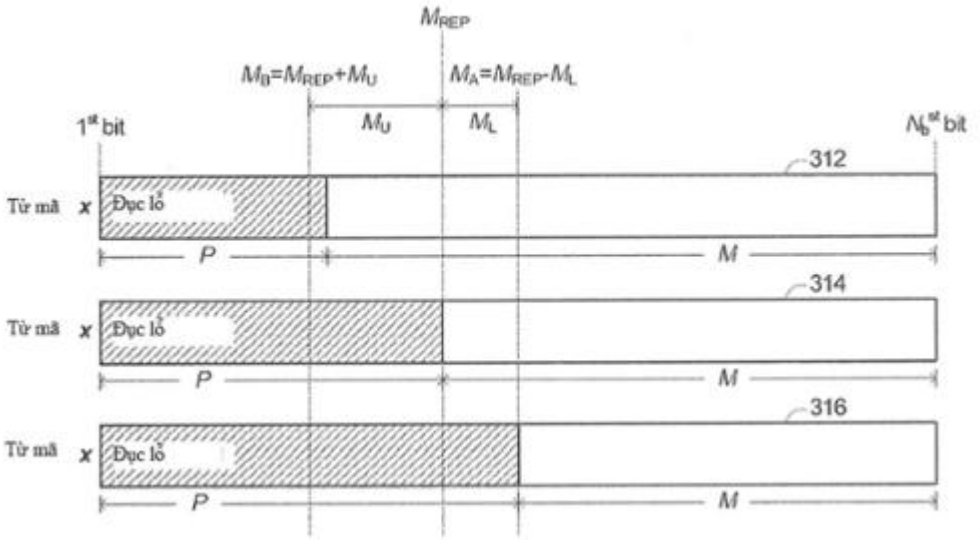
(12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) **Cộng hòa xã hội chủ nghĩa Việt Nam (VN)**
CỤC SỞ HỮU TRÍ TUỆ
(11) **1-0036365**
(51)[7] **H03M 13/00**
(13) **B**

(21) 1-2019-00152
(22) 06/06/2017
(86) PCT/CN2017/087351 06/06/2017
(87) WO2017/215494 21/12/2017
(30) 62/351,438 17/06/2016 US; 15/607,591 29/05/2017 US
(45) 25/07/2023 424
(43) 25/03/2019 372A
(73) HUAWEI TECHNOLOGIES CO., LTD. (CN)
Huawei Administration Building, Bantian, Longgang District Shenzhen, Guangdong 518129, China
(72) ZHANG, Ran (CA); SHI, Wuxian (CA); CHENG, Nan (CA); GE, Yiqun (CA).
(74) Công ty Luật TNHH Phạm và Liên danh (PHAM & ASSOCIATES)

(54) BỘ TRUYỀN VÀ PHƯƠNG PHÁP TRUYỀN ĐƯỢC THỰC HIỆN Ở BỘ TRUYỀN

(57) Sáng chế đề cập đến hệ thống và phương pháp liên quan đến việc thực hiện so khớp tốc độ khi sử dụng các mã phân cực. Theo một phương án thực hiện, các bit ở được nhận ở bộ mã hóa phân cực. Giá trị thu được tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit, và số lượng bit được mã hóa được sử dụng để truyền các bit. Xác định khoảng giá trị mà giá trị này nằm trong, và chuỗi thông tin thu được tương ứng với khoảng mà giá trị nằm trong. Các bit được ánh xạ đến tập con các vị trí của vectơ đầu vào theo chuỗi thông tin. Các vị trí còn lại của vectơ đầu vào được thiết lập làm các giá trị đóng băng được bộ giải mã biết đến. Vectơ đầu vào sau đó được mã hóa trong bộ mã hóa phân cực để tạo từ mã.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến mã hóa điều khiển lỗi và so khớp tốc độ mã hóa, và cụ thể hơn đến việc so khớp tốc độ mã hóa khi sử dụng các mã phân cực tổng quát.

Tình trạng kỹ thuật của sáng chế

Trong hệ thống truyền thông, thông tin được truyền từ bộ truyền, trên kênh, đến bộ nhận. Chẳng hạn, trong hệ thống truyền thông không dây, bộ truyền trong thiết bị di động có thể truyền thông tin trên kênh không dây đến bộ nhận trong trạm cơ sở (base station – BS).

Kênh có thể đưa lỗi vào thông tin được truyền trên kênh. Mã hóa điều khiển lỗi có thể được sử dụng để dò và/hoặc sửa các lỗi này. Chẳng hạn, k bit thông tin được truyền từ bộ truyền đến bộ nhận có thể trước hết được mã hóa bằng bộ mã hóa trong bộ truyền để thu thập từ mã có độ dài N_b bit, trong đó $N_b > k$. Sau đó, từ mã có thể được truyền trên kênh này. Sau đó từ mã được nhận được giải mã bằng bộ giải mã trong bộ nhận để thu thập quyết định về việc k bit nào đã được truyền. Dư thừa được thêm vào bằng cách truyền từ mã có độ dài bit $N_b > k$ làm tăng xác suất mà k bit được giải mã đúng ở bộ nhận, ngay cả nếu một số lỗi được đưa vào từ mã bởi nhiễu trong kênh này.

Có các loại mã dò lỗi và sửa lỗi khác nhau. Một loại mã sửa lỗi, được gọi là mã phân cực Arikan, được bộc lộ trong tài liệu "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels" by E. Arıkan, *IEEE Transactions on Information Theory*, vol. 55, no. 7 (July 2009). Mã phân cực Arikan là mã phân cực nhị phân, nghĩa là mã phân cực Arikan chỉ

thực hiện mã hóa điều khiển lỗi trên bảng chữ cái ký tự nhị phân. k bit được mã hóa bằng cách sử dụng mã phân cực Arikan, đại diện k ký hiệu thông tin. Mỗi một ký hiệu trong k ký hiệu thông tin có thể chỉ có một trong hai giá trị.

Độ dài bit N_b của mỗi từ mã trong mã phân cực Arikan cũng phải là lũy thừa hai, tức là $N_b = 2^n$, trong đó n là số tự nhiên. Tuy nhiên, do yêu cầu tốc độ mã hóa, độ dài từ mã có thể thực sự được truyền vào kênh này có thể không phải chính xác là độ dài bit $N_b = 2^n$.

Bản chất kỹ thuật của sáng chế

Các hệ thống và các phương pháp được bộc lộc có thể được thực hiện như là một phần so khớp tốc độ mã khi sử dụng các mã phân cực.

Theo một phương án thực hiện, sáng chế đề cập đến phương pháp được thực hiện ở bộ truyền. Phương pháp gồm tiếp nhận các bit ở bộ mã hóa phân cực. Phương pháp có thể còn gồm thu thập giá trị tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit, và số lượng bit được mã hóa được sử dụng để truyền các bit. Phương pháp có thể còn gồm xác định khoảng giá trị mà giá trị nằm trong đó. Phương pháp có thể còn gồm thu thập chuỗi thông tin tương ứng với khoảng mà giá trị nằm trong đó. Phương pháp có thể còn gồm ánh xạ các bit đến tập con các vị trí của vectơ đầu vào theo chuỗi thông tin. Phương pháp có thể còn gồm thiết lập các vị trí còn lại của vectơ đầu vào làm các giá trị đóng băng được bộ giải mã biết đến. Phương pháp có thể còn gồm mã hóa vectơ đầu vào trong bộ mã hóa phân cực để tạo từ mã.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, thu thập chuỗi thông tin tương ứng với khoảng có thể gồm truy xuất từ bộ nhớ chuỗi thông tin được lưu trữ tương ứng với khoảng.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, chuỗi thông tin tương ứng với khoảng có thể là chuỗi thông tin được xác định dựa trên giá trị đại diện trong khoảng. Giá trị đại diện có thể gần hơn với một đầu của khoảng so với đầu còn lại của khoảng. Giá trị đại diện có thể bằng giá trị ở hoặc gần đầu của khoảng.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, phương pháp có thể còn gồm đục lỗ từ mã theo mẫu hình đục lỗ để thu thập từ mã được đục lỗ. Độ dài bit của từ mã được đục lỗ bằng số lượng bit được mã hóa được sử dụng để truyền các bit. Theo một số phương án thực hiện, đục lỗ từ mã theo mẫu hình đục lỗ bao gồm rút ngắn từ mã. Theo một số phương án thực hiện, từ mã có độ dài bit N_b, từ mã được đục lỗ có độ dài bit M, và mẫu hình đục lỗ là để đục lỗ $(N_b - M)$ bit thứ nhất của từ mã. Các mẫu hình đục lỗ khác là khả thi thay vào đó.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, các bit có thể là các bit thứ nhất, giá trị này có thể là giá trị thứ nhất, chuỗi thông tin có thể là chuỗi thông tin thứ nhất, vectơ đầu vào có thể là vectơ đầu vào thứ nhất, từ mã có thể là từ mã thứ nhất, và từ mã được đục lỗ có thể là từ mã được đục lỗ thứ nhất. Sau đó, phương pháp có thể còn gồm tiếp nhận các bit thứ hai ở bộ mã hóa phân cực. Phương pháp có thể còn gồm thu thập giá trị thứ hai tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit thứ hai, và độ dài bit của từ mã được đục lỗ thứ hai được sử dụng để truyền các bit thứ hai. Phương pháp có thể còn gồm xác định khoảng giá trị mà giá trị thứ hai nằm trong đó. Phương pháp có thể còn gồm thu thập chuỗi thông tin thứ hai tương ứng với khoảng mà giá trị thứ hai nằm trong. Phương pháp có thể còn gồm ánh xạ các bit thứ hai đến tập con các vị trí của vectơ đầu vào thứ hai theo chuỗi thông tin thứ hai. Phương pháp có thể còn gồm thiết lập các vị trí còn lại của vectơ đầu vào thứ hai làm các giá trị đóng băng được bộ giải mã biết

đến. Phương pháp có thể còn gồm mã hóa vectơ đầu vào thứ hai trong bộ mã hóa phân cực để tạo từ mã thứ hai. Phương pháp có thể còn gồm đục lỗ từ mã thứ hai theo mẫu hình đục lỗ để thu thập từ mã được đục lỗ thứ hai. Theo một số phương án thực hiện, từ mã được đục lỗ thứ nhất và từ mã được đục lỗ thứ hai có độ dài bit khác nhau. Theo một số phương án thực hiện, khoảng mà giá trị thứ hai nằm trong giống như khoảng mà giá trị thứ nhất nằm trong. Theo một số phương án thực hiện, chuỗi thông tin thứ nhất giống như chuỗi thông tin thứ hai. Theo một số phương án thực hiện, khoảng mà giá trị thứ hai nằm trong khác với khoảng mà giá trị thứ nhất nằm trong. Theo một số phương án thực hiện, chuỗi thông tin thứ nhất khác với chuỗi thông tin thứ hai.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, ít nhất một số bit có thể biểu diễn các ký hiệu q phân, trong đó $q > 2$.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, các bit có thể biểu diễn các ký hiệu q phân, trong đó $q > 2$, và từ mã có thể bao gồm ký hiệu q phân được biểu diễn bởi các bit.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, mã hóa vectơ đầu vào có thể gồm mã hóa vectơ đầu vào bằng cách sử dụng ít nhất một nhân bộ mã hóa phân cực để tạo từ mã. Việc mã hóa có thể gồm tiếp nhận, ở nhân bộ mã hóa phân cực, tập của ký hiệu đầu vào q phân được biểu diễn bởi các bit. Việc mã hóa có thể gồm biến đổi tập của ký hiệu đầu vào q phân, theo ma trận hạt của nhân bộ mã hóa phân cực, để tạo tập của ký hiệu đầu ra q phân được biểu diễn bởi các bit.

Theo phương án thực hiện khác, bộ truyền có bộ mã hóa phân cực để nhận các bit được đề xuất. Bộ truyền có thể còn gồm (chẳng hạn trong hoặc như là một phần của bộ mã hóa phân cực) bộ tạo chuỗi thông tin. Bộ tạo chuỗi thông tin có thể được tạo cấu hình để thu thập giá trị tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit, và số

lượng bit được mã hóa được sử dụng để truyền các bit. Bộ tạo chuỗi thông tin có thể còn được tạo cấu hình để xác định khoảng giá trị mà giá trị này nằm trong và thu thập chuỗi thông tin tương ứng với khoảng mà giá trị nằm trong. Bộ truyền có thể còn gồm (chẳng hạn trong hoặc như là một phần của bộ mã hóa phân cực) bộ tạo vectơ đầu vào. Bộ tạo vectơ đầu vào có thể được tạo cấu hình để ánh xạ các bit đến tập con các vị trí của vectơ đầu vào theo chuỗi thông tin. Bộ tạo vectơ đầu vào có thể còn được tạo cấu hình để thiết lập các vị trí còn lại của vectơ đầu vào làm các giá trị đóng băng được bộ giải mã biết đến. Bộ mã hóa phân cực có thể còn được tạo cấu hình để mã hóa vectơ đầu vào để tạo từ mã.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên bộ truyền có thể còn gồm bộ nhớ. Bộ tạo chuỗi thông tin có thể được tạo cấu hình để thu thập chuỗi thông tin tương ứng với khoảng bằng cách truy xuất từ bộ nhớ chuỗi thông tin được lưu trữ tương ứng với khoảng.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, chuỗi thông tin tương ứng với khoảng có thể là chuỗi thông tin được xác định dựa trên giá trị đại diện trong khoảng. Giá trị đại diện có thể gần hơn với một đầu của khoảng so với đầu còn lại của khoảng. Giá trị đại diện có thể ở gần hoặc ở một đầu của khoảng.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, bộ truyền có thể còn gồm bộ đục lỗ để đục lỗ từ mã theo mẫu hình đục lỗ để thu thập từ mã được đục lỗ. Độ dài bit của từ mã được đục lỗ bằng số lượng bit được mã hóa được sử dụng để truyền các bit. Theo một số phương án thực hiện, bộ đục lỗ được tạo cấu hình để đục lỗ từ mã theo mẫu hình đục lỗ bằng cách rút ngắn từ mã. Theo một số phương án thực hiện, từ mã có độ dài bit N_b, từ mã được đục lỗ có độ dài bit M, và mẫu hình đục lỗ là để đục lỗ $(N_b - M)$ bit thứ nhất của từ mã. Các mẫu hình đục lỗ khác là khả thi.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, các bit có thể là các bit thứ nhất, giá trị này có thể là giá trị thứ nhất, chuỗi thông tin có thể là chuỗi thông tin thứ nhất, vectơ đầu vào có thể là vectơ đầu vào thứ nhất, từ mã có thể là từ mã thứ nhất, và từ mã được đục lỗ có thể là từ mã được đục lỗ thứ nhất. Bộ mã hóa phân cực có thể còn để nhận các bit thứ hai. Bộ tạo chuỗi thông tin có thể còn được tạo cấu hình để thu thập giá trị thứ hai tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit thứ hai, và độ dài bit của từ mã được đục lỗ thứ hai được sử dụng để truyền các bit thứ hai. Bộ tạo chuỗi thông tin có thể còn được tạo cấu hình để xác định khoảng giá trị mà giá trị thứ hai nằm trong. Bộ tạo chuỗi thông tin có thể còn được tạo cấu hình để thu thập chuỗi thông tin thứ hai tương ứng với khoảng mà giá trị thứ hai nằm trong. Bộ tạo vectơ đầu vào có thể còn được tạo cấu hình để ánh xạ các bit thứ hai đến tập con các vị trí của vectơ đầu vào thứ hai theo chuỗi thông tin thứ hai. Bộ tạo vectơ đầu vào có thể còn được tạo cấu hình để thiết lập các vị trí còn lại của vectơ đầu vào thứ hai làm các giá trị đóng băng được bộ giải mã biết đến. Bộ mã hóa phân cực có thể còn được tạo cấu hình để mã hóa vectơ đầu vào thứ hai để tạo từ mã thứ hai. Bộ đục lỗ có thể còn được tạo cấu hình để đục lỗ từ mã thứ hai theo mẫu hình đục lỗ để thu thập từ mã được đục lỗ thứ hai. Theo một số phương án thực hiện, từ mã được đục lỗ thứ nhất và từ mã được đục lỗ thứ hai có độ dài bit khác nhau. Theo một số phương án thực hiện, khoảng mà giá trị thứ hai nằm trong giống như khoảng mà giá trị thứ nhất nằm trong. Theo một số phương án thực hiện, chuỗi thông tin thứ nhất giống như chuỗi thông tin thứ hai. Theo một số phương án thực hiện, khoảng mà giá trị thứ hai nằm trong khác với khoảng mà giá trị thứ nhất nằm trong. Theo một số phương án thực hiện, chuỗi thông tin thứ nhất khác với chuỗi thông tin thứ hai.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, ít nhất một số bit có thể biểu diễn ký hiệu q phân, trong đó $q > 2$.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, các bit có thể biểu diễn ký hiệu q phân, trong đó $q > 2$, và từ mã bao gồm ký hiệu q phân được biểu diễn bởi các bit.

Theo phương án bất kỳ trong các phương án thực hiện nêu trên, bộ mã hóa phân cực có thể mã hóa vectơ đầu vào bằng cách mã hóa vectơ đầu vào bằng cách sử dụng ít nhất một nhân bộ mã hóa phân cực để tạo từ mã. Việc mã hóa có thể gồm tiếp nhận, ở nhân bộ mã hóa phân cực, tập của ký hiệu đầu vào q phân được biểu diễn bởi các bit. Việc mã hóa có thể gồm biến đổi tập của ký hiệu đầu vào q phân, theo ma trận hạt của nhân bộ mã hóa phân cực, để tạo tập của ký hiệu đầu ra q phân được biểu diễn bởi các bit.

Mô tả vắn tắt các hình vẽ

Các phương án thực hiện sẽ được mô tả, chỉ qua ví dụ, dựa vào các hình vẽ đi kèm trong đó:

Fig.1 là sơ đồ khối của hệ thống truyền thông theo một phương án thực hiện;

Fig.2 là hình vẽ thể hiện cách thức ma trận nhân Kronecker có thể được tạo từ ma trận hạt $\boldsymbol{G_2}$;

Fig.3 là lưu đồ minh họa cách thức từ mã được tạo bằng cách sử dụng mã phân cực nhị phân, theo một phương án thực hiện;

Fig.4 là sơ đồ của nhân Arikan;

Fig.5 và Fig.6 là sơ đồ của cấu trúc triển khai ví dụ ở bước 208 trên Fig.3;

Fig.7 minh họa nhân tổng quát lấy làm ví dụ;

Fig.8 minh họa nhân dựa trên Reed-Solomon (RS);

Fig.9 và Fig.10 là sơ đồ của cấu trúc bộ mã hóa phân cực RS(4) cho $n = 2$, tức là, 16 ký hiệu;

Fig.11 minh họa sơ đồ khối của hệ thống truyền thông theo phương án thực hiện khác;

Fig.12 và Fig.13 minh họa một ví dụ của bộ giải mã cho từ mã 32 bit được tạo bằng cách sử dụng 2 lớp của các nhân RS(4) khi xem xét đục lỗ;

Fig.14 minh họa các đường cong FER (frame error rate – tỷ lệ lỗi khung) để giải mã phân cực có hỗ trợ CRC;

Fig.15 là lưu đồ của phương pháp được thực hiện bởi bộ truyền theo một phương án thực hiện;

Fig.16 minh họa LUT (look up table – bảng tra cứu) được lưu trữ trong bộ nhớ trong bộ truyền;

Fig.17 thể hiện ba từ mã minh họa khái niệm so khớp độ dài theo đoạn theo một phương án thực hiện;

Fig.18 là lưu đồ của phương pháp được thực hiện bởi bộ truyền theo phương án thực hiện khác;

Fig.19 đến Fig.22 minh họa các đường cong FER khác nhau;

Fig.23 là lưu đồ của phương pháp được thực hiện bởi bộ truyền theo phương án thực hiện khác;

Fig.24 minh họa hệ thống truyền thông lấy làm ví dụ; và

Fig.25 và Fig.26 minh họa các thiết bị lấy làm ví dụ có thể triển khai chức năng và/hoặc các phương án thực hiện được mô tả ở đây.

Mô tả chi tiết sáng chế

Nhằm mục đích minh họa, các phương án thực hiện lấy làm ví dụ cụ thể sẽ được giải thích chi tiết hơn dưới đây cùng với các hình vẽ.

Fig.1 là sơ đồ khối của hệ thống truyền thông 122 theo một phương án thực hiện. Hệ thống truyền thông 122 gồm bộ truyền 124 và bộ nhận

126 truyền thông trên kênh 128. Bộ truyền 124 gồm bộ mã hóa phân cực 130 và bộ nhận 126 gồm bộ giải mã phân cực 132.

Bộ mã hóa phân cực 130, cũng như các chức năng xử lý tín hiệu tín hiệu/dữ liệu của bộ truyền 124, chẳng hạn bộ đục lỗ được mô tả sau đây, có thể được triển khai bởi bộ xử lý thực thi các lệnh khiến bộ xử lý thực thi một số hoặc tất cả các hoạt động của bộ mã hóa phân cực 130 và bộ truyền 124. Theo cách khác, bộ mã hóa phân cực 130, cũng như các chức năng xử lý tín hiệu/dữ liệu còn lại của bộ truyền 124, có thể được triển khai trong phần cứng hoặc hệ mạch (chẳng hạn, trong một hoặc nhiều chipset, bộ vi xử lý, các mạch tích hợp ứng dụng cụ thể (application-specific integrated circuit – ASIC), các mảng cổng dạng trường lập trình được (field-programmable gate array – FPGA), hệ mạch dành riêng, hoặc các tổ hợp của nó) và được tạo cấu hình để triển khai các hoạt động của bộ mã hóa phân cực 130 và bộ truyền 124. Mặc dù không được thể hiện trên hình vẽ, bộ truyền 124 có thể gồm bộ điều biến, bộ khuếch đại, anten và/hoặc các môđun khác hoặc các thành phần của chuỗi truyền hoặc theo cách khác có thể được tạo cấu hình để tạo giao diện với môđun truyền tần số vô tuyến (Radio Frequency – RF) riêng rẽ sao cho các từ mã có thể được tạo như được mô tả ở đây và được truyền trực tiếp hoặc bởi khối hoặc môđun truyền riêng rẽ. Bộ truyền 124 cũng có thể gồm vật ghi máy tính đọc được bất biến (không được thể hiện trên hình vẽ), gồm các lệnh để thực thi (chẳng hạn, bởi bộ xử lý hoặc hệ mạch khác như được mô tả trên đây) để triển khai và/hoặc điều khiển hoạt động của bộ mã hóa phân cực 130 và bộ truyền 124, và/hoặc để điều khiển theo cách khác việc thực thi chức năng và/hoặc các phương án thực hiện được mô tả ở đây. Một số phương án thực hiện có thể được triển khai bằng cách sử dụng chỉ phần cứng. Theo một số phương án thực hiện, các lệnh để thực thi bởi bộ xử lý có thể được triển khai ở dạng sản phẩm phần mềm. Sản phẩm phần mềm

có thể được lưu trữ trong vật lưu trữ bất biến hoặc bộ nhớ, có thể là, chẳng hạn, CD-ROM, ổ nhớ nhanh USB, hoặc đĩa cứng tháo được.

Một cách tương tự, bộ giải mã phân cực 132, cũng như các chức năng xử lý tín hiệu/dữ liệu khác của bộ nhận 126, có thể được triển khai bởi bộ xử lý mà thực thi các lệnh khiến bộ xử lý thực hiện một số hoặc tất cả các hoạt động của bộ giải mã phân cực 132 và bộ nhận 126. Theo cách khác, bộ giải mã phân cực 132, cũng như các chức năng xử lý tín hiệu/dữ liệu còn lại của bộ nhận 126, có thể được triển khai trong phần cứng hoặc hệ mạch (chẳng hạn trong một hoặc nhiều chipset, các bộ vi xử lý, các ASIC, các FPGA, hệ mạch dành riêng, hoặc các tổ hợp của nó) và được tạo cấu hình để triển khai một số hoặc tất cả các hoạt động của bộ giải mã phân cực 132 và bộ nhận 126. Mặc dù không được thể hiện trên hình vẽ, bộ nhận 126 có thể gồm anten, bộ giải điều biến, bộ khuếch đại, và/hoặc các mô đun khác hoặc các thành phần của chuỗi nhận hoặc theo cách khác có thể được tạo cấu hình để tạo giao diện với môđun nhận RF riêng để xử lý và/hoặc giải mã các từ dựa trên các từ mã của mã phân cực được tiếp nhận bởi bộ nhận 126 trực tiếp hoặc gián tiếp từ khối hoặc môđun nhận riêng rẽ. Bộ nhận 126 cũng có thể gồm vật ghi máy tính đọc được bất biến (không được thể hiện trên hình vẽ), gồm các lệnh để thực thi (chẳng hạn, bởi bộ xử lý hoặc một số hệ mạch khác như được mô tả trên đây) để triển khai và/hoặc điều khiển hoạt động của bộ giải mã phân cực 132 và bộ nhận 126, và/hoặc để điều khiển theo cách khác việc thực thi chức năng và/hoặc các phương án thực hiện được mô tả ở đây. Một số phương án thực hiện có thể được triển khai bằng cách sử dụng chỉ phần cứng. Theo một số phương án thực hiện, các lệnh để thực thi bởi bộ xử lý có thể được triển khai ở dạng sản phẩm phần mềm. Sản phẩm phần mềm có thể được lưu trữ trong vật lưu trữ hoặc bộ nhớ bất biến, có thể là, chẳng hạn, CD-ROM, đĩa nhớ nhanh USB, hoặc đĩa cứng tháo được.

Khi đang hoạt động, bộ mã hóa phân cực 130 mã hóa nhóm m bit để thu thập từ mã $\boldsymbol{x}$ tương ứng có độ dài từ mã bằng N_b bit, trong đó $N_b > m$. Mặc dù bộ mã hóa phân cực 130 nhận các bit, trong bộ mã hóa phân cực 130 bit có thể biểu diễn ký hiệu của bảng chữ cái q phân. Việc sử dụng từ "ký hiệu" không có nghĩa là đề cập đến ký hiệu trong tập hợp điều biến, nhưng có nghĩa là đề cập đến các phần tử của bảng chữ cái q phân. Chẳng hạn, bảng chữ cái tứ phân có thể có bốn ký hiệu được ký hiệu sử dụng ký hiệu "0", "1", "2", và "3", và lần lượt được biểu diễn bởi các bit 00, 01, 10, và 11

Từ mã $\boldsymbol{x}$ được xuất ra bởi bộ mã hóa phân cực 130 được truyền trên kênh 128 này và được nhận ở bộ giải mã phân cực 132 của bộ nhận 126. Bộ giải mã phân cực 132 thực hiện giải mã để quyết định nhóm m bit nào được truyền. Việc giải mã được xem là thành công nếu m bit được giải mã bởi bộ giải mã phân cực 132 so khớp m bit được truyền ban đầu. Nếu kênh 128 nhiễu và đưa quá nhiều lỗi vào từ mã $\boldsymbol{x}$, thì việc giải mã phân cực không thể sửa tất cả các lỗi.

Mã phân cực có thể được tạo sử dụng ma trận tích Kronecker $\boldsymbol{G}$ được tạo từ ma trận hạt $\boldsymbol{G_s}$. Đối với mã phân cực nhị phân, mỗi ký hiệu thông tin là bit, và ma trận hạt $\boldsymbol{G_s} = \boldsymbol{G_2}$ có thể được sử dụng, trong đó $\boldsymbol{G_2} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$.

Fig.2 thể hiện cách thức ma trận tích Kronecker có thể được tạo từ ma trận hạt $\boldsymbol{G_2}$. Được thể hiện trên Fig.2 là ma trận tích Kronecker $\boldsymbol{G_2}^{\otimes 2}$ 142 gập hai và ma trận tích Kronecker gập 3 $\boldsymbol{G_2}^{\otimes 3}$ 144. Cách thức tiếp cận tích Kronecker có thể được lặp lại để tạo ma trận tích Kronecker $\mathbf{G}_2^{\otimes n}$ gập n. Đối với mã phân cực nhị phân có các từ mã có độ dài $N = 2^n$, ma trận tích Kronecker $\boldsymbol{G}$ là ma trận sinh cho mã phân cực, và bằng $\boldsymbol{G} = \boldsymbol{G_2}^{\otimes n}$.

Fig.3 là lưu đồ minh họa cách thức từ mã được tạo sử dụng mã phân cực nhị phân, theo một phương án thực hiện. Một ví dụ cụ thể được thể hiện trong các bọt chấm cho mã phân cực nhị phân có các từ mã có độ dài bit $N_b = 8$, tức là, mã phân cực nhị phân có ma trận sinh $\boldsymbol{G} = \boldsymbol{G_2}^{\otimes 3}$. Ở bước 202, k bit thông tin sẽ được truyền đến bộ nhận 126 được thu thập. Trong ví dụ, $k = 4$ bit được dán nhãn $b_1 b_2 b_3 b_4$. Một cách tùy chọn, ở bước 204, bit hỗ trợ hoặc các bit mã dò lỗi (error-detecting code – EDC), chẳng hạn các bit kiểm tra dư thừa tuần hoàn (cyclic redundancy check – CRC), được thêm vào k bit (để dẫn đến m bit thông tin) để hỗ trợ trong việc giải mã. Nên hiểu rằng nhiều hơn một EDC có thể được sử dụng trong một từ mã. Cũng nên hiểu rằng các loại EDC khác có thể được sử dụng thay thế hoặc thêm vào, chẳng hạn các mã tổng kiểm tra, các mã Fletcher, các mã băm hoặc các mã kiểm tra chẵn lẻ khác. Một số EDC cũng có thể được sử dụng làm các ECC và có thể được sử dụng khi chọn tuyến để giải mã danh sách, chẳng hạn, để cải thiện hiệu suất mã phân cực.

Các bit CRC, chẳng hạn, được tạo dựa trên k bit thông tin và thường được đặt ở các vị trí đáng tin cậy hơn ở vectơ đầu vào. Tuy nhiên, tùy thuộc vào mục đích dự kiến (chẳng hạn, được sử dụng để dò lỗi hoặc sửa lỗi hoặc cả hai), các bit CRC cũng có thể hoặc thay vào đó được phân tán hoặc theo cách khác được đặt ở các vị trí khác ở vectơ đầu vào. Trong ví dụ này, bắt đầu với k bit thông tin, CRC được tính toán và bổ sung vào k bit thông tin để tạo m bit thông tin gồm k bit thông tin và các bit CRC.

Trong ví dụ trên Fig.3, không bit EDC nào được thêm vào ở bước 204, và do vậy $k = m$. m bit được đưa vào bộ mã hóa phân cực 130. Ở bước 206, bộ mã hóa phân cực 130 tạo vectơ đầu vào $\boldsymbol{u} = [u_1\, u_2\, u_3 \ldots\, u_{N_b}]$ tức là dài N_b bit bằng cách ánh xạ từng bit trong m bit đến vị trí tương ứng trong N_b vị trí ở vectơ đầu vào $\boldsymbol{u}$, và sau đó bằng cách đặt các bit

"đóng băng" ở các vị trí còn lại của vectơ đầu vào $\boldsymbol{u}$. Giá trị này và vị trí của các bit đóng băng đã quen thuộc với cả bộ mã hóa phân cực 130 lẫn bộ giải mã phân cực 132. Theo lý thuyết phân cực kênh đằng sau việc xây dựng mã phân cực, một số vị trí của vectơ đầu vào $\boldsymbol{u}$ sẽ có độ tin cậy cao hơn việc được giải mã đúng so với các vị trí khác của vectơ đầu vào $\boldsymbol{u}$. Khi xây dựng mã phân cực, thực hiện phép thử để đặt m bit vào các vị trí tin cậy hơn của vectơ đầu vào $\boldsymbol{u}$, và đặt các bit đóng băng vào các vị trí không đáng tin cậy hơn của vectơ đầu vào $\boldsymbol{u}$. Trong ví dụ trên Fig.3, $N_b = 8$, và các vị trí u_4, u_6, u_7, và u_8 là các vị trí đáng tin cậy hơn của vectơ đầu vào $\boldsymbol{u}$. Do vậy, m bit được đặt ở các vị trí u_4, u_6, u_7, và u_8. Mỗi bit đóng băng có giá trị bằng 0, mặc dù tổng quát hơn các bit đóng băng có thể được gán bằng giá trị khác đã được biết đến với cả bộ mã hóa phân cực 130 lẫn bộ giải mã phân cực 132.

Ở bước 208, sau đó vectơ đầu vào $\boldsymbol{u}$ được nhân với ma trận sinh $\boldsymbol{G}$ để thu thập từ mã $\boldsymbol{x} = [x_1\ x_2\ x_3\ \ldots\ x_{N_b}]$. Trong ví dụ trên Fig.3, $\boldsymbol{x} = \boldsymbol{uG_2}^{\otimes 3}$.

Bộ mã hóa phân cực 130 có thể sử dụng các nhân để thực hiện phép nhân của vectơ đầu vào $\boldsymbol{u}$ với bộ tạo $\boldsymbol{G}$, chẳng hạn để thực hiện bước 208 trên Fig.3. Fig.4 là sơ đồ của nhân 148 để triển khai phép nhân của các đầu vào $[u\ v]$ với ma trận hạt $\boldsymbol{G_2} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. Ma trận hạt cho nhân đôi lúc được gọi là ma trận sinh nhân thay thế. Nhân 148 là nhân nhị phân, và cụ thể là nhân nhị phân Arikan. Các loại nhân nhị phân khác là khả thi. Nhân nhị phân Arikan 148 nhận hai đầu vào u và v, và xuất ra $u + v$ và v, đại diện đầu ra của phép nhân $[u\ v]\begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$. Ký hiệu vòng tròn cộng đại diện phép cộng modulo 2.

Fig.5 là sơ đồ của cấu trúc lấy làm ví dụ để triển khai ví dụ ở bước 208 trên Fig.3, tức là, phép nhân ma trận $\boldsymbol{x} = [x_1\ x_2\ x_3\ \ldots\ x_{N_b}] = \boldsymbol{uG_2}^{\otimes 3}$.

Mỗi nhân 148 trên Fig.5 giống như được minh họa trên Fig.4, và do vậy được chỉ định sử dụng số tham chiếu 148 tương tự. Mỗi nhân cũng được chỉ báo bởi chữ cái "A" trên Fig.5 do nó là nhân nhị phân Arikan. Ba lớp mã hóa được sử dụng, được dán nhãn L_1, L_2, và L_3, với mỗi lớp mã hóa có bốn nhân nhị phân. Lớp mã hóa cũng có thể được gọi là giai đoạn mã hóa.

Mã hóa phân cực có thể được thực hiện có hoặc không có đảo bit. Cấu trúc ví dụ trên Fig.5 không có đảo bit. Cấu trúc ví dụ khác để triển khai bước 208 được thể hiện trên Fig.6. Ví dụ trên Fig.6 thực hiện đảo bit. Nói chung, đầu ra của bộ mã hóa phân cực có thể được biểu diễn dưới dạng $x_0^{N-1} = u_0^{N-1}G_N$, trong đó, không có đảo bit, $G_N = F^{\otimes n}$ là ma trận sinh NxN, $N = 2^n$, $n \geq 1$. Chẳng hạn, đối với $n = 1$, $G_2 = F$. Để đảo bit, $G_N = B_N F^{\otimes n}$, trong đó B_N là ma trận hoán vị đảo bit NxN.

Mã phân cực tổng quát

Các mã phân cực nhị phân được mô tả trên đây dựa trên ma trận hạt $\boldsymbol{G_2} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ bị giới hạn ở bảng chữ cái ký hiệu nhị phân. Mã phân cực tổng quát có thể được xây dựng các mã hóa ký hiệu của bảng chữ cái q-ary, trong đó $q \geq 2$. Vectơ đầu vào $\boldsymbol{u}$ được mã hóa bằng cách sử dụng ma trận sinh $\boldsymbol{G}$ để dẫn đến từ mã $\boldsymbol{x}$. Vectơ đầu vào $\boldsymbol{u}$ có m ký hiệu thông tin, với mỗi ký hiệu thông tin được đại diện bằng cách sử dụng $log_2 q$ bit thông tin. Các vị trí còn lại của vectơ đầu vào $\boldsymbol{u}$ bị đóng băng, tức là đã quen thuộc với cả bộ mã hóa phân cực 130 lẫn bộ giải mã phân cực 132. Phép nhân của vectơ đầu vào $\boldsymbol{u}$ với ma trận sinh $\boldsymbol{G}$ có thể được triển khai bằng cách sử dụng các lớp nhân, trong đó mỗi nhân thực hiện các toán tử trường hữu hạn trong trường Galois GF(q) để thực hiện phép nhân với ma trận hạt $\boldsymbol{G_s}$ của ma trận sinh $\boldsymbol{G}$. Nhân tổng quát 121 lấy làm ví dụ

được minh họa trên Fig.7. Nhân 121 thực hiện toán tử $\boldsymbol{x} = \boldsymbol{u}\boldsymbol{G}_s$, trong đó $\boldsymbol{u} = [u_1\, u_2 \ldots u_{q-1}\, u_q]$ và $\boldsymbol{x} = [x_1\, x_2 \ldots x_{q-1}\, x_q]$. Cả $\boldsymbol{u}$ lẫn $\boldsymbol{x}$ là các vectơ của ký hiệu, mỗi ký hiệu được đại diện bởi $log_2 q$ bit. Nhân nhị phân là trường hợp đặc biệt của nhân tổng quát 121 cho $q = 2$.

Chẳng hạn, mã phân cực tổng quát có thể được định nghĩa rằng mã hóa ký hiệu tứ phân, tức là $q = 4$. Mỗi ký hiệu đầu vào là một trong bốn giá trị khả thi, lần lượt được đại diện bằng cách sử dụng ký hiệu 0, 1, $\propto$, và $\propto^2$. Hai bit được sử dụng khi triển khai để đại diện mỗi giá trị ký hiệu khả thi: 00, 01, 10, và 11. Phần sau là một ví dụ của ma trận hạt $\boldsymbol{G}_s$ có thể được sử dụng:

$$\boldsymbol{G}_s = \begin{bmatrix} 1 & 1 & 1 & 0 \\ \propto & \propto^2 & 1 & 0 \\ \propto^2 & \propto & 1 & 0 \\ 1 & 1 & 1 & 0 \end{bmatrix},$$

trong đó các toán tử trường hữu hạn dưới đây được định nghĩa: $1 + \propto = \propto^2$, $1 + \propto^2 = \propto$, $1 + 0 = 1$, $\propto + 0 = \propto$, $\propto^2 + 0 = \propto^2$, $\propto + \propto^2 = 1$, $1 + 1 = 0$, $\propto + \propto = 0$, $\propto^2 + \propto^2 = 0$, $0 + 0 = 0$, $\propto \propto^2 = 1$, $\propto 1 = \propto$, $\propto \propto = \propto^2$, $0 \times \propto = 0$, $0 \times \propto^2 = 0$, $0 \times 1 = 0$, $0 \times 0 = 0$, $\propto^2 \times 1 = \propto^2$, $\propto^2 \propto^2 = \propto$, và $1 \times 1 = 1$. Nhân tương ứng sẽ được gọi là nhân dựa trên RS và được minh họa trên Fig.8 bằng cách sử dụng số tham chiếu 150. Ký hiệu "RS(4)" được sử dụng để chỉ báo rằng nhân 150 là nhân dựa trên RS để mã hóa ký hiệu tứ phân. Tổng quát hơn, nhân "RS(q)" là nhân dựa trên RS để mã hóa ký hiệu q-ary. Nhân dựa trên RS có thể có các ưu điểm hiệu năng, một số sẽ được mô tả chi tiết hơn dưới đây. Các nhân RS không nhị phân chỉ là một ví dụ. Các nhân phi nhị phân có thể được sử dụng, chẳng hạn nhân Hermitian.

Độ dài từ mã cho ký hiệu tứ phân bị giới hạn ở $N_b = 2 \times 4^n$ bit, trong đó n là số tự nhiên.

Fig.9 là sơ đồ của các cấu trúc bộ mã hóa phân cực RS(4) cho $n = 2$, tức là, 16 ký hiệu. Độ dài từ mã bằng N_b =$2 \times 16 = 32$ bit do mỗi ký hiệu tứ phân được đại diện bởi 2 bit. Hai lớp mã hóa L_1 và L_2 được sử dụng, với mỗi lớp mã hóa có bốn nhân RS(4). Mỗi nhân RS(4) chấp nhận bốn ký hiệu tứ phân, tức là 8 bit, làm đầu vào và tạo đầu ra của bốn ký hiệu tứ phân, tức là 8 bit. Giống với cấu trúc bộ mã hóa phân cực với nhân nhị phân, cấu trúc bộ mã hóa phân cực RS(4) có thể có phép giao hoan chỉ mục ký hiệu hoặc không. Fig.9 minh họa cấu trúc có phép giao hoán chỉ mục ký hiệu. Fig.10 minh họa cấu trúc không có phép giao hoán chỉ mục ký hiệu. Đối với cấu trúc có phép giao hoán chỉ mục ký hiệu (Fig.9), ma trận sinh là ma trận giao hoán lần ma trận sinh của cấu trúc mã hóa không có phép giao hoán chỉ mục ký hiệu. Các phương án thực hiện được bộc lộ ở đây có thể được triển khai có hoặc không có phép giao hoán chỉ mục ký hiệu.

Đối với bảng chữ cái bát phân, nhân sẽ thực hiện phép nhân ma trận $\boldsymbol{x} = \boldsymbol{u}\boldsymbol{G}_s$, trong đó $\boldsymbol{u}$ là vectơ đầu vào của tám ký hiệu và $\boldsymbol{G}_s$ là 8×8 ma trận hạt. Các toán tử trường hữu hạn trong trường Galois GF(8) sẽ được thực hiện. Do vậy, đầu vào của nhân sẽ là tám ký hiệu bát phân, tức là 24 bit do ba bit được sử dụng để đại diện mỗi ký hiệu đầu vào. Đầu ra của nhân sẽ là tám ký hiệu bát phân, tức là lại là 24 bit.

Nói chung, bộ mã hóa phân cực 130 có thể được xây dựng để tạo từ mã $\boldsymbol{x}$ có độ dài N_b bit bằng cách triển khai $\boldsymbol{x} = \boldsymbol{u}\boldsymbol{G}$, trong đó $\boldsymbol{x}$ và $\boldsymbol{u}$ đều đại diện vectơ tương ứng của ký hiệu q phân. Mỗi vectơ $\boldsymbol{x}$ và $\boldsymbol{u}$ có độ dài ký hiệu $N_s = q^n$, tương ứng với độ dài bit $N_b = log_2 q \times q^n$. Mỗi lớp trong n lớp mã hóa có q^{n-1} nhân, và mỗi nhân có q đầu vào và q đầu ra.

Phần sau là các lợi ích khả thi khi sử dụng mã phân cực tổng quát phi nhị phân, chẳng hạn mã dựa trên RS. Với cùng độ dài từ mã, FER thấp hơn cho kích thước danh sách cụ thể trong bộ giải mã có thể đạt được,

hoặc kích thước danh sách nhỏ hơn trong bộ giải mã cho FER cụ thể có thể đạt được. Việc sử dụng kích thước danh sách nhỏ hơn có thể giảm độ phức tạp triển khai và tăng thông lượng giải mã, chẳng hạn, bằng cách có bộ nhớ liên quan danh sách nhỏ hơn để sao chép, di chuyển, và sắp xếp.

So khớp tốc độ/độ dài

Tốc độ mã hóa R được định nghĩa như là $R = k_b/N_b$, trong đó k_b là số lượng bit thông tin, và N_b là độ dài bit của từ mã được tạo tương ứng với k_b bit thông tin. Khi bộ truyền 124 là để truyền k_b bit đến bộ nhận 126, bộ truyền 124 có thể được yêu cầu sử dụng tốc độ mã hóa R/độ dài mã cụ thể, có thể thay đổi theo thời gian, chẳng hạn dựa trên các tài nguyên mạng khả dụng, chẳng hạn băng thông.

Mã phân cực giới hạn ở các giá trị này của N_b. Chẳng hạn, khi thực hiện mã hóa phân cực bằng cách sử dụng chỉ các nhân nhị phân, độ dài từ mã bằng bit, N_b, bị giới hạn ở lũy thừa hai: $N_b = 2^n$ bit. Bảng sau tóm tắt độ dài từ mã N_b, và số lượng lớp mã hóa tương ứng, n, cho các giá trị n khác nhau, lên đến $n = 12$:

n	1	2	3	4	5	6	7	8	9	10	11	12
N_b	2	4	8	16	32	64	128	256	512	1024	2048	4096

Như là ví dụ khác, khi thực hiện mã hóa phân cực bằng cách sử dụng chỉ các nhân RS(4), độ dài từ mã N_b bị giới hạn ở $N_b = 2 \times 4^n$ bit. Bảng sau tóm tắt độ dài từ mã N_b, và số lượng lớp mã hóa tương ứng, n, cho các giá trị khác nhau của n, lên đến $n = 7$:

n	1	2	3	4	5	6	7
N_b	8	32	128	512	2048	8192	32768

Như là ví dụ khác, khi thực hiện mã hóa phân cực bằng cách sử dụng chỉ các nhân RS(8), độ dài từ mã N_b bị giới hạn ở $N_b = 3 \times 8^n$ bit. Bảng sau tóm tắt độ dài từ mã N_b, và số lượng lớp mã hóa tương ứng, n, cho các giá trị khác nhau của n, lên đến $n = 6$:

n	1	2	3	4	5	6
N_b	24	192	1536	12288	98304	786432

Nếu bộ truyền 124 là để truyền ở tốc độ mã hóa R, và bộ truyền 124 có k_b bit để truyền, sau đó độ dài từ mã, bằng bit, được sử dụng bởi bộ mã hóa phân cực 130 một cách lý tưởng là $N_b = k_b/R$. Tuy nhiên, việc giới hạn lên N_b do sử dụng mã phân cực không thể cho phép đối với giá trị N_b chính xác bằng $N_b = k_b/R$. Chẳng hạn, khi thực hiện mã hóa phân cực bằng cách sử dụng chỉ nhân nhị phân, thì N_b bị giới hạn ở lũy thừa hai. Nếu bộ truyền 124 có $k_b = 700$ bit để truyền và tốc độ mã hóa R bộ truyền 124 phải sử dụng là $R = 1/3$, thì một cách lý tưởng $N_b = \frac{700}{\frac{1}{3}} =$ 2100 bit. Tuy nhiên, độ dài từ mã của chính xác N_b =2100 bit không thể được tạo bằng cách sử dụng các nhân nhị phân. Do vậy, bộ truyền 124 thực hiện so khớp tốc độ bằng cách đệm (kéo dài) hoặc đục lỗ (rút ngắn) từ mã để có chính xác $M = 2100$ bit. Việc so khớp tốc độ do vậy được thực hiện bằng cách so khớp độ dài, tức là, chỉnh sửa độ dài của từ mã sao cho tốc độ mã hóa được thỏa mãn.

Fig.11 minh họa hệ thống truyền thông 122 trên Fig.1 theo phương án thực hiện khác. Bộ truyền 124 còn gồm bộ đục lỗ 152. Bộ mã hóa phân cực 130 cụ thể gồm bộ tạo vectơ đầu vào 147, bộ tạo chuỗi thông tin 149, và một hoặc nhiều lớp nhân 151. Bộ truyền 124 có thể cũng gồm vật ghi máy tính đọc được bất biến (không được thể hiện trên hình vẽ), gồm các lệnh để thực thi (chẳng hạn, bởi bộ xử lý hoặc một số hệ mạch khác như

được mô tả trên đây) để triển khai và/hoặc điều khiển hoạt động của bộ mã hóa phân cực 130, và/hoặc bộ đục lỗ 152, và/hoặc bộ truyền 124, và/hoặc để điều khiển theo cách khác việc thực hiện chức năng và/hoặc các phương án thực hiện được mô tả ở đây. Một số phương án thực hiện có thể được triển khai bằng cách sử dụng chỉ phần cứng. Theo một số phương án thực hiện, các lệnh để thực thi bởi bộ xử lý có thể được thực hiện ở dạng sản phẩm phần mềm. Sản phẩm phần mềm có thể được lưu trữ trong bộ nhớ hoặc vật lưu trữ bất biến, có thể là, chẳng hạn, CD-ROM, đĩa nhớ nhanh USB, hoặc đĩa cứng tháo được.

Bộ giải mã phân cực 132 trong bộ nhận 126 cụ thể gồm máy tính tỷ lệ khả năng lôga (log-likelihood ratio – LLR) dạng bit 154, bit LLR đến bộ biến đổi LLR ký hiệu 156, và bộ giải mã 158. Bộ nhận 126 có thể cũng gồm vật ghi máy tính đọc được bất biến (không được thể hiện trên hình vẽ), gồm các lệnh để thực thi (chẳng hạn bởi bộ xử lý hoặc một số hệ mạch khác như được mô tả trên đây) để triển khai và/hoặc điều khiển hoạt động của bộ giải mã phân cực 132 và bộ nhận 126, và/hoặc để điều khiển theo cách khác việc thực hiện chức năng và/hoặc các phương án thực hiện được mô tả ở đây. Một số phương án thực hiện có thể được triển khai bằng cách sử dụng chỉ phần cứng. Theo một số phương án thực hiện, các lệnh để thực thi bởi bộ xử lý có thể được thực hiện ở dạng sản phẩm phần mềm. Sản phẩm phần mềm có thể được lưu trữ trong bộ nhớ hoặc vật lưu trữ bất biến, có thể là, chẳng hạn, CD-ROM, đĩa nhớ nhanh USB, hoặc đĩa cứng tháo được.

Fig.11 sẽ hoặc có thể gồm các linh kiện khác không được minh họa, chẳng hạn bộ điều biến trong bộ truyền 124 và bộ giải điều biến trong bộ nhận 126. Việc điều biến và giải điều biến tương ứng có thể được sử dụng để kích hoạt các LLR tính toán. Các linh kiện khác trong bộ truyền 124 có thể cũng gồm bộ khuếch đại, anten và/hoặc các môđun khác hoặc các

linh kiện của chuỗi truyền hoặc theo cách khác bộ truyền 124 có thể được tạo cấu hình để giao diện với môđun truyền RF riêng rẽ sao cho các từ mã có thể được tạo như được mô tả ở đây và được truyền trực tiếp hoặc bởi khối hoặc môđun truyền riêng rẽ. Các linh kiện khác trong bộ nhận 126 có thể gồm anten, bộ khuếch đại, và/hoặc các môđun khác hoặc các thành phần của chuỗi nhận hoặc theo cách khác có thể được tạo cấu hình để giao diện với môđun tiếp nhận riêng rẽ để xử lý và/hoặc giải mã các từ dựa trên các từ mã của mã phân cực được tiếp nhận bởi bộ nhận 126 trực tiếp hoặc giám tiếp từ môđun hoặc khối tiếp nhận riêng rẽ.

Khi vận hành, k_b bit ở được nhận ở bộ mã hóa phân cực 130. Bộ tạo vectơ đầu vào 147 ánh xạ k_b bit đến các vị trí cụ thể của vectơ đầu vào $\boldsymbol{u}$. Các vị trí còn lại của vectơ đầu vào $\boldsymbol{u}$ được thiết lập làm các vị trí đóng băng. Chuỗi thông tin 155 chỉ báo các vị trí nào của vectơ đầu vào $\boldsymbol{u}$ để nhận mỗi bit trong k_b bit và các vị trí nào của vectơ đầu vào $\boldsymbol{u}$ để nhận các giá trị đóng băng. Chuỗi thông tin 155 được tạo bởi bộ tạo chuỗi thông tin 149 (được tạo trực tuyến hoặc được đọc từ bộ nhớ) dựa trên các hệ số chẳng hạn tốc độ mã hóa R bộ truyền 124 sử dụng để truyền k_b bit, và nhiễu của kênh này 128 hoặc SNR đang hoạt động định trước dựa trên tốc độ mã hóa và độ dài mã hóa. Bộ tạo chuỗi thông tin 149 tạo chuỗi thông tin 155 sao cho cố gắng đặt k_b bit vào các vị trí đáng tin cậy hơn của vectơ đầu vào $\boldsymbol{u}$ và các giá trị đóng băng vào các vị trí ít tin cậy hơn của vectơ đầu vào $\boldsymbol{u}$.

Như đã biết, chuỗi có thứ tự chẳng hạn chuỗi thông tin 155 đại diện của "độ tin cậy" tương đối của các kênh phụ, trong đó kênh phụ đề cập đến kênh tổng hợp sau quá trình phân cực hóa. Nói theo cách khác, một số kênh phụ có dung lượng cao, và một số kênh phụ có dung lượng thấp. Nói cách khác, một số kênh phụ có tỷ lệ tín trên tạp (Signal-to-Noise Ratio – SNR) cao và một số kênh khác có SNR thấp. Các hệ đo này là các

ví dụ của các đặc tính có thể được sử dụng để định lượng hoặc phân loại "độ tin cậy" kênh phụ. Các hệ đó khác chỉ báo độ tin cậy kênh phụ cũng có thể được sử dụng. Việc lựa chọn kênh phụ dựa trên các độ tin cậy của các kênh phụ, và đặc tinh các kênh phụ có độ tin cậy cao nhất được lựa chọn làm các kênh phụ thông tin để mang bit thông tin.

Đối với các mã phân cực chung, chuỗi thông tin 155 có thể là chuỗi các vị trí ký hiệu hoặc chuỗi các vị trí bit. Đối với từ mã có N_b bit sử dụng các nhân q phân, độ dài đủ của chuỗi các vị trí ký hiệu là $N_b/log_2(q)$; trong khi độ dài của chuỗi các vị trí bit là N_b. Lấy các mã phân cực RS(4) làm ví dụ. Đối với chuỗi các vị trí ký hiệu, các vị trí đặt bit thông tin được lựa chọn liên quan đến ký hiệu, tức là, 2 bit lân cận đại diện một ký hiệu được mã hóa nên là cả bit thông tin hoặc cả bit đóng băng; trong khi đối với chuỗi vị trí bit, hai bit lân cận đại diện một ký hiệu có thể chứa 0, một, hoặc hai bit thông tin. Theo một số phương án thực hiện, việc sử dụng chuỗi vị trí ký hiệu có thể có hiệu năng sửa lỗi tốt hơn sử dụng chuỗi vị trí bit, ít nhất theo phương pháp tạo chuỗi được Genie hỗ trợ. Điều này gắn với việc sử dụng chuỗi vị trí ký hiệu có thể triển khai tốt hơn các độ khuếch đại phân cực khi các nhân dựa trên ký hiệu được sử dụng. Chuỗi vị trí ký hiệu thực sự là trường hợp đặc biệt của chuỗi vị trí bit, theo nghĩa là chuỗi vị trí ký hiệu tương đương với chuỗi vị trí bit với ràng buộc rằng hai vị trí bit lân cận $2*i$ và $2*i+1$ $(0 \leq i < N_b/2)$ nên là cả hai vị trí bit thông tin hoặc các vị trí bit đóng băng. Do vậy, nhằm mục đích tổng quát, chuỗi thông tin trong các đoạn văn bản sau đều đề cập đến chuỗi vị trí bit.

Một chuỗi được sắp thứ tự độc lập SNR, được lồng 155 của các kênh phụ có thể được tính toán cho độ dài mã N_{max}, với các chuỗi được sắp thứ tự cho các độ dài mã ngắn hơn N được lựa chọn từ chuỗi N_{max} dài hơn. Nhiều chuỗi sắp thứ tự liên quan đến các độ dài mã mẹ khác nhau N_i có

thể được tính toán thay vào đó, và một trong các chuỗi độ dài mã mẹ có thể được lựa chọn cho mã cụ thể dựa trên độ dài mã được ưu tiên. Tùy chọn khả thi khác bao gồm việc tính toán nhiều chuỗi thứ tự liên quan đến các giá trị SNR, chẳng hạn, và lựa chọn chuỗi thứ tự dựa trên SNR được đo.

Bộ tạo chuỗi thông tin 149 có thể thực hiện các phép tính toán chuỗi thứ tự theo nhiều cách khác nhau. Chẳng hạn, các phép tính toán có thể được thực hiện trực tuyến, tạo chuỗi thứ tự có thể được điều chỉnh hoặc tính toán lại động dựa trên, chẳng hạn, các điều kiện được giám sát. Các phép tính toán có thể theo cách khác được thực hiện ngoại tuyến (chẳng hạn, trước) để tạo các chuỗi thứ tự được tính toán trước (và tĩnh) có thể được lưu trữ và truy xuất hoặc được đọc từ bộ nhớ trong suốt hoạt động mã hóa tiếp theo. Theo tùy chọn khác, các phép tính toán có thể được thực hiện bán phần ngoại tuyến và bán phần trực tuyến.

Vectơ đầu vào $\boldsymbol{u}$ được xuất ra từ bộ tạo vectơ đầu vào 147 được mã hóa bởi một hoặc nhiều lớp nhân 151, mỗi lớp nhân có ít nhất một nhân, dẫn đến từ mã x tương ứng có độ dài N_b bit. Vectơ đầu vào $\boldsymbol{u}$ nên được biến đổi thành vectơ của ký hiệu q phân (được ký hiệu là u_s) trước khi được vận hành với các lớp nhân, và từ mã $\boldsymbol{x}$ có N_b bit nên được biến đổi từ vectơ đầu ra của ký hiệu q phân (được ký hiệu là $\boldsymbol{x_s}$). Một hoặc nhiều lớp nhân 151 thực hiện toán tử $\boldsymbol{x_s = u_s G}$. Mặc dù $\boldsymbol{x_s}$ và $\boldsymbol{u_s}$ là ký hiệu, chúng vẫn được biểu diễn bởi các bit trong phần cứng.

Bộ mã hóa phân cực 130 triển khai mã phân cực trong đó N_b vượt qua số lượng bit $M = k_b/R$ có thể thực sự được truyền dựa trên tốc độ mã hóa R. Từ mã $\boldsymbol{x}$ có độ dài N_b bit do vậy được đục lỗ bởi bộ đục lỗ 152 để loại bỏ các bit, theo mẫu hình đục lỗ 153, để dẫn đến M bit. Sau đó M bit được truyền trên kênh này 128. "Đục lỗ" như được sử dụng ở đây đề cập đến việc loại bỏ các bit khỏi từ mã. Khi thực hiện đục lỗ, từ mã có độ dài

được rút gọn. Từ "rút ngắn" đôi lúc được sử dụng để đề cập đến tình huống cụ thể trong đó mỗi bit được loại bỏ khỏi từ mã có giá trị mà bộ giải mã đã biết, chẳng hạn có thể là trường hợp nếu các bit được loại bỏ khỏi từ mã là tổ hợp tuyến tính của các bit được đóng băng. "Đục lỗ", như được sử dụng ở đây, bao gồm cả "rút ngắn", cũng như các triển khai khác trong đó một, một số, hoặc tất cả các bit được loại bỏ khỏi từ mã có giá trị không được bộ giải mã biết đến. Mẫu hình bất kỳ chỉ báo các bit nào để loại bỏ khỏi từ mã được gọi là mẫu hình đục lỗ. Mẫu hình đục lỗ bao gồm các mẫu hình rút ngắn rút ngắn tập con của các vị trí của từ mã về độ dài được rút ngắn. Mẫu hình đục lỗ cũng bao gồm các loại mẫu hình khác mà đục lỗ các bit nhưng không phải là mẫu hình rút ngắn.

Tín hiệu nhận được mang M bit được xử lý ở bộ giải mã phân cực 132. Máy tính LLR bit 154 trước hết tính toán LLR bit cho một trong M bit. Sau đó giải đục lỗ được thực hiện bằng cách thiết lập LLR bit cho mỗi bit trong các bit được đục lỗ bằng 0, như được thể hiện ở bước 160. Bộ biến đổi LLR bit thành LLR ký hiệu 156 sau đó biến đổi LLR bit thành các LLR ký hiệu tương ứng cho ký hiệu được đại diện bởi N_b bit. Nếu mã hóa phân cực 130 chỉ sử dụng các nhân nhị phân, thì mỗi bit đại diện ký hiệu và do vậy mỗi LLR bit là LLR ký hiệu, và không yêu cầu biến đổi LLR bit thành LLR ký hiệu. Sau đó, các giá trị LLR ký hiệu được xử lý bởi bộ giải mã 158 để tạo quyết định về việc liệu k_b bit được truyền về đây. Thuật toán giải mã lấy làm ví dụ được triển khai bởi bộ giải mã 158 là triệt tiêu liên tiếp dựa trên ký hiệu (SC) hoặc giải mã SCL (successive cancellation list – danh sách triệt tiêu liên tiếp). Fig.12 minh họa một ví dụ về bộ giải mã 158 cho từ mã 32 bit được tạo sử dụng 2 lớp của các nhân RS(4) với phép giao hoán chỉ mục ký hiệu, và trong đó 7 bit thứ nhất của từ mã đã được đục lỗ. Việc giải đục lỗ được thực hiện bằng cách thiết lập các giá trị LLR bit bằng 0 cho mỗi bit trong 7 bit được đục

lỗ, như được thể hiện ở bước 160. Fig.13 minh họa ví dụ khác của bộ giải mã 158, trong đó mẫu hình đục lỗ tương tự (tức là, đục lỗ 7 bit thứ nhất của từ mã) cũng áp dụng cho bộ giải mã sử dụng các lớp nhân RS(4), mà không có phép giao hoán chỉ mục ký hiệu.

Trở về Fig.11, mẫu hình đục lỗ 153 được sử dụng để đục lỗ các bit của từ mã $\boldsymbol{x}$ sao cho N_b bit của từ mã $\boldsymbol{x}$ được rút gọn về $M < N_b$ bit. Mẫu hình đục lỗ 153 chỉ báo các bit nào sẽ được đục lỗ, tức là, cụ thể các bit nào của từ mã $\boldsymbol{x}$ sẽ được loại bỏ. Khi mã phân cực tổng quát được sử dụng mà mã hóa ký hiệu của bảng chữ cái q phân, trong đó $q > 2$, thì việc đục lỗ phải xem xét thực tế là các nhóm bit đại diện ký hiệu khác nhau.

Quyết định về việc mẫu hình đục lỗ 153 và chuỗi thông tin 155 nào để sử dụng phụ thuộc lẫn nhau. Một cách thức để thu thập mẫu hình đục lỗ tối ưu 153 giả thiết chuỗi thông tin cố định 155. Chẳng hạn, trước hết xác định chuỗi thông tin 155, và sau đó giả thiết các vị trí đóng băng được chỉ báo trong chuỗi thông tin 155 (tức là, "tập đóng băng"), tạo mẫu hình đục lỗ tối ưu 153. Như là ví dụ, chuỗi thông tin 155 có thể chỉ báo đặt k_b bit trong k_b vị trí bit cuối cùng trong vectơ đầu vào $\boldsymbol{u}$ do bộ tạo chuỗi thông tin 149 đã xác định rằng k_b vị trí bit cuối cùng trong vectơ đầu vào $\boldsymbol{u}$ là các vị trí đáng tin cậy nhất giả thiết nhiễu trong kênh này và tốc độ mã hóa R. Sau đó mẫu hình đục lỗ 153 có thể được tính toán dựa trên chuỗi thông tin cụ thể 155 để cố gắng đục lỗ các bit trong từ mã $\boldsymbol{x}$ mà tương ứng tốt nhất với các giá trị đóng băng.

Phương pháp khác trước hết là chọn mẫu hình đục lỗ 153 và sau đó, dựa trên mẫu hình đục lỗ 153, tạo chuỗi thông tin tối ưu 155. Chẳng hạn, đối với mẫu hình đục lỗ 153 cụ thể, tập đóng băng, tức là, các vị trí đóng băng trong chuỗi thông tin 155, có thể được tối ưu hóa. Tập đóng băng tối ưu có thể được xác định sử dụng phương pháp tiến hóa mật độ qua phép

xấp xỉ Gaussian hoặc các phương pháp hỗ trợ Gene bằng cách sử dụng các phép mô phỏng. Khi tập đóng băng tối ưu được xác định dựa trên mẫu hình đục lỗ được chọn, thì bộ tạo chuỗi thông tin 149 được chỉnh sửa để không chỉ tạo chuỗi thông tin 155 dựa trên, chẳng hạn, nhiễu kênh (hoặc SNR làm việc) và tốc độ mã hóa R, mà cũng tạo chuỗi thông tin 155 dựa trên mẫu hình đục lỗ 153 được chọn. Chẳng hạn, mẫu hình đục lỗ 153 có thể được chọn để đục lỗ đơn giản $P = N_b - M$ bit thứ nhất của từ mã $\boldsymbol{x}$. Sau đó chuỗi thông tin tối ưu 155 được tạo bởi bộ tạo chuỗi thông tin 149 để xác định các vị trí đóng băng ở vectơ đầu vào $\boldsymbol{u}$ làm các vị trí mà tương ứng tốt nhất với các bit được đục lỗ trong từ mã $\boldsymbol{x}$.

Theo cách khác, phương pháp tối ưu hóa kết hợp có thể được thực hiện trong đó chuỗi thông tin 155 và mẫu hình đục lỗ 153 được tạo cùng nhau để tối ưu hóa kết hợp chuỗi thông tin 155 và mẫu hình đục lỗ 153. Phép tìm kiếm vét cạn hoặc rút gọn thông minh của không gian tìm kiếm có thể được thực hiện khi thực hiện tối ưu hóa kết hợp.

Fig.14 minh họa các đường cong FER cho giải mã phân cực SCL được hỗ trợ CRC trong đó $N_b = 2048$ bit, $k_b = 600$ bit, và $R = 1/3$. Do vậy, $M = 1800$ bit, và do vậy 248 bit được đục lỗ. Các vị trí đóng băng của chuỗi thông tin được tối ưu hóa dựa trên mẫu hình đục lỗ. Để giải mã SC với kích thước danh sách L = 1, độ khuếch đại hiệu năng sử dụng các nhân RS(4) trên các nhân Arikan bằng 0,45dB ở FER = 0,1, và 0,5dB ở FER = 0,01. Để giải mã SCL với kích thước danh sách L = 8, độ khuếch đại hiệu năng bằng 0,15dB ở FER = 0,1, và 0,17dB ở FER = 0,01.

Theo phương án thực hiện, Fig.15 là lưu đồ của phương pháp được thực hiện bởi bộ truyền 124, trong đó mẫu hình đục lỗ 153 được chọn trước và sau đó chuỗi thông tin 155 được tạo. Ở bước 222, bộ truyền 124 xác định độ dài bit M sẽ được truyền trên kênh này 128. Chẳng hạn, độ dài bit M có thể được tính toán như là $M = k_b/R$. Ở bước 224, bộ truyền

124 tạo hoặc xác định mẫu hình đục lỗ 153 bằng cách xây dựng mẫu hình đục lỗ mà đục lỗ $P = N_b - M$ bit thứ nhất của từ mã $\boldsymbol{x}$.

Ở bước 224, việc đục lỗ $P = N_b - M$ bit thứ nhất chỉ là ví dụ. Tập $P = N_b - M$ bit khác có thể được đục lỗ (chẳng hạn, được rút ngắn) thay thế theo mẫu hình đục lỗ khác. Mẫu hình đục lỗ ở bước 224 có thể gồm các mẫu hình rút ngắn xác định tập con nào của các bit từ mã được rút ngắn về độ dài được rút gọn. Theo một số phương án thực hiện, mẫu hình đục lỗ có thể được tạo hoặc được xác định xem xét các tham số khác bên cạnh M, chẳng hạn, độ dài khối thông tin K và/hoặc tốc độ mã hóa R. Chẳng hạn, để hiệu năng sửa lỗi tốt hơn, việc đục lỗ $N_b - M$ thứ nhất có thể được chấp thuận cho các tốc độ mã hóa thấp và trung bình R, trong khi rút ngắn dựa trên khối hoặc rút ngắn đảo bit (BIV) có thể được sử dụng cho các tốc độ mã hóa cao R. Theo các phương án thực hiện khác, mẫu hình đục lỗ 153 có thể được xác định bằng cách lựa chọn mẫu hình đục lỗ (chẳng hạn, dựa trên một hoặc nhiều tham số nêu trên) từ các mẫu hình đục lỗ khả dụng. Sau đó, ở bước 226, bộ truyền 142 gửi hoặc ngược lại làm cho mẫu hình đục lỗ khả dụng được xác định đến bộ tạo chuỗi thông tin 149. Theo cách khác, bộ tạo chuỗi thông tin 149 có thể thu thập độc lập mẫu hình đục lỗ 153 được xác định. Ở bước 228, bộ tạo chuỗi thông tin 149 tạo chuỗi thông tin 155 dựa một phần vào mẫu hình đục lỗ 153.

Theo một số phương án thực hiện, bước 228 bao gồm tính toán chuỗi thông tin 155 cho mỗi từ mã $\boldsymbol{x}$ sử dụng mẫu hình đục lỗ 153 cho từ mã $\boldsymbol{x}$ đó và một số hoặc tất cả các tham số bổ sung sau: (i) độ dài của vectơ đầu vào $\boldsymbol{u}$, bằng N_b bit; (ii) số lượng bit thông tin k_b, (iii) SNR của kênh này 128 (hoặc SNR đang làm việc), và (iv) độ dài bit M sẽ được truyền. Chẳng hạn, phép xấp xỉ Gaussian để tiến hóa mật độ có thể được thực hiện để tìm các vị trí đóng băng tối ưu hóa trong vectơ đầu vào $\boldsymbol{u}$ và nhờ

đó tạo chuỗi thông tin 155. Tuy nhiên, việc thực hiện tính toán ở bước 228 có thể dẫn đến độ phức tạp tính toán tăng lên và độ trễ tăng. Cụ thể là, khi mã phân cực tổng quát được sử dụng mã hóa ký hiệu của bảng chữ cái q phân, trong đó $q > 2$, thì thực hiện bước 228 theo cách trực tuyến trong suốt quá trình vận hành bộ truyền 124 có thể không thực tế. Tùy chọn khác là tiền tính toán ngoại tuyến và lưu trữ trong bộ truyền 124 tất cả các chuỗi thông tin cho tất cả các giá trị khả thi của M. Tuy nhiên, điều này có thể yêu cầu phần lớn bộ nhớ để lưu trữ tất cả các chuỗi thông tin được tính toán trước. Các ràng buộc bộ nhớ trên bộ truyền 124, chẳng hạn không gian bộ nhớ và/hoặc thời gian truy nhập bộ nhớ, có thể không cho phép lưu trữ tất cả các chuỗi thông tin được tiền tính toán.

Do vậy, theo một phương án thực hiện, như được thể hiện trên Fig.16, LUT 302 được lưu trữ trong bộ nhớ 304 trong bộ truyền 124. LUT 302 có thể tương ứng với mẫu hình đục lỗ cụ thể. LUT 302 chỉ báo chuỗi thông tin nào để sử dụng cho các khoảng khác nhau của M. Đối với mỗi khoảng, một chuỗi thông tin được tính toán ngoại tuyến cho giá trị đại diện của M trong khoảng, và sau đó việc chuỗi thông tin được sử dụng giá trị bất kỳ của M trong khoảng. Chẳng hạn, đối với khoảng $M_A \leq M < M_B$, độ dài khối đại diện M_{Rep} được chọn, trong đó $M_A \leq M_{Rep} < M_B$. Chuỗi thông tin được tính toán ngoại tuyến cho M_{Rep} và được lưu trữ trong LUT 302. Sau đó, trong quá trình hoạt động, bất cứ khi nào giá trị này của M trong khoảng $M_A \leq M < M_B$, chuỗi thông tin được lưu trữ trong LUT 302 tương ứng với khoảng $M_A \leq M < M_B$ được sử dụng. Cụ thể, đối với giá trị M bất kỳ trong khoảng $M_A \leq M < M_B$, K vị trí bit đáng tin cậy nhất được chọn để chứa K bit thông tin như được chỉ báo bởi chuỗi thông tin tương ứng của khoảng (tức là, chuỗi thông tin được tạo cho M_{Rep}). Khi M lớn hơn M_{Rep}, việc lựa chọn có thể bỏ qua một hoặc một số vị trí bit bị ép là các vị trí đóng băng do đục lỗ bổ sung so với

M_{Rep}. LUT có thể được viết tắt đáng kể do phân vùng M thành các khoảng và lưu trữ chỉ một chuỗi thông tin trên khoảng, thay vì lưu trữ chuỗi thông tin cho mỗi giá trị khả thi của M.

Fig.17 minh họa ba từ mã 312, 314, và 316. Tốc độ mã hóa R mà ở đó từ mã 312 được truyền sao cho M hơi nhỏ hơn M_B. Tốc độ mã hóa R mà ở đó từ mã 314 được truyền sao cho M bằng M_{Rep}. Tốc độ mã hóa R mà ở đó từ mã 316 được truyền sao cho M bằng M_A. Đối với mỗi từ mã, số lượng bit để đục lỗ ($P = N_b - M$) là khác nhau, nhưng cùng chuỗi thông tin, tương ứng với M_{Rep}, được sử dụng. Trong ví dụ trên Fig.17, các giá trị M_U và M_L được minh họa, trong đó $M_U = M_B - M_{Rep}$ và $M_L = M_{Rep} - M_A$. M_U là một phần của khoảng giữa M_{Rep} và M_B, và M_L là một phần của khoảng giữa M_A và M_{Rep}. M_U lớn hơn M_L. Mặc dù không nhất thiết M_U lớn hơn M_L, khi triển khai M_U có thể lớn hơn M_L đáng kể. Việc sử dụng giá trị M nhỏ hơn M_{Rep} đòi hỏi đục lỗ nhiều hơn M_{Rep}, dẫn đến tốc độ mã hóa cao hơn với ít thông tin hơn khả dụng để dò và/hoặc sửa lỗi, có thể dẫn đến hỏng hóc giải mã và giảm hiệu năng hơn nữa. Do vậy, có thể mong muốn cho M_A gần hơn với M_{Rep}, tức là, M_L nhỏ hơn M_U, như được minh họa trên Fig.17. Tuy nhiên, việc sử dụng giá trị của M lớn hơn M_{Rep} không dẫn đến đục lỗ bổ sung so với M_{Rep}, và do vậy thông tin mã hóa quan trọng có thể được giữ với hầu như không có tổn hao. Do vậy, M_B có thể không cần được giữ gần với M_{Rep}, tức là, M_U có thể lớn hơn M_L, như được minh họa trên Fig.17.

Fig.18 là lưu đồ của phương pháp được thực hiện bởi bộ truyền 124 theo một phương án thực hiện. Ở bước 242, bộ truyền 124 xác định độ dài bit M sẽ được truyền trên kênh này 128. Chẳng hạn, độ dài bit M có thể được tính toán làm $M = k_b/R$. Ở bước 244, bộ truyền 124 tạo hoặc xác định mẫu hình đục lỗ 153 bằng cách xây dựng mẫu hình đục lỗ mà

đục lỗ $P = N_b - M$ bit thứ nhất của từ mã $\boldsymbol{x}$. Ở bước 246, sau đó bộ truyền 142 gửi hoặc khiến giá trị này M khả dụng đến bộ tạo chuỗi thông tin 149. Theo cách khác, bộ tạo chuỗi thông tin 149 có thể tính toán độc lập giá trị này của M. Ở bước 248, bộ tạo chuỗi thông tin 149 tạo chuỗi thông tin 155 bằng cách đọc từ LUT 302 chuỗi thông tin 155 tương ứng với khoảng chứa giá trị này của M. Phương pháp trên Fig.18 có thể được gọi là phương tiện so khớp độ dài/tốc độ ngoại tuyến theo đoạn do chuỗi thông tin được sử dụng cho M_{Rep} đại diện cũng được sử dụng cho các giá trị lân cận của M được chứa trong khoảng được chỉ định.

Ở bước 244, đục lỗ $P = N_b - M$ bit thứ nhất chỉ là ví dụ. Tập khác của $P = N_b - M$ bit có thể được đục lỗ thay vào đó (chẳng hạn, được rút ngắn) theo mẫu hình đục lỗ khác. Mẫu hình đục lỗ ở bước 244 có thể gồm các mẫu hình rút ngắn mà xác định tập con nào của các từ mã bit sẽ được rút ngắn. Theo một số phương án thực hiện, mẫu hình đục lỗ có thể được tạo hoặc được xác định khi xem xét các tham số khác bên cạnh M, chẳng hạn, độ dài khối thông tin K và/hoặc tốc độ mã hóa R. Chẳng hạn, đối với hiệu năng sửa lỗi tốt hơn, việc đục lỗ $N_b - M$ thứ nhất có thể được chấp thuận cho các tốc độ mã hóa thấp và trung bình R, trong khi rút ngắn dựa trên khối hoặc rút ngắn đảo bit (BIV) có thể được sử dụng cho các tốc độ mã hóa cao R. Theo các phương án thực hiện khác, mẫu hình đục lỗ có thể được xác định bằng cách lựa chọn mẫu hình đục lỗ (chẳng hạn, dựa trên một hoặc nhiều tham số lưu ý trên) từ các mẫu hình đục lỗ khả dụng. Theo một số phương án thực hiện, xác định mẫu hình đục lỗ gồm xác định tập đục lỗ, gồm số lượng và các chỉ mục của các kênh phụ được đục lỗ. Phương pháp theo đoạn được mô tả liên quan đến Fig.16 đến Fig.18 cũng áp dụng cho cả mẫu hình đục lỗ mà đục lỗ các bit liên tiếp trong từ mã và các mẫu hình đục lỗ không đục lỗ các bit liên tiếp trong từ mã.

Ở bước 248, các LUT khác nhau có thể được sử dụng nếu các mẫu hình đục lỗ khác nhau được xem xét, chẳng hạn, mỗi mẫu hình đục lỗ tương ứng với LUT. Các LUT khác nhau có thể chứa các số lượng các khoảng khác nhau của M và các chuỗi đại diện liên kết. Chẳng hạn, trên Fig.16, nếu rút ngắn dựa trên khối hoặc rút ngắn BIV được xem xét thay vì đục lỗ vài bit thứ nhất, số lượng khoảng của M có thể thay đổi với các biên khoảng khác nhau một cách khả thi (chẳng hạn, $M_1^{'} \leq M < M_2^{'}, M_2^{'} \leq M < M_3^{'}$, v.v.). Ngoài ra, đối với cùng giá trị của M, LUT có thể chứa chuỗi đại diện khác nhau (chẳng hạn, đối với M mà thỏa mãn $M_1 \leq M < M_2$ và $M_1^{'} \leq M < M_2^{'}$, Chuỗi #1 có thể được sử dụng trong một LUT và chuỗi #1' khác với chuỗi #1 có thể được sử dụng trong LUT khác). Ngoài ra, mặc dù theo một số phương án thực hiện độ dài của mỗi chuỗi trong các chuỗi được lưu là N_b, theo các phương án thực hiện khác, độ dài của một, một số, hoặc tất cả các chuỗi có thể không phải là độ dài khối mã mẹ, tức là, lũy thừa 2. Thay vào đó, chuỗi có độ dài nhỏ hơn N_b, nhưng không nhỏ hơn M_2 có thể được lưu trữ cho khoảng tương ứng của M, trong đó M_2 là biên trên của khoảng tương ứng của M.

Theo một số phương án thực hiện, các chuỗi thông tin khác nhau cho các khoảng khác nhau của M (chẳng hạn chuỗi #1, chuỗi #2,...v.v. trên Fig.16) mà mỗi khoảng có thể tương ứng với cùng độ dài mã mẹ N_b. Theo các phương án thực hiện khác, các chuỗi thông tin khác cho các khoảng khác nhau của M có thể tương ứng với các độ dài mã mẹ khác nhau (chẳng hạn, mỗi chuỗi thông tin/khoảng của M trên Fig.16 có thể tương ứng với độ dài mã mẹ khác nhau).

Lưu đồ trên Fig.18 chỉ báo rằng việc lựa chọn chuỗi, hoặc tạo chuỗi, phụ thuộc vào các mẫu hình đục lỗ cụ thể cũng như độ dài khối mã được truyền M.

Các lợi ích khả thi được liên kết với phương pháp trên Fig.18 như sau. Bằng cách phân chia thành hợp không gian độ dài bit mã hóa và chọn độ dài khối đại diện M_{Rep}, một cho mỗi khoảng, hiệu năng của mã hóa/giải mã phân cực có thể so với hiệu năng của bộ mã hóa phân cực mà tính toán chuỗi thông tin khác nhau cho mỗi giá trị của M. Việc tạo ngoại tuyến cũng khả thi, có thể giảm độ phức tạp triển khai do LUT 302 đang được truy nhập trong suốt quá trình hoạt động, thay vì tính toán chuỗi thông tin trong suốt quá trình hoạt động. Các yêu cầu bộ nhớ có thể được giảm do chuỗi thông tin đang được lưu trữ cho mỗi khoảng của các giá trị độ dài khối, thay vì cho mỗi giá trị độ dài khối khả thi M. Phương pháp trên Fig.18 áp dụng được cho bộ mã hóa mà sử dụng chỉ các mã phân cực nhị phân, chẳng hạn mã phân cực Arikan, cũng như cho bộ mã hóa mà sử dụng các mã phân cực tổng quát phi nhị phân, chẳng hạn các mã phân cực dựa trên RS.

Các phương án thực hiện nêu trên tạo hoặc chọn chuỗi thông tin 155 dựa trên khoảng mà giá trị này M nằm trong, chẳng hạn qua LUT 302. Theo cách khác, khoảng có thể dựa trên tốc độ mã hóa R thay vào đó. Tức là, các khoảng khác nhau của các giá trị cho tốc độ mã hóa R có thể có tốc độ mã hóa đại diện R_{Rep}. Chuỗi thông tin tương ứng với R_{Rep} có thể được sử dụng bất kỳ khi nào mà tốc độ mã hóa nằm trong khoảng được đại diện bởi R_{Rep}. Theo các phương án thực hiện khác, chuỗi thông tin của độ dài khối mã mẹ có thể được tiền cố định, theo đó tập đục lỗ chung được tạo hoặc xác định cho khoảng các độ dài khối mã M hoặc các tốc độ mã hóa R.

Fig.19 đến Fig.22 minh họa các đường cong FER khác nhau. $N_b = 2048$ bit và $k_b = 600$ bit. Tốc độ mã hóa đại diện $R_{Rep} = 1/3$, và do vậy $M_{Rep} = \frac{k_b}{R_{Rep}} = 1800$ bit. Việc giải mã SCL được thực hiện sử dụng

kích thước danh sách L. Các mã phân cực RS(4) được sử dụng trong các phép mô phỏng tương ứng với Fig.19 và Fig.20, và các mã phân cực nhị phân được sử dụng trong các phép mô phỏng tương ứng với Fig.21 và Fig.22. FER cho các kích thước danh sách khác nhau L và các giá trị của M được vẽ. Như có thể thấy từ Fig.19, FER thay đổi nhiều hơn, tương đối với $M_{Rep} = 1800$ bit, khi sử dụng giá trị của M nhỏ hơn 100 bit với M_{Rep} so với việc sử dụng giá trị của M lớn hơn 100 bit so với M_{Rep}.

Fig.23 là lưu đồ của phương pháp được thực hiện bởi bộ truyền, chẳng hạn, bộ truyền 124, theo một phương án thực hiện. Ở bước 402, các bit ở được nhận ở bộ mã hóa phân cực. Ở bước 404, giá trị thu được tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit, và số lượng bit được mã hóa được sử dụng để truyền các bit. Ở bước 406, bộ truyền xác định khoảng của các giá trị mà giá trị này nằm trong đó. Khoảng giá trị này nằm trong sẽ được gọi là khoảng tương ứng, do chính là khoảng mà chứa giá trị này và do vậy là khoảng tương ứng với giá trị này. Chuỗi thông tin cho khoảng tương ứng thu được. Ở bước 408, các bit được ánh xạ đến tập con các vị trí của vectơ đầu vào theo chuỗi thông tin. Ở bước 410, các vị trí còn lại của vectơ đầu vào được thiết lập làm các giá trị đóng băng được bộ giải mã biết đến. Ở bước 412, vectơ đầu vào được mã hóa trong bộ mã hóa phân cực để tạo từ mã.

Theo một số phương án thực hiện, thu thập chuỗi thông tin cho khoảng tương ứng gồm truy xuất từ bộ nhớ chuỗi thông tin được lưu trữ cho khoảng tương ứng. Theo một số phương án thực hiện, chuỗi thông tin cho khoảng tương ứng là chuỗi thông tin được xác định dựa trên giá trị đại diện trong khoảng tương ứng. Theo một số phương án thực hiện, giá trị đại diện gần hơn với một đầu của khoảng tương ứng so với đầu còn lại của khoảng tương ứng.

Theo một số phương án thực hiện, phương pháp còn gồm đục lỗ từ mã theo mẫu hình đục lỗ (chẳng hạn, rút ngắn) để thu thập từ mã được đục lỗ (chẳng hạn được rút ngắn). Độ dài bit của từ mã được đục lỗ (chẳng hạn được rút ngắn) bằng số lượng bit được mã hóa được sử dụng để truyền các bit.

Theo một số phương án thực hiện, từ mã mẹ có độ dài bit N_b, từ mã được đục lỗ (chẳng hạn, được rút ngắn) có độ dài bit M, và mẫu hình đục lỗ là để đục lỗ $(N_b - M)$ bit thứ nhất của từ mã.

Theo một số phương án thực hiện, các bit là các bit thứ nhất, giá trị này là giá trị thứ nhất, chuỗi thông tin là chuỗi thông tin thứ nhất, vectơ đầu vào là vectơ đầu vào thứ nhất, từ mã là từ mã thứ nhất, và từ mã được đục lỗ là từ mã được đục lỗ thứ nhất, và phương pháp bổ sung gồm: (1) tiếp nhận các bit thứ hai ở bộ mã hóa phân cực; (2) thu thập giá trị thứ hai tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit thứ hai, và độ dài bit của từ mã được đục lỗ thứ hai được sử dụng để truyền các bit thứ hai; (3) xác định khoảng giá trị mà giá trị thứ hai nằm trong đó, và thu thập chuỗi thông tin thứ hai tương ứng với khoảng mà giá trị thứ hai nằm trong; (4) ánh xạ các bit thứ hai đến tập con các vị trí của vectơ đầu vào thứ hai theo chuỗi thông tin thứ hai; (5) thiết lập các vị trí còn lại của vectơ đầu vào thứ hai làm các giá trị đóng băng được bộ giải mã biết đến; (6) mã hóa vectơ đầu vào thứ hai trong bộ mã hóa phân cực để tạo từ mã thứ hai; và (7) đục lỗ (chẳng hạn rút ngắn) từ mã thứ hai theo mẫu hình đục lỗ (chẳng hạn, rút ngắn) để thu thập từ mã được đục lỗ thứ hai (chẳng hạn, được rút ngắn).

Theo một số phương án thực hiện, từ mã được đục lỗ thứ nhất (chẳng hạn, được rút ngắn) và từ mã được đục lỗ thứ hai (chẳng hạn, được rút ngắn) có độ dài bit khác nhau. Theo một số phương án thực hiện, khoảng tương ứng chứa giá trị thứ hai giống như khoảng tương ứng chứa giá trị

thứ nhất, và chuỗi thông tin thứ nhất giống như chuỗi thông tin thứ hai. Theo các phương án thực hiện khác, khoảng tương ứng chứa giá trị thứ hai khác với khoảng tương ứng chứa giá trị thứ nhất, và chuỗi thông tin thứ nhất khác với chuỗi thông tin thứ hai.

Theo một số phương án thực hiện, ít nhất một số bit biểu diễn các ký hiệu q phân, trong đó $q > 2$. Chẳng hạn, các bit có thể đều biểu diễn các ký hiệu q phân, và từ mã có thể bao gồm ký hiệu q phân được biểu diễn bởi các bit.

Theo một số phương án thực hiện, mã hóa vectơ đầu vào gồm mã hóa vectơ đầu vào bằng cách sử dụng ít nhất một nhân bộ mã hóa phân cực để tạo từ mã. Việc mã hóa có thể gồm: tiếp nhận, ở nhân bộ mã hóa phân cực, tập của ký hiệu q phân đầu vào được biểu diễn bởi các bit; và biến đổi tập của ký hiệu q phân đầu vào, theo ma trận hạt của nhân bộ mã hóa phân cực, để tạo tập của ký hiệu q phân đầu ra được biểu diễn bởi các bit. q có thể lớn hơn hoặc bằng 2. Ma trận hạt của nhân bộ mã hóa phân cực đôi lúc được gọi là ma trận sinh nhân.

Fig.24 minh họa hệ thống truyền thông 1300 lấy làm ví dụ trong đó các phương án thực hiện sáng chế có thể được triển khai. Nói chung, hệ thống 1300 cho phép nhiều phần tử không dây hoặc có dây để truyền thông dữ liệu và nội dung khác. Mục đích của hệ thống 1300 có thể là dề xuất nội dung (thoại, dữ liệu, video, văn bản) qua phát quảng bá, phát hẹp, thiết bị người dùng đến thiết bị người dùng, v.v.. Hệ thống 1300 có thể hoạt động hiệu quả bằng cách chia sẻ các tài nguyên chẳng hạn băng thông.

Trong ví dụ này, hệ thống truyền thông 1300 gồm các thiết bị điện tử (electronic device – ED) từ 1310a đến 1310c, các mạng truy nhập vô tuyến (radio access network – RAN) từ 1320a đến 1320b, mạng lõi (core network – CN) 1330, mạng điện thoại chuyển mạch công cộng (public

switched telephone network – PSTN) 1340, mạng Internet 1350, và các mạng khác 1360. Trong khi số lượng thành phần hoặc phần tử cụ thể được thể hiện trên Fig.24, số lượng thành phần hoặc phần tử hợp lý bất kỳ có thể được bao gồm trong hệ thống 1300.

Các ED từ 1310a đến 1310c và các BS từ 1370a đến 1370b là các ví dụ của thiết bị truyền thông có thể được tạo cấu hình để triển khai một số hoặc tất cả các chức năng và/hoặc các phương án thực hiện được mô tả ở đây. Chẳng hạn, thiết bị bất kỳ trong các ED từ 1310a đến 1310c và các BS từ 1370a đến 1370b có thể được tạo cấu hình để triển khai chức năng mã hóa hoặc giải mã (hoặc cả hai) được mô tả trên đây. Trong ví dụ khác, thiết bị bất kỳ trong các ED từ 1310a đến 1310c và các BS từ 1370a đến 1370b có thể gồm bộ truyền 124, bộ nhận 126, hoặc cả hai được mô tả trên đây.

Các ED từ 1310a đến 1310c được tạo cấu hình để vận hành, truyền thông, hoặc cả hai, trong hệ thống 1300. Chẳng hạn, các ED từ 1310a đến 1310c được tạo cấu hình để truyền, nhận, hoặc cả hai qua các kênh truyền thông không dây hoặc có dây. Mỗi ED từ 1310a đến 1310c đại diện thiết bị người dùng cuối thích hợp bất kỳ cho hoạt động không dây và có thể gồm các thiết bị này (hoặc có thể được gọi là) như là thiết bị người dùng (user equipment – UE), khối thu/phát không dây (wireless transmit/receive unit – WTRU), trạm di động, khối thuê bao di động hoặc cố định, điện thoại tế bào, trạm (station – STA), thiết bị truyền thông loại máy (machine type communication – MTC), hỗ trợ số cá nhân (personal digital assistant – PDA), điện thoại thông minh, máy tính xách tay, máy tính, vùng phím chạm, bộ cảm biến không dây, hoặc thiết bị điện tử người tiêu dùng.

Trên Fig.24, các RAN từ 1320a đến 1320b lần lượt gồm các BS từ 1370a đến 1370b. Mỗi BS từ 1370a đến 1370b được tạo cấu hình để tạo

giao diện không dây với một hoặc nhiều ED từ 1310a đến 1310c để cho phép truy nhập vào BS khác bất kỳ từ 1370a đến 1370b, CN 1330, PSTN 1340, Internet 1350, và/hoặc các mạng khác 1360. Chẳng hạn, các BS từ 1370a đến 1370b có thể gồm (hoặc là) một hoặc nhiều thiết bị đã biết, chẳng hạn BTS (trạm thu phát station gốc), Node-B (NodeB), nút B tiến hóa (evolved NodeB – eNB), eNB thường trú, gNodeB (đôi lúc được gọi là "gigabit" NodeB), điểm truyền (transmission point – TP), bộ điều khiển tại địa điểm, điểm truy nhập (access point – AP), hoặc bộ định tuyến không dây. ED từ 1310a đến 1310c bất kỳ có thể theo cách khác hoặc đồng thời được tạo cấu hình để tạo giao diện, truy nhập, hoặc truyền thông với BS bất kỳ từ 1370a đến 1370b, Internet 1350, CN 1330, PSTN 1340, các mạng khác 1360, hoặc tổ hợp bất kỳ của các thiết bị nêu trên. Một cách tùy chọn, hệ thống có thể gồm các RAN, chẳng hạn RAN 1320b, trong đó BS 1370b tương tự truy nhập CN 1330 qua Internet 1350, như được thể hiện trên hình vẽ.

Theo phương án thực hiện được thể hiện trên Fig.24, BS 1370a tạo một phần của RAN 1320a, có thể gồm các BS khác, các bộ điều khiển trạm gốc (base station controller – BSC), các bộ điều khiển mạng vô tuyến (radio network controller – RNC), các nút chuyển tiếp, các phần tử, và/hoặc các thiết bị. BS 1370a, 1370b bất kỳ có thể là một phần tử, như được thể hiện trên hình vẽ, hoặc nhiều phần tử, được phân tán trong RAN tương ứng, hoặc ngược lại. BS 1370b cũng tạo một phần RAN 1320b, có thể gồm các BS khác, các phần tử, và/hoặc các thiết bị. Mỗi BS từ 1370a đến 1370b có thể được tạo cấu hình để vận hành để truyền và/hoặc nhận các tín hiệu không dây trong khu vực hoặc vùng địa lý cụ thể, đôi lúc được gọi là "tế bào." Tế bào có thể còn được phân chia thành các đoạn tế bào, và BS từ 1370a đến 1370b có thể, chẳng hạn, tận dụng nhiều bộ thu phát để cung cấp dịch vụ cho nhiều đoạn. Theo một số phương án thực

hiện, BS từ 1370a đến 1370b có thể thiết lập các pico tế bào hoặc các femto tế bào trong đó công nghệ truy nhập vô tuyến (radio access technology – RAT) hỗ trợ điều này. Theo một số phương án thực hiện, công nghệ nhiều đầu vào nhiều đầu ra (multiple-input multiple-output – MIMO) có thể được tận dụng có nhiều bộ thu phát cho mỗi tế bào. Số lượng RAN từ 1320a đến 1320b được thể hiện chỉ lấy làm ví dụ. Số lượng RAN bất kỳ có thể được xem xét khi phát minh hệ thống 1300.

Các BS từ 1370a đến 1370b truyền thông với một hoặc nhiều ED từ 1310a đến 1310c trên một hoặc nhiều giao diện không khí 1390 sử dụng các liên kết truyền thông không dây chẳng hạn RF, μWave, IR, v.v.. Các giao diện không khí 1390 có thể tận dụng RAT thích hợp bất kỳ. Chẳng hạn, hệ thống 1300 có thể triển khai một hoặc nhiều phương pháp truy nhập kênh, chẳng hạn đa truy nhập phân chia mã (code division multiple access – CDMA), đa truy nhập phân chia thời gian (time division multiple access – TDMA), đa truy nhập phân chia tần số (frequency division multiple access – FDMA), FDMA trực giao (orthogonal FDMA – OFDMA), hoặc đơn kênh mang (single-carrier – SC) FDMA trong các giao diện không khí 1390.

BS từ 1370a đến 1370b có thể triển khai UMTS (hệ thống viễn thông di động toàn cầu) truy nhập vô tuyến mặt đất (Terrestrial Radio Access – UTRA) để thiết lập giao diện không khí 1390 sử dụng WCDMA. Khi làm vậy, BS từ 1370a đến 1370b có thể triển khai các giao thức chẳng hạn HSPA, HSPA+ một cách tùy chọn gồm HSDPA, HSUPA hoặc cả hai. Theo cách khác, BS từ 1370a đến 1370b có thể thiết lập giao diện không khí 1390 với UTRA tiến hóa (Evolved UTRA – E-UTRA) sử dụng LTE, LTE-A, và/hoặc LTE-B. Cân nhắc rằng hệ thống 1300 có thể sử dụng chức năng đa truy nhập kênh, gồm các phương tiện như được mô tả trên đây. Các công nghệ vô tuyến để triển khai các giao diện không khí gồm

IEEE 802.11, 802.15, 802.16, CDMA2000, CDMA2000 1X, CDMA2000 EV-DO, IS-2000, IS-95, IS-856, GSM, EDGE, và GERAN. Dĩ nhiên, các phương tiện đa truy nhập khác và các giao thức không dây có thể được tận dụng.

Các RAN từ 1320a đến 1320b truyền thông với CN 1330 để cấp cho các ED từ 1310a đến 1310c các dịch vụ khác nhau chẳng hạn giọng nói, dữ liệu, và các dịch vụ khác nhau. Có thể hiểu rằng, các RAN từ 1320a đến 1320b và/hoặc CN 1330 có thể truyền thông trực tiếp hoặc gián tiếp với một hoặc nhiều RAN khác (không được thể hiện trên hình vẽ), có thể hoặc không thể được phục vụ trực tiếp bởi CN 1330, và có thể hoặc không thể sử dụng cùng RAN với RAN 1320a, RAN 1320b hoặc cả hai. CN 1330 cũng có thể phục vụ truy nhập cổng nối giữa (i) các RAN từ 1320a đến 1320b hoặc các ED từ 1310a đến 1310c hoặc cả hai, và (ii) các mạng khác (chẳng hạn PSTN 1340, Internet 1350, và các mạng khác 1360). Ngoài ra, một số hoặc tất cả các ED từ 1310a đến 1310c có thể gồm chức năng truyền thông với các mạng không dây khác nhau trên các liên kết không dây khác nhau sử dụng các công nghệ không dây khác nhau và/hoặc các giao thức. Thay cho việc truyền thông không dây (hoặc bên cạnh đó), các ED từ 1310a đến 1310c có thể truyền thông qua các kênh truyền thông hữu tuyến đến nhà cung cấp dịch vụ hoặc bộ chuyển mạch (không được thể hiện trên hình vẽ), và với mạng Internet 1350. PSTN 1340 có thể gồm các mạng điện thoại chuyển mạch để cung cấp dịch vụ điện thoại cũ thông thường (plain old telephone service – POTS). Internet 1350 có thể gồm mạng máy tính và các mạng con (các mạng Intranet) hoặc cả hai, và đưa vào các giao thức, chẳng hạn IP, TCP, UDP. Các ED từ 1310a đến 1310c có thể các thiết bị đa chế độ có thể vận hành theo nhiều RAT, và đưa vào nhiều bộ thu phát cần để hỗ trợ điều này.

Fig.25 và Fig.26 minh họa các thiết bị lấy làm ví dụ có thể triển khai chức năng và/hoặc các phương án thực hiện được mô tả trên đây. Cụ thể là, Fig.25 minh họa ED 1310 lấy làm ví dụ, và Fig.26 minh họa BS 1370 lấy làm ví dụ. Các thành phần này có thể được sử dụng trong hệ thống 1300 hoặc trong hệ thống thích hợp khác bất kỳ.

Như được thể hiện trên Fig.25, ED 1310 gồm ít nhất một khối xử lý 1400. Khối xử lý 1400 triển khai các hoạt động xử lý khác nhau của ED 1310. Chẳng hạn, khối xử lý 1400 có thể thực hiện mã hóa tín hiệu, xử lý dữ liệu, điều khiển công suất, xử lý nhập/xuất (input/output – I/O), hoặc chức năng khác bất kỳ cho phép ED 1310 hoạt động trong hệ thống 1300. Khối xử lý 1400 cũng có thể được tạo cấu hình để triển khai một số hoặc tất cả chức năng và/hoặc các phương án thực hiện được mô tả trên đây. Mỗi khối xử lý 1400 gồm thiết bị tính toán hoặc xử lý thích hợp bất kỳ được tạo cấu hình để thực hiện một hoặc nhiều hoạt động. Mỗi khối xử lý 1400 có thể, chẳng hạn, gồm bộ vi xử lý, bộ vi điều khiển, DSP, FPGA, hoặc ASIC.

ED 1310 cũng gồm ít nhất một bộ thu phát 1402. Bộ thu phát 1402 được tạo cấu hình để điều biến dữ liệu hoặc nội dung khác để truyền bằng ít nhất một anten hoặc bộ điều khiển giao diện mạng (Network Interface Controller – NIC) 1404. Bộ thu phát 1402 cũng được tạo cấu hình để giải điều biến dữ liệu hoặc nội dung khác được tiếp nhận bởi ít nhất một anten 1404. Mỗi bộ thu phát 1402 gồm cấu trúc thích hợp bất kỳ để tạo các tín hiệu cho phiên truyền không dây hoặc có dây và/hoặc các tín hiệu xử lý được tiếp nhận không dây hoặc có dây. Mỗi anten 1404 gồm cấu trúc thích hợp bất kỳ để truyền và/hoặc tiếp nhận các tín hiệu không dây hoặc có dây. Một hoặc nhiều bộ thu phát 1402 có thể được sử dụng trong ED 1310, và một hoặc nhiều anten 1404 có thể được sử dụng trong ED 1310. Mặc dù được thể hiện như là một khối chức năng, bộ thu phát 1402 có thể

cũng được triển khai bằng cách sử dụng ít nhất một bộ truyền và ít nhất một bộ nhận riêng rẽ. Theo một số phương án thực hiện, bộ thu phát 1402 có thể triển khai bộ truyền 124 và/hoặc bộ nhận 126 được mô tả trước đó.

ED 1310 còn gồm một hoặc nhiều thiết bị I/O 1406 hoặc các giao diện (chẳng hạn giao diện có dây với mạng Internet 1350). Các thiết bị I/O 1406 tạo thuận tiện tương tác với người dùng hoặc các thiết bị người dùng khác (truyền thông mạng) trong mạng. Mỗi thiết bị I/O 1406 gồm cấu trúc thích hợp bất kỳ để cấp thông tin cho hoặc tiếp nhận/cấp thông tin từ người dùng, chẳng hạn loa, micrô, vùng phím bấm, bàn phím, phần hiển thị, hoặc màn hình cảm ứng, gồm truyền thông giao diện mạng.

Ngoài ra, ED 1310 gồm ít nhất một bộ nhớ 1408. Bộ nhớ 1408 lưu trữ lệnh và dữ liệu được sử dụng, được tạo, hoặc được tập hợp bởi ED 1310. Chẳng hạn, bộ nhớ 1408 có thể lưu trữ các lệnh phần mềm hoặc các môđun được tạo cấu hình để triển khai một số hoặc tất cả chức năng và/hoặc các phương án thực hiện được mô tả trên đây và được thực thi bởi các khối xử lý 1400. Mỗi bộ nhớ 1408 gồm bộ lưu trữ bất biến và/hoặc khả biến thích hợp bất kỳ và các thiết bị truy xuất. Loại bộ nhớ thích hợp bất kỳ có thể được sử dụng, chẳng hạn bộ nhớ truy xuất ngẫu nhiên (random access memory – RAM), bộ nhớ chỉ đọc (read only memory – ROM), đĩa cứng, đĩa quang, thẻ môđun danh tính thuê bao (subscriber identity module – SIM), thẻ nhớ, thẻ nhớ số bảo mật (secure digital – SD), và tương tự.

Như được thể hiện trên Fig.26, BS 1370 gồm ít nhất một khối xử lý 1450, ít nhất một bộ truyền 1452 (có thể hoặc không thể giống bộ truyền 124 được mô tả trước đó), ít nhất một bộ nhận 1454 (có thể hoặc không thể giống với bộ nhận 126 được mô tả trước đó), một hoặc nhiều anten 1456, ít nhất một bộ nhớ 1458, và một hoặc nhiều thiết bị hoặc giao diện I/O 1466. Bộ thu phát, không được thể hiện trên hình vẽ, có thể được sử

dụng thay cho bộ truyền 1452 và bộ nhận 1454. Bộ lập lịch 1453 có thể được ghép nối với khối xử lý 1450. Bộ lập lịch 1453 có thể được bao gồm trong hoặc được hoạt động riêng rẽ với BS 1370. Khối xử lý 1450 triển khai các hoạt động xử lý khác nhau của BS 1370, chẳng hạn mã hóa tín hiệu, xử lý dữ liệu, điều khiển công suất, xử lý I/O, hoặc chức năng khác bất kỳ. Khối xử lý 1450 cũng có thể được tạo cấu hình để triển khai một số hoặc tất cả chức năng và/hoặc các phương án thực hiện được mô tả chi tiết hơn trên đây. Mỗi khối xử lý 1450 gồm thiết bị tính toán hoặc xử lý thích hợp bất kỳ được tạo cấu hình để thực hiện một hoặc nhiều hoạt động. Mỗi khối xử lý 1450 có thể, chẳng hạn, gồm bộ vi xử lý, bộ vi xử lý, DSP, FPGA, hoặc ASIC.

Mỗi bộ truyền 1452 gồm cấu trúc thích hợp bất kỳ để tạo các tín hiệu cho phiên truyền có dây hoặc không dây đến một hoặc nhiều ED hoặc các thiết bị khác. Mỗi bộ nhận 1454 gồm cấu trúc thích hợp bất kỳ để xử lý các tín hiệu được nhận không dây hoặc có dây từ một hoặc nhiều ED hoặc các thiết bị khác. Mặc dù được thể hiện dưới dạng các thành phần riêng rẽ, ít nhất một bộ truyền 1452 và ít nhất một bộ nhận 1454 có thể được kết hợp vào bộ thu phát. Mỗi anten 1456 gồm cấu trúc thích hợp để truyền và/hoặc tiếp nhận các tín hiệu không dây hoặc có dây. Trong khi anten chung 1456 được thể hiện ở đây như được ghép nối với cả bộ truyền 1452 lẫn bộ nhận 1454, một hoặc nhiều anten 1456 có thể được ghép nối với các bộ truyền 1452, và một hoặc nhiều anten riêng rẽ 1456 có thể được ghép nối với các bộ nhận 1454. Mỗi bộ nhớ 1458 gồm bộ nhớ bất biến và/hoặc khả biến thích hợp bất kỳ và các thiết bị truy xuất chẳng hạn các thiết bị được mô tả trên đây kết nối với ED 1310. Bộ nhớ 1458 lưu trữ lệnh và dữ liệu được sử dụng, được tạo, hoặc được thu thập bởi BS 1370. Chẳng hạn, bộ nhớ 1458 có thể lưu trữ các lệnh hoặc các môđun phần mềm được tạo cấu hình để triển khai một số hoặc tất cả chức

năng và/hoặc các phương án thực hiện được mô tả trên đây và được thực thi bởi các khối xử lý 1450.

Mỗi thiết bị I/O 1466 tạo thuận tiện tương tác với người dùng hoặc các thiết bị khác (truyền thông mạng) trong mạng. Mỗi thiết bị I/O 1466 gồm cấu trúc thích hợp bất kỳ cung cấp thông tin đến hoặc tiếp nhận/cấp thông tin từ người dùng, gồm các truyền thông giao diện mạng.

Mặc dù sáng chế được mô tả dựa vào các dấu hiệu cụ thể và các phương án thực hiện của nó, các chỉnh sửa và các tổ hợp khác nhau có thể được thực hiện với nó mà không xa rời sáng chế. Phần mô tả và các hình vẽ do đó được đề cập đơn giản như là minh họa của một số phương án thực hiện sáng chế như được định nghĩa bởi các điểm yêu cầu bảo hộ đi kèm, và được cân nhắc bao gồm bất kỳ hoặc tất cả các chỉnh sửa, biến thể, tổ hợp hoặc các tương đương nằm trong phạm vi của sáng chế. Do vậy, mặc dù sáng chế và các ưu điểm của nó được mô tả chi tiết, các thay đổi khác nhau, các thay thế và các biến đổi có thể được hiện trong đó mà không xa rời sáng chế như được định nghĩa bởi các điểm yêu cầu bảo hộ đi kèm. Ngoài ra, phạm vi của sáng chế không được nhằm bị giới hạn ở các phương án thực hiện cụ thể của quá trình, máy móc, sản xuất, hợp chất, các phương tiện, các phương pháp và các bước được mô tả trong bản mô tả. Khi người có kiến thức trung bình trong lĩnh vực sẽ dễ hiểu từ bản mô tả của sáng chế, các quá trình, máy móc, sản xuất, hợp chất, phương tiện, các phương pháp, hoặc các bước, hiện có hoặc sau này được phát triển, thực hiện chức năng gần như tương tự hoặc đạt được kết quả gần như giống với các phương án thực hiện tương ứng được mô tả ở đây có thể được tận dụng theo sáng chế. Do đó, các điểm yêu cầu bảo hộ đi kèm được nhằm bao gồm trong phạm vi các quá trình, máy móc, sản xuất, hợp chất, phương tiện, phương pháp, hoặc các bước.

Ngoài ra, môđun bất kỳ, thành phần, hoặc thiết bị được lấy ví dụ ở đây mà thực thi các lệnh có thể gồm hoặc theo cách khác truy nhập vào phương tiện lưu trữ bộ xử lý/ máy tính đọc được bất biến để lưu trữ thông tin, chẳng hạn các lệnh máy tính/bộ xử lý đọc được, các cấu trúc dữ liệu, các môđun chương trình, và/hoặc dữ liệu khác. Danh sách vét cạn của các ví dụ của phương tiện lưu trữ máy tính/bộ xử lý bất biến đọc được gồm các băng cát xét từ tính, băng từ, bộ lưu trữ đĩa từ hoặc các thiết bị lưu trữ từ tính khác, đĩa quang chẳng hạn CD-ROM, các đĩa viđêo số hoặc các đĩa đa dụng số (digital versatile disc – DVD), Blu-ray Disc™, hoặc bộ lưu trữ quang khác, phương tiện không tháo được, tháo được và bất biến được triển khai trong phương pháp hoặc công nghệ bất kỳ, RAM, ROM, ROM lập trình được xóa được bằng điện (electrically erasable programmable ROM – EEPROM), bộ nhớ nhanh hoặc công nghệ bộ nhớ khác. Phương tiện lưu trữ máy tính/bộ xử lý bất biến bất kỳ có thể là một phần của thiết bị hoặc truy nhập được hoặc kết nối được với nó. Ứng dụng hoặc môđun bất kỳ được mô tả ở đây có thể được triển khai bằng cách sử dụng các lệnh máy tính/bộ xử lý đọc được/thực thi được có thể được lưu trữ hoặc theo cách khác được giữ bởi phương tiện lưu trữ đọc được bằng máy tính/bộ xử lý bất biến.

YÊU CẦU BẢO HỘ

1. Phương pháp truyền được thực hiện ở bộ truyền bao gồm các bước:

tiếp nhận các bit ở bộ mã hóa phân cực;

thu thập giá trị tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit, và số lượng bit được mã hóa được sử dụng để truyền các bit;

xác định khoảng giá trị mà giá trị nêu trên nằm trong đó;

thu thập chuỗi thông tin tương ứng với khoảng mà giá trị nằm trong đó, trong đó chuỗi thông tin là tương tự đối với bất kỳ giá trị nào nằm trong khoảng này, và trong đó chuỗi thông tin này chỉ báo tập hợp con của các vị trí của vectơ đầu vào cần nhận các bit;

ánh xạ các bit đến tập con các vị trí của vectơ đầu vào theo chuỗi thông tin;

thiết lập các vị trí còn lại của vectơ đầu vào làm các giá trị đóng băng được bộ giải mã biết đến; và

mã hóa vectơ đầu vào trong bộ mã hóa phân cực để tạo từ mã.

2. Phương pháp theo điểm 1, trong đó chuỗi thông tin tương ứng với khoảng là chuỗi thông tin được xác định dựa trên giá trị đại diện trong khoảng này.

3. Phương pháp theo điểm 2, trong đó giá trị đại diện gần hơn với một đầu của khoảng so với đầu còn lại của khoảng.

4. Phương pháp theo điểm bất kỳ trong số các điểm 1 đến 3, còn bao gồm đục lỗ từ mã theo mẫu hình đục lỗ để thu thập từ mã được đục lỗ, trong đó độ dài bit của từ mã được đục lỗ bằng số lượng bit được mã hóa được sử dụng để truyền các bit.

5. Phương pháp theo điểm 4, trong đó việc đục lỗ từ mã theo mẫu hình đục lỗ bao gồm rút ngắn từ mã.

6. Phương pháp theo điểm 4 hoặc 5, trong đó từ mã có độ dài bit N_b, từ mã được đục lỗ có độ dài bit M, và trong đó mẫu hình đục lỗ là để đục lỗ $(N_b - M)$ bit thứ nhất của từ mã.

7. Phương pháp theo điểm bất kỳ trong số các điểm 4 đến 6, trong đó các bit là các bit thứ nhất, giá trị này là giá trị thứ nhất, chuỗi thông tin là chuỗi thông tin thứ nhất, vectơ đầu vào là vectơ đầu vào thứ nhất, từ mã là từ mã thứ nhất, và từ mã được đục lỗ là từ mã được đục lỗ thứ nhất; và trong đó phương pháp còn bao gồm các bước:

tiếp nhận các bit thứ hai ở bộ mã hóa phân cực;

thu thập giá trị thứ hai tương ứng với ít nhất một trong: tốc độ mã hóa được sử dụng để truyền các bit thứ hai, và độ dài bit của từ mã được đục lỗ thứ hai được sử dụng để truyền các bit thứ hai;

xác định khoảng giá trị mà giá trị thứ hai nằm trong đó, và thu thập chuỗi thông tin thứ hai tương ứng với khoảng mà giá trị thứ hai nằm trong đó;

ánh xạ các bit thứ hai đến tập con các vị trí của vectơ đầu vào thứ hai theo chuỗi thông tin thứ hai;

thiết lập các vị trí còn lại của vectơ đầu vào thứ hai làm các giá trị đóng băng được bộ giải mã biết đến;

mã hóa vectơ đầu vào thứ hai trong bộ mã hóa phân cực để tạo từ mã thứ hai; và

đục lỗ từ mã thứ hai theo mẫu hình đục lỗ để thu thập từ mã được đục lỗ thứ hai.

8. Phương pháp theo điểm 7, trong đó từ mã được đục lỗ thứ nhất và từ mã được đục lỗ thứ hai có độ dài bit khác nhau.

9. Phương pháp theo điểm 7 hoặc 8, trong đó khoảng mà giá trị thứ hai nằm trong đó giống như khoảng mà giá trị thứ nhất nằm trong đó, và trong đó chuỗi thông tin thứ nhất giống như chuỗi thông tin thứ hai.

10. Phương pháp theo điểm 7 hoặc 8, trong đó khoảng mà giá trị thứ hai nằm trong đó khác với khoảng mà giá trị thứ nhất nằm trong đó, và trong đó chuỗi thông tin thứ nhất khác với chuỗi thông tin thứ hai.

11. Phương pháp theo điểm bất kỳ trong số các điểm 1 đến 10, trong đó ít nhất một số bit biểu diễn các ký hiệu q phân, trong đó $q > 2$.

12. Phương pháp theo điểm bất kỳ trong số các điểm 1 đến 10, trong đó các bit biểu diễn các ký hiệu q phân, trong đó $q > 2$, và từ mã bao gồm các ký hiệu q phân được biểu diễn bởi các bit.

13. Phương pháp theo điểm 11 hoặc 12, trong đó việc mã hóa vectơ đầu vào bao gồm:

mã hóa vectơ đầu vào bằng cách sử dụng ít nhất một nhân bộ mã hóa phân cực để tạo từ mã, việc mã hóa gồm các bước:

tiếp nhận, ở nhân bộ mã hóa phân cực, tập của các ký hiệu q phân đầu vào được biểu diễn bởi các bit; và

biến đổi tập của các ký hiệu q phân đầu vào, theo ma trận hạt của nhân bộ mã hóa phân cực, để tạo tập của các ký hiệu q phân đầu ra được biểu diễn bởi các bit.

14. Phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 13, trong đó việc thu thập chuỗi thông tin tương ứng với khoảng bao gồm:

truy xuất từ bộ nhớ chuỗi thông tin được lưu trữ tương ứng với khoảng này.

15. Bộ truyền (124, 1310) bao gồm:

bộ xử lý (1400); và

bộ nhớ máy tính đọc được bất biến (1408) lưu trữ các lệnh để thực thi bởi bộ xử lý để triển khai các bước ở phương pháp theo điểm bất kỳ trong số các điểm từ 1 đến 14.

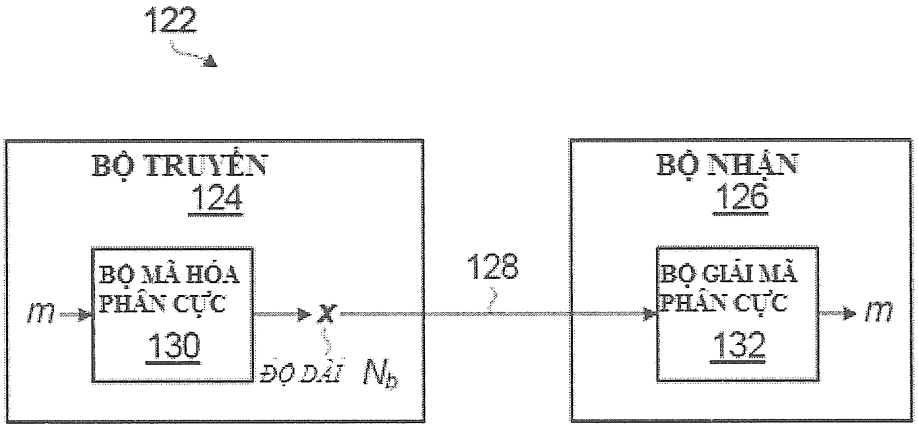

122
BỘ TRUYỀN
124
BỘ MÃ HÓA
PHÂN CỰC
130
m
x
ĐỘ DÀI N_b
128
BỘ NHẬN
126
BỘ GIẢI MÃ
PHÂN CỰC
132
m

Fig.1

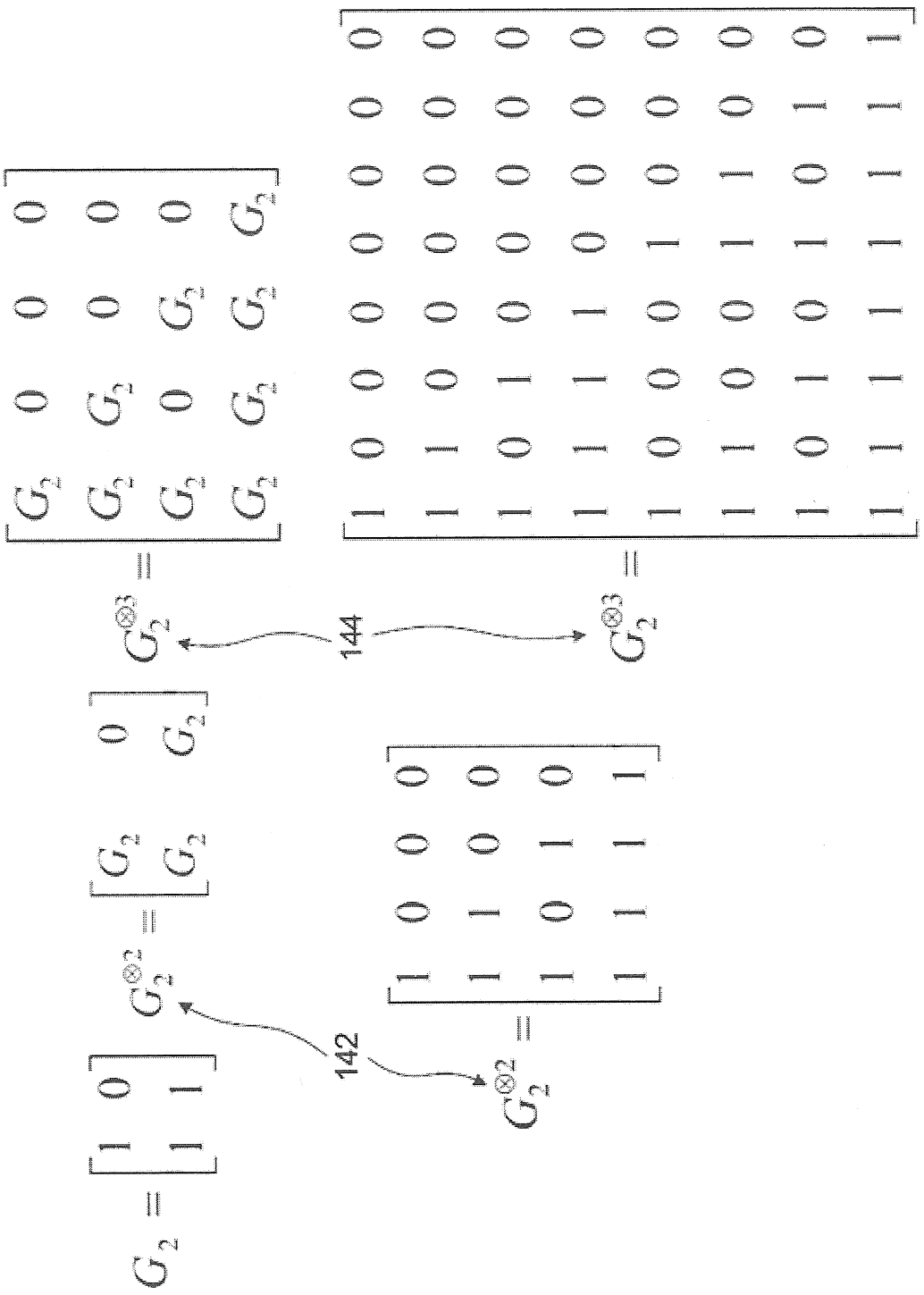


Fig.2

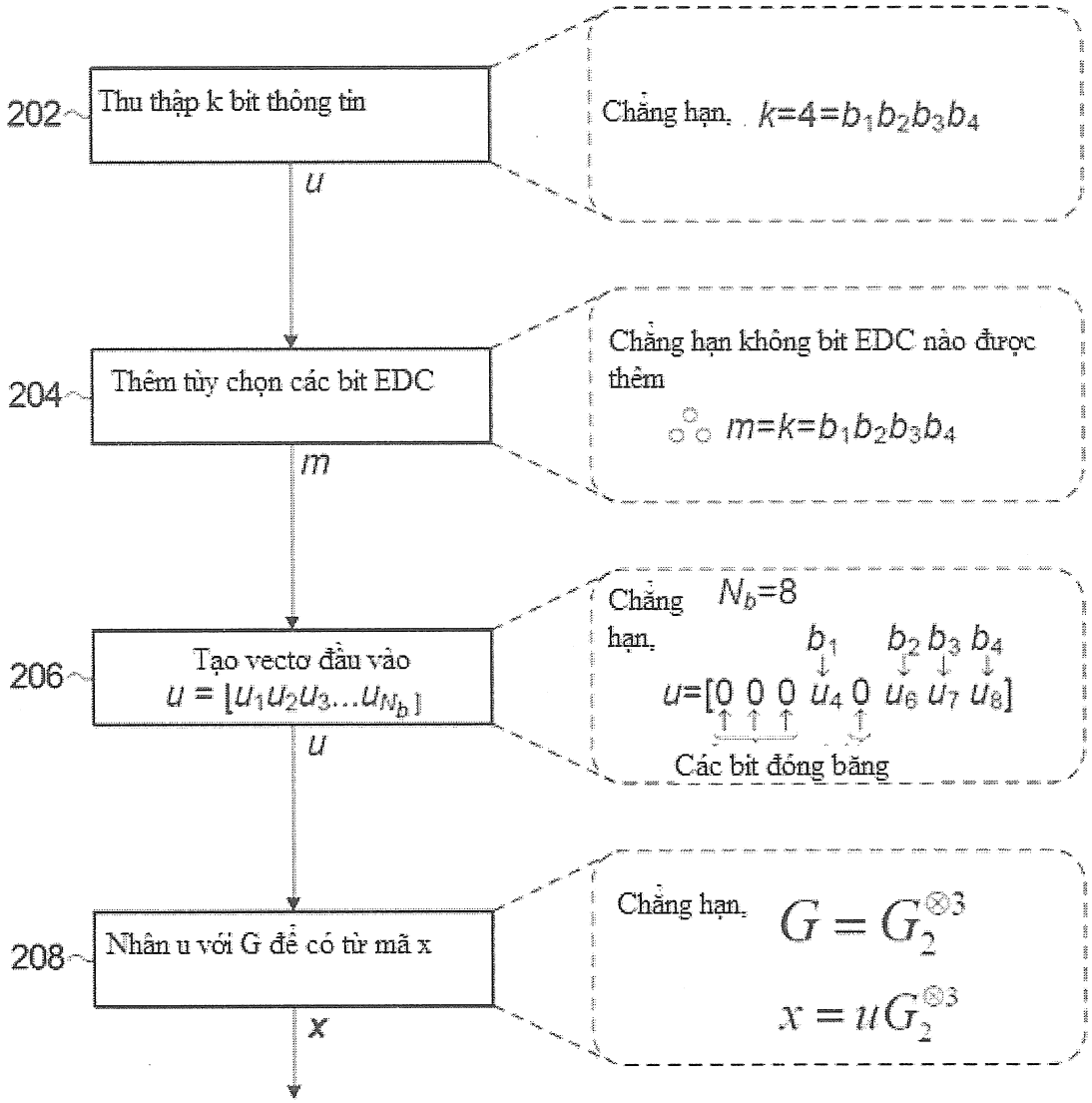
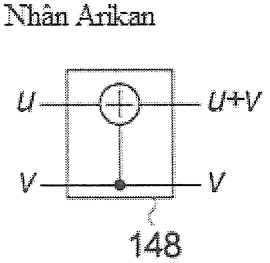

202
Thu thập k bit thông tin
Chẳng hạn, $k=4=b_1b_2b_3b_4$
u
204
Thêm tùy chọn các bit EDC
Chẳng hạn không bit EDC nào được thêm
$m=k=b_1b_2b_3b_4$
m
206
Tạo vectơ đầu vào
$u = [u_1u_2u_3...u_{N_b}]$
Chẳng hạn, $N_b=8$
$u=[0\ 0\ 0\ u_4\ 0\ u_6\ u_7\ u_8]$
Các bit đóng băng
u
208
Nhân u với G để có từ mã x
Chẳng hạn, $G = G_2^{\otimes 3}$
$x = uG_2^{\otimes 3}$
x

Fig.3

Fig.4


Nhân Arikan
u
u+v
v
v
148

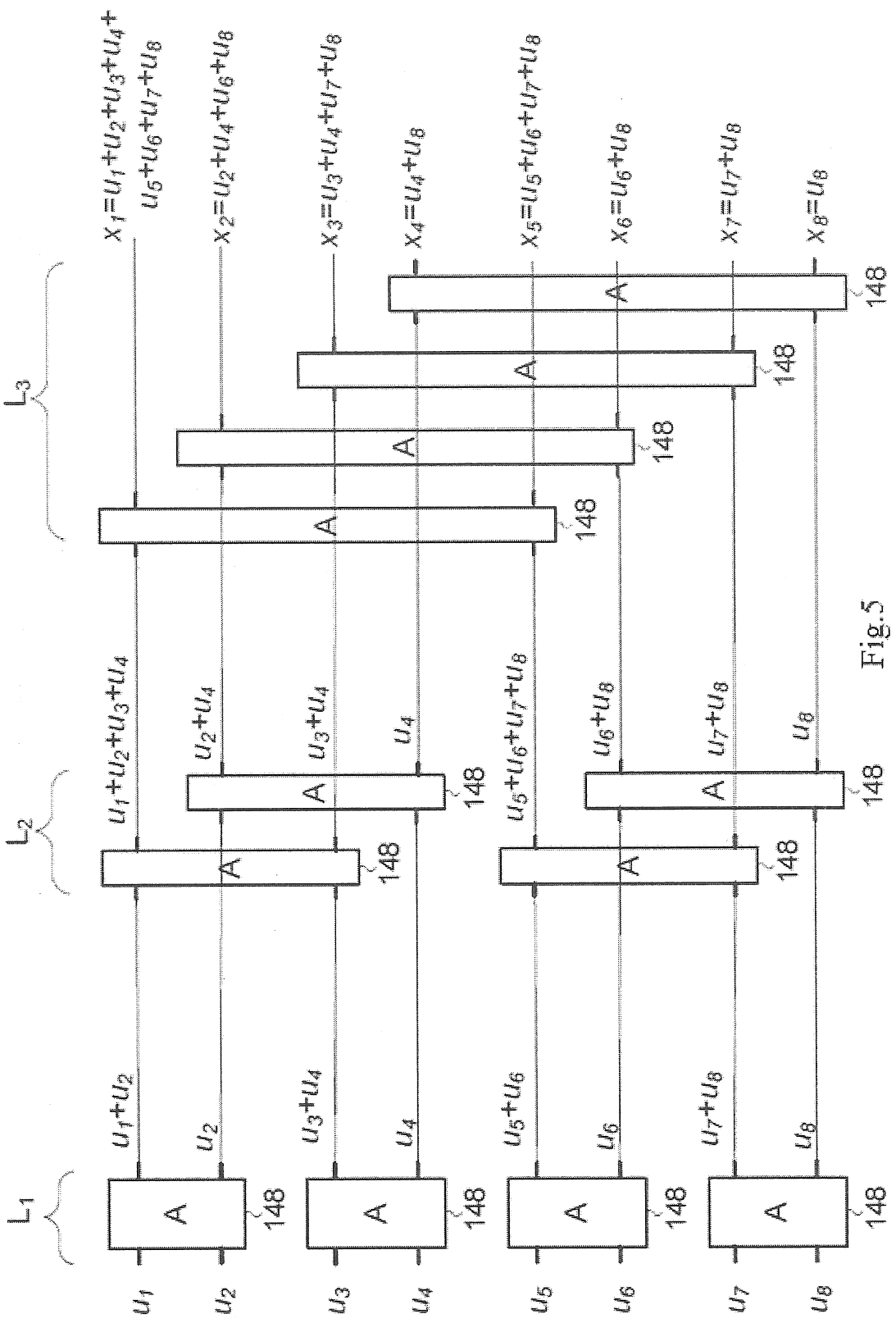

L1
L2
L3
A
148
u1
u2
u3
u4
u5
u6
u7
u8
u1+u2
u3+u4
u5+u6
u7+u8
u2+u4
u6+u8
u1+u2+u3+u4
u5+u6+u7+u8
x1=u1+u2+u3+u4+ u5+u6+u7+u8
x2=u2+u4+u6+u8
x3=u3+u4+u7+u8
x4=u4+u8
x5=u5+u6+u7+u8
x6=u6+u8
x7=u7+u8
x8=u8

Fig.5

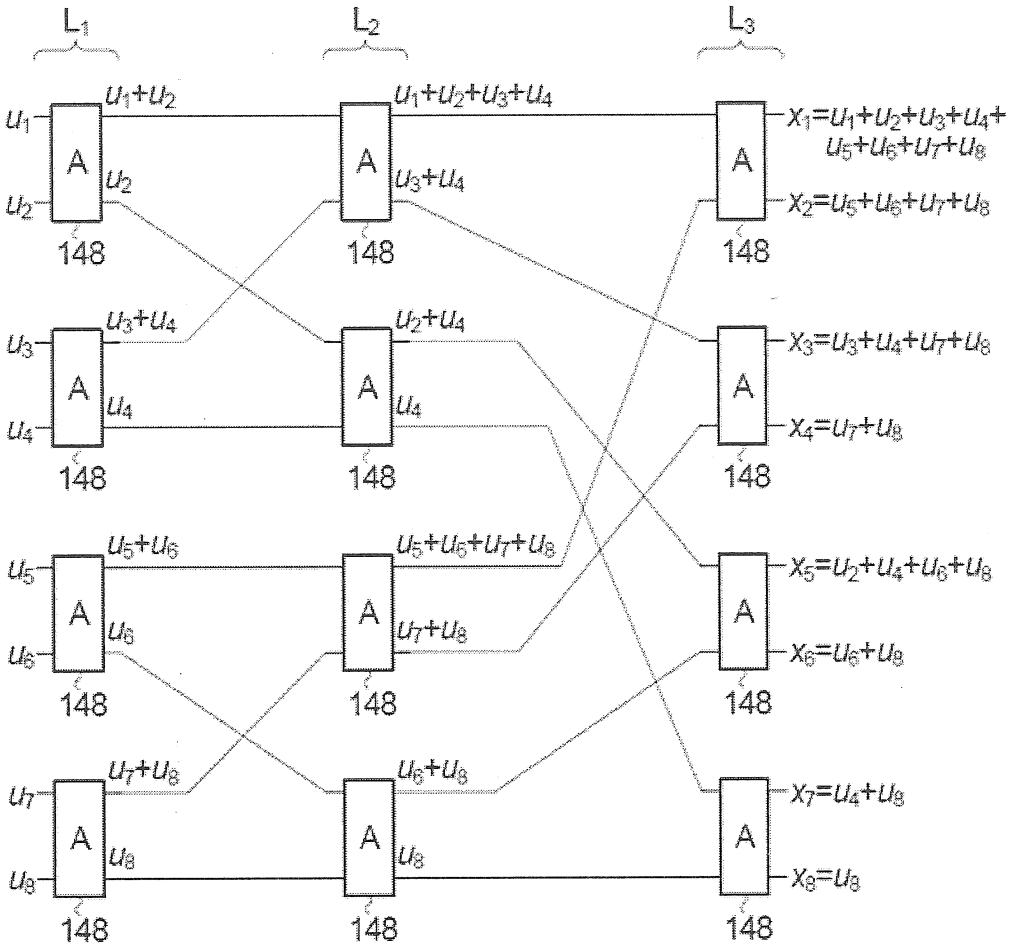

L1
L2
L3
u1
u2
u3
u4
u5
u6
u7
u8
A
148
u1+u2
u2
u1+u2+u3+u4
u3+u4
X1=u1+u2+u3+u4+
u5+u6+u7+u8
X2=u5+u6+u7+u8
u3+u4
u4
u2+u4
X3=u3+u4+u7+u8
X4=u7+u8
u5+u6
u6
u5+u6+u7+u8
u7+u8
X5=u2+u4+u6+u8
X6=u6+u8
u7+u8
u8
u6+u8
X7=u4+u8
X8=u8

Fig.6

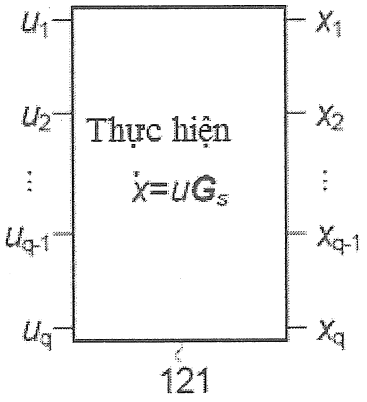
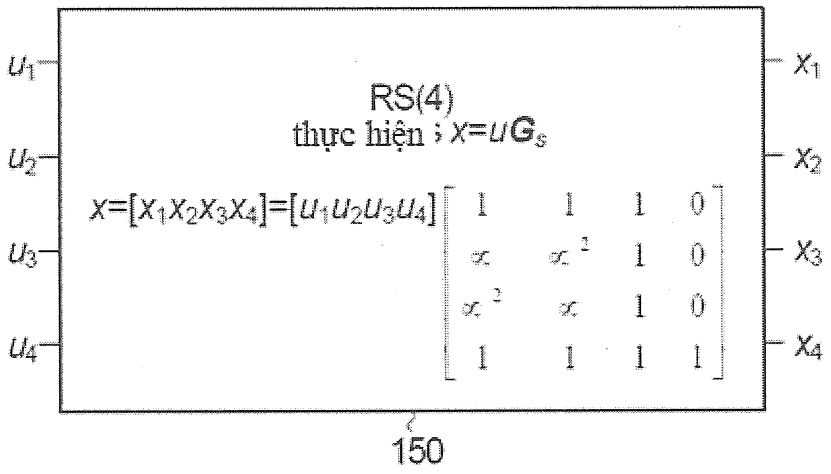

u_1
u_2
u_{q-1}
u_q
Thực hiện
$\dot{X}=uG_s$
x_1
x_2
x_{q-1}
x_q
121

Fig.7

Fig.8


u_1
u_2
u_3
u_4
RS(4)
thực hiện ; $X=uG_s$
$X=[X_1X_2X_3X_4]=[u_1u_2u_3u_4]$
x_1
x_2
x_3
x_4
150

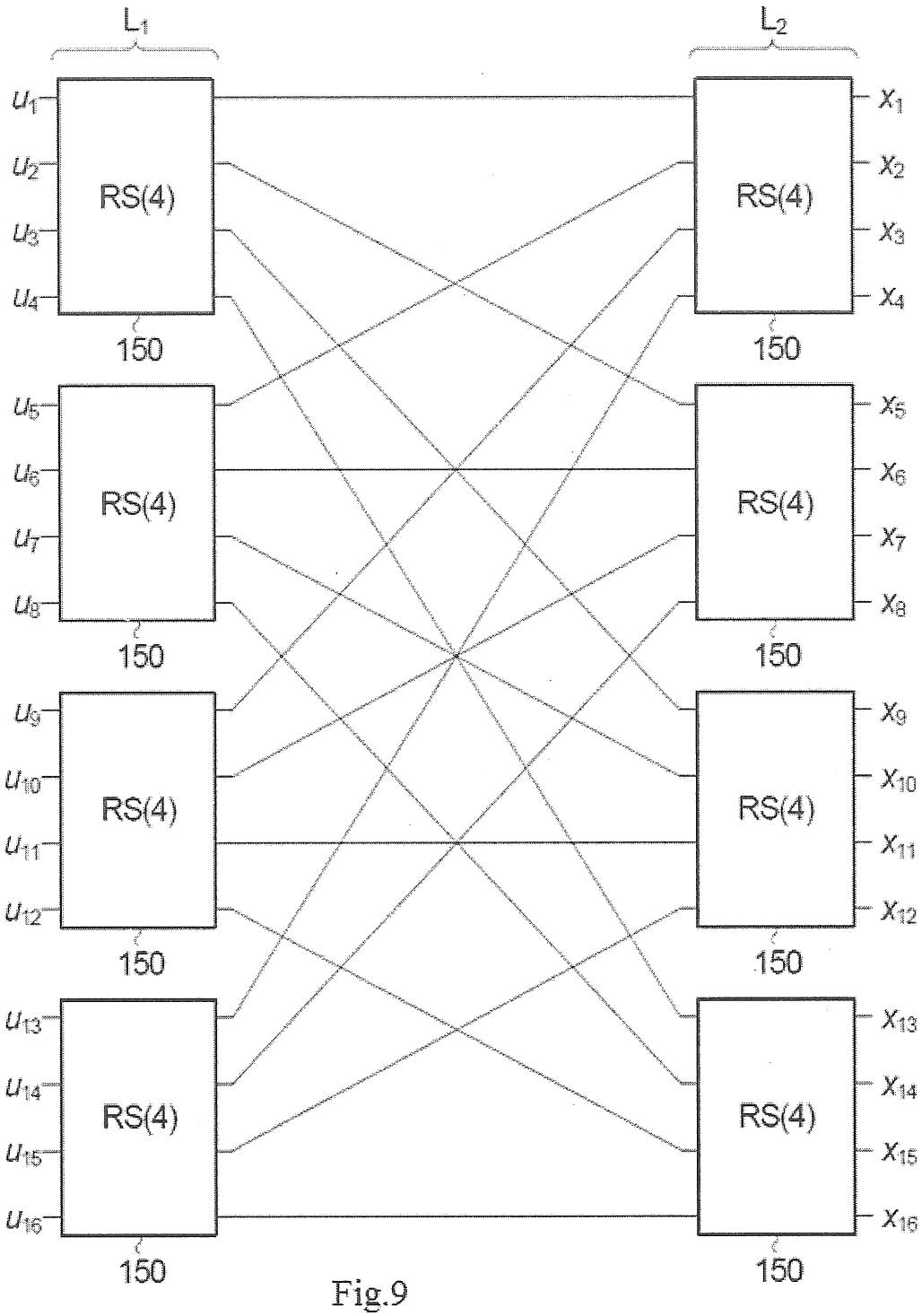
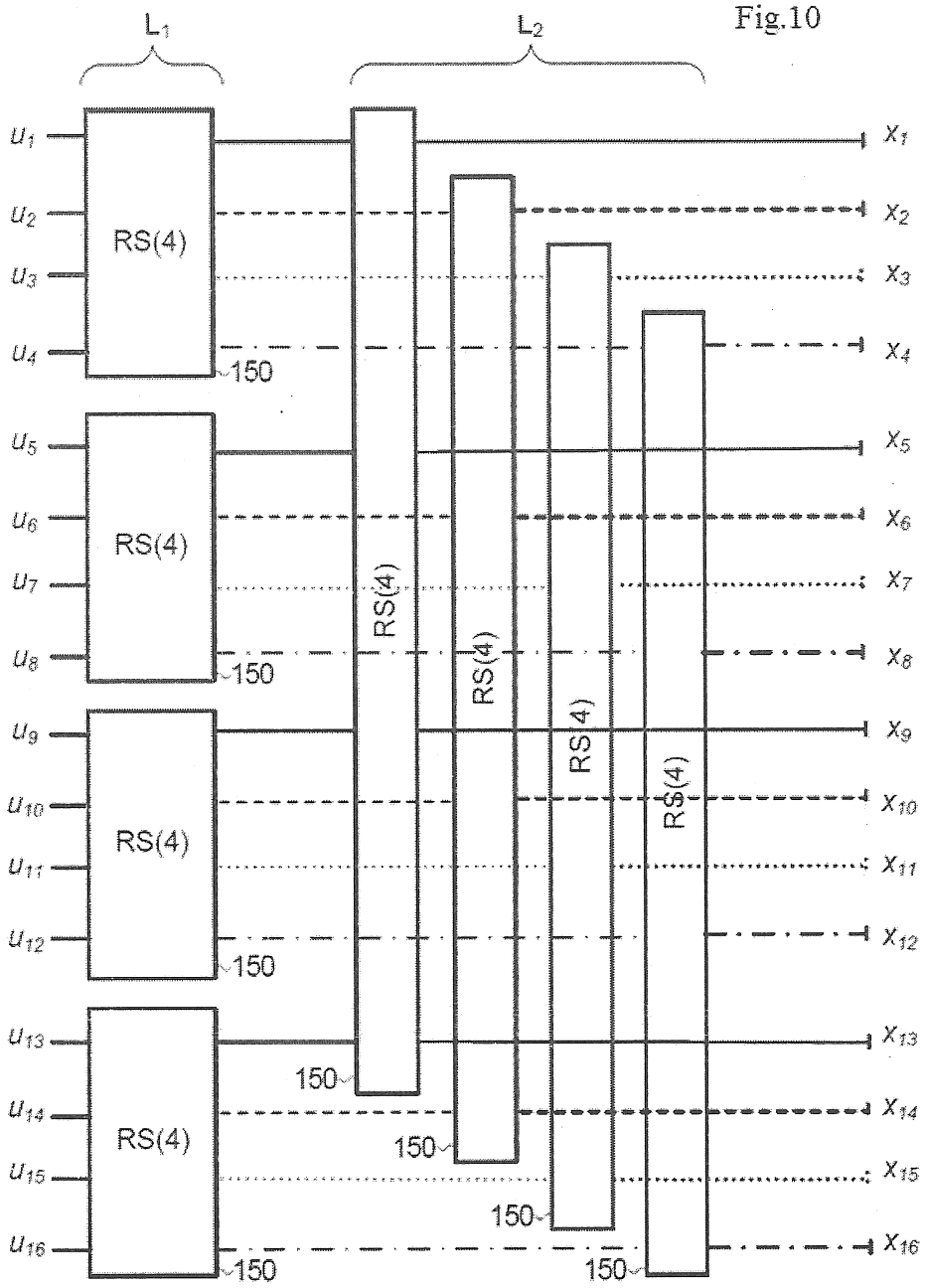

L1
L2
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
RS(4)
150
x1
x2
x3
x4
x5
x6
x7
x8
x9
x10
x11
x12
x13
x14
x15
x16

Fig.9

Fig.10


L1
L2
u1
u2
u3
u4
u5
u6
u7
u8
u9
u10
u11
u12
u13
u14
u15
u16
RS(4)
150
x1
x2
x3
x4
x5
x6
x7
x8
x9
x10
x11
x12
x13
x14
x15
x16

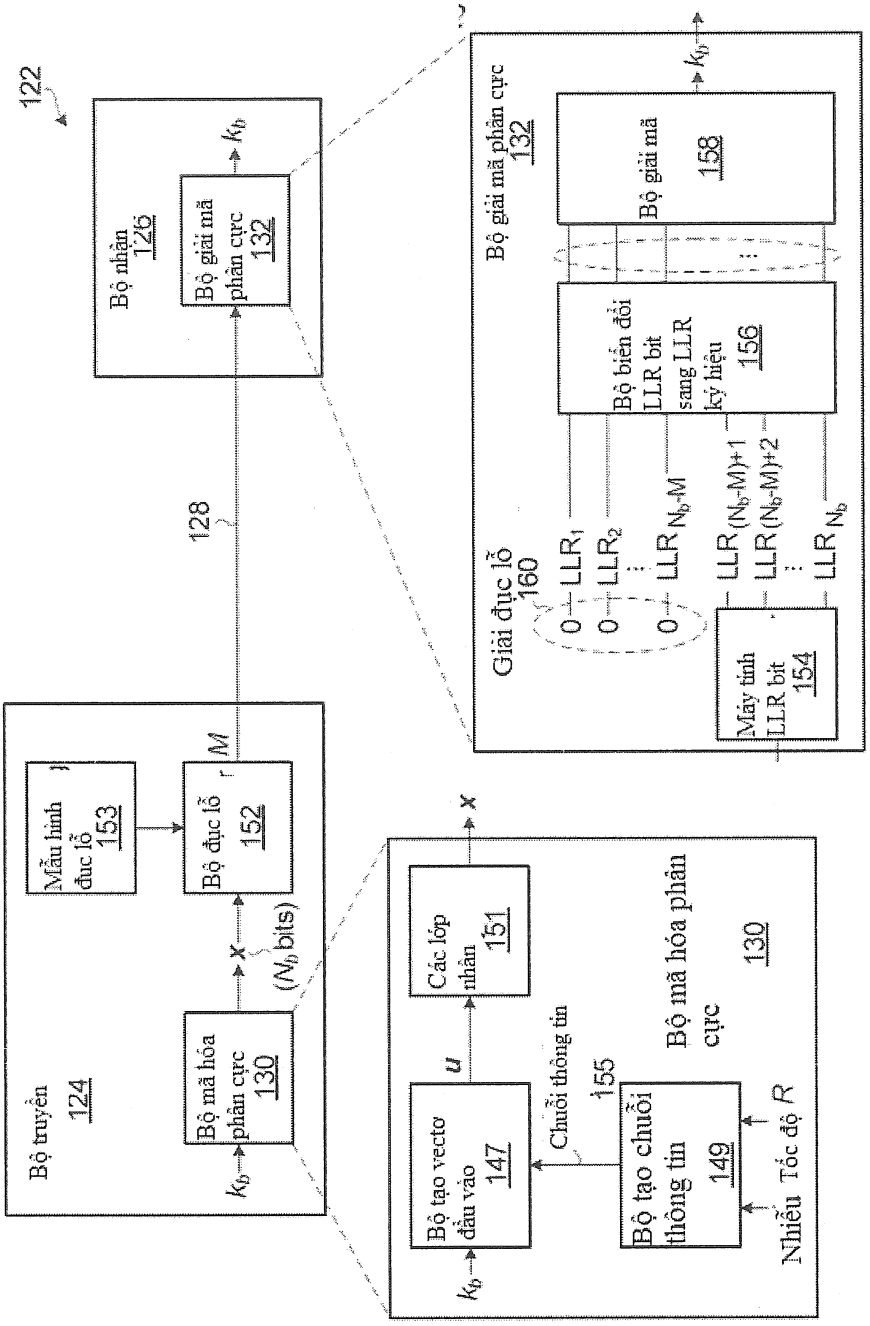

122
Bộ nhận
126
Bộ giải mã phân cực
132
k_b
128
Bộ truyền
124
Mẫu hình đục lỗ
153
Bộ đục lỗ
152
M
Bộ mã hóa phân cực
130
x
(N_b bits)
k_b
Bộ giải mã phân cực
132
Giải đục lỗ
160
LLR_1
LLR_2
LLR_{N_b-M}
$LLR_{(N_b-M)+1}$
$LLR_{(N_b-M)+2}$
LLR_{N_b}
Bộ biến đổi LLR bit sang LLR ký hiệu
156
Bộ giải mã
158
Máy tính LLR bit
154
Bộ mã hóa phân cực
130
Các lớp nhân
151
u
x
Bộ tạo vecto đầu vào
147
Chuỗi thông tin
155
Bộ tạo chuỗi thông tin
149
Nhiễu
Tốc độ R
k_b

Fig.11

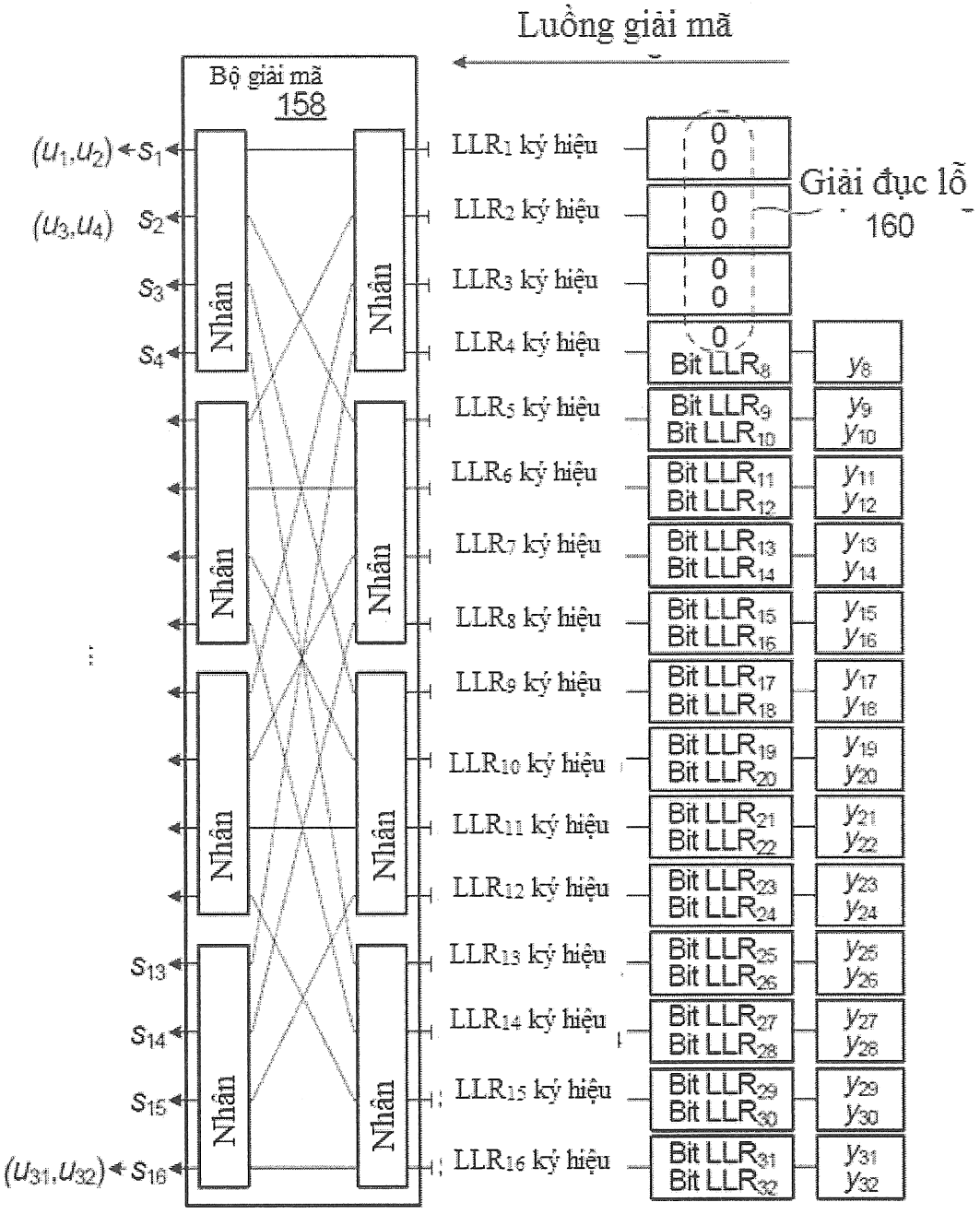

Luồng giải mã
Bộ giải mã
158
Giải đục lỗ
160
Nhân
(u_1,u_2) s_1
(u_3,u_4) s_2
s_3
s_4
s_{13}
s_{14}
s_{15}
(u_{31},u_{32}) s_{16}
LLR1 ký hiệu
LLR2 ký hiệu
LLR3 ký hiệu
LLR4 ký hiệu
LLR5 ký hiệu
LLR6 ký hiệu
LLR7 ký hiệu
LLR8 ký hiệu
LLR9 ký hiệu
LLR10 ký hiệu
LLR11 ký hiệu
LLR12 ký hiệu
LLR13 ký hiệu
LLR14 ký hiệu
LLR15 ký hiệu
LLR16 ký hiệu
0 0
0 0
0 0
0 Bit LLR8
Bit LLR9 Bit LLR10
Bit LLR11 Bit LLR12
Bit LLR13 Bit LLR14
Bit LLR15 Bit LLR16
Bit LLR17 Bit LLR18
Bit LLR19 Bit LLR20
Bit LLR21 Bit LLR22
Bit LLR23 Bit LLR24
Bit LLR25 Bit LLR26
Bit LLR27 Bit LLR28
Bit LLR29 Bit LLR30
Bit LLR31 Bit LLR32
y8
y9 y10
y11 y12
y13 y14
y15 y16
y17 y18
y19 y20
y21 y22
y23 y24
y25 y26
y27 y28
y29 y30
y31 y32

Fig.12

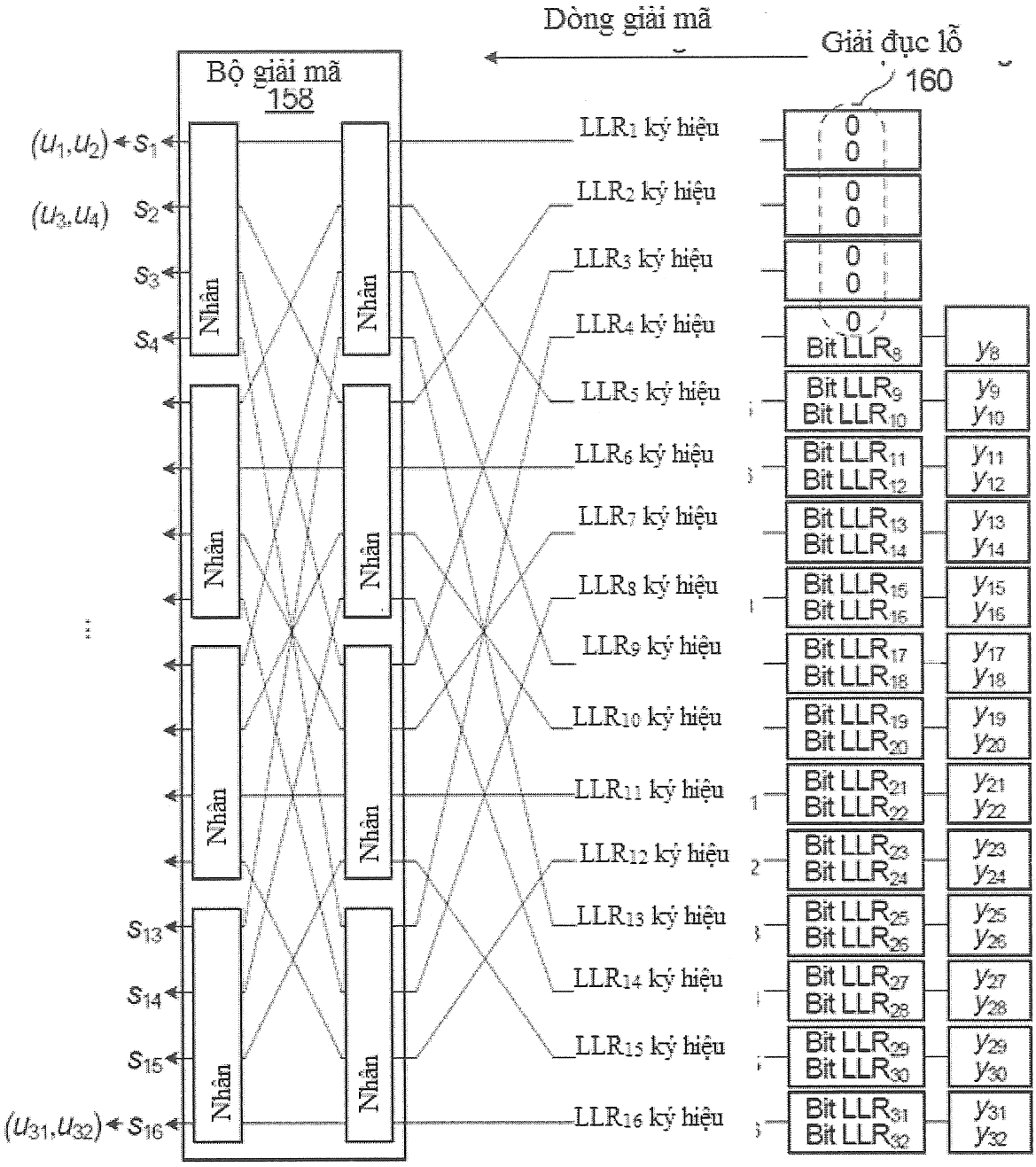
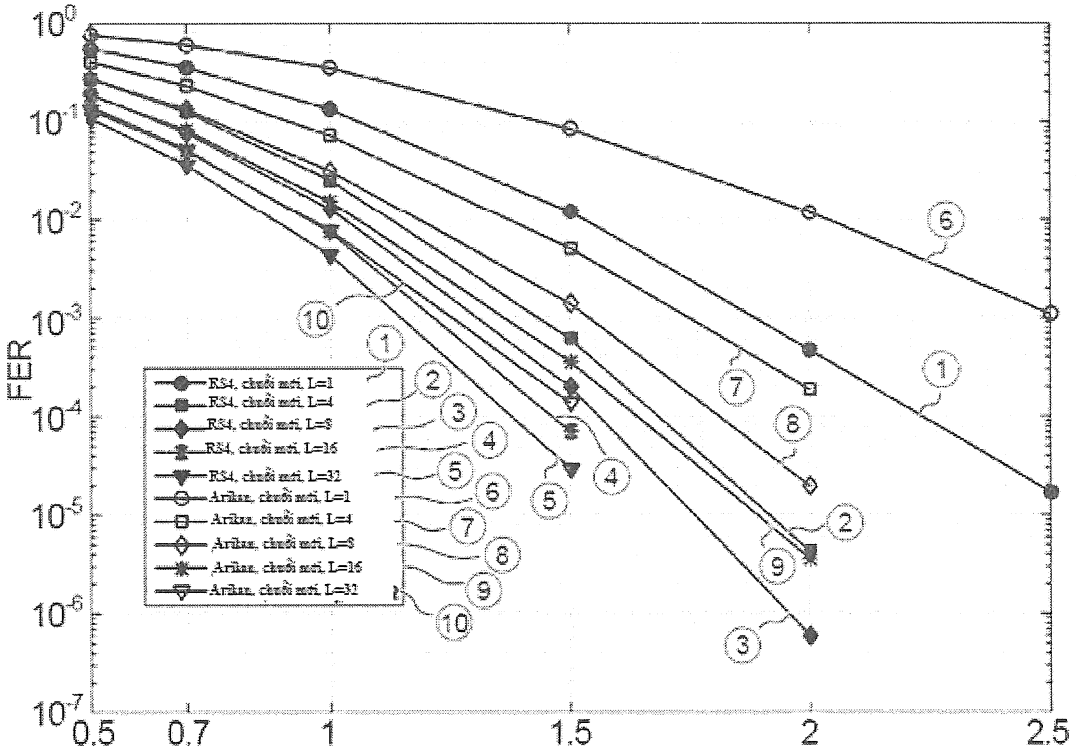

Dòng giải mã
Giải đục lỗ
160
Bộ giải mã
158
Nhân
(U_1,U_2)
(U_3,U_4)
S_1
S_2
S_3
S_4
S_{13}
S_{14}
S_{15}
(U_{31},U_{32})
S_{16}
LLR_1 ký hiệu
LLR_2 ký hiệu
LLR_3 ký hiệu
LLR_4 ký hiệu
LLR_5 ký hiệu
LLR_6 ký hiệu
LLR_7 ký hiệu
LLR_8 ký hiệu
LLR_9 ký hiệu
LLR_{10} ký hiệu
LLR_{11} ký hiệu
LLR_{12} ký hiệu
LLR_{13} ký hiệu
LLR_{14} ký hiệu
LLR_{15} ký hiệu
LLR_{16} ký hiệu
0 0
Bit LLR_8
Bit LLR_9 Bit LLR_{10}
Bit LLR_{11} Bit LLR_{12}
Bit LLR_{13} Bit LLR_{14}
Bit LLR_{15} Bit LLR_{16}
Bit LLR_{17} Bit LLR_{18}
Bit LLR_{19} Bit LLR_{20}
Bit LLR_{21} Bit LLR_{22}
Bit LLR_{23} Bit LLR_{24}
Bit LLR_{25} Bit LLR_{26}
Bit LLR_{27} Bit LLR_{28}
Bit LLR_{29} Bit LLR_{30}
Bit LLR_{31} Bit LLR_{32}
y_8
y_9 y_{10}
y_{11} y_{12}
y_{13} y_{14}
y_{15} y_{16}
y_{17} y_{18}
y_{19} y_{20}
y_{21} y_{22}
y_{23} y_{24}
y_{25} y_{26}
y_{27} y_{28}
y_{29} y_{30}
y_{31} y_{32}

Fig.13



CRC8 cho RS4, CRC16 cho Arikan, CRC = 0 cho L = 1, trong cả hai trường hợp mã hóa

Fig.14

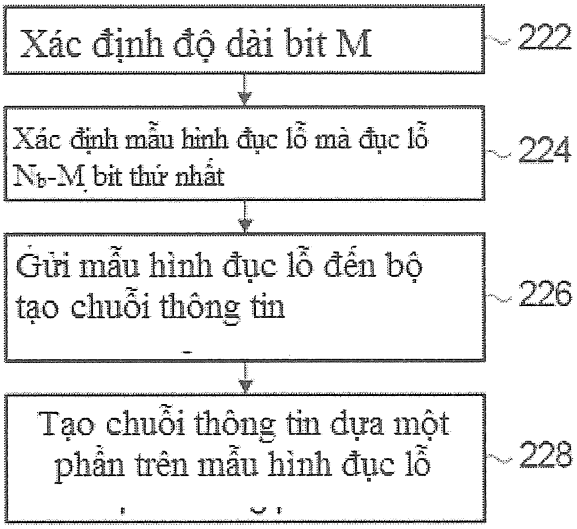

Xác định độ dài bit M
222
Xác định mẫu hình đục lỗ mà đục lỗ N_b-M bit thứ nhất
224
Gửi mẫu hình đục lỗ đến bộ tạo chuỗi thông tin
226
Tạo chuỗi thông tin dựa một phần trên mẫu hình đục lỗ
228

Fig.15

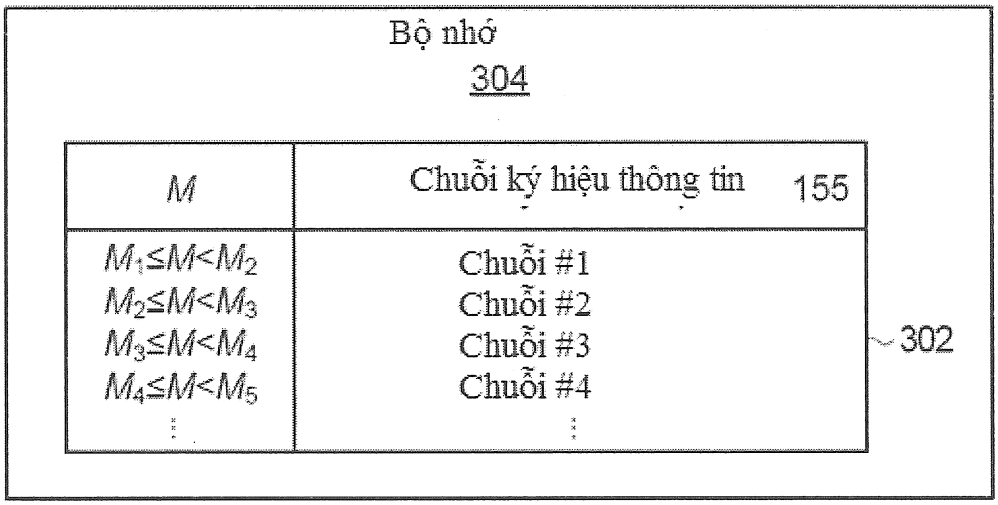

Bộ nhớ
304
M
Chuỗi ký hiệu thông tin
155
$M_1 \le M < M_2$
$M_2 \le M < M_3$
$M_3 \le M < M_4$
$M_4 \le M < M_5$
Chuỗi #1
Chuỗi #2
Chuỗi #3
Chuỗi #4
302

Fig.16

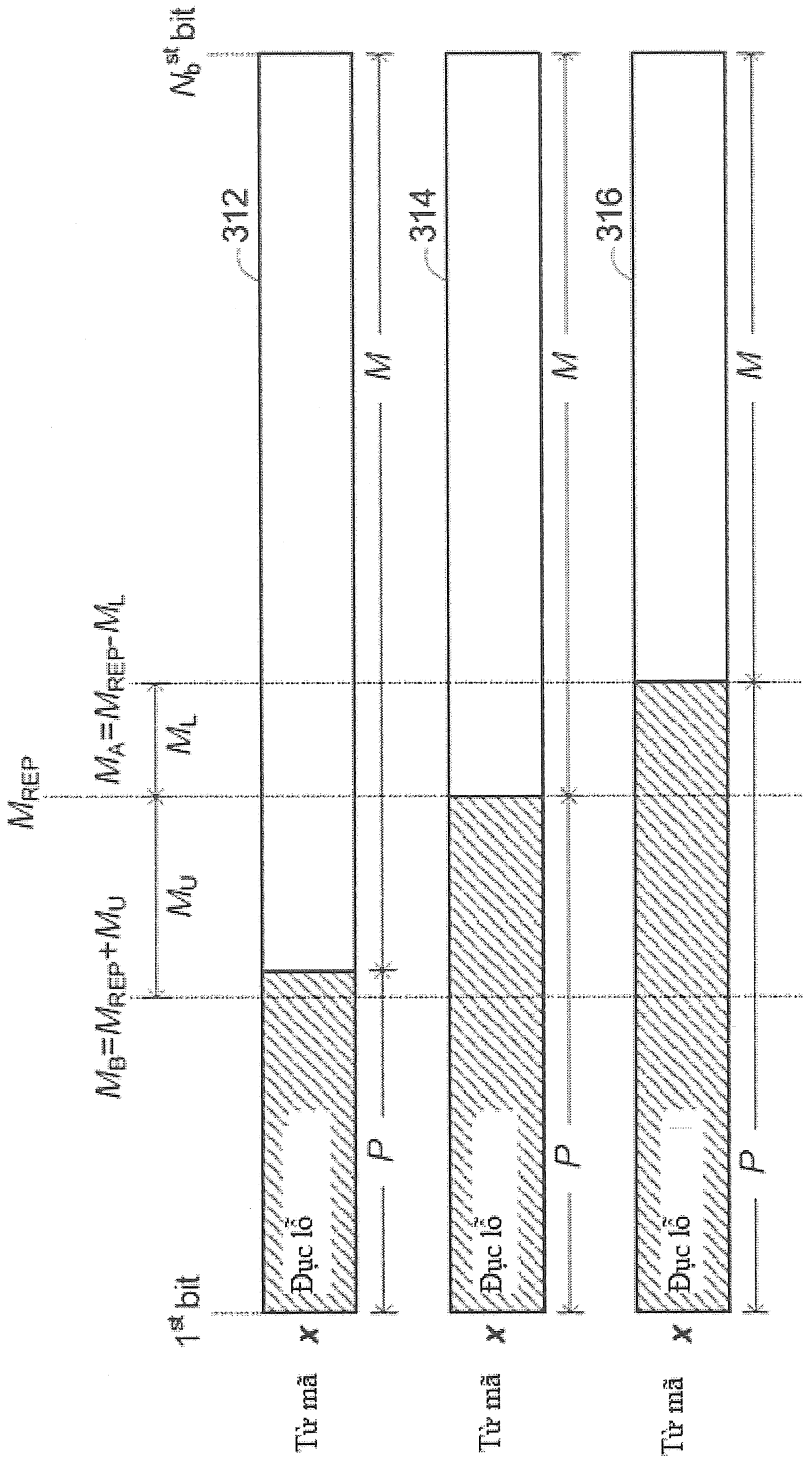

N_b^{st} bit
312
314
316
M_{REP}
$M_A=M_{REP}-M_L$
M_L
$M_B=M_{REP}+M_U$
M_U
M
P
1st bit
Đục lỗ
x
Từ mã

Fig.17

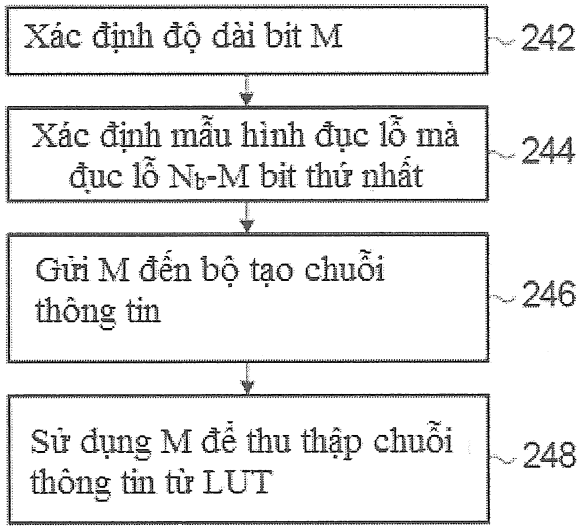
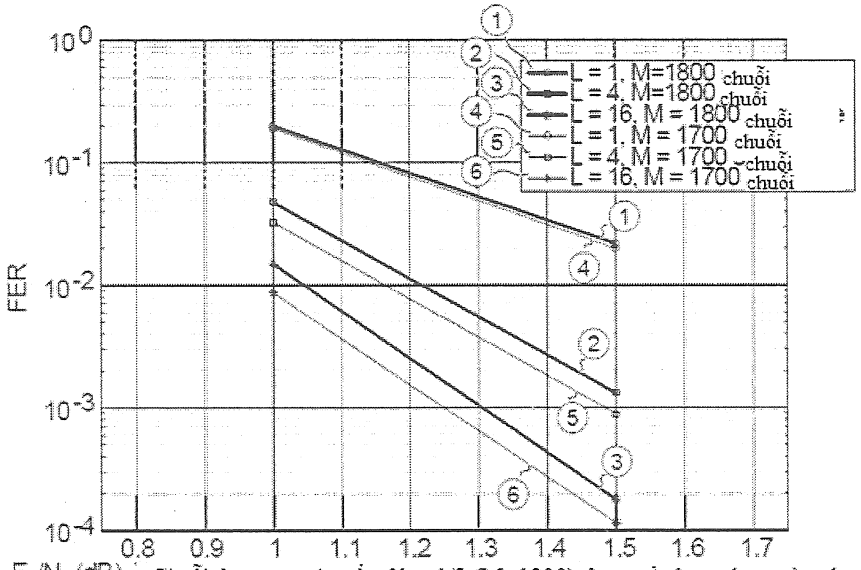
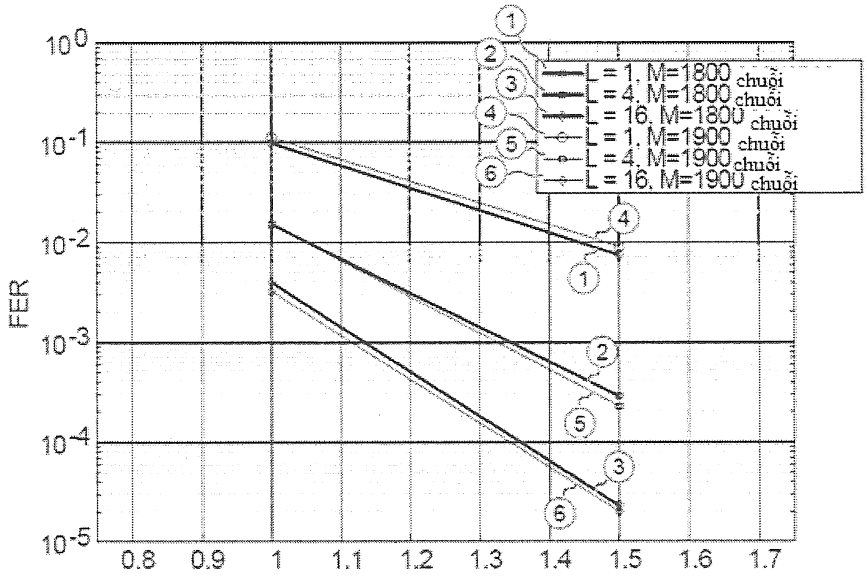

Xác định độ dài bit M
242
Xác định mẫu hình đục lỗ mà đục lỗ N_b-M bit thứ nhất
244
Gửi M đến bộ tạo chuỗi thông tin
246
Sử dụng M để thu thập chuỗi thông tin từ LUT
248

Fig.18

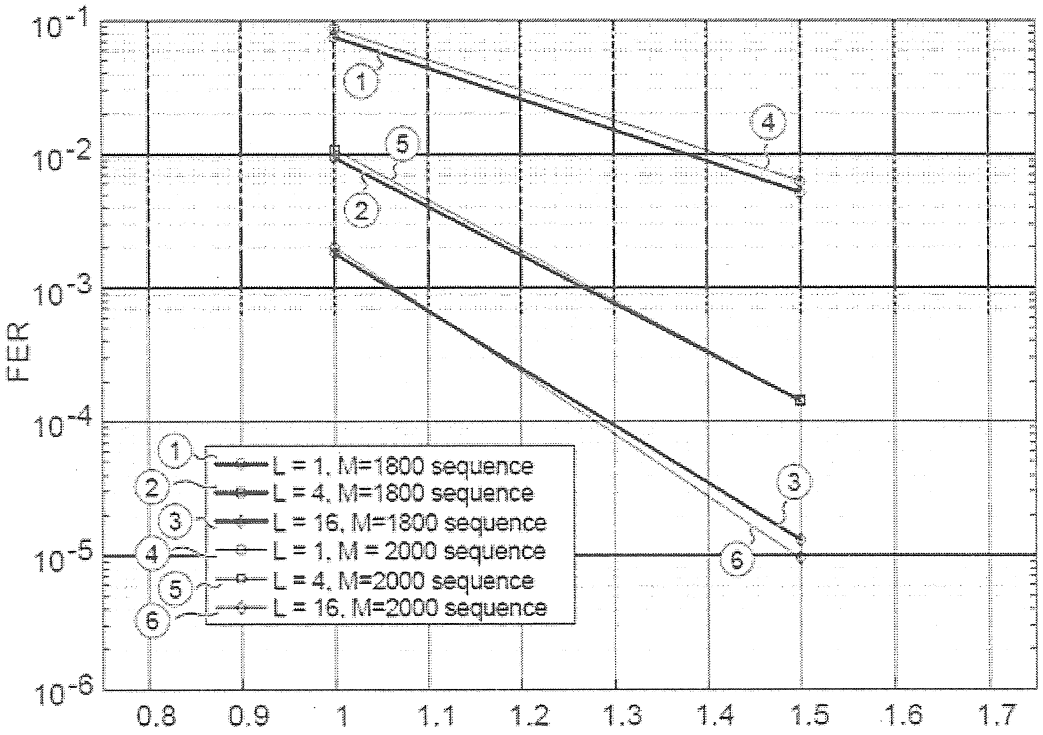


Chuỗi được tạo cho tốc độ = 1/3 (M=1800) được sử dụng cho trường hợp tốc độ = 0,353 (M=1700)



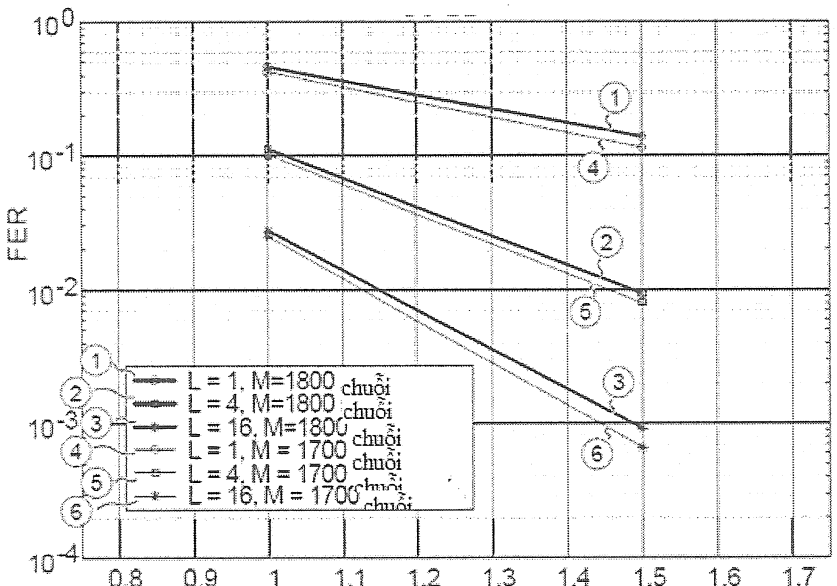
Chuỗi được tạo cho tốc độ = 1/3 (M=1800) được sử dụng cho trường hợp tốc độ = 0,316 (M=1900)

Fig.19

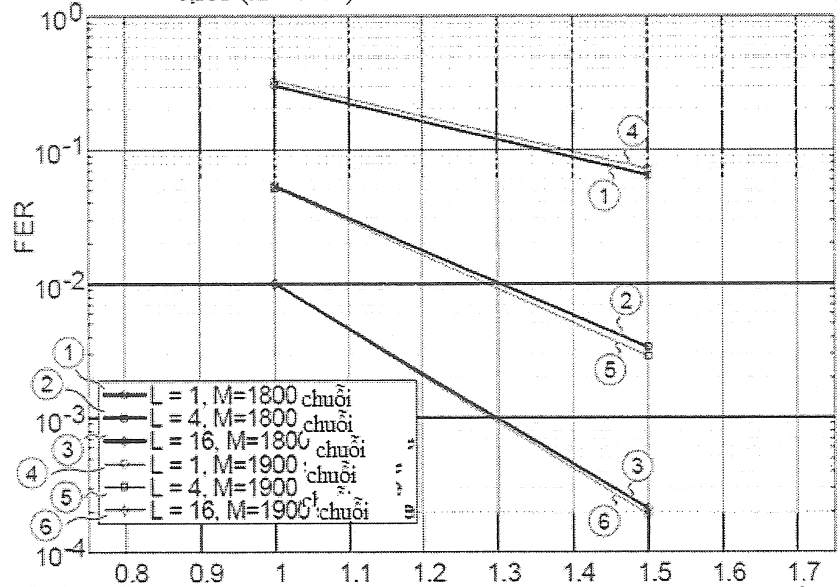


E_b/N_0 (dB). Tốc độ tạo chuỗi = 1/3 (M=1800) được sử dụng cho trường hợp tốc độ = 0,3 (M = 2000)

Fig.20

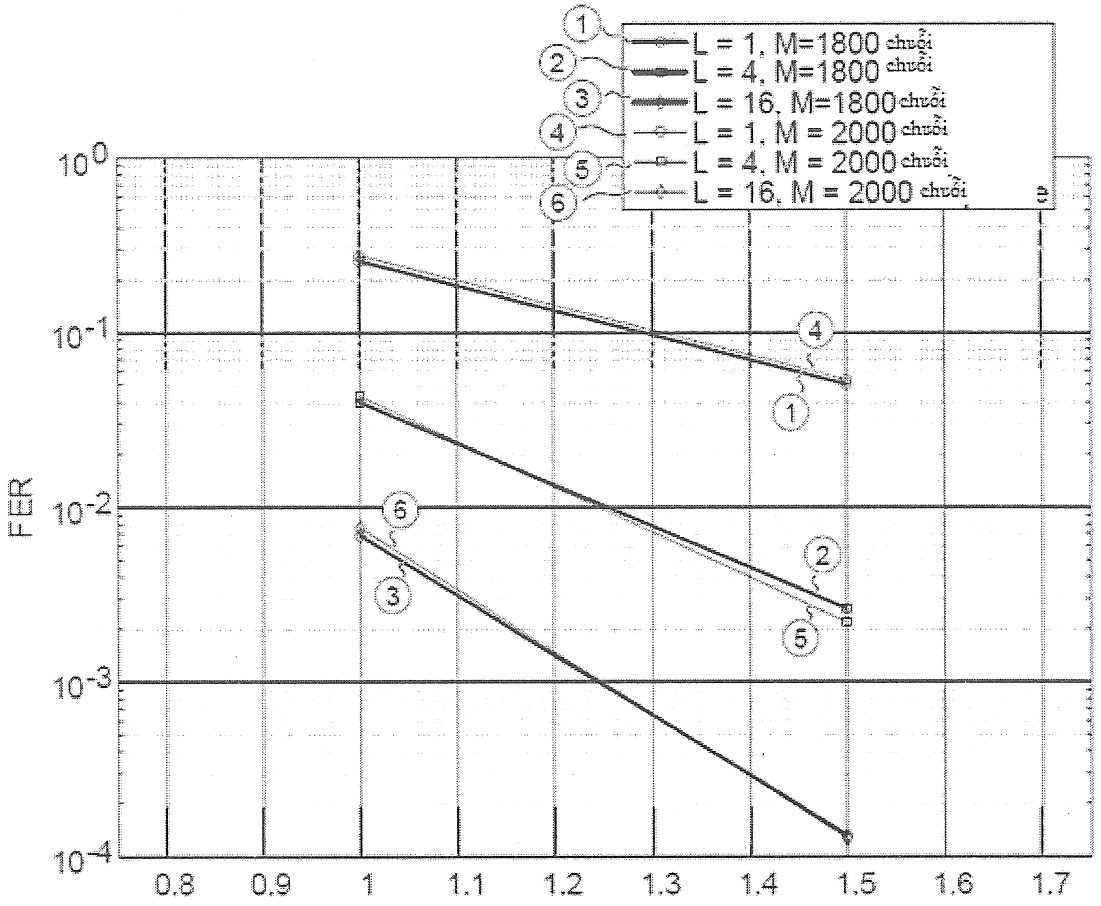


E_b/N_0 (dB). Tốc độ tạo chuỗi = 1/3 (M= 1800) được sử dụng cho trường hợp tốc độ = 0,353 (M = 1700)



E_b/N_0 (dB). Tốc độ tạo chuỗi = 1/3 (M= 1800) được sử dụng cho trường hợp tốc độ = 0,316 (M = 1900)

Fig.21



Chuỗi được tạo cho tốc độ = 1/3 (M = 1800) được sử dụng cho trường hợp tốc độ = 0,3 (M = 2000)

Fig.22

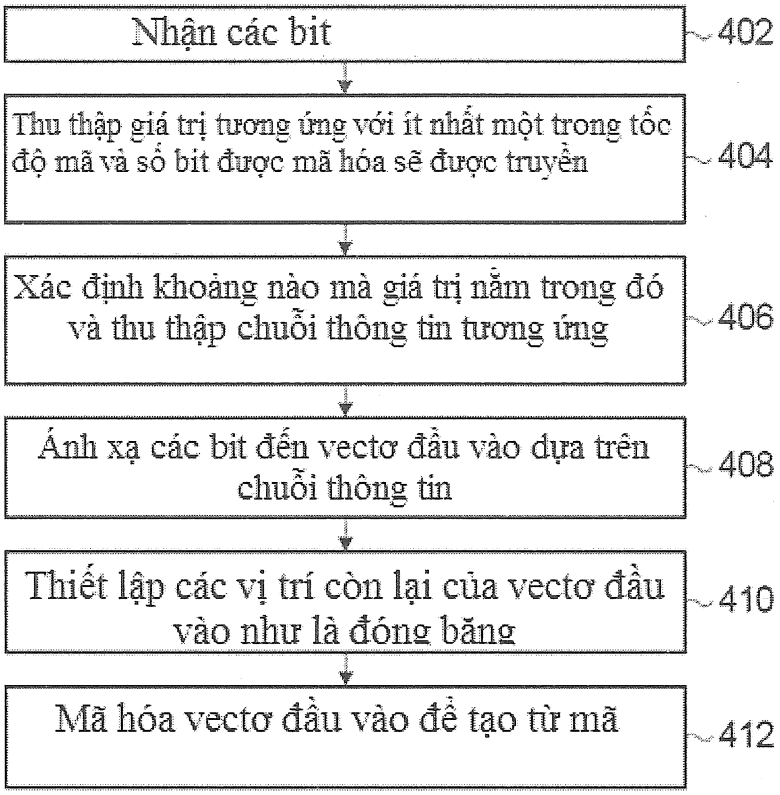

Nhận các bit
402
Thu thập giá trị tương ứng với ít nhất một trong tốc độ mã và số bit được mã hóa sẽ được truyền
404
Xác định khoảng nào mà giá trị nằm trong đó và thu thập chuỗi thông tin tương ứng
406
Ánh xạ các bit đến vectơ đầu vào dựa trên chuỗi thông tin
408
Thiết lập các vị trí còn lại của vectơ đầu vào như là đóng băng
410
Mã hóa vectơ đầu vào để tạo từ mã
412

Fig.23

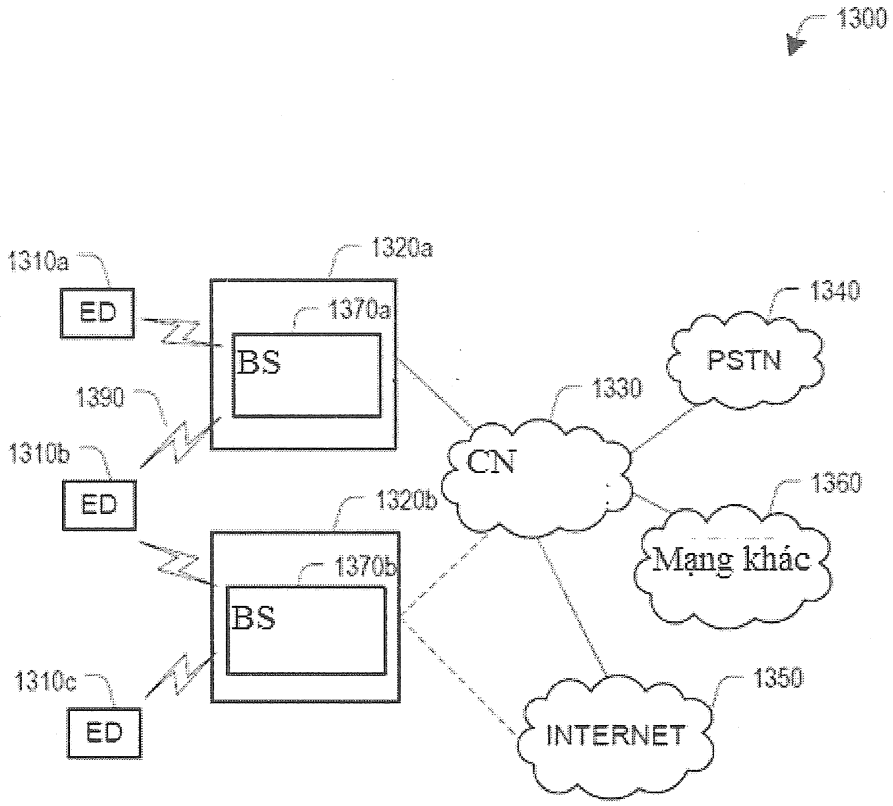

1300
1310a
ED
1320a
1370a
BS
1340
PSTN
1330
1390
1310b
ED
CN
1360
Mạng khác
1320b
1370b
BS
1310c
ED
1350
INTERNET

Fig.24

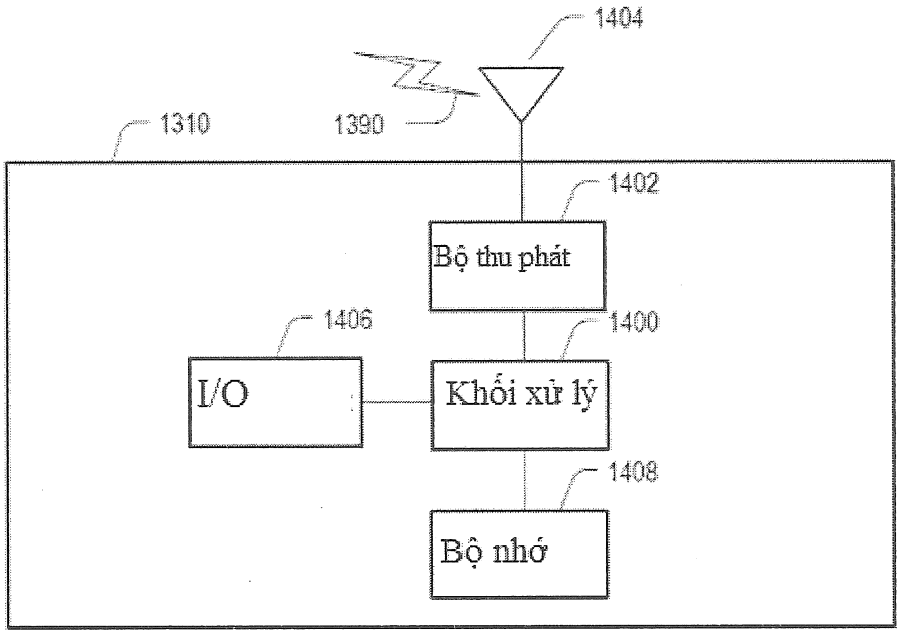

1404
1390
1310
1402
Bộ thu phát
1406
1400
I/O
Khối xử lý
1408
Bộ nhớ

Fig.25

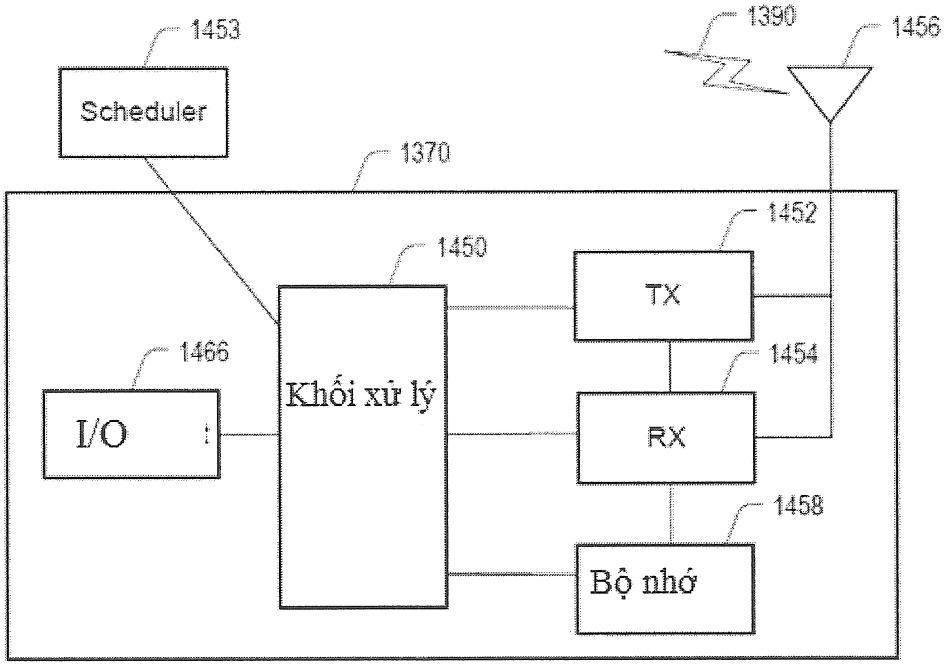

1453
Scheduler
1390
1456
1370
1452
1450
TX
1466
Khối xử lý
1454
I/O
RX
1458
Bộ nhớ

Fig.26