



(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)



1-0034406

(51)⁷ G10L 19/008; G10L 19/028

(13) B

(21) 1-2018-04070

(22) 14/02/2017

(86) PCT/EP2017/053272 14/02/2017

(87) WO 2017/140666 24/08/2017

(30) 16156209.5 17/02/2016 EP

(45) 26/12/2022 417

(43) 26/11/2018 368A

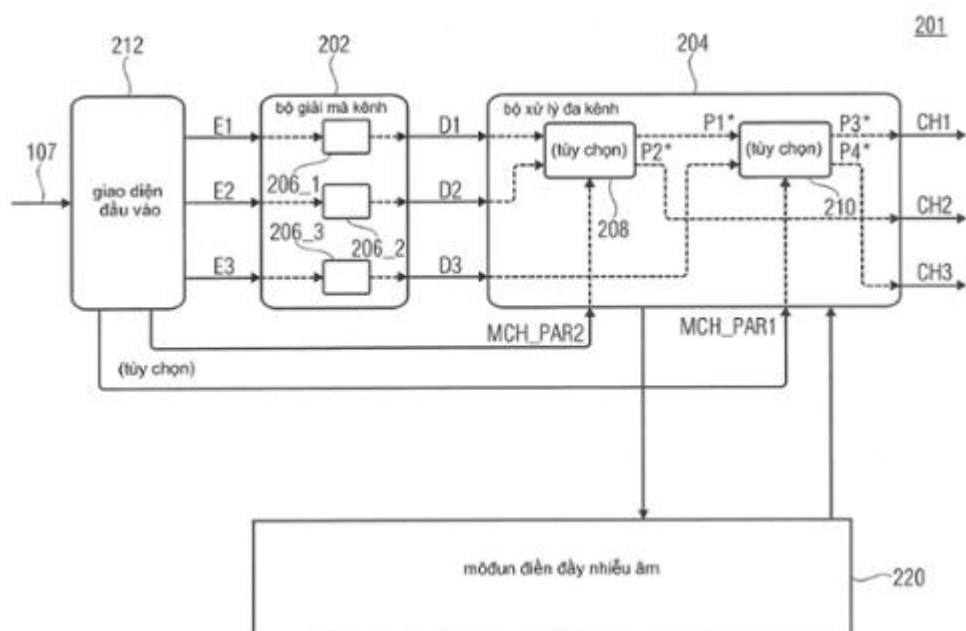
(73) Fraunhofer-Gesellschaft zur Foerderung der angewandten Forschung e.V. (DE)
Hansastraße 27c, 80686 Muenchen, Germany

(72) DICK, Sascha (DE); HELMRICH, Christian (DE); RETTELACH, Nikolaus (DE);
SCHUH, Florian (DE); FUEG, Richard (DE); NAGEL, Frederik (DE).

(74) Công ty Luật TNHH Ambys Hà Nội (AMBYS HANOI)

(54) THIẾT BỊ VÀ PHƯƠNG PHÁP GIẢI MÃ TÍN HIỆU ĐA KÊNH ĐƯỢC MÃ HÓA
TRƯỚC VÀ HỆ THỐNG MÃ HÓA VÀ GIẢI MÃ

(57) Sáng chế đề cập đến thiết bị và phương pháp giải mã tín hiệu đa kênh được mã hóa trước và hệ thống mã hóa và giải mã. Thiết bị giải mã tín hiệu đa kênh được mã hóa của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời được đề xuất. Bộ xử lý đa kênh được làm thích ứng để lựa chọn hai kênh được giải mã từ ba hoặc nhiều hơn ba kênh được giải mã phụ thuộc vào các tham số đa kênh thứ nhất. Hơn nữa, bộ xử lý đa kênh được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý dựa trên các kênh được lựa chọn đã nêu. Môđun điền đầy nhiều âm được làm thích ứng để nhận biết đối với ít nhất một trong số các kênh được lựa chọn, một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và để tạo kênh trộn sử dụng, phụ thuộc vào thông tin phụ, tập hợp con thích hợp gồm ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước mà được giải mã và để điền đầy các vạch phổ của các băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra sử dụng các vạch phổ của kênh trộn.



Lĩnh vực kỹ thuật được đề cập

Sáng chế đề cập đến mã hóa tín hiệu âm thanh và cụ thể là thiết bị và phương pháp điền đầy âm lập thể trong mã hóa đa kênh.

Tình trạng kỹ thuật của sáng chế

Mã hóa âm thanh là miền nén mà xử lý việc khai thác phần dư và phần không liên quan trong các tín hiệu âm thanh.

Trong MPEG USAC (xem ví dụ [3]), mã hóa âm lập thể liên kết của hai kênh được thực hiện sử dụng dự báo phức hợp, MPS 2-1-2 hoặc âm lập thể chưa được hợp nhất với các tín hiệu dư giới hạn băng hoặc toàn băng. Âm thanh nổi MPEG (xem, ví dụ [4]) được tổ hợp theo kiểu phân cấp các hộp Một-đến-Hai (One-To-Two - OTT) và Hai-đến-ba (Two-To-Three (TTT)) để mã hóa liên kết âm thanh đa kênh có hoặc không có việc truyền dẫn tín hiệu dư.

Trong MPEG-H, các phần tử bốn kênh áp dụng theo kiểu phân cấp các hộp âm lập thể MPS 2-1-2 theo sau bởi các hộp âm lập thể dự báo phức hợp/MS xây dựng cây phối lại 4x4 cố định (xem, ví dụ [1]).

AC4 (xem, ví dụ, [6]) giới thiệu các phần tử 3, 4 và 5 kênh mà cho phép phối lại các kênh được truyền dẫn qua ma trận trộn được truyền dẫn và thông tin mã hóa âm lập thể liên kết theo sau. Hơn nữa, các công bố trong tình trạng kỹ thuật gợi ý sử dụng các phép biến đổi trực giao như Phép biến đổi Karhunen-Loeve (KLT) để mã hóa âm thanh đa kênh được nâng cao (xem, ví dụ [7]).

Ví dụ, trong ngữ cảnh âm thanh ba chiều, các kênh loa phát thanh được bố trí theo một vài lớp chiều cao, tạo ra các cặp kênh ngang và dọc. Mã hóa liên kết chỉ của hai kênh được định rõ trong USAC là không đủ để xem xét các mối quan hệ không gian và cảm quan giữa các kênh. Âm thanh nổi MPEG được áp dụng trong bước xử lý/sau bổ sung, các tín hiệu dư được truyền dẫn một cách riêng biệt mà không có khả năng mã hóa âm lập thể liên kết, ví dụ, để sử dụng sự phụ thuộc giữa các kênh dư dọc bên trái và bên phải. Trong AC-4 dành cho các phần tử N kênh được giới thiệu mà cho phép mã hóa một cách hiệu quả các tham số mã hóa liên kết, nhưng thất bại trong các thiết lập loa phát thanh chung với nhiều kênh như được đề xuất cho các cảnh phát lại

đắm chìm (7.1+4, 22.2). Phần tử bốn kênh MPEG cũng được hạn chế chỉ cho 4 kênh và không được áp dụng linh hoạt vào các kênh tùy chọn mà chỉ cho các kênh được cấu hình trước và có số lượng cố định.

Công cụ mã hóa đa kênh MPEG-H cho phép tạo cây tùy chọn của các hộp âm lập thể được mã hóa rời rạc, tức là, các cặp kênh được mã hóa liên kết, xem [2].

Vấn đề mà thường xảy ra trong mã hóa tín hiệu âm thanh thường do lượng tử hóa, ví dụ, lượng tử hóa quang phổ. Việc lượng tử hóa có thể gây ra các lỗi quang phổ. Ví dụ, tất cả các giá trị quang phổ trong băng tần cụ thể có thể thiết lập bằng không trên phía bộ mã hóa như là kết quả của việc lượng tử hóa. Ví dụ, giá trị chính xác của các vạch phổ trước khi lượng tử hóa có thể tương đối thấp và việc lượng tử hóa sau đó có thể dẫn đến tình huống mà trong đó các giá trị quang phổ của tất cả các vạch quang phổ, ví dụ trong băng tần nhất định được thiết lập bằng không. Trên phía bộ giải mã, khi giải mã, nó có thể dẫn đến các lỗi quang phổ không mong muốn.

Các hệ thống mã hóa tiếng nói/âm thanh trong miền tần số hiện đại ví dụ như bộ mã hóa-giải mã Opus/Celt của IETF [9], MPEG-4 (HE-)AAC [10] hoặc, cụ thể, MPEG-D xHE-AAC (USAC) [11], thường có nghĩa là mã hóa các khung âm thanh sử dụng một phép biến đổi dài - khối dài- hoặc tám phép biến đổi ngắn - các khối ngắn - phụ thuộc vào tính dừng của tín hiệu. Ngoài ra, đối với mã hóa bit thấp, các sơ đồ này cung cấp công cụ để khôi phục các hệ số tần số của kênh sử dụng nhiễu âm ngẫu nhiên giả hoặc các hệ số tần số thấp của kênh tương tự. Trong xHE-AAC, các công cụ này lần lượt được biết đến như là điền đầy nhiễu âm và phép lặp băng quang phổ.

Tuy nhiên, đối với đầu vào âm thanh nổi có âm và tạm thời, việc điền đầy nhiễu âm và lặp một mình hạn chế chất lượng mã hóa có thể đạt được tại tốc độ bit rất thấp, chủ yếu do nhiễu hệ số quang phổ của cả hai kênh cần được truyền một cách rõ ràng.

Điền đầy âm lập thể MPEG-H là công cụ theo tham số mà phụ thuộc và việc sử dụng phép trộn giảm khung trước để cải thiện việc điền đầy các lỗi quang phổ gây ra bởi việc lượng tử hóa trong miền tần số. Giống như điền đầy nhiễu âm, điền đầy âm lập thể hoạt động trực tiếp trong miền MDCT của bộ giải mã lõi MPEG, xem [1], [5], [8].

Tuy nhiên, việc sử dụng âm thanh nổi MPEG và điền đầy âm lập thể trong MPEG-H bị hạn chế ở các phần tử cặp kênh cố định và do đó không thể sử dụng các

sự phụ thuộc liên kênh biến thiên theo thời gian.

Công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) trong MPEG-H cho phép làm thích ứng để thay đổi các sự phụ thuộc liên kênh nhưng không cho phép điền đầy âm lập thể do việc sử dụng các phần tử kênh đơn trong các cấu hình hoạt động điển hình. Tình trạng kỹ thuật không bộc lộ các cách tối ưu theo cảm quan để tạo ra các phép trộn giảm của khung trước trong trường hợp biến thiên theo thời gian, các cặp kênh được mã hóa liên kết một cách tùy chọn. Sử dụng điền đầy nhiễu âm như là cách thay thế cho điền đầy âm lập thể trong sự kết hợp với MCT để điền đầy các lỗ quang phổ sẽ dẫn đến các thành phần lạ nhiễu âm, đặc biệt là cho các tín hiệu âm.

Tài liệu tham khảo

- [1] ISO/IEC international standard 23008-3:2015, “Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 3: 3D audio,” March 2015
- [2] ISO/IEC amendment 23008-3:2015/PDAM3, “Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 3: 3D audio, Amendment 3: MPEG-H 3D Audio Phase 2,” July 2015
- [3] International Organization for Standardization, ISO/IEC 23003-3:2012, “Information Technology – MPEG audio – Part 3: Unified speech and audio coding,” Geneva, Jan. 2012
- [4] ISO/IEC 23003-1:2007 - Information technology – MPEG audio technologies Part 1: MPEG Surround
- [5] C. R. Helmrich, A. Niedermeier, S. Bayer, B. Edler, “Low-Complexity Semi-Parametric Joint-Stereo Audio Transform Coding,” in Proc. EUSIPCO, Nice, September 2015
- [6] ETSI TS 103 190 V1.1.1 (2014-04) – Digital Audio Compression (AC-4) Standard
- [7] Yang, Dai and Ai, Hongmei and Kyriakakis, Chris and Kuo, C.-C. Jay, 2001: Adaptive Karhunen-Loeve Transform for Enhanced Multichannel Audio Coding, <http://ict.usc.edu/pubs/Adaptive%20Karhunen->

Loeve%20Transform%20for

%20Enhanced%20Multichannel%20Audio%20Coding.pdf

- [8] European Patent Application, Publication EP 2 830 060 A1: “Noise filling in multichannel audio coding”, published on 28 January 2015
- 9] Internet Engineering Task Force (IETF), RFC 6716, “Definition of the Opus Audio Codec,” Int. Standard, Sep. 2012. Available online at:
<http://tools.ietf.org/html/rfc6716>
- [10] International Organization for Standardization, ISO/IEC 14496-3:2009, “Information Technology – Coding of audio-visual objects – Part 3: Audio,” Geneva, Switzerland, Aug. 2009
- [11] M. Neuendorf et al., “MPEG Unified Speech and Audio Coding – The ISO/MPEG Standard for High-Efficiency Audio Coding of All Content Types,” in Proc. 132nd AES Convention, Budapest, Hungary, Apr. 2012. Also to appear in the Journal of the AES, 2013

Mục đích của sáng chế

Mục đích của sáng chế là cung cấp các khái niệm mã hóa âm thanh được cải thiện. Mục đích của sáng chế được giải quyết bằng thiết bị giải mã theo điểm 1, thiết bị mã hóa theo điểm 15, phương pháp giải mã theo điểm 18, phương pháp mã hóa theo điểm 19, chương trình máy tính theo điểm 20 và tín hiệu đa kênh được mã hóa theo điểm 21.

Bản chất kỹ thuật của sáng chế

Thiết bị giải mã tín hiệu đa kênh được mã hóa của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời được cung cấp. Bộ xử lý đa kênh được làm thích ứng để lựa chọn hai kênh được giải mã từ ba hoặc nhiều hơn ba kênh được giải mã phụ thuộc vào các tham số đa kênh thứ nhất ba hoặc nhiều hơn ba kênh đầu ra âm thanh thứ nhất. Hơn nữa, bộ xử lý đa kênh được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý dựa trên các kênh được lựa chọn đã nêu. Môđun điền đầy nhiễu âm được làm thích ứng để nhận biết đối với ít nhất một trong số nhiều kênh được lựa chọn, một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và để tạo kênh trộn sử dụng, phụ

thuộc vào thông tin phụ, tập hợp con thích hợp của ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước mà được giải mã và để điền đầy các vạch phổ của các băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra sử dụng các vạch phổ của kênh trộn.

Theo các phương án, thiết bị giải mã tín hiệu đa kênh được mã hóa trước của khung trước để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời và giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời được cung cấp.

Thiết bị bao gồm giao diện, bộ giải mã kênh, bộ xử lý đa kênh để tạo ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời, môđun điền đầy nhiều âm.

Giao diện được làm thích ứng để nhận tín hiệu đa kênh được mã hóa hiện thời và để nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất.

Bộ giải mã kênh được làm thích ứng để giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh được giải mã của khung hiện thời.

Bộ xử lý đa kênh được làm thích ứng để lựa chọn cặp được lựa chọn gồm hai kênh được giải mã từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã phụ thuộc vào các tham số đa kênh thứ nhất.

Hơn nữa, bộ xử lý đa kênh được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý dựa trên cặp được lựa chọn thứ nhất gồm hai kênh được giải mã để thu được tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã.

trong đó, bộ xử lý đa kênh tạo ra cặp thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý dựa trên cặp được lựa chọn thứ nhất gồm hai kênh được giải mã, môđun điền đầy nhiều âm được làm thích ứng để nhận biết ít nhất một trong số hai kênh của cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã, một hoặc nhiều băng tần mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và để tạo kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải tất cả trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và để điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra sử dụng các vạch phổ của kênh trộn, trong đó môđun điền đầy nhiều

âm được làm thích ứng để lựa chọn hai hoặc nhiều hơn hai kênh đầu ra âm thanh trước mà được sử dụng để tạo kênh trộn từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào thông tin phụ.

Khái niệm cụ thể của các phương án mà có thể được sử dụng bởi môđun điền đầy nhiều âm mà cụ thể làm thế nào để tạo ra và điền đầy nhiều âm được đề cập đến như là điền đầy âm thanh nổi.

Hơn nữa, thiết bị mã hóa tín hiệu đa kênh có ít nhất ba kênh được cung cấp.

Thiết bị bao gồm bộ xử lý phép lặp được làm thích ứng để tính toán, trong bước lặp thứ nhất, các giá trị tương quan liên kênh giữa mỗi cặp của ít nhất ba kênh, để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị cao hơn ngưỡng, và để xử lý cặp được lựa chọn sử dụng phép toán xử lý đa kênh để suy ra các tham số đa kênh ban đầu cho các cặp được lựa chọn và để suy ra các kênh được xử lý thứ nhất.

trong đó bộ xử lý phép lặp được làm thích ứng để thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai sử dụng ít nhất một kênh trong số các kênh được xử lý để suy ra thêm các tham số đa kênh và các kênh được xử lý thứ hai.

Hơn nữa, thiết bị bao gồm bộ mã hóa kênh được làm thích ứng để mã hóa các kênh từ xử lý phép lặp được thực hiện bởi bộ xử lý phép lặp để thu được các kênh được mã hóa.

Hơn nữa, thiết bị bao gồm giao diện đầu ra được làm thích ứng để tạo tín hiệu đa kênh được mã hóa có các kênh được mã hóa, các tham số đa kênh ban đầu và các tham số đa kênh khác và có thông tin biểu thị liệu thiết bị giải mã có điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà được giải mã trước bởi thiết bị giải mã.

Hơn nữa, phương pháp giải mã tín hiệu đa kênh được mã hóa trước hoặc khung trước để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh kênh đầu ra âm thanh trước, và giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh. Phương pháp bao gồm:

- Nhận tín hiệu đa kênh được mã hóa hiện thời, và nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất.
- Giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu

được ba hoặc nhiều hơn ba kênh được giải mã của - khung hiện thời.

- Lựa chọn cặp được lựa chọn gồm hai kênh được giải mã từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã phụ thuộc vào các tham số đa kênh thứ nhất.

- Tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý dựa trên cặp được lựa chọn thứ nhất gồm hai kênh được giải mã để thu được tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã.

Trước khi cặp thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý được tạo ra dựa trên cặp được lựa chọn thứ nhất đã nêu của hai kênh được giải mã, các bước sau đây được thực hiện:

- Nhận biết ít nhất một trong số hai kênh của cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2), một hoặc nhiều băng tần mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và tạo kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải tất cả trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiễu âm được tạo ra sử dụng các vạch phổ của kênh trộn, trong đó việc lựa chọn hai hoặc nhiều hơn hai kênh đầu ra âm thanh trước mà được sử dụng để tạo kênh trộn từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước được thực hiện phụ thuộc vào thông tin phụ.

Hơn nữa, thiết bị mã hóa tín hiệu đa kênh có ít nhất ba kênh được cung cấp. Phương pháp bao gồm:

- Tính toán, trong bước lặp thứ nhất, các giá trị tương quan liên kênh giữa mỗi cặp của ít nhất ba kênh, để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị cao hơn ngưỡng, và để xử lý cặp được lựa chọn sử dụng phép toán xử lý đa kênh để suy ra các tham số đa kênh ban đầu cho các cặp được lựa chọn và để suy ra các kênh được xử lý thứ nhất.

- Thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai sử dụng ít nhất một kênh trong số các kênh được xử lý để suy ra thêm các tham số đa kênh và các kênh được xử lý thứ hai.

- Mã hóa các kênh từ xử lý phép lặp được thực hiện bởi bộ xử lý phép lặp để thu các kênh được mã hóa. Và:

- Tạo tín hiệu đa kênh được mã hóa có các kênh được mã hóa, các tham số đa

kênh ban đầu và các tham số đa kênh khác và có thông tin biểu thị liệu thiết bị giải mã có điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà được giải mã trước bởi thiết bị giải mã.

Hơn nữa, chương trình máy tính được đề xuất, trong đó mỗi chương trình máy tính được tạo cấu hình để thực hiện một trong các phương pháp đã nêu khi chạy trên máy tính hoặc bộ xử lý tín hiệu, sao cho mỗi phương pháp đã nêu được thực hiện bởi một chương trình máy tính.

Ngoài ra, các tín hiệu đa kênh được mã hóa được đề xuất. Tín hiệu đa kênh được mã hóa bao gồm các kênh được mã hóa và các tham số đa kênh và thông tin biểu thị liệu thiết bị giải mã có điền đầy các vạch phổ của một hoặc nhiều băng tần hay không, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với dữ liệu quang phổ được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà được giải mã trước bởi thiết bị giải mã.

Mô tả vắn tắt các hình vẽ

Sau đây, các phương án của sáng chế được mô tả cụ thể với sự tham khảo các hình vẽ, trong đó:

Fig.1a thể hiện thiết bị giải mã theo phương án;

Fig.1b thể hiện thiết bị giải mã theo phương án khác;

Fig.2 thể hiện sơ đồ khối của bộ giải mã miền tần số theo tham số theo phương án của sáng chế;

Fig.3 thể hiện sơ đồ dưới dạng giản đồ minh họa chuỗi quang phổ tạo thành các ảnh phổ các các kênh của tín hiệu âm thanh đa kênh để hiểu dễ hơn phần mô tả của bộ giải mã trên Fig.2;

Fig.4 thể hiện sơ đồ dưới dạng giản đồ minh họa quang phổ hiện thời bên ngoài các ảnh phổ được thể hiện trên Fig.3 để hiểu dễ hơn phần mô tả của Fig.2;

Fig.5a và Fig.5b thể hiện sơ đồ khối của bộ giải mã âm thanh miền tần số tương ứng với phương án thay mà theo đó phép trộn giảm của khung trước được sử dụng như là cơ sở cho việc điền đầy nhiều âm liên kênh;

Fig.6 thể hiện sơ đồ khối của bộ mã hóa âm thanh miền tần số theo tham số tương ứng với một phương án;

Fig.7 thể hiện sơ đồ khối dạng giản lược của thiết bị mã hóa tín hiệu đa kênh có ít nhất ba kênh, theo một phương án;

Fig.8 thể hiện sơ đồ khối dạng giản lược của thiết bị mã hóa tín hiệu đa kênh có ít nhất ba kênh, theo một phương án;

Fig.9 thể hiện sơ đồ khối dạng giản lược của hộp âm lập thể theo một phương án;

Fig.10 thể hiện sơ đồ khối dạng giản lược của thiết bị giải mã tín hiệu đa kênh được mã hóa có các kênh được mã hóa và ít nhất hai tham số đa kênh theo phương án;

Fig.11 thể hiện lưu đồ của phương pháp mã hóa tín hiệu đa kênh có ít nhất ba kênh theo một phương án;

Fig.12 thể hiện lưu đồ của phương pháp giải mã tín hiệu đa kênh được mã hóa có các kênh được mã hóa và ít nhất hai tham số đa kênh theo một phương án;

Fig.13 thể hiện hệ thống theo một phương án;

Fig.14 thể hiện trong kịch bản, (a) sự tạo ra của các kênh tổ hợp cho khung thứ nhất trong kịch bản, và trong kịch bản (b), sự tạo ra các kênh tổ hợp cho khung thứ hai kế tiếp khung thứ nhất theo một phương án; và

Fig.15 thể hiện sơ đồ chỉ số cho các tham số đa kênh theo các phương án.

Mô tả chi tiết sáng chế

Các phần tử bằng hoặc tương đương hoặc các phần tử với chức năng bằng hoặc tương đương được biểu thị trong phần mô tả bằng các số tham chiếu bằng hoặc tương đương.

Trong phần mô tả sau đây, nhiều chi tiết được đưa ra để cung cấp giải thích kỹ lưỡng hơn về các phương án của sáng chế. Tuy nhiên, sẽ rõ ràng đối với những người có hiểu biết trong lĩnh vực kỹ thuật rằng các phương án của sáng chế có thể được thực hiện mà không có các chi tiết cụ thể. Trong các trường hợp khác, các cấu hình và các thiết bị đã biết được thể hiện trong dạng sơ đồ khối hơn là chi tiết để tránh làm khó hiểu các phương án của sáng chế. Ngoài ra, các dấu hiệu kỹ thuật của các phương án khác nhau được mô tả sau đây có thể được tổ hợp với nhau trừ khi được lưu ý cụ thể.

Trước khi mô tả thiết bị 201 giải mã của Fig.1a, trước tiên, việc điền đầy nhiều âm cho mã hóa âm thanh đa kênh được mô tả. Trong các phương án, môđun điền đầy nhiều âm 220 của Fig.1 có thể, ví dụ, được tạo cấu hình để tiến hành một hoặc nhiều

kỹ thuật dưới đây mà được mô tả liên quan đến việc điền đầy nhiễu âm cho mã hóa âm thanh đa kênh.

Fig.2 thể hiện bộ giải mã âm thanh miền tần số tương ứng với một phương án của sáng chế. Bộ giải mã thường được biểu thị sử dụng số tham chiếu 10 và bao gồm ký hiệu nhận biết thừa số tỉ lệ 12, bộ giải lượng tử hóa 14, bộ điền đầy nhiễu âm 16 và bộ biến đổi nghịch đảo 18 cũng như bộ chiết xuất vạch phổ 20 và bộ chiết xuất thừa số tỉ lệ 22. Các phần tử tùy chọn khác mà có thể bao gồm trong bộ giải mã 10 bao gồm bộ dự báo âm lập thể phức hợp 24, bộ giải mã giữa - bên (MS - mid-side) 26 và công cụ lọc TNS nghịch đảo (tạo hình nhiễu âm theo thời gian) mà trong đó hai thuyết minh bằng ví dụ cụ thể 28a và 28b được thể hiện trên Fig.2. Ngoài ra, bộ cấp trộn giảm được thể hiện và được tóm tắt cụ thể hơn dưới đây sử dụng số tham chiếu 30.

Bộ giải mã âm thanh miền tần số 10 của Fig.2 là bộ giải mã theo tham số hỗ trợ điền đầy nhiễu âm mà theo thừa số tỉ lệ được lượng tử hóa bằng không nhất định được điền đầy với nhiễu âm sử dụng thừa số tỉ lệ của băng thừa số tỉ lệ như là phương tiện để điều khiển mức nhiễu âm được điền đầy vào băng thừa số tỉ lệ đó. Ngoài ra, bộ giải mã 10 của Fig.2 thể hiện bộ giải mã âm thanh đa kênh được tạo cấu hình để khôi phục tín hiệu âm thanh đa kênh từ dòng dữ liệu luân phiên về 30. Tuy nhiên, Fig.2 tập trung vào các phần tử của bộ giải mã 10 liên quan đến khôi phục một trong các tín hiệu âm thanh đa kênh được mã hóa vào dòng dữ liệu 30 và xuất ra kênh (đầu ra) này tại đầu ra 32. Số tham chiếu 34 biểu thị rằng bộ giải mã 10 có thể bao gồm các phần tử khác hoặc có thể bao gồm một số điều khiển hoạt động đường dây để khôi phục các kênh khác của tín hiệu âm thanh đa kênh trong đó phần mô tả dưới đây biểu thị việc khôi phục bộ giải mã 10 của kênh quan tâm tại đầu ra 32 tương tác với giải mã các kênh khác.

Tín hiệu âm thanh đa kênh được biểu diễn bởi dòng dữ liệu 30 có thể bao gồm hai hoặc nhiều hơn hai kênh. Sau đây, phần mô tả các phương án của sáng chế tập trung vào trường hợp âm lập thể trong đó tín hiệu âm thanh đa kênh chỉ bao gồm hai kênh, nhưng theo nguyên tắc, các phương án sau đây có thể dễ dàng chuyển sang các phương án thay thế liên quan đến các tín hiệu âm thanh đa kênh và việc mã hóa chung bao gồm nhiều hơn hai kênh.

Như sẽ trở nên rõ ràng hơn từ phần mô tả của Fig.2 dưới đây, bộ giải mã 10 của

Fig.2 là bộ giải mã biến đổi. Tức là, theo kỹ thuật mã hóa của bộ giải mã ưu tiên 10, các kênh được mã hóa trong miền biến đổi ví dụ như sử dụng phép biến đổi chống lặp của các kênh. Hơn nữa, phụ thuộc vào bộ tạo của tín hiệu, các pha thời gian trong đó các kênh của tín hiệu âm thanh phần lớn biểu diễn nội dung âm thanh, lệch khỏi nhau chỉ một ít hoặc thay đổi nhất định giữa chúng, ví dụ các biên độ và/hoặc pha khác nhau để biểu diễn cảnh âm thanh mà trong đó các chênh lệch giữa các kênh cho phép định vị ảo nguồn âm thanh của cảnh âm thanh liên quan đến các vị trí loa phát thanh ảo kết hợp với các kênh đầu ra của tín hiệu âm thanh đa kênh. Tuy nhiên, tại một số pha theo thời gian, các kênh khác của tín hiệu âm thanh có thể tương quan nhiều hoặc ít hơn với nhau và thậm chí có thể biểu diễn, ví dụ, các nguồn âm thanh khác nhau một cách hoàn thiện.

Để xem xét khả năng môi liên hệ biến thiên theo thời gian có thể xảy ra giữa các kênh của tín hiệu âm thanh, bộ mã hóa-giải mã âm thanh nằm dưới bộ giải mã 10 của Fig.2 cho phép sử dụng biến thiên thời gian của các ước số khác nhau để sử dụng các phần dư liên kênh. Ví dụ, mã hóa MS cho phép chuyển đổi giữa việc biểu diễn các kênh bên trái và bên phải của tín hiệu âm thanh âm lập thể như chung hoặc như cặp gồm các kênh M (giữa) và S (bên) biểu diễn lần lượt phép trộn giảm của các kênh bên trái và bên phải và chênh lệch được giảm một nửa của chúng. Tức là, theo nghĩa thời gian phổ, ảnh phổ liên tục của hai kênh được truyền dẫn bởi dòng dữ liệu 30, nhưng ý nghĩa của các kênh này (được truyền dẫn) có thể lần lượt thay đổi theo thời gian và tương ứng với các kênh đầu ra.

Dự báo âm lập thể phức hợp - Công cụ sử dụng phần dư liên kênh khác- cho phép, trong miền quang phổ, dự báo các thừa số miền tần số của một kênh hoặc các vạch phổ sử dụng các vạch phổ được định vị quang phổ của kênh khác. Chi tiết liên quan đến vấn đề này được mô tả như sau.

Để tạo điều kiện cho việc hiểu phần mô tả sau của Fig.2 và các thành phần của nó được thể hiện trên đó, Fig.3 thể hiện trường hợp minh họa của tín hiệu âm thanh âm lập thể được biểu diễn bởi dòng dữ liệu 30, cách có thể mà làm thế nào các giá trị mẫu cho các vạch phổ của hai kênh có thể được mã hóa vào dòng dữ liệu 30 để được xử lý bởi bộ giải mã của Fig.2. Cụ thể, tại nửa phía trên của Fig.3, ảnh phổ 40 của kênh thứ nhất của tín hiệu âm thanh âm lập thể được thể hiện, nửa phía dưới của Fig.3 minh họa

ảnh phổ 42 của kênh khác của tín hiệu âm thanh âm lập thể. Một lần nữa, cần lưu ý rằng "ý nghĩa" của các ảnh phổ 40 và 42 có thể thay đổi theo thời gian, ví dụ, do chuyển đổi biến thiên theo thời gian giữa miền được mã hóa MS và miền không được mã hóa MS. Trong trường hợp thứ nhất, các ảnh phổ 40 và 42 lần lượt liên quan đến kênh M và S, trong đó trong trường hợp sau các ảnh phổ 40 và 42 liên quan đến các kênh bên trái và bên phải. Việc chuyển đổi giữa miền được mã hóa MS và miền không được mã hóa MS có thể được truyền tín hiệu trong dòng dữ liệu 30.

Fig.3 thể hiện các ảnh phổ 40 và 42 có thể được mã hóa vào dòng dữ liệu 30 tại độ phân giải thời gian phổ biến thiên theo thời gian. Ví dụ, cả hai kênh (được truyền dẫn), theo cách được căn chỉnh theo thời gian) có thể được chia nhỏ thành chuỗi gồm các khung biểu thị bằng việc sử dụng các ngoặc móc 44 mà có thể dài bằng nhau và tiếp giáp với nhau mà không chồng lấp. Như vừa được đề cập, độ phân giải quang phổ mà tại đó các ảnh phổ 40 và 42 được biểu diễn trong dòng dữ liệu 30 có thể thay đổi theo thời gian. Ban đầu, giả sử rằng độ phân giải quang phổ thay đổi theo thời gian như nhau đối với các ảnh phổ 40 và 42 nhưng sự mở rộng của việc đơn giản hóa này cũng khả thi khi sẽ trở nên rõ ràng từ phần mô tả sau. Việc thay đổi độ phân giải thời gian phổ, ví dụ, được truyền tín hiệu trong dòng dữ liệu 30 trong các đơn vị của các khung 44. Tức là, độ phân giải thời gian phổ thay đổi trong các đơn vị của các khung 44. Thay đổi trong độ phân giải thời gian phổ của các ảnh phổ 40 và 42 đạt được bằng việc chuyển độ dài biến đổi và số lượng phép biến đổi được sử dụng để mô tả các ảnh phổ 40 và 42 trong mỗi khung 44. Trong ví dụ của Fig.3, các khung 44a và 44b ví dụ các khung mà trong đó phép biến đổi dài được sử dụng để lấy mẫu các kênh của tín hiệu âm thanh trong đó, từ đó dẫn đến độ phân giải quang phổ cao nhất với một giá trị mẫu vạch phổ cho mỗi vạch phổ đối với mỗi khung cho mỗi kênh. Trên Fig.3, các giá trị mẫu của các vạch phổ được biểu thị sử dụng các dấu gạch nhỏ trong các hộp, trong đó, các hộp, lần lượt, được bố trí theo các hàng và các cột và sẽ biểu diễn lưới thời gian quang phổ với mỗi hàng tương ứng với một vạch phổ và mỗi cột tương ứng với các đoạn con của các khung 44 tương ứng với các phép biến đổi ngắn nhất liên quan đến việc tạo thành các ảnh phổ 40 và 42. Cụ thể, ví dụ, ngoài ra, Fig.3 minh họa, đối với khung 44d, ngoài ra khung có thể trải qua các phép biến đổi liên tiếp về độ dài ngắn hơn từ đó, đối với các khung như khung 44d, dẫn đến một số quang phổ tiếp theo

theo thời gian của độ phân giải quang phổ được giảm. Tám phép biến đổi ngắn được sử dụng làm ví dụ cho khung 44d, dẫn đến việc lấy mẫu thời gian phổ của các ảnh phổ 40 và 42 mà trong đó khung 42d, tại các vạch phổ được đặt cách nhau sao cho chỉ tám vạch phổ được bố trí, nhưng với giá trị mẫu cho mỗi tám cửa sổ biến đổi hoặc các biến đổi của độ dài ngắn hơn được sử dụng cho khung biến đổi. Nhằm mục đích minh họa, Fig.3 thể hiện số lượng khác của phép biến đổi đối với khung có thể thực hiện được, ví dụ như sử dụng hai phép biến đổi của độ dài biến đổi mà, ví dụ, bằng một độ độ dài biến đổi của các phép biến đổi dài đối với các khung 44a và 44b, từ đó dẫn đến việc lấy mẫu lưới thời gian phổ hoặc ảnh các ảnh phổ 40 và 42 trong đó các giá trị mẫu vạch phổ thu được đối với mỗi hai vạch phổ, một trong đó liên quan đến phép biến đổi dẫn trước, còn lại là phép biến đổi theo sau.

Các cửa sổ biến đổi đối với các phép biến đổi mà các khung được chia nhỏ được minh họa trên Fig.3 dưới đây, mỗi ảnh phổ sử dụng các đường giống cửa sổ chồng lấp. Ví dụ, chồng lấp theo thời gian phục vụ mục đích triệt tiêu răng cưa trong miền thời gian (TDAC - Time-Domain Aliasing Cancellation).

Mặc dù các phương án được mô tả thêm dưới đây cũng có thể được thực hiện theo kiểu khác, Fig.3 minh họa trường hợp trong đó việc chuyển giữa các độ phân giải thời gian phổ khác nhau cho các khung riêng lẻ 44 được thực hiện theo cách sao cho đối với mỗi khung 44, số lượng tương tự của các giá trị vạch phổ được biểu thị bởi các gạch chéo nhỏ trên kết quả Fig.3 đối với ảnh phổ 40 và ảnh phổ 42, sự chênh lệch chỉ nằm trong cách các vạch phổ lấy mẫu thời gian phổ tám thời gian phổ tương ứng tương ứng với khung tương ứng 44, được mở rộng theo thời gian trên thời gian của khung tương ứng 44 và mở rộng theo thời gian từ tần số không đến tần số cực đại $f_{\max}$.

Sử dụng các mũi tên trên Fig.3, Fig.3 minh họa khung 44d mà quang phổ tương tự có thể thu được cho tất cả các khung 44 bằng việc phân bố một cách thích hợp các giá trị mẫu vạch phổ thuộc về vạch phổ tương tự nhưng các cửa sổ biến đổi ngắn nằm trong một khung của một kênh, lên trên các vạch phổ trống (rỗng) mà trong đó khung lên đến vạch phổ bị chiếm tiếp theo của khung tương tự đó. Các quang phổ này sau đây được gọi là "quang phổ được xen kẽ". Trong n biến đổi xen kẽ của một khung của một kênh, ví dụ, các giá trị quang phổ được đồng định vị theo quang phổ của n biến đổi ngắn theo sau nhau trước khi tập hợp của n giá trị vạch phổ được đồng định vị treo

quang phổ của n biến đổi ngắn của vạch phổ kế tiếp theo quang phổ theo sau. Dạng trung gian xen kẽ có thể được thực hiện như sau: thay vì xen kẽ tất cả các hệ số quang phổ của một khung, có thể thực hiện xen kẽ các hệ số vạch phổ của tập con thích hợp của các biến đổi ngắn của khung 44d. Trong bất kỳ trường hợp nào, bất kỳ khi nào các khung của hai kênh tương ứng với các ảnh phổ 40 và 42 được thảo luận, các quang phổ này có thể liên quan đến các quang phổ xen kẽ hoặc quang phổ không xen kẽ.

Để mã hóa một cách hiệu quả các hệ số quang phổ biểu diễn các ảnh phổ 40 và 42 qua dòng dữ liệu 30 đi qua bộ giải mã 10, tương tự được lượng tử hóa. Để điều khiển nhiều âm lượng tử hóa theo thời gian phổ, kích thước bước lượng tử hóa được điều khiển qua các thừa số tỉ lệ mà được thiết lập trong lưới thời gian phổ nhất định. Cụ thể, trong mỗi chuỗi của quang phổ của mỗi ảnh phổ, các vạch phổ được tạo nhóm thành các nhóm thừa số tỉ lệ không chồng lấp liên tiếp theo quang phổ. Fig.4 thể hiện quang phổ 46 của ảnh phổ 40 tại nửa phía trên của nó, và quang phổ đồng thời 48 ra khỏi ảnh phổ 42. Như được thể hiện trong đó, các quang phổ 46 và 48 được chia nhỏ thành các băng thừa số tỉ lệ dọc theo trục quang phổ f để tạo nhóm các vạch phổ thành các nhóm không chồng lấp. Các băng thừa số tỉ lệ được minh họa trên Fig.4 sử dụng các dấu ngoặc móc 50. Để đơn giản, giả sử rằng các đường biên giữa các băng thừa số tỉ lệ trùng giữa quang phổ 46 và 48, nhưng không nhất thiết là trường hợp này.

Tức là, bằng cách mã hóa trong dòng dữ liệu 30, mỗi ảnh phổ 40 và 42 được chia nhỏ thành các chuỗi theo thời gian của quang phổ và mỗi quang phổ được chia nhỏ theo quang phổ thành các băng thừa số tỉ lệ, và đối với mỗi băng thừa số tỉ lệ, dòng dữ liệu 30 mã hóa hoặc truyền thông tin về thừa số tỉ lệ tương ứng với băng thừa số tỉ lệ tương ứng. Các hệ số vạch phổ nằm trong băng thừa số tỉ lệ tương ứng 50 được lượng tử hóa sử dụng thừa số tỉ lệ tương ứng hoặc, theo như bộ giải mã 10 được quan tâm, có thể giải lượng tử hóa sử dụng thừa số tỉ lệ của băng thừa số tỉ lệ tương ứng.

Trước khi việc thay đổi trở lại Fig.2 và mô tả của nó, giả sử rằng sau đây kênh được xử lý cụ thể, tức là giải mã các phần tử của bộ giải mã của Fig.2 trừ phần tử 34 có liên quan, là kênh được truyền dẫn của ảnh phổ 40 mà, như được đề cập ở trên, có thể biểu diễn một kênh trong số các kênh trái và phải, và kênh M hoặc kênh S với giả sử rằng tín hiệu âm thanh đa kênh được mã hóa thành dòng dữ liệu 30 là tín hiệu âm thanh âm lập thể.

Trong khi bộ chiết suất vạch phổ 20 được tạo cấu hình để chiết suất dữ liệu vạch phổ, tức là các hệ số vạch phổ cho các khung 44 từ dòng dữ liệu 30, bộ chiết suất thừa số tỉ lệ 22 được tạo cấu hình để chiết suất các thừa số tỉ lệ tương ứng cho mỗi khung 44. Để đạt được mục đích này, bộ chiết suất 20 và 22 có thể sử dụng giải mã entropy. Tương ứng với một phương án, bộ chiết suất thừa số tỉ lệ 22 được chiết xuất để chiết suất theo theo thứ tự các thừa số tỉ lệ của, ví dụ quang phổ 46 trên Fig.4, tức là các thừa số tỉ lệ của các băng thừa số tỉ lệ 50, từ dòng dữ liệu 30, sử dụng giải mã entropy thích ứng ngữ cảnh. Thứ tự của giải mã theo thứ tự có thể theo thứ tự quang phổ được định rõ giữa thừa số tỉ lệ dẫn trước, ví dụ, từ tần số thấp đến tần số cao. Bộ chiết suất thừa số tỉ lệ 22 có thể sử dụng giải mã entropy thích ứng ngữ cảnh và có thể xác định ngữ cảnh đối mỗi thừa số tỉ lệ phụ thuộc và các thừa số tỉ lệ đã được chiết suất trong vùng lân cận quang phổ của các thừa số tỉ lệ được chiết xuất hiện thời, ví dụ phụ thuộc vào thừa số tỉ lệ của băng thừa số tỉ lệ ngay trước đó. Cách khác, bộ chiết suất thừa số tỉ lệ 22 có thể giải mã theo cách dự báo các thừa số tỉ lệ từ dòng dữ liệu 30, ví dụ như sử dụng giải mã chênh lệch trong khi dự báo thừa số tỉ lệ được giải mã hiện thời dựa trên bất kỳ các thừa số tỉ lệ được giải mã trước đó ví dụ như thừa số tỉ lệ ngay trước đó. Đáng chú ý là, quy trình chiết suất thừa số tỉ lệ là bất khả tri đối với thừa số tỉ lệ thuộc về băng thừa số tỉ lệ được bố trí bởi các vạch phổ được lượng tử hóa bằng không dành riêng, hoặc được bố trí bởi các vạch phổ mà giữa chúng ít nhất một vạch phổ được lượng tử hóa tới giá trị bằng không. Các thừa số tỉ lệ thuộc về băng thừa số tỉ lệ chỉ bởi các vạch phổ được lượng tử hóa bằng không có thể đóng vai trò như là cơ sở dự báo cho thừa số tỉ lệ được giải mã sau đó mà có thể thuộc về băng thừa số tỉ lệ được bố trí bởi các vạch phổ mà trong đó có vạch phổ khác không, và được dự đoán dựa trên thừa số tỉ lệ được giải mã trước đó mà có thể thuộc về băng thừa số tỉ lệ được phân bố bởi các vạch phổ mà trong đó có một vạch phổ khác không.

Để hoàn chỉnh, cần lưu ý rằng bộ chiết suất vạch phổ 20 chiết suất các thừa số vạch phổ mà các băng thừa số tỉ lệ 50 được bố trí tương tự như vậy sử dụng, ví dụ, giải mã entropy và/hoặc mã hóa dự báo. Giải mã entropy có thể sử dụng sự thích ứng ngữ cảnh dựa trên các hệ số vạch phổ trong vùng lân cận thời gian phổ của hệ số vạch phổ được giải mã hiện thời, và tương tự như vậy, việc dự báo có thể là dự báo quang phổ, dự báo thời gian, hoặc dự báo thời gian phổ dự báo thừa số vạch phổ được giải mã

hiện thời dựa trên các thừa số vạch phổ được giải mã trước đó trong vùng lân cận thời gian phổ của nó. Để hiệu quả mã hóa được tăng, bộ chiết xuất vạch phổ 20 có thể được tạo cấu hình để thực hiện giải mã vạch phổ theo thời gian hoặc các hệ số vạch theo bộ dữ liệu, mà thu thập hoặc tạo nhóm các vạch phổ theo trục tần số.

Do đó, tại đầu ra của bộ chiết suất vạch phổ 20, các thừa số vạch phổ được cung cấp, ví dụ trong các đơn vị của quang phổ ví dụ như quang phổ 46 thu thập, ví dụ tất cả các thừa số vạch phổ của khung tương ứng, hoặc cách khác thu thập tất cả các thừa số vạch phổ của các phép biến đổi ngắn nhất định của khung tương ứng. Tại đầu ra của bộ chiết xuất 22, theo thứ tự, các thừa số tỉ lệ tương ứng của quang phổ tương ứng được xuất ra.

Các ký hiệu nhận biết băng thừa số tỉ lệ 12 cũng như bộ giải lượng tử hóa 14 có các đầu vào vạch phổ được nối với đầu ra của bộ chiết xuất vạch phổ 20, và bộ giải lượng tử hóa 14 và bộ điền đầy nhiễu âm 15 có các đầu ra thừa số tỉ lệ được nối với đầu ra của bộ chiết suất thừa số tỉ lệ 22. Ký hiệu nhận biết băng thừa số tỉ lệ 12 được tạo cấu hình để nhận biết cái được gọi là các băng thừa số tỉ lệ được lượng tử hóa bằng không trong quang phổ hiện thời 46, tức là các băng thừa số tỉ lệ mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, ví dụ băng thừa số tỉ lệ 50c trên Fig.4 và các băng thừa số còn lại của quang phổ này mà trong đó ít nhất một vạch phổ được lượng tử hóa khác không. Cụ thể, trên Fig.4, các hệ số vạch phổ được biểu thị sử dụng các vùng được gạch chéo trên Fig.4. Từ đó có thể thấy rằng trên quang phổ 46, tất cả các băng thừa số trừ băng thừa số tỉ lệ 50b có ít nhất một vạch phổ mà trong số hệ số vạch phổ được lượng tử hóa bằng giá trị khác không. Sau đây sẽ trở nên rõ ràng rằng các băng thừa số tỉ lệ được lượng tử hóa bằng không ví dụ như 50 tạo thành đối tượng của việc điền đầy nhiễu âm liên kênh được mô tả sau đây. Trước khi tiếp tục với phần mô tả, cần lưu ý rằng ký hiệu nhận biết băng thừa số tỉ lệ 12 có thể giới hạn nhận biết của nó chỉ trên tập con thích hợp của các băng thừa số tỉ lệ 50 ví dụ như trên các băng thừa số tỉ lệ ở trên tần số bắt đầu nhất định 52. Trên Fig.4, điều này hạn chế quy trình nhận biết trên các băng thừa số tỉ lệ 50d, 50e và 50f.

Kí hiệu băng thừa số tỉ lệ 12 thông báo bộ điền đầy nhiễu âm 16 trên các băng thừa số tỉ lệ mà là các băng thừa số tỉ lệ được lượng tử hóa bằng không. Bộ giải lượng tử hóa 14 sử dụng các thừa số tỉ lệ được kết hợp với quang phổ luân phiên lại 46 để

giải lượng tử hóa, hoặc tỉ lệ, các hệ số vạch phổ của các vạch phổ của quang phổ 46 theo các thừa số tỉ lệ được kết hợp, tức là các thừa số tỉ lệ được kết hợp với các băng thừa định tỉ lệ 50. Cụ thể, bộ giải lượng tử hóa 14 giải lượng tử hóa và định tỉ lệ các hệ số quang phổ nằm trong băng thừa số tỉ lệ tương ứng với thừa số tỉ lệ được kết hợp với băng thừa số tỉ lệ tương ứng. Fig.4 có thể được diễn giải thể hiện kết quả của việc giải lượng tử hóa các vạch quang phổ.

Bộ điền đầy nhiễu âm 16 thu thông tin trên các băng thừa số tỉ lệ được lượng tử hóa bằng không mà tạo ra đối tượng của việc điền đầy nhiễu âm sau đó, quang phổ được giải lượng tử hóa cũng như các thừa số tỉ lệ của ít nhất các băng thừa số tỉ lệ này được nhận biết như các băng thừa số tỉ lệ được lượng tử hóa bằng không và việc truyền tín hiệu được thu từ dòng dữ liệu 30 đối với khung hiện thời cho biết việc điền đầy nhiễu âm liên kênh có được thực hiện cho khung hiện thời hay không.

Quy trình điền đầy nhiễu âm lên kênh được mô tả trong ví dụ sau đây thực sự liên quan đến hai loại điền đầy nhiễu âm, cụ thể là việc chèn sán nhiễu 54 liên quan đến tất cả các vạch phổ được lượng tử hóa bằng không không tương ứng với thành phần tiềm năng của chúng vào băng thừa số tỉ lệ được lượng tử hóa bằng không và quy trình điền đầy nhiễu âm liên kênh thực. Mặc dù việc tổ hợp này được mô tả sau đây, nhấn mạnh rằng việc chèn sán nhiễu có thể được bỏ qua theo một phương án thay thế khác. Hơn nữa, việc truyền tín hiệu đề cập đến việc bật và tắt điền đầy nhiễu âm liên quan đến khung hiện thời và được thu từ dòng dữ liệu 30 có thể chỉ liên quan đến việc điền đầy nhiễu âm liên kênh, hoặc có thể điều khiển tổ hợp của cả hai loại điền đầy nhiễu âm với nhau.

Theo việc chèn sán nhiễu, bộ điền đầy nhiễu âm 16 có thể vận hành như sau. Cụ thể, bộ điền đầy nhiễu âm 16 có thể sử dụng việc tạo nhiễu âm nhân tạo ví dụ như bộ tạo số ngẫu nhiên giả hoặc các nguồn ngẫu nhiên để điền đầy các vạch phổ, các hệ số vạch phổ của nó bằng không. Do đó, mức sán nhiễu 54 được chèn vào các vạch phổ được lượng tử hóa bằng không có thể được thiết lập theo việc truyền tín hiệu rõ ràng trong dòng dữ liệu 30 đối với khung hiện thời hoặc quang phổ hiện thời 46. "Mức" của sán nhiễu 54 có thể được xác định, ví dụ sử dụng căn bậc hai của trung bình bình phương (root-mean-square - RMS) hoặc ước số năng lượng.

Do đó, việc chèn sán nhiễu biểu diễn loại điền đầy trước cho các băng thừa số tỉ

lệ được nhận biết như băng thừa số tỉ lệ được lượng tử hóa bằng không cũng như băng thừa số tỉ lệ 50d trên Fig.4. Nó cũng ảnh hưởng đến các băng thừa số tỉ lệ vượt quá các băng được lượng tử hóa bằng không, như sau đó cái sau không trải qua điền đầy nhiều âm liên kênh. Như được mô tả sau đây, quy trình điền đầy nhiều âm liên kênh là để điền đầy các băng thừa số tỉ lệ được lượng tử hóa bằng không lên đến mức mà được điều khiển qua thừa số tỉ lệ của băng thừa số tỉ lệ được lượng tử hóa bằng không tương ứng. Cái sau có thể được sử dụng trực tiếp để đạt được mục đích này do tất cả các vạch phổ của băng thừa số tỉ lệ được lượng tử hóa bằng không tương ứng được lượng tử hóa bằng không. Tuy nhiên, dòng dữ liệu 30 có thể gồm việc truyền dữ liệu bổ sung của tham số, đối với mỗi khung hoặc mỗi quang phổ 46, mà thường được áp dụng cho các thừa số tỉ lệ của tất cả các băng thừa số tỉ lệ được lượng tử hóa bằng không của khung hoặc quang phổ 46 và các kết quả tương ứng, khi được áp dụng lên các thừa số tỉ lệ của các băng thừa số tỉ lệ được lượng tử hóa bằng không bởi bộ điền đầy nhiều âm 16, mức điền đầy tương ứng mà riêng biệt đối với các băng thừa số tỉ lệ được lượng tử hóa bằng không. Tức là, bộ điền đầy nhiều âm 16 có thể cải biên, sử dụng hàm cải biên, đối với mỗi băng thừa số tỉ lệ được lượng tử hóa bằng không của quang phổ 46, thừa số tỉ lệ của băng thừa số tỉ lệ tương ứng sử dụng tham số vừa được đề cập đến được chứa trong dòng dữ liệu 30 cho quang phổ 46 này của khung hiện thời để thu được mức điền đầy đích cho việc đo băng thừa số tỉ lệ được lượng tử hóa bằng không tương ứng, về mặt năng lượng hoặc RMS, ví dụ mức mà xử lý điền đầy nhiều âm liên kênh có thể điền đầy băng thừa số tỉ lệ được lượng tử hóa bằng không tương ứng với nhiều âm bổ sung (tùy chọn) (ngoài sản nhiễu 54).

Cụ thể, để thực hiện điền đầy nhiều âm liên kênh 56, bộ điền đầy nhiều âm 16 thu phần được đồng định vị của quang phổ của kênh khác 48, trong trạng thái đã được giải mã đầy đủ hoặc phần lớn, và sao chép phần thu được của quang phổ 48 vào băng thừa số tỉ lệ được lượng tử hóa bằng không đến chỗ mà phần này được đồng định vị theo quang phổ, được định tỉ lệ theo cách mà mức nhiễu âm toàn phần kết quả mà trong đó băng thừa số tỉ lệ được lượng tử hóa bằng không - được suy ra bằng việc lấy tích phân theo các vạch phổ của băng thừa số tỉ lệ tương ứng - bằng với mức điền đầy đích được thu thừa số tỉ lệ của các băng thừa số tỉ lệ được lượng tử hóa. Bằng cách này, âm điệu của nhiễu âm được điền đầy vào băng thừa số tỉ lệ được lượng tử hóa

bằng không tương ứng được cải hiện so với nhiều âm được tạo ra một cách nhân tạo ví dụ như nhiều âm tạo ra cơ sở sản nhiều 54, và cũng tốt hơn việc sao chép/ lặp quang phổ không được điều khiển từ các vạch tần số rất thấp nằm trong quang phổ 46 tương đương.

Để chính xác hơn, bộ điền đầy nhiều âm 16 định vị, đối với băng hiện thời, phần được đồng định vị theo quang phổ trong quang phổ 48 của kênh khác, định tỉ lệ các vạch phổ của nó phụ thuộc và thừa số tỉ lệ của băng thừa số tỉ lệ được lượng tử hóa 50d theo cách vừa được mô tả liên quan, tùy chọn, tham số thừa số nhiều âm hoặc bù bổ sung được chứa trong dòng dữ liệu 30 đối với khung hoặc quang phổ hiện thời 6, để kết quả của nó điền đầy băng thừa số tỉ lệ được lượng tử hóa bằng không tương ứng 50d lên đến mức mong muốn như được định bởi thừa số tỉ lệ của băng tham số tham số định tỉ lệ bằng không 50d. Trong phương án của sáng chế, điều này nghĩa là việc điền đầy được thực hiện theo cách bổ sung liên quan đến sản nhiều 54.

Theo phương án đơn giản hóa, quang phổ được điền đầy nhiều âm thu được 46 sẽ được đưa trực tiếp vào đầu vào của bộ biến đổi nghịch đảo 18 để thu được, đối với mỗi cửa sổ biến đổi mà các hệ số vạch phổ của quang phổ 46 thuộc về, phần miền thời gian của tín hiệu thời gian âm thanh kênh tương ứng mà ở đó (không thể hiện trên Fig.2) quy trình chồng lấp-cộng có thể kết hợp với các phần miền thời gian này. Tức là, nếu quang phổ 46 là quang phổ không xen kẽ, các hệ số vạch phổ mà chỉ thuộc về một phép biến đổi, thì bộ biến đổi nghịch đảo 18 trải qua phép biến đổi này để thu được một phần miền thời gian và các đầu phía trước và theo sau mà trải qua quy trình chồng lấp cộng với các phần miền thời gian phía trước và theo sau được thu bởi các phép biến đổi nghịch đảo trước và phép biến đổi nghịch đảo kế tiếp để thực hiện, ví dụ, việc hủy răng cưa miền thời gian. Tuy nhiên, nếu quang phổ 46 có các hệ số vạch phổ xen kẽ trong đó của nhiều hơn một phép biến đổi liên tiếp, thì bộ biến đổi nghịch đảo 18 sẽ trải qua các phép biến đổi tương tự các phép biến đổi nghịch đảo riêng lẻ để thu được một phần miền thời gian cho mỗi biến đổi nghịch đảo, và tương ứng với thứ tự thời gian được định rõ giữa chúng, các phần miền thời gian này sẽ trải qua quy trình chồng lấp cộng giữa chúng, cũng như đối với các phần miền thời gian phía trước và kế tiếp của các quang phổ hoặc khung khác.

Tuy nhiên, để hoàn thành, cần lưu ý rằng việc xử lý khác có thể được thực hiện

trên quang phổ được điền đầy nhiễu âm. Như được thể hiện trên Fig.3, bộ lọc TNS nghịch đảo có thể thực hiện lọc TNS trên quang phổ điền đầy nhiễu âm. Tức là, được điều khiển qua các hệ số bộ lọc TNS cho khung hoặc quang phổ hiện thời 46, quang phổ thu được đến lúc này trải qua lọc tuyến tính dọc theo hướng quang phổ.

Có hoặc không có TNS, bộ dự báo âm lập thể phức hợp 24 có thể sau đó xử lý quang phổ như phần dư dự báo của phần dư liên kênh. Cụ thể là, bộ dự báo liên kênh 24 có thể sử dụng phần đồng định vị theo quang phổ của kênh khác để dự báo quang phổ 46 hoặc ít nhất tập con của các băng thừa số tỉ lệ 50 của nó. Quy trình dự báo phức hợp được minh họa trên Fig.4 với hộp gạch nối liên quan đến băng thừa số tỉ lệ 50b. Tức là, dòng dữ liệu 30 có thể gồm điều khiển các tham số dự báo liên kênh, ví dụ, mà của các băng thừa số tỉ lệ sẽ được dự báo liên kênh mà không được dự báo theo cách này. Ngoài ra, các tham số dự báo liên kênh trong dòng dữ liệu 30 có thể còn bao gồm các thừa số dự báo liên kênh phức hợp được áp dụng bởi bộ dự báo liên kênh 24 để thu kết quả dự báo liên kênh. Các thừa số này có thể được chứa trong dòng dữ liệu 30 một cách riêng lẻ cho mỗi băng thừa số tỉ lệ, hoặc cách khác cho mỗi nhóm gồm một hoặc nhiều băng thừa số tỉ lệ, mà dự báo liên kênh được kích hoạt hoặc được truyền tín hiệu để kích hoạt trong dòng dữ liệu 30.

Như được biểu thị trên Fig.4, nguồn dự báo liên kênh có thể là quang phổ 48 của kênh khác. Để chính xác hơn, nguồn dự báo liên kênh có thể là phần được đồng định vị theo quang phổ của quang phổ 48, được đồng định vị cho băng thừa số tỉ lệ 50b để được dự báo liên kênh, được mở rộng bởi việc ước lượng các phản trường tượng của nó. Việc ước lượng phản trường tượng có thể được thực hiện dựa trên phần được đồng định vị theo quang phổ 60 của bản thân quang phổ 48, và/hoặc có thể sử dụng phép trộn giảm của các kênh đã được giải mã của khung trước, tức là khung ngay trước khung được giải mã hiện thời mà quang phổ 46 thuộc về. Trên thực tế, bộ dự báo liên kênh 24 bổ sung vào các băng thừa số tỉ lệ để được dự báo liên kênh ví dụ như băng thừa số tỉ lệ 50b trên Fig.4, tín hiệu dự báo được thu như vừa được mô tả.

Như được lưu ý trong phần mô tả trước, kênh mà quang phổ 46 thuộc về có thể là kênh được mã hóa MS, hoặc có thể là kênh được liên quan loa phát thanh, ví dụ như kênh bên trái hoặc bên phải của tín hiệu âm thanh âm lập thể. Theo đó, bộ giải mã MS tùy chọn 26 đưa quang phổ được dự báo liên kênh một cách tùy chọn 46 đến giải mã

MS. theo cách thực hiện tương tự, mỗi vạch phổ hoặc quang phổ 46, phép cộng hoặc phép trừ với các vạch phổ tương ứng theo quang phổ của kênh khác tương ứng với quang phổ 48. Ví dụ, mặc dù không được thể hiện trên Fig.2, quang phổ 48 như được thể hiện trên Fig.4 được thu bằng phần 34 của bộ giải mã 10 theo cách tương tự phần mô tả phía trên liên quan đến kênh mà quang phổ 46 thuộc về, và môđun giải mã MS 26, trong thực hiện giải mã MS, đưa các quang phổ 46 và 48 qua phép cộng theo vạch phổ hoặc phép trừ theo vạch phổ, với cả hai quang phổ 46 và 48 ở cùng một giai đoạn trong dây chuyền xử lý, tức là, cả hai đều vừa được thu được bởi việc dự báo liên kênh, ví dụ, cả hai đều được thu bằng việc điền đầy nhiễu âm hoặc lọc TNS nghịch đảo.

Một cách tùy chọn, lưu ý rằng, giải mã MS có thể được thực hiện theo cách chung liên quan đến toàn bộ quang phổ 46, hoặc có thể được kích hoạt một cách riêng lẻ bởi dòng dữ liệu 30 trong các đơn vị, ví dụ, của các băng thừa số tỉ lệ 50. Nói cách khác, giải mã MS có thể được bật hoặc tắt sử dụng việc truyền tín hiệu tương ứng trong dòng dữ liệu 30 trong các đơn vị, ví dụ, của các khung hoặc độ phân giải thời gian phổ mịn ví dụ như, một cách riêng lẻ cho các băng thừa số tỉ lệ của quang phổ 46 và/hoặc 48 của ảnh phổ 40 và/hoặc 42, trong đó giả sử rằng các đường biên giống nhau của các băng thừa số tỉ lệ của cả hai kênh được định rõ.

Như được minh họa trên Fig.2, việc lọc TNS nghịch đảo bởi bộ lọc TNS nghịch đảo 28 có thể cũng được thực hiện sau bất kỳ xử lý liên kênh vào ví dụ như dự báo liên kênh 58 hoặc giải mã MS bởi bộ giải mã MS 26. Hiệu suất phía trước hoặc xuôi dòng của xử lý liên kênh có thể được cố định hoặc có thể được điều khiển thông qua việc truyền tín hiệu tương ứng đối với mỗi khung trong dòng dữ liệu 30 hoặc ở mức chi tiết khác. Bất cứ nơi nào việc lọc TNS nghịch đảo được thực hiện các hệ số bộ lọc TNS tương ứng có mặt trong dòng dữ liệu cho quang phổ hiện thời 46 điều khiển bộ lọc TNS, tức là bộ lọc dự báo tuyến tính chạy dọc theo hướng quang phổ để lọc một cách tuyến tính quang phổ luân phiên lại vào môđun bộ lọc TNS nghịch đảo tương ứng 28a và/hoặc 28b.

Do đó, quang phổ 46 đến đầu vào của bộ biến đổi nghịch đảo 18 có thể trải qua quy trình khác như vừa được mô tả. Một lần nữa, phần mô tả ở trên không có nghĩa được hiểu theo cách là tất cả các công cụ tùy chọn phải có mặt một cách đồng thời

hoặc không. Các công cụ này có thể có mặt trong bộ giải mã 10 một cách cục bộ hoặc một cách tập thể.

Trong bất kỳ trường hợp nào, quang phổ thu được tại đầu vào của bộ biến đổi nghịch đảo thể hiện sự khôi phục cuối cùng của tín hiệu đầu ra của kênh và tạo ra cơ sở của phép trộn giảm đã đề cập trước đó cho khung hiện thời mà đóng vai trò, như được mô tả liên quan đến bộ dự báo phức 58, như cơ sở cho việc ước lượng phần tương tự tiềm năng cho khung tiếp theo sẽ được giải mã. Nó có thể còn đóng vai trò như việc khôi phục cuối cùng cho việc dự báo liên kênh khác kênh mà các phần tử ngoại trừ phần tử 34 trên Fig.2 liên quan đến.

Phép trộn giảm được thực hiện bởi bộ cấp trộn giảm 31 bằng việc tổ hợp quang phổ cuối cùng 46 này với phiên bản cuối cùng tương ứng của quang phổ 48. Các đối tượng khác, tức là phiên bản cuối cùng tương ứng của quang phổ 48, tạo ra cơ sở cho việc dự báo liên kênh phức hợp trong bộ dự báo 24.

Fig.5 thể hiện cách khác so với Fig.2 đến mức cơ sở cho điền đầy nhiều âm liên kênh được biểu diễn bởi phép trộn giảm các vạch phổ được đồng định vị theo quang phổ của khung trước đó, sao cho, trong trường hợp tùy chọn của việc sử dụng dự báo liên kênh phức, nguồn của việc dự báo liên kênh phức được sử dụng hai lần, như là nguồn cho điền đầy nhiều âm liên kênh cũng như nguồn cho việc ước lượng phần tương tự trong dự báo liên kênh phức hợp. Fig.5 thể hiện bộ giải mã 10 gồm có phần 70 liên quan đến việc giải mã kênh thứ nhất mà quang phổ 46 thuộc về, cũng như cấu hình bên trong của phần khác 34 được đề cập trước đó mà liên quan đến việc giải mã kênh khác bao gồm quang phổ 48. Các ký hiệu tham chiếu giống nhau một mặt được sử dụng cho các phần tử bên trong phần 70 và mặt khác là phần 34. Như có thể thấy, việc khôi phục là tương tự. Tại đầu ra 32, một kênh của tín hiệu âm thanh âm lập thể được xuất ra, và tại đầu ra của bộ biến đổi nghịch đảo 18 của phần bộ giải mã thứ hai 34, kênh (đầu ra) khác của các kết quả tín hiệu âm thanh âm lập thể, với đầu ra này được biểu thị bằng ký hiệu tham chiếu 74. Một lần nữa, các phương án dực mô tả ở trên có thể được chuyển sang trường hợp sử dụng nhiều hơn hai kênh.

Bộ cấp trộn giảm 31 được đồng sử dụng bởi cả hai phần 70 và 34 và nhận các quang phổ được đồng định vị theo thời gian 48 và 46 của các ảnh phổ 40 và 42 để tạo ra phép trộn giảm dựa trên đó bằng việc cộng các quang phổ này trên vạch phổ bằng

cơ sở vạch phổ, có khả năng tạo ra trung bình từ đó bằng việc chia tổng này tại mỗi vạch phổ với số lượng các kênh được trộn giảm, tức là, chia hai trong trường hợp của Fig.5. Tại đầu ra của bộ cấp trộn giảm 31, phép trộn giảm của các kết quả khung trước bằng cách này. Lưu ý rằng trong trường hợp khung trước chứa nhiều hơn một quang phổ trong số một ảnh phổ của các ảnh phổ 40 và 42, các khả năng khác nhau tồn tại như cách bộ cấp trộn giảm 31 vận hành trong trường hợp này. Ví dụ, trong trường hợp bộ cấp trộn giảm 31 có thể sử dụng quang phổ của các phép biến đổi theo sau của khung hiện thời, hoặc có thể sử dụng kết quả xen kẽ của việc xen kẽ tất cả các hệ số vạch phổ của khung hiện thời của ảnh phổ 40 và 42. Phần tử độ trễ 74 trên Fig.5 như được kết nối với đầu ra của bộ cấp trộn giảm 31, do đó thể hiện rằng phép trộn giảm được cung cấp tại đầu ra của bộ cấp trộn giảm 31 tạo ra phép trộn giảm của khung trước đó 76 (xem Fig.4 lần lượt liên quan đến việc điền đầy nhiều âm liên kênh 56 và dự báo phức hợp 58). Do đó, đầu ra của phần tử độ trễ 74 được một mặt được nối với các đầu ra của bộ dự báo liên kênh 24 của các phân bộ giải mã 34 và 70 và mặt khác với các đầu ra của bộ điền đầy nhiều âm 16 của các phân bộ giải mã 70 và 34.

Tức là, trong Fig.2, bộ điền đầy nhiều âm 16 nhận quang phổ được đồng định vị theo quang phổ được khôi phục cuối cùng của kênh khác của khung hiện thời như là cơ sở của việc điền đầy nhiều âm liên kênh, trên Fig.5, việc điền đầy nhiều âm liên kênh được thực hiện thay vì dựa trên phép trộn giảm khung trước như được cung cấp bởi bộ cấp trộn giảm 31. Cách mà việc điền đầy nhiều âm liên kênh được thực hiện vẫn giữ nguyên. Tức là, bộ điền đầy nhiều âm liên kênh 16 lấy ra phần được đồng định vị theo quang phổ của quang phổ tương ứng của quang phổ của kênh khác của khung hiện thời, trong trường hợp Fig.2, và được giải mã phần lớn hoặc đầy đủ, quang phổ cuối cùng như được thu được từ khung trước biểu diễn phép trộn giảm của khung trước, trong trường hợp Fig.5, và bổ sung phần "nguồn" tương tự vào các vạch phổ trong đó bằng thừa số tỉ lệ được điền đầy nhiều âm, ví dụ 50d trên Fig.4, được định tỉ lệ theo mức nhiều âm đích được xác định bởi thừa số tỉ lệ của băng thừa số tỉ lệ tương ứng.

Kết thúc thảo luận ở trên về các phương án mô tả điền đầy nhiều âm liên kênh trong bộ giải mã âm thanh, cần rõ ràng với người đọc có hiểu biết trong lĩnh vực kỹ thuật rằng, trước khi bổ sung phần được đồng định vị theo quang phổ hoặc theo thời

gian được lấy ra của quang phổ "nguồn" vào các vạch phổ của băng thừa số tỉ lệ "đích" việc xử lý trước nhất định có thể được áp dụng cho các vạch "nguồn" mà không lạc khỏi khái niệm chung về việc điền đầy liên kênh. Cụ thể, có thể có lợi khi áp dụng vận hành lọc như, ví dụ, làm phẳng quang phổ hoặc loại bỏ độ nghiêng, đối với các vạch phổ của vùng "nguồn" được thêm vào băng thừa số tỉ lệ "đích" như 50d trên Fig.4, để cải thiện chất lượng âm thanh của quy trình điền đầy nhiều âm liên kênh. Tương tự, và như ví dụ về quang phổ được giải mã phần lớn (thay vì đầy đủ), phần "nguồn" được đề cập trước đó có thể được thu được từ quang phổ mà chưa được lọc bởi bộ lọc TNS nghịch đảo (tức là tổng hợp) có sẵn.

Do đó, các phương án trên liên quan tới khái niệm điền đầy nhiều âm liên kênh. Trong phần sau đây, khả năng được mô tả là cách mà khái niệm trên về điền đầy nhiều âm liên kênh có thể được xây dựng thành bộ mã hóa-giải mã hiện tại, cụ thể là xHE-AAC, theo cách bán tương thích theo chiều ngược. Cụ thể, việc thực hiện được ưu tiên của các phương án ở trên được mô tả, mà theo đó công cụ điền đầy âm lập thể được xây dựng thành bộ mã hóa-giải mã âm thanh dựa trên xHE-AAC theo cách báo hiệu bán tương thích theo chiều ngược. Bằng việc sử dụng cách thực hiện được mô tả thêm dưới đây, cho các tín hiệu âm lập thể nhất định, phép điền đầy âm lập thể của các hệ số biến đổi trong một trong số hai kênh trong bộ mã hóa-giải mã âm thanh dựa trên MPEG-D xHE-AAC (USAC) là khả thi, do đó cải thiện chất lượng mã hóa của các tín hiệu âm thanh nhất định đặc biệt là ở các tốc độ bit thấp. Công cụ điền đầy âm lập thể được báo hiệu theo cách bán tương thích theo chiều ngược sao cho các bộ giải mã xHE-AAC kế thừa có thể phân tách và giải mã các dòng bit mà không có các lỗi hoặc sự bỏ qua âm thanh hiển nhiên. Như đã được mô tả ở trên, chất lượng chung tốt hơn có thể đạt được nếu bộ mã hóa âm thanh có thể sử dụng tổ hợp của các hệ số được giải mã/được lượng tử hóa trước của hai kênh âm lập thể để khôi phục các hệ số được lượng tử hóa bằng không (không được truyền) của một trong số các kênh được giải mã hiện thời. Do đó mong muốn là cho phép điền đầy âm lập thể (từ trước tới các hệ số hiện tại) ngoài sự sao chép băng quang phổ (từ các hệ số kênh tần số thấp đến cao) và điền đầy nhiều âm (từ nguồn ngẫu nhiên giả không tương quan) trong bộ mã hóa âm thanh, đặc biệt là xHE-AAC hoặc bộ mã hóa dựa trên nó.

Để cho phép các dòng bit được mã hóa với phép điền đầy nhiều âm sẽ được đọc

và được phân tách bởi các bộ giải mã xHE-AAC kế thừa, công cụ điền đầy âm lập thể mong muốn sẽ được sử dụng theo cách bán tương thích theo chiều ngược: sự có mặt của nó không làm cho các bộ giải mã kế thừa dừng - hoặc thậm chí không bắt đầu - giải mã. Khả năng đọc dòng bit bởi cơ sở xHE-AAC cũng có thể tạo điều kiện thuận lợi cho việc chọn lựa thị trường.

Để đạt được mong muốn được đề cập ở trên cho tính bán tương thích theo chiều ngược đối với công cụ điền đầy âm lập thể trong ngữ cảnh xHE-AAC hoặc ngữ cảnh bắt nguồn tiềm năng của xHE-AAC, việc thực hiện sau đây bao gồm chức năng của điền đầy âm lập thể cũng như khả năng báo hiệu tương tự thông qua cú pháp trong dòng dữ liệu thực sự quan tâm đến điền đầy nhiễu âm. Công cụ điền đầy âm lập thể sẽ thực hiện theo thứ tự mô tả ở trên. Trong cặp kênh với cấu hình cửa sổ chung, hệ số của băng thừa số tỉ lệ được lượng tử hóa bằng không là, khi công cụ điền đầy âm lập thể được kích hoạt, cách khác (hoặc, như được mô tả, ngoài ra) để điền đầy nhiễu âm, được khôi phục bởi tổng hoặc chênh lệch của các hệ số của khung trước trong một trong số hai kênh, tốt hơn là kênh bên phải. Điền đầy âm lập thể được thực hiện tương tự với điền đầy nhiễu âm. Việc báo hiệu được thực hiện thông qua báo hiệu điền đầy nhiễu âm của xHE-AAC. Điền đầy âm lập thể được chuyển tải bởi thông tin phụ điền đầy nhiễu âm 8 bit. Điều này là có khả năng vì tiêu chuẩn MPEG-D USAC [3] nêu rằng tất cả 8 bit được truyền ngay cả nếu mức nhiễu âm được áp dụng là không. Trong trường hợp này, một số bit điền đầy nhiễu âm có thể được sử dụng lại cho công cụ điền đầy nhiễu âm.

Bán tương thích theo chiều ngược liên quan đến phân tách dòng bit và phát lại bởi các bộ giải mã xHE-AAC kế thừa được đảm bảo như sau. Điền đầy âm lập thể được báo hiệu thông qua mức nhiễu âm bằng không (tức là cả ba bit điền đầy nhiễu âm thứ nhất có giá trị bằng không) được theo sau bởi năm bit khác không (mà thường biểu diễn phần bù nhiễu âm) chứa thông tin phụ cho công cụ điền đầy nhiễu âm cũng như mức nhiễu âm khuyết. Vì bộ giải mã xHE-AAC kế thừa không quan tâm đến phần bù nhiễu âm 5-bit nếu mức nhiễu âm 3-bit bằng không, sự có mặt của công cụ điền đầy nhiễu âm chỉ có hiệu quả trên phép điền đầy nhiễu âm trong bộ giải mã kế thừa: điền đầy nhiễu âm bị ngắt vì ba bit thứ nhất bằng không, và phần dư của phép toán mã hóa chạy như dự định. Cụ thể, điền đầy nhiễu âm không được thực hiện do thực tế là nó

được hoạt động như quy trình điền đầy nhiễu âm, mà được ngắt kích hoạt. Do đó, bộ giải mã kế thừa vẫn cung cấp việc giải mã “từ từ” của dòng bit tăng cường 30 vì không cần làm cam tín hiệu đầu ra hoặc thậm chí là hủy bỏ việc giải mã ngay khi tiếp cận khung với phép điền đầy âm lập thể được bật. Tuy nhiên, đương nhiên là không thể cung cấp sự khôi phục được dự định, chính xác của các hệ số vạch được điền đầy âm lập thể, dẫn đến chất lượng giảm theo các khung bị ảnh hưởng so với giải mã bởi bộ giải mã thích hợp có khả năng xử lý công cụ điền đầy âm lập thể mới một cách phù hợp. Tuy nhiên, giả định công cụ điền đầy âm lập thể được sử dụng như dự định, tức là chỉ trên đầu vào âm lập thể ở các tốc độ bit thấp, chất lượng qua các bộ giải mã xHE-AAC sẽ tốt hơn nếu các khung bị ảnh hưởng sẽ biến mất do việc làm cam hoặc dẫn đến các lỗi phát lại hiển nhiên khác.

Trong phần sau đây, sự mô tả chi tiết được biểu diễn cách mà công cụ điền đầy âm lập thể có thể được xây dựng thành, như là sự mở rộng, bộ mã hóa-giải mã xHE-AAC.

Khi được xây dựng thành tiêu chuẩn, công cụ điền đầy âm lập thể có thể được mô tả như sau. Cụ thể là, công cụ điền đầy âm lập thể (stereo filling - SF) này sẽ biểu diễn công cụ mới trong phần miền tần số (frequency-domain FD) của âm thanh MPEG-H 3D. Đi đôi với phần mô tả ở trên, mục đích của công cụ điền đầy âm lập thể sẽ là sự khôi phục theo tham số của các hệ số quang phổ MDCT ở tốc độ bit thấp, tương tự với những gì đã có thể đạt được với phép điền đầy nhiễu âm theo mục 7.2 của tiêu chuẩn được mô tả trong [3]. Tuy nhiên, không giống điền đầy nhiễu âm, mà sử dụng nguồn nhiễu âm ngẫu nhiên giả để tạo ra các giá trị quang phổ MDCT của kênh FD bất kỳ, SF cũng sẽ khả dụng để khôi phục các giá trị DCT của kênh bên phải thuộc cặp kênh âm lập thể được mã hóa kết hợp sử dụng phép trộn giảm của các quang phổ MDCT bên phải và bên trái của khung trước. SF, theo phép thực hiện được nêu dưới đây, được báo hiệu bán tương thích theo chiều ngược bởi thông tin phụ điền đầy nhiễu âm mà có thể được phân tách chính xác bởi bộ giải mã MPEG-D USAC kế thừa.

Phần mô tả công cụ có thể như sau. Khi SF hoạt động trong khung FD âm lập thể kết hợp, các hệ số MDCT của các băng thừa số tỉ lệ (tức là hoàn toàn được lượng tử hóa bằng không) trống của kênh (thứ hai) bên phải, như 50d, được thay thế bởi tổng hoặc chênh lệch của các hệ số MDCT của các kênh bên trái và bên phải được giải mã

của khung trước (nếu FD). Nếu phép điền đầy nhiều âm kế thừa hoạt động cho kênh thứ hai, các giá trị ngẫu nhiên giả cũng được cộng vào từng hệ số. Các hệ số thu được của từng băng thừa số tỉ lệ sau đó được định tỉ lệ sao cho RMS của từng băng phù hợp với giá trị được truyền bởi cách của thừa số tỉ lệ của băng. Xem phần 7.3 của tiêu chuẩn trong [3].

Một số ràng buộc về hoạt động có thể được cung cấp cho việc sử dụng công cụ SF mới trong tiêu chuẩn MPEG-D USAC. Ví dụ, công cụ SF có thể khả dụng cho việc sử dụng chỉ trong kênh FD bên phải của cặp kênh FD chung, tức là phần tử cặp kênh truyền StereoCoreToolInfo() với common_window == 1. Bên cạnh đó, do sự báo hiệu bán tương thích theo chiều ngược, công cụ SF có thể là khả dụng để sử dụng chỉ khi noiseFilling == 1 trong phần chứa cú pháp UsacCoreConfig(). Nếu một trong hai kênh trong cặp này nằm trong LPD core_mode, công cụ SF có thể được sử dụng, thậm chí nếu kênh bên phải nằm trong chế độ FD.

Các thuật ngữ và định nghĩa được sử dụng sau đây nhằm mô tả rõ ràng sự mở rộng của tiêu chuẩn như được mô tả trong [3].

Cụ thể, khi các phần tử dữ liệu được nói, phần tử dữ liệu sau đây mới được đưa vào:

stereo_filling binary flag biểu thị liệu SF có được sử dụng trong khung và kênh hiện thời

Hơn nữa, các phần tử hỗ trợ mới được đưa vào:

noise_offset noise-fill offset để cải biên các thừa số tỉ lệ của các băng lượng tử hóa băng không (phần 7.2)

noise_level noise-fill level biểu diễn biên độ của nhiễu âm quang phổ được bổ sung (phần 7.2)

downmix_prev[] downmix (i.e. sum or difference) của các kênh bên trái và bên phải của khung phía trước

sf_index[g][sfb] chỉ số thừa số tỉ lệ (i.e. số nguyên được truyền) cho nhóm cửa sổ g và băng sfb

Quy trình giải mã của tiêu chuẩn này sẽ được mở rộng theo cách sau đây. Cụ thể, việc giải mã kênh FD được mã hóa liên kết âm lập thể với công cụ SF được kích hoạt được chạy theo ba bước như sau:

Đầu tiên, việc giải mã stereo_filling flag diễn ra.

stereo_filling không biểu diễn phần tử dòng bit độc lập mà được suy ra từ các phần tử điền đầy nhiễu âm, noise_offset và noise_level, in a UsacChannelPairElement() and the common_window flag in StereoCoreToolInfo(). Nếu noiseFilling == 0 or common_window == 0 hoặc kênh hiện thời là kênh (thứ nhất) bên trái trong phần tử này, stereo_filling bằng 0, và quy trình điền đầy âm lập thể kết thúc Otherwise,

```
if ((noiseFilling != 0) && (common_window != 0) && (noise_level == 0)) {
    stereo_filling = (noise_offset & 16) / 16;
    noise_level = (noise_offset & 14) / 2;
    noise_offset = (noise_offset & 1) * 16;
}
else {
    stereo_filling = 0;
}
```

Nói cách khác, nếu noise_level == 0, noise_offset chứa stereo_filling flag được theo sau bởi 4 bit của dữ liệu điền đầy nhiễu âm mà sau đó được bố trí lại. Do việc vận động này thay đổi các giá trị của noise_level and noise_offset, nó cần được thực hiện trước quy trình điền đầy nhiễu âm của phần 7.2. Hơn nữa, mã giả ở trên không được chạy trong kênh (thứ nhất) bên trái của UsacChannelPairElement() hoặc bất kỳ phần tử nào khác.

Sau đó, việc tính toán downmix_prev sẽ diễn ra.

downmix_prev[], phép trộn giảm quang phổ mà được sử dụng cho việc điền đầy âm lập thể, mà tương tự với dmx_re_prev[] được sử dụng cho ước lượng quang phổ MDST trong dự báo âm lập thể phức hợp (phần 7.7.2.3). Điều này có nghĩa là

- Tất cả các hệ số downmix_prev[] phải bằng không nếu bất kỳ kênh nào của khung và phần tử mà phép trộn giảm được thực hiện - tức là nếu khung trước khung được giải mã hiện thời - sử dụng core_mode == 1 (LPD) hoặc các kênh sử dụng các độ dài biến đổi không bằng nhau (split_transform == 1 hoặc khối chuyển sang window_sequence == EIGHT_SHORT_SEQUENCE trong chỉ một kênh) hoặc usacIndependencyFlag == 1.

- Tất cả các hệ số của `downmix_prev[ ]` phải bằng không trong quy trình điền đầy âm lập thể nếu độ dài biến đổi của kênh thay đổi từ khung cuối cùng đến khung hiện thời (tức là `split_transform == 1` đặt trước bởi `split_transform == 0`, hoặc `window_sequence == EIGHT_SHORT_SEQUENCE` đặt trước bởi `window_sequence != EIGHT_SHORT_SEQUENCE`, hoặc ngược lại tương ứng) trong phần tử hiện thời.
- Nếu việc chia phép biến đổi được áp dụng trong các kênh của khung hiện thời hoặc khung trước đó, `downmix_prev[ ]` biến diễn phép trộn giảm quang phổ được xen kẽ theo từng vạch. Chi tiết xem công cụ chia phép biến đổi.
- Nếu việc dự báo âm lập thể phức hợp không được sử dụng trong khung và phần tử hiện thời, `pred_dir equals 0`.

Kết quả là, phép trộn giảm trước chỉ được tính toán một lần cho cả hai công cụ, giảm độ phức tạp. Khác biệt duy nhất giữa `downmix_prev[ ]` và `dmx_re_prev[ ]` trong phần 7.7.2 là hành vi khi dự báo âm lập thể phức hợp không được sử dụng hiện thời, hoặc khi nó được kích hoạt nhưng `use_prev_frame == 0`. Trong trường hợp này, `downmix_prev[ ]` được tính toán cho việc giải mã điền đầy âm lập thể theo phần 7.7.2.3 mặc dù `dmx_re_prev[ ]` không cần cho việc giải mã dự báo âm lập thể phức hợp và do đó không được định rõ/bằng không.

Sau đó, điền đầy âm lập thể của các băng thừa số tỉ lệ trống sẽ được thực hiện.

Nếu `stereo_filling == 1`, quy trình sau đây được tiến hành sau quy trình điền đầy âm lập thể trong các băng thừa số định tính lệ trống ban đầu `fb[ ]` below `max_sfb_ste`, tức là tất cả các băng mà trong đó tất cả các vạch MDCT được lượng tử hóa bằng không. Thứ nhất, năng lượng của `sfb[ ]` đã cho và các vạch tương ứng trong `downmix_prev[ ]` được tính toán thông qua tổng của các bình phương đường. Sau đó, `sfbWidth` đã cho chứa số lượng vạch cho mỗi `sfb[ ]`,

```
if (energy[sfb] < sfbWidth[sfb]) { /* noise level isn't maximum, or band starts below noise-fill region */
```

```
    facDmx = sqrt((sfbWidth[sfb] - energy[sfb]) / energy_dmx[sfb]);
```

```
    factor = 0.0;
```

```
    /* if the previous downmix isn't empty, add the scaled downmix lines such that band reaches unity energy */
```

```
    for (index = swb_offset[sfb]; index < swb_offset[sfb+1]; index++) {
```

```

spectrum>window>[index] += downmix_prev>window>[index] * facDmx;
factor += spectrum>window>[index] * spectrum>window>[index];
}
if ((factor != sfbWidth[sfb]) && (factor > 0)) { /* unity energy isn't reached, so
modify band */
factor = sqrt(sfbWidth[sfb] / (factor + 1e-8));
for (index = swb_offset[sfb]; index < swb_offset[sfb+1]; index++) {
spectrum>window>[index] *= factor;
}
}
}

```

Đối với quang phổ của mỗi cửa sổ nhóm. Sau đó các thừa số tỉ lệ được áp dụng cho quang phổ thu được như trong phần 7.3 với các thừa số tỉ lệ của các băng trống được xử lý giống các thừa số tỉ lệ thông thường.

Cách khác cho phần mở rộng ở trên của tiêu chuẩn xHE-AAC sẽ sử dụng phương pháp truyền tín hiệu bán tương thích theo chiều ngược.

Phương án thực hiện ở trên trong khung mã hóa xHE-AAC mô tả phương pháp mà sử dụng một bit trong dòng bit cho việc sử dụng tín hiệu của công cụ điền đầy âm lập thể mới, được chứa trong stereo_filling, theo bộ giải mã tương ứng với Fig.2. Chính xác hơn, việc truyền tín hiệu này (gọi là truyền tín hiệu bán tương thích theo chiều ngược) cho phép dữ liệu dòng bit kế thừa sau đây - ở đây là thông tin phụ điền đầy nhiễu âm - được sử dụng một cách độc lập cho việc truyền tín hiệu: Trong phương án của sáng chế, dữ liệu điền đầy nhiễu âm không phụ thuộc vào thông tin phụ và ngược lại. Ví dụ, dữ liệu điền đầy nhiễu âm bao gồm tất cả băng không (noise_level = noise_offset = 0) có thể được truyền dẫn trong khi stereo_filling có thể truyền tín hiệu bất kỳ giá trị nào có thể (là cờ hiệu nhị phân, 0 hoặc 1).

Trong trường hợp mà ở đó tính độc lập chặt chẽ giữa dữ liệu dòng bit kế thừa và theo sáng chế không được yêu cầu và tín hiệu theo sáng chế là quyết định nhị phân, việc truyền tín hiệu rõ ràng của bit truyền tín hiệu có thể được tránh, và quyết định nhị phân đã nêu có thể được truyền tín hiệu bởi sự có mặt hoặc vắng mặt của cái được gọi là truyền tín hiệu bán tương thích theo chiều ngược ẩn. Xem phương án trên như ví dụ,

việc sử dụng điền đầy âm lập thể có thể được truyền dẫn bằng việc sử dụng đơn giản việc truyền tín hiệu mới: Nếu `noise_level` bằng không và cùng lúc, `noise_offset` bằng không, cờ hiệu `stereo_filling` flag được thiết lập bằng 1. Nếu cả hai `noise_level` và `noise_offset` được thiết lập bằng không, `stereo_filling` bằng không. Việc phụ thuộc của tín hiệu ẩn vào tín hiệu điền đầy nhiều âm kế thừa xảy ra khi cả `noise_level` và `noise_offset` bằng không. Trong trường hợp này, không rõ là việc truyền tín hiệu ẩn kế thừa hoặc SF mới được sử dụng. Để tránh sự mơ hồ như vậy, giá trị của `stereo_filling` phải được xác định trước. Trong ví dụ sáng chế, thích hợp để định rõ `stereo_filling` = 0 nếu dữ liệu điền đầy nhiều âm bao gồm tất cả bằng không, do những bộ mã hóa kế thừa không có tín hiệu có khả năng điền đầy âm lập thể khi việc điền đầy nhiều âm không được áp dụng vào khung.

Vấn đề còn lại để được giải quyết trong trường hợp việc truyền tín hiệu bán tương thích theo chiều ngược là việc là làm thế nào truyền tín hiệu `stereo_filling` == 1 và không có việc điền đầy nhiều âm cùng lúc. Như được giải thích, dữ liệu điền đầy nhiều âm phải khác không, và cường độ nhiều âm bằng không được yêu cầu, `noise_level` $((\text{noise_offset} \& 14)/2)$ như được đề cập ở trên) phải bằng không. Điều này chỉ để lại `noise_offset` $((\text{noise_offset} \& 1) * 16)$ như đã đề cập ở trên) lớn hơn 0 như là một giải pháp. Tuy nhiên, `noise_offset` mà được xem xét trong trường hợp điền đầy âm lập thể khi áp dụng các thừa số tỉ lệ, thậm chí nếu `noise_offset` bằng không. May mắn là, bộ mã hóa có thể bù cho thực tế rằng một `noise_offset` bằng 0 có thể không được truyền bằng cách thay đổi các thừa số tỉ lệ bị ảnh hưởng sao cho khi ghi dòng bit, chúng chứa phần bù mà chưa được hoàn thành trong bộ giải mã qua `noise_offset`. Điều này cho phép việc truyền tín hiệu ẩn đã nêu trong phương án ở trên với sự tăng tiềm tàng trong tốc độ dữ liệu thừa số tỉ lệ. Do đó, việc truyền tín hiệu của việc điền đầy âm lập thể trong mã giả của phần mô tả ở trên có thể được thay đổi như sau, sử dụng bit truyền tín hiệu SF được giảm để truyền `noise_offset` với 2 bit (4 giá trị) thay cho 1 bit:

```
if ((noiseFilling) && (common_window) && (noise_level == 0) &&
(noise_offset > 0)) {
    stereo_filling = 1;
    noise_level = (noise_offset & 28) / 4;
    noise_offset = (noise_offset & 3) * 8;
```

```

    }
    else {
        stereo_filling = 0;
    }

```

Nhằm mục đích hoàn thiện, Fig.6 thể hiện bộ mã hóa âm thanh theo tham số tương ứng với phương án của sáng chế. Đầu tiên, bộ mã hóa của Fig.6 mà thường được biểu thị sử dụng ký hiệu tham chiếu 90 bao gồm bộ biến đổi 92 để thực hiện phép biến đổi của phiên bản ban đầu, phiên bản không bị biến dạng của tín hiệu âm thanh được khôi phục tại đầu ra 32 của Fig.2. Như được mô tả trên Fig.3, phép biến đổi chồng lấp có thể được sử dụng với sự chuyển giữa các độ dài biến đổi khác nhau với các cửa sổ biến đổi tương ứng trong các đơn vị của các khung 44. Độ dài biến đổi khác nhau và các cửa sổ tương ứng được minh họa trên Fig.3 sử dụng ký hiệu tham chiếu 104. Theo cách tương tự với Fig.2, Fig.6 tập trung vào phần của bộ mã hóa 90 chịu trách nhiệm mã hóa một kênh của tín hiệu âm thanh đa kênh, trong đó phần miền kênh còn lại của bộ giải mã 90 thường được biểu thị sử dụng ký hiệu tham chiếu 96 trên Fig.6.

Tại đầu ra của bộ biến đổi 92 các vạch phổ và các thừa số tỉ lệ không được lượng tử hóa và về cơ bản sự mất mát mã hóa chưa xảy ra. Ảnh phổ được xuất ra bởi bộ mã hóa 92 đi vào bộ lượng tử hóa 98, mà được tạo cấu hình để lượng tử hóa các vạch phổ của ảnh phổ xuất ra bởi bộ biến đổi 92, theo từng quang phổ, thiết lập và sử dụng các thừa số tỉ lệ sơ bộ của các băng thừa số tỉ lệ. Tức là, tại đầu ra của bộ lượng tử hóa 98, các thừa số tỉ lệ sơ bộ và kết quả của các hệ số vạch phổ tương ứng và chuỗi của bộ điền đầy nhiễu âm 161, bộ lọc TNS nghịch đảo tùy chọn 28a', bộ dự báo liên kênh 24', bộ giải mã MS 26' và bộ lọc TNS nghịch đảo 28b' được nối theo thứ tự để cung cấp bộ mã hóa 90 của Fig.6 với khả năng thu được phiên bản cuối cùng được khôi phục của quang phổ hiện thời mà có thể thu được tại phía bộ giải mã ở đầu vào của bộ cung trộn giảm (xem Fig.2). Trong trường hợp sử dụng bộ dự báo liên kênh 24' và/hoặc sử dụng điền đầy nhiễu âm liên kênh trong phiên bản tạo ra nhiễu âm liên kênh sử dụng phép trộn giảm khung trước, bộ mã hóa 90 cũng bao gồm bộ cấp trộn giảm 31' để tạo ra phép trộn giảm của các phiên bản được khôi phục, các phiên bản cuối cùng của quang phổ của các kênh của tín hiệu âm thanh đa kênh. Tất nhiên, để

giảm sự tính toán, thay vì các phiên bản cuối cùng, ban đầu, được lượng tử hóa của quang phổ đã nêu của các kênh có thể được sử dụng bởi bộ cấp trộn giảm 31' trong việc tạo thành phép trộn giảm.

Bộ mã hóa 90 có thể sử dụng thông tin trên phiên bản được khôi phục, cuối cùng có sẵn của quang phổ để thực hiện dự báo quang phổ liên khung ví dụ như phiên bản có thể có nói trên của việc thực hiện dự báo liên kênh sử dụng ước lượng phần tương tự, và/hoặc để thực hiện điều khiển tốc độ, tức là để xác định, trong vòng lặp điều khiển tốc độ, các tham số có thể có được mã hóa vào dòng dữ liệu 30 bằng bộ mã hóa 90 được thiết lập theo nghĩa tối ưu về tốc độ/biến dạng.

Ví dụ, một tham số thiết lập trong vòng lặp dự báo và/hoặc vòng lặp điều khiển tốc độ của bộ mã hóa 90, đối với mỗi băng thừa số tỉ lệ được lượng tử hóa bằng không được nhận biết bởi ký hiệu nhận biết 12', thừa số tỉ lệ của băng thừa số tỉ lệ tương ứng mà chỉ được thiết lập sơ bộ bởi bộ lượng tử hóa 98. Trong việc dự đoán và/hoặc vòng lặp dự báo và/hoặc điều khiển tốc độ của bộ mã hóa 90, thừa số tỉ lệ của các băng thừa số tỉ lệ được lượng tử hóa bằng không được thiết lập theo nghĩa tối ưu và mặt tâm thính hoặc tốc độ/biến dạng để xác định mức nhiễu âm đích, như được mô tả ở trên cùng với tham số cải biên tùy chọn cũng được truyền bởi dòng dữ liệu cho khung tương ứng với phía bộ giải mã. Cần lưu ý rằng thừa số tỉ lệ này có thể được tính toán chỉ sử dụng các vạch phổ của quang phổ và kênh mà chúng thuộc về (tức là quang phổ đích, như được mô tả trước đó), hoặc cách khác, có thể được xác định sử dụng cả các vạch phổ của quang phổ kênh "đích" và, ngoài ra, các vạch phổ của quang phổ kênh khác hoặc quang phổ trộn giảm từ khung trước đó (tức là, quang phổ "nguồn" như được giới thiệu trước đó) được thu từ bộ cấp trộn giảm 31'. Cụ thể để ổn định mức nhiễu âm đích và để giảm dao động mức theo thời gian trong các kênh âm thanh được giải mã mà trên đó việc điền đầy nhiễu âm liên kênh được áp dụng, thừa số tỉ lệ đích có thể được tính toán sử dụng mối liên hệ giữa ước số năng lượng của các vạch phổ trong băng thừa số "đích" và ước số năng lượng của các vạch phổ đồng định vị trong vùng "nguồn" tương ứng. Cuối cùng, như được lưu ý ở trên, vùng "nguồn" này có thể bắt nguồn từ phiên bản cuối cùng, được khôi phục của kênh khác hoặc của phép trộn giảm của khung trước, hoặc nếu độ phức tạp của bộ mã hóa được giảm, phiên bản ban đầu, không được lượng tử hóa của kênh khác tương tự hoặc của các phiên bản ban

đầu, không được lượng tử hóa của quang phổ của khung trước.

Sau đây, việc mã hóa đa kênh và giải mã đa kênh theo các phương án được giải thích. Trong các phương án, bộ xử lý đa kênh 204 của thiết bị 201 để giải mã của Fig.1a có thể, ví dụ, được tạo cấu hình để thực hiện trên nhiều kỹ thuật dưới đây mà được mô tả liên quan đến giải mã đa kênh nhiều âm.

Đầu tiên, trước khi mô tả giải mã đa kênh, mã hóa đa kênh theo các phương án được giải thích với sự tham chiếu từ Fig.7 đến Fig.9 và sau đó, giải mã đa kênh được giải thích với sự tham chiếu đến các hình vẽ từ Fig.10 đến Fig.12.

Bây giờ, việc mã hóa đa kênh theo các phương án được giải thích với sự tham chiếu đến các hình vẽ từ Fig.7 đến Fig.9 và Fig.11:

Fig.7 thể hiện sơ đồ khối dạng giản lược của thiết bị (bộ mã hóa) 100 để mã hóa tín hiệu đa kênh 101 có ít nhất ba kênh CH1 đến CH3.

Thiết bị 100 bao gồm bộ xử lý phép lặp 102, bộ mã hóa kênh 104 và giao diện đầu ra 106.

Bộ xử lý phép lặp 102 được tạo cấu hình để tính toán, trong bước lặp thứ nhất, các giá trị tương quan liên kênh giữa mỗi cặp của ít nhất ba kênh từ CH1 đến CH3, để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị cao hơn ngưỡng, và để xử lý cặp được lựa chọn sử dụng phép toán xử lý đa kênh để suy ra các tham số đa kênh ban đầu MCH_PAR1 cho các cặp được lựa chọn và để suy ra các kênh được xử lý thứ nhất P1 và P2. Trong phần sau đây, các kênh được xử lý P1 và các kênh được xử lý P2 cũng có thể được đề cập đến lần lượt là kênh tổ hợp P1 và kênh tổ hợp P2. Hơn nữa, bộ xử lý phép lặp 102 được tạo cấu hình để thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai sử dụng ít nhất một kênh trong số các kênh được xử lý P1 hoặc P2 để suy ra các tham số đa kênh MCH_PAR2 và các kênh được xử lý thứ hai P3 và P4.

Ví dụ, như được thể hiện trên Fig.7, bộ xử lý phép lặp 102 có thể tính toán, trong bước lặp thứ nhất, giá trị tương quan liên kênh giữa cặp thứ nhất của ít nhất ba kênh CH1 đến CH3, cặp thứ nhất gồm kênh thứ nhất CH1 và kênh thứ hai CH2, giá trị tương quan liên kênh giữa cặp thứ hai của ít nhất ba kênh CH1 đến CH3, cặp thứ hai bao gồm kênh thứ hai CH2 và kênh thứ ba CH3, và giá trị tương quan liên kênh giữa cặp thứ ba của ít nhất ba kênh CH1 đến CH3, cặp thứ ba bao gồm kênh thứ nhất CH1

và kênh thứ ba CH3.

Trên Fig.7 giả định rằng trong bước lặp thứ nhất, cặp thứ ba bao gồm kênh thứ nhất CH1 và kênh thứ ba CH3 chứa giá trị tương quan liên kênh cao nhất, sao cho bộ xử lý phép lặp 102 lựa chọn, trong bước lặp thứ nhất, cặp thứ ba có giá trị tương quan liên kênh cao nhất và xử lý cặp được lựa chọn, tức là cặp thứ ba, sử dụng phép toán xử lý đa kênh để suy ra các tham số đa kênh MCH_PAR1 cho cặp được chọn và để suy ra các kênh được xử lý thứ nhất P1 và P2.

Hơn nữa, bộ xử lý phép lặp 102 có thể được tạo cấu hình để tính toán, trong bước lặp thứ hai, các giá trị tương quan liên kênh giữa từng cặp của ít nhất ba kênh từ CH1 đến CH3 và các kênh được xử lý P1 và P2, để lựa chọn, trong bước lặp thứ hai, cặp có giá trị tương quan liên kênh cao nhất hoặc có giá trị trên ngưỡng. Do đó, bộ xử lý phép lặp 102 có thể được tạo cấu hình để không lựa chọn cặp được lựa chọn của bước lặp thứ nhất trong bước lặp thứ hai (hoặc trong bất kỳ phép lặp nào nữa).

Viện dẫn đến ví dụ được thể hiện trên Fig.7, bộ xử lý phép lặp 102 có thể còn tính toán giá trị tương quan liên kênh giữa cặp kênh thứ tư bao gồm kênh thứ nhất CH1 và kênh được xử lý thứ nhất P1, giá trị tương quan liên kênh giữa cặp thứ năm bao gồm kênh thứ nhất CH1 và kênh được xử lý thứ hai P2, giá trị tương quan liên kênh giữa cặp thứ sáu bao gồm kênh thứ hai CH2 và kênh được xử lý thứ nhất P1, giá trị tương quan liên kênh giữa cặp thứ bảy bao gồm kênh thứ hai CH2 và kênh được xử lý thứ hai P2, giá trị tương quan liên kênh giữa cặp thứ tám bao gồm kênh thứ ba CH3 và kênh được xử lý thứ nhất P1, giá trị tương quan liên kênh giữa cặp thứ chín bao gồm kênh thứ ba CH3 và kênh được xử lý thứ hai P2, và giá trị tương quan liên kênh giữa cặp thứ mười bao gồm kênh được xử lý thứ nhất P1 và kênh được xử lý thứ hai P2.

Trên Fig.7 giả định rằng trong bước lặp thứ hai, cặp thứ sáu bao gồm kênh thứ hai CH2 và kênh được xử lý thứ nhất P1 chứa giá trị tương quan liên kênh cao nhất, sao cho bộ xử lý phép lặp 102 lựa chọn, trong bước lặp thứ hai, cặp thứ sáu và xử lý cặp được lựa chọn, tức là cặp thứ ba, sử dụng phép toán xử lý đa kênh để suy ra các tham số đa kênh MCH_PAR2 cho cặp được chọn và để suy ra các kênh được xử lý thứ hai P3 và P4.

Bộ xử lý phép lặp 102 có thể được tạo cấu hình để chỉ lựa chọn cặp khi chênh lệch mức của cặp là nhỏ hơn ngưỡng, ngưỡng nhỏ hơn 40dB, 25dB, 12dB hoặc nhỏ

hơn 6dB. Do đó, các ngưỡng 25 hoặc 40dB tương ứng với các góc luân phiên 3 hoặc 0,5 độ.

Bộ xử lý phép lặp 102 có thể được tạo cấu hình để tính toán các giá trị tương quan nguyên được chuẩn hóa, trong đó bộ xử lý phép lặp 102 có thể được tạo cấu hình để lựa chọn cặp, khi giá trị tương quan nguyên lớn hơn ví dụ 0,2 hoặc tốt hơn là 0,3.

Hơn nữa, bộ xử lý phép lặp 102 có thể cung cấp các kênh thu được từ phép xử lý đa kênh tới bộ mã hóa 104. Ví dụ, viện dẫn đến Fig.7, bộ xử lý phép lặp 102 có thể cung cấp kênh được xử lý thứ ba P3 và kênh được xử lý thứ tư P4 thu được từ phép xử lý đa kênh được thực hiện trong bước lặp thứ hai và kênh được xử lý thứ hai P2 thu được từ phép xử lý đa kênh được thực hiện trong bước lặp thứ nhất cho bộ mã hóa kênh 104. Do đó, bộ xử lý phép lặp 102 có thể chỉ cung cấp các kênh được xử lý cho bộ mã hóa kênh 104 mà không được xử lý (nữa) trong bước lặp sau đó. Như được thể hiện trên Fig.7, kênh được xử lý thứ nhất P1 không được cung cấp cho bộ mã hóa kênh 104 vì nó không được xử lý thêm trong bước lặp thứ hai.

Bộ mã hóa kênh 104 có thể được tạo cấu hình để mã hóa các kênh từ P2 đến P4 thu được từ việc xử lý phép lặp (hoặc xử lý đa kênh) được thực hiện bởi bộ xử lý phép lặp 102 để thu được các kênh được mã hóa từ E1 đến E3.

Ví dụ, bộ mã hóa kênh 104 có thể được tạo cấu hình để sử dụng các bộ mã hóa đơn (hoặc các hộp đơn hoặc các công cụ đơn) 120_1 đến 120_3 để mã hóa các kênh từ P2 đến P4 thu được từ việc xử lý phép lặp (hoặc xử lý đa kênh). Các hộp đơn có thể được tạo cấu hình để mã hóa các kênh sao cho ít bit hơn được yêu cầu để mã hóa kênh có năng lượng ít hơn (hoặc biên độ nhỏ hơn) so với để mã hóa kênh có năng lượng lớn hơn (hoặc biên độ cao hơn). Các hộp đơn 120_1 đến 120_3 có thể là, ví dụ, các bộ mã hóa âm thanh dựa trên phép biến đổi. Hơn nữa, bộ mã hóa kênh 104 có thể được tạo cấu hình để sử dụng các bộ mã hóa âm lập thể (ví dụ các bộ mã hóa âm lập thể theo tham số, hoặc các bộ mã hóa âm lập thể suy giảm) để mã hóa các kênh từ P2 đến P4 thu được từ việc xử lý phép lặp (hoặc xử lý đa kênh).

Giao diện đầu ra 106 có thể được tạo cấu hình để tạo ra tín hiệu đa kênh được mã hóa 107 có các kênh được mã hóa E1 đến E3 và các tham số đa kênh MCH_PAR1 và MCH_PAR2.

Ví dụ, giao diện đầu ra 106 có thể được tạo cấu hình để tạo ra tín hiệu đa kênh

được mã hóa 107 như chuỗi tín hiệu hoặc chuỗi dòng bit, và sao cho các tham số đa kênh MCH_PAR2 ở trong tín hiệu được mã hóa 107 trước các tham số đa kênh MCH_PAR1. Do đó, bộ giải mã, phương án mà sẽ được mô tả sau đây có viện dẫn đến Fig.10, sẽ nhận các tham số đa kênh MCH_PAR2 trước các tham số đa kênh MCH-PAR1.

Trên Fig.7 bộ xử lý phép lặp 102 thực hiện hai phép toán xử lý đa kênh theo cách làm ví dụ, phép toán xử lý đa kênh trong bước lặp thứ nhất và phép toán xử lý đa kênh trong bước lặp thứ hai. Đương nhiên, bộ xử lý phép lặp 102 cũng có thể thực hiện các phép toán xử lý đa kênh nữa trong các bước lặp tiếp theo. Do đó, bộ xử lý phép lặp 102 có thể được tạo cấu hình để thực hiện các bước lặp cho đến khi đạt được tiêu chuẩn kết thúc phép lặp. Tiêu chuẩn kết thúc phép lặp có thể là tiêu chuẩn mà số lượng cực đại của bước lặp bằng hoặc cao hơn tổng số lượng kênh của tín hiệu đa kênh 101 là hai, hoặc trong đó tiêu chuẩn kết thúc phép lặp là, khi giá trị tương quan liên kênh không có giá trị lớn hơn ngưỡng, ngưỡng tốt hơn là lớn hơn 0,2 hoặc ngưỡng tốt hơn là bằng 0,3. Trong phương án nữa, tiêu chuẩn kết thúc phép lặp có thể là tiêu chuẩn mà số lượng cực đại của bước lặp bằng hoặc cao hơn tổng số lượng kênh của tín hiệu đa kênh 101, hoặc trong đó tiêu chuẩn kết thúc phép lặp là, khi giá trị tương quan liên kênh không có giá trị lớn hơn ngưỡng, ngưỡng tốt hơn là lớn hơn 0,2 hoặc ngưỡng tốt hơn là bằng 0,3.

Với mục đích minh họa, các phép toán xử lý đa kênh được thực hiện bởi bộ xử lý phép lặp 102 trong bước lặp thứ nhất và bước lặp thứ hai được minh họa rõ ràng trên Fig.7 bởi các hộp xử lý 110 và 112. Các hộp xử lý 110 và 112 có thể được thực hiện trên phần cứng hoặc phần mềm. Ví dụ, các hộp xử lý 110 và 112 có thể là các hộp âm lập thể.

Do đó, sự phụ thuộc tín hiệu liên kênh có thể được khai thác bằng cách áp dụng theo cấp các công cụ mã hóa âm lập thể kết hợp. Trái lại với các phương pháp MPEG trước, các cặp tín hiệu sẽ được xử lý không được xác định trước bởi đường dẫn tín hiệu cố định (ví dụ cây mã hóa âm lập thể) nhưng có thể được thay đổi động để thích ứng với các đặc tính tín hiệu đầu vào. Các đầu vào của hộp âm lập thể thực có thể là (1) các kênh chưa được xử lý, ví dụ như các kênh CH1 đến CH3, (2) các đầu ra của hộp âm lập thể trước, ví dụ như các tín hiệu được xử lý P1 đến P4, hoặc (3) kênh tổ

hợp của kênh chưa được xử lý và đầu ra của âm lập thể trước.

Việc xử lý bên trong hộp âm lập thể 110 và 112 có thể là dựa trên dự báo (giống hộp dự báo phức hợp trong USAC) hoặc dựa trên KLT/PCAT (các kênh đầu vào được luân phiên (ví dụ, qua ma trận luân phiên) trong bộ mã hóa để tối đa hóa việc nén năng lượng, tức là tập trung năng lượng tín hiệu vào một kênh, trong bộ giải mã, các tín hiệu được luân phiên sẽ được biến đổi theo các hướng tín hiệu đầu vào ban đầu).

Trong phương án có thể thực hiện của bộ mã hóa 100, (1) bộ mã hóa tính toán tương quan liên kênh giữa mỗi cặp kênh và lựa chọn một cặp tín hiệu thích hợp của các tín hiệu đầu vào và áp dụng công cụ âm lập thể vào các kênh được lựa chọn; (2) bộ mã hóa tính toán lại tương quan liên kênh giữa tất cả các kênh (các kênh chưa được xử lý cũng như các kênh đầu ra trung gian được xử lý) và lựa chọn một cặp tín hiệu thích hợp trong các tín hiệu đầu vào và áp dụng công cụ âm lập thể vào các kênh được lựa chọn; và (3) bộ mã hóa lặp lại bước (2) cho đến khi tất cả tương quan liên kênh thấp hơn ngưỡng hoặc nếu số lượng cực đại của các phép biến đổi được áp dụng.

Như đã được đề cập, các cặp tín hiệu được xử lý bởi bộ mã hóa 100, hoặc chính xác hơn là bộ xử lý phép lặp 102 không được định trước bởi đường tín hiệu cố định (ví dụ cây mã hóa âm lập thể) mà có thể thay đổi một cách linh hoạt để thích ứng với các đặc tính tín hiệu đầu vào. Từ đó, bộ mã hóa 100 (hoặc bộ xử lý phép lặp 102) có thể được tạo cấu hình để khôi phục cây âm lập thể phụ thuộc vào ít nhất ba kênh CH1 đến CH3 của tín hiệu (đầu vào) đa kênh 101. Nói cách khác, bộ mã hóa 100 (hoặc bộ xử lý phép lặp 102) có thể được tạo cấu hình để xây dựng cây âm lập thể dựa trên tương quan liên kênh (ví dụ bằng việc tính toán, trong bước lặp thứ nhất, các giá trị tương quan liên kênh giữa mỗi cặp kênh của ít nhất ba kênh CH1 đến CH3, để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc giá trị lớn hơn ngưỡng và bằng việc tính toán, trong bước lặp thứ hai, các giá trị tương quan liên kênh giữa mỗi cặp của ít nhất ba kênh và các kênh được xử lý trước, để lựa chọn, trong bước lặp thứ hai, cặp có giá trị cao nhất hoặc giá trị lớn hơn ngưỡng. Theo phương pháp một bước, ma trận tương quan có thể được tính toán để mỗi phép lặp có thể chứa các tương quan của tất cả các kênh, trong các phép lặp trước mà có khả năng được xử lý.

Như được biểu thị ở trên, bộ xử lý phép lặp 102 có thể được tạo cấu hình để suy

ra các tham số đa kênh MCH_PAR1 cho cặp kênh được lựa chọn trong bước lắp thứ nhất và suy ra các tham số đa kênh MCH_PAR2 cho cặp được lựa chọn trong bước lắp thứ hai. Các tham số đa kênh MCH_PAR1 có thể bao gồm việc nhận biết cặp kênh thứ nhất (hoặc chỉ số) mà nhận biết (hoặc truyền tín hiệu) cặp gồm các kênh được lựa chọn trong bước lắp thứ nhất, tổng đó các tham số đa kênh MCH_PAR2 có thể bao gồm việc nhận biết cặp kênh thứ hai (hoặc chỉ số) nhận biết (hoặc truyền tín hiệu) cặp gồm các kênh được lựa chọn trong bước lắp thứ hai.

Sau đây, việc ghi chỉ số hiệu quả của các tín hiệu đầu vào được mô tả. Ví dụ, các cặp kênh có thể được truyền tín hiệu một cách hiệu quả sử dụng chỉ số duy nhất cho mỗi cặp, phụ thuộc vào tổng số lượng các kênh. Ví dụ, chỉ số của các cặp cho sáu kênh có thể được thể hiện trong bảng dưới đây:

					0	1
					2	3
						4

Ví dụ, trong bảng trên, chỉ số 5 có thể truyền tín hiệu cặp gồm có kênh thứ nhất và kênh thứ hai. Tương tự, chỉ số 6 có thể truyền tín hiệu cặp gồm có kênh thứ nhất và kênh thứ ba.

Tổng số lượng các chỉ số cặp kênh có thể cho n kênh có thể được tính toán bằng:

$$\text{numPairs} = \text{numChannels} * (\text{numChannels} - 1) / 2$$

Do đó, số lượng các bit cần cho việc truyền tín hiệu một cặp kênh lên đến:

$$\text{numBits} = \text{floor}(\log_2(\text{numPairs}-1))+1$$

Hơn nữa, bộ mã hóa 100 có thể sử dụng mặt nạ kênh. Cấu hình của công cụ đa kênh có thể chứa mặt nạ kênh biểu thị kênh mà công cụ được kích hoạt. Do đó, các kênh hiệu quả/nâng cao tần số thấp (LFE = low frequency effects/enhancement channels) có thể được loại bỏ khỏi việc ghi chỉ số cặp kênh, cho phép mã hóa hiệu quả hơn. Ví dụ, đối với thiết lập 11.1, việc này giảm số cặp kênh từ $12 \times 11/2 = 66$ đến $11 \times 10/2 = 55$, cho phép việc truyền tín hiệu với 6 bit thay vì 7 bit. Cơ cấu này có thể được sử dụng để loại trừ các kênh dành cho các đối tượng đơn (ví dụ dấu đa ngôn ngữ). Trên giải mã mặt nạ kênh (channelMask), ánh xạ kênh (channelMap) có thể được tạo ra để cho phép ánh xạ lại các chỉ số cặp kênh tới các kênh của bộ giải mã.

Hơn nữa, bộ xử lý phép lặp 102 có thể được tạo cấu hình để suy ra, cho khung thứ nhất, nhiều chỉ thị cặp được lựa chọn, trong đó giao diện đầu ra 106 có thể được tạo cấu hình để bao gồm, vào tín hiệu đa kênh 107, đối với khung thứ hai, theo sau khung thứ nhất, bộ chỉ thị duy trì, chỉ thị rằng khung thứ hai có số lượng chỉ thị cặp được lựa chọn giống khung thứ nhất.

Bộ chỉ thị hoặc cờ hiệu cây duy trì có thể được sử dụng để truyền tín hiệu mà rằng không cây mới nào được truyền dẫn, mà cây âm lập thể cuối cùng sẽ được sử dụng. Điều này có thể được sử dụng để tránh đa truyền dẫn của cấu hình cây âm lập thể tương tự nếu các đặc tính tương quan kênh duy trì cố định trong thời gian dài.

Fig.8 thể hiện sơ đồ khối dạng giản lược của hộp âm lập thể 110, 112. Hộp âm lập thể 110, 112 bao gồm các đầu ra cho tín hiệu đầu vào thứ nhất I1 và tín hiệu đầu vào thứ hai I2, và các đầu ra cho tín hiệu đầu ra thứ nhất O1 và tín hiệu đầu ra thứ hai O2. Như được biểu thị trên Fig.8, sự phụ thuộc vào các tín hiệu đầu ra O1 và O2 từ tín hiệu đầu vào I1 và I2 có thể được mô tả bởi các tham số - s S1 đến S4.

Bộ xử lý phép lặp 102 có thể sử dụng (hoặc bao gồm) các ô âm lập thể 110, 112 để thực hiện các phép toán xử lý đa kênh trên các kênh đầu vào và/hoặc các kênh được xử lý để suy ra các kênh được xử lý (khác). Ví dụ, bộ xử lý phép lặp 102 có thể được tạo cấu hình để sử dụng các hộp âm lập thể luân phiên dựa trên dự báo chung hoặc dựa trên phép biến đổi Karhunen-Loève (KLT - Karhunen-Loève-Transformation) 110, 112.

Bộ mã hóa chung (hộp âm lập thể phía bộ mã hóa) có thể được tạo cấu hình để mã hóa các tín hiệu đầu vào I_1 và I_2 để thu được các tín hiệu đầu ra O_1 và O_2 dựa trên phương trình:

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = \begin{bmatrix} s_1 & s_2 \\ s_3 & s_4 \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}.$$

Bộ giải chung (hộp âm lập thể phía bộ giải mã) có thể được tạo cấu hình để mã hóa các tín hiệu đầu vào I_1 và I_2 để thu được các tín hiệu đầu ra O_1 và O_2 dựa trên phương trình:

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = \begin{bmatrix} s_1 & s_2 \\ s_3 & s_4 \end{bmatrix}^{-1} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}.$$

Bộ mã hóa dựa trên dự báo (hộp âm lập thể phía bộ mã hóa) có thể được tạo cấu hình để mã hóa các tín hiệu đầu vào I_1 và I_2 để thu được các tín hiệu đầu ra O_1 và O_2 dựa trên phương trình

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = 0.5 \cdot \begin{bmatrix} 1 & 1 \\ 1-p & -(1+p) \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix},$$

trong đó p là hệ số dự báo.

Bộ giải mã dựa trên dự báo (hoặc âm lập thể phía bộ giải mã) có thể được tạo cấu hình để giải mã các tín hiệu đầu vào I_1 và I_2 để thu các tín hiệu đầu ra O_1 và O_2 dựa trên phương trình:

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = \begin{bmatrix} 1+p & 1 \\ 1-p & -1 \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}.$$

Bộ mã hóa luân phiên dựa trên KLT (hoặc hộp âm lập thể phía bộ mã hóa) có thể được tạo cấu hình để mã hóa các tín hiệu đầu vào I_1 đến I_2 để thu được các tín hiệu đầu ra O_1 và O_2 dựa trên phương trình:

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}.$$

Bộ giải mã luân phiên dựa trên KLT (hoặc hộp âm lập thể phía bộ giải mã) có thể được tạo cấu hình để giải mã các tín hiệu đầu vào I_1 và I_2 để thu các tín hiệu đầu ra O_1 và O_2 dựa trên phương trình:

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}.$$

Sau đây, việc tính toán góc luân phiên α cho phép luân phiên dựa trên KLT được mô tả.

Góc luân phiên α cho phép luân phiên dựa trên KLT có thể được xác định như:

$$\alpha = \frac{1}{2} \tan^{-1} \left(\frac{2c_{12}}{c_{11} - c_{22}} \right)$$

với c_{xy} là các mục nhập của ma trận tương quan không được chuẩn hóa, trong đó c_{11} , c_{22} là năng lượng các kênh.

Việc này có thể được thực hiện sử dụng hàm atan2 cho phép lấy vi phân giữa các tương quan âm trong tử số và chênh lệch năng lượng âm trong mẫu số:

```
alpha = 0,5*atan2(2*correlation[ch1][ch2],
  (correlation[ch1][ch1] - correlation[ch2][ch2]));
```

Ngoài ra, bộ xử lý phép lặp 102 có thể được tạo cấu hình để tính toán tương quan liên kênh sử dụng khung của mỗi kênh bao gồm nhiều băng sao cho giá trị tương quan liên kênh đơn cho nhiều băng được thu được, trong đó bộ xử lý phép lặp 102 có thể được tạo cấu hình để thực hiện xử lý đa kênh cho mỗi băng của nhiều băng sao cho các tham số đa kênh được thu từ mỗi băng trong số nhiều băng.

Từ đó, bộ xử lý phép lặp 102 có thể được tạo cấu hình để tính toán các tham số âm lập thể trong xử lý đa kênh, trong đó bộ xử lý phép lặp 102 có thể được tạo cấu hình để chỉ thực hiện xử lý âm lập thể trong các băng, trong đó tham số âm lập thể cao hơn ngưỡng được lượng tử hóa bằng không được xác định bởi bộ lượng tử hóa âm lập thể (ví dụ bộ mã hóa luân phiên dựa trên KLT). Ví dụ, các tham số âm lập thể, có thể là MS Bật/Tắt hoặc các góc luân phiên hoặc các hệ số dự báo).

Ví dụ, bộ xử lý phép lặp 102 có thể được tạo cấu hình để tính toán các góc luân phiên trong xử lý đa kênh, trong đó bộ xử lý phép lặp 102 có thể được tạo cấu hình chỉ để thực hiện xử lý luân phiên trong các băng, trong đó góc luân phiên cao hơn ngưỡng được lượng tử hóa bằng không được lượng tử hóa bằng không được định rõ bởi bộ lượng tử hóa góc luân phiên (ví dụ bộ mã hóa luân phiên dựa trên KLT).

Do đó bộ mã hóa 100 (hoặc giao diện đầu vào 106) có thể được tạo cấu hình để truyền dẫn thông tin phép biến đổi/ phép luân phiên như là một tham số cho quang phổ hoàn thiện (hộp băng đầy đủ) hoặc các tham số phụ thuộc đa tần số cho các đường quang phổ.

Bộ mã hóa 100 có thể được tạo cấu hình để tạo ra dòng bit 107 dựa trên các bảng sau:

Bảng 1 — Cú pháp mpeg3daExtElementConfig()

Cú pháp	Số lượng các bit	Thuật nhớ
<pre> mpeg3daExtElementConfig() { usacExtElementType = escapedValue(4, 8, 16); usacExtElementConfigLength = escapedValue(4, 8, 16); if (usacExtElementDefaultLengthPresent) { usacExtElementType = escapedValue(8, 16, 0); } else { usacExtElementDefaultLength = 0; } usacExtElementPayloadFrag; switch (usacExtElementType) { case ID_EXT_ELE_FILL: /* No configuration element */ break; case ID_EXT_ELE_MPEGS: SpatialSpecificConfig(); break; case ID_EXT_ELE_SAOC: SAOCSpecificConfig(); break; case ID_EXT_ELE_AUDIOPREROLL: </pre>	1	uimbsf
<pre> usacExtElementPayloadFrag; </pre>	1	uimbsf

```

        /* No configuration element */
        break;
    case ID_EXT_ELE_UNI_DRC:
        mpeg3daUniDrcConfig();
        break;
    case ID_EXT_ELE_OBJ_METADATA:
        ObjectMetadataConfig();
        break;
    case ID_EXT_ELE_SAOC:
        SAOC3DSpecificConfig();
        break;
    case ID_EXT_ELE_SAOC:
        HOAConfig();
        break;
    case ID_EXT_ELE_MCC: /* multi channel
coding */
        MCCConfig(grp);
        break;
    case ID_EXT_ELE_FMT_CNVRTR
        /* No configuration element */
        break;
    default:
        while (usacExtElementConfigLength--) {
            tmp;
        }
        break;
    }
}

```

NOTE

8 uimbsf

Lưu ý: Mục nhập mặc định cho usacExtElementType được sử dụng cho unknown extElementTypes để các bộ giải mã kế thừa có thể xử lý các phần mở rộng sau này

Table 21 — Syntax of MCCCConfig(),

Cú pháp	Số lượng các bit	Thuật nhớ
<pre> MCCCConfig(grp) { nChannels = 0 for(chan=0;chan < bsNumberOfSignals[grp]; chan++) chanMask[chan] if(chanMask[chan] > 0) { mctChannelMap[nChannels]=chan; nChannels++; } } </pre>	1	
NOTE: Phần tử ID_USAC_EXT tương ứng sẽ ở trước phần tử âm thanh bất kỳ của nhóm tín hiệu nhất định grp..		

Bảng 32 — Cú pháp MultichannelCodingBoxBandWise()

Cú pháp	Số lượng các bit	Thuật nhớ
<pre> MultichannelCodingBoxBandWise() { for(pair=0; pair<numPairs;pair++) { if (keepTree == 0) { channelPairIndex[pair] } else { </pre>	nBits NOTE 1)	

```

        channelPairIndex[pair]=
        lastChannelPairIndex[pair];
    }

    hasMctMask 1
    hasBandwiseAngles 1

    if (hasMctMask ||
hasBandwiseAngles) {
        isShort 1
        numMaskBands; 5
        if (isShort) {
            numMaskBands =
numMaskBands*8
        }
        } else { NOTE 2)
            numMaskBands =
MAX_NUM_MC_BANDS;
        }
        if (hasMctMask ||
hasBandwiseAngles) {
            for(j=0;j<numMaskBands;j++)
{
                msMask[pair][j]; 1
            } else {

for(j=0;j<numMaskBands;j++) {
            msMask[pair][j] = 1;
        }
    }
}

```

<pre> } if (qFactorIndex > 0) { delta_code_time = 0; } else { delta_code_time; } if (hasBandwiseAngles == 0) { </pre>		
	1	
	1..10	vlclbf
<pre> hcod_angle[dpcm_alpha[pair][0]]; } else { for(j=0;j< numMaskBands;j++) { if (msMask[pair][j] == 1) { </pre>		
	1..10	vlclbf
<pre> hcod_angle[dpcm_alpha[pair][j]]; } } } } } </pre>		
NOTE 1) $nBits = \text{floor}(\log_2(nChannels * (nChannels - 1) / 2 - 1)) + 1$		

Bảng 4 — Cú pháp of MultichannelCodingBoxFullband()

Cú pháp	Số lượng các bit	Thuật nhớ
<pre> MultichannelCodingBoxFullband() { for (pair=0; pair<numPairs; pair++) { If(keepTree == 0) { </pre>		

channelPairIndex[pair]	nBits
}	NOTE 1)
else {	
numPairs = lastNumPairs;	
}	
alpha;	8
}	
NOTE: 1) $nBits = \text{floor}(\log_2(nChannels * (nChannels - 1) / 2 - 1)) + 1$	

Bảng 5 — Cú pháp MultichannelCodingFrame()

Cú pháp	Số lượng các bit	Thuật nhớ
<pre> MultichannelCodingFrame() { MCCSignalingType keepTree if(bit == 0) { numPairs } else { numPairs=lastNumPairs; } if(MCCSignalingType == 0) { /* tree of standard stereo boxes */ for(i=0;i<numPairs;i++) { MCCBox[i] = StereoCoreToolInfo(0); } } } </pre>	2 1 5	

```
        if(MCCSignalingType == 1) {  
/* arbitrary mct trees */  
  
        MultichannelCodingBoxBandWise();  
        }  
        if(MCCSignalingType == 2) {  
/* arbitrary mct trees */  
  
        }  
        if(MCCSignalingType == 3) {  
/* arbitrary mct trees */  
  
        MultichannelCodingBoxFullband();  
        }  
    }
```

Bảng 6 — Giá trị usacExtElementType

usacExtElementType	Giá trị
ID_EXT_ELE_FILL	0
ID_EXT_ELE_MPEGS	1
ID_EXT_ELE_SAOC	2
ID_EXT_ELE_AUDIOPRER OLL	3
ID_EXT_ELE_UNI_DRC	4
ID_EXT_ELE_OBJ_METAD ATA	5
ID_EXT_ELE_SAOC_3D	6
ID_EXT_ELE_HOA	7
ID_EXT_ELE_FMT_CNVRT R	8
ID_EXT_ELE_MCC	9, 12000, 10
/* reserved for ISO use */	10,-127
/* reserved for use outside of ISO scope */	128 and higher
NOTE: Các giá trị usacExtElementType ứng dụng cụ thể được chỉ thị nằm trong không gian dành riêng cho việc sử dụng bên ngoài phạm vi ISO. Chúng được bỏ qua bởi bộ giải mã như là cấu trúc tối thiểu được yêu cầu bởi bộ giải mã để bỏ qua các phần mở rộng.	

Bảng 7 - Diễn giải các khối dữ liệu cho việc giải mã tải mở rộng

usacExtElementType	usacExtElementSegmentData được ghép nối biểu diễn
ID_EXT_ELE_FILL	Series of fill_byte
ID_EXT_ELE_MPEGS	SpatialFrame()

ID_EXT_ELE_SAOC	SaocFrame()
ID_EXT_ELE_AUDIOPRE ROLL	AudioPreRoll()
ID_EXT_ELE_UNI_DRC	uniDrcGain() as defined in ISO/IEC 23003-4
ID_EXT_ELE_OBJ_META DATA	object_metadata()
ID_EXT_ELE_SAOC_3D	Saoc3DFrame()
ID_EXT_ELE_HOA	HOAFrame()
ID_EXT_ELE_FMT_CNV RTR	FormatConverterFrame()
ID_EXT_ELE_MCC	MultichannelCodingFrame()
unknown	unknown data. Khối dữ liệu sẽ được loại bỏ.

Fig.9 thể hiện sơ đồ khối dạng giản lược của bộ xử lý phép lặp 102 theo phương án. Trong phương án được thể hiện trên Fig.9, tín hiệu đa kênh 101 là tín hiệu kênh 5.1 có sáu kênh: kênh bên trái L, kênh bên phải R, kênh lập thể bên trái Ls, kênh lập thể bên phải Rs, kênh trung tâm C và kênh ảnh hưởng tần số thấp LFE.

Như được biểu thị trên Fig.9, kênh LFE không được xử lý bởi bộ xử lý phép lặp 102. Điều này có thể là trường hợp do các giá trị tương quan liên kênh giữa kênh LFE và mỗi kênh trong số năm kênh L, R, Ls, Rs, and C là nhỏ, hoặc do các mặt nạ kênh biểu thị không xử lý kênh LFE mà sẽ được giả định sau đây.

Trong bước lặp thứ nhất, bộ xử lý phép lặp 102 tính toán các giá trị tương quan liên kênh giữa từng cặp của năm kênh L, R, Ls, Rs, và C, để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị trên ngưỡng. Trên Fig.9, giả định rằng kênh bên trái L và kênh bên phải R có giá trị cao nhất, sao cho bộ xử lý phép lặp 102 xử lý kênh bên trái L và kênh bên phải R bằng cách sử dụng hộp âm lập thể (hoặc công cụ âm lập thể) 110, mà thực hiện phép toán xử lý phép toán đa kênh, để suy ra các kênh được xử lý thứ nhất và thứ hai P1 và P2.

Trong bước lặp thứ hai, bộ xử lý phép lặp 102 tính toán giá trị tương quan liên kênh giữa từng cặp của năm kênh L, R, Ls, Rs, và C và các kênh được xử lý P1 và P2, để lựa chọn, trong bước lặp thứ hai, cặp có giá trị cao nhất hoặc có giá trị trên ngưỡng. Trên Fig.9, giả định rằng kênh bao quanh bên trái Ls và kênh bao quanh bên phải Rs có giá trị cao nhất, sao cho bộ xử lý phép lặp 102 xử lý kênh bao quanh bên trái Ls và kênh bao quanh bên phải Rs bằng cách sử dụng hộp âm lập thể (hoặc công cụ âm lập thể) 112, để suy ra các kênh được xử lý thứ ba và thứ tư P3 và P4.

Trong bước lặp thứ ba, bộ xử lý phép lặp 102 tính toán giá trị tương quan liên kênh giữa từng cặp của năm kênh L, R, Ls, Rs, và C và các kênh được xử lý P1 đến P4, để lựa chọn, trong bước lặp thứ ba, cặp có giá trị cao nhất hoặc có giá trị trên ngưỡng. Trên Fig.9, giả định rằng kênh được xử lý thứ nhất P1 và kênh được xử lý thứ ba P3 có giá trị cao nhất, sao cho bộ xử lý phép lặp 102 xử lý kênh được xử lý thứ nhất P1 và kênh được xử lý thứ ba P3 bằng cách sử dụng hộp âm lập thể (hoặc công cụ âm lập thể) 114, để suy ra các kênh được xử lý thứ năm và thứ sáu P5 và P6.

Trong bước lặp thứ tư, bộ xử lý phép lặp 102 tính toán giá trị tương quan liên kênh giữa từng cặp của năm kênh L, R, Ls, Rs, và C và các kênh được xử lý P1 đến P6, để lựa chọn, trong bước lặp thứ tư, cặp có giá trị cao nhất hoặc có giá trị trên ngưỡng. Trên Fig.9, giả định rằng kênh được xử lý thứ năm P5 và kênh trung tâm C có giá trị cao nhất, sao cho bộ xử lý phép lặp 102 xử lý kênh được xử lý thứ năm P5 và kênh trung tâm C bằng cách sử dụng hộp âm lập thể (hoặc công cụ âm lập thể) 115, để suy ra các kênh được xử lý thứ bảy và thứ tám P7 và P8.

Các hộp âm lập thể 110 và 116 có thể là các hộp âm lập thể MS, tức là các hộp âm thanh lập thể ở giữa/ở phía bên được tạo cấu hình để cung cấp kênh giữa và kênh bên. Kênh giữa có thể là tổng của các kênh đầu vào của hộp âm lập thể, trong đó kênh bên có thể là chênh lệch giữa các kênh đầu vào của hộp âm lập thể. Hơn nữa, các hộp âm lập thể 110 và 116 có thể là các hộp luân phiên hoặc các hộp dự báo âm lập thể.

Trên Fig.9, kênh được xử lý thứ nhất P1, kênh được xử lý thứ ba P3 và kênh được xử lý thứ năm P5 có thể là các kênh giữa, trong đó kênh được xử lý thứ hai P2, kênh được xử lý thứ tư P4 và kênh được xử lý thứ sáu P6 có thể là các kênh bên.

Hơn nữa, như được biểu thị trên Fig.9, bộ xử lý phép lặp 102 có thể được tạo cấu hình để thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai và, nếu có thể, trong bước lặp nữa bất kỳ, sử dụng các kênh đầu vào L, R, Ls, Rs, và C và (chỉ) các kênh giữa P1, P3, P5 trong số các kênh được xử lý. Nói cách khác, bộ xử lý phép lặp 102 có thể được tạo cấu hình để không sử dụng các kênh bên P1, P3 và P5 trong số các kênh được xử lý trong việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai và, nếu có thể, trong bước lặp nữa bất kỳ.

Fig.11 thể hiện lưu đồ của phương pháp 300 để mã hóa tín hiệu đa kênh có ít nhất ba kênh. Phương pháp 300 bao gồm bước 302 là tính toán, trong bước lặp thứ nhất, các giá trị tương quan liên kênh giữa từng cặp của ít nhất ba kênh, lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị trên ngưỡng, và xử lý cặp được chọn bằng cách sử dụng phép toán xử lý đa kênh để suy ra các tham số đa kênh MCH_PAR1 cho cặp được chọn và để suy ra các kênh được xử lý thứ nhất; bước 304 là thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai bằng cách sử dụng ít nhất một trong số các kênh được xử lý để suy ra các tham số đa kênh MCH_PAR2 và các kênh được xử lý thứ hai; bước 306 là mã hóa các kênh thu được từ việc xử lý phép lặp được thực hiện bởi bộ xử lý phép lặp để thu được các kênh được mã hóa; và bước 308 là tạo ra tín hiệu đa kênh được mã hóa có các kênh được mã hóa và các tham số thứ nhất và các tham số đa kênh MCH_PAR2.

Trong phần sau đây, việc xử lý đa kênh được giải thích.

Fig.10 thể hiện sơ đồ khối dạng giản lược của thiết bị (bộ giải mã) 200 để giải mã tín hiệu đa kênh được mã hóa 107 có các kênh được mã hóa E1 đến E3 và ít nhất hai tham số đa kênh MCH_PAR1 và MCH_PAR2.

Thiết bị bao gồm bộ giải mã kênh 202 và bộ xử lý đa kênh 204.

Bộ giải mã kênh 202 được tạo cấu hình để giải mã các kênh được mã hóa từ E1 đến E3 để thu được các kênh được giải mã trong D1 đến D3.

Ví dụ, bộ giải mã kênh 202 có thể bao gồm ít nhất ba bộ giải mã đơn (hoặc các hộp đơn, hoặc các công cụ đơn) 206_1 đến 206_3, trong đó từng bộ giải mã đơn 206_1 đến 206_3 có thể được tạo cấu hình để giải mã một trong số ít nhất ba kênh được mã

hóa E1 đến E3, để thu được kênh được giải mã tương ứng E1 đến E3. Các bộ giải mã đơn 206_1 đến 206_3 có thể là, ví dụ, các bộ giải mã âm thanh dựa trên phép biến đổi.

Bộ xử lý đa kênh 204 được tạo cấu hình để thực hiện phép xử lý đa kênh bằng cách sử dụng cặp kênh được giải mã thứ hai được nhận biết bởi các tham số đa kênh MCH_PAR2 và sử dụng các tham số đa kênh MCH_PAR2 để thu được các kênh được xử lý, và thực hiện phép xử lý đa kênh nữa bằng cách sử dụng cặp kênh thứ nhất được nhận biết bởi các tham số đa kênh MCH_PAR1 và sử dụng các tham số đa kênh MCH_PAR1, trong đó cặp kênh thứ nhất bao gồm ít nhất một kênh được xử lý.

Như được biểu thị trên Fig.10 bằng cách ví dụ, các tham số đa kênh MCH_PAR2 có thể biểu thị (hoặc báo hiệu) rằng cặp kênh được giải mã thứ hai bao gồm kênh được giải mã thứ nhất D1 và kênh được giải mã thứ hai D2. Do đó, bộ xử lý đa kênh 204 thực hiện phép xử lý đa kênh sử dụng cặp kênh được giải mã thứ hai bao gồm kênh được giải mã thứ nhất D1 và kênh được giải mã thứ hai D2 (được nhận biết bởi các tham số đa kênh MCH_PAR2) và sử dụng các tham số đa kênh MCH_PAR2, để thu được các kênh được xử lý P1* và P2*. Các tham số đa kênh MCH_PAR1 có thể biểu thị rằng cặp kênh được giải mã thứ nhất bao gồm kênh được xử lý thứ nhất P1* và kênh được giải mã thứ ba D3. Do đó, bộ xử lý đa kênh 204 thực hiện phép xử lý đa kênh nữa sử dụng cặp kênh được giải mã thứ nhất này bao gồm kênh được xử lý thứ nhất P1* và kênh được giải mã thứ ba D3 (được nhận biết bởi các tham số đa kênh MCH_PAR1) và sử dụng các tham số đa kênh MCH_PAR12, để thu được các kênh được xử lý P3* và P4*.

Hơn nữa, bộ xử lý đa kênh 204 có thể cung cấp kênh được xử lý thứ ba P3* như kênh thứ nhất CH1, kênh được xử lý thứ tư P4* như kênh thứ ba CH3 và kênh được xử lý thứ hai P2* như kênh thứ hai CH2.

Giả định rằng bộ giải mã 200 được thể hiện trên Fig.10 nhận tín hiệu đa kênh được mã hóa 107 từ bộ mã hóa 100 được thể hiện trên Fig.7, kênh được giải mã thứ nhất D1 của bộ giải mã 200 có thể tương đương với kênh được xử lý thứ ba P3 của bộ mã hóa 100, trong đó kênh được giải mã thứ hai D2 của bộ giải mã 200 có thể tương đương với kênh được xử lý thứ tư P4 của bộ mã hóa 100, và trong đó kênh được giải mã thứ ba D3 của bộ giải mã 200 có thể tương đương với kênh được xử lý thứ hai P2

của bộ mã hóa 100. Hơn nữa, kênh được xử lý thứ nhất P1* của bộ giải mã 200 có thể tương đương với kênh được xử lý thứ nhất P1 của bộ mã hóa 100.

Hơn nữa, tín hiệu đa kênh được mã hóa 107 có thể là tín hiệu chuỗi, trong đó các tham số đa kênh MCH_PAR2 được nhận, tại bộ giải mã 200, trước các tham số đa kênh MCH_PAR1. Trong trường hợp này, bộ xử lý đa kênh 204 có thể được tạo cấu hình để xử lý các kênh được giải mã theo thứ tự, trong đó các tham số đa kênh MCH_PAR1 và MCH_PAR2 được nhận bởi bộ giải mã. Trong ví dụ được thể hiện trên Fig.10, bộ giải mã nhận các tham số đa kênh MCH_PAR2 trước các tham số đa kênh MCH_PAR1, và do đó thực hiện phép xử lý đa kênh bằng cách sử dụng cặp kênh được giải mã thứ hai (bao gồm các kênh được giải mã thứ nhất và thứ hai D1 và D2) được nhận biết bởi các tham số đa kênh MCH_PAR2 trước khi thực hiện phép xử lý đa kênh bằng cách sử dụng cặp kênh được giải mã thứ nhất (bao gồm kênh được xử lý thứ nhất P1* và kênh được giải mã thứ ba D3) được nhận biết bởi tham số đa kênh MCH_PAR1.

Trên Fig.10, bộ xử lý đa kênh 204 thực hiện theo cách làm mẫu hai phép toán xử lý đa kênh. Với mục đích minh họa, các phép toán xử lý đa kênh được thực hiện bởi bộ xử lý đa kênh 204 được minh họa trên Fig.10 bởi các hộp xử lý 208 và 210. Các hộp xử lý 208 và 210 có thể được thực hiện trong phần cứng hoặc phần mềm. Các hộp xử lý 208 và 210 có thể là, ví dụ, các hộp âm lập thể, như được mô tả ở trên có tham khảo bộ mã hóa 100, như các bộ giải mã chung (hoặc các hộp âm lập thể phía bộ giải mã), phép dự báo dựa trên các bộ giải mã (hoặc các hộp âm lập thể phía bộ giải mã) hoặc KLT dựa trên các bộ giải mã luân phiên (hoặc các hộp âm lập thể phía bộ giải mã).

Ví dụ, bộ mã hóa 100 có thể sử dụng KTL dựa trên các bộ mã hóa luân phiên (hoặc các hộp âm lập thể phía bộ mã hóa). Trong trường hợp đó, bộ mã hóa 100 có thể suy ra các tham số đa kênh MCH_PAR1 và MCH_PAR2 sao cho các tham số đa kênh MCH_PAR1 và MCH_PAR2 bao gồm hai góc luân phiên. Các góc luân phiên có thể được mã hóa vi sai. Do đó, bộ xử lý đa kênh 204 của bộ giải mã 200 có thể bao gồm bộ giải mã vi sai để giải mã vi sai các góc luân phiên được mã hóa vi sai.

Thiết bị 200 có thể còn bao gồm giao diện đầu vào 212 được tạo cấu hình để nhận và xử lý tín hiệu đa kênh được mã hóa 107, để cung cấp các kênh được mã hóa

E1 đến E3 cho bộ giải mã kênh 202 và các tham số đa kênh MCH_PAR1 và MCH_PAR2 cho bộ xử lý đa kênh 204.

Như đã được đề cập, bộ chỉ thị duy trì (hoặc cờ hiệu cây duy trì) có thể được sử dụng để báo hiệu rằng không có cây mới nào được truyền, nhưng cây âm lập thể cuối cùng sẽ được sử dụng. Việc này có thể được sử dụng để tránh sự truyền đa phân của cùng cấu hình cây âm lập thể nếu các đặc tính tương quan kênh duy trì không đổi trong thời gian lâu hơn.

Do đó, khi tín hiệu đa kênh được mã hóa 107 bao gồm, đối với khung thứ nhất, các tham số đa kênh MCH_PAR1 và MCH_PAR2 và, đối với khung thứ hai, theo sau khung thứ nhất, bộ chỉ thị duy trì, bộ xử lý đa kênh 204 có thể được tạo cấu hình để thực hiện phép xử lý đa kênh hoặc phép xử lý đa kênh nửa trong khung thứ hai để cùng cặp kênh thứ hai hoặc cùng cặp kênh thứ nhất được sử dụng trong khung thứ nhất.

Phép xử lý đa kênh và phép xử lý đa kênh nửa có thể bao gồm phép xử lý âm lập thể bằng cách sử dụng tham số âm lập thể, trong đó đối với các băng thừa số tỉ lệ riêng lẻ hoặc các nhóm băng thừa số tỉ lệ của các kênh được giải mã D1 đến D3, tham số âm lập thể thứ nhất được chứa trong tham số đa kênh MCH_PAR1 và tham số âm lập thể thứ hai được chứa trong tham số đa kênh MCH_PAR2. Do đó, tham số âm lập thể thứ nhất và tham số âm lập thể thứ hai có thể là thuộc cùng loại, như các góc luân phiên hoặc các hệ số dự báo. Đương nhiên, tham số âm lập thể thứ nhất và tham số âm lập thể thứ hai có thể là thuộc các loại khác nhau. Ví dụ, tham số âm lập thể thứ nhất có thể là góc luân phiên, trong đó tham số âm lập thể thứ hai có thể là hệ số dự báo, hoặc vân vân.

Hơn nữa, các tham số đa kênh MCH_PAR1 và MCH_PAR2 có thể bao gồm mạng che xử lý đa kênh biểu thị các băng thừa số tỉ lệ nào được xử lý đa kênh và các băng thừa số tỉ lệ nào không được xử lý đa kênh. Do đó, bộ xử lý đa kênh 204 có thể được tạo cấu hình để không thực hiện phép xử lý đa kênh theo băng thừa số tỉ lệ được biểu thị bởi mạng che xử lý đa kênh.

Các tham số đa kênh MCH_PAR1 và MCH_PAR2, từng tham số có thể bao gồm ký hiệu (hoặc chỉ số) cặp kênh, trong đó bộ xử lý đa kênh 204 có thể được tạo cấu

hình để giải mã các ký hiệu (hoặc các chỉ số) cặp kênh bằng cách sử dụng quy tắc giải mã được định trước hoặc quy tắc giải mã được biểu thị trong tín hiệu đa kênh được mã hóa.

Ví dụ, các cặp kênh có thể được báo hiệu một cách hiệu quả bằng cách sử dụng chỉ số đơn nhất cho từng cặp, phụ thuộc vào tổng số lượng kênh, như được mô tả ở trên có tham khảo bộ mã hóa 100.

Hơn nữa, quy tắc giải mã có thể là quy tắc giải mã Huffman, trong đó bộ xử lý đa kênh 204 có thể được tạo cấu hình để thực hiện phép giải mã Huffman của các ký hiệu cặp kênh.

Tín hiệu đa kênh được mã hóa 107 có thể còn bao gồm bộ chỉ thị cho phép xử lý đa kênh biểu thị chỉ nhóm con của các kênh được giải mã, mà phép xử lý đa kênh được cho phép và biểu thị ít nhất một kênh được giải mã mà phép xử lý đa kênh không được cho phép. Do đó, bộ xử lý đa kênh 204 có thể được tạo cấu hình để không thực hiện bất kỳ phép xử lý đa kênh nào cho ít nhất một kênh được giải mã, mà phép xử lý đa kênh không được cho phép như được biểu thị bởi bộ chỉ thị cho phép xử lý đa kênh.

Ví dụ, khi tín hiệu đa kênh là tín hiệu kênh 5.1, bộ chỉ thị cho phép xử lý đa kênh có thể biểu thị rằng phép xử lý đa kênh chỉ được cho phép đối với 5 kênh, tức là các kênh bên phải R, bên trái L, bao quanh bên phải Rs, bao quanh bên trái LS và trung tâm C, trong đó phép xử lý đa kênh không được cho phép đối với kênh LFE.

Đối với quy trình giải mã (giải mã các chỉ số cặp kênh), mã c sau đây có thể được sử dụng. Do đó, đối với tất cả các cặp kênh, số lượng của các kênh với phép xử lý KLT hoạt động (nChannels) cũng như số lượng các cặp kênh (numPairs) của khung hiện thời được cần đến.

```
maxNumPairIdx = nChannels*(nChannels-1)/2 - 1;
```

```
numBits = floor(log2(maxNumPairIdx)+1);
```

```
pairCounter = 0;
```

```
for (chan1=1; chan1 < nChannels; chan1++) {
```

```
    for (chan0=0; chan0 < chan1; chan0++) {
```

```
        if (pairCounter == pairIdx) {
```

```

    channelPair[0] = chan0;
    channelPair[1] = chan1;
    return;
}
else
    pairCounter++;
}
}
}

```

Để giải mã các hệ số cho các góc không theo băng, mã c sau đây có thể được sử dụng.

```

for(pair=0; pair<numPairs; pair++) {
    mctBandsPerWindow = numMaskBands[pair]/windowsPerFrame;
    if(delta_code_time[pair] > 0) {
        lastVal = alpha_prev_fullband[pair];
    } else {
        lastVal = DEFAULT_ALPHA;
    }
    newAlpha = lastVal + dpcm_alpha[pair][0];
    if(newAlpha >= 64) {
        newAlpha -= 64;
    }
    for (band=0; band < numMaskBands; band++){
        /* set all angles to fullband angle */
        pairAlpha[pair][band] = newAlpha;
        /* set previous angles according to mctMask */
        if(mctMask[pair][band] > 0) {
            alpha_prev_frame[pair][band%mctBandsPerWindow] = newAlpha;

```

```

    }
    else {
        alpha_prev_frame[pair][band%mctBandsPerWindow] = DEFAULT_ALPHA;
    }
}
alpha_prev_fullband[pair] = newAlpha;
for(band=bandsPerWindow ; band<MAX_NUM_MC_BANDS; band++) {
    alpha_prev_frame[pair][band] = DEFAULT_ALPHA;
}
}

```

Để giải mã các hệ số cho các góc KTL không theo băng, mã c sau đây có thể được sử dụng.

```

for(pair=0; pair<numPairs; pair++) {
    mctBandsPerWindow = numMaskBands[pair]/windowsPerFrame;
    for(band=0; band<numMaskBands[pair]; band++) {
        if(delta_code_time[pair] > 0) {
            lastVal = alpha_prev_frame[pair][band%mctBandsPerWindow];
        }
        else {
            if ((band % mctBandsPerWindow) == 0) {
                lastVal = DEFAULT_ALPHA;
            }
        }
        if (msMask[pair][band] > 0 ) {

            newAlpha = lastVal + dpcm_alpha[pair][band];
            if(newAlpha >= 64) {
                newAlpha -= 64;
            }
        }
    }
}

```

```

    }
    pairAlpha[pair][band] = newAlpha;
    alpha_prev_frame[pair][band%mctBandsPerWindow] = newAlpha;
    lastVal = newAlpha;
}
else {
    alpha_prev_frame[pair][band%mctBandsPerWindow] = DEFAULT_ALPHA; /* -
45° */
}
/* reset fullband angle */
alpha_prev_fullband[pair] = DEFAULT_ALPHA;
}
for(band=bandsPerWindow ; band<MAX_NUM_MC_BANDS; band++) {
    alpha_prev_frame[pair][band] = DEFAULT_ALPHA;
}
}

```

Để tránh các chênh lệch dấu phẩy động của hàm lượng giác trên các nền tảng khác nhau, các bảng dò tìm để chuyển đổi các chỉ số góc trực tiếp thành sin/cos sẽ được sử dụng:

```

tabIndexToSinAlpha[64] = {
-1.000000f,-0.998795f,-0.995185f,-0.989177f,-0.980785f,-0.970031f,-0.956940f,-
0.941544f,
-0.923880f,-0.903989f,-0.881921f,-0.857729f,-0.831470f,-0.803208f,-0.773010f,-
0.740951f,
-0.707107f,-0.671559f,-0.634393f,-0.595699f,-0.555570f,-0.514103f,-0.471397f,-
0.427555f,
-0.382683f,-0.336890f,-0.290285f,-0.242980f,-0.195090f,-0.146730f,-0.098017f,-
0.049068f,

```

```

0.000000f, 0.049068f, 0.098017f, 0.146730f, 0.195090f, 0.242980f, 0.290285f,
0.336890f,
0.382683f, 0.427555f, 0.471397f, 0.514103f, 0.555570f, 0.595699f, 0.634393f,
0.671559f,
0.707107f, 0.740951f, 0.773010f, 0.803208f, 0.831470f, 0.857729f, 0.881921f,
0.903989f,
0.923880f, 0.941544f, 0.956940f, 0.970031f, 0.980785f, 0.989177f, 0.995185f,
0.998795f
};
tabIndexToCosAlpha[64] = {
0.000000f, 0.049068f, 0.098017f, 0.146730f, 0.195090f, 0.242980f, 0.290285f,
0.336890f,
0.382683f, 0.427555f, 0.471397f, 0.514103f, 0.555570f, 0.595699f, 0.634393f,
0.671559f,
0.707107f, 0.740951f, 0.773010f, 0.803208f, 0.831470f, 0.857729f, 0.881921f,
0.903989f,
0.923880f, 0.941544f, 0.956940f, 0.970031f, 0.980785f, 0.989177f, 0.995185f,
0.998795f,
1.000000f, 0.998795f, 0.995185f, 0.989177f, 0.980785f, 0.970031f, 0.956940f,
0.941544f,
0.923880f, 0.903989f, 0.881921f, 0.857729f, 0.831470f, 0.803208f, 0.773010f,
0.740951f,
0.707107f, 0.671559f, 0.634393f, 0.595699f, 0.555570f, 0.514103f, 0.471397f,
0.427555f,
0.382683f, 0.336890f, 0.290285f, 0.242980f, 0.195090f, 0.146730f, 0.098017f,
0.049068f
};

```

Để giải mã phép mã hóa đa kênh, mã c sau đây có thể được sử dụng cho phương pháp tiếp cận dựa trên sự luân phiên KLT.

```

decode_mct_rotation()
{
    for (pair=0; pair < self->numPairs; pair++) {
        mctBandOffset = 0;
        /* inverse MCT rotation */
        for (win = 0, group = 0; group < num_window_groups; group++) {
            for (groupwin = 0; groupwin < window_group_length[group]; groupwin++, win++)
            {
                *dmx = spectral_data[ch1][win];
                *res = spectral_data[ch2][win];
                apply_mct_rotation_wrapper(self,dmx,res,&alphaSfb[mctBandOffset],
                                           &mctMask[mctBandOffset],mctBandsPerWindow, alpha,
                                           totalSfb,pair,nSamples);
            }
            mctBandOffset += mctBandsPerWindow;
        }
    }
}

```

Để xử lý theo bảng, mã c sau đây có thể được sử dụng.

```

apply_mct_rotation_wrapper(self, *dmx, *res, *alphaSfb, *mctMask,
mctBandsPerWindow,
                        alpha, totalSfb, pair, nSamples)
{
    sfb = 0;
    if (self->MCCSignalingType == 0) {
    }
    else if (self->MCCSignalingType == 1) {
        /* apply fullband box */

```

```

if (!self->bHasBandwiseAngles[pair] && !self->bHasMctMask[pair]) {
    apply_mct_rotation(dmx, res, alphaSfb[0], nSamples);
}
else {
    /* apply bandwise processing */
    for (i = 0; i < mctBandsPerWindow; i++) {
        if (mctMask[i] == 1) {
            startLine = swb_offset [sfb];
            stopLine = (sfb+2<totalSfb)? swb_offset [sfb+2] : swb_offset [sfb+1];
            nSamples = stopLine-startLine;
            apply_mct_rotation(&dmx[startLine], &res[startLine], alphaSfb[i], nSamples);
        }
        sfb += 2;
        /* break condition */
        if (sfb >= totalSfb) {
            break;
        }
    }
}
else if (self->MCCSignalingType == 2) {
}
else if (self->MCCSignalingType == 3) {
    apply_mct_rotation(dmx, res, alpha, nSamples);
}
}

```

Đối với ứng dụng của phép luân phiên KLT, mã c sau đây có thể được sử dụng.

```

apply_mct_rotation(*dmx, *res, alpha, nSamples)
{
    for (n=0;n<nSamples;n++) {
        L = dmx[n] * tabIndexToCosAlpha [alphaIdx] - res[n] * tabIndexToSinAlpha
[alphaIdx];
        R = dmx[n] * tabIndexToSinAlpha [alphaIdx] + res[n] * tabIndexToCosAlpha
[alphaIdx];
        dmx[n] = L;
        res[n] = R;
    }
}

```

Fig.12 thể hiện lưu đồ của phương pháp 400 để giải mã tín hiệu đa kênh được mã hóa có các kênh được mã hóa và ít nhất hai tham số đa kênh MCH_PAR1 và MCH_PAR2. Phương pháp 400 bao gồm bước 402 là giải mã các kênh được mã hóa để thu được các kênh được giải mã; và bước 404 là thực hiện phép xử lý đa kênh bằng cách sử dụng cặp kênh được giải mã thứ hai được nhận biết bởi các tham số đa kênh MCH_PAR2 và sử dụng các tham số đa kênh MCH_PAR2 để thu được các kênh được xử lý, và thực hiện phép xử lý đa kênh nữa bằng cách sử dụng cặp kênh thứ nhất được nhận biết bởi các tham số đa kênh MCH_PAR1 và sử dụng các tham số đa kênh MCH_PAR1, trong đó cặp kênh thứ nhất bao gồm ít nhất một kênh được xử lý.

Trong phần sau đây, việc điền đầy âm lập thể trong phép mã hóa đa kênh theo các phương án được giải thích:

Như đã được tóm tắt, tác động không mong muốn của phép lượng tử hóa quang phổ có thể là ở chỗ phép lượng tử hóa có thể có khả năng dẫn đến các lỗi quang phổ. Ví dụ, tất cả các giá trị quang phổ trong băng tần cụ thể có thể được thiết lập bằng không trên phía bộ mã hóa như kết quả của phép lượng tử hóa. Ví dụ, giá trị chính xác của các vạch quang phổ trước khi lượng tử hóa có thể là tương đối thấp và phép lượng tử hóa sau đó có thể dẫn đến trạng thái, trong đó các giá trị quang phổ của tất cả các vạch phổ, ví dụ, trong băng tần cụ thể đã được thiết lập bằng không. Trên phía bộ giải mã, khi giải mã, điều này có thể dẫn đến các lỗi quang phổ không mong muốn.

Công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) trong MPEG-H cho phép thích ứng với sự phụ thuộc liên kênh khác nhau nhưng, do việc sử dụng các phân tử kênh đơn trong các cấu hình hoạt động điển hình, không cho phép điền đầy âm lập thể.

Như có thể được thấy trên Fig.14, công cụ mã hóa đa kênh tổ hợp ba hoặc nhiều kênh mà được mã hóa theo kiểu phân cấp. Tuy nhiên, cách mà công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) tổ hợp các kênh khác nhau khi mã hóa biến thiên từ khung sang khung phụ thuộc vào các đặc tính tín hiệu hiện thời của các kênh.

Ví dụ, trên Fig.14, kịch bản (a), để tạo ra khung tín hiệu âm thanh được mã hóa thứ nhất, công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) có thể tổ hợp kênh thứ nhất CH1 và kênh thứ hai CH2 để thu được kênh tổ hợp thứ nhất (kênh được xử lý) P1 và kênh tổ hợp thứ hai P2. Sau đó, công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) có thể tổ hợp kênh tổ hợp thứ nhất P1 và kênh thứ ba CH3 để thu được kênh tổ hợp thứ ba P3 và kênh tổ hợp thứ tư P4. Công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) sau đó có thể mã hóa kênh tổ hợp thứ hai P2, kênh tổ hợp thứ ba P3 và kênh tổ hợp thứ tư P4 để tạo ra khung thứ nhất.

Sau đó, ví dụ, trên Fig.14, kịch bản (b), để tạo ra khung tín hiệu âm thanh được mã hóa thứ hai (theo thời gian) kế tiếp khung tín hiệu âm thanh được mã hóa thứ nhất, công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) có thể tổ hợp kênh thứ nhất CH1' và kênh thứ ba CH3' để thu được kênh tổ hợp thứ nhất P1' và kênh tổ hợp thứ hai P2'. Sau đó, công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) có thể tổ hợp kênh tổ hợp thứ nhất P1' và kênh thứ hai CH2' để thu được kênh tổ hợp thứ ba P3' và kênh tổ hợp thứ tư P4'. Công cụ mã hóa đa kênh (Multichannel Coding Tool - MCT) sau đó có thể mã hóa kênh tổ hợp thứ hai P2', kênh tổ hợp thứ ba P3' và kênh tổ hợp thứ tư P4' để tạo ra khung thứ hai.

Như có thể được thấy từ Fig.14, cách mà kênh tổ hợp thứ hai, thứ ba và thứ tư của khung thứ nhất đã được tạo ra theo kịch bản của Fig.14 (a) khác đáng kể với cách mà lần lượt các kênh tổ hợp thứ hai, thứ ba và thứ tư của khung thứ hai đã được tạo ra theo kịch bản của Fig.14 (b), vì các tổ hợp khác nhau của các kênh đã được sử dụng để tạo ra các kênh tổ hợp tương ứng P2, P3 và P4 và P2', P3', P4', một cách lần lượt.

Không kể đến những điều khác, các phương án của sáng chế được dựa trên các phát hiện sau đây:

Như có thể thấy trên Fig.7 và Fig.14, các kênh tổ hợp P3, P4 và P2 (hoặc P2', P3' và P4' trong kịch bản (b) của Fig.14) được nạp vào bộ mã hóa kênh 104. Không kể đến những điều khác, bộ mã hóa kênh 104 có thể, ví dụ, tiến hành phép lượng tử hóa, sao cho các giá trị quang phổ của các kênh P2, P3 và P4 có thể được thiết lập bằng không do phép lượng tử hóa. Các mẫu quang phổ lân cận theo quang phổ có thể được mã hóa như băng quang phổ, trong đó từng băng quang phổ có thể bao gồm số lượng các mẫu quang phổ.

Số lượng các mẫu quang phổ của băng tần có thể là khác nhau đối với các băng tần khác nhau. Ví dụ, các băng tần trong phạm vi tần số thấp hơn có thể, ví dụ, gồm ít mẫu quang phổ hơn, (ví dụ, 4 mẫu quang phổ) so với các băng tần trong phạm vi tần số cao hơn, mà có thể, ví dụ, bao gồm 16 mẫu tần số. Ví dụ, các băng tới hạn tỉ lệ Bark có thể xác định các băng tần được sử dụng.

Trường hợp không mong muốn cụ thể có thể xảy ra, khi tất cả các mẫu quang phổ của băng tần đã được thiết lập bằng không sau khi lượng tử hóa. Nếu trường hợp này có thể xảy ra, theo sáng chế, nên tiến hành điền đầy âm lập thể. Hơn nữa, sáng chế dựa trên việc tìm ra rằng ít nhất là không chỉ nhiều âm ngẫu nhiên (-giả) cần được tạo ra.

Thay vì hoặc ngoài việc bổ sung nhiều âm ngẫu nhiên (-giả), theo phương án của sáng chế, nếu, ví dụ, trên Fig.14, kịch bản (b), tất cả giá trị quang phổ của băng tần của kênh P4' đã được thiết lập đến không, kênh tổ hợp mà đã được tạo ra theo cùng cách hoặc theo cách tương tự như kênh P3' sẽ là cơ sở rất phù hợp để tạo ra nhiều âm để điền đầy trong các băng tần mà đã được lượng tử hóa bằng không.

Tuy nhiên, theo các phương án của sáng chế, tốt hơn là không sử dụng các giá trị quang phổ của kênh tổ hợp P3' của khung hiện thời/của thời điểm hiện thời như cơ sở để điền đầy băng tần của kênh tổ hợp P4', mà chỉ bao gồm các giá trị quang phổ là không, vì cả kênh tổ hợp P3' cũng như kênh tổ hợp P4' đã được tạo ra dựa trên kênh P1' và P2', và do đó, sử dụng kênh tổ hợp P3' của thời điểm hiện thời sẽ dẫn đến chỉ phép ghép nhẵn.

Ví dụ, nếu $P3'$ là kênh giữa của $P1'$ và $P2'$ (ví dụ, $P3' = 0,5 * (P1' + P2')$) và $P4'$ nếu là kênh bên của $P1'$ và $P2'$ (ví dụ, $P4' = 0,5 * (P1' - P2')$), hơn là đưa, ví dụ các giá trị quang phổ yếu dần của $P3'$, vào trong băng tần của $P4'$ sẽ chỉ dẫn đến sự ghép nhẵn.

Thay vào đó sử dụng các kênh của thời điểm trước để tạo ra các giá trị quang phổ cho việc điền đầy các lỗ quang phổ trong kênh tổ hợp $P4'$ hiện thời sẽ được ưu tiên. Theo các phát hiện trong sáng chế, các tổ hợp kênh của khung trước mà tương ứng với kênh tổ hợp $P3'$ của khung hiện thời sẽ là cơ sở mong muốn để tạo ra các mẫu quang phổ để điền đầy các khoảng trống quang phổ của $P4'$.

Tuy nhiên, kênh tổ hợp $P3$ đã được tạo ra trong kịch bản của Fig.10 (a) cho khung trước không tương ứng với kênh tổ hợp $P3'$ của khung hiện thời, vì kênh tổ hợp $P3$ của khung trước đã được tạo ra theo cách khác kênh tổ hợp $P3'$ của khung hiện thời.

Theo các phát hiện trong các phương án của sáng chế, sự gần đúng của kênh tổ hợp $P3'$ sẽ được tạo ra dựa trên các kênh được khôi phục của khung trước trên phía bộ giải mã.

Fig.10 (a) minh họa kịch bản bộ mã hóa trong đó các kênh CH1, CH2, CH3 được mã hóa cho khung trước bằng cách tạo ra E1, E2 và E3. Bộ giải mã nhận các kênh E1, E2 và E3 và khôi phục các kênh CH1, CH2 và CH3 mà đã được mã hóa. Một số tổn hao mã hóa đã xảy ra, nhưng các kênh được tạo ra CH1*, CH2* và CH3* mà xấp xỉ CH1, CH2 và CH3 vẫn sẽ khá giống với các kênh ban đầu CH1, CH2 và CH3, sao cho $CH1* \approx CH1$; $CH2* \approx CH2$ và $CH3* \approx CH3$. Theo các phương án, bộ giải mã duy trì các kênh CH1*, CH2* và CH3*, được tạo ra cho khung trước trong vùng đệm để sử dụng chúng để điền đầy nhiều âm trong khung hiện thời.

Fig.1a minh họa thiết bị 201 để giải mã theo phương án của sáng chế sẽ được mô tả chi tiết hơn trong phần sau đây:

Thiết bị 201 của Fig.1a được làm thích ứng để giải mã tín hiệu đa kênh được mã hóa trước của khung trước để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và được tạo cấu hình để giải mã thiết bị đa kênh được mã hóa hiện thời 107 của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời.

Thiết bị bao gồm giao diện 212, bộ giải mã kênh 202, bộ xử lý đa kênh 204 để tạo ra ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời CH1, CH2, CH3, và môđun điền đầy nhiễu âm 220.

Giao diện 212 được làm thích ứng để nhận tín hiệu đa kênh được mã hóa hiện thời 107, và để nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất MCH_PAR2.

Bộ giải mã kênh 202 được làm thích ứng để giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu được tập hợp của ba hoặc nhiều hơn ba kênh được giải mã D1, D2, D3 của khung hiện thời.

Bộ xử lý đa kênh 204 được làm thích ứng để chọn cặp được chọn thứ nhất gồm hai kênh được giải mã D1, D2 từ tập hợp của ba hoặc nhiều hơn ba kênh được giải mã D1, D2, D3 phụ thuộc vào các tham số đa kênh thứ nhất MCH_PAR2.

Như ví dụ, điều này được minh họa trên Fig.1a bởi hai kênh D1, D2 mà được nạp vào hộp xử lý 208 (tùy chọn).

Hơn nữa, bộ xử lý đa kênh 204 được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý P1*, P2* dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã D1, D2 để thu được tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã D3, P1*, P2*.

Trong ví dụ, trong đó hai kênh D1 và D2 được nạp vào hộp 208 (tùy chọn), hai kênh được xử lý P1* và P2* được tạo ra từ hai kênh được chọn D1 và D2. Thiết lập được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã sau đó bao gồm kênh D3 mà đã được để lại và không được biến đổi và còn bao gồm P1* và P2* mà đã được tạo ra từ D1 và D2.

Trước khi bộ xử lý đa kênh 204 tạo ra cặp thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý P1*, P2* dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã D1, D2, môđun điền đầy nhiễu âm 220 được làm thích ứng để nhận biết ít nhất một trong số hai kênh của cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã D1, D2, một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và tạo ra kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải cả ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và để điền đầy các

vạch phổ của một hoặc nhiều băng tần, mà trong đó các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra sử dụng các vạch phổ của kênh trộn, trong đó môđun điền đầy nhiều âm 220 được làm thích ứng để lựa chọn hai hoặc nhiều hơn hai kênh đầu ra âm thanh trước mà được sử dụng để tạo ra kênh trộn từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào thông tin phụ.

Do đó, môđun điền đầy nhiều âm 220 phân tích, liệu có các băng tần mà chỉ có các giá trị quang phổ bằng không hay không, và hơn nữa có điền đầy các băng tần trống được tìm thấy với nhiều âm được tạo ra hay không. Ví dụ, băng tần có thể, ví dụ có 4 hoặc 8 hoặc 16 vạch phổ và khi tất cả các vạch phổ của băng tần đã được lượng tử hóa bằng không thì môđun điền đầy nhiều âm 220 điền đầy nhiều âm được tạo ra.

Khía cạnh cụ thể của các phương án là có thể được sử dụng bởi môđun điền đầy nhiều âm 220 mà chỉ ra cách để tạo ra và điền đầy nhiều âm được đề cập đến như điền đầy âm lập thể.

Trong các phương án của Fig.1a, môđun điền đầy nhiều âm 220 tương tác với bộ xử lý đa kênh 204. Ví dụ, trong phương án, khi môđun điền đầy nhiều âm muốn xử lý hai kênh, ví dụ, bởi hộp xử lý, cần nạp các kênh này vào môđun điền đầy nhiều âm 220, và môđun điền đầy nhiều âm 220 kiểm tra, liệu các băng tần đã được lượng tử hóa bằng không hay chưa, và có điền đầy các băng tần này hay không, nếu đã phát hiện.

Trong các phương án khác được minh họa bởi Fig.1b, môđun điền đầy nhiều âm 220 tương tác với bộ giải mã kênh 202. Ví dụ, khi bộ giải mã kênh giải mã tín hiệu đa kênh được mã hóa để thu được ba hoặc nhiều hơn ba kênh được giải mã D1, D2 và D3, môđun điền đầy nhiều âm có thể, ví dụ, kiểm tra liệu các băng tần đã được lượng tử hóa đến không hay chưa và, ví dụ, có điền đầy các băng tần này hay không, nếu đã phát hiện. Trong phương án này, bộ xử lý đa kênh 204 có thể chắc chắn rằng tất cả các lỗ quang phổ đã được đóng trước bằng cách điền đầy nhiều âm.

Trong các phương án nữa (không được thể hiện), môđun điền đầy nhiều âm 220 có thể tương tác với cả bộ giải mã kênh và bộ xử lý đa kênh. Ví dụ, khi bộ giải mã kênh 202 tạo ra các kênh được giải mã D1, D2 và D3, môđun điền đầy nhiều âm 220 có thể đã kiểm tra liệu các băng tần đã được lượng tử hóa bằng không hay chưa, chỉ

sau khi bộ giải mã kênh 202 đã tạo ra chúng, nhưng có thể chỉ tạo ra nhiều âm và điền đầy các băng tần tương ứng, khi bộ xử lý đa kênh 204 xử lý thực sự các kênh này.

Ví dụ, nhiều âm ngẫu nhiên, phép toán ít tính toán có thể được chèn vào bất kỳ băng tần nào đã được lượng tử hóa bằng không, nhưng môđun điền đầy nhiều âm có thể điền đầy nhiều âm mà đã được tạo ra từ các kênh đầu ra âm thanh đã được tạo ra trước đó chỉ nếu chúng được xử lý thực sự bởi bộ xử lý đa kênh 204. Tuy nhiên, trong các phương án này, trước khi chèn nhiều âm ngẫu nhiên, việc phát hiện liệu có các lỗ quang phổ hay không được thực hiện trước khi chèn nhiều âm ngẫu nhiên, và thông tin cần được giữ trong bộ nhớ, vì sau khi chèn nhiều âm ngẫu nhiên, các băng tần tương ứng có các giá trị quang phổ khác không, vì nhiều âm ngẫu nhiên đã được chèn vào.

Trong các phương án, nhiều âm ngẫu nhiên được chèn vào các băng tần mà đã được lượng tử hóa bằng không ngoài nhiều âm được tạo ra dựa trên các tín hiệu đầu ra âm thanh trước.

Trong một số phương án, giao diện 212 có thể, ví dụ, được làm thích ứng để nhận tín hiệu đa kênh được mã hóa hiện thời 107, và để nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất MCH_PAR2 và các tham số đa kênh thứ hai MCH_PAR1.

Bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để lựa chọn cặp được lựa chọn thứ hai gồm hai kênh được giải mã P1*, D3 từ tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã D3, P1*, P2* phụ thuộc vào các tham số đa kênh thứ hai MCH_PAR1, trong đó ít nhất một kênh P1* của cặp được lựa chọn thứ hai gồm hai kênh được giải mã (P1*, D3) là một kênh thuộc cặp thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý P1*, P2*, và

Bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để tạo ra nhóm thứ hai gồm hai hoặc nhiều hơn hai kênh được xử lý P3*, P4* dựa trên cặp được lựa chọn thứ hai đã nêu gồm hai kênh được giải mã D1, D2 để cập nhật thêm tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã.

Ví dụ cho phương án này có thể được thấy trên các hình vẽ Fig.1a và Fig.1b, trong đó hộp xử lý 210 (tùy chọn) nhận kênh D3 và kênh được xử lý P1* và xử lý chúng để thu được các kênh được xử lý P3* và P4* sao cho việc thiết lập được cập

nhập thêm của ba kênh được giải mã bao gồm P2*, mà đã không được biến đổi bởi hộp xử lý 210, và P3* và P4* được tạo ra.

Các hộp xử lý 208 và 210 đã được đánh dấu trên Fig.1a và Fig.1 là tùy chọn. Điều này thể hiện là dù có thể sử dụng các hộp xử lý 208 và 210 để thực hiện bộ xử lý đa kênh 204, nhiều khả năng khác vẫn tồn tại, cách để thực hiện chính xác bộ xử lý đa kênh 204. Ví dụ, thay vì sử dụng hộp xử lý 208, 210 khác nhau cho mỗi việc xử lý khác nhau của hai (hoặc nhiều) kênh, hộp xử lý giống nhau có thể được dùng lại, hoặc bộ xử lý đa kênh 204 có thể thực hiện việc xử lý hai kênh mà không sử dụng các hộp xử lý 208, 210 (như các bộ phận con của bộ xử lý đa kênh 204) nữa.

Theo phương án nữa, bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý P1*, P2* bằng cách tạo ra nhóm thứ nhất gồm chính xác hai kênh được xử lý P1*, P2* dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã D1, D2. Bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để thay thế cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã D1, D2 trong tập hợp của ba hoặc nhiều hơn ba kênh được giải mã D1, D2, D3 bởi nhóm thứ nhất gồm chính xác hai kênh được xử lý P1*, P2* để thu được tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã D3, P1*, P2*. Bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để tạo ra nhóm thứ hai gồm hai hoặc nhiều hơn hai kênh được xử lý P3*, P4* bằng cách tạo ra nhóm thứ hai gồm chính xác hai kênh được xử lý P3*, P4* dựa trên cặp được lựa chọn thứ hai đã nêu gồm hai kênh được giải mã P1*, D3. Hơn nữa, bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để thay thế cặp được lựa chọn thứ hai đã nêu gồm hai kênh được giải mã P1*, D3 trong tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã D3, P1*, P2* bởi nhóm thứ hai gồm chính xác hai kênh được xử lý P3*, P4* để cập nhật thêm tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã.

Trong các phương án này, từ hai kênh được lựa chọn (ví dụ, hai kênh đầu vào của hộp xử lý 208 hoặc 210), chính xác hai kênh được xử lý được tạo ra và chính xác hai kênh được xử lý này thay thế các kênh được chọn trong tập hợp của ba hoặc nhiều hơn ba kênh được giải mã. Ví dụ, hộp xử lý 208 của bộ xử lý đa kênh 204 thay thế các kênh được chọn D1 và D2 bởi P1* và P2*.

Tuy nhiên, trong các phương án khác, phép trộn tăng có thể diễn ra trong thiết bị 201 để giải mã, và nhiều hơn hai kênh được xử lý có thể được tạo ra từ hai kênh được lựa chọn, hoặc không phải tất cả các kênh được lựa chọn có thể bị xóa khỏi tập hợp được cập nhật của các kênh được giải mã.

Vấn đề nữa là cách để tạo ra kênh trộn mà được sử dụng để tạo ra nhiều âm được tạo ra bởi môđun điền đầy nhiều âm 220.

Theo một số phương án, môđun điền đầy nhiều âm 220 có thể, ví dụ, được làm thích ứng để tạo ra kênh trộn bằng cách sử dụng chính xác hai trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước như hai hoặc nhiều hơn hai trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước; trong đó môđun điền đầy nhiều âm 220 có thể, ví dụ, được làm thích ứng để lựa chọn chính xác hai kênh đầu ra âm thanh trước từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào thông tin phụ.

Sử dụng chỉ hai trong số ba hoặc nhiều hơn ba kênh đầu ra trước giúp giảm độ phức tạp tính toán của việc tính toán kênh trộn.

Tuy nhiên, trong các phương án khác, nhiều hơn hai kênh trong số các kênh đầu ra âm thanh trước được sử dụng để tạo ra kênh trộn, nhưng số lượng của các kênh đầu ra âm thanh trước mà được tính đến là nhỏ hơn tổng số lượng của ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước.

Trong các phương án, khi chỉ hai kênh đầu ra âm thanh trước được tính đến, kênh trộn có thể, ví dụ, được tính toán như sau:

Trong một phương án, môđun điền đầy nhiều âm 220 được làm thích ứng để tạo ra kênh trộn bằng cách sử dụng chính xác hai kênh đầu ra âm thanh trước dựa trên công thức

$$D_{ch} = (\hat{O}_1 + \hat{O}_2) \cdot d \text{ hoặc dựa trên công thức}$$

$$D_{ch} = (\hat{O}_1 - \hat{O}_2) \cdot d$$

trong đó D_{ch} là kênh trộn; trong đó $\hat{O}_1$ là kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước; trong đó $\hat{O}_2$ là kênh thứ hai trong số chính xác hai kênh đầu ra âm thanh trước, khác kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước, và trong đó d là vô hướng dương, thực.

Trong trạng thái điển hình, kênh giữa $D_{ch} = (\hat{O}_1 + \hat{O}_2) \cdot d$ có thể là kênh trộn phù hợp. Phương thức này tính toán kênh trộn như kênh giữa của hai kênh đầu ra âm thanh trước mà được tính đến.

Tuy nhiên, trong một số kịch bản, kênh trộn gần với không có thể xuất hiện khi áp dụng $D_{ch} = (\hat{O}_1 + \hat{O}_2) \cdot d$, ví dụ khi $\hat{O}_1 \approx -\hat{O}_2$. Sau đó, có thể, ví dụ, tốt hơn là sử dụng $D_{ch} = (\hat{O}_1 - \hat{O}_2) \cdot d$ như tín hiệu trộn. Do đó, sau đó kênh bên (đối với các kênh đầu vào dịch pha) được sử dụng.

Theo một phương thức khác, môđun điều chỉnh nhiều âm 220 được làm thích ứng để tạo ra kênh trộn bằng cách sử dụng chính xác hai kênh đầu ra âm thanh trước dựa trên công thức

$$\hat{I}_{ch} = (\cos \alpha \cdot \hat{O}_1 + \sin \alpha \cdot \hat{O}_2) \cdot d \text{ hoặc dựa trên công thức}$$

$$\hat{I}_{ch} = (-\sin \alpha \cdot \hat{O}_1 + \cos \alpha \cdot \hat{O}_2) \cdot d$$

trong đó $\hat{I}_{ch}$ là kênh trộn, trong đó $\hat{O}_1$ là kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước, trong đó $\hat{O}_2$ là kênh thứ hai trong số chính xác hai kênh đầu ra âm thanh trước, khác kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước, và trong đó α là góc luân phiên.

Phương thức này tính toán kênh trộn bằng cách tiến hành phép luân phiên của hai kênh đầu ra âm thanh trước mà được tính đến.

Góc luân phiên α có thể, ví dụ, nằm trong khoảng: $-90^\circ < \alpha < 90^\circ$.

Trong một phương án, góc luân phiên có thể, ví dụ, nằm trong khoảng: $30^\circ < \alpha < 60^\circ$.

Ngoài ra, trong các trạng thái điển hình, kênh $\hat{I}_{ch} = (\cos \alpha \cdot \hat{O}_1 + \sin \alpha \cdot \hat{O}_2) \cdot d$ có thể là kênh trộn phù hợp. Phương thức này tính toán kênh trộn như kênh giữa của hai kênh đầu ra âm thanh trước mà được tính đến.

Tuy nhiên, trong một số kịch bản, kênh trộn gần với không có thể xuất hiện khi áp dụng $\hat{I}_{ch} = (\cos \alpha \cdot \hat{O}_1 + \sin \alpha \cdot \hat{O}_2) \cdot d$, ví dụ khi $\cos \alpha \cdot \hat{O}_1 \approx -\sin \alpha \cdot \hat{O}_2$. Sau đó, có thể, ví dụ, tốt hơn là sử dụng $\hat{I}_{ch} = (-\sin \alpha \cdot \hat{O}_1 + \cos \alpha \cdot \hat{O}_2) \cdot d$ như tín hiệu trộn.

Theo phương án cụ thể, thông tin phụ có thể, ví dụ, là thông tin phụ hiện thời được gán cho khung hiện thời, trong đó giao diện 212 có thể, ví dụ, được làm thích ứng để nhận thông tin phụ trước mà được gán cho khung trước, trong đó thông tin phụ trước bao gồm góc trước; trong đó giao diện 212 có thể, ví dụ, được làm thích ứng để nhận thông tin phụ hiện thời bao gồm góc hiện thời, và trong đó môđun điền đầy nhiều âm 220 có thể, ví dụ, được làm thích ứng để sử dụng góc hiện thời của thông tin phụ hiện thời như góc luân phiên α , và được làm thích ứng để không sử dụng góc trước của thông tin phụ trước như góc luân phiên α .

Do đó, trong phương án này, ngay cả nếu kênh trộn được tính toán dựa trên kênh đầu ra âm thanh trước thì vẫn có góc hiện thời mà được truyền trong thông tin phụ được sử dụng như góc luân phiên và không phải góc luân phiên được nhận trước, dù kênh trộn được tính dựa trên các kênh đầu ra âm thanh trước mà đã được tạo ra dựa trên khung trước.

Khía cạnh khác của một số phương án của sáng chế đề liên quan đến các thừa số tỉ lệ.

Các băng tần có thể, ví dụ, là các băng thừa số tỉ lệ.

Theo một số phương án, trước khi bộ xử lý đa kênh 204 tạo ra cặp thứ nhất của hai hoặc nhiều hơn hai kênh được xử lý P1*, P2* dựa trên cặp được chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2), môđun điền đầy nhiều âm (220) có thể, ví dụ, được làm thích ứng để nhận biết đối với ít nhất một trong số hai kênh của cặp được chọn thứ nhất đã nêu gồm hai kênh được giải mã D1, D2, một hoặc nhiều băng thừa số tỉ lệ là một hoặc nhiều băng tần, mà trong đó tất cả cách vạch phổ được lượng tử hóa bằng không, và có thể, ví dụ, được làm thích ứng để tạo ra kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải cả ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước đã nêu, và để điền đầy các vạch phổ của một hoặc nhiều băng thừa số tỉ lệ, mà trong đó tất cả cách vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra sử dụng các vạch phổ của kênh trộn phụ thuộc vào thừa số tỉ lệ của từng băng trong số một hoặc nhiều băng thừa số tỉ lệ mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không.

Trong các phương án này, thừa số tỉ lệ có thể, ví dụ, được gán cho từng băng thừa số tỉ lệ, và thừa số tỉ lệ được tính đến khi tạo ra nhiều âm sử dụng kênh trộn.

Trong phương án cụ thể, giao diện nhận 212 có thể, ví dụ, được tạo cấu hình để nhận thừa số tỉ lệ của từng băng trong số một hoặc nhiều băng thừa số tỉ lệ, và thừa số tỉ lệ của từng băng trong số một hoặc nhiều băng thừa số tỉ lệ đã nêu biểu thị năng lượng của các vạch phổ của băng thừa số tỉ lệ đã nêu trước khi lượng tử hóa. Môđun điền đầy nhiều âm 220 có thể, ví dụ, được làm thích ứng để tạo ra nhiều âm cho từng băng trong số một hoặc nhiều băng thừa số tỉ lệ, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, sao cho năng lượng của các vạch phổ sau khi bổ sung nhiều âm vào một trong số các băng tần tương ứng với năng lượng được biểu thị bởi thừa số tỉ lệ cho băng thừa số tỉ lệ đã nêu.

Ví dụ, kênh trộn có thể biểu thị cho các trị số quang phổ cho bốn vạch phổ của băng thừa số tỉ lệ mà trong đó nhiều âm sẽ được chèn vào, và các trị số quang phổ này có thể, ví dụ, là: 0,2; 0,3; 0,5; 0,1.

Năng lượng của băng thừa số tỉ lệ của kênh trộn có thể, ví dụ, được tính toán như sau:

$$(0,2)^2 + (0,3)^2 + (0,5)^2 + (0,1)^2 = 0,39$$

Tuy nhiên, thừa số tỉ lệ mà băng thừa số tỉ lệ của kênh mà trong đó nhiều âm sẽ được điền đầy có thể, ví dụ, chỉ là 0,0039.

Thừa số suy giảm có thể, ví dụ, được tính toán như sau:

$$\text{Thừa số suy giảm} = \frac{\text{Năng lượng được biểu thị bởi thừa số tỷ lệ}}{\text{Năng lượng của kênh trộn}}$$

Do đó, trong ví dụ ở trên,

$$\text{thừa số tỷ lệ} = \frac{0,0039}{0,39} = 0,01$$

Trong phương án, từng giá trị quang phổ của các băng thừa số tỉ lệ của kênh trộn mà sẽ được sử dụng như nhiều âm, được nhân với thừa số suy giảm.

Do đó, từng giá trị quang phổ trong số bốn giá trị quang phổ của băng thừa số tỉ lệ của ví dụ ở trên được nhân với thừa số suy giảm và thu được các giá trị quang phổ suy giảm:

$$0,2 \cdot 0,01 = 0,002$$

$$0,3 \cdot 0,01 = 0,003$$

$$0,5 \cdot 0,01 = 0,005$$

$$0,1 \cdot 0,01 = 0,001$$

Các giá trị quang phổ suy giảm có thể, ví dụ, sau đó được chèn vào các bảng thừa số tỉ lệ của kênh mà trong đó nhiều âm sẽ được điền đầy.

Ví dụ ở trên là có khả năng áp dụng như nhau lên các giá trị loga bằng cách thay thế các phép toán ở trên bởi các phép toán loga tương ứng của chúng, ví dụ, bằng cách thay thế phép nhân bởi phép cộng, v.v..

Hơn nữa, ngoài phần mô tả các phương án cụ thể được cung cấp ở trên, các phương án khác về môđun điền đầy nhiều âm 220 áp dụng một, một số hoặc tất cả các khái niệm được mô tả dựa vào các hình vẽ từ Fig.2 đến Fig.6.

Các khía cạnh khác của các phương án của sáng chế liên quan đến câu hỏi dựa trên kênh thông tin nào từ các kênh đầu ra âm thanh trước được chọn để được sử dụng để tạo ra kênh trộn để thu được nhiều âm sẽ được chèn.

Theo phương án, thiết bị theo môđun điền đầy nhiều âm 220 có thể, ví dụ, được làm thích ứng để lựa chọn chính xác hai kênh đầu ra âm thanh trước từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào các tham số đa kênh thứ nhất MCH_PAR2.

Do đó, trong phương án này, các tham số đa kênh thứ nhất mà hướng theo các kênh sẽ được chọn để được xử lý, cũng hướng theo các kênh đầu ra âm thanh trước sẽ được sử dụng để tạo ra kênh trộn để tạo ra nhiều âm sẽ được chèn.

Trong một phương án, các tham số đa kênh thứ nhất MCH_PAR2 có thể, ví dụ, biểu thị hai kênh được giải mã D1, D2 từ tập hợp ba hoặc nhiều hơn ba kênh được giải mã; và bộ xử lý đa kênh 204 được làm thích ứng để lựa chọn cặp lựa chọn thứ nhất gồm hai kênh âm thanh được giải mã D1, D2 từ tập hợp ba hoặc nhiều hơn ba kênh được giải mã D1, D2, D3 bằng cách lựa chọn hai kênh được giải mã D1, D2 được biểu thị bởi các tham số đa kênh thứ nhất MCH_PAR2. Hơn nữa, các tham số đa kênh thứ hai MCH_PAR1 có thể, ví dụ, biểu thị hai kênh được giải mã P1*, D3 từ tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã. Bộ xử lý đa kênh 204 có thể, ví

dụ, được làm thích ứng để lựa chọn cặp được lựa chọn thứ hai gồm hai kênh được giải mã P1*, D3 từ tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã D3, P1*, P2* bằng cách lựa chọn hai kênh được giải mã P1*, D3 được biểu thị bởi các tham số đa kênh thứ hai MCH_PAR1.

Do đó, trong phương án này, các kênh được lựa chọn cho bước xử lý thứ nhất, ví dụ, bước xử lý của hộp xử lý 208 trên Fig.1a hoặc Fig.1b không chỉ phụ thuộc vào các tham số đa kênh thứ nhất MCH_PAR2. Hơn thế, hai kênh được chọn này được định rõ trong các tham số đa kênh thứ nhất MCH_PAR2.

Cũng như vậy, trong phương án này, các kênh được lựa chọn cho bước xử lý thứ hai, ví dụ, bước xử lý của hộp xử lý 210 trên Fig.1a hoặc Fig.1b không chỉ phụ thuộc vào các tham số đa kênh thứ hai MCH_PAR1. Hơn thế, hai kênh được chọn này được định rõ trong các tham số đa kênh thứ hai MCH_PAR1.

Các phương án của sáng chế giới thiệu sơ đồ chỉ số hóa tinh vi cho các tham số đa kênh mà được giải thích dựa vào Fig.15.

Fig.15 (a) thể hiện việc mã hóa của năm kênh, cụ thể là các kênh bên trái, bên phải, trung tâm, bao quanh bên trái và bao quanh bên phải, trên phía bộ mã hóa. Fig.15 (b) thể hiện việc giải mã của các kênh được mã hóa E0, E1, E2, E3, E4 để khôi phục các kênh bên trái, bên phải, trung tâm, bao quanh bên trái và bao quanh bên phải.

Được giả định rằng chỉ số được gán cho từng kênh trong số năm kênh bên trái, bên phải, trung tâm, bao quanh bên trái và bao quanh bên phải, cụ thể là

Chỉ số	Tên kênh
0	Bên trái
1	Bên phải
2	Trung tâm
3	Bao quanh bên trái
4	Bao quanh bên phải

Trên Fig.15 (a), trên phía bộ mã hóa, phép toán thứ nhất mà được thực hiện có thể, ví dụ, là việc trộn của kênh 0 (bên trái) và kênh 3 (bao quanh bên trái) trong hộp xử lý 192 để thu được hai kênh được xử lý. Có thể được giả định rằng một trong số các

kênh được xử lý là kênh giữa và kênh khác là kênh bên. Tuy nhiên, khía cạnh khác của việc tạo thành hai kênh được xử lý cũng có thể được áp dụng, ví dụ, xác định hai kênh được xử lý bằng cách thực hiện phép toán luân phiên.

Ngay sau đây, hai kênh được xử lý đã tạo ra có được cùng chỉ số như các chỉ số của các kênh mà đã được sử dụng để xử lý. Cụ thể là, kênh thứ nhất trong số các kênh được xử lý có chỉ số 0 và kênh thứ hai trong số các kênh được xử lý có chỉ số 3. Các tham số đa kênh được xác định cho bước xử lý này có thể, ví dụ, là (0; 3).

Phép toán thứ hai trên phía bộ mã hóa mà được thực hiện có thể, ví dụ, là phép trộn của kênh 1 (bên phải) và kênh 4 (bao quanh bên phải) trong hộp xử lý 194 để thu được hai kênh được xử lý nữa. Ngoài ra, hai kênh được xử lý đã tạo ra nữa có được cùng chỉ số như các chỉ số của các kênh mà đã được sử dụng để xử lý. Cụ thể là, kênh thứ nhất trong số các kênh được xử lý nữa có chỉ số 1 và kênh thứ hai trong số các kênh được xử lý có chỉ số 4. Các tham số đa kênh được xác định cho bước xử lý này có thể, ví dụ, là (1; 4).

Phép toán thứ ba trên phía bộ mã hóa mà được thực hiện có thể, ví dụ, là phép trộn của kênh được xử lý 0 và kênh được xử lý 1 trong hộp xử lý 196 để thu được hai kênh được xử lý nữa. Ngoài ra, hai kênh được xử lý đã tạo ra này có được cùng chỉ số như các chỉ số của các kênh mà đã được sử dụng để xử lý. Cụ thể là, kênh thứ nhất trong số các kênh được xử lý nữa có chỉ số 0 và kênh thứ hai trong số các kênh được xử lý có chỉ số 1. Các tham số đa kênh được xác định cho bước xử lý này có thể, ví dụ, là (0; 1).

Các kênh được mã hóa E0, E1, E2, E3 và E4 được phân biệt bởi các chỉ số của chúng, cụ thể là, E0 có chỉ số 0, E1 có chỉ số 1, E2 có chỉ số 2, v.v..

Ba phép toán trên phía bộ mã hóa thu được ba tham số đa kênh:

(0; 3), (1; 4), (0; 1).

Vì thiết bị giải mã sẽ thực hiện các phép toán của bộ mã hóa theo thứ tự nghịch đảo, thứ tự của các tham số đa kênh có thể, ví dụ, được nghịch đảo khi được truyền đến thiết bị giải mã, thu được các tham số đa kênh:

(0; 1), (1; 4), (0; 3).

Đối với thiết bị giải mã, (0;1) có thể được đề cập đến như các tham số đa kênh thứ nhất, (1; 4) có thể được đề cập đến như các tham số đa kênh thứ hai và (0; 3) có thể được đề cập đến như các tham số đa kênh thứ ba.

Trên phía bộ giải mã được thể hiện trên Fig.15 (b), từ việc nhận các tham số đa kênh thứ nhất (0; 1), thiết bị giải mã kết luận rằng như phép toán xử lý thứ nhất trên phía bộ giải mã, các kênh 0 (E0) và 1 (E1) sẽ được xử lý. Điều này được tiến hành trong hộp 296 của Fig.15 (b). Cả hai kênh được xử lý đã tạo ra thừa hưởng các chỉ số từ các kênh E0 và E1 mà đã được sử dụng để tạo ra chúng, và do đó, các kênh được xử lý đã tạo ra cũng có các chỉ số 0 và 1.

Từ việc nhận các tham số đa kênh thứ hai (1; 4), thiết bị giải mã kết luận rằng như phép toán xử lý thứ hai trên phía bộ giải mã, kênh được xử lý 1 và kênh 4 (E4) sẽ được xử lý. Điều này được tiến hành trong hộp 294 của Fig.15 (b). Cả hai kênh được xử lý đã tạo ra thừa hưởng các chỉ số từ các kênh 1 và 4 mà đã được sử dụng để tạo ra chúng, và do đó, các kênh được xử lý đã tạo ra cũng có các chỉ số 1 và 4.

Từ việc nhận các tham số đa kênh thứ ba (0; 3), thiết bị giải mã kết luận rằng như phép toán xử lý thứ ba trên phía bộ giải mã, kênh được xử lý 0 và kênh 3 (E3) sẽ được xử lý. Điều này được tiến hành trong hộp 292 của Fig.15 (b). Cả hai kênh được xử lý đã tạo ra thừa hưởng các chỉ số từ các kênh 0 và 3 mà đã được sử dụng để tạo ra chúng, và do đó, các kênh được xử lý đã tạo ra cũng có các chỉ số 0 và 3.

Như kết quả của việc xử lý của thiết bị giải mã, các kênh bên trái (chỉ số 0), bên phải (chỉ số 1), trung tâm (chỉ số 2), bao quanh bên trái (chỉ số 3) và bao quanh bên phải (chỉ số 4) được khôi phục.

Giả định rằng trên phía bộ giải mã, do sự lượng tử hóa, tất cả các giá trị của kênh E1 (chỉ số 1) trong băng thừa số tỉ lệ nhất định đã được lượng tử hóa bằng không. Khi thiết bị giải mã muốn tiến hành xử lý trong hộp 296, kênh được điền đầy nhiễu âm 1 (kênh E1) được mong muốn.

Như đã được tóm tắt, các phương án ngay sau đây sử dụng hai tín hiệu đầu ra âm thanh trước để điền đầy lỗ quang phổ của kênh 1.

Trong phương án cụ thể, nếu kênh mà phép toán sẽ được thực hiện có các băng thừa số tỉ lệ mà được lượng tử hóa bằng không, thì hai kênh đầu ra âm thanh trước

được sử dụng để tạo ra nhiều âm mà có cùng số chỉ số như hai kênh mà việc xử lý sẽ được thực hiện. Trong ví dụ, nếu lỗ quang phổ của kênh 1 được phát hiện trước khi xử lý trong hộp xử lý 296, thì các kênh đầu ra âm thanh trước có chỉ số không (kênh bên trái trước) và có chỉ số 1 (kênh bên phải trước) được sử dụng để tạo ra nhiều âm để điền đầy khoảng trống quang phổ của kênh 1 trên phía bộ giải mã.

Vì các chỉ số được kế thừa một cách phù hợp bởi các kênh được xử lý mà thu được từ quá trình xử lý, có thể giả định rằng các kênh đầu ra trước sẽ đóng vai trò để tạo ra các kênh mà tham gia vào việc xử lý thực tế của phía bộ giải mã, nếu các kênh đầu ra âm thanh trước là các kênh đầu ra âm thanh hiện thời. Do đó, có thể đạt được ước lượng tốt cho băng thừa số tỷ mà đã được lượng tử hóa bằng không.

Theo các phương án, thiết bị có thể, ví dụ, được làm thích ứng để gán ký hiệu nhận biết từ tập hợp các ký hiệu nhận biết cho từng kênh đầu ra âm thanh trước trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, sao cho từng kênh đầu ra âm thanh trước trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước được gán cho chính xác một ký hiệu nhận biết trong số tập hợp các ký hiệu nhận biết, và sao cho từng ký hiệu nhận biết trong số các ký hiệu nhận biết được gán cho chính xác một kênh đầu ra âm thanh trước trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước. Hơn nữa, thiết bị có thể, ví dụ, được làm thích ứng để gán ký hiệu nhận biết từ tập hợp các ký hiệu nhận biết đã nêu cho từng kênh trong số ba hoặc nhiều hơn ba kênh được giải mã, sao cho từng kênh trong số tập hợp của ba hoặc nhiều hơn ba kênh được giải mã được gán cho chính xác một ký hiệu nhận biết trong tập hợp các ký hiệu nhận biết, và sao cho từng ký hiệu nhận biết trong số tập hợp các ký hiệu nhận biết được gán cho chính xác một kênh trong số tập hợp của ba hoặc nhiều hơn ba kênh được giải mã.

Hơn nữa, các tham số đa kênh thứ nhất MCH_PAR2 có thể, ví dụ, biểu thị cặp thứ nhất gồm hai ký hiệu nhận biết trong tập hợp của ba hoặc nhiều hơn ba ký hiệu nhận biết. Bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để lựa chọn cặp được lựa chọn thứ nhất gồm hai kênh được giải mã D1, D2 từ tập hợp của ba hoặc nhiều hơn ba kênh được giải mã D1, D2, D3 bằng cách lựa chọn hai kênh được giải mã D1, D2 được gán cho hai ký hiệu nhận biết của cặp thứ nhất gồm hai ký hiệu nhận biết.

Thiết bị có thể, ví dụ, được làm thích ứng để gán ký hiệu nhận biết thứ nhất trong số hai ký hiệu nhận biết của cặp thứ nhất gồm hai ký hiệu nhận biết cho kênh được xử lý thứ nhất của nhóm thứ nhất gồm chính xác hai kênh được xử lý P1*, P2*. Hơn nữa, thiết bị có thể, ví dụ, được làm thích ứng để gán ký hiệu nhận biết thứ hai trong số hai ký hiệu nhận biết của cặp thứ nhất gồm hai ký hiệu nhận biết cho kênh được xử lý thứ hai của nhóm thứ nhất gồm chính xác hai kênh được xử lý P1*, P2*.

Tập hợp các ký hiệu nhận biết có thể, ví dụ, là tập hợp của các chỉ số, ví dụ như tập hợp của các số nguyên không âm (ví dụ, tập hợp bao gồm các ký hiệu nhận biết 0; 1; 2; 3 và 4)

Trong các phương án cụ thể, các tham số đa kênh thứ hai MCH_PAR1 có thể, ví dụ, biểu thị cặp thứ hai gồm hai ký hiệu nhận biết trong tập hợp của ba hoặc nhiều hơn ba ký hiệu nhận biết. Bộ xử lý đa kênh 204 có thể, ví dụ, được làm thích ứng để lựa chọn cặp được lựa chọn thứ hai gồm hai kênh được giải mã P1*, D3 từ tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã D3, P1*, P2* bằng cách lựa chọn hai kênh được giải mã (D3, P1*) được gán cho hai ký hiệu nhận biết của cặp thứ hai gồm hai ký hiệu nhận biết. Hơn nữa, thiết bị có thể, ví dụ, được làm thích ứng để gán ký hiệu nhận biết thứ nhất trong số hai ký hiệu nhận biết của cặp thứ hai gồm hai ký hiệu nhận biết cho kênh được xử lý thứ nhất của nhóm thứ hai gồm chính xác hai kênh được xử lý P3*, P4*. Hơn nữa, thiết bị có thể, ví dụ, được làm thích ứng để gán ký hiệu nhận biết thứ hai trong số hai ký hiệu nhận biết của cặp thứ hai gồm hai ký hiệu nhận biết cho kênh được xử lý thứ hai của nhóm thứ hai gồm chính xác hai kênh được xử lý P3*, P4*.

Trong phương án cụ thể, các tham số đa kênh thứ nhất MCH_PAR2 có thể, ví dụ, biểu thị cặp thứ nhất đã nêu gồm hai ký hiệu nhận biết trong tập hợp của ba hoặc nhiều hơn ba ký hiệu nhận biết. Môđun điền đầy nhiễu âm 220 có thể, ví dụ, được làm thích ứng để lựa chọn chính xác hai kênh đầu ra âm thanh trước từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước bằng cách lựa chọn hai kênh đầu ra âm thanh trước được gán cho hai ký hiệu nhận biết của cặp thứ nhất đã nêu gồm hai ký hiệu nhận biết.

Như đã được tóm tắt, Fig.7 minh họa thiết bị 100 để mã hóa tín hiệu đa kênh 101 có ít nhất ba kênh (CH1:CH3) theo phương án.

Thiết bị bao gồm bộ xử lý phép lặp 102 được làm thích ứng để tính toán, trong bước lặp thứ nhất, giá trị tương quan liên kênh giữa từng cặp trong số ít nhất ba kênh (CH:CH3), để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị trên ngưỡng, và để xử lý cặp được lựa chọn bằng cách sử dụng phép toán xử lý đa kênh 110,112 để suy ra các tham số đa kênh ban đầu MCH_PAR1 cho cặp được lựa chọn thứ nhất và để suy ra các kênh được xử lý thứ nhất P1,P2.

Bộ xử lý phép lặp 102 được làm thích ứng để thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai bằng cách sử dụng ít nhất một trong số các kênh được xử lý P1 để suy ra các tham số đa kênh nữa MCH_PAR2 và các kênh được xử lý thứ hai P3, P4.

Hơn nữa, thiết bị bao gồm bộ mã hóa kênh được làm thích ứng để mã hóa các kênh (P2:P4) thu được từ việc xử lý phép lặp được thực hiện bởi bộ xử lý phép lặp 104 để thu được các kênh được mã hóa (E1:E3).

Hơn nữa, thiết bị bao gồm giao diện đầu ra 106 được làm thích ứng để tạo ra tín hiệu đa kênh được mã hóa 107 có các kênh được mã hóa (E1:E3), các tham số đa kênh ban đầu và các tham số đa kênh nữa MCH_PAR1, MCH_PAR2.

Hơn nữa, thiết bị bao gồm giao diện đầu ra 106 được làm thích ứng để tạo ra tín hiệu đa kênh được mã hóa 107 để bao gồm thông tin biểu thị liệu có hay không có việc thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiễu âm được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã.

Do đó, thiết bị mã hóa có khả năng báo hiệu liệu có hay không có việc thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiễu âm được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã.

Theo phương án, từng tham số trong số các tham số đa kênh ban đầu và các tham số đa kênh nữa MCH_PAR1, MCH_PAR2 biểu thị chính xác hai kênh, từng kênh trong số chính xác hai kênh là một trong số các kênh được mã hóa (E1:E3) hoặc

là một trong số kênh được xử lý thứ nhất hoặc thứ hai P1, P2, P3, P4 hoặc là một trong số ít nhất ba kênh (CH1:CH3).

Giao diện đầu ra 106 có thể, ví dụ, được làm thích ứng để tạo ra tín hiệu đa kênh được mã hóa 107, sao cho thông tin biểu thị liệu có hay không việc thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó các vạch phổ được lượng tử hóa bằng không, bao gồm thông tin biểu thị từng tham số trong số các tham số đa kênh ban đầu và các tham số đa kênh MCH_PAR1, MCH_PAR2, liệu có hay không việc đối với ít nhất một kênh trong số chính xác hai kênh mà được biểu thị bởi một trong số các tham số đa kênh ban đầu và các tham số đa kênh MCH_PAR1, MCH_PAR2, thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, của ít nhất một kênh đã nêu, với dữ liệu quang phổ được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã.

Dưới đây, các phương án cụ thể được mô tả trong đó các thông tin này được truyền bằng cách sử dụng giá trị hasStereoFilling[pair] mà biểu thị liệu có hay không việc điền đầy nhiều âm trong cặp kênh MCT được xử lý hiện thời sẽ được áp dụng.

Fig.13 minh họa hệ thống theo các phương án.

Hệ thống bao gồm thiết bị 100 để mã hóa như được mô tả ở trên, thiết bị 201 để giải mã theo một trong số các phương án được mô tả ở trên.

Thiết bị 201 để giải mã được tạo cấu hình để nhận tín hiệu đa kênh được mã hóa 107, được tạo ra bởi thiết bị 100 để mã hóa, từ thiết bị 100 để mã hóa.

Hơn nữa, tín hiệu đa kênh được mã hóa 107 được cung cấp.

Tín hiệu đa kênh được mã hóa bao gồm

- các kênh được mã hóa (E1:E3), và
- các tham số đa kênh MCH_PAR1, MCH_PAR2, và
- thông tin biểu thị liệu có hay không có việc thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với dữ liệu quang phổ được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã.

Theo phương án của sáng chế, tín hiệu đa kênh được mã hóa có thể, ví dụ, bao gồm, như các tham số đa kênh MCH_PAR1, MCH_PAR2, hai hoặc nhiều hơn hai tham số đa kênh.

Từng tham số trong số hai hoặc nhiều hơn hai tham số đa kênh MCH_PAR1, MCH_PAR2 có thể, ví dụ, biểu thị chính xác hai kênh, mỗi kênh trong số chính xác hai kênh là một trong số các kênh được mã hóa (E1:E3) hoặc là một trong số nhiều kênh được xử lý P1, P2, P3, P4 hoặc là một trong số ít nhất ba kênh gốc (ví dụ, không được xử lý) (CH:CH3).

Thông tin biểu thị liệu có hay không việc thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó các vạch phổ được lượng tử hóa bằng không có thể, ví dụ, bao gồm thông tin biểu thị từng tham số trong số hai hoặc nhiều hơn hai tham số đa kênh MCH_PAR1, MCH_PAR2, liệu có hay không việc đối với ít nhất một kênh trong số chính xác hai kênh mà được biểu thị bởi một trong số hai hoặc nhiều tham số đa kênh, thiết bị giải mã sẽ điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, của ít nhất một kênh đã nêu, với dữ liệu quang phổ được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã.

Như đã được tóm tắt, ở dưới đây, các phương án cụ thể được mô tả trong đó các thông tin này được truyền bằng cách sử dụng giá trị hasStereoFilling[pair] mà biểu thị liệu có hay không việc điền đầy âm lập thể trong cặp kênh MCT được xử lý hiện thời sẽ được áp dụng.

Trong phần sau đây, các khái niệm chung và các phương án cụ thể được mô tả chi tiết hơn.

Các phương án thực hiện chế độ mã hóa tốc độ bit thấp theo tham số với sự linh hoạt trong việc sử dụng các cây âm lập thể tùy chọn, tổ hợp của phép điền đầy âm lập thể và MCT.

Các phần phụ thuộc tín hiệu liên kênh được khai thác bằng cách áp dụng theo thứ bậc các công cụ mã hóa âm lập thể liên kết đã biết. Đối với các tốc độ bit thấp, các phương án mở rộng MCT để sử dụng tổ hợp của các hộp mã hóa âm lập thể và các hộp điền đầy âm lập thể rời rạc. Do đó, việc mã hóa bán tham số có thể được áp dụng, ví

dự, cho các kênh với ngữ cảnh tương tự, tức là các cặp kênh với sự tương quan cao nhất, trong khi các kênh khác có thể được mã hóa độc lập hoặc thông qua phép biểu diễn không theo tham số. Do đó, cú pháp dòng bit MCT được mở rộng để có thể báo hiệu nếu phép điền đầy âm lập thể được cho phép mà trong đó nó đang hoạt động.

Các phương án thực hiện việc tạo ra phép trộn giảm trước cho các cặp điền đầy âm lập thể tùy chọn.

Điền đầy âm lập thể dựa vào việc sử dụng phép trộn giảm của khung trước để cải thiện việc điền đầy các lỗ trống quang phổ tạo ra bởi việc lượng tử hóa trong miền quang phổ. Tuy nhiên, trong tổ hợp với MCT, tập hợp các cặp âm lập thể được mã hóa kết hợp được cho phép ngay sẽ là biến thiên thời gian. Do đó, hai kênh được mã hóa kết hợp có thể đã không được mã hóa kết hợp trong khung trước, tức là khi cấu hình dạng cây được thay đổi.

Để ước lượng phép trộn giảm trước, các kênh đầu ra được giải mã trước được lưu và được xử lý với phép toán âm lập thể nghịch đảo. Đối với hộp âm lập thể đã cho, việc này được thực hiện bằng cách sử dụng các tham số của khung hiện thời và các kênh đầu ra được giải mã của khung trước tương ứng với các chỉ số kênh của hộp âm lập thể được xử lý.

Nếu tín hiệu kênh đầu ra trước không có sẵn, ví dụ, do khung độc lập (khung mà có thể được giải mã mà không tính đến khung âm thanh trước) hoặc thay đổi độ dài biến đổi, vùng đệm kênh trước của kênh tương ứng được thiết lập bằng không. Do đó, phép trộn giảm trước khác không có thể vẫn được tính, miễn là ít nhất một trong số các tín hiệu kênh trước có sẵn.

Nếu MCT được tạo cấu hình để sử dụng các hộp âm lập thể dựa trên phép dự báo, phép trộn giảm trước được tính toán với phép toán MS nghịch đảo như được chỉ rõ cho các cặp điền đầy âm lập thể, tốt hơn là sử dụng một trong số hai phương trình sau dựa trên cờ hiệu hướng dự báo (*pred_dir* trong cú pháp MPEG-H).

$$D_1 = (\overline{O}_1 + \overline{O}_2) \cdot d$$

$$D_2 = (\overline{O}_1 - \overline{O}_2) \cdot d,$$

trong đó d là vô hướng dương và thực tùy chọn.

Nếu MCT được tạo cấu hình để sử dụng các hộp âm lập thể dựa trên sự luân phiên, phép trộn giảm trước được tính toán bằng cách sử dụng phép luân phiên với góc luân phiên được phủ định.

Do đó, đối với phép luân phiên được cho như sau:

$$\begin{bmatrix} O_1 \\ O_2 \end{bmatrix} = \begin{bmatrix} \cos \alpha & -\sin \alpha \\ \sin \alpha & \cos \alpha \end{bmatrix} \cdot \begin{bmatrix} I_1 \\ I_2 \end{bmatrix}$$

phép luân phiên nghịch đảo được tính toán như sau:

$$\begin{bmatrix} \widehat{I}_1 \\ \widehat{I}_2 \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} \cdot \begin{bmatrix} \widehat{O}_1 \\ \widehat{O}_2 \end{bmatrix}$$

với $\widehat{I}_1$ là phép trộn giảm trước được mong muốn của các kênh đầu ra trước $\widehat{O}_1$ và $\widehat{O}_2$.

Các phương án thực hiện ứng dụng của điện đầy âm lập thể trong MCT.

Ứng dụng của điện đầy âm lập thể đối với hộp âm lập thể đơn được mô tả trong tài liệu [1], [5].

Như đối với hộp âm lập thể đơn, điện đầy âm lập thể được áp dụng cho kênh thứ hai thuộc cặp kênh MCT đã cho.

Không kể đến những điều khác, các chênh lệch của điện đầy âm lập thể kết hợp với MCT là như sau:

Cấu hình cây MCT được mở rộng bởi một bit báo hiệu trên mỗi khung để có thể báo hiệu nếu điện đầy âm lập thể được cho phép trong khung hiện thời.

Trong phương án được ưu tiên, nếu điện đầy âm lập thể được cho phép trong khung hiện thời, một bit bổ sung để kích hoạt phép điện đầy âm lập thể trong hộp âm lập thể được truyền cho từng hộp âm lập thể. Đây là phương án được ưu tiên vì cho phép sự điều khiển phía bộ giải mã mà qua đó các hộp sẽ có phép điện đầy âm lập thể được áp dụng cho bộ giải mã.

Trong phương án thứ hai, nếu điện đầy âm lập thể được cho phép trong khung hiện thời, điện đầy âm lập thể được cho phép trong tất cả các hộp âm lập thể và không có bit bổ sung được truyền cho từng hộp âm lập thể riêng rẽ. Trong trường hợp này, ứng dụng chọn lọc của điện đầy âm lập thể trong các hộp MCT riêng rẽ được điều khiển bởi bộ giải mã.

Các khái niệm và các phương án chi tiết nữa sẽ được mô tả sau đây:

Các phương án cải thiện chất lượng cho các điểm hoạt động đa kênh có tốc độ bit thấp.

Trong phần tử cặp kênh (channel pair element - CPE) được mã hóa miền tần số (frequency-domain - FD), tiêu chuẩn âm thanh MPEG-H 3D cho phép sử dụng công cụ điền đầy âm lập thể, được mô tả trong phần nhỏ 5.5.5.4.9 của tài liệu [1], cho việc điền đầy các lỗ quang phổ được cải thiện theo cảm quan gây ra bởi sự lượng tử hóa rất thô trong bộ mã hóa. Công cụ này đã được thể hiện để đặc biệt có lợi cho âm lập thể hai kênh được mã hóa ở tốc độ bit trung bình và thấp.

Công cụ mã hóa đa kênh (MCT), được mô tả trong phần 7 của tài liệu [2], đã được giới thiệu, mà cho phép các định nghĩa thích ứng tín hiệu linh hoạt của cặp kênh được mã hóa kết hợp trên cơ sở trên mỗi khung để khai thác các phụ thuộc liên kênh biến thiên theo thời gian trong thiết lập đa kênh. Giá trị của MCT là đặc biệt đáng kể khi được sử dụng cho mã hóa kết hợp động có hiệu quả của các thiết lập đa kênh khi từng kênh có trong phần tử kênh đơn (single channel element - SCE) riêng rẽ của nó vì, không giống các cấu hình CPE + SCE (+ LFE) truyền thống mà phải được thiết lập trước, nó cho phép mã hóa kênh kết hợp để được ghép tầng và/hoặc được tạo cấu hình lại từ một khung tới khung tiếp theo.

Mã hóa âm thanh nổi đa kênh mà không sử dụng các CPE đang mang lại các bất lợi mà các công cụ âm thanh nổi kết hợp chỉ khả dụng trong các CPE - mã hóa M/S dự báo và điền đầy âm lập thể - không thể được khai thác, mà đặc biệt bất lợi ở các tốc độ bit trung bình và thấp. MCT có thể tác động như phần thay thế cho công cụ M/S, như phần thay thế cho công cụ điền đầy âm lập thể hiện không khả dụng.

Các phương án cho phép việc sử dụng công cụ điền đầy âm lập thể cũng trong các cặp kênh của MCT bằng cách mở rộng cú pháp dòng bit MCT với bit báo hiệu tương ứng và bằng cách khái quát hóa ứng dụng của điền đầy âm lập thể vào các cặp kênh tùy chọn bất kể loại phần tử kênh nào của chúng.

Một số phương án có thể, ví dụ, thực hiện việc báo hiệu phép điền đầy âm lập thể trong MCT như sau:

Trong CPE, việc sử dụng công cụ điền đầy âm lập thể được báo hiệu trong thông tin điền đầy nhiều âm FD cho kênh thứ hai, như được mô tả trong phần phụ 5.5.5.4.9.4 của [1]. Khi sử dụng MCT, mọi kênh tiềm ẩn là "kênh thứ hai" (do khả năng là cặp kênh phân tử chéo. Do đó được đề xuất để báo hiệu rõ ràng phép điền đầy âm lập thể bởi bit bổ sung trên mỗi cặp kênh được mã hóa MCT. Để tránh nhu cầu cho bit bổ sung này khi điền đầy nhiều âm không được sử dụng trong bất kỳ cặp kênh nào của trường hợp "cây" MCT cụ thể, hai mục nhập dành riêng hiện thời của phần tử *MCTSignalingType* trong *MultichannelCodingFrame()* [2] được sử dụng để báo hiệu sự có mặt của bit bổ sung trên mỗi cặp kênh được đề cập ở trên.

Sự mô tả chi tiết được cung cấp dưới đây.

Một số phương án có thể, ví dụ, thực hiện việc tính toán phép trộn giảm trước như sau:

Điền đầy âm lập thể trong CPE điền đầy các băng thừa số tỉ lệ "trống" của kênh thứ hai bằng việc bổ sung các hệ số MDCT tương ứng của phép trộn giảm của khung trước, được định tỉ lệ theo các thừa số tỉ lệ được truyền của các băng tương ứng (mà không được sử dụng vì các băng được lượng tử hóa hoàn toàn bằng không). Quy trình bổ sung được gán trọng số, được điều khiển bằng cách sử dụng các băng thừa số tỉ lệ của kênh mục tiêu, có thể được sử dụng giống nhau trong ngữ cảnh MCT. Tuy nhiên, quang phổ nguồn để điền đầy âm lập thể, tức là, phép trộn giảm của khung trước phải được tính toán theo cách khác trong các CPE, cụ thể là do cấu hình "cây" MCT có thể biến đổi theo thời gian.

Trong MCT, phép trộn giảm trước có thể được suy ra từ các kênh đầu ra được giải mã của khung trước (mà được lưu trữ sau khi giải mã MCT) sử dụng các tham số MCT của khung hiện thời cho cặp kênh kết hợp đã cho. Đối với cặp áp dụng phép mã hóa kết hợp dựa trên M/S dự báo, phép trộn giảm trước bằng, như trong điền đầy âm lập thể CPE, hoặc là tổng hoặc là chênh lệch của các quang phổ kênh thích hợp, phụ thuộc vào bộ chỉ thị hướng của khung hiện thời. Đối với cặp âm lập thể sử dụng phép mã hóa kết hợp dựa trên phép luân phiên Karhunen-Loève, phép trộn giảm trước biểu

diễn phép luân phiên nghịch đảo được tính với (các) góc luân phiên của khung hiện thời. Một lần nữa, sự mô tả chi tiết được cung cấp dưới đây.

Sự đánh giá độ phức tạp thể hiện rằng điền đầy âm lập thể trong MCT, là công cụ có tốc độ bit trung bình và thấp, không được mong đợi để làm tăng độ phức tạp trong trường hợp xấu nhất khi được đo trên cả tốc độ bit thấp/trung bình và tốc độ bit cao. Hơn nữa, sử dụng phép điền đầy âm lập thể thường trùng khớp với nhiều hệ số quang phổ được lượng tử hóa bằng không, bằng cách đó làm giảm độ phức tạp thuật toán của bộ giải mã số học dựa trên ngữ cảnh. Giả định sử dụng nhiều nhất $N/3$ kênh điền đầy nhiều âm trong cấu hình bao quanh N kênh và 02 WMOPS bổ sung trên mỗi phép thực hiện điền đầy nhiều âm, độ phức tạp cao nhất tăng chỉ 0,4 WMOPS đối với kênh 5.1 và 0,8 WMOPS đối với kênh 11.1 khi tốc độ lấy mẫu bộ mã hóa là 48kHz và công cụ IGF hoạt động chỉ trên 12kHz. Lượng này nhỏ hơn 2% trong tổng độ phức tạp bộ giải mã.

Các phương án thực hiện phần tử MultichannelCodingFrame() như sau:

Cú pháp	Số bit	Thuật nhớ
<pre> MultichannelCodingFrame() { MCTSignalingType; keepTree; if(keepTree==0) { numPairs=escapedValue(5,8,16); } else { numPairs=lastNumPairs; } MCTStereoFilling = 0; if (MCTSignalingType > 1) { </pre>	2 1	uimbsf uimbsf

<pre> MCTSignalingType = MCTSignalingType - 2; MCTStereoFilling = 1; } for(pair=0; pair<numPairs;pair++) { hasStereoFilling[pair] = 0; if(MCTStereoFilling == 1) { hasStereoFilling[pair]; } if(MCTSignalingType == 0) { /* tree of stereo prediction boxes */ MultichannelCodingBoxPrediction(); } if(MCTSignalingType == 1) { /* tree of rotation boxes */ MultichannelCodingBoxRotation(); } if((MCTSignalingType == 2) - (MCTSignalingType == 3)) { /* reserved */ } } } </pre>	1	uimsbf
--	----------	--------

Điền đầy âm lập thể trong MCT có thể, theo một số phương án, được thực hiện như sau:

Giống như điền đầy âm lập thể cho IGF trong phần tử cặp kênh, được mô tả trong mục nhỏ 5.5.5.4.9 của tài liệu [1], điền đầy âm lập thể trong công cụ mã hóa đa kênh (MCT) điền đầy các băng thừa số tỉ lệ "trống" (mà được lượng tử hóa đầy đủ

bằng không) tại và trên tần số bắt đầu điền đầy nhiều âm sử dụng phép trộn giảm của phổ đầu ra của khung trước.

Khi phép điền đầy nhiều âm hoạt động trong cặp kênh kết hợp MCT ($\text{hasStereoFilling}[\text{pair}] \neq 0$ trong bảng AMD4.4), tất cả các băng thừa số tỉ lệ “trống” trong vùng điền đầy nhiều âm (tức là bắt đầu tại hoặc trên $\text{noiseFillingStartOffset}$) của kênh thứ hai của cặp được điền đầy với năng lượng đích cụ thể sử dụng phép trộn giảm của quang phổ đầu ra tương ứng (sau ứng dụng MCT) của khung trước. Điều này được thực hiện sau khi điền đầy nhiều âm FD (xem mục nhỏ 7.2 trong ISO/IEC 23003-3:2012) và trước thừa số tỉ lệ và ứng dụng âm lập thể kết hợp MCT. Tất cả các quang phổ đầu ra sau khi xử lý MCT hoàn toàn được lưu cho phép điền đầy âm lập thể tiềm năng trong khung tiếp theo.

Sự hạn chế hoạt động có thể, ví dụ, là phép thực hiện theo từng bậc của thuật toán điền đầy âm lập thể ($\text{hasStereoFilling}[\text{pair}] \neq 0$) trong các băng trống của kênh thứ hai không được hỗ trợ cho bất kỳ cặp âm lập thể MCT theo sau nào với $\text{hasStereoFilling}[\text{pair}] \neq 0$ nếu kênh thứ hai là tương tự. Trong phần tử cặp kênh, phép điền đầy âm lập thể IGF hoạt động trong kênh (dư) thứ hai theo mục nhỏ 5.5.5.4.9 trong tài liệu [1] được ưu tiên hơn và do đó, làm bất hoạt - bất kỳ ứng dụng nào sau đó của phép điền đầy âm lập thể MCT trong cùng kênh của cùng khung.

Các thuật ngữ và định nghĩa có thể được nêu như sau:

$\text{hasStereoFilling}[\text{pair}]$	biểu thị việc sử dụng phép điền đầy âm lập thể trong cặp kênh MCT được xử lý hiện thời
$\text{ch1}, \text{ch2}$	biểu thị các kênh trong cặp kênh MCT được xử lý hiện thời
$\text{spectral_data}[\text{ }][\text{ }]$	là các hệ số quang phổ của các kênh trong cặp kênh MCT được xử lý hiện thời.
$\text{spectral_data_prev}[\text{ }][\text{ }]$	là các quang phổ đầu ra sau phép xử lý MCT hoàn thành trong khung trước
$\text{downmix_prev}[\text{ }][\text{ }]$	là phép trộn giảm ước lượng của các kênh đầu ra của khung trước với các chỉ số được cho bởi cặp kênh MCT được xử lý hiện thời

<code>num_swb</code>	là tổng số lượng các băng thừa số tỉ lệ, xem ISO/IEC 23003-3, mục nhỏ 6.2.9.4
<code>ccfl</code>	<code>coreCoderFrameLength</code> , độ dài biến đổi, xem ISO/IEC 23003-3, mục nhỏ 6.1.
<code>noiseFillingStartOffset</code>	đường bắt đầu điền đầy nhiễu âm, được xác định phụ thuộc vào <code>ccfl</code> trong ISO/IEC 23003-3, bảng 109.
<code>igf_WhiteningLevel</code>	làm trắng quang phổ trong IGF, xem ISO/IEC 23008-3, mục nhỏ 5.5.5.4.7
<code>seed[]</code>	sự gieo mầm phép điền đầy nhiễu âm được sử dụng bởi <code>randomSign()</code> , xem ISO/IEC 23003-3, mục nhỏ 7.2.

Đối với một số phương án cụ thể, quy trình giải mã có thể, ví dụ, được mô tả như sau:

Điền đầy âm lập thể MCT được thực hiện bằng cách sử dụng các thao tác liên tiếp, mà được mô tả trong phần sau:

Bước 1: Chuẩn bị quang phổ của các kênh thứ hai cho thuật toán điền đầy âm lập thể

Nếu bộ chỉ thị điền đầy âm lập thể `tgheer` cho cặp kênh MCT đã cho, `hasStereoFilling[pair]`, bằng không, phép điền đầy nhiễu âm không được sử dụng, và các bước sau không được thực hiện. Nếu không thì ứng dụng thừa số tỉ lệ không được thực hiện nếu được áp dụng trước cho quang phổ kênh thứ hai của cặp, `spectral_data[ch2]`.

Bước 2: Tạo ra quang phổ trộn giảm trước cho cặp kênh MCT đã cho

Phép trộn giảm trước được ước lượng từ các tín hiệu đầu ra của khung trước `spectral_data_prev[ ][ ]` mà được lưu trữ sau ứng dụng của phép xử lý MCT. Nếu tín hiệu kênh đầu ra trước không khả dụng, ví dụ do khung phụ thuộc (`indepFlag > 0`), thay đổi độ dài biến đổi hoặc `core_mode == 1`, vùng đệm kênh trước của kênh tương ứng sẽ được thiết lập bằng không.

Đối với các cặp âm lập thể dự báo, tức là `MCTSignalingType == 0`, phép trộn giảm trước được tính toán từ các kênh đầu ra trước là `downmix_prev[ ][ ]` được xác định trong bước 2 của mục nhỏ 5.5.5.4.9.4 của tài liệu [1], do đó `spectrum>window[ ][ ]` được biểu diễn bởi `spectral_data[ ][window]`.

Đối với các cặp âm lập thể luân phiên, tức là `MCTSignalingType == 1`, phép trộn giảm trước được tính toán từ các kênh đầu ra trước bằng cách nghịch đảo phép toán luân phiên được xác định trong mục nhỏ 5.5.X.3.7.1 của tài liệu [2].

```
apply_mct_rotation_inverse(*R, *L, *dmx, aIdx, nSamples)
{
    for (n=0; n<nSamples; n++) {
        dmx = L[n] * tabIndexToCosAlpha[aIdx] + R[n] * tabIndexToSinAlpha[aIdx];
    }
}

sử dụng L = spectral_data_prev[ch1][ ], R = spectral_data_prev[ch2][ ], dmx =
downmix_prev[ ] của khung trước và sử dụng aIdx, nSamples của khung hiện thời và
cặp MCT.
```

Bước 3: Việc thực hiện thuật toán điền đầy âm lập thể trong các băng trống của kênh thứ hai

Phép điền đầy âm lập thể được áp dụng trong kênh thứ hai của cặp MCT như trong bước 3 của mục nhỏ 5.5.5.4.9.4 của tài liệu [1], do đó `spectrum>window` được biểu diễn bởi

`spectral_data[ch2][window]` và `max_sfb_ste` được cho bởi `num_swb`.

Bước 4: Ứng dụng thừa số tỉ lệ và sự đồng bộ hóa thích ứng của các phép gieo mầm điền đầy nhiễu âm.

Sau bước 3 của mục nhỏ 5.5.5.4.9.4 của tài liệu [1], các thừa số tỉ lệ được áp dụng lên quang phổ kết quả như trong 7.3 của ISO/IEC 23003-3, với các thừa số tỉ lệ của các băng trống được xử lý giống như các thừa số tỉ lệ đều đặn. Trong trường hợp thừa số tỉ lệ không được xác định, ví dụ do nó được định vị trên `max_sfb`, giá trị của nó sẽ bằng không. Nếu IGF được sử dụng, `igf_WhiteningLevel` bằng 2 trong bất kỳ ô nào của kênh thứ hai, và cả hai kênh không sử dụng tám phép biến đổi ngắn, năng lượng quang phổ của cả hai kênh trong cặp MCT được tính toán trong khoảng từ chỉ số `noiseFillingStartOffset` đến chỉ số `ccfl/2 - 1` trước khi thực hiện `decode_mct( )`. Nếu

năng lượng được tính của kênh thứ nhất lớn hơn trên tám lần năng lượng của kênh thứ hai, seed[ch2] của kênh thứ hai được thiết lập bằng seed[ch1] của kênh thứ nhất.

Dù một số khía cạnh đã được mô tả trong ngữ cảnh của thiết bị, sẽ là rõ ràng rằng các khía cạnh này cũng biểu diễn sự mô tả của phương pháp tương ứng, trong đó khối hoặc thiết bị tương ứng với bước phương pháp hoặc dấu hiệu của bước trong phương pháp. Ngoài ra, các khía cạnh được mô tả trong ngữ cảnh của bước trong phương pháp cũng biểu diễn sự mô tả của khối hoặc mục hoặc dấu hiệu tương ứng của thiết bị tương ứng. Một số hoặc tất cả các bước phương pháp có thể được chạy bởi (hoặc sử dụng) thiết bị phần cứng, ví dụ như bộ vi xử lý, máy tính có thể lập trình hoặc mạch điện tử. Trong một số phương án, một hoặc nhiều trong số các bước quan trọng nhất trong phương pháp có thể được chạy bởi thiết bị này.

Phụ thuộc vào các yêu cầu nhất định, các phương án của sáng chế có thể được thực hiện trong phần cứng hoặc phần mềm hoặc ít nhất một phần trong phần cứng hoặc ít nhất một phần trong phần mềm. Phương án thực hiện có thể được thực hiện bằng cách sử dụng vật ghi lưu trữ số, ví dụ đĩa mềm, DVD, Blu-Ray, CD, ROM, PROM, EPROM, EEPROM hoặc bộ nhớ FLASH, có tín hiệu điều khiển có thể đọc được bằng điện tử được lưu trữ trên đó, mà kết hợp (hoặc có khả năng kết hợp) với hệ thống máy tính có thể lập trình sao cho phương pháp tương ứng được thực hiện. Do đó, vật ghi lưu trữ số có thể có khả năng đọc được bằng máy tính.

Một số phương án theo sáng chế bao gồm vật mang dữ liệu có các tín hiệu điều khiển có thể đọc được bằng điện tử, mà có khả năng kết hợp với hệ thống máy tính có thể lập trình được, sao cho một trong số các phương pháp được mô tả ở đây được thực hiện.

Nói chung, các phương án của sáng chế có thể được thực hiện như sản phẩm chương trình máy tính với mã chương trình, mã chương trình có tác dụng thực hiện một trong số các phương pháp khi sản phẩm chương trình máy tính chạy trên máy tính. Mã chương trình có thể, ví dụ, được lưu trữ trên vật ghi đọc được bằng máy.

Các phương án khác bao gồm chương trình máy tính để thực hiện một trong số các phương pháp được mô tả ở đây, được lưu trữ trên vật mang có thể đọc được bằng máy.

Do đó, nói cách khác, phương án của phương pháp theo sáng chế là chương trình máy tính có mã chương trình để thực hiện một trong số các phương pháp được mô tả ở đây, khi chương trình máy tính chạy trên máy tính.

Do đó, phương án nữa của phương pháp theo sáng chế là vật mang dữ liệu (hoặc vật ghi lưu trữ số, hoặc vật ghi có thể đọc được bằng máy tính) bao gồm chương trình máy tính đã được lưu trữ trên máy tính để thực hiện một trong số các phương pháp được mô tả ở đây. Vật mang dữ liệu, vật ghi lưu trữ số hoặc vật ghi đã được ghi là thường là hữu hình và/hoặc không tạm thời.

Do đó, phương án nữa của phương pháp theo sáng chế là dòng dữ liệu hoặc chuỗi các tín hiệu biểu diễn chương trình máy tính để thực hiện một trong số các phương pháp được mô tả ở đây. Dòng dữ liệu hoặc chuỗi tín hiệu có thể, ví dụ, được tạo cấu hình để được truyền thông qua kết nối truyền thông dữ liệu, ví dụ thông qua internet.

Phương án nữa bao gồm phương tiện xử lý, ví dụ máy tính hoặc thiết bị logic có khả năng lập trình, được tạo cấu hình để hoặc được làm thích ứng để thực hiện một trong số các phương pháp được mô tả ở đây.

Phương án nữa bao gồm chương trình máy tính đã được cài đặt trên đó chương trình máy tính để thực hiện một trong số các phương pháp được mô tả ở đây.

Phương án nữa theo sáng chế bao gồm thiết bị hoặc hệ thống được tạo cấu hình để chuyển (ví dụ bằng điện tử hoặc quang học) chương trình máy tính để thực hiện một trong số các phương pháp được mô tả ở đây đến bộ nhận. Bộ nhận có thể, ví dụ, là máy tính, thiết bị điện tử, thiết bị nhớ hoặc tương tự. Thiết bị hoặc hệ thống có thể, ví dụ, bao gồm máy chủ tệp tin để truyền chương trình máy tính đến bộ nhận.

Trong một số phương án, thiết bị logic có thể lập trình được (ví dụ mảng cổng lập trình được dạng trường) có thể được sử dụng để thực hiện một số hoặc tất cả các chức năng của các phương pháp được mô tả ở đây. Trong một số phương án, mảng cổng lập trình được dạng trường có thể kết hợp với bộ vi xử lý để thực hiện một trong số các phương pháp được mô tả ở đây. Nói chung, các phương pháp tốt hơn là được thực hiện bởi thiết bị phân cứng bất kỳ.

Thiết bị được mô tả ở đây có thể được thực hiện bằng cách sử dụng thiết bị phần cứng, hoặc sử dụng máy tính, hoặc sử dụng tổ hợp của thiết bị phần cứng và máy tính.

Phương pháp được mô tả ở đây có thể được thực hiện bằng cách sử dụng thiết bị phần cứng, hoặc sử dụng máy tính, hoặc sử dụng tổ hợp của thiết bị phần cứng và máy tính.

Các phương án được mô tả ở trên chỉ minh họa cho các nguyên tắc của sáng chế. Được hiểu là các biến đổi và thay đổi về cách sắp đặt hoặc các chi tiết được mô tả ở đây sẽ là hiển nhiên với người có hiểu biết trung bình trong lĩnh vực kỹ thuật tương ứng. Do đó, ý định là sáng chế chỉ được giới hạn bởi phạm vi của các điểm yêu cầu bảo hộ sáng chế sau đây và không bị giới hạn bởi các chi tiết cụ thể được thể hiện bằng cách mô tả và giải thích của các phương án được nêu trong bản mô tả.

YÊU CẦU BẢO HỘ

1. Thiết bị (201) giải mã tín hiệu đa kênh được mã hóa trước của khung trước để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và giải mã tín hiệu đa kênh được mã hóa hiện thời (107) của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh,

trong đó thiết bị (201) bao gồm giao diện (212), bộ giải mã kênh (202), bộ xử lý đa kênh (204) để tạo ra ba hoặc nhiều hơn ba kênh đầu ra âm thanh hiện thời và môđun điền đầy nhiễu âm (220),

trong đó giao diện (212) được làm thích ứng để nhận tín hiệu đa kênh được mã hóa hiện thời (107) và để nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất (MCH_PAR2),

trong đó bộ giải mã kênh (202) được làm thích ứng để giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu được tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) của khung hiện thời,

trong đó bộ xử lý đa kênh (204) được làm thích ứng để lựa chọn cặp được lựa chọn thứ nhất gồm hai kênh được giải mã (D1, D2) từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) phụ thuộc vào các tham số đa kênh thứ nhất (MCH_PAR2),

trong đó bộ xử lý đa kênh (204) được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý (P1*,P2*) dựa trên cặp được lựa chọn thứ nhất gồm hai kênh được giải mã (D1, D2) để thu được tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã (D3, P1*, P2*),

trong đó, trước khi bộ xử lý đa kênh (204) tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý (P1*,P2*) dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2), môđun điền đầy nhiễu âm (220) được làm thích ứng để nhận biết ít nhất một trong số hai kênh của cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2), một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và để tạo ra kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải tất cả ba hoặc nhiều hơn ba kênh đầu ra âm thanh

trước, và để điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiều âm được tạo ra sử dụng các vạch phổ của kênh trộn, trong đó môđun điền đầy nhiều âm (220) được làm thích ứng để lựa chọn hai hoặc nhiều hơn hai kênh đầu ra âm thanh trước mà được sử dụng để tạo ra kênh trộn từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào thông tin phụ.

2. Thiết bị (201) theo điểm 1,

trong đó môđun điền đầy nhiều âm (220) được làm thích ứng để tạo kênh trộn sử dụng một cách chính xác hai kênh đầu ra âm thanh trước trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước như là hai hoặc nhiều hơn hai trong số ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước;

trong đó môđun điền đầy nhiều âm (220) được làm thích ứng để lựa chọn một cách chính xác hai kênh đầu ra âm thanh trước từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào thông tin phụ.

3. Thiết bị (201) theo điểm 2,

trong đó môđun điền đầy nhiều âm (220) được làm thích ứng để tạo kênh trộn sử dụng một cách chính xác hai kênh đầu ra âm thanh trước dựa trên công thức

$$D_{ch} = (\hat{O}_1 + \hat{O}_2) \cdot d \quad \text{hoặc dựa trên công thức}$$

$$D_{ch} = (\hat{O}_1 - \hat{O}_2) \cdot d$$

trong đó D_{ch} là kênh trộn,

trong đó $\hat{O}_1$ là kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước,

trong đó $\hat{O}_2$ là kênh thứ hai trong số chính xác hai kênh đầu ra âm thanh trước, mà khác với kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước, và

trong đó d là vô hướng dương, thực.

4. Thiết bị (201) theo điểm 2,

trong đó môđun điền đầy nhiễu âm (220) được làm thích ứng để tạo kênh trộn sử dụng một cách chính xác hai kênh đầu ra âm thanh trước dựa trên công thức

$$\hat{I}_{ch} = (\cos \alpha \cdot \hat{O}_1 + \sin \alpha \cdot \hat{O}_2) \cdot d \quad \text{hoặc dựa trên công thức}$$

$$\hat{I}_{ch} = (-\sin \alpha \cdot \hat{O}_1 + \cos \alpha \cdot \hat{O}_2) \cdot d$$

trong đó $\hat{I}_{ch}$ là kênh trộn,

trong đó $\hat{O}_1$ là kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước,

trong đó $\hat{O}_2$ là kênh thứ hai trong số chính xác hai kênh đầu ra âm thanh trước, mà khác với kênh thứ nhất trong số chính xác hai kênh đầu ra âm thanh trước, và

trong đó α là góc quay.

5. Thiết bị (201) theo điểm 4,

trong đó thông tin phụ là thông tin phụ hiện thời được gán cho khung hiện thời,

trong đó giao diện (212) được làm thích ứng để nhận thông tin phụ trước được gán cho khung trước, trong đó thông tin phụ trước bao gồm góc trước,

trong đó giao diện (212) được làm thích ứng để nhận thông tin phụ hiện thời bao gồm góc hiện thời, và

trong đó môđun điền đầy nhiễu âm (220) được làm thích ứng để sử dụng góc hiện thời của thông tin phụ hiện thời như là góc quay α , và được làm thích ứng để không sử dụng góc trước của thông tin phụ trước như là góc quay α .

6. Thiết bị (201) theo một điểm trong số các điểm từ điểm 2 đến 5, trong đó môđun điền đầy nhiễu âm (220) được làm thích ứng để lựa chọn một cách chính xác hai kênh đầu ra âm thanh trước từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước phụ thuộc vào các tham số đa kênh thứ nhất (MCH_PAR2).

7. Thiết bị (201) theo một điểm trong số các điểm từ 2 đến 6,

trong đó giao diện (212) được làm thích ứng để nhận tín hiệu đa kênh được mã hóa hiện thời (107) và để nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất

(MCH_PAR2) và các tham số đa kênh thứ hai (MCH_PAR1),

trong đó bộ xử lý đa kênh (204) được làm thích ứng để lựa chọn cặp được lựa chọn thứ hai gồm hai kênh được giải mã ($P1^*$, D3) từ tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã (D3, $P1^*$, $P2^*$) phụ thuộc vào các tham số đa kênh thứ hai (MCH_PAR1), ít nhất một kênh ($P1^*$) của cặp được lựa chọn thứ hai gồm hai kênh ($P1^*$, D3) là một kênh của nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý ($P1^*$, $P2^*$), và

trong đó bộ xử lý đa kênh (204) được làm thích ứng để tạo nhóm thứ hai gồm hai hoặc nhiều hơn hai kênh được xử lý ($P3^*$, $P4^*$) dựa trên cặp được lựa chọn thứ hai đã nêu gồm hai kênh được giải mã ($P1^*$, D3) để cập nhật thêm tập hợp được cập nhật của ba hoặc nhiều hơn ba kênh được giải mã.

8. Thiết bị (201) theo điểm 7,

trong đó, bộ xử lý đa kênh (204) được làm thích ứng để tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý ($P1^*$, $P2^*$) bằng cách tạo ra nhóm thứ nhất gồm chính xác hai kênh được xử lý ($P1^*$, $P2^*$) dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2);

trong đó bộ xử lý đa kênh (204) được làm thích ứng để thay thế cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2) trong tập hợp của ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) bằng nhóm thứ nhất gồm chính xác hai kênh được xử lý ($P1^*$, $P2^*$) để thu được tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã (D3, $P1^*$, $P2^*$);

trong đó bộ xử lý đa kênh (204) được làm thích ứng để tạo nhóm thứ hai gồm hai hoặc nhiều hơn hai kênh được xử lý ($P3^*$, $P4^*$) bằng cách tạo nhóm thứ hai gồm chính xác hai kênh được xử lý ($P3^*$, $P4^*$) dựa trên cặp được lựa chọn thứ hai đã nêu gồm hai kênh được giải mã ($P1^*$, D3), và

trong đó bộ xử lý đa kênh (204) được làm thích ứng để thay thế cặp được lựa chọn thứ hai gồm hai kênh được giải mã ($P1^*$, D3) trong tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã (D3, $P1^*$, $P2^*$) bởi nhóm thứ hai gồm chính xác hai kênh được xử lý ($P3^*$, $P4^*$) để cập nhật thêm tập hợp được cập nhật gồm ba

hoặc nhiều hơn ba kênh được giải mã.

9. Thiết bị (201) theo điểm 8,

trong đó các tham số đa kênh thứ nhất (MCH_PAR2) biểu thị hai kênh được giải mã (D1, D2) từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã;

trong đó bộ xử lý đa kênh (204) được làm thích ứng để lựa chọn cặp được lựa chọn thứ nhất gồm hai kênh được giải mã (D1, D2) từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) bằng việc lựa chọn hai kênh được giải mã (D1, D2) được biểu thị bởi các tham số đa kênh thứ nhất (MCH_PAR2);

trong đó các tham số đa kênh thứ hai (MCH_PAR1) biểu thị hai kênh được giải mã (P1*, D3) từ tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã;

trong đó bộ xử lý đa kênh (204) được làm thích ứng để lựa chọn cặp được lựa chọn thứ hai gồm hai kênh được giải mã (P1*, D3) từ tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã (D3, P1*, P2*) bằng cách lựa chọn hai kênh được giải mã (P1*, D3) được biểu thị bởi các tham số đa kênh thứ hai (MCH_PAR1).

10. Thiết bị (201) theo điểm 9,

trong đó thiết bị (201) được làm thích ứng để gán ký hiệu nhận biết từ tập hợp gồm các ký hiệu nhận biết cho mỗi kênh đầu ra âm thanh trước của ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, sao cho mỗi kênh đầu ra âm thanh trước của ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước được gán cho chính xác một ký hiệu nhận biết của tập hợp gồm các ký hiệu nhận biết, và sao cho mỗi ký hiệu nhận biết của tập hợp gồm các ký hiệu nhận biết được gán cho chính xác một kênh đầu ra âm thanh trước của ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước,

trong đó thiết bị (201) được làm thích ứng để gán ký hiệu nhận biết từ tập hợp đã nêu gồm các ký hiệu nhận biết cho mỗi kênh của tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3), sao cho mỗi kênh của tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã được gán cho chính xác một ký hiệu nhận biết của tập hợp gồm các ký hiệu nhận biết và sao cho mỗi ký hiệu nhận biết của tập hợp gồm các ký hiệu nhận biết được gán cho chính xác một kênh của tập hợp gồm ba hoặc nhiều hơn

ba kênh được giải mã (D1, D2, D3),

trong đó các tham số đa kênh thứ nhất (MCH_PAR2) biểu thị cặp thứ nhất gồm hai ký hiệu nhận biết của tập hợp gồm ba hoặc nhiều hơn ba ký hiệu nhận biết,

trong đó bộ xử lý đa kênh (204) được làm thích ứng để lựa chọn cặp được lựa chọn thứ nhất gồm hai kênh được giải mã (D1, D2) từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) bằng việc lựa chọn hai kênh được giải mã (D1, D2) được gán cho hai ký hiệu nhận biết của cặp thứ nhất gồm hai ký hiệu nhận biết;

trong đó thiết bị (201) được làm thích ứng để gán ký hiệu nhận biết thứ nhất trong số hai ký hiệu nhận biết của cặp thứ nhất gồm hai ký hiệu nhận biết cho kênh được xử lý thứ nhất của nhóm thứ nhất gồm chính xác hai kênh được xử lý (P1*,P2*), và trong đó thiết bị (201) được làm thích ứng để gán ký hiệu nhận biết thứ hai trong số hai ký hiệu nhận biết của cặp thứ nhất gồm hai ký hiệu nhận biết cho kênh được xử lý thứ hai của nhóm thứ nhất gồm chính xác hai kênh được xử lý (P1*,P2*).

11. Thiết bị (201) theo điểm 10,

trong đó các tham số đa kênh thứ hai (MCH_PAR1) biểu thị cặp thứ hai gồm hai ký hiệu của tập hợp gồm ba hoặc nhiều hơn ba ký hiệu nhận biết,

trong đó bộ xử lý đa kênh (204) được làm thích ứng để lựa chọn cặp được lựa chọn thứ hai gồm hai kênh được giải mã (P1*, D3) từ tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã (D3, P1*, P2*) bằng việc lựa chọn hai kênh được giải mã (D3, P1*) được gán cho hai ký hiệu nhận biết của cặp thứ hai gồm hai ký hiệu nhận biết;

trong đó thiết bị (201) được làm thích ứng để gán ký hiệu nhận biết thứ nhất trong số hai ký hiệu nhận biết của cặp thứ hai gồm hai ký hiệu nhận biết cho kênh được xử lý thứ nhất của nhóm thứ hai gồm chính xác hai kênh được xử lý (P3*,P4*), và trong đó thiết bị (201) được làm thích ứng để gán ký hiệu nhận biết thứ hai trong số hai ký hiệu nhận biết của cặp thứ hai gồm hai ký hiệu nhận biết cho kênh được xử lý thứ hai của nhóm thứ hai gồm chính xác hai kênh được xử lý (P3*,P4*).

12. Thiết bị (201) theo điểm 10 hoặc điểm 11,

trong đó các tham số đa kênh thứ nhất (MCH_PAR2) biểu thị cặp thứ nhất đã nêu gồm hai ký hiệu nhận biết của tập hợp gồm ba hoặc nhiều hơn ba ký hiệu nhận biết, và

trong đó môđun điền đầy nhiễu âm (220) được làm thích ứng để lựa chọn chính xác hai kênh đầu ra âm thanh trước từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước bằng việc lựa chọn hai kênh đầu ra âm thanh trước được gán cho hai ký hiệu nhận biết của cặp thứ nhất đã nêu gồm hai ký hiệu nhận biết.

13. Thiết bị (201) theo một trong số các điểm đã nêu, trong đó, trước khi bộ xử lý đa kênh (204) tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý (P1*,P2*) dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2), môđun điền đầy nhiễu âm (220) được làm thích ứng để nhận biết ít nhất một trong số hai kênh của cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã (D1, D2), một hoặc nhiều băng thừa số tỉ lệ là một hoặc nhiều băng tần mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và để tạo kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải tất cả ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và để điền đầy các vạch phổ của một hoặc nhiều băng thừa số tỉ lệ, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiễu âm được tạo ra sử dụng các vạch phổ của kênh trộn phụ thuộc vào mỗi băng thừa số tỉ lệ trong số một hoặc nhiều băng thừa số tỉ lệ mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không.

14. Thiết bị (201) theo điểm 13,

trong đó giao diện nhận (212) được tạo cấu hình để nhận thừa số tỉ lệ của mỗi băng trong số một hoặc nhiều băng thừa số tỉ lệ, và

trong đó thừa số tỉ lệ của mỗi băng thừa số tỉ lệ trong số một hoặc nhiều băng thừa số tỉ lệ đã nêu biểu thị năng lượng của các vạch phổ của thừa số của băng thừa số tỉ lệ đã nêu trước khi lượng tử hóa, và

trong đó môđun điền đầy nhiễu âm (220) được làm thích ứng để tạo nhiễu âm cho mỗi băng thừa số tỉ lệ, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, sao cho năng lượng của các vạch phổ sau khi bổ sung nhiễu âm vào một băng

tần trong số các băng tần tương ứng với năng lượng được biểu thị bằng thừa số tỉ lệ cho băng thừa số tỉ lệ đã nêu.

15. Hệ thống mã hóa và giải mã bao gồm:

thiết bị (100) mã hóa tín hiệu đa kênh (101) có ít nhất ba kênh (CH1:CH3), và
thiết bị (201) giải mã theo một điểm trong số các điểm từ 1 đến 14,
trong đó thiết bị (201) giải mã được tạo kết cấu để nhận tín hiệu đa kênh được mã hóa (107), được tạo ra bởi thiết bị (100) mã hóa từ thiết bị mã hóa (100),
trong đó thiết bị (100) mã hóa tín hiệu đa kênh (101) bao gồm:

bộ xử lý phép lặp (102) được làm thích ứng để tính toán, trong bước lặp thứ nhất, các giá trị tương quan liên kênh giữa mỗi cặp của ít nhất ba kênh (CH:CH3), để lựa chọn, trong bước lặp thứ nhất, cặp có giá trị cao nhất hoặc có giá trị cao hơn ngưỡng, và để xử lý cặp được lựa chọn sử dụng phép toán xử lý đa kênh (110, 112) để suy ra các tham số đa kênh ban đầu (MCH_PAR1) cho cặp được lựa chọn và để suy ra các kênh được xử lý thứ nhất (P1, P2),

trong đó bộ xử lý phép lặp (102) được làm thích ứng để thực hiện việc tính toán, lựa chọn và xử lý trong bước lặp thứ hai sử dụng ít nhất một kênh trong số các kênh được xử lý (P1) để suy ra các tham số đa kênh nữa (MCH_PAR2) và các kênh được xử lý thứ hai (P3, P4);

bộ mã hóa kênh được làm thích ứng để mã hóa các kênh (P2:P4) từ việc xử lý phép lặp được thực hiện bởi bộ xử lý phép lặp (104) để thu các kênh được mã hóa (E1:E3); và

giao diện đầu ra (106) được làm thích ứng để tạo tín hiệu đa kênh được mã hóa (107) có các kênh được mã hóa (E1:E3), các tham số đa kênh ban đầu và các tham số đa kênh khác (MCH_PAR1, MCH_PAR2) và có thông tin biểu thị liệu thiết bị giải mã có điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó các vạch phổ được lượng tử hóa bằng không, với nhiễu âm được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã hay không.

16. Hệ thống mã hóa và giải mã theo điểm 15,

trong đó mỗi tham trong số các tham số đa kênh ban đầu và các tham số đa kênh nữa (MCH_PAR1, MCH_PAR2) biểu thị một cách chính xác hai kênh, mỗi kênh trong số chính xác hai kênh là một trong số các kênh được mã hóa (E1:E3) hoặc là một trong số các kênh được xử lý thứ nhất hoặc thứ hai (P1, P2, P3, P4) hoặc là một trong số ít nhất ba kênh (CH:CH3), và

trong đó giao diện đầu ra (106) của thiết bị (100) mã hóa tín hiệu đa kênh (101) được làm thích ứng để tạo ra tín hiệu đa kênh (107), để thông tin biểu thị liệu thiết bị giải mã có điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, bao gồm thông tin mà biểu thị mỗi tham số trong số các tham số đa kênh và tham số đa kênh ban đầu (MCH_PAR1, MCH_PAR2) hay không, liệu đối với ít nhất một kênh trong số chính xác hai kênh mà được biểu thị bởi một tham số đã nêu trong số các tham số đa kênh ban đầu và tham số đa kênh khác (MCH_PAR1, MCH_PAR2), thiết bị giải mã có điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không của ít nhất một kênh đã nêu, với dữ liệu quang phổ được tạo ra dựa trên các kênh đầu ra âm thanh được giải mã trước mà đã được giải mã trước bởi thiết bị giải mã hay không

17. Phương pháp giải mã tín hiệu đa kênh được mã hóa trước của khung trước để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và giải mã tín hiệu đa kênh được mã hóa hiện thời (107) của khung hiện thời để thu được ba hoặc nhiều hơn ba kênh đầu ra âm thanh, phương pháp bao gồm:

nhận tín hiệu đa kênh được mã hóa hiện thời (107), và nhận thông tin phụ bao gồm các tham số đa kênh thứ nhất (MCH_PAR2);

giải mã tín hiệu đa kênh được mã hóa hiện thời của khung hiện thời để thu được tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) của khung hiện thời;

lựa chọn cặp được lựa chọn thứ nhất gồm hai kênh được giải mã (D1, D2) từ tập hợp gồm ba hoặc nhiều hơn ba kênh được giải mã (D1, D2, D3) phụ thuộc vào các tham số đa kênh thứ nhất (MCH_PAR2);

tạo ra nhóm thứ nhất gồm hai hoặc nhiều hơn hai kênh được xử lý ($P1^*, P2^*$) dựa trên cặp được lựa chọn thứ nhất gồm hai kênh được giải mã ($D1, D2$) để thu được tập hợp được cập nhật gồm ba hoặc nhiều hơn ba kênh được giải mã ($D3, P1^*, P2^*$);

trong đó, trước khi nhóm thứ nhất gồm hai hoặc nhiều kênh được xử lý ($P1^*, P2^*$) được tạo ra dựa trên cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã ($D1, D2$), các bước sau đây được thực hiện:

nhận biết ít nhất một trong số hai kênh của cặp được lựa chọn thứ nhất đã nêu gồm hai kênh được giải mã ($D1, D2$), một hoặc nhiều băng tần mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, và tạo kênh trộn sử dụng hai hoặc nhiều hơn hai, nhưng không phải tất cả ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước, và điền đầy các vạch phổ của một hoặc nhiều băng tần, mà trong đó tất cả các vạch phổ được lượng tử hóa bằng không, với nhiễu âm được tạo ra sử dụng các vạch phổ của kênh trộn, trong đó việc lựa chọn hai hoặc nhiều hơn hai kênh đầu ra âm thanh trước mà được sử dụng để tạo kênh trộn từ ba hoặc nhiều hơn ba kênh đầu ra âm thanh trước được thực hiện phụ thuộc vào thông tin phụ.

18. Vật ghi đọc được bằng máy tính bao gồm chương trình máy tính để thực hiện phương pháp theo điểm 17 khi chạy trên máy tính hoặc bộ xử lý tín hiệu.

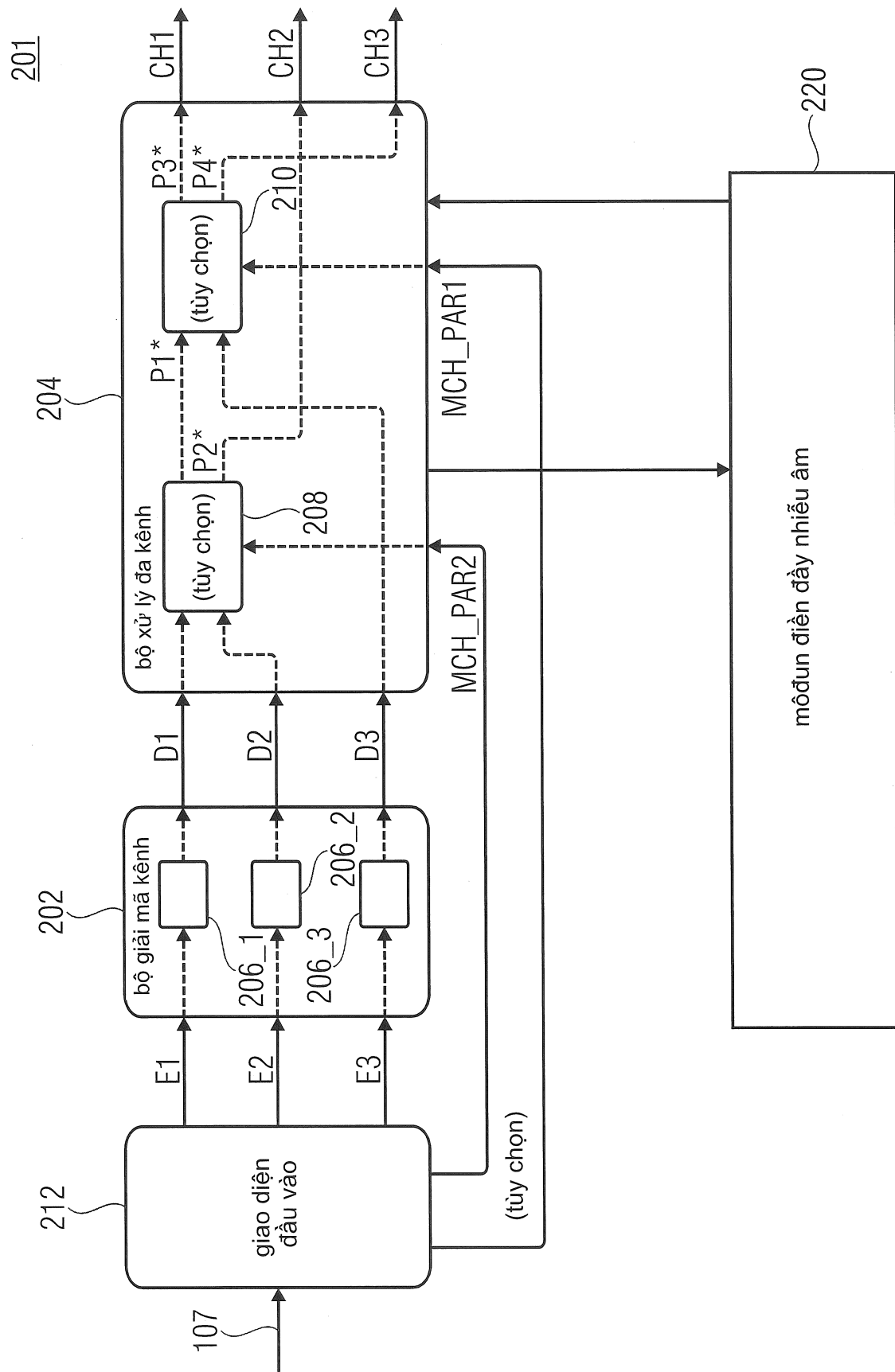


Fig. 1a

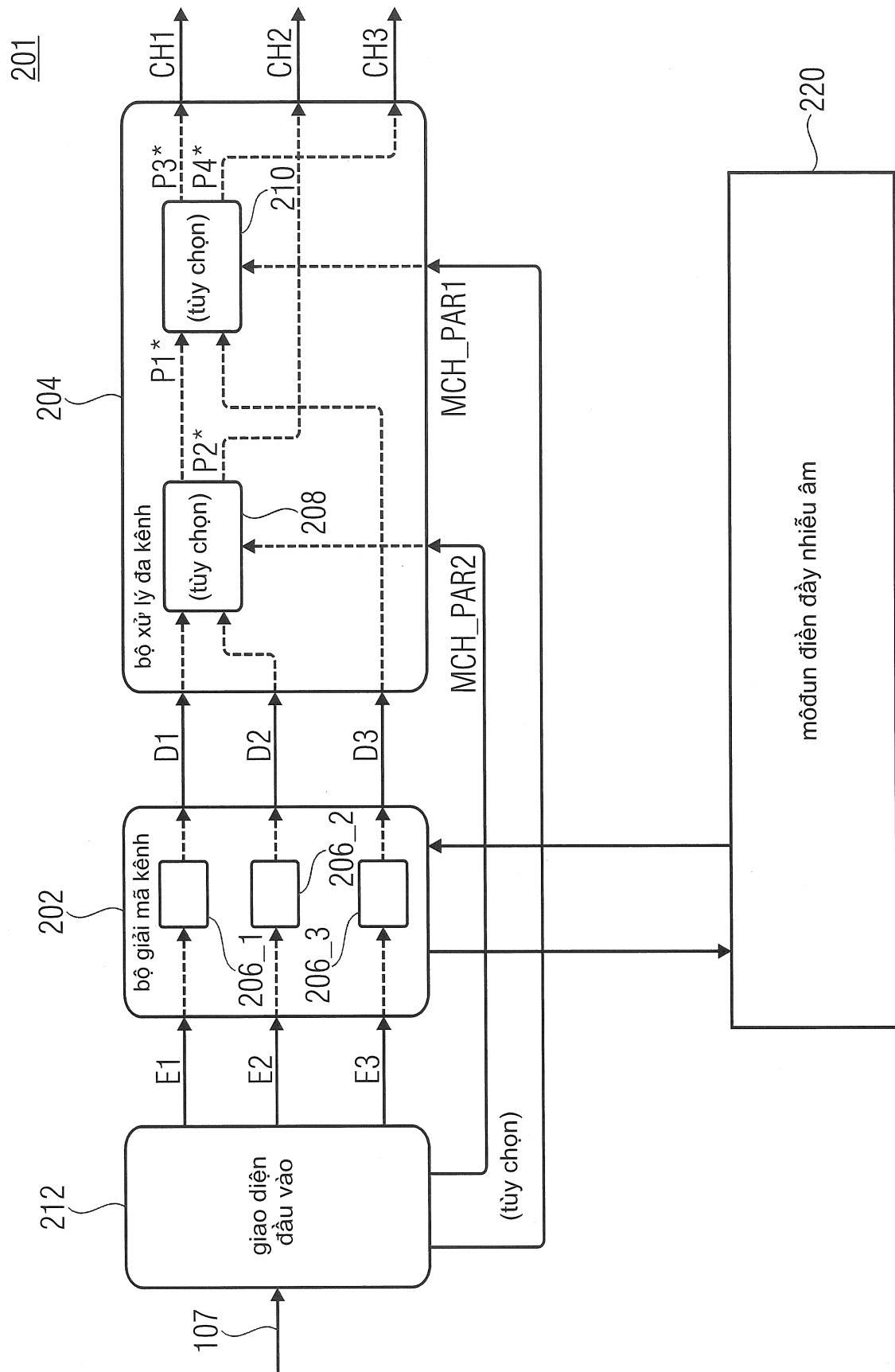


Fig. 1b

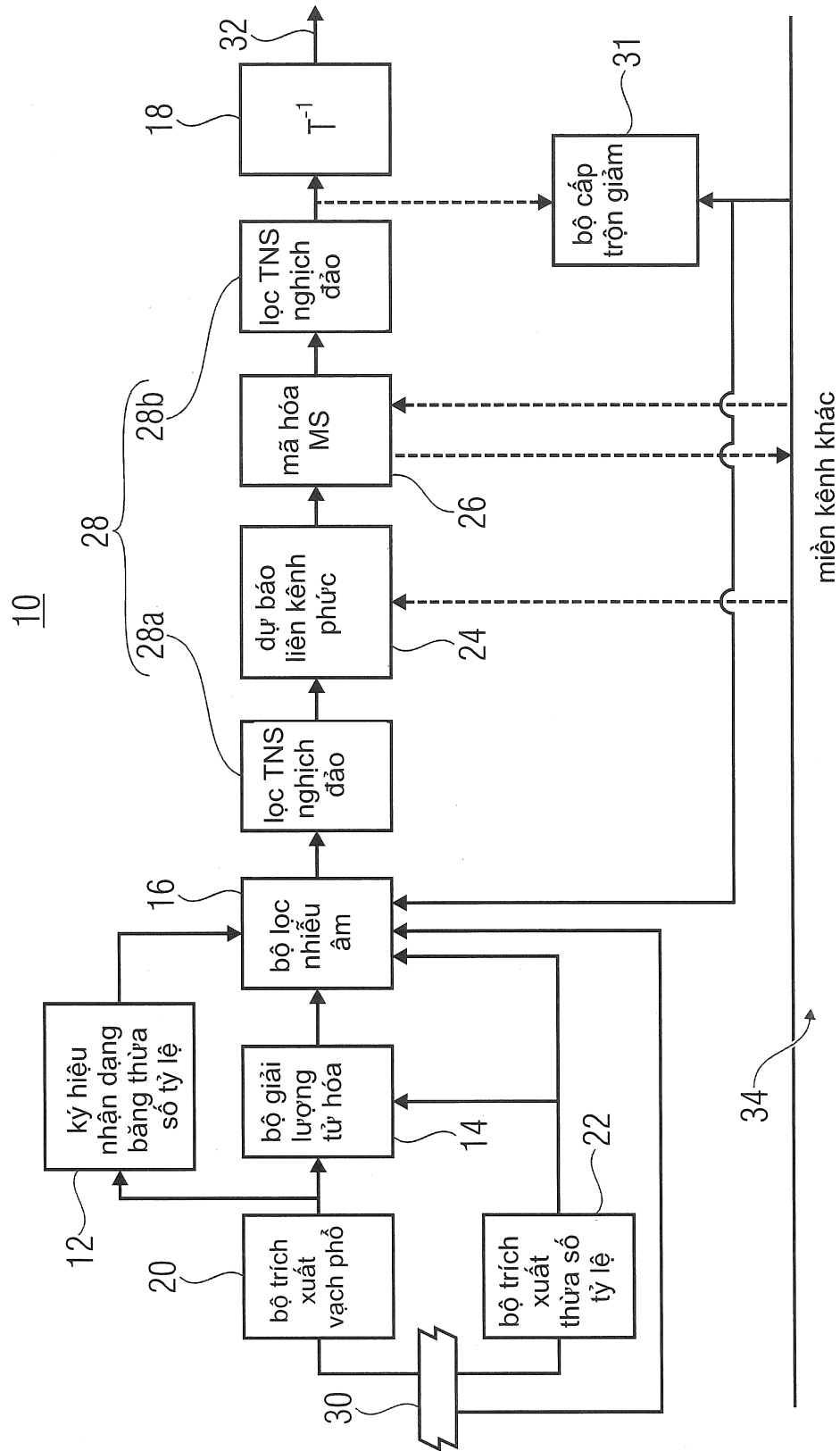


Fig. 2

4/17

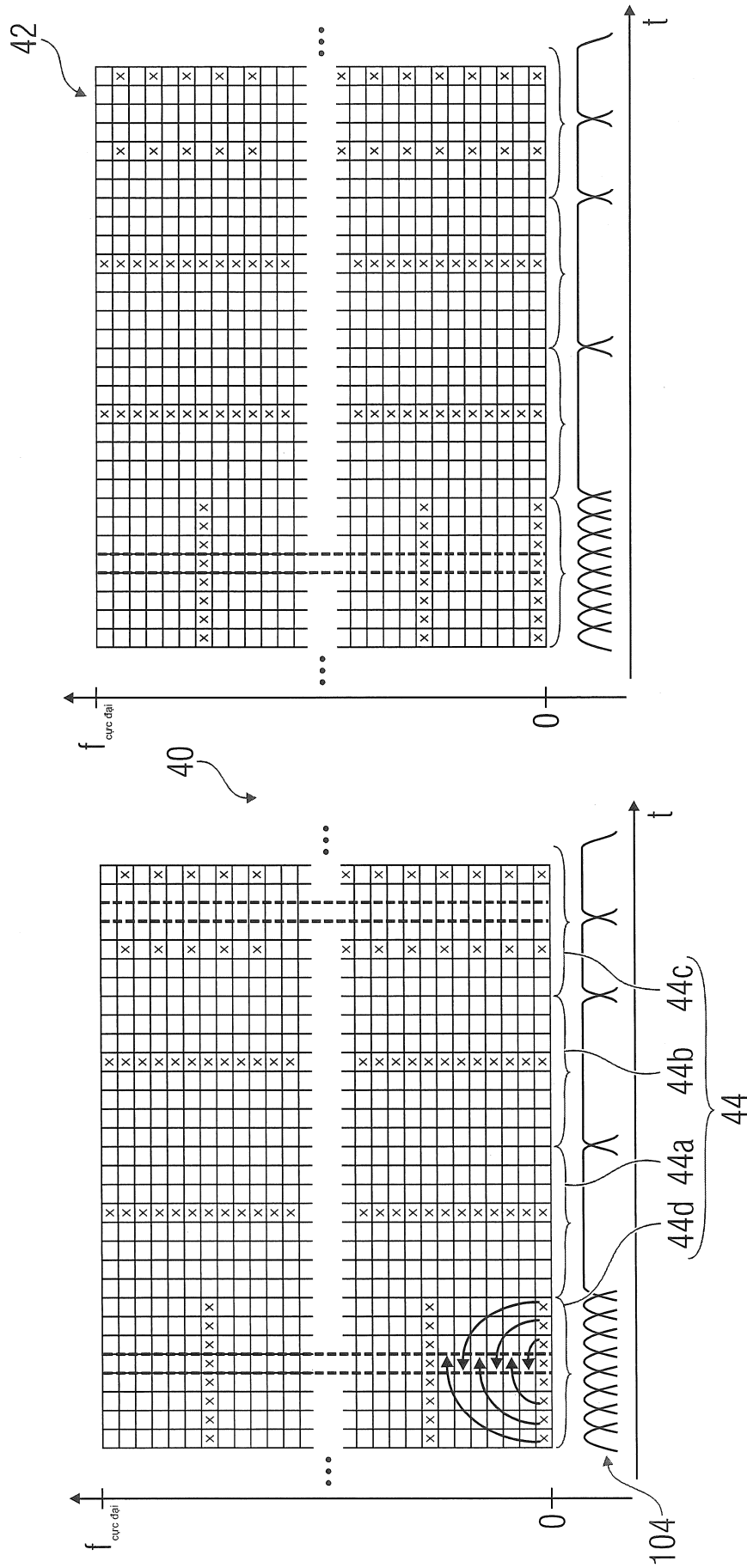


Fig. 3

5/17

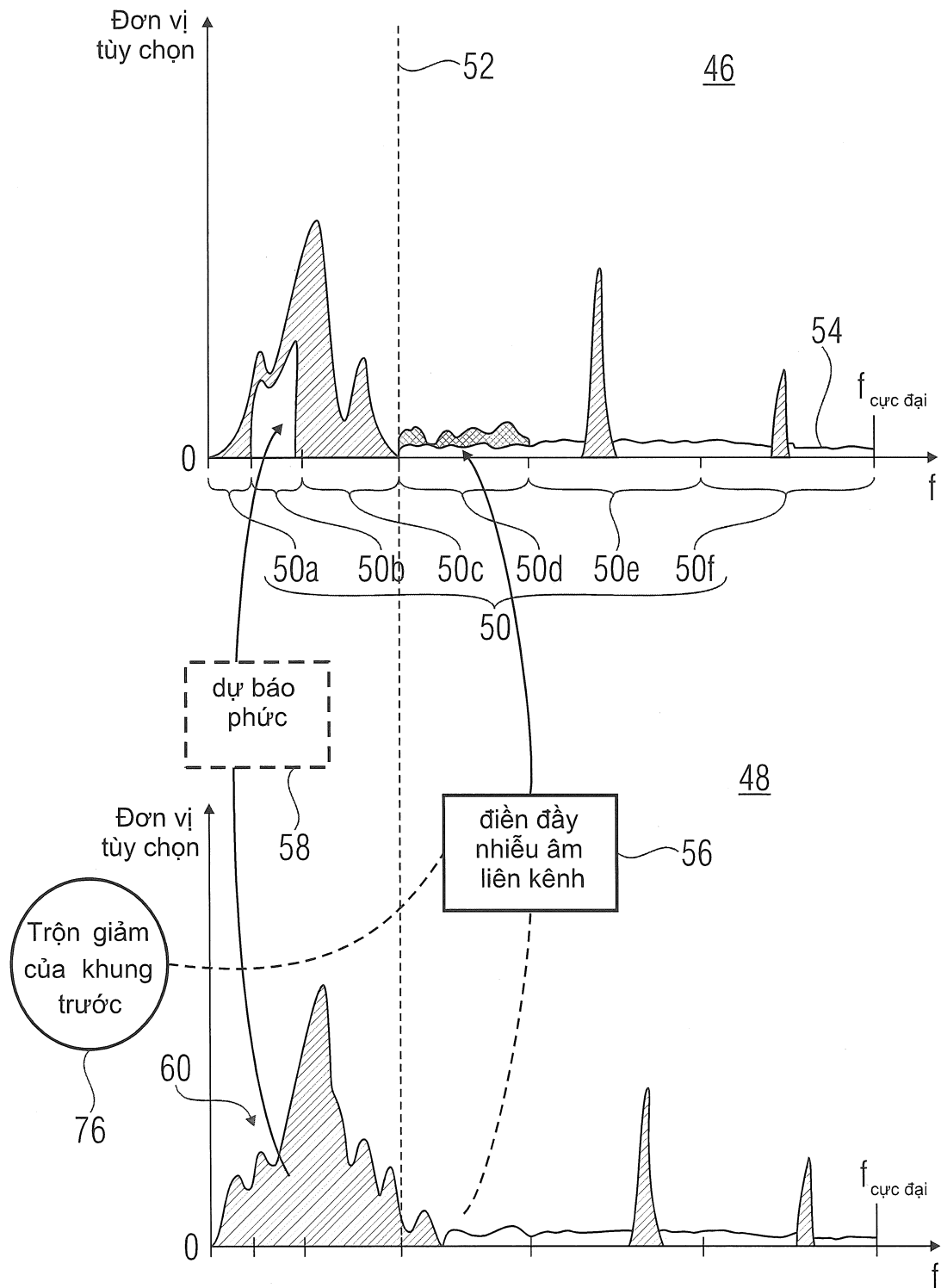


Fig. 4

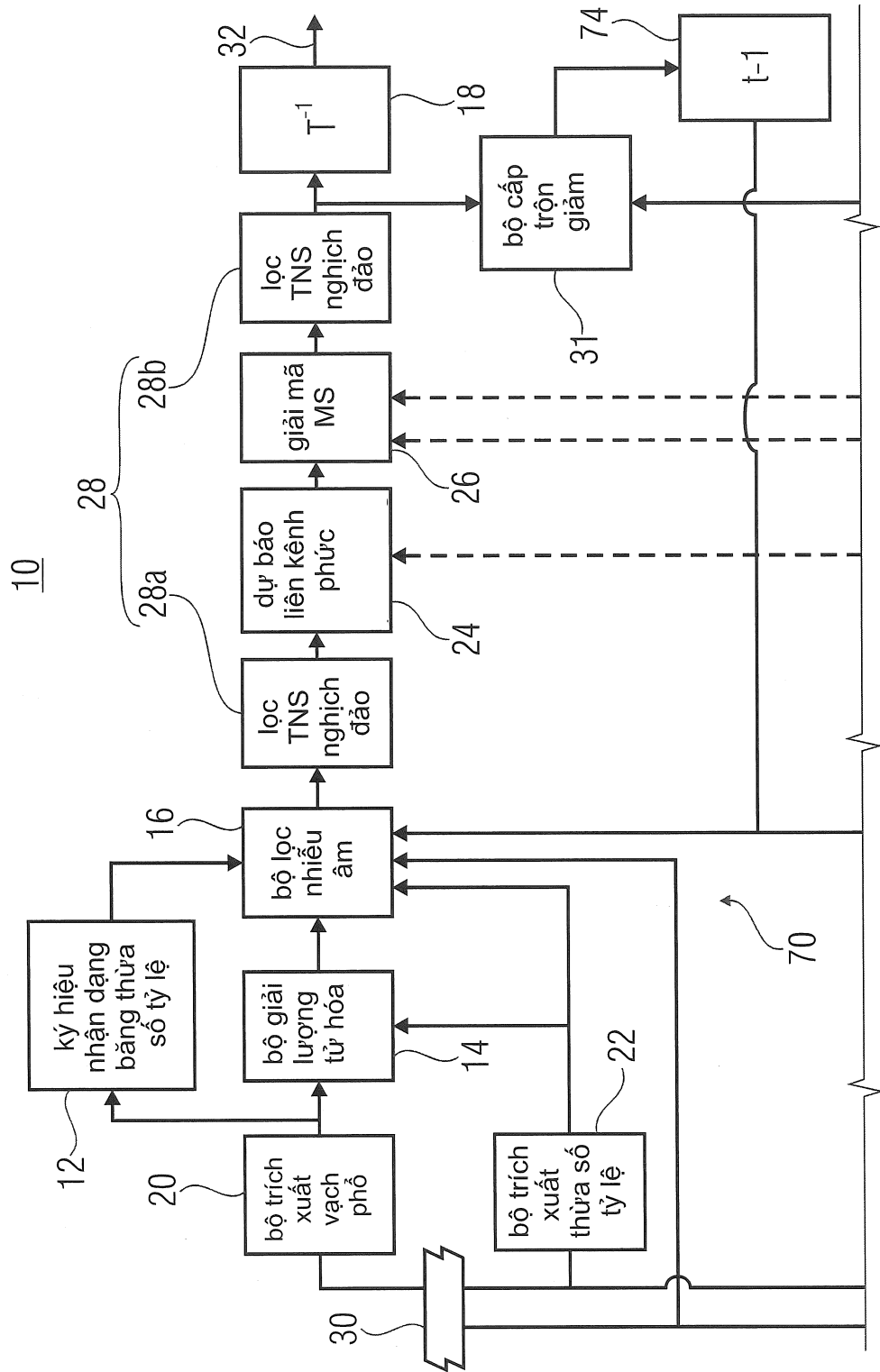


Fig. 5a

Fig. 5a

Fig. 5b

Fig. 5

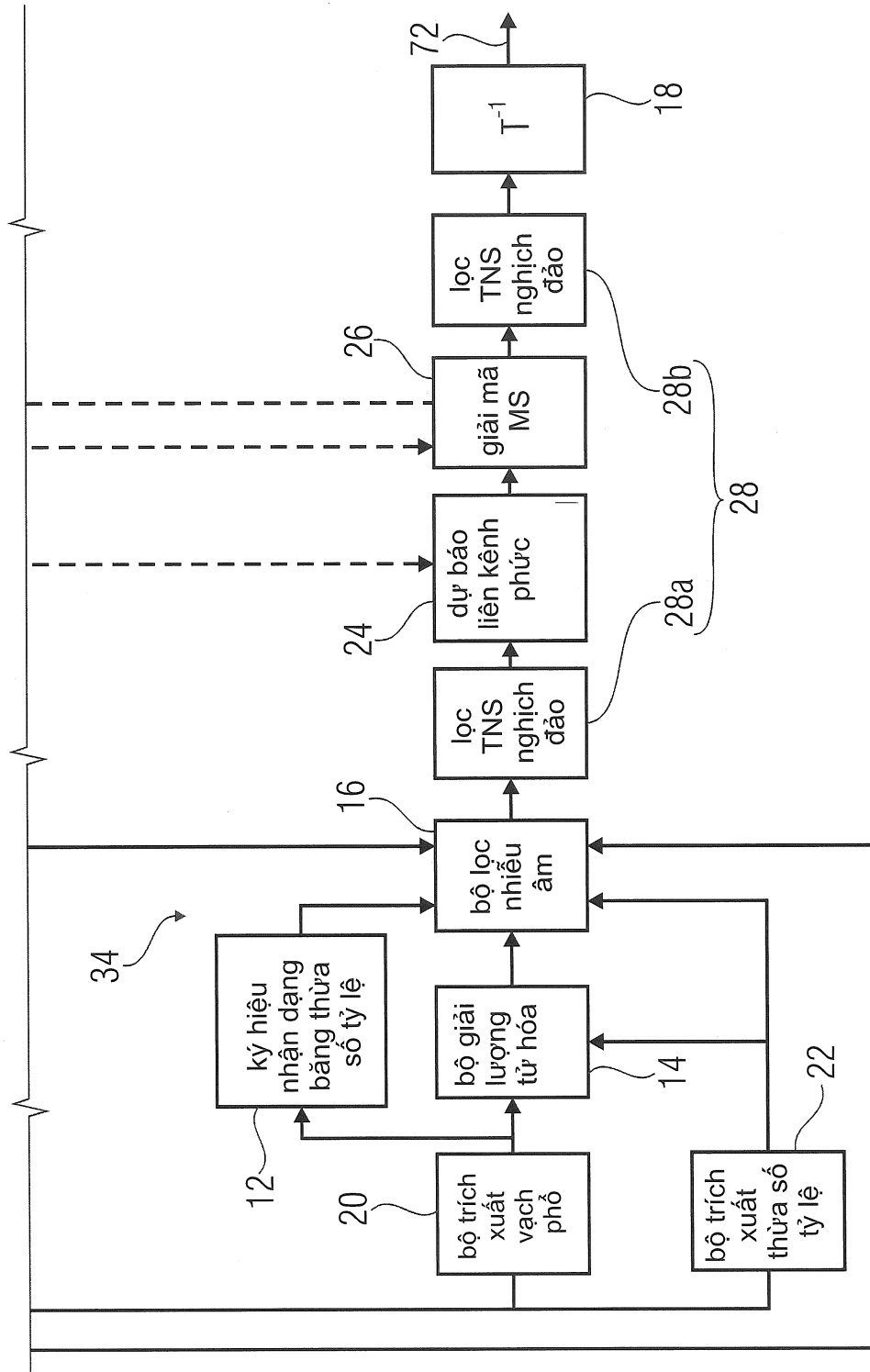


Fig. 5b

Fig. 5	Fig. 5a
	Fig. 5b

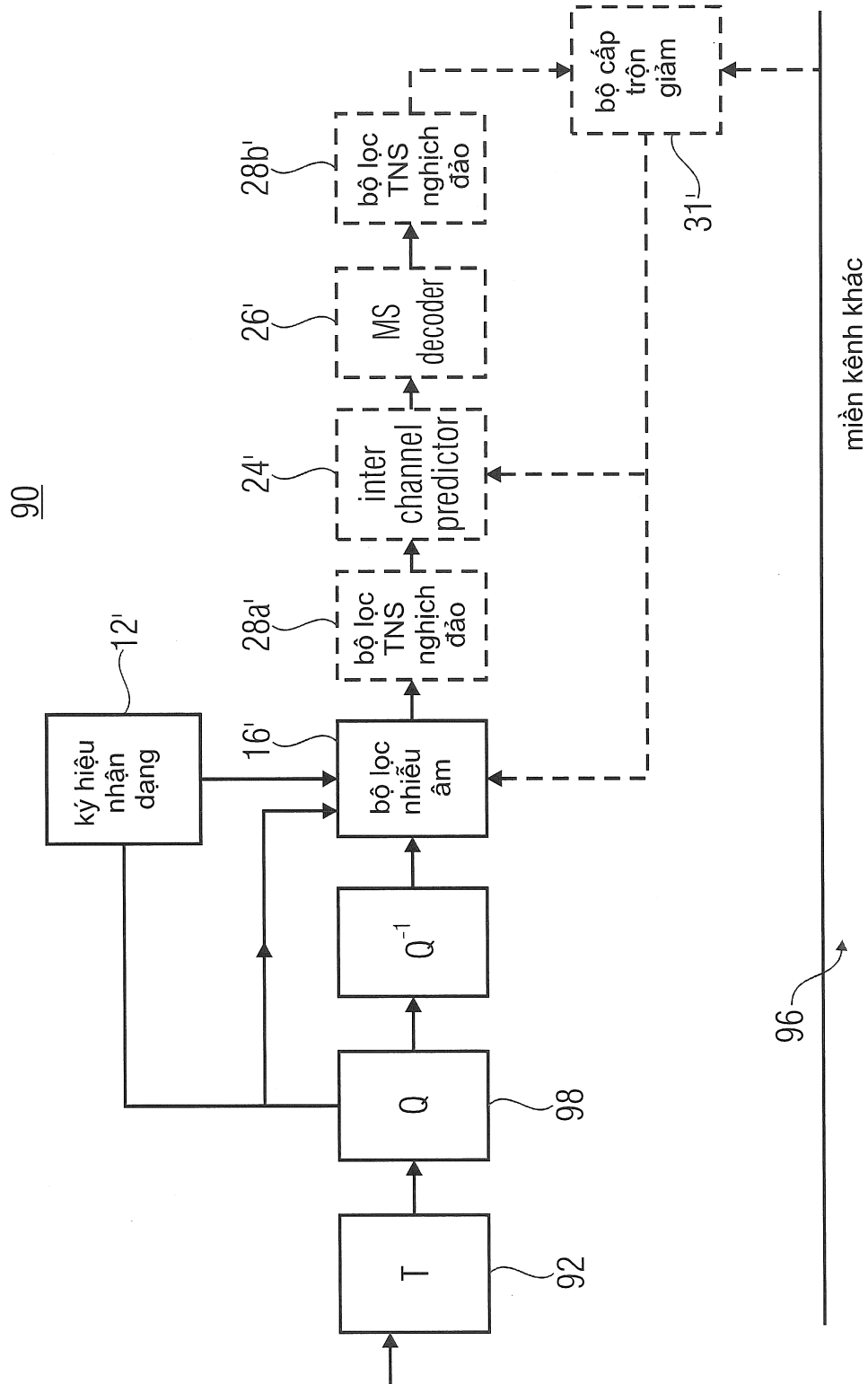


Fig. 6

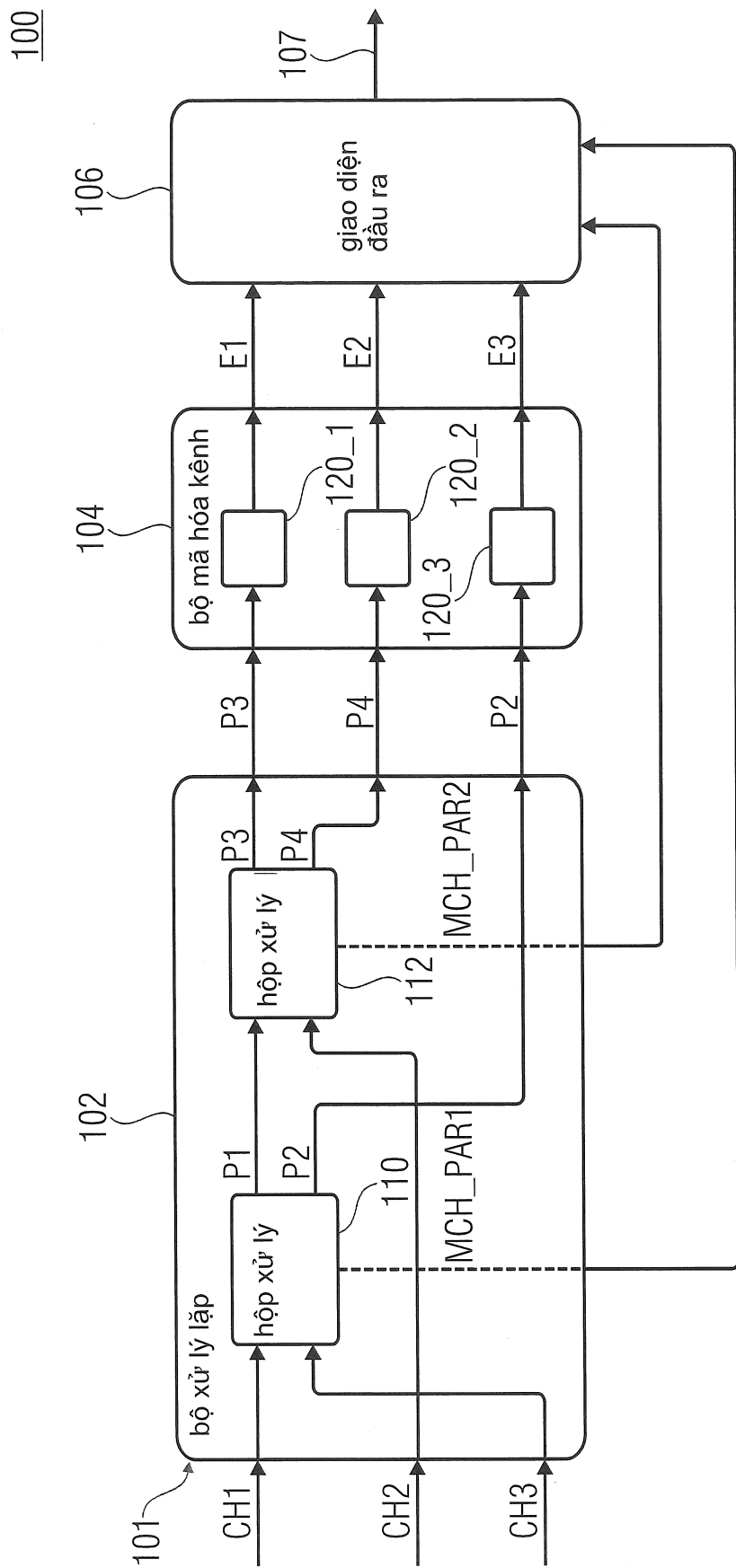


Fig. 7

10/17

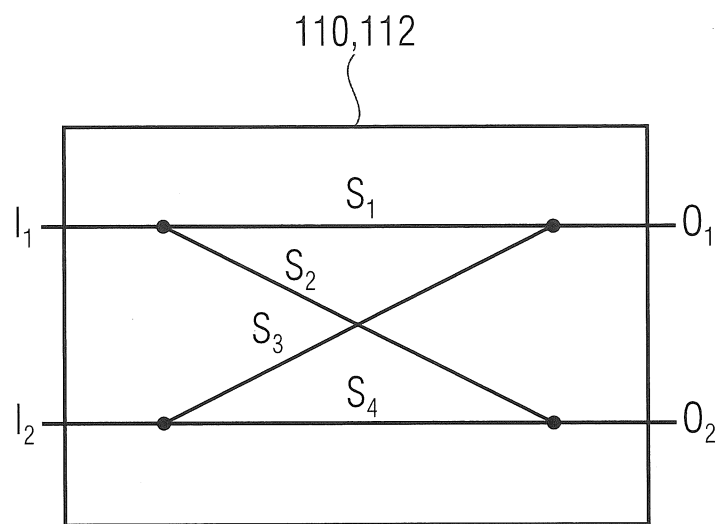


Fig. 8

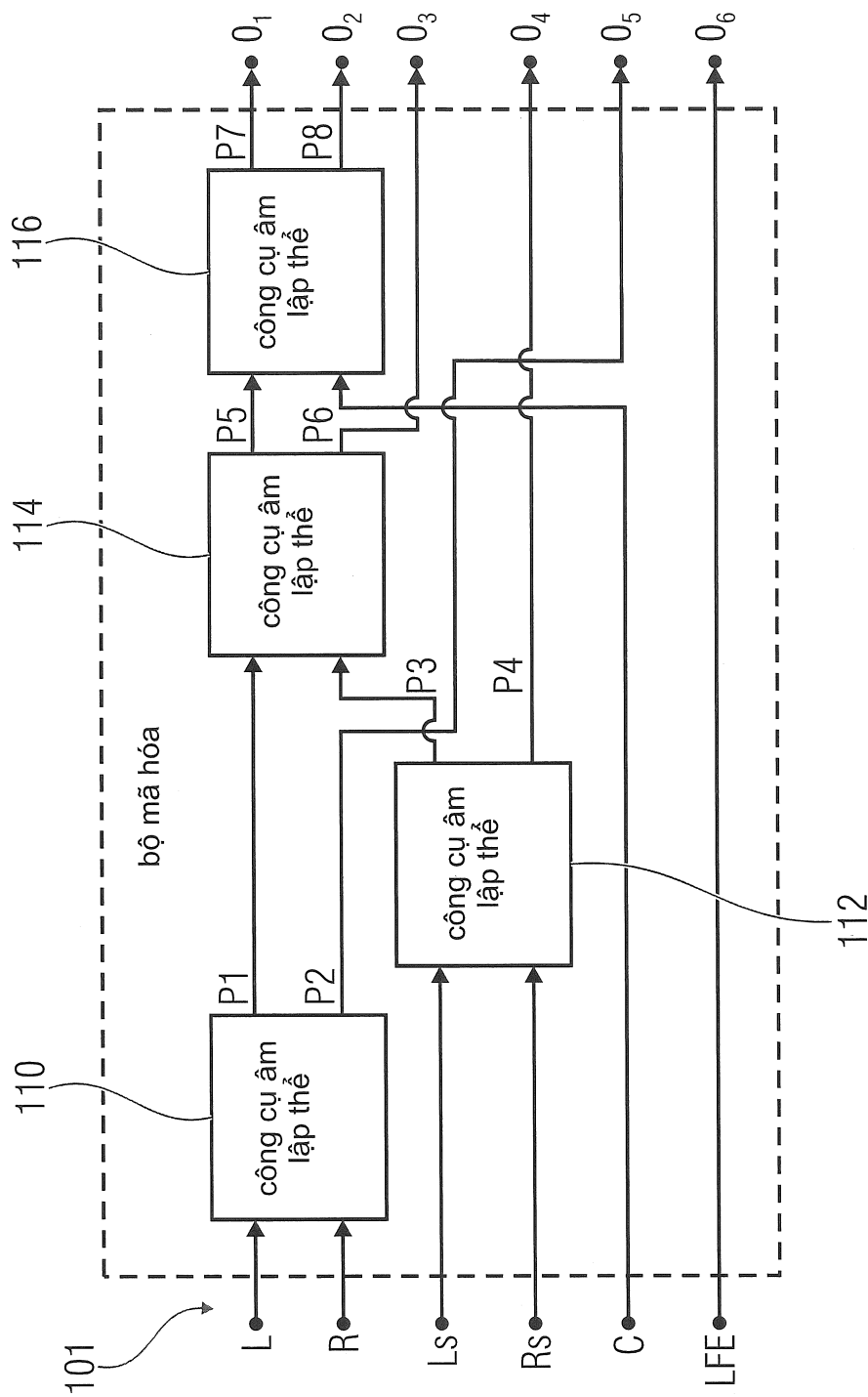


Fig. 9

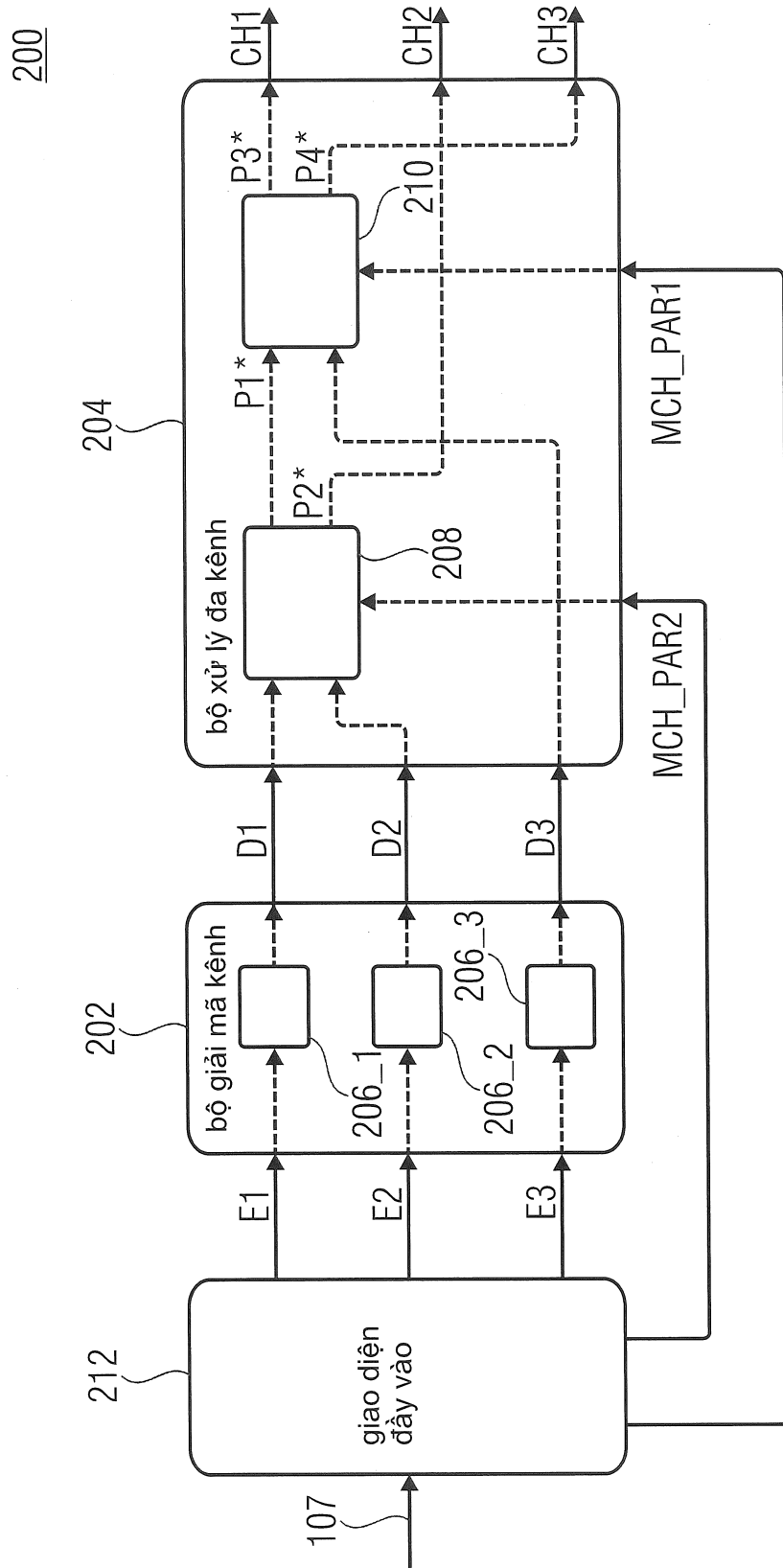


Fig. 10

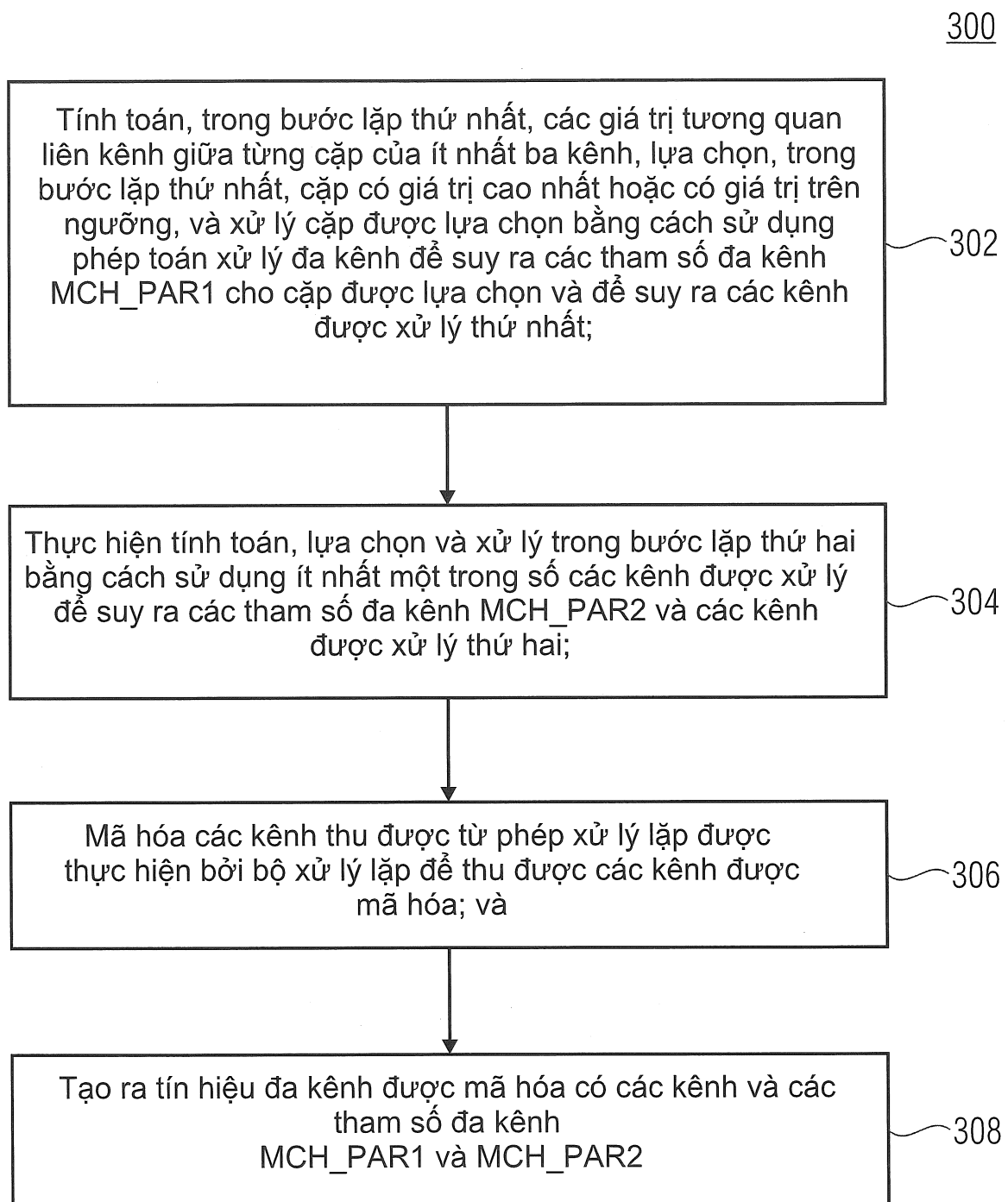


Fig. 11

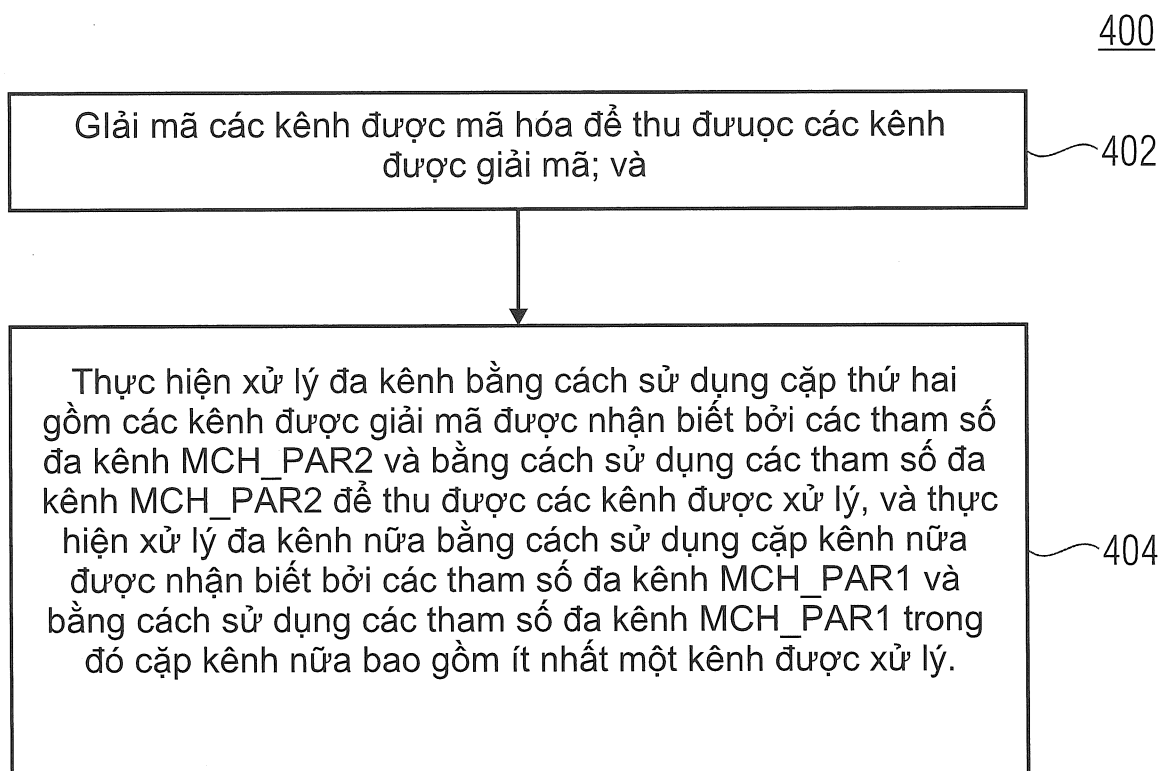


Fig. 12

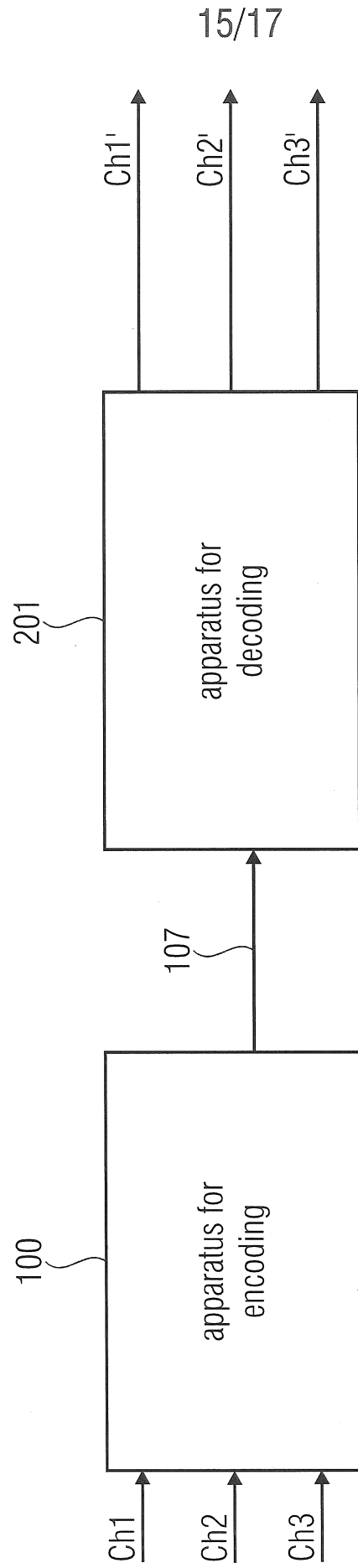


Fig. 13

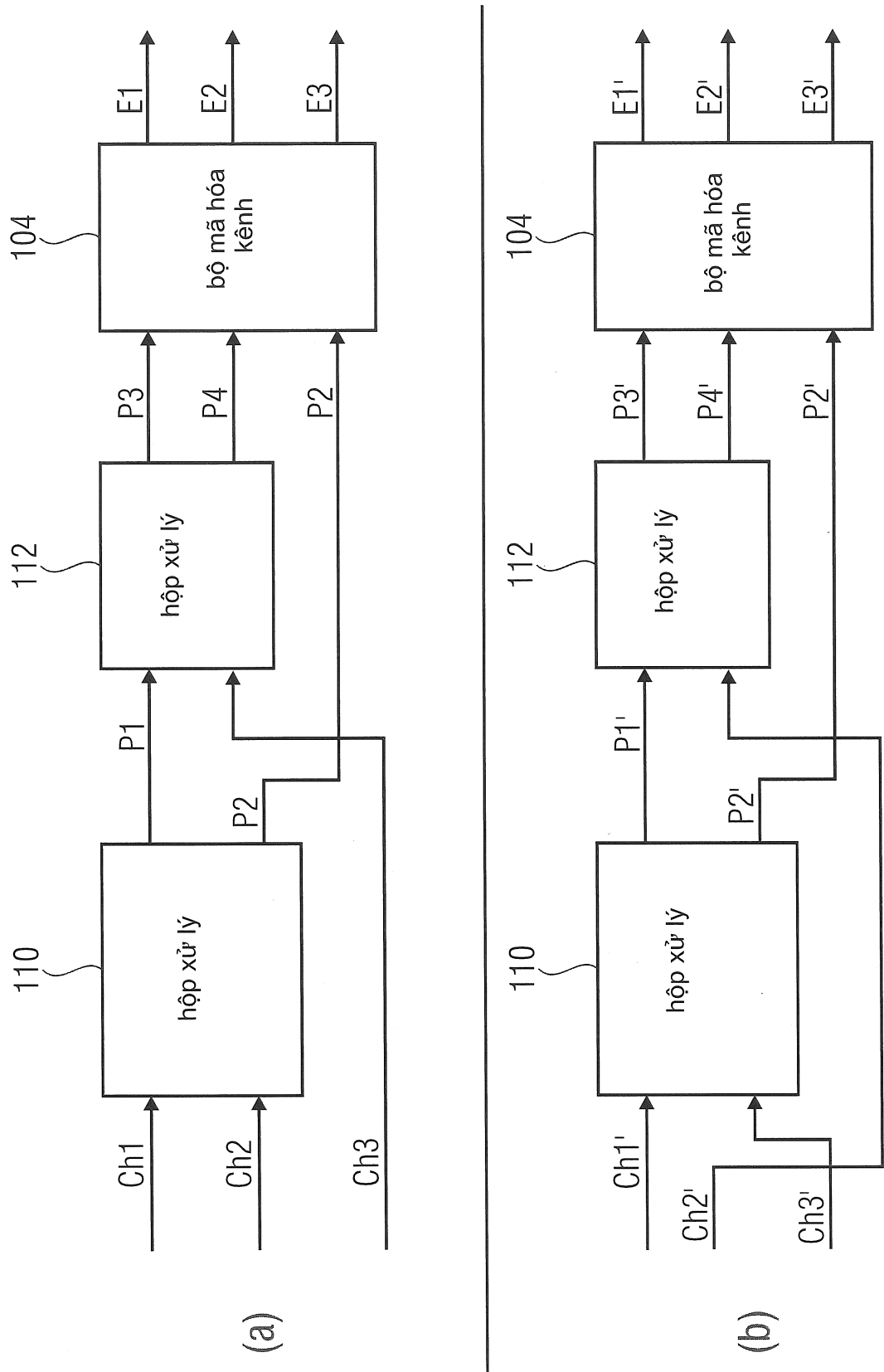


Fig. 14

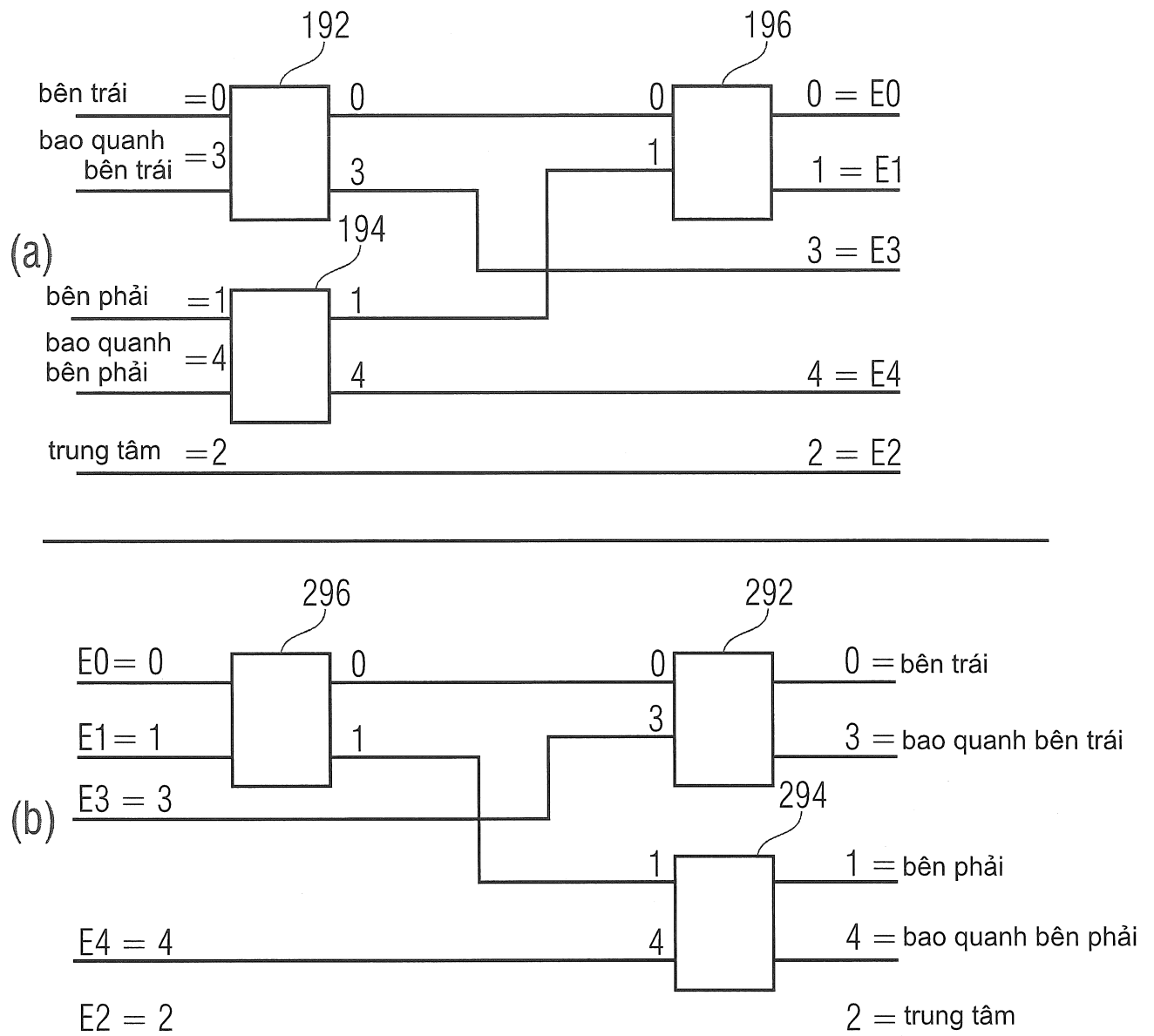


Fig. 15