



(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)



1-0036734

(51)⁷ H04N 7/32

(13) B

(21) 1-2019-04648

(22) 27/06/2012

(62) 1-2013-03986

(86) PCT/JP2012/067023 27/06/2012

(87) WO 2013/002412 A8 03/01/2013

(30) 13/174,564 30/06/2011 US

(45) 25/08/2023 425

(43) 25/11/2019 380A

(73) VELOS MEDIA INTERNATIONAL LIMITED (IE)

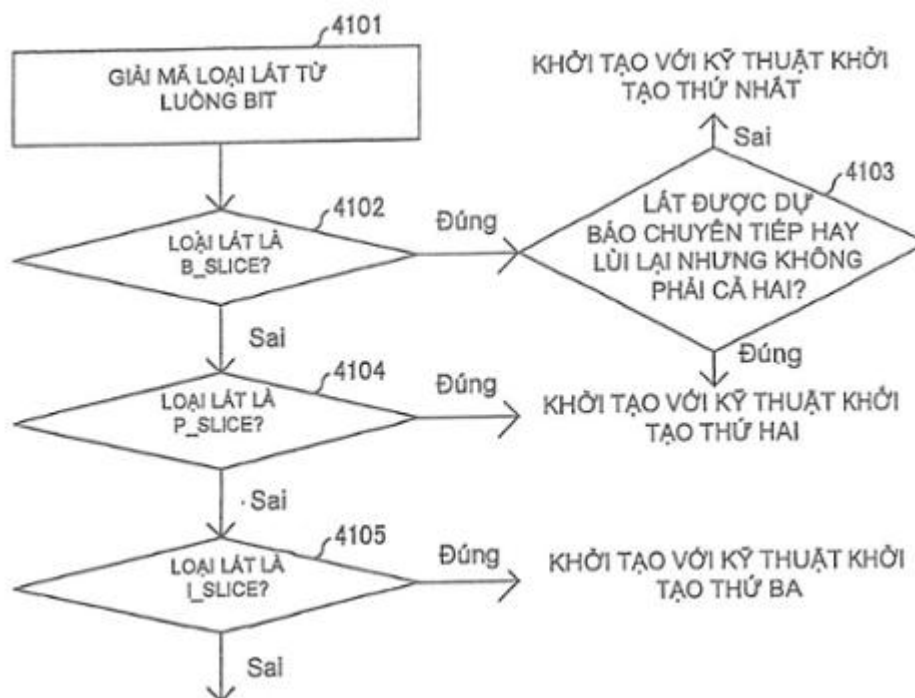
Unit 32, the Hyde Building, The Park, Carrickmines, Dublin 18 Ireland

(72) MISRA, Kiran (IN); SEGALL, Christopher A. (US).

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

(54) PHƯƠNG PHÁP VÀ THIẾT BỊ GIẢI MÃ KHUNG VIDEO

(57) Sáng chế đề cập đến phương pháp giải mã khung video của chuỗi video, phương pháp này bao gồm các bước: (a) thu lát ở bộ giải mã video; (b) nhận dạng kiểu của lát như là một trong ít nhất lát được dự báo thứ nhất và lát được dự báo thứ hai; (c) thu thông tin để xác định phương pháp khởi tạo ngữ cảnh tương ứng với lát này; (d) khởi tạo ngữ cảnh được kết hợp với lát bằng cách sử dụng một trong số phương pháp khởi tạo ngữ cảnh thứ nhất và phương pháp khởi tạo ngữ cảnh thứ hai dựa vào thông tin thu được; và (e) giải mã khung video sử dụng kiểu của lát và một trong phương pháp khởi tạo ngữ cảnh thứ nhất và phương pháp khởi tạo ngữ cảnh thứ hai.



Lĩnh vực kỹ thuật được đề cập

Nói chung, sáng chế đề cập đến phương pháp mã hóa video và cụ thể là sáng chế đề cập đến phương pháp khởi tạo ngữ cảnh.

Tình trạng kỹ thuật của sáng chế

Các tiêu chuẩn và các phương pháp mã hóa video đã biết là, chẳng hạn H.264/MPEG-4 AVC (H.264/AVC) và mô hình thử nghiệm đang xem xét (JCT-VC Test Model under Consideration (TMuC)), có thể tạo ra hiệu quả mã hóa cao hơn so với các tiêu chuẩn và các phương pháp đã biết trước phải chấp nhận tính phức tạp cao hơn. Các tiêu chuẩn và các phương pháp mã hóa video yêu cầu về độ phân giải và yêu cầu về chất lượng tăng có thể còn tăng tính phức tạp của chúng. Các bộ giải mã hỗ trợ giải mã song song có thể cải thiện tốc độ giải mã và giảm đòi hỏi về bộ nhớ. Ngoài ra, sự cải tiến ở các bộ xử lý đa lõi có thể tạo ra bộ mã hóa và giải mã hỗ trợ phương pháp giải mã song song mong muốn.

H.264/MPEG-4 AVC (nhóm cộng tác video về ITU-T VCEG và ISO/IEC MPEG, “H.264: mã hóa video cải tiến cho các dịch vụ nghe nhìn chung,” ITU-T Rec. H.264 và ISO/IEC 14496-10 (MPEG4 – phần 10), tháng 11 năm 2007) – [Joint Video Team of ITU-T VCEG và ISO/IEC MPEG, “H.264: Advanced video coding for generic audiovisual services,” ITU-T Rec. H.264 và ISO/IEC 14496-10 (MPEG4 - Part 10), November 2007], sẽ được viện dẫn kết hợp trong bản mô tả này nhằm tham khảo, là tiêu chuẩn (bộ mã hóa/bộ giải mã) mã hóa và giải mã video sử dụng sự dự báo khối macrô sau khi mã hóa dư để giảm lượng dư không gian và thời gian trong chuỗi video để nén một cách có hiệu quả.

Mô hình thử nghiệm đang xem xét (Test Model under Consideration (TMuC)) [JCT-VC A205, “Test Model under Consideration”, ngày 16 tháng 6 năm 2010], sẽ được viện dẫn kết hợp trong bản mô tả này nhằm tham khảo, là mô hình thử nghiệm khởi đầu của JCT-VC. TMuC, bằng cách sử dụng bộ mã hóa cơ sở gọi là khối cây mã hóa (CTB) có thể có các kích thước biến đổi, có thể tạo ra tính linh hoạt hơn so với H.264/AVC.

Bản chất kỹ thuật của sáng chế

Mục đích của sáng chế là đề xuất các phương pháp và các hệ thống để mã hóa entropi song song, và đề xuất các phương pháp và các hệ thống để giải mã entropi song song.

Theo một phương án thực hiện sáng chế, phương pháp để giải mã khung video của chuỗi video bao gồm:

(a) thu lát ở bộ giải mã video;

(b) nhận dạng kiểu của lát bao gồm ít nhất lát được dự báo thứ nhất và lát được dự báo thứ hai; và

(c) khởi tạo ngữ cảnh được kết hợp với lát bằng cách sử dụng phương pháp khởi tạo ngữ cảnh tương ứng với kiểu của lát;

trong đó phương pháp khởi tạo ngữ cảnh được thay bằng phương pháp khởi tạo ngữ cảnh khác nhau tương ứng với lát được dự báo thứ nhất, nếu kiểu của lát được nhận dạng là lát được dự báo thứ hai.

Theo một phương án thực hiện sáng chế, phương pháp để giải mã khung video trong chuỗi video bao gồm:

(a) thu lát trong bộ giải mã video;

(b) nhận dạng nếu lát là lát B được dự báo;

(c) khởi tạo, bằng cách sử dụng kỹ thuật khác so với kỹ thuật được dùng cho lát B, ngữ cảnh được kết hợp với lát này.

Các mục đích nêu trên và các mục đích khác, các dấu hiệu kỹ thuật khác, và các hiệu quả của sáng chế sẽ rõ ràng hơn thông qua phần mô tả sáng chế dưới đây có dựa vào các hình vẽ kèm theo.

Mô tả vắn tắt các hình vẽ

Fig.1 là hình vẽ thể hiện bộ mã hóa video H.264/AVC (đã biết);

Fig.2 là hình vẽ thể hiện bộ giải mã video H.264/AVC (đã biết);

Fig.3 là hình vẽ thể hiện làm ví dụ về cấu hình lát (đã biết);

Fig.4 là hình vẽ thể hiện cấu hình nhóm lát làm ví dụ (đã biết);

Fig.5 là hình vẽ thể hiện phân chia lát làm ví dụ theo các phương án theo sáng chế, trong đó ảnh có thể được phân chia trong ít nhất một lát phức hồi và lát

phục hồi có thể được phân chia thành nhiều hơn một lát entropi;

Fig.6 là lưu đồ thể hiện các phương án làm ví dụ về sáng chế bao gồm lát entropi;

Fig.7 là lưu đồ thể hiện các phương án thực hiện sáng chế bao gồm phương pháp giải mã entropi song song của các lát entropi sau khi phục hồi lát;

Fig.8 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc dồn kênh dữ liệu dự báo/dữ liệu dư ở mức độ ảnh để tạo cấu hình lát entropi;

Fig.9 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm dồn kênh mặt phẳng màu ở mức độ ảnh để tạo cấu hình lát entropi;

Fig.10 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc chuyển mã luồng bit bằng việc giải mã entropi, tạo các lát entropi và mã hóa entropi;

Fig.11 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm phân chia lát phục hồi thành các lát entropi, trong đó số lượng các bin (ngăn nhỏ) được kết hợp với mỗi lát entropi trong các lát entropi không vượt quá số lượng bin định trước;

Fig.12 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc phân chia lát phục hồi thành các lát entropi, trong đó các bin có thể được kết hợp với lát entropi cho đến khi số lượng các bin trong lát entropi vượt giá trị ngưỡng dựa vào số lượng bin tối đa định trước;

Fig.13 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc phân chia lát phục hồi thành các lát entropi, trong đó số lượng các bin được kết hợp với mỗi lát entropi trong các lát entropi không vượt quá số lượng bin định trước và mỗi lát phục hồi chứa không vượt quá số lượng định trước của các khối macro;

Fig.14 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc phân chia lát phục hồi thành các lát entropi, trong đó các bin có thể được kết hợp với lát entropi cho đến khi số lượng các bin trong lát entropi vượt quá giá trị ngưỡng dựa vào số lượng bin tối đa định trước và mỗi lát phục hồi chứa không vượt quá số lượng định trước của các khối macro;

Fig.15 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao

gồm việc phân chia lát phục hồi thành các lát entropi, trong đó số lượng các bit được kết hợp với mỗi lát entropi trong các lát entropi không vượt quá số lượng các bit định trước;

Fig.16 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc phân chia lát phục hồi thành các lát entropi, trong đó các bit có thể được kết hợp với lát entropi cho đến khi số lượng các bit trong các lát entropi vượt quá giá trị ngưỡng dựa vào số lượng các bit tối đa định trước;

Fig.17 là hình vẽ thể hiện các phương án thực hiện làm ví dụ của sáng chế bao gồm các bộ mã hóa bin;

Fig.18 là hình vẽ thể hiện các phương án thực hiện làm ví dụ của sáng chế bao gồm các bộ thích ứng ngữ cảnh;

Fig.19 là hình vẽ thể hiện các phương án thực hiện làm ví dụ của sáng chế bao gồm các bộ mã hóa bin và các bộ thích ứng ngữ cảnh;

Fig.20 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc phân chia lát phục hồi thành các lát entropi, trong đó kích thước của lát entropi được hạn chế để giới hạn số lượng các bin được sử dụng, trong lát entropi, bởi mỗi bộ mã hóa entropi được hạn chế;

Fig.21 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc phân chia lát phục hồi thành các lát entropi, trong đó kích thước của lát entropi được hạn chế để giới hạn số lượng các bin được sử dụng, trong lát entropi, bởi mỗi bộ mã hóa entropi được hạn chế;

Fig.22 là hình vẽ thể hiện các phương án thực hiện làm ví dụ của sáng chế bao gồm các bộ giải mã bin;

Fig.23 là hình vẽ thể hiện các phương án thực hiện làm ví dụ của sáng chế bao gồm các bộ thích ứng ngữ cảnh;

Fig.24 là hình vẽ thể hiện các phương án thực hiện làm ví dụ của sáng chế bao gồm các bộ giải mã bin bội và các bộ thích ứng ngữ cảnh;

Fig.25 là hình vẽ thể hiện sự phân chia làm ví dụ của khối phục hồi thành các lát entropi trong đó các khối macro trong lát entropi là liên tiếp;

Fig.26 là hình vẽ thể hiện sự phân chia làm ví dụ của khối phục hồi thành các lát entropi trong đó các khối macro trong lát entropi không liên tiếp;

Fig.27 là hình vẽ thể hiện các khối lân cận không liên tiếp, được sử dụng khi giải mã entropi, đối với sự phân chia làm ví dụ của khối phục hồi thành các lát entropi trong đó các khối macro trong lát entropi không liên tiếp;

Fig.28 là hình vẽ thể hiện các khối lân cận được sử dụng khi giải mã entropi và sự phục hồi khối trong lát entropi đối với sự phân chia làm ví dụ của khối phục hồi thành lát entropi trong đó các khối macro trong lát entropi không liên tiếp;

Fig.29 là hình vẽ thể hiện một phần làm ví dụ của luồng bit làm ví dụ minh họa các giới hạn vị trí của mào đầu của lát entropi;

Fig.30 là hình vẽ thể hiện một phần làm ví dụ của luồng bit làm ví dụ minh họa các giới hạn vị trí của mào đầu của lát entropi;

Fig.31 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm bộ giải mã entropi xử lý các phần được giới hạn của luồng bit để nhận dạng mào đầu lát entropi;

Fig.32 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm bộ giải mã entropi xử lý các phần được giới hạn của luồng bit để nhận dạng mào đầu lát entropi;

Fig.33 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm bộ giải mã entropi xử lý các phần được giới hạn của luồng bit để nhận dạng mào đầu lát entropi;

Fig.34 là hình vẽ thể hiện sơ đồ khởi tạo bảng ngữ cảnh làm ví dụ trong các lát entropi theo các phương án thực hiện sáng chế;

Fig.35 là hình vẽ thể hiện việc quét thích ứng theo các phương án làm ví dụ thực hiện sáng chế.

Fig.36 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm bộ mã hóa entropi với việc tìm nạp ngữ cảnh được tách với thứ tự quét hệ số;

Fig.37 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm bộ giải mã entropi với việc tìm nạp ngữ cảnh được tách với thứ tự quét hệ số;

Fig.38 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao gồm việc thích ứng ngữ cảnh trên cơ sở đếm bin;

Fig.39 là hình vẽ thể hiện làm ví dụ, phân chia số lượng các bin được xử lý;

Fig.40 là lưu đồ thể hiện các phương án làm ví dụ thực hiện sáng chế bao

gồm thích ứng ngữ cảnh trên cơ sở tham số lượng tử hóa;

Fig.41 là hình vẽ thể hiện kỹ thuật khởi tạo ngữ cảnh;

Fig.42 là hình vẽ thể hiện kỹ thuật khởi tạo ngữ cảnh khác; và

Fig.43 là hình vẽ thể hiện kỹ thuật khởi tạo ngữ cảnh.

Mô tả chi tiết sáng chế

Các phương án thực hiện sáng chế sẽ được mô tả chi tiết dưới đây có dựa vào các hình vẽ kèm theo, trong đó các thành phần cấu thành tương đương nhau được ký hiệu bởi cùng các ký hiệu chỉ dẫn trong toàn bộ bản mô tả. Các hình vẽ được liệt kê trên đây chỉ nhằm để minh họa cho phần mô tả chi tiết của sáng chế.

Cần hiểu rằng các thành phần cấu thành của sáng chế, như được mô tả và được minh họa trên các hình vẽ trong bản mô tả này, có thể được thay đổi và được bố trí theo khía cạnh rộng nhất theo các cấu hình khác nhau. Vì thế, phần mô tả chi tiết dưới đây theo các phương án của các phương pháp và các hệ thống của sáng chế không nhằm giới hạn phạm vi của sáng chế, mà chỉ là các phương án ưu tiên thực hiện sáng chế.

Các thành phần cấu thành theo các phương án thực hiện sáng chế có thể được thực hiện bằng phần cứng, phần sụn và/hoặc phần mềm. Mặc dù các phương án làm ví dụ được thể hiện trong bản mô tả này chỉ mô tả một trong các dạng phương án thực hiện, nhưng cần hiểu rằng người có hiểu biết trung bình trong lĩnh vực kỹ thuật liên quan có thể thực hiện sửa đổi theo dạng bất kỳ khác mà không nằm ngoài phạm vi của sáng chế.

Mặc dù bộ mã hóa/giải mã video (codec) dùng để mã hóa/giải mã entropi có thể được sử dụng theo các phương án thực hiện sáng chế, nhưng nhiều phương án thực hiện làm ví dụ khác của sáng chế sẽ được minh họa liên quan đến bộ mã hóa H.264/AVC và bộ giải mã H.264/AVC. Điều này chỉ nhằm minh họa các phương án thực hiện sáng chế và không nhằm giới hạn sáng chế.

Các phương án làm ví dụ thực hiện sáng chế có thể được mô tả liên quan đến khối macrô làm đơn vị cơ bản. Điều này chỉ nhằm minh họa và không nhằm giới hạn sáng chế.

Đơn sáng chế Mỹ số 12/058301 đề xuất “Các phương pháp và các hệ thống

đề mã hóa và giải mã video song song” nộp ngày 28 tháng 3 năm 2008 sẽ được trích dẫn dưới đây nhằm tham khảo. Đơn sáng chế Mỹ số 12/579236 đã đề xuất “Các phương pháp và các hệ thống đề mã hóa và giải mã video song song” nộp ngày 14 tháng 10 năm 2009 sẽ được trích dẫn dưới đây nhằm tham khảo.

Các tiêu chuẩn và các phương pháp mã hóa video đã biết, chẳng hạn H.264/AVC và TMuC, có thể tạo ra hiệu suất mã hóa cao hơn so với các phương pháp và các tiêu chuẩn trước đó với sự chấp nhận tính phức tạp cao hơn. Việc tăng các đòi hỏi về chất lượng và độ phân giải ở các phương pháp và các tiêu chuẩn mã hóa video có thể còn làm tăng tính phức tạp của chúng. Bộ giải mã hỗ trợ giải mã song song có thể cải thiện tốc độ giải mã và giảm các đòi hỏi về bộ nhớ. Ngoài ra, sự cải tiến ở các bộ xử lý đa lõi có thể tạo ra các bộ mã hóa và các bộ giải mã hỗ trợ giải mã song song mong muốn.

H.264/AVC, và các tiêu chuẩn và các phương pháp mã hóa video khác, dựa vào việc đạt được phương pháp mã hóa video lai trên cơ sở khối, trong đó thuật toán mã hóa nguồn là phương pháp lai của liên ảnh, còn gọi là dự báo liên khung, nội ảnh còn gọi là dự báo nội khung và mã hóa biến đổi của lượng dư dự báo. Dự báo liên khung có thể sử dụng các lượng dư theo thời gian, và mã hóa nội khung và biến đổi của lượng dư dự báo có thể sử dụng các lượng dư không gian.

Fig.1 là hình vẽ thể hiện sơ đồ khối làm ví dụ của bộ mã hóa video H.264/AVC 2. Ảnh đầu vào 4, còn gọi là khung, có thể được biểu thị để mã hóa. Tín hiệu được dự báo 6 và tín hiệu dư 8 có thể được tạo ra, trong đó tín hiệu được dự báo 6 có thể được dựa vào hoặc dự báo liên khung 10 hoặc dự báo nội khung 12. Dự báo liên khung 10 có thể được xác định bởi bộ bù chuyển động 14 bằng cách sử dụng ảnh tham chiếu được lưu trữ 16, còn gọi là khung tham chiếu, và thông tin chuyển động 19 được xác định bởi bộ đánh giá chuyển động 18 xử lý giữa khung đầu vào 4 và khung tham chiếu 16. Dự báo nội khung 12 có thể được xác định bởi bộ dự báo nội khung 20 bằng cách sử dụng tín hiệu được giải mã 22. Tín hiệu dư 8 có thể được xác định bằng cách trừ tín hiệu đầu vào 4 từ tín hiệu dự báo 6. Tín hiệu dư 8 được biến đổi, định tỷ lệ và được lượng tử hóa bởi bộ biến đổi/định tỷ lệ/lượng tử hóa 24, nhờ đó tạo ra các hệ số biến đổi được lượng tử hóa 26. Tín hiệu được giải mã 22 có thể được tạo ra bằng cách cộng tín hiệu được dự

báo 6 tới tín hiệu 28 được tạo ra bởi bộ đảo (biến đổi/định tỷ lệ/lượng tử hóa) 30 bằng cách sử dụng các hệ số biến đổi được lượng tử hóa 26. Thông tin chuyển động 19 và các hệ số biến đổi được lượng tử hóa 26 có thể là entropi được mã hóa bởi bộ mã hóa entropi 32 và được ghi vào luồng bit video được nén 34. Các vùng ảnh đầu ra 38, chẳng hạn một phần của khung tham chiếu, có thể được tạo ra ở bộ mã hóa 2 bởi bộ lọc tách khối 36 bằng cách sử dụng tín hiệu được lọc trước, được phục hồi 22.

Fig.2 thể hiện sơ đồ khối của bộ giải mã video H.264/AVC 50 làm ví dụ. tín hiệu đầu vào 52, còn gọi là luồng bit, có thể được biểu thị để giải mã. Các ký hiệu thu được có thể là entropi được giải mã bởi bộ giải mã entropi 54, do đó tạo ra thông tin chuyển động 56 và các hệ số biến đổi được lượng tử hóa được định tỷ lệ 58. Thông tin chuyển động 56 có thể được kết hợp bởi bộ bù chuyển động 60 với một phần của khung tham chiếu 84 có thể lưu trữ trong bộ nhớ khung 64, và dự báo liên khung 68 có thể được tạo ra. Các hệ số biến đổi được lượng tử hóa được định tỷ lệ 58 có thể là các hệ số được định tỷ lệ, được lượng tử hóa ngược và được biến đổi ngược bởi bộ (biến đổi/định tỷ lệ/lượng tử hóa) ngược 62, do đó tạo ra tín hiệu dư được giải mã 70. Tín hiệu dư 70 có thể được cộng với tín hiệu dự báo 78: hoặc tín hiệu dự báo liên khung 68 hoặc tín hiệu dự báo nội khung 76. Tín hiệu dự báo nội khung 76 có thể được dự báo bởi bộ dự báo nội khung 74 từ thông tin được giải mã trước đó trong khung hiện hành 72. Tín hiệu được kết hợp 72 có thể được lọc bởi bộ lọc tách khối 80 và tín hiệu được lọc 82 có thể được ghi vào bộ nhớ khung 64.

Theo H.264/AVC, ảnh đầu vào được phân chia thành các khối macrô có kích thước không đổi, trong đó mỗi khối macrô bao phủ vùng ảnh hình chữ nhật có 16x16 mẫu của thành phần độ chói (luma) và 8x8 mẫu của mỗi trong số hai thành phần màu sắc. Theo các bộ mã hóa/giải mã và các tiêu chuẩn khác, đơn vị cơ bản, hoặc đơn vị mã hóa cơ bản, khác với khối macrô, chẳng hạn khối cây mã hóa, có thể được sử dụng. Quá trình xử lý giải mã theo tiêu chuẩn H.264/AVC được định rõ cho các đơn vị xử lý mà là các khối macrô. Bộ giải mã entropi 54 phân tích cú pháp các phần tử cú pháp của luồng bit video được nén 52 và phân kênh chúng. H.264/AVC định rõ hai phương pháp lựa chọn cho giải mã entropi: phương pháp

độ phức tạp thấp dựa vào việc sử dụng các tập thay đổi tính thích ứng ngữ cảnh của các mã độ dài thay đổi, được gọi là CAVLC, và thuật toán tăng yêu cầu tính toán của việc mã hóa số học nhị phân thích ứng trên cơ sở ngữ cảnh, gọi là CABAC. Trong cả hai phương pháp giải mã entropi, việc giải mã ký hiệu hiện hành có thể dựa vào các ký hiệu được giải mã một cách chính xác trước đó và các mô hình ngữ cảnh được cập nhật một cách thích ứng. Ngoài ra, thông tin dữ liệu khác nhau, chẳng hạn thông tin dữ liệu dự báo, thông tin dữ liệu dư và các mặt phẳng màu khác nhau, có thể được dồn kênh với nhau. Việc phân kênh có thể không cần được thực hiện cho đến khi các phần tử được giải mã entropi.

Sau khi giải mã entropi, khối macro có thể được phục hồi nhờ thu được: tín hiệu dư thông qua lượng tử hóa ngược và biến đổi ngược, và tín hiệu dự báo, hoặc tín hiệu dự báo nội khung hoặc tín hiệu dự báo liên khung. Sự suy giảm tạo khối có thể giảm bằng cách áp dụng bộ lọc tách khối cho mọi khối macro được giải mã. Không có quá trình xử lý nào được bắt đầu cho đến khi tín hiệu đầu vào được giải mã entropi, do đó làm cho việc giải mã entropi tắc nghẽn tiềm ẩn khi giải mã.

Tương tự, trong các bộ mã hóa/giải mã trong đó các biện pháp dự báo thay thế có thể được cho phép, chẳng hạn dự báo liên lớp theo H.264/AVC hoặc dự báo liên lớp theo các bộ mã hóa/giải mã khả biến tỷ lệ khác, giải mã entropi có thể yêu cầu trước khi xử lý tất cả ở bộ giải mã, do đó làm cho việc giải mã entropi tắc nghẽn tiềm ẩn.

Theo H.264/AVC, ảnh đầu vào bao gồm các khối macro có thể được phân chia thành một hoặc một số lát. Các giá trị của các mẫu trong vùng của ảnh mà lát này biểu thị có thể được giải mã một cách chính xác không sử dụng dữ liệu khác với các lát khác được tạo ra mà các ảnh tham chiếu được sử dụng ở bộ mã hóa và bộ giải mã đều giống nhau. Do đó, giải mã entropi và phục hồi khối macro cho lát này không phụ thuộc vào các lát khác. Cụ thể là, trạng thái mã hóa entropi được thiết lập lại ở trạng thái bắt đầu của từng lát. Dữ liệu trong các lát khác được đánh dấu là không sử dụng được khi xác định tính khả dụng lân cận cho cả giải mã entropi và phục hồi. Theo H.264/AVC, các lát có thể được giải mã entropi và được phục hồi theo cách song song. Không có sự dự báo nội và dự báo vectơ chuyển động được phép vượt qua ranh giới lát. Việc lọc tách khối có thể sử dụng thông tin

vượt qua các ranh giới lát.

Fig.3 thể hiện ảnh video làm ví dụ 90 bao gồm mười một khối macro theo chiều ngang và chín khối macro theo chiều dọc (chín khối macro làm ví dụ được ghi nhãn 91-99). Fig.3 thể hiện ba lát làm ví dụ: lát thứ nhất được ký hiệu là “SLICE #0” 100, lát thứ hai được ký hiệu là “SLICE #1” 101 và lát thứ ba được ký hiệu là “SLICE #2” 102. Bộ giải mã theo H.264/AVC có thể giải mã và phục hồi ba lát 100, 101, 102 theo cách song song. Khi bắt đầu quá trình xử lý giải mã/phục hồi đối với mỗi lát, các mô hình ngữ cảnh được khởi đầu hoặc được thiết lập lại và các khối macro trong các lát khác được ghi dấu là bất khả dụng cho cả giải mã entropi và phục hồi khối macro. Vì thế, đối với khối macro, chẳng hạn khối macro được ghi nhãn 93, trong “SLICE #1,” các khối macro (chẳng hạn các khối macro được ghi nhãn 91 và 92) trong “SLICE #0” có thể không được sử dụng để chọn mô hình ngữ cảnh hoặc phục hồi. Trái lại, đối với khối macro, chẳng hạn khối macro được ghi nhãn 95, trong “SLICE #1,” các khối macro khác (chẳng hạn các khối macro được ghi nhãn 93 và 94) trong “SLICE #1” có thể được sử dụng để chọn mô hình ngữ cảnh hoặc phục hồi. Do đó, giải mã entropi và phục hồi khối macro phải được xử lý một cách liên tiếp trong lát. Trừ khi các lát được xác định bằng cách sử dụng thứ tự khối macro linh hoạt được (FMO), các khối macro trong lát được xử lý theo thứ tự quét màn hình.

Thứ tự khối macro linh hoạt được xác định nhóm lát để thay đổi cách ảnh được phân chia thành các lát. Các khối macro trong nhóm lát được xác định bởi sơ đồ ánh xạ khối macro với nhóm lát, nó được báo hiệu bởi nội dung của tập tham số ảnh và thông tin bổ sung trong các mào đầu lát. Sơ đồ ánh xạ khối macro với nhóm lát gồm có số nhận dạng nhóm lát đối với từng khối macro trong ảnh. Số nhận dạng nhóm lát định rõ nhóm lát mà khối macro được kết hợp thuộc vào. Từng nhóm lát có thể được phân chia thành một hoặc nhiều lát, trong đó lát là chuỗi các khối macro trong cùng nhóm lát mà được xử lý theo thứ tự quét màn hình trong tập các khối macro của nhóm lát nhất định. Giải mã entropi và phục hồi khối macro phải được xử lý một cách liên tiếp trong lát.

Fig.4 là hình vẽ thể hiện sự phân phối khối macro làm ví dụ thành ba nhóm lát: nhóm lát thứ nhất được ký hiệu là “SLICE GROUP #0” 103, nhóm lát thứ hai

được ký hiệu là “SLICE GROUP #1” 104 và nhóm lát thứ ba được ký hiệu là “SLICE GROUP #2” 105. Các nhóm lát 103, 104, 105 này có thể được kết hợp với hai vùng tiền cảnh và vùng nền, một cách tương ứng, trong ảnh 90.

Một số phương án thực hiện sáng chế có thể bao gồm phân chia ảnh thành một hoặc nhiều lát phục hồi, trong đó lát phục hồi có thể đã tự mang các giá trị của các mẫu trong vùng của ảnh mà lát phục hồi biểu thị có thể được phục hồi một cách chính xác không cần sử dụng dữ liệu từ các lát phục hồi khác, được tạo ra sao cho các ảnh tham chiếu được sử dụng đều giống nhau ở bộ mã hóa và bộ giải mã. Tất cả các khối macro được phục hồi trong lát phục hồi có thể đã sẵn có khi xác định vùng lân cận để phục hồi.

Một số phương án thực hiện sáng chế có thể bao gồm việc phân chia lát phục hồi thành nhiều hơn một lát entropi, trong đó lát entropi có thể đã tự mang các giá trị ký hiệu trong vùng ảnh lát entropi biểu thị có thể được giải mã entropi một cách chính xác không sử dụng dữ liệu từ các lát entropi khác. Theo một số phương án, trạng thái mã hóa entropi có thể được thiết lập lại khi bắt đầu giải mã từng lát entropi. Theo một số phương án thực hiện sáng chế, dữ liệu trong các lát entropi khác có thể được ghi dấu bất khả dụng khi xác định tính khả dụng lân cận đối với giải mã entropi. Theo một số phương án thực hiện sáng chế, các khối macro trong các lát entropi khác không thể được sử dụng trong khi mô hình ngữ cảnh của khối hiện hành. Theo một số phương án thực hiện sáng chế, các mô hình ngữ cảnh có thể được cập nhật chỉ trong lát entropi. Theo các phương án thực hiện này của sáng chế, mỗi bộ giải mã entropi được kết hợp với lát entropi có thể duy trì tập các mô hình ngữ cảnh của nó.

Nội dung của tài liệu Lĩnh vực chuẩn hóa viễn thông thuộc Tổ chức Viễn thông quốc tế, nhóm nghiên cứu 16, bài viết 405 (ITU Telecommunication Standardization Sector, Study Group 16 – Contribution 405) với tựa đề “các lát entropi dùng cho giải mã entropi song song-Entropy slices for parallel entropy decoding”, tháng 4 năm 2008, được trích dẫn kết hợp trong phần mô tả dưới đây nhằm tham khảo.

Theo một số phương án thực hiện sáng chế có thể bao gồm quá trình xử lý mã hóa/giải mã CABAC. Quá trình xử lý mã hóa CABAC bao gồm bốn bước cơ

bản sau: nhị phân hóa; chọn mô hình ngữ cảnh; mã hóa số học nhị phân; và cập nhật xác suất.

Nhị phân hóa: ký hiệu có giá trị chưa nhị phân hóa (chẳng hạn hệ số biến đổi, vectơ chuyển động, hoặc dữ liệu mã hóa khác) được biến đổi thành mã nhị phân, còn được gọi là chuỗi bin hoặc ký hiệu được nhị phân hóa. Khi phần tử cú pháp có giá trị nhị phân được định trước, bước bắt đầu nhị phân hóa có thể được phân nhánh. Phần tử cú pháp có giá trị nhị phân hoặc phần tử ký hiệu được nhị phân hóa có thể được gọi là bin.

Đối với từng bin, phần sau đây có thể được thực hiện:

Chọn mô hình ngữ cảnh: mô hình ngữ cảnh là mô hình xác suất đối với một hoặc nhiều bin. Mô hình ngữ cảnh bao gồm, đối với mỗi bin, xác suất của bin sẽ là “1” hoặc “0”. Mô hình này có thể được chọn để chọn các mô hình khả dụng tùy theo thống kê của các ký hiệu dữ liệu được mã hóa trước đó, thông thường, dựa vào các ký hiệu lân cận bên trái và phía trên, nếu nó khả dụng.

Mã hóa số học nhị phân: bộ mã hóa số học mã hóa từng bin theo mô hình xác suất được chọn và dựa vào sự chia nhỏ thêm thời khoảng theo cách đệ quy.

Cập nhật xác suất: mô hình ngữ cảnh được chọn được cập nhật dựa vào giá trị được mã hóa thực sự.

Việc thích ứng ngữ cảnh có thể gọi là quá trình xử lý chọn, dựa vào các giá trị ký hiệu lân cận, trạng thái mô hình ngữ cảnh, còn gọi là trạng thái, được kết hợp với bin và cập nhật sự phân bố mô hình xác suất được định rõ cho các ký hiệu định trước. Vị trí của các ký hiệu lân cận có thể được xác định theo bảng mẫu ngữ cảnh.

Theo một số phương án thực hiện sáng chế bao gồm mã hóa/giải mã CABAC, lúc bắt đầu giải mã lát entropi, tất cả các mô hình ngữ cảnh có thể được khởi tạo hoặc được thiết lập lại cho các mô hình định trước.

Một số phương án thực hiện sáng chế có thể được mô tả có dựa vào Fig.5. Fig.5 thể hiện khung video làm ví dụ 110 bao gồm mười một khối macrô theo chiều ngang và chín khối macrô theo chiều dọc (chín khối macrô làm ví dụ được ghi nhãn 115-123). Fig.5 thể hiện ba lát phục hồi làm ví dụ: lát phục hồi thứ nhất được ký hiệu là “R_SLICE #0” 111, lát phục hồi thứ hai được ký hiệu là

“R_SLICE #1” 112 và lát phục hồi thứ ba được ký hiệu là “R_SLICE #2” 113. Fig.5 còn thể hiện sự phân chia lát phục hồi thứ hai “R_SLICE #1” 112 thành ba lát entropi: lát entropi thứ nhất được ký hiệu là “E_SLICE #0” được thể hiện trong phần gạch chéo 114, lát entropi thứ hai được ký hiệu là “E_SLICE #1” được thể hiện trong phần gạch dọc 115 và lát entropi thứ ba được ký hiệu là “E_SLICE #2” được thể hiện trong phần gạch ô vuông 116. Mỗi lát entropi 114, 115, 116 có thể được giải mã entropi theo cách song song.

Theo một số phương án thực hiện sáng chế, chỉ dữ liệu từ các khối macrô trong lát entropi có thể khả dụng để chọn mô hình ngữ cảnh trong khi giải mã entropi của lát entropi. Tất cả các khối macrô khác có thể được ghi dấu là bất khả dụng. Đối với việc phân chia làm ví dụ này, các khối macrô được ghi nhãn 117 và 118 là bất khả dụng để chọn mô hình ngữ cảnh khi các ký hiệu giải mã tương ứng với vùng của khối macrô được ghi nhãn 119 bởi vì các khối macrô được ghi nhãn 117 và 118 nằm ngoài lát entropi chứa khối macrô 119. Tuy nhiên, các khối macrô 117, 118 này khả dụng khi khối macrô 119 được phục hồi.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể xác định xem cần hay không phân chia lát phục hồi thành các lát entropi, và bộ mã hóa có thể báo hiệu quyết định trong luồng bit. Theo một số phương án thực hiện sáng chế, tín hiệu báo hiệu này có thể bao gồm cờ lát entropi, dưới đây gọi là “entropy_slice_flag” theo một số phương án thực hiện sáng chế.

Bộ giải mã theo một số phương án thực hiện có thể được mô tả có dựa vào Fig.6. Theo các phương án này, cờ lát entropi có thể được qua quá trình kiểm tra ở bước 130, và nếu cờ lát entropi xác định rằng không có ở bước 132 thì các lát entropi được kết hợp với ảnh, hoặc lát phục hồi, sau đó phần mào đầu có thể được phân tích cú pháp ở bước 134 là mào đầu lát ổn định. Trạng thái bộ giải mã entropi có thể được thiết lập lại là trạng thái thiết lập lại ở bước 136, và thông tin lân cận để giải mã entropi và phục hồi có thể được xác định ở bước 138. Sau đó, dữ liệu lát có thể được giải mã entropi ở bước 140, và lát này có thể được phục hồi ở bước 142. Nếu cờ lát entropi chỉ báo có ở bước 146 thì các lát entropi được kết hợp với ảnh, hoặc lát phục hồi, sau đó mào đầu có thể được phân tích cú pháp ở bước 148 là mào đầu lát entropi. Trạng thái bộ giải mã entropi có thể được thiết lập lại ở

bước 150, thông tin lân cận để giải mã entropi có thể được xác định 152 và dữ liệu lát entropi có thể được giải mã entropi ở bước 154. Thông tin lân cận để phục hồi sau đó được xác định ở bước 156, và lát này có thể được phục hồi ở bước 142. Sau đó, phục hồi lát ở bước 142, lát tiếp theo, hoặc ảnh, có thể được kiểm tra ở bước 158.

Một số bộ giải mã theo các phương án thực hiện khác của sáng chế sẽ được mô tả có dựa vào Fig.7. Theo các phương án này, bộ giải mã có thể giải mã song song và có thể xác định mức độ song song của nó, chẳng hạn bộ giải mã bao gồm khả năng giải mã N lát entropi theo cách song song. Bộ giải mã có thể nhận dạng ở bước 170 N lát entropi. Theo một số phương án thực hiện sáng chế, nếu không nhiều hơn N lát entropi khả dụng trong ảnh hiện hành, hoặc lát phục hồi, bộ giải mã có thể giải mã các lát entropi từ các ảnh tiếp theo, hoặc các lát phục hồi, nếu chúng là khả dụng. Theo các phương án thực hiện khác, bộ giải mã có thể đợi cho đến khi ảnh hiện hành, hoặc lát phục hồi, được xử lý hoàn toàn trước khi giải mã các phần của ảnh tiếp theo, hoặc lát phục hồi tiếp theo. Sau khi nhận dạng ở bước 170 đến N lát entropi, mỗi trong số các lát entropi được xác định có thể được giải mã entropi một cách độc lập. Lát entropi thứ nhất có thể được giải mã ở các bước từ bước 172 đến bước 176. Việc giải mã ở bước 172-176 của lát entropi thứ nhất có thể bao gồm việc thiết lập lại trạng thái bộ giải mã ở bước 172. Theo một số phương án thực hiện bao gồm giải mã entropi CABAC, trạng thái CABAC có thể được thiết lập lại. Thông tin lân cận để giải mã entropi của lát entropi thứ nhất có thể được xác định ở bước 174, và dữ liệu lát entropi thứ nhất có thể được giải mã ở bước 176. Đối với mỗi trong số N lát entropi, các bước này có thể được thực hiện (ở bước 178-182 đối với lát entropi thứ N). Theo một số phương án thực hiện sáng chế, bộ giải mã có thể phục hồi 184 các lát entropi khi tất cả các lát entropi được giải mã entropi. Theo các phương án thực hiện khác của sáng chế, bộ giải mã có thể bắt đầu phục hồi ở bước 184 sau khi một hoặc nhiều lát entropi được giải mã.

Theo một số phương án thực hiện sáng chế, khi có nhiều hơn N lát entropi, việc giải mã có thể bắt đầu giải mã entropi lát entropi tiếp theo đến khi hoàn thành giải mã entropi của lát entropi này. Vì thế, khi quá trình xử lý giải mã kết thúc việc giải mã entropi lát entropi có độ phức tạp nhỏ, quá trình xử lý giải mã có thể bắt

đầu việc giải mã bổ sung các lát entropi mà không đợi các quá trình xử lý khác kết thúc việc giải mã của chúng.

Theo một số phương án thực hiện sáng chế, sáng chế có thể kết hợp phương pháp hoặc tiêu chuẩn hiện có, lát entropi có thể được dùng chung hầu hết các thuộc tính lát của lát ổn định theo tiêu chuẩn hoặc phương pháp này. Do đó, lát entropi có thể đòi hỏi mào đầu nhỏ. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi có thể cho phép bộ giải mã nhận dạng phần bắt đầu của lát entropi và bắt đầu giải mã entropi. Theo một số phương án, ở phần bắt đầu của ảnh, hoặc lát phục hồi, mào đầu lát entropi có thể là mào đầu ổn định, hoặc mào đầu lát phục hồi.

Theo một số phương án thực hiện sáng chế bao gồm bộ mã hóa/giải mã H.264/AVC, lát entropi có thể được báo hiệu bằng cách bổ sung bit mới, “entropy_slice_flag” vào mào đầu lát hiện có. Bảng 1 liệt kê cú pháp đối với mào đầu lát entropi theo các phương án thực hiện sáng chế, trong đó C chỉ báo loại (Category) và phần tử mô tả (Descriptor) u(1), ue(v) biểu thị một số phương pháp mã hóa độ dài không đổi hoặc độ dài biến đổi. Các phương án thực hiện sáng chế bao gồm “entropy_slice_flag” có thể cải thiện được hiệu suất mã hóa.

“first_mb_in_slice” định rõ địa chỉ của khối macro thứ nhất trong lát entropi được kết hợp với mào đầu lát entropi. Theo một số phương án, lát entropi có thể bao gồm chuỗi các khối macro.

“cabac_init_idc” định rõ chỉ số để xác định bảng khởi tạo được sử dụng trong quá trình xử lý khởi tạo đối với chế độ ngữ cảnh.

slice_header() {	C	Phần tử mô tả
entropy_slice_flag	2	u(1)
(entropy_slice_flag) {		
first_mb_in_slice	2	ue(v)
(entropy_coding_mode_flag && slice_type != I && slice_type != SI)		
cabac_init_idc	2	ue(v)

}		
}		
else {		
mào đầu lát ổn định ...		
}		
}		

Bảng 1: Bảng cú pháp làm ví dụ đối với mào đầu lát entropi

Theo một số phương án thực hiện sáng chế, lát entropi có thể được gán kiểu đơn vị (NAL) lớp rút gọn mạng khác nhau từ các lát ổn định. Theo các phương án này, bộ giải mã có thể phân biệt giữa các lát ổn định và các lát entropi dựa vào kiểu đơn vị NAL. Theo các phương án này, trường bit “entropy_slice_flag” là không cần thiết.

Theo một số phương án thực hiện sáng chế, trường bit “entropy_slice_flag” có thể không được truyền theo tất cả các biên dạng. Theo một số phương án thực hiện sáng chế, trường bit “entropy_slice_flag” có thể không được truyền biên dạng đường cơ sở, mà trường bit “entropy_slice_flag” có thể được truyền theo các biên dạng cao hơn như là chính, mở rộng hoặc biên dạng chuyên nghiệp. Theo một số phương án thực hiện sáng chế, trường bit “entropy_slice_flag” có thể chỉ được truyền theo các luồng bit được kết hợp với các đặc trưng lớn hơn các giá trị đặc trưng cố định. Các đặc trưng làm ví dụ có thể bao gồm độ phân giải không gian, tốc độ khung, độ sâu bit, tốc độ bit và các đặc trưng luồng bit khác. Theo một số phương án thực hiện sáng chế, trường bit “entropy_slice_flag” có thể chỉ được truyền trong các luồng bit được kết hợp với độ phân giải không gian lớn hơn 1920x1080 được xen kẽ. Theo một số phương án thực hiện sáng chế, trường bit “entropy_slice_flag” có thể chỉ được truyền theo các luồng bit được kết hợp với độ phân giải không gian lớn hơn 1920x1080 liên tục. Theo một số phương án thực hiện sáng chế, nếu trường bit “entropy_slice_flag” không được truyền, giá trị mặc định có thể được sử dụng.

Theo một số phương án thực hiện sáng chế, lát entropi có thể được tạo cấu hình bằng cách sử dụng luôn phiên dồn kênh dữ liệu. Theo một số phương án thực hiện sáng chế, nhóm các ký hiệu có trong lát entropi có thể được dồn kênh ở mức

độ khối macrô. Theo các phương án thực hiện khác của sáng chế, nhóm các ký hiệu có trong lát entropi có thể được dồn kênh ở mức độ ảnh. Theo các phương án thực hiện khác của sáng chế, nhóm các ký hiệu có trong lát entropi có thể được dồn kênh theo kiểu dữ liệu. Theo các phương án thực hiện khác của sáng chế, nhóm các ký hiệu có trong lát entropi có thể được dồn kênh theo cách kết hợp các phương pháp nêu trên.

Một số phương án thực hiện sáng chế bao gồm tạo cấu hình lát entropi dựa vào việc dồn kênh mức độ ảnh sẽ được mô tả có dựa vào Fig.8 và Fig.9. Theo một số phương án thực hiện sáng chế được thể hiện trên Fig.8, dữ liệu dự báo 190 và dữ liệu dư 192 có thể được mã hóa entropi bởi bộ mã hóa dự báo 194 và bộ mã hóa dư 196 một cách riêng biệt và dữ liệu dự báo được mã hóa và dữ liệu dư được mã hóa có thể được dồn kênh bởi bộ dồn kênh mức độ ảnh 198 ở mức độ ảnh. Theo một số phương án thực hiện sáng chế, dữ liệu dự báo cho ảnh 190 có thể được kết hợp với lát entropi thứ nhất, và dữ liệu dư cho ảnh 192 có thể được kết hợp với lát entropi thứ hai. Dữ liệu dự báo được mã hóa và dữ liệu entropi được mã hóa có thể được giải mã theo cách song song. Theo một số phương án thực hiện sáng chế, mỗi phần chia bao gồm dữ liệu dự báo hoặc dữ liệu dư có thể được phân chia thành các lát entropi có thể được giải mã theo cách song song.

Theo một số phương án thực hiện sáng chế, như được thể hiện trên Fig.9, phần dư của mỗi mặt phẳng màu, chẳng hạn phần dư độ chói 200 và hai dữ liệu màu sắc 202, 204, có thể được mã hóa entropi bởi bộ mã hóa Y 206, bộ mã hóa U 208, và bộ mã hóa V 210 một cách riêng biệt và các dữ liệu dư được mã hóa entropi có thể được dồn kênh bởi bộ dồn kênh mức độ ảnh 212 ở mức độ ảnh. Theo một số phương án thực hiện sáng chế, phần dư độ chói cho ảnh 200 có thể được kết hợp với lát entropi thứ nhất, dữ liệu dư màu sắc thứ nhất cho ảnh 202 có thể được kết hợp với lát entropi thứ hai, và phần dư thứ hai cho ảnh 204 có thể được kết hợp với lát entropi thứ ba. Dữ liệu dư được mã hóa đối với ba mặt phẳng màu có thể được giải mã theo cách song song. Theo một số phương án thực hiện sáng chế, mỗi phần chia bao gồm dữ liệu dư mặt phẳng màu có thể được phân chia thành các lát entropi có thể được giải mã theo cách song song. Theo một số phương án thực hiện sáng chế, phần dư độ chói 200 có thể có tương đối nhiều lát

entropi hơn so với các dữ liệu dư màu sắc 202, 204.

Theo một số phương án thực hiện sáng chế, luồng bit video được nén có thể được mã hóa biến đổi để bao gồm các lát entropi, do đó cho phép giải mã entropi song song là phương án ưu tiên thực hiện của sáng chế, như được mô tả trên đây. Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.10. Luồng bit đầu vào không có các lát entropi có thể được xử lý từng ảnh một, như được thể hiện trên Fig.10. Theo các phương án thực hiện này của sáng chế, ảnh từ luồng bit đầu vào có thể được giải mã entropi ở bước 220. Dữ liệu mà đã được mã hóa, chẳng hạn dữ liệu chế độ, thông tin chuyển động, thông tin dư và dữ liệu khác, có thể thu được. Các lát entropi có thể được phục hồi ở bước 222 một phần từ dữ liệu này. Mào đầu lát entropi tương ứng với lát entropi có thể được chèn ở bước 224 trong luồng bit mới. Trạng thái bộ mã hóa có thể được thiết lập lại và thông tin lân cận được xác định ở bước 226. Lát entropi có thể được mã hóa entropi ở bước 228 và được ghi vào luồng bit mới. Nếu có dữ liệu ảnh chưa được xử lý ở bước 232 các lát entropi được tạo cấu hình, thì lát entropi khác có thể được tạo cấu hình ở bước 222, và quá trình xử lý ở bước 224-230 có thể được tiếp tục cho đến khi tất cả dữ liệu ảnh đã được xử lý ở bước 234 bởi các lát entropi được tạo cấu hình, và sau đó ảnh tiếp theo có thể được xử lý.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi trong đó kích thước của mỗi lát entropi có thể nhỏ hơn, hoặc có thể không vượt quá, số lượng bin nhất định. Theo một số phương án trong đó bộ mã hóa có thể hạn chế kích thước của mỗi lát entropi, số lượng bin tối đa có thể được báo hiệu trong luồng bit. Theo một số phương án khác trong đó bộ mã hóa có thể giới hạn kích thước của mỗi lát entropi, số lượng bin tối đa có thể được xác định bởi biên dạng và điểm mức độ thích hợp của bộ mã hóa. Chẳng hạn, Phụ lục của đặc tả mã hóa video H.264/AVC có thể được mở rộng để bao gồm định nghĩa về số lượng bin tối đa được cho phép trong lát entropi.

Theo một số phương án thực hiện sáng chế, số lượng bin tối đa được cho phép trong lát entropi có thể được chỉ báo cho từng điểm mức độ thích hợp của bộ mã hóa theo bảng, chẳng hạn như được thể hiện trong bảng 2, trong đó $M_{m,n}$ biểu thị số lượng bin tối đa được cho phép trong lát entropi đối với điểm mức độ m.n

thích hợp.

Mức độ	Số lượng bin tối đa trên lát entropi
1.1	M1.1
1.2	M1.2
:	:
m.n	Mm.n
:	:
5.1	M5.1

Bảng 2: Số lượng bin tối đa trên lát entropi đối với từng mức độ

Số lượng bin tối đa được cho phép làm ví dụ trong lát entropi là $M_{1.1} = 1.000$ bin, $M_{1.2} = 2.000$ bin, ..., và $M_{5.1} = 40.000$ bin. Số lượng bin tối đa được cho phép khác làm ví dụ trong lát entropi là $M_{1.1} = 2.500$ bin, $M_{1.2} = 4.200$ bin, ..., và $M_{5.1} = 150.000$ bin.

Theo một số phương án, tập số lượng bin tối đa được cho phép trong lát entropi có thể được xác định đối với tất cả các mức độ dựa vào tốc độ bit, kích thước ảnh, số lượng các khối macro và các tham số mã hóa khác. Theo một số phương án thực hiện sáng chế số lượng bin tối đa được cho phép trong lát entropi có thể được thiết lập là cùng một số lượng cho tất cả các mức độ. Các giá trị làm ví dụ là 38.000 bin và 120.000 bin.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể xác định số lượng bin của trường hợp xấu nhất được kết hợp với khối macro, và bộ mã hóa có thể ghi bin được kết hợp với:

$$\frac{ESLICE_MaxNumberBins}{BinsPerMB}$$

khối macro cho từng lát entropi, trong đó *ESLICE_MaxNumberBins* có thể biểu thị số lượng bin tối đa được cho phép trong lát entropi và *BinsPerMB* có thể biểu thị số lượng bin của trường hợp xấu nhất được kết hợp với khối macro. Theo một số phương án, các khối macro có thể được chọn theo thứ tự quét mảnh. Theo các phương án thực hiện khác, các khối macro có thể được chọn theo thứ tự được định trước, khác. Theo một số phương án, số lượng bin của trường hợp xấu nhất

được kết hợp với khối macrô có thể là một số không đổi. Theo các phương án thực hiện khác, bộ mã hóa có thể cập nhật số lượng bin của trường hợp xấu nhất dựa vào các giá trị đo kích thước của các khối macrô được xử lý trước đó.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.11. Theo các phương án này, bộ mã hóa có thể, đối với lát phục hồi, phân chia lát phục hồi thành các lát entropi trong đó không có lát entropi có thể có kích thước lớn hơn so với số lượng bin định trước. Bộ mã hóa có thể khởi tạo ở bước 240 cho bộ đếm không được kết hợp với số lượng các bin trong lát entropi hiện hành. Giá trị của bộ đếm có thể được biểu thị là A nhằm mục đích minh họa trong phần mô tả dưới đây về các phương án thực hiện sáng chế, có dựa vào Fig.11. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 242. Khối macrô tiếp theo có thể được xác định theo thứ tự xử lý khối macrô định trước. Theo một số phương án, thứ tự xử lý khối macrô có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macrô có thể được biến đổi ở bước 244 thành chuỗi các bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Số lượng các bin được kết hợp với khối macrô có thể được xác định ở bước 246. Số lượng các bin được kết hợp với khối macrô có thể bao gồm các bin trong các chuỗi bin được kết hợp với các phần tử cú pháp phi nhị phân ngoài các phần tử cú pháp nhị phân, và số lượng các bin được kết hợp với khối macrô có thể được biểu thị là num nhằm mục đích minh họa trong phần mô tả dưới đây của các phương án thực hiện sáng chế, có dựa vào Fig.11.

Nếu số lượng các bin được kết hợp với khối macrô có thể được bổ sung ở bước 248 với số lượng các bin đã được tích lũy kết hợp với lát entropi hiện hành không ở bước 249 vượt quá số lượng bin tối đa được cho phép đối với lát entropi, thì số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 250 để bao gồm các bin được kết hợp với khối macrô, và các bin được kết hợp với khối macrô có thể được ghi ở bước 252, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 242, và quá trình xử lý phân chia có thể tiếp tục.

Nếu ở bước 248 tổng số lượng các bin được kết hợp với khối macrô và số

lượng các bin được tích lũy có sẵn được kết hợp với lát entropi hiện hành ở bước 253 vượt quá số lượng bin tối đa được cho phép đối với lát entropi, thì bộ mã hóa có thể bắt đầu ở bước 254 lát entropi mới được kết hợp với lát phục hồi hiện hành và có thể kết thúc lát entropi hiện hành. Tiếp theo, bộ đếm được kết hợp với số lượng các bin trong lát entropi mới, hiện hành có thể được khởi tạo ở bước 256 về không. Số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 250 để bao gồm các bin được kết hợp với khối macrô, và các bin được kết hợp với khối macrô có thể được ghi ở bước 252, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 242, và quá trình xử lý phân chia có thể được tiếp tục.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.12. Theo các phương án này, bộ mã hóa có thể, đối với lát phục hồi, phân chia lát phục hồi thành các lát entropi trong đó không có lát entropi nào có thể có kích thước lớn hơn so với số lượng bin tối đa đã định. Theo các phương án này, bộ mã hóa có thể kết hợp các phần tử cú pháp khối macrô với lát entropi cho đến khi kích thước của lát entropi đạt đến ngưỡng được kết hợp với số lượng bin tối đa đã định được cho phép trong lát entropi. Theo một số phương án, ngưỡng có thể là phần trăm của số lượng bin tối đa được cho phép trong lát entropi. Theo một phương án thực hiện làm ví dụ, ngưỡng có thể 90% của số lượng bin tối đa được cho phép trong lát entropi, giả sử rằng số lượng bin lớn nhất mong muốn trong khối macrô là nhỏ hơn 10% số lượng bin tối đa. Theo phương án thực hiện khác làm ví dụ, ngưỡng có thể là phần trăm của số lượng bin tối đa được cho phép trong lát entropi trong đó phần trăm có thể được dựa vào số lượng bin lớn nhất mong muốn trong khối macrô. Theo các phương án này, một khi kích thước của lát entropi vượt quá kích thước ngưỡng, thì lát entropi khác có thể được tạo ra. Kích thước ngưỡng có thể được chọn để đảm bảo rằng lát entropi không vượt quá số lượng bin tối đa được cho phép trong lát entropi. Theo một số phương án, kích thước ngưỡng có thể là hàm của số lượng bin tối đa được cho phép trong lát entropi và cần đánh giá số lượng bin tối đa đối với khối macrô.

Bộ mã hóa có thể khởi tạo ở bước 270 về không bộ đếm được kết hợp với số

lượng các bin trong lát entropi hiện hành. Giá trị của bộ đếm có thể được biểu thị là A nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.12. Các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 272. Khối macro tiếp theo có thể được xác định theo thứ tự xử lý khối macro định trước. Theo một số phương án, thứ tự xử lý khối macro có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macro có thể được biến đổi ở bước 274 thành chuỗi bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Các bin được kết hợp với khối macro có thể được ghi ở bước 276, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. Số lượng các bin được kết hợp với khối macro có thể được xác định ở bước 278, và số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 280 để bao gồm các bin được kết hợp với khối macro. Nếu ở bước 282 số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành lớn hơn so với ngưỡng ở bước 284, sau đây ngưỡng này được biểu thị là $TH(MaxNumBins)$, dựa vào số lượng bin tối đa được cho phép trong lát entropi, thì bộ mã hóa có thể bắt đầu ở bước 286 lát entropi mới và có thể kết thúc lát entropi hiện hành. Tiếp theo, bộ mã hóa có thể khởi tạo ở bước 288 về không bộ đếm được kết hợp với số lượng các bin trong lát entropi mới, hiện hành. Các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 272, và quá trình xử lý phân chia có thể được tiếp tục. Nếu số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành không lớn hơn ở bước 283 so với ngưỡng dựa vào số lượng bin tối đa được cho phép trong lát entropi, thì các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 272, và quá trình xử lý phân chia có thể được tiếp tục.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể kết thúc lát phục hồi hiện hành và bắt đầu lát phục hồi mới khi số lượng xác định trước của các khối macro đã được gán cho lát phục hồi hiện hành.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.13. Theo các phương án này, bộ mã hóa có thể kết thúc lát phục hồi hiện hành và bắt đầu lát phục hồi mới khi số lượng xác định trước của các khối macro đã được gán cho lát phục hồi hiện hành. Bộ mã hóa có thể khởi tạo ở bước 300 về không bộ đếm được kết hợp với số lượng các khối macro trong lát phục hồi hiện hành. Giá trị của bộ

đếm được biểu thị là *AMB* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.13. Bộ mã hóa có thể khởi tạo ở bước 310 về không bộ đếm được kết hợp với số lượng các bin trong lát entropi hiện hành. Giá trị của bộ đếm được ký hiệu là *ABin* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.13. Nếu ở bước 312 giá trị đếm của bộ đếm được kết hợp với số lượng các khối macro trong lát phục hồi hiện hành ở bước 331 không nhỏ hơn so với số lượng các khối macro tối đa định trước được cho phép trong lát phục hồi, thì lát entropi mới có thể được bắt đầu ở bước 332 và lát phục hồi mới có thể được bắt đầu ở bước 334, kết thúc lát phục hồi hiện hành và lát entropi hiện hành. Số lượng các khối macro tối đa được cho phép trong lát phục hồi được ký hiệu là *MaxMBperRSlice* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.13.

Nếu giá trị bộ đếm của bộ đếm được kết hợp với số lượng các khối macro trong lát phục hồi hiện hành nhỏ hơn ở bước 313 so với số lượng các khối macro tối đa định trước được cho phép trong lát phục hồi, thì các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 314. Khối macro tiếp theo có thể được xác định theo thứ tự xử lý khối macro định trước. Theo một số phương án, thứ tự xử lý khối macro có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macro có thể được biến đổi ở bước 316 thành chuỗi bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Số lượng các bin được kết hợp với khối macro có thể được xác định ở bước 318. Số lượng các bin được kết hợp với khối macro có thể bao gồm các bin trong các chuỗi bin được kết hợp với các phần tử cú pháp phi nhị phân ngoài các phần tử cú pháp nhị phân, và số lượng các bin được kết hợp với khối macro được ký hiệu là *num* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.13.

Nếu số lượng các bin được kết hợp với khối macro có thể được bổ sung ở bước 320 vào số lượng các bin được tích lũy đã có được kết hợp với lát entropi hiện hành không ở bước 321 vượt quá số lượng bin tối đa được cho phép đối với lát entropi, thì số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 322 để bao gồm các bin được kết hợp với khối macro, các bin được kết hợp với khối macro có thể được ghi ở bước 324, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành, và số

lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được tăng ở bước 326. Số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được so sánh ở bước 312 với số lượng các khối macrô tối đa định trước được cho phép trong lát phục hồi, và quá trình xử lý phân chia có thể được tiếp tục.

Nếu ở bước 320 tổng số lượng các bin được kết hợp với khối macrô và số lượng các bin được tích lũy đã có được kết hợp với lát entropi hiện hành vượt quá ở bước 327 số lượng bin tối đa được cho phép đối với lát entropi, thì bộ mã hóa có thể bắt đầu ở bước 328 lát entropi mới, hiện hành được kết hợp với lát phục hồi hiện hành, và bộ đếm được kết hợp với số lượng các bin trong lát entropi hiện hành có thể được khởi tạo ở bước 330 về không. Số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 322 để bao gồm các bin được kết hợp với khối macrô, các bin được kết hợp với khối macrô có thể được ghi ở bước 324, bởi bộ mã hóa entropi, với luồng bit và được kết hợp với lát entropi hiện hành, và số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được tăng ở bước 326. Số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được so sánh ở bước 312 với số lượng các khối macrô tối đa định trước được cho phép trong lát phục hồi, và quá trình xử lý phân chia có thể được tiếp tục.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.14. Theo các phương án này, bộ mã hóa có thể bắt đầu lát phục hồi mới khi số lượng xác định trước của các khối macrô được gán cho lát phục hồi hiện hành. Theo các phương án này, bộ mã hóa có thể kết hợp các phần tử cú pháp khối macrô với lát entropi cho đến khi kích thước của lát entropi đạt tới ngưỡng được kết hợp với số lượng bin tối đa đã định được cho phép trong lát entropi. Theo một số phương án, ngưỡng có thể là phần trăm của số lượng bin tối đa được cho phép trong lát entropi. Theo một phương án thực hiện làm ví dụ, ngưỡng có thể là 90% của số lượng bin tối đa được cho phép trong lát entropi, giả sử rằng số lượng bin lớn nhất mong muốn trong khối macrô nhỏ hơn 10% của số lượng bin tối đa. Theo phương án thực hiện khác làm ví dụ, ngưỡng có thể là phần trăm của số lượng bin tối đa được cho phép trong lát entropi trong đó phần trăm có thể được dựa vào số lượng bin lớn nhất mong muốn trong khối macrô. Theo các phương án này, một khi kích

thước của lát entropi vượt quá kích thước ngưỡng, thì lát entropi khác có thể được tạo ra. Kích thước ngưỡng có thể được chọn để đảm bảo rằng lát entropi không vượt quá số lượng bin tối đa được cho phép trong lát entropi. Theo một số phương án, kích thước ngưỡng có thể là hàm của số lượng bin tối đa được cho phép trong lát entropi và cần đánh giá số lượng bin tối đa đối với khối macro.

Bộ mã hóa có thể khởi tạo ở bước 350 về không bộ đếm được kết hợp với số lượng các khối macro trong lát phục hồi hiện hành. giá trị của bộ đếm được ký hiệu là *AMB* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.14. Bộ mã hóa có thể khởi tạo ở bước 352 về không bộ đếm được kết hợp với số lượng các bin trong lát entropi hiện hành. Giá trị của bộ đếm được ký hiệu là *ABin* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.14. Nếu ở bước 354 giá trị bộ đếm của bộ đếm được kết hợp với số lượng các khối macro trong lát phục hồi hiện hành không nhỏ hơn ở bước 373 so với số lượng các khối macro tối đa định trước được cho phép trong lát phục hồi, thì lát entropi mới có thể được bắt đầu ở bước 374, và lát phục hồi mới có thể được bắt đầu ở bước 376. Số lượng các khối macro tối đa được cho phép trong lát phục hồi được ký hiệu là *MaxMBperRSlice* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.14.

Nếu giá trị bộ đếm của bộ đếm được kết hợp với số lượng các khối macro trong lát phục hồi hiện hành nhỏ hơn ở bước 355 so với số lượng các khối macro tối đa định trước được cho phép trong lát phục hồi, thì các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 356. Khối macro tiếp theo có thể được xác định theo thứ tự xử lý khối macro định trước. Theo một số phương án, thứ tự xử lý khối macro có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macro có thể được biến đổi ở bước 358 thành chuỗi các bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Các bin được kết hợp với khối macro có thể được ghi ở bước 360, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. Số lượng các bin được kết hợp với khối macro có thể được xác định ở bước 362, và số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 364 để bao gồm các bin được kết hợp với khối macro. Nếu ở bước 366 số lượng các bin được tích lũy

được kết hợp với lát entropi hiện hành lớn hơn ở bước 369 so với ngưỡng, ngưỡng này được ký hiệu là $TH(MaxNumBins)$, dựa vào số lượng bin tối đa được cho phép trong lát entropi, thì bộ mã hóa có thể bắt đầu ở bước 370 lát entropi mới, và khởi tạo ở bước 372 về không bộ đếm được kết hợp với số lượng các bin trong lát entropi hiện hành. Số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được tăng ở bước 368. Số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được so sánh 354 với số lượng các khối macrô tối đa định trước được cho phép trong lát phục hồi, và quá trình xử lý phân chia có thể được tiếp tục. Nếu số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành không lớn hơn ở bước 367 so với ngưỡng dựa vào số lượng bin tối đa được cho phép trong lát entropi, thì số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được tăng ở bước 368, và số lượng các khối macrô được kết hợp với lát phục hồi hiện hành có thể được so sánh ở bước 354 với số lượng các khối macrô tối đa định trước được cho phép trong lát phục hồi, và quá trình xử lý phân chia có thể được tiếp tục.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó mỗi lát entropi có thể được kết hợp với không lớn hơn số lượng các bit định trước.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.15. Theo các phương án này, bộ mã hóa có thể, đối với lát phục hồi, phân chia lát phục hồi thành các lát entropi trong đó không có lát entropi có thể có kích thước lớn hơn so với số lượng xác định trước của các bit. Bộ mã hóa có thể khởi tạo ở bước 400 về không bộ đếm được kết hợp với số lượng các bit trong lát entropi hiện hành. Giá trị của bộ đếm được ký hiệu là A nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.15. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 402. Khối macrô tiếp theo có thể được xác định theo thứ tự xử lý khối macrô định trước. Theo một số phương án, thứ tự xử lý khối macrô có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macrô có thể được biến đổi ở bước 404 thành chuỗi các bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Các bin, các phần tử phi nhị phân được biến đổi và các phần tử nhị phân, được kết hợp với khối macrô có thể được biểu thị cho

bộ mã hóa entropi, và các bin có thể được mã hóa entropi ở bước 406. Số lượng các bit được kết hợp với khối macrô có thể được xác định ở bước 408. Số lượng các bit được kết hợp với khối macrô được ký hiệu là *num* nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.15.

Nếu số lượng các bit được kết hợp với khối macrô có thể được bổ sung ở bước 410 với số lượng các bit được tích lũy đã có được kết hợp với lát entropi hiện hành không ở bước 411 vượt quá số lượng các bit tối đa được cho phép đối với lát entropi, thì số lượng các bit được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 412 để bao gồm các bit được kết hợp với khối macrô, và các bit được kết hợp với khối macrô có thể được ghi ở bước 414 vào luồng bit và được kết hợp với lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 402, và quá trình xử lý phân chia có thể được tiếp tục.

Nếu ở bước 410 tổng số lượng các bit được kết hợp với khối macrô và số lượng các bit được tích lũy đã có được kết hợp với lát entropi hiện hành vượt quá ở bước 415 số lượng các bit tối đa được cho phép đối với lát entropi, thì bộ mã hóa có thể bắt đầu ở bước 416 lát entropi mới được kết hợp với lát phục hồi hiện hành, và bộ đếm được kết hợp với số lượng các bit trong lát entropi hiện hành có thể được khởi tạo ở bước 418 về không. Số lượng các bit được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 412 để bao gồm các bit được kết hợp với khối macrô, và các bit được kết hợp với khối macrô có thể được ghi ở bước 414 vào luồng bit và được kết hợp với lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 402, và quá trình xử lý phân chia có thể được tiếp tục.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.16. Theo các phương án này, bộ mã hóa có thể, đối với lát phục hồi, phân chia lát phục hồi thành các lát entropi trong đó không có lát entropi nào có thể có kích thước lớn hơn so với số lượng tối đa định trước của các bit. Theo các phương án này, bộ mã hóa có thể kết hợp các phần tử cú pháp khối macrô với lát entropi cho đến khi kích thước của lát entropi đạt tới ngưỡng được kết hợp với số lượng các bit tối đa được định trước được cho phép trong lát entropi. Theo một số phương án, ngưỡng có thể

là phần trăm của số lượng các bit tối đa được cho phép trong lát entropi. Theo một phương án thực hiện làm ví dụ, ngưỡng có thể là 90% số lượng các bit tối đa được cho phép trong lát entropi, giả sử rằng số lượng các bit lớn nhất được mong muốn trong khối macrô nhỏ hơn 10% số lượng tối đa của các bit. Theo phương án thực hiện khác làm ví dụ, ngưỡng có thể là phần trăm của số lượng các bit tối đa được cho phép trong lát entropi trong đó phần trăm có thể được dựa vào số lượng các bit lớn nhất mong muốn trong khối macrô. Theo các phương án này, một khi kích thước của lát entropi vượt quá kích thước ngưỡng, thì lát entropi khác có thể được tạo ra. Kích thước ngưỡng có thể được chọn để đảm bảo rằng lát entropi không vượt quá số lượng các bit tối đa được cho phép trong lát entropi. Theo một số phương án, kích thước ngưỡng có thể là hàm của số lượng các bit tối đa được cho phép trong lát entropi và cần đánh giá số lượng các bit tối đa đối với khối macrô.

Bộ mã hóa có thể khởi tạo ở bước 440 về không bộ đếm được kết hợp với số lượng các bit trong lát entropi hiện hành. giá trị của bộ đếm được ký hiệu là A nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.16. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 442. Khối macrô tiếp theo có thể được xác định theo thứ tự xử lý khối macrô định trước. Theo một số phương án, thứ tự xử lý khối macrô có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macrô có thể được biến đổi ở bước 444 thành chuỗi các bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Các bin được kết hợp với khối macrô có thể được mã hoá entropi ở bước 446, và số lượng các bin được kết hợp với khối macrô có thể được xác định ở bước 448. Số lượng các bit được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 450 để bao gồm các bin được kết hợp với khối macrô, và các bin được mã hóa entropi được kết hợp với khối macrô có thể được ghi ở bước 452 vào luồng bit. Nếu ở bước 454 số lượng các bit được tích lũy được kết hợp với lát entropi hiện hành lớn hơn ở bước 456 so với ngưỡng dựa vào số lượng các bit tối đa được cho phép trong lát entropi, thì bộ mã hóa có thể bắt đầu ở bước 458 lát entropi mới, và khởi tạo ở bước 460 về không bộ đếm được kết hợp với số lượng các bit trong lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 442, và quá trình xử lý phân chia có thể được tiếp tục. Nếu số lượng

các bit được tích lũy được kết hợp với lát entropi hiện hành không lớn hơn ở bước 455 so với ngưỡng dựa vào số lượng các bit tối đa được cho phép trong lát entropi, thì các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 442, và quá trình xử lý phân chia có thể được tiếp tục.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó each lát entropi có thể được kết hợp với không lớn hơn số lượng định trước của các khối macro.

Theo một số phương án thực hiện sáng chế, giới hạn về số lượng các khối macro tối đa trong lát phục hồi có thể được sử dụng ngoài giới hạn về kích thước của lát entropi.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được giới hạn nhỏ hơn so với số lượng định trước của các khối macro và nhỏ hơn so với số lượng bin định trước.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được giới hạn nhỏ hơn so với số lượng định trước của các khối macro và nhỏ hơn so với số lượng các bit định trước.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được giới hạn nhỏ hơn so với số lượng định trước của các khối macro, nhỏ hơn so với số lượng bin định trước và nhỏ hơn so với số lượng các bit định trước.

Theo một số phương án thực hiện sáng chế, việc mã hoá bin trong bộ mã hóa entropi có thể được cho phép thực hiện mã hóa song song nhiều hơn một bin, nhờ đó có thể giảm thời gian mã hóa. Cần hiểu rằng các phương án thực hiện này của sáng chế liên quan đến bộ mã hóa entropi được minh họa trên Fig.17. Theo các phương án này, bộ mã hóa entropi 480 có thể bao gồm bộ thích ứng ngữ cảnh 482, bộ chọn bộ mã hóa bin, dựa vào trạng thái 484 và nhiều bộ mã hóa bin, còn gọi là các bộ mã hoá, (ba bộ được thể hiện) 486, 488, 500 có thể vận hành theo cách song song. Các bin 502 có thể khả dụng cho bộ mã hóa entropi 480 từ bộ nhị phân hóa 504 có thể tạo ra các bin 502 từ các ký hiệu đầu vào 506. Các bin 502 có thể khả

dụng cho bộ thích ứng ngữ cảnh 482 và bộ chọn bộ mã hóa bin, dựa vào trạng thái 484. Bộ thích ứng ngữ cảnh 482 có thể thực hiện thích ứng ngữ cảnh và tạo ra trạng thái mô hình, còn gọi là trạng thái, 508 có thể được sử dụng để chọn bộ mã hóa bin từ đó bin 502 có thể được chọn trong số các bộ mã hóa bin 486, 488, 500. Bộ chọn bộ mã hóa bin, dựa vào trạng thái 484 có thể chọn bộ mã hóa bin được kết hợp với trạng thái mô hình được tạo ra 508 trong số các bộ mã hóa bin 486, 488, 500 để mã hoá bin 502. Trong một số phương án thực hiện (không được thể hiện), trạng thái được tạo ra 508 có thể khả dụng cho bộ mã hóa bin được chọn. Các bit đầu ra 510, 512, 514 có thể được tạo ra bởi các bộ mã hóa bin 486, 488, 500, và các bit đầu ra 510, 512, 514 có thể được kết hợp thành luồng bit. Theo một số phương án thực hiện sáng chế, các bit đầu ra 510, 512, 514 có thể được nhớ đệm và được kết hợp thành luồng bit bằng cách ghép. Theo các phương án thực hiện khác, các bit đầu ra 510, 512, 514 có thể được nhớ đệm và được kết hợp thành luồng bit theo sơ đồ xen kẽ.

Theo các phương án thực hiện sáng chế có dựa vào Fig.17, bin thứ nhất có thể được gửi tới bộ mã hóa bin thứ nhất đáp lại trạng thái mô hình thứ nhất được tạo ra liên quan đến bin thứ nhất. Bộ thích ứng ngữ cảnh 482, khi hoàn thành việc xử lý bin thứ nhất, có thể bắt đầu quá trình xử lý của bin thứ hai, gửi bin thứ hai tới bộ mã hóa bin thứ hai đáp lại trạng thái mô hình thứ hai được tạo ra liên quan đến bin thứ hai, do đó về cơ bản cho phép quá trình xử lý song song nhiều hơn một bin.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa entropi có thể bao gồm các bộ thích ứng ngữ cảnh có thể hoạt động theo cách song song và bộ mã hóa bin đơn. Trong các hệ thống trong đó các bộ thích ứng ngữ cảnh đòi hỏi thời gian xử lý lâu hơn so với bộ mã hóa bin, các bộ thích ứng ngữ cảnh hoạt động theo cách song song có thể giảm thời gian mã hoá. Một số trong các phương án thực hiện sáng chế liên quan đến bộ mã hóa entropi sẽ được mô tả có dựa vào Fig.18. Theo các phương án này, bộ mã hóa entropi 530 có thể bao gồm các bộ thích ứng ngữ cảnh (ba bộ được thể hiện) 532, 534, 536, bộ chọn bộ thích ứng ngữ cảnh 538, bộ chọn trạng thái 540 và bộ mã hóa bin 542. Các bin 544 có thể được khiến cho khả dụng với bộ mã hóa entropi 530 từ bộ nhị phân hóa 546 có thể tạo ra các bin 544 từ các ký hiệu đầu vào 548. Các bin 544 có thể được khiến cho khả dụng với

bộ chọn bộ thích ứng ngữ cảnh 538, bộ chọn trạng thái 540 và bộ mã hóa bin 542. Bộ chọn bộ thích ứng ngữ cảnh 538 có thể được sử dụng để chọn, hoặc lập lịch biểu, bộ thích ứng ngữ cảnh 532, 534, 536 theo đó bin 544 có thể được chọn và từ đó giá trị trạng thái 550, 552, 554 có thể được tạo ra. Theo một số phương án làm ví dụ, bộ chọn bộ thích ứng ngữ cảnh 538 có thể chọn bộ thích ứng ngữ cảnh trong số các bộ thích ứng ngữ cảnh 532, 534, 536 dựa vào cú pháp được kết hợp với bin, chẳng hạn bộ nhận dạng bộ thích ứng ngữ cảnh có thể được kết hợp với bin nhận dạng bộ thích ứng ngữ cảnh theo đó bin có thể được chọn để xử lý. Theo một số phương án thực hiện khác làm ví dụ, bộ chọn bộ thích ứng ngữ cảnh 538 có thể chọn bộ thích ứng ngữ cảnh trong số các bộ thích ứng ngữ cảnh 532, 534, 536 dựa vào giao thức lập lịch biểu hoặc ràng buộc cân bằng tải được kết hợp với các bộ thích ứng ngữ cảnh 532, 534, 536. Theo một số phương án, trạng thái được tạo ra giá trị có thể được chọn bởi bộ chọn trạng thái 540, theo tiêu chuẩn được sử dụng ở bộ chọn bộ thích ứng ngữ cảnh 538, với thời gian thích hợp cần cho bộ mã hóa bin 542. Bộ mã hóa bin 542 có thể sử dụng giá trị trạng thái 556 được thực hiện bởi bộ chọn trạng thái 540 khi mã hoá bin 544. Theo các phương án thực hiện khác của sáng chế, (không được thể hiện), giá trị trạng thái có thể không được yêu cầu bởi bộ mã hóa bin và, do đó, không được khiến cho khả dụng với bộ mã hóa bin. Các bit đầu ra 558 có thể được tạo ra bởi bộ mã hóa bin 542, và các bit đầu ra 558 có thể được kết hợp thành luồng bit. Theo một số phương án thực hiện sáng chế, các bit đầu ra 558 có thể được nhớ đệm và được kết hợp thành luồng bit bằng cách ghép. Theo các phương án thực hiện khác, các bit đầu ra 558 có thể được nhớ đệm và được kết hợp thành luồng bit theo sơ đồ xen kẽ.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa entropi có thể bao gồm các bộ thích ứng ngữ cảnh có thể hoạt động theo cách song song và các bộ mã hóa bin có thể hoạt động theo cách song song. Các phương án thực hiện này của sáng chế liên quan đến bộ mã hóa entropi làm ví dụ được minh họa trên Fig.19. Theo các phương án này, bộ mã hóa entropi 570 có thể bao gồm các bộ thích ứng ngữ cảnh (ba bộ được thể hiện) 572, 574, 576, bộ chọn bộ thích ứng ngữ cảnh 578, bộ chọn trạng thái 580, bộ chọn bộ mã hóa bin, dựa vào trạng thái 582 và các bộ mã hóa bin (ba bộ được thể hiện) 584, 586, 588. Bin 590 có thể được

khiến cho khả dụng với bộ mã hóa entropi 570 từ bộ nhị phân hóa 592 có thể tạo ra các bin 590 từ các ký hiệu đầu vào 594. Các bin 590 có thể được khiến cho khả dụng với bộ chọn bộ thích ứng ngữ cảnh 578, bộ chọn trạng thái 580 và bộ chọn bộ mã hóa bin 582. Bộ chọn bộ thích ứng ngữ cảnh 578 có thể được sử dụng để chọn, hoặc lập lịch biểu, bộ thích ứng ngữ cảnh 572, 574, 576 theo đó bin 590 có thể được chọn và từ đó giá trị trạng thái 596, 598, 600 có thể được tạo ra. Giá trị trạng thái được tạo ra có thể được chọn bởi bộ chọn trạng thái 580 với thời gian thích hợp cần được thực hiện cho bộ chọn bộ mã hóa bin, dựa vào trạng thái 582. Bộ chọn bộ mã hóa bin, dựa vào trạng thái 582 có thể sử dụng giá trị trạng thái 602 được thực hiện bởi bộ chọn trạng thái 580 để chọn bộ mã hóa bin trong số các bộ mã hóa bin 584, 586, 588 theo đó bin 590 có thể được chọn. Theo các phương án thực hiện (không được thể hiện), giá trị trạng thái 602 có thể được khiến cho khả dụng với bộ mã hóa bin được chọn. Bộ mã hóa bin được chọn có thể sử dụng giá trị trạng thái 602 khi mã hóa bin 590. Theo các phương án thực hiện khác của sáng chế, (không được thể hiện), giá trị trạng thái có thể không được yêu cầu bởi bộ mã hóa bin và, do đó, không khả dụng với bộ mã hóa bin. Các bit đầu ra 604, 606, 608 có thể được tạo ra bởi các bộ mã hóa bin 584, 586, 588 và các bit đầu ra 604, 606, 608 có thể được kết hợp thành luồng bit. Theo một số phương án thực hiện sáng chế, các bit đầu ra 604, 606, 608 có thể được nhớ đệm và được kết hợp thành luồng bit bằng cách ghép. Theo các phương án thực hiện khác, các bit đầu ra 604, 606, 608 có thể được nhớ đệm và được kết hợp thành luồng bit theo sơ đồ xen kẽ

Theo một phương án thực hiện sáng chế, sáng chế có thể bao gồm các bộ mã hóa/giải mã mã hóa độ dài thay đổi có thể hoạt động theo cách song song.

Theo một phương án thực hiện sáng chế, bộ mã hóa bin có thể bao gồm mã hóa số học nhị phân. Theo một phương án thực hiện khác của sáng chế, bộ mã hóa bin có thể bao gồm mã hoá độ dài thay đổi. Theo một phương án thực hiện khác nữa của sáng chế, bộ mã hóa bin có thể bao gồm mã hoá độ dài không đổi.

Nói chung, bộ mã hóa entropi có thể bao gồm N_{ca} bộ thích ứng ngữ cảnh và N_{bc} bộ mã hoá bin, trong đó N_{ca} là số nguyên lớn hơn hoặc bằng một và N_{bc} là số nguyên lớn hơn hoặc bằng một.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát

phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho một hoặc nhiều trong số N_{ca} bộ thích ứng ngữ cảnh và N_{bc} bộ mã hóa bin có thể mỗi trong số các bộ này hoạt động dựa vào số bin không lớn hơn số lượng bin giới hạn trong suốt quá trình xử lý lát entropi. Các bộ thích ứng ngữ cảnh và các bộ mã hóa bin với giới hạn như vậy có thể được coi là các bộ mã hóa entropi được hạn chế.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho không bộ nào trong số N_{ca} bộ thích ứng ngữ cảnh có thể hoạt động với số bin nhiều hơn so với B_{ca} bin trong suốt quá trình xử lý lát entropi. Theo một số phương án thực hiện sáng chế, giá trị của B_{ca} có thể được báo hiệu, chẳng hạn trong luồng bit, ràng buộc biên dạng, ràng buộc mức độ hoặc cơ chế thông thường khác.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho không bộ nào trong số N_{bc} bộ mã hóa bin có thể hoạt động với số bin nhiều hơn so với B_{bc} bin trong suốt quá trình xử lý của lát entropi. Theo một số phương án thực hiện sáng chế, giá trị của B_{bc} có thể được báo hiệu, chẳng hạn trong luồng bit, ràng buộc biên dạng, ràng buộc mức độ hoặc cơ chế thông thường khác.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho không bộ nào trong số N_{ca} bộ thích ứng ngữ cảnh có thể hoạt động với số bin nhiều hơn so với B_{ca} bin và không bộ nào trong số N_{bc} bộ mã hóa bin có thể hoạt động với số bin nhiều hơn so với B_{bc} bin trong suốt quá trình xử lý của lát entropi. Theo một số phương án thực hiện sáng chế, giá trị của B_{bc} và giá trị của B_{ca} có thể được báo hiệu, chẳng hạn trong luồng bit, ràng buộc biên dạng, ràng buộc mức độ hoặc cơ chế thông thường khác.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho bộ thích ứng ngữ cảnh N_{ca} thứ i , ký hiệu là $N_{ca}(i)$, trong

đó $i=1,\dots,N_{ca}$, có thể hoạt động với không lớn hơn $B_{ca}(i)$ bin và bộ mã hoá bin N_{bc} thứ i , $N_{bc}(i)$, trong đó $i=1,\dots,N_{bc}$, có thể hoạt động với không lớn hơn $B_{bc}(i)$ bin trong suốt quá trình xử lý của lát entropi. Theo một số phương án thực hiện sáng chế, các giá trị của $B_{bc}(i)$ và các giá trị của $B_{ca}(i)$ có thể được báo hiệu, chẳng hạn trong luồng bit, ràng buộc biên dạng, ràng buộc mức độ hoặc cơ chế thông thường khác.

Một số phương án thực hiện sáng chế làm ví dụ sẽ được mô tả có dựa vào Fig.20. Theo các phương án này, bộ mã hóa có thể, đối với lát phục hồi, phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho một, hoặc nhiều, bộ trong số N_{ca} bộ thích ứng ngữ cảnh và N_{bc} bộ mã hóa bin có thể hoạt động với số bin không lớn hơn số bin được giới hạn. Bộ mã hóa có thể khởi tạo ở bước 650 về không bộ đếm, đối với từng bộ mã hóa entropi được hạn chế, được kết hợp với số lượng các bin được xử lý trong lát entropi hiện hành. Nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.20, giá trị của bộ đếm được ký hiệu là A , trong đó A biểu thị vector với mỗi mục nhập trong vector tương ứng với số lượng tích lũy của các bin được xử lý, đối với lát entropi hiện hành, bởi bộ mã hóa entropi được hạn chế. Các phần tử cú pháp đối với khối macro tiếp theo có thể thu được ở bước 652. Khối macro tiếp theo có thể được xác định theo thứ tự xử lý khối macro định trước. Theo một số phương án, thứ tự xử lý khối macro có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp phi nhị phân trong khối macro có thể được biến đổi ở bước 654 thành chuỗi các bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Số lượng bin, được kết hợp với khối macro, được xử lý bởi mỗi bộ mã hóa entropi được hạn chế có thể được xác định ở bước 656. Số lượng các bin được kết hợp với khối macro có thể bao gồm các bin các chuỗi bin được kết hợp với các phần tử cú pháp phi nhị phân ngoài các phần tử cú pháp nhị phân. nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.20, số lượng bin, được kết hợp với khối macro, được xử lý bởi mỗi bộ mã hóa entropi được hạn chế được ký hiệu là num , trong đó num biểu thị vector với mỗi mục nhập trong vector tương ứng với số lượng bin được xử lý, đối với khối macro hiện hành, bởi bộ mã hóa entropi được hạn chế.

Nếu số lượng các bin được kết hợp với khối macro đối với mỗi bộ mã hóa

entropi được hạn chế có thể được bổ sung ở bước 658 cho số lượng các bin được tích lũy đã có, được kết hợp với lát entropi hiện hành, đối với mỗi bộ mã hóa entropi được hạn chế, mà ở bước 659 không vượt quá số lượng bin tối đa được cho phép đối với bộ mã hóa entropi được hạn chế bất kỳ, thì số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 660 để bao gồm các bin được kết hợp với khối macrô, và các bin được kết hợp với khối macrô có thể được ghi ở bước 662, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 652, và quá trình xử lý phân chia có thể được tiếp tục.

Nếu ở bước 658 tổng số lượng các bin được kết hợp với khối macrô và số lượng các bin được tích lũy đã có được kết hợp với lát entropi hiện hành vượt quá ở bước 663 số lượng bin tối đa được cho phép đối với bộ mã hóa entropi được hạn chế bất kỳ, thì bộ mã hóa có thể bắt đầu ở bước 664 lát entropi mới được kết hợp với lát phục hồi hiện hành, và bộ đếm được kết hợp với số lượng các bin trong lát entropi hiện hành có thể được khởi tạo ở bước 666 về không. Số lượng các bin được tích lũy được kết hợp với lát entropi hiện hành có thể được cập nhật ở bước 660 để bao gồm các bin được kết hợp với khối macrô, và các bin được kết hợp với khối macrô có thể được ghi ở bước 662, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 652, và quá trình xử lý phân chia có thể được tiếp tục.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.21. Theo các phương án này, bộ mã hóa có thể, đối với lát phục hồi, phân chia lát phục hồi thành các lát entropi, trong đó kích thước của mỗi lát entropi có thể được hạn chế sao cho một hoặc nhiều, trong số N_{ca} bộ thích ứng ngữ cảnh và N_{bc} bộ mã hóa bin có thể hoạt động với không lớn hơn số lượng bin được giới hạn. Bộ mã hóa có thể khởi tạo ở bước 700 về không bộ đếm, đối với mỗi bộ mã hóa entropi được hạn chế, được kết hợp với số lượng các bin được xử lý trong lát entropi hiện hành bởi bộ mã hóa entropi được hạn chế. nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.21, giá trị của bộ đếm được ký hiệu là A, trong đó A biểu thị vector với mỗi mục nhập trong vector tương ứng với số lượng tích lũy của các bin

được xử lý, đối với lát entropi hiện hành, bởi bộ mã hóa entropi được hạn chế. Theo các phương án này, bộ mã hóa có thể kết hợp các phần tử cú pháp khối macrô với lát entropi cho đến khi số lượng các bin được xử lý bởi bộ mã hóa entropi được hạn chế đạt tới ngưỡng được kết hợp với số lượng bin tối đa đã định được phép xử lý, trong lát entropi, bởi bộ mã hóa entropi được hạn chế. Theo một số phương án, ngưỡng có thể là phần trăm của số lượng bin tối đa được phép xử lý, trong lát entropi, bởi bộ mã hóa entropi được hạn chế. Theo một phương án thực hiện làm ví dụ, ngưỡng có thể là 90% của số lượng bin tối đa được phép xử lý, trong lát entropi, bởi bộ mã hóa entropi được hạn chế, giả sử rằng số lượng bin lớn nhất mong muốn trong khối macrô cần được xử lý bởi bộ mã hóa entropi được hạn chế nhỏ hơn 10% số lượng bin tối đa được phép xử lý, trong lát entropi, bởi bộ mã hóa entropi được hạn chế. Theo phương án thực hiện khác làm ví dụ, ngưỡng có thể là phần trăm của số lượng bin tối đa được phép xử lý, trong lát entropi, bởi bộ mã hóa entropi được hạn chế trong đó phần trăm có thể được dựa vào số lượng bin lớn nhất mong muốn trong khối macrô cần được xử lý bởi bộ mã hóa entropi được hạn chế. Theo các phương án này, một khi kích thước của lát entropi vượt quá kích thước ngưỡng, thì lát entropi khác có thể được tạo ra. Kích thước ngưỡng có thể được chọn để đảm bảo rằng lát entropi không vượt quá số lượng bin tối đa được phép xử lý bởi một bộ mã hóa entropi được hạn chế bất kỳ trong lát entropi. Theo một số phương án, kích thước ngưỡng có thể là hàm của số lượng bin tối đa được cho phép trong lát entropi và cần đánh giá số lượng bin tối đa đối với khối macrô.

Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 702. Khối macrô tiếp theo có thể được xác định theo thứ tự xử lý khối macrô định trước. Theo một số phương án, thứ tự xử lý khối macrô có thể tương ứng với thứ tự quét mảnh. Các phần tử cú pháp nhị phân trong khối macrô có thể được biến đổi ở bước 704 thành chuỗi các bin. Các phần tử cú pháp nhị phân có thể không cần biến đổi. Các bin được kết hợp với khối macrô có thể được ghi ở bước 706, bởi bộ mã hóa entropi, vào luồng bit và được kết hợp với lát entropi hiện hành. số lượng bin, được kết hợp với khối macrô, được xử lý bởi mỗi bộ mã hóa entropi được hạn chế có thể được xác định ở bước 708. Số lượng các bin được kết hợp với khối macrô có thể bao gồm các bin trong các chuỗi bin được kết hợp với các phần

tử cú pháp phi nhị phân ngoài các phần tử cú pháp nhị phân. Nhằm mục đích minh họa trong phần mô tả dưới đây, có dựa vào Fig.21, số lượng bin, được kết hợp với khối macrô, được xử lý bởi mỗi bộ mã hóa entropi được hạn chế được ký hiệu là num , trong đó num biểu thị vector với mỗi mục nhập trong vector tương ứng với số lượng bin được xử lý, đối với khối macrô hiện hành, bởi bộ mã hóa entropi được hạn chế tương ứng. Số lượng các bin được tích lũy, được kết hợp với lát entropi hiện hành, được xử lý bởi mỗi bộ mã hóa entropi được hạn chế có thể được cập nhật ở bước 710 để bao gồm các bin được kết hợp với khối macrô. Nếu ở bước 712 số lượng các bin được tích lũy, được kết hợp với lát entropi hiện hành, được xử lý bởi bộ mã hóa entropi được hạn chế lớn hơn 714 so với ngưỡng, ngưỡng này được ký hiệu là $TH(MaxNumBins)(i)$ đối với bộ mã hóa entropi được hạn chế i , thì bộ mã hóa có thể bắt đầu ở bước 716 lát entropi mới, và khởi tạo ở bước 718 về không bộ đếm được kết hợp với số lượng các bin được xử lý bởi mỗi bộ mã hóa entropi được hạn chế trong lát entropi hiện hành. Các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 702, và quá trình xử lý phân chia có thể được tiếp tục. Nếu số lượng các bin được tích lũy, được kết hợp với lát entropi hiện hành, được xử lý bởi bộ mã hóa entropi được hạn chế không lớn hơn ở bước 713 so với ngưỡng, thì các phần tử cú pháp đối với khối macrô tiếp theo có thể thu được ở bước 702, và quá trình xử lý phân chia có thể được tiếp tục.

Một số phương án thực hiện sáng chế có thể bao gồm kết hợp của tiêu chuẩn mô tả trên đây để phân chia lát entropi.

Cần hiểu rằng mặc dù một số phương án thực hiện sáng chế có thể hạn chế kích thước của lát entropi nhỏ hơn so với kích thước định trước thứ nhất, kích thước của lát entropi có thể được hạn chế theo cách tương đương không vượt quá kích thước định trước thứ hai. Các phương án được mô tả ở đây là các phương án làm ví dụ thực hiện sáng chế, và người có hiểu biết trung bình trong lĩnh vực kỹ thuật có thể hiểu rằng có thể có các phương án tương đương khác để hạn chế kích thước của lát entropi.

Theo một số phương án thực hiện sáng chế, bắt đầu lát entropi mới có thể bao gồm kết thúc lát và xét đến lát entropi mới lát entropi hiện hành.

Theo một số phương án thực hiện sáng chế, việc giải mã các bit trong lát

entropi có thể được thực hiện song song trong bộ giải mã entropi bao gồm các bộ giải mã bin, mà có thể giảm thời gian giải mã. Các phương án thực hiện làm ví dụ của sáng chế liên quan đến bộ giải mã entropi 750 làm ví dụ sẽ được mô tả dưới đây, có dựa vào Fig.22, bao gồm (ba bộ được thể hiện) các bộ giải mã bin 762, 764, 766. Các bit 752 trong lát entropi và các ký hiệu được giải mã trước đó 754 có thể được khiến cho khả dụng với bộ giải mã entropi 750. Các bit 752 có thể được khiến cho khả dụng với bộ chọn bộ giải mã bin 756 có thể chọn bin bộ giải mã, dựa vào trạng thái ngữ cảnh 758 được tạo ra từ bộ thích ứng ngữ cảnh 760, trong số các bộ giải mã bin 762, 764, 766. Bộ thích ứng ngữ cảnh 760 có thể tạo ra trạng thái ngữ cảnh 758 dựa vào các ký hiệu được giải mã trước đó 754 được khiến cho khả biến với bộ thích ứng ngữ cảnh 760. Bộ chọn bộ giải mã bin 756 có thể gán bộ giải mã bin 762, 764, 766 dựa vào trạng thái ngữ cảnh 758. Bit cần được giải mã 752 có thể được thực hiện bởi bộ chọn bộ giải mã bin 756 đến bin bộ giải mã được chọn. Các bộ giải mã bin 762, 764, 766 có thể tạo ra các bin được giải mã 768, 770, 772 có thể được dồn kênh bởi bộ dồn kênh 774 và các bin được dồn kênh 776 có thể được gửi tới bộ ký hiệu hóa 778 có thể tạo ra các ký hiệu 754 được kết hợp với các bin 776.

Theo một số phương án thực hiện sáng chế, việc giải mã các bit trong lát entropi có thể được thực hiện song song trong bộ giải mã entropi bao gồm các bộ thích ứng ngữ cảnh, nó có thể giảm thời gian giải mã. Các phương án thực hiện sáng chế liên quan đến bộ giải mã entropi 800 làm ví dụ sẽ được mô tả dưới đây, có dựa vào Fig.23, bộ giải mã entropi 800 bao gồm (ba bộ được thể hiện) các bộ thích ứng ngữ cảnh 814, 816, 818. Các bit 802 trong lát entropi và các ký hiệu được giải mã trước đó 810 có thể được khiến cho khả dụng với bộ giải mã entropi 800. Các bit 802 có thể được khiến cho khả dụng với bộ chọn bộ thích ứng ngữ cảnh 812 có thể chọn từ các bộ thích ứng ngữ cảnh 814, 816, 818 bộ thích ứng ngữ cảnh đối với quá trình xử lý giải mã của bit đầu vào. Theo một số phương án thực hiện sáng chế, bộ chọn bộ thích ứng ngữ cảnh 812 có thể chọn bộ thích ứng ngữ cảnh thứ N khi thu mọi bit thứ N. Bộ thích ứng ngữ cảnh được chọn có thể tạo ra trạng thái ngữ cảnh 820, 822, 824 dựa vào các ký hiệu được giải mã trước đó 810 được khiến cho khả biến với bộ thích ứng ngữ cảnh được chọn. Bộ chọn trạng thái

826, ở thời gian thích hợp, có thể chọn trạng thái ngữ cảnh được tạo ra khi được kết hợp với bit đầu vào. Theo một số phương án thực hiện sáng chế, bộ chọn trạng thái 826 có thể chọn bộ thích ứng ngữ cảnh thứ N khi thu mọi bit thứ N theo cùng một quy trình như bộ chọn bộ thích ứng ngữ cảnh 812. Trạng thái được chọn 828 có thể được khiến cho khả dụng với bin bộ giải mã 804. Bin bộ giải mã 804 có thể giải mã bit 802 và gửi bin được giải mã 806 tới bộ ký hiệu hóa 808 có thể tạo ra ký hiệu 810 được kết hợp với bin được giải mã 806.

Theo một số phương án thực hiện sáng chế, việc giải mã các bit trong lát entropi có thể được thực hiện song song trong bộ giải mã entropi bao gồm các bộ thích ứng ngữ cảnh và các bộ giải mã bin, có thể giảm thời gian giải mã. Các phương án thực hiện sáng chế liên quan đến bộ giải mã entropi 850 làm ví dụ sẽ được mô tả dưới đây, có dựa vào Fig.24, bộ giải mã entropi 850 bao gồm (ba bộ được thể hiện) các bộ thích ứng ngữ cảnh 852, 854, 856 và (ba bộ được thể hiện) các bộ giải mã bin 858, 860, 862. Các bit 864 trong lát entropi và các ký hiệu được giải mã trước đó 866 có thể được khiến cho khả dụng với bộ giải mã entropi 850. Các bit 864 có thể được khiến cho khả dụng với bộ chọn bộ thích ứng ngữ cảnh 868 có thể chọn từ các bộ thích ứng ngữ cảnh 852, 854, 856 bộ thích ứng ngữ cảnh đối với quá trình xử lý giải mã bit đầu vào. Theo một số phương án thực hiện sáng chế, bộ chọn bộ thích ứng ngữ cảnh 868 có thể chọn bộ thích ứng ngữ cảnh thứ N khi thu mọi bit thứ N. Bộ thích ứng ngữ cảnh được chọn có thể tạo ra trạng thái ngữ cảnh 870, 872, 874 dựa vào các ký hiệu được giải mã trước đó 866 được khiến cho khả biến với bộ thích ứng ngữ cảnh được chọn. Bộ chọn trạng thái 876, với thời gian thích hợp, có thể chọn trạng thái ngữ cảnh được tạo ra khi được kết hợp với bit đầu vào. Theo một số phương án thực hiện sáng chế, bộ chọn trạng thái 876 có thể chọn bộ thích ứng ngữ cảnh thứ N khi thu mọi bit thứ N theo cùng một quá trình xử lý như bộ chọn bộ thích ứng ngữ cảnh 868. Trạng thái được chọn 878 có thể được khiến cho khả dụng với bộ chọn bộ giải mã bin 880, có thể chọn, dựa vào trạng thái ngữ cảnh được chọn 878, bộ giải mã bin 858, 860, 862. Bộ chọn bộ giải mã bin 880 có thể gán bộ giải mã bin 858, 860, 862 dựa vào trạng thái ngữ cảnh 878. Bit cần được giải mã 864 có thể được thực hiện bởi bộ chọn bộ giải mã bin 880 đến bộ giải mã bin được chọn. Các bộ giải mã bin 858, 860, 862 có thể tạo ra

các bin được giải mã 882, 884, 886 có thể được dồn kênh bởi bộ dồn kênh 888 và các bin được dồn kênh 890 có thể được gửi tới bộ ký hiệu hóa 892 có thể tạo ra các ký hiệu 866 được kết hợp với các bin 890.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó các khối macrô trong lát entropi là kề nhau. Fig.25 thể hiện lát phục hồi 950 làm ví dụ được phân chia thành ba lát entropi: lát entropi 0 được thể hiện ở phần gạch chéo 952, lát entropi 1 được thể hiện ở phần trắng 954 và lát entropi 2 được thể hiện ở phần chấm 956. Các khối macrô trong mỗi lát entropi 952, 954, 956, trong lát phục hồi 950 làm ví dụ này, đều kề nhau.

Theo các phương án thực hiện khác của sáng chế, bộ mã hóa có thể phân chia lát phục hồi thành các lát entropi, trong đó các khối macrô trong lát entropi có thể không kề nhau. Fig.26 thể hiện lát phục hồi 960 làm ví dụ được phân chia thành ba lát entropi: lát entropi 0 được thể hiện trong phần gạch chéo 962, lát entropi 1 được thể hiện trong phần trắng 964 và lát entropi 2 được thể hiện trong phần chấm 966. Các khối macrô trong mỗi lát entropi 962, 964, 966, lát phục hồi 960 làm ví dụ này, không liên tiếp. Sự phân chia của lát phục hồi trong đó các khối macrô trong lát entropi không liên tiếp có thể gọi là phân chia xen kẽ.

Theo một số phương án thực hiện sáng chế, trong khi giải mã entropi khối trong lát entropi hiện hành, bộ giải mã có thể sử dụng các khối khác từ cùng lát entropi để dự báo thông tin liên quan đến giải mã entropi của khối hiện hành. Theo một số phương án thực hiện sáng chế, trong khi phục hồi khối hiện hành trong lát phục hồi, các khối khác từ cùng một lát phục hồi có thể được sử dụng để dự báo thông tin liên quan đến phục hồi khối hiện hành.

Theo một số phương án thực hiện sáng chế trong đó lát phục hồi bao gồm phân chia xen kẽ, các khối lân cận trong lát entropi được dùng khi giải mã khối hiện hành trong lát entropi có thể không trực tiếp lân cận, hoặc kề nhau. Fig.27 thể hiện trường hợp này đối với phân chia xen kẽ làm ví dụ được thể hiện Fig.26.

Như được thể hiện trên Fig.27, đối với khối hiện hành 970 trong lát entropi 964, khối lân cận bên trái được dùng để giải mã entropi của khối hiện hành 970 là liền kề, khối lân cận bên trái 972 trong lát entropi 964. khối lân cận phía trên được dùng để giải mã entropi của khối hiện hành 970 là không liền kề, khối lân cận phía

trên 974 trong cùng lát entropi 964. Để phục hồi khối hiện hành 970, khối lân cận bên trái là liền kề, khối lân cận bên trái 972 trong lát phục hồi 960, và khối lân cận phía trên là liền kề, khối lân cận phía trên 976 trong lát phục hồi 960.

Theo một số phương án thực hiện sáng chế trong đó lát phục hồi bao gồm phân chia xen kẽ, không có khối lân cận thích hợp trong lát entropi cần được dùng trong khi giải mã khối hiện hành trong lát entropi. Fig.28 thể hiện trường hợp này cho sự phân chia xen kẽ làm ví dụ được thể hiện trên Fig.26.

Như được thể hiện trên Fig.28, đối với khối hiện hành 980 trong lát entropi 964, không có khối lân cận bên trái trong lát entropi 964 cần được dùng để giải mã entropi của khối hiện hành 980. Khối lân cận phía trên được dùng để giải mã entropi của khối hiện hành 980 là không liền kề, khối lân cận phía trên 982 trong cùng lát entropi 964. Để phục hồi khối hiện hành 980, khối lân cận bên trái là liền kề, khối lân cận bên trái 984 trong lát phục hồi 960, và khối lân cận phía trên là liền kề, khối lân cận phía trên 986 trong lát phục hồi 960.

Theo một số phương án thực hiện sáng chế, bộ giải mã có thể xử lý trước luồng bit đến hoàn chỉnh để nhận dạng các vị trí của các lát entropi. Theo một số phương án thực hiện sáng chế, bộ giải mã có thể xử lý trước toàn phần lát phục hồi để nhận dạng các vị trí của các lát entropi trong lát phục hồi. Theo một số phương án, các vị trí của các lát entropi có thể được xác định bằng cách nhận dạng các vị trí của các mào đầu lát entropi. Theo các phương án này, bộ giải mã có thể đọc các bit trong luồng bit và các giá trị bắt đầu mã xác định trước có thể được nhận dạng.

Theo các phương án thực hiện khác, các mào đầu lát entropi có thể được ràng buộc và phạm vi của các bit được đặt ở các vị trí xác định trước trong luồng bit đến. Theo các phương án thực hiện khác, các mào đầu lát entropi có thể được ràng buộc vào phạm vi của các byte được đặt ở các vị trí xác định trước trong luồng bit đến. Theo các phương án này, hoặc bit sắp đặt hoặc byte được sắp đặt, bộ giải mã một cách đáng kể không cần xử lý trước phần lớn của luồng bit đến để định vị các lát entropi.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể báo hiệu, trong luồng bit, thông tin vị trí lát entropi (thông tin entropy-slice-location), sau đây gọi là các tham số vị trí lát entropi (tham số entropy-slice-location), chẳng hạn thông

tin dịch vị và phạm vi, nó có thể ràng buộc các vị trí của các mào đầu lát entropi. Theo các phương án thực hiện khác, thông tin vị trí lát entropi có thể không được báo hiệu trong luồng bit, nhưng có thể được xác định từ các tham số lát entropi (tham số entropy-slice), chẳng hạn số lượng bin không đổi được phép trong lát entropi bất kỳ định trước, số lượng các bit không đổi được phép trong lát entropi định trước bất kỳ và các tham số lát entropi khác. Theo các phương án thực hiện khác của sáng chế, thông tin vị trí lát entropi có thể được xác định bởi phương tiện thông thường khác, chẳng hạn thông tin có thể được định rõ trong ràng buộc biên dạng, ràng buộc mức độ, ràng buộc ứng dụng, hoặc ràng buộc khác, hoặc thông tin có thể được báo hiệu là thông tin bổ sung hoặc được báo hiệu bởi phương tiện bên ngoài khác.

Theo một số phương án thực hiện sáng chế, một tập các giá trị tham số vị trí lát entropi có thể được dùng cho tất cả các lát entropi trong luồng bit. Theo các phương án thực hiện khác, các giá trị tham số vị trí lát entropi có thể được xác định cho nhóm các điểm ảnh được biểu thị bởi một phần của chuỗi. Theo các phương án thực hiện khác, các giá trị tham số vị trí lát entropi có thể được xác định đối với mỗi ảnh trong luồng bit và có thể được dùng cho tất cả các lát entropi trong ảnh được kết hợp. Theo các phương án thực hiện khác, các giá trị tham số vị trí lát entropi có thể được xác định cho mỗi lát phục hồi trong luồng bit và có thể được dùng cho tất cả các lát entropi trong lát phục hồi được kết hợp. Theo các phương án thực hiện khác của sáng chế, các tập các giá trị tham số vị trí lát entropi có thể được dùng bởi bộ giải mã. Theo các phương án thực hiện khác của sáng chế, các giá trị tham số vị trí lát entropi có thể được gán cho các phần tử nhận dạng lát entropi, chẳng hạn mào đầu lát entropi thứ nhất có thể sử dụng tập thứ nhất của các giá trị tham số vị trí lát entropi, mào đầu lát entropi thứ hai có thể sử dụng tập thứ hai của các giá trị tham số vị trí lát entropi và, nói chung, mào đầu lát entropi thứ N có thể sử dụng tập thứ N của các giá trị tham số vị trí lát entropi. Theo một số phương án thực hiện sáng chế, các giá trị tham số lát entropi có thể được gán cho phần tử nhận dạng khung. Theo một phương án thực hiện làm ví dụ, ảnh thứ nhất có thể sử dụng tập thứ nhất của các giá trị tham số lát entropi, ảnh thứ hai có thể sử dụng tập thứ hai của các giá trị tham số lát entropi và, nói chung, ảnh thứ N có thể

sử dụng tập thứ N của các giá trị tham số vị trí lát entropi. Theo phương án thực hiện khác làm ví dụ, ảnh có kiểu thứ nhất có thể sử dụng tập thứ nhất các giá trị tham số vị trí lát entropi và ảnh có kiểu thứ hai có thể sử dụng tập thứ hai của các giá trị tham số vị trí lát entropi. Các kiểu làm ví dụ của các ảnh là các nội ảnh, các ảnh được dự báo và các kiểu khác của các ảnh.

Theo một số phương án thực hiện sáng chế bao gồm bộ mã hóa/giải mã H.264/AVC, dịch vị lát entropi và phạm vi lát entropi có thể được báo hiệu trong RBSP (RBSP: Raw Byte Sequence Payload (tải tin của chuỗi byte thô)) tập tham số chuỗi bằng cách bổ sung tham số “entropy_slice_offset” và tham số “entropy_slice_range” vào tập tham số chuỗi. Bảng 3 liệt kê cú pháp RBSP tập tham số chuỗi theo các phương án thực hiện sáng chế.

Theo một số phương án thực hiện sáng chế bao gồm bộ mã hóa/giải mã H.264/AVC, dịch vị entropy-slice và phạm vi entropy-slice có thể được báo hiệu trong RBSP tập tham số ảnh bằng cách bổ sung tham số “entropy_slice_offset” và “entropy_slice_range” vào tập tham số ảnh. Bảng 4 liệt kê cú pháp RBSP tập tham số ảnh làm ví dụ theo các phương án thực hiện sáng chế.

Theo một số phương án thực hiện sáng chế bao gồm bộ mã hóa/giải mã H.264/AVC, dịch vị lát entropi và phạm vi lát entropi có thể được báo hiệu vào đầu lát bằng cách bổ sung tham số “entropy_slice_offset” và “entropy_slice_range” vào đầu lát. Bảng 5 liệt kê cú pháp vào đầu lát làm ví dụ theo các phương án thực hiện sáng chế.

Theo một số phương án thực hiện sáng chế, dịch vị lát entropi và phạm vi lát entropi có thể được chỉ báo cho từng điểm mức độ thích hợp của bộ mã hóa theo bảng này, chẳng hạn như được thể hiện trong bảng 6, trong đó $O_{m,n}$ biểu thị dịch vị dịch vị lát entropi đối với điểm thích hợp mức độ m.n và $R_{m,n}$ biểu thị phạm vi phạm vi lát entropi đối với điểm thích hợp m.n.

seq_parameter_set_rbsp() {	C	Phần tử mô tả
profile_idc	0	u(8)
reserved_zero_8bits /* equal 0 */	0	u(8)
level_idc	0	u(8)

seq_parameter_set_id	0	ue(v)
bit_depth_luma_minus8	0	ue(v)
bit_depth_chroma_minus8	0	ue(v)
increased_bit_depth_luma	0	ue(v)
increased_bit_depth_chroma	0	ue(v)
log2_max_frame_num_minus4	0	ue(v)
log2_max_pic_order_cnt_lsb_minus4	0	ue(v)
max_num_ref_frames	0	ue(v)
gaps_in_frame_num_value_allowed_flag	0	u(1)
log2_min_coding_unit_size_minus3	0	ue(v)
max_coding_unit_hierarchy_depth	0	ue(v)
log2_min_transform_unit_size_minus2	0	ue(v)
max_transform_unit_hierarchy_depth	0	ue(v)
pic_width_in_luma_samples	0	u(16)
pic_height_in_luma_samples	0	u(16)
entropy_slice_offset	0	ue(v)
entropy_slice_range	0	ue(v)
rbsp_trailing_bits()	0	
}		

Bảng 3: Bảng cú pháp RBSP tập tham số chuỗi làm ví dụ

pic_parameter_set_rbsp() {	C	Phần tử mô tả
pic_parameter_set_id	1	ue(v)
seq_parameter_set_id	1	ue(v)
entropy_coding_mode_flag	1	u(1)
num_ref_idx_l0_default_active_minus1	1	ue(v)
num_ref_idx_l1_default_active_minus1	1	ue(v)
pic_init_qp_minus26 /* relative 26 */	1	se(v)
constrained_intra_pred_flag	1	u(1)
entropy_slice_offset	0	ue(v)

entropy_slice_range	0	ue(v)
rbsp_trailing_bits()	1	
}		

Bảng 4: Bảng cú pháp RBSP tập tham số ảnh làm ví dụ

slice_header() {	C	Phần tử mô tả
first_lctb_in_slice	2	ue(v)
entropy_slice_flag		u(1)
if(!entropy_slice_flag){		
slice_type	2	ue(v)
pic_parameter_set_id	2	ue(v)
frame_num	2	u(v)
if(IdrPicFlag)		
idr_pic_id	2	ue(v)
pic_order_cnt_lsb	2	u(v)
if(slice_type == P slice_type == B) {		
num_ref_idx_active_override_flag	2	u(1)
if(num_ref_idx_active_override_flag) {		
num_ref_idx_l0_active_minus1	2	ue(v)
if(slice_type == B)		
num_ref_idx_l1_active_minus1	2	ue(v)
}		
}		
if(nal_ref_idc != 0)		
dec_ref_pic_marking()	2	
if(entropy_coding_mode_flag && slice_type != I)		
cabac_init_idc	2	ue(v)
slice_qp_delta	2	se(v)
alf_param()		
if(slice_type == P slice_type == B) {		

mc_interpolation_idc	2	ue(v)
mv_competition_flag	2	u(1)
(mv_competition_flag) {		
mv_competition_temporal_flag	2	u(1)
}		
}		
if (slice_type == B && mv_competition_flag)		
collocated_from_l0_flag	2	u(1)
} else		
(entropy_coding_mode_flag && slice_type != I)		
cabac_init_idc		ue(v)
entropy_slice_offset	0	ue(v)
entropy_slice_range	0	ue(v)
}		

Bảng 5: Bảng cú pháp làm ví dụ đối với mào đầu lát

Mức độ	Dịch vị lát entropi	Phạm vi lát entropi
1.1	O1.1	R1.1
1.2	O1.2	R1.2
:	:	:
m.n	Om.n	Rm.n
:	:	:
5.1	O5.1	R5.1

Bảng 6: Dịch vị lát entropi và phạm vi lát entropi làm ví dụ đối với mỗi mức độ

Theo một số phương án, thông tin vị trí lát entropi có thể bao gồm thông tin có thể ràng buộc các vị trí của các mào đầu lát entropi. Theo một ví dụ, thông tin vị trí lát entropi có thể bao gồm dịch vị, sau đây gọi là dịch vị chu kỳ hoặc dịch vị cơ bản, giá trị và phạm vi, sau đây gọi là độ lệch hoặc dịch vị đối với chu kỳ, giá trị. Vị trí mào đầu lát entropi có thể được ràng buộc dựa vào giá trị dịch vị và giá trị

phạm vi.

Theo một số phương án thực hiện sáng chế, giá trị dịch vị và giá trị phạm vi có thể được xác định một cách rõ ràng. Theo các phương án thực hiện khác của sáng chế, giá trị dịch vị và giá trị phạm vi có thể được xác định một cách ẩn là giá trị dịch vị tối thiểu và giá trị dịch vị tối đa. Theo các phương án thực hiện sáng chế, giá trị dịch vị và giá trị phạm vi có thể được xác định theo cách ẩn là giá trị dịch vị tối đa và độ chênh lệch giữa giá trị dịch vị tối đa và giá trị dịch vị tối thiểu. Theo các phương án thực hiện khác của sáng chế, giá trị dịch vị và giá trị phạm vi có thể được xác định theo cách ẩn là giá trị dịch vị tối thiểu và độ chênh lệch giữa giá trị dịch vị tối thiểu và giá trị dịch vị tối đa. Theo các phương án thực hiện khác, giá trị dịch vị và giá trị phạm vi có thể được xác định theo cách ẩn là giá trị thứ ba và độ chênh lệch giữa giá trị thứ ba và giá trị dịch vị tối đa và giá trị dịch vị tối thiểu. Theo các phương án thực hiện khác của sáng chế, giá trị dịch vị và giá trị phạm vi có thể được xác định thông qua chỉ số đưa vào trong bảng tìm kiếm chứa các giá trị bit tối thiểu và tối đa tương ứng. Theo một số phương án, giá trị dịch vị và giá trị phạm vi có thể được xác định bằng cách sử dụng cây tìm kiếm trên cơ sở dịch vị. Theo một số phương án, giá trị dịch vị và giá trị phạm vi có thể được xác định bằng cách sử dụng chỉ số chi phí tối thiểu hóa. Người có hiểu biết trung bình trong lĩnh vực kỹ thuật liên quan có thể hiểu rằng có nhiều phương pháp đã biết để xác định một cách ẩn giá trị phạm vi và giá trị dịch vị và giả thiết rằng bộ mã hóa và bộ giải mã hoạt động với cùng giá trị đối với dịch vị xác định trước và các giá trị phạm vi xác định trước.

Theo một số phương án thực hiện sáng chế, việc báo hiệu giá trị phạm vi có thể là tùy chọn. Theo một số phương án, khi giá trị phạm vi không được báo hiệu, thì giá trị phạm vi có thể được thiết lập là giá trị xác định trước. Theo phương án làm ví dụ này, giá trị xác định trước có thể là không. Theo phương án thực hiện khác làm ví dụ, giá trị xác định trước có thể là giá trị nguyên khác không.

Theo phương án được mô tả có dựa vào Fig.29, mào đầu lát entropi được kết hợp với lát entropi, số lượng lát N trong lát phục hồi, có thể được ràng buộc để bắt đầu sau $Nk - p$ bit từ khi bắt đầu, hoặc vị trí cố định khác bên trong, mào đầu lát phục hồi, trong đó k biểu thị giá trị dịch vị và p biểu thị phạm vi. Vị trí từ đó

$Nk - p$ bit có thể được đo có thể được gọi là vị trí tham chiếu. Theo các phương án thực hiện khác, vị trí tham chiếu có thể không được kết hợp với lát phục hồi nhất định và có thể là cùng vị trí cố định trong luồng bit đối với tất cả các lát entropi. Theo các phương án thực hiện khác, mào đầu lát entropi có thể là byte được sắp xếp, và sự ràng buộc có thể được kết hợp với số lượng byte. Mặc dù ví dụ này, như được thể hiện trên Fig.29, được mô tả theo thuật ngữ bit, nhưng người có hiểu biết trung bình trong lĩnh vực kỹ thuật có thể hiểu rằng có là thực hiện theo phương án sắp xếp byte, theo lựa chọn khác.

Fig.29 là hình vẽ thể hiện phần 1000 làm ví dụ của luồng bit làm ví dụ. Phần luồng bit 1000 bao gồm mào đầu lát phục hồi 1002, được biểu thị bởi hình chữ nhật màu đen đặc, bốn mào đầu lát entropi (mào đầu lát entropi tương ứng với lát entropi thứ không 1003, gọi là mào đầu lát entropi thứ không, mào đầu lát entropi tương ứng với lát entropi thứ nhất 1004, gọi là mào đầu lát entropi thứ nhất, mào đầu lát entropi tương ứng với lát entropi thứ hai 1005, gọi là mào đầu lát entropi thứ hai, mào đầu lát entropi tương ứng với lát entropi thứ ba 1006, gọi là mào đầu lát entropi thứ ba), được biểu thị bởi các hình chữ nhật màu xám đặc, và các phần còn lại của các lát entropi, được biểu thị bằng nét vẽ mảnh, các dải đen và trắng trong ví dụ này. Trong ví dụ này, vị trí tham chiếu có thể bắt đầu 1001 của mào đầu lát phục hồi 1002. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi tương ứng với lát entropi thứ không 1003 có thể được ràng buộc cần được đặt ngay sau mào đầu lát phục hồi 1002. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi tương ứng với lát entropi thứ không là một phần của phần mào đầu lát phục hồi. Nói cách khác, mào đầu lát phục hồi có thể còn dùng làm mào đầu lát entropi tương ứng với lát entropi thứ không. Theo các phương án này, mào đầu lát phục hồi có thể bao gồm phần phục hồi và phần entropi. Theo một số phương án thực hiện sáng chế, như được thể hiện trên Fig.29, mào đầu lát entropi thứ nhất 1004 có thể được ràng buộc cần được đặt sau $k - p$ bit 1007 tính từ vị trí tham chiếu 1001, mào đầu lát entropi thứ hai 1005 có thể được ràng buộc cần được đặt sau $2k - p$ bit 1008 tính từ vị trí tham chiếu 1001, mào đầu lát entropi thứ hai 1006 có thể được ràng buộc cần được đặt sau $3k - p$ bit 1009 tính từ vị trí tham chiếu 1001. Theo các phương án này, bộ giải mã entropi được gán để giải mã lát

entropi N có thể bắt đầu tìm kiếm đối với mào đầu lát entropi tương ứng sau $Nk - p$ bit tính từ vị trí tham chiếu 1001.

Theo các phương án thực hiện khác của sáng chế, thông tin vị trí lát entropi có thể không bao gồm tham số phạm vi. Theo các phương án này, bộ giải mã entropi có thể bắt đầu tìm kiếm đối với mào đầu lát entropi thứ N sau Nk bit tính từ vị trí tham chiếu.

Theo các phương án làm ví dụ được minh họa trên Fig.30, mào đầu lát entropi được kết hợp với lát entropi, số lượng lát N trong lát phục hồi, có thể được ràng buộc để bắt đầu sau $Nk - p$ bit tính từ vị trí bắt đầu, hoặc vị trí cố định khác trong đó, mào đầu lát phục hồi, trong đó k biểu thị giá trị dịch vị và p biểu thị phạm vi, và mào đầu lát entropi còn được ràng buộc vào trong phạm vi $2p$ của các bit tính từ vị trí bắt đầu ràng buộc. Vị trí tính từ đó $Nk - p$ bit có thể được đo có thể được gọi là vị trí tham chiếu. Theo các phương án thực hiện khác, vị trí tham chiếu có thể không được kết hợp với lát phục hồi nhất định và có thể là cùng vị trí cố định trong luồng bit đối với tất cả các lát entropi. Theo các phương án thực hiện khác, mào đầu lát entropi có thể là byte được sắp xếp, và ràng buộc có thể được kết hợp với số lượng byte. Mặc dù ví dụ này, như được minh họa trên Fig.30, mô tả theo bit, nhưng người có hiểu biết trung bình trong lĩnh vực kỹ thuật có liên quan có thể hiểu rằng các phương án thay thế khác sử dụng byte có thể được thực hiện.

Fig.30 là hình vẽ thể hiện phần 1020 của luồng bit làm ví dụ. Phần luồng bit 1020 bao gồm mào đầu lát phục hồi 1022, được biểu thị bởi hình chữ nhật màu đen đặc, bốn mào đầu lát entropi (mào đầu lát entropi tương ứng với lát entropi thứ không 1023, gọi là mào đầu lát entropi thứ không, mào đầu lát entropi tương ứng với lát entropi thứ nhất 1024, gọi là mào đầu lát entropi thứ nhất, mào đầu lát entropi tương ứng với lát entropi thứ hai 1025, gọi là mào đầu lát entropi thứ hai, mào đầu lát entropi tương ứng với lát entropi thứ ba 1026, gọi là mào đầu lát entropi thứ ba), được biểu thị bởi các hình chữ nhật màu xám, và các phần còn lại của các lát entropi, được biểu thị bởi nét vẽ mảnh, các dải đen và trắng. Trong ví dụ này, vị trí tham chiếu có thể là phần bắt đầu 1021 của mào đầu lát phục hồi 1022. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi tương ứng với lát entropi thứ không 1023 có thể được ràng buộc cần được đặt ngay sau mào

đầu lát phục hồi 1022. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi tương ứng với lát entropi thứ không là một phần của mào đầu lát phục hồi. Theo các phương án này, mào đầu lát phục hồi có thể bao gồm phần phục hồi và phần entropi. Theo một số phương án thực hiện sáng chế, như được thể hiện trên Fig.30, mào đầu lát entropi thứ nhất 1024 có thể được ràng buộc cần được đặt trong $2p$ bit 1031 sau $k - p$ bit 1027 tính từ vị trí tham chiếu 1021, mào đầu lát entropi thứ hai 1025 có thể được ràng buộc cần được đặt trong $2p$ bit 1032 sau $2k - p$ bit 1028 tính từ vị trí tham chiếu 1021, mào đầu lát entropi thứ hai 1026 có thể được ràng buộc cần được đặt trong $2p$ bit 1033 sau $3k - p$ bit 1029 tính từ vị trí tham chiếu 1021. Theo các phương án này, bộ giải mã entropi được gán để giải mã lát entropi N có thể bắt đầu tìm kiếm đối với mào đầu lát entropi tương ứng sau $Nk - p$ bit tính từ vị trí tham chiếu và có thể kết thúc việc tìm kiếm sau khi nhận dạng mào đầu lát entropi hoặc sau khi tìm kiếm $2p$ bit.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.31. Theo các phương án này, bộ giải mã entropi có thể thu ở bước 1050 số lượng lát entropi chỉ báo số lượng của lát entropi trong khối hiện hành phục hồi để giải mã entropi. Bộ giải mã entropi có thể xác định ở bước 1052 thông tin vị trí lát entropi. Theo một số phương án thực hiện sáng chế, thông tin vị trí lát entropi, còn gọi là các tham số vị trí lát entropi, có thể được báo hiệu trong luồng bit, và bộ giải mã có thể xác định ở bước 1052 thông tin lát entropi bằng cách kiểm tra luồng bit. Theo các phương án thực hiện khác, thông tin vị trí lát entropi có thể không được báo hiệu trong luồng bit, nhưng có thể được xác định ở bước 1052, bởi bộ giải mã, từ các tham số lát entropi, chẳng hạn số lượng bin cố định được phép trong lát entropi định trước bất kỳ, số lượng các bit không đổi được phép trong lát entropi định trước bất kỳ và các tham số lát entropi khác. Theo các phương án thực hiện sáng chế, thông tin vị trí lát entropi có thể được xác định và được xác định ở bước 1052 bởi phương tiện thông thường khác, chẳng hạn thông tin này có thể được định rõ trong ràng buộc biên dạng, ràng buộc mức độ, ràng buộc ứng dụng, hoặc ràng buộc khác, hoặc thông tin này có thể được báo hiệu là thông tin bổ sung hoặc được báo hiệu bởi phương tiện bên ngoài khác.

Bộ giải mã entropi có thể tính ở bước 1054 vị trí bắt đầu tìm kiếm lát entropi

trước đó, trong luồng bit, mào đầu lát entropi được giới hạn từ khi được ghi bởi mã hóa. Theo một số phương án thực hiện sáng chế, vị trí bắt đầu tìm kiếm lát entropi có thể được tính ở bước 1054 bằng cách sử dụng giá trị dịch vị và giá trị phạm vi được xác định từ thông tin vị trí lát entropi. Theo các phương án thực hiện khác của sáng chế, vị trí bắt đầu tìm kiếm lát entropi có thể được tính ở bước 1054 bằng cách sử dụng giá trị dịch vị được xác định từ thông tin vị trí lát entropi. Bộ giải mã entropi có thể dịch chuyển ở bước 1056, trong luồng bit, đến vị trí bắt đầu tìm kiếm lát entropi, và có thể kiểm tra ở bước 1058 luồng bit đối với mào đầu lát entropi. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi có thể được biểu thị bởi mã bắt đầu.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.32. Theo các phương án này, bộ giải mã entropi có thể thu ở bước 1070 số lượng lát entropi chỉ báo số lượng của lát entropi trong khối hiện hành phục hồi để giải mã entropi. Bộ giải mã entropi có thể xác định ở bước 1072 thông tin vị trí lát entropi. Theo một số phương án thực hiện sáng chế, thông tin vị trí lát entropi, còn gọi là các tham số vị trí lát entropi, có thể được báo hiệu trong luồng bit, và bộ giải mã có thể xác định ở bước 1072 thông tin lát entropi bằng cách kiểm tra luồng bit. Theo các phương án thực hiện khác, thông tin vị trí lát entropi có thể không được báo hiệu trong luồng bit, nhưng có thể được xác định ở bước 1072, bởi bộ giải mã, từ các tham số lát entropi, chẳng hạn số lượng bin cố định được phép trong lát entropi định trước bất kỳ, số lượng các bit không đổi được phép trong lát entropi định trước bất kỳ và các tham số lát entropi khác. Theo các phương án thực hiện sáng chế, thông tin vị trí lát entropi có thể được xác định và được xác định ở bước 1072 bởi các phương tiện thông thường khác, chẳng hạn thông tin này có thể được định rõ trong ràng buộc biên dạng, ràng buộc mức độ, ràng buộc ứng dụng, hoặc ràng buộc khác, hoặc thông tin này có thể được báo hiệu là thông tin bổ sung hoặc được báo hiệu bởi các phương tiện bên ngoài khác.

Bộ giải mã entropi có thể tính ở bước 1074 vị trí bắt đầu tìm kiếm lát entropi trước đó, trong luồng bit, mào đầu lát entropi được giới hạn từ khi được ghi bởi bộ mã hóa. Theo một số phương án thực hiện sáng chế, vị trí bắt đầu tìm kiếm lát entropi có thể được tính ở bước 1074 bằng cách sử dụng giá trị dịch vị và giá trị

phạm vi được xác định từ thông tin vị trí lát entropi. Theo các phương án thực hiện khác của sáng chế, vị trí bắt đầu tìm kiếm lát entropi có thể được tính ở bước 1074 bằng cách sử dụng giá trị dịch vị được xác định từ thông tin vị trí lát entropi. Bộ giải mã entropi có thể dịch chuyển ở bước 1076, trong luồng bit, đến vị trí bắt đầu tìm kiếm lát entropi và có thể kiểm tra ở bước 1078 luồng bit đối với mào đầu lát entropi. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi có thể được chỉ báo bởi mã bắt đầu.

Các bit, trong luồng bit, có thể được kiểm tra ở bước 1078 theo chuỗi bắt đầu ở vị trí bắt đầu tìm kiếm lát entropi. Nếu ở bước 1080 mào đầu lát entropi được nhận dạng 1081, thì bộ giải mã entropi có thể giải mã entropi ở bước 1082 lát entropi được kết hợp với mào đầu lát entropi được nhận dạng. Nếu ở bước 1080 mào đầu lát entropi không được nhận dạng ở bước 1083, thì bộ giải mã entropi có thể kết thúc ở bước 1084 tìm kiếm. Theo một số phương án, bộ giải mã entropi có thể chỉ báo lỗi khi không có mào đầu lát entropi được nhận dạng 1083.

Một số phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.33. Theo các phương án này, bộ giải mã entropi có thể thu ở bước 1100 số lượng lát entropi chỉ báo số lượng của lát entropi, trong khôi phục hồi hiện hành để giải mã entropi. Bộ giải mã entropi có thể xác định ở bước 1102 thông tin vị trí lát entropi. Theo một số phương án thực hiện sáng chế, thông tin vị trí lát entropi, còn gọi là các tham số vị trí lát entropi, có thể được báo hiệu trong luồng bit, và bộ giải mã có thể xác định ở bước 1102 thông tin lát entropi bằng cách kiểm tra luồng bit. Theo các phương án thực hiện khác, thông tin vị trí lát entropi có thể không được báo hiệu trong luồng bit, nhưng có thể được xác định ở bước 1102, bởi bộ giải mã, từ các tham số lát entropi, chẳng hạn số lượng bin cố định được phép trong lát entropi định trước bất kỳ, số lượng các bit không đổi được phép trong lát entropi định trước bất kỳ và các tham số lát entropi khác. Theo các phương án thực hiện sáng chế, thông tin vị trí lát entropi có thể được xác định và được xác định ở bước 1102 bởi các phương tiện thông thường khác, chẳng hạn thông tin này có thể được định rõ trong ràng buộc biên dạng, ràng buộc mức độ, ràng buộc ứng dụng, hoặc ràng buộc khác, hoặc thông tin này có thể được báo hiệu là thông tin bổ sung hoặc được báo hiệu bởi các phương tiện bên ngoài khác.

Bộ giải mã entropi có thể tính ở bước 1104 vị trí bắt đầu tìm kiếm lát entropi trước đó, trong luồng bit, mào đầu lát entropi được giới hạn từ khi được ghi bởi bộ mã hóa. Theo một số phương án thực hiện sáng chế, vị trí bắt đầu tìm kiếm lát entropi có thể được tính ở bước 1104 bằng cách sử dụng giá trị dịch vị và giá trị phạm vi được xác định từ thông tin vị trí lát entropi. Theo các phương án thực hiện khác của sáng chế, vị trí bắt đầu tìm kiếm lát entropi có thể được tính ở bước 1104 bằng cách sử dụng giá trị dịch vị được xác định từ thông tin vị trí lát entropi. Bộ giải mã entropi có thể dịch chuyển ở bước 1106, trong luồng bit, đến vị trí bắt đầu tìm kiếm lát entropi và có thể kiểm tra ở bước 1108 luồng bit đối với mào đầu lát entropi. Theo một số phương án thực hiện sáng chế, mào đầu lát entropi có thể được chỉ báo bởi mã bắt đầu.

Các bit, trong luồng bit, có thể được kiểm tra ở bước 1108 khi bắt đầu chuỗi ở vị trí bắt đầu tìm kiếm lát entropi. Nếu ở bước 1110 mào đầu lát entropi được nhận dạng 1111, thì bộ giải mã entropi có thể giải mã entropi ở bước 1112 lát entropi được kết hợp với mào đầu lát entropi được nhận dạng. Nếu ở bước 1110 mào đầu lát entropi không được nhận dạng ở bước 1113, thì nếu ở bước 1114 tiêu chuẩn tìm kiếm được thỏa mãn ở bước 1115, bộ giải mã entropi có thể kết thúc ở bước 1116. Tiêu chuẩn tìm kiếm có thể cung cấp tiêu chuẩn theo đó việc xác định có thể được thực hiện để xét xem có hay không các vị trí hợp lệ để bắt đầu mào đầu lát entropi còn lại cần được tìm kiếm. Theo một số phương án thực hiện (không được thể hiện), tiêu chuẩn tìm kiếm có thể được thỏa mãn nếu các vị trí hợp lệ còn lại được kiểm tra. Theo các phương án thực hiện khác, tiêu chuẩn tìm kiếm có thể được thỏa mãn nếu không có các vị trí hợp lệ còn lại cần được kiểm tra ở bước 1115, và quá trình tìm kiếm có thể kết thúc ở bước 1116. Theo một số phương án, bộ giải mã entropi có thể chỉ báo lỗi khi không có mào đầu lát entropi được nhận dạng ở bước 1115. Nếu ở bước 1114 tiêu chuẩn tìm kiếm không được thỏa mãn ở bước 1117, việc kiểm tra ở bước 1108 luồng bit có thể được tiếp tục sau khi dịch chuyển ở bước 1118, trong luồng bit đến vị trí tìm kiếm tiếp theo.

Theo một số phương án thực hiện sáng chế, tiêu chuẩn tìm kiếm có thể liên quan đến giá trị phạm vi, chẳng hạn vị trí bắt đầu của mào đầu lát entropi có thể được hạn chế vào phạm vi $2p$ bit được định tâm ở Nk , trong đó k biểu thị giá trị

dịch vị, p biểu thị giá trị phạm vi và N lát entropi trong lát phục hồi. Theo các phương án này, vị trí của bắt đầu của mào đầu lát entropi được kết hợp với lát entropi N có thể được giới hạn nằm trong khoảng từ $Nk - p$ đến $Nk + p$. Theo một số phương án, tiêu chuẩn tìm kiếm có thể liên quan đến giới hạn hoặc các giới hạn, về kích thước của lát entropi. Theo một số phương án, tiêu chuẩn tìm kiếm có thể liên quan đến sự kết hợp các giới hạn.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể tạo đệm lát entropi để đáp ứng giới hạn về vị trí của mào đầu lát entropi tiếp theo.

Theo một số phương án thực hiện sáng chế, bộ mã hóa có thể kết thúc lát entropi trước khi các giới hạn kích thước lát entropi khác sẽ đáp ứng để đáp ứng giới hạn về vị trí của mào đầu lát entropi tiếp theo.

Theo một số phương án thực hiện sáng chế, khi lát entropi sau cùng trong lát phục hồi không chứa số lượng các bit (hoặc các byte, theo phương án sắp xếp byte) cần thỏa mãn ràng buộc về vị trí của mào đầu lát entropi tiếp theo, bộ mã hóa có thể tạo đệm lát entropi sau cùng trong lát phục hồi để thỏa mãn ràng buộc về vị trí của mào đầu lát entropi tiếp theo.

Theo các phương án thực hiện khác, mào đầu lát entropi có thể bao gồm cờ last-entropy-slice, trong đó giá trị của cờ last-entropy-slice có thể chỉ báo xem có hay không lát entropi được kết hợp với mào đầu lát entropi là lát entropi sau cùng trong lát phục hồi. Theo một số phương án, giá trị cờ last-entropy-slice là không có thể được kết hợp với lát entropi sau cùng. Theo các phương án thực hiện khác, giá trị cờ last-entropy-slice là một có thể được kết hợp với lát entropi sau cùng. Theo một số phương án, khi giá trị của cờ last-entropy-slice chỉ báo rằng lát entropi là lát entropi sau cùng trong lát phục hồi, thì mào đầu lát entropi tiếp theo có thể được đặt ngay sau lát entropi hiện hành không cần tạo đệm.

Bảng 7 thể hiện cú pháp làm ví dụ và ngữ nghĩa để báo hiệu cờ last-entropy-slice, gọi là “next_entropy_slice_flag.” theo phương án làm ví dụ bao gồm cú pháp làm ví dụ và ngữ nghĩa làm ví dụ được thể hiện trong bảng 7, cờ “next_entropy_slice_flag” báo hiệu nếu có các lát entropi bổ sung đối với lát phục hồi hiện hành. Nếu cờ “next_entropy_slice_flag” chỉ báo rằng không có các lát entropi bổ sung đối với lát phục hồi hiện hành, thì vị trí của mào đầu lát entropi

tiếp theo trong luồng bit có thể không được ràng buộc bởi các tham số vị trí lát entropi.

Theo một số phương án thực hiện sáng chế, vị trí của các mào đầu lát entropi có thể được xác định theo định dạng cây với nút gốc trở đến vị trí mào đầu lát entropi. Theo một số phương án, vị trí mào đầu lát entropi được trở bởi nút gốc có thể là tương đối. Theo các phương án thực hiện khác, vị trí mào đầu lát entropi được trở bởi nút gốc có thể là tuyệt đối. Các nút còn lại của cây này các rãnh chứa các khoảng cách dịch vị so với nút cha mẹ của chúng. Cây này có thể được định rõ ràng buộc, chẳng hạn để giảm thời gian trung bình để xác định vị trí mào đầu lát entropi, giới hạn thời gian có trường hợp xấu nhất cần để xác định vị trí mào đầu lát entropi, để báo hiệu thứ tự ưu tiên của lát giải mã entropi, để tối thiểu hóa chi phí lưu trữ đối với cây này và các ràng buộc thiết kế khác. Theo một số phương án, số lượng nút con của mỗi nút trong cây này có thể được điều khiển dựa vào mức độ mong muốn của độ song song ở khi xác định vị trí mào đầu lát entropi.

slice_header() {	C	Phần tử mô tả
entropy_slice_flag	2	u(1)
next_entropy_slice_flag	2	ue(v)
if (entropy_slice_flag) {		
first_mb_in_slice	2	ue(v)
if(entropy_coding_mode_flag && slice_type != I && slice_type != SI)		
cabac_init_idc	2	ue(v)
}		
}		
else {		
mào đầu lát ổn định		
}		
}		

Bảng 7: Bảng cú pháp làm ví dụ đối với cờ last-entropy-slice

Theo một số phương án thực hiện sáng chế, các mô hình ngữ cảnh có thể

được thiết lập trong lát entropi mỗi khi điều kiện context-model-reset (thiết lập lại mô hình ngữ cảnh) được đáp ứng. Theo các phương án này, các giá trị theo đó các mô hình ngữ cảnh có thể thiết lập lại có thể được dựa vào mô hình ngữ cảnh đơn vị cơ bản lân cận trong lát entropi, và nếu đơn vị cơ bản lân cận không có trong lát entropi, thì các giá trị mặc định có thể được sử dụng. Theo các phương án thực hiện khác, các mô hình ngữ cảnh có thể được thiết lập vào các giá trị mặc định. Theo các phương án thực hiện khác của sáng chế, các mô hình ngữ cảnh có thể được thiết lập dựa vào mô hình ngữ cảnh mà phần tử nhận dạng của nó có thể được báo hiệu trong luồng bit, phần tử nhận dạng này chỉ báo một trong số các mô hình ngữ cảnh xác định trước. Mô hình ngữ cảnh xác định trước có thể tùy thuộc vào một hoặc nhiều tham số trong luồng bit. Theo các phương án làm ví dụ này, các mô hình ngữ cảnh có thể được thiết lập dựa vào giá trị “cabac_init_idc” được báo hiệu, trong luồng bit, chỉ báo một trong số các mô hình ngữ cảnh được xác định trước.

Theo một số phương án, bảng ngữ cảnh có thể được sử dụng để khởi tạo các mô hình ngữ cảnh, trong đó bảng ngữ cảnh nêu ra tập các mô hình ngữ cảnh. Theo một số phương án, tập các mô hình ngữ cảnh trong bảng ngữ cảnh có thể được làm thích ứng dựa vào một hoặc nhiều, tham số trong luồng bit, chẳng hạn tham số lượng tử hóa, tham số kiểu lát hoặc tham số khác.

Theo phương án thực hiện làm ví dụ được thể hiện trên Fig.34, các mô hình ngữ cảnh có thể được thiết lập lại, trong lát entropi, khi khởi hiện hành macrô là khối macrô thứ nhất theo hàng, ngoài việc thiết lập lại khi bắt đầu khối macrô trong lát entropi. Fig.34 minh họa lát phục hồi 1200 làm ví dụ chứa 48 các khối macrô 1208-1255 được phân chia thành ba lát entropi: lát entropi “0” (được thể hiện trong phần gạch chéo) 1202, lát entropi “1” (được thể hiện trong phần màu trắng) 1204 và lát entropi “2” (được thể hiện trong phần chấm) 1206. Lát entropi “0” 1202 chứa 15 các khối macrô 1208-1222. Lát entropi “1” 1204 chứa 17 các khối macrô 1223-1239, và lát entropi “2” 1206 chứa 16 các khối macrô 1240-1255. Các khối macrô ở đó các mô hình ngữ cảnh có thể được thiết lập lại được chỉ báo với biên màu đen đậm 1260-1266 và các khối macrô 1208, 1223, 1240 đó ở phần bắt đầu của mỗi lát entropi và khối macrô thứ nhất trong mỗi hàng 1216, 1224,

1232, 1240, 1248.

Đơn vị cơ bản, chẳng hạn khối macrô, ở phần bắt đầu của lát entropi có thể được gọi là đơn vị cơ bản bắt đầu lát (đơn vị cơ bản slice-start). Chẳng hạn, đối với các lát entropi 1202, 1204, 1206 trong lát phục hồi 1200 làm ví dụ trên Fig.34, các đơn vị cơ bản slice-start tương ứng là 1208, 1223 và 1240. Đơn vị cơ bản đó là đơn vị cơ bản thứ nhất ở hàng trong lát entropi có thể gọi là đơn vị cơ bản bắt đầu hàng (đơn vị cơ bản row-start), chẳng hạn các khối macrô 1208, 1216, 1224, 1232, 1240 và 1248 trên Fig.34.

Theo một số phương án thực hiện sáng chế, các mô hình ngữ cảnh có thể được thiết lập lại dựa vào các mô hình ngữ cảnh của khối lân cận macrô nếu khối lân cận macrô có trong lát entropi và các giá trị mặc định nếu khối lân cận macrô không có trong lát entropi. Chẳng hạn, các mô hình ngữ cảnh có thể được thiết lập lại dựa vào các mô hình ngữ cảnh của khối macrô ở trên khối hiện hành macrô nếu khối macrô ở trên khối hiện hành macrô có trong cùng lát entropi, nhưng thiết lập vào các giá trị mặc định nếu khối macrô ở trên khối hiện hành macrô không có trong cùng một lát entropi.

Theo các phương án thực hiện sáng chế, các mô hình ngữ cảnh có thể được thiết lập lại, trong lát entropi, khi đơn vị cơ bản hiện hành là đơn vị cơ bản thứ nhất trong hàng. Theo các phương án thực hiện khác, điều kiện context-model-reset có thể được dựa vào tiêu chuẩn khác, chẳng hạn số lượng các bin được xử lý trong lát entropi, số lượng các bit được xử lý trong lát, vị trí theo không gian của đơn vị cơ bản hiện hành và tiêu chuẩn khác.

Theo một số phương án thực hiện sáng chế, cờ context-model-reset có thể được sử dụng để chỉ báo xem có hay không các mô hình ngữ cảnh có thể được thiết lập lại trong lát entropi mỗi khi điều kiện context-model-reset được đáp ứng. Theo một số phương án, cờ context-model-reset có thể có trong mào đầu lát entropi. Theo các phương án thực hiện khác, cờ context-model-reset có thể có trong mào đầu lát phục hồi. Theo một số phương án, cờ context-model-reset có thể có trong cờ nhị phân, và điều kiện context-model-reset có thể là điều kiện mặc định. Theo các phương án thực hiện khác, cờ context-model-reset có thể là cờ đa trị còn chỉ báo điều kiện context-model-reset.

Theo một phương án thực hiện, sáng chế bao gồm mã hóa context-adaptive (mã hóa thích ứng ngữ cảnh), chẳng hạn mã hóa CABAC, mã hóa CAV2V và mã hóa context-adaptive khác, cờ “`lcu_row_cabac_init_flag`” có thể là tín hiệu nếu giải mã entropi có thể được khởi tạo ở phần đầu của hàng bộ mã hóa lớn nhất (LCU). Theo một số phương án, LCU là phần suy rộng của khối macro được sử dụng trong H.264 để mã hóa video hiệu suất cao (HEVC), và ảnh được phân chia thành các lát, trong đó lát được tạo ra theo chuỗi các LCU. Theo các phương án thực hiện khác, LCU là khối lớn nhất có các vị trí mang giá trị điểm ảnh mà có thể được biểu thị bằng giá trị chế độ được truyền, đơn. Theo các phương án thực hiện khác, LCU là khối lớn nhất của các vị trí mang giá trị điểm ảnh mà có thể được biểu thị bằng giá trị chế độ dự báo được truyền, đơn. Theo một số phương án thực hiện sáng chế, giá trị cờ “`lcu_row_cabac_init_flag`” là “1” có thể báo hiệu rằng ngữ cảnh mã hóa entropi được thiết lập lại. Ngữ cảnh mã hóa entropi có thể biểu thị tập của tất cả các mô hình ngữ cảnh được kết hợp với bộ mã hóa entropi. Theo một số phương án thực hiện sáng chế, giá trị cờ “`lcu_row_cabac_init_flag`” là “1” có thể báo hiệu rằng ngữ cảnh mã hóa entropi được thiết lập lại và quá trình quét thích ứng được thiết lập lại. Quá trình quét thích ứng có thể gọi là quá trình trong đó bộ mã hóa/giải mã được làm thích ứng với thứ tự quét của các hệ số biến đổi dựa vào các giá trị của hệ số biến đổi được biến đổi trước đó. Phần 7.6.1 trong tài liệu JCTVC JCTVC-B205_draft005, được kết hợp trích dẫn trong phần mô tả dưới đây nhằm tham khảo, có nêu ví dụ trong đó việc quét thích ứng chọn giữa hai thứ tự quét phân biệt dựa vào các hệ số đáng kể trong phần lân cận. Theo một phương án, việc quét thích ứng có thể được thiết lập lại ở phần bắt đầu của mọi hàng LCU bằng cách chọn thứ tự quét xác định trước. Theo một phương án, thứ tự quét được xác định bằng cách tạo ra sơ đồ ánh xạ hệ số có ý nghĩa, và các giá trị của hệ số biến đổi tương ứng với các giá trị hệ số có ý nghĩa lớn hơn so với giá trị được xác định trước có thể được truyền trước cho các giá trị của hệ số biến đổi tương ứng với các giá trị hệ số có ý nghĩa nhỏ hơn hoặc bằng giá trị được xác định trước. Theo một phương án, các giá trị hệ số có ý nghĩa tương ứng với các giá trị của hệ số biến đổi lớn hơn giá trị được xác định trước có thể tăng sau đó. Theo phương án thực hiện khác của sáng chế, các giá trị hệ số có ý nghĩa tương ứng với các giá trị của hệ số

biến đổi nhỏ hơn hoặc bằng giá trị được xác định trước có thể giảm sau đó. Quá trình xử lý quét thích ứng có thể được thiết lập lại bằng cách thiết lập sơ đồ ánh xạ hệ số có ý nghĩa tới các giá trị xác định trước. Theo một số phương án, giá trị mặc định, được tiếp tục khi còn không được gửi, đối với cờ “`lcu_row_cabac_init_flag`” có thể là “0.” cờ “`lcu_row_cabac_init_idc_flag`” có thể báo hiệu nếu các giá trị `cabac_init_idc` sẽ được truyền ở phần bắt đầu của mỗi hàng LCU. Theo một số phương án, khi giá trị của cờ “`lcu_row_cabac_init_idc_flag`” là các giá trị “1” thì sẽ được truyền ở phần bắt đầu của mỗi hàng LCU. Theo một số phương án, giá trị mặc định, được thừa nhận khi cờ này không được gửi, đối với cờ “`lcu_row_cabac_init_idc_flag`” có thể là “0”. Theo một phương án thực hiện, cờ “`cabac_init_idc_present_flag`” có thể báo hiệu nếu giá trị `cabac_init_idc` được truyền đối với LCU. Theo một số phương án, khi giá trị `cabac_init_idc` không được truyền đối với LCU thì ngữ cảnh mã hóa entropi được thiết lập lại bằng cách sử dụng giá trị trước đối với `cabac_init_idc` trong luồng bit. Theo một số phương án thực hiện sáng chế, “`lcu_row_cabac_init_flag`” và “`lcu_row_cabac_init_idc_flag`” có thể được báo hiệu trong mào đầu lát ổn định, chẳng hạn khi giá trị của “`entropy_slice_flag`” là “0”. Bảng 8 và bảng 9 thể hiện cho phép làm ví dụ đối với các phương án này. Theo một số phương án thực hiện sáng chế, “`lcu_row_cabac_init_flag`” và “`lcu_row_cabac_init_idc_flag`” có thể được báo hiệu trong mào đầu lát entropi, chẳng hạn khi giá trị của “`entropy_slice_flag`” là “1”. Bảng 8 thể hiện cho phép mào đầu lát làm ví dụ, và bảng 9 thể hiện cho phép dữ liệu lát làm ví dụ (`coding_unit`).

<code>slice_header() {</code>	C	Phần tử mô tả
<code>entropy_slice_flag</code>	2	u(1)
<code>if(entropy_slice_flag) {</code>		
<code>first_lcu_in_slice</code>	2	ue(v)
<code>(entropy_coding_mode_flag) {</code>		
<code>lcu_row_cabac_init_flag</code>	1	u(1)
<code>if(lcu_row_cabac_init_flag){</code>		
<code>lcu_row_cabac_init_idc_flag</code>	1	u(1)

}		
}		
if(entropy_coding_mode_flag && slice_type != I) {		
cabac_init_idc	2	ue(v)
}		
}		
else {		
lcu_row_cabac_init_flag	1	u(1)
if(lcu_row_cabac_init_flag){		
lcu_row_cabac_init_idc_flag	1	u(1)
}		
mào đầu lát ổn định		
}		
}		

Bảng 8: Bảng cho phép làm ví dụ để báo hiệu khởi tạo việc mã hóa entropi ở phần bắt đầu của hàng LCU.

coding_unit(x0, y0, currCodingUnitSize) {	C	Phần tử mô tả
(x0==0 && currCodingUnitSize==MaxCodingUnitSize && lcu_row_cabac_init_idc_flag==true && lcu_id!=first_lcu_in_slice) {		
cabac_init_idc_present_flag	1	u(1)
if(cabac_init_idc_present_flag)		
cabac_init_idc	2	ue(v)
}		
regular coding unit ...		
}		

Bảng 9: Bảng cho phép làm ví dụ để báo hiệu ngữ cảnh khởi tạo đối với LCU

Theo phương án thực hiện khác, sáng chế bao gồm mã hóa thích ứng ngữ

cảnh, chẳng hạn mã hóa CABAC, mã hóa CAV2V và mã hóa thích ứng ngữ cảnh khác, cờ “mb_row_cabac_init_flag” có thể báo hiệu nếu giải mã entropi có thể được khởi tạo ở khối macro thứ nhất trong hàng. Theo một số phương án thực hiện sáng chế, giá trị cờ “mb_row_cabac_init_flag” là “1” có thể báo hiệu rằng ngữ cảnh mã hóa entropi được thiết lập lại ở phần bắt đầu của mỗi hàng khối macro. Theo các phương án thực hiện khác của sáng chế, giá trị cờ “mb_row_cabac_init_flag” là “1” có thể báo hiệu rằng ngữ cảnh mã hóa entropi được thiết lập lại và việc quét thích ứng được thiết lập lại ở phần bắt đầu của mỗi hàng khối macro. Theo một số phương án, giá trị mặc định, được thừa nhận khi cờ không được gửi, đối với cờ “mb_row_cabac_init_flag” là “0.” cờ “mb_row_cabac_init_idc_flag” có thể báo hiệu rằng nếu các giá trị cabac_init_idc sẽ được truyền ở phần bắt đầu của mỗi hàng khối macro. Theo một số phương án, khi giá trị của cờ “mb_row_cabac_init_idc_flag” là các giá trị “1” sẽ được truyền ở phần bắt đầu của mỗi hàng khối macro. Theo một số phương án, giá trị mặc định, được thừa nhận khi cờ này không được gửi, đối với cờ “mb_row_cabac_init_idc_flag” là “0”. Theo một số phương án thực hiện sáng chế, cờ “cabac_init_idc_present_flag” có thể báo hiệu nếu giá trị cabac_init_idc được truyền đối với khối macro. Theo một số phương án, khi giá trị cabac_init_idc không được truyền cho khối macro, thì ngữ cảnh mã hóa entropi được thiết lập lại bằng cách sử dụng giá trị trước đối với cabac_init_idc trong luồng bit. Theo một số phương án thực hiện sáng chế, cờ “mb_row_cabac_init_flag” và cờ “mb_row_cabac_init_idc_flag” có thể được báo hiệu trong mào đầu lát ổn định, chẳng hạn khi giá trị của “entropy_slice_flag” là “0”. Theo một số phương án thực hiện sáng chế, cờ “mb_row_cabac_init_flag” và cờ “mb_row_cabac_init_idc_flag” có thể được báo hiệu trong mào đầu lát entropi, chẳng hạn khi giá trị của “entropy_slice_flag” là “1”. Bảng 10 và bảng 11 thể hiện cho phép làm ví dụ đối với các phương án thực hiện này. Bảng 10 thể hiện cho phép mào đầu lát làm ví dụ, và bảng 11 thể hiện cho phép dữ liệu lát làm ví dụ (coding_unit).

slice_header() {	C	Phần tử mô tả
entropy_slice_flag	2	u(1)

if(entropy_slice_flag) {		
first_mb_in_slice	2	ue(v)
(entropy_coding_mode_flag) {		
mb_row_cabac_init_flag	1	u(1)
if(mb_row_cabac_init_flag){		
mb_row_cabac_init_idc_flag	1	u(1)
}		
}		
if(entropy_coding_mode_flag && slice_type != I) {		
cabac_init_idc	2	ue(v)
}		
}		
else {		
mb_row_cabac_init_flag	1	u(1)
if(mb_row_cabac_init_flag){		
mb_row_cabac_init_idc_flag	1	u(1)
}		
mào đầu lát ổn định		
}		
}		

Bảng 10: Bảng cho phép làm ví dụ để báo hiệu việc khởi tạo mã hóa entropi ở phần bắt đầu của hàng khối macro

coding_unit(x0, y0, currCodingUnitSize) {	C	Phần tử mô tả
(x0==0 && currCodingUnitSize==MaxCodingUnitSize && mb_row_cabac_init_idc_flag==true && mb_id!=first_mb_in_slice) {		
cabac_init_idc_present_flag	1	u(1)
if(cabac_init_idc_present_flag)		
cabac_init_idc	2	ue(v)

}		
regular coding unit ...		
}		

Bảng 11: Bảng cho phép làm ví dụ để báo hiệu ngữ cảnh khởi tạo cho khối macro

Theo một số phương án thực hiện sáng chế, các vị trí, trong luồng bit, của các lát entropi có thể được báo hiệu trong luồng bit. Theo một số phương án, còn này có thể được dùng để báo hiệu rằng các vị trí, trong luồng bit, của các lát entropi tiếp tục cần được báo hiệu trong luồng bit. Theo các phương án thực hiện, sáng chế có thể bao gồm “entropy_slice_locations_flag” nếu “true” (đúng) có thể chỉ báo rằng các vị trí, trong luồng bit, của các mào đầu lát entropi cần được báo hiệu tiếp trong luồng bit. Theo một số phương án, dữ liệu vị trí có thể được mã hóa theo cách khác nhau. Theo một số phương án, dữ liệu vị trí có thể được gửi trong mỗi lát phục hồi. Theo các phương án thực hiện khác, dữ liệu vị trí có thể được gửi một lần trên ảnh.

Theo một số phương án thực hiện sáng chế, các vị trí, trong luồng bit, của các hàng LCU có thể được báo hiệu trong luồng bit. Theo một số phương án, còn này có thể được sử dụng để báo hiệu rằng vị trí, trong luồng bit, của LCU thứ nhất trong mỗi hàng cần được báo hiệu tiếp trong luồng bit. Theo phương án thực hiện, sáng chế có thể bao gồm “lcu_row_location_flag” nếu “true” (đúng) có thể chỉ báo rằng vị trí, trong luồng bit, của LCU thứ nhất trong mỗi hàng cần được báo hiệu tiếp trong luồng bit. Theo một số phương án, dữ liệu vị trí có thể được mã hóa theo cách khác. Theo một số phương án, dữ liệu vị trí có thể được gửi trong mỗi lát entropi. Theo các phương án thực hiện khác, dữ liệu vị trí có thể được gửi một lần trên lát phục hồi.

Bảng 12 thể hiện cú pháp làm ví dụ để báo hiệu các vị trí, trong luồng bit, của các hàng LCU và các lát entropi. Đối với cú pháp làm ví dụ này, ngữ nghĩa như sau:

Các tín hiệu “entropy_slice_locations_flag” nếu vị trí mào đầu lát entropi được truyền. Nếu giá trị của “entropy_slice_locations_flag” được thiết lập là “1”, thì vị trí mào đầu lát entropi được truyền, trái lại nó không được truyền. giá trị mặc

định đối với "entropy_slice_locations_flag" là "0".

"num_of_entropy_slice_minus1" báo hiệu số lượng của các lát entropi trong lát phục hồi trừ 1.

"entropy_slice_offset [i]" chỉ báo dịch vị của lát entropi thứ i tính từ lát entropi trước đó.

"lcu_row_locations_flag" báo hiệu nếu thông tin vị trí hàng LCU được truyền hoặc không được truyền. Nếu giá trị của "lcu_row_locations_flag" là "1", thì thông tin vị trí hàng LCU được truyền, trái lại nó không được truyền. Giá trị mặc định đối với "lcu_row_locations_flag" là "0".

"num_of_lcu_rows_minus1" báo hiệu số lượng của các hàng LCU trong lát entropi trừ 1.

"lcu_row_offset [i]" chỉ báo dịch vị của hàng LCU thứ i tính từ hàng LCU trước đó.

Theo một số phương án thực hiện sáng chế, "lcu" trong bảng 12 có thể được thay thế bởi "khối macro". Chẳng hạn, "first_lcu_in_slice", "lcu_row_cabac_init_flag", "lcu_row_cabac_init_idc_flag", "lcu_row_locations_flag", "lcu_row_locations ()", "num_of_lcu_rows_minus1", và "lcu_row_offset[i]" trong bảng 12 có thể được thay bằng "first_mb_in_slice", "mb_row_cabac_init_flag", "mb_row_cabac_init_idc_flag", "mb_row_locations_flag", "mb_row_locations ()", "num_of_mb_rows_minus1", và "mb_row_offset[i]", một cách tương ứng.

slice_header() {	C	Phần tử mô tả
entropy_slice_flag	2	u(1)
(entropy_slice_flag) {		
first_lcu_in_slice	2	ue(v)
(entropy_coding_mode_flag) {		
lcu_row_cabac_init_flag	1	u(1)
if(lcu_row_cabac_init_flag){		
lcu_row_cabac_init_idc_flag	1	u(1)
lcu_row_locations_flag	1	u(1)

(lcu_row_locations_flag) {		
lcu_row_locations ()		
}		
}		
}		
if(entropy_coding_mode_flag && slice_type != I) {		
cabac_init_idc	2	ue(v)
}		
}		
else {		
entropy_slice_locations_flag	1	u(1)
(entropy_slice_locations_flag) {		
entropy_slice_locations()		
}		
(entropy_coding_mode_flag) {		
lcu_row_cabac_init_flag	1	u(1)
if(lcu_row_cabac_init_flag){		
lcu_row_cabac_init_idc_flag	1	u(1)
lcu_row_locations_flag	1	u(1)
(lcu_row_locations_flag) {		
lcu_row_locations ()		
}		
}		
}		
mào đầu lát ổn định		
}		
}		

entropy_slice_locations()	C	Phần tử mô tả
---------------------------	---	---------------

{		
num_entropy_slices_minus1	2	ue(v)
(i=0; i<num_of_entropy_slices_minus1; i++)		
entropy_slice_offset[i]	2	ue(v)
}		

lcu_row_locations()	C	Phần tử mô tả
{		
num_of_lcu_rows_minus1	2	ue(v)
(i=0; i<num_of_lcu_rows_minus1_slice; i++) {		
lcu_row_offset[i]	2	ue(v)
}		
}		

Bảng 12: Bảng cú pháp làm ví dụ để báo hiệu các vị trí, trong luồng bit, của LCU thứ nhất trong hàng.

Việc truyền có hiệu quả dữ liệu dư từ bộ mã hóa tới bộ giải mã có thể đạt được bằng cách báo hiệu vị trí của các hệ số biến đổi có giá trị không và các giá trị mức độ của các hệ số biến đổi khác không đối với đơn vị cơ bản, chẳng hạn khối macrô. Nhiều hệ thống mã hóa có thể cố gắng đặt các hệ số biến đổi có giá trị không ở phần cuối của dữ liệu dư đối với đơn vị cơ bản, do đó cho phép sử dụng mã “end-of-block” (khối cuối) sau khi hệ số biến đổi có ý nghĩa cuối cùng báo hiệu có hiệu quả rằng các giá trị của hệ số biến đổi còn lại là không.

Một số hệ thống mã hóa có thể tìm vết các vị trí của các hệ số biến đổi có giá trị không trong dữ liệu dư được truyền trước đó đối với đơn vị cơ bản được xử lý trước đó, mà có thể cho phép các vị trí với các hệ số biến đổi có giá trị không trước đó cần được truyền sau cùng trong dữ liệu dư sau đó. Theo cách khác, một số hệ thống mã hóa có thể tìm vết các vị trí của các hệ số biến đổi có giá trị khác không trong dữ liệu dư được truyền trước đó. Mặc dù các hệ thống này có thể cải thiện hiệu suất mã hóa, nhưng nó khiến cho cần giải mã toàn phần dữ liệu dư trước đó để

giải mã dữ liệu dư hiện hành do thực tế rằng việc mã hóa dữ liệu dư sử dụng các mô hình ngữ cảnh, còn gọi là các mô hình xác suất, được xác định bởi phân tử nhận dạng hệ số biến đổi mà có thể chỉ được xác định khi đã biết rằng các vị trí được nhận dạng cần được truyền ở phần cuối của dữ liệu dư.

Chẳng hạn, như được thể hiện trong bảng Fig.35, nếu việc thích ứng quét đã tạo ra thứ tự quét: $S = \{coeff_0, coeff_{10}, coeff_1, \dots\}$ for đối với quá trình xử lý mã hóa entropi được kết hợp với đơn vị cơ bản hiện hành, trong đó $coeff_i$ biểu thị hệ số biến đổi thứ i , thì ngữ cảnh, được ký hiệu là $ctxt_0$, tương ứng với $coeff_0$ cần được tìm nạp đối với để mã hóa hệ số biến đổi $coeff_0$. Tiếp theo, ngữ cảnh $ctxt_{10}$, tương ứng với $coeff_{10}$ cần được tìm nạp để mã hóa hệ số biến đổi $coeff_{10}$, và .v.v.. Vì thế, thứ tự theo thời gian để mã hóa các đơn vị cơ bản có thể là bắt buộc do cần phải biết thứ tự quét $S = \{coeff_0, coeff_{10}, coeff_1, \dots\}$, không thể thu được cho đến khi các đơn vị cơ bản trước đó đã được mã hóa.

Theo một số phương án thực hiện sáng chế, để cho phép mã hóa song song các lát entropi, việc quét thích ứng có thể được thiết lập lại cho thứ tự quét mặc định lát entropi ở đơn vị cơ bản slice-start của mỗi lát entropi, do đó cho phép tách riêng các lát entropi cần được mã hóa theo cách song song.

Theo một số phương án thực hiện sáng chế, thứ tự quét của việc xác định quét thích ứng có thể được thiết lập để biết trước, sau đây gọi là thứ tự quét, mặc định hàng, với đơn vị cơ bản row-start của mỗi hàng LCU trong lát entropi.

Theo các phương án thực hiện khác của sáng chế, thứ tự quét hệ số biến đổi khối và mô hình ngữ cảnh tương ứng, còn gọi là ngữ cảnh, có thể được tìm nạp để mã hóa hệ số biến đổi có thể được tách, do đó cho phép mã hóa song song. Theo các phương án này, hệ số biến đổi được đặt ở vị trí thứ nhất trong luồng bit có thể được kết hợp, dựa vào vị trí tương đối của nó với các hệ số biến đổi khác trong luồng bit, với ngữ cảnh được đặt tương ứng theo thứ tự tìm nạp ngữ cảnh. Theo các phương án này, thứ tự tìm nạp ngữ cảnh, được ký hiệu là $F = \{ctxt_A, ctxt_B, ctxt_C, \dots\}$, trong đó $ctxt_i$ biểu thị ngữ cảnh mà không được kết hợp với vị trí hệ số biến đổi trong miền biến đổi, nhưng hơn là được kết hợp với vị trí tương đối của hệ số biến đổi trong luồng bit, có thể được xác định trước. Vì thế, đối với thứ tự quét hệ số biến đổi $S = \{coeff_0, coeff_{10}, coeff_1, \dots\}$, quá trình xử lý mã hóa có thể mã hóa $coeff_0$ với

$ctxt_A$, $coeff_{10}$ với $ctxt_B$, $coeff_1$ với $ctxt_C$ và .v.v.. Theo các phương án này, quá trình xử lý mã hóa entropi có thể hoạt động một cách độc lập theo thứ tự quét. Một số bộ mã hóa theo phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.36. Bộ mã hóa có thể tìm nạp ở bước 1280 hệ số biến đổi tiếp theo cần được mã hóa và có thể tìm nạp ở bước 1282 ngữ cảnh tiếp theo, tính danh mục tìm nạp xác định trước của các ngữ cảnh. Hệ số biến đổi được tìm nạp có thể được mã hóa entropi ở bước 1284 bsc ngữ cảnh tìm nạp, và việc xác định ở bước 1286 có thể được thực hiện để xem có hay không các hệ số biến đổi có ý nghĩa còn lại để mã hóa. Nếu có 1287 hệ số biến đổi có ý nghĩa còn lại cần được mã hóa, thì hệ số biến đổi có ý nghĩa tiếp theo có thể được tìm nạp ở bước 1280, và quá trình xử lý này có thể được tiếp tục. Nếu không có 1289, thì quá trình xử lý này có thể kết thúc ở bước 1290. Một số bộ giải mã theo phương án thực hiện sáng chế sẽ được mô tả có dựa vào Fig.37. bộ giải mã có thể tìm nạp ở bước 1300 ngữ cảnh tiếp theo và giải mã entropi ở bước 1302 hệ số biến đổi có ý nghĩa tiếp theo từ luồng bit bằng cách sử dụng ngữ cảnh được tìm nạp. Hệ số biến đổi được giải mã có thể được lưu trữ ở bước 1304, và việc xác định ở bước 1306 có thể được thực hiện để xem có hay không các hệ số biến đổi có ý nghĩa còn lại cần được giải mã. Nếu có 1307, thì ngữ cảnh tiếp theo có thể được tìm nạp ở bước 1300, và quá trình xử lý này có thể được tiếp tục. Nếu không có 1309, thì quá trình xử lý này có thể kết thúc ở bước 1310.

Theo các phương án thực hiện khác của sáng chế, thứ tự quét hệ số có thể được hạn chế vào tập con của tất cả các kết hợp quét có thể và có thể được báo hiệu rõ ràng. Ở phần bắt đầu của lát entropi, thứ tự quét có thể được thiết lập thứ tự quét được báo hiệu. Theo một số phương án, thứ tự quét có thể được báo hiệu là cú pháp thông thường. Theo các phương án thực hiện khác, thứ tự quét có thể được báo hiệu bằng thông báo thông báo không quy chuẩn, chẳng hạn thông báo SEI hoặc thông báo không quy chuẩn khác.

Theo các phương án thực hiện khác của sáng chế, thứ tự quét hệ số có thể được hạn chế vào tập con của tất cả các kết hợp quét có thể và có thể được báo hiệu một cách rõ ràng. Ở phần đầu của hàng LCU trong lát entropi, thứ tự quét có thể được thiết lập cho thứ tự quét được báo hiệu. Theo một số phương án, thứ tự quét có thể được báo hiệu là cú pháp thông thường. Theo các phương án thực hiện

khác, thứ tự quét có thể được báo hiệu với thông báo không quy chuẩn, chẳng hạn thông báo SEI hoặc thông báo không quy chuẩn khác.

Theo các phương án thực hiện khác của sáng chế, ở phần bắt đầu lát entropi, thứ tự quét hệ số có thể được thiết lập cho thứ tự quét của đơn vị cơ bản được giải mã trước. Theo một số phương án, thứ tự quét có thể được thiết lập cho thứ tự quét được sử dụng theo đơn vị cơ bản nêu trên. Theo các phương án thực hiện khác, thứ tự quét có thể được thiết lập cho thứ tự quét được sử dụng theo đơn vị cơ bản nêu trên và phần bên phải.

Theo các phương án thực hiện khác của sáng chế, ở phần bắt đầu của hàng LCU trong lát entropi, thứ tự quét hệ số có thể được thiết lập cho thứ tự quét của đơn vị cơ bản được giải mã trước. Theo một số phương án, thứ tự quét có thể được thiết lập cho thứ tự quét được sử dụng theo đơn vị cơ bản nêu trên. Theo các phương án thực hiện khác, thứ tự quét có thể được thiết lập cho thứ tự quét được sử dụng theo đơn vị cơ bản nêu trên và phần bên phải.

Như được thể hiện trên Fig.43, theo một số phương án thực hiện sáng chế, phương pháp khởi tạo ngữ cảnh đối với các lát P có thể được áp dụng cho các lát B được dự báo chuyển tiếp, phương pháp có thể đạt được hiệu suất nén cao hơn do mức độ ngẫu nhiên lớn hơn được tạo cho các lát B và tính đa giả thiết của các dự báo B. Các lát tham chiếu được sử dụng theo lát B được dự báo chuyển tiếp luôn từ các khung/các ảnh trước đó theo thời gian như được phân biệt với lát B chính tắc trong đó việc tham chiếu có thể được chọn từ các khung/ảnh quá khứ và/hoặc đặc điểm theo thời gian. Vì thế, lát B được dự báo chuyển tiếp có thể bao gồm dữ liệu dư với đặc điểm thống kê khác với các đặc điểm đó của lát B chính tắc. Như được thể hiện trên Fig.43, phương pháp này có thể thu các lát B, các lát P, và các lát I và dịch vị kiểu của lát thu được, phương pháp khởi tạo ngữ cảnh được chọn. Ngoài ra, trong trường hợp của lát B được dự báo chuyển tiếp kỹ thuật khởi tạo ngữ cảnh đối với lát P được sử dụng. Cụ thể hơn là, các bước dưới đây được thực hiện:

4301: kiểu lát có thể được giải mã từ luồng bit;

4302: xác định xem có hay không kiểu lát là lát B;

4303: nếu kiểu lát là lát B (Đúng ở bước 4302), thì xác định xem lát này có

được dự báo chuyển tiếp hay không. Nếu lát được dự báo chuyển tiếp, thì ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ hai, trái lại ngữ cảnh được khởi tạo bởi phương pháp khởi tạo thứ nhất;

4304: nếu kiểu lát không là lát B (Sai ở bước 4302), thì xác định xem có hay không kiểu lát là lát P. Nếu kiểu lát là lát P, thì ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ hai; và

4305: nếu kiểu lát không là lát P (Sai ở bước 4304), thì xác định xem có hay không kiểu lát là lát I. Nếu kiểu lát là lát I, thì ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ ba.

Theo khía cạnh của sáng chế, phân bố xác suất khởi tạo được sử dụng để khởi tạo bộ mã hóa entropi có thể được tạo ra bằng cách chỉ dẫn cho chỉ các khung B được dự báo chuyển tiếp. Theo khía cạnh khác của sáng chế, việc khởi tạo ngữ cảnh có thể được làm thích ứng dựa vào tham số lượng tử hóa, nó được ký hiệu là QP , được sử dụng để mã hóa dữ liệu video hiện hành.

Theo một số phương án thực hiện sáng chế, bộ mã hóa theo cách khác có thể áp dụng phương pháp khởi tạo ngữ cảnh đối với các lát P đến các lát B được dự báo chuyển tiếp và có thể báo hiệu sự xuất hiện của lựa chọn này. Theo một số phương án thực hiện sáng chế, việc báo hiệu có thể là rõ ràng. Theo các phương án thực hiện khác của sáng chế, việc báo hiệu có thể là ẩn. Theo một số phương án thực hiện sáng chế bao gồm việc báo hiệu rõ ràng, còn có thể được gửi tới bộ giải mã mỗi khi lát P được thay thế bằng lát B được dự báo chuyển tiếp. Theo một phương án thực hiện sáng chế, còn có thể được báo hiệu là cú pháp thông thường. Theo các phương án thực hiện khác, còn có thể được báo hiệu trong thông báo không quy chuẩn, chẳng hạn thông báo SEI hoặc thông báo không quy chuẩn khác.

Như được thể hiện trên Fig.41, theo phương án thực hiện sáng chế, phương pháp khởi tạo ngữ cảnh đối với các lát P có thể được áp dụng cho các lát B được dự báo đơn, dẫn đến hiệu suất nén cao hơn do đó mức độ ngẫu nhiên lớn hơn được tạo cho các lát B. Các lát tham chiếu được sử dụng theo lát B được dự báo đơn là một phần của hoặc các khung/ảnh sớm hơn theo thời gian hoặc các khung/ảnh muộn hơn theo thời gian. Vì thế, lát B được dự báo đơn có thể bao gồm dữ liệu dư với đặc điểm thống kê khác với các đặc điểm đó của lát B hai chiều. Theo thời gian

có thể tham chiếu thứ tự hiển thị của các ảnh/các khung được giải mã. Như được thể hiện trên Fig.41, phương pháp này có thể thu các lát B, các lát P, và các lát I và dựa vào các kiểu của lát thu được, kỹ thuật khởi tạo ngữ cảnh được chọn. Ngoài ra, trong trường hợp của lát B được dự báo đơn kỹ thuật khởi tạo ngữ cảnh đối với lát P được sử dụng. Cụ thể hơn là, các bước dưới đây được thực hiện:

4101: Kiểu lát có thể được giải mã từ luồng bit;

4102: Xác định xem có hay không kiểu lát là lát B;

4103: Nếu kiểu lát là lát B (Đúng ở bước 4102), thì xác định xem lát có được dự báo chuyển tiếp hay lùi lại, hay là chuyển tiếp và lùi lại. Nếu lát được dự báo chuyển tiếp hoặc lùi lại, nhưng không là cả chuyển tiếp và lùi lại, thì ngữ cảnh được khởi tạo bằng phương pháp khởi tạo thứ hai, trái lại ngữ cảnh được khởi tạo bằng phương pháp khởi tạo thứ nhất;

4104: nếu kiểu lát không là lát B (Sai ở bước 4102), thì xác định xem có hay không kiểu lát là lát P. Nếu kiểu lát là lát P, thì ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ hai; và

4105: Nếu kiểu lát không là lát P (Sai ở bước 4104), thì xác định xem có hay không kiểu lát là lát I. Nếu kiểu lát là lát I, ngữ cảnh được khởi tạo bằng phương pháp khởi tạo thứ ba.

Như được thể hiện trên Fig.42, theo các phương án thực hiện sáng chế, lát B được dự báo chuyển tiếp có thể được khởi tạo theo cách khác với lát B được dự báo lùi lại và/hoặc lát B được dự báo hai hướng, nó là, dẫn đến hiệu suất nén cao hơn do mức độ ngẫu nhiên khác nhau được gán cho các lát B. Các lát tham chiếu được sử dụng theo lát B được dự báo chuyển tiếp là một phần của các khung/ảnh sớm hơn theo thời gian. Vì thế, lát B được dự báo chuyển tiếp có thể bao gồm dữ liệu dư với đặc điểm thống kê khác với các đặc điểm đó của lát B hai hướng và hoặc lùi lại. Theo thời gian có thể tham chiếu thứ tự hiển thị của các ảnh/khung được giải mã. Như được thể hiện trên Fig.42, phương pháp có thể thu các lát B, các lát P, và các lát I và dịch vị kiểu của lát thu được, phương pháp khởi tạo ngữ cảnh được chọn. Ngoài ra, trong trường hợp của lát B được dự báo chuyển tiếp phương pháp khởi tạo ngữ cảnh khác với các lát B, các lát P, và các lát I có thể được sử dụng. Cụ thể hơn là, các bước dưới đây được thực hiện:

4201: Kiểu lát có thể được giải mã từ luồng bit;

4202: Xác định xem có hay không kiểu lát là lát B;

4203: Nếu kiểu lát là lát B (Đúng ở bước 4202), thì xác định xem có hay không lát được dự báo chuyển tiếp. Nếu lát được dự báo chuyển tiếp, thì ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ tư, trái lại ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ nhất;

4204: Nếu kiểu lát không là lát B (Sai ở bước 4202), thì xác định xem có hay không kiểu lát là lát P. Nếu kiểu lát là lát P, thì ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ hai; và

4205: Nếu kiểu lát không là lát P (Sai ở bước 4204), xác định xem có hay không kiểu lát là lát I. Nếu kiểu lát là lát I, ngữ cảnh được khởi tạo với phương pháp khởi tạo thứ ba.

Phương pháp khởi tạo ngữ cảnh có thể bao gồm việc thiết lập các giá trị của bộ mã hóa entropi cần cho hoạt động của bộ mã hóa entropi. Chẳng hạn, đánh giá xác suất thứ nhất có thể được thiết lập cho bộ mã hóa entropi sử dụng các đánh giá xác suất để mã hóa dữ liệu, như là trong bộ mã hóa số học. Phương pháp khởi tạo ngữ cảnh có thể sử dụng phương pháp khởi tạo để xác định các giá trị. Theo một phương án, phương pháp khởi tạo thứ nhất tương ứng với việc sử dụng bảng thứ nhất của các giá trị được tính trước để thiết lập các giá trị. Tương tự, phương pháp khởi tạo thứ hai tương ứng với việc sử dụng bảng thứ hai của các giá trị được tính trước để thiết lập các giá trị. Theo phương án khác, phương pháp khởi tạo thứ nhất sử dụng bảng thứ nhất của các giá trị được tính trước mà được định rõ cho các khung mà sử dụng nhiều hơn một dự báo giữa các khung để thiết lập các giá trị, trong đó (các) khung được dùng để dự báo cả hai, theo thời gian, khung trước và khung sau khung hiện hành khi được tính thứ tự theo thứ tự hiển thị. Phương pháp khởi tạo thứ hai sử dụng bảng thứ hai của các giá trị được tính trước được định rõ từ các khung sử dụng nhiều hơn một dự báo giữa các khung để thiết lập các giá trị, trong đó các khung được dùng để dự báo trước hoặc sau theo thời gian, nhưng do hoặc là cả hai khung trước hoặc sau theo thời gian, khung hiện hành khi xét thứ tự theo thứ tự hiển thị. phương pháp khởi tạo thứ ba sử dụng bảng thứ ba của các giá trị được tính trước được định rõ đối với các khung sử dụng hầu hết một dự báo

giữa các khung để thiết lập các giá trị. phương pháp khởi tạo thứ tự sử dụng bảng thứ tự của các giá trị được tính trước được định rõ đối với các khung không sử dụng dự báo giữa các khung để thiết lập các giá trị.

Theo một phương án thực hiện sáng chế, RefPicList0 chứa danh mục các khung được giải mã trước đó được hiển thị trước khi hiển thị khung hiện hành tiếp theo là danh mục của các khung được giải mã trước đó được hiển thị sau đó để hiển thị khung hiện hành và RefPicList1 chứa danh mục các khung được giải mã trước đó được hiển thị tiếp theo việc hiển thị khung hiện hành tiếp theo là danh mục các khung được giải mã trước đó được hiển thị trước khi hiển thị khung hiện hành. Theo phương án này, điều kiện (2) nêu trên là đúng đối với lát B đơn hướng vì số lượng của các khung được giải mã trước đó được hiển thị tiếp theo việc hiển thị khung hiện hành là không, hoặc số lượng các khung được giải mã được hiển thị trước khi hiển thị khung hiện hành là không. Theo phương án thực hiện khác của sáng chế, PicOrderCnt biểu thị thứ tự hiển thị của các khung được giải mã. Theo phương án này, điều kiện (3) nêu trên là đúng đối với lát B đơn hướng vì các khung được giải mã đều được hiển thị trước khi hiển thị khung hiện hành hoặc sau việc hiển thị khung hiện hành.

Theo một số phương án thực hiện sáng chế bao gồm việc báo hiệu ẩn, theo phương pháp khởi tạo ngữ cảnh, sự xuất hiện của các lát P được áp dụng cho lát B được dự báo chuyển tiếp, có thể được suy ra bộ giải mã khi các lát tham chiếu (các khung/các ảnh) được sử dụng theo dự báo là tất cả các lát trước đó (các khung/các ảnh) dựa vào thứ tự này trong đó các lát cần được hiển thị. Theo một số phương án, sự xuất hiện của lát P được thay bởi lát B đơn hướng có thể được suy ra khi các điều kiện sau là đúng: (1) danh mục ảnh tham chiếu RefPicList1 có một hoặc nhiều mục nhập, (2) danh mục ảnh tham chiếu RefPicList1 và RefPicList0 chứa cùng các tập ảnh, và (3) tất cả các khung trong danh mục ảnh tham chiếu RefPicList1 và RefPicList0 có PicOrderCnt nhỏ hơn so với PicOrderCnt của khung hiện hành, trong đó PicOrderCnt chỉ báo thứ tự hiển thị của các khung theo thứ tự tăng. Theo một số phương án, sự xuất hiện của lát P được thay bởi lát B đơn hướng có thể được suy ra khi các điều kiện sau là đúng: (1) danh mục ảnh tham chiếu RefPicList1 có một hoặc nhiều mục nhập, (2) danh mục ảnh tham chiếu

RefPicList1 và RefPicList0 chứa cùng các tập ảnh, và (3) tất cả các khung trong danh mục ảnh tham chiếu RefPicList1 và RefPicList0 có PicOrderCnt lớn hơn PicOrderCnt của khung hiện hành, trong đó PicOrderCnt chỉ báo thứ tự hiển thị của các khung theo thứ tự tăng. Theo một số phương án, sự xuất hiện của lát P được thay bởi lát B đơn hướng có thể được suy ra khi các điều kiện sau là đúng: (1) danh mục ảnh tham chiếu RefPicList1 có một hoặc nhiều mục nhập, và (2) khung thứ nhất trong danh mục ảnh tham chiếu RefPicList1 và khung thứ nhất trong RefPicList0 có PicOrderCnt nhỏ hơn (hoặc lớn hơn) so với PicOrderCnt của khung hiện hành, trong đó PicOrderCnt chỉ báo thứ tự hiển thị của các khung thứ tự tăng. Theo một số phương án, sự xuất hiện của lát P được thay bởi lát B đơn hướng có thể được suy ra khi các điều kiện sau là đúng: (1) danh mục ảnh tham chiếu RefPicList1 có nhiều hơn một mục nhập, và (2) các khung [0, N] trong danh mục ảnh tham chiếu RefPicList1 và các khung [0, N] trong RefPicList0 có PicOrderCnt nhỏ hơn (hoặc lớn hơn) so với PicOrderCnt của khung hiện hành, trong đó PicOrderCnt chỉ báo thứ tự hiển thị của các khung theo thứ tự tăng. Theo một số phương án, sự xuất hiện của lát P được thay bởi lát B đơn hướng có thể được suy ra khi các điều kiện sau là đúng: (1) danh mục ảnh tham chiếu RefPicList1 có một hoặc nhiều mục nhập, (2) khung thứ nhất trong danh mục ảnh tham chiếu RefPicList1 là giống như khung thứ hai trong danh mục ảnh tham chiếu RefPicList0, (3) khung thứ hai trong danh mục ảnh tham chiếu RefPicList1 giống như khung thứ nhất trong danh mục ảnh tham chiếu RefPicList0, (4) các khung [0, N] trong danh mục ảnh tham chiếu RefPicList1 và các khung [0, N] trong RefPicList0 có PicOrderCnt nhỏ hơn (hoặc lớn hơn) so với PicOrderCnt của khung hiện hành, trong đó PicOrderCnt chỉ báo thứ tự hiển thị của các khung theo thứ tự tăng. Theo một số phương án thực hiện sáng chế, thứ tự trong RefPicList0 và RefPicList1 không cần đồng nhất chứa cùng các tập ảnh. Theo phương án thực hiện sáng chế, khi danh mục ảnh tham chiếu RefPicList1 có nhiều hơn một mục nhập và RefPicList1 giống với danh mục ảnh tham chiếu RefPicList0, thì hai mục nhập đầu RefPicList1[0] và RefPicList1[1] có thể được chuyển mạch. N có thể nhỏ hơn so với tổng số lượng các khung/ảnh trong danh mục ảnh tham chiếu. PicOrderCnt có thể còn nêu ra thứ tự hiển thị tương đối trong bộ nhớ đệm của bộ

giải mã ảnh.

Khi sự xuất hiện của lát P được thay bởi lát B được dự báo chuyển tiếp được chỉ báo, ngữ cảnh đối với lát entropi có thể được khởi tạo bằng cách sử dụng phương pháp lát P. Theo phương án thực hiện khác của sáng chế, ngữ cảnh của lát có thể được khởi tạo bằng cách sử dụng phương pháp lát P.

Bảng 13 thể hiện cú pháp làm ví dụ để báo hiệu một cách rõ ràng rằng ngữ cảnh ban đầu của lát B cần được khởi tạo bằng cách sử dụng phương pháp lát P. Theo phương án thực hiện sáng chế liên quan đến bảng 13, “cabac_init_P_flag” là cờ chỉ báo, để khởi tạo mã hóa entropi lát B, xem phương pháp lát B hay phương pháp lát P cần được chọn. Theo một số phương án, nếu giá trị của cờ “cabac_init_P_flag” là “0,” thì phương pháp lát B được chọn để khởi tạo, và nếu giá trị của cờ “cabac_init_P_flag” là “1,” thì phương pháp lát P được chọn để khởi tạo.

slice_header() {	C	Phần tử mô tả
entropy_slice_flag	2	u(1)
if (entropy_slice_flag) {		
first_lcu_in_slice	2	ue(v)
(entropy_coding_mode_flag) {		
lcu_row_cabac_init_flag	1	u(1)
if(lcu_row_cabac_init_flag){		
lcu_row_cabac_init_idc_flag	1	u(1)
}		
}		
if(entropy_coding_mode_flag && slice_type != I) {		
cabac_init_idc	2	ue(v)
(slice_type==B_SLICE && lát B được dự báo chuyển tiếp)		
cabac_init_P_flag	1	u(1)
}		
}		
else {		

if(entropy_coding_mode_flag) {		
lcu_row_cabac_init_flag	1	u(1)
if(lcu_row_cabac_init_flag){		
lcu_row_cabac_init_idc_flag	1	u(1)
}		
}		
first_lcu_in_slice	2	ue(v)
slice_type	2	ue(v)
Some elements mào đầu lát ổn định		
if(entropy_coding_mode_flag && slice_type != I) {		
cabac_init_idc	2	ue(v)
(slice_type==B_SLICE && lát B được dự báo chuyển tiếp)		
cabac_init_P_flag	1	u(1)
}		
Remainder mào đầu lát ổn định		
}		
}		

Bảng 13: Bảng cú pháp làm ví dụ thể hiện việc báo hiệu một cách rõ ràng sự khởi tạo lát B bằng cách sử dụng phương pháp lát P

Theo một số phương án thực hiện sáng chế, các trạng thái khởi tạo ngữ cảnh đối với lát entropi có thể được dựa vào số lượng các bin được xử lý bởi bộ mã hóa entropi. Bộ mã hóa entropi có thể đồng quy nhanh hơn so với thống kê nguồn khi được khởi tạo một cách chính xác. Sự đồng quy nhanh hơn có thể đạt được sau một số bit bị loại và vì thế hiệu suất nén cao hơn. Theo một số phương án thực hiện sáng chế, số lượng các bin có thể được truyền có thể được đánh giá, và khi số lượng bin được đánh giá đáp ứng tiêu chuẩn thứ nhất, thì phương pháp khởi tạo thứ nhất có thể được sử dụng. khi số lượng bin được đánh giá không đáp ứng tiêu chuẩn thứ nhất, phương pháp khởi tạo thứ hai có thể được sử dụng.

Phương án làm ví dụ thực hiện sáng chế có thể được hiểu rõ hơn thông qua phần mô tả có dựa vào Fig.38. Theo các phương án này, số lượng các bin được xử lý có thể được đánh giá ở bước 1320. Số lượng bin được đánh giá được xử lý, được

ký hiệu là Nbins, có thể được so sánh ở bước 1322 với giá trị ngưỡng, được ký hiệu là Tbins. Khi số lượng các bin được xử lý tăng độ chính xác dự báo của việc khởi tạo ngưỡng cảnh trên cơ sở QP có thể giảm. Độ chính xác dự báo cao hơn để khởi tạo ngưỡng cảnh có thể dẫn đến hiệu suất nén tốt hơn. Nếu số lượng bin được đánh giá được xử lý là lớn hơn 1324 giá trị ngưỡng, so với giá trị khởi tạo ngưỡng cảnh đơn có thể được chọn ở bước 1326. Nếu số lượng các bin được đánh giá được xử lý là không 1328 lớn hơn giá trị ngưỡng, thì ngưỡng cảnh có thể được khởi tạo một cách thích ứng ở bước 1330 dựa vào QP. Giá trị khởi tạo ngưỡng cảnh đơn có thể được chọn dựa vào chỉ dẫn và sự tối ưu hóa theo giá trị đo được chọn, chẳng hạn lỗi bình phương, entropi tương đối và khoảng cách khác theo hệ mét. Sự khởi tạo trên cơ sở QP thích ứng có thể là sự thích ứng afin có dạng $C_A * QP + C_B$, trong đó C_A và C_B là các hằng số. Theo một số phương án, số lượng các bin có thể được đánh giá dựa vào số lượng các bin được xử lý trong lát trước. Theo các phương án thực hiện khác, số lượng các bin có thể được đánh giá dựa vào quá trình xử lý số lượng các bin trong khung trước.

Theo một số phương án thực hiện sáng chế, như được thể hiện trên Fig.39 dưới dạng ảnh biểu thị 1340 phạm vi của số lượng bin được xử lý, nhân, các phạm vi rời rạc (ba bộ được thể hiện 1342, 1344, 1346) của số lượng bin được xử lý có thể được xác định và được mô tả liên quan đến số lượng các ngưỡng (hai được thể hiện 1348, 1350), và giá trị khởi tạo ngưỡng cảnh có thể được chọn dựa vào trong số các phạm vi 1342, 1344, 1346 số lượng bin được đánh giá được xử lý giảm, chẳng hạn đối với ba phạm vi 1342, 1344, 1346, khi $Nbins \leq T_{min}$ 1342, ngưỡng cảnh có thể được khởi tạo một cách thích ứng dựa vào QP, khi $T_{min} < Nbins \leq T_1$ 1344, ngưỡng cảnh có thể được khởi tạo cho giá trị ngưỡng cảnh cố định thứ nhất và khi $T_1 < Nbins$ 1346, ngưỡng cảnh có thể được khởi tạo cho giá trị ngưỡng cảnh cố định khác nhau thứ hai.

Phương án thực hiện khác của sáng chế có thể được rõ ràng hơn khi mô tả có dựa vào Fig.40. Theo phương án này, giá trị của QP có thể được xác định ở bước 1400 và được kiểm tra ở bước 1402 liên quan đến giá trị ngưỡng, được ký hiệu là T_{QP} . Nói chung, khi QP giảm số lượng các bin được xử lý có thể tăng. Nếu QP không nhỏ hơn 1404 so với giá trị ngưỡng, thì ngưỡng cảnh có thể được khởi tạo một cách thích ứng ở bước 1406 dựa vào QP. Nếu giá trị của QP nhỏ hơn 1408 so với

giá trị ngưỡng, thì giá trị khởi tạo ngữ cảnh đơn có thể chọn ở bước 1410. Giá trị khởi tạo ngữ cảnh đơn có thể được chọn dựa vào chỉ dẫn và sự tối ưu hóa của các metric được chọn, chẳng hạn lỗi bình phương, liên quan đến entropi và các metric khoảng cách khác.

Theo một số phương án thực hiện sáng chế, các phạm vi bộ, rời rạc của QP có thể được xác định, và giá trị khởi tạo ngữ cảnh có thể được chọn dựa vào trong số các phạm vi giá trị QP giảm.

Bảng 14 thể hiện sự so sánh về khả năng suy giảm tốc độ đối với mã hóa nội toàn phần. So sánh thứ nhất là, được thể hiện trong hai cột con của cột ba, là so sánh, bằng cách sử dụng phần mềm H.264/AVC Joint Model (JM), phiên bản 13.0, giữa mã hóa bằng cách sử dụng các lát bội, trong đó giải mã entropi và phục hồi khối macrô đối với lát không phụ thuộc vào các lát khác, và mã hóa bằng cách không sử dụng các lát. Thông thường là, đối với cùng một tốc độ bit, chất lượng được giảm -0,3380 dB nhờ mã hóa bằng cách sử dụng các lát bội trên toàn bộ các lát không sử dụng. Thông thường là, đối với cùng mức độ chất lượng, tốc độ bit tăng 7% nhờ mã hóa bằng cách sử dụng các lát bội trên toàn bộ các lát không sử dụng.

So sánh thứ hai là, như được thể hiện trong hai cột con của cột bốn, là so sánh, giữa mã hóa bằng cách sử dụng một lát phục hồi được phân chia, theo các phương án thực hiện sáng chế, thành các lát entropi bội (hai hàng của các khối macrô trên một lát entropi) và mã hóa bằng cách sử dụng JM 13.0 không sử dụng lát. Thông thường là, đối với cùng tốc độ bit, chất lượng bị giảm -0,0860 dB bằng cách sử dụng một lát phục hồi với các lát entropi bội qua việc mã hóa không sử dụng lát. Thông thường là, đối với cùng mức độ chất lượng, tốc độ bit tăng 1,83% bằng cách mã hóa sử dụng một lát phục hồi với các lát entropi bội qua việc mã hóa bằng cách không sử dụng lát.

Mã hóa nội toàn phần

Chuỗi	Độ phân giải	JM 13.0 các lát so sánh với JM 13.0 không sử dụng lát		Một lát phục hồi với các lát entropi so với JM 13.0 không sử dụng lát	
		BD SNR [dB]	BD tốc độ bit [%]	BD SNR [dB]	BD tốc độ bit [%]
BigShip	720p	-0,22	4,54	-0,08	1,61
City	720p	-0,28	4,03	-0,06	0,84
Crew	720p	-0,42	11,67	-0,11	2,98
Night	720p	-0,38	5,64	-0,06	0,91
ShuttleStart	720p	-0,39	9,12	-0,12	2,81
AVERAGE		-0,3380	7,00	-0,0860	1,83

Bảng 14: So sánh khả năng suy giảm tốc độ đối với mã hóa nội toàn phần

Bảng 15 thể hiện sự so sánh khả năng suy giảm tốc độ đối với mã hóa IBBP.

So sánh thứ nhất là, như được thể hiện trong hai cột con của cột ba, là so sánh, bằng cách sử dụng phần mềm H.264/AVC Joint Model (JM), phiên bản 13.0, giữa mã hóa bằng cách sử dụng các lát bội, trong đó giải mã entropi và phục hồi khối macro đối với lát không tùy thuộc vào các lát khác, và mã hóa bằng cách không sử dụng lát. Thông thường là, đối với cùng một tốc độ bit, chất lượng giảm -0,5460 dB do mã hóa bằng cách sử dụng các lát bội. Thông thường là, đối với cùng mức độ chất lượng, tốc độ bit tăng 21,41% do mã hóa bằng cách sử dụng các lát bội bỏ qua lát không sử dụng.

So sánh thứ hai, như được thể hiện trong hai cột con của cột bốn, là so sánh giữa mã hóa bằng cách sử dụng một lát phục hồi được phân chia, theo các phương án thực hiện sáng chế, thành các lát entropi bội (hai hàng của các khối macro trên lát entropi) và mã hóa bằng cách sử dụng JM 13.0 không sử dụng lát. Thông thường là, đối với cùng tốc độ bit, chất lượng giảm -0,31 dB bằng cách sử dụng một lát phục hồi với các lát entropi bội bỏ qua mã hóa không sử dụng lát. Thông thường là, đối với cùng mức độ chất lượng, tốc độ bit tăng 11,45% do mã hóa bằng cách sử dụng một lát phục hồi với các lát entropi bội bỏ qua mã hóa không sử dụng lát.

Mã hóa IBBP					
Chuỗi	Độ phân giải	JM 13.0 các lát được so sánh với JM 13.0 không sử dụng lát		Một lát phục hồi với các lát entropi bội được so sánh với JM 13.0 không sử dụng lát	
		BD SNR	BD tốc độ	BD SNR	BD tốc độ bit
		[dB]	bit [%]	[dB]	[%]
BigShip	720p	-0,45	19,34	-0,26	10,68
City	720p	-0,48	17,83	-0,22	7,24
Crew	720p	-0,62	30,10	-0,33	14,93
Night	720p	-0,36	11,11	-0,19	5,5
ShuttleStart	720p	-0,82	28,69	-0,55	18,89
AVERAGE		-0,5460	21,41	-0,31	11,45

Bảng 15: So sánh khả năng suy giảm tốc độ đối với mã hóa IBBP

Kết quả so sánh là, mã hóa bằng cách sử dụng các lát entropi bội trong một lát phục hồi thu được sự tăng tốc độ bit 5,17% và 9,96% đối với mã hóa nội toàn phần và mã hóa IBBP, một cách tương ứng, vượt mã hóa bằng cách sử dụng các lát, trong đó giải mã entropi và phục hồi khối macrô cho lát không phụ thuộc vào các lát khác, qua đó cả hai đề cho phép mã hóa song song.

Bảng 16 thể hiện so sánh khả năng suy giảm tốc độ đối với mã hóa nội toàn phần và mã hóa IBBP. Trong bảng này, sự so sánh là so sánh giữa mã hóa bằng cách không sử dụng các lát và mã hóa bằng cách sử dụng một lát phục hồi được phân chia thành các lát entropi, theo các phương án thực hiện sáng chế, các bin kích thước tối đa 26k trên một lát entropi. So sánh thứ nhất là, như được thể hiện trong hai cột con của cột hai, là so sánh bằng cách sử dụng mã hóa nội toàn phần. Thông thường là, đối với cùng tốc độ bit, chất lượng diamr -0,062 dB do mã hóa sử dụng lát phục hồi với các lát entropi bội. Thông thường là, đối với cùng mức độ chất lượng, tốc độ bit tăng 1,86% do mã hóa sử dụng lát phục hồi với các lát entropi bội. Vì thế, đối với mã hóa nội toàn phần bằng cách sử dụng các lát entropi của các bin kích thước tối đa 26k trên lát entropi, có tốc độ bit trung bình tăng

khoảng 0,64 % so với khi các kích thước lát entropi không đổi của hai hàng của các khối macro.

So sánh thứ hai, như được thể hiện trong hai cột con của cột ba, là so sánh bằng cách sử dụng mã hóa IBBP. Thông thường là, đối với dùng tốc độ bit, chất lượng giảm -0,022 dB bằng cách sử dụng một lát phục hồi với các lát entropi bội so với mã hóa không sử dụng lát. Thông thường là, đối với cùng mức độ chất lượng, tốc độ bit tăng 0,787% do mã hóa bằng cách sử dụng một lát phục hồi với các lát entropi bội trên mã hóa không sử dụng lát. Vì thế, đối với mã hóa IBBP bằng cách sử dụng các lát entropi của các bin kích thước tối đa 26k trên lát entropi, tốc độ bit trung bình tăng khoảng 10,66 % so với khi các kích thước lát entropi không đổi của hai hàng của các khối macro.

Lát entropi được so sánh với JM 15.1 không sử dụng lát.				
Thử nghiệm (1): các bin tối đa 26k trên lát entropi				
Chuỗi (720p)	Mã hóa nội toàn phần		Mã hóa IBBP	
	BD SNR [dB]	BD tốc độ bit [%]	BD SNR [dB]	BD tốc độ bit [%]
BigShip	-0,07	1,40	-0,02	0,70
City	-0,07	1,02	-0,02	0,51
Crew	-0,05	1,31	-0,03	1,25
Night	-0,07	1,00	-0,02	0,66
ShuttleStart	-0,05	1,20	-0,03	-0,82
AVERAGE	-0,062	1,187	-0,022	0,787

Bảng 16: So sánh khả năng suy giảm tốc độ đối với mã hóa nội toàn phần và mã hóa IBBP bằng cách sử dụng các lát entropi nhỏ hơn các bin 26k trên lát entropi

Sử dụng các lát entropi cho phép giải mã song song, và bộ mã hóa phân chia lát phục hồi thành các lát entropi, trong đó mỗi lát entropi nhỏ hơn số lượng bin tối đa có thể tăng đáng kể tốc độ bit trên các lát entropi có số lượng các khối macro không đổi.

Mặc dù các sơ đồ và các lưu đồ trên các hình vẽ có thể thể hiện thứ tự thực hiện cụ thể, nhưng cần hiểu rằng thứ tự thực hiện có thể khác với thứ tự được minh họa. Chẳng hạn, thứ tự thực hiện các khối có thể được thay đổi liên quan đến thứ tự được thể hiện. Ngoài ra, ví dụ, hai hoặc nhiều hơn hai khối được thể hiện liên tiếp trên hình vẽ có thể được thực hiện đồng thời toàn bộ, hoặc đồng thời riêng phần. Rõ ràng với người có hiểu biết trung bình trong lĩnh vực phương pháp có liên quan rằng, sáng chế có thể thực hiện bằng phần mềm, phần cứng và/hoặc phần mềm cơ sở bằng các tạo ra các chức năng với mặt logic như đã mô tả trên.

Một số phương án thực hiện, sáng chế có thể bao gồm sản phẩm chương trình máy tính bao gồm vật ghi đọc được bằng máy tính có các chỉ lệnh được lưu trữ trên đó có thể được sử dụng để lập trình hệ thống tính thực hiện các chức năng và phương pháp bất kỳ được mô tả trên đây. Vật ghi đọc được bằng máy tính làm ví dụ có thể bao gồm, nhưng không giới hạn, các thiết bị nhớ truy cập nhanh, vật ghi dạng đĩa, chẳng hạn các đĩa mềm, các đĩa quang, các đĩa quang từ, các đĩa phương pháp số đa năng (các DVD), các đĩa compact (các CD), các vật ghi dạng thẻ nhớ cỡ nhỏ và vật ghi dạng đĩa khác, bộ nhớ chỉ đọc (ROM), bộ nhớ chỉ đọc khả trình (PROM), bộ nhớ chỉ đọc khả trình xóa được (EPROMS), bộ nhớ chỉ đọc khả trình xóa được bằng điện (EEPROM), bộ nhớ truy cập ngẫu nhiên (RAMS), bộ nhớ truy cập ngẫu nhiên video (VRAM), bộ nhớ truy cập ngẫu nhiên động (DRAM) và vật ghi hoặc thiết bị lưu trữ kiểu bất kỳ thích hợp để lưu trữ các chỉ dẫn và/hoặc dữ liệu.

Theo một số phương án thực hiện sáng chế, mẫu quét có thể được khởi tạo ở phần bắt đầu của lát entropi.

Theo một số phương án thực hiện sáng chế, mẫu quét có thể được khởi tạo với đơn vị cơ bản bắt đầu trong hàng trong lát entropi.

Theo một số phương án thực hiện sáng chế, trạng thái được kết hợp với việc xác định quét thích ứng có thể được khởi tạo ở phần bắt đầu của lát entropi.

Theo một số phương án thực hiện sáng chế, trạng thái được kết hợp với việc xác định quét thích ứng có thể được khởi tạo với đơn vị cơ bản bắt đầu trong hàng trong lát entropi.

Theo một số phương án thực hiện sáng chế, thứ tự quét hệ số có thể được

tách với thứ tự tìm nạp ngữ cảnh.

Theo một số phương án thực hiện sáng chế, lát B được dự báo chuyển tiếp có thể được phát hiện và ngữ cảnh được kết hợp với mã hoá entropi lát B được dự báo chuyển tiếp có thể được khởi tạo theo phương pháp lát P.

Theo một số phương án thực hiện sáng chế, ngữ cảnh có thể được khởi tạo dựa vào đếm bin.

Theo một số phương án thực hiện sáng chế, ngữ cảnh có thể được khởi tạo dựa vào giá trị tham số lượng tử hóa.

Một phương án thực hiện sáng chế bộc lộ phương pháp giải mã khung video của chuỗi video bao gồm:

- (a) thu lát trong bộ giải mã video;
- (b) nhận dạng nếu lát là hoặc lát B được dự báo chuyển tiếp hoặc lát B được dự báo lùi lại, và hoặc là cả hai lát B được dự báo chuyển tiếp và lùi lại;
- (c) khởi tạo, bằng cách sử dụng phương pháp lát P, ngữ cảnh được kết hợp với lát.

Theo một số phương án thực hiện sáng chế, việc nhận dạng có thể bao gồm chỉ lát B được dự báo lùi lại.

Theo một số phương án thực hiện sáng chế, việc nhận dạng có thể dựa vào: (1) các ảnh tham chiếu nhận dạng danh mục thứ nhất có ít nhất một mục nhập, (2) các ảnh tham chiếu nhận dạng danh mục thứ nhất và danh mục thứ hai chứa cùng một tập các ảnh, (3) tất cả các ảnh trong danh mục thứ nhất và danh mục thứ hai được hiển thị trước lát hiện hành.

Theo một số phương án thực hiện sáng chế, việc nhận dạng có thể dựa vào: (1) các ảnh tham chiếu nhận dạng danh mục thứ nhất có ít nhất một mục nhập, (2) các ảnh tham chiếu nhận dạng danh mục thứ nhất và danh mục thứ hai chứa cùng tập ảnh, (3) tất cả các ảnh trong danh mục thứ nhất và danh mục thứ hai cần được hiển thị sau lát hiện hành.

Theo một số phương án thực hiện sáng chế, việc nhận dạng có thể dựa vào: (1) các ảnh tham chiếu nhận dạng danh mục thứ nhất có ít nhất một mục nhập, (2) các ảnh tham chiếu nhận dạng danh mục thứ nhất và danh mục thứ hai chứa tập không đồng mở rộng của các ảnh, và tập không đồng mở rộng của các ảnh cần

được hiển thị trước lát hiện hành.

Theo một số phương án thực hiện sáng chế, việc nhận dạng có thể dựa vào: (1) các ảnh tham chiếu nhận dạng danh mục thứ nhất có ít nhất một mục nhập, (2) các ảnh tham chiếu nhận dạng danh mục thứ nhất và danh mục thứ hai chứa tập không đồng mở rộng của các ảnh, và tập không đồng mở rộng của các ảnh cần được hiển thị sau lát hiện hành.

Theo một số phương án thực hiện sáng chế, việc nhận dạng dựa vào: (1) các ảnh tham chiếu nhận dạng danh mục thứ nhất có ít nhất một mục nhập, (2) ảnh tham chiếu thứ nhất trong danh mục thứ nhất là giống như ảnh tham chiếu thứ hai trong danh mục thứ hai, (3) ảnh tham chiếu thứ nhất trong danh mục thứ hai là giống như ảnh tham chiếu thứ hai trong danh mục thứ nhất, (4) các ảnh tham chiếu nhận dạng danh mục thứ nhất và danh mục thứ hai chứa tập không đồng mở rộng của các ảnh, và tập không đồng mở rộng của các ảnh cần được hiển thị trước lát hiện hành.

Một phương án thực hiện sáng chế bộc lộ phương pháp giải mã khung video của chuỗi video bao gồm các bước:

- (a) thu lát trong bộ giải mã video;
- (b) nhận dạng nếu lát là lát B được dự báo chuyển tiếp;
- (c) khởi tạo, bằng cách sử dụng phương pháp khác với phương pháp được dùng đối với lát B bất kỳ, lát P bất kỳ, và lát I bất kỳ, ngữ cảnh được kết hợp với lát.

Các thuật ngữ và các cách diễn tả được dùng trong phần mô tả trên đây không nhằm giới hạn sáng chế mà phạm vi của sáng chế chỉ giới hạn bởi yêu cầu bảo hộ kèm theo.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã khung video, phương pháp này bao gồm các bước:

thu luồng bit của dữ liệu video được mã hóa;

giải mã, từ luồng bit, chỉ thị phân tử cú pháp của kiểu lát của lát;

khi kiểu lát chỉ báo kiểu lát B hoặc kiểu lát P, giải mã, từ luồng bit, cờ mà được sử dụng để xác định phương pháp khởi tạo ngữ cảnh mã hóa số học nhị phân thích ứng ngữ cảnh - Context-Adaptive Binary Arithmetic Coding (CABAC) đối với kiểu lát B hoặc kiểu lát P; và

khởi tạo ngữ cảnh CABAC được kết hợp với kiểu lát, trong đó

trong trường hợp mà kiểu lát chỉ báo kiểu lát B và trị số của cờ bằng với trị số thứ nhất, ngữ cảnh CABAC được kết hợp với kiểu lát B được khởi tạo bằng phương pháp khởi tạo thứ nhất,

trong trường hợp mà kiểu lát chỉ báo kiểu lát B và trị số của cờ bằng với trị số thứ hai, ngữ cảnh CABAC được kết hợp với kiểu lát B được khởi tạo bằng phương pháp khởi tạo thứ hai,

trong trường hợp mà kiểu lát chỉ báo kiểu lát P và trị số của cờ bằng với trị số thứ nhất, ngữ cảnh CABAC được kết hợp với kiểu lát P được khởi tạo bằng phương pháp khởi tạo thứ hai, và

trong trường hợp mà kiểu lát chỉ báo kiểu lát I, ngữ cảnh CABAC được kết hợp với kiểu lát I được khởi tạo bằng phương pháp khởi tạo thứ ba.

2. Phương pháp theo điểm 1, còn bao gồm các bước:

giải mã khung video; và

hiển thị khung video được giải mã.

3. Phương pháp theo điểm 1, còn bao gồm các bước:

giải mã luồng bit của dữ liệu video được mã hóa; và

hiển thị dữ liệu video được giải mã.

4. Thiết bị giải mã khung video, thiết bị này bao gồm bộ xử lý và bộ nhớ, bộ xử lý, làm việc cùng với bộ nhớ, được tạo cấu hình để:

thu luồng bit của dữ liệu video được mã hóa;

giải mã, từ luồng bit, chỉ thị phần tử cú pháp của kiểu lát của lát;

khi kiểu lát chỉ báo kiểu lát B hoặc kiểu lát P, giải mã, từ luồng bit, cờ mà được sử dụng để xác định phương pháp khởi tạo ngữ cảnh mã hóa số học nhị phân thích ứng ngữ cảnh - Context-Adaptive Binary Arithmetic Coding (CABAC) đối với kiểu lát B hoặc kiểu lát P; và

khởi tạo ngữ cảnh CABAC được kết hợp với kiểu lát, trong đó

trong trường hợp mà kiểu lát chỉ báo kiểu lát B và trị số của cờ bằng với trị số thứ nhất, ngữ cảnh CABAC được kết hợp với kiểu lát B được khởi tạo bằng phương pháp khởi tạo thứ nhất,

trong trường hợp mà kiểu lát chỉ báo kiểu lát B và trị số của cờ bằng với trị số thứ hai, ngữ cảnh CABAC được kết hợp với kiểu lát B được khởi tạo bằng phương pháp khởi tạo thứ hai,

trong trường hợp mà kiểu lát chỉ báo kiểu lát P và trị số của cờ bằng với trị số thứ nhất, ngữ cảnh CABAC được kết hợp với kiểu lát P được khởi tạo bằng phương pháp khởi tạo thứ hai, và

trong trường hợp mà kiểu lát chỉ báo kiểu lát I, ngữ cảnh CABAC được kết hợp với kiểu lát I được khởi tạo bằng phương pháp khởi tạo thứ ba.

5. Thiết bị theo điểm 4, trong đó bộ xử lý, làm việc cùng với bộ nhớ, còn được tạo cấu hình để:

giải mã khung video; và

hiển thị khung video được giải mã.

6. Thiết bị theo điểm 4, trong đó bộ xử lý, làm việc cùng với bộ nhớ, còn được tạo cấu hình để:

giải mã luồng bit của dữ liệu video được mã hóa; và

hiển thị dữ liệu video được giải mã.

FIG. 1

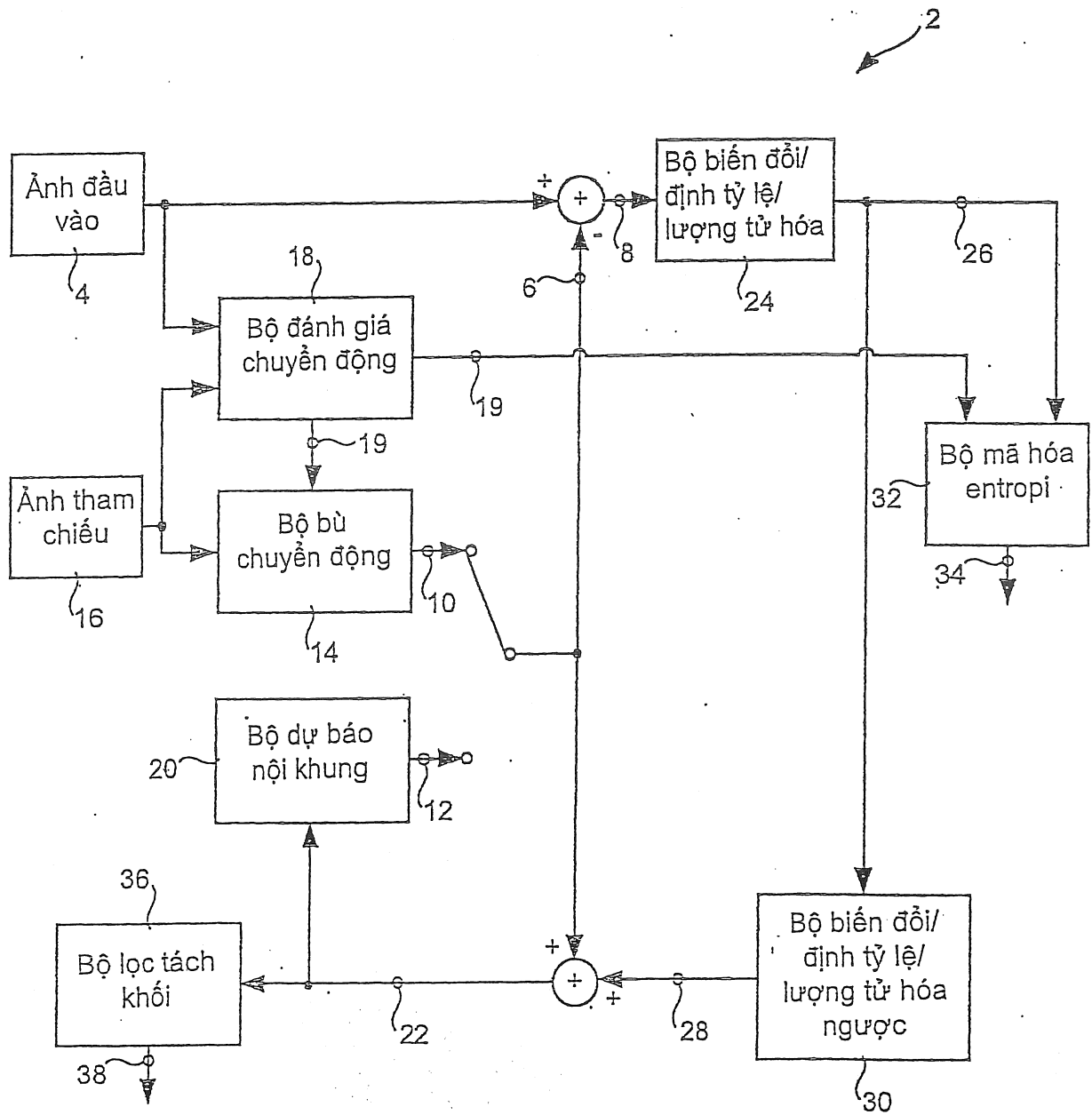
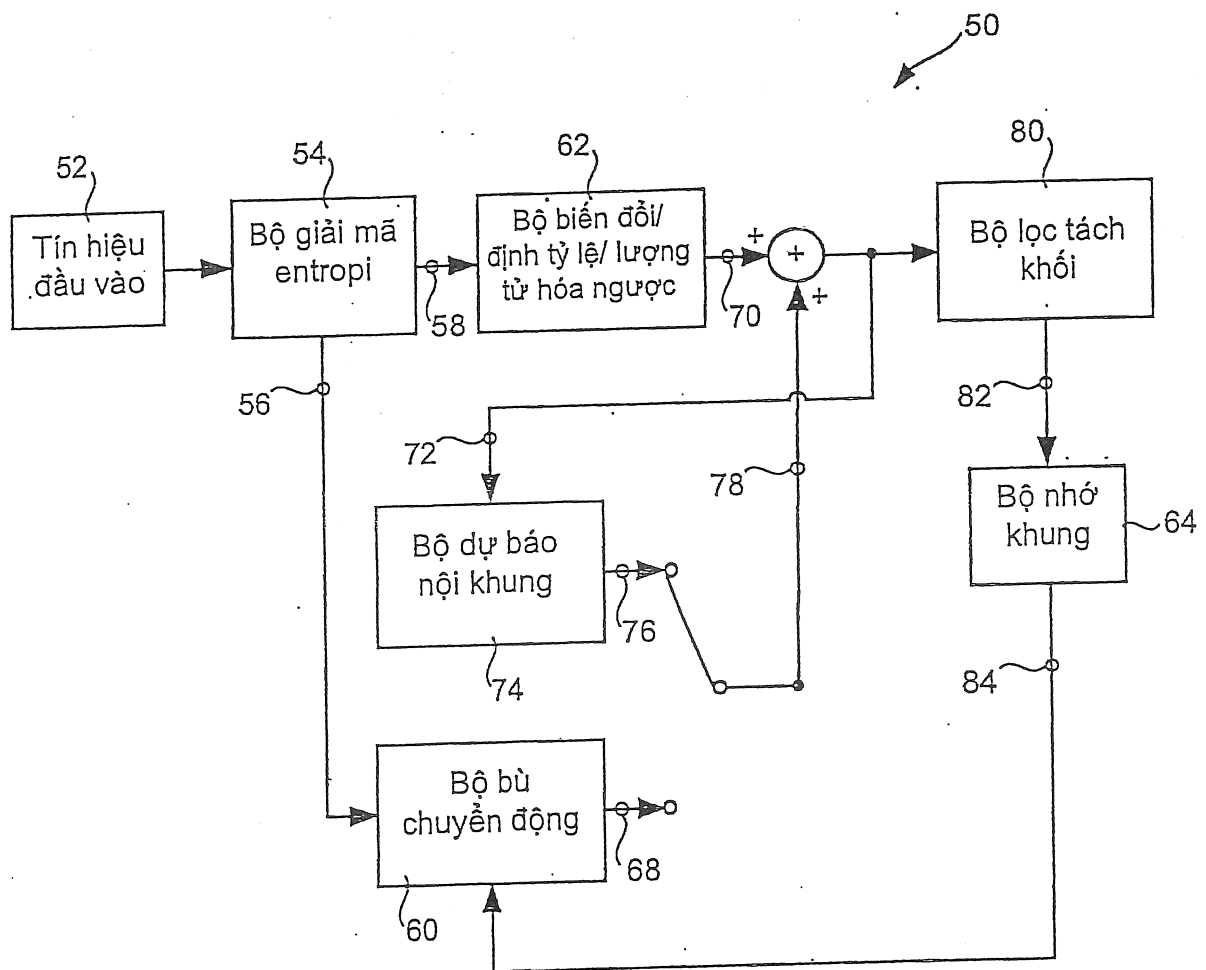


FIG. 2



3/41

FIG. 3

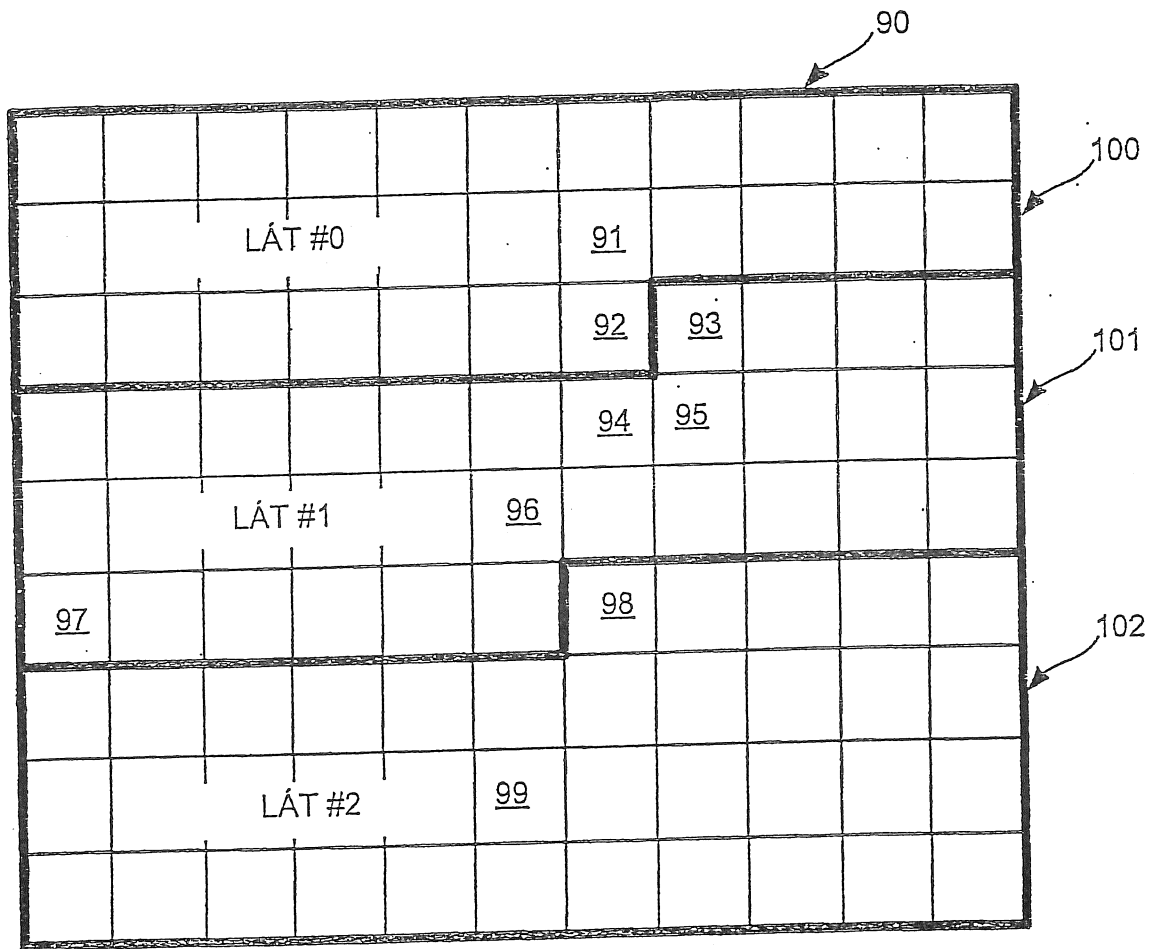


FIG. 5

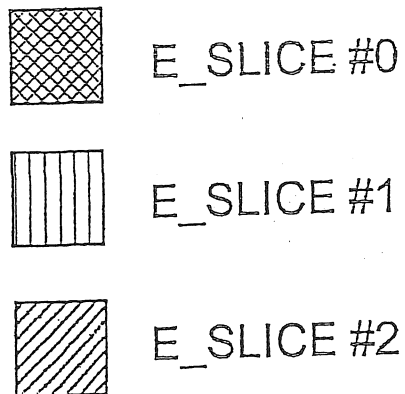
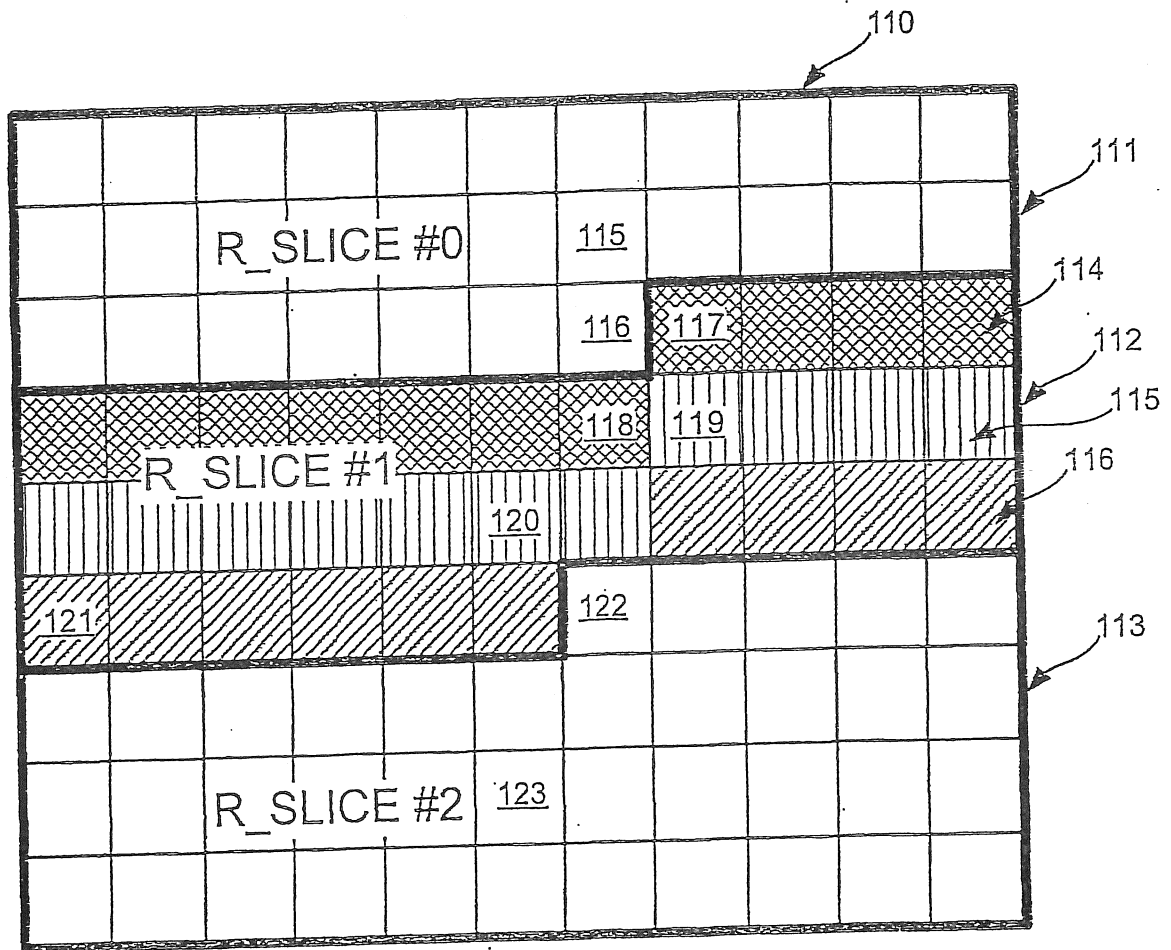
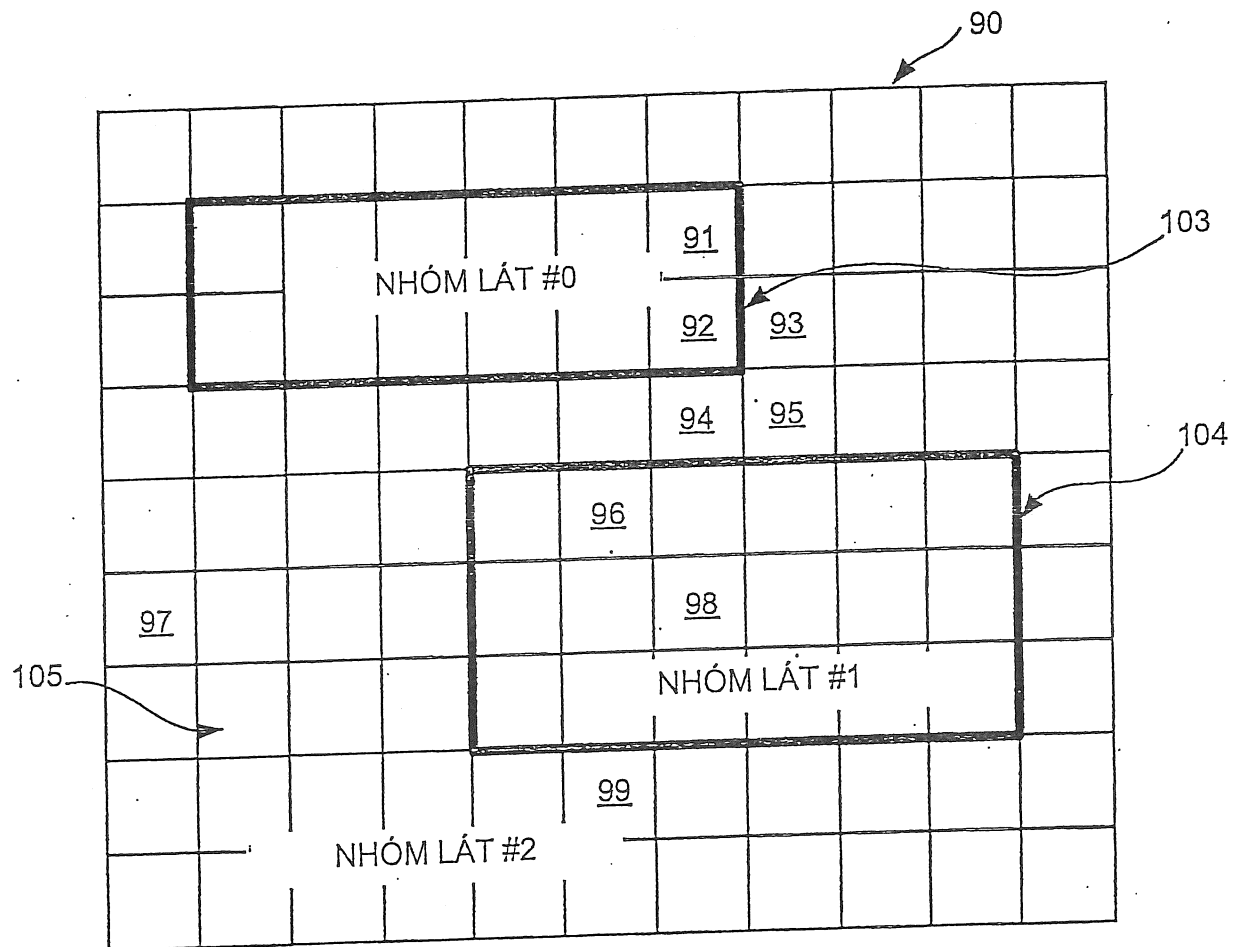


FIG. 4



6/41

FIG. 6

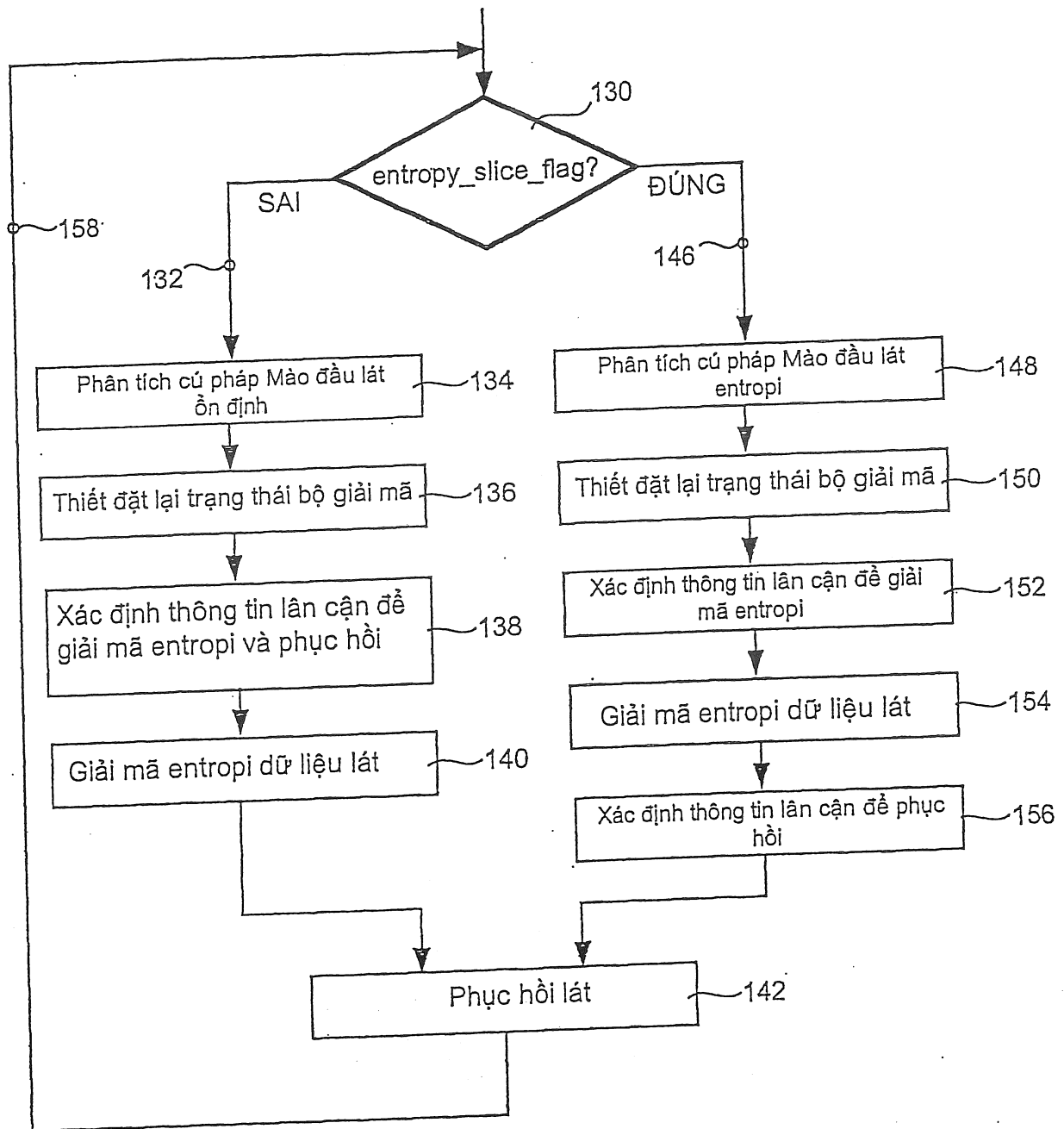


FIG. 7

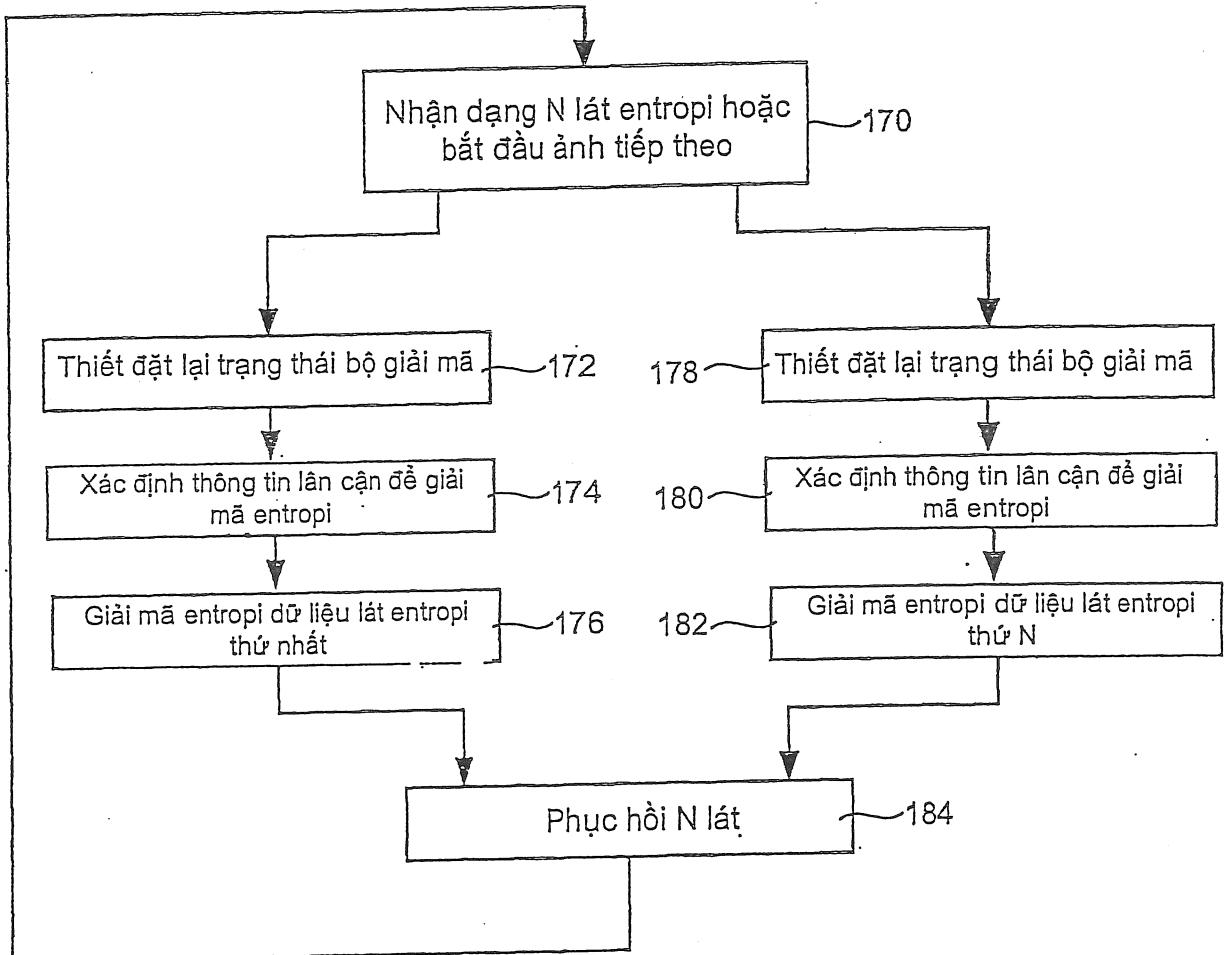


FIG. 8

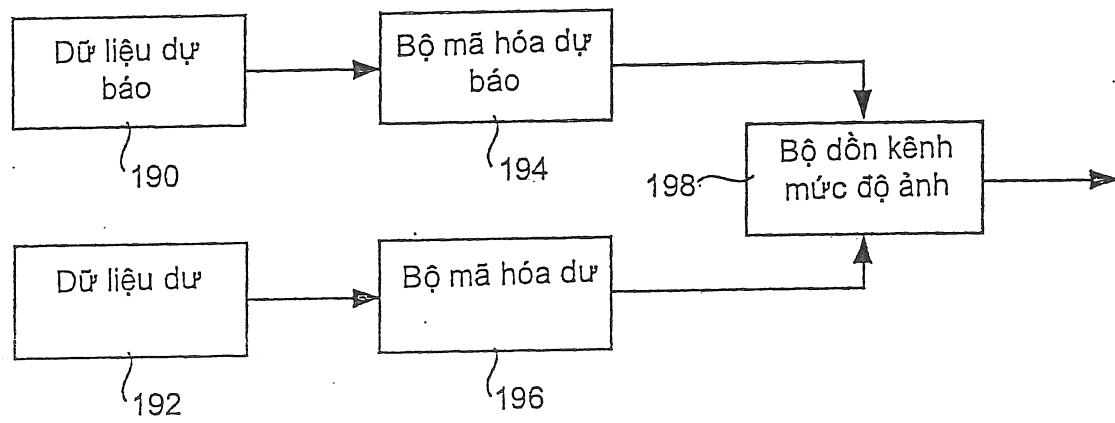
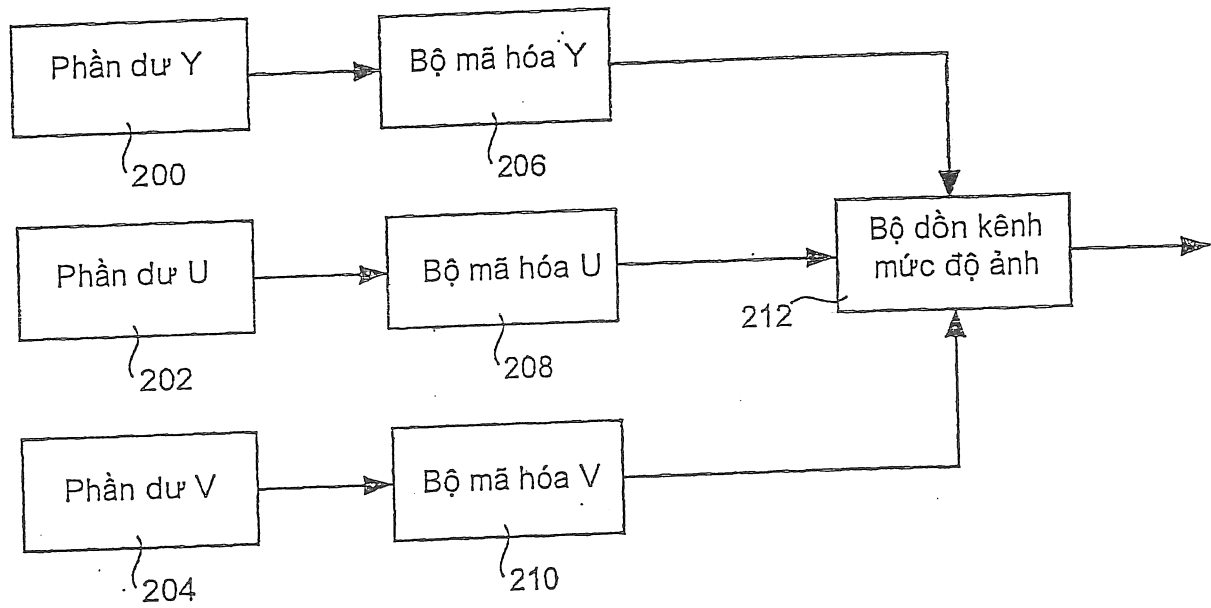


FIG. 9



10/41

FIG. 10

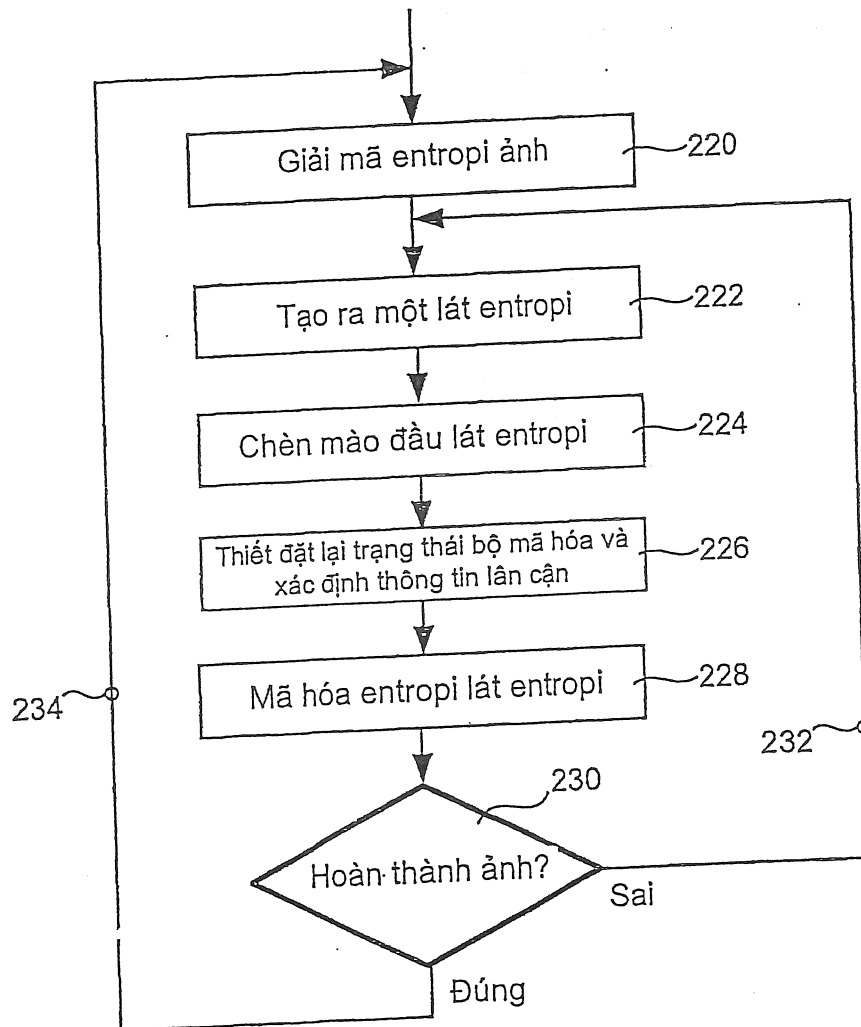
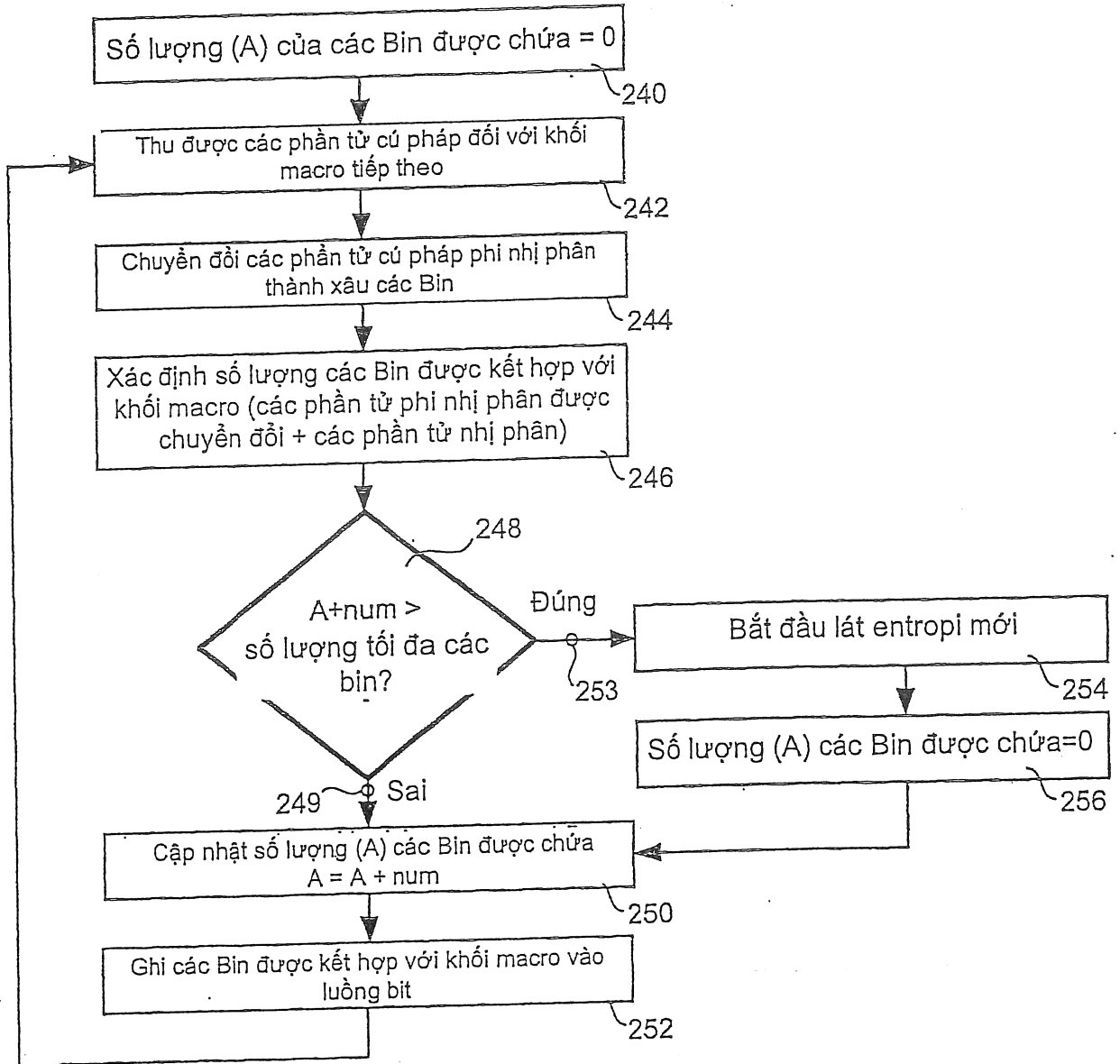


FIG. 11



12/41

FIG. 12

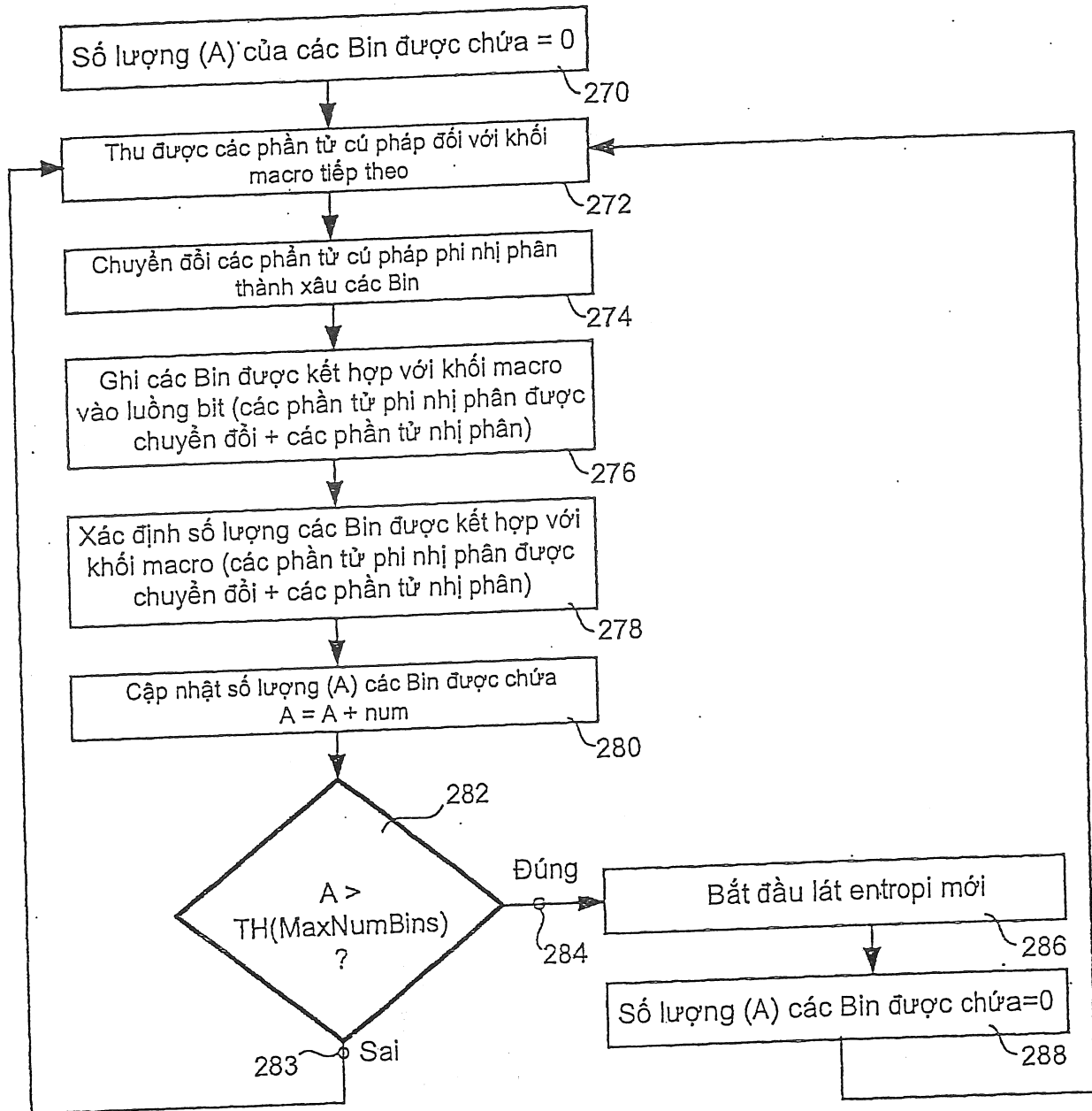


FIG. 13

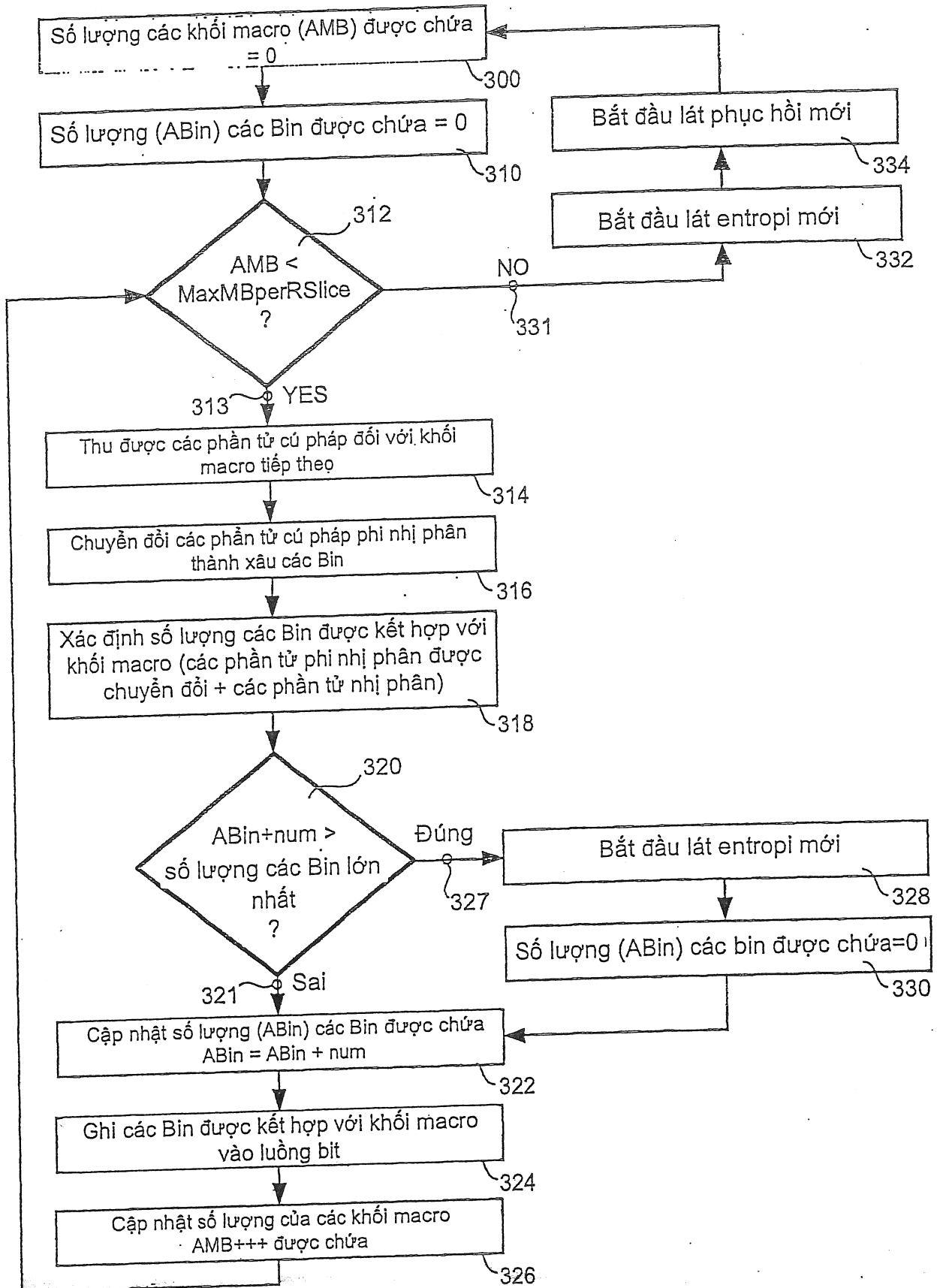


FIG. 14

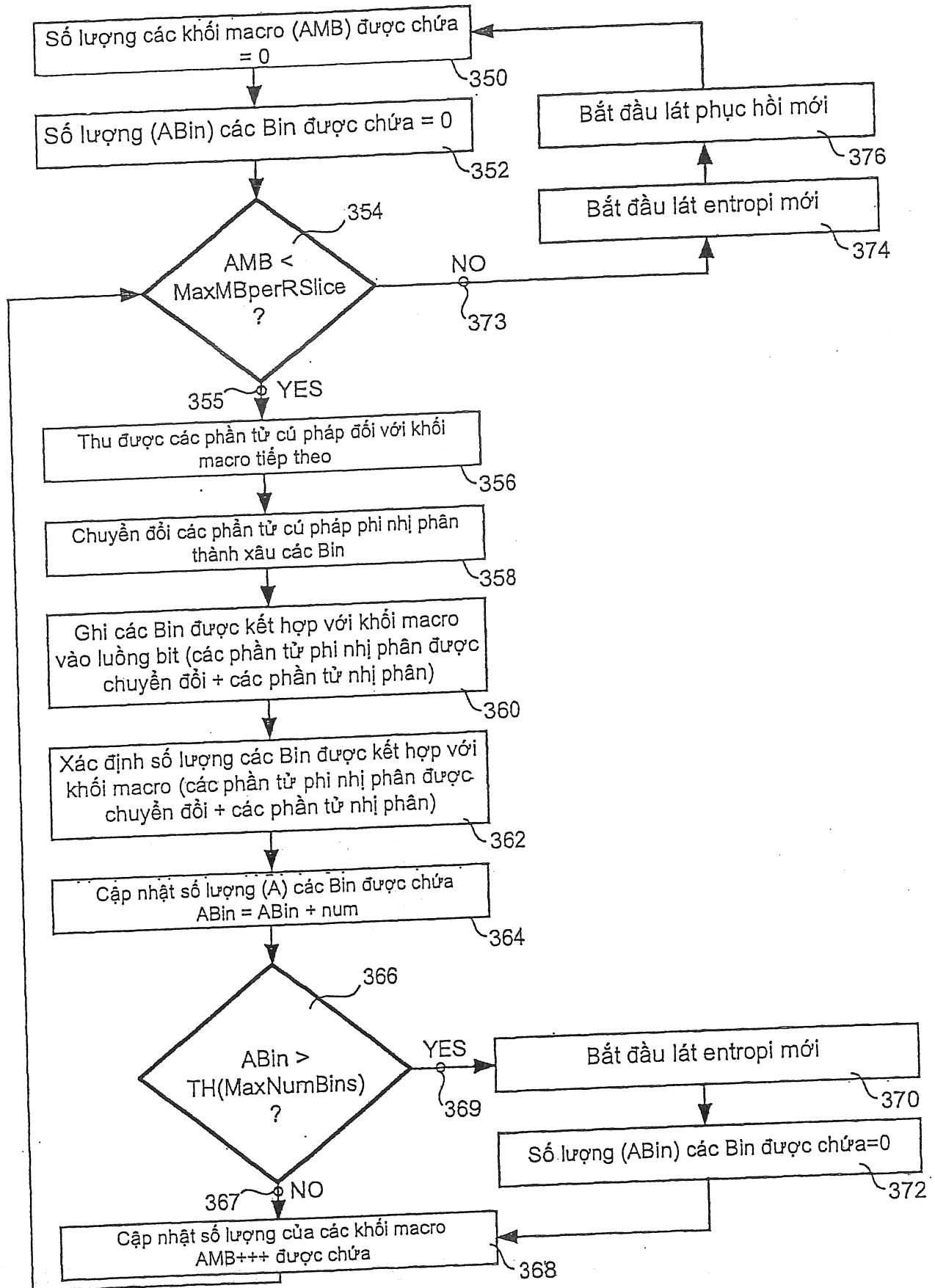


FIG. 15

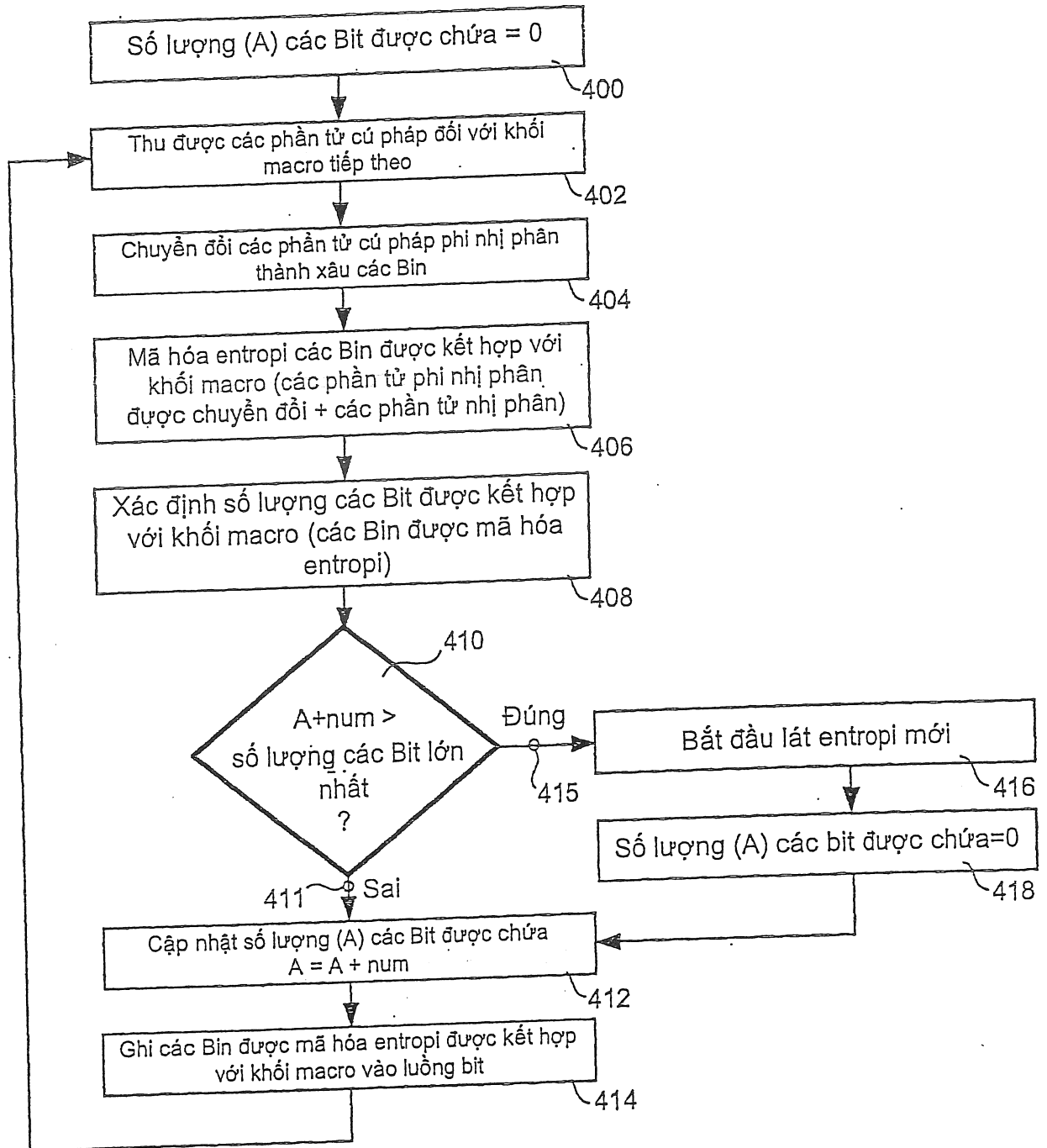


FIG. 16

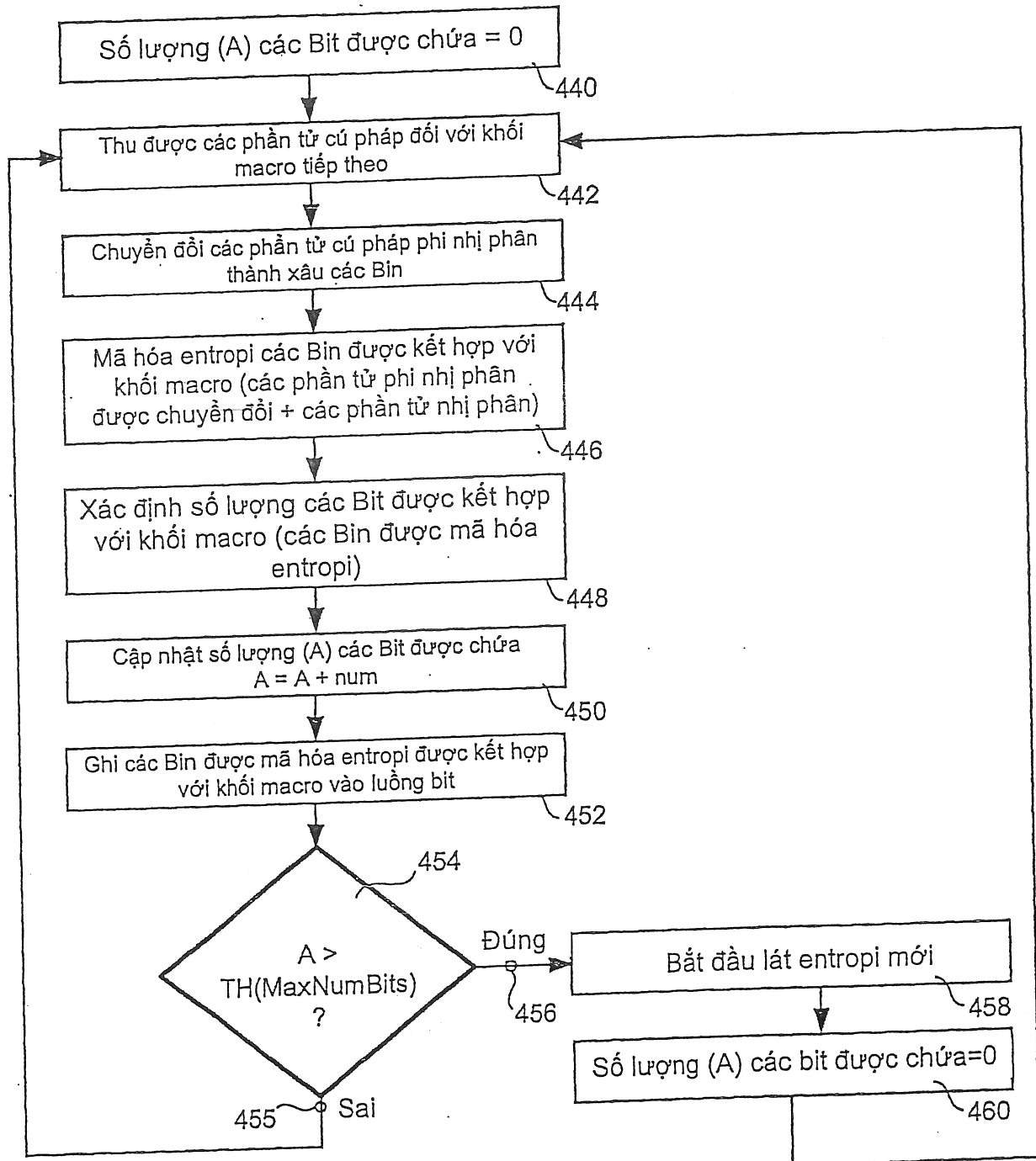
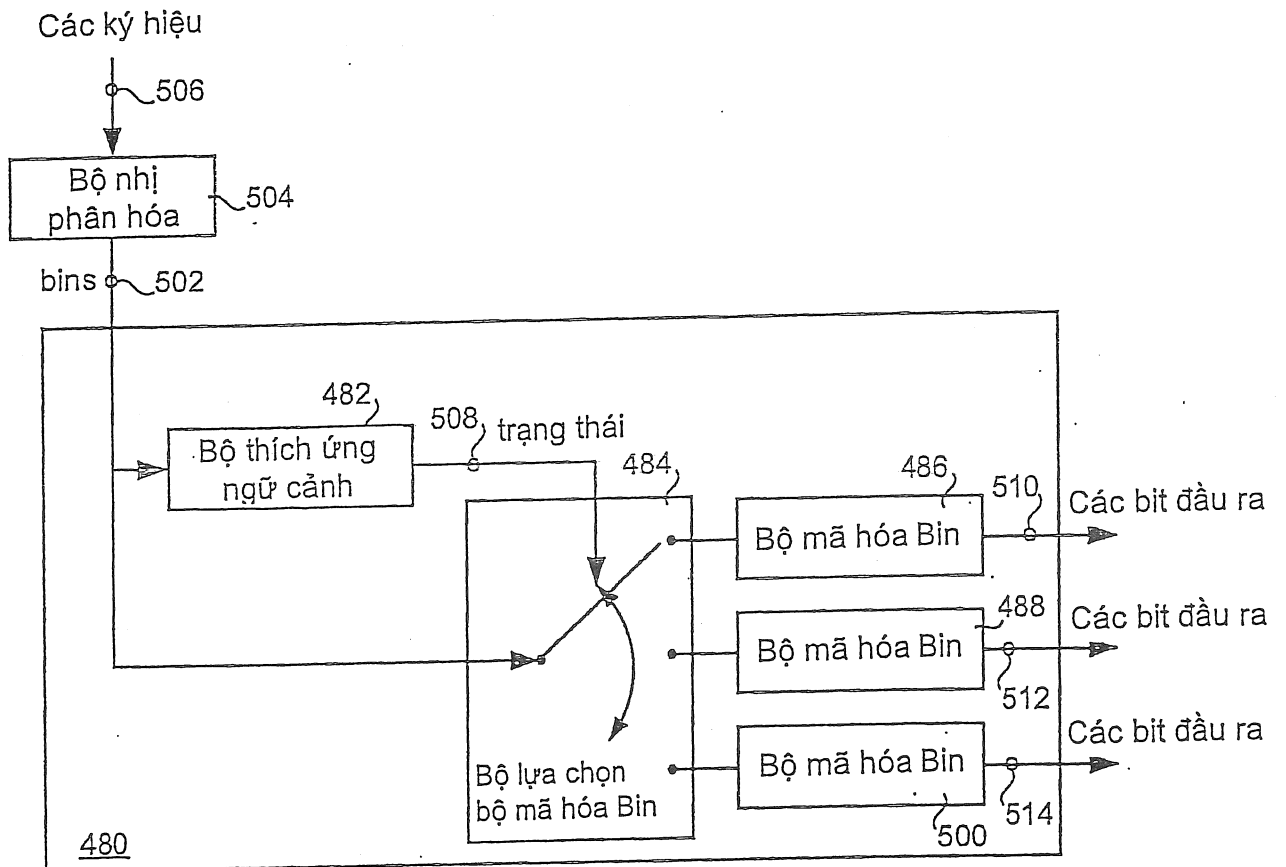


FIG. 17



18/41

FIG. 18

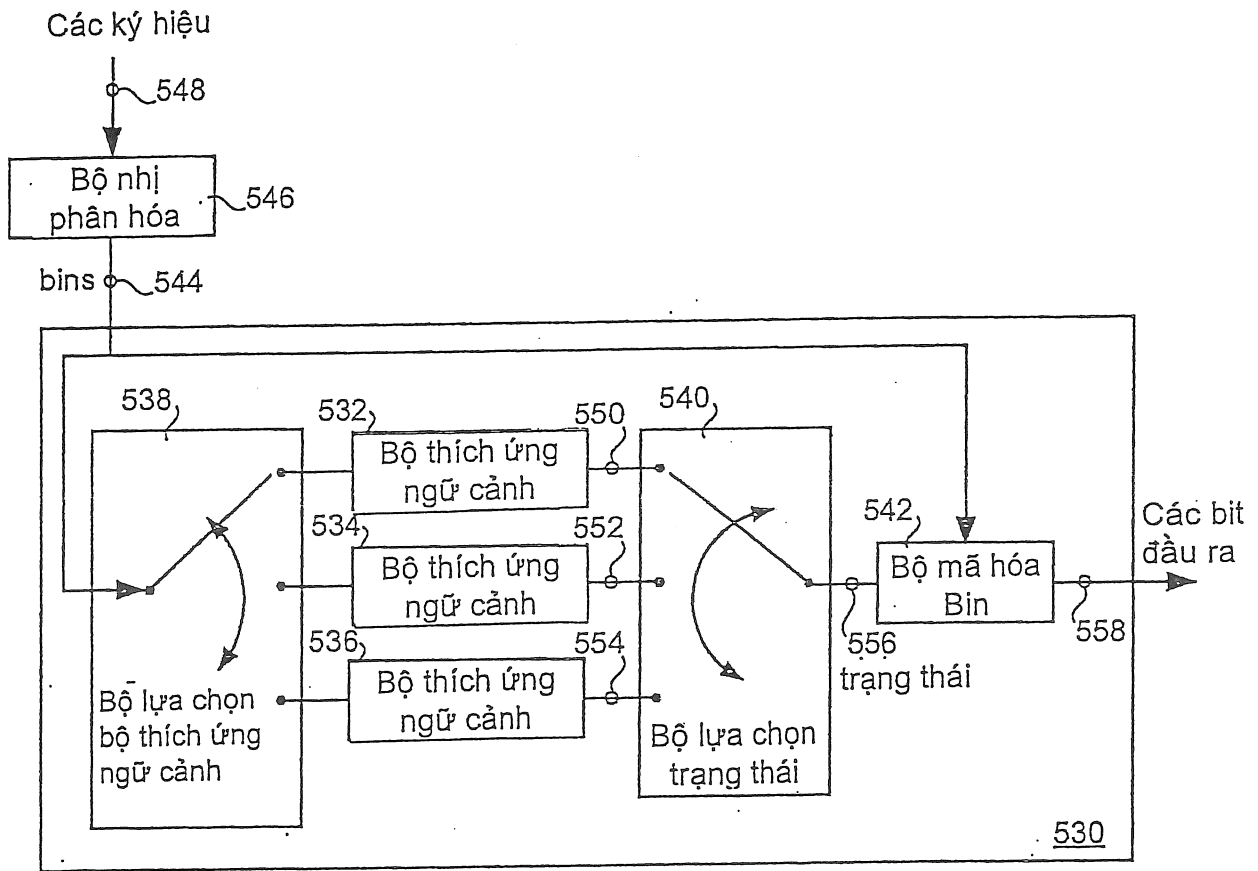
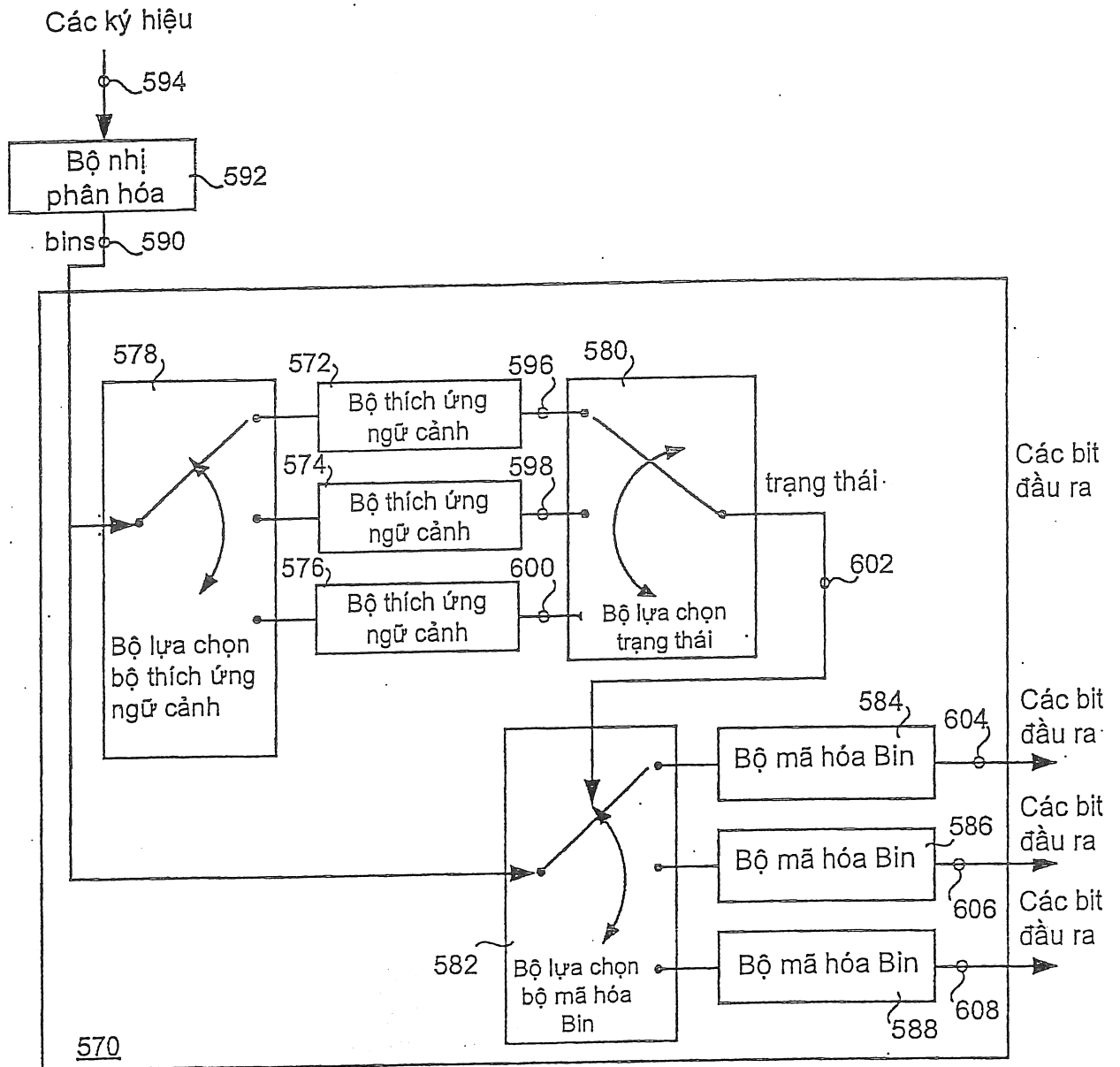


FIG. 19



20/41

FIG. 20

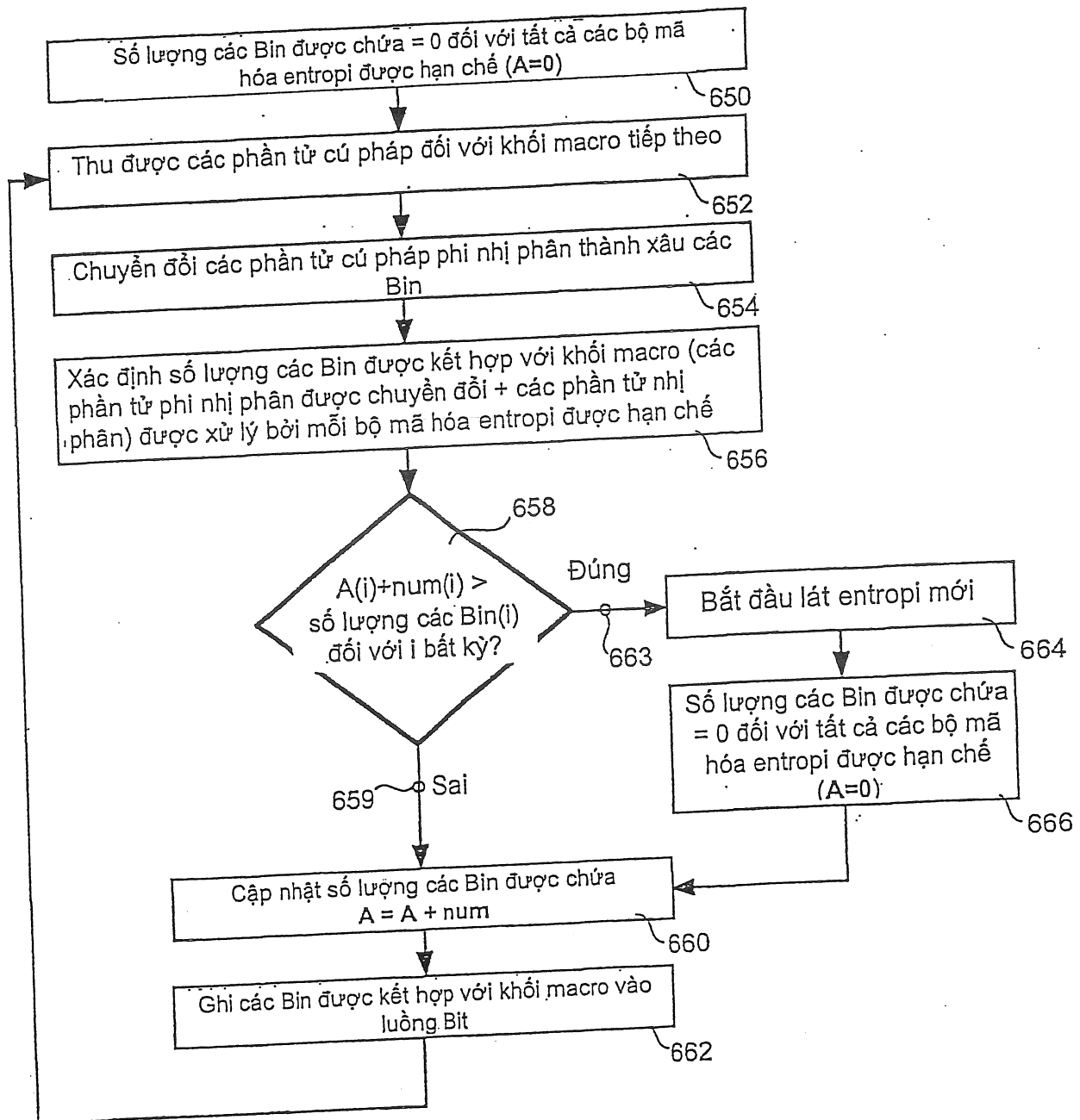


FIG. 21

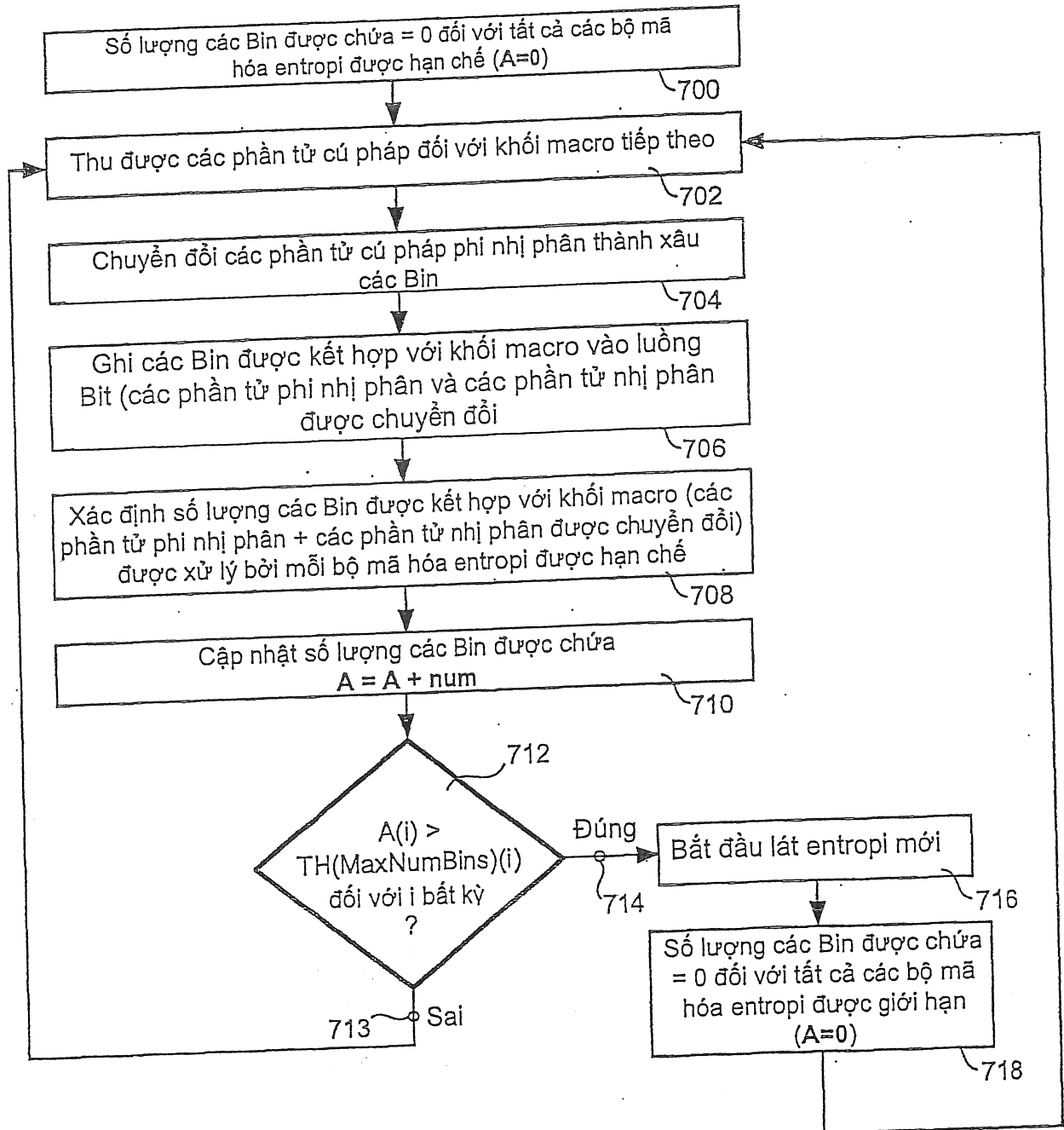
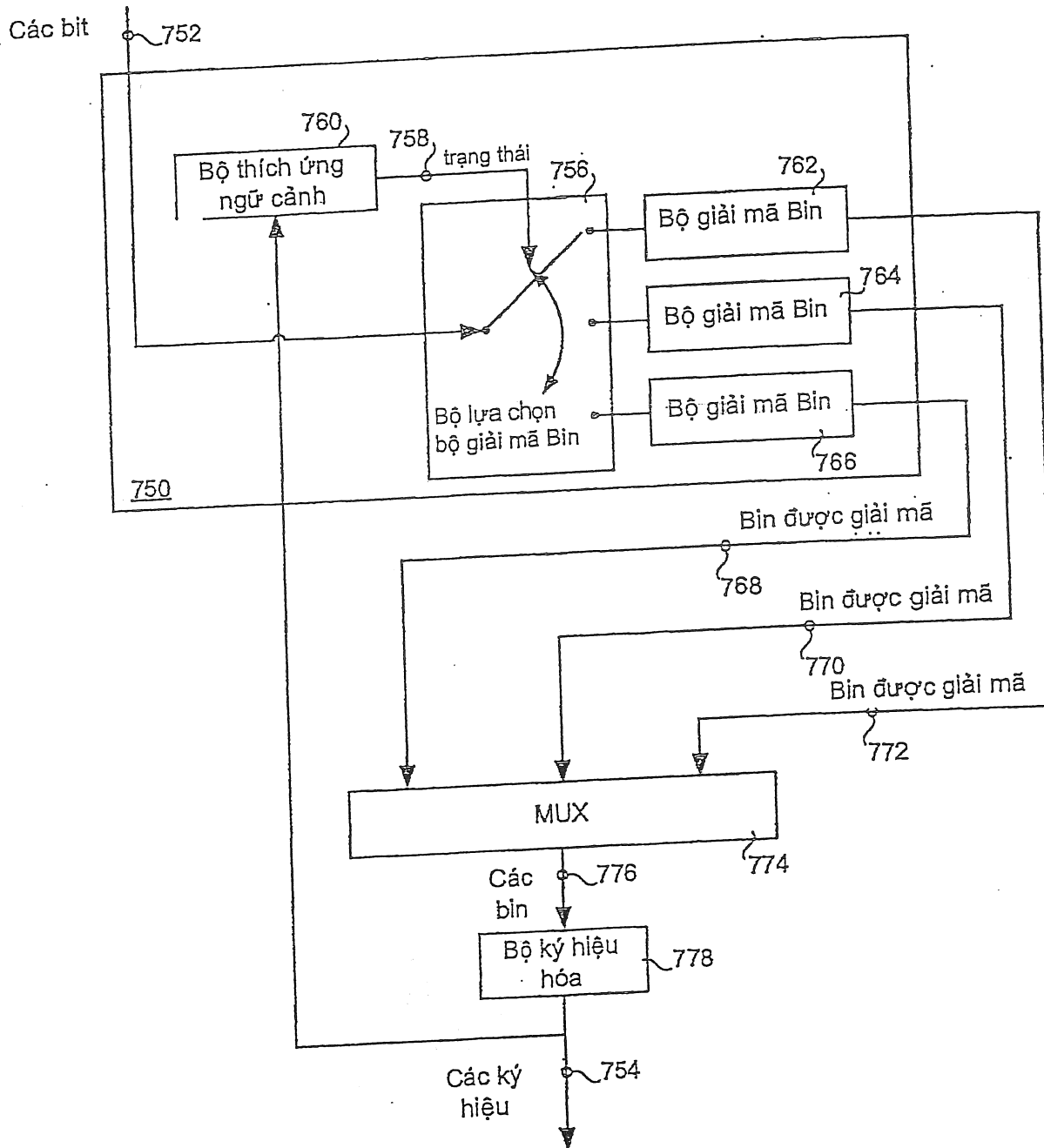
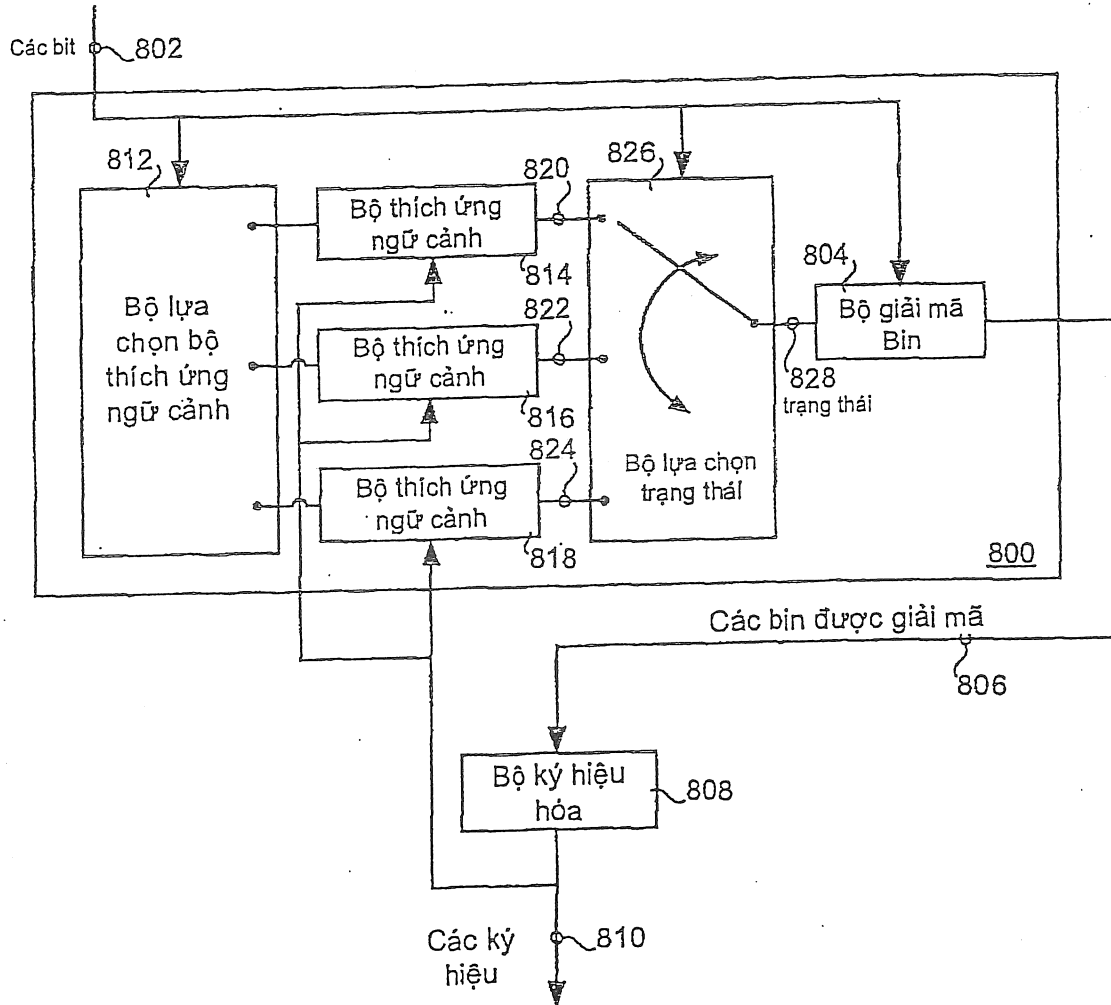


FIG. 22



23/41

FIG. 23



24/41

FIG. 24

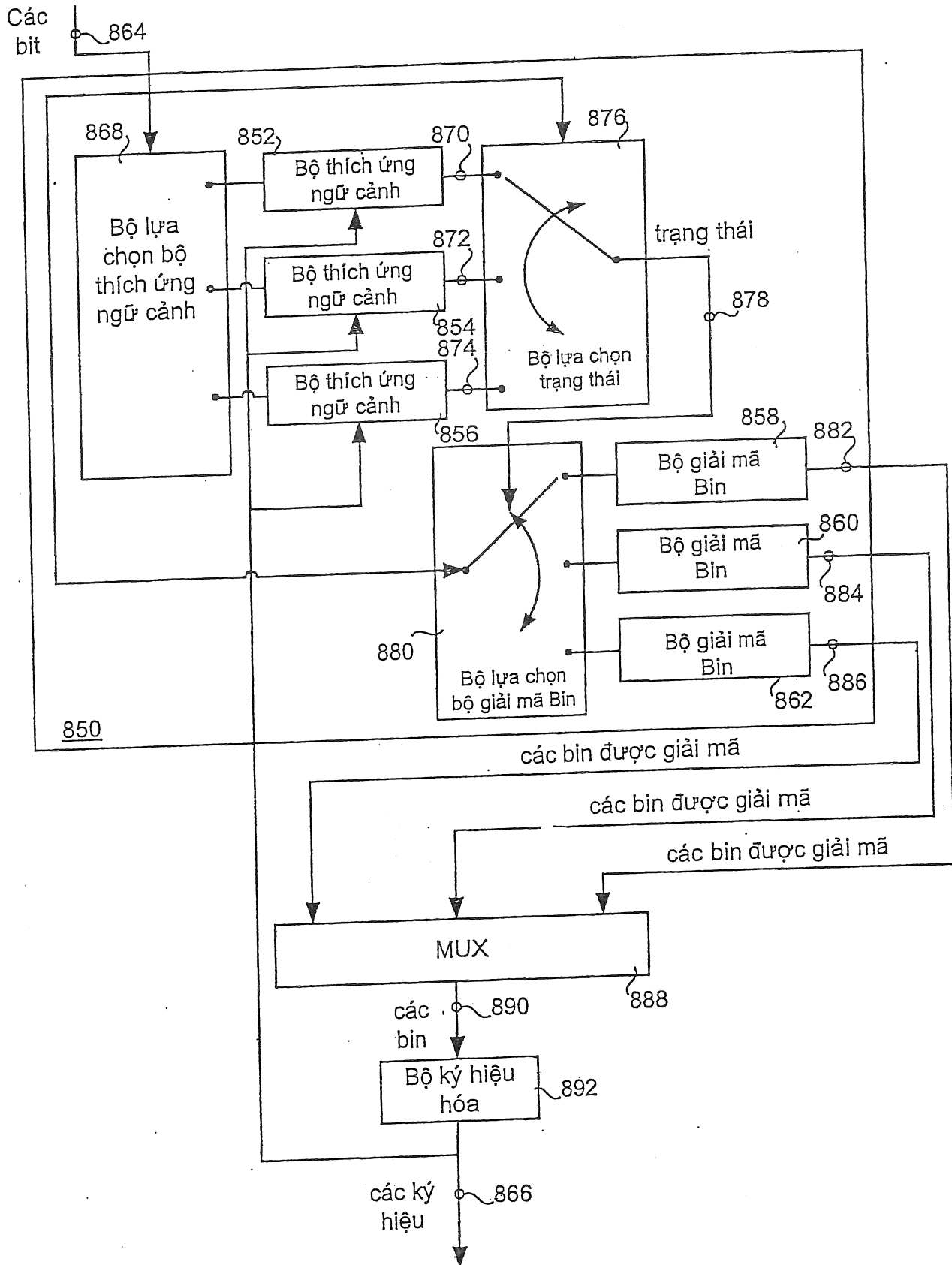


FIG. 25

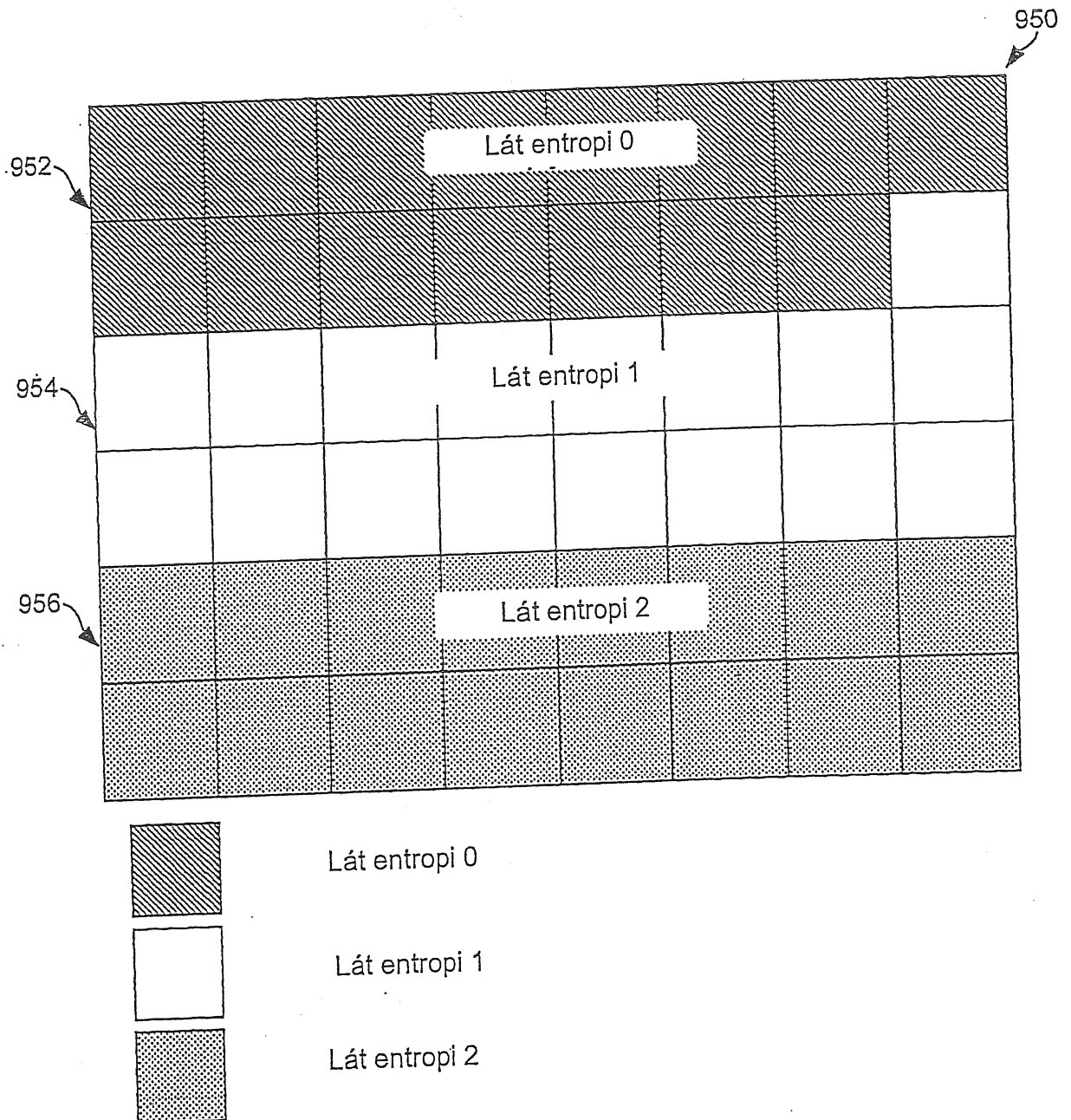


FIG. 26

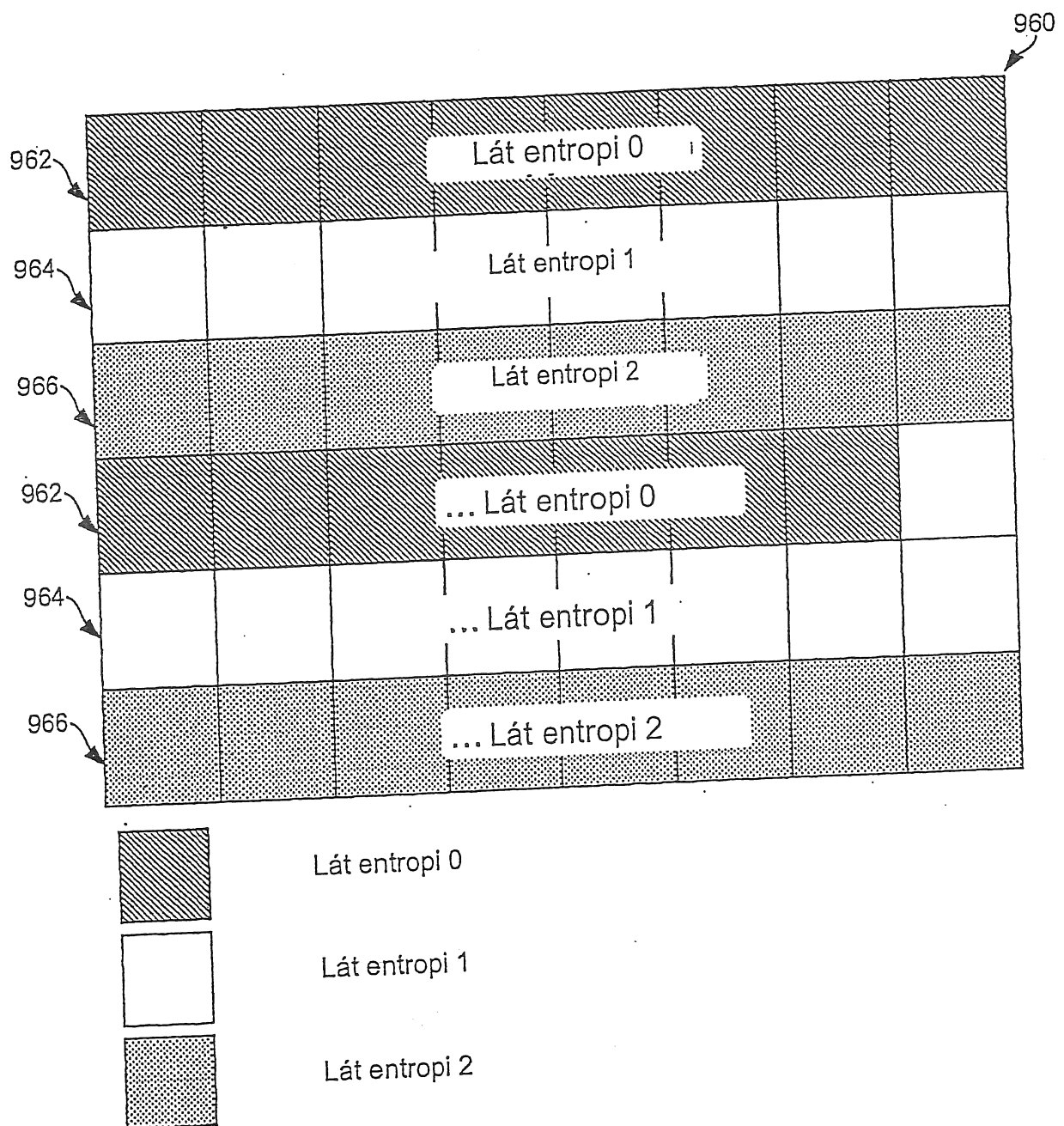


FIG. 27

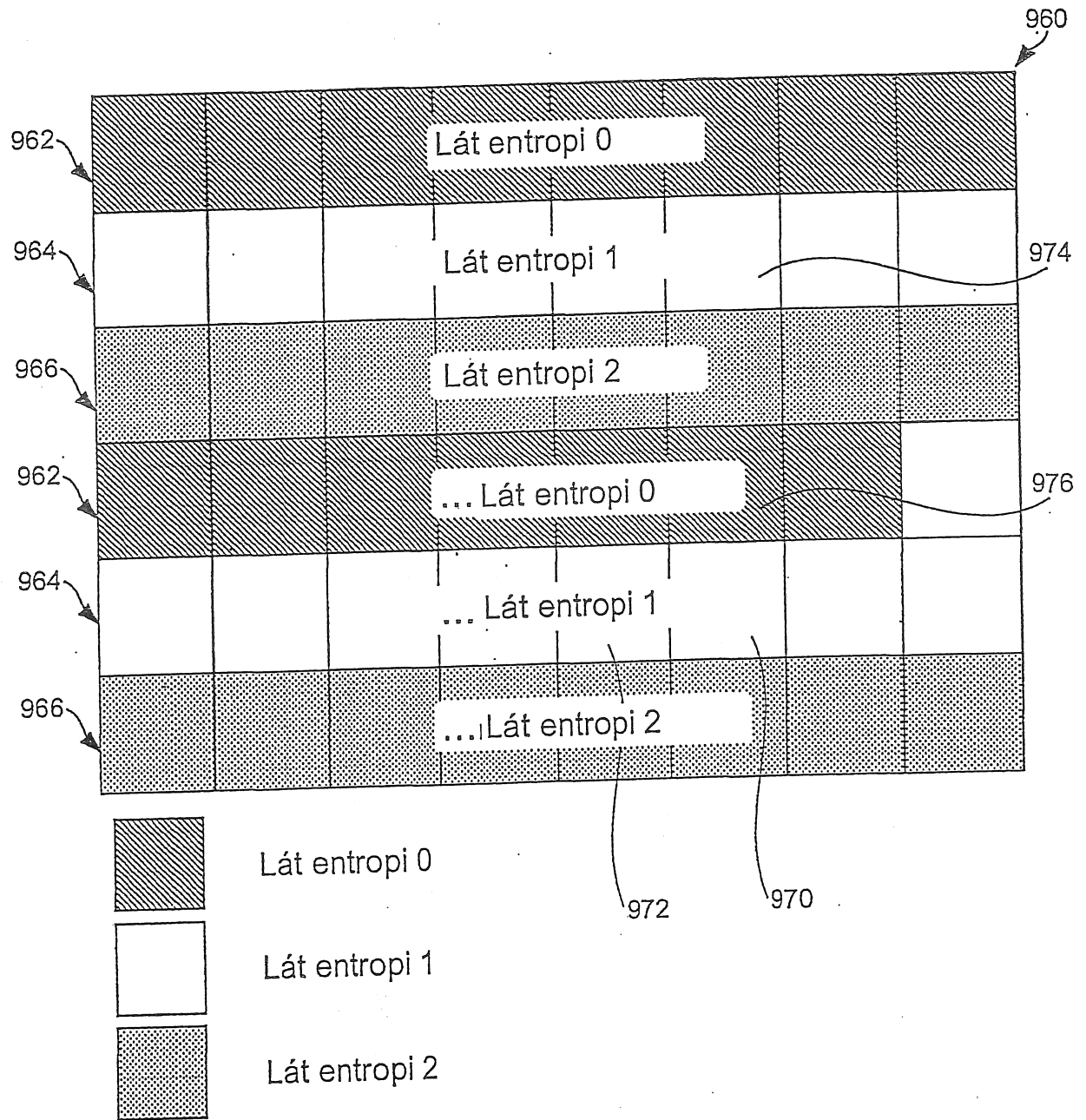
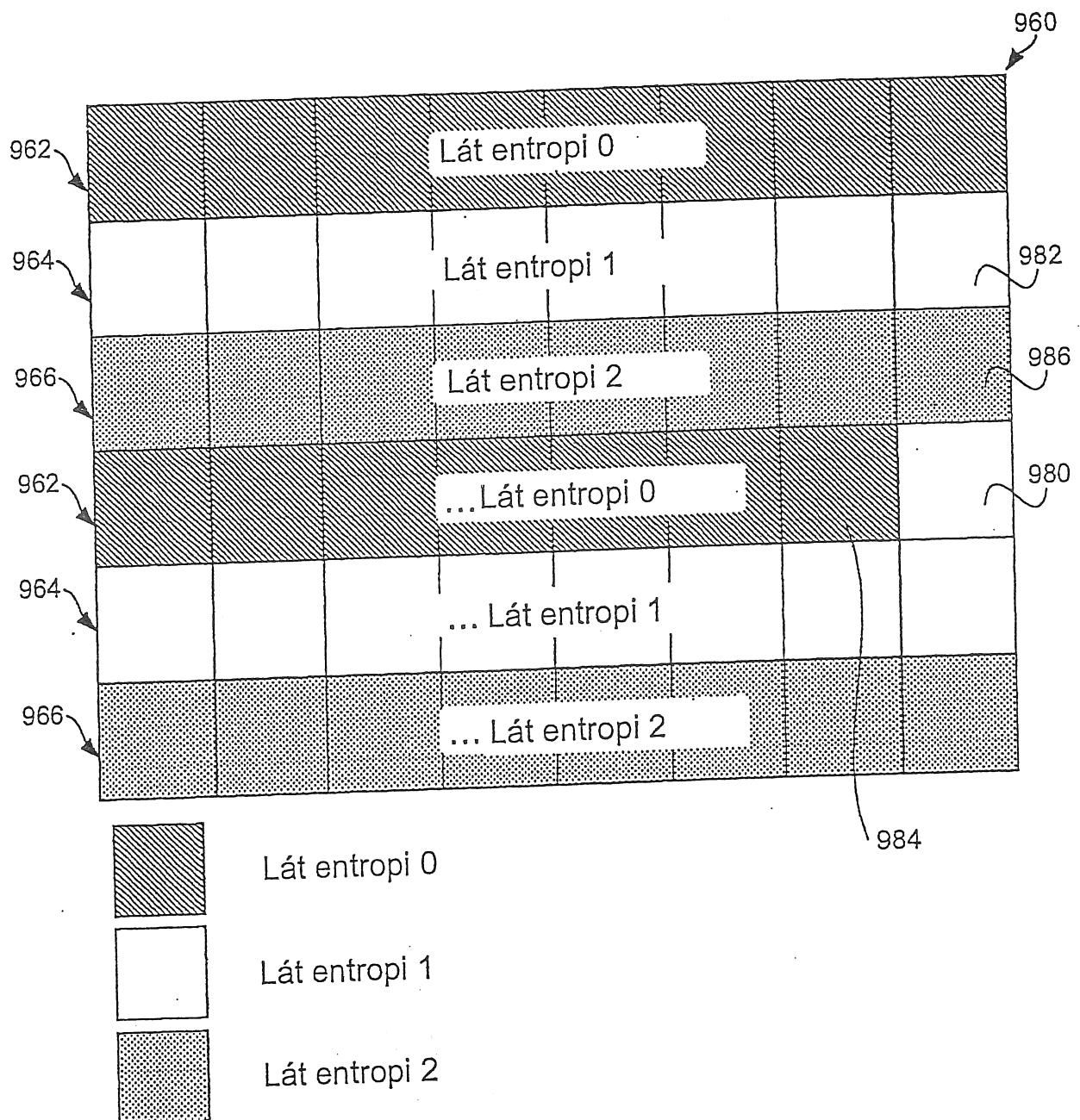


FIG. 28



29/41

FIG. 29

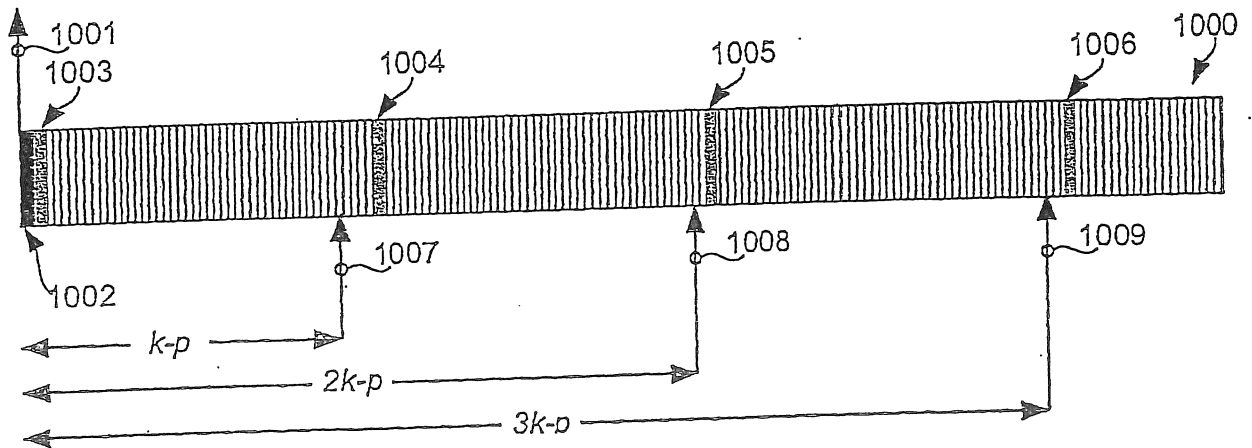


FIG. 30

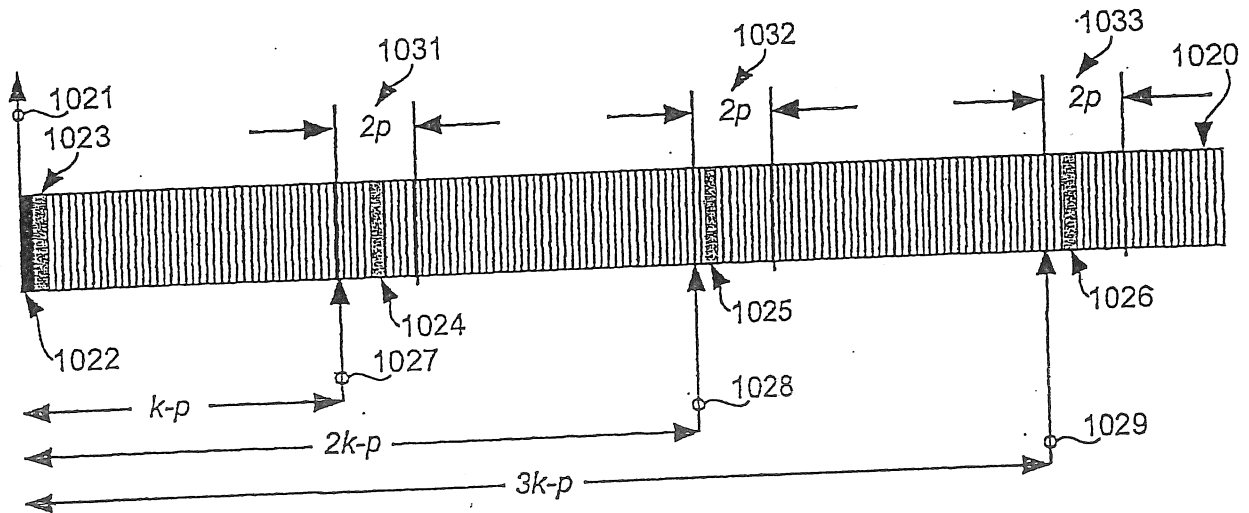
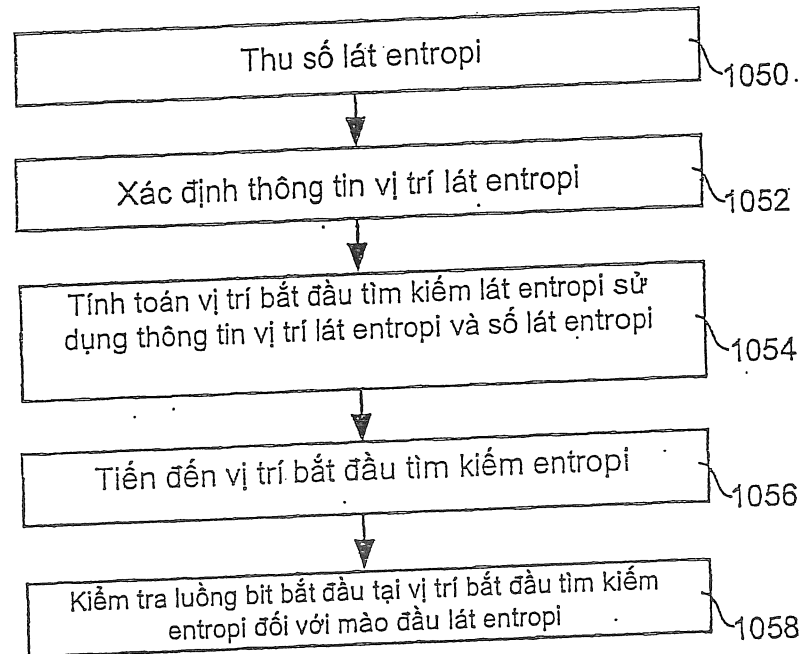


FIG. 31



32/41

FIG. 32

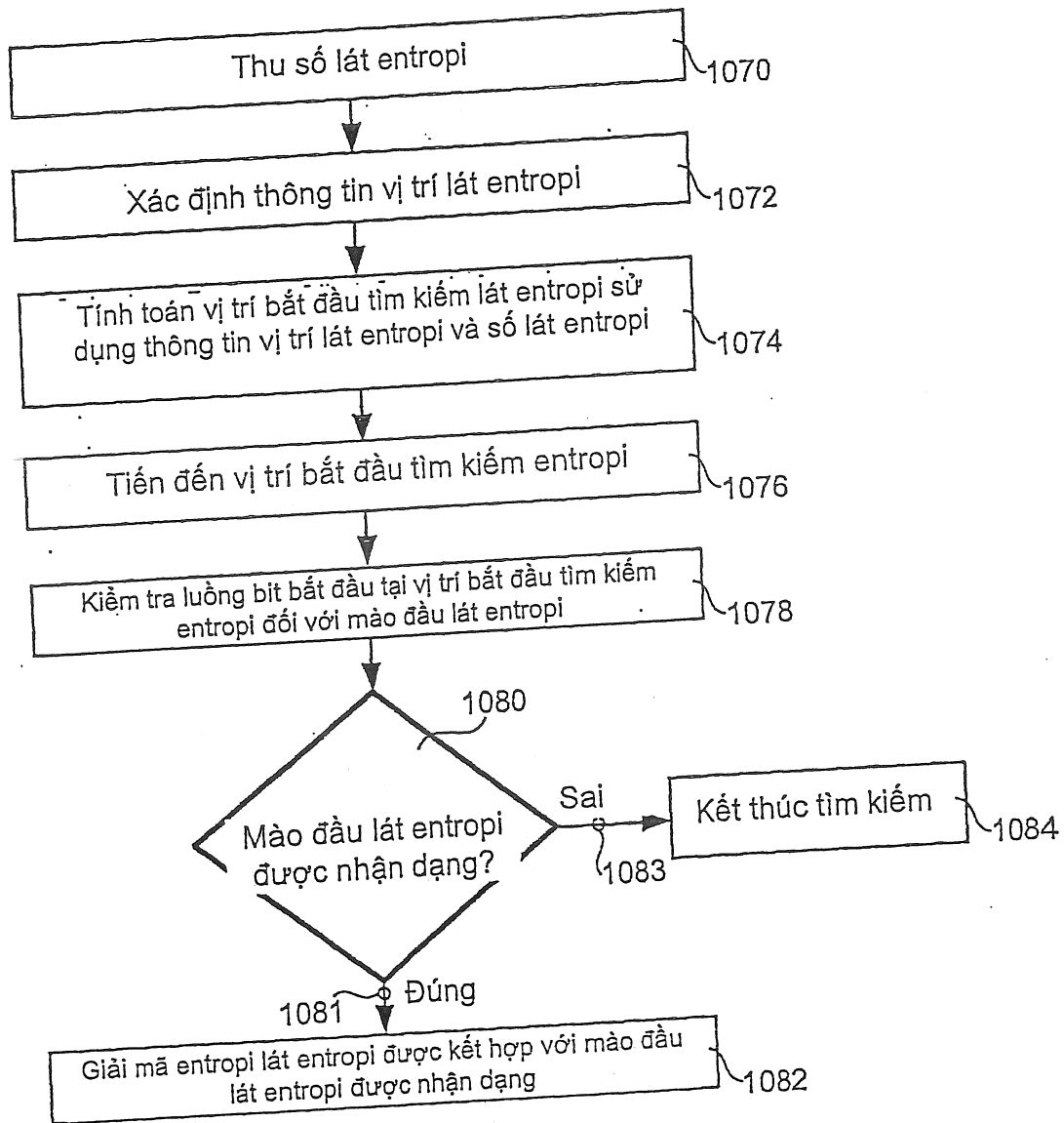


FIG. 33

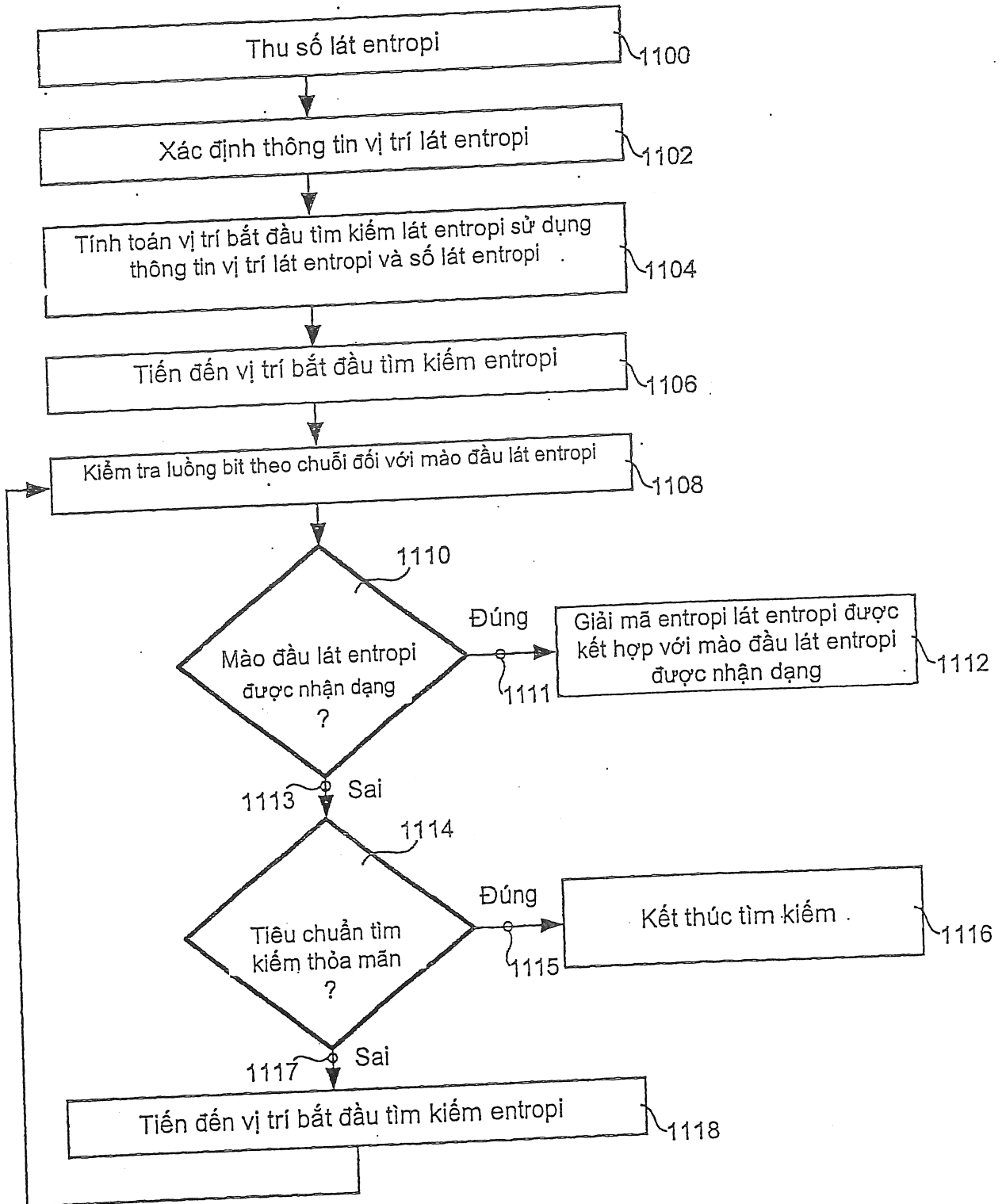


FIG. 34

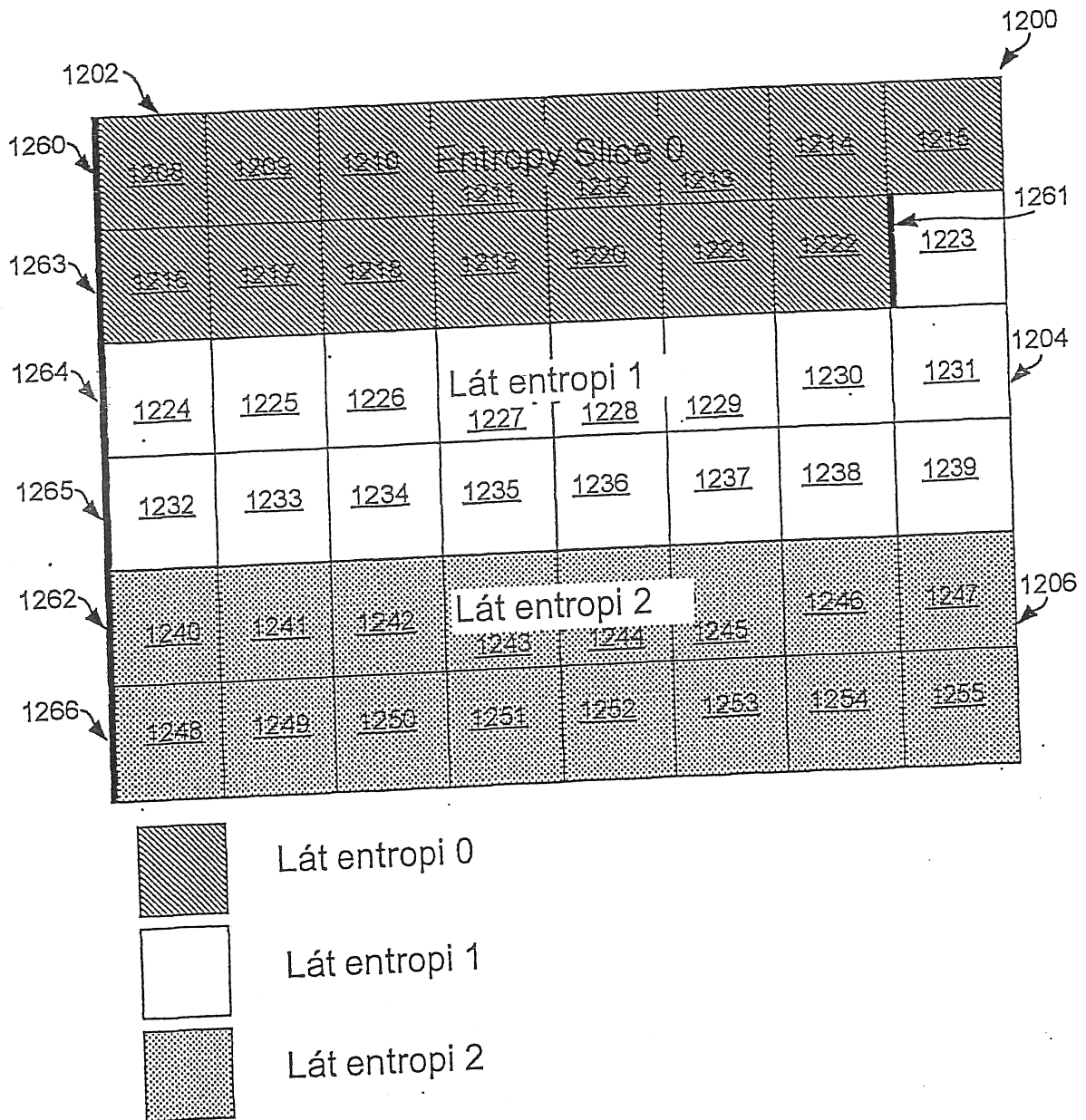


FIG. 35

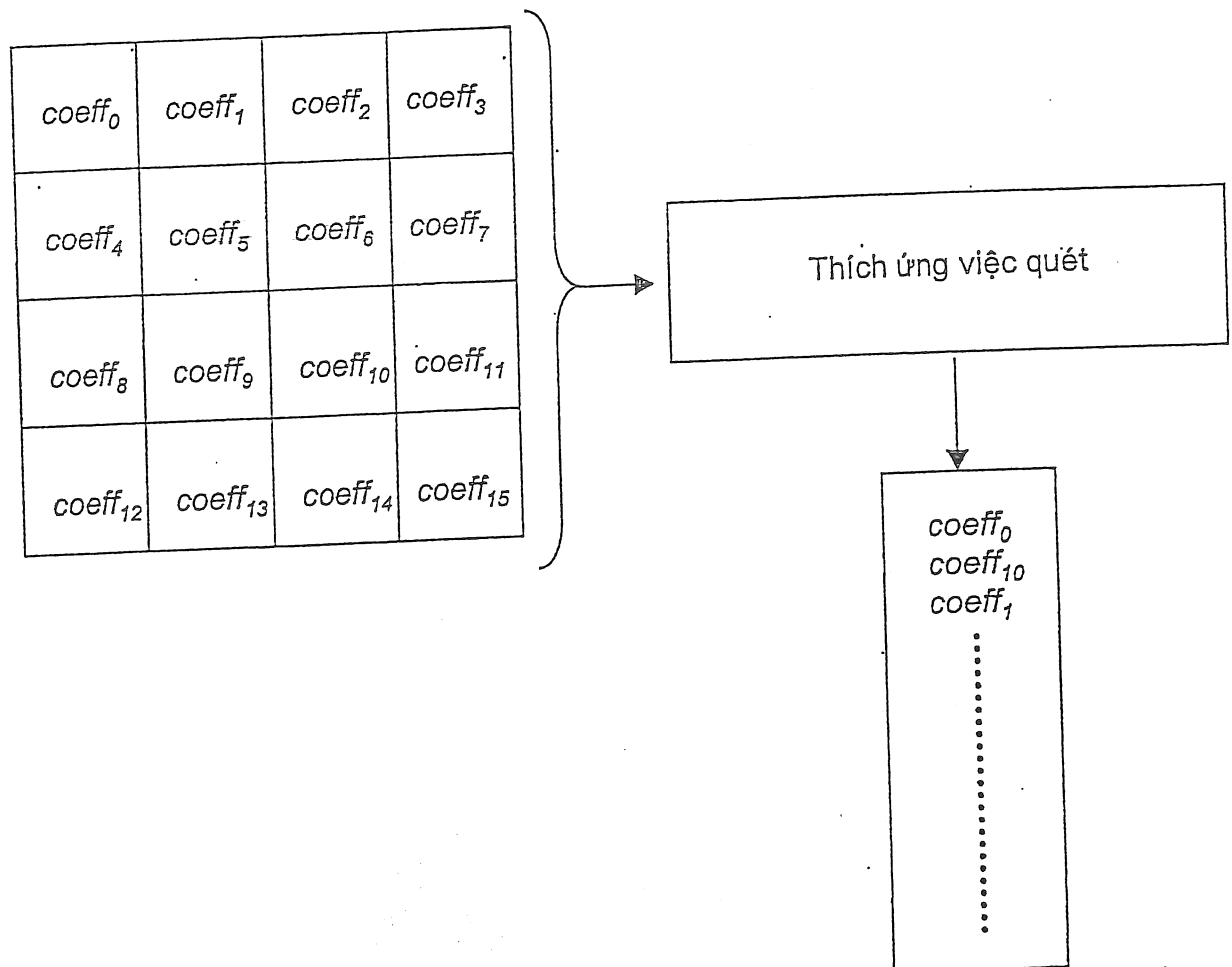


FIG. 36

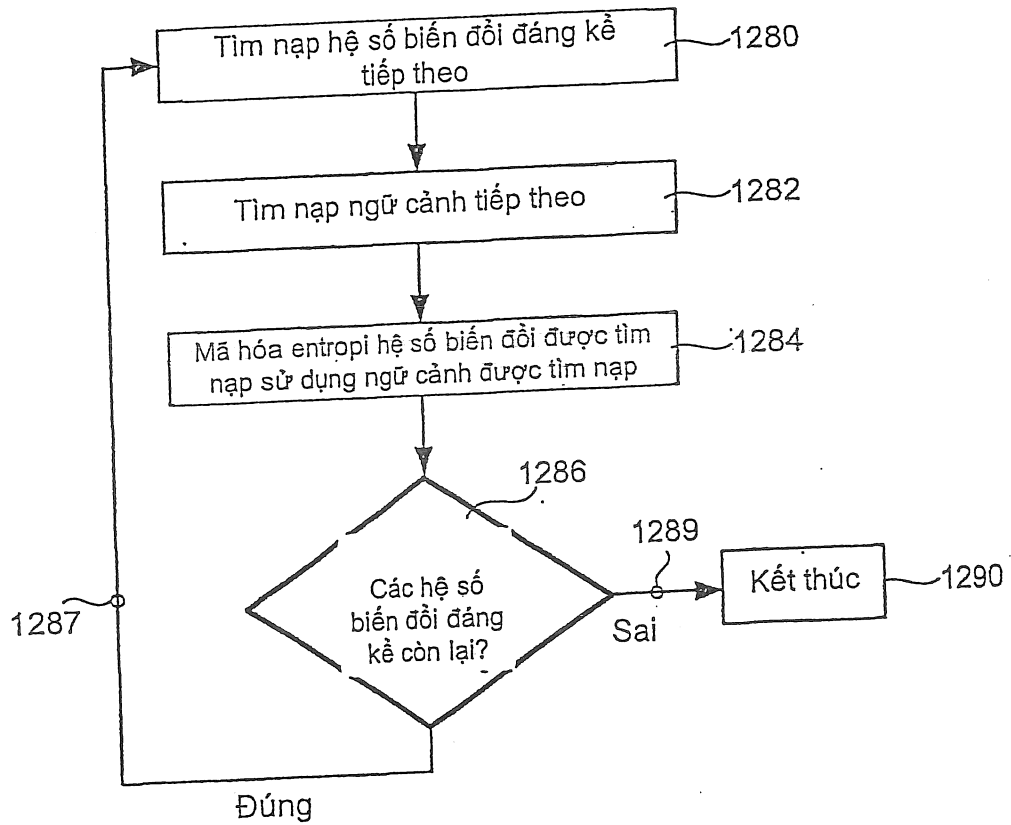


FIG. 37

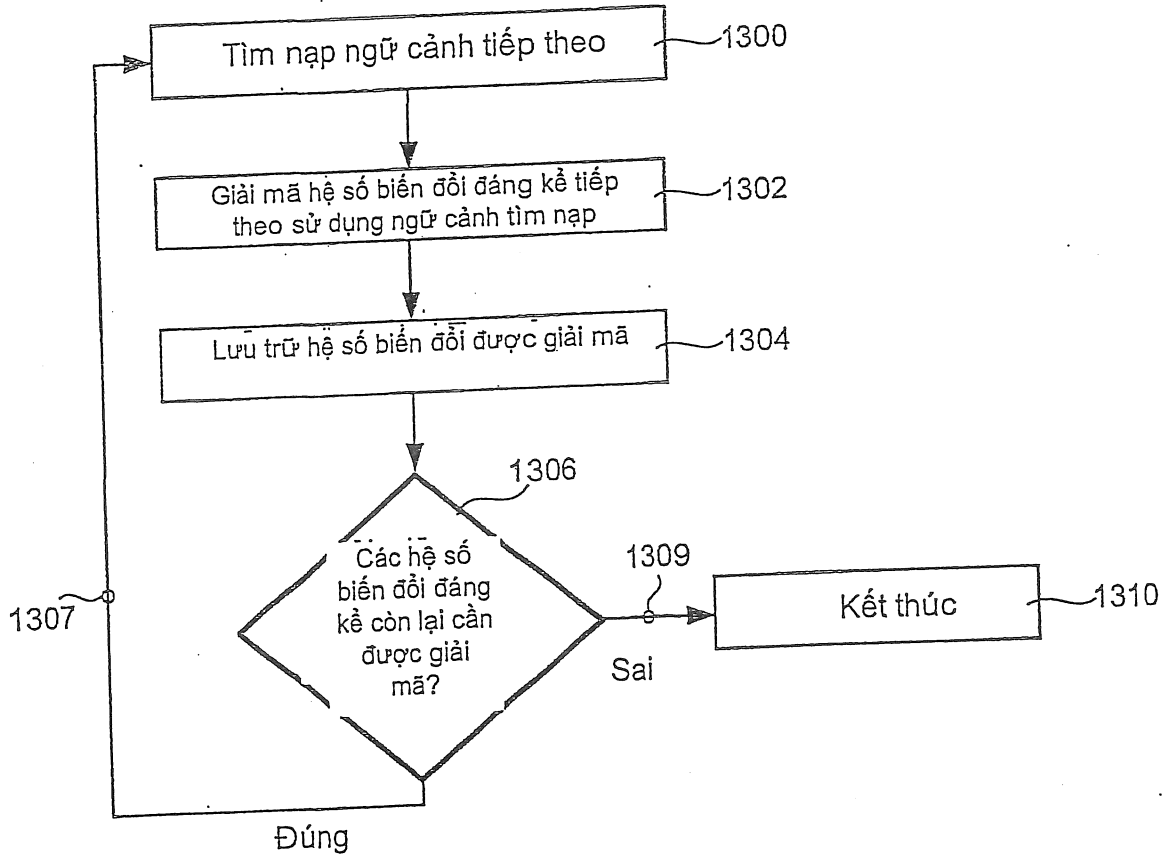


FIG. 38

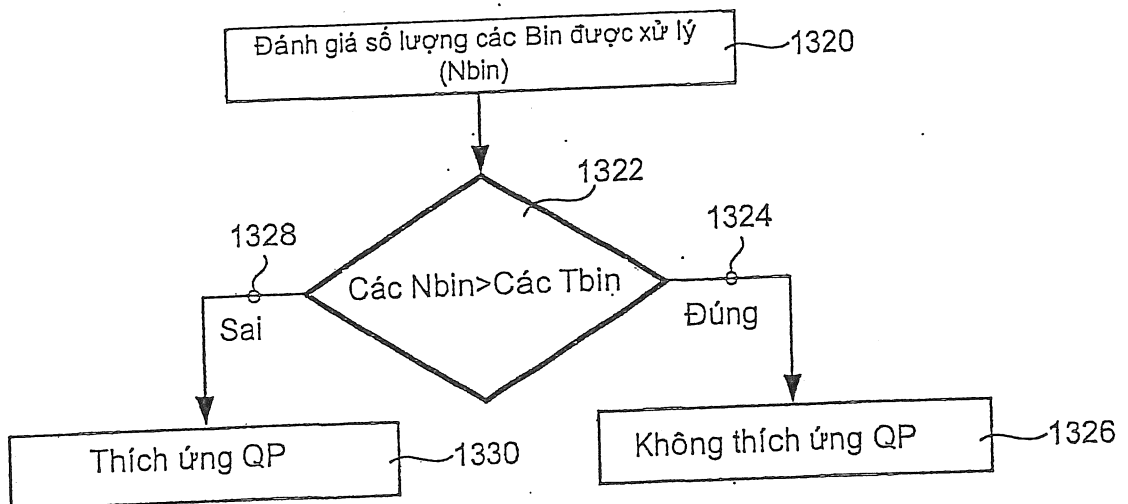


FIG. 39

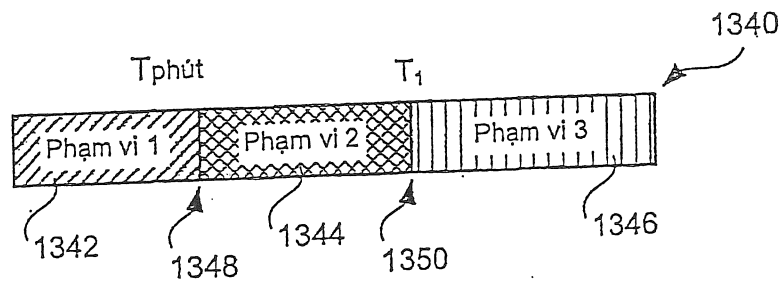


FIG. 40

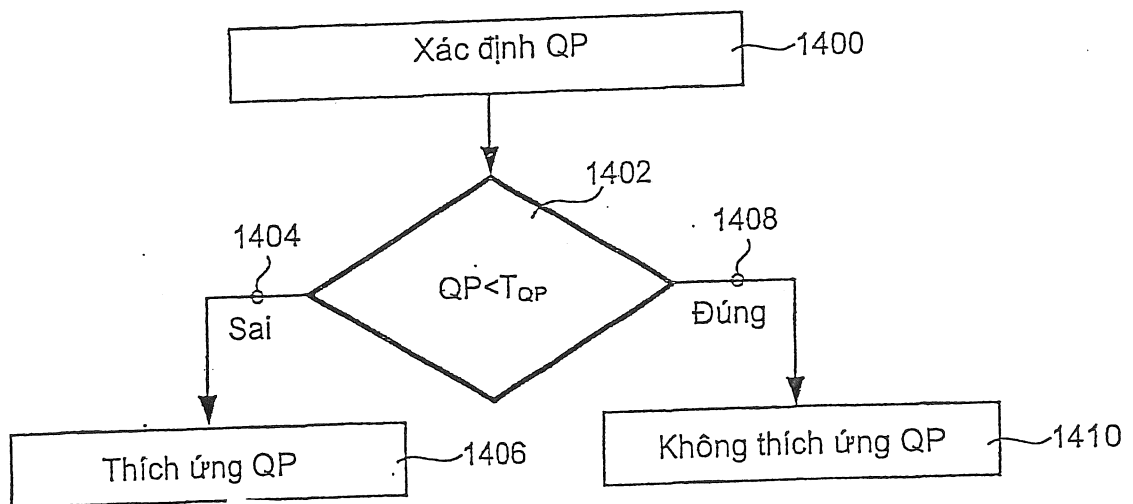


FIG. 41

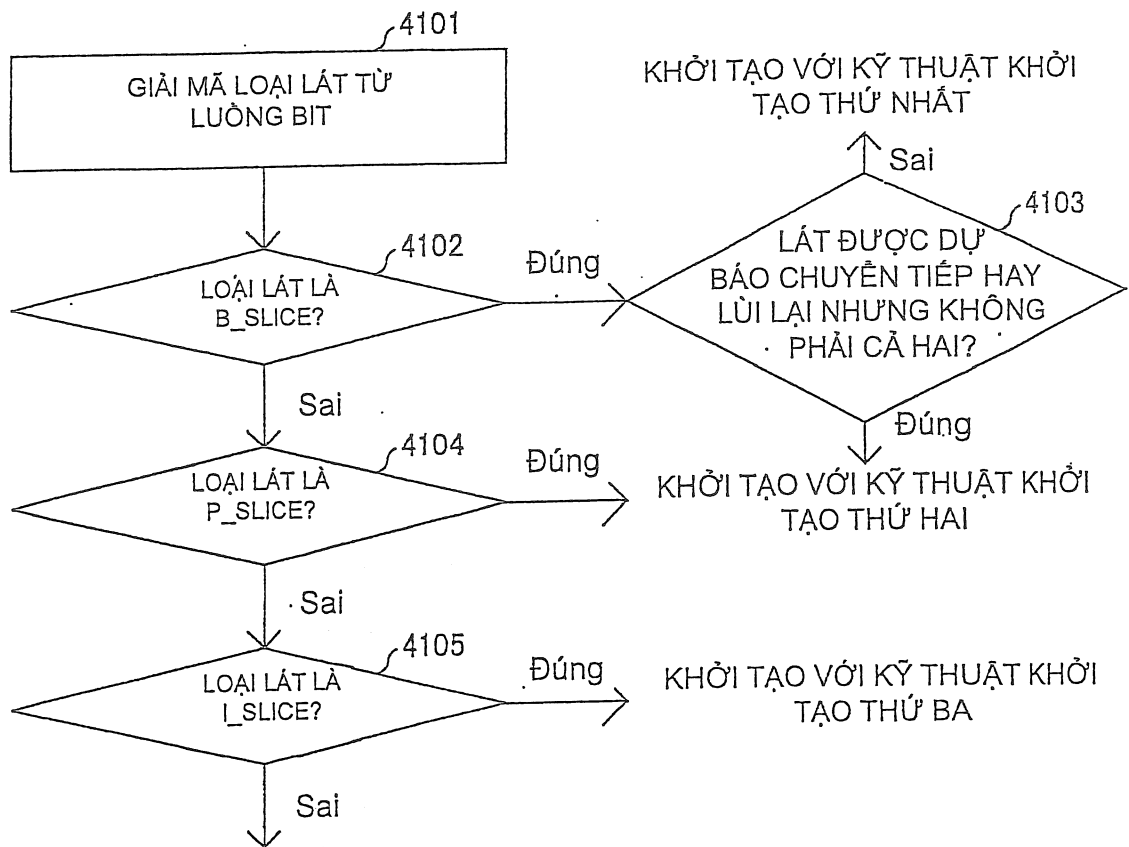
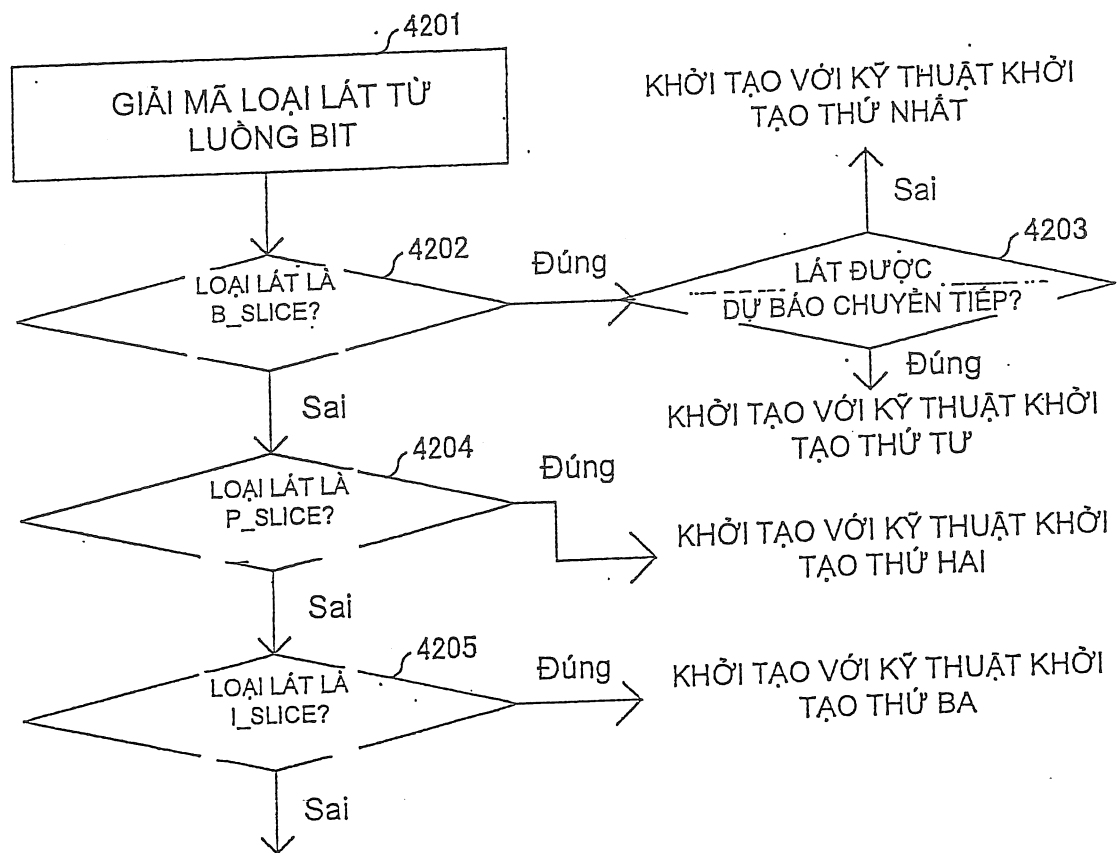


FIG. 42



4 1 / 4 1

FIG. 43

