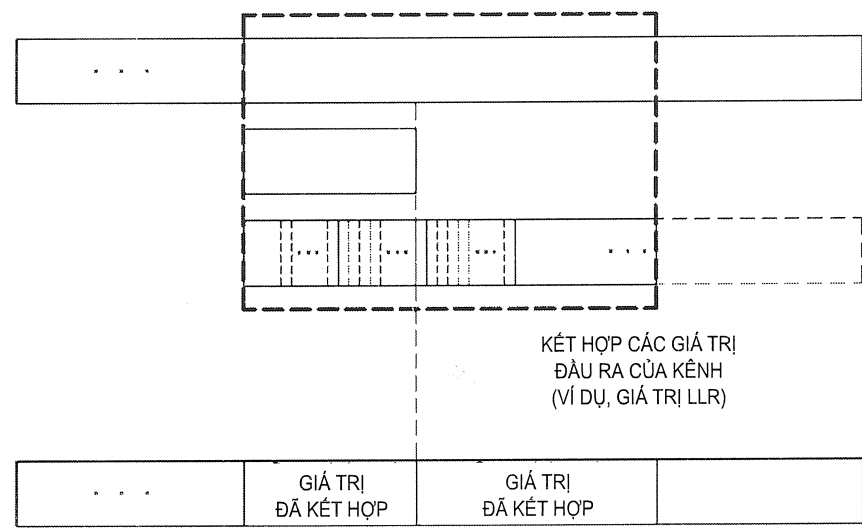


Fig. 62

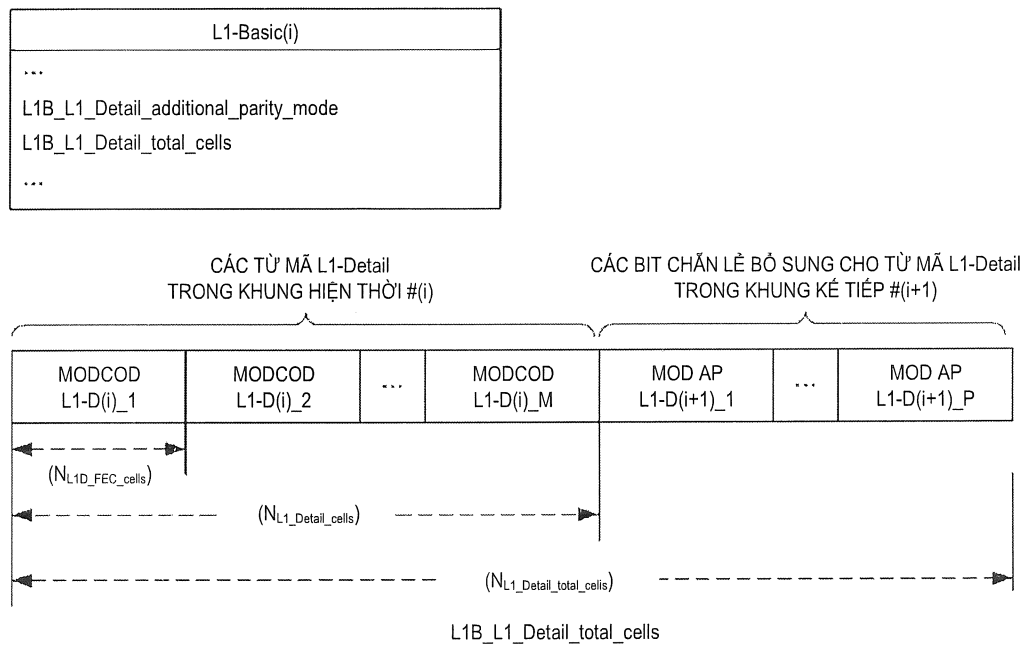


Fig. 63

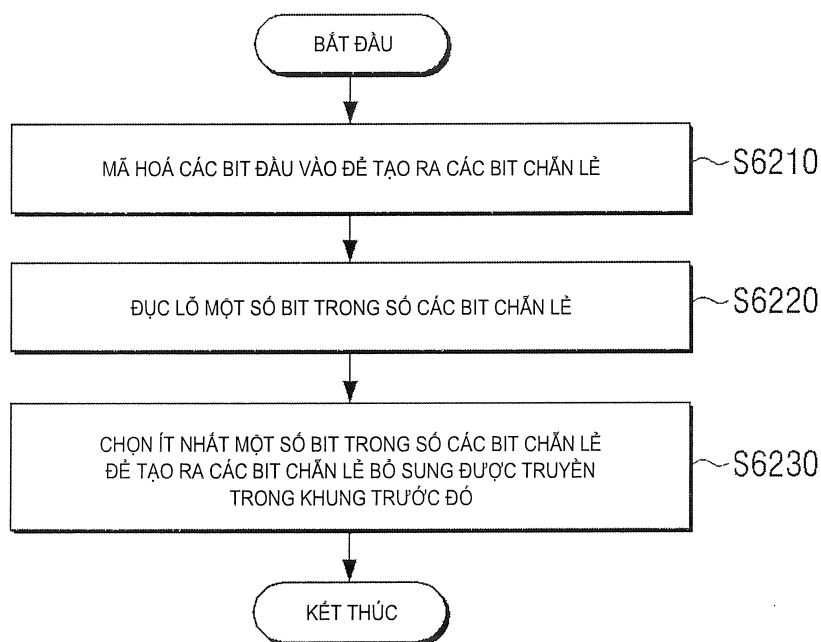


Fig. 64

