



- (12) BẢN MÔ TẢ SÁNG CHẾ THUỘC BẰNG ĐỘC QUYỀN SÁNG CHẾ
(19) Cộng hòa xã hội chủ nghĩa Việt Nam (VN) (11) 
CỤC SỞ HỮU TRÍ TUỆ
1-0052171
(51)^{2020.01} H04N 19/124; H04N 19/70; H04N 19/186; H04N 19/137; H04N 19/18 (13) B

-
- (21) 1-2022-00180 (22) 24/06/2020
(86) PCT/KR2020/008126 24/06/2020 (87) WO2020/262909 30/12/2020
(30) 62/867,879 28/06/2019 US
(45) 27/10/2025 451 (43) 25/03/2022 408A
(73) LG ELECTRONICS INC. (KR)
128, Yeoui-daero, Yeongdeungpo-gu Seoul 07336, Korea
(72) PALURI, Seethal (IN); KIM, Seunghwan (KR); ZHAO, Jie (US).
(74) Công ty Luật TNHH T&G (TGVN)
-

- (54) PHƯƠNG PHÁP GIẢI MÃ VÀ MÃ HÓA ẢNH, PHƯƠNG TIỆN LƯU TRỮ PHI
CHUYỂN TIẾP ĐƯỢC ĐỌC BỞI MÁY TÍNH VÀ PHƯƠNG PHÁP TRUYỀN DỮ
LIỆU CHO ẢNH

(21) 1-2022-00180

(57) Sáng chế liên quan đến phương pháp giải mã ảnh được thực hiện bởi thiết bị giải mã bao gồm các bước: thu được cờ chỉ báo liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không; thu được dữ liệu tham số lượng tử hóa sắc độ dựa trên cờ; dẫn xuất bảng tham số lượng tử hóa sắc độ dựa trên dữ liệu tham số lượng tử hóa sắc độ; dẫn xuất tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng tham số lượng tử hóa sắc độ; dẫn xuất các mẫu phần dư cho các thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ; và tạo ra hình ảnh được tái dựng dựa trên các mẫu phần dư. Sáng chế cũng liên quan đến phương pháp mã hóa ảnh, phương tiện lưu trữ phi chuyển tiếp đọc được bởi máy tính và phương pháp truyền dữ liệu cho ảnh.

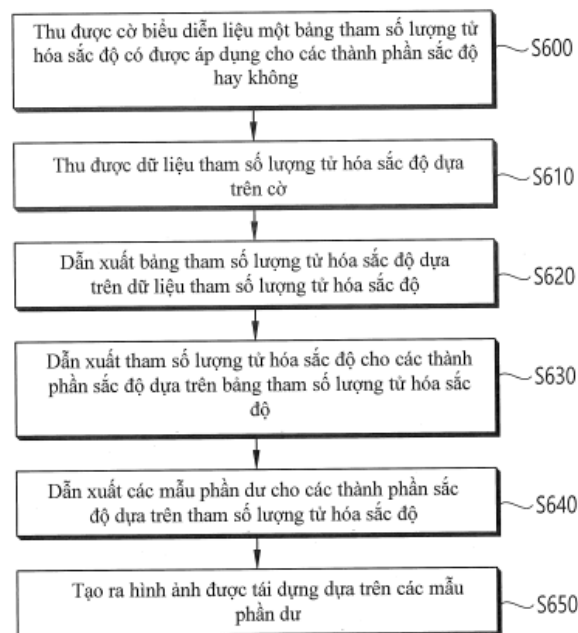


FIG. 6

Lĩnh vực kỹ thuật được đề cập

Sáng chế liên quan đến công nghệ lập mã ảnh, và cụ thể hơn là, đến phương pháp giải mã ảnh và thiết bị để lập mã thông tin ảnh để dẫn xuất bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ trong hệ thống lập mã ảnh.

Tình trạng kỹ thuật của sáng chế

Gần đây, nhu cầu về các ảnh độ phân giải cao, chất lượng cao, chẳng hạn các ảnh độ phân giải cao (High Definition, HD) và các ảnh độ phân giải siêu cao (Ultra High Definition, UHD), đã và đang ngày càng tăng trong các lĩnh vực khác nhau. Vì dữ liệu ảnh này có độ phân giải cao và chất lượng cao, nên lượng thông tin hoặc lượng bit cần được truyền tải so với dữ liệu ảnh kế thừa. Do đó, khi dữ liệu ảnh được truyền bằng các phương tiện như đường băng rộng dùng dây/không dây thông thường, hoặc dữ liệu ảnh được lưu trữ bằng phương tiện lưu trữ hiện có, thì chi phí truyền và chi phí lưu trữ bị tăng lên.

Theo đó, cần có kỹ thuật nén ảnh hiệu quả cao để truyền, lưu trữ, và tái tạo một cách hiệu quả thông tin của các ảnh có độ phân giải cao và chất lượng cao.

Bản chất kỹ thuật của sáng chế

Mục đích kỹ thuật của sáng chế là đề xuất phương pháp và thiết bị để nâng cao hiệu quả lập mã ảnh.

Mục đích kỹ thuật khác của sáng chế là để cung cấp phương pháp và thiết bị để dẫn xuất tham số lượng tử hóa cho thành phần sắc độ.

Theo phương án, sáng chế đề xuất phương pháp giải mã hình ảnh được thực hiện bởi thiết bị giải mã. Phương pháp bao gồm bước thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, bước thu được dữ liệu tham số lượng tử hóa sắc độ dựa trên cờ, bước dẫn xuất bảng tham số lượng tử hóa sắc độ dựa trên dữ liệu tham số lượng tử hóa sắc độ, bước dẫn

xuất tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng tham số lượng tử hóa sắc độ, dẫn xuất các mẫu phần dư cho các thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ và tạo ra hình ảnh được tái dựng dựa trên các mẫu phần dư, trong đó khi giá trị của cờ là 0, thì dữ liệu tham số lượng tử hóa sắc độ bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr.

Theo phương án khác của sáng chế, thiết bị giải mã thực hiện việc giải mã ảnh được cung cấp. Thiết bị giải mã bao gồm bộ mã hóa entropy được tạo cấu hình để thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, để thu được dữ liệu tham số lượng tử hóa sắc độ dựa trên cờ, bộ xử lý phần dư được tạo cấu hình để dẫn xuất bảng tham số lượng tử hóa sắc độ dựa trên dữ liệu tham số lượng tử hóa sắc độ, để dẫn xuất tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng tham số lượng tử hóa sắc độ, để dẫn xuất các mẫu phần dư cho các thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ, bộ cộng được tạo cấu hình để tạo ra hình ảnh được tái dựng dựa trên các mẫu phần dư, trong đó khi giá trị của cờ là 0, dữ liệu tham số lượng tử hóa sắc độ bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr.

Vẫn theo phương án khác của sáng chế này, phương pháp mã hóa video được thực hiện bởi thiết bị mã hóa được cung cấp. Phương pháp bao gồm bước dẫn xuất các mẫu phần dư cho các thành phần sắc độ, bước tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, bước tạo ra dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ và bước giải mã dữ liệu tham số lượng tử hóa sắc độ và cờ, trong đó khi giá trị của cờ là 0, thì dữ liệu tham số lượng tử hóa sắc độ bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr.

Vẫn theo phương án khác nữa của sáng chế này, thiết bị mã hóa video được cung cấp. Thiết bị mã hóa bao gồm bộ xử lý phần dư được tạo cấu hình để dẫn xuất các mẫu phần dư cho các thành phần sắc độ, để tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, để tạo ra dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ, và bộ mã hóa entropy

được tạo cấu hình để mã hóa dữ liệu tham số lượng tử hóa sắc độ và cờ, trong đó khi giá trị của cờ là 0, thì dữ liệu tham số lượng tử hóa sắc độ bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr.

Theo sáng chế này, bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được xác định dựa trên cờ biểu diễn liệu cùng một bảng tham số lượng tử hóa sắc độ có được sử dụng để dẫn xuất tham số lượng tử hóa cho các thành phần sắc độ hay không, thì hiệu quả lập mã có thể được nâng cao bằng cách thực hiện việc lập mã dựa trên tham số lượng tử hóa theo các đặc tính của ảnh.

Theo sáng chế này, bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được xác định dựa trên dữ liệu lượng tử hóa sắc độ được báo hiệu riêng lẻ hoặc chung cho các thành phần sắc độ, hiệu quả lập mã có thể được nâng cao bằng cách thực hiện việc lập mã dựa trên tham số lượng tử hóa theo các đặc tính của ảnh.

Mô tả vắn tắt các hình vẽ

Fig.1 là hình vẽ minh họa vắn tắt ví dụ về thiết bị lập mã video/ảnh mà các phương án của sáng chế có thể áp dụng.

Fig.2 là giản đồ sơ lược minh họa cấu hình của thiết bị mã hóa video/ảnh mà phương án (các phương án) của sáng chế này có thể được áp dụng.

Fig.3 là sơ đồ giản lược minh họa cấu hình của thiết bị giải mã video/ảnh mà phương án (các phương án) của sáng chế có thể được áp dụng.

Fig.4 thể hiện theo cách giản lược phương pháp mã hóa ảnh bởi thiết bị mã hóa theo sáng chế này.

Fig.5 thể hiện theo cách giản lược thiết bị mã hóa để thực hiện phương pháp mã hóa ảnh theo sáng chế này.

Fig.6 thể hiện theo cách giản lược phương pháp giải mã ảnh bởi thiết bị giải mã theo sáng chế này.

Fig.7 thể hiện theo cách giản lược thiết bị giải mã để thực hiện phương pháp giải mã ảnh theo sáng chế này.

Fig.8 là hình vẽ minh họa sơ đồ cấu trúc của hệ thống tạo luồng các nội dung mà

sáng chế được áp dụng.

Mô tả chi tiết sáng chế

Sáng chế có thể được cải biến theo nhiều dạng khác nhau, và các phương án cụ thể của sáng chế sẽ được mô tả và được thể hiện trên các hình vẽ. Tuy nhiên, các phương án thực hiện này không nhằm giới hạn sáng chế. Các thuật ngữ được sử dụng trong phần mô tả sau đây là chỉ được dùng để mô tả các phương án cụ thể, chứ không nhằm giới hạn sáng chế. Biểu diễn của số ít cũng bao gồm biểu diễn của số nhiều, miễn là nó được đọc khác nhau một cách rõ ràng. Các từ ngữ như “bao gồm” và “có” là nhằm thể hiện các dấu hiệu, các số lượng, các bước, các hoạt động, các phần tử, các thành phần, hoặc những sự kết hợp của chúng, mà được sử dụng trong phần mô tả sau đây, là tồn tại, và do đó, cần được hiểu rằng khả năng tồn tại hoặc bổ sung của một hoặc nhiều dấu hiệu, số lượng, bước, hoạt động, phần tử, thành phần khác, hoặc những sự kết hợp của chúng, là không bị loại trừ.

Trong khi đó, các phần tử trên các hình vẽ mà được mô tả theo sáng chế là được vẽ độc lập để thuận tiện cho việc mô tả các chức năng cụ thể khác nhau, và không có nghĩa rằng các phần tử này là được thực hiện bởi phần cứng độc lập hoặc phần mềm độc lập. Ví dụ, hai hoặc nhiều phần tử trong số các phần tử là có thể được kết hợp để tạo thành một phần tử, hoặc một phần tử có thể được chia thành nhiều phần tử. Các phương án mà trong đó các phần tử được kết hợp và/hoặc được chia là thuộc sáng chế chứ không nằm ngoài phạm vi của sáng chế.

Dưới đây, các phương án về sáng chế này sẽ được mô tả chi tiết với việc tham chiếu đến các hình vẽ kèm theo. Ngoài ra, các số chỉ dẫn giống nhau được dùng để chỉ các phần tử giống nhau trong suốt các hình vẽ, và các phần mô tả giống nhau của các phần tử giống nhau sẽ được lược bỏ.

Fig.1 là hình vẽ minh họa vắn tắt ví dụ về thiết bị lập mã video/ảnh mà các phương án của sáng chế có thể áp dụng.

Tham chiếu đến Fig.1, hệ thống lập mã video/ảnh có thể bao gồm thiết bị thứ nhất (thiết bị nguồn) và thiết bị thứ hai (thiết bị thu). Thiết bị nguồn có thể phân phối thông tin hoặc dữ liệu video/ảnh được mã hóa ở dạng của tệp hoặc tạo luồng tới thiết bị thu thông qua phương tiện lưu trữ số hoặc mạng.

Thiết bị nguồn có thể gồm nguồn video, thiết bị mã hóa, và bộ truyền. Thiết bị thu có thể bao gồm bộ thu, thiết bị giải mã, và bộ kết xuất. Thiết bị mã hóa có thể được gọi là thiết bị mã hóa video/ảnh, và thiết bị giải mã có thể được gọi là thiết bị giải mã video/ảnh. Bộ truyền có thể được bao gồm trong thiết bị mã hóa. Bộ thu có thể được bao gồm trong thiết bị giải mã. Bộ kết xuất có thể chứa bộ phận hiển thị, bộ phận hiển thị có thể được tạo cấu hình như là thiết bị tách rời hoặc thành phần ngoài.

Nguồn video có thể thu được video/ảnh qua quy trình chụp, tổng hợp, hoặc tạo ra video/ảnh. Nguồn video có thể bao gồm thiết bị chụp video/ảnh và/hoặc thiết bị tạo ra video/ảnh. Thiết bị chụp video/ảnh có thể bao gồm, ví dụ, một hoặc nhiều camera, phần lưu trữ video/ảnh chứa các video/ảnh đã được chụp trước đó, và dạng tương tự. Thiết bị tạo ra video/ảnh có thể bao gồm, ví dụ, máy tính, máy tính bảng, và điện thoại thông minh, và có thể tạo ra các video/ảnh (theo cách điện tử). Ví dụ, video/hình ảnh ảo có thể được tạo ra qua máy tính hoặc tương tự. Trong trường hợp này, quy trình chụp video/ảnh có thể được thay thế bởi quy trình tạo ra dữ liệu liên quan.

Thiết bị mã hóa có thể mã hóa ảnh/ảnh đầu vào. Thiết bị mã hóa có thể thực hiện chuỗi các thủ tục như dự đoán, biến đổi, và lượng tử hóa cho việc nén và mã hóa hiệu quả. Dữ liệu được mã hóa (thông tin video/ảnh được mã hóa) có thể được xuất ra ở dạng luồng bit.

Bộ truyền có thể truyền thông tin hoặc dữ liệu ảnh/video được mã hóa được xuất ra ở dạng luồng bit đến bộ thu của thiết bị thu qua phương tiện lưu trữ số hoặc mạng ở dạng của tệp hoặc tạo luồng. Phương tiện lưu trữ số có thể bao gồm phương tiện lưu trữ khác nhau như USB, SD, CD, DVD, Blu-ray, HDD, SSD, và tương tự. Bộ truyền có thể bao gồm phần tử để tạo ra tệp tin phương tiện thông qua định dạng tệp tin định trước và có thể bao gồm phần tử để truyền qua mạng quảng bá/truyền thông. Bộ thu có thể thu/tách luồng bit và truyền luồng bit thu được đến thiết bị giải mã.

Thiết bị giải mã có thể giải mã video/ảnh bằng cách thực hiện chuỗi các thủ tục như giải lượng tử hóa, biến đổi ngược, và dự đoán tương ứng với hoạt động vận hành của thiết bị mã hóa.

Bộ kết xuất có thể kết xuất video/ảnh được giải mã. Video/ảnh được kết xuất có thể được hiển thị qua bộ hiển thị.

Sáng chế này liên quan tới việc lập mã video/ảnh. Ví dụ, các phương pháp/các phương án được bộc lộ trong sáng chế này có thể được áp dụng cho phương pháp được bộc lộ trong chuẩn lập mã video đa năng (Versatile video coding, VVC), chuẩn lập mã video cốt yếu (Essential video coding, EVC), chuẩn audio video 1 (AOMedia Video 1, AV1), chuẩn lập mã video thế hệ thứ hai (2nd generation of audio video coding standard, AVS2), hoặc chuẩn lập mã video/ảnh thế hệ tiếp theo (ví dụ, H.267 hoặc H.268, v.v.).

Sáng chế này thể hiện các phương án khác nhau của việc lập mã video/ảnh, và các phương án có thể được thực hiện kết hợp với nhau trừ khi được đề cập theo cách khác.

Trong sáng chế này, video có thể đề cập đến chuỗi các ảnh theo thời gian. Hình ảnh nói chung đề cập tới đơn vị đang biểu diễn một ảnh trong vùng thời gian cụ thể, và hình ảnh con/lát/ngói là đơn vị cấu thành một phần của hình ảnh trong khi lập mã. Hình ảnh con/lát/ô xếp có thể gồm một hoặc nhiều đơn vị cây lập mã (Coding tree unit, CTU). Một hình ảnh có thể bao gồm một hoặc nhiều hình ảnh con/lát/ô xếp. Một hình ảnh có thể bao gồm một hoặc nhiều nhóm ô xếp. Một nhóm ô xếp có thể chứa một hoặc nhiều ô xếp. Gạch có thể biểu diễn vùng hình chữ nhật của các hàng CTU bên trong ô xếp trong hình ảnh. Ô xếp có thể được phân chia thành nhiều gạch, trong số đó mỗi gạch bao gồm một hoặc nhiều hàng CTU bên trong ô xếp. Ô xếp mà không được phân chia thành nhiều gạch có thể cũng được đề cập đến như gạch. Quét gạch là lập thứ tự các CTU theo trình tự cụ thể đang phân chia hình ảnh trong đó các CTU được lập thứ tự liên tiếp theo quét mảnh CTU trong gạch, các gạch bên trong ô xếp được lập thứ tự liên tiếp theo quét mảnh các gạch của ô xếp, và các ô xếp trong hình ảnh được lập thứ tự liên tiếp theo quét mảnh các ô xếp của hình ảnh. Ngoài ra, hình ảnh con có thể biểu diễn vùng hình chữ nhật gồm một hoặc nhiều lát bên trong hình ảnh. Tức là, hình ảnh con chứa một hoặc nhiều lát mà nhìn chung bao gồm cả vùng hình chữ nhật của hình ảnh. Ô xếp là vùng hình chữ nhật của các CTU nằm trong cột ô xếp cụ thể và hàng ngôi cụ thể trong hình ảnh. Cột ô xếp là vùng hình chữ nhật của các CTU có chiều cao bằng với chiều cao của hình ảnh và chiều rộng được đặc tả bằng các phần tử cú pháp trong tập tham số hình ảnh. Hàng ô xếp là vùng hình chữ nhật của các CTU có chiều cao được đặc tả bởi các phần tử cú pháp trong tập tham số hình ảnh và chiều rộng bằng với chiều rộng của hình ảnh. Quét ô xếp là việc lập thứ tự các CTU theo trình tự cụ thể đang phân chia hình ảnh trong đó các CTU được lập thứ tự liên tiếp trong quét mảnh CTU trong ô xếp, trái lại

các ô xếp trong hình ảnh được lập thứ tự liên tiếp trong quét màn hình các ô xếp của hình ảnh. Lát bao gồm một số gạch nguyên của hình ảnh mà có thể được chứa dành riêng theo đơn vị NAL đơn lẻ. Lát có thể bao gồm một số ô xếp hoàn chỉnh hoặc chỉ trình tự liên tiếp của các gạch hoàn chỉnh của một ô xếp. Các nhóm ô xếp và lát có thể được sử dụng hoán đổi cho nhau trong sáng chế này. Ví dụ, trong sáng chế này, nhóm ô xếp/tiêu đề nhóm ô xếp có thể được gọi là lát/tiêu đề lát.

Điểm ảnh (pixel) hoặc pel có thể nghĩa là đơn vị nhỏ nhất cấu thành một hình ảnh (hoặc ảnh). Cũng như vậy, “mẫu” có thể được sử dụng như thuật ngữ tương ứng với điểm ảnh. Mẫu nhìn chung có thể biểu diễn điểm ảnh hoặc giá trị của điểm ảnh, và có thể biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần độ chói hoặc chỉ điểm ảnh/giá trị điểm ảnh của thành phần sắc độ.

Đơn vị có thể biểu diễn đơn vị cơ sở của việc xử lý ảnh. Đơn vị có thể bao gồm ít nhất một trong số vùng cụ thể của hình ảnh và thông tin liên quan tới vùng. Một đơn vị có thể chứa một khối độ chói và hai khối sắc độ (ví dụ Cb, cr). Đơn vị có thể được sử dụng theo kiểu thay thế cho nhau theo nghĩa như là khối hoặc khu vực trong một số trường hợp. Trong trường hợp chung, khối $M \times N$ có thể bao gồm các mẫu (hoặc các mảng mẫu) hoặc tập (hoặc mảng) các hệ số biến đổi gồm M cột và N hàng.

Trong sáng chế, “A hoặc B” có thể nghĩa là “chỉ A”, “chỉ B” hoặc “cả A và B”. Nói cách khác, trong sáng chế này, “A hoặc B” có thể được diễn dịch như “A và/hoặc B”. Ví dụ, trong sáng chế này, “A, B hoặc C” nghĩa là “chỉ A”, “chỉ B”, “chỉ C”, hoặc “sự kết hợp bất kỳ của A, B và C”.

Dấu gạch chéo (/) hoặc dấu phẩy được sử dụng trong sáng chế này có thể nghĩa là “và/hoặc”. Ví dụ, “A/B” có nghĩa là “A và/hoặc B”. Theo đó, “A/B” có nghĩa là “chỉ A”, “chỉ B”, hoặc “cả A và B”. Ví dụ, “A, B, C” có thể có nghĩa là “A, B hoặc C”.

Trong sáng chế, “ít nhất một trong số A và B” có thể chỉ “chỉ A”, “chỉ B”, hoặc “cả A và B”. Ngoài ra, theo sáng chế này, cách biểu diễn “ít nhất một trong số A hoặc B” hoặc “ít nhất một trong số A và/hoặc B” có thể được diễn dịch giống như “ít nhất một trong số A và B”.

Ngoài ra, trong sáng chế này, “ít nhất một trong số A, B và C” nghĩa là “chỉ A”, “chỉ B”, “chỉ C”, hoặc “sự kết hợp bất kỳ của A, B và C”. Cũng như vậy, “ít nhất một

trong số A, B hoặc C” hoặc “ít nhất một trong số A, B và/hoặc C” có thể có nghĩa là “ít nhất một trong số A, B và C”.

Ngoài ra, các dấu ngoặc đơn được sử dụng trong sáng chế này có thể nghĩa là “ví dụ”. Cụ thể là, khi “việc dự đoán (nội dự đoán)” được chỉ báo, thì “nội dự đoán” có thể được đề xuất như ví dụ của “dự đoán”. Nói cách khác, “dự đoán” trong sáng chế này có thể không được giới hạn ở “nội dự đoán”, và “nội dự đoán” có thể được đề xuất như ví dụ của “dự đoán”. Cũng như vậy, thậm chí khi “việc dự đoán (nội dự đoán)” được thể hiện, thì “nội dự đoán” có thể được đề xuất như ví dụ của “dự đoán”.

Trong sáng chế này, các đặc điểm kỹ thuật mà được mô tả riêng lẻ trong phạm vi một hình vẽ có thể được thực hiện riêng lẻ hoặc có thể được thực hiện cùng lúc.

Các hình vẽ sau đây được tạo ra để giải thích ví dụ cụ thể của sáng chế. Bởi vì các tên của các thiết bị cụ thể được mô tả trong các hình vẽ hoặc các tên của các tín hiệu/thông điệp/trường cụ thể được biểu thị theo cách ví dụ, nên các đặc điểm kỹ thuật của sáng chế này không được giới hạn ở các tên cụ thể được sử dụng trong các hình vẽ sau đây.

Fig.2 là giản đồ sơ lược minh họa cấu hình của thiết bị mã hóa video/ảnh mà phương án (các phương án) của sáng chế này có thể được áp dụng. Dưới đây, thiết bị mã hóa video có thể bao gồm thiết bị mã hóa ảnh.

Tham chiếu đến Fig.2, thiết bị mã hóa 200 bao gồm bộ phân chia ảnh 210, bộ dự đoán 220, bộ xử lý phần dư 230, và bộ mã hóa entropy 240, bộ cộng 250, bộ lọc 260, và bộ nhớ 270. Bộ dự đoán 220 có thể bao gồm bộ liên dự đoán 221 và bộ nội dự đoán 222. Bộ xử lý phần dư 230 có thể bao gồm bộ biến đổi 232, bộ lượng tử hóa 233, bộ giải lượng tử hóa 234, và bộ biến đổi ngược 235. Bộ xử lý phần dư 230 có thể còn bao gồm bộ trừ 231. Bộ cộng 250 có thể được gọi là bộ tái dựng hoặc bộ tạo ra khối được tái dựng. Bộ phân chia ảnh 210, bộ dự đoán 220, bộ xử lý phần dư 230, bộ mã hóa entropy 240, bộ cộng 250, và bộ lọc 260 có thể được tạo cấu hình bằng ít nhất một thành phần phần cứng (ví dụ, bộ chip của bộ mã hóa hoặc bộ xử lý) theo phương án. Ngoài ra, bộ nhớ 270 có thể bao gồm bộ đệm hình ảnh được giải mã (decoded picture buffer, DPB) hoặc có thể được tạo cấu hình bằng phương tiện lưu trữ số. Thành phần phần cứng có thể còn bao gồm bộ nhớ 270 như thành phần bên trong/bên ngoài.

Bộ phân vùng ảnh 210 có thể phân vùng ảnh đầu vào (hoặc hình ảnh hoặc khung) được nhập vào thiết bị mã hóa 200 vào trong một hoặc nhiều bộ xử lý. Ví dụ, bộ xử lý có thể được gọi là đơn vị lập mã (Coding unit, CU). Trong trường hợp này, đơn vị lập mã có thể được phân vùng theo cách đệ quy theo cấu trúc cây tứ phân nhị phân tam phân (Quad-tree binary-tree ternary-tree, QTBTNTT) từ đơn vị cây lập mã (Coding tree unit, CTU) hoặc đơn vị lập mã lớn nhất (Largest coding unit, LCU). Ví dụ, một đơn vị lập mã có thể được phân chia thành nhiều đơn vị lập mã có độ sâu sâu hơn dựa trên cấu trúc cây tứ phân, cấu trúc cây nhị phân, và/hoặc cấu trúc tam phân. Trong trường hợp này, ví dụ, cấu trúc cây tứ phân có thể được áp dụng đầu tiên và cấu trúc cây nhị phân và/hoặc cấu trúc tam phân có thể được áp dụng sau. Thay vào đó, cấu trúc cây nhị phân có thể được áp dụng trước. Thủ tục lập mã theo sáng chế có thể được thực hiện dựa trên đơn vị lập mã cuối cùng mà không được phân chia tiếp nữa. Trong trường hợp này, đơn vị lập mã lớn nhất có thể được dùng làm đơn vị lập mã cuối cùng dựa trên hiệu quả lập mã theo các đặc tính của ảnh, hoặc nếu cần thiết, đơn vị lập mã có thể được chia theo cách đệ quy thành các đơn vị lập mã có chiều sâu sâu hơn, và đơn vị lập mã có kích thước tối ưu có thể được dùng làm đơn vị lập mã cuối cùng. Ở đây, thủ tục lập mã có thể bao gồm thủ tục dự đoán, biến đổi, và tái dựng, mà sẽ được mô tả sau. Theo ví dụ khác, bộ xử lý có thể còn bao gồm đơn vị dự đoán (Prediction unit, PU) hoặc đơn vị biến đổi (Transform Unit, TU). Trong trường hợp này, đơn vị dự đoán và đơn vị biến đổi có thể được chia hoặc phân chia từ đơn vị lập mã cuối cùng được đề cập đến ở trên. Đơn vị dự đoán có thể là đơn vị của việc dự đoán mẫu, và đơn vị biến đổi có thể là đơn vị để dẫn xuất hệ số biến đổi và/hoặc đơn vị để dẫn xuất tín hiệu phần dư từ hệ số biến đổi.

Đơn vị có thể được sử dụng theo kiểu thay thế cho nhau theo nghĩa như là khối hoặc khu vực trong một số trường hợp. Trong trường hợp chung, khối $M \times N$ có thể biểu diễn tập các mẫu hoặc các hệ số biến đổi bao gồm M cột và N hàng. Mẫu nói chung là có thể biểu diễn điểm ảnh hoặc giá trị của điểm ảnh, có thể biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần độ chói hoặc biểu diễn chỉ điểm ảnh/giá trị điểm ảnh của thành phần sắc độ. Mẫu có thể được sử dụng như thuật ngữ tương ứng với một hình ảnh (hoặc ảnh) cho điểm ảnh hoặc pel.

Trong thiết bị mã hóa 200, tín hiệu dự đoán (khối được dự đoán, mảng mẫu dự đoán) được xuất ra từ bộ liên dự đoán 221 hoặc bộ nội dự đoán 222 được trừ từ tín hiệu

ảnh đầu vào (khối gốc, mảng mẫu gốc) để tạo ra khối dư tín hiệu phần dư, mảng mẫu phần dư), và tín hiệu phần dư được tạo ra được truyền đến bộ biến đổi 232. Trong trường hợp này, như được thể hiện, đơn vị để trừ tín hiệu dự đoán (khối được dự đoán, mảng mẫu dự đoán) từ tín hiệu ảnh đầu vào (khối gốc, mảng mẫu gốc) trong bộ mã hóa 200 có thể được gọi là bộ trừ 231. Bộ dự đoán có thể thực hiện việc dự đoán trên khối cần được xử lý (ở dưới đây, được đề cập đến như là khối hiện tại) và tạo ra khối được dự đoán bao gồm các mẫu dự đoán cho khối hiện tại. Bộ dự đoán có thể xác định liệu việc nội dự đoán hay liên dự đoán được áp dụng trên cơ sở khối hiện tại hay CU. Như được mô tả sau trong phần mô tả về mỗi chế độ dự đoán, bộ dự đoán có thể tạo ra thông tin khác nhau liên quan đến dự đoán, như thông tin chế độ dự đoán, và truyền thông tin được tạo ra đến bộ mã hóa entropy 240. Thông tin về việc dự đoán có thể được mã hóa trong bộ mã hóa entropy 240 và được xuất ra dưới dạng luồng bit.

Bộ nội dự đoán 222 có thể dự đoán khối hiện tại bằng cách tham chiếu đến các mẫu trong hình ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt trong vùng lân cận của khối hiện tại hoặc có thể đặt tách ra theo chế độ dự đoán. Trong việc nội dự đoán, các chế độ dự đoán có thể bao gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Các chế độ không hướng có thể bao gồm, ví dụ, chế độ DC và chế độ phẳng. Chế độ có hướng có thể bao gồm, ví dụ, 33 chế độ dự đoán có hướng hoặc 65 chế độ dự đoán có hướng theo cấp độ chi tiết của hướng dự đoán. Tuy nhiên, điều này chỉ đơn thuần là ví dụ, nhiều hoặc ít chế độ dự đoán có hướng hơn có thể được sử dụng phụ thuộc vào việc cài đặt. Bộ nội dự đoán 222 có thể xác định chế độ dự đoán được áp dụng cho khối hiện tại bằng cách sử dụng chế độ dự đoán được áp dụng cho khối lân cận.

Bộ liên dự đoán 221 có thể dẫn xuất khối được dự đoán cho khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được đặc tả bằng vector chuyển động trên hình ảnh tham chiếu. Ở đây, để làm giảm lượng thông tin chuyển động được truyền trong chế độ liên dự đoán, thông tin chuyển động có thể được dự đoán trong các đơn vị là các khối, các khối con, hoặc các mẫu dựa trên sự tương quan của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể bao gồm vector chuyển động và chỉ số hình ảnh tham chiếu. Thông tin chuyển động có thể còn bao gồm thông tin hướng liên dự đoán (dự đoán L0, dự đoán L1, song dự đoán, v.v.). Trong trường hợp liên dự đoán, khối lân cận có thể bao gồm khối lân cận không gian có mặt trong hình ảnh hiện

tại và khối lân cận thời gian có mặt trong hình ảnh tham chiếu. Hình ảnh tham chiếu chứa khối tham chiếu và hình ảnh tham chiếu bao gồm khối lân cận theo thời gian có thể là giống nhau hoặc khác nhau. Khối lân cận thời gian có thể được gọi là khối tham chiếu được sắp xếp vào một chỗ, CU được đặt cùng vị trí (Co-located CU, colCU), và tương tự, và hình ảnh tham chiếu bao gồm khối lân cận thời gian có thể được gọi là hình ảnh được sắp xếp vào một chỗ (Collocated picture, colPic). Ví dụ, bộ liên dự đoán 221 có thể tạo cấu hình danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận và tạo ra thông tin chỉ báo ứng viên nào được sử dụng để dẫn xuất vector chuyển động và/hoặc chỉ số hình ảnh tham chiếu của khối hiện tại. Việc liên dự đoán có thể được thực hiện dựa trên các chế độ dự đoán khác nhau. Ví dụ, trong trường hợp chế độ bỏ cóc và chế độ hợp nhất, bộ liên dự đoán 221 có thể sử dụng thông tin chuyển động của khối lân cận như thông tin chuyển động của khối hiện tại. Trong chế độ bỏ cóc, không giống chế độ hợp nhất, tín hiệu phần dư có thể không được truyền. Trong trường hợp của chế độ dự đoán vector chuyển động (Motion vector prediction, MVP), vector chuyển động của khối lân cận có thể được sử dụng làm bộ dự đoán vector chuyển động và vector chuyển động của khối hiện tại có thể được chỉ thị bằng cách báo hiệu sự chênh lệch vector chuyển động.

Bộ dự đoán 220 có thể tạo ra tín hiệu dự đoán dựa trên các phương pháp dự đoán khác nhau được mô tả bên dưới. Ví dụ, bộ dự đoán có thể không chỉ áp dụng việc nội dự đoán hoặc việc liên dự đoán để dự đoán một khối mà còn có thể đồng thời áp dụng cả việc nội dự đoán và việc liên dự đoán. Điều này có thể được gọi là liên dự đoán và nội dự đoán kết hợp (Combined inter and intra prediction, CIIP). Ngoài ra, bộ dự đoán có thể được dựa trên chế độ nội dự đoán sao chép khối (intra block copy, IBC) hoặc chế độ bảng màu để dự đoán khối. Chế độ dự đoán IBC hoặc chế độ bảng màu (palette) có thể được sử dụng cho việc lập mã ảnh/video nội dung của trò chơi hoặc dạng tương tự, ví dụ, lập mã nội dung màn (screen content coding, SCC). Về cơ bản, IBC thực hiện việc dự đoán trong hình ảnh hiện tại nhưng có thể được thực hiện tương tự cho việc liên dự đoán trong đó khối tham chiếu được dẫn xuất trong hình ảnh hiện tại. Tức là, IBC có thể sử dụng ít nhất một trong các kỹ thuật liên dự đoán được mô tả trong sáng chế. Chế độ bảng màu có thể được xem xét như là ví dụ của việc nội lập mã hoặc việc nội dự đoán. Khi chế độ bảng màu được áp dụng, giá trị mẫu nằm trong hình ảnh có thể được

báo hiệu dựa trên thông tin về bảng của bảng màu và chỉ số bảng màu.

Tín hiệu dự đoán được tạo ra bởi bộ dự đoán (bao gồm bộ liên dự đoán 221 và/hoặc bộ nội dự đoán 222) có thể được sử dụng để tạo ra tín hiệu được tái dựng hoặc để tạo ra tín hiệu dư. Bộ biến đổi 232 có thể tạo ra các hệ số biến đổi bằng cách áp dụng kỹ thuật biến đổi cho tín hiệu dư. Ví dụ, kỹ thuật biến đổi có thể bao gồm ít nhất một trong số biến đổi cosin rời rạc (Discrete cosine transform, DCT), biến đổi sin rời rạc (Discrete sine transform, DST), biến đổi karhunen-loève (Karhunen-loève transform, KLT), biến đổi dựa trên đồ thị (Graph-based transform, GBT), hoặc biến đổi phi tuyến có điều kiện (Conditionally non-linear transform, CNT). Ở đây, GBT nghĩa là việc biến đổi thu được từ đồ thị khi thông tin mối quan hệ giữa các điểm ảnh được biểu diễn bằng đồ thị. CNT đề cập đến việc biến đổi được tạo ra dựa trên tín hiệu dự đoán được tạo ra sử dụng tất cả các điểm ảnh được tái dựng từ trước. Thêm vào đó, quy trình biến đổi có thể được áp dụng cho các khối điểm ảnh hình vuông có cùng kích thước hoặc có thể được áp dụng cho các khối có kích thước thay đổi ngoài hình vuông.

Bộ lượng tử hóa 233 có thể lượng tử hóa các hệ số biến đổi và truyền chúng tới bộ mã hóa entropy 240 và bộ mã hóa entropy 240 có thể mã hóa tín hiệu đã lượng tử hóa (thông tin về các hệ số biến đổi đã được lượng tử hóa) và xuất ra luồng bit. Thông tin về các hệ số biến đổi đã được lượng tử hóa có thể được đề cập đến như thông tin phần dư. Bộ lượng tử hóa 233 có thể sắp xếp lại các hệ số biến đổi đã được lượng tử hóa dạng khối thành dạng vector một chiều dựa trên thứ tự quét hệ số và tạo ra thông tin về các hệ số biến đổi đã được lượng tử hóa dựa trên các hệ số biến đổi đã được lượng tử hóa ở dạng vector một chiều. Thông tin về các hệ số biến đổi có thể được tạo ra. Bộ mã hóa entropy 240 có thể thực hiện các phương pháp mã hóa khác nhau như, ví dụ Golomb hàm số mũ, lập mã chiều dài biến đổi thích ứng ngữ cảnh (Context-Adaptive Variable Length Coding, CAVLC), lập mã số học nhị phân thích ứng ngữ cảnh (Context-Adaptive Binary Arithmetic Coding, CABAC), và dạng tương tự. Bộ mã hóa entropy 240 có thể mã hóa thông tin cần thiết cho việc tái dựng video/ảnh chứ không phải các hệ số biến đổi được lượng tử hóa (ví dụ, các giá trị của các phần tử cú pháp, v.v.) cùng nhau hoặc riêng biệt. Thông tin được mã hóa (ví dụ, thông tin video/ảnh được mã hóa) có thể được truyền hoặc được lưu giữ trong các đơn vị là lớp trừu tượng mạng (Network Abstraction Layer, NAL) ở dạng luồng bit. Thông tin video/ảnh có thể còn bao gồm thông tin về các

tập tham số khác nhau như tập tham số thích ứng (adaptation parameter set, APS), tập tham số hình ảnh (picture parameter set, PPS), tập tham số trình tự (sequence parameter set, SPS), hoặc tập tham số video (video parameter set, VPS). Ngoài ra, thông tin video/ảnh có thể còn bao gồm thông tin ràng buộc chung. Trong sáng chế này, thông tin và/hoặc các phân tử cú pháp được truyền/được báo hiệu từ thiết bị mã hóa đến thiết bị giải mã có thể được bao gồm trong thông tin video/hình ảnh. Thông tin video/hình ảnh có thể được mã hóa qua thủ tục mã hóa được mô tả ở trên và được bao gồm trong luồng bit. Luồng bit có thể được truyền qua mạng hoặc có thể được lưu trong phương tiện lưu trữ số. Mạng có thể bao gồm mạng quảng bá và/hoặc mạng truyền thông, và phương tiện lưu trữ số có thể bao gồm các phương tiện lưu trữ khác nhau như USB, SD, CD, DVD, Blu-ray, HDD, SSD, và dạng tương tự. Bộ truyền (không được thể hiện) truyền tín hiệu xuất ra từ bộ mã hóa entropy 240 và/hoặc đơn vị lưu trữ (không được thể hiện) đang lưu tín hiệu có thể được bao gồm như thành phần bên trong/bên ngoài của thiết bị mã hóa 200, và theo cách khác, bộ truyền có thể được bao gồm trong bộ mã hóa entropy 240.

Các hệ số biến đổi đã được lượng tử hóa được xuất ra từ bộ lượng tử hóa 233 có thể được sử dụng để tạo ra tín hiệu dự đoán. Ví dụ, tín hiệu phần dư (khối phần dư hoặc các mẫu phần dư) có thể được tái dựng bằng cách áp dụng việc giải lượng tử hóa và biến đổi ngược đối với các hệ số biến đổi đã được lượng tử hóa qua bộ giải lượng tử hóa 234 và bộ biến đổi ngược 235. Bộ cộng 250 cộng tín hiệu phần dư được tái dựng vào tín hiệu dự đoán được xuất ra từ bộ liên dự đoán 221 hoặc bộ nội dự đoán 222 để tạo ra tín hiệu được tái dựng (hình ảnh được tái dựng, khối được tái dựng, mảng mẫu được tái dựng). Nếu không có phần dư nào cho khối cần được xử lý, như trường hợp mà trong đó chế độ bỏ cóc được áp dụng, thì khối được dự đoán có thể được sử dụng làm khối được tái dựng. Bộ cộng 250 có thể được gọi là bộ tái dựng hoặc bộ tạo ra khối được tái dựng. Tín hiệu được tái dựng được tạo ra có thể được sử dụng cho việc nội dự đoán khối tiếp theo cần được xử lý trong hình ảnh hiện tại và có thể được sử dụng cho việc liên dự đoán hình ảnh tiếp theo qua việc lọc như được mô tả bên dưới.

Trong khi đó, phép ánh xạ độ chói cùng với việc định tỷ lệ sắc độ (Luma mapping with chroma scaling, LMCS) có thể được áp dụng trong suốt quá trình mã hóa và/hoặc tái dựng hình ảnh.

Bộ lọc 260 có thể nâng cao chất lượng ảnh chủ quan/khách quan bằng cách áp dụng việc lọc cho tín hiệu được tái dựng. Ví dụ, bộ lọc 260 có thể tạo ra hình ảnh được tái dựng được điều chỉnh bằng cách áp dụng các phương pháp lọc khác nhau đối với hình ảnh được tái dựng và lưu trữ hình ảnh được tái dựng được điều chỉnh trong bộ nhớ 270, cụ thể, DPB của bộ nhớ 270. Các phương pháp lọc khác nhau có thể bao gồm, ví dụ, lọc khử khối, bù tương thích mẫu, bộ lọc vòng lặp tương thích, bộ lọc song phương, và dạng tương tự. Bộ lọc 260 có thể tạo ra thông tin khác nhau liên quan tới việc lọc và truyền thông tin được tạo ra đến bộ mã hóa entropy 240 như sẽ được mô tả sau trong phần mô tả về mỗi phương pháp lọc. Thông tin liên quan đến việc lọc có thể được mã hóa bởi bộ mã hóa entropy 240 và được xuất ra trong dạng luồng bit.

Hình ảnh được tái dựng được điều chỉnh mà được truyền đến bộ nhớ 270 là có thể được sử dụng làm hình ảnh tham chiếu ở bộ liên dự đoán 221. Khi việc liên dự đoán được áp dụng qua thiết bị mã hóa, thì việc không khớp dự đoán giữa thiết bị mã hóa 200 và thiết bị giải mã 300 có thể tránh được và hiệu quả mã hóa có thể được nâng cao.

DPB của bộ nhớ DPB 270 có thể lưu hình ảnh được tái dựng được biến đổi để sử dụng như là hình ảnh tham chiếu trong bộ liên dự đoán 221. Bộ nhớ 270 có thể lưu trữ thông tin chuyển động của khối mà thông tin chuyển động trong ảnh hiện tại được dẫn xuất (hoặc được mã hóa) từ đó và/hoặc thông tin chuyển động của các khối trong hình ảnh mà đã được tái dựng. Thông tin chuyển động được lưu trữ có thể được truyền đến bộ liên dự đoán 221 và được sử dụng như là thông tin chuyển động của khối lân cận không gian hoặc thông tin chuyển động của khối lân cận thời gian. Bộ nhớ 270 có thể lưu trữ các mẫu được tái dựng của các khối được tái dựng trong hình ảnh hiện tại và có thể chuyển các mẫu được tái dựng đến bộ nội dự đoán 222.

Fig.3 là sơ đồ giản lược minh họa cấu hình của thiết bị giải mã video/ảnh mà phương án (các phương án) của sáng chế có thể được áp dụng.

Tham chiếu đến Fig.3, thiết bị giải mã 300 có thể bao gồm bộ giải mã entropy 310, bộ xử lý phần dư 320, bộ dự đoán 330, bộ cộng 340, bộ lọc 350, bộ nhớ 360. Bộ dự đoán 330 có thể bao gồm bộ liên dự đoán 331 và bộ nội dự đoán 332. Bộ xử lý phần dư 320 có thể bao gồm bộ giải lượng tử hóa 321 và bộ biến đổi ngược 322. Bộ giải mã entropy 310, bộ xử lý phần dư 320, bộ dự đoán 330, bộ cộng 340, và bộ lọc 350 có thể được tạo cấu hình bởi thành phần phần cứng (ví dụ, bộ chip của bộ giải mã hoặc bộ xử lý) theo

một phương án. Ngoài ra, bộ nhớ 360 có thể bao gồm bộ đệm hình ảnh được giải mã (DPB) hoặc có thể được tạo cấu hình bằng phương tiện lưu trữ số. Thành phần phần cứng có thể còn bao gồm bộ nhớ 360 như thành phần bên trong/bên ngoài.

Khi luồng bit bao gồm thông tin video/ảnh được nhập vào, thì thiết bị giải mã 300 có thể tái dựng ảnh tương ứng với quy trình trong đó thông tin video/ảnh được xử lý trong thiết bị mã hóa của Fig.2. Ví dụ, thiết bị giải mã 300 có thể dẫn xuất các đơn vị/các khối dựa trên thông tin liên quan đến sự phân vùng khối được thu được từ luồng bit. Thiết bị giải mã 300 có thể thực hiện việc giải mã sử dụng bộ xử lý được áp dụng trong thiết bị mã hóa. Do đó, bộ xử lý việc giải mã có thể đơn vị lập mã chẳng hạn và đơn vị lập mã có thể được phân vùng theo cấu trúc cây tứ phân, cấu trúc cây nhị phân và/hoặc cấu trúc cây tam phân từ đơn vị cây lập mã hoặc đơn vị lập mã lớn nhất. Một hoặc nhiều đơn vị biến đổi có thể được dẫn xuất từ đơn vị lập mã. Tín hiệu ảnh được tái dựng được giải mã và xuất ra qua thiết bị giải mã 300 có thể được tái tạo thông qua thiết bị tái tạo.

Thiết bị giải mã 300 có thể thu tín hiệu mà được xuất ra từ thiết bị mã hóa trên Fig. 2 dưới dạng luồng bit, và tín hiệu thu được có thể được giải mã thông qua bộ giải mã entropy 310. Ví dụ, bộ giải mã entropy 310 có thể phân tích cú pháp luồng bit để dẫn xuất thông tin (ví dụ, thông tin video/ảnh) cần thiết cho việc tái dựng ảnh (hoặc tái dựng hình ảnh). Thông tin video/ảnh có thể còn bao gồm thông tin về các tập tham số khác nhau như tập tham số thích ứng (APS), tập tham số hình ảnh (PPS), tập tham số trình tự (SPS), hoặc tập tham số video (VPS). Ngoài ra, thông tin video/ảnh có thể còn bao gồm thông tin ràng buộc chung. Thiết bị giải mã có thể còn giải mã hình ảnh dựa trên thông tin về tập tham số và/hoặc thông tin ràng buộc chung. Thông tin được báo hiệu/thu được và/hoặc các phần tử cú pháp được mô tả ở sau trong sáng chế này có thể được giải mã có thể giải mã thủ tục giải mã và thu được từ luồng bit. Ví dụ, bộ giải mã entropy 310 giải mã thông tin trong luồng bit dựa trên phương pháp lập mã như lập mã Golomb hàm số mũ, CAVLC hoặc CABAC, và xuất ra các phần tử cú pháp được yêu cầu cho việc tái dựng ảnh và các giá trị được lượng tử hóa của các hệ số biến đổi cho phần dư. Cụ thể hơn, phương pháp giải mã entropy CABAC có thể thu ngắn (bin) tương ứng với mỗi phần tử cú pháp trong luồng bit, xác định mô hình ngữ cảnh sử dụng thông tin về phần tử cú pháp mục tiêu giải mã, thông tin về khối mục tiêu giải mã hoặc thông tin về ký hiệu/bin được giải mã trong pha trước, và thực hiện việc giải mã số học trên bin này

bằng cách dự đoán xác suất xuất hiện bin theo mô hình ngữ cảnh được xác định, và tạo ra ký hiệu tương ứng với giá trị của mỗi phần tử cú pháp. Trong trường hợp này, phương pháp giải mã entropy CABAC có thể cập nhật mô hình ngữ cảnh bằng cách sử dụng thông tin về ký hiệu/bin được giải mã cho mô hình ngữ cảnh của ký hiệu/bin tiếp theo sau khi xác định mô hình ngữ cảnh. Thông tin liên quan đến việc dự đoán trong số thông tin được giải mã bởi bộ giải mã entropy 310 có thể được cấp đến bộ dự đoán (bộ liên dự đoán 332 và bộ nội dự đoán 331), và giá trị dư mà việc giải mã entropy được thực hiện trên đó trong bộ giải mã entropy 310, tức là, các hệ số biến đổi được lượng tử hóa và thông tin tham số liên quan, có thể được cấp vào bộ xử lý phần dư 320. Bộ xử lý phần dư 320 có thể dẫn xuất tín hiệu phần dư (khối phần dư, các mẫu phần dư, mảng mẫu phần dư). Ngoài ra, thông tin về việc lọc trong số thông tin được giải mã bởi bộ giải mã entropy 310 có thể được cấp đến bộ lọc 350. Trong khi đó, bộ thu (không được thể hiện) để thu tín hiệu được xuất ra từ thiết bị mã hóa có thể còn được tạo cấu hình như phần tử bên trong/bên ngoài của thiết bị giải mã 300, hoặc bộ thu có thể là thành phần của bộ giải mã entropy 310. Trong khi đó, thiết bị giải mã theo sáng chế này có thể được đề cập đến như là thiết bị giải mã video/ảnh/hình ảnh, và thiết bị giải mã có thể được phân loại thành bộ giải mã thông tin (bộ giải mã thông tin video/ảnh/hình ảnh) và bộ giải mã mẫu (bộ giải mã mẫu video/ảnh/hình ảnh). Bộ giải mã thông tin có thể bao gồm bộ giải mã entropy 310, và bộ giải mã mẫu có thể bao gồm ít nhất một trong bộ giải lượng tử hóa 321, bộ biến đổi ngược 322, bộ cộng 340, bộ lọc 350, bộ nhớ 360, bộ liên dự đoán 332, và bộ nội dự đoán 331.

Bộ giải lượng tử hóa 321 có thể giải lượng tử hóa các hệ số biến đổi được lượng tử hóa và xuất ra các hệ số biến đổi. Bộ giải lượng tử hóa 321 có thể sắp xếp lại các hệ số biến đổi được lượng tử hóa dưới dạng của loại khối hai chiều. Trong trường hợp này, việc sắp xếp lại có thể được thực hiện dựa trên thứ tự quét hệ số được thực hiện trong thiết bị mã hóa. Bộ giải lượng tử hóa 321 có thể thực hiện việc giải lượng tử hóa trên các hệ số biến đổi được lượng tử hóa bằng cách sử dụng tham số lượng tử hóa (ví dụ, thông tin kích thước bước lượng tử hóa) và thu được các hệ số biến đổi.

Bộ biến đổi ngược 322 biến đổi ngược các hệ số biến đổi để thu được tín hiệu phần dư (khối phần dư, mảng mẫu phần dư).

Bộ dự đoán có thể thực hiện việc dự đoán trên khối hiện tại và tạo ra khối được dự

đoán bao gồm các mẫu dự đoán cho khối hiện tại. Bộ dự đoán có thể xác định liệu việc nội dự đoán hay việc liên dự đoán được áp dụng cho khối hiện tại dựa trên thông tin về dự đoán được xuất ra từ bộ giải mã entropy 310 và có thể xác định chế độ nội/liên dự đoán cụ thể.

Bộ dự đoán 320 có thể tạo ra tín hiệu dự đoán dựa trên các phương pháp dự đoán khác nhau được mô tả bên dưới. Ví dụ, bộ dự đoán có thể không chỉ áp dụng việc nội dự đoán hoặc liên dự đoán để dự đoán một khối mà còn có thể áp dụng đồng thời việc nội dự đoán và liên dự đoán. Điều này có thể được gọi là liên dự đoán và nội dự đoán kết hợp (CIIP). Ngoài ra, bộ dự đoán có thể được dựa trên chế độ nội dự đoán sao chép khối (IBC) hoặc chế độ bảng màu để dự đoán khối. Chế độ dự đoán IBC hoặc chế độ bảng màu có thể được sử dụng cho việc lập mã ảnh/video nội dung của trò chơi hoặc dạng tương tự, ví dụ, lập mã nội dung màn (SCC). Về cơ bản, IBC thực hiện việc dự đoán trong hình ảnh hiện tại nhưng có thể được thực hiện tương tự cho việc liên dự đoán trong đó khối tham chiếu được dẫn xuất trong hình ảnh hiện tại. Tức là, IBC có thể sử dụng ít nhất một trong các kỹ thuật liên dự đoán được mô tả trong sáng chế. Chế độ bảng màu có thể được xem xét như là ví dụ của việc nội lập mã hoặc việc nội dự đoán. Khi chế độ bảng màu được áp dụng, giá trị mẫu nằm trong hình ảnh có thể được báo hiệu dựa trên thông tin về bảng của bảng màu và chỉ số bảng màu.

Bộ nội dự đoán 331 có thể dự đoán khối hiện tại bằng cách tham chiếu đến các mẫu trong hình ảnh hiện tại. Các mẫu được tham chiếu có thể được đặt trong vùng lân cận của khối hiện tại hoặc có thể đặt tách ra theo chế độ dự đoán. Trong việc nội dự đoán, các chế độ dự đoán có thể bao gồm nhiều chế độ không có hướng và nhiều chế độ có hướng. Bộ nội dự đoán 331 có thể xác định chế độ dự đoán được áp dụng cho khối hiện tại bằng cách sử dụng chế độ dự đoán được áp dụng cho khối lân cận.

Bộ liên dự đoán 332 có thể dẫn xuất khối được dự đoán cho khối hiện tại dựa trên khối tham chiếu (mảng mẫu tham chiếu) được đặc tả bằng vector chuyển động trên hình ảnh tham chiếu. Trong trường hợp này, để làm giảm lượng thông tin chuyển động được truyền trong chế độ liên dự đoán, thông tin chuyển động có thể được dự đoán trong các đơn vị là các khối, các khối con, hoặc các mẫu dựa trên mối tương quan của thông tin chuyển động giữa khối lân cận và khối hiện tại. Thông tin chuyển động có thể bao gồm vector chuyển động và chỉ số hình ảnh tham chiếu. Thông tin chuyển động có thể còn

bao gồm thông tin hướng liên dự đoán (dự đoán L0, dự đoán L1, song dự đoán, v.v.). Trong trường hợp liên dự đoán, khối lân cận có thể bao gồm khối lân cận không gian có mặt trong hình ảnh hiện tại và khối lân cận thời gian có mặt trong hình ảnh tham chiếu. Ví dụ, bộ liên dự đoán 332 có thể tạo cấu hình danh sách ứng viên thông tin chuyển động dựa trên các khối lân cận và dẫn xuất vector chuyển động của khối hiện tại và/hoặc chỉ số hình ảnh tham chiếu dựa trên thông tin chọn ứng viên nhận được. Việc liên dự đoán có thể được thực hiện dựa trên các chế độ dự đoán khác nhau, và thông tin về việc dự đoán có thể bao gồm thông tin chỉ báo chế độ liên dự đoán cho khối hiện tại.

Bộ cộng 340 có thể tạo ra tín hiệu được tái dựng (hình ảnh được tái dựng, khối được tái dựng, mảng mẫu được tái dựng) bằng cộng tín hiệu phần dư thu được vào tín hiệu dự đoán (khối được dự đoán, mảng mẫu được dự đoán) được xuất ra từ bộ dự đoán (bao gồm bộ liên dự đoán 332 và/hoặc bộ nội dự đoán 331). Nếu không có phần dư nào cho khối cần được xử lý, như khi chế độ bỏ cóc được áp dụng, thì khối được dự đoán có thể được sử dụng như khối được tái dựng.

Bộ cộng 340 có thể được gọi là bộ tái dựng hoặc bộ tạo khối được tái dựng. Tín hiệu được tái dựng được tạo ra có thể được sử dụng cho việc nội dự đoán khối tiếp theo cần được xử lý trong hình ảnh hiện tại, có thể được xuất ra qua bước lọc được mô tả ở trên, hoặc có thể được sử dụng cho việc liên dự đoán hình ảnh tiếp theo.

Trong khi đó, phép ánh xạ độ chói cùng với việc định tỷ lệ sắc độ (LMCS) có thể được áp dụng trong quy trình giải mã ảnh.

Bộ lọc 350 có thể nâng cao chất lượng ảnh chủ quan/khách quan bằng cách áp dụng việc lọc cho tín hiệu được tái dựng. Ví dụ, bộ lọc 350 có thể tạo ra hình ảnh được tái dựng được điều chỉnh bằng cách áp dụng các phương pháp lọc khác nhau đối với hình ảnh được tái dựng và lưu trữ hình ảnh được tái dựng được điều chỉnh trong bộ nhớ 360, cụ thể, DPB của bộ nhớ 360. Các phương pháp lọc khác nhau có thể bao gồm, ví dụ, lọc khử khối, bù tương thích mẫu, bộ lọc vòng lặp tương thích, bộ lọc song phương, và dạng tương tự.

Hình ảnh được tái dựng (được điều chỉnh) được lưu trữ trong DPB của bộ nhớ 360 có thể được sử dụng như hình ảnh tham chiếu trong bộ liên dự đoán 332. Bộ nhớ 360 có thể lưu trữ thông tin chuyển động của khối mà thông tin chuyển động trong hình ảnh

hiện tại được dẫn xuất (hoặc được giải mã) từ đó và/hoặc thông tin chuyển động của các khối trong hình ảnh mà đã được tái dựng. Thông tin chuyển động được lưu trữ có thể được truyền đến bộ liên dự đoán 260 để được sử dụng như thông tin chuyển động của khối lân cận không gian hoặc thông tin chuyển động của khối lân cận thời gian. Bộ nhớ 360 có thể lưu trữ các mẫu được tái dựng của các khối được tái dựng trong hình ảnh hiện tại và chuyển các mẫu được tái dựng đến bộ nội dự đoán 331.

Trong sáng chế này, các phương án được mô tả trong bộ lọc 260, bộ liên dự đoán 221, và bộ nội dự đoán 222 của thiết bị mã hóa 200 có thể là giống như hoặc được áp dụng lần lượt để tương ứng với bộ lọc 350, bộ liên dự đoán 332, và bộ nội dự đoán 331 của thiết bị giải mã 300. Nguyên tắc tương tự có thể cũng áp dụng cho đơn vị 332 và bộ nội dự đoán 331.

Trong sáng chế này, ít nhất một trong việc lượng tử hóa/lượng tử hóa ngược và/hoặc biến đổi/biến đổi ngược có thể được lược bỏ. Khi việc lượng tử hóa/lượng tử hóa ngược được lược bỏ, thì các hệ số biến đổi được lượng tử hóa có thể được gọi là các hệ số biến đổi. Khi việc biến đổi/biến đổi ngược được lược bỏ, thì các hệ số biến đổi có thể được gọi là các hệ số hoặc các hệ số dư, hoặc có thể vẫn được gọi là các hệ số biến đổi cho sự đồng nhất về diễn đạt.

Trong sáng chế này, hệ số biến đổi được lượng tử hóa và hệ số biến đổi có thể được đề cập đến lần lượt như hệ số biến đổi và hệ số được định tỷ lệ. Trong trường hợp này, thông tin phần dư có thể bao gồm thông tin về hệ số (các hệ số) biến đổi, và thông tin về hệ số (các hệ số) biến đổi có thể được báo hiệu qua cú pháp lập mã phần dư. Các hệ số biến đổi có thể được dẫn xuất dựa trên thông tin dư (hoặc thông tin về hệ số (các hệ số) biến đổi), và các hệ số biến đổi được định tỷ lệ có thể được dẫn xuất bằng cách biến đổi ngược (định tỷ lệ) trên các hệ số biến đổi. Các mẫu phần dư có thể được dẫn xuất dựa trên việc biến đổi ngược (biến đổi) trên các hệ số biến đổi được định tỷ lệ. Điều này có thể cũng được áp dụng/được thể hiện trong các phần khác của sáng chế này.

Trong khi đó, như được mô tả nêu trên, bộ lượng tử hóa của thiết bị mã hóa có thể dẫn xuất các hệ số biến đổi được lượng tử hóa bằng cách áp dụng việc lượng tử hóa cho các hệ số biến đổi. Bộ giải lượng tử hóa của thiết bị mã hóa hoặc bộ giải lượng tử hóa của thiết bị giải mã có thể dẫn xuất các hệ số biến đổi bằng cách áp dụng việc giải lượng tử hóa cho các hệ số biến đổi được lượng tử hóa.

Nhìn chung, trong việc lập mã video/ảnh, tỷ lệ lượng tử hóa có thể được thay đổi, và tốc độ nén có thể được điều chỉnh sử dụng tỷ lệ lượng tử hóa được thay đổi này. Trong khía cạnh thực hiện, tham số lượng tử hóa (quantization parameter, QP) có thể được sử dụng thay vì trực tiếp sử dụng tỷ lệ lượng tử hóa khi xét đến độ phức tạp. Ví dụ, các tham số lượng tử hóa có các giá trị nguyên từ 0 đến 63 có thể được sử dụng, và mỗi giá trị tham số lượng tử hóa có thể tương ứng với tỷ lệ lượng tử hóa thực sự. Thêm nữa, ví dụ, tham số lượng tử hóa QP_Y cho thành phần độ chói và tham số lượng tử hóa QP_C cho thành phần sắc độ có thể được tạo cấu hình khác nhau.

Trong quy trình lượng tử hóa, hệ số biến đổi C có thể là đầu vào, tỷ lệ lượng tử hóa (Q_{step}) có thể được chia, và hệ số biến đổi được lượng tử hóa C' có thể thu được dựa trên tỷ lệ lượng tử hóa. Trong trường hợp này, tỷ lệ lượng tử hóa có thể được tạo ra dưới dạng số nguyên bằng cách nhân tỷ lệ lượng tử hóa với một thang bằng việc xét đến độ phức tạp tính toán, và hoạt động vận hành luân phiên có thể được thực hiện bằng giá trị tương ứng với giá trị thang. Thang lượng tử hóa có thể được dẫn xuất dựa trên tích số của tỷ lệ lượng tử hóa và giá trị thang. Tức là, thang lượng tử hóa có thể được dẫn xuất dựa trên QP. Ví dụ, thang lượng tử hóa có thể được áp dụng cho hệ số biến đổi C' , và hệ số biến đổi được lượng tử hóa C' có thể được dẫn xuất dựa trên kết quả áp dụng.

Quy trình giải lượng tử hóa là quy trình ngược của quy trình lượng tử hóa. Trong quy trình này, hệ số biến đổi được lượng tử hóa C' có thể được nhân với tỷ lệ lượng tử hóa (Q_{step}), và hệ số biến đổi được tái dựng C'' có thể thu được dựa trên kết quả của phép nhân. Trong trường hợp này, thang cấp độ có thể được dẫn xuất dựa trên tham số lượng tử hóa, thang cấp độ có thể được áp dụng cho hệ số biến đổi được lượng tử hóa C' , và hệ số biến đổi được tái dựng C'' có thể được dẫn xuất. Hệ số biến đổi được tái dựng C'' có thể có một số chênh lệch so với hệ số biến đổi thứ nhất C do sự suy hao trong quy trình biến đổi và/hoặc lượng tử hóa. Theo đó, việc giải lượng tử hóa được thực hiện trong thiết bị mã hóa như trong thiết bị giải mã.

Trong khi đó, công nghệ lượng tử hóa đo tần số thích ứng để điều chỉnh độ mạnh lượng tử hóa phụ thuộc vào tần số có thể được áp dụng. Công nghệ lượng tử hóa đo tần số thích ứng là phương pháp áp dụng độ mạnh lượng tử hóa theo cách khác nhau cho mỗi tần số. Trong khi lượng tử hóa đo tần số thích ứng, độ mạnh lượng tử hóa cho mỗi tần số có thể được áp dụng theo cách khác nhau sử dụng ma trận định tỷ lệ lượng tử hóa

định trước. Tức là, quy trình lượng tử hóa/giải lượng tử hóa được đề cập ở trên có thể được thực hiện dựa trên ma trận định tỷ lệ lượng tử hóa. Ví dụ, để tạo ra kích thước của khối hiện tại và/hoặc tín hiệu phần dư của khối hiện tại, ma trận định tỷ lệ lượng tử hóa khác nhau có thể được sử dụng phụ thuộc vào việc liệu chế độ dự đoán được áp dụng cho khối hiện tại là liên dự đoán hay nội dự đoán. Ma trận định tỷ lệ lượng tử hóa có thể được gọi là ma trận lượng tử hóa hoặc ma trận định tỷ lệ. Ma trận định tỷ lệ lượng tử hóa có thể được xác định trước. Thêm nữa, đối với việc định tỷ lệ thích ứng tần số, thông tin thang lượng tử hóa cho mỗi tần số cho ma trận định tỷ lệ lượng tử hóa có thể được tái dựng/được mã hóa trong thiết bị mã hóa và được báo hiệu đến thiết bị giải mã. Thông tin thang lượng tử hóa cho mỗi tần số có thể được gọi là thông tin định tỷ lệ lượng tử hóa. Thông tin thang lượng tử hóa cho mỗi tần số có thể bao gồm dữ liệu danh sách định tỷ lệ (`scaling_list_data`). Ma trận định tỷ lệ lượng tử hóa (được điều chỉnh) có thể được dẫn xuất dựa trên dữ liệu danh sách định tỷ lệ. Thêm nữa, thông tin thang lượng tử hóa cho mỗi tần số có thể bao gồm thông tin cờ biểu diễn liệu dữ liệu danh sách định tỷ lệ là có mặt hay không. Thay vào đó, nếu dữ liệu danh sách định tỷ lệ được báo hiệu trong cấp cao hơn (ví dụ, SPS), thì thông tin biểu diễn liệu dữ liệu danh sách định tỷ lệ được điều chỉnh trong cấp thấp hơn (ví dụ, PPS hoặc tiêu đề nhóm ô xếp, v.v.) của cấp cao hơn, v.v. có thể còn được bao gồm.

Như trong các nội dung được đề cập đến ở trên, việc lượng tử hóa/giải lượng tử hóa có thể được áp dụng cho thành phần độ chói và thành phần sắc độ dựa trên tham số lượng tử hóa.

Các tham số lượng tử hóa cho đơn vị lập mã có thể được xác định dựa trên thông tin được báo hiệu trong hình ảnh và/hoặc cấp lát. Ví dụ, các tham số lượng tử hóa có thể được dẫn xuất như trong các nội dung được mô tả sau đây.

Ví dụ, thông tin liên quan đến việc dẫn xuất các tham số lượng tử hóa có thể được báo hiệu như trong bảng sau đây thông qua tập tham số trình tự (SPS).

Bảng 1

seq_parameter_set_rbsp() {	Descriptor
sps_decoding_parameter_set_id	u(4)
...	
bit_depth_luma_minus8	uc(v)
bit_depth_chroma_minus8	ue(v)
...	
rbsp_trailing_bits()	
}	

Các ngữ nghĩa của các phần tử cú pháp trong Bảng 1 có thể là giống như bảng sau đây.

Bảng 2

bit_depth_luma_minus8 specifies the bit depth of the samples of the luma array BitDepth_Y and the value of the luma quantization parameter range offset QpBdOffset_Y as follows: $\text{BitDepth}_Y = 8 + \text{bit_depth_luma_minus8}$ $\text{QpBdOffset}_Y = 6 * \text{bit_depth_luma_minus8}$ $\text{bit_depth_luma_minus8}$ shall be in the range of 0 to 8, inclusive. bit_depth_chroma_minus8 specifies the bit depth of the samples of the chroma arrays BitDepth_C and the value of the chroma quantization parameter range offset QpBdOffset_C as follows: $\text{BitDepth}_C = 8 + \text{bit_depth_chroma_minus8}$ $\text{QpBdOffset}_C = 6 * \text{bit_depth_chroma_minus8}$ $\text{bit_depth_chroma_minus8}$ shall be in the range of 0 to 8, inclusive.
--

Ví dụ, phần tử cú pháp $\text{bit_depth_luma_minus8}$ có thể biểu diễn BitDepth_Y , tức là, chiều sâu bit của các mẫu của mảng độ chói, và QpBdOffset_Y mà là độ bù ngưỡng tham số lượng tử hóa độ chói. Tức là, ví dụ, BitDepth_Y và QpBdOffset_Y có thể được dẫn xuất dựa trên phần tử cú pháp $\text{bit_depth_luma_minus8}$. Ví dụ, BitDepth_Y có thể được dẫn xuất như giá trị thu được bằng cách cộng 8 vào giá trị của phần tử cú pháp $\text{bit_depth_luma_minus8}$. QpBdOffset_Y có thể được dẫn xuất như giá trị thu được bằng cách nhân giá trị của phần tử cú pháp $\text{bit_depth_luma_minus8}$ với 6. Thêm nữa, $\text{bit_depth_luma_minus8}$ có thể là trong phạm vi từ 0 đến 8.

Thêm nữa, ví dụ, phần tử cú pháp `bit_depth_chroma_minus8` có thể biểu diễn `BitDepthc`, tức là, chiều sâu bit của các mẫu của mảng sắc độ, và `QpBdOffsetc`, tức là, độ bù phạm vi tham số lượng tử hóa sắc độ. Tức là, ví dụ, `BitDepthc` và `QpBdOffsetc` có thể được dẫn xuất dựa trên phần tử cú pháp `bit_depth_chroma_minus8`. Ví dụ, `BitDepthc` có thể được dẫn xuất như giá trị thu được bằng cách cộng 8 vào giá trị của phần tử cú pháp `bit_depth_chroma_minus8`. `QpBdOffsetc` có thể được dẫn xuất như giá trị thu được bằng cách nhân giá trị của phần tử cú pháp `bit_depth_chroma_minus8` với 6. Thêm nữa, `bit_depth_chroma_minus8` có thể là trong phạm vi từ 0 đến 8.

Thêm nữa, thông tin liên quan đến việc dẫn xuất các tham số lượng tử hóa có thể được báo hiệu như trong bảng sau đây, ví dụ, thông qua tập tham số hình ảnh (picture parameter set, PPS). Thông tin có thể bao gồm độ bù sắc độ Cb, độ bù sắc độ Cr, độ bù sắc độ kết hợp, và tham số lượng tử hóa ban đầu. Tức là, thông tin có thể bao gồm các phần tử cú pháp cho độ bù sắc độ Cb, độ bù sắc độ Cr, độ bù sắc độ kết hợp, và tham số lượng tử hóa ban đầu.

Bảng 3

<code>pic_parameter_set_rbsp()</code> {	Descriptor
<code>pps_pic_parameter_set_id</code>	<code>ue(v)</code>
...	
<code>init_qp_minus26</code>	<code>se(v)</code>
<code>transform_skip_enabled_flag</code>	<code>u(1)</code>
<code>pps_cb_qp_offset</code>	<code>se(v)</code>
<code>pps_cr_qp_offset</code>	<code>se(v)</code>
<code>pps_joint_cbr_qp_offset</code>	<code>se(v)</code>
<code>pps_slice_chroma_qp_offsets_present_flag</code>	<code>u(1)</code>
...	
<code>rbp_trailing_bits()</code>	
}	

Các ngữ nghĩa của các phần tử cú pháp trong Bảng 3 có thể là giống như bảng sau đây.

Bảng 4

init_qp_minus26 plus 26 specifies the initial value of SliceQp_Y for each slice referring to the PPS. The initial value of SliceQp_Y is modified at the slice layer when a non-zero value of slice_qp_delta is decoded. The value of init_qp_minus26 shall be in the range of $-(26 + \text{QpBdOffset}_Y)$ to +37, inclusive.

pps_cb_qp_offset and **pps_cr_qp_offset** specify the offsets to the luma quantization parameter Qp'_Y used for deriving Qp'_{Cb} and Qp'_{Cr} , respectively. The values of **pps_cb_qp_offset** and **pps_cr_qp_offset** shall be in the range of -12 to +12, inclusive. When **ChromaArrayType** is equal to 0, **pps_cb_qp_offset** and **pps_cr_qp_offset** are not used in the decoding process and decoders shall ignore their value.

pps_joint_cbr_qp_offset specifies the offset to the luma quantization parameter Qp'_Y used for deriving Qp'_{CbCr} . The value of **pps_joint_cbr_qp_offset** shall be in the range of -12 to +12, inclusive. When **ChromaArrayType** is equal to 0, **pps_joint_cbr_qp_offset** is not used in the decoding process and decoders shall ignore its value.

pps_slice_chroma_qp_offsets_present_flag equal to 1 indicates that the **slice_cb_qp_offset** and **slice_cr_qp_offset** syntax elements are present in the associated slice headers. **pps_slice_chroma_qp_offsets_present_flag** equal to 0 indicates that these syntax elements are not present in the associated slice headers. When **ChromaArrayType** is equal to 0, **pps_slice_chroma_qp_offsets_present_flag** shall be equal to 0.

Ví dụ, giá trị thu được bằng cách cộng 26 vào phần tử cú pháp **init_qp_minus26** có thể biểu diễn giá trị ban đầu của SliceQp_Y cho mỗi lát mà đề cập đến PPS. Nếu giá trị khác không của **slice_qp_delta** được giải mã, thì giá trị ban đầu của SliceQp_Y có thể được điều chỉnh trong lớp lát. **init_qp_minus26 0** có thể là trong phạm vi là $-(26 + \text{QpBdOffset}_Y)$ đến +37.

Thêm nữa, ví dụ, các phần tử cú pháp **pps_cb_qp_offset** và **pps_cr_qp_offset** có thể biểu diễn các độ bù cho tham số lượng tử hóa độ chói Qp'_Y được sử dụng lần lượt để dẫn xuất Qp'_{Cb} và Qp'_{Cr} . **pps_cb_qp_offset** và **pps_cr_qp_offset** có thể là trong phạm vi là -12 đến +12. Thêm nữa, khi **ChromaArrayType** là 0, trong quy trình giải mã, **pps_cb_qp_offset** và **pps_cr_qp_offset** có thể không được sử dụng, và thiết bị giải mã có thể bỏ qua các giá trị của các phần tử cú pháp.

Thêm nữa, ví dụ, phần tử cú pháp **pps_joint_cbr_qp_offset** có thể biểu diễn độ dịch cho tham số lượng tử hóa độ chói Qp'_Y được sử dụng để dẫn xuất Qp'_{CbCr} . **pps_joint_cbr_qp_offset** có thể là trong phạm vi là -12 đến +12. Thêm nữa, khi **ChromaArrayType** là 0, trong quy trình giải mã, **pps_joint_cbr_qp_offset** có thể không được sử dụng, và thiết bị giải mã có thể bỏ qua giá trị của phần tử cú pháp.

Thêm nữa, ví dụ, phần tử cú pháp `pps_slice_chroma_qp_offsets_present_flag` có thể biểu diễn liệu các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` có mặt trong các tiêu đề lát liên quan đến các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` hay không. Ví dụ, `pps_slice_chroma_qp_offsets_present_flag` có giá trị là 1 có thể biểu diễn rằng các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` là có mặt trong các tiêu đề lát liên quan đến các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset`. Thêm nữa, ví dụ, `pps_slice_chroma_qp_offsets_present_flag` có giá trị là 0 có thể biểu diễn rằng các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset` là không có mặt trong các tiêu đề lát liên quan đến các phần tử cú pháp `slice_cb_qp_offset` và `slice_cr_qp_offset`. Thêm nữa, khi `ChromaArrayType` là 0, trong quy trình giải mã, `pps_slice_chroma_qp_offsets_present_flag` có thể là giống nhau như 0.

Như trong các nội dung được đề cập đến ở trên, các phần tử cú pháp được phân tích cú pháp trong PPS có thể là `init_qp_minus26`, `pps_cb_qp_offset`, `pps_cr_qp_offset`, `pps_joint_cbr_qp_offset`, và `pps_slice_chroma_qp_offsets_present_flag`. Phần tử cú pháp `init_qp_minus26` có thể biểu diễn giá trị ban đầu của `SliceQpY` cho mỗi lát mà đề cập đến PPS. Thêm nữa, các phần tử cú pháp `pps_cb_qp_offset`, `pps_cr_qp_offset`, và `pps_joint_cbr_qp_offset` có thể biểu diễn các độ dịch cho tham số lượng tử hóa độ chói $Qp'Y$. Thêm nữa, phần tử cú pháp `pps_slice_chroma_qp_offsets_present_flag` có thể biểu diễn liệu tham số độ bù có mặt trong tiêu đề lát hay không.

Thêm nữa, thông tin liên quan đến việc dẫn xuất các tham số lượng tử hóa có thể được báo hiệu như trong bảng sau đây thông qua tiêu đề lát chẳng hạn.

Bảng 5

	Descriptor
<code>slice_header() {</code>	
<code>slice_pic_parameter_set_id</code>	<code>uc(v)</code>
...	
<code>slice_qp_delta</code>	<code>se(v)</code>
<code>if(pps_slice_chroma_qp_offsets_present_flag) {</code>	
<code>slice_cb_qp_offset</code>	<code>se(v)</code>
<code>slice_cr_qp_offset</code>	<code>se(v)</code>
<code>slice_joint_cbr_qp_offset</code>	<code>se(v)</code>
<code>}</code>	
...	
<code>byte_alignment()</code>	
<code>}</code>	

Các ngữ nghĩa của các phần tử cú pháp trong Bảng 5 có thể là giống như bảng sau đây.

Bảng 6

slice_qp_delta specifies the initial value of Qp_Y to be used for the coding blocks in the slice until modified by the value of $CuQpDeltaVal$ in the coding unit layer. The initial value of the Qp_Y quantization parameter for the slice, $SliceQp_Y$, is derived as follows:

$$SliceQp_Y = 26 + init_qp_minus26 + slice_qp_delta$$

The value of $SliceQp_Y$ shall be in the range of $-QpBdOffset_Y$ to +63, inclusive.

slice_cb_qp_offset specifies a difference to be added to the value of $pps_cb_qp_offset$ when determining the value of the Qp'_{Cb} quantization parameter. The value of $slice_cb_qp_offset$ shall be in the range of -12 to +12, inclusive. When $slice_cb_qp_offset$ is not present, it is inferred to be equal to 0. The value of $pps_cb_qp_offset + slice_cb_qp_offset$ shall be in the range of -12 to +12, inclusive.

slice_cr_qp_offset specifies a difference to be added to the value of $pps_cr_qp_offset$ when determining the value of the Qp'_{Cr} quantization parameter. The value of $slice_cr_qp_offset$ shall be in the range of -12 to +12, inclusive. When $slice_cr_qp_offset$ is not present, it is inferred to be equal to 0. The value of $pps_cr_qp_offset + slice_cr_qp_offset$ shall be in the range of -12 to +12, inclusive.

slice_joint_cbr_qp_offset specifies a difference to be added to the value of $pps_joint_cbr_qp_offset$ when determining the value of the Qp'_{CbCr} . The value of $slice_joint_cbr_qp_offset$ shall be in the range of -12 to +12, inclusive. When $slice_joint_cbr_qp_offset$ is not present, it is inferred to be equal to 0. The value of $pps_joint_cbr_qp_offset + slice_joint_cbr_qp_offset$ shall be in the range of -12 to +12, inclusive.

The derivation process for the luma and chroma quantization parameters begin with the inputs to the process being a luma location, variables specifying the width and height of the current coding block and a variable specifying whether it is a single or a dual tree. The luma, chroma and the joint chroma quantization parameters are denoted as follows: Qp'_Y , Qp'_{Cb} , Qp'_{Cr} and Qp'_{CbCr} .

cu_qp_delta_sign_flag specifies the sign of **CuQpDeltaVal** as follows:

- If **cu_qp_delta_sign_flag** is equal to 0, the corresponding **CuQpDeltaVal** has a positive value.
- Otherwise (**cu_qp_delta_sign_flag** is equal to 1), the corresponding **CuQpDeltaVal** has a negative value.

When **cu_qp_delta_sign_flag** is not present, it is inferred to be equal to 0.

When **cu_qp_delta_abs** is present, the variables **IsCuQpDeltaCoded** and **CuQpDeltaVal** are derived as follows:

$$\text{IsCuQpDeltaCoded} = 1$$

$$\text{CuQpDeltaVal} = \text{cu_qp_delta_abs} * (1 - 2 * \text{cu_qp_delta_sign_flag})$$

The value of **CuQpDeltaVal** shall be in the range of $-(32 + \text{QpBdOffsetY} / 2)$ to $+(31 + \text{QpBdOffsetY} / 2)$, inclusive.

Ví dụ, **slice_qp_delta** có thể biểu diễn giá trị ban đầu của Q_{pY} cần được sử dụng trong khối lập mã trong lát cho đến khi nó được điều chỉnh bằng giá trị của **CuQpDeltaVal** trong lớp đơn vị lập mã. Ví dụ, giá trị ban đầu của Q_{pY} cho lát, **SliceQpY**, có thể được dẫn xuất như $26 + \text{init_qp_minus26} + \text{slice_qp_delta}$. Giá trị của **SliceQpY** có thể là trong phạm vi từ $-\text{QpBdOffsetY}$ đến $+63$.

Thêm nữa, ví dụ, **slice_cb_qp_offset** có thể biểu diễn mức chênh lệch cần được cộng vào giá trị của **pps_cb_qp_offset** khi giá trị của tham số lượng tử hóa $Q_{p'cb}$ được xác định. Giá trị của **slice_cb_qp_offset** có thể là trong phạm vi từ -12 đến $+12$. Thêm nữa, ví dụ, nếu **slice_cb_qp_offset** là không có mặt, thì **slice_cb_qp_offset** có thể được suy luận như 0. Giá trị của **pps_cb_qp_offset** + **slice_cb_qp_offset** có thể là trong phạm vi từ -12 đến $+12$.

Thêm nữa, ví dụ, **slice_cr_qp_offset** có thể biểu diễn mức chênh lệch cần được cộng vào giá trị của **pps_cr_qp_offset** khi giá trị của tham số lượng tử hóa $Q_{p'cr}$ được xác định. Giá trị của **slice_cr_qp_offset** có thể là trong phạm vi từ -12 đến $+12$. Thêm nữa, ví dụ, nếu **slice_cr_qp_offset** là không có mặt, thì **slice_cr_qp_offset** có thể được suy luận như 0. Giá trị của **pps_cr_qp_offset** + **slice_cr_qp_offset** có thể là trong phạm vi từ -12 đến $+12$.

Thêm nữa, ví dụ, **slice_cbr_qp_offset** có thể biểu diễn mức chênh lệch cần được

cộng vào giá trị của pps_cbr_qp_offset khi giá trị của tham số lượng tử hóa Qp'_{CbCr} được xác định. Giá trị của $\text{slice_cbr_qp_offset}$ có thể là trong phạm vi từ -12 đến +12. Thêm nữa, ví dụ, nếu $\text{slice_cbr_qp_offset}$ là không có mặt, thì $\text{slice_cbr_qp_offset}$ có thể được suy luận như 0. Giá trị của $\text{pps_cbr_qp_offset} + \text{slice_cbr_qp_offset}$ có thể là trong phạm vi từ 12 đến +12.

Quy trình dẫn xuất các tham số lượng tử hóa độ chói và sắc độ có thể được bắt đầu dựa trên thực tế là đầu vào cho quy trình này là vị trí độ chói, tham số để chỉ định chiều rộng và chiều cao của khối lập mã hiện tại, và tham số để chỉ định cây đơn hoặc cây kép. Trong khi đó, như trong các nội dung được đề cập đến ở trên, tham số lượng tử hóa độ chói, tham số lượng tử hóa sắc độ và tham số lượng tử hóa sắc độ hợp lại có thể được biểu diễn như Qp'_Y , Qp'_{Cb} , Qp'_{Cr} và Qp'_{CbCr} .

Trong khi đó, ví dụ, phần tử cú pháp $\text{cu_qp_delta_sign_flag}$ biểu diễn dấu của CuQpDeltaVal có thể được phân tích cú pháp. Ví dụ, $\text{cu_qp_delta_sign_flag}$ có thể biểu diễn dấu của CuQpDeltaVal như sau.

Ví dụ, khi $\text{cu_qp_delta_sign_flag}$ là 0, thì CuQpDeltaVal tương ứng với $\text{cu_qp_delta_sign_flag}$ có thể có giá trị dương. Thay vào đó, ví dụ, khi $\text{cu_qp_delta_sign_flag}$ là 1, thì CuQpDeltaVal tương ứng với $\text{cu_qp_delta_sign_flag}$ có thể có giá trị âm. Thêm nữa, nếu $\text{cu_qp_delta_sign_flag}$ là không có mặt, thì $\text{cu_qp_delta_sign_flag}$ có thể được suy luận như 0.

Thêm nữa, ví dụ, nếu cu_qp_delta_abs là có mặt, tham số IsCuQpDeltaCoded có thể được dẫn xuất như 1. Tham số CuQpDeltaVal có thể được dẫn xuất như $\text{cu_qp_delta_abs} * (1 - 2 * \text{cu_qp_delta_sign_flag})$. CuQpDeltaVal có thể là trong phạm vi là từ $-(32 + \text{QpBdOffset}_Y/2)$ đến $+(31 + \text{QpBdOffset}_Y/2)$.

Sau đó, ví dụ, tham số lượng tử hóa độ chói Qp'_Y có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 1

$$\text{Qp}_Y = ((\text{qp}_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y$$

Thêm nữa, nếu ChromaArrayType không là 0 và, treeType là SINGLE_TREE hay DUAL_TREE_CHROMA , thì được áp dụng như sau.

- Khi treeType là bằng DUAL_TREE_CHROMA , thì tham số Q_{pY} có thể được thiết đặt giống với tham số lượng tử hóa độ sáng Q_{pY} của đơn vị lập mã độ chói bao gồm vị trí độ chói ($x_{Cb} + cbWidth/2, y_{Cb} + cbHeight/2$).

- Các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất như sau.

Phương trình 2

$$qP_{iCb} = \text{Clip3}(-QpBdOffset_C, 69, Q_{pY} + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qP_{iCr} = \text{Clip3}(-QpBdOffset_C, 69, Q_{pY} + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qP_{iCbCr} = \text{Clip3}(-QpBdOffset_C, 69, Q_{pY} + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

Ví dụ, khi ChromaArrayType là 1, thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được thiết đặt giống với giá trị Q_{pC} được chỉ định trong Bảng 7 dựa trên các chỉ số qP_i lần lượt giống như qP_{iCb} , qP_{iCr} và qP_{iCbCr} .

Bảng 7

qP_i	< 30	30	31	32	33	34	35	36	37	38	39	40	41	42	43	> 43
Q_{pC}	= qP_i	29	30	31	32	33	33	34	34	35	35	36	36	37	37	= $qP_i - 6$

Thay vào đó, khi ChromaArrayType không là 1, thì các tham số qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được thiết đặt giống với $\text{Min}(qP_i, 63)$ dựa trên các chỉ số qP_i lần lượt giống như qP_{iCb} , qP_{iCr} , và qP_{iCbCr} .

- Các tham số lượng tử hóa sắc độ cho thành phần Cb và thành phần Cr, $Q_{p'Cb}$ và $Q_{p'Cr}$, và tham số lượng tử hóa sắc độ $Q_{p'CbCr}$ cho việc lập mã Cb-Cr hợp lại có thể được dẫn xuất như sau.

Phương trình 3

$$Q_{p'Cb} = qP_{Cb} + QpBdOffset_C$$

$$Q_{p'Cr} = qP_{Cr} + QpBdOffset_C$$

$$Q_{p'CbCr} = qP_{CbCr} + QpBdOffset_C$$

Trong khi đó, sáng chế này đề xuất các sơ đồ để nâng cao hiệu quả lập mã trong quy trình lượng tử hóa/giải lượng tử hóa.

Trong một phương án, sáng chế này đề xuất phương pháp để xác định và sử dụng bảng ánh xạ lượng tử hóa sắc độ được xác định bởi người dùng, không phải phương

pháp để thu được giá trị tham số lượng tử hóa sắc độ từ giá trị tham số lượng tử hóa độ chói thông qua bảng ánh xạ lượng tử hóa sắc độ được xác định trước trong VVC Draft5 v.7 hiện có khi ChromaArrayType không là 0 (ví dụ, khi ChromaArrayType là 1). Trong phần đặc tả VVC (ví dụ, VVC Draft5 v.7), khi qPi (giá trị tham số lượng tử hóa độ chói) được cho, thì Qpc (giá trị tham số lượng tử hóa sắc độ) được dẫn xuất thông qua bảng lượng tử hóa sắc độ định trước (ví dụ, Bảng 7), nhưng sáng chế này đề xuất phương pháp để dẫn xuất Qpc từ qPi dựa trên bảng ánh xạ lượng tử hóa sắc độ được xác định mới bởi người dùng. Theo một phương án của sáng chế này, có phương pháp trong đó giá trị Qpc có thể được dẫn xuất thông qua quan hệ hàm của giá trị qPi được cung cấp, hàm có thể được báo hiệu như cú pháp, như APS, SPS hoặc PPS, thông qua phương pháp chức năng được xác định bằng người dùng, mà bao gồm các giá trị truyền quan hệ hàm của các phần tử cú pháp định trước, và người dùng xác định sự ánh xạ bảng lượng tử hóa sắc độ dựa trên các giá trị được truyền. Ví dụ, bởi vì giá trị Qpc có thể được dẫn xuất thông qua quan hệ hàm có giá trị qPi, nếu các giá trị phần tử cú pháp biểu diễn hàm tương ứng được truyền, bảng ánh xạ lượng tử hóa sắc độ được xác định bởi người dùng có thể được dẫn xuất dưới một hình thức, như trong Bảng 7.

Trong một phương án, sơ đồ để báo hiệu thông tin về các phần tử cú pháp (Qpc_data) biểu diễn hàm liên quan đến việc ánh xạ lượng tử hóa sắc độ như trong bảng sau đây mà sẽ được mô tả sau trong tập tham số thích ứng (APS).

Bảng 8

<code>adaptation_parameter_set_rbsp() {</code>	Descriptor
<code>adaptation_parameter_set_id</code>	<code>u(5)</code>
<code>aps_params_type</code>	<code>u(3)</code>
<code>if(aps_params_type == ALF_APS)</code>	
<code>alf_data(adaptation_parameter_set_id)</code>	
<code>else if(aps_params_type == LMCS_APS)</code>	
<code>lmcs_data()</code>	
<code>else if(aps_params_type == Qpc_APS) // 2</code>	
<code>Qpc_data()</code>	
<code>aps_extension_flag</code>	<code>u(1)</code>
<code>if(aps_extension_flag)</code>	
<code>while(more_rbsp_data())</code>	
<code>aps_extension_data_flag</code>	<code>u(1)</code>
<code>rbp_trailing_bits()</code>	
<code>}</code>	

Tham chiếu đến Bảng 8, nếu `aps_params_type` biểu diễn `Qpc_APS`, chẳng hạn, khi giá trị của `aps_params_type` là 2, thì `Qpc_data()` có thể được báo hiệu.

Các ngữ nghĩa của các phần tử cú pháp trong Bảng 8 có thể là giống như bảng sau đây.

Bảng 9

adaptation_parameter_set_id provides an identifier for the APS for reference by other syntax elements.

NOTE – APSs can be shared across pictures and can be different in different slices within a picture.

aps_extension_flag equal to 0 specifies that no `aps_extension_data_flag` syntax elements are present in the APS RBSP syntax structure. `aps_extension_flag` equal to 1 specifies that there are `aps_extension_data_flag` syntax elements present in the APS RBSP syntax structure.

aps_extension_data_flag may have any value. Its presence and value do not affect decoder conformance to profiles specified in this version of this Specification. Decoders conforming to this version of this Specification shall ignore all `aps_extension_data_flag` syntax elements.

aps_params_type specifies the type of APS parameters carried in the APS as specified in the Table 2 shown below.

Ví dụ, phần tử cú pháp `adaptation_parameter_set_id` có thể cung cấp bộ định danh của APS được tham chiếu bởi các phần tử cú pháp khác.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_flag` có thể biểu diễn liệu các phần tử cú pháp `aps_extension_data_flag` có mặt trong cấu trúc cú pháp RBSP APS hay không. Ví dụ, phần tử cú pháp `aps_extension_flag` có giá trị là 1 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS. Phần tử cú pháp `aps_extension_flag` có giá trị là 0 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là không có mặt trong cấu trúc cú pháp RBSP APS.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_data_flag` có thể có giá trị bất kỳ. Sự có mặt (sự có mặt và giá trị) của `aps_extension_data_flag` có thể không ảnh hưởng đến sự thích ứng giải mã cho đối tượng (profile) được đặc tả trong một phiên bản của chuẩn này. Ví dụ, thiết bị giải mã mà theo một phiên bản của chuẩn này có thể bỏ qua tất cả các phần tử cú pháp `aps_extension_data_flag`.

Thêm nữa, ví dụ, phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các tham số APS được bao gồm trong APS như được minh họa trong Bảng 10.

Bảng 10

aps_params_type	Name of aps_params_type	Type of APS parameters
0	ALF_APS	ALF parameters
1	LMCS_APS	LMCS parameters
2	Qp _C _APS	Qpc data parameters
3..7	Reserved	Reserved

Ví dụ, tham chiếu đến Bảng 10, khi giá trị của phần tử cú pháp `aps_params_type` là 0, thì phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các tham số APS là các tham số ALF. Khi giá trị của phần tử cú pháp `aps_params_type` là 1, thì phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các tham số APS là các tham số LMCS. Khi giá trị của phần tử cú pháp `aps_params_type` là 2, thì phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các tham số APS là các tham số dữ liệu Qpc. Tham số dữ liệu Qpc có thể biểu diễn tham số dữ liệu lượng tử hóa sắc độ.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất sơ đồ để báo hiệu dữ liệu Qpc được xác định bằng người dùng trong tập tham số hình ảnh (PPS). Như ví dụ để thực hiện sơ đồ được đề xuất trong phương án này, cờ biểu diễn liệu PPS có bao gồm dữ liệu được xác định bởi người dùng trong SPS hay không có thể được giới thiệu. Tức là, cờ biểu diễn liệu PPS có bao gồm dữ liệu được xác định bởi người dùng trong SPS hay không, có thể được báo hiệu. Thêm nữa, theo phương án này, dữ liệu được xác định bởi người dùng có thể được báo hiệu trong PPS. Thay vào đó, dữ liệu được xác định bởi người dùng có thể được báo hiệu trong tiêu đề lát và/hoặc tập tiêu đề khác.

Cờ biểu diễn liệu PPS có bao gồm dữ liệu được xác định bởi người dùng hay không có thể được báo hiệu trong bảng sau đây.

Bảng 11

seq_parameter_set_rbsp() {	Descriptor
sps_decoding_parameter_set_id	u(4)
...	
Qpc_data_default_flag	u(1)
...	
rbsp_trailing_bits()	
}	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể là phần tử cú pháp của cờ được đề cập đến ở trên. Phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu các tham số `Qpc_data()` là có mặt trong cấu trúc cú pháp RBSP PPS hay không. Ví dụ, `Qpc_data_default_flag` là 0 có thể biểu diễn rằng các tham số `Qpc_data()` là không có mặt trong cấu trúc cú pháp RBSP PPS và bảng mặc định được sử dụng để giúp xác định việc lượng tử hóa sắc độ. Trong trường hợp này, bảng mặc định có thể là giống như Bảng 7. Thêm nữa, ví dụ, `Qpc_data_default_flag` là 1 có thể biểu diễn là các tham số `Qpc_data()` có thể là có mặt trong cấu trúc cú pháp RBSP PPS.

Thêm nữa, dữ liệu được xác định bởi người dùng được báo hiệu trong PPS theo phương án này có thể là giống như bảng sau đây.

Bảng 12

pic_parameter_set_rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
...	
if(Qpc_data_default_flag)	
Qpc_data()	
...	
rbsp_trailing_bits()	
}	

Trong khi đó, ví dụ, `Qpc_data()` có thể bao gồm thông tin cần cho việc dẫn xuất lượng tử hóa sắc độ khi `ChromaArrayType` là 1.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất cấu trúc linh hoạt cho việc dẫn xuất tham số lượng tử hóa (QP) sắc độ và việc dẫn xuất QP sắc độ kết hợp. Phương án này đề xuất sơ đồ để báo hiệu cờ ban đầu biểu diễn liệu chế độ được xác định bởi người dùng trong đó các

tham số biểu diễn hàm được sử dụng để dẫn xuất tham số lượng tử hóa (QP) sắc độ trong SPS và/hoặc PPS có thể được sử dụng là có mặt hay không.

Ví dụ, thông tin cờ được báo hiệu trong cú pháp cấp cao được đề xuất trong phương án này có thể là giống như bảng mà sẽ được mô tả sau.

Bảng 13

high_level_syntax_parameter_set {	Descriptor
...	
Qpc_data_present_flag	u(1)
...	
}	

Ví dụ, Qpc_data_present_flag có thể biểu diễn liệu các tham số để dẫn xuất tham số lượng tử hóa sắc độ có mặt trong cấu trúc cú pháp RBSP của cú pháp cấp cao hay không. Ví dụ, Qpc_data_present_flag có giá trị là 0 có thể biểu diễn là các tham số lượng tử hóa sắc độ là không có mặt trong cấu trúc cú pháp RBSP của cấu trúc cú pháp cấp cao. Thêm nữa, ví dụ, Qpc_data_present_flag có giá trị là 1 có thể biểu diễn rằng các tham số lượng tử hóa sắc độ là có mặt trong cấu trúc cú pháp RBSP của cú pháp cấp cao.

Thay vào đó, phần tử cú pháp Qpc_data_present_flag có thể được sử dụng để chỉ báo sơ đồ sử dụng việc dẫn xuất lượng tử hóa sắc độ trong luồng bit. Ví dụ, phần tử cú pháp Qpc_data_present_flag có thể biểu diễn công cụ được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hoặc sử dụng chế độ được xác định bởi người dùng như sau.

Ví dụ, Qpc_data_present_flag có thể biểu diễn liệu việc lượng tử hóa sắc độ được xác định bởi người dùng có được sử dụng trong luồng bit hay không. Ví dụ, Qpc_data_present_flag có giá trị là 0 có thể biểu diễn liệu việc lượng tử hóa sắc độ được xác định bởi người dùng có không được sử dụng trong luồng bit hay không. Thêm nữa, ví dụ, Qpc_data_present_flag có giá trị là 1 có thể biểu diễn việc lượng tử hóa sắc độ được xác định bởi người dùng được sử dụng một mình hay cùng với cờ khác.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất phương án trong đó các tham số lượng tử hóa (QP) sắc độ, tức là, Qp_{Cb} , Qp_{Cr} và Qp_{CbCr} có thể được dẫn xuất sử dụng thông tin được xác định bởi người dùng được báo hiệu trong một hàm như thế nào. Ví dụ, theo phương án

này, dữ liệu biểu diễn hàm để dẫn xuất tham số lượng tử hóa (QP) sắc độ có thể được báo hiệu, và các tham số lượng tử hóa sắc độ có thể được dẫn xuất dựa trên dữ liệu lượng tử hóa sắc độ. Dữ liệu (hoặc bảng ánh xạ QP được xác định bởi người dùng) cho việc dẫn xuất tham số lượng tử hóa sắc độ có thể được báo hiệu như trong bảng sau đây.

Bảng 14

Qp_C data() {	Descriptor
qPi_min_idx	ue(v)
qPi_delta_max_idx	ue(v)
for (i = qPi_min_idx; i <= qPiMaxIdx; i++)	
Qp_C qPi_val[i]	ue(v)
QpOffset_C	ue(v)
}	

Các ngữ nghĩa của các phần tử cú pháp trong Bảng 14 có thể là giống như bảng sau đây.

Bảng 15

qPi_min_idx specifies the minimum qPi index used in the chroma quantization. qPi_delta_max_idx specifies the delta value between the Qp_{C_min_idx} and the maximum qPi index used for the chroma Qp_C derivation. The value of qPiMaxIdx shall be greater than or equal to qPi_min_idx. The maximum index qPiMaxIdx used in Qp_C derivation is derived as follows: $qPiMaxIdx = qPi_min_idx + qPi_delta_max_idx$ Qp_C qPi_val[i] specifies the Qp_C value for the ith index. QpOffset_C specifies the offset value to be used the derivation of the Qp_C. The variable Qp_CIdx[qPi] for qPi with qPi = 0 ... qPiMaxIdx, is derived as follows: - For qPi < qPi_min_idx, Qp_CIdx[qPi] is set equal to qPi. - For qPi = qPi_min_idx ... qPiMaxIdx, the following applies: $Qp_C_Idx[qPi] = Qp_C_qPi_val[qPi]$ - For qPi > qPiMaxIdx, Qp_CIdx[qPi] = qPi - QpOffset_C The value of Qp_C is derived as Qp_CIdx[qPi].

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa Qpi_min_idx và chỉ số qPi lớn nhất được sử dụng trong khi dẫn xuất sắc độ Qpc . Giá trị của $qPiMaxIdx$ có thể là lớn hoặc bằng qPi_min_idx . Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qpc có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 4

$$qPiMaxIdx = qPi_min_idx + qPi_delta_max_idx$$

Thêm nữa, ví dụ, phần tử cú pháp $Qpc_qPi_val[i]$ có thể biểu diễn giá trị Qpc cho chỉ số thứ i (i-th).

Thêm nữa, ví dụ, phần tử cú pháp $QpOffsetc$ có thể biểu diễn giá trị bù được sử dụng cho việc dẫn xuất Qpc .

Thêm nữa, ví dụ, tham số $QpcIdx[qPi]$ cho qPi có thể được dẫn xuất như sau. Trong trường hợp này, qPi có thể là từ 0 đến $qPiMaxIdx$.

- Khi $qPi < qPi_min_idx$, thì $QpcIdx[qPi]$ có thể được thiết đặt giống với qPi .
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $QpcIdx[qPi]$ có thể được thiết đặt giống với $Qpc_qPi_val[qPi]$.
- Khi $qPi > qPiMaxIdx$, thì $QpcIdx[qPi]$ có thể được thiết đặt như $qPi - QpOffsetc$.

Sau đó, giá trị của Qpc có thể được dẫn xuất như $QpcIdx[qPi]$.

Ví dụ, theo phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được mô tả trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 16

Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg - 1, yQg) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location (xQg - 1, yQg) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (xQg - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= yQg \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp \parallel yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg - 1, yQg).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg, yQg - 1) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location $(xCb + cbWidth / 2, yCb + cbHeight / 2)$.
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as specified in-Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively clause 7.x.x
- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Tham chiếu đến Bảng 16, quy trình dẫn xuất cho tham số lượng tử hóa độ chói và sắc độ có thể được bắt đầu dựa vào thực tế là đầu vào cho quy trình là vị trí độ chói (xCb, yCb) , tham số $cbWidth$ và $cbHeight$ chỉ định chiều rộng và chiều cao của khối lập mã hiệu tại, và tham số $treeType$ chỉ định cây đơn hoặc cây kép. Trong khi đó, như trong các nội dung được mô tả ở trên, tham số lượng tử hóa độ chói và tham số lượng tử hóa sắc độ có thể được biểu diễn như Qp'_Y , Qp'_{Cb} và Qp'_{Cr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số

lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất ví dụ trong đó cờ bên trong như SPS có chế độ được xác định bởi người dùng hoặc chế độ mặc định, thì các phần tử cú pháp vốn có thể được sử dụng để điều khiển việc dẫn xuất tham số lượng tử hóa được sử dụng. Ví dụ về phần tử cú pháp vốn có thể được sử dụng để dẫn xuất tham số lượng tử hóa có thể là giống như trong các bảng sau đây. Trong khi đó, cấu trúc của các phần tử cú pháp không được giới hạn ở cấu trúc được minh họa trong các bảng sau đây, ví dụ.

Bảng 17

	Descriptor
pic_parameter_set_rbsp() {	
pps_pic_parameter_set_id	ue(v)
...	
Qpc_data_default_flag	u(1)
...	

Bảng 18

	Descriptor
slice_header() {	
slice_pic_parameter_set_id	ue(v)
...	
if(!Qpc_data_default_flag)	
slice_QpC_aps_id	u(5)
byte_alignment()	
}	

Bảng 19

adaptation_parameter_set_rbsp() {	Descriptor
adaptation_parameter_set_id	u(5)
aps_params_type	u(3)
if(aps_params_type == ALF_APS)	
alf_data(adaptation_parameter_set_id)	
else if(aps_params_type == LMCS_APS)	
lmcs_data()	
else if(aps_params_type == Qpc_APS) // 2	
Qpc_data()	
aps_extension_flag	u(1)
if(aps_extension_flag)	
while(more_rbsp_data())	
aps_extension_data_flag	u(1)
rbsp_trailing_bits()	
}	

Ví dụ, phần tử cú pháp Qpc_data_default_flag có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng cho việc dẫn xuất tham số lượng tử hóa hay không. Ví dụ, Qpc_data_default_flag có giá trị là 0 có thể biểu diễn là chế độ được xác định bởi người dùng được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Thêm nữa, ví dụ, Qpc_data_default_flag có giá trị là 1 có thể biểu diễn là bảng mặc định được sử dụng cho việc dẫn xuất tham số lượng tử hóa sắc độ. Trong trường hợp này, bảng mặc định có thể là giống như Bảng 7. Thêm nữa, nếu phần tử cú pháp Qpc_data_default_flag là không có mặt, thì phần tử cú pháp Qpc_data_default_flag có thể được suy luận như 1.

Trong khi đó, nếu chế độ được xác định bởi người dùng được sử dụng, thì tiêu đề lát, nhóm ô xếp/tiêu đề, hoặc tiêu đề thích hợp khác tương ứng có thể được sử dụng để báo hiệu ID APS. Ví dụ, như trong Bảng 18, phần tử cú pháp biểu diễn ID APS thông qua tiêu đề lát có thể được báo hiệu.

Ví dụ, phần tử cú pháp slice_Qpc_aps_id có thể biểu diễn adaptation_parameter_set_id của APS Qpc được đề cập đến bởi lát. TemporalId của đơn vị NAL APS Qpc có adaptation_parameter_set_id, như slice_Qpc_aps_id, có thể là nhỏ hơn hoặc bằng TemporalId của đơn vị NAL lát được lập mã. Nếu nhiều APS Qpc có adaptation_parameter_set_id có cùng giá trị được đề cập đến bởi hai hoặc nhiều lát của cùng hình ảnh, thì nhiều APS Qpc có adaptation_parameter_set_id có cùng giá trị có thể có cùng nội dung.

Thêm nữa, cấu trúc APS mà chuyển dữ liệu lượng tử hóa sắc độ, vốn được đề xuất trong phương án này, có thể là giống như Bảng 19.

Ví dụ, phần tử cú pháp `adaptation_parameter_set_id` có thể cung cấp bộ định danh của APS được đề cập đến bởi các phần tử cú pháp khác.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_flag` có thể biểu diễn liệu các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS hay không. Ví dụ, phần tử cú pháp `aps_extension_flag` có giá trị là 1 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS. Phần tử cú pháp `aps_extension_flag` có giá trị là 0 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là không có mặt trong cấu trúc cú pháp RBSP APS.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_data_flag` có thể có giá trị bất kỳ. Sự có mặt (sự có mặt và giá trị) của `aps_extension_data_flag` có thể không ảnh hưởng đến sự thích ứng giải mã cho đối tượng được đặc tả trong một phiên bản của chuẩn này. Ví dụ, thiết bị giải mã mà theo một phiên bản của chuẩn này có thể bỏ qua tất cả các phần tử cú pháp `aps_extension_data_flag`.

Thêm nữa, ví dụ, phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các tham số APS được bao gồm trong APS như được minh họa trong Bảng 10.

`Qpc_data()` được bộc lộ trong Bảng 19 có thể được báo hiệu như bảng sau đây.

Bảng 20

<code>Qp_C_data () {</code>	Descriptor
<code>qPi_min_idx</code>	<code>ue(v)</code>
<code>qPi_delta_max_idx</code>	<code>ue(v)</code>
<code>Qp_C_prec_minus1</code>	<code>ue(v)</code>
<code>Qp_C_init_val</code>	<code>ue(v)</code>
<code>for (i = qPi_min_idx + 1; i <= qPiMaxIdx; i++)</code>	
<code> Qp_C_qPi_delta_val[i]</code>	<code>ue(v)</code>
<code>QpOffset_C</code>	<code>ue(v)</code>
<code>}</code>	

Ví dụ, phần tử cú pháp `qPi_min_idx` có thể biểu diễn chỉ số `qPi` nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị

đenta giữa $Q_{pi_min_idx}$ và chỉ số q_{pi} lớn nhất được sử dụng trong khi dẫn xuất sắc độ Q_{pc} . Giá trị của $q_{piMaxIdx}$ có thể là lớn hoặc bằng $q_{pi_min_idx}$. Ví dụ, chỉ số lớn nhất $q_{piMaxIdx}$ được sử dụng trong khi dẫn xuất Q_{pc} có thể được dẫn xuất giống Phương trình 4.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp $Q_{pc_prec_minus1}$ có thể biểu diễn số lượng bit được sử dụng cho việc biểu diễn cú pháp $l_{mcs_delta_abs_cw}[i]$. Giá trị của $Q_{pc_prec_minus1}$ có thể là trong phạm vi là 0 đến $BitDepth - 2$.

Thêm nữa, ví dụ, phần tử cú pháp $Q_{pc_init_val}$ có thể biểu diễn giá trị Q_{pc} tương ứng với $q_{pi_min_idx}$.

Thêm nữa, ví dụ, phần tử cú pháp $Q_{pc_q_{pi_delta_val}}[i]$ có thể biểu diễn đenta của giá trị Q_{pc} cho chỉ số thứ i .

Thêm nữa, ví dụ, phần tử cú pháp $Q_{pOffsetc}$ có thể biểu diễn giá trị bù được sử dụng cho việc dẫn xuất Q_{pc} .

Ví dụ, tham số $Q_{pcIdx}[q_{pi}]$ cho q_{pi} có thể được dẫn xuất như sau. Trong trường hợp này, q_{pi} có thể là từ 0 đến $q_{piMaxIdx}$.

- Khi $q_{pi} < q_{pi_min_idx}$, thì $Q_{pcIdx}[q_{pi}]$ có thể được thiết đặt giống với q_{pi} .
- Khi $q_{pi} = q_{pi_min_idx} \dots q_{piMaxIdx}$, thì $Q_{pcIdx}[q_{pi}]$ có thể được thiết đặt như $Q_{pc_q_{pi_delta_val}}[q_{pi}] + Q_{pcIdx}[q_{pi}-1]$.
- Khi $q_{pi} > q_{piMaxIdx}$, thì $Q_{pcIdx}[q_{pi}]$ có thể được thiết đặt như $q_{pi} - Q_{pOffsetc}$.

Sau đó, giá trị của Q_{pc} có thể được dẫn xuất như $Q_{pcIdx}[q_{pi}]$.

Như trong phương án được đề cập ở trên, tham số lượng tử hóa sắc độ, tức là, $Q_{p'Cb}$, $Q_{p'Cr}$ và $Q_{p'CbCr}$ có thể được dẫn xuất sử dụng thông tin được xác định bởi người dùng được báo hiệu hoặc bằng mặc định được minh họa trong bảng mặc định, như Bảng 7.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 21

Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location $(xCb + cbWidth / 2, yCb + cbHeight / 2)$.

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as specified in clause 7.x.x
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 21, khi ChromaArrayType là 1 và Qpc_data_default_flag chỉ báo sai (false) (tức là, ví dụ, khi Qpc_data_default_flag là 0), thì các tham số qP_{Cb} ,

qP_{Cr} và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Khi $ChromaArrayType$ là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (true) (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất các phần tử cú pháp mà có thể được sử dụng để điều khiển việc dẫn xuất tham số lượng tử hóa bằng cách chỉ báo chế độ được xác định bởi người dùng hoặc chế độ mặc định thông qua cờ của SPS. Cụ thể, phương án này đề xuất sơ đồ để báo hiệu các phần tử cú pháp của cấu trúc cú pháp sau đây. Trong khi đó, cấu trúc của các phần tử cú pháp là một ví dụ, và không được giới hạn đến cấu trúc được minh họa trong bảng sau đây.

Bảng 22

$Qp_c_data () \{$	Descriptor
qPi_min_idx	ue(v)
$qPi_delta_max_idx$	ue(v)
for ($i = qPi_min_idx$; $i \leq qPiMaxIdx$; $i++$)	
$Qp_c_qPi_delta_val[i]$	ue(v)
$QpOffset_c$	ue(v)
$\}$	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số qPi lớn nhất được sử dụng trong khi dẫn xuất sắc độ Qp_c . Giá trị của $qPiMaxIdx$ có thể là lớn hoặc bằng qPi_min_idx . Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qp_c có thể được dẫn xuất như Phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $Qp_c_qPi_delta_val[i]$ có thể biểu diễn delta của giá trị Qp_c cho chỉ số thứ i .

Thêm nữa, ví dụ, phần tử cú pháp $QpOffset_c$ có thể biểu diễn giá trị bù được sử

dụng trong khi dẫn xuất Qp_c , như các nội dung được đề cập đến ở trên.

Như trong phương án được đề cập ở trên, tham số lượng tử hóa sắc độ, tức là, Qp'_{Cb} , Qp'_{Cr} và Qp'_{CbCr} có thể được dẫn xuất sử dụng thông tin được xác định bởi người dùng được báo hiệu hoặc bằng mặc định, như Bảng 7.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 23

Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp \parallel yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg - 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg - 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg - 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively as described below:

The variable $QpCIdx[i]$ is derived as follows:

- For $i < qPi_min_idx$, $QpCIdx[i]$ is set equal to qPi .

- For $i = qPi_min_idx \dots qPiMaxIdx$, the following applies:

$$QpCIdx[i] = QpC_qPi_delta_val[i] + QpCIdx[i - 1]$$

For $i > qPiMaxIdx$, $QpCIdx[i] = qPi - QpOffset_C$

The value of Qp_C is derived as $QpCIdx[qPi]$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 23, khi ChromaArrayType là 1 và $Qp_c_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Ví dụ, khi ChromaArrayType là 1 và $Qp_c_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất tương tự với giá trị của Qp_C dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} như sau.

Ví dụ, tham số $QpCIdx[i]$ có thể được dẫn xuất như sau.

- Khi $i < qPi_min_idx$, $QpCIdx[qPi]$ có thể được thiết đặt giống với qPi .
- Khi $i = qPi_min_idx \dots qPiMaxIdx$, thì $QpCIdx[i]$ có thể được thiết đặt như $QpC_qPi_delta_val[i] + QpCIdx[i-1]$.
- Khi $i > qPiMaxIdx$, thì $QpCIdx[i]$ có thể được thiết đặt như $qPi - QpOffset_C$.

Sau đó, Qp_C có thể được thiết đặt như $QpCIdx[i]$.

Thêm nữa, tham chiếu đến Bảng 23, khi ChromaArrayType là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất các phần tử cú pháp cho tham số dẫn xuất lượng tử hóa sắc độ (Q_{pc}) trong tập tham số thích ứng (APS). Ví dụ, ID APS có thể được báo hiệu trong tiêu đề lát. Thêm nữa, ví dụ, có thể có cờ bên trong tập tham số hình ảnh (PPS) mà biểu diễn liệu bảng mặc định được sử dụng hay bảng được dẫn xuất từ thông tin được báo hiệu trong APS được sử dụng được đề xuất. Thêm nữa, ví dụ, nếu bảng mặc định là không được sử dụng, sơ đồ điều khiển bổ sung để hỗ trợ truy cập vào APS bao gồm dữ liệu Q_{pc} có thể được thêm vào tiêu đề lát hay không.

Trong khi đó, theo chuẩn video/ảnh hiện có, QP sắc độ có thể được dẫn xuất từ QP độ chói, và có thể được cập nhật bằng độ bù QP được báo hiệu bổ sung. Bảng tham số lượng tử hóa sắc độ Q_{pc} hiện có có thể là bảng mặc định, như Bảng 7.

Phương án này đề xuất bổ sung hàm để báo hiệu tham số lượng tử hóa sắc độ Q_{pc} như hàm của chỉ số q_{Pi} . APS có thể được sử dụng để hợp nhất các sơ đồ báo hiệu gồm các giá trị Q_{pc} .

Ví dụ, APS theo phương án này có thể là giống như bảng sau đây.

Bảng 24

<code>adaptation_parameter_set_rbsp() {</code>	Descriptor
<code>adaptation_parameter_set_id</code>	<code>u(5)</code>
<code>aps_params_type</code>	<code>u(3)</code>
<code>if(aps_params_type == ALF_APS)</code>	
<code> alf_data(adaptation_parameter_set_id)</code>	
<code>else if(aps_params_type == LMCS_APS)</code>	
<code> lmcs_data()</code>	
<code>else if(aps_params_type == Q_{pc}_APS) // 2</code>	
<code> Q_{pc}_data()</code>	
<code>aps_extension_flag</code>	<code>u(1)</code>
<code>if(aps_extension_flag)</code>	
<code> while(more_rbsp_data())</code>	
<code>aps_extension_data_flag</code>	<code>u(1)</code>
<code> rbsp_trailing_bits()</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `adaptation_parameter_set_id` có thể cung cấp bộ định danh của APS được đề cập đến bởi các phần tử cú pháp khác.

Thêm nữa, ví dụ, phần tử cú pháp `aps_params_type` có thể biểu diễn loại của các tham số APS được bao gồm trong APS như được minh họa trong Bảng 10.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_flag` có thể biểu diễn liệu các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS hay không. Ví dụ, phần tử cú pháp `aps_extension_flag` có giá trị là 1 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là có mặt trong cấu trúc cú pháp RBSP APS. Phần tử cú pháp `aps_extension_flag` có giá trị là 0 có thể biểu diễn các phần tử cú pháp `aps_extension_data_flag` là không có mặt trong cấu trúc cú pháp RBSP APS.

Thêm nữa, ví dụ, phần tử cú pháp `aps_extension_data_flag` có thể có giá trị bất kỳ. Sự có mặt (sự có mặt và giá trị) của `aps_extension_data_flag` có thể không ảnh hưởng đến sự thích ứng giải mã cho đối tượng được đặc tả trong một phiên bản của chuẩn này. Ví dụ, thiết bị giải mã mà theo một phiên bản của chuẩn này có thể bỏ qua tất cả các phần tử cú pháp `aps_extension_data_flag`.

`Qp_c_data()` được bộc lộ trong Bảng 24 có thể được báo hiệu trong bảng sau đây.

Bảng 25

<code>Qp_c_data () {</code>	Descriptor
<code>qPi_min_idx</code>	<code>ue(v)</code>
<code>qPi_delta_max_idx</code>	<code>ue(v)</code>
<code>for (i = qPi_min_idx; i <= qPiMaxIdx; i++)</code>	
<code>Qp_c_qPi_delta_val[i]</code>	<code>ue(v)</code>
<code>Qp_cOffset_c_present_flag</code>	<code>u(1)</code>
<code>if(Qp_cOffset_c_present_flag)</code>	
<code>QpOffset_c</code>	<code>ue(v)</code>
<code>}</code>	

Ví dụ, phần tử cú pháp `qPi_min_idx` có thể biểu diễn chỉ số `qPi` nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của `qPi_min_idx` có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `qPi_min_idx` và chỉ số `qPi` lớn nhất được sử dụng trong khi dẫn xuất sắc độ `Qp_c`. Giá trị của `qPiMaxIdx` có thể là lớn hoặc bằng `qPi_min_idx`. Thêm nữa, ví dụ, giá trị của `qPi_delta_max_idx` có thể là trong phạm vi là 0 đến 63. Ví dụ, chỉ số lớn nhất `qPiMaxIdx` được sử dụng trong khi dẫn xuất `Qp_c` có thể được dẫn xuất giống Phương

trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $Q_{pc_qPi_delta_val}[i]$ có thể biểu diễn mức chênh lệch giữa các giá trị Q_{pc} cho chỉ số thứ i . Mức chênh lệch có thể cũng được gọi là δ_{enta} .

Thêm nữa, ví dụ, phần tử cú pháp $Q_{pcOffsetc_present_flag}$ có thể biểu diễn liệu $Q_{pcOffsetc}$ là có mặt trong luồng bit hay không. Ví dụ, $Q_{pcOffsetc_present_flag}$ có giá trị là 1 có thể biểu diễn là $Q_{pcOffsetc}$ là có mặt trong luồng bit. Thêm nữa, ví dụ, $Q_{pcOffsetc_present_flag}$ có giá trị là 0 có thể biểu diễn là $Q_{pcOffsetc}$ là không có mặt trong luồng bit. Khi $Q_{pcOffsetc_present_flag}$ là không có mặt, thì $Q_{pcOffsetc_present_flag}$ có thể được suy luận như 0.

Thêm nữa, ví dụ, phần tử cú pháp $Q_{pcOffsetc}$ có thể biểu diễn giá trị bù được sử dụng trong khi dẫn xuất Q_{pc} .

Ví dụ, tham số $Q_{pcIdx}[qPi]$ cho qPi có thể được dẫn xuất như sau. Trong trường hợp này, qPi có thể là 0 đến 63.

- Khi $qPi < qPi_min_idx$, thì $Q_{pcIdx}[qPi]$ có thể được thiết đặt giống với qPi .
- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $Q_{pcIdx}[qPi]$ có thể được thiết đặt như $Q_{pc_qPi_delta_val}[qPi] + Q_{pcIdx}[qPi-1]$.
- Nếu $qPi > qPiMaxIdx$, $Q_{pcOffsetc_present_flag}$ là 1, thì $Q_{pcIdx}[qPi]$ có thể được thiết đặt như $qPi - Q_{pcOffsetc}$. Nếu $Q_{pcOffsetc_present_flag}$ không là 1, tức là, nếu $Q_{pcOffsetc_present_flag}$ là 0, thì $Q_{pcIdx}[qPi]$ có thể được thiết đặt như $qPi - (qPiMaxIdx - Q_{pcIdx}[qPiMaxIdx])$.

Sau đó, giá trị của Q_{pc} có thể được dẫn xuất như $Q_{pcIdx}[qPi]$.

Thêm nữa, phương án này đề xuất cờ được báo hiệu như PPS như trong bảng sau đây.

Bảng 26

pic parameter set rbsp() {	Descriptor
pps_pic_parameter_set_id	ue(v)
...	
Qpc_data_default_flag	u(1)
...	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng cho việc dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là chế độ được xác định bởi người dùng được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn là bảng mặc định được đề cập đến ở trên được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như Bảng 7. Nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Thêm nữa, phương án này đề xuất phần tử cú pháp được báo hiệu như tiêu đề lát như trong bảng sau đây.

Bảng 27

<code>slice_header() {</code>	Descriptor
<code>slice_pic_parameter_set_id</code>	<code>ue(v)</code>
...	
<code>if(!Qpc_data_default_flag)</code>	
<code>slice_Qpc_aps_id</code>	<code>u(5)</code>
...	
<code>byte_alignment()</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `slice_Qpc_aps_id` có thể biểu diễn `adaptation_parameter_set_id` của APS `Qpc` được đề cập đến bởi lát. `TemporalId` của đơn vị NAL APS `Qpc` có `adaptation_parameter_set_id`, như `slice_Qpc_aps_id`, có thể là nhỏ hơn hoặc giống như `TemporalId` của đơn vị NAL lát được lập mã. Nếu nhiều APS `Qpc` có `adaptation_parameter_set_id` có cùng giá trị được đề cập đến bởi hai hoặc nhiều lát của cùng hình ảnh, thì nhiều APS `Qpc` có `adaptation_parameter_set_id` có cùng giá trị có thể có cùng nội dung.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 28

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 28, khi ChromaArrayType là 1 và $Qpc_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qpc_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và

qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi $ChromaArrayType$ là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, trong phương án này đề xuất việc dẫn xuất được xác định bởi người dùng đối với việc lượng tử hóa sắc độ được báo hiệu trong SPS như sau. Ví dụ, phương án này đề xuất việc lượng tử hóa sắc độ được xác định bởi người dùng (Qp_c). Ví dụ, cờ SPS có thể biểu diễn liệu bảng mặc định được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay các nội dung của bảng cho việc dẫn xuất lượng tử hóa sắc độ được dẫn xuất trong thông tin được báo hiệu trong SPS.

Ví dụ, phương án này đề xuất sơ đồ để thực hiện việc lượng tử hóa như hàm của chỉ số qPi bằng cách sử dụng các phần tử cú pháp được minh họa trong bảng sau đây.

Bảng 29

$Qp_c_data()$ {	Descriptor
qPi_min_idx	$ue(v)$
$qPi_delta_max_idx$	$ue(v)$
for ($i = qPi_min_idx$; $i \leq qPiMaxIdx$; $i++$)	
$Qp_c_qPi_delta_val[i]$	$ue(v)$
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số qPi lớn nhất được sử dụng trong khi dẫn xuất sắc độ Qp_c . Giá trị của $qPiMaxIdx$ có thể là lớn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi là 0 đến 63. Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qp_c có thể được dẫn xuất giống Phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $Qp_c_qPi_delta_val[i]$ có thể biểu diễn delta

của giá trị Q_{pC} cho chỉ số thứ i .

Ví dụ, tham số $Q_{pCIdx}[q_{Pi}]$ có thể được dẫn xuất như sau.

- Khi $q_{Pi} < q_{Pi_min_idx}$, thì $Q_{pCIdx}[q_{Pi}]$ có thể được thiết đặt giống với q_{Pi} .
- Khi $q_{Pi} = q_{Pi_min_idx} \dots q_{PiMaxIdx}$, thì $Q_{pCIdx}[q_{Pi}]$ có thể được thiết đặt như $Q_{pC_q_{Pi_delta_val}}[q_{Pi}] + Q_{pCIdx}[q_{Pi}-1]$.
- Khi $q_{Pi} > q_{PiMaxIdx}$, $Q_{pCIdx}[q_{Pi}]$ có thể được thiết đặt như $q_{Pi} - (q_{PiMaxIdx} - Q_{pCIdx}[q_{PiMaxIdx}])$.

Theo đó, Q_{pC} có thể được thiết đặt như $Q_{pCIdx}[q_{Pi}]$.

Thêm nữa, cờ của SPS biểu diễn liệu bảng mặc định được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay liệu thông tin được báo hiệu được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ, vốn được đề xuất trong phương án này, có thể là giống như bảng sau đây.

Bảng 30

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code> Qpc_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là chế độ được xác định bởi người dùng được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn là bảng mặc định được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như Bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, theo phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 31

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 31, khi ChromaArrayType là 1 và $Qpc_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qpc_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và

qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi $ChromaArrayType$ là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất việc bổ sung hàm để báo hiệu tham số lượng tử hóa sắc độ Qp_c như hàm của chỉ số qPi . Ví dụ, có thể có sơ đồ để báo hiệu các phần tử cú pháp cho bảng được xác định bởi người dùng cho việc dẫn xuất tham số lượng tử hóa trong PPS được đề xuất. Theo đó, sự linh hoạt liên quan đến việc thay đổi bảng được xác định bởi người dùng và bảng mặc định trong mỗi hình ảnh mà tham chiếu đến PPS có thể được cung cấp.

Các phần tử cú pháp cho bảng được xác định bởi người dùng được báo hiệu trong PPS được đề xuất trong phương án này có thể là giống như bảng sau đây.

Bảng 32

$Qp_c_data()$ {	Descriptor
qPi_min_idx	ue(v)
$qPi_delta_max_idx$	ue(v)
for ($i = qPi_min_idx$; $i \leq qPiMaxIdx$; $i++$)	
$Qp_c_qPi_delta_val[i]$	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số qPi lớn nhất được sử dụng trong khi dẫn xuất sắc độ Qp_c . Giá trị của $qPiMaxIdx$ có thể là lớn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi là 0 đến 63. Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qp_c có thể được dẫn xuất giống Phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $Qp_c_qPi_delta_val[i]$ có thể biểu diễn delta

của giá trị Q_{pC} cho chỉ số thứ i .

Ví dụ, tham số $Q_{pCIdx}[q_{Pi}]$ có thể được dẫn xuất như sau.

- Khi $q_{Pi} < q_{Pi_min_idx}$, thì $Q_{pCIdx}[q_{Pi}]$ có thể được thiết đặt giống với q_{Pi} .
- Khi $q_{Pi} = q_{Pi_min_idx} \dots q_{PiMaxIdx}$, thì $Q_{pCIdx}[q_{Pi}]$ có thể được thiết đặt như $Q_{pC_q_{Pi_delta_val}}[q_{Pi}] + Q_{pCIdx}[q_{Pi}-1]$.
- Khi $q_{Pi} > q_{PiMaxIdx}$, $Q_{pCIdx}[q_{Pi}]$ có thể được thiết đặt như $q_{Pi} - (q_{PiMaxIdx} - Q_{pCIdx}[q_{PiMaxIdx}])$.

Theo đó, Q_{pC} có thể được thiết đặt như $Q_{pCIdx}[q_{Pi}]$.

Thêm nữa, cờ của SPS biểu diễn liệu bảng mặc định được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay liệu thông tin được báo hiệu được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ, vốn được đề xuất trong phương án này, có thể là giống như bảng sau đây.

Bảng 33

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là chế độ được xác định bởi người dùng được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Khi `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn là bảng mặc định được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như Bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 34

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to $availableA$. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - $availableA$ is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to $availableB$. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffsct}_Y) \% (64 + \text{QpBdOffsct}_Y)) - \text{QpBdOffsct}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 34, khi ChromaArrayType là 1 và $Qp_c_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và

qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, $ChromaArrayType$ là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất chế độ chung trong đó tham số lượng tử hóa sắc độ Qp_c được dẫn xuất và được báo hiệu.

Dữ liệu tham số lượng tử hóa sắc độ $Qp_c_data()$ cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 35

$Qp_c_data()$ {	Descriptor
qPi_min_idx	ue(v)
$qPi_delta_max_idx$	ue(v)
for ($i = qPi_min_idx$; $i \leq qPiMaxIdx$; $i++$)	
$Qp_c_qPi_delta_val[i]$	ue(v)
}	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số qPi lớn nhất được sử dụng trong khi dẫn xuất sắc độ Qp_c . Giá trị của $qPiMaxIdx$ có thể là lớn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi là 0 đến 63. Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qp_c có thể được dẫn xuất giống Phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $Qp_c_qPi_delta_val[i]$ có thể biểu diễn delta của giá trị Qp_c cho chỉ số thứ i .

Ví dụ, tham số $Qp_cIdx[qPi]$ có thể được dẫn xuất như sau.

- Khi $qPi < qPi_min_idx$, thì $Qp_cIdx[qPi]$ có thể được thiết đặt giống với qPi .

- Khi $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $QpcIdx[qPi]$ có thể được thiết đặt như $Qpc_qPi_delta_val[qPi] + QpcIdx[qPi-1]$.

- Khi $qPi > qPiMaxIdx$, $QpcIdx[qPi]$ có thể được thiết đặt như $qPi - (qPiMaxIdx - QpcIdx[qPiMaxIdx])$.

Theo đó, Qpc có thể được thiết đặt như $QpcIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay liệu thông tin được báo hiệu được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ. Cờ có thể được báo hiệu thông qua cấu trúc cấp cao, như tập tham số trình tự (SPS), hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cấu trúc cấp cao có thể là giống như bảng sau đây.

Bảng 36

<code>high_level_parameter_set() {</code>	Descriptor
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code> Qpc_data()</code>	
<code>...</code>	

Ví dụ, phân tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là chế độ được xác định bởi người dùng được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Khi `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn là bảng mặc định được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như Bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 37

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg - 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg - 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg - 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).
- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively
- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 37, khi ChromaArrayType là 1 và $Qpc_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qpc_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và

qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi $ChromaArrayType$ là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án này đề xuất sơ đồ để dẫn xuất bảng tham số lượng tử hóa sắc độ Qp_c mà không gồm độ bù. Phương án này có thể được đề xuất để được sử dụng cùng với APS hoặc để được sử dụng độc lập. Ví dụ, cấu trúc cú pháp của APS được tích hợp với dữ liệu lượng tử hóa sắc độ có thể là giống như bảng sau đây.

Bảng 38

$Qp_c_data() \{$	Descriptor
qPi_min_idx	ue(v)
$qPi_delta_max_idx$	ue(v)
$for (i = qPi_min_idx; i \leq qPiMaxIdx; i++)$	
$Qp_c_qPi_delta_val[i]$	ue(v)
$\}$	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa qPi_min_idx và chỉ số qPi lớn nhất được sử dụng trong khi dẫn xuất sắc độ Qp_c . Giá trị của $qPiMaxIdx$ có thể là lớn hoặc bằng qPi_min_idx . Giá trị của $qPi_delta_max_idx$ có thể là trong phạm vi là 0 đến 63. Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng trong khi dẫn xuất Qp_c có thể được dẫn xuất giống Phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp $Qp_c_qPi_delta_val[i]$ có thể biểu diễn mức chênh lệch giữa các giá trị Qp_c cho chỉ số thứ i . Mức chênh lệch có thể cũng được gọi là delta.

Ví dụ, tham số $Qp_cIdx[qPi]$ có thể được dẫn xuất như sau. Trong trường hợp này, qPi có thể là 0 đến 63.

- Khi $qPi < qPi_min_idx$, thì $QpcIdx[qPi]$ có thể được thiết đặt giống với qPi .
- $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $QpcIdx[qPi]$ có thể được thiết đặt như $Qpc_qPi_delta_val[qPi] + QpcIdx[qPi-1]$.
- Khi $qPi > qPiMaxIdx$, $QpcIdx[qPi]$ có thể được thiết đặt như $qPi - (qPiMaxIdx - QpcIdx[qPiMaxIdx])$.

Theo đó, Qpc có thể được thiết đặt như $QpcIdx[qPi]$.

Thêm nữa, sáng chế này đề xuất phương án khác trong đó thông tin cho tham số lượng tử hóa được báo hiệu.

Ví dụ, phương án đề xuất, như một ví dụ, sơ đồ trong đó delta (hoặc mức chênh lệch) giữa các giá trị Qpc liên tục được giới hạn là 1.

Ví dụ, phương án này đề xuất sơ đồ trong đó việc lượng tử hóa sắc độ được xác định bởi người dùng (Qpc) được bao gồm thêm trong chuẩn ảnh/video hiện có. Ví dụ, cờ của tập tham số trình tự (SPS) được đề xuất trong phương án này có thể biểu diễn liệu bảng mặc định hiện có được sử dụng cho việc dẫn xuất tham số lượng tử hóa sắc độ hay liệu các nội dung của bảng được dẫn xuất dựa trên thông tin được báo hiệu trong SPS. Sơ đồ thích hợp cho ảnh được lập mã có thể được chọn bằng cách điều tiết việc lượng tử hóa sắc độ theo phương án này, và hiệu quả lập mã có thể được nâng cao.

Ví dụ, phương án này đề xuất việc bổ sung hàm để báo hiệu việc lượng tử hóa sắc độ Qpc như hàm của chỉ số qPi bằng cách sử dụng các phân tử cú pháp như trong bảng sau đây.

Bảng 39

$Qpc_data()$ {	Descriptor
qPi_min_idx	ue(v)
$qPi_delta_max_idx$	ue(v)
for ($i = qPi_min_idx$; $i \leq qPiMaxIdx$; $i++$)	
$Qpc_qPi_flag[i]$	u(1)
}	

Ví dụ, phân tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng trong việc lượng tử hóa sắc độ. Giá trị của qPi_min_idx có thể là trong phạm vi từ 1 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `Qpi_min_idx` và chỉ số `qPi` lớn nhất được sử dụng trong khi dẫn xuất sắc độ `Qpc`. Giá trị của `qPiMaxIdx` có thể là lớn hoặc bằng `qPi_min_idx`. Giá trị của `qPi_delta_max_idx` có thể là trong phạm vi là 1 đến 63. Ví dụ, chỉ số lớn nhất `qPiMaxIdx` được sử dụng trong khi dẫn xuất `Qpc` có thể được dẫn xuất giống Phương trình 4.

Thêm nữa, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `Qpc` có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp `QpC_qPi_flag[i]` có thể biểu diễn liệu giá trị `Qpc` thứ *i* đã tăng thêm 1 hay không so với giá trị `Qpc` thứ (*i*-1). Ví dụ, `QpC_qPi_flag[i]` có giá trị là 1 có thể biểu diễn giá trị `Qpc` được tăng thêm 1. `QpC_qPi_flag[i]` có giá trị là 0 có thể biểu diễn rằng giá trị `Qpc` không được tăng thêm.

Ví dụ, tham số `QpcIdx[qPi]` có thể được dẫn xuất như sau. Trong trường hợp này, `qPi` có thể là 0 đến 63.

- Khi `qPi < qPi_min_idx`, thì `QpcIdx[qPi]` có thể được thiết đặt giống với `qPi`.
- Khi `qPi = qPi_min_idx...qPiMaxIdx`, thì `QpcIdx[qPi]` có thể được thiết đặt như `Qpc_qPi_flag[qPi] + QpcIdx[qPi-1]`.
- Khi `qPi > qPiMaxIdx`, `QpcIdx[qPi]` có thể được thiết đặt như `qPi - (qPiMaxIdx - QpcIdx[qPiMaxIdx])`.

Theo đó, `Qpc` có thể được thiết đặt như `QpcIdx[qPi]`.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ hay liệu thông tin được báo hiệu được sử dụng cho việc dẫn xuất lượng tử hóa sắc độ. Cờ có thể được báo hiệu thông qua cấu trúc cấp cao, như tập tham số trình tự (SPS), hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cấu trúc cấp cao có thể là giống như bảng sau đây.

Bảng 40

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là chế độ được xác định bởi người dùng được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn là dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Khi `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn là bảng mặc định được sử dụng cho việc dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như Bảng 7. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, trong phương án này, nếu quy trình để dẫn xuất tham số lượng tử hóa được viết trong định dạng chuẩn, thì quy trình có thể được biểu diễn như trong bảng sau đây.

Bảng 41

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (`SINGLE_TREE`) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (`DUAL_TREE_LUMA`) or chroma components (`DUAL_TREE_CHROMA`) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to `SINGLE_TREE` or `DUAL_TREE_LUMA`, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to $availableA$. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - $availableA$ is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (CtbLog2SizeY - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to $availableB$. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 41, khi ChromaArrayType là 1 và $Qpc_data_default_flag$ chỉ báo sai (tức là, ví dụ, khi $Qpc_data_default_flag$ là 0), thì các tham số qP_{Cb} , qP_{Cr} và

qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, khi $ChromaArrayType$ là 1 và $Qp_c_data_default_flag$ chỉ báo đúng (tức là, ví dụ, khi $Qp_c_data_default_flag$ là 1), thì các tham số qP_{Cb} , qP_{Cr} và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên các chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa.

Ví dụ, phương án này đề xuất ví dụ về cấu trúc báo hiệu dữ liệu cho việc dẫn xuất QP sắc độ. Cụ thể, phương án này đề xuất sơ đồ để bổ sung $chroma_qp_mapping_flag$ mà là phần tử cú pháp mới trong SPS. Ví dụ, nếu giá trị $chroma_qp_mapping_flag$ là 0, thì bảng ánh xạ QP sắc độ mặc định có thể được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Thêm nữa, ví dụ, nếu giá trị $chroma_qp_mapping_flag$ là 1, thì các phần tử cú pháp được sử dụng để dẫn xuất bảng ánh xạ QP sắc độ có thể được báo hiệu như trong bảng sau đây.

Bảng 42

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qp_c_data_default_flag)</code>	
<code>qPi_delta_max_idx_minus1</code>	<code>ue(v)</code>
<code>qPi_min_idx_minus1</code>	<code>ue(v)</code>
<code>for (i = 1; i <= qPi_delta_max_idx_minus1 + 1; i++) {</code>	
<code>Qp_c_qPi_flag[i]</code>	<code>ue(v)</code>
<code>...</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp $Qp_c_data_default_flag$ có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, $Qp_c_data_default_flag$ có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, $Qp_c_data_default_flag$ có giá trị là 0 có thể biểu diễn rằng bảng ánh xạ QP sắc độ được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 42 như được mô tả nêu trên được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Nếu

Qpc_data_default_flag là 0, thì dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 42 như được mô tả ở trên có thể được báo hiệu. Thêm nữa, ví dụ, Qpc_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như bảng mặc định trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu Qpc_data_default_flag là không có mặt, thì Qpc_data_default_flag có thể được suy luận như 1.

Thêm nữa, ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp qPi_delta_max_idx_minus1 có thể biểu diễn số lượng điểm mà có hàm ánh xạ không được tăng.

Thêm nữa, ví dụ, phần tử cú pháp qPi_min_idx_minus1 có thể biểu diễn phần tử thứ nhất của tập điểm mà có hàm ánh xạ không được tăng.

Thêm nữa, ví dụ, phần tử cú pháp Qpc_qPi_flag[i] có thể biểu diễn giá trị delta giữa phần tử thứ i và phần tử thứ (i-1) của tập điểm mà có hàm ánh xạ không được tăng.

Bảng ánh xạ QP sắc độ có thể được dẫn xuất như sau dựa trên dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 42.

Ví dụ, biến cQpFlatSize có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 5

$$cQpFlatSize = qPi_delta_max_idx_minus1 + 1$$

Thêm nữa, ví dụ, biến cQpFlat[] có thể được dẫn xuất như trong bảng sau đây.

Bảng 43

The variable cQpFlat[] is derived as follows:

```

cQpFlat[ 0 ] = qPi_min_idx_minus1 + 1;
for( i = 1; i < cQpFlatSize; i++ ) {
    cQpFlat[ i ] = Qpc_qPi_flag[ i ] + cQpFlat[ i - 1 ]
}

```

Sau đó, dựa trên biến cQpFlatSize và biến cQpFlat[], bảng ánh xạ QP sắc độ có thể được dẫn xuất như trong bảng sau đây.

Bảng 44

The chroma QP mapping table `cqpMappingTable[]` is derived as follows:

```

cqpMappingTable [ 0 ] = 0;
for( i = 1; i <= maxQP; i++ ) {
    incStep = 1
    for ( j = 0; j < cQpFlatSize; j++){
        if ( i == cQpFlat[ j ] )
            incStep = 0
    }
    cqpMappingTable [ i ] = cqpMappingTable [ i - 1 ] + incStep;

```

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa.

Ví dụ, phương án này đề xuất sơ đồ để bổ sung `chroma_qp_mapping_flag` mà là phần tử cú pháp mới trong SPS. Ví dụ, nếu giá trị `chroma_qp_mapping_flag` là 0, thì bảng ánh xạ QP sắc độ mặc định có thể được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Thêm nữa, ví dụ, nếu giá trị `chroma_qp_mapping_flag` là 1, thì các phần tử cú pháp được sử dụng để dẫn xuất bảng ánh xạ QP sắc độ có thể được báo hiệu như trong bảng sau đây.

Bảng 45

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>qPi_delta_max_idx_minus1</code>	<code>ue(v)</code>
<code>qPi_min_idx_minus1</code>	<code>ue(v)</code>
<code>for (i = 1; i <= qPi_delta_max_idx_minus1 + 1; i++) {</code>	
<code>Qpc_qPi_idx_minus1[i]</code>	<code>ue(v)</code>
<code>...</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng ánh xạ QP sắc độ được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 42 như được mô tả nêu trên được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Nếu

Qpc_data_default_flag là 0, thì dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 42 như được mô tả ở trên có thể được báo hiệu. Thêm nữa, ví dụ, Qpc_data_default_flag có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu Qpc_data_default_flag là không có mặt, thì Qpc_data_default_flag có thể được suy luận như 1.

Thêm nữa, ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp qPi_delta_max_idx_minus1 có thể biểu diễn số lượng điểm mà có hàm ánh xạ không được tăng.

Thêm nữa, ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp qPi_min_idx_minus1 có thể biểu diễn phần tử thứ nhất của tập điểm mà có bảng ánh xạ không được tăng.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp Qpc_qPi_idx_minus1[i] có thể biểu diễn giá trị delta giữa phần tử thứ i và phần tử thứ (i-1) của tập điểm mà có hàm ánh xạ không được tăng.

Bảng ánh xạ QP sắc độ có thể được dẫn xuất như sau dựa trên dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 45.

Ví dụ, biến cQpFlatSize có thể được dẫn xuất như trong Phương trình 5 như được mô tả nêu trên.

Thêm nữa, ví dụ, biến cQpFlat[] có thể được dẫn xuất như trong bảng sau đây.

Bảng 46

The variable cQpFlat[] is derived as follows:

```
cQpFlat[ 0 ] = qPi_min_idx_minus1 + 1;
for( i = 1; i < cQpFlatSize; i++ ) {
    cQpFlat[ i ] = Qpc_qPi_idx_minus1[ i ] + 1 + cQpFlat[ i - 1 ]
}
```

Sau đó, dựa trên biến cQpFlatSize và biến cQpFlat[], thì bảng ánh xạ QP sắc độ có thể được dẫn xuất. Ví dụ, bảng ánh xạ QP sắc độ có thể được dẫn xuất như trong Bảng 44 như được mô tả nêu trên.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham

số lượng tử hóa.

Ví dụ, phương án này đề xuất sơ đồ để báo hiệu các bảng riêng lẻ cho các thành phần sắc độ tương ứng. Tức là, ví dụ, phương án này đề xuất sơ đồ để báo hiệu các phần tử cú pháp được sử dụng để dẫn xuất các bảng ánh xạ QP sắc độ cho các thành phần sắc độ tương ứng.

Ví dụ, các bảng ánh xạ QP sắc độ cho các thành phần sắc độ tương ứng có thể được dẫn xuất, và các phần tử cú pháp cho các thành phần sắc độ tương ứng có thể được báo hiệu như trong bảng sau đây.

Bảng 47

Qp _C data() {	Descriptor
qp_luma_to_chroma_joint_map_flag	u(1)
for(i = 0; i < (qp_luma_to_chroma_joint_map_flag ? 1 : 3); i++) {	
qPi_min_idx_minus1[i]	uc(v)
qPi_delta_max_idx_minus1[i]	ue(v)
for (j = qPi_min_idx_minus1 + 1 ; j <= qPiMaxIdx; i++)	
Qp_C qPi_flag[i][j]	u(1)
}	
}	

Ví dụ, phần tử cú pháp qp_luma_to_chroma_joint_map_flag có thể biểu diễn liệu bảng ánh xạ tham số lượng tử hóa độ chói-sắc độ có được sử dụng cho các thành phần sắc độ Cb, Cr, và CbCr hay không. Tức là, ví dụ, phần tử cú pháp qp_luma_to_chroma_joint_map_flag có thể biểu diễn liệu một bảng ánh xạ tham số lượng tử hóa độ chói-sắc độ có được áp dụng cho phần dư Cb, phần dư Cr, và phần dư CbCr hay không. Ví dụ, nếu giá trị của qp_luma_to_chroma_joint_map_flag là 1, thì bảng ánh xạ tham số lượng tử hóa độ chói-sắc độ chung có thể được sử dụng cho các thành phần sắc độ Cb, Cr, và CbCr, và nếu giá trị của qp_luma_to_chroma_joint_map_flag là 0, thì bảng ánh xạ tham số lượng tử hóa độ chói-sắc độ riêng lẻ có thể được sử dụng cho mỗi thành phần sắc độ trong số các thành phần sắc độ Cb, Cr, và CbCr.

Thêm nữa, ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp qPi_min_idx_minus1 có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx_minus1 có thể là trong phạm vi từ 1 đến 63.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp $qPi_delta_max_idx_minus1$ có thể biểu diễn giá trị delta giữa Qpi_min_idx và chỉ số qPi lớn nhất được sử dụng cho việc dẫn xuất sắc độ Qpc . Giá trị $qPiMaxIdx$ có thể là bằng hoặc lớn hơn giá trị qPi_min_idx . Ví dụ, giá trị $qPi_delta_max_idx_minus1$ có thể là trong phạm vi từ 1 đến 63. Chỉ số lớn nhất $qPiMaxIdx$ được sử dụng cho việc dẫn xuất Qpc có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 6

```
for( i = 0; i < 3; i++) {
     $qPiMaxIdx[i] = qPi\_min\_idx\_minus1[i] + 1 + qPi\_delta\_max\_idx\_minus1[i] + 1$ 
```

Thêm nữa, ví dụ, phần tử cú pháp $Qpc_qPi_flag[j]$ có thể biểu diễn liệu giá trị Qpc có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $Qpc_qPi_flag[i][j]$ có thể biểu diễn liệu giá trị Qpc thứ j của thành phần sắc độ thứ i có được tăng thêm 1 như được so sánh với giá trị Qpc thứ $(j-1)$ hay không. Ví dụ, $Qpc_qPi_flag[j]$ bằng 1 có thể biểu diễn rằng giá trị Qpc được tăng thêm 1, và $Qpc_qPi_flag[j]$ bằng 0 có thể biểu diễn rằng giá trị Qpc không được tăng.

Ví dụ, biến $QpcIdx[i][qPi]$ có thể được dẫn xuất như sau. Ở đây, qPi có thể là $maxQp$ trong trường hợp giá trị là 0.

- Trong trường hợp $qPi < qPi_min_idx_minus1 + 1$, thì $QpcIdx[qPi]$ có thể được tạo cấu hình để là giống như qPi .

- Trong trường hợp $qPi = qPi_min_idx_minus1 + 1 \dots qPiMaxIdx$, thì $QpcIdx[qPi]$ có thể được tạo cấu hình như $Qpc_qPi_flag[qPi] + QpcIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpcIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpcIdx[qPiMaxIdx])$.

Sau đó, giá trị Qpc có thể được dẫn xuất như $QpcIdx[i][qPi]$.

Trong khi đó, theo phương án này, cờ biểu diễn liệu các phần tử cú pháp được sử dụng để dẫn xuất bảng ánh xạ QP sắc độ trong SPS được báo hiệu hay bảng mặc định được sử dụng có thể được báo hiệu. Ví dụ, cờ có thể được báo hiệu như trong bảng sau đây.

Bảng 48

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
<code>...</code>	

Phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng ánh xạ QP sắc độ được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 47 như được mô tả nêu trên được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ được minh họa trong Bảng 47 như được mô tả ở trên có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là giống như bảng mặc định trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 49

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.X.X based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
	Pi	9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 49 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (FALSE) (ví dụ, nếu Qpc_data_default_flag là

0), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu $ChromaArrayType$ là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (TRUE) (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu mức chênh lệch lớn nhất giữa điểm bắt đầu và điểm kết thúc như delta cho QP lớn nhất. Tức là, ví dụ, theo phương án này, phần tử cú pháp biểu diễn giá trị delta giữa $maxQp$ được sử dụng để dẫn xuất sắc độ Qpc và chỉ số qPi lớn nhất có thể được báo hiệu.

Dữ liệu tham số lượng tử hóa sắc độ $Qpc_data()$ cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 50

$Qpc_data() \{$	Descriptor
qPi_min_idx	ue(v)
$qPi_delta_max_idx$	ue(v)
for ($i = qPi_min_idx; i \leq qPiMaxIdx; i++$)	
$Qpc_qPi_flag[i]$	u(1)
$\}$	

Ví dụ, phần tử cú pháp qPi_min_idx có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 0 đến 63.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa $maxQp$ và chỉ số qPi lớn nhất được sử dụng để dẫn xuất sắc độ Qpc . Giá trị $qPiMaxIdx$ có thể là bằng hoặc lớn hơn giá trị qPi_min_idx . Giá trị $qPi_delta_max_idx$ có thể là trong phạm vi từ 1 đến 63. Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng để dẫn xuất Qpc có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 7

$$qPiMaxIdx = maxQp - qPi_delta_max_idx$$

Thêm nữa, ví dụ, phần tử cú pháp $Qpc_qPi_flag[i]$ có thể biểu diễn liệu giá trị

Q_{pc} có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $Q_{pc_qPi_flag}[i]$ có thể biểu diễn liệu giá trị Q_{pc} thứ i có được tăng thêm 1 như được so sánh với giá trị Q_{pc} thứ $(i-1)$ hay không. Ví dụ, $Q_{pc_qPi_flag}[i]$ có giá trị là 1 có thể biểu diễn rằng giá trị Q_{pc} được tăng thêm 1, và $Q_{pc_qPi_flag}[i]$ có giá trị là 0 có thể biểu diễn rằng giá trị Q_{pc} không được tăng.

Ví dụ, biến $Q_{pcIdx}[qPi]$ có thể được dẫn xuất như sau. Ở đây, qPi có thể là từ 0 đến 63.

- Trong trường hợp $qPi < qPi_min_idx$, thì $Q_{pcIdx}[qPi]$ có thể được tạo cấu hình giống như qPi .

- Trong trường hợp $qPi = qPi_min_idx \dots qPiMaxIdx$, thì $Q_{pcIdx}[qPi]$ có thể được tạo cấu hình như $Q_{pc_qPi_flag}[qPi] + Q_{pcIdx}[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $Q_{pcIdx}[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - Q_{pcIdx}[qPiMaxIdx])$.

Sau đó, Q_{pc} có thể được tạo cấu hình như $Q_{pcIdx}[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hay thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ được sử dụng. Cờ có thể được báo hiệu qua cú pháp cấp cao như tập tham số trình tự (SPS) hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cú pháp cấp cao có thể là như trong bảng sau đây.

Bảng 51

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp $Q_{pc_data_default_flag}$ có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, $Q_{pc_data_default_flag}$ có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ,

`Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 52

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg - 1, yQg) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location (xQg - 1, yQg) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (xQg - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= yQg \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg - 1, yQg).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg, yQg - 1) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).

4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:

- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
- Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($xCb + cbWidth / 2$, $yCb + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 52 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (ví dụ, nếu Qpc_data_default_flag là 0), thì các

biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu $ChromaArrayType$ là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu mức chênh lệch lớn nhất giữa điểm bắt đầu và điểm kết thúc bằng cách báo hiệu điểm kết thúc như delta cho QP lớn nhất hoặc như mức chênh lệch giữa điểm bắt đầu và giá trị thu được bằng cách cộng delta vào điểm bắt đầu.

Dữ liệu tham số lượng tử hóa sắc độ $Qpc_data()$ cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 53

$Qpc_data() \{$	Descriptor
$qPi_min_idx_minus1$	ue(v)
is_delta_maxQp	u(1)
$qPi_delta_max_idx_minus1$	uc(v)
$for (i = qPi_min_idx_minus1 + 1; i \leq qPiMaxIdx; i++)$	
$Qpc_qPi_flag[i]$	u(1)
$\}$	

Ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp $qPi_min_idx_minus1$ có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến $maxQp$.

Thêm nữa, ví dụ, phần tử cú pháp is_delta_maxQp có thể biểu diễn liệu chỉ số lớn nhất $qPiMaxIdx$ có được dẫn xuất từ giá trị $maxQp$ hay không. Ví dụ, is_delta_maxQp có giá trị là 1 có thể biểu diễn rằng $qPiMaxIdx$ được dẫn xuất từ giá trị $maxQp$. Thêm nữa, ví dụ, is_delta_maxQp có giá trị là 0 có thể biểu diễn rằng $qPiMaxIdx$ được dẫn xuất từ phần tử cú pháp $qPi_min_idx_minus1$.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp $qPi_delta_max_idx_minus1$ có thể biểu diễn dữ liệu delta giữa $maxQp$ và chỉ số qPi lớn nhất được sử dụng để dẫn xuất sắc độ Qpc . Giá trị $qPiMaxIdx$ có thể là bằng hoặc lớn hơn qPi_min_idx . Ví dụ, giá trị $qPi_delta_max_idx_minus1$ có thể là trong phạm vi

từ 1 đến 63. Chỉ số lớn nhất $qPiMaxIdx$ được sử dụng cho việc dẫn xuất QpC có thể được dẫn xuất như trong bảng sau đây.

Bảng 54

```
index qPiMaxIdx used in QpC derivation is derived as follows:
if(is_delta_maxQp){
  qPiMaxIdx = maxQp - qPi_delta_max_idx
} else {
  qPiMaxIdx = qPi_min_idx + qPi_delta_max_idx
}
```

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_flag[i]$ có thể biểu diễn liệu giá trị QpC có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $QpC_qPi_flag[i]$ có thể biểu diễn liệu giá trị QpC thứ i có được tăng thêm 1 như được so sánh với giá trị QpC thứ $(i-1)$ hay không. Ví dụ, $QpC_qPi_flag[i]$ có giá trị là 1 có thể biểu diễn rằng giá trị QpC được tăng thêm 1, và $QpC_qPi_flag[i]$ có giá trị là 0 có thể biểu diễn rằng giá trị QpC không được tăng.

Ví dụ, biến $QpCIdx[qPi]$ có thể được dẫn xuất như sau. Ở đây, qPi có thể là từ 0 đến $maxQp$.

- Trong trường hợp là $qPi < qPi_min_idx_minus1+1$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình giống như qPi .

- Trong trường hợp là $qPi = qPi_min_idx_minus1 \dots qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $QpC_qPi_flag[qPi] + QpCIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Sau đó, QpC có thể được tạo cấu hình như $QpCIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hay thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ được sử dụng. Cờ có thể được báo hiệu qua cú pháp cấp cao như tập tham số trình tự (SPS) hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cú pháp cấp cao có thể là như trong bảng sau đây.

Bảng 55

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code> sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code> ...</code>	
<code> Qpc_data_default_flag</code>	<code>u(1)</code>
<code> if(!Qp_C_data_default_flag)</code>	
<code> Qp_C_data()</code>	
<code> ...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ `QpC_data()` có được sử dụng hay không. Nếu `QpC_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `QpC_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu `QpC_data_default_flag` là không có mặt, thì `QpC_data_default_flag` có thể được suy luận như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 56

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PRED} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PRED} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg - 1, yQg) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location (xQg - 1, yQg) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (xQg - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= yQg \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg - 1, yQg).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg, yQg - 1) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2$, $y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qP_{Cb} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qP_{Cr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qP_{CbCr} = \text{Clip3}(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $\text{Min}(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 56 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (ví dụ, nếu Qpc_data_default_flag là 0), thì các

biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu $ChromaArrayType$ là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu mức chênh lệch lớn nhất giữa điểm bắt đầu và điểm kết thúc bằng cách báo hiệu điểm kết thúc như delta cho QP lớn nhất hoặc như mức chênh lệch giữa điểm bắt đầu và giá trị thu được bằng cách cộng delta vào điểm bắt đầu.

Dữ liệu tham số lượng tử hóa sắc độ $Qpc_data()$ cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 57

$Qpc_data() \{$	Descriptor
$qPi_min_idx_minus1$	ue(v)
is_delta_maxQp	u(1)
$qPi_delta_max_idx_minus1$	uc(v)
for ($i = qPi_min_idx_minus1 + 1$; $i \leq qPiMaxIdx$; $i++$)	
$Qpc_qPi_flag[i]$	u(1)
$\}$	

Ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp $qPi_min_idx_minus1$ có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến $maxQp$.

Thêm nữa, ví dụ, phần tử cú pháp is_delta_maxQp có thể biểu diễn liệu chỉ số lớn nhất $qPiMaxIdx$ có được dẫn xuất từ giá trị $maxQp$ hay không. Ví dụ, is_delta_maxQp có giá trị là 1 có thể biểu diễn rằng $qPiMaxIdx$ được dẫn xuất từ giá trị $maxQp$. Thêm nữa, ví dụ, is_delta_maxQp có giá trị là 0 có thể biểu diễn rằng $qPiMaxIdx$ được dẫn xuất từ phần tử cú pháp $qPi_min_idx_minus1$.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp $qPi_delta_max_idx_minus1$ có thể biểu diễn dữ liệu delta giữa $maxQp$ và chỉ số qPi lớn nhất được sử dụng để dẫn xuất sắc độ Qpc . Giá trị $qPiMaxIdx$ có thể là bằng hoặc lớn hơn qPi_min_idx . Ví dụ, giá trị $qPi_delta_max_idx_minus1$ có thể là trong phạm vi

từ 1 đến 63. Chỉ số lớn nhất $qPiMaxIdx$ được sử dụng để dẫn xuất QpC có thể được dẫn xuất như trong Bảng 54 như được mô tả nêu trên.

Thêm nữa, ví dụ, phần tử cú pháp $QpC_qPi_flag[i]$ có thể biểu diễn liệu giá trị QpC có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $QpC_qPi_flag[i]$ có thể biểu diễn liệu giá trị QpC thứ i có được tăng thêm 1 như được so sánh với giá trị QpC thứ $(i-1)$ hay không. Ví dụ, $QpC_qPi_flag[i]$ có giá trị là 1 có thể biểu diễn rằng giá trị QpC được tăng thêm 1, và $QpC_qPi_flag[i]$ có giá trị là 0 có thể biểu diễn rằng giá trị QpC không được tăng.

Ví dụ, biến $QpCIdx[qPi]$ có thể được dẫn xuất như sau. Ở đây, qPi có thể là từ 0 đến $maxQp$.

- Trong trường hợp là $qPi < qPi_min_idx_minus1+1$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình giống như qPi .

- Trong trường hợp là $qPi = qPi_min_idx_minus1 \dots qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $QpC_qPi_flag[qPi] + QpCIdx[qPi-1]$.

- Trong trường hợp $qPi > qPiMaxIdx$, thì $QpCIdx[qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Sau đó, QpC có thể được tạo cấu hình như $QpCIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hay thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ được sử dụng. Cờ có thể được báo hiệu qua cú pháp cấp cao như tập tham số trình tự (SPS) hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cú pháp cấp cao có thể là giống như cờ được báo hiệu trong bảng sau đây.

Bảng 58

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>QpC_data_default_flag</code>	<code>u(1)</code>
<code>if(!QpC_data_default_flag)</code>	
<code>QpC_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp $QpC_data_default_flag$ có thể biểu diễn liệu chế độ được

xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có được sử dụng hay không. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu hay không. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được coi như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 59

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (`SINGLE_TREE`) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (`DUAL_TREE_LUMA`) or chroma components (`DUAL_TREE_CHROMA`) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to `SINGLE_TREE` or `DUAL_TREE_LUMA`, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:
 - If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned}
 x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\
 y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\
 \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY}))
 \end{aligned}$$
 - Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).
3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:
 - If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * QpBdOffset_Y) \% (64 + QpBdOffset_Y)) - Qp \\
 &\text{BdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 59 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (ví dụ, nếu Qpc_data_default_flag là 0), thì các

biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu $ChromaArrayType$ là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa. Phương án này đề xuất sơ đồ để báo hiệu chỉ số cho bảng ánh xạ QP sắc độ bằng cách sử dụng hệ ký hiệu $minus1$ thay vì giá trị thực.

Dữ liệu tham số lượng tử hóa sắc độ $Qpc_data()$ cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 60

$Qpc_data() \{$	Descriptor
$qPi_min_idx_minus1$	$ue(v)$
$qPi_delta_max_idx_minus1$	$ue(v)$
$for (i = qPi_min_idx_minus1; i \leq qPiMaxIdx; i++)$	
$Qpc_qPi_flag[i]$	$u(1)$
$\}$	

Ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp $qPi_min_idx_minus1$ có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến 63.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp $qPi_delta_max_idx_minus1$ có thể biểu diễn dữ liệu delta giữa qPi_min_idx và chỉ số qPi lớn nhất được sử dụng để dẫn xuất sắc độ Qpc . Giá trị $qPiMaxIdx$ có thể là bằng hoặc lớn hơn qPi_min_idx . Giá trị $qPi_delta_max_idx$ có thể là trong phạm vi từ 1 đến 63. Ví dụ, chỉ số lớn nhất $qPiMaxIdx$ được sử dụng để dẫn xuất Qpc có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 8

$$qPiMaxIdx = qPi_min_idx_minus1 + 1 + qPi_delta_max_idx_minus1 + 1$$

Thêm nữa, ví dụ, phần tử cú pháp $Qpc_qPi_flag[i]$ có thể biểu diễn liệu giá trị Qpc có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $Qpc_qPi_flag[i]$ có thể biểu diễn liệu giá trị Qpc thứ i có được tăng thêm 1 như được so sánh với giá trị Qpc

thứ (i-1) hay không. Ví dụ, `QpC_qPi_flag[i]` có giá trị là 1 có thể biểu diễn rằng giá trị `QpC` được tăng thêm 1, và `QpC_qPi_flag[i]` có giá trị là 0 có thể biểu diễn rằng giá trị `QpC` không được tăng.

Ví dụ, biến `QpCIdx[qPi]` có thể được dẫn xuất như sau. Ở đây, `qPi` có thể là từ 0 đến 63.

- Trong trường hợp là $qPi < qPi_min_idx_minus1+1$, thì `QpCIdx[qPi]` có thể được tạo cấu hình giống như `qPi`.

- Trong trường hợp là $qPi = qPi_min_idx_minus1+1 \dots qPiMaxIdx$, thì `QpCIdx[qPi]` có thể được tạo cấu hình như `QpC_qPi_flag[qPi] + QpCIdx[qPi-1]`.

- Trong trường hợp $qPi > qPiMaxIdx$, thì `QpCIdx[qPi]` có thể được tạo cấu hình như $qPi - (qPiMaxIdx - QpCIdx[qPiMaxIdx])$.

Sau đó, `QpC` có thể được tạo cấu hình như `QpCIdx[qPi]`.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hay thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ được sử dụng. Cờ có thể được báo hiệu qua cú pháp cấp cao như tập tham số trình tự (SPS) hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cú pháp cấp cao có thể là giống như cờ được báo hiệu trong bảng sau đây.

Bảng 61

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
...	
<code>QpC_data_default_flag</code>	<code>u(1)</code>
<code>if(!QpC_data_default_flag)</code>	
<code>QpC_data()</code>	
...	

Ví dụ, phần tử cú pháp `QpC_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `QpC_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ `QpC_data()` được sử dụng. Nếu `QpC_data_default_flag` là 0, thì dữ liệu tham số

lượng tử hóa sắc độ $Qpc_data()$ có thể được báo hiệu. Thêm nữa, ví dụ, $Qpc_data_default_flag$ có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu $Qpc_data_default_flag$ là không có mặt, thì $Qpc_data_default_flag$ có thể được coi như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 62

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg - 1, yQg) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location (xQg - 1, yQg) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} xTmp &= (xQg - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= yQg \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg - 1, yQg).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (xCurr, yCurr) set equal to (xCb, yCb) and the neighbouring location (xNbY, yNbY) set equal to (xQg, yQg - 1) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($x_{Cb} + cbWidth / 2, y_{Cb} + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbcr_qp_offset + slice_joint_cbcr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 62 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (ví dụ, nếu Qpc_data_default_flag là 0), thì các

biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu $ChromaArrayType$ là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa. Phương án này đề xuất sơ đồ trong đó bảng lượng tử hóa sắc độ riêng lẻ được sử dụng cho mỗi thành phần sắc độ trong các thành phần sắc độ.

Dữ liệu tham số lượng tử hóa sắc độ cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 63

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>.....</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag){</code>	
<code>sps_separate_qpc_table_flag</code>	<code>u(1)</code>
<code>qPi_cb_min_idx_minus1</code>	<code>u(7)</code>
<code>qPi_cb_delta_max_idx_minus1</code>	<code>u(7)</code>
<code>for(i = qPi_cb_min_idx_minus1+1; i <= qPiMaxIdxCb; i++)</code>	
<code>Qpc_cb_qPi_flag[i]</code>	<code>u(1)</code>
<code>if(sps_separate_qpc_table_flag) {</code>	
<code>qPi_cb_min_idx_minus1</code>	<code>u(7)</code>
<code>qPi_cb_delta_max_idx_minus1</code>	<code>u(7)</code>
<code>for(i = qPi_cr_min_idx_minus1+1; i <= qPiMaxIdxCr; i++)</code>	
<code>Qpc_cr_qPi_flag[i]</code>	<code>u(1)</code>
<code>}</code>	
<code>}</code>	
<code>.....</code>	
<code>}</code>	

Ví dụ, phần tử cú pháp $Qpc_data_default_flag$ có thể biểu diễn liệu bảng tham số lượng tử hóa sắc độ mặc định có được sử dụng hay không. Ví dụ, $Qpc_data_default_flag$ có giá trị là 1 có thể biểu diễn rằng bảng tham số lượng tử hóa sắc độ mặc định được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Bảng mặc định

có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng tham số lượng tử hóa sắc độ mặc định không được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng bảng tham số lượng tử hóa sắc độ được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ được báo hiệu để dẫn xuất tham số lượng tử hóa sắc độ được sử dụng.

Thêm nữa, ví dụ, phần tử cú pháp `sps_separate_qpc_table_flag` có thể biểu diễn liệu hai bảng Qpc riêng lẻ có được sử dụng cho mẫu Cb và mẫu Cr hay không. Tức là, ví dụ, phần tử cú pháp `sps_separate_qpc_table_flag` có thể biểu diễn liệu các bảng ánh xạ tham số lượng tử hóa độ chói-sắc độ riêng lẻ có được sử dụng lần lượt cho phần dư Cb và phần dư Cr hay không. Ví dụ, `sps_separate_qpc_table_flag` có giá trị là 1 có thể biểu diễn rằng các bảng Qpc riêng lẻ lần lượt được sử dụng cho mẫu Cb và mẫu Cr, và `sps_separate_qpc_table_flag` có giá trị là 0 có thể biểu diễn rằng một bảng Qpc được sử dụng cho mẫu Cb và mẫu Cr.

Trong khi đó, ví dụ, biến `Qpcb[i]` có thể biểu diễn bảng Qpc được sử dụng cho mẫu Cb. Thêm nữa, ví dụ, biến `Qpcr[i]` có thể biểu diễn bảng Qpc được sử dụng cho mẫu Cr. Thêm nữa, ví dụ, nếu giá trị của `sps_separate_qpc_table_flag` là 0, thì `Qpcr[i]` có thể là giống như `Qpcb[i]`. Ở đây, *i* có thể là từ 0 đến 69.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 vào phần tử cú pháp `qPi_cb_min_idx_minus1` có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho thành phần sắc độ Cb. Giá trị `qPi_cb_min_idx_minus1` có thể là trong phạm vi từ 1 đến 69.

Thêm nữa, ví dụ, giá trị thu được bằng cách cộng 1 và phần tử cú pháp `qPi_cb_delta_max_idx_minus1` có thể biểu diễn giá trị delta giữa `qPi_cb_min_idx_minus1` và `qPi_cb_delta_max_idx_minus1` lớn nhất được sử dụng để dẫn xuất sắc độ Qpc Cb. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn `qPi_min_idx`. Giá trị `qPi_cb_delta_max_idx_minus1` có thể là trong phạm vi từ 1 đến 69. Ví dụ, chỉ số lớn nhất `qPiMaxIdxcb` được sử dụng để dẫn xuất Qpc cho thành phần Cb có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 9

$$qPiMaxIdxCb = qPi_cb_min_idx_minus1 + 1 + qPi_cb_delta_max_idx_minus1 + 1$$

Thêm nữa, ví dụ, phần tử cú pháp $QpC_cb_qPi_flag[i]$ có thể biểu diễn giá trị delta giữa giá trị QpC thứ i $QpCb[i]$ cho thành phần Cb và giá trị QpC thứ $(i-1)$ $QpCb[i-1]$. Giá trị $QpC_cb_qPi_flag[i]$ có thể là trong phạm vi từ 0 đến 1.

Ví dụ, biến $QpCb[i]$ có thể được dẫn xuất như sau. Ở đây, i có thể là từ 0 đến 69.

- Trong trường hợp là $i = 0 \dots qPiMaxIdxCb$, thì $QpCb[i]$ có thể được tạo cấu hình giống như i .

- Trong trường hợp là $i = qPi_cb_min_idx_minus1 + 1 \dots qPiMaxIdxCb$, thì $QpCb[i]$ có thể được tạo cấu hình như $QpCb[i-1] + QpC_cb_qPi_flag[i]$.

- Trong trường hợp là $i = qPiMaxIdxCb + 1 \dots 69$, thì $QpCb[i]$ có thể được tạo cấu hình như $i - deltaEnd$, và $deltaEnd$ có thể được dẫn xuất như $qPiMaxIdxCb - QpCb[qPiMaxIdxCb]$.

Thêm nữa, ví dụ, $qPi_cr_min_idx_minus1$, $qPiMaxIdxCr$, và $QpC_cr_qPi_flag[i]$, vốn là các phần tử cú pháp cho thành phần Cr, có thể có cùng nghĩa như nghĩa của các phần tử cú pháp cho thành phần Cb.

Thêm nữa, sáng chế này đề xuất phương án khác để báo hiệu thông tin về tham số lượng tử hóa. Ví dụ, phương án này đề xuất sơ đồ để báo hiệu các tham số cho nhiều bảng QP sắc độ. Thêm nữa, phương án này có thể được kết hợp với ít nhất một trong các phương án như được mô tả nêu trên. Tức là, ví dụ, các phương án của sáng chế này có thể được áp dụng chung.

Cụ thể, ví dụ, phương án này đề xuất để bao gồm tham số lượng tử hóa sắc độ được xác định bởi người dùng (QpC) trong tài liệu đặc tả VVC. Ví dụ, theo phương án này, cờ của tập tham số trình tự (sequence parameter set, SPS) có thể biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa sắc độ hay bảng ánh xạ sắc độ QP được dẫn xuất dựa trên thông tin được báo hiệu trong SPS. Thông qua điều này, tham số lượng tử hóa sắc độ được xác định bởi người dùng có thể được sử dụng xét đến các đặc tính của ảnh trong khi lập mã ảnh, và hiệu quả lập mã có thể được nâng cao. Thêm nữa, phương án này có thể cung cấp sự linh hoạt thông qua lựa chọn trong đó một bảng được xác định bởi người dùng được sử dụng cho các thành phần sắc độ và lựa chọn trong đó các bảng được xác định bởi người dùng riêng biệt được sử dụng cho thành phần Cb và thành phần Cr.

Ví dụ, dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` cho tham số lượng tử hóa sắc độ được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 64

<code>Qpc_data() {</code>	Descriptor
<code>qPi_min_idx_minus1</code>	<code>ue(v)</code>
<code>qPi_delta_max_idx</code>	<code>ue(v)</code>
<code>for (i = qPi_min_idx; i <= qPiMaxIdx; i++)</code>	
<code> Qpc_qPi_flag[i]</code>	<code>u(1)</code>
<code>}</code>	

Ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp `qPi_min_idx_minus1` có thể biểu diễn chỉ số `qPi` nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị `qPi_min_idx` có thể là trong phạm vi từ 1 đến 69.

Thêm nữa, ví dụ, phần tử cú pháp `qPi_delta_max_idx` có thể biểu diễn giá trị delta giữa `qPi_min_idx` và chỉ số `qPi` lớn nhất được sử dụng để dẫn xuất sắc độ `Qpc`. Giá trị `qPiMaxIdx` có thể là bằng hoặc lớn hơn `qPi_min_idx`. Giá trị `qPi_delta_max_idx` có thể là trong phạm vi từ 0 đến 69. Ví dụ, chỉ số lớn nhất `qPiMaxIdx` được sử dụng để dẫn xuất `Qpc` có thể được dẫn xuất như trong Phương trình 4 như được mô tả nêu trên.

Thêm nữa, ví dụ, phần tử cú pháp `Qpc_qPi_flag[i]` có thể biểu diễn liệu giá trị `Qpc` có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp `Qpc_qPi_flag[i]` có thể biểu diễn liệu giá trị `Qpc` thứ `i` có được tăng thêm 1 như được so sánh với giá trị `Qpc` thứ `(i-1)` hay không. Ví dụ, `Qpc_qPi_flag[i]` có giá trị là 1 có thể biểu diễn rằng giá trị `Qpc` được tăng thêm 1, và `Qpc_qPi_flag[i]` có giá trị là 0 có thể biểu diễn rằng giá trị `Qpc` không được tăng.

Ví dụ, biến `QpcIdx[qPi]` có thể được dẫn xuất như sau. Ở đây, `qPi` có thể là từ 0 đến 69.

- Trong trường hợp là `qPi < qPi_min_idx`, thì `QpcIdx[qPi]` có thể được tạo cấu hình giống như `qPi`.

- Trong trường hợp là `qPi = qPi_min_idx...qPiMaxIdx`, thì `QpcIdx[qPi]` có thể được tạo cấu hình như `Qpc_qPi_flag[qPi] + QpcIdx[qPi-1]`.

- Trong trường hợp `qPi > qPiMaxIdx`, thì `QpcIdx[qPi]` có thể được tạo cấu hình như `qPi - (qPiMaxIdx - QpcIdx[qPiMaxIdx])`.

Sau đó, Qpc có thể được tạo cấu hình như $QpcIdx[qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ để báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ hay bảng ánh xạ QP sắc độ được dẫn xuất dựa trên thông tin được báo hiệu được sử dụng. Cờ có thể được báo hiệu qua cú pháp cấp cao như tập tham số trình tự (SPS) hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cú pháp cấp cao có thể là như trong bảng sau đây.

Bảng 65

<code>scq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
...	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
...	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 66

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address ctbAddrA of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to CtbAddrInBs, where ctbAddrA is derived as follows:

$$\begin{aligned} x_{Tmp} &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ y_{Tmp} &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[x_{Tmp}][y_{Tmp}] \\ \text{ctbAddrA} &= \text{minTbAddrA} \gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffset}_Y) \% (64 + \text{QpBdOffset}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($xCb + cbWidth / 2, yCb + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbr_qp_offset + slice_joint_cbr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.
- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_c	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 66 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (ví dụ, nếu Qpc_data_default_flag là 0), thì các

biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu $ChromaArrayType$ là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ $Qpc_data()$ trong trường hợp sử dụng các bảng được xác định bởi người dùng cho các thành phần sắc độ tương ứng được đề xuất trong phương án này có thể được báo hiệu như trong bảng sau đây.

Bảng 67

$Qpc_data()$ {	Descriptor
is_separate_chroma_table	$u(1)$
for ($i = 0; i < is_separate_chroma_table ? 2 : 1; i++$) {	
$qPi_min_idx_minus1[i]$	$ue(v)$
$qPi_delta_max_idx[i]$	$uc(v)$
for ($j = qPi_min_idx[i]; j <= qPiMaxIdx[i]; j++$)	
$Qpc_qPi_flag[i][j]$	$u(1)$
}	
}	

Ví dụ, phần tử cú pháp $is_separate_chroma_table$ có thể biểu diễn liệu các tham số liên quan đến bảng lượng tử hóa sắc độ riêng biệt có được báo hiệu cho thành phần Cb và thành phần Cr hay không. Tức là, ví dụ, phần tử cú pháp $is_separate_chroma_table$ có thể biểu diễn liệu hai bảng ánh xạ tham số lượng tử hóa sắc độ riêng biệt có được sử dụng cho thành phần Cb và thành phần Cr hay không. Ví dụ, phần tử cú pháp $is_separate_chroma_table$ có thể biểu diễn liệu các bảng ánh xạ tham số lượng tử hóa độ chói-sắc độ riêng lẻ có được sử dụng lần lượt cho phần dư Cb và phần dư Cr hay không. Ví dụ, $is_separate_chroma_table$ có giá trị là 1 có thể biểu diễn rằng bảng ánh xạ tham số lượng tử hóa sắc độ riêng biệt được báo hiệu cho thành phần Cb và thành phần Cr, và $is_separate_chroma_table$ có giá trị là 0 có thể biểu diễn rằng một bảng ánh xạ tham số lượng tử hóa sắc độ được sử dụng cho thành phần Cb, thành phần Cr, và các thành phần CbCr liên kết. Ví dụ, nếu $is_separate_chroma_table$ value là 1, thì $qPi_min_idx_minus1[i]$, $qPi_delta_max_idx[i]$, và $Qpc_qPi_flag[i][j]$ cho thành phần Cb, và $qPi_min_idx_minus1[i]$, $qPi_delta_max_idx[i]$, và $Qpc_qPi_flag[i][j]$ cho thành phần Cr có thể được báo hiệu. Thêm nữa, ví dụ, nếu giá trị của

is_separate_chroma_table là 0, thì $qPi_min_idx_minus1[i]$, $qPi_delta_max_idx[i]$, và $Qpc_qPi_flag[i][j]$ cho thành phần Cb, thành phần Cr, và thành phần CbCr liên kết có thể được báo hiệu.

Thêm nữa, ví dụ, giá trị thu được bằng cách thêm 1 vào phần tử cú pháp $qPi_min_idx_minus1$ có thể biểu diễn chỉ số qPi nhỏ nhất được sử dụng cho việc lượng tử hóa sắc độ. Giá trị qPi_min_idx có thể là trong phạm vi từ 1 đến 69. Biến $qPi_min_idx[i]$ có thể được tạo cấu hình giống như giá trị thu được bằng cách cộng 1 vào $qPi_min_idx_minus1[i]$.

Thêm nữa, ví dụ, phần tử cú pháp $qPi_delta_max_idx$ có thể biểu diễn giá trị delta giữa $qPi_min_idx[i]$ và chỉ số qPi lớn nhất được sử dụng để dẫn xuất Qpc sắc độ. Giá trị $qPiMaxIdx[i]$ có thể là bằng hoặc lớn hơn $qPi_min_idx[i]$. Giá trị $qPi_delta_max_idx$ có thể là trong phạm vi từ 0 đến 69. Ví dụ, chỉ số lớn nhất $qPiMaxIdx[i]$ được sử dụng để dẫn xuất Qpc có thể được dẫn xuất như trong phương trình sau đây.

Phương trình 10

$$qPiMaxIdx[i] = qPi_min_idx[i] + qPi_delta_max_idx_minus1[i] + 1$$

Giá trị $qPiMaxIdx[i]$ có thể là bằng hoặc lớn hơn $qPi_min_idx_minus1[i]$.

Thêm nữa, ví dụ, phần tử cú pháp $Qpc_qPi_flag[i][j]$ có thể biểu diễn liệu giá trị Qpc thứ j có thành phần sắc độ thứ i có được tăng thêm 1 hay không. Tức là, ví dụ, phần tử cú pháp $Qpc_qPi_flag[i][j]$ có thể biểu diễn liệu giá trị Qpc thứ j của thành phần sắc độ thứ i có được tăng thêm 1 như được so sánh với giá trị Qpc thứ $(j-1)$ hay không. Ví dụ, $Qpc_qPi_flag[j]$ có giá trị là 1 có thể biểu diễn rằng giá trị Qpc thứ j của thành phần sắc độ thứ i được tăng thêm 1 hay không, và $Qpc_qPi_flag[j]$ có giá trị là 0 có thể biểu diễn rằng giá trị Qpc thứ j của thành phần sắc độ thứ i không được tăng.

Ví dụ, biến $QpcIdx[i][qPi]$ có thể được dẫn xuất như trong bảng sau đây. Ở đây, qPi có thể là từ 0 đến 69.

Bảng 68

For $i=0; i < \text{is_separate_chroma_table} ? 2 : 1, i++$

The variable $Q_{pC}Idx[i][qPi]$ for qPi with $qPi = 0 \dots 69$, is derived as follows:

- For $qPi < qPi_min_idx[i]$, $Q_{pC}Idx[i][qPi]$ is set equal to qPi .
- For $qPi = qPi_min_idx[i] \dots qPiMaxIdx[i]$, the following applies:

$$Q_{pC}Idx[i][qPi] = Q_{pC_qPi_flag}[i][qPi] + Q_{pC}Idx[i][qPi - 1]$$
- For $qPi > qPiMaxIdx$

$$Q_{pC}Idx[i][qPi] = qPi - (qPiMaxIdx[i] - Q_{pC}Idx[i][qPiMaxIdx])$$

The value of Q_{pC} of the i th chroma component is derived as $Q_{pC}Idx[i][qPi]$.

Tham chiếu đến Bảng 68, nếu giá trị của `is_separate_chroma_table` là 1, thì dữ liệu tham số lượng tử hóa sắc độ thứ 0 ($i=0$) và dữ liệu tham số lượng tử hóa sắc độ thứ nhất ($i=1$) có thể được báo hiệu. Ở đây, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ 0 ($i=0$) có thể là dữ liệu tham số lượng tử hóa sắc độ để dẫn xuất bảng ánh xạ tham số lượng tử hóa sắc độ cho thành phần Cb, và dữ liệu tham số lượng tử hóa sắc độ thứ nhất ($i=1$) có thể là dữ liệu tham số lượng tử hóa sắc độ để dẫn xuất bảng ánh xạ tham số lượng tử hóa sắc độ cho thành phần Cr.

Thêm nữa, tham chiếu đến Bảng 68, nếu giá trị `is_separate_chroma_table` là 0, thì chỉ dữ liệu tham số lượng tử hóa sắc độ thứ 0 ($i=0$) có thể được báo hiệu. Ở đây, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ 0 ($i=0$) có thể là dữ liệu tham số lượng tử hóa sắc độ để dẫn xuất bảng ánh xạ tham số lượng tử hóa sắc độ cho thành phần Cb, thành phần Cr, và thành phần CbCr kết hợp. Tức là, một bảng ánh xạ tham số lượng tử hóa sắc độ có thể được sử dụng cho các thành phần sắc độ.

Thêm nữa, tham chiếu đến Bảng 68, $Q_{pC}Idx[i][qPi]$ có thể được dẫn xuất như sau.

- Trong trường hợp là $qPi < qPi_min_idx[i]$, thì $Q_{pC}Idx[i][qPi]$ có thể được tạo cấu hình giống như qPi .
- Trong trường hợp là $qPi = qPi_min_idx[i] \dots qPiMaxIdx[i]$, $Q_{pC}Idx[i][qPi]$ có thể được tạo cấu hình như $Q_{pC_qPi_flag}[i][qPi] + Q_{pC}Idx[i][qPi - 1]$.
- Trong trường hợp là $qPi > qPiMaxIdx$, thì $Q_{pC}Idx[i][qPi]$ có thể được tạo cấu hình như $qPi - (qPiMaxIdx[i] - Q_{pC}Idx[i][qPiMaxIdx])$.

Sau đó, giá trị Qpc có thể được dẫn xuất như $QpcIdx[i][qPi]$.

Thêm nữa, phương án này đề xuất sơ đồ báo hiệu cờ biểu diễn liệu bảng mặc định được sử dụng để dẫn xuất việc lượng tử hóa sắc độ, hay thông tin được báo hiệu để dẫn xuất việc lượng tử hóa sắc độ được sử dụng. Cờ có thể được báo hiệu qua cú pháp cấp cao như tập tham số trình tự (SPS) hoặc tập tham số hình ảnh (PPS). Cờ được báo hiệu qua cú pháp cấp cao có thể là giống như cờ được báo hiệu trong bảng sau đây.

Bảng 69

<code>seq_parameter_set_rbsp() {</code>	Descriptor
<code>sps_decoding_parameter_set_id</code>	<code>u(4)</code>
<code>...</code>	
<code>Qpc_data_default_flag</code>	<code>u(1)</code>
<code>if(!Qpc_data_default_flag)</code>	
<code>Qpc_data()</code>	
<code>...</code>	

Ví dụ, phần tử cú pháp `Qpc_data_default_flag` có thể biểu diễn liệu chế độ được xác định bởi người dùng có được sử dụng để dẫn xuất tham số lượng tử hóa hay không. Ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng chế độ được xác định bởi người dùng được sử dụng để dẫn xuất tham số lượng tử hóa. Tức là, ví dụ, `Qpc_data_default_flag` có giá trị là 0 có thể biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` được sử dụng. Nếu `Qpc_data_default_flag` là 0, thì dữ liệu tham số lượng tử hóa sắc độ `Qpc_data()` có thể được báo hiệu. Thêm nữa, ví dụ, `Qpc_data_default_flag` có giá trị là 1 có thể biểu diễn rằng bảng mặc định được sử dụng để dẫn xuất tham số lượng tử hóa. Bảng mặc định có thể là như trong Bảng 7 như được mô tả nêu trên. Thêm nữa, nếu `Qpc_data_default_flag` là không có mặt, thì `Qpc_data_default_flag` có thể được suy luận như 1.

Ví dụ, theo phương án này, thông qua phần mô tả trong định dạng chuẩn, quy trình để dẫn xuất tham số lượng tử hóa có thể được biểu diễn như trong bảng sau đây.

Bảng 70

8.7.1 Derivation process for quantization parameters

Inputs to this process are:

- a luma location (x_{Cb} , y_{Cb}) specifying the top-left luma sample of the current coding block relative to the top-left luma sample of the current picture,
- a variable $cbWidth$ specifying the width of the current coding block in luma samples,
- a variable $cbHeight$ specifying the height of the current coding block in luma samples,
- a variable $treeType$ specifying whether a single tree (SINGLE_TREE) or a dual tree is used to partition the CTUs and, when a dual tree is used, whether the luma (DUAL_TREE_LUMA) or chroma components (DUAL_TREE_CHROMA) are currently processed.

In this process, the luma quantization parameter Qp'_Y and the chroma quantization parameters Qp'_{Cb} and Qp'_{Cr} are derived.

The luma location (x_{Qg} , y_{Qg}), specifies the top-left luma sample of the current quantization group relative to the top left luma sample of the current picture. The horizontal and vertical positions x_{Qg} and y_{Qg} are set equal to $CuQgTopLeftX$ and $CuQgTopLeftY$, respectively.

NOTE – : The current quantization group is a rectangular region inside a coding tree block that shares the same qP_{Y_PREV} . Its width and height are equal to the width and height of the coding tree node of which the top-left luma sample position is assigned to the variables $CuQgTopLeftX$ and $CuQgTopLeftY$.

When $treeType$ is equal to SINGLE_TREE or DUAL_TREE_LUMA, the predicted luma quantization parameter qP_{Y_PREV} is derived by the following ordered steps:

1. The variable qP_{Y_PREV} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_PREV} is set equal to $SliceQp_Y$:
 - The current quantization group is the first quantization group in a slice.
 - The current quantization group is the first quantization group in a brick.
- Otherwise, qP_{Y_PREV} is set

2. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to ($x_{Qg} - 1$, y_{Qg}) as inputs, and the output is assigned to availableA. The variable qP_{Y_A} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_A} is set equal to qP_{Y_PREV} :
 - availableA is equal to FALSE.
 - the CTB address $ctbAddrA$ of the CTB containing the luma coding block covering the luma location ($x_{Qg} - 1$, y_{Qg}) is not equal to $CtbAddrInBs$, where $ctbAddrA$ is derived as follows:

$$\begin{aligned} xTmp &= (x_{Qg} - 1) \gg \text{MinTbLog2SizeY} \\ yTmp &= y_{Qg} \gg \text{MinTbLog2SizeY} \\ \text{minTbAddrA} &= \text{MinTbAddrZs}[xTmp][yTmp] \\ ctbAddrA &= \text{minTbAddrA} \gg (2 * (CtbLog2SizeY - \text{MinTbLog2SizeY})) \end{aligned}$$

- Otherwise, qP_{Y_A} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering ($x_{Qg} - 1$, y_{Qg}).

3. The availability derivation process for a block as specified in clause 6.4.X [Ed. (BB): Neighbouring blocks availability checking process tbd] is invoked with the location (x_{Curr} , y_{Curr}) set equal to (x_{Cb} , y_{Cb}) and the neighbouring location (x_{NbY} , y_{NbY}) set equal to (x_{Qg} , $y_{Qg} - 1$) as inputs, and the output is assigned to availableB. The variable qP_{Y_B} is derived as follows:

- If one or more of the following conditions are true, qP_{Y_B} is set equal to qP_{Y_PREV} :

- availableB is equal to FALSE.
- the CTB address ctbAddrB of the CTB containing the luma coding block covering the luma location (xQg, yQg – 1) is not equal to CtbAddrInBs, where ctbAddrB is derived as follows:

$$\begin{aligned}
 xTmp &= xQg \gg \text{MinTbLog2SizeY} \\
 yTmp &= (yQg - 1) \gg \text{MinTbLog2SizeY} \\
 \text{minTbAddrB} &= \text{MinTbAddrZs}[xTmp][yTmp] \\
 \text{ctbAddrB} &= \\
 \text{minTbAddrB} &\gg (2 * (\text{CtbLog2SizeY} - \text{MinTbLog2SizeY})) \quad (8-922)
 \end{aligned}$$

- Otherwise, qP_{Y_B} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1).
4. The predicted luma quantization parameter qP_{Y_PRED} is derived as follows:
- If all the following conditions are true, then qP_{Y_PRED} is set equal to the luma quantization parameter Qp_Y of the coding unit containing the luma coding block covering (xQg, yQg – 1):
 - availableB is equal to TRUE.
 - the current quantization group is the first quantization group in a CTB row within a brick
 - Otherwise, qP_{Y_PRED} is derived as follows:

$$qP_{Y_PRED} = (qP_{Y_A} + qP_{Y_B} + 1) \gg 1$$

The variable Qp_Y is derived as follows:

$$\begin{aligned}
 Qp_Y &= \\
 &((qP_{Y_PRED} + \text{CuQpDeltaVal} + 64 + 2 * \text{QpBdOffsct}_Y) \% (64 + \text{QpBdOffsct}_Y)) - \text{QpBdOffset}_Y
 \end{aligned}$$

The luma quantization parameter Qp'_Y is derived as follows:

$$Qp'_Y = Qp_Y + QpBdOffset_Y$$

When ChromaArrayType is not equal to 0 and treeType is equal to SINGLE_TREE or DUAL_TREE_CHROMA, the following applies:

- When treeType is equal to DUAL_TREE_CHROMA, the variable Qp_Y is set equal to the luma quantization parameter Qp_Y of the luma coding unit that covers the luma location ($xCb + cbWidth / 2$, $yCb + cbHeight / 2$).

- The variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are derived as follows:

$$qPi_{Cb} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cb_qp_offset + slice_cb_qp_offset)$$

$$qPi_{Cr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_cr_qp_offset + slice_cr_qp_offset)$$

$$qPi_{CbCr} = Clip3(-QpBdOffset_C, 69, Qp_Y + pps_joint_cbr_qp_offset + slice_joint_cbr_qp_offset)$$

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to FALSE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to the value of Qp_C as-specified in clause 7.x.x based on the index $qPi[i]$ equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- If ChromaArrayType is equal to 1, and Qpc_data_default_flag is equal to TRUE, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are specified in Table 8-15 based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively

- Otherwise, the variables qP_{Cb} , qP_{Cr} and qP_{CbCr} are set equal to $Min(qPi, 63)$, based on the index qPi equal to qPi_{Cb} , qPi_{Cr} and qPi_{CbCr} , respectively.

- The chroma quantization parameters for the Cb and Cr components, Qp'_{Cb} and Qp'_{Cr} , and joint Cb-Cr coding Qp'_{CbCr} are derived as follows:

$$Qp'_{Cb} = qP_{Cb} + QpBdOffset_C$$

$$Qp'_{Cr} = qP_{Cr} + QpBdOffset_C$$

$$Qp'_{CbCr} = qP_{CbCr} + QpBdOffset_C$$

Table 8-15 – Specification of Qp_C as a function of qPi for ChromaArrayType equal to 1

qPi	< 3	3	3	3	3	34	3	3	3	3	3	4	4	4	4	> 43
	0	0	1	2	3		5	6	7	8	9	0	1	2	3	
Qp_C	= qPi	2	3	3	3	33	3	3	3	3	3	3	3	3	3	= $qPi - 6$
		9	0	1	2		3	4	4	5	5	6	6	7	7	

Tham chiếu đến Bảng 70 như được mô tả nêu trên, nếu ChromaArrayType là 1, và Qpc_data_default_flag biểu diễn SAI (ví dụ, nếu Qpc_data_default_flag là 0), thì các

biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất dựa trên thông tin được xác định bởi người dùng được báo hiệu như được đề xuất trong phương án này. Thêm nữa, ví dụ, nếu ChromaArrayType là 1, và $Qpc_data_default_flag$ biểu diễn ĐÚNG (ví dụ, nếu $Qpc_data_default_flag$ là 1), thì các biến qP_{Cb} , qP_{Cr} , và qP_{CbCr} có thể được dẫn xuất bằng bảng mặc định dựa trên chỉ số qPi lần lượt giống như qPi_{Cb} , qPi_{Cr} , và qPi_{CbCr} .

Fig. 4 thể hiện theo cách giản lược phương pháp mã hóa ảnh bởi thiết bị mã hóa theo sáng chế này. Phương pháp được bộc lộ trong Fig. 4 có thể được thực hiện bằng thiết bị mã hóa được thực hiện bằng thiết bị mã hóa được bộc lộ trên Fig. 2. Cụ thể, ví dụ, S400 đến S420 của Fig. 4 có thể được thực hiện bởi bộ xử lý phần dư của thiết bị mã hóa, và S430 có thể được thực hiện bằng bộ mã hóa entropy của thiết bị mã hóa. Thêm nữa, mặc dù không được minh họa, quy trình để tạo ra các mẫu được tái dựng và hình ảnh được tái dựng dựa trên các mẫu phần dư và các mẫu dự đoán có thể được thực hiện bằng bộ cộng của thiết bị mã hóa.

Thiết bị mã hóa dẫn xuất các mẫu phần dư cho các thành phần sắc độ (S400). Ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu phần dư thông qua việc trừ các mẫu gốc và các mẫu dự đoán cho khối hiện tại trong hình ảnh hiện tại. Các thành phần sắc độ có thể bao gồm thành phần Cb, thành phần Cr và/hoặc thành phần CbCr liên kết.

Trong khi đó, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán của khối hiện tại dựa trên chế độ dự đoán. Trong trường hợp này, các phương pháp dự đoán khác nhau được bộc lộ trong sáng chế này, như liên dự đoán hoặc nội dự đoán, có thể được áp dụng.

Ví dụ, thiết bị mã hóa có thể xác định liệu thực hiện việc liên dự đoán hay nội dự đoán trên khối hiện tại, và có thể xác định chế độ liên dự đoán cụ thể hoặc chế độ nội dự đoán cụ thể dựa trên chi phí RD. Theo chế độ được xác định, thiết bị mã hóa có thể dẫn xuất các mẫu dự đoán cho khối hiện tại.

Thiết bị mã hóa tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không (S410). Thiết bị mã hóa có thể xác định liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không, và có thể tạo ra cờ.

Ví dụ, thiết bị mã hóa tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ dựa trên loại sắc độ hay không. Ở

đây, loại sắc độ có thể nghĩa là ChromaArrayType được mô tả ở trên. Ví dụ, khi giá trị của loại sắc độ không là 0, thì d mã hóa có thể tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không. Ví dụ, khi giá trị của loại sắc độ là 1, thì thiết bị mã hóa có thể tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không. Ở đây, nếu giá trị của loại sắc độ là 0, loại sắc độ có thể là định dạng đơn sắc, và nếu giá trị của loại sắc độ là 1, thì loại sắc độ có thể là định dạng 4:2:0. Nếu giá trị của loại sắc độ là 2, thì loại sắc độ có thể là định dạng 4:2:2, và nếu giá trị của loại sắc độ là 3, loại sắc độ có thể là định dạng 4:4:4. Ví dụ, phần tử cú pháp cho cờ có thể là `qp_luma_to_chroma_joint_map_flag` flag, `sps_separate_qpc_table_flag`, hoặc `is_separate_chroma_table` được mô tả nêu trên.

Ví dụ, nếu giá trị của cờ là 1, thì cờ có thể biểu diễn rằng một bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Tức là, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng bảng tham số lượng tử hóa sắc độ riêng lẻ được áp dụng cho mỗi thành phần sắc độ.

Thêm nữa, ví dụ, cờ có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thiết bị mã hóa tạo ra dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ (S420). Thiết bị mã hóa có thể tạo ra dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ.

Ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho thành phần sắc độ), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và/hoặc dữ liệu tham số lượng

tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Trong khi đó, ví dụ, thiết bị mã hóa có thể tạo ra cờ cho phép CbCr liên kết biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ liên kết cho thành phần CbCr liên kết là có mặt hay không. Tức là, ví dụ, thiết bị mã hóa có thể xác định liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr là có mặt hay không, và có thể tạo ra cờ cho phép CbCr liên kết. Thêm nữa, ví dụ, thiết bị mã hóa có thể tạo ra cờ cho phép CbCr liên kết biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr liên kết là có mặt dựa trên loại sắc độ hay không. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType như được mô tả nêu trên. Ví dụ, nếu giá trị của loại sắc độ không là 0, thì thiết bị mã hóa có thể tạo ra được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt hay không. Ví dụ, nếu giá trị của loại sắc độ là 1, thì thiết bị mã hóa có thể tạo ra cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt hay không. Thêm nữa, ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Trong trường hợp này, nếu giá trị của cờ là 0 (tức là, xác định được rằng nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), và giá trị của cờ cho phép CbCr kết hợp là 1 (tức là, xác định được rằng dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ nhất và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ thứ nhất. Tức là, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp

biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ nhất và/hoặc phần tử cú pháp cho giá trị tham số lượng tử hóa của mỗi chỉ số của bảng tham số lượng tử hóa sắc độ thứ nhất. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` or `qPi_cb_min_idx_minus1` được mô tả nêu trên. Cũng như vậy, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Cũng như vậy, phần tử cú pháp cho giá trị tham số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả nêu trên. Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu qua cú pháp cấp cao. Ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ hai và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ thứ hai. Tức là, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ hai và/hoặc phần tử cú pháp cho giá trị tham số lượng tử hóa của mỗi chỉ số của bảng tham số lượng tử hóa sắc độ thứ hai. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` or `qPi_cb_min_idx_minus1` được mô tả nêu trên. Cũng như vậy, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Cũng như vậy, phần tử cú pháp cho giá trị tham số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả nêu trên. Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể được báo hiệu qua cú pháp cấp cao. Ví

dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ ba và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ thứ ba. Tức là, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ ba và/hoặc phần tử cú pháp cho giá trị tham số lượng tử hóa của mỗi chỉ số của bảng tham số lượng tử hóa sắc độ thứ ba. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là qPi_min_idx , $qPi_min_idx_minus1[i]$ or $qPi_cb_min_idx_minus1$ được mô tả nêu trên. Cũng như vậy, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là $qPi_delta_max_idx$, $qPi_cb_delta_max_idx_minus1$ hoặc $qPi_delta_max_idx[i]$. Cũng như vậy, phần tử cú pháp cho giá trị tham số lượng tử hóa của chỉ số có thể là $QpC_qPi_val[i]$, $QpC_cb_qPi_flag[i]$ hoặc $QpC_qPi_flag[i][j]$ được mô tả nêu trên. Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể được báo hiệu qua cú pháp cấp cao. Ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thêm nữa, ví dụ, khi giá trị của cờ là 1 (tức là, xác định được rằng một bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ cho thành phần Cb, thành phần Cr và thành phần CbCr kết hợp.

Thiết bị mã hóa mã hóa dữ liệu tham số lượng tử hóa sắc độ và cờ (S430). Thiết bị mã hóa có thể mã hóa dữ liệu tham số lượng tử hóa sắc độ và cờ. Thông tin ảnh có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ và cờ.

Thiết bị mã hóa có thể mã hóa thông tin ảnh bao gồm dữ liệu tham số lượng tử hóa sắc độ, cờ và/hoặc cờ cho phép CbCr kết hợp.

Trong khi đó, ví dụ, thiết bị mã hóa có thể tạo ra và mã hóa thông tin dự đoán cho khối hiện tại. Thông tin dự đoán có thể là thông tin chế độ dự đoán cho khối hiện tại. Thông tin ảnh có thể bao gồm thông tin dự đoán.

Thêm nữa, ví dụ, thiết bị mã hóa có thể mã hóa thông tin phần dư cho các mẫu phần dư.

Ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu phần dư được lượng tử hóa bằng cách lượng tử hóa các mẫu phần dư dựa trên tham số lượng tử hóa sắc độ, dẫn xuất các hệ số biến đổi dựa trên các mẫu phần dư được lượng tử hóa, và tạo ra và mã hóa thông tin phần dư dựa trên các hệ số biến đổi. Thêm nữa, ví dụ, thiết bị mã hóa có thể dẫn xuất các mẫu phần dư được lượng tử hóa bằng cách lượng tử hóa các mẫu phần dư dựa trên tham số lượng tử hóa sắc độ, dẫn xuất các hệ số biến đổi bằng cách biến đổi các mẫu phần dư được lượng tử hóa, và tạo ra và mã hóa thông tin phần dư dựa trên các hệ số biến đổi.

Ví dụ, khi giá trị của cờ là 0, thì thiết bị mã hóa có thể dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cb bằng giải lượng tử hóa các mẫu phần dư cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, và có thể dẫn xuất các mẫu phần dư cho thành phần Cr bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, khi giá trị của cờ là 0, thì thiết bị mã hóa có thể dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cb bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, có thể dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cr bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và có thể dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần CbCr kết hợp bằng cách lượng tử hóa các mẫu phần dư cho thành phần CbCr kết hợp dựa trên tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp. Thêm nữa, ví dụ, khi giá trị của cờ là 1, thì thiết bị mã hóa có thể dẫn xuất các mẫu phần dư được lượng tử hóa cho các thành phần sắc độ bằng cách lượng tử hóa các mẫu phần dư cho thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ.

Thông tin hình ảnh có thể gồm thông tin phần dư. Thêm nữa, ví dụ, thiết bị mã hóa có thể xuất ra thông tin ảnh dưới dạng luồng bit.

Trong khi đó, luồng bit bao gồm thông tin ảnh có thể được truyền đến thiết bị giải mã thông qua mạng hoặc phương tiện lưu trữ (số). Ở đây, mạng có thể bao gồm mạng quảng bá và/hoặc mạng truyền thông, và phương tiện lưu trữ kỹ thuật số có thể bao gồm các loại phương tiện lưu trữ khác nhau như đĩa USB, SD, CD, DVD, đĩa Blu-ray, HDD, và SSD.

Fig.5 thể hiện theo cách giản lược thiết bị mã hóa để thực hiện phương pháp mã hóa ảnh theo sáng chế này. Phương pháp được bộc lộ trong Fig.4 có thể được thực hiện bằng thiết bị mã hóa được thực hiện bằng thiết bị mã hóa được bộc lộ trên Fig.5. Cụ thể, ví dụ, bộ xử lý phần dư của thiết bị mã hóa của Fig.5 có thể thực hiện từ S400 đến S420, bộ mã hóa entropy của thiết bị mã hóa có thể thực hiện S430. Thêm nữa, mặc dù không được minh họa, nhưng quy trình tạo ra các mẫu được tái dựng và hình ảnh được tái dựng dựa trên các mẫu phần dư và các mẫu dự đoán có thể được thực hiện bằng bộ cộng của thiết bị mã hóa.

Fig.6 thể hiện theo cách giản lược phương pháp giải mã ảnh bởi thiết bị giải mã theo sáng chế này. Phương pháp được bộc lộ trong Fig.6 có thể được thực hiện bởi thiết bị giải mã được bộc lộ ở Fig.3. Cụ thể, ví dụ, S600 đến S610 của Fig. 6 có thể được thực hiện bằng bộ giải mã entropy của thiết bị giải mã, S620 đến S640 của Fig. 6 có thể được thực hiện bằng bộ xử lý phần dư của thiết bị giải mã, và S650 của Fig. 6 có thể được thực hiện bằng bộ cộng của thiết bị giải mã.

Thiết bị giải mã thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không (S600). Thiết bị giải mã có thể thu được thông tin ảnh thông qua luồng bit. Ví dụ, thông tin ảnh có thể bao gồm thông tin về tham số lượng tử hóa sắc độ. Ví dụ, thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không. Tức là, ví dụ, thông tin ảnh có thể bao gồm cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ hay không. Ví dụ, thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng cho các thành phần sắc độ dựa trên loại sắc độ hay không. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType như được mô tả nêu trên. Ví dụ, nếu giá trị của loại sắc độ không là 0, thì thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng hay không. Ví dụ, nếu giá trị của

loại sắc độ là 1, thì thiết bị giải mã có thể thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng hay không. Ở đây, nếu giá trị của loại sắc độ là 0, loại sắc độ có thể là định dạng đơn sắc, và nếu giá trị của loại sắc độ là 1, thì loại sắc độ có thể là định dạng 4:2:0. Nếu giá trị của loại sắc độ là 2, thì loại sắc độ có thể là định dạng 4:2:2, và nếu giá trị của loại sắc độ là 3, loại sắc độ có thể là định dạng 4:4:4. Thêm nữa, ví dụ, các thành phần sắc độ có thể bao gồm thành phần Cb, thành phần Cr và/hoặc thành phần CbCr kết hợp. Ví dụ, phần tử cú pháp cho cờ có thể là `qp_luma_to_chroma_joint_map_flag`, `sps_separate_qpc_table_flag`, hoặc `is_separate_chroma_table` được mô tả nêu trên.

Ví dụ, nếu giá trị của cờ là 1, thì cờ có thể biểu diễn rằng một bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Thêm nữa, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ. Tức là, ví dụ, nếu giá trị của cờ là 0, cờ có thể biểu diễn rằng bảng tham số lượng tử hóa sắc độ riêng lẻ được áp dụng cho mỗi thành phần sắc độ.

Thêm nữa, ví dụ, cờ có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thiết bị giải mã thu được dữ liệu tham số lượng tử hóa sắc độ dựa trên cờ (S610). Thiết bị giải mã có thể thu được dữ liệu tham số lượng tử hóa sắc độ dựa trên cờ.

Ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho thành phần sắc độ), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb và dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, khi giá trị của cờ là 0 (tức là, khi xác định được là nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và/hoặc dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Trong khi đó, ví dụ, thiết bị mã hóa có thể thu được cờ cho phép CbCr kết hợp

biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr là có mặt hay không. Ví dụ, thông tin ảnh có thể bao gồm cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt hay không. Thêm nữa, ví dụ, thiết bị giải mã có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt dựa trên loại sắc độ hay không. Ở đây, loại sắc độ có thể nghĩa là ChromaArrayType như được mô tả nêu trên. Ví dụ, nếu giá trị của loại sắc độ không là 0, thì thiết bị giải mã có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt hay không. Ví dụ, nếu giá trị của loại sắc độ là 1, thì thiết bị giải mã có thể thu được cờ cho phép CbCr kết hợp biểu diễn liệu dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt hay không. Thêm nữa, ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu thông qua cú pháp cấp cao. Ví dụ, cờ cho phép CbCr kết hợp có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Trong trường hợp này, nếu giá trị của cờ là 0 (tức là, cờ biểu diễn nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), và giá trị của cờ cho phép CbCr kết hợp là 1 (tức là, cờ cho phép CbCr kết hợp biểu diễn rằng dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp là có mặt), dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ nhất và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ thứ nhất. Tức là, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ nhất, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ nhất và/hoặc phần tử cú pháp cho giá trị tham số lượng tử hóa

của mỗi chỉ số của bảng tham số lượng tử hóa sắc độ thứ nhất. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` or `qPi_cb_min_idx_minus1` được mô tả nêu trên. Cũng như vậy, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Cũng như vậy, phần tử cú pháp cho giá trị tham số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả nêu trên. Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu qua cú pháp cấp cao. Ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ nhất có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ hai và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ thứ hai. Tức là, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ hai, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ hai và/hoặc phần tử cú pháp cho giá trị tham số lượng tử hóa của mỗi chỉ số của bảng tham số lượng tử hóa sắc độ thứ hai. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` or `qPi_cb_min_idx_minus1` được mô tả nêu trên. Cũng như vậy, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Cũng như vậy, phần tử cú pháp cho giá trị tham số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả nêu trên. Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể được báo hiệu qua cú pháp cấp cao. Ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ hai có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ ba và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ thứ ba. Tức là, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể bao gồm phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử hóa sắc độ thứ ba, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ thứ ba và/hoặc phần tử cú pháp cho giá trị tham số lượng tử hóa của mỗi chỉ số của bảng tham số lượng tử hóa sắc độ thứ ba. Phần tử cú pháp biểu diễn chỉ số bắt đầu có thể là `qPi_min_idx`, `qPi_min_idx_minus1[i]` or `qPi_cb_min_idx_minus1` được mô tả nêu trên. Cũng như vậy, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng có thể là `qPi_delta_max_idx`, `qPi_cb_delta_max_idx_minus1` hoặc `qPi_delta_max_idx[i]`. Cũng như vậy, phần tử cú pháp cho giá trị tham số lượng tử hóa của chỉ số có thể là `QpC_qPi_val[i]`, `QpC_cb_qPi_flag[i]` hoặc `QpC_qPi_flag[i][j]` được mô tả nêu trên. Thêm nữa, ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể được báo hiệu qua cú pháp cấp cao. Ví dụ, dữ liệu tham số lượng tử hóa sắc độ thứ ba có thể được báo hiệu qua tập tham số trình tự (SPS), tập tham số hình ảnh (PPS), tiêu đề lát, hoặc tập tham số thích ứng (APS).

Thêm nữa, ví dụ, khi giá trị của cờ là 1 (tức là, khi cờ biểu diễn một bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì dữ liệu tham số lượng tử hóa sắc độ có thể bao gồm dữ liệu tham số lượng tử hóa sắc độ cho thành phần Cb, thành phần Cr và thành phần CbCr kết hợp.

Thiết bị giải mã dẫn xuất bảng tham số lượng tử hóa sắc độ dựa trên dữ liệu tham số lượng tử hóa sắc độ (S620). Thiết bị giải mã có thể dẫn xuất bảng tham số lượng tử hóa sắc độ dựa trên dữ liệu tham số lượng tử hóa sắc độ. Bảng tham số lượng tử hóa sắc độ có thể được gọi là bảng ánh xạ tham số lượng tử hóa sắc độ, hoặc bảng ánh xạ tham số lượng tử hóa được xác định bởi người dùng.

Ví dụ, như được mô tả nêu trên, bảng tham số lượng tử hóa sắc độ có thể được dẫn xuất dựa trên phần tử cú pháp biểu diễn chỉ số bắt đầu của bảng tham số lượng tử

hóa sắc độ, phần tử cú pháp biểu diễn mức chênh lệch giữa chỉ số bắt đầu và chỉ số cuối cùng của bảng tham số lượng tử hóa sắc độ và/hoặc các phần tử cú pháp cho các giá trị tham số lượng tử hóa của các chỉ số của bảng tham số lượng tử hóa sắc độ. Tức là, ví dụ, bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa.

Ví dụ, nếu giá trị của c_r là 0 (tức là, c_r biểu diễn nhiều bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần sắc độ Cb có thể được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb. Thêm nữa, ví dụ, nếu giá trị của c_r là 0, thì bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần sắc độ Cr có thể được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, nếu giá trị của c_r là 0, thì bảng tham số lượng tử hóa sắc độ thứ ba cho thành phần sắc độ CbCr có thể được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp.

Thêm nữa, ví dụ, nếu giá trị của c_r là 1 (tức là, c_r biểu diễn một bảng tham số lượng tử hóa sắc độ được áp dụng cho các thành phần sắc độ), thì bảng tham số lượng tử hóa sắc độ thứ nhất cho các thành phần sắc độ có thể được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ. Các thành phần sắc độ có thể bao gồm thành phần Cb, thành phần Cr, và/hoặc thành phần CbCr kết hợp.

Thiết bị giải mã dẫn xuất tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng tham số lượng tử hóa sắc độ (S630). Thiết bị giải mã có thể dẫn xuất tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên bảng tham số lượng tử hóa sắc độ.

Ví dụ, nếu giá trị của c_r là 0, thì tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb có thể được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ nhất, và tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr có thể được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ hai. Thêm nữa, ví dụ, nếu giá trị của c_r là 0, thì tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb có thể được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ nhất, tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr có thể được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ hai, và tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp có thể được dẫn

xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ ba. Ở đây, tham số lượng tử hóa cho thành phần Cb có thể biểu diễn QP'_{Cb} như được mô tả nêu trên, tham số lượng tử hóa cho thành phần Cr có thể biểu diễn QP'_{Cr} như được mô tả nêu trên, và tham số lượng tử hóa cho thành phần CbCr kết hợp có thể biểu diễn QP'_{CbCr} như được mô tả nêu trên.

Ví dụ, chỉ số cho các thành phần sắc độ (thành phần Cb, thành phần Cr, hoặc thành phần CbCr kết hợp) có thể được dẫn xuất dựa trên tham số lượng tử hóa cho thành phần độ chói, và tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên tham số lượng tử hóa sắc độ cho chỉ số của bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ. Tức là, ví dụ, tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên tham số lượng tử hóa sắc độ cho chỉ số giống như tham số lượng tử hóa của thành phần độ chói trong bảng tham số lượng tử hóa sắc độ.

Thêm nữa, ví dụ, tham số lượng tử hóa sắc độ (tức là, QP'_{Cb} , QP'_{Cr} , hoặc QP'_{CbCr}) cho các thành phần sắc độ có thể được dẫn xuất bằng cách cộng độ bù vào tham số lượng tử hóa sắc độ (ví dụ, QP_{Cb} , QP_{Cr} , hoặc QP_{CbCr}) cho chỉ số của bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ (thành phần Cb, thành phần Cr, hoặc thành phần CbCr kết hợp). Độ bù có thể được dẫn xuất dựa trên phần tử cú pháp biểu diễn độ bù để dẫn xuất tham số lượng tử hóa cho các thành phần sắc độ.

Thêm nữa, ví dụ, nếu giá trị của cờ là 1, tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ. Tức là, ví dụ, nếu giá trị của cờ là 1, thì tham số lượng tử hóa sắc độ cho các thành phần sắc độ được dẫn xuất dựa trên một bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ. Theo đó, tham số lượng tử hóa sắc độ có thể được áp dụng ngang bằng với các thành phần sắc độ.

Ví dụ, chỉ số cho các thành phần sắc độ (thành phần Cb, thành phần Cr, và thành phần CbCr kết hợp) có thể được dẫn xuất dựa trên tham số lượng tử hóa cho thành phần độ chói, và tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên tham số lượng tử hóa sắc độ cho chỉ số của bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ. Tức là, ví dụ, tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất dựa trên tham số lượng tử hóa sắc độ cho chỉ số giống như tham số lượng tử hóa của thành phần độ chói trong bảng tham số lượng tử hóa sắc độ.

độ.

Thêm nữa, ví dụ, tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được dẫn xuất bằng cách cộng độ bù vào tham số lượng tử hóa sắc độ cho chỉ số của bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ. Độ bù có thể được dẫn xuất dựa trên phần tử cú pháp biểu diễn độ bù để dẫn xuất tham số lượng tử hóa cho các thành phần sắc độ.

Thiết bị giải mã dẫn xuất các mẫu phần dư cho các thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ (S640). Thiết bị giải mã có thể dẫn xuất các mẫu phần dư dựa trên tham số lượng tử hóa sắc độ.

Ví dụ, thiết bị giải mã có thể dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư thu được. Sau đó, nếu giá trị còn lại là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu phần dư cho thành phần Cb bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, và có thể dẫn xuất các mẫu phần dư cho thành phần Cr bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, nếu giá trị còn lại là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu phần dư cho thành phần Cb bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dẫn xuất các mẫu phần dư cho thành phần Cr bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dẫn xuất các mẫu phần dư cho thành phần CbCr kết hợp bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần CbCr kết hợp dựa trên tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp. Thêm nữa, ví dụ, nếu giá trị còn lại là 1, thì thiết bị giải mã có thể dẫn xuất các mẫu phần dư cho các thành phần sắc độ bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ.

Thêm nữa, ví dụ, thiết bị giải mã có thể dẫn xuất các hệ số biến đổi dựa trên thông tin phần dư, và có thể dẫn xuất các hệ số biến đổi được biến đổi ngược bằng cách biến đổi ngược các hệ số biến đổi. Sau đó, nếu giá trị còn lại là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu phần dư cho thành phần Cb bằng cách giải lượng tử hóa các hệ số biến đổi được biến đổi ngược cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, và có thể dẫn xuất các mẫu phần dư cho thành phần Cr bằng

cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr. Thêm nữa, ví dụ, nếu giá trị còn là 0, thì thiết bị giải mã có thể dẫn xuất các mẫu phần dư cho thành phần Cb bằng cách giải lượng tử hóa các hệ số biến đổi được biến đổi ngược cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, dẫn xuất các mẫu phần dư cho thành phần Cr bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và dẫn xuất các mẫu phần dư cho thành phần CbCr kết hợp bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần CbCr kết hợp dựa trên tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp. Thêm nữa, ví dụ, nếu giá trị còn là 1, thì thiết bị giải mã có thể dẫn xuất các mẫu phần dư cho các thành phần sắc độ bằng cách giải lượng tử hóa các hệ số biến đổi được biến đổi ngược cho thành phần sắc độ dựa trên tham số lượng tử hóa sắc độ.

Thiết bị giải mã tạo ra hình ảnh được tái dựng dựa trên các mẫu phần dư (S650). Ví dụ, thiết bị giải mã có thể tạo ra hình ảnh được tái dựng dựa trên các mẫu phần dư.

Trong khi đó, ví dụ, thiết bị giải mã có thể dẫn xuất các mẫu dự đoán bằng cách thực hiện chế độ liên dự đoán hoặc chế độ nội dự đoán cho khôi hiện tại dựa trên thông tin dự đoán thu được qua luồng bit, và có thể tạo ra các mẫu được tái dựng và/hoặc hình ảnh được tái dựng bằng cách cộng các mẫu dự đoán và các mẫu phần dư.

Sau đó, nếu cần thiết, để nâng cao chất lượng hình ảnh chủ quan/khách quan, thủ tục lọc trong vòng, như lọc khử khối, SAO và/hoặc thủ tục ALF, có thể được áp dụng cho các mẫu được tái dựng như được mô tả ở trên.

Fig.7 thể hiện theo cách giản lược thiết bị giải mã để thực hiện phương pháp giải mã ảnh theo sáng chế này. Phương pháp được bộc lộ trong Fig.6 có thể được thực hiện bởi thiết bị giải mã được bộc lộ ở Fig.7. Cụ thể, ví dụ, bộ giải mã entropy của thiết bị giải mã của Fig.7 có thể thực hiện S600 đến S610 của Fig.6, bộ xử lý phần dư của thiết bị giải mã của Fig.7 có thể thực hiện S620 đến S640 của Fig. 6, và bộ cộng của thiết bị giải mã của Fig.7 có thể thực hiện S650 của Fig. 6.

Theo sáng chế này như được mô tả nêu trên, để dẫn xuất tham số lượng tử hóa cho các thành phần sắc độ, bảng tham số lượng tử hóa sắc độ có thể được xác định dựa trên cờ biểu diễn liệu cùng một bảng tham số lượng tử hóa sắc độ được sử dụng hay

không, và thông qua điều này, hiệu quả lập mã có thể được nâng cao thông qua cách thực hiện việc lập mã dựa trên tham số lượng tử hóa theo các đặc tính của ảnh.

Thêm nữa, theo sáng chế này, bảng tham số lượng tử hóa sắc độ cho các thành phần sắc độ có thể được xác định dựa trên dữ liệu lượng tử hóa sắc độ được báo hiệu riêng lẻ hoặc chung cho các thành phần sắc độ, và thông qua điều này, hiệu quả lập mã có thể được nâng cao thông qua thực hiện việc lập mã dựa trên tham số lượng tử hóa theo các đặc tính của ảnh.

Ở phương án được mô tả ở trên, các phương pháp được mô tả dựa trên sơ đồ tiến trình có chuỗi các bước hoặc các khối. Sáng chế không được giới hạn ở thứ tự của các bước hoặc các khối ở trên. Một số bước hoặc khối có thể xuất hiện đồng thời hoặc theo trật tự khác với các bước hoặc các khối khác được mô tả ở trên. Thêm nữa, người có hiểu biết trung bình trong lĩnh vực kỹ thuật này sẽ hiểu rằng các bước được thể hiện trong sơ đồ tiến trình ở trên không phải là riêng biệt và các bước thêm nữa cũng thể được bao gồm, hoặc một hoặc nhiều bước trong các sơ đồ tiến trình làm ví dụ có thể được xóa mà không ảnh hưởng tới phạm vi bảo hộ của sáng chế này.

Các phương án được mô tả trong bản mô tả này có thể được thực hiện trên bộ xử lý, bộ vi xử lý, bộ điều khiển hoặc chip. Ví dụ, các đơn vị chức năng được thể hiện ở mỗi hình vẽ có thể được thực hiện bằng cách được triển khai trên máy tính, bộ xử lý, bộ vi xử lý, bộ điều khiển hoặc chip. Trong trường hợp này, thông tin cho việc triển khai (ví dụ, thông tin về các lệnh) hoặc thuật toán có thể được lưu trữ trong phương tiện lưu trữ số.

Ngoài ra, thiết bị giải mã và thiết bị mã hóa mà sáng chế được áp dụng cho có thể được bao gồm trong thiết bị truyền/thu quảng bá đa phương tiện, thiết bị đầu cuối truyền thông di động, thiết bị video chiếu phim gia đình, thiết bị video chiếu phim số, camera giám sát, thiết bị trò chuyện video, thiết bị truyền thông thời gian thực như truyền thông video, thiết bị tạo luồng (streaming) di động, phương tiện lưu trữ, máy ghi hình cầm tay, thiết bị cung cấp dịch vụ video theo yêu cầu (Video on Demand, VoD), thiết bị cung cấp dịch vụ nội dung trên nền mạng viễn thông (Over The Top, OTT), thiết bị cung cấp dịch vụ tạo luồng Internet, thiết bị video ba chiều (Three Dimensional, 3D), thiết bị video dùng cho các cuộc họp tổ chức qua điện thoại, thiết bị người dùng cho vận tải (ví dụ, thiết bị người dùng cho xe cộ, thiết bị người dùng cho máy bay, thiết bị người dùng

cho tàu, v.v.) và thiết bị video y tế và có thể được dùng để xử lý các tín hiệu video hoặc các tín hiệu dữ liệu. Ví dụ, thiết bị video của dịch vụ cung cấp nội dung trên nền mạng viễn thông (OTT) có thể bao gồm bảng điều khiển chơi trò chơi, máy đọc đĩa Blu-ray, tivi truy cập Internet, hệ thống rạp hát gia đình, điện thoại thông minh, máy tính bảng, đầu ghi video kỹ thuật số (Digital Video Recorder, DVR), và tương tự.

Ngoài ra, phương pháp xử lý mà sáng chế được áp dụng cho có thể được tạo ra dưới dạng chương trình mà để được thực thi bởi máy tính, và có thể được lưu trữ trên phương tiện ghi có thể đọc được bằng máy tính. Dữ liệu đa phương tiện mà có cấu trúc dữ liệu theo sáng chế thì cũng có thể được lưu giữ trên phương tiện ghi đọc được bằng máy tính. Phương tiện ghi đọc được bởi máy tính bao gồm tất cả các loại thiết bị lưu trữ mà dữ liệu đọc được bởi máy tính được lưu trong đó. Phương tiện ghi đọc được bằng máy tính này có thể bao gồm BD, thiết bị buýt nối tiếp vạn năng (Universal Serial Bus, USB), ROM, PROM, EPROM, RAM, CD-ROM, băng từ, đĩa mềm, và thiết bị lưu trữ dữ liệu quang học chẳng hạn. Hơn nữa, phương tiện ghi đọc được bởi máy tính chứa phương tiện được thực hiện ở dạng của các sóng bộ mang (ví dụ, truyền qua Internet). Hơn nữa, luồng bit được tạo ra bằng phương pháp mã hóa có thể được lưu trong phương tiện ghi đọc được bởi máy tính hoặc có thể được truyền qua các mạng truyền thông có dây hoặc không dây.

Ngoài ra, các phương án theo sáng chế có thể được thực hiện với sản phẩm chương trình máy tính bằng các mã chương trình, và các mã chương trình này có thể được thực hiện trên máy tính bởi các phương án theo sáng chế. Các mã chương trình có thể được lưu trữ trên vật mang mà có thể được đọc bởi máy tính.

Fig.8 là hình vẽ minh họa sơ đồ cấu trúc của hệ thống tạo luồng các nội dung mà sáng chế được áp dụng.

Hệ thống tạo luồng nội dung mà phương án thực hiện (các phương án thực hiện) của sáng chế này được áp dụng, có thể phần lớn bao gồm máy chủ mã hóa, máy chủ tạo luồng, máy chủ web, bộ phận lưu trữ phương tiện, thiết bị người dùng, và thiết bị nhập vào đa phương tiện.

Máy chủ mã hóa nén các nội dung được nhập vào từ các thiết bị đầu vào đa phương tiện như điện thoại thông minh, camera, máy quay video, v.v. thành dữ liệu dạng

số để tạo ra luồng bit và truyền luồng bit tới máy chủ tạo luồng. Như là một ví dụ khác, khi thiết bị đầu vào đa phương tiện như điện thoại thông minh, camera, máy quay video, v.v. trực tiếp tạo ra luồng bit, thì máy chủ mã hóa có thể được lược bỏ.

Luồng bit có thể được tạo ra bởi phương pháp mã hóa hoặc phương pháp tạo ra luồng bit mà phương án (các phương án) của sáng chế được áp dụng, và máy chủ tạo luồng có thể lưu trữ tạm thời luồng bit trong quy trình truyền hoặc thu luồng bit.

Máy chủ tạo luồng truyền dữ liệu trung gian đến thiết bị người dùng dựa trên yêu cầu của người dùng thông qua máy chủ web, và máy chủ web phục vụ như phương tiện để thông báo người dùng về dịch vụ. Khi người dùng yêu cầu các dịch vụ mong muốn từ máy chủ web, thì máy chủ web chuyển yêu cầu này đến máy chủ tạo luồng, và máy chủ tạo luồng truyền dữ liệu đa phương tiện đến người dùng. Trong trường hợp này, hệ thống tạo luồng các nội dung có thể bao gồm máy chủ điều khiển riêng biệt. Trong trường hợp này, máy chủ điều khiển có chức năng để điều khiển mệnh lệnh/phản hồi giữa các thiết bị trong hệ thống tạo luồng nội dung.

Máy chủ tạo luồng có thể nhận nội dung từ bộ phận lưu trữ phương tiện và/hoặc máy chủ mã hóa. Ví dụ, khi nội dung được thu từ máy chủ mã hóa, thì nội dung có thể được thu theo thời gian thực. Trong trường hợp này, để cung cấp dịch vụ tạo luồng tron tru, thì máy chủ phát luồng có thể lưu trữ luồng bit trong thời gian được xác định trước.

Các ví dụ về thiết bị người dùng có thể bao gồm điện thoại di động, điện thoại thông minh, máy tính xách tay, thiết bị đầu cuối truyền quảng bá số, thiết bị trợ giúp số cá nhân (Personal digital assistant, PDA), thiết bị chơi đa phương tiện di động (Portable multimedia player, PMP), thiết bị định vị, PC dạng phiên, các PC dạng bảng, các máy tính siêu di động, các thiết bị đeo được (ví dụ, Các đồng hồ thông minh, các kính thông minh, các bộ hiển thị gắn trên đầu), các ti vi số, máy tính để bàn, bảng ký hiệu dạng số và tương tự. Mỗi máy chủ trong hệ thống tạo luồng các nội dung có thể được vận hành như là máy chủ phân tán, mà trong trường hợp đó dữ liệu được thu từ từng máy chủ có thể được phân tán.

Các yêu cầu bảo hộ được mô tả trong sáng chế này có thể được kết hợp theo các cách khác nhau. Ví dụ, các đặc điểm kỹ thuật của các yêu cầu bảo hộ phương pháp của sáng chế có thể được kết hợp để được thực hiện như thiết bị, và các đặc điểm kỹ thuật

của các yêu cầu bảo hộ thiết bị của sáng chế có thể được kết hợp để được thực hiện như phương pháp. Ngoài ra, các đặc điểm kỹ thuật của yêu cầu bảo hộ phương pháp của sáng chế và các đặc điểm kỹ thuật của yêu cầu bảo hộ thiết bị có thể được kết hợp để được thực hiện như thiết bị, và các đặc điểm kỹ thuật của yêu cầu bảo hộ phương pháp của sáng chế và các đặc điểm kỹ thuật của yêu cầu bảo hộ thiết bị có thể được kết hợp để được thực hiện như phương pháp.

YÊU CẦU BẢO HỘ

1. Phương pháp giải mã ảnh, được thực hiện bởi thiết bị giải mã, phương pháp bao gồm các bước:

thu được cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng hay không;

thu được ít nhất một dữ liệu tham số lượng tử hóa sắc độ dựa trên cờ;

dẫn xuất ít nhất một bảng tham số lượng tử hóa sắc độ dựa trên ít nhất một dữ liệu tham số lượng tử hóa sắc độ;

dẫn xuất ít nhất một tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên ít nhất một bảng tham số lượng tử hóa sắc độ;

dẫn xuất các mẫu phần dư cho các thành phần sắc độ dựa trên ít nhất một tham số lượng tử hóa sắc độ; và

tạo ra hình ảnh được tái dựng dựa trên các mẫu phần dư,

trong đó cờ bằng 0 biểu diễn rằng bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp được báo hiệu,

trong đó bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ nhất, và bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ hai,

trong đó tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ nhất, và tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ hai, và

trong đó bước dẫn xuất các mẫu phần dư cho các thành phần sắc độ dựa trên ít nhất một tham số lượng tử hóa sắc độ bao gồm:

dẫn xuất các hệ số biến đổi cho các thành phần sắc độ dựa trên thông tin phần dư thu được;

dẫn xuất các mẫu phần dư cho thành phần Cb bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất; và

dẫn xuất các mẫu phần dư cho thành phần Cr bằng cách giải lượng tử hóa các hệ số biến đổi cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai.

2. Phương pháp mã hóa ảnh, được thực hiện bởi thiết bị mã hóa, phương pháp bao gồm các bước:

dẫn xuất các mẫu phần dư cho các thành phần sắc độ;

tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng hay không;

tạo ra ít nhất một dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ; và

mã hóa ít nhất một dữ liệu tham số lượng tử hóa sắc độ và cờ,

trong đó cờ bằng 0 biểu diễn rằng bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp được báo hiệu,

trong đó bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ nhất, và bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ hai,

trong đó tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ nhất, và tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ hai, và

trong đó bước dẫn xuất các mẫu phần dư cho các thành phần sắc độ bao gồm:

dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cb bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất;

dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cr bằng cách lượng

tử hóa các mẫu phần dư cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai; và

dẫn xuất các hệ số biến đổi dựa trên việc biến đổi các thành phần sắc độ được lượng tử hóa.

3. Phương tiện lưu trữ phi chuyên tiếp đọc được bởi máy tính lưu trữ luồng bit được tạo ra bằng phương pháp, phương pháp bao gồm các bước:

dẫn xuất các mẫu phần dư cho các thành phần sắc độ;

tạo ra cờ biểu diễn liệu một bảng tham số lượng tử hóa sắc độ có được áp dụng hay không;

tạo ra ít nhất một dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ dựa trên cờ;

mã hóa ít nhất một dữ liệu tham số lượng tử hóa sắc độ và cờ; và

tạo ra luồng bit bao gồm ít nhất một dữ liệu tham số lượng tử hóa sắc độ và cờ,

trong đó cờ bằng 0 biểu diễn rằng bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp được báo hiệu,

trong đó bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ nhất, và bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ hai,

trong đó tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ nhất, và tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ hai, và

trong đó bước dẫn xuất các mẫu phần dư cho các thành phần sắc độ bao gồm:

dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cb bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất;

dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cr bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai; và

dẫn xuất các hệ số biến đổi dựa trên việc biến đổi các thành phần sắc độ được lượng tử hóa.

4. Phương pháp truyền dữ liệu cho ảnh, phương pháp bao gồm các bước:

thu được luồng bit của thông tin ảnh bao gồm cờ và ít nhất một dữ liệu tham số lượng tử hóa sắc độ cho các thành phần sắc độ; và

truyền dữ liệu bao gồm luồng bit của thông tin ảnh bao gồm cờ và ít nhất một dữ liệu tham số lượng tử hóa sắc độ,

trong đó các mẫu phần dư cho các thành phần sắc độ được dẫn xuất,

trong đó cờ bằng 0 biểu diễn rằng bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb, bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr, và bảng tham số lượng tử hóa sắc độ thứ ba cho thành phần CbCr kết hợp được báo hiệu,

trong đó bảng tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ nhất, và bảng tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên dữ liệu tham số lượng tử hóa sắc độ thứ hai,

trong đó tham số lượng tử hóa sắc độ thứ nhất cho thành phần Cb được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ nhất, và tham số lượng tử hóa sắc độ thứ hai cho thành phần Cr được dẫn xuất dựa trên bảng tham số lượng tử hóa sắc độ thứ hai, và

trong đó bước dẫn xuất các mẫu phần dư cho các thành phần sắc độ bao gồm:

dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cb bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cb dựa trên tham số lượng tử hóa sắc độ thứ nhất;

dẫn xuất các mẫu phần dư được lượng tử hóa cho thành phần Cr bằng cách lượng tử hóa các mẫu phần dư cho thành phần Cr dựa trên tham số lượng tử hóa sắc độ thứ hai; và

dẫn xuất các hệ số biến đổi dựa trên việc biến đổi các thành phần sắc độ được lượng tử hóa.

FIG. 1

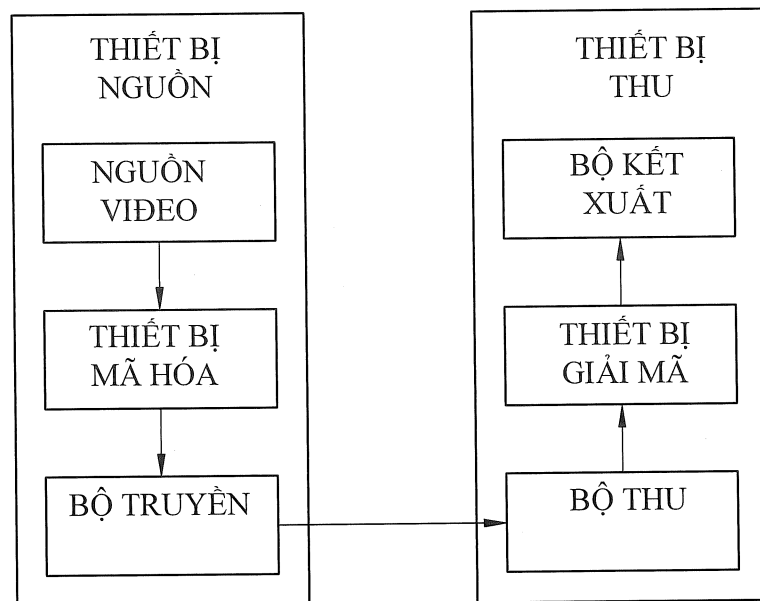


FIG. 2

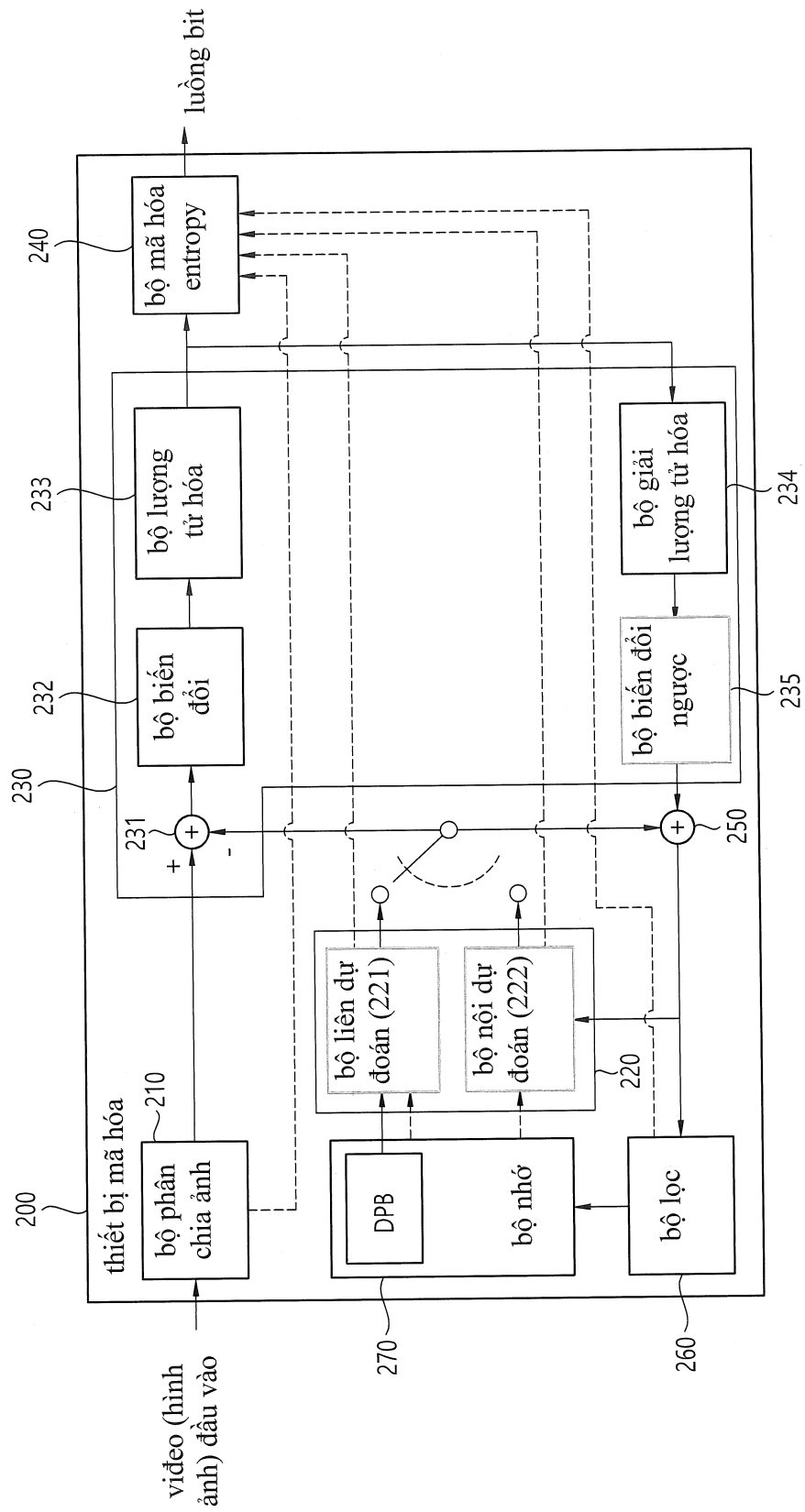


FIG. 3

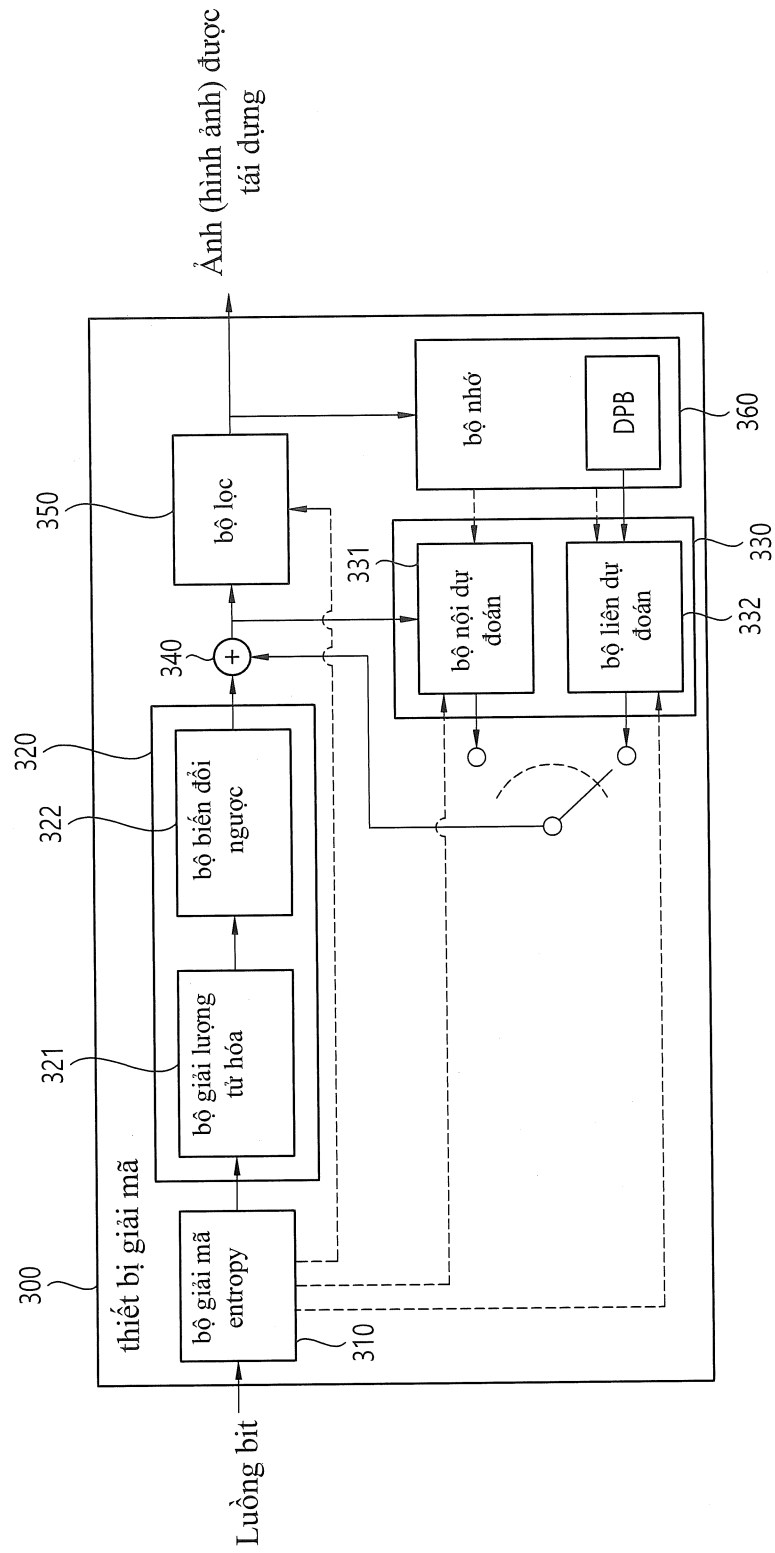


FIG. 4

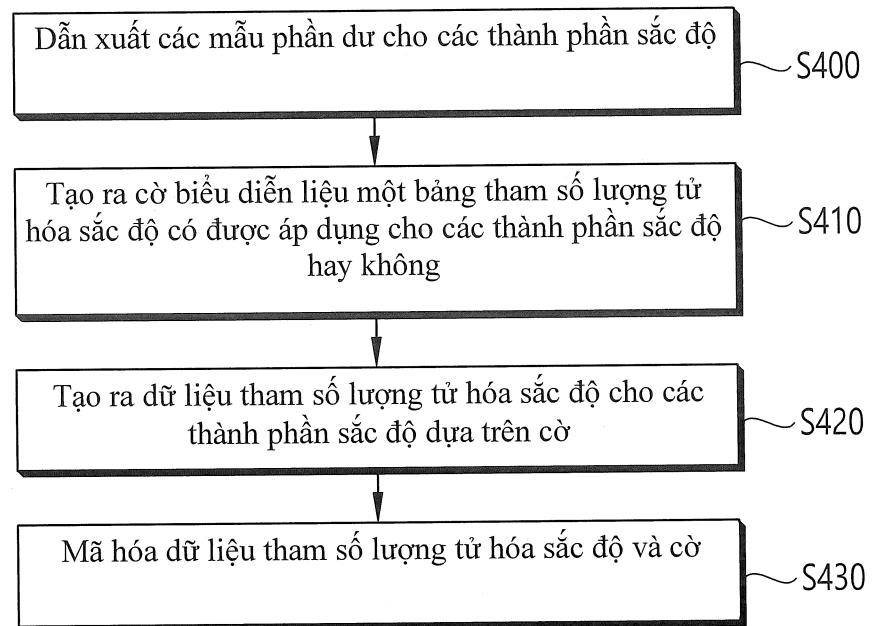


FIG. 5

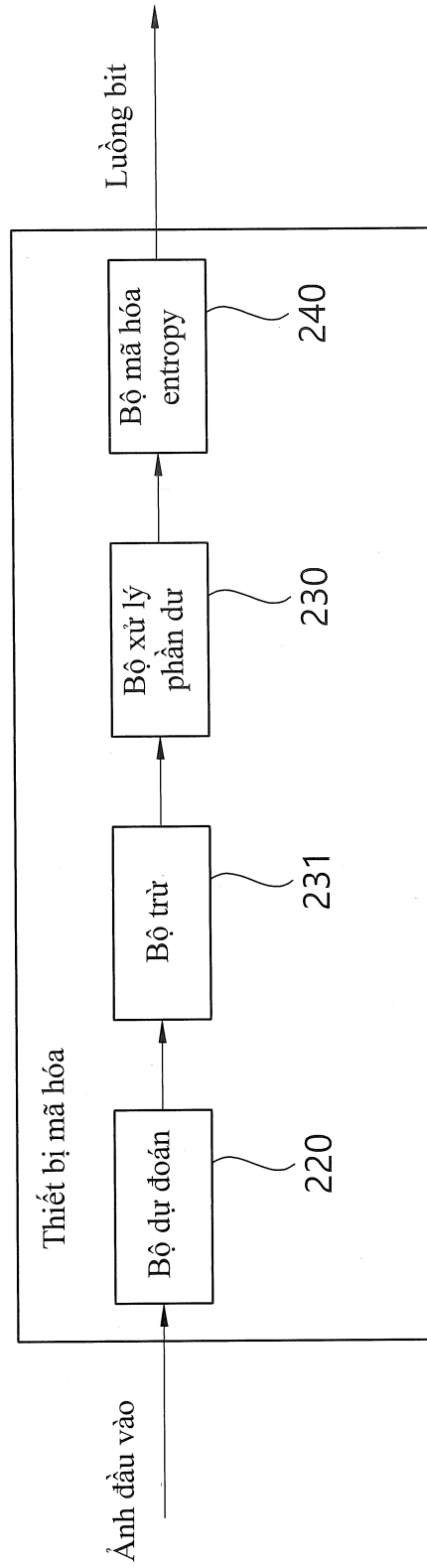


FIG. 6

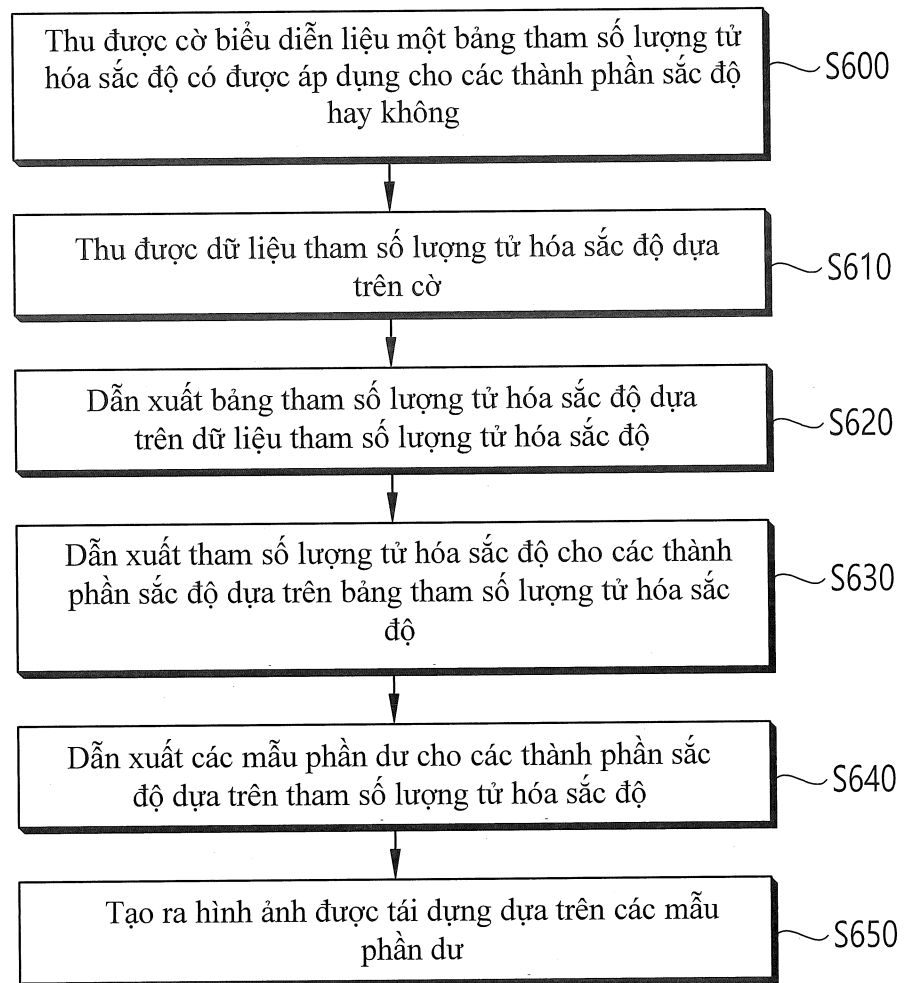


FIG. 7

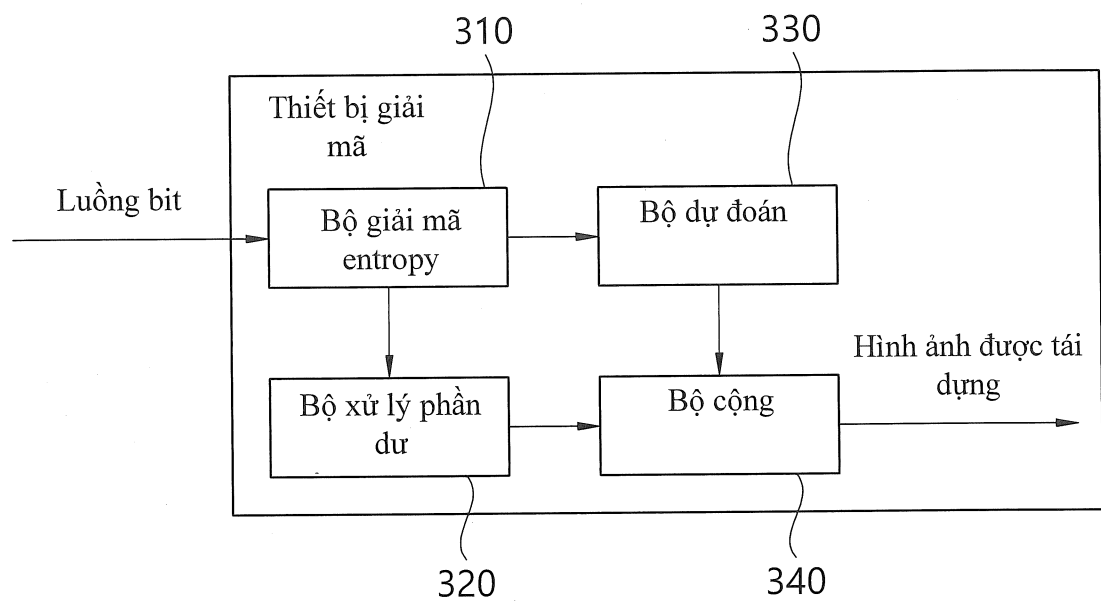


FIG. 8

