



(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)



1-0048688

(51)^{2020.01} H04L 1/00

(13) B

(21) 1-2020-06182

(22) 27/10/2020

(45) 25/07/2025 448

(43) 25/02/2021 395A

(73) TẬP ĐOÀN CÔNG NGHIỆP - VIỆN THÔNG QUÂN ĐỘI (VN)

Lô D26 Khu đô thị mới Cầu Giấy, phường Yên Hoà, quận Cầu Giấy, thành phố Hà Nội

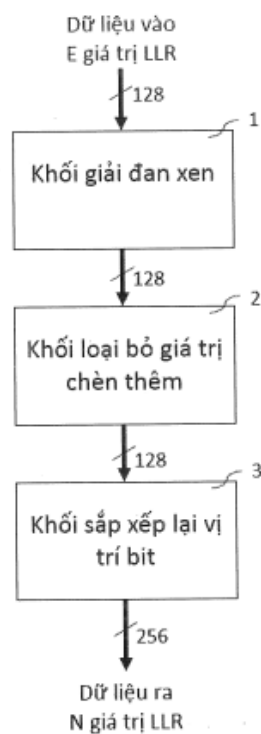
(72) Vương Đăng Huy (VN).

(74) Công ty TNHH Tư vấn Quốc Dân (NACILAW)

(54) KHÔI GIẢI ĐIỀU CHỈNH TỶ LỆ BÍT KÊNH DỮ LIỆU VẬT LÝ DÙNG CHUNG ĐƯỜNG LÊN TRONG HỆ THỐNG 5G

(21) 1-2020-06182

(57) Sáng chế đề xuất khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên trong hệ thống 5G bao gồm ba khối chức năng chính: giải đan xen, loại bỏ giá trị chèn thêm và sắp xếp lại vị trí bit. Sáng chế có thể đáp ứng được toàn bộ các giá trị HARQ từ không đến ba theo như chuẩn 5G yêu cầu, có thể dễ dàng thực hiện đường ống hóa, song song hóa để tối ưu tốc độ xử lý, tốc độ xử lý cao, có thể sử dụng cùng với phần mềm để giảm tải, tài nguyên cho phần mềm.



Hình 3

Lĩnh vực kỹ thuật được đề cập

Sáng chế đề xuất khỏi giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên trong hệ thống 5G. Cụ thể, khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên trong hệ thống 5G được ứng dụng trong việc giải mã dữ liệu người dùng kênh vật lý dùng chung trong hệ thống thông tin vô tuyến thế hệ năm.

Tình trạng kỹ thuật của sáng chế

Trong hệ thống thông tin vô tuyến nói chung và trong hệ thống 5G nói riêng, dữ liệu của người dùng trước khi được thực hiện điều chế và gửi qua kênh truyền sẽ được thực hiện mã hóa kênh để giảm bớt đi các tác động xấu của kênh truyền bằng cách thêm vào dữ liệu các bit dư thừa nhằm hai mục đích: thứ nhất, giúp phía thu kiểm tra xem dữ liệu nhận được có đúng hay không, thứ hai phía thu dùng để tự thực hiện sửa lỗi dữ liệu nhận được. Trong kênh dữ liệu vật lý dùng chung đường lên, loại mã hóa kênh được sử dụng là mã hạng nhẹ (Low Density Parity-check Code) hay gọi tắt là mã LDPC. Tuy nhiên, việc thêm lượng bit dư thừa có thể làm cho số lượng bit cần truyền đi bị lệch so với lượng tài nguyên mà người dùng được cấp phát. Chính vì vậy sau khi thực hiện mã hóa LDPC cần có một bước thực hiện điều chỉnh tỷ lệ bit (rate matching) để sao cho lượng bit cần truyền khớp với số tài nguyên hiện có. Nếu số bit quá ít thì cần lặp lại dữ liệu một vài lần, còn nếu số bit lớn hơn thì cần thực hiện cắt bớt.

Về lý thuyết, việc điều chỉnh tỷ lệ bit bao gồm hai bước: bước một là lựa chọn bit, bước hai là đan xen các bit được chọn. Dữ liệu sau khi mã hóa LDPC sẽ có độ dài N bit bao gồm: dữ liệu gốc trước khi được mã hóa, phần bit có giá trị không xác định và phần dữ liệu dư thừa được thêm vào. E bit liên tiếp được ký hiệu $e_0, e_1, e_2, \dots, e_{E-1}$ sẽ được lựa chọn từ dữ liệu N bit, ngoại trừ các bit có giá trị không xác định gọi là bit NULL như được thể hiện trên Hình 1. Vị trí lấy ra bit đầu tiên sẽ được quyết định bởi giá trị yêu cầu lặp lại tự động kết hợp (Hybird Auto Repeat reQuest) HARQ. Việc thực hiện đan xen các bit được mô tả trên Hình 2. Đầu tiên E bit được chọn ở bước lựa chọn bit sẽ được sắp thành một ma trận Q_m hàng và E/Q_m cột trong đó Q_m là mức điều chế. E bit này sẽ được sắp vào liên tiếp theo

thứ tự từng hàng một, sau đó sẽ được đọc ra lần lượt theo từng cột theo chiều từ trên xuống dưới như của mũi tên như trong hình. Ví dụ trong trường hợp mức điều chế $Q_m = 2$ tức điều chế pha trực giao (Quadrature Phase Shift Keying - QPSK) thì đầu ra sau bước xen kẽ dữ liệu sẽ là $e_0, e_{\frac{E}{2}}, e_1, e_{\frac{E}{2}+1}, \dots, e_{\frac{E}{2}-1}, e_{E-1}$.

Bên phía thu, dữ liệu trước khi được vào giải mã LDPC cần phải được thực hiện giải điều chế và giải điều chỉnh tỷ lệ bit. Có hai phương pháp chính thực hiện giải điều chế: một là giải điều chế cứng và thứ hai là giải điều chế mềm. Trên thực tế phương pháp giải điều chế mềm được sử dụng phổ biến hơn bởi nó cho kết quả tốt hơn. Đầu ra của khối giải điều chế mềm là giá trị ước lượng hợp lý (Log-likelihood hay viết tắt là LLR) biểu diễn xác suất một bit nhận giá trị 0 hay 1. Do đó đầu vào của khối giải điều chỉnh tỷ lệ bit là E giá trị LLR và đầu ra của nó sẽ là N giá trị LLR.

Hiện tại các phương pháp thực hiện giải điều chỉnh tỷ lệ bit đều được thực hiện bằng phần mềm trên ngôn ngữ C/C++. Hạn chế của phương pháp này đó là sẽ tốn tài nguyên cũng như thời gian xử lý của máy tính.

Bản chất kỹ thuật của sáng chế

Mục đích của sáng chế là đề xuất khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên của hệ thống 5G. Để đạt được mục đích trên hệ thống được đề xuất bao gồm các khối chức năng:

·Khối giải đan xen: chức năng của khối này là thực hiện giải đan xen dữ liệu đầu vào.

·Khối loại bỏ giá trị chèn thêm: chức năng của khối này là thực hiện loại bỏ đi những bit chèn thêm dư thừa trong dữ liệu.

·Khối sắp xếp lại vị trí bit: chức năng của khối này là thực hiện sắp xếp lại các bit theo đúng thứ tự như bên phía phát.

Mô tả vắn tắt các hình vẽ

Hình 1 mô tả việc lựa chọn E bit từ N bit dữ liệu sau mã hóa LDPC,

Hình 2 mô tả cách thực hiện xen kẽ bit,

Hình 3 mô tả kiến trúc đề xuất của khối giải điều chỉnh tỷ lệ bit,

Hình 4 mô tả cấu trúc dữ liệu vào/ra của khối giải điều chế tỷ lệ bit,

Hình 5 mô tả một dữ liệu đầu vào của hệ thống trong trường hợp $Q_m = 2$,

Hình 6 mô tả kiến trúc cụ thể của khối giải đan xen,

Hình 7 mô tả cách sắp xếp lại dữ liệu của khối giải đan xen trong trường hợp $Q_m = 2$,

Hình 8 mô tả thứ tự của dữ liệu đầu ra khối giải đan xen,

Hình 9 mô tả kiến trúc cụ thể của khối loại bỏ giá trị chèn thêm,

Hình 10 mô tả hoạt động ghi đọc dữ liệu vào bộ nhớ của khối loại bỏ các giá trị chèn thêm,

Hình 11 mô tả cách thức loại bỏ phần các giá trị 0 chèn thêm của khối loại bỏ các giá trị chèn thêm,

Hình 12 mô tả kiến trúc cụ thể của khối sắp xếp lại vị trí bit,

Hình 13 mô tả quy trình đọc ghi dữ liệu vào bộ nhớ của khối sắp xếp lại vị trí bit.

Mô tả chi tiết sáng chế

Sáng chế đề xuất khối giải điều chỉnh tỷ lệ bit được thể hiện như trên Hình 3 bao gồm ba khối chức năng chính: khối giải đan xen 1, khối loại bỏ các giá trị chèn thêm 2 và khối sắp xếp lại vị trí bit 3.

Đầu vào của khối là E dữ liệu LLR được chia ra thành từng đoạn dữ liệu nhỏ hơn được đưa vào khối theo chu kỳ. Đường dữ liệu đầu vào có độ rộng 128 bit bao gồm 16 LLR, mỗi LLR có độ rộng tám bit. Đầu ra của khối là N dữ liệu LLR cũng được chia thành từng đoạn dữ liệu và đưa ra theo chu kỳ. Đường dữ liệu đầu ra có độ rộng 256 bit bao gồm 16 LLR, mỗi LLR có độ rộng mười sáu bit. Các LLR được sắp xếp trong một chu kỳ dữ liệu theo thứ tự từ bit có trọng số thấp nhất (Least Significant Bit – LSB) đến bit có trọng số cao nhất (Most Significant Bit – MSB) như trong Hình 4, nghĩa là LLR0 sẽ được sắp xếp ở LSB, LLR15 sắp xếp ở vị trí MSB. Do dữ liệu được sắp xếp vào đường dữ liệu 128/256 bit nên sẽ có trường hợp chu kỳ dữ liệu cuối cùng không có đủ mười sáu LLR, do đó cần chèn thêm thêm các bit 0 vào chu kỳ dữ liệu cuối cho đủ 128/256 bit. Cụ thể chi tiết của từng khối chức năng trong khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên của hệ thống 5G như sau:

Khối giải đan xen 1 thực hiện đảo lại dữ liệu theo đúng thứ tự. Đầu vào của khối này, cũng là dữ liệu đầu vào của khối chính, là dữ liệu đã được xen kẽ như đã mô tả trong Hình 2, trong một chu kỳ dữ liệu vào sẽ bao gồm dữ liệu của cả Q_m hàng. Ví dụ với $Q_m = 2$ thì một chu kỳ dữ liệu vào sẽ bao gồm tám LLR của hàng không và tám LLR của hàng một được sắp xen kẽ như trên Hình 5, trong đó ký hiệu $LLR_{x,y}$ là giá trị LLR của bit nằm ở hàng x cột y . Đầu ra của khối có độ rộng 128 bit, bao gồm 16 LLR của từng hàng, cụ thể chu kỳ dữ liệu ra đầu tiên sẽ là dữ liệu của hàng 0, chu kỳ tiếp theo là hàng một, ..., chu kỳ thứ $Q_m - 1$ là hàng $Q_m - 1$.

Kiến trúc cụ thể của khối giải đan xen 1 được thể hiện trong Hình 6 bao gồm: bộ đếm dữ liệu đầu vào 11 và bộ đảo vị trí 12, ngoài ra còn có tám thanh ghi dùng để lưu lại dữ liệu vào. Do mỗi chu kỳ dữ liệu vào chỉ chứa $16/Q_m$ LLR của một hàng, ví dụ $Q_m = 2$ mỗi chu kỳ chứa tám LLR của hàng 0 và hàng một, do đó để lấy đủ 16 LLR cho một chu kỳ dữ liệu ra ta cần phải chờ đủ Q_m chu kỳ dữ liệu vào. Bộ đếm dữ liệu vào 11 sẽ có nhiệm vụ đếm cho đến khi nào đủ Q_m dữ liệu. Bộ tám thanh ghi dịch sẽ lưu dữ liệu vào ở từng chu kỳ. Đầu ra của từng thanh ghi sẽ được đưa vào bộ đảo vị trí 12 để đảo các LLR của chung một hàng về lại với nhau rồi đẩy ra đầu ra. Bộ đảo vị trí 12 sẽ dựa vào giá trị của Q_m để lựa chọn đầu vào phù hợp để tính toán đầu ra. Trong trường hợp $Q_m = 2$, chúng ta chỉ cần hai dữ liệu vào là có thể tính được đầu ra, nên chỉ cần dữ liệu của reg_0 và reg_1 , còn nếu $Q_m = 4$ thì chọn $reg_0, reg_1, reg_2, reg_3, \dots$. Việc đảo xen kẽ dữ liệu đầu vào của khối đảo vị trí trong trường hợp $Q_m = 2$ được thể hiện như trong Hình 7. Các giá trị $llr_{0,0}, llr_{0,1}, \dots, llr_{0,7}$ trong chu kỳ dữ liệu vào 0, $llr_{0,8}, llr_{0,9}, \dots, llr_{0,15}$ trong chu kỳ dữ liệu vào một sẽ được lấy ra và ghép lại theo đúng cấu trúc dữ liệu của một chu kỳ để tạo ra chu kỳ dữ liệu ra 0 là dữ liệu của hàng 0. Các giá trị $llr_{1,0}, llr_{1,1}, \dots, llr_{1,7}$ trong chu kỳ dữ liệu vào 0, $llr_{1,8}, llr_{1,9}, \dots, llr_{1,15}$ trong chu kỳ dữ liệu vào một sẽ được ghép lại để tạo ra chu kỳ dữ liệu ra một là dữ liệu của hàng một. Cứ tiếp tục như vậy, chu kỳ dữ liệu vào hai, ba sẽ tạo ra chu kỳ dữ liệu ra hai là dữ liệu của hàng 0, chu kỳ dữ liệu ra ba là dữ liệu của hàng một.

Dữ liệu LLR đầu ra của khối giải đan xen 1 có thứ tự như trên Hình 8. Các thứ tự này vẫn chưa đúng với thứ tự bit trước khi xen kẽ bên phía phát là $e_0, e_1, e_2, \dots, e_{E-1}$. Đồng

thời có thể có trường hợp số LLR ở mỗi hàng không chia hết cho mười sáu nên dẫn đến chu kỳ dữ liệu cuối của mỗi hàng sẽ bị chèn thêm giá trị 0.

Khối loại bỏ giá trị chèn thêm 2 có nhiệm vụ loại bỏ phần giá trị 0 được chèn thêm ở chu kỳ dữ liệu cuối của từng hàng và đọc dữ liệu ra theo đúng thứ tự. Kiến trúc cụ thể của khối được thể hiện trong Hình 9 bao gồm ba bộ chính: bộ tạo địa chỉ ghi 21, bộ tạo địa chỉ đọc 22 và bộ ghép dữ liệu 23. Ngoài ra còn có bộ nhớ 24 và một thanh ghi để lưu trữ dữ liệu.

Bộ nhớ 24 có nhiệm vụ lưu trữ toàn bộ E giá trị LLR. Dữ liệu đầu vào của khối loại bỏ giá trị chèn thêm 2 là dữ liệu đầu ra của khối giải đan xen 1 và là dữ liệu LLR chưa đúng thứ tự. Bộ tạo địa chỉ ghi 21 sẽ tính toán địa chỉ đúng cho từng dữ liệu đầu vào dựa trên giá trị Q_m và số bit E rồi lưu chúng vào bộ nhớ. Trên Hình 10 mô tả quá trình ghi và đọc vào bộ nhớ 24 cho trường hợp $Q_m = 2$. Dữ liệu đầu vào $d_0, d_1, d_2, d_3, \dots, d_{A-1}, d_A$ là đầu ra của khối giải đan xen 1 trong đó d_0, d_2, \dots, d_{A-1} thuộc hàng 0 còn dữ liệu d_1, d_3, \dots, d_A thuộc hàng một. Do có tổng cộng E giá trị LLR được chia vào Q_m hàng, mỗi hàng có E/Q_m giá trị được sắp xếp thành từng cụm 16 LLR, nên để lưu hết một hàng cần A địa chỉ trong đó:

$$A = \left\lceil \frac{E/Q_m}{16} \right\rceil$$

Dữ liệu d_0 là dữ liệu đầu tiên thuộc hàng 0 nên sẽ được lưu vào địa chỉ 0. Dữ liệu d_1 là dữ liệu đầu tiên của hàng một nên sẽ được lưu vào địa chỉ A. d_2 là dữ liệu thứ hai của hàng 0 nên được lưu vào địa chỉ một, còn d_3 được lưu vào địa chỉ $A + 1 \dots d_{A-1}$ là dữ liệu cuối của hàng 0 sẽ được lưu vào địa chỉ $A - 1$. Sau khi toàn bộ dữ liệu đã được lưu hết, bộ tạo địa chỉ đọc 22 sẽ đọc bộ nhớ theo địa chỉ lần lượt từ 0 đến địa chỉ cuối cùng để đọc dữ liệu ra theo đúng thứ tự $e_0, e_1, e_2, \dots, e_{E-1}$ như trước khi được xen kẽ bên phía phát. Tuy nhiên nếu như dữ liệu cuối cùng của mỗi hàng có chèn thêm dữ liệu 0, dữ liệu đầu ra sẽ trở thành $e_0, e_1, \dots, e_{E/Q_m-1}, 0, \dots, 0, e_{E/Q_m}, \dots$. Do đó cần phải loại bỏ phần giá trị 0 được chèn thêm trước khi đẩy ra đầu ra, đó là nhiệm vụ của bộ ghép dữ liệu 23.

Giả sử dữ liệu cuối cùng của hàng 0 là d_{A-1} không có đủ dữ liệu và cần phải chèn thêm 0. Khi đó d_{A-1} sẽ được lưu lại vào thanh ghi để chờ dữ liệu d_1 của hàng một trước khi được đẩy vào bộ ghép dữ liệu 23 để loại bỏ đi phần giá trị chèn thêm. Hoạt động của bộ

ghép dữ liệu 23 được mô tả như trên Hình 11. Phần giá trị 0 bao gồm a bit được chèn thêm ở đoạn cuối dữ liệu của d_{A-1} sẽ được thay thế bằng a bit đầu tiên có giá trị y của d_1 . Tương tự, phần bit còn lại sau khi bị cắt mất a bit của d_1 có giá trị z sẽ được ghép nối với a bit đầu của d_3 có giá trị t, ... để tạo thành một dữ liệu hoàn chỉnh.

Khối sắp xếp lại vị trí bit 3 sẽ sắp dữ liệu vào đúng vị trí mà nó được cắt ra ở bên phía phát, dựa trên giá trị HARQ. Kiến trúc cụ thể của khối sắp xếp lại vị trí bit 3 được thể hiện như trên Hình 12 bao gồm: bộ tạo địa chỉ ghi 31, bộ tạo địa chỉ đọc 32 và khối điều khiển 33. Ngoài ra có bộ nhớ 34 dùng để lưu N dữ liệu LLR đầu ra, một bộ cộng 35 dùng để tổng hợp dữ liệu đầu vào và dữ liệu có sẵn trong bộ nhớ 34 và một khối lựa chọn 36 dùng để lựa chọn dữ liệu nào sẽ được ghi vào bộ nhớ 34.

Bộ nhớ 34 được dùng để lưu toàn bộ N giá trị LLR, trong đó 16 LLR được gộp thành một dữ liệu nên cần có B địa chỉ để lưu hết N, trong đó:

$$B = \lceil \frac{N}{16} \rceil$$

Địa chỉ lưu dữ liệu vào đầu tiên sẽ được tính toán bởi bộ tạo địa chỉ ghi 31. Như trên Hình 13, d_0 là dữ liệu vào đầu tiên sẽ được lưu tại vị trí địa chỉ bắt đầu, giá trị của địa chỉ này được tính toán dựa trên HARQ và cũng chính là vị trí bắt đầu thực hiện cắt E bit từ N bit bên phía phát. Khối điều khiển 33 sẽ điều khiển khối lựa chọn 36 để ghi vào bộ nhớ 34 cho đến khi nào địa chỉ ghi còn chưa đạt đến $B - 1$, đây là lượt 0. Hết lượt 0, chuyển qua lượt một, bộ tạo địa chỉ đọc 32 sẽ cho đọc dữ liệu trong bộ nhớ 34 bắt đầu từ địa chỉ 0 để đem cộng với dữ liệu đầu vào thông qua bộ cộng 35. Đầu ra của bộ cộng 35 được khối điều khiển 33 điều khiển bộ lựa chọn 36 để ghi vào bộ nhớ 34 tại địa chỉ ghi lúc này chính bằng địa chỉ đọc dữ liệu do bộ tạo địa chỉ đọc 32 tạo ra. Việc thực hiện lượt một cứ tiếp diễn cho đến khi nào hết toàn bộ dữ liệu vào. Sau khi toàn bộ dữ liệu vào đã được đem tổng hợp và được lưu lại trong bộ nhớ 34, khi đó bộ tạo địa chỉ đọc 32 sẽ cho đọc bộ nhớ 34 một lần nữa từ địa chỉ 0 cho đến $B - 1$ để đọc hết N dữ liệu ra, quá trình hoạt động của cả khối sắp xếp lại vị trí bit kết thúc.

Đầu ra của khối sắp xếp lại vị trí bit 3 cũng chính là đầu ra cuối cùng của cả hệ thống là N giá trị LLR được sắp xếp lại theo đúng thứ tự như bên phía phát, sẵn sàng được đem đi giải mã hóa kênh.

Hiệu quả đạt được của sáng chế

Phương pháp theo sáng chế là khả thi, dễ áp dụng và đảm bảo được các yêu cầu về hiệu năng của hệ thống. Sáng chế có thể đáp ứng được toàn bộ các giá trị HARQ 0, 1, 2, 3 theo như chuẩn 5G yêu cầu, có thể dễ dàng thực hiện đường ống hóa, song song hóa để tối ưu tốc độ xử lý.

Yêu cầu bảo hộ

1. Khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên trong hệ thống 5G gồm các khối chức năng:

khối giải đan xen: thực hiện đảo lại dữ liệu theo đúng thứ tự; đầu vào của khối này, cũng là dữ liệu đầu vào của khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên trong hệ thống 5G, là dữ liệu đã được xen kẽ, trong một chu kỳ dữ liệu vào sẽ bao gồm dữ liệu của cả Q_m hàng; cụ thể chu kỳ dữ liệu ra đầu tiên sẽ là dữ liệu của hàng 0, chu kỳ tiếp theo là hàng một, ..., chu kỳ thứ $Q_m - 1$ là hàng $Q_m - 1$; kiến trúc cụ thể của khối giải đan xen bao gồm: bộ đếm dữ liệu đầu vào và bộ đảo vị trí, ngoài ra còn có tám thanh ghi dùng để lưu lại dữ liệu vào; do mỗi chu kỳ dữ liệu vào chỉ chứa $16/Q_m$ LLR của một hàng, do đó để lấy đủ 16 LLR cho một chu kỳ dữ liệu ra ta cần phải chờ đủ Q_m chu kỳ dữ liệu vào; bộ đếm dữ liệu đầu vào sẽ có nhiệm vụ đếm cho đến khi nào đủ Q_m dữ liệu; bộ tám thanh ghi dịch sẽ lưu dữ liệu vào ở từng chu kỳ; đầu ra của từng thanh ghi sẽ được đưa vào bộ đảo vị trí để đảo các LLR của chung một hàng về lại với nhau rồi đẩy ra đầu ra; bộ đảo vị trí sẽ dựa vào giá trị của Q_m để lựa chọn đầu vào phù hợp để tính toán đầu ra; dữ liệu LLR đầu ra của khối giải đan xen có thứ tự chưa đúng với thứ tự bit trước khi xen kẽ bên phía phát; đồng thời có thể có trường hợp số LLR ở mỗi hàng không chia hết cho mười sáu nên dẫn đến chu kỳ dữ liệu cuối của mỗi hàng sẽ bị chèn thêm giá trị 0;

khối loại bỏ giá trị chèn thêm có nhiệm vụ loại bỏ phần giá trị 0 được chèn thêm ở chu kỳ dữ liệu cuối của từng hàng và đọc dữ liệu ra theo đúng thứ tự; kiến trúc cụ thể của khối loại bỏ giá trị chèn thêm bao gồm ba bộ chính: bộ tạo địa chỉ ghi, bộ tạo địa chỉ đọc và bộ ghép dữ liệu; ngoài ra còn có bộ nhớ và một thanh ghi để lưu trữ dữ liệu; trong đó: bộ nhớ có nhiệm vụ lưu trữ toàn bộ E giá trị LLR; dữ liệu đầu vào của khối loại bỏ giá trị chèn thêm là dữ liệu đầu ra của khối giải đan xen và là dữ liệu LLR chưa đúng thứ tự; bộ tạo địa chỉ ghi sẽ tính toán địa chỉ đúng cho từng dữ liệu đầu vào dựa trên giá trị Q_m và số bit E rồi lưu chúng vào bộ nhớ; do có tổng cộng E giá trị LLR được chia vào Q_m hàng, mỗi hàng có E/Q_m giá trị được sắp xếp thành từng cụm 16 LLR, nên để lưu hết một hàng cần A địa chỉ trong đó:

$$A = \lceil \frac{E/Q_m}{16} \rceil$$

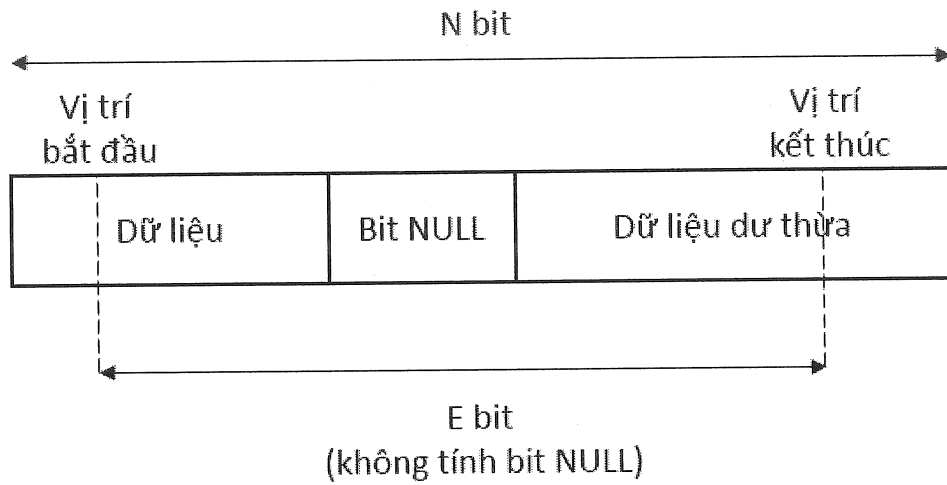
dữ liệu d_0 là dữ liệu đầu tiên thuộc hàng 0 nên sẽ được lưu vào địa chỉ 0; dữ liệu d_1 là dữ liệu đầu tiên của hàng một nên sẽ được lưu vào địa chỉ A; dữ liệu d_2 là dữ liệu thứ hai của hàng 0 nên được lưu vào địa chỉ một, còn dữ liệu d_3 được lưu vào địa chỉ $A + 1 \dots d_{A-1}$ là dữ liệu cuối của hàng 0 sẽ được lưu vào địa chỉ $A - 1$; sau khi toàn bộ dữ liệu đã được lưu hết, bộ tạo địa chỉ đọc sẽ đọc bộ nhớ theo địa chỉ lần lượt từ 0 đến địa chỉ cuối cùng để đọc dữ liệu ra theo đúng thứ tự $e_0, e_1, e_2, \dots e_{E-1}$ như trước khi được xen kẽ bên phía phát; tuy nhiên nếu như dữ liệu cuối cùng của mỗi hàng có chèn thêm dữ liệu 0, dữ liệu đầu ra sẽ trở thành $e_0, e_1, \dots e_{E/Q_m-1}, 0, \dots 0, e_{E/Q_m}, \dots$; do đó cần phải loại bỏ phần giá trị 0 được chèn thêm trước khi đẩy ra đầu ra, đó là nhiệm vụ của bộ ghép dữ liệu; hoạt động của bộ ghép dữ liệu được mô tả như sau: phần giá trị 0 bao gồm a bit được chèn thêm ở đoạn cuối dữ liệu của d_{A-1} sẽ được thay thế bằng a bit đầu tiên có giá trị y của d_1 ; tương tự, phần bit còn lại sau khi bị cắt mất a bit của d_1 có giá trị z sẽ được ghép nối với a bit đầu của d_3 có giá trị t, ... để tạo thành một dữ liệu hoàn chỉnh;

khối sắp xếp lại vị trí bit sẽ sắp dữ liệu vào đúng vị trí mà nó được cắt ra ở bên phía phát, dựa trên giá trị HARQ; kiến trúc cụ thể của khối sắp xếp lại vị trí bit bao gồm: bộ tạo địa chỉ ghi, bộ tạo địa chỉ đọc và khối điều khiển; ngoài ra có bộ nhớ dùng để lưu N dữ liệu LLR đầu ra, một bộ cộng dùng để tổng hợp dữ liệu đầu vào và dữ liệu có sẵn trong bộ nhớ và một khối lựa chọn dùng để lựa chọn dữ liệu nào sẽ được ghi vào bộ nhớ; trong đó: bộ nhớ được dùng để lưu toàn bộ N giá trị LLR, trong đó 16 LLR được gộp thành một dữ liệu nên cần có B địa chỉ để lưu hết N, trong đó:

$$B = \lceil \frac{N}{16} \rceil$$

địa chỉ lưu dữ liệu vào đầu tiên sẽ được tính toán bởi bộ tạo địa chỉ ghi, d_0 là dữ liệu vào đầu tiên sẽ được lưu tại vị trí địa chỉ bắt đầu, giá trị của địa chỉ này này được tính toán dựa trên HARQ và cũng chính là vị trí bắt đầu thực hiện cắt E bit từ N bit bên phía phát; khối điều khiển sẽ điều khiển khối lựa chọn để ghi vào bộ nhớ cho đến khi nào địa chỉ ghi còn chưa đạt đến $B - 1$, đây là lượt 0; hết lượt 0, chuyển qua lượt một, bộ tạo địa chỉ đọc sẽ cho đọc dữ liệu trong bộ nhớ bắt đầu từ địa chỉ 0 để đem cộng với dữ liệu đầu vào thông

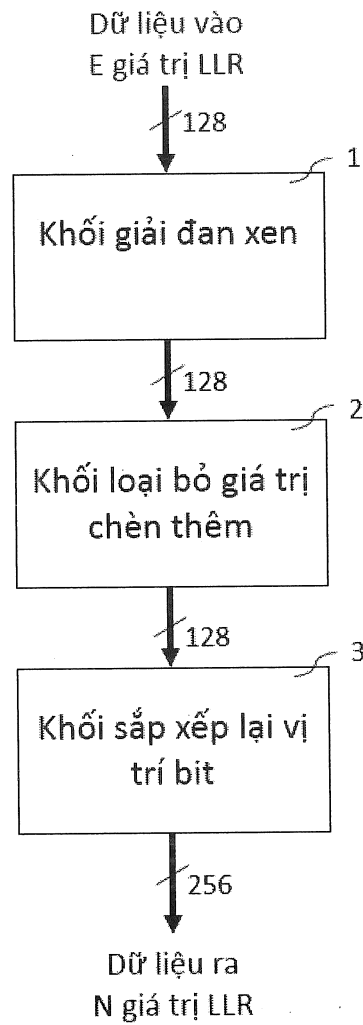
qua bộ cộng; đầu ra của bộ cộng được khối điều khiển điều khiển bộ lựa chọn để ghi vào bộ nhớ tại địa chỉ ghi lúc này chính bằng địa chỉ đọc dữ liệu do bộ tạo địa chỉ đọc tạo ra; việc thực hiện lượt một cứ tiếp diễn cho đến khi nào hết toàn bộ dữ liệu vào; sau khi toàn bộ dữ liệu vào đã được đem tổng hợp và được lưu lại trong bộ nhớ, khi đó bộ tạo địa chỉ đọc sẽ cho đọc bộ nhớ một lần nữa từ địa chỉ 0 cho đến $B - 1$ để đọc hết N dữ liệu ra, quá trình hoạt động của cả khối sắp xếp lại vị trí bit kết thúc; đầu ra của khối sắp xếp lại vị trí bit cũng chính là đầu ra cuối cùng của cả khối giải điều chỉnh tỷ lệ bit kênh dữ liệu vật lý dùng chung đường lên trong hệ thống 5G là N giá trị LLR được sắp xếp lại theo đúng thứ tự như bên phía phát, sẵn sàng được đem đi giải mã hóa kênh.



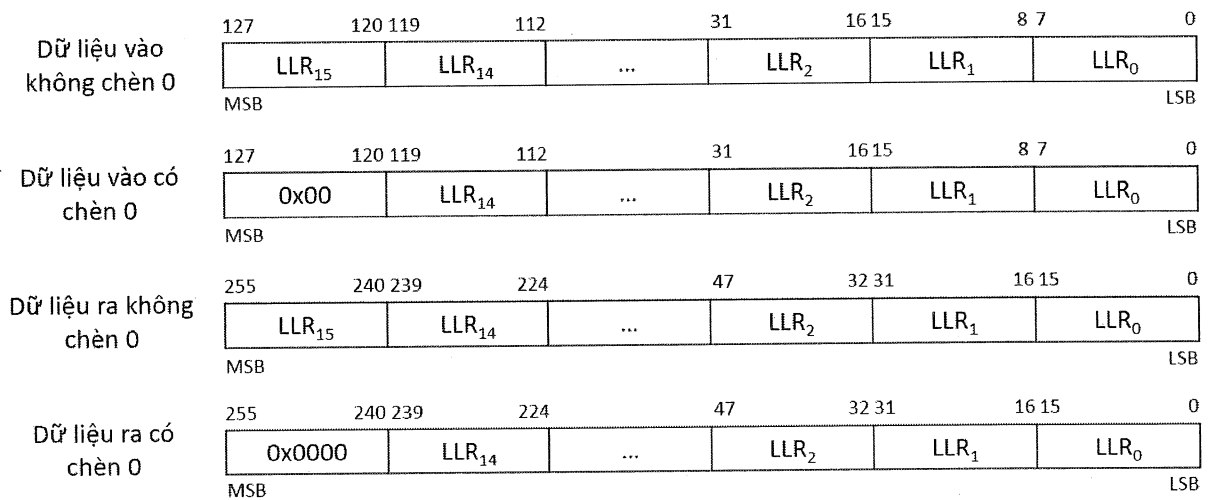
Hình 1

	cột 0	cột 1	cột 2	cột 3	...	cột $E/Q_m - 1$
hàng 0	e_0	e_1	e_2	e_3	...	$e_{E/Q_m - 1}$
hàng 1	e_{E/Q_m}	$e_{E/Q_m + 1}$	$e_{E/Q_m + 2}$	$e_{E/Q_m + 3}$...	$e_{2E/Q_m - 1}$
...
hàng $Q_m - 1$	$e_{(Q_m - 1) * E / Q_m}$	$e_{(Q_m - 1) * E / Q_m + 1}$	$e_{(Q_m - 1) * E / Q_m + 2}$	$e_{(Q_m - 1) * E / Q_m + 3}$...	$e_{E - 1}$

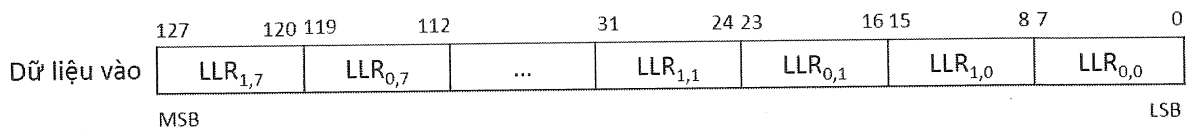
Hình 2



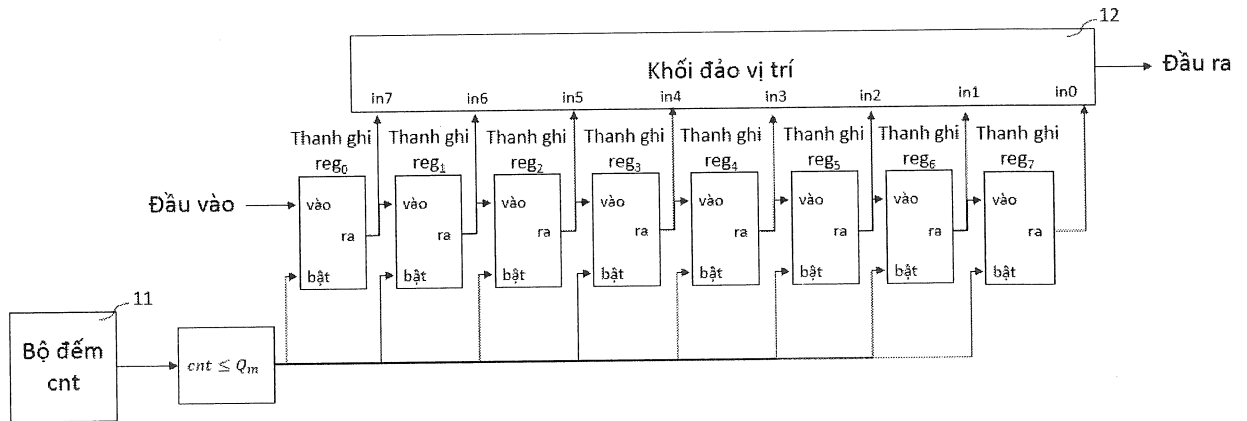
Hình 3



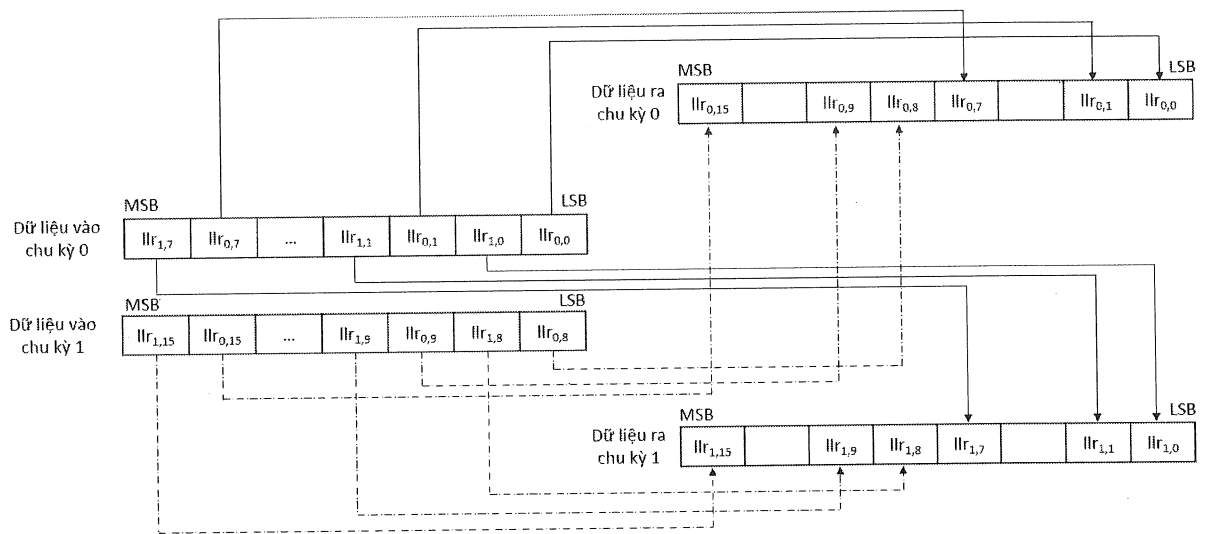
Hình 4



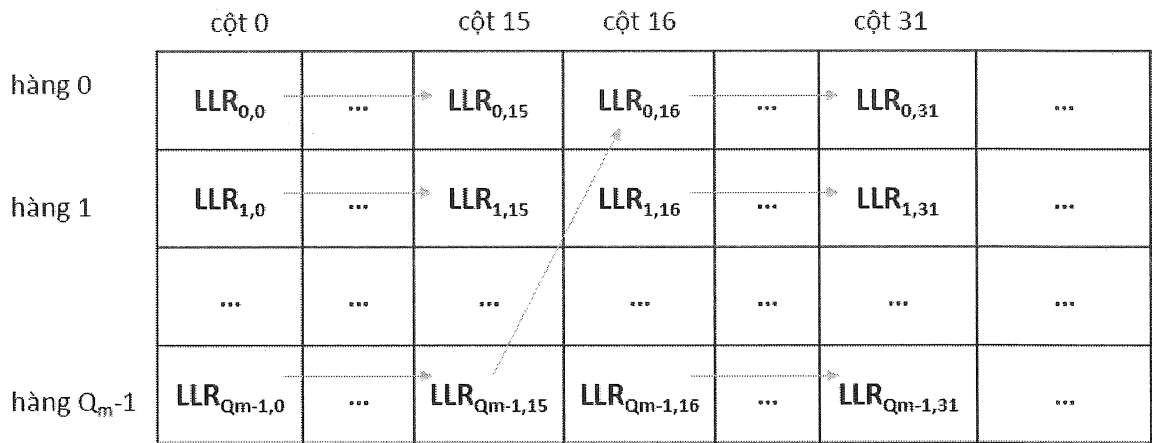
Hình 5



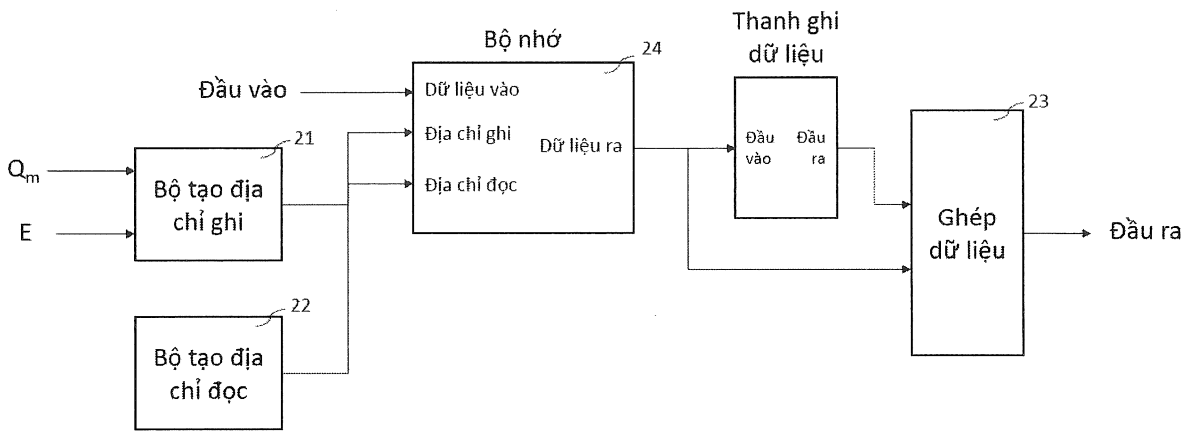
Hình 6



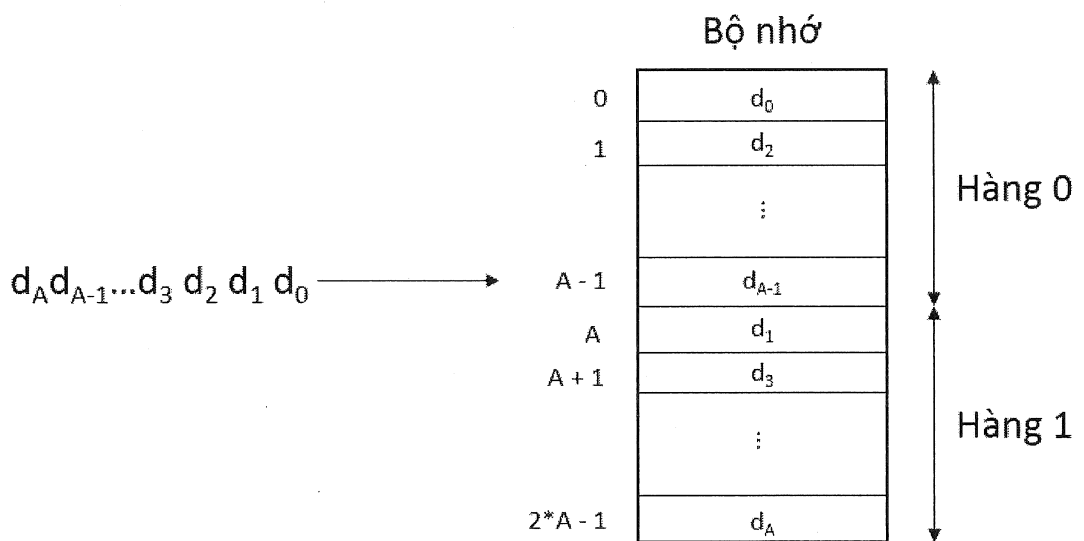
Hình 7



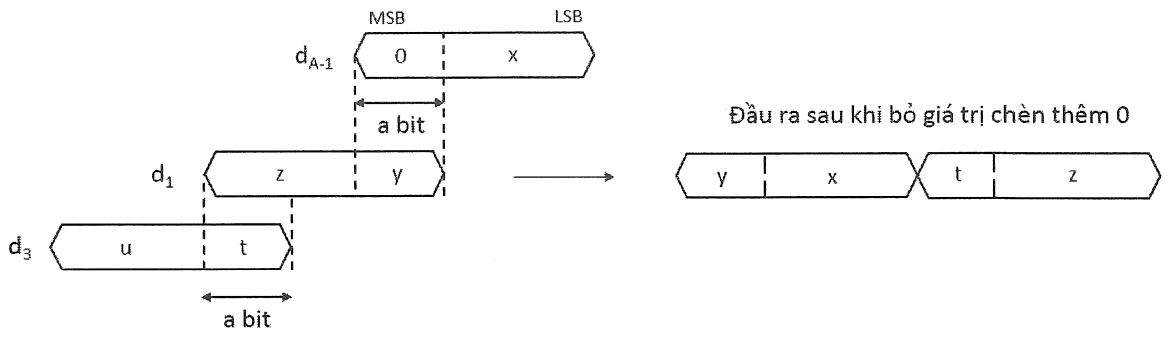
Hình 8



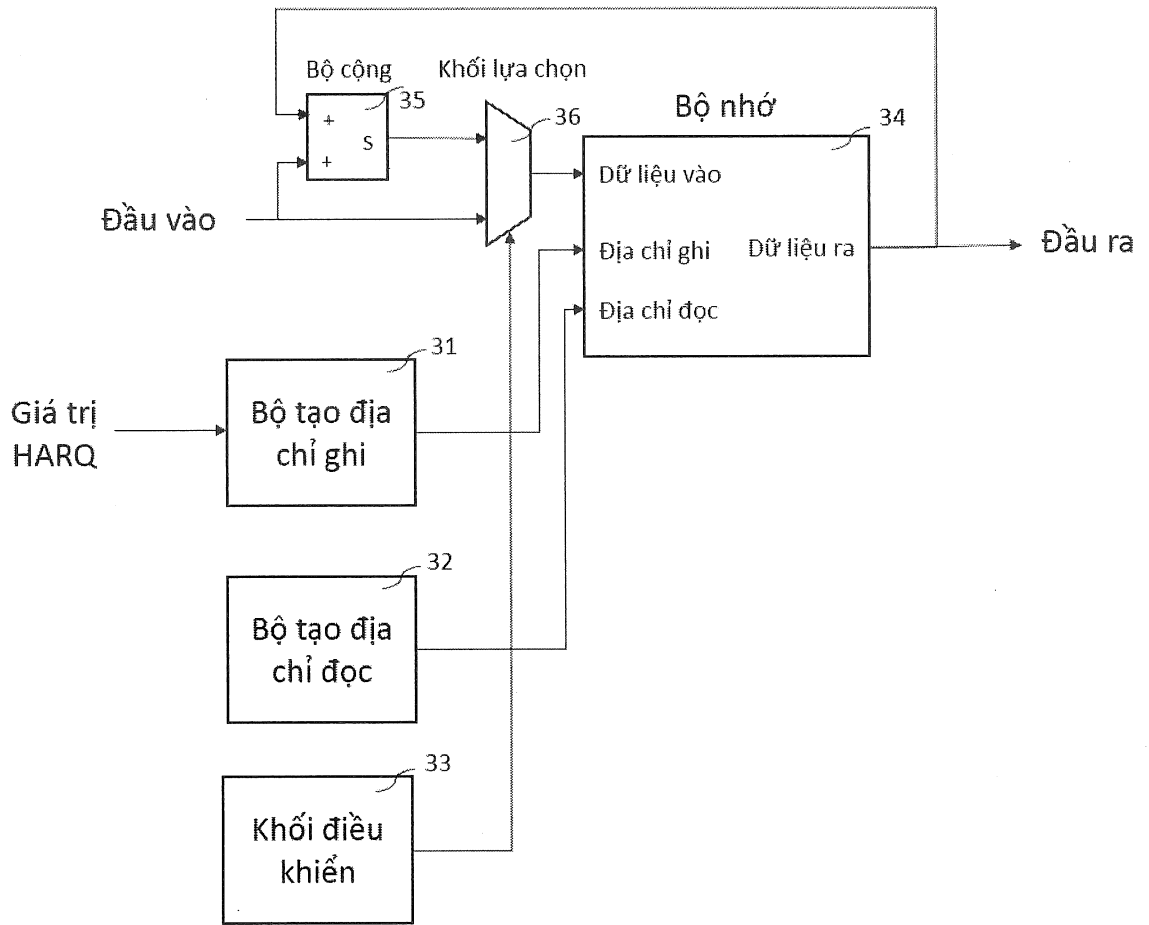
Hình 9



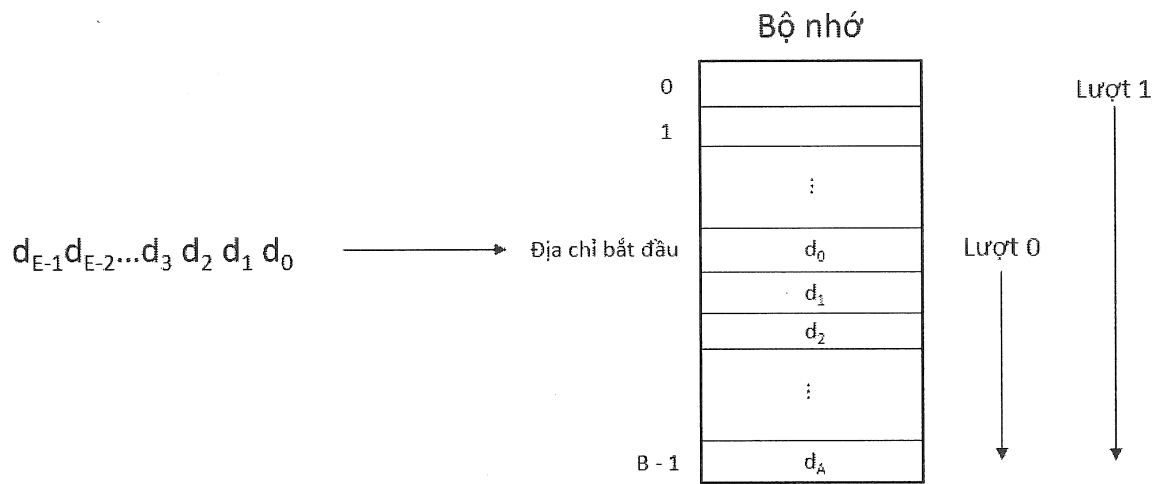
Hình 10



Hình 11



Hình 12



Hình 13