

# TR-1022

## オーディオビジュアルサービス全般 のための高度ビデオ符号化方式

(ADVANCED VIDEO CODING FOR GENERIC  
AUDIOVISUAL SERVICES)

第1版

2009年5月27日制定

社団法人  
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE



本書は、(社) 情報通信技術委員会が著作権を保有しています。

内容の一部又は全部を (社) 情報通信技術委員会の許諾を得ることなく複製、転載、  
改変、転用及びネットワーク上での送信、配布を行うことを禁止します。

## オーディオビジュアルサービス全般のための高度ビデオ符号化方式

## 要約

本標準は存在するビデオ符号化標準(JT-H261、JT-H262 と JT-H263)の進化に相当し、またビデオ会議、デジタル蓄積メディア、テレビ放送、インターネットストリーミング、および通信等様々なアプリケーションのための、動画像のより高い圧縮技法に対する要求の高まりに応えるために開発された。これは、多種多様なネットワーク環境において、柔軟性の高い方法で符号化ビデオの表現が可能となるよう設計されている。本標準の使用により、動画ビデオをコンピュータデータの形で取り扱うことができ、様々なストレージメディアへの保存、既存および将来のネットワーク上での送受信、および、既存および将来の放送チャネル上での配信が可能である。

本標準に対応する ITU-T 勧告において、ITU-T で 2005 年 3 月に承認された改定は、ビデオ品質能力を改良するために、また、標準によって焦点が当てられるアプリケーションの範囲を広げるために（例えば、より大きな範囲のピクチャサンプル精度や、より高い解像度を持つ色差フォーマットへの対応を含むことにより）、ハイ(High)、ハイ 10(High10)、ハイ 4:2:2(High4:2:2)、ハイ 4:4:4(High4:4:4)プロファイルと呼ばれる 4 つの新しいプロファイルを追加するビデオ符号化標準の修正を含んでいた。さらに、付加データの新しいタイプの定義は、ビデオ符号化標準の適用性をさらに広げるために規定された。最後に、発行された文書に含まれるいくつかの誤りの訂正も含まれていた。

ITU-T Rec.H.264 の訂正 1(Corrigendum 1)は、対応する共通開発され技術提携された文書 ISO/IEC14496-10 の新版として承認された 2005 年 4 月出力状態に対して、ITU-T 版の文書を新しい物にするために種々の細かな側面を訂正し更新した。さらに、それはいくつかの小さな誤りと明確化に対する必要性を修正し、そして 3 個の以前予約されていたサンプルアスペクト比指示子を定義した。

改正 1(Amendment 1) “追加色空間のサポートとハイ 4:4:4 プロファイルの削除”は、本標準に対し、追加の色空間を規定し、ハイ 4:4:4 プロファイルを削除する変更を含んでいた。

記 本標準が対応する ITU-T Rec.H.264 は、ISO/IEC 14496-10 との対の文書であるが、この改正は ISO/IEC シリーズでは 2 つの異なる文書として発行された：

- ハイ 4:4:4 プロファイルの削除は ISO/IEC 14496-10:2005/Cor.2 として見つかる。
- 追加色空間サポートの規定は ISO/IEC 14496-10:2005/Amd.1 として見つかるだろう。

改正 2 (Amendment 2) “専門的アプリケーションのための新プロファイル” は、主に専門的アプリケーションを意図する 5 つの追加プロファイル(ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、CAVLC 4:4:4 INTRA、ハイ 4:4:4 予測プロファイル)および 2 つの新しい型の付加拡張情報(SEI)メッセージ(ポストフィルタヒント SEI メッセージおよびトーンマッピング情報 SEI メッセージ)のサポートを規定する、本標準への拡張を含んでいた。

改正 3(Amendment 3) “スケーラブルビデオ符号化” は、3 つのプロファイル(スケーラブル基本、スケーラブルハイ、スケーラブルハイ INTRA プロファイル)によるスケーラブル符号化拡張を規定する、本標準への拡張を含んでいた。

本標準に対応する ITU-T 勧告における 2005 年 11 月発行の版は、ITU-T で 2005 年 3 月に承認された文書と 2005 年 9 月に承認されたその訂正 1(Corrigendum 1)を含んでいた。ITU-T Rec. H.264 (2005) Amd.2 (2007) は未発行文書としてのみ入手可能であったが、それは正式発行以前に Amd.3 (11/2007)により置換されたためである。また、ITU-T Rec. H.264 Amd.3 は分離して発行されなかった。この第 3 版は 2005 年 11 月の版に対して、改正 1(2006 年 6 月)、同 2(2007 年 4 月)、同 3(2007 年 11 月)により承認された全ての変更を統合する。

## 目 次

序文.....	14
0 序論.....	15
0.6.1 予測符号化.....	18
0.6.2 プログレッシブおよびインタレースビデオの符号化.....	18
0.6.3 マクロブロックおよびより小さいパーティションへのピクチャパーティション.....	19
0.6.4 空間冗長性削減.....	19
1 本標準の規定範囲.....	21
2 標準としての参照.....	21
3 定義.....	21
4 略語.....	32
5 規則.....	33
6 情報源、符号化、復号、および出力データフォーマット、走査処理、隣接関係.....	38
6.4.1 逆マクロブロック走査処理.....	45
6.4.2 逆マクロブロックパーティションおよびサブマクロブロックパーティション走査処理.....	46
6.4.3 逆4×4輝度ブロック走査処理.....	47
6.4.4 3に等しいChromaArrayTypeに対する逆4×4CbまたはCrブロック走査処理.....	48
6.4.5 逆8×8輝度ブロック走査処理.....	48
6.4.7 マクロブロックアドレスに対する利用可能性の導出処理.....	49
6.4.8 隣接マクロブロックアドレスおよびそれらの利用可能性の導出処理.....	49
6.4.9 MBAFFフレームにおける隣接マクロブロックアドレスおよびそれらの利用可能性の導出処理.....	50
6.4.10 隣接マクロブロック、ブロック、およびパーティションのための導出処理.....	51
6.4.11 隣接位置の導出処理.....	57
7 シンタックスおよび意味.....	61
7.3.2 未加工のバイトシーケンスペイロードおよびRBSP追従ビットシンタックス.....	65
7.4.1 NALユニットの意味.....	88
7.4.2 未加工のバイトシーケンスペイロードおよびRBSP追従ビットの意味.....	101
7.4.3 スライスヘッダの意味.....	119
7.4.4 スライスデータの意味.....	135
7.4.5 マクロブロックレイヤの意味.....	135
8 復号処理.....	151
8.2.1 ピクチャ順序カウントに対する復号処理.....	154
8.2.2 マクロブロックからスライスグループへのマップの復号処理.....	159

8.2.3	スライスデータパーティションの復号処理	164
8.2.4	参照ピクチャリスト構築の復号処理	165
8.2.5	復号参照ピクチャマーキング処理	174
8.3.1	輝度サンプルのIntra_4x4 予測処理	180
8.3.2	輝度サンプルのIntra_8x8 予測処理	187
8.3.3	輝度サンプルのIntra_16x16 予測処理	197
8.3.4	色差サンプルのINTRA予測処理	200
8.3.5	I_PCMマクロブロックに対するサンプル構築処理	205
8.4.1	動きベクトル成分と参照インデックスの導出処理	209
8.4.2	INTER予測サンプルの復号処理	224
8.4.3	予測重みの導出処理	235
8.5	変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理	237
8.5.1	4x4 輝度残差ブロックに対する変換復号処理の規定	238
8.5.2	Intra_16x16 マクロブロック予測モードの輝度サンプルに対する変換復号処理の規定	239
8.5.3	8x8 輝度残差ブロックに対する変換復号処理の規定	240
8.5.4	色差サンプルに対する変換復号処理の規定	241
8.5.5	3に等しいChromaArrayTypeの色差サンプルに対する変換復号処理の規定	243
8.5.6	変換係数に対する逆走査処理	243
8.5.7	8x8 変換係数に対する逆走査処理	244
8.5.8	色差量子化パラメータとスケーリング関数のための導出処理	247
8.5.9	Intra_16x16 マクロブロックタイプに対するDC変換係数のスケーリングと変換処理	250
8.5.10	色差DC変換係数に対するスケーリングと変換処理	251
8.5.11	残差4x4 ブロックに対するスケーリングと変換処理	253
8.5.12	残差8x8 ブロックに対するスケーリングと変換処理	256
8.5.13	デブロックフィルタ処理前のピクチャ構築処理	260
8.5.14	INTRA残差変換バイパス復号処理	262
8.6	SPスライス中のPマクロブロックまたはSIマクロブロックに対する復号処理	262
8.6.1	非スイッチングピクチャに対するSP復号処理	263
8.6.2	スイッチングピクチャに対するSPとSIスライスの復号処理	266
8.7	デブロックフィルタ処理	268
8.7.1	ブロックエッジにおけるフィルタ処理	274
8.7.2	水平または垂直ブロックエッジを交差するサンプルの組に対するフィルタ処理	276
9	構文解析処理	283

9.1	指数Golomb符号に対する構文解析処理.....	283
9.1.1	符号付き指数Golomb符号に対するマッピング処理.....	285
9.1.2	符号化ブロックパターンに対するマッピング処理.....	286
9.2	変換係数レベルに対するCAVLC構文解析処理.....	288
9.2.1	変換係数レベル総数および追従1に対する構文解析処理.....	289
9.2.2	レベル情報に対する構文解析処理.....	294
9.2.3	ラン情報に対する構文解析処理.....	296
9.2.4	レベルおよびラン情報の結合.....	299
9.3	スライスデータに対するCABAC構文解析処理.....	299
9.3.1	初期化処理.....	301
9.3.2	バイナリ化処理.....	325
9.3.3	復号処理フロー.....	337
9.3.4	算術符号化処理（参考）.....	363
A.1	ビデオ復号器能力の要件.....	370
A.2	プロファイル.....	371
A.2.1	基本プロファイル（Baseline）.....	371
A.2.2	メインプロファイル（Main）.....	371
A.2.3	拡張プロファイル（Extended）.....	372
A.2.4	ハイプロファイル（High）.....	372
A.2.5	ハイ10プロファイル（High 10）.....	373
A.2.6	ハイ4:2:2プロファイル（High 4:2:2）.....	374
A.2.7	ハイ4:4:4予測プロファイル（High 4:4:4 Predictive）.....	375
A.2.8	ハイ10 INTRAプロファイル（High 10 Intra）.....	375
A.2.9	ハイ4:2:2 INTRAプロファイル（High 4:2:2 Intra）.....	376
A.2.10	ハイ4:4:4 INTRAプロファイル（High 4:4:4 Intra）.....	377
A.2.11	CAVLC 4:4:4 INTRAプロファイル（CAVLC 4:4:4 Intra）.....	378
A.3	レベル.....	378
A.3.1	基本、メイン、および拡張プロファイルに対する共通のレベル限定.....	378
A.3.2	ハイ、ハイ10、ハイ4:2:2、ハイ4:4:4予測、ハイ10 INTRA、ハイ4:2:2 INTRA、ハイ4:4:4 INTRA、 およびCAVLC 4:4:4 INTRAプロファイルに対する共通のレベル限定.....	382
A.3.3	プロファイルに特有のレベル限定.....	383
A.3.4	フレームレートに対するレベル限定の影響（参考）.....	388
B.1	バイトストリームNALユニットのシンタックスと意味.....	392
B.1.1	バイトストリームNALユニットシンタックス.....	392

B.1.2	バイトストリームNALユニットの意味 .....	393
B.2	バイトストリームNALユニットの復号処理 .....	393
B.3	復号器のバイト整列の回復 (参考) .....	394
C.1	符号化ピクチャバッファ(CPB)の動作 .....	398
C.1.1	ビットストリームの到達タイミング .....	398
C.1.2	符号化ピクチャ除去のタイミング .....	400
C.2	復号ピクチャバッファ(DPB)の動作 .....	401
C.2.1	frame_numのギャップの復号と“非存在”フレームの保存 .....	401
C.2.2	ピクチャの復号と出力 .....	401
C.2.3	現ピクチャの挿入が起こりうる前のDPBからのピクチャ除去 .....	402
C.2.4	現復号ピクチャのマーキングと保存 .....	403
C.3	ビットストリームの適合 .....	404
C.4	復号器の適合 .....	405
C.4.1	出力順序DPBの動作 .....	407
C.4.2	frame_numのギャップの復号と“非存在”ピクチャの保存 .....	407
C.4.3	ピクチャの復号 .....	407
C.4.4	現ピクチャの挿入が起こりうる前のDPBからのピクチャ除去 .....	407
C.4.5	現復号ピクチャのマーキングと保存 .....	408
D.1	SEIペイロードシンタックス .....	412
D.2	SEIペイロードの意味 .....	425
D.2.1	バッファリング期間SEIメッセージの意味 .....	425
D.2.2	ピクチャタイミングSEIメッセージの意味 .....	425
D.2.3	パン・スキャン矩形SEIメッセージの意味 .....	431
D.2.4	フィルペイロードSEIメッセージの意味 .....	433
D.2.5	ITU-T勧告T.35による登録ユーザデータSEIメッセージの意味 .....	433
D.2.6	非登録ユーザデータSEIメッセージの意味 .....	433
D.2.7	復旧点SEIメッセージの意味 .....	433
D.2.8	復号参照ピクチャマーキング繰り返しSEIメッセージの意味 .....	435
D.2.9	予備ピクチャSEIメッセージの意味 .....	436
D.2.10	シーン情報SEIメッセージの意味 .....	438
D.2.11	サブシーケンス情報SEIメッセージの意味 .....	440
D.2.12	サブシーケンスレイヤ特性SEIメッセージの意味 .....	442
D.2.13	サブシーケンス特性SEIメッセージの意味 .....	444

D.2.14	フルフレーム凍結SEIメッセージの意味 .....	445
D.2.15	フルフレーム凍結解除SEIメッセージの意味 .....	446
D.2.16	フルフレームスナップショットSEIメッセージの意味.....	446
D.2.17	プログレッシブリファインメント区分開始SEIメッセージの意味.....	447
D.2.18	プログレッシブリファインメント区分終了SEIメッセージの意味.....	447
D.2.19	動き制約スライスグループセットSEIメッセージの意味.....	448
D.2.20	フィルムグレイン特性SEIメッセージの意味 .....	449
D.2.21	デブロックフィルタ表示選好SEIメッセージシンタックス.....	456
D.2.22	ステレオビデオ情報SEIメッセージの意味 .....	459
D.2.23	ポストフィルタヒントSEIメッセージの意味 .....	460
D.2.24	トーンマッピング情報SEIメッセージの意味 .....	461
D.2.25	予約SEIメッセージの意味.....	464
E.1	VUIシンタックス .....	465
E.2	VUIの意味.....	466
E.2.1	VUIパラメータの意味 .....	466
E.2.2	HRDパラメータの意味.....	483
G.6.1	参照レイヤマクロブロックの導出処理.....	492
G.6.1.1	フィールド参照レイヤマクロブロックからフレーム参照レイヤマクロブロックへの変換処理	494
G.6.1.2	フレーム参照レイヤマクロブロックからフィールド参照レイヤマクロブロックへの変換処理	495
G.6.2	参照レイヤパーティションのための導出処理.....	496
G.6.3	再サンプリングにおける参照レイヤサンプル位置に対する導出処理 .....	498
G.7.1	表形式におけるシンタックス規定の手法.....	500
G.7.2	シンタックス関数、カテゴリ、記述子の規定.....	500
G.7.3	表形式におけるシンタックス .....	501
G.7.3.1	NALユニットシンタックス .....	501
G.7.3.2	未加工のバイトシーケンスペイロードおよびRBSP追従ビットシンタックス.....	501
G.7.3.3	スライスヘッダシンタックス.....	504
G.7.3.4	スライスデータシンタックス.....	508
G.7.3.5	マクロブロックレイヤシンタックス.....	509
G.7.3.6	スケラブル拡張内マクロブロックレイヤシンタックス .....	510
G.7.4	意味.....	516
G.7.4.1	NALユニットの意味.....	516
G.7.4.2	未加工のバイトシーケンスペイロードおよびRBSP追従ビットの意味.....	528

G7.4.3	スライスヘッダの意味	533
G7.4.4	スライスデータの意味	547
G7.4.5	マクロブロックレイヤの意味	548
G7.4.6	スケーラブル拡張内マクロブロックレイヤの意味	549
G8	SVC復号処理	551
G8.1	SVC初期化および復号処理	553
G8.1.1	復号に要求されるレイヤ表現集合に対する導出処理	553
G8.1.2	配列の割り当て、初期化、および再構成処理	554
G8.1.3	レイヤ表現の復号処理	558
G8.1.4	スライス復号処理	562
G8.1.5	マクロブロック初期化および復号処理	563
G8.2	SVC参照ピクチャリスト構築と復号参照ピクチャマーキング処理	577
G8.2.1	ピクチャ順序カウントに対するSVC復号処理	579
G8.2.2	ピクチャ番号に対するSVC復号処理	580
G8.2.3	参照ピクチャリスト構築のSVC復号処理	582
G8.2.4	SVC 復号参照ピクチャマーキング処理	584
G8.2.5	frame_numのギャップのSVC復号処理	588
G8.3	SVC INTRA復号処理	588
G8.3.1	INTRA予測モードのためのSVC導出処理	588
G8.3.2	SVC INTRAサンプル予測および構築処理	593
G8.4	SVC INTER予測処理	600
G8.4.1	動きベクトル成分と参照インデックスのSVC導出処理	601
G8.4.2	INTER予測サンプルのSVC復号処理	609
G8.5	SVC変換係数復号処理とサンプル配列構築処理	616
G8.5.1	変換係数スケーリングおよび改善処理	616
G8.5.2	変換係数改善前の変換係数レベルスケーリング処理	626
G8.5.3	残差構築および累算処理	628
G8.5.4	サンプル配列の累算処理	634
G8.5.5	サンプル配列の再初期化処理	638
G8.6	予測データ、INTRAサンプル、および残差サンプルに対する再サンプリング処理	638
G8.6.1	マクロブロックタイプ、サブマクロブロックタイプ、参照インデックス、および動きベクトルに対するレイヤ間予測因子の導出処理	639
G8.6.2	INTRAサンプルの再サンプリング処理	651
G8.6.3	残差サンプルのための再サンプリング処理	664

G8.7	SVCデブロックフィルタ処理.....	672
G8.7.1	Intra_Base予測のためのデブロックフィルタ処理.....	672
G8.7.2	ターゲット表現のためのデブロックフィルタ処理.....	673
G8.7.3	デブロックフィルタで使用される量子化パラメータの導出処理.....	674
G8.7.4	マクロブロックデブロックフィルタ処理.....	675
G8.8	ビットストリームサブセットの規定.....	687
G8.8.1	サブビットストリーム抽出処理.....	688
G8.8.2	ベースレイヤビットストリームの規定.....	689
G9	構文解析処理.....	689
G9.1	符号化ブロックパターンに対する代替構文解析処理.....	690
G9.2	変換係数レベルに対する代替CAVLC構文解析処理.....	691
G9.2.1	変換係数レベル総数および追従 1 に対する追加構文解析処理.....	692
G9.2.2	ラン情報に対する代替構文解析処理.....	694
G9.3	スケーラブル拡張内スライスデータに対する代替CABAC構文解析処理.....	695
G9.3.1	初期化処理.....	695
G9.3.2	バイナリ化処理.....	697
G9.3.3	復号処理フロー.....	697
G10	プロファイルとレベル.....	699
G10.1	プロファイル.....	699
G10.1.1	スケーラブル基本プロファイル (Scalable Baseline).....	699
G10.1.2	スケーラブルハイプロファイル (Scalable High).....	701
G10.1.3	スケーラブルハイINTRAプロファイル (Scalable High Intra).....	702
G10.2	レベル.....	703
G10.2.1	スケーラブル基本、スケーラブルハイ、およびスケーラブルハイINTRAプロファイルに対する共通のレベル限定.....	703
G10.2.2	プロファイルに特有のレベル限定.....	706
G11	バイトストリームフォーマット.....	710
G12	仮想標準復号器.....	710
G13	付加拡張情報.....	710
G13.1	SEIペイロードシンタックス.....	710
G13.1.1	スケーラビリティ情報SEIメッセージシンタックス.....	710
G13.1.2	レイヤ非存在SEIメッセージシンタックス.....	713
G13.1.3	レイヤ依存変更SEIメッセージシンタックス.....	713
G13.1.4	サブピクチャスケラブルレイヤSEIメッセージシンタックス.....	714

G.13.1.5	非要求レイヤ表現SEIメッセージシンタックス .....	714
G.13.1.6	優先レイヤ情報SEIメッセージシンタックス .....	714
G.13.1.7	スケラブルネスティングSEIメッセージシンタックス .....	715
G.13.1.8	ベースレイヤ時間HRD SEIメッセージシンタックス .....	715
G.13.1.9	品質レイヤ整合性検査SEIメッセージシンタックス .....	716
G.13.1.10	冗長ピクチャプロパティSEIシンタックス .....	716
G.13.1.11	時間レベルゼロ依存表現インデックスSEIメッセージシンタックス .....	717
G.13.1.12	時間レベルスイッチング点SEIメッセージ .....	717
G.13.2	SEIペイロードの意味 .....	717
G.13.2.1	スケラビリティ情報SEIメッセージの意味 .....	719
G.13.2.2	レイヤ非存在SEIメッセージの意味 .....	731
G.13.2.3	レイヤ依存変更SEIメッセージの意味 .....	731
G.13.2.4	サブピクチャスケラブルレイヤSEIメッセージの意味 .....	732
G.13.2.5	非要求レイヤ表現SEIメッセージの意味 .....	732
G.13.2.6	優先レイヤ情報SEIメッセージの意味 .....	733
G.13.2.7	スケラブルネスティングSEIメッセージの意味 .....	734
G.13.2.8	ベースレイヤ時間HRD SEIメッセージの意味 .....	735
G.13.2.9	品質レイヤ整合性検査SEIメッセージの意味 .....	736
G.13.2.10	冗長ピクチャプロパティSEIメッセージの意味 .....	736
G.13.2.11	時間レベルゼロ依存表現インデックスSEIメッセージの意味 .....	737
G.13.2.12	時間レベルスイッチング点SEIメッセージの意味 .....	739
G.14.1	SVC VUIパラメータ拡張シンタックス .....	741
G.14.2	SVC VUIパラメータ拡張の意味 .....	741

## <参考>

### 1. 英文記述の適用レベル

適用レベル：なし

### 2. 国際標準等との関連

本技術レポートは、2007年11月22日にITU-T SG16により承認されたITU-T勧告H.264(11/07)の翻訳である。

ITU-T勧告およびISO標準の各版との詳しい関係は、本文の0.4節を参照されたい。

上記ITU-T勧告H.264(11/07)に対する訂正1(Corrigendum 1)が、2009年1月13日にITU-T SG16により承認され、既に出版されているが、本技術レポートはその内容を含んでいない。

### 3. 上記国際勧告に対する追加項目

#### 3.1 オプション選択項目

なし

#### 3.2 ナショナルマター項目

なし

#### 3.3 その他

- (1) 本技術レポートは、上記ITU-T勧告に対し、先行している項目はない。
- (2) 本技術レポートは、上記ITU-T勧告に対し、追加した項目はない。
- (3) 本技術レポートは、上記ITU-T勧告に対し、削除した項目はない。
- (4) 上記ITU-T勧告は、shallとmustを、各々、必須部分ではshall、必須部分以外(実際には“記”の内部)ではmustと使い分けている。しかし、本技術レポートでは日本語としてそれらを区別していない。“記”の内部が必須でない事が明記されている為、それらを区別していない事で混乱は発生しない。これらに関係した定義文(3章)を一部変更削除した。それ以外に変更した項目はない。

#### 3.4 原勧告との章立て構成の相違

なし

#### 4. 改訂の履歴

版 数	制 定 日	改 版 内 容
第 1 版	2009 年 5 月 27 日	制定

#### 5. 工業所有権

本技術レポートに関わる「工業所有権の実施の権利に係る確認書」の提出状況は、TTC ホームページでご覧になれます。

#### 6. その他

- (1) 本文書は参考資料として提供するものであり、正式には何も規定しない。ITU-T 勧告 H.264 に対応する TTC 標準は JT-H264（第 3 版より簡略標準）である。
- (2) 但し、本文書は ITU-T 勧告 H264 の対応する版が和文の TTC 標準として制定されたと仮定した場合にそう書かれるであろう表現によって書かれている。したがって、文章中に「本標準」や「TTC により予約」等の表現があるが、前項の通り、それらは正式には何も規定しない。

## 序文

国際電気通信連合 (ITU) は、国際連合の電気通信分野における専門機関である。この ITU の電気通信標準化セクタ (ITU-T) は、ITU の恒久的な機関である。ITU-T は世界的に電気通信を標準化する見地より、技術、運営および課金に関する問題の研究を行い、それらについての勧告を刊行する責任を負う。世界電気通信標準化総会 (WTSA) は、4 年に 1 度開催され、ITU-T の研究委員会による研究課題の設定を行い、研究委員会は順次これらの課題に関連した勧告を制定する。ITU-T 勧告の承認は、WTSA 決議 1 に付される手続きによって取り扱われる。ITU-T の範囲に入る、いくつかの情報技術分野においては、必要な標準は ISO および IEC と協調して作られる。

ISO (国際標準化機構) および IEC (国際電気標準化会議) は、世界的な標準化のための専門的な組織を形成する。ISO および IEC のメンバであるナショナルボディは、特定分野の技術的な活動を行うために各々の組織によって設立された技術委員会を通じて国際標準の開発に参加する。ISO および IEC の技術委員会は、共通の関心事の分野で協力している。一方その他の国際機関、政府機関、非政府機関もまた、ISO および IEC とのリエゾンとして、その作業に参加する。情報技術分野において ISO および IEC は、合同技術委員会 ISO/IEC JTC1 を設立した。また、合同技術委員会によって採択された国際標準草案は、投票のためナショナルボディに配布される。国際標準として発行するためには、投票したナショナルボディの最低 75 パーセントの承認が必要となる。

本標準は、VCEG (ビデオ符号化専門家グループ) として知られる ITU-T SG16 Q.6、そして MPEG (動画専門家グループ) として知られる ISO/IEC JTC1/SC29/WG11 によって、共同で作成された。VCEG は、既存の ITU-T ビデオ符号化標準の管理、また広範囲の会話型或いは非会話型サービスに適した新しいビデオ符号化標準の開発のため、1997 年に設立された。MPEG は、デジタル蓄積メディア、配信、通信など様々なアプリケーション向けの動画像や付随音声の符号化のための標準を制定するため、1988 年に設立された。

本標準において、付属資料 A から E および G は、標準としての要件を含み、本標準の必須部分である。

## オーディオビジュアルサービス全般のための高度ビデオ符号化方式

## 0 序論

この章は、本標準の必須部分を形成しない。

## 0.1 プロローグ

この節は、本標準の必須部分を形成しない。

演算処理能力とメモリの両方のコストが削減され、符号化ビデオデータに対するネットワークの対応が多様化し、またビデオ符号化技術が進歩してきたため、実質的に高められた符号化効率とネットワーク環境に対する強化された頑強性を有する圧縮ビデオ表現の業界標準が必要となってきた。このような目的に対し、ITU-T ビデオ符号化専門家グループ (VCEG) と ISO/IEC 動画像専門家グループ (MPEG) は、新しい勧告 | 国際標準の開発のため、2001年にジョイントビデオチーム (JVT) を編成した。

## 0.2 目的

この節は、本標準の必須部分を形成しない。

本標準は、ビデオ会議、デジタル蓄積メディア、テレビ放送、インターネットストリーミング、および通信等様々なアプリケーションのための、動画像のより高い圧縮技法に対する要求の高まりに応えるために開発された。これは、多種多様なネットワーク環境において、柔軟性の高い方法で符号化ビデオの表現が可能となるよう設計されている。本標準の使用は、動画ビデオのコンピュータデータの形での取り扱い、および、様々なストレージメディアへの保存、既存および将来のネットワーク上での送受信、および、既存および将来の放送チャンネル上での配信、を可能にする。

## 0.3 アプリケーション

この節は、本標準の必須部分を形成しない。

本標準は、以下を含むような、しかしこれらに限定されない、ビデオ内容のためのアプリケーションの広い範囲をカバーするよう設計されている。

CATV	Cable TV	光ネットワーク、銅線ケーブル等経由のケーブルテレビ
DBS	Direct broadcast satellite video services	直接放送衛星ビデオサービス
DSL	Digital subscriber line video services	デジタル加入者線ビデオサービス
DTTB	Digital terrestrial television broadcasting	デジタル地上テレビ放送
ISM	Interactive storage media	

インタラクティブ蓄積メディア(光ディスク等)

MMM Multimedia mailing

マルチメディアメール

MSPN Multimedia services over packet networks

パケットネットワーク上のマルチメディアサービス

RTC Real-time conversational services

リアルタイム会話サービス (ビデオ会議、テレビ電話等)

RVS Remote video surveillance

遠隔ビデオ監視

SSM Serial storage media

シリアル蓄積メディア (デジタルVTR等)

#### 0.4 本標準の出版と版

この節は、本標準の必須部分を形成しない。

本標準は ITU-T Video Coding Experts Group(VCEG)と ISO/IEC Moving Picture Experts Group によって共同で開発された規定に対応する。その規定は、ITU-T と ISO/IEC の両組織内で技術的に提携した対の文書として出版される。

ITU-T Rec.H.264|ISO/IEC14496-10 version 1 は、この ITU-T 勧告の最初に承認された版を参照する。version 1 は 2003 年 5 月 30 日に ITU-T により承認された。ISO/IEC の最初に出版された版は version 1 に対応する。

ITU-T Rec.H.264|ISO/IEC14496-10 version 2 は、最初の技術的な訂正で規定された補正を含む統合文書を参照する。TTC 標準 JT-H264 第 1 版はこの文書に対応する。2004 年 5 月 7 日に ITU-T により承認され、出版過程の間に訂正が作成されたため、ITU-T における最初の完全に出版された版は version 2 であった。version 2 も ISO/IEC との統合形式で出版された。

ITU-T Rec.H.264|ISO/IEC14496-10 version 3 は、最初の技術的な訂正(2004 年)と“フィデリティ範囲拡張”と呼ばれる最初の改正をとともに含む統合文書を参照する。version 3 は 2005 年 3 月 1 日に ITU-T により承認された。

ITU-T Rec.H.264 | ISO/IEC14496-10 version 4 は、最初の技術的な訂正(2004 年)と最初の改正(“フィデリティ範囲拡張”)と追加の技術的な訂正(2005 年)を含む統合文書を参照する。version 4 は ITU-T により 2005 年 9 月 13 日に承認された。ITU-T および ISO/IEC の双方において、version 2 の後の次の完全に出版された版は、version 4 であった。TTC 標準 JT-H264 第 2 版はこの文書に対応する。

ITU-T Rec.H.264 | ISO/IEC14496-10 version 5 は、ハイ 4:4:4 プロファイルの規定が削除されて統合された version 4 文書を参照する。

ITU-T Rec.H.264 | ISO/IEC14496-10 version 6 は、追加の色空間指示子をサポートするための改正の後の統合された version 5 文書を参照する。ITU-T では、version 5 および 6 の変更は 2006 年 6 月 13 日に承認され、

一つの改正として出版された。

ITU-T Rec.H.264 | ISO/IEC14496-10 version 7は、主に専門的アプリケーションを意図する5つの新プロファイル（ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、CAVLC 4:4:4 INTRA、そしてハイ 4:4:4 予測プロファイル）および2つの新しい型の付加拡張情報(SEI)メッセージ(ポストフィルタヒント SEI メッセージおよびトーンマッピング情報 SEI メッセージ)を定義するための改正の後の統合された version 6 文書を参照する。version 7はITU-Tにより2007年4月6日に承認された。

ITU-T Rec.H.264 | ISO/IEC14496-10 version 8は、3つのプロファイル(スケーラブル基本、スケーラブルハイ、スケーラブルハイ INTRA プロファイル)によるスケーラブルビデオ符号化を規定する改正の後の統合された version 7 文書を参照する。version 8はITU-Tにより2007年11月22日に承認された。TTC 標準 JT-H264の今回の版はこの文書に対応する。

## 0.5 プロファイルとレベル

この節は、本標準の必須部分を形成しない。

本標準は、広範囲にわたるアプリケーション、ビットレート、解像度、品質およびサービス等を提供するという意味において、汎用に設計されている。アプリケーションとして、とりわけ、デジタル蓄積メディア、テレビ放送およびリアルタイム通信をカバーすべきである。本標準の策定経過において、代表的アプリケーションからの多様な要件が考慮され、必要なアルゴリズム的要素が開発され、そして、それらが1つのシンタックスに統合されてきた。従って、本標準は異なるアプリケーションの間でのビデオデータの相互交換を容易にするだろう。

しかし、本標準の全てのシンタックスを実装することの現実性を考慮し、シンタックスの限られた数のサブセットが“プロファイル”と“レベル”によって設定される。これらや他の関連する用語は、第3章で正式に定義される。

“プロファイル”は本標準により規定されるビットストリームシンタックス全体のサブセットである。それでも、与えられたプロファイルのシンタックスにより課される範囲内で、ビットストリーム内のシンタックス要素の取る値によって、例えば復号ピクチャの規定サイズのように、符号器や復号器の性能に非常に大きな多様性を要求することがあり得る。多くのアプリケーションにおいて、現在のところは、特定のプロファイル内のシンタックスの仮説的使用法全てに対処できる復号器を実装することは、非実用的かつ不経済である。

この問題に対処するために、各プロファイルに“レベル”が規定される。レベルはビットストリーム内のシンタックス要素の値に対して課する制約の規定された組である。これらの制約は、値の単純な限定であることもある。あるいは、値の算術的組み合わせ（例：ピクチャ幅とピクチャ高の積と1秒間に復号されるピクチャ数の積）における制約の形をとることもある。

本標準に適合した符号化ビデオ内容は、共通のシンタックスを用いる。完全なシンタックスのサブセットを実現するために、その後のビットストリームに現れるシンタックス要素の存在または非存在を示すフラグ、パラメータ、およびその他のシンタックス的な要素がビットストリームに含まれている。

## 0.6 設計特性概要

この節は、本標準の必須部分を形成しない。

シンタックスで規定された符号化表現は、望まれる画像品質に対する高い圧縮能力を可能とする様に設計

されている。ハイ 4:4:4 INTRA、CAVLC 4:4:4 INTRA、およびハイ 4:4:4 予測プロファイルのロスレス符号化における変換バイパスモード操作と全てのプロファイルの I\_PCM モード操作を除き、アルゴリズムは、一般的に正確な原サンプル値が符号化および復号処理を通じて維持されないため、一般的にロスレスではない。高効率の圧縮を達成するために多くの技術を使用することができる。符号化アルゴリズム（本標準では規定されない）は、各ピクチャのブロック形状の領域で INTER と INTRA 符号化を選択することができる。INTER 符号化では、異なるピクチャ間の時間方向の統計的依存性を利用するため、ブロックベース INTER 予測の動きベクトルを用いる。INTRA 符号化では、単一のピクチャに対する原信号の空間方向の統計的依存性を利用するため、様々な空間予測モードを用いる。動きベクトルと INTRA 予測モードは、ピクチャにおける様々なブロックサイズに対して指定することができる。次に、変換ブロック内部の空間方向の相関を取り除くため、変換を用いて予測残差はさらに圧縮される。続いて、それに対して、原サンプルの良い近似を形成しつつ、一般的により重要でない視覚情報を削除する非可逆処理である量子化が実施される。最後に、動きベクトルや INTRA 予測モードは、量子化された変換係数の情報と組み合わせられ、可変長符号または算術符号化のいずれかを用いて符号化される。

スケーラブルビデオ符号化は、本標準の付属資料 G に規定され、本標準に適合するサブビットストリームを含むビットストリームの構築を可能にする。時間ビットストリームスケーラビリティ、すなわちビットストリームより小さい時間的サンプリングレートのサブビットストリームの存在、においては、サブビットストリームが導出される時、アクセスユニット全体がビットストリームから除去される。この場合、ビットストリームの高レベルシンタックスと INTER 予測参照ピクチャはそれに応じて構築される。空間および品質ビットストリームスケーラビリティ、すなわちビットストリームより低い空間解像度あるいは品質のサブビットストリームの存在、においては、サブビットストリームが導出される時、NAL ユニットがビットストリームから除去される。この場合、効率的符号化のために、レイヤ間予測、すなわち高位の解像度または品質の信号の低位の解像度または品質の信号のデータからの予測、が典型的には用いられる。

それ以外の場合、上の段落に示したような符号化アルゴリズムが用いられる。

### 0.6.1 予測符号化

この小節は、本標準の必須部分を形成しない。

ランダムアクセスと高効率圧縮の要件は相反するため、2 種類の主な符号化タイプが規定される。INTRA 符号化は、他のピクチャを参照しないで行われる。INTRA 符号化は、符号化シーケンスに対して正しく復号を開始したり再開したりするためのアクセス点の提供を可能にするが、一般には、並みの圧縮効率でしかない。INTER 符号化（予測符号化、双予測符号化）は、符号器によって選択された過去のいくつかの復号ピクチャの中から、各ブロックのサンプル値の INTER 予測を用いて、より効率良く符号化される。他のいくつかのビデオ符号化標準とは対照的に、双予測式 INTER 予測を使って符号化されたピクチャが、他のピクチャの INTER 予測の参照として使われても良い。

シーケンス中のピクチャに対する 3 つの符号化タイプの適用は柔軟であり、また、復号処理の順序は、符号器の原ピクチャのキャプチャ処理順序や表示のための復号器からの出力順序とは一般的に同じではない。これらの選択は符号器に任されており、アプリケーションの要件に依存することになるだろう。復号順序は、INTER ピクチャ予測を用いるピクチャの復号が復号処理で参照される他のピクチャよりも復号順序で後に続く様に規定される。

### 0.6.2 プログレッシブおよびインタレースビデオの符号化

この小節は、本標準の必須部分を形成しない。

本標準では、元がプログレッシブ走査あるいはインタレース走査のいずれかの形式であるビデオに対するシンタックスと復号処理を規定しており、これらは同じシーケンス中に混合しても良い。インタレースフレーム中の2つのフィールドはキャプチャ時刻が分かれており、一方、プログレッシブフレーム中の2つのフィールドは同じキャプチャ時刻を共有する。各々のフィールドは別々に符号化されてもよいし、フレームとして2つのフィールドが一緒に符号化されてもよい。プログレッシブフレームは、一般的にフレームとして符号化される。インタレースビデオでは、符号器はフレーム符号化およびフィールド符号化を選択できる。フレーム符号化またはフィールド符号化は、ピクチャ単位、および、符号化フレーム内のさらに局所的な単位で、適応的に選択することができる。フレーム符号化は一般的に、ビデオシーンの動きが限定されていて多くの精細部分を含んでいる時に選ばれる。フィールド符号化は一般的に、ピクチャ間で速い動きがある場合により適している。

### 0.6.3 マクロブロックおよびより小さいパーティションへのピクチャパーティション

この小節は、本標準の必須部分を形成しない。

以前のビデオ符号化の標準、勧告および国際標準のように、マクロブロックは、輝度サンプルの  $16 \times 16$  ブロックと対応する2つの色差サンプルブロックで構成され、ビデオ復号処理の基本処理単位として使用される。

マクロブロックは、INTER 予測のためにさらにパーティションすることができる。INTER 予測パーティションサイズの選択は、より小さいブロックの動き補償を用いることにより得られる符号化利得と、その動き補償のデータを表すのに必要なデータ量とのトレードオフの結果となる。本標準において、INTER 予測処理は、 $4 \times 4$  輝度サンプルの小さいサイズで動き表現のための分割を形成することが可能であり、輝度サンプルグリッド間隔変位の4分の1の動きベクトル精度を用いる。サンプルブロックのINTER 予測処理はまた、数多くの蓄積された復号済みピクチャから、参照ピクチャとして使われるピクチャを選択することもできる。動きベクトルは、符号化された近傍の動きベクトルから形成された予測値に対する差分として符号化される。

一般的に、符号器は、適切な動きベクトルやその他のビデオデータストリーム中に表現されるデータ要素を計算する。符号器のこの動き推定処理や、ビデオ内容の各領域を表現するために INTER 予測を用いるか否かの選択は、本標準では規定されない。

### 0.6.4 空間冗長性削減

この小節は、本標準の必須部分を形成しない。

原ピクチャおよび予測残差はどちらも高い空間的冗長性を持っている。本標準では、空間的冗長性除去のため、ブロックベース変換手法の使用を基本とする。他のピクチャの復号済みサンプルからの INTER 予測や、現ピクチャの復号済みサンプルからの空間ベース予測の後、その結果である予測残差は  $4 \times 4$  のブロックに分けられる。これらは変換領域に変換され、そこで量子化される。量子化後、多くの変換係数はゼロまたは低い振幅となり、そのため少ない量の符号化データで表現できる。符号器における変換や量子化の処理については、本標準では規定されない。

## 0.7 本標準の読み方

この節は、本標準の必須部分を形成しない。

読者は始めに第1章（本標準の規定範囲）を読み、続けて第3章（定義）へ移ることを提案する。第6章は、復号器の情報源、入力、出力の幾何学的な関係を理解するために読むべきである。第7章（シンタック

スおよび意味)は、ビットストリームからシンタックス要素を解析する順序を規定する。シンタックスの順序は、7.1-7.3節を参照し、意味、すなわちシンタックス要素に課せられる有効範囲、制限、条件については、7.4節を参照せよ。ほとんどのシンタックス要素の実際の構文解析は、第9章(構文解析処理)にて規定される。最後に第8章(復号処理)は、シンタックス要素がどのように復号サンプルにマップされるかを規定する。本標準を読む際、必要に応じて、第2章(標準としての参照)、第4章(略語)、第5章(規則)を参照されたい。付属資料Aから付属資料Eまでは、本標準の必須部分を形成する。

本標準の付属資料Aでは、各々一定のアプリケーション領域に適合するように作られた11のプロファイル(基本(Baseline)、メイン(Main)、拡張(Extended)、ハイ(High)、ハイ10(High10)、ハイ4:2:2(High4:2:2)、ハイ4:4:4予測(High4:4:4 Predictive)、ハイ10 INTRA(High 10 Intra)、ハイ4:2:2 INTRA(High 4:2:2 Intra)、ハイ4:4:4 INTRA(High 4:4:4 Intra)、およびCAVLC 4:4:4 INTRA(CAVLC 4:4:4 Intra))を規定し、そして、プロファイルのいわゆるレベルを定義する。付属資料Bでは、順序付けられたバイトのストリームとして符号化ビデオを配送するための、バイトストリームフォーマットのシンタックスと意味を規定する。付属資料Cでは、仮想標準復号器と、ビットストリームと復号器の適合性を検査するためのその使用方法を規定する。付属資料Dでは、付加拡張情報メッセージペイロードのためのシンタックスと意味を規定する。最後に、付属資料Eでは、シーケンスパラメータセットのビデオ有用性情報パラメータのシンタックスと意味を規定する。

付属資料Gは、スケーラブルビデオ符号化(SVC)を規定する。読者はSVCの全復号処理について付属資料Gを参照すること。それはそこで2-9章および付属資料A-Eへの参照により規定される。G.10節はSVCのための3つのプロファイル(スケーラブル基本(Scalable Baseline)、スケーラブルハイ(Scalable High)、およびスケーラブルハイ INTRA(Scalable High Intra))を規定する。

本標準を通じて、“記”の前置き表記のあるステートメントは、参考としての取り扱いであり、本標準において必須部分ではない。

## 1 本標準の規定範囲

本標準は、TTC 標準 JT-H264 のビデオ符号化方式を規定するものである。

## 2 標準としての参照

以下に示す勧告および国際標準は、本文書中の参照を通じて、本標準の規定を構成する条項を含んでいる。発刊時においては、示された版は有効であった。全ての勧告および標準は改定されることがあり、本標準に基づく規定に関連して下記勧告および標準の最新版の適用が可能かを調べることが望ましい。IEC および ISO のメンバは現在有効な国際標準の登録を維持している。ITU の電気通信標準化事務局は、現在有効な ITU-T 勧告のリストを維持している。

- ITU-T勧告T.35(2000),*Procedure for the allocation of ITU-T defined codes for non-standard facilities.*
- ISO/IEC 11578:1996,Annex A,*Universal Unique Identifier.*
- ISO/CIE 10527:1991,*Colorimetric Observers.*

## 3 定義

本標準の目的のために以下の定義が適用される。

- 3.1 **アクセスユニット**：ちょうど1つの主符号化ピクチャを常に含む1組のNALユニット。主符号化ピクチャに加えて、1つのアクセスユニットは、1つ以上の冗長符号化ピクチャ、1つの補助符号化ピクチャ、または、符号化ピクチャのスライスやスライスデータパーティションを含んでいない他のNALユニットを含んでもよい。1つのアクセスユニットの復号により、常に1つの復号ピクチャが生成される。
- 3.2 **AC 変換係数**：片方、または、両方の次元の周波数インデックスが非ゼロである、任意の変換係数。
- 3.3 **適応バイナリ算術復号処理**：適応バイナリ算術符号化処理により生成されるビットストリームから(複数)bin値を導出するエントロピ復号処理。
- 3.4 **適応バイナリ算術符号化処理**：本標準で必須項目として規定されない、(複数)binのシーケンスを符号化し適応バイナリ算術復号処理を使用して復号できるビットストリームを生成するエントロピ符号化処理。
- 3.5 **アルファブレンディング**：表示処理で、補助符号化ピクチャが主符号化ピクチャと本標準で規定されない他のデータと組み合わせて使用される本標準で規定されない処理。アルファブレンディング処理では、補助符号化ピクチャのサンプルは、主符号化ピクチャの対応する輝度サンプルに関連付けられた不透明度の次元（または、等価に、透明度の次元）の指示として解釈される。
- 3.6 **任意スライス順序(ASO)**：スライスグループのあるスライスの先頭マクロブロックのマクロブロックアドレスが、同じスライスグループの他のある先行するスライスの、もしくは3つの分離された色プレーンを用いて符号化されたピクチャの場合は同じ色プレーン内の同じスライスグループの他のある先行するスライスの、先頭マクロブロックのマクロブロックアドレスよりも小さくてもよく、あるいは、ピクチャのスライスグループのスライスが、ピクチャの他の1つ以上のスライスグループの中のスライスと、もしくは3つの分離された色プレーンを用いて符号化されたピクチャの場合は同じ色プレーン内の他の1つ以上のスライスグループの中のスライスと、インターリーブされてもよい、スライスの復号順序。

- 3.7 **補助符号化ピクチャ**：表示処理で、本標準で規定されない他のデータと組み合わせて使用されるかも知れない**主符号化ピクチャ**を補うピクチャ。補助符号化ピクチャは、モノクローム冗長ピクチャと同じシンタックスと意味に関する制限をもつ。補助符号化ピクチャは、**主符号化ピクチャ**と同じマクロブロック数を含まなければならない。補助符号化ピクチャは、**復号処理**に必須な影響を与えない。**主符号化ピクチャ**と**冗長符号化ピクチャ**も見よ。
- 3.8 **B スライス**：各ブロックのサンプル値を予測するために、*INTRA* 予測を用いるか、最大 2 つの動きベクトルと**参照インデックス**を用いる *INTER* 予測を用いて、復号してもよいスライス。
- 3.9 **bin**：*bin* 記号列の 1 ビット。
- 3.10 **バイナリ化**：シンタックス要素の全ての取りうる値に対する *bin* 記号列集合。
- 3.11 **バイナリ化処理**：*bin* 記号列集合へのシンタックス要素の全てのとりうる値の一意マッピング処理。
- 3.12 **bin 記号列**：(複数) *bin* の記号列。*bin* 記号列は、シンタックス要素のバイナリ化からのシンタックス要素値の中間バイナリ表現。
- 3.13 **双予測スライス**：*B* スライス参照。
- 3.14 **ビットストリーム**：1 つ以上の符号化ビデオシーケンスを形成する符号化ピクチャと関連するデータの表現を形成するビットの並び。ビットストリームは、*NAL* ユニットストリームまたはバイトストリームのいずれかを呼ぶのに使用される総称。
- 3.15 **ブロック**：サンプルの  $M \times N$  ( $N$  行  $M$  列)配列、または、**変換係数**の  $M \times N$  配列。
- 3.16 **ボトムフィールド**：フレームを構成する 2 つのフィールドのうちの片方。ボトムフィールドの各行は、空間的にトップフィールドの対応する行のすぐ下に位置する。
- 3.17 **(マクロブロックペアの) ボトムマクロブロック**：マクロブロックペアに対するサンプルのボトム行のサンプルを含むマクロブロックペアの中のマクロブロック。フィールドマクロブロックペアでは、ボトムマクロブロックは、マクロブロックペアの空間領域に位置するフレームのボトムフィールド領域からのサンプルを表す。フレームマクロブロックペアでは、ボトムマクロブロックは、マクロブロックペアの空間領域の下半分に位置するフレームのサンプルを表す。
- 3.18 **ブロークンリンク**：ビットストリーム生成時に行われた無規定の操作により、**復号順序**で連続するいくつかのピクチャが、重大な視覚上の劣化を含むかもしれないことを示すビットストリーム中の個所。
- 3.19 **バイト**：8 ビットの並びであり、左側が最上位ビットで、右側が最下位ビットとして読み書きされる。データビットの並びにおいて表現される時は、バイトの最上位ビットが始めとなる。
- 3.20 **バイト整列**：ビットストリームのある位置が、ビットストリームの最初のビット位置から数えて、8 ビットの整数倍に位置している場合、その位置はバイト整列となる。ビットまたはバイトまたはシンタックス要素は、それがビットストリームに現れる位置がバイト整列の時、バイト整列と言われる。
- 3.21 **バイトストリーム**：スタートコードプリフィックスと *NAL* ユニットを含む *NAL* ユニットストリームをカプセル化したもので、**付属資料 B** で規定される。
- 3.22 **～できる、～し得る (または類似な表現)**：許されるが必ずしも要求されない動作に言及するために使用される用語。
- 3.23 **カテゴリ**：各シンタックス要素に関連する数。カテゴリは、**スライスデータパーティション**に対し、シンタックス要素の *NAL* ユニットへの割り付けを規定するために使用される。加えて、本標準で規定されない方法によるシンタックス要素のクラスを参照するために、アプリケーションによって決定される方法で使用されても良い。
- 3.24 **色差**：サンプル配列または、1 つのサンプルが、原色に関連する 2 つの色差信号のいずれか 1 つを表すことを規定する形容詞的な用語。色差配列またはサンプルに対して使われる記号は  $C_b$  と

Crである。

記 クロミナンスは、しばしばリニア光変換特性と関連づけられ、これを避けるため、用語として“クロミナンス”ではなく“色差(クロマ)”が使用される。

- 3.25 **符号化フィールド**：フィールドの符号化表現。
- 3.26 **符号化フレーム**：フレームの符号化表現。
- 3.27 **符号化ピクチャ**：ピクチャの符号化表現。符号化ピクチャは、符号化フィールドまたは、符号化フレームのいずれでもよい。符号化ピクチャは、主符号化ピクチャまたは、冗長符号化ピクチャのいずれかを呼ぶ総称であるが、両方同時には呼ばない。
- 3.28 **符号化ピクチャバッファ (CPB)**：付属資料 C の仮想標準復号器で規定される、復号順序でアクセスユニットを含む先入れ先出しバッファ。
- 3.29 **符号化表現**：符号化形式で表現されるデータ要素。
- 3.30 **符号化スライスデータパーティションNALユニット**：スライスデータパーティションを含むNALユニット
- 3.31 **符号化スライスNALユニット**：補助符号化ピクチャのスライスではないスライスを含むNALユニット
- 3.32 **符号化ビデオシーケンス**：復号順序で、あるIDRアクセスユニットとそれに続く、後続するIDRアクセスユニットを含まずそれまでの後続する全てのアクセスユニットを含む、0個以上の非IDRアクセスユニットから成るアクセスユニットの並び。
- 3.33 **成分**：4:2:0、4:2:2、または4:4:4色フォーマットにおけるフィールドやフレームを構成する3つの配列（輝度と2つの色差）の1つからの配列または1つのサンプル、あるいは、モノクロームフォーマットにおけるフィールドやフレームを構成する配列または1つのサンプル。
- 3.34 **相補的フィールドペア**：相補的参照フィールドペアまたは相補的非参照フィールドペアに対する総称。
- 3.35 **相補的非参照フィールドペア**：復号順序で連続するアクセスユニット中の2つの非参照フィールドであって、第1フィールドがすでにペアのフィールドでない逆パリティの2つの符号化フィールド。
- 3.36 **相補的参照フィールドペア**：2つの符号化フィールドとして、復号順序で連続するアクセスユニット中の2つの参照フィールドであり、それらは同じframe\_numシンタックス要素の値を持つ。ここで復号順序で2番目のフィールドはIDRピクチャでなく、5に等しいmemory\_management\_control\_operationシンタックス要素を含まない。
- 3.37 **コンテキスト変数**：最近復号した（複数）binを含む等式による、binの適応バイナリ算術復号処理のために規定される変数。
- 3.38 **DC変換係数**：全ての次元で周波数インデックスがゼロである変換係数。
- 3.39 **復号ピクチャ**：復号ピクチャは、符号化ピクチャを復号して得られる。復号ピクチャは、復号フレームまたは、復号フィールドのいずれかである。また復号フィールドは、復号トップフィールド、または、復号ボトムフィールドのいずれかである。
- 3.40 **復号ピクチャバッファ (DPB)**：付属資料 C の仮想標準復号器のために規定される、参照や出力の並び替え、出力遅延のために復号ピクチャを保持するバッファ。
- 3.41 **復号器**：復号処理を行う実体。
- 3.42 **復号順序**：復号処理によってシンタックス要素が処理される順番。
- 3.43 **復号処理**：本標準で規定される、ビットストリームを読み出し、復号ピクチャをそれから導出する処理。
- 3.44 **直接予測**：動きベクトルの復号されないブロックのINTER予測。空間方向直接予測と時間方向直

接予測モードと呼ばれる2つの直接予測モードが規定される。

- 3.45 **表示処理**：復号処理の出力である切り出された復号ピクチャをその入力とする本標準で規定されない処理。
- 3.46 **試験される復号器 (DUT)**：本標準への適合を試験される復号器。その試験は、仮想ストリームスケジューラを操作し復号器と仮想標準復号器に適合ビットストリームの配達を行い、それら2つの復号器の出力からの値やタイミングを比較することにより行う。
- 3.47 **エミュレーション防止バイト**：NAL ユニット内に存在するかも知れない値 0x03 のバイト。エミュレーション防止バイトの存在は、NAL ユニットにおいて、スタートコードプリフィックスを含む連続するバイト整列されたバイトのシーケンスがないことを保証するものである。
- 3.48 **符号器**：符号化処理を行う実体。
- 3.49 **符号化処理**：本標準に適合するビットストリームを生成する処理。処理そのものは、本標準では規定されない。
- 3.50 **フィールド**：フレーム上の交互の行を集めたもの。フレームは、2つのフィールド、すなわちトップフィールドとボトムフィールドで構成される。
- 3.51 **フィールドマクロブロック**：1つのフィールドからのサンプルを含むマクロブロック。符号化フィールドの全てのマクロブロックは、フィールドマクロブロックである。マクロブロック適応フレーム/フィールド復号が使用される場合、符号化フレームのいくつかのマクロブロックは、フィールドマクロブロックであってもよい。
- 3.52 **フィールドマクロブロックペア**：2つのフィールドマクロブロックとして復号されるマクロブロックペア。
- 3.53 **フィールド走査**：変換係数の特定のシーケンシャルな順序。それは列を行より早めに走査する点がジグザグ走査と異なる。フィールド走査は、フィールドマクロブロックにおける変換係数のために使用される。
- 3.54 **フラグ**：2つの値 0、1 のいずれかをとりうる変数。
- 3.55 **フレーム**：フレームは、モノクロームフォーマットにおいては1つの輝度サンプルの配列を、あるいは 4:2:0、4:2:2、または 4:4:4 色フォーマットにおいては1つの輝度サンプルの配列と対応する2つの色差サンプルの配列を、含む。1つのフレームは、2つのフィールド、すなわちトップフィールドとボトムフィールドからなる。
- 3.56 **フレームマクロブロック**：符号化フレームの2つのフィールドのサンプルを表現するマクロブロック。マクロブロック適応フレーム/フィールド復号が使用されない場合、符号化フレームの全てのマクロブロックは、フレームマクロブロックになる。マクロブロック適応フレーム/フィールド復号が使用される場合、符号化フレームのいくつかのマクロブロックは、フレームマクロブロックであってもよい。
- 3.57 **フレームマクロブロックペア**：2つのフレームマクロブロックとして復号されるマクロブロックペア。
- 3.58 **周波数インデックス**：復号処理の逆変換部前の変換係数に関連する、1次元または、2次元のインデックス。
- 3.59 **仮想標準復号器 (HRD)**：符号化処理が生成するかも知れない、適合する NAL ユニットストリームや、適合するバイトストリームの変動に対する制約を規定した仮想の復号器モデル。
- 3.60 **仮想ストリームスケジューラ (HSS)**：仮想標準復号器へのビットストリーム入力のタイミングや、データフローのための仮想配達機構。HSS はビットストリームや復号器の適合の確認のために使用される。
- 3.61 **I スライス**：INTRA 予測のみを使用して復号される SI スライスではないスライス。

- 3.62 **参考**：本標準で与えられる本標準の必須部分でない内容に言及するために使用される用語。参考の内容は本標準への適合に対するいかなる必須要件も確立させない。
- 3.63 **瞬時復号更新 (IDR) アクセスユニット**：主符号化ピクチャが IDR ピクチャであるアクセスユニット。
- 3.64 **瞬時復号更新 (IDR) ピクチャ**：IDR ピクチャを復号直後に、復号処理が全ての参照ピクチャを“参照不使用”としてマークすることになる、全てのスライスが I または SI スライスである符号化ピクチャ。IDR ピクチャ復号後、復号順序に続く全ての符号化ピクチャは、IDR ピクチャの前に復号されたどんなピクチャからの INTER 予測なしで復号が可能である。各符号化ビデオシーケンスの最初のピクチャは IDR ピクチャである。
- 3.65 **INTER 符号化**：INTER 予測を使用するブロック、マクロブロック、スライス、または、ピクチャの符号化。
- 3.66 **INTER 予測**：現復号ピクチャ以外の参照ピクチャの復号サンプルから導出される予測。
- 3.67 **解釈サンプル値**：表示処理での使用のために生成されるかもしれない、補助符号化ピクチャの復号サンプル値に対応する変化した可能性のある値。解釈サンプル値は復号処理で使用されず、復号処理に必須な影響を与えない。
- 3.68 **INTRA 符号化**：INTRA 予測を用いるブロック、マクロブロック、スライス、または、ピクチャの符号化。
- 3.69 **INTRA 予測**：同一復号スライスの復号サンプルから得られる予測。
- 3.70 **INTRA スライス**：I スライス参照。
- 3.71 **逆変換**：復号処理の一部で、これにより変換係数の組が空間領域の値に変換されたり、変換係数の組が DC 変換係数に変換されたりする。
- 3.72 **レイヤ**：分岐を行わない階層関係を持ったシンタックス的な構造の集合の 1 つ。上位レイヤは、下位レイヤを含む。符号化のレイヤは、符号化ビデオシーケンス、ピクチャ、スライス、そして、マクロブロックレイヤである。
- 3.73 **レベル**：本標準のシンタックス要素や変数がとりうる値を制約するように定義された組である。全てのプロファイルに対して同じレベルの集合が定義されており、各レベルの定義のほとんどの様相は、異なるプロファイル間で共通となっている。個々の実装では規定された制約内で、対応するプロファイル毎に、異なったレベルをサポートしてもよい。異なる文脈では、レベルは、スケリングする前の変換係数の値を意味する。
- 3.74 **list0(list1)動きベクトル**：参照ピクチャリスト 0(リスト 1)内を指す参照インデックスに関連する動きベクトル。
- 3.75 **list0(list1)予測**：参照ピクチャリスト 0(リスト 1)内を指す参照インデックスを使用したスライスの内容の INTER 予測。
- 3.76 **輝度**：サンプル配列または、1 つのサンプルが、原色に関連するモノクロームの信号を表すことを規定する形容詞的な用語。輝度に対して使われる記号または下付き添字は Y または L である。  
記 ルミナンスは、しばしばリニア光変換特性と関連づけられ、これを避けるため、用語として“ルミナンス”ではなく“輝度(ルマ)”が使用される。シンボル L は、垂直位置に使用されるシンボル y との混乱を避けるため、シンボル Y の代わりとして時々使用される。
- 3.77 **マクロブロック**：3 つのサンプル配列を持つピクチャの輝度サンプルの 16×16 ブロックと 2 つの対応する色差サンプルのブロック、あるいはモノクロームピクチャまたは 3 つの分離された色プレーンを用いて符号化されたピクチャのサンプルの 16×16 ブロック。スライスまたはマクロブロックペアのマクロブロックへの分割は、パーティションである。
- 3.78 **マクロブロック適応フレーム/フィールド復号**：いくつかのマクロブロックはフレームマクロブロックとして復号されてよく、他はフィールドマクロブロックとして復号されてよい、符号化フ

レームの復号処理。

- 3.79 **マクロブロックアドレス**：マクロブロック適応フレーム／フィールド復号が使用されない場合、マクロブロックアドレスは、ピクチャのマクロブロックラスタ走査でのマクロブロックのインデックスであり、ピクチャの左上のマクロブロックに対するゼロから始まる。マクロブロック適応フレーム／フィールド復号が使用される場合、マクロブロックペアのトップマクロブロックのマクロブロックアドレスは、ピクチャのマクロブロックペアラスタ走査でのマクロブロックペアのインデックスの2倍である。そして、マクロブロックペアのボトムマクロブロックのマクロブロックアドレスは、対応するトップマクロブロックのマクロブロックアドレスに1を加算したものである。各マクロブロックペアのトップマクロブロックのマクロブロックアドレスは偶数であり、各マクロブロックペアのボトムマクロブロックのマクロブロックアドレスは奇数である。
- 3.80 **マクロブロック位置**：(x,y) で表記するピクチャ中のマクロブロック2次元座標。ピクチャの最も左上のマクロブロックは、(x,y) = (0,0) となる。マクロブロック列が左から右に向かう毎、xは1ずつ増加する。マクロブロック適応フレーム／フィールド復号が使用されない場合、各マクロブロック行が上から下へ向かう毎に、yは1ずつ増加する。マクロブロック適応フレーム／フィールド復号が使用される場合、各マクロブロックペア行が上から下へ向かう毎に、yは2ずつ増加し、マクロブロックがボトムマクロブロックの場合には更に1増加する。
- 3.81 **マクロブロックペア**：フレーム内の垂直に隣接するマクロブロックのペアで、マクロブロック適応フレーム／フィールド復号において、組み合わせて使用される。スライスのマクロブロックペアへの分割は、パーティションである。
- 3.82 **マクロブロックパーティション**：3つのサンプル配列を持つピクチャに対する *INTER* 予測のためのマクロブロックのパーティションの結果となる輝度サンプルと対応する2つの色差サンプルのブロック、あるいは、モノクロームピクチャまたは3つの分離された色プレーンを用いて符号化されたピクチャに対する *INTER* 予測のためのマクロブロックのパーティションの結果となる輝度サンプルのブロック。
- 3.83 **マクロブロックからスライスグループへのマップ**：ピクチャのマクロブロックからスライスグループへのマッピングの手段。マクロブロックからスライスグループへのマップは、各符号化マクロブロックに対して1つ存在する数値のリストである。それは、各符号化マクロブロックが属するスライスグループを規定する。
- 3.84 **マップユニットからスライスグループへのマップ**：ピクチャのスライスグループマップユニットからスライスグループへのマッピングの手段。マップユニットからスライスグループへのマップは、各スライスグループマップユニットに対して1つ存在する数値のリストである。それは、各符号化スライスグループマップユニットが属するスライスグループを規定する。
- 3.85 **～してもよい、～かもしれない (または類似な表現)**：許されるが必ずしも要求されない動作に言及するために使用される用語。記述される動作のオプション性が強調されることを意図するいくつかの場所では、言い回し「してもよいし、または、しなくてもよい (または類似な表現)」が、強調を与えるため使用される。
- 3.86 **メモリ管理制御操作**：参照ピクチャマーキングを制御する7つの操作。
- 3.87 **動きベクトル**：復号ピクチャにおける座標から参照ピクチャの座標へのオフセットを与える、*INTER* 予測に使用される2次元ベクトル。
- 3.88 **(参考文脈内での) ～しなければならない (または類似な表現)**：本標準以外で規定される要件または要件の含意に関する意見を表現するのに使用される用語。この用語は参考文脈でのみ使用される。
- 3.89 **NAL ユニット**：その後続くデータのタイプの指示と必要に応じてエミュレーション防止バイト

- を点在させた *RBSP* 形式のデータを含む(複数)バイトを含むシンタックス構造。
- 3.90 **NAL ユニットストリーム** : *NAL* ユニットのシーケンス。
  - 3.91 **非ペアフィールド** : 非ペア参照フィールドまたは非ペア非参照フィールドに対する総称。
  - 3.92 **非ペア非参照フィールド** : 相補的非参照フィールドペアの一部でない復号された非参照フィールド。
  - 3.93 **非ペア参照フィールド** : 相補的参照フィールドペアの一部でない復号された参照フィールド。
  - 3.94 **非参照フィールド** : *nal\_ref\_idc* が 0 に等しく符号化されたフィールド。
  - 3.95 **非参照フレーム** : *nal\_ref\_idc* が 0 に等しく符号化されたフレーム。
  - 3.96 **非参照ピクチャ** : *nal\_ref\_idc* が 0 に等しく符号化されたピクチャ。非参照ピクチャは、他のピクチャの *INTER* 予測のためには使用されない。
  - 3.97 **記** : 参考注釈の接頭辞に使用される用語。この用語は排他的に参考文脈で使用される。
  - 3.98 **逆パリティ** : トップの逆パリティはボトムである。逆もまた同様。
  - 3.99 **出力順序** : 復号ピクチャが復号ピクチャバッファから出力される際の順番。
  - 3.100 **P スライス** : 各ブロックのサンプル値を予測するために、*INTRA* 予測、または、最大 1 つの動きベクトルと参照インデックスを使う *INTER* 予測を用いて復号されてよい、*SP* スライスではないスライス。
  - 3.101 **パラメータ** : シーケンスパラメータセット或いはピクチャパラメータセットのシンタックス要素。また、パラメータは、量子化パラメータとして定義された用語の一部としても使用される。
  - 3.102 **パリティ** : フィールドのパリティは、トップかボトムである。
  - 3.103 **パーティション** : 集合の各要素が、ただ 1 つの部分集合に含まれるような、集合から部分集合への分割。
  - 3.104 **ピクチャ** : フィールドまたは、フレームを指す総称。
  - 3.105 **ピクチャパラメータセット** : それぞれのスライスヘッダで見つかる *pic\_parameter\_set\_id* シンタックス要素で決定されるような 0 個以上の全符号化ピクチャにあてはまるシンタックス要素を含むシンタックス構造。
  - 3.106 **ピクチャ順序カウント** : 復号順序で直前の *IDR* ピクチャに相対の、あるいは全ての参照ピクチャを“参照不使用”としてマークするメモリ管理制御操作を含む直前のピクチャに相対の、出力順序で増加するピクチャ位置で非減少する値を持つ変数。
  - 3.107 **予測** : 予測処理の実体。
  - 3.108 **予測処理** : 現在復号されているサンプル値やデータ要素の推定を行うため予測因子を用いること。
  - 3.109 **予測スライス** : P スライス参照。
  - 3.110 **予測因子** : 後続するサンプル値やデータ要素の復号処理に用いられる、規定された値や過去に復号されたサンプル値やデータ要素の組み合わせ。
  - 3.111 **主符号化ピクチャ** : 本標準に適合するビットストリームの復号処理で使用される、ピクチャの符号化表現。主符号化ピクチャは、ピクチャの全てのマクロブロックを含む。復号処理に関して必須な影響をもつピクチャは、主符号化ピクチャだけである。冗長符号化ピクチャも参照のこと。
  - 3.112 **プロファイル** : 本標準のシンタックスの規定されたサブセット。
  - 3.113 **量子化パラメータ** : 復号処理によって変換係数レベルのスケーリングに使用される変数。
  - 3.114 **ランダムアクセス** : ストリームの始め以外の位置で、ビットストリームに対する復号処理を開始する動作。
  - 3.115 **ラスト走査** : 矩形の 2 次元パターンから 1 次元パターンへのマッピング。1 次元パターンの始めのエントリは左から右へ走査される 2 次元パターンの上部第 1 番目の行からであり、同じように第 2 番目の行、第 3 番目の行のようにパターンの行が続き (下がって行き)、各々の行は左から右

- へ走査される。
- 3.116 **未加工のバイトシーケンスペイロード (RBSP)** : *NAL* ユニットでカプセル化された、整数個のバイトを含むシンタックス構造。**RBSP** は、空であるか、シンタックス要素に **RBSP** ストップビットと、0 個以上の 0 である連続ビットが続く、データビット記号列の形式を持つ。
- 3.117 **未加工のバイトシーケンスペイロード (RBSP) ストップビット** : データビット記号列後、未加工のバイトシーケンスペイロード (**RBSP**) 内に存在する、値が 1 であるビット。**RBSP** 内のデータビット記号列の最後の位置は、**RBSP** の最後の非ゼロのビットである **RBSP** ストップビットを **RBSP** の最後から検索することによって識別できる。
- 3.118 **復旧点** : ランダムアクセスあるいはブローケンリンク後、ビットストリームによって表現される復号ピクチャの正確または近似の表示復旧が達成されるビットストリーム上の場所。
- 3.119 **冗長符号化ピクチャ** : ピクチャまたはピクチャの一部の符号化表現。冗長符号化ピクチャの内容は、本標準に適合したビットストリームの復号処理では使用されてはならない。冗長符号化ピクチャは、主符号化ピクチャの全てのマクロブロックを含むことを要求されない。冗長符号化ピクチャは、復号処理に関して必須な影響を与えない。主符号化ピクチャも参照のこと。
- 3.120 **参照フィールド** : 参照フィールドは、符号化フィールドの *P*、*SP*、*B* スライス、または符号化フレームのフィールドマクロブロック復号時、*INTER* 予測のために用いられて良い。参照ピクチャも参照のこと。
- 3.121 **参照フレーム** : 参照フレームは、符号化フレームの *P*、*SP*、*B* スライスの復号時、*INTER* 予測のために用いられて良い。参照ピクチャも参照のこと。
- 3.122 **参照インデックス** : 参照ピクチャリストへのインデックス。
- 3.123 **参照ピクチャ** : *nal\_ref\_idc* が 0 に等しくないピクチャ。参照ピクチャは、復号順序で後続するピクチャの復号処理において *INTER* 予測に使用されても良いサンプルを含む。
- 3.124 **参照ピクチャリスト** : *P*、*B*、または *SP* スライスの *INTER* 予測に使用される参照ピクチャのリスト。*P* または *SP* スライスの復号処理においては 1 つの参照ピクチャリストがある。*B* スライスの復号処理においては 2 つの参照ピクチャリストがある。
- 3.125 **参照ピクチャリスト 0** : *P*、*B*、*SP* スライスの *INTER* 予測で使用される参照ピクチャリスト。*P* と *SP* スライスで使用される全ての *INTER* 予測は、参照ピクチャリスト 0 を使用する。参照ピクチャリスト 0 は、*B* スライスの *INTER* 予測に用いられる 2 つの参照ピクチャリストの片方であり、他方は参照ピクチャリスト 1 である。
- 3.126 **参照ピクチャリスト 1** : *B* スライスの *INTER* 予測に用いられる参照ピクチャリスト。参照ピクチャリスト 1 は、*B* スライスの *INTER* 予測で用いられる 2 つの参照ピクチャリストの片方であり、他方は参照ピクチャリスト 0 である。
- 3.127 **参照ピクチャマーキング** : ビットストリーム上で復号ピクチャが *INTER* 予測のためマークされる方法に関して規定する。
- 3.128 **予約** : 予約という用語は、特定のシンタックス要素のいくつかの値を規定する節の中で使用される場合は、TTC による将来使用のためにある。これらの値は本標準に適合するビットストリームの中では使用してはならない。しかし、TTC による本標準の将来拡張においては、使用されるかもしれない。
- 3.129 **残差** : サンプルやデータ要素の予測と、その復号値との差分が復号されたもの。
- 3.130 **ラン** : 復号処理において表現される連続するデータ要素の数。ある状況においては、ジグザグ走査またはフィールド走査によって生成された変換係数レベルのリストにおける、非ゼロ変換係数レベルに先立つゼロ値変換係数レベルの数である。別の状況では、ランはマクロブロックの数を呼ぶ。

- 3.131 **サンプルアスペクト比**：本標準では規定されない表示処理を支援するために、フレームにおける輝度サンプル配列の、列の間の意図した水平距離と、行の間の意図した垂直距離との間の比率を規定する。サンプルアスペクト比は  $h:v$  で表される。ここで、(空間距離の任意の単位で)  $h$  は水平方向の幅であり  $v$  は垂直方向の高さである。
- 3.132 **スケーリング**：変換係数レベルに因子を掛け合わせる処理であり、その結果変換係数を得る。
- 3.133 **シーケンスパラメータセット**：それぞれのスライスヘッダで見つかる `pic_parameter_set_id` シンタックス要素で参照されるピクチャパラメータセットで見つかる `seq_parameter_set_id` シンタックス要素の内容で決定されるような 0 個以上の全符号化ビデオシーケンスにあてはまるシンタックス要素を含むシンタックス構造。
- 3.134 **～しなければならない (または類似な表現)**：本標準への適合に対しての必須要件を表現するために使用される用語。シンタックス要素の値または規定された復号処理の操作で得られた結果の必須制約を表現することに使用される時、その制約が満たされることを保証することは符号器の責任である。復号処理で行われる操作に関して使用される時、ここで記述された復号処理と同一結果が生じる任意の復号処理は、本標準の復号処理要件に適合する。
- 3.135 **～すべきである (または類似な表現)**：予期される通常状況下で奨励される実装の動作に言及するために使用される用語だが、本標準への適合に対する必須要件ではない。
- 3.136 **SI スライス**：*INTRA* 予測のみを用い、さらに予測サンプルの量子化を用いて符号化されるスライス。SI スライスは、その復号サンプルが *SP* スライスと同一に構築できるよう符号化できる。
- 3.137 **スキップマクロブロック**：マクロブロックが“スキップ”として復号されるという指示以外、符号化されたデータが存在しないマクロブロックのこと。この指示は、いくつかのマクロブロックで共通であってもよい。
- 3.138 **スライス**：特定のスライスグループ内のラスタ走査で連続した順番の、整数個のマクロブロックまたはマクロブロックペア。主符号化ピクチャにおいては、各スライスグループからスライスへの分割はパーティションとなる。スライスは、スライスグループの中のラスタ走査で連続した順番のマクロブロックやマクロブロックペアを含んでいるが、そのマクロブロックやマクロブロックペアは、ピクチャの内部で連続している必要はない。マクロブロックアドレスは、(スライスヘッダで表現される) スライスにおける最初のマクロブロックアドレスと、マクロブロックからスライスグループへのマップ、および、ピクチャが 3 つの分離された色プレーンを用いて符号化されている場合は色プレーン識別子から導出される。
- 3.139 **スライスデータパーティション**：各シンタックス要素に関連するカテゴリに基づき、選択されたシンタックス要素からシンタックス構造へパーティションする方法。
- 3.140 **スライスグループ**：ピクチャのマクロブロックまたはマクロブロックペアの部分集合。ピクチャからスライスグループへの分割は、ピクチャのパーティションである。このパーティションは、マクロブロックからスライスグループへのマップによって規定される。
- 3.141 **スライスグループマップユニット**：マップユニットからスライスグループへのマップの単位。
- 3.142 **スライスヘッダ**：スライス内で表された最初のまたは全てのマクロブロックに属するデータ要素を含む符号化スライスの一部。
- 3.143 **情報源 (原～)**：符号化する前のビデオ素材やそのいくつかの属性を記述する用語。
- 3.144 **SP スライス**：各ブロックのサンプル値を予測するために、*INTRA* 予測または最大 1 つの動きベクトルと参照インデックスを用いた予測サンプル量子化つきの *INTER* 予測を用いて符号化されてよいスライス。SP スライスは、その復号サンプルが他の SP スライス、*SI* スライスと同一に構築できるように符号化される。
- 3.145 **スタートコードプリフィックス**：各 *NAL* ユニットに対するプリフィックスとしてバイトストリー

ムに組み込まれた 0x000001 に等しい 3 バイトの唯一のシーケンス。スタートコードプリフィックスの位置は、新たな NAL ユニットの開始と先行する NAL ユニットの終了を識別するために復号器によって使える。エミュレーション防止バイトを含むことにより、NAL ユニット内でのスタートコードプリフィックスのエミュレーションが防止される。

- 3.146 **データビット記号列 (SODB)** : 未加工のバイトシーケンスペイロードストップビットの前にある未加工のバイトシーケンスペイロードの中に存在する、(複数の)シンタックス要素を表すいくつかのビットの並び。SODB 内では、最も左側のビットは、最初の最上位ビットと考えられ、最も右側のビットは、最後の最下位ビットと考えられる。
- 3.147 **サブマクロブロック** : マクロブロックのサンプルの 4 分の 1、すなわち、3 つのサンプル配列を持つピクチャに対してはその 1 つの角がマクロブロックの角に位置する 8×8 の輝度ブロックと対応する 2 個の色差ブロック、モノクロームピクチャまたは 3 つの分離された色プレーンを用いて符号化されたピクチャに対してはその 1 つの角がマクロブロックの角に位置する 8×8 の輝度ブロックである。
- 3.148 **サブマクロブロックパーティション** : 3 つのサンプル配列を持つピクチャに対しては *INTER* 予測のサブマクロブロックのパーティションに起因する輝度サンプルのブロックと対応する色差サンプルの 2 つのブロック、あるいは、モノクロームピクチャまたは 3 つの分離された色プレーンを用いて符号化されたピクチャに対しては *INTER* 予測のサブマクロブロックのパーティションに起因する輝度サンプルのブロック。
- 3.149 **スイッチング I スライス** : *SI* スライス参照。
- 3.150 **スイッチング P スライス** : *SP* スライス参照。
- 3.151 **シンタックス要素** : ビットストリームの中で表現されるデータの要素。
- 3.152 **シンタックス構造** : 規定された順番でビットストリーム中に共に存在する、0 個以上のシンタックス要素。
- 3.153 **トップフィールド** : フレームを構成する 2 つのフィールドの 1 つ。トップフィールドの各行は、ボトムフィールドの対応する行に対し、空間的にすぐ上に位置する。
- 3.154 **(マクロブロックペアの) トップマクロブロック** : マクロブロックペアに対するサンプルのトップ行のサンプルを含むマクロブロックペアの中のマクロブロック。フィールドマクロブロックペアでは、トップマクロブロックは、マクロブロックペアの空間領域に位置するフレームのトップフィールド領域からのサンプルを表す。フレームマクロブロックペアでは、トップマクロブロックは、マクロブロックペアの空間領域の上半分に位置するフレームのサンプルを表す。
- 3.155 **変換係数** : 復号処理の逆変換部における特定の 1 次元または 2 次元周波数インデックスに関連する周波数領域におけるスカラー量。
- 3.156 **変換係数レベル** : 変換係数値の計算におけるスケールリング前の復号処理で、特定の 2 次元周波数インデックスに関連する値を表す整数量。
- 3.157 **ユニバーサルユニーク識別子 (UUID)** : 全てのユニバーサルユニーク識別子の空間に関して、唯一の識別子。
- 3.158 **無規定** : 無規定という用語は、特定のシンタックス要素のいくつかの値を規定する節で使用される時、その値は、本標準で規定された意味を持たず、本標準の必須部分として将来も規定された意味を持たないことを示す。
- 3.159 **可変長符号化 (VLC)** : より頻度の高いと予想されるシンボルにより短いビット記号列を、より頻度の低いと予想されるシンボルにより長いビット記号列を割り当てるエントロピ符号化のための可逆である手続き。
- 3.160 **VCL NAL ユニット** : 符号化スライス NAL ユニットと符号化スライスデータパーティション NAL

ユニットの総称。

- 3.161 **ジグザグ走査**：(おおよそ) 最も低い空間周波数から最も高い周波数への順になっている、変換係数レベルの連続した順番づけの規定。ジグザグ走査はフレームマクロブロック内の変換係数レベルに使用される。

## 4 略語

本標準の目的のために以下の略語が適用される

- CABAC** コンテキスト適応バイナリ算術符号化(Context-based Adaptive Binary Arithmetic Coding)
- CAVLC** コンテキスト適応可変長符号化(Context-based Adaptive Variable Length Coding)
- CBR** 固定ビットレート(Constant Bit Rate)
- CPB** 符号化ピクチャバッファ(Coded Picture Buffer)
- DPB** 復号ピクチャバッファ(Decoded Picture Buffer)
- DUT** 試験される復号器(Decoder Under Test)
- FIFO** 先入れ先出し(First-In, First-Out)
- HRD** 仮想標準復号器(Hypothetical Reference Decoder)
- HSS** 仮想ストリームスケジューラ(Hypothetical Stream Scheduler)
- IDR** 瞬時復号更新(Instantaneous Decoding Refresh)
- LSB** 最下位ビット(Least Significant Bit)
- MB** マクロブロック(Macroblock)
- MBAFF** マクロブロック適応フレーム/フィールド符号化(Macroblock-Adaptive Frame-Field Coding)
- MSB** 最上位ビット(Most Significant Bit)
- NAL** ネットワーク抽象レイヤ(Network Abstraction Layer)
- RBSP** 未加工のバイトシーケンスペイロード(Raw Byte Sequence Payload)
- SEI** 付加拡張情報(Supplemental Enhancement Information)
- SODB** データビット記号列(String Of Data Bits)
- SVC** スケーラブルビデオ符号化(Scalable Video Coding)
- UUID** ユニバーサルユニーク識別子(Universal Unique Identifier)
- VBR** 可変ビットレート(Variable Bit Rate)
- VCL** ビデオ符号化レイヤ(Video Coding Layer)
- VLC** 可変長符号化(Variable Length Coding)
- VUI** ビデオ有用性情報(Video Usability Information)

## 5 規則

記 本標準の記述で使用される数学的演算子は、Cプログラミング言語で使われるものと同様なものである。しかし整数除算や算術シフト演算は明確に定義する。また、番号振りや数え上げは、通常ゼロから始める。

### 5.1 算術演算子

以下の算術演算子は、下記のように定義する。

+	加算。
-	減算（2項演算）。または、符号反転（単項演算）。
*	乗算。
$x^y$	べき乗。 $x$ の $y$ 乗を規定する。別の文脈では、このような表記はべき乗としての解釈を意図されず、上付き文字として使用される。
/	演算結果をゼロに近づけるように切り捨てる整数除算。例えば、 $7/4$ や $-7/-4$ は、いずれも1に切り捨てられ、また $-7/4$ や $7/-4$ は、 $-1$ に切り捨てられる。
÷	切り捨てや丸めを行わない、数学的な等式における除算を示す。
$\frac{x}{y}$	切り捨てや丸めを行わない、数学的な等式における除算を示す。
$\sum_{i=x}^y f(i)$	$i$ が $x$ から $y$ までの $y$ を含む全ての整数値をとる、 $f(i)$ の総和。
$x\%y$	モジュロ。 $x$ を $y$ で割った余り。 $x$ および $y$ は整数で $x \geq 0$ かつ $y > 0$ の場合のみ定義。

先行の順序が括弧の使用によって明示的に示されない場合、以下のルールを適用する。

- 乗算と除算演算は、加算と減算の前に行うと考える。
- 連続した乗算と除算演算は、左から右の順に連続して評価される。
- 連続した加算と減算演算は、左から右の順に連続して評価される。

### 5.2 論理演算子

以下の論理演算子は、下記のように定義される。

$x \&\& y$	$x$ 、 $y$ のブール論理積
$x \parallel y$	$x$ 、 $y$ のブール論理和
!	ブール論理否定
$x?y:z$	$x$ が真または非ゼロなら、 $y$ の値を評価する。そうでないならば、 $z$ の値を評価する。

### 5.3 関係演算子

以下の関係演算子は、下記のように定義される。

>	大なり
>=	大なりまたは、等しい

<	小なり
<=	小なりまたは、等しい
=	等しい
!=	等しくない

値”na”（適用可能でない）が割り当てられたあるシンタックス要素または変数に関係演算子が適用される時、値”na”はそのシンタックス要素または変数について別個の値として扱われる。値”na”は他のどのような値とも等しくないとみなされる。

## 5.4 ビット演算子

以下のビット演算子は、下記のように定義される。

&	ビット毎の“AND”。整数引数の演算の場合には、整数値の2の補数表現での演算である。他方の引数より少ないビットを含むバイナリ引数の演算では、短い引数は0に等しいさらなる上位ビットを加えて拡張される。
	ビット毎の“OR”。整数引数の演算の場合には、整数値の2の補数表現での演算である。他方の引数より少ないビットを含むバイナリ引数の演算では、短い引数は0に等しいさらなる上位ビットを加えて拡張される。
x>>y	2の補数表現整数xに対して、バイナリ桁数yの算術的な右シフト。この関数は、yが正の整数値でのみ定義される。右シフトの結果としてMSBに入るシフトされたビットは、シフト演算前のxのMSBに等しい値を持つ。
x<<y	2の補数表現整数xに対して、バイナリ桁数yの算術的な左シフト。この関数は、yが正の整数値でのみ定義される。左シフトの結果としてLSBに入るシフトされたビットは、ゼロに等しい値を持つ。

## 5.5 代入演算子

以下の算術演算子は、下記のように定義される。

=	代入演算子。
++	1だけ加算。すなわち、x++は、x=x+1と等価である。配列のインデックスで使用時には、加算演算より、変数値の評価を優先する。
--	1だけ減算。すなわち、x--は、x=x-1と等価である。配列のインデックスで使用時には、減算演算より、変数値の評価を優先する。
+=	指定値の加算。すなわち、x+=3は、x=x+3であり、x+=(-3)は、x=x+(-3)と等価である。
-=	指定値の減算。すなわち、x-=3は、x=x-3であり、x-=(-3)は、x=x-(-3)と等価である。

## 5.6 範囲表記

以下の表記法は、範囲の値を示すのに使用される。

x=y..z	xは、yからzまでのそれぞれの値を含む整数値をとり、またx、y、zは整数値である。
--------	---

## 5.7 数学的関数

以下の数学的関数は、下記のように定義される。

$$\text{Abs}(x) = \begin{cases} x & ; x \geq 0 \\ -x & ; x < 0 \end{cases} \quad (5-1)$$

$$\text{Ceil}(x) \quad x \text{より大きいか、または等しい、最小の整数。} \quad (5-2)$$

$$\text{Clip1}_y(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_y) - 1, x) \quad (5-3)$$

$$\text{Clip1}_c(x) = \text{Clip3}(0, (1 \ll \text{BitDepth}_c) - 1, x) \quad (5-4)$$

$$\text{Clip3}(x, y, z) = \begin{cases} x & ; z < x \\ y & ; z > y \\ z & ; \text{それ以外} \end{cases} \quad (5-5)$$

$$\text{Floor}(x) \quad x \text{より小さいか、または等しい、最大の整数。} \quad (5-6)$$

$$\text{InverseRasterScan}(a, b, c, d, e) = \begin{cases} (a \% (d / b)) * b; & e == 0 \\ (a / (d / b)) * c; & e == 1 \end{cases} \quad (5-7)$$

$$\text{Log2}(x) \text{は} x \text{に対して底2の対数を取る。} \quad (5-8)$$

$$\text{Log10}(x) \text{は} x \text{に対して底10の対数を取る。} \quad (5-9)$$

$$\text{Median}(x, y, z) = x + y + z - \text{Min}(x, \text{Min}(y, z)) - \text{Max}(x, \text{Max}(y, z)) \quad (5-10)$$

$$\text{Min}(x, y) = \begin{cases} x & ; x \leq y \\ y & ; x > y \end{cases} \quad (5-11)$$

$$\text{Max}(x, y) = \begin{cases} x & ; x \geq y \\ y & ; x < y \end{cases} \quad (5-12)$$

$$\text{Round}(x) = \text{Sign}(x) * \text{Floor}(\text{Abs}(x) + 0.5) \quad (5-13)$$

$$\text{Sign}(x) = \begin{cases} 1 & ; x \geq 0 \\ -1 & ; x < 0 \end{cases} \quad (5-14)$$

$$\text{Sqrt}(x) = \sqrt{x} \quad (5-15)$$

## 5.8 変数、シンタックス要素、表

ビットストリーム中のシンタックス要素は**ボールド(太文字)**タイプで表現される。各シンタックス要素は、その名前 (アンダスコアを含む、全て小文字)、1つないし2つのシンタックスカテゴリ、そして符号化表現方法のための、1つないし2つの記述子により記述される。復号処理は、シンタックス要素の値に、そして、以前復号されたシンタックス要素の値に従って動作する。シンタックス要素の値がシンタックス表あるいは文書中で使用される場合、それは通常タイプで (つまり、ボールドでない) 現れる。

いくつかの場合において、シンタックス表は、シンタックス要素から導出された他の変数値を使用するかもしれない。シンタックス表、或は文書に現れるそのような変数は、アンダスコアなしの小文字および大文字の混合で命名される。大文字で始まる変数は、現在のシンタックス構造および全ての依存するシンタックス構造の復号のため導出される。大文字から始まる変数は、その変数の元のシンタックス構造に言及する、後のシンタックス構造の復号処理で使用されてもよい。小文字で始まる変数は、それらが導出される節内部でのみ使用される。

いくつかの場合において、シンタックス要素値、変数値のための“ニーモニック”名が、それらの数値と交換可能に使用される。しばしば、“ニーモニック”名は、関連する数値なしで使用される。そして、値と名前の関連は、文書中で規定される。その名前は、アンダスコアによって分離された1つ以上の文字のグループから構成される。各グループは大文字で始まり、より多くの大文字を含んでいてもよい。

記 シンタックスは、C言語のシンタックス構成に近い方法で記述される。

関数は、シンタックス要素名として構築される名前と、(複数変数時には)コンマで分離される、0個以上の変数名(定義)または値(使用法)を含んだ左右の小括弧で記述される。

1次元配列はリストと呼ばれる。2次元配列は行列と呼ばれる。配列はシンタックス要素あるいは変数となりうる。下付き添字または大括弧は配列の中でインデックスを付けるために使用される。行列の視覚的描写を参照する際には、第1下付き添字は行(垂直)インデックスとして使用され、第2下付き添字は列(水平)インデックスとして使用される。索引に関する下付き添字の代わりに大括弧を使用する時、インデックス順序は逆になる。従って、水平位置  $x$ 、垂直位置  $y$  である行列  $s$  の要素は  $s[x,y]$  として記されても  $s_{yx}$  として記されてもよい。

バイナリ表記は、シングルクォーテーションマークによってビット値の記号列を囲むことで示される。例えば、'01000001'は、2番目と、最後のビットだけが、1となる、8ビットの記号列を表す。

16進表記は、16進数の先頭に“0x”を付けることで示され、ビット数が4の倍数である場合、バイナリ表記の代わりに使用してもよい。例えば、0x41は、2番目と、最後のビットが1である8ビットの記号列を表す。

シングルクォーテーションで囲まれず、そして、“0x”も前に付かない数値は、10進の値である。

テストステートメントで、0に等しい値は偽条件(FALSE)を表す。値、真(TRUE)は、ゼロ以外の任意の値で表される。

## 5.9 論理演算の文書表記

文書において、次の擬似コードで記述される論理演算のステートメントは、

```
if(条件 0)
    ステートメント 0
else if(条件 1)
    ステートメント 1
...
else /* 残りの条件の参考注釈 */
    ステートメント n
```

以下の様式で記述されてもよい。

～次（以下）の通り（ように）～／～次（以下）が適用される。

- もし条件0ならば、ステートメント0
- それ以外、もし条件1ならば、ステートメント1
- …
- それ以外(残りの条件の参考注釈)、ステートメントn

文書中の各々の「もし...それ以外、もし...それ以外、...」ステートメントが「もし...」の直前の「～次（以下）の通り（ように）～」または「～次（以下）が適用される」で始められる。「もし...それ以外、もし...それ以外、...」の最後の条件はいつも1つの「それ以外、...」である。インターリーブされた「もし...それ以外、もし...それ以外、...」ステートメントは、最後の「それ以外、...」にマッチする「～次（以下）の通り（ように）～」または「～次（以下）が適用される」によって識別できる。

文書において、次の擬似コードで記述される論理演算のステートメントは、

```
if(条件 0a && 条件 0b)
    ステートメント 0
else if(条件 1a || 条件 1b)
    ステートメント 1
...
else
    ステートメント n
```

以下の様式で記述されてもよい。

～次（以下）の通り（ように）～／～次（以下）が適用される。

- もし以下の条件の全てが真ならば、ステートメント0
  - 条件0a
  - 条件0b
- それ以外、もし以下の条件のいずれかが真ならば、ステートメント1
  - 条件1a
  - 条件1b
- …
- それ以外、ステートメントn

文書において、次の擬似コードで記述される論理演算のステートメントは、

```
if(条件 0)
    ステートメント 0
if(条件 1)
    ステートメント 1
```

以下の様式で記述されてもよい。

条件 0 ならば (条件 0 の時、条件 0 の場合 など)、ステートメント 0

条件 1 ならば (条件 1 の時、条件 1 の場合 など)、ステートメント 1

## 5.10 処理

処理は、シンタックス要素の復号を記述するために使用される。処理は、個別の規定と起動を持っている。現在のシンタックス構造および依存するシンタックス構造に属する全てのシンタックス要素および大文字変数は、その処理規定と起動で利用可能である。処理規定は、入力として明示的に規定された小文字変数を持っていても良い。各処理規定は、明示的に規定された出力を持っている。その出力は大文字変数あるいは小文字変数のいずれかの変数である。

変数の割り当てが次の通り規定される：

- もし処理を起動する時同じ名前を持たないならば、変数は処理規定の小文字入力あるいは出力変数に明示的に割り当てられる。
- それ以外(起動する変数と規定が同じ名前を持つ場合)、割り当てが暗に意味される。

処理の規定では、特定のマクロブロックが、その特定のマクロブロックのアドレスに等しい値をもつ変数名で呼ばれるかもしれない。

## 6 情報源、符号化、復号、および出力データフォーマット、走査処理、隣接関係

### 6.1 ビットストリームフォーマット

この節は、いずれもビットストリームと呼ばれる NAL ユニットストリームとバイトストリームの関係を規定する。

ビットストリームは、2つのフォーマット、NAL ユニットストリームフォーマットまたは、バイトストリームフォーマットのいずれかの1つにありうる。NAL ユニットストリームフォーマットは、概念的に、より“基本的”なタイプである。それは、NAL ユニットと呼ばれるシンタックス構造のシーケンスから成る。このシーケンスは、復号順序で並ぶ。NAL ユニットストリームの NAL ユニットの復号順序 (および、内容) が負う制約が存在する。

バイトストリームフォーマットは、NAL ユニットの復号順序に配列し、バイトのストリームを形成するために、スタートコードプリフィックスと 0 個以上のゼロ値バイトを、各 NAL ユニットの前に付けることで NAL ユニットストリームフォーマットから構築することができる。NAL ユニットストリームフォーマットは、このバイトのストリームにある唯一のスタートコードプリフィックスパターンの位置を検索することで、バイトストリームフォーマットから抽出することができる。バイトストリームフォーマットの使用以外の方法による NAL ユニットのフレーム化する手段は、本標準の範囲外である。バイトストリームフォーマットは、付属資料 B で規定される。

### 6.2 情報源、復号、および出力ピクチャフォーマット

この節では、情報源と、ビットストリームによって与えられる復号フレームおよびフィールドとの関係に

ついて規定する。

ビットストリームによって表現されるビデオ情報源は、復号順序でのフレームとフィールドのいずれか、または両方（総称的にピクチャと呼ばれる）のシーケンスで表現される。

情報源や復号ピクチャ（フレームまたは、フィールド）は、各々1以上のサンプル配列からなる。

- 補助配列をもつ、またはもたない、輝度(Y)のみ（モノクローム）
- 補助配列をもつ、またはもたない、輝度と2つの色差（YCbCr または YCgCo）
- 補助配列をもつ、またはもたない、緑、青および赤(GBR、RGB としても知られる)
- 補助配列をもつ、またはもたない、他の無規定のモノクロームまたは 3 刺激色サンプリング（例えば、YZX、XYZ としても知られる）を表現する配列

本規定では記法と語法の便利のため、これらの配列に関連した変数と用語は輝度(または L または Y)と色差と呼ばれ、ここでは、実際の使用される色表現によらず、2つの色差配列は Cb と Cr と呼ばれる。実際の使用される色表現は付属資料 E で規定されるシンタックスで指示されうる。符号化ビデオシーケンスに補助ピクチャとして存在しても良いし、しなくても良い（モノクローム）補助配列は、復号においてオプションであり、アルファブレンディングのような目的に対して使用されうる。

変数 SubWidthC と SubHeightC は、chroma\_format\_idc および separate\_colour\_plane\_flag を通して規定される色差フォーマットサンプリング構造に従って、表 6-1/JT-H264 で規定される。表 6-1/JT-H264 で“-”としてマークされるエントリは SubWidthC または SubHeightC において定義されない値を示す。他の chroma\_format\_idc、SubWidthC と SubHeightC の値は、TTC により将来規定されるかもしれない。

表6-1/JT-H264 chroma\_format\_idcおよびseparate\_colour\_plane\_flagから導出されるSubWidthCとSubHeightCの値  
(ITU-T H.264)

chroma_format_idc	separate_colour_plane_flag	色差フォーマット	SubWidthC	SubHeightC
0	0	モノクローム	-	-
1	0	4:2:0	2	2
2	0	4:2:2	2	1
3	0	4:4:4	1	1
3	1	4:4:4	-	-

モノクロームサンプリングでは、名目上輝度配列と考えられる1つのサンプル配列のみがある。

4:2:0 サンプリングでは、2つの色差配列の各々は輝度配列の半分の高さと半分の幅をもつ。

4:2:2 サンプリングでは、2つの色差配列の各々は輝度配列と同じ高さと半分の幅をもつ。

4:4:4 サンプリングでは、separate\_colour\_plane\_flag の値に依存して、次が適用される。

- もし separate\_colour\_plane\_flag が 0 に等しいならば、2つの色差配列の各々は輝度配列と同じ高さと幅をもつ。
- それ以外(separate\_colour\_plane\_flag が 1 に等しい)、3つの色プレーンはモノクロームサンプルのピクチャ

ャとして分離して処理される。

輝度サンプル配列の幅と高さは、各々16の倍数である。4:2:0色差サンプリングを使用する符号化ビデオシーケンスでは、色差サンプル配列の幅と高さは、各々8の倍数である。4:2:2色差サンプリングを使用する符号化ビデオシーケンスでは、色差サンプル配列の幅は8の倍数であり高さは16の倍数である。2つの分離したフィールドとして符号化される、またはマクロブロック適応フレーム/フィールド符号化(下記参照)による輝度配列の高さは32の倍数である。4:2:0色差サンプリングを使用する符号化ビデオシーケンスでは、2つの分離したフィールドとして符号化される、またはマクロブロック適応フレーム/フィールド符号化(下記参照)による各々の色差配列の高さは16の倍数である。復号処理から出力されるピクチャの幅と高さは、16の整数倍である必要はなく、切り出し矩形を使用して規定できる。

輝度と色差(存在する時)配列に対するシンタックスは、3つ全ての色成分のデータが存在する時、他で規定されない限り、輝度配列に対するデータが最初であり、Cb配列に対する任意のデータが続き、Cr配列に対する任意のデータが続くように順序付けられる。

特定のシーケンスパラメータセットを参照して符号化されたフィールドの幅は、同じシーケンスパラメータセットを参照して符号化されたフレームの幅と同じである(下記参照)。特定のシーケンスパラメータセットを参照して符号化されたフィールドの高さは、同じシーケンスパラメータセットを参照して符号化されたフレームの高さの半分である(下記参照)。

ビデオシーケンスにおける輝度と色差配列のサンプルの各々の表現に対して必要なビットの数は、8から14の範囲であり、輝度配列で使用されるビットの数は色差配列で使用されるビットの数と異なっても良い。

chroma\_format\_idcの値が1に等しい場合、フレームにおける輝度、色差サンプルの名目上の垂直、水平の相対位置は、図6-1/JT-H264で示される。代替りの色差サンプル相対位置が、ビデオ有用性情報で示されても良い。(付属資料E参照。)

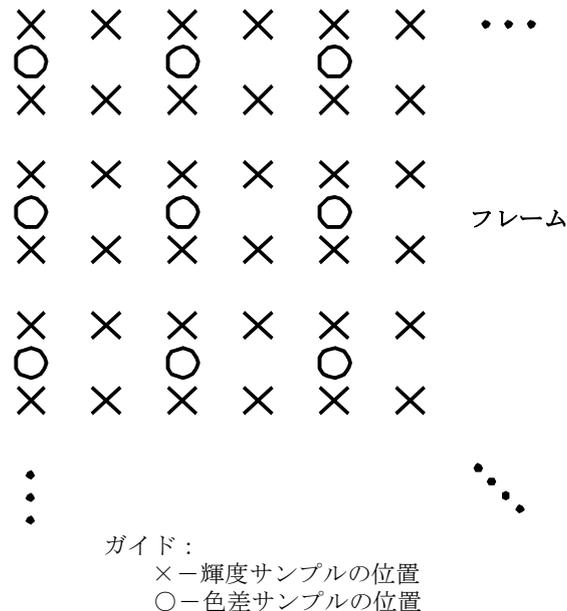


図6-1/JT-H264 フレームにおける4:2:0輝度、色差サンプルの名目上の垂直、水平位置  
(ITU-T H.264)

1枚のフレームは、下記に示すように、2枚のフィールドからなる。符号化ピクチャは、符号化フレーム、

或は別々に符号化されたフィールドを表現してもよい。本標準に適合する符号化ビデオシーケンスは、符号化フレームおよび符号化フィールドの任意の組み合わせを含んでも構わない。また、復号処理は、マクロブロック適応フレーム／フィールド符号化の使用によって、符号化フレームのより小さな領域が、フレームまたはフィールドの領域のいずれかとして符号化されることを許容する方法でも規定される。

情報源および復号されたフィールドは、次の2つのタイプの1つである：トップフィールド、ボトムフィールド。2つのフィールドが同時に出力されるか、参照フレーム（以下を参照）として使用されるために組み合わせられる場合、2つのフィールド（それらは逆パリティでなければならない）はインターリーブされる。復号フレームの第1番目（すなわちトップ）、第3番目、第5番目などの行はトップフィールド行である。復号フレームの第2番目、第4番目、第6番目などの行は、ボトムフィールド行である。トップフィールドは復号フレームのトップフィールド行だけから成る。復号フレームのトップフィールドかボトムフィールドが参照フィールド（以下を参照）として使用される場合、復号フレームの（トップフィールドに対しては）偶数行あるいは、（ボトムフィールドに対しては）奇数行のみ使用される。

chroma\_format\_idc の値が1に等しい場合、トップフィールドと、ボトムフィールドの輝度、色差サンプルの名目上の垂直、水平軸上の相対位置が、図6-2/JT-H264に示される。トップフィールドにおける色差サンプルの名目上の垂直サンプリング相対位置は、フィールドサンプリンググリッドから相対で、4分の1輝度サンプル高だけ上にずれるように規定される。ボトムフィールドにおける色差サンプルの垂直サンプリング位置は、フィールドサンプリンググリッドから相対で、4分の1輝度サンプル高だけ下にずれるように規定される。代替の色差サンプル相対位置が、ビデオ有用性情報の中で指示されてもよい（付属資料E参照）。

記 色差サンプルのずれは、図6-1/JT-H264で示されるような全フレームサンプリンググリッド相対の通常位置に、これらのサンプルを垂直方向で合わせるためである。

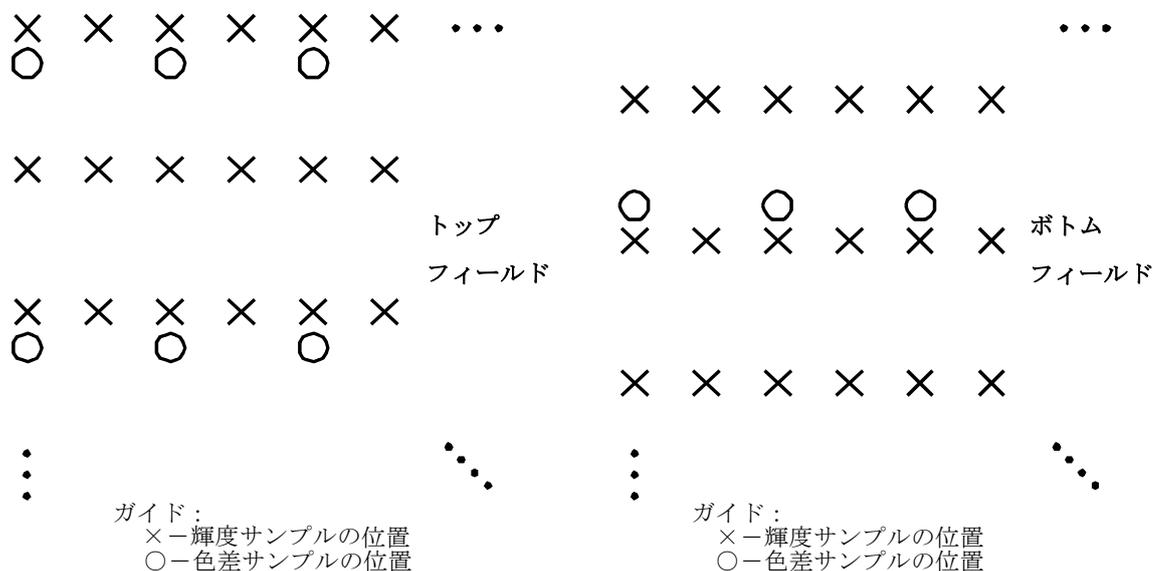


図6-2/JT-H264 トップ、ボトムフィールドにおける4:2:0サンプルの名目上の垂直、水平サンプリング位置 (ITU-T H.264)

chroma\_format\_idc の値が2に等しい場合、色差サンプルは対応する輝度サンプルと共通位置にあり、フレームとフィールドの名目上の位置は図6-3/JT-H264と図6-4/JT-H264で、それぞれ示される。

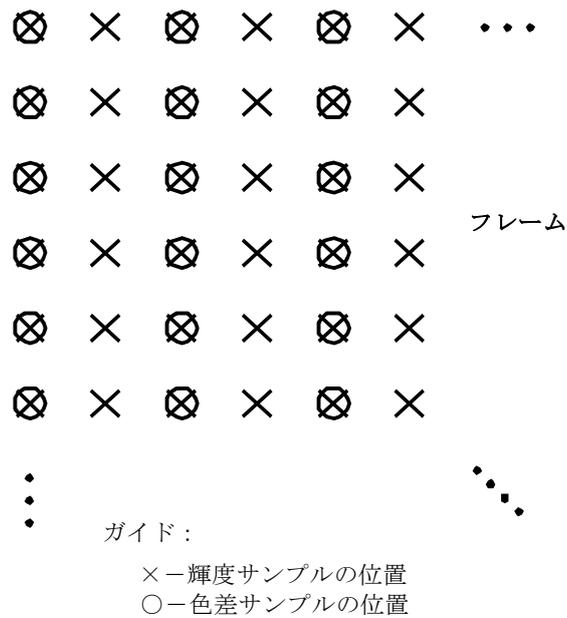


図6-3/JT-H264 フレームにおける4:2:2輝度、色差サンプルの名目上の垂直、水平位置  
(ITU-T H.264)

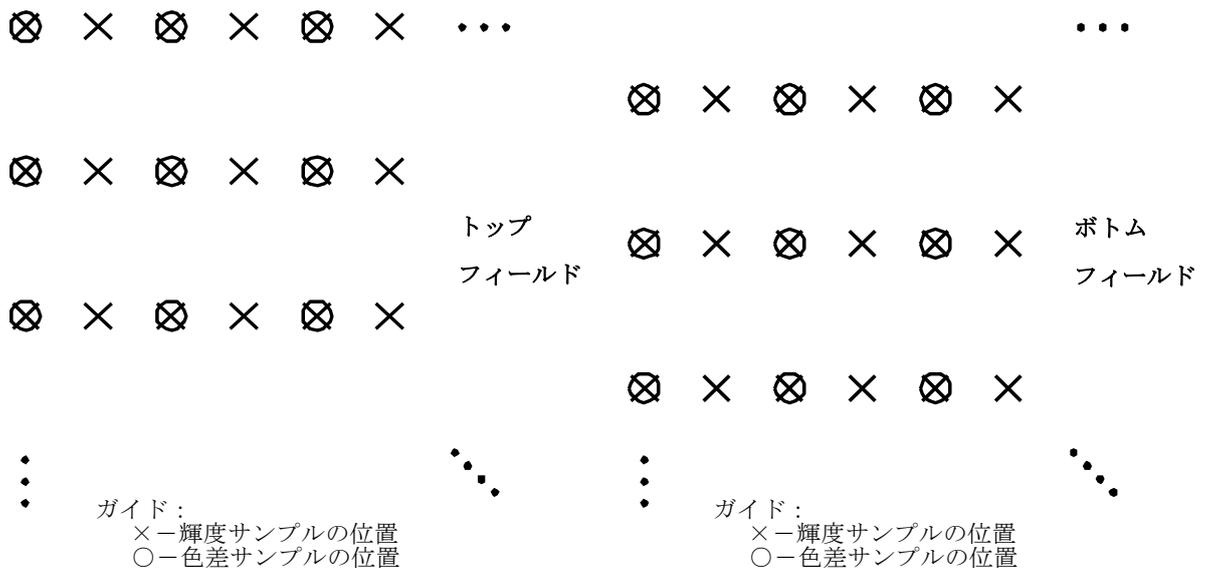


図6-4/JT-H264 トップ、ボトムフィールドにおける4:2:2サンプルの名目上の垂直、水平サンプリング位置  
(ITU-T H.264)

chroma\_format\_idc の値が 3 に等しい場合、全ての配列サンプルは、フレームとフィールドの全ての場合で共通位置にあり、フレームとフィールドの名目上の位置は図 6-5/JT-H264 と図 6-6/JT-H264 で、それぞれ示される。

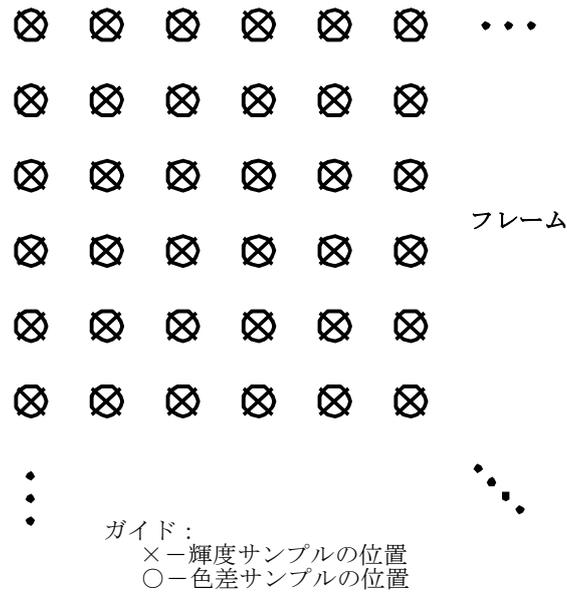


図6-5/JT-H264 フレームにおける4:4:4輝度、色差サンプルの名目上の垂直、水平位置  
(ITU-T H.264)

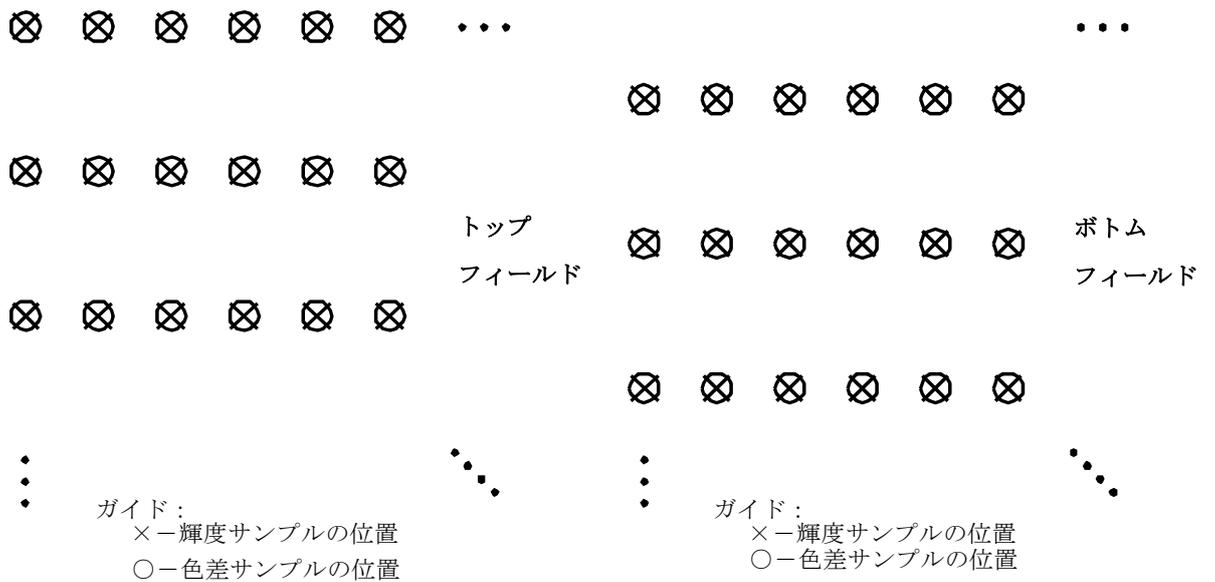


図6-6/JT-H264 トップ、ボトムフィールドにおける4:4:4サンプルの名目上の垂直、水平サンプリング位置  
(ITU-T H.264)

サンプルはマクロブロックの単位で処理される。各々のマクロブロックに対する輝度配列は、幅と高さともに 16 サンプルである。各々のマクロブロックに対する色差配列の幅と高さをそれぞれ規定する、変数 MbWidthC と MbHeightC は次の通り導出される。

- もし chroma\_format\_idc の値が 0 (モノクローム) に等しいか、または separate\_colour\_plane\_flag が 1 に等しいならば、MbWidthC と MbHeightC は両者とも 0 に等しい。

- それ以外、MbWidthC と MbHeightC は以下で導出される。

$$\text{MbWidthC} = 16 / \text{SubWidthC} \quad (6-1)$$

$$\text{MbHeightC} = 16 / \text{SubHeightC} \quad (6-2)$$

### 6.3 ピクチャとスライスの空間的細分化

この節は、ピクチャがスライスとマクロブロックへ、どのようにしてパーティションされるかを規定する。ピクチャは、スライスに分割される。スライスはマクロブロックの並びに、また、マクロブロック適応フレーム／フィールド復号が使用される場合には、マクロブロックペアの並びとなる。

各マクロブロックは、16×16の輝度配列と、色差サンプリングフォーマットが4:0:0に等しくなく、かつ `separate_colour_plane_flag` が 0 に等しい時、2つの対応する色差サンプル配列からなる。`separate_colour_plane_flag` が 1 に等しい時、各マクロブロックは16×16の1個の輝度または色差サンプル配列からなる。マクロブロック適応フレーム／フィールド復号が使用されない場合には、各マクロブロックは、ピクチャの空間的な矩形領域を表現する。例えば、あるピクチャは、図 6-7/JT-H264 で示される通り、2つのスライスに分割されてもよい。

ピクチャが3つの分離された色プレーンを用いて符号化されている時(`separate_colour_plane_flag` は 1 に等しい)、スライスは、対応する `colour_plane_id` の値により識別される1つの色成分のマクロブロックのみを含み、ピクチャの各色成分配列は同じ `colour_plane_id` の値を持つスライスから構成される。アクセスユニット内の異なる `colour_plane_id` の値を持つ符号化スライスは、`colour_plane_id` の各値について、その `colour_plane_id` 値の符号化スライス NAL ユニットの、各符号化スライス NAL ユニットの最初のマクロブロックがマクロブロックアドレス昇順となる順序にならなければならないという制約のもとで、お互いにインターリーブすることができる。

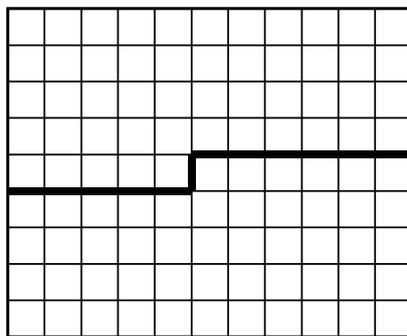


図6-7/JT-H264 2つのスライスにパーティションされる11×9マクロブロック構成のピクチャ (ITU-T H.264)

マクロブロック適応フレーム／フィールド復号が使用される場合、ピクチャは、図 6-8/JT-H264 に示されるような整数個のマクロブロックペアを含むスライスにパーティションされる。それぞれのマクロブロックペアは、2つのマクロブロックから構成される。

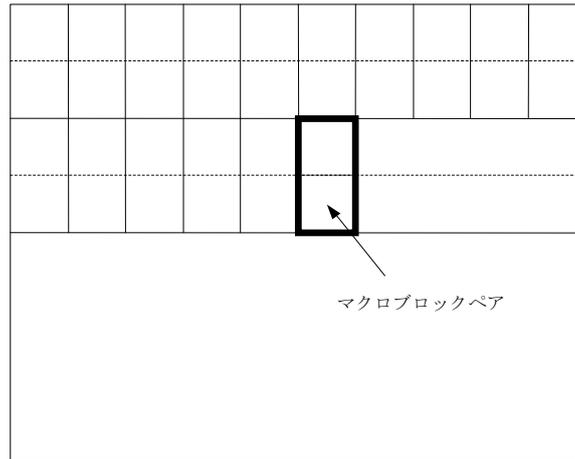


図6-8/JT-H264 復号フレームのマクロブロックペアへのパーティション  
(ITU-T H.264)

#### 6.4 逆走査処理および隣接のための導出処理

この節は、逆走査処理、つまりインデックスから位置へのマッピング、および隣接のための導出処理を規定する。

##### 6.4.1 逆マクロブロック走査処理

この処理への入力は、マクロブロックアドレス  $mbAddr$  である。

この処理の出力は、アドレス  $mbAddr$  を持つマクロブロックに対する、ピクチャ左上端サンプル相対の左上端の輝度サンプルの位置  $(x,y)$  である。

逆マクロブロック走査処理は、次の通り規定される。

- もし  $MbaffFrameFlag$  が 0 に等しいならば、

$$x = \text{InverseRasterScan}(mbAddr, 16, 16, \text{PicWidthInSamples}_L, 0) \quad (6-3)$$

$$y = \text{InverseRasterScan}(mbAddr, 16, 16, \text{PicWidthInSamples}_L, 1) \quad (6-4)$$

- それ以外 ( $MbaffFrameFlag$  が 1 に等しい)、次が適用される。

$$xO = \text{InverseRasterScan}(mbAddr/2, 16, 32, \text{PicWidthInSamples}_L, 0) \quad (6-5)$$

$$yO = \text{InverseRasterScan}(mbAddr/2, 16, 32, \text{PicWidthInSamples}_L, 1) \quad (6-6)$$

現マクロブロックに従って次が適用される。

- もし現マクロブロックがフレームマクロブロックならば、

$$x = xO \quad (6-7)$$

$$y = yO + (mbAddr \% 2) * 16 \quad (6-8)$$

- それ以外（現マクロブロックがフィールドマクロブロック）、

$$x=x0 \tag{6-9}$$

$$y=y0+(mbAddr\%2) \tag{6-10}$$

#### 6.4.2 逆マクロブロックパーティションおよびサブマクロブロックパーティション走査処理

マクロブロックあるいはサブマクロブロックは、パーティションされてもよく、パーティションは図 6-9 /JT-H264 に示すように INTER 予測のために走査される。外部の矩形は、マクロブロックかサブマクロブロックの中のサンプルをそれぞれ言う。矩形はパーティションに当てはまる。各矩形における数字は、逆マクロブロックパーティション走査あるいは逆サブマクロブロックパーティション走査のインデックスを規定する。

マクロブロックパーティションおよびサブマクロブロックパーティションの幅および高さを記述する関数 MbPartWidth(), MbPartHeight(), SubMbPartWidth(), および SubMbPartHeight()は、表 7-13 /JT-H264、7-14 /JT-H264、7-17 /JT-H264、および 7-18 /JT-H264 で規定される。MbPartWidth()および MbPartHeight()は、そのマクロブロックタイプに従って、それぞれのマクロブロックにおいて適切な値に設定される。SubMbPartWidth()および SubMbPartHeight()は、そのサブマクロブロックタイプに従って、P\_8x8、P\_8x8ref0、または B\_8x8 に等しい mb\_type を持つマクロブロックのそれぞれのサブマクロブロックにおいて適切な値に設定される。

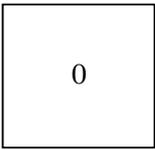
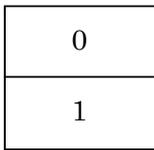
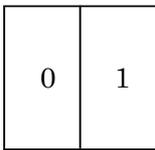
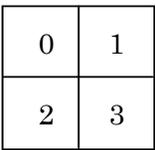
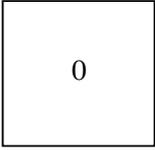
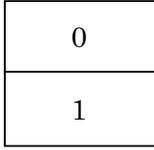
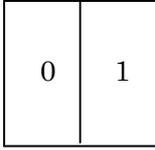
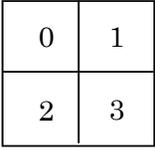
	16×16 輝度サンプルと関連する色差サンプルの1つのマクロブロックパーティション	16×8 輝度サンプルと関連する色差サンプルの2つのマクロブロックパーティション	8×16 輝度サンプルと関連する色差サンプルの2つのマクロブロックパーティション	8×8 輝度サンプルと関連する色差サンプルの4つのマクロブロックパーティション
マクロブロックパーティション				
サブマクロブロックパーティション				

図6-9 /JT-H264 マクロブロックパーティション、サブマクロブロックパーティション、マクロブロックパーティション走査、およびサブマクロブロックパーティション走査 (ITU-T H.264)

##### 6.4.2.1 逆マクロブロックパーティション走査処理

この処理への入力は、マクロブロックパーティションのインデックス mbPartIdx である。

この処理の出力は、マクロブロックパーティション mbPartIdx に対する、そのマクロブロック左上端サン

プル相対の、左上端輝度サンプルの位置 (x,y) である。

逆マクロブロックパーティション走査処理は、次のように規定される。

$$x=\text{InverseRasterScan}(\text{mbPartIdx},\text{MbPartWidth}(\text{mb\_type}),\text{MbPartHeight}(\text{mb\_type}),16,0) \quad (6-11)$$

$$y=\text{InverseRasterScan}(\text{mbPartIdx},\text{MbPartWidth}(\text{mb\_type}),\text{MbPartHeight}(\text{mb\_type}),16,1) \quad (6-12)$$

#### 6.4.2.2 逆サブマクロブロックパーティション走査処理

この処理への入力は、マクロブロックパーティションのインデックス `mbPartIdx`、およびサブマクロブロックパーティションのインデックス `subMbPartIdx` である。

この処理の出力は、サブマクロブロックパーティション `subMbPartIdx` に対する、サブマクロブロック左上端サンプル相対の左上端輝度サンプル位置 (x,y) である。

逆サブマクロブロックパーティション走査処理は、次の通り規定される。

- もし `mb_type` が `P_8x8`、`P_8x8ref0`、あるいは `B_8x8` に等しいならば、

$$x=\text{InverseRasterScan}(\text{subMbPartIdx},\text{SubMbPartWidth}(\text{sub\_mb\_type}[\text{mbPartIdx}]), \\ \text{SubMbPartHeight}(\text{sub\_mb\_type}[\text{mbPartIdx}]),8,0) \quad (6-13)$$

$$y=\text{InverseRasterScan}(\text{subMbPartIdx},\text{SubMbPartWidth}(\text{sub\_mb\_type}[\text{mbPartIdx}]), \\ \text{SubMbPartHeight}(\text{sub\_mb\_type}[\text{mbPartIdx}]),8,1) \quad (6-14)$$

- それ以外、

$$x=\text{InverseRasterScan}(\text{subMbPartIdx},4,4,8,0) \quad (6-15)$$

$$y=\text{InverseRasterScan}(\text{subMbPartIdx},4,4,8,1) \quad (6-16)$$

#### 6.4.3 逆4×4輝度ブロック走査処理

この処理への入力は、4×4輝度ブロックのインデックス `luma4x4BlkIdx` である。

この処理の出力は、インデックス `luma4x4BlkIdx` を持つ4×4輝度ブロックに対する、マクロブロックの左上端輝度サンプル相対の左上端輝度サンプル位置 (x,y) である。

図 6-10/JT-H264 は、4×4輝度ブロックにおける走査を示す。

0	1	4	5
2	3	6	7
8	9	12	13
10	11	14	15

図6-10/JT-H264 4×4輝度ブロックにおける走査  
(ITU-T H.264)

逆4×4輝度ブロック走査処理は、次のように規定される。

$$x = \text{InverseRasterScan}(\text{luma4x4BlkIdx}/4, 8, 8, 16, 0) + \text{InverseRasterScan}(\text{luma4x4BlkIdx}\%4, 4, 4, 8, 0) \quad (6-17)$$

$$y = \text{InverseRasterScan}(\text{luma4x4BlkIdx}/4, 8, 8, 16, 1) + \text{InverseRasterScan}(\text{luma4x4BlkIdx}\%4, 4, 4, 8, 1) \quad (6-18)$$

#### 6.4.4 3に等しいChromaArrayTypeに対する逆4×4CbまたはCrブロック走査処理

この処理はChromaArrayTypeが3に等しい時のみ起動される。

逆4x4色差ブロック走査処理は、6.4.3小節の全箇所において用語「輝度」を用語「Cb」または「Cr」に、用語「luma4x4BlkIdx」を用語「cb4x4BlkIdx」または用語「cr4x4BlkIdx」に置き換えた時、6.4.3小節に規定された逆4×4輝度ブロック走査処理と同一である。

#### 6.4.5 逆8×8輝度ブロック走査処理

この処理への入力は、8×8輝度ブロックのインデックス luma8x8BlkIdx である。

この処理の出力は、インデックス luma8x8BlkIdx を持つ8×8輝度ブロックに対する、マクロブロックの左上端輝度サンプル相対の左上端輝度サンプル位置 (x,y) である。

図6-11/JT-H264は、8×8輝度ブロックにおける走査を示す。

0	1
2	3

図6-11/JT-H264 8×8輝度ブロックにおける走査  
(ITU-T H.264)

逆8×8輝度ブロック走査処理は、次のように規定される。

$$x = \text{InverseRasterScan}(\text{luma8x8BlkIdx}, 8, 8, 16, 0) \quad (6-19)$$

#### 6.4.6 3に等しいChromaArrayTypeに対する逆8×8CbまたはCrブロック走査処理

この処理はChromaArrayTypeが3に等しい時のみ起動される。

逆8x8色差ブロック走査処理は、6.4.5小節の全箇所において用語「輝度」を用語「Cb」または「Cr」に、用語「luma8x8BlkIdx」を用語「cb8x8BlkIdx」または用語「cr8x8BlkIdx」に置き換えた時、6.4.5小節に規定された逆8x8輝度ブロック走査処理と同一である。

#### 6.4.7 マクロブロックアドレスに対する利用可能性の導出処理

この処理への入力は、マクロブロックアドレス mbAddr である。

この処理の出力は、マクロブロック mbAddr の利用可能性である。

記 この処理が起動される時、利用可能性の意味が決定される。

次の条件の1つが真の時マクロブロックは利用不可としてマークされ、それ以外マクロブロックは利用可能としてマークされる：

- mbAddr<0
- mbAddr>CurrMbAddr
- アドレス mbAddr を持つマクロブロックは、アドレス CurrMbAddr を持つマクロブロックとは異なるスライスに属する

#### 6.4.8 隣接マクロブロックアドレスおよびそれらの利用可能性の導出処理

MbaffFrameFlag が 0 に等しい時のみ、この処理は起動することができる。

この処理の出力は、次の通りである。

- mbAddrA：現マクロブロックの左のマクロブロックのアドレスおよび利用可能性ステータス。
- mbAddrB：現マクロブロックの上のマクロブロックのアドレスおよび利用可能性ステータス。
- mbAddrC：現マクロブロックの右上のマクロブロックのアドレスおよび利用可能性ステータス。
- mbAddrD：現マクロブロックの左上のマクロブロックのアドレスおよび利用可能性ステータス。

図 6-12/JT-H264 は、CurrMbAddr を持つ現マクロブロックに関する mbAddrA、mbAddrB、mbAddrC、および mbAddrD を持つマクロブロックの空間的な相対位置を示す。

mbAddrD	mbAddrB	mbAddrC
mbAddrA	CurrMbAddr	

図6-12/JT-H264 与えられたマクロブロックに対する隣接マクロブロック  
(ITU-T H.264)

6.4.7 小節における処理への入力は、 $mbAddrA = CurrMbAddr - 1$  であり、出力は、マクロブロック  $mbAddrA$  が利用可能かどうかである。さらに、 $CurrMbAddr \% PicWidthInMbs$  が 0 に等しい時、 $mbAddrA$  は利用不可としてマークされる。

6.4.7 小節における処理への入力は、 $mbAddrB = CurrMbAddr - PicWidthInMbs$  であり、出力は、マクロブロック  $mbAddrB$  が利用可能かどうかである。

6.4.7 小節における処理への入力は、 $mbAddrC = CurrMbAddr - PicWidthInMbs + 1$  であり、出力は、マクロブロック  $mbAddrC$  が利用可能かどうかである。さらに、 $(CurrMbAddr + 1) \% PicWidthInMbs$  が 0 に等しい時、 $mbAddrC$  は利用不可としてマークされる。

6.4.7 小節における処理への入力は、 $mbAddrD = CurrMbAddr - PicWidthInMbs - 1$  であり、出力は、マクロブロック  $mbAddrD$  が利用可能かどうかである。さらに、 $CurrMbAddr \% PicWidthInMbs$  が 0 に等しい時、 $mbAddrD$  は利用不可としてマークされる。

#### 6.4.9 MBAFFフレームにおける隣接マクロブロックアドレスおよびそれらの利用可能性の導出処理

$MbaffFrameFlag$  が 1 に等しい時のみ、この処理は起動することができる。

この処理の出力は

- $mbAddrA$  : 現マクロブロックペアの左のマクロブロックペアのトップマクロブロックのアドレスおよび利用可能性ステータス。
- $mbAddrB$  : 現マクロブロックペアの上のマクロブロックペアのトップマクロブロックのアドレスおよび利用可能性ステータス。
- $mbAddrC$  : 現マクロブロックペアの右上のマクロブロックペアのトップマクロブロックのアドレスおよび利用可能性ステータス。
- $mbAddrD$  : 現マクロブロックペアの左上のマクロブロックペアのトップマクロブロックのアドレスおよび利用可能性ステータス。

図 6-13/JT-H264 は、 $CurrMbAddr$  を持つ現マクロブロックに関する  $mbAddrA$ 、 $mbAddrB$ 、 $mbAddrC$ 、および  $mbAddrD$  を持つマクロブロックの空間的な相対位置を示す。

mbAddrA、mbAddrB、mbAddrC、および mbAddrD は、現マクロブロックがマクロブロックペアのトップマクロブロックであるかボトムマクロブロックであるかに関係なく、同一の値を持つ。

mbAddrD	mbAddrB	mbAddrC
mbAddrA	CurrMbAddr または	
	CurrMbAddr	

図6-13/JT-H264 MBAFFフレームにおける与えられたマクロブロックに対する隣接マクロブロック (ITU-T H.264)

6.4.7 小節における処理への入力は、 $mbAddrA=2*(CurrMbAddr/2-1)$ であり、出力はマクロブロック mbAddrA が利用可能かどうかである。さらに、 $(CurrMbAddr/2)\%PicWidthInMbs$  が 0 に等しい時、mbAddrA は利用不可としてマークされる。

6.4.7 小節における処理への入力は、 $mbAddrB=2*(CurrMbAddr/2-PicWidthInMbs)$ であり、出力はマクロブロック mbAddrB が利用可能かどうかである。

6.4.7 小節における処理への入力は、 $mbAddrC=2*(CurrMbAddr/2-PicWidthInMbs+1)$ であり、出力はマクロブロック mbAddrC が利用可能かどうかである。さらに、 $(CurrMbAddr/2+1)\%PicWidthInMbs$  が 0 に等しい時、mbAddrC は利用不可としてマークされる。

6.4.7 小節における処理への入力は、 $mbAddrD=2*(CurrMbAddr/2-PicWidthInMbs-1)$ であり、出力はマクロブロック mbAddrD が利用可能かどうかである。さらに、 $(CurrMbAddr/2)\%PicWidthInMbs$  が 0 に等しい時、mbAddrD は利用不可としてマークされる。

#### 6.4.10 隣接マクロブロック、ブロック、およびパーティションのための導出処理

6.4.10.1 小小節は、隣接マクロブロックのための導出処理を規定する。

6.4.10.2 小小節は、隣接 8×8 輝度ブロックのための導出処理を規定する。

6.4.10.3 小小節は、3 に等しい ChromaArrayType に対する隣接 8×8 色差ブロックのための導出処理を規定する。

6.4.10.4 小小節は、隣接 4×4 輝度ブロックのための導出処理を規定する。

6.4.10.5 小小節は、隣接 4×4 色差ブロックのための導出処理を規定する。

6.4.10.6 小小節は、3 に等しい ChromaArrayType に対する隣接 4×4 色差ブロックのための導出処理を規定する。

6.4.10.7 小小節は、隣接パーティションのための導出処理を規定する。

表 6-2/JT-H264 は、出力 mbAddrN、mbPartIdxN、subMbPartIdxN、luma8x8BlkIdxN、cb8x8BlkIdxN、cr8x8BlkIdxN、luma4x4BlkIdxN、cb4x4BlkIdxN、cr4x4BlkIdxN、および chroma4x4BlkIdxN の N の置き換えと、入力である輝度位置の差分値 (xD,yD) を規定する。これら入出力の割り当ては、6.4.10.1 小小節から 6.4.10.7 小小節において使用される。表 6-2/JT-H264 が参照される時、変数 predPartWidth が規定される。

表6-2/JT-H264 6.4.10.1小小節から6.4.10.7小小節のための入出力割り当ての規定  
(ITU-T H.264)

N	xD	yD
A	-1	0
B	0	-1
C	predPartWidth	-1
D	-1	-1

現マクロブロック、パーティションまたはブロックがフレーム符号化モードの時、図 6-14/JT-H264 は現マクロブロック、パーティションまたはブロックに対する、隣接マクロブロック、ブロック、あるいはパーティション A、B、C、および D の相対位置を示す。

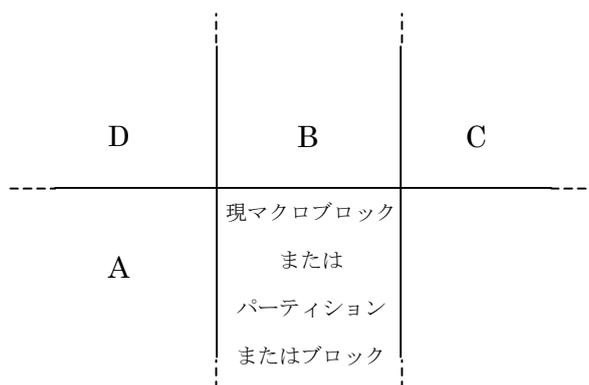


図6-14/JT-H264 隣接マクロブロック、ブロックおよびパーティションの決定 (参考)  
(ITU-T H.264)

#### 6.4.10.1 隣接マクロブロックのための導出処理

この処理の出力は

- mbAddrA : 現マクロブロックの左のマクロブロックのアドレスおよびその利用可能性ステータスと
- mbAddrB : 現マクロブロックの上のマクロブロックのアドレスおよびその利用可能性ステータス。

mbAddrN (N は A または B) は、以下のように導出される。

- 輝度位置の差分 (xD,yD) は、表 6-2/JT-H264 に従って設定される。
- 6.4.11 小節において規定されるような隣接位置の導出処理が、(xD,yD) に等しい (xN,yN) を持つ輝度位置に対して起動され、出力は mbAddrN に割り当てられる。

#### 6.4.10.2 隣接 8×8 輝度ブロックのための導出処理

この処理への入力は、8×8 輝度ブロックインデックス  $luma8x8BlkIdx$  である。

$luma8x8BlkIdx$  は、マクロブロックの 8×8 輝度ブロックをラスタ走査で規定する。

この処理の出力は

- $mbAddrA$  :  $CurrMbAddr$  に等しいか、あるいは現マクロブロックの左側のマクロブロックのアドレスのいずれか、およびその利用可能性ステータス、
- $luma8x8BlkIdxA$  : インデックス  $luma8x8BlkIdx$  を持つその 8×8 ブロックの左側の 8×8 輝度ブロックのインデックス、およびその利用可能性ステータス、
- $mbAddrB$  :  $CurrMbAddr$  に等しいか、あるいは現マクロブロックの上側のマクロブロックのアドレスのいずれか、およびその利用可能性ステータス、
- $luma8x8BlkIdxB$  : インデックス  $luma8x8BlkIdx$  を持つその 8×8 ブロックの上側の 8×8 輝度ブロックのインデックス、およびその利用可能性ステータス。

$mbAddrN$  および  $luma8x8BlkIdxN$  (N は A または B) は、以下のように導出される。

- 輝度位置の差分( $xD,yD$ )は、表 6-2/JT-H264 に従って設定される。
- 輝度位置 ( $xN,yN$ ) は次式から規定される。

$$xN=(luma8x8BlkIdx\%2)*8+xD \quad (6-21)$$

$$yN=(luma8x8BlkIdx/2)*8+yD \quad (6-22)$$

- 6.4.11 小節において規定されるような隣接位置の導出処理が、入力として ( $xN,yN$ ) である輝度位置に対して起動され、出力は  $mbAddrN$  および ( $xW,yW$ ) に割り当てられる。
- 変数  $luma8x8BlkIdxN$  は次の通り導出される。
  - もし  $mbAddrN$  が利用不可ならば、 $luma8x8BlkIdxN$  は利用不可としてマークされる。
  - それ以外 ( $mbAddrN$  は利用可能)、マクロブロック  $mbAddrN$  中の輝度位置 ( $xW,yW$ ) を含む 8×8 輝度ブロックが  $luma8x8BlkIdxN$  に割り当てられる。

#### 6.4.10.3 3に等しいChromaArrayTypeに対する隣接 8×8 色差ブロックのための導出処理

この処理はChromaArrayTypeが3に等しい時のみ起動される。

隣接8x8色差ブロックのための導出処理は、6.4.10.2小節の全箇所において用語「輝度」を用語「Cb」または「Cr」に、用語「 $luma8x8BlkIdx$ 」を用語「 $cb8x8BlkIdx$ 」または用語「 $cr8x8BlkIdx$ 」に置き換えた時、6.4.10.2小節に規定された隣接8x8輝度ブロックのための導出処理と同一である。

#### 6.4.10.4 隣接 4×4 輝度ブロックのための導出処理

この処理への入力は、4×4 輝度ブロックインデックス  $luma4x4BlkIdx$  である。

この処理の出力は、

- **mbAddrA** : CurrMbAddr に等しいか、あるいは現マクロブロックの左側のマクロブロックのアドレスのいずれか、およびその利用可能性ステータス、
- **luma4x4BlkIdxA** : インデックス **luma4x4BlkIdx** を持つその 4×4 ブロックの左側の 4×4 輝度ブロックのインデックス、およびその利用可能性ステータス、
- **mbAddrB** : CurrMbAddr に等しいか、あるいは現マクロブロックの上側のマクロブロックのアドレスのいずれか、およびその利用可能性ステータス、
- **luma4x4BlkIdxB** : インデックス **luma4x4BlkIdx** を持つその 4×4 ブロックの上側の 4×4 輝度ブロックのインデックス、およびその利用可能性ステータス。

**mbAddrN** および **luma4x4BlkIdxN** (N は A または B) は以下のように導出される。

- 輝度位置の差分 (**xD,yD**) は、表 6-2/JT-H264 に従って設定される。
- 6.4.3 小節において規定されるような逆 4×4 輝度ブロック走査処理が、入力として **luma4x4BlkIdx** を、および出力として (**x,y**) を持って起動される。
- 輝度位置 (**xN,yN**) は次式から規定される。

$$xN=x+xD \quad (6-23)$$

$$yN=y+yD \quad (6-24)$$

- 6.4.11 小節において規定されるような隣接位置の導出処理が、輝度位置に対して (**xN,yN**) を入力として起動され、出力は **mbAddrN** および (**xW,yW**) に割り当てられる。
- 変数 **luma4x4BlkIdxN** は次の通り導出される。
  - もし **mbAddrN** が利用不可ならば、**luma4x4BlkIdxN** は利用不可としてマークされる。
  - それ以外 (**mbAddrN** は利用可能)、マクロブロック **mbAddrN** 中の輝度位置 (**xW,yW**) を含む 4×4 輝度ブロックが、**luma4x4BlkIdxN** に割り当てられる。

#### 6.4.10.5 隣接 4×4 色差ブロックのための導出処理

この処理への入力は、4×4 色差ブロックインデックス **chroma4x4BlkIdx** である。

この処理の出力は、

- **mbAddrA** (CurrMbAddr に等しいか、あるいは現マクロブロックの左側のマクロブロックのアドレスのいずれか) およびその利用可能性ステータス、
- **chroma4x4BlkIdxA** (インデックス **chroma4x4BlkIdx** を持つその 4×4 色差ブロックの左側の 4×4 色差ブロックのインデックス) およびその利用可能性ステータス、
- **mbAddrB** (CurrMbAddr に等しいか、あるいは現マクロブロックの上側のマクロブロックのアドレスのいずれか) およびその利用可能性ステータス、
- **chroma4x4BlkIdxB** (インデックス **chroma4x4BlkIdx** を持つその 4×4 色差ブロックの上側の 4×4 色差ブロックのインデックス) およびその利用可能性ステータス。

mbAddrN と chroma4x4BlkIdxN (N は A または B) は以下のように導出される。

- 色差位置の差分 (xD,yD) は表 6-2/JT-H264 に従って設定される。
- インデックス chroma4x4BlkIdx を持つその 4×4 色差ブロックの左上端サンプルの位置(x,y)は、次の通り導出される。

$$x = \text{InverseRasterScan}(\text{chroma4x4BlkIdx}, 4, 4, 8, 0) \quad (6-25)$$

$$y = \text{InverseRasterScan}(\text{chroma4x4BlkIdx}, 4, 4, 8, 1) \quad (6-26)$$

- 色差位置(xN,yN)は以下で規定される。

$$xN = x + xD \quad (6-27)$$

$$yN = y + yD \quad (6-28)$$

- 6.4.11 小節において規定されるような隣接位置の導出処理が、色差位置に対して (xN,yN) を入力として起動され、出力は mbAddrN および (xW,yW) に割り当てられる。
- 変数 chroma4x4BlkIdxN は次の通り導出される。
  - もし mbAddrN が利用不可ならば、chroma4x4BlkIdxN は利用不可としてマークされる。
  - それ以外 (mbAddrN は利用可能)、マクロブロック mbAddrN 中の色差位置 (xW,yW) を含む 4×4 色差ブロックが、chroma4x4BlkIdxN に割り当てられる。

#### 6.4.10.6 3に等しいChromaArrayTypeに対する隣接8×8色差ブロックのための導出処理

この処理はChromaArrayTypeが3に等しい時のみ起動される。

4:4:4色差フォーマットの隣接4x4色差ブロックのための導出処理は、6.4.10.4小節の全箇所において用語「輝度」を用語「Cb」または「Cr」に、用語「luma4x4BlkIdx」を用語「cb4x4BlkIdx」または用語「cr4x4BlkIdx」に置き換えた時、6.4.10.4小節に規定された隣接4x4輝度ブロックのための導出処理と同一である。

#### 6.4.10.7 隣接パーティションのための導出処理

この処理への入力は、

- マクロブロックパーティションインデックス mbPartIdx
- 現サブマクロブロックタイプ currSubMbType
- サブマクロブロックパーティションインデックス subMbPartIdx

この処理の出力は、

- mbAddrA¥mbPartIdxA¥subMbPartIdxA : 現マクロブロックの左側の、マクロブロックまたはサブマクロブロックパーティションおよびその利用可能性ステータス、またはサブマクロブロックパーティション CurrMbAddr¥mbPartIdx¥subMbPartIdx およびその利用可能性ステータスを規定、
- mbAddrB¥mbPartIdxB¥subMbPartIdxB : 現マクロブロックの上側の、マクロブロックまたはサブマクロブ

ロックパーティションおよびその利用可能性ステータス、またはサブマクロブロックパーティション CurrMbAddr¥mbPartIdx¥subMbPartIdx およびその利用可能性ステータスを規定、

- mbAddrC¥mbPartIdxC¥subMbPartIdxC : 現マクロブロックの右上側の、マクロブロックまたはサブマクロブロックパーティションおよびその利用可能性ステータス、またはサブマクロブロックパーティション CurrMbAddr¥mbPartIdx¥subMbPartIdx およびその利用可能性ステータスを規定、
- mbAddrD¥mbPartIdxD¥subMbPartIdxD : 現マクロブロックの左上側の、マクロブロックまたはサブマクロブロックパーティションおよびその利用可能性ステータス、またはサブマクロブロックパーティション CurrMbAddr¥mbPartIdx¥subMbPartIdx およびその利用可能性ステータスを規定。

mbAddrN、mbPartIdxN および subMbPartIdxN (N は A、B、C または D) は以下のように導出される。

- 6.4.2.1 小節に記述されるような逆マクロブロックパーティション走査処理が、入力として mbPartIdx、および出力として (x,y) を持ち起動される。
- マクロブロックパーティション内部の、その左上端輝度サンプル位置 (xS,yS) は、次の通り導出される。
  - もし mb\_type が P\_8x8、P\_8x8ref0 あるいは B\_8x8 に等しいならば、6.4.2.2 小節において記述されるような逆サブマクロブロックパーティション走査処理が、入力として subMbPartIdx を、および出力として (xS,yS) を持ち起動される。
  - それ以外、(xS,yS) は (0,0) に設定される。
- 表 6-2/JT-H264 における変数 predPartWidth は、次の通り規定される。
  - もし mb\_type が P\_Skip、B\_Skip、あるいは B\_Direct\_16x16 に等しいならば、predPartWidth=16 となる。
  - それ以外、もし mb\_type が B\_8x8 に等しいならば、次が適用される。
    - もし currSubMbType が B\_Direct\_8x8 に等しいならば、predPartWidth=16 となる。  
 記1 currSubMbTypeがB\_Direct\_8x8と等しく、かつdirect\_spatial\_mv\_pred\_flagが1に等しい時、その予測された動きベクトルはそのマクロブロック全体に対して予測された動きベクトルになる。
    - それ以外、predPartWidth=SubMbPartWidth(sub\_mb\_type[mbPartIdx])となる。
  - それ以外、もし mb\_type が P\_8x8 または P\_8x8ref0 と等しいならば、predPartWidth=SubMbPartWidth(sub\_mb\_type[mbPartIdx])となる。
  - それ以外、predPartWidth=MbPartWidth(mb\_type)となる。
- 輝度位置の差分 (xD,yD) は、表 6-2/JT-H264 に従って設定される。
- 隣接輝度位置 (xN,yN) は、次式により規定される。

$$xN=x+xS+xD \tag{6-29}$$

$$yN=y+yS+yD \tag{6-30}$$

- 6.4.11 小節に規定されるような隣接位置の導出処理が、輝度位置に対して (xN,yN) を入力として起動され、出力が mbAddrN および (xW,yW) に割り当てられる。

- mbAddrN に従って、次が適用される。
  - もし mbAddrN が利用不可ならば、マクロブロックまたはサブマクロブロックパーティション mbAddrN¥mbPartIdxN¥subMbPartIdxN は利用不可としてマークされる。
  - それ以外 (mbAddrN は利用可能)、次が適用される。
    - 輝度位置 (xW,yW) を含むマクロブロック mbAddrN におけるマクロブロックパーティションが、mbPartIdxN に割り当てられ、そのマクロブロック mbAddrN においてサンプル(xW,yW) を含むマクロブロックパーティション mbPartIdxN の内部のサブマクロブロックパーティションが subMbPartIdxN に割り当てられる。
    - mbPartIdxN および subMbPartIdxN によって与えられるパーティションがまだ復号されていない時、マクロブロックパーティション mbPartIdxN およびサブマクロブロックパーティション subMbPartIdxN は利用不可としてマークされる。

記2 後者の条件は、例えば、mbPartIdx=2、subMbPartIdx=3、xD=4、yD=-1である場合、すなわち、第3のサブマクロブロックの最後の4×4輝度ブロックの隣接Cが要求される時である。

#### 6.4.11 隣接位置の導出処理

この処理への入力は、現マクロブロックの左上隅相対で表現された輝度または色差の位置 (xN,yN) である。

この処理の出力は、

- mbAddrN : CurrMbAddr、あるいは (xN,yN) を含む隣接マクロブロックのアドレスのいずれかに等しい、およびその利用可能性ステータス、
- (xW,yW) : (現マクロブロックの左上隅相対ではなく) マクロブロック mbAddrN の左上隅相対で表現された位置 (xN,yN)。

maxW と maxH を、それぞれ位置成分 xN、xW および yN、yW の最大値を規定する変数とする。maxW と maxH は、次の通り導出される。

- もしこの処理が隣接輝度位置のために起動されるならば、

$$\text{maxW}=\text{maxH}=16 \quad (6-31)$$

- それ以外 (この処理が隣接色差位置のために起動される)、

$$\text{maxW}=\text{MbWidthC} \quad (6-32)$$

$$\text{maxH}=\text{MbHeightC} \quad (6-33)$$

変数 MbaffFrameFlag に従って、隣接位置は次の通り導出される。

- もし MbaffFrameFlag が 0 に等しいならば、6.4.11.1 小節で述べられるようなフィールドおよび非 MBAFF フレームにおける隣接位置に対する規定が適用される。
- それ以外 (MbaffFrameFlag が 1 に等しい)、6.4.11.2 小節で述べられるような MBAFF フレームにおける隣接位置に対する規定が適用される。

#### 6.4.11.1 フィールドおよび非MBAFFフレームにおける隣接位置に対する規定

MbaffFrameFlag が 0 に等しい場合、この小節における規定が適用される。

6.4.8 小節の隣接マクロブロックアドレスおよびそれらの利用可能性のための導出処理が、出力として mbAddrA、mbAddrB、mbAddrC および mbAddrD、および各々の利用可能性ステータスをもち起動される。

表 6-3/JT-H264 は、(xN,yN) に従って mbAddrN を規定する。

表6-3/JT-H264 mbAddrNの規定  
(ITU-T H.264)

xN	yN	mbAddrN
<0	<0	mbAddrD
<0	0..maxH-1	mbAddrA
0..maxW-1	<0	mbAddrB
0..maxW-1	0..maxH-1	CurrMbAddr
>maxW-1	<0	mbAddrC
>maxW-1	0..maxH-1	利用不可
	>maxH-1	利用不可

マクロブロック mbAddrN の左上隅相対の隣接位置 (xW,yW) は、次のように導出される。

$$xW=(xN+\max W)\% \max W \quad (6-34)$$

$$yW=(yN+\max H)\% \max H \quad (6-35)$$

#### 6.4.11.2 MBAFFフレームにおける隣接位置に対する規定

MbaffFrameFlag が 1 に等しい場合、この小節における規定が適用される。

6.4.9 小節における隣接マクロブロックアドレスおよびそれらの利用可能性のための導出処理が、出力として mbAddrA、mbAddrB、mbAddrC および mbAddrD、および、各々の利用可能性ステータスをもち起動される。

表 6-4/JT-H264 は、2つの順序付けられたステップによりマクロブロックアドレス mbAddrN と yM を規定する：

1. (xN,yN) および以下に示す変数に従うマクロブロックアドレス mbAddrX の規定：

- 変数 currMbFrameFlag は次の通り導出される。
- もしアドレス CurrMbAddr を持つマクロブロックがフレームマクロブロックならば、currMbFrameFlag は 1 に等しく設定され、
- それ以外 (アドレス CurrMbAddr を持つマクロブロックがフィールドマクロブロックである)、currMbFrameFlag は 0 に等しく設定される。

- 変数 `mbIsTopMbFlag` は次の通り導出される。
    - もしアドレス `CurrMbAddr` を持つマクロブロックがトップマクロブロックである (`CurrMbAddr%2` が 0 に等しい) ならば、`mbIsTopMbFlag` は 1 に等しく設定され、
    - それ以外 (アドレス `CurrMbAddr` を持つマクロブロックがボトムマクロブロックである。すなわち `CurrMbAddr%2` が 1 に等しい)、`mbIsTopMbFlag` は 0 に等しく設定される。
2. `mbAddrX` の利用可能性に従って、次が適用される。
- もし `mbAddrX` が利用不可であるならば、`mbAddrN` は利用不可としてマークされる。
  - それ以外 (`mbAddrX` は利用可能)、`mbAddrN` は利用可能としてマークされ、表 6-4/JT-H264 は (`xN,yN`)、`currMbFrameFlag`、`mbIsTopMbFlag` および、次の通り導出される変数 `mbAddrXFrameFlag` に従って `mbAddrN` および `yM` を規定する。
    - もし `mbAddrX` がフレームマクロブロックであるならば、`mbAddrXFrameFlag` は 1 に等しく設定され、
    - それ以外 (マクロブロック `mbAddrX` がフィールドマクロブロックである)、`mbAddrXFrameFlag` は 0 に等しく設定される。

表 6-4/JT-H264 における上記フラグの無規定値 (na) は、対応するフラグの値が、その行には適切ではないことを示す。

表6-4/JT-H264 mbAddrNおよびyMの規定

(ITU-T H.264)

$xN$	$yN$	currMbFrameFlag	mbIsTopMbFlag	mbAddrX	mbAddrXFrameFlag	追加条件	mbAddrN	$yM$
<0	<0	1	1	mbAddrD			mbAddrD+1	$yN$
			0	mbAddrA	1		mbAddrA	$yN$
		0	1	mbAddrD	1		mbAddrD+1	$(yN+maxH)>>1$
			0	mbAddrD	0		mbAddrD	$2*yN$
<0	0..maxH-1	1	1	mbAddrA	1		mbAddrA	$yN$
					0	$yN\%2==0$	mbAddrA	$yN>>1$
					0	$yN\%2!=0$	mbAddrA+1	$yN>>1$
			0	mbAddrA	1		mbAddrA+1	$yN$
					0	$yN\%2==0$	mbAddrA	$(yN+maxH)>>1$
					0	$yN\%2!=0$	mbAddrA+1	$(yN+maxH)>>1$
		0	1	mbAddrA	1	$yN<(maxH/2)$	mbAddrA	$yN<<1$
					1	$yN\geq(maxH/2)$	mbAddrA+1	$(yN<<1)-maxH$
					0		mbAddrA	$yN$
			0	mbAddrA	1	$yN<(maxH/2)$	mbAddrA	$(yN<<1)+1$
					1	$yN\geq(maxH/2)$	mbAddrA+1	$(yN<<1)+1-maxH$
					0		mbAddrA+1	$yN$
0..maxW-1	<0	1	1	mbAddrB			mbAddrB+1	$yN$
			0	CurrMbAddr			CurrMbAddr-1	$yN$
		0	1	mbAddrB	1		mbAddrB+1	$2*yN$
			0	mbAddrB	0		mbAddrB	$yN$
0..maxW-1	0..maxH-1			CurrMbAddr		CurrMbAddr	$yN$	
>maxW-1	<0	1	1	mbAddrC			mbAddrC+1	$yN$
			0	利用不可			利用不可	na
		0	1	mbAddrC	1		mbAddrC+1	$2*yN$
			0	mbAddrC	0		mbAddrC	$yN$
>maxW-1	0..maxH-1			利用不可		利用不可	na	

	>maxH-1		利用不可		利用不可	na
--	---------	--	------	--	------	----

マクロブロック mbAddrN の左上隅相対の隣接輝度位置 (xW,yW) は、次のように導出される。

$$xW=(xN+maxW)\%maxW \quad (6-36)$$

$$yW=(yM+maxH)\%maxH \quad (6-37)$$

## 7 シンタックスおよび意味

### 7.1 表形式におけるシンタックス規定の手法

シンタックス表は、全ての許可されたビットストリームのシンタックスのスーパーセットを規定する。シンタックスに対する追加制約が他の節において、直接にあるいは間接に規定されるかもしれない。

記 実際の復号器は、ビットストリームへのエントリ点を識別するための手段、および規定に従わないビットストリームを識別し扱うための手段を実装すべきである。エラーおよび他のそのような状況を識別し扱う方法はここでは規定されない。

次の表は、シンタックスを記述するために使われる擬似コードの例をリストする。**syntax\_element** が現われる時、それはビットストリーム構文解析処理で、シンタックス要素がビットストリームから構文解析され、ビットストリームポインタがシンタックス要素を超えて次の位置まで進められることを規定する。

	C	記述子
/*次の2例のように、ステートメントは関連するシンタックスカテゴリおよび記述子を備えたシンタックス要素であり得るか、あるいはシンタックス要素の存在、タイプおよび量のための条件を規定するために使用される表現であり得る。*/		
<b>syntax_element</b>	3	ue(v)
コンディショニングステートメント		
/*中括弧で囲まれたステートメントのグループは複合ステートメントであり、そのステートメントは単一のステートメントとして機能的に扱われる。*/		
{		
ステートメント		
ステートメント		
...		
}		
/* “while” 構造は、条件が真かどうかのテストを規定し、真の場合、条件がもはや真でなくなるまで、ステートメント（あるいは複合ステートメント）の繰り返し実行を規定する。*/		
while(条件)		

ステートメント		
/* “do…while” 構造は、条件が真かどうかのテストが後に続く、ステートメント1回の実行を規定し、真の場合、条件がもはや真実ではなくなるまで、ステートメントの繰り返し実行を規定する。*/		
do		
ステートメント		
while(条件)		
/* “if…else” 構造は、条件が真かどうかのテストを規定し、条件が真なら、主要ステートメントの実行を規定し、そうでなければ代替ステートメントの実行を規定する。構造の “else” パートおよび関連する代替ステートメントは、代替ステートメントの実行が必要ない場合、省略される。*/		
if(条件)		
主要ステートメント		
else		
代替ステートメント		
/* “for” 構造は、条件のテストが後に続く、初期ステートメントの実行を規定し、条件が真の場合、条件がもはや真でなくなるまで、後続ステートメントが後ろに続く主要ステートメントの繰り返し実行が続くことを規定する。*/		
for(初期ステートメント;条件;後続ステートメント)		
主要ステートメント		

## 7.2 シンタックス関数、カテゴリ、記述子の規定

ここに示された関数は、構文的な記述において使用される。これらの関数は、復号処理によってビットストリームから読まれる次のビットの位置指示を備えたビットストリームポインタの存在を仮定する。

`byte_aligned()`は次の通り規定される：

- もしビットストリームにおける現在の位置がバイト境界上にあるならば、つまりビットストリーム内の次のビットがあるバイト内の最初のビットの時、`byte_aligned()`の戻り値は TRUE となる。
- それ以外、`byte_aligned()`の戻り値は FALSE となる。

付属資料 B で規定されるバイトストリーム NAL ユニットシンタックス構造においてのみ使用される `more_data_in_byte_stream()`は、次の通り規定される：

- もしバイトストリームに更にデータが続くならば、`more_data_in_byte_stream()`の戻り値は TRUE となる。
- それ以外、`more_data_in_byte_stream()`の戻り値は FALSE となる。

`more_rbsp_data()`は次の通り規定される：

- もし `rbsp_trailing_bits()`の前の RBSP にさらにデータがあるならば、`more_rbsp_data()`の戻り値は TRUE となる。
- それ以外、`more_rbsp_data()`の戻り値は FALSE となる。

さらに多くのデータが RBSP にあるかどうかの決定を可能にする手法は、アプリケーションによって（あるいは付属資料 B においてはバイトストリームフォーマットを使用するアプリケーションにおいて）規定される。

`more_rbsp_trailing_data()`は次の通り規定される：

- もし RBSP 内にさらにデータが存在するならば、`more_rbsp_trailing_data()`の戻り値は TRUE となる。
- それ以外、`more_rbsp_trailing_data()`の戻り値は FALSE となる。

`next_bits(n)`はビットストリームポインタを進めずに、比較目的にビットストリーム内の次の（複数）ビットを与える。その引数 `n` で、ビットストリームにおける次の `n` ビットへの注視を与える。付属資料 B の中で規定されるようなバイトストリーム内で使用される場合、バイトストリーム内に残ったビットが `n` ビット未満であるならば `next_bits(n)`は 0 の値を返す。

`read_bits(n)`はビットストリームから次の `n` ビットを読み、`n` ビット位置だけビットストリームポインタを進める。`n` が 0 に等しい場合、`read_bits(n)`は 0 に等しい値を返し、かつビットストリームポインタを進めないとして規定される。

カテゴリ（表では C として表記される）は、多くとも 3 つのスライスデータパーティションへのスライスデータのパーティションを規定する。スライスデータパーティション A は、カテゴリ 2 の全てのシンタックス要素を含む。スライスデータパーティション B は、カテゴリ 3 の全てのシンタックス要素を含む。スライスデータパーティション C は、カテゴリ 4 の全てのシンタックス要素を含む。他のカテゴリ値の意味は規定されない。いくつかのシンタックス要素においては、縦線で分けられた 2 つのカテゴリ値が用いられる。これらの場合において、適用されるべきカテゴリ値が、文書中でさらに規定される。他のシンタックス構造内にて使用されるシンタックス構造については、含まれたシンタックス構造内に見つかった全てのシンタックス要素のカテゴリが、縦線によって分離されてリストされる。“All”としてマークされたカテゴリを備えたシンタックス要素またはシンタックス構造は、そのシンタックス要素あるいはシンタックス構造を含む全てのシンタックス構造内に存在する。他のシンタックス構造内に使用されるシンタックス構造については、“All”としてマークされたカテゴリを備えたシンタックス要素を含むシンタックス構造の包含の位置で、シンタックス表の中で提供される数値のカテゴリ値が、そのカテゴリ “All”を備えたシンタックス要素に適用されると考えられる。

次の記述子は、各シンタックス要素の構文解析処理を規定する。いくつかのシンタックス要素に対しては、縦線によって分離された 2 つの記述子が使用される。これらの場合は、`entropy_coding_mode_flag` が 0 に等しい時は左の記述子を適用し、`entropy_coding_mode_flag` が 1 に等しい時は右の記述子を適用する。

- `ac(v)`：コンテキスト適応算術エントロピ符号化シンタックス要素。この記述子に対する構文解析処理は 9.3 節で規定される。
- `b(8)`：任意のパターンのビット記号列を持つバイト（8 ビット）。この記述子に対する構文解析処理は、関数 `read_bits(8)`が返す値によって規定される。

- **ce(v)**: コンテキスト適応可変長エントロピ符号化された、左のビットから始まるシンタックス要素。この記述子に対する解析処理は 9.2 節で規定される。
- **f(n)**: 左のビットから (左から右へ) 書かれた  $n$  ビットを使った固定パターンのビット記号列。この記述子に対する構文解析処理は、関数 `read_bits(n)` が返す値によって規定される。
- **i(n)**:  $n$  ビットを使った符号付き整数。シンタックス表において  $n$  が “v” の場合、ビットの数は、他のシンタックス要素の値に従う方法によって異なる。この記述子に対する構文解析処理は、最初に書かれたビットを最上位ビットとする 2 の補数による整数表現として解釈された、関数 `read_bits(n)` が返す値によって規定される。
- **me(v)**: 左のビットから始まる、マップされた指数 Golomb 符号シンタックス要素。この記述子に対する構文解析処理は 9.1 節で規定される。
- **se(v)**: 左のビットから始まる、符号付き整数指数 Golomb 符号シンタックス要素。この記述子に対する構文解析処理は 9.1 節で規定される。
- **te(v)**: 左のビットから始まる、切り詰め指数 Golomb 符号シンタックス要素。この記述子に対する構文解析処理は 9.1 節で規定される。
- **u(n)**:  $n$  ビットを使った符号無し整数。シンタックス表において  $n$  が “v” である場合、ビットの数は、他のシンタックス要素の値に依存する方法によって異なる。この記述子に対する構文解析処理は、最上位ビットから書かれた、符号無しの整数のバイナリ表現として解釈された、関数 `read_bits(n)` が返す値によって規定される。
- **ue(v)**: 左のビットから始まる、符号無し整数指数 Golomb 符号シンタックス要素。この記述子のための構文解析処理は 9.1 節で規定される。

### 7.3 表形式におけるシンタックス

#### 7.3.1 NALユニットシンタックス

nal_unit( NumBytesInNALunit ) {	C	記述子
<b>forbidden_zero_bit</b>	All	f(1)
<b>nal_ref_idc</b>	All	u(2)
<b>nal_unit_type</b>	All	u(5)
NumBytesInRBSP = 0		
nalUnitHeaderBytes = 1		
if( nal_unit_type == 14    nal_unit_type == 20 ) {		
nal_unit_header_svc_extension() /* 付属資料Gにて規定 */		
nalUnitHeaderBytes += 3		
}		
for( i = nalUnitHeaderBytes; i < NumBytesInNALunit; i++ ) {		
if( i + 2 < NumBytesInNALunit && next_bits( 24 ) == 0x000003 ) {		
<b>rbsp_byte</b> [ NumBytesInRBSP++ ]	All	b(8)
<b>rbsp_byte</b> [ NumBytesInRBSP++ ]	All	b(8)
i += 2		
<b>emulation_prevention_three_byte</b> /* 0x03に等しい */	All	f(8)
} else		
<b>rbsp_byte</b> [ NumBytesInRBSP++ ]	All	b(8)
}		
}		

#### 7.3.2 未加工のバイトシーケンスペイロードおよびRBSP追従ビットシンタックス

##### 7.3.2.1 シーケンスパラメータセットRBSPシンタックス

seq_parameter_set_rbsp( ) {	C	記述子
seq_parameter_set_data( )	0	
rbsp_trailing_bits( )	0	
}		

7.3.2.1.1 シーケンスパラメータセットデータシンタックス

seq_parameter_set_data() {	C	記述子
<b>profile_idc</b>	0	u(8)
<b>constraint_set0_flag</b>	0	u(1)
<b>constraint_set1_flag</b>	0	u(1)
<b>constraint_set2_flag</b>	0	u(1)
<b>constraint_set3_flag</b>	0	u(1)
<b>reserved_zero_4bits</b> /* 0に等しい */	0	u(4)
<b>level_idc</b>	0	u(8)
<b>seq_parameter_set_id</b>	0	ue(v)
if( profile_idc == 100    profile_idc == 110    profile_idc == 122    profile_idc == 244    profile_idc == 44    profile_idc == 83    profile_idc == 86 ) {		
<b>chroma_format_idc</b>	0	ue(v)
if( chroma_format_idc == 3 )		
<b>separate_colour_plane_flag</b>	0	u(1)
<b>bit_depth_luma_minus8</b>	0	ue(v)
<b>bit_depth_chroma_minus8</b>	0	ue(v)
<b>qpprime_y_zero_transform_bypass_flag</b>	0	u(1)
<b>seq_scaling_matrix_present_flag</b>	0	u(1)
if( seq_scaling_matrix_present_flag )		
for( i = 0; i < ( ( chroma_format_idc != 3 ) ? 8 : 12 ); i++ ) {		
<b>seq_scaling_list_present_flag[ i ]</b>	0	u(1)
if( seq_scaling_list_present_flag[ i ] )		
if( i < 6 )		
scaling_list( ScalingList4x4[ i ], 16, UseDefaultScalingMatrix4x4Flag[ i ] )	0	
else		
scaling_list( ScalingList8x8[ i - 6 ], 64, UseDefaultScalingMatrix8x8Flag[ i - 6 ] )	0	
}		
}		
<b>log2_max_frame_num_minus4</b>	0	ue(v)
<b>pic_order_cnt_type</b>	0	ue(v)
if( pic_order_cnt_type == 0 )		
<b>log2_max_pic_order_cnt_lsb_minus4</b>	0	ue(v)
else if( pic_order_cnt_type == 1 ) {		
<b>delta_pic_order_always_zero_flag</b>	0	u(1)
<b>offset_for_non_ref_pic</b>	0	se(v)
<b>offset_for_top_to_bottom_field</b>	0	se(v)
<b>num_ref_frames_in_pic_order_cnt_cycle</b>	0	ue(v)

for( i = 0; i < num_ref_frames_in_pic_order_cnt_cycle; i++ )		
<b>offset_for_ref_frame[ i ]</b>	0	se(v)
}		
<b>num_ref_frames</b>	0	ue(v)
<b>gaps_in_frame_num_value_allowed_flag</b>	0	u(1)
<b>pic_width_in_mbs_minus1</b>	0	ue(v)
<b>pic_height_in_map_units_minus1</b>	0	ue(v)
<b>frame_mbs_only_flag</b>	0	u(1)
if( !frame_mbs_only_flag )		
<b>mb_adaptive_frame_field_flag</b>	0	u(1)
<b>direct_8x8_inference_flag</b>	0	u(1)
<b>frame_cropping_flag</b>	0	u(1)
if( frame_cropping_flag ) {		
<b>frame_crop_left_offset</b>	0	ue(v)
<b>frame_crop_right_offset</b>	0	ue(v)
<b>frame_crop_top_offset</b>	0	ue(v)
<b>frame_crop_bottom_offset</b>	0	ue(v)
}		
<b>vui_parameters_present_flag</b>	0	u(1)
if( vui_parameters_present_flag )		
vui_parameters( )	0	
}		

#### 7.3.2.1.1.1 スケーリングリストシンタックス

	C	記述子
scaling_list( scalingList, sizeofScalingList, useDefaultScalingMatrixFlag ) {		
lastScale = 8		
nextScale = 8		
for( j = 0; j < sizeofScalingList; j++ ) {		
if( nextScale != 0 ) {		
<b>delta_scale</b>	0   1	se(v)
nextScale = ( lastScale + delta_scale + 256 ) % 256		
useDefaultScalingMatrixFlag = ( j == 0 && nextScale == 0 )		
}		
scalingList[ j ] = ( nextScale == 0 ) ? lastScale : nextScale		
lastScale = scalingList[ j ]		
}		
}		

### 7.3.2.1.2 シーケンスパラメータセット拡張Rbspシンタックス

seq_parameter_set_extension_rbsp() {	C	記述子
<b>seq_parameter_set_id</b>	10	ue(v)
<b>aux_format_idc</b>	10	ue(v)
if( aux_format_idc != 0 ) {		
<b>bit_depth_aux_minus8</b>	10	ue(v)
<b>alpha_incr_flag</b>	10	u(1)
<b>alpha_opaque_value</b>	10	u(v)
<b>alpha_transparent_value</b>	10	u(v)
}		
<b>additional_extension_flag</b>	10	u(1)
rbsp_trailing_bits()	10	
}		

### 7.3.2.2 ピクチャパラメータセットRbspシンタックス

pic_parameter_set_rbsp() {	C	記述子
<b>pic_parameter_set_id</b>	1	ue(v)
<b>seq_parameter_set_id</b>	1	ue(v)
<b>entropy_coding_mode_flag</b>	1	u(1)
<b>pic_order_present_flag</b>	1	u(1)
<b>num_slice_groups_minus1</b>	1	ue(v)
if( num_slice_groups_minus1 > 0 ) {		
<b>slice_group_map_type</b>	1	ue(v)
if( slice_group_map_type == 0 )		
for( iGroup = 0; iGroup <= num_slice_groups_minus1; iGroup++ )		
<b>run_length_minus1[ iGroup ]</b>	1	ue(v)
else if( slice_group_map_type == 2 )		
for( iGroup = 0; iGroup < num_slice_groups_minus1; iGroup++ ) {		
<b>top_left[ iGroup ]</b>	1	ue(v)
<b>bottom_right[ iGroup ]</b>	1	ue(v)
}		
else if( slice_group_map_type == 3    slice_group_map_type == 4    slice_group_map_type == 5 ) {		
<b>slice_group_change_direction_flag</b>	1	u(1)
<b>slice_group_change_rate_minus1</b>	1	ue(v)
} else if( slice_group_map_type == 6 ) {		
<b>pic_size_in_map_units_minus1</b>	1	ue(v)
for( i = 0; i <= pic_size_in_map_units_minus1; i++ )		
<b>slice_group_id[ i ]</b>	1	u(v)

}		
}		
<b>num_ref_idx_l0_active_minus1</b>	1	ue(v)
<b>num_ref_idx_l1_active_minus1</b>	1	ue(v)
<b>weighted_pred_flag</b>	1	u(1)
<b>weighted_bipred_idc</b>	1	u(2)
<b>pic_init_qp_minus26</b> /* 26相对 */	1	se(v)
<b>pic_init_qs_minus26</b> /* 26相对 */	1	se(v)
<b>chroma_qp_index_offset</b>	1	se(v)
<b>deblocking_filter_control_present_flag</b>	1	u(1)
<b>constrained_intra_pred_flag</b>	1	u(1)
<b>redundant_pic_cnt_present_flag</b>	1	u(1)
if( more_rbsp_data() ) {		
<b>transform_8x8_mode_flag</b>	1	u(1)
<b>pic_scaling_matrix_present_flag</b>	1	u(1)
if( pic_scaling_matrix_present_flag )		
for( i = 0; i < 6 + ( (chroma_format_idc != 3) ? 2 : 6 ) * transform_8x8_mode_flag; i++ ) {		
<b>pic_scaling_list_present_flag[ i ]</b>	1	u(1)
if( pic_scaling_list_present_flag[ i ] )		
if( i < 6 )		
scaling_list( ScalingList4x4[ i ], 16, UseDefaultScalingMatrix4x4Flag[ i ] )	1	
else		
scaling_list( ScalingList8x8[ i - 6 ], 64, UseDefaultScalingMatrix8x8Flag[ i - 6 ] )	1	
}		
<b>second_chroma_qp_index_offset</b>	1	se(v)
}		
rbsp_trailing_bits()	1	
}		

### 7.3.2.3 付加拡張情報RBSPシンタックス

sei_rbsp() {	C	記述子
do		
sei_message()	5	
while( more_rbsp_data() )		
rbsp_trailing_bits()	5	
}		

#### 7.3.2.3.1 付加拡張情報メッセージシンタックス

sei_message() {	C	記述子
payloadType = 0		
while( next_bits( 8 ) == 0xFF ) {		
<b>ff_byte</b> /* 0xFFに等しい */	5	f(8)
payloadType += 255		
}		
<b>last_payload_type_byte</b>	5	u(8)
payloadType += last_payload_type_byte		
payloadSize = 0		
while( next_bits( 8 ) == 0xFF ) {		
<b>ff_byte</b> /* 0xFFに等しい */	5	f(8)
payloadSize += 255		
}		
<b>last_payload_size_byte</b>	5	u(8)
payloadSize += last_payload_size_byte		
sei_payload( payloadType, payloadSize )	5	
}		

#### 7.3.2.4 アクセスユニット境界RBSPシンタックス

access_unit_delimiter_rbsp() {	C	記述子
<b>primary_pic_type</b>	6	u(3)
rbsp_trailing_bits()	6	
}		

### 7.3.2.5 シーケンス終了RBSPシンタックス

end_of_seq_rbsp() {	C	記述子
}		

### 7.3.2.6 ストリーム終了RBSPシンタックス

end_of_stream_rbsp() {	C	記述子
}		

### 7.3.2.7 フィルデータRBSPシンタックス

filler_data_rbsp() {	C	記述子
while( next_bits( 8 ) == 0xFF )		
<b>ff_byte</b> /* 0xFFに等しい */	9	f(8)
rbsp_trailing_bits()	9	
}		

### 7.3.2.8 パーティションのないスライスレイヤRBSPシンタックス

slice_layer_without_partitioning_rbsp() {	C	記述子
slice_header()	2	
slice_data() /* slice_data()シンタックスの全てのカテゴリ */	2 3 4	
rbsp_slice_trailing_bits()	2	
}		

### 7.3.2.9 スライスデータパーティションRBSPシンタックス

#### 7.3.2.9.1 スライスデータパーティションA RBSPシンタックス

slice_data_partition_a_layer_rbsp() {	C	記述子
slice_header()	2	
<b>slice_id</b>	All	ue(v)
slice_data() /* slice_data()シンタックスのカテゴリ2の部分のみ */	2	
rbsp_slice_trailing_bits()	2	
}		

7.3.2.9.2 スライスデータパーティションB RBSPシンタックス

slice_data_partition_b_layer_rbsp() {	C	記述子
<b>slice_id</b>	All	ue(v)
if( separate_colour_plane_flag == 1 )		
<b>colour_plane_id</b>	All	u(2)
if( redundant_pic_cnt_present_flag )		
<b>redundant_pic_cnt</b>	All	ue(v)
slice_data() /* slice_data()シンタックスのカテゴリ3の部分のみ */	3	
rbsp_slice_trailing_bits()	3	
}		

7.3.2.9.3 スライスデータパーティションC RBSPシンタックス

slice_data_partition_c_layer_rbsp() {	C	記述子
<b>slice_id</b>	All	ue(v)
if( separate_colour_plane_flag == 1 )		
<b>colour_plane_id</b>	All	u(2)
if( redundant_pic_cnt_present_flag )		
<b>redundant_pic_cnt</b>	All	ue(v)
slice_data() /* slice_data() シンタックスのカテゴリ4の部分のみ */	4	
rbsp_slice_trailing_bits()	4	
}		

7.3.2.10 RBSPスライス追従ビットシンタックス

rbsp_slice_trailing_bits() {	C	記述子
rbsp_trailing_bits()	All	
if( entropy_coding_mode_flag )		
while( more_rbsp_trailing_data() )		
<b>cabac_zero_word</b> /* 0x0000に等しい */	All	f(16)
}		

### 7.3.2.11 RBSP追従ビットシンタックス

<code>rbsp_trailing_bits() {</code>	<b>C</b>	記述子
<code>    <b>rbsp_stop_one_bit</b> /* 1に等しい */</code>	All	f(1)
<code>    while( !byte_aligned() )</code>		
<code>        <b>rbsp_alignment_zero_bit</b> /* 0に等しい */</code>	All	f(1)
<code>}</code>		

### 7.3.3 スライスヘッダシンタックス

slice_header() {	C	記述子
<b>first_mb_in_slice</b>	2	ue(v)
<b>slice_type</b>	2	ue(v)
<b>pic_parameter_set_id</b>	2	ue(v)
if( separate_colour_plane_flag == 1 )		
<b>colour_plane_id</b>	2	u(2)
<b>frame_num</b>	2	u(v)
if( !frame_mbs_only_flag ) {		
<b>field_pic_flag</b>	2	u(1)
if( field_pic_flag )		
<b>bottom_field_flag</b>	2	u(1)
}		
if( nal_unit_type == 5 )		
<b>idr_pic_id</b>	2	ue(v)
if( pic_order_cnt_type == 0 ) {		
<b>pic_order_cnt_lsb</b>	2	u(v)
if( pic_order_present_flag && !field_pic_flag )		
<b>delta_pic_order_cnt_bottom</b>	2	se(v)
}		
if( pic_order_cnt_type == 1 && !delta_pic_order_always_zero_flag ) {		
<b>delta_pic_order_cnt[ 0 ]</b>	2	se(v)
if( pic_order_present_flag && !field_pic_flag )		
<b>delta_pic_order_cnt[ 1 ]</b>	2	se(v)
}		
if( redundant_pic_cnt_present_flag )		
<b>redundant_pic_cnt</b>	2	ue(v)
if( slice_type == B )		
<b>direct_spatial_mv_pred_flag</b>	2	u(1)
if( slice_type == P    slice_type == SP    slice_type == B ) {		
<b>num_ref_idx_active_override_flag</b>	2	u(1)
if( num_ref_idx_active_override_flag ) {		
<b>num_ref_idx_l0_active_minus1</b>	2	ue(v)
if( slice_type == B )		
<b>num_ref_idx_l1_active_minus1</b>	2	ue(v)
}		
}		
ref_pic_list_reordering( )	2	
if( ( weighted_pred_flag && ( slice_type == P    slice_type == SP ) )    ( weighted_bipred_idc == 1 && slice_type == B ) )		
pred_weight_table( )	2	
if( nal_ref_idc != 0 )		

dec_ref_pic_marking( )	2	
if( entropy_coding_mode_flag && slice_type != I && slice_type != SI )		
<b>cabac_init_idc</b>	2	ue(v)
<b>slice_qp_delta</b>	2	se(v)
if( slice_type == SP    slice_type == SI ) {		
if( slice_type == SP )		
<b>sp_for_switch_flag</b>	2	u(1)
<b>slice_qs_delta</b>	2	se(v)
}		
if( deblocking_filter_control_present_flag ) {		
<b>disable_deblocking_filter_idc</b>	2	ue(v)
if( disable_deblocking_filter_idc != 1 ) {		
<b>slice_alpha_c0_offset_div2</b>	2	se(v)
<b>slice_beta_offset_div2</b>	2	se(v)
}		
}		
if( num_slice_groups_minus1 > 0 && slice_group_map_type >= 3 && slice_group_map_type <= 5)		
<b>slice_group_change_cycle</b>	2	u(v)
}		

### 7.3.3.1 参照ピクチャリスト並替シンタックス

ref_pic_list_reordering() {	C	記述子
if( slice_type % 5 != 2 && slice_type % 5 != 4 ) {		
<b>ref_pic_list_reordering_flag_l0</b>	2	u(1)
if( ref_pic_list_reordering_flag_l0 )		
do {		
<b>reordering_of_pic_nums_idc</b>	2	ue(v)
if( reordering_of_pic_nums_idc == 0    reordering_of_pic_nums_idc == 1 )		
<b>abs_diff_pic_num_minus1</b>	2	ue(v)
else if( reordering_of_pic_nums_idc == 2 )		
<b>long_term_pic_num</b>	2	ue(v)
} while( reordering_of_pic_nums_idc != 3 )		
}		
if( slice_type % 5 == 1 ) {		
<b>ref_pic_list_reordering_flag_l1</b>	2	u(1)
if( ref_pic_list_reordering_flag_l1 )		
do {		
<b>reordering_of_pic_nums_idc</b>	2	ue(v)
if( reordering_of_pic_nums_idc == 0    reordering_of_pic_nums_idc == 1 )		
<b>abs_diff_pic_num_minus1</b>	2	ue(v)
else if( reordering_of_pic_nums_idc == 2 )		
<b>long_term_pic_num</b>	2	ue(v)
} while( reordering_of_pic_nums_idc != 3 )		
}		
}		

### 7.3.3.2 予測重み表シンタックス

	C	記述子
pred_weight_table() {		
<b>luma_log2_weight_denom</b>	2	ue(v)
if( ChromaArrayType != 0 )		
<b>chroma_log2_weight_denom</b>	2	ue(v)
for( i = 0; i <= num_ref_idx_l0_active_minus1; i++ ) {		
<b>luma_weight_l0_flag</b>	2	u(1)
if( luma_weight_l0_flag ) {		
<b>luma_weight_l0[ i ]</b>	2	se(v)
<b>luma_offset_l0[ i ]</b>	2	se(v)
}		
if( ChromaArrayType != 0 ){		
<b>chroma_weight_l0_flag</b>	2	u(1)
if( chroma_weight_l0_flag )		
for( j = 0; j < 2; j++ ) {		
<b>chroma_weight_l0[ i ][ j ]</b>	2	se(v)
<b>chroma_offset_l0[ i ][ j ]</b>	2	se(v)
}		
}		
}		
if( slice_type % 5 == 1 )		
for( i = 0; i <= num_ref_idx_l1_active_minus1; i++ ) {		
<b>luma_weight_l1_flag</b>	2	u(1)
if( luma_weight_l1_flag ) {		
<b>luma_weight_l1[ i ]</b>	2	se(v)
<b>luma_offset_l1[ i ]</b>	2	se(v)
}		
if( ChromaArrayType != 0 ){		
<b>chroma_weight_l1_flag</b>	2	u(1)
if( chroma_weight_l1_flag )		
for( j = 0; j < 2; j++ ) {		
<b>chroma_weight_l1[ i ][ j ]</b>	2	se(v)
<b>chroma_offset_l1[ i ][ j ]</b>	2	se(v)
}		
}		
}		
}		

### 7.3.3.3 復号参照ピクチャマーキングシンタックス

	C	記述子
dec_ref_pic_marking( ) {		
if( IdrPicFlag ) {		
<b>no_output_of_prior_pics_flag</b>	2   5	u(1)
<b>long_term_reference_flag</b>	2   5	u(1)
} else {		
<b>adaptive_ref_pic_marking_mode_flag</b>	2   5	u(1)
if( adaptive_ref_pic_marking_mode_flag )		
do {		
<b>memory_management_control_operation</b>	2   5	ue(v)
if( memory_management_control_operation == 1    memory_management_control_operation == 3 )		
<b>difference_of_pic_nums_minus1</b>	2   5	ue(v)
if( memory_management_control_operation == 2 )		
<b>long_term_pic_num</b>	2   5	ue(v)
if( memory_management_control_operation == 3    memory_management_control_operation == 6 )		
<b>long_term_frame_idx</b>	2   5	ue(v)
if( memory_management_control_operation == 4 )		
<b>max_long_term_frame_idx_plus1</b>	2   5	ue(v)
} while( memory_management_control_operation != 0 )		
}		
}		
}		

### 7.3.4 スライスデータシンタックス

	C	記述子
slice_data( ) {		
if( entropy_coding_mode_flag )		
while( !byte_aligned( ) )		
<b>cabac_alignment_one_bit</b>	2	f(1)
CurrMbAddr = first_mb_in_slice * ( 1 + MbaffFrameFlag )		
moreDataFlag = 1		
prevMbSkipped = 0		
do {		
if( slice_type != I && slice_type != SI )		
if( !entropy_coding_mode_flag ) {		
<b>mb_skip_run</b>	2	ue(v)
prevMbSkipped = ( mb_skip_run > 0 )		
for( i=0; i<mb_skip_run; i++ )		
CurrMbAddr = NextMbAddress( CurrMbAddr )		
moreDataFlag = more_rbsp_data( )		
}		

} else {		
<b>mb_skip_flag</b>	2	ae(v)
moreDataFlag = !mb_skip_flag		
}		
if( moreDataFlag ) {		
if( MbaffFrameFlag && ( CurrMbAddr % 2 == 0    ( CurrMbAddr % 2 == 1 && prevMbSkipped ) ) )		
<b>mb_field_decoding_flag</b>	2	u(1)   ae(v)
macroblock_layer( )	2   3   4	
}		
if( !entropy_coding_mode_flag )		
moreDataFlag = more_rbsp_data( )		
else {		
if( slice_type != I && slice_type != SI )		
prevMbSkipped = mb_skip_flag		
if( MbaffFrameFlag && CurrMbAddr % 2 == 0 )		
moreDataFlag = 1		
else {		
<b>end_of_slice_flag</b>	2	ae(v)
moreDataFlag = !end_of_slice_flag		
}		
}		
CurrMbAddr = NextMbAddress( CurrMbAddr )		
} while( moreDataFlag )		
}		

### 7.3.5 マクロブロックレイヤシンタックス

macroblock_layer( ) {	<b>C</b>	<b>記述子</b>
<b>mb_type</b>	2	ue(v)   ae(v)
if( mb_type == I_PCM ) {		
while( !byte_aligned( ) )		
<b>pcm_alignment_zero_bit</b>	3	f(1)
for( i = 0; i < 256; i++ )		
<b>pcm_sample_luma[ i ]</b>	3	u(v)
for( i = 0; i < 2 * MbWidthC * MbHeightC; i++ )		
<b>pcm_sample_chroma[ i ]</b>	3	u(v)
} else {		
noSubMbPartSizeLessThan8x8Flag = 1		
if( mb_type != I_NxN && MbPartPredMode( mb_type, 0 ) != Intra_16x16 && NumMbPart( mb_type ) == 4 ) {		

sub_mb_pred( mb_type )	2	
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( sub_mb_type[ mbPartIdx ] != B_Direct_8x8 ) {		
if( NumSubMbPart( sub_mb_type[ mbPartIdx ] ) > 1 )		
noSubMbPartSizeLessThan8x8Flag = 0		
} else if( !direct_8x8_inference_flag )		
noSubMbPartSizeLessThan8x8Flag = 0		
} else {		
if( transform_8x8_mode_flag && mb_type == I_NxN )		
<b>transform_size_8x8_flag</b>	2	u(1)   ae(v)
mb_pred( mb_type )	2	
}		
if( MbPartPredMode( mb_type, 0 ) != Intra_16x16 ){		
<b>coded_block_pattern</b>	2	me(v)   ae(v)
if( CodedBlockPatternLuma > 0 && transform_8x8_mode_flag && mb_type != I_NxN && noSubMbPartSizeLessThan8x8Flag && ( mb_type != B_Direct_16x16    direct_8x8_inference_flag ) )		
<b>transform_size_8x8_flag</b>	2	u(1)   ae(v)
}		
if( CodedBlockPatternLuma > 0    CodedBlockPatternChroma > 0    MbPartPredMode( mb_type, 0 ) == Intra_16x16 ) {		
<b>mb_qp_delta</b>	2	se(v)   ae(v)
residual( 0, 0, 15 )	3   4	
}		
}		
}		

7.3.5.1 マクロブロック予測シンタックス

mb_pred( mb_type ) {	C	記述子
if( MbPartPredMode( mb_type, 0 ) == Intra_4x4    MbPartPredMode( mb_type, 0 ) == Intra_8x8    MbPartPredMode( mb_type, 0 ) == Intra_16x16 ) {		
if( MbPartPredMode( mb_type, 0 ) == Intra_4x4 )		
for( luma4x4BlkIdx=0; luma4x4BlkIdx<16; luma4x4BlkIdx++ ) {		
<b>prev_intra4x4_pred_mode_flag</b> [ luma4x4BlkIdx ]	2	u(1)   ae(v)
if( !prev_intra4x4_pred_mode_flag[ luma4x4BlkIdx ] )		
<b>rem_intra4x4_pred_mode</b> [ luma4x4BlkIdx ]	2	u(3)   ae(v)
}		
if( MbPartPredMode( mb_type, 0 ) == Intra_8x8 )		
for( luma8x8BlkIdx=0; luma8x8BlkIdx<4; luma8x8BlkIdx++ ) {		
<b>prev_intra8x8_pred_mode_flag</b> [ luma8x8BlkIdx ]	2	u(1)   ae(v)
if( !prev_intra8x8_pred_mode_flag[ luma8x8BlkIdx ] )		
<b>rem_intra8x8_pred_mode</b> [ luma8x8BlkIdx ]	2	u(3)   ae(v)
}		
if( ChromaArrayType == 1    ChromaArrayType == 2 )		
<b>intra_chroma_pred_mode</b>	2	ue(v)   ae(v)
} else if( MbPartPredMode( mb_type, 0 ) != Direct ) {		
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++ )		
if( ( num_ref_idx_l0_active_minus1 > 0    mb_field_decoding_flag ) && MbPartPredMode( mb_type, mbPartIdx ) != Pred_L1 )		
<b>ref_idx_l0</b> [ mbPartIdx ]	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++ )		
if( ( num_ref_idx_l1_active_minus1 > 0    mb_field_decoding_flag ) && MbPartPredMode( mb_type, mbPartIdx ) != Pred_L0 )		
<b>ref_idx_l1</b> [ mbPartIdx ]	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++ )		
if( MbPartPredMode( mb_type, mbPartIdx ) != Pred_L1 )		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_l0</b> [ mbPartIdx ][ 0 ][ compIdx ]	2	se(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++ )		
if( MbPartPredMode( mb_type, mbPartIdx ) != Pred_L0 )		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_l1</b> [ mbPartIdx ][ 0 ][ compIdx ]	2	se(v)   ae(v)
}		
}		

### 7.3.5.2 サブマクロブロック予測シンタックス

sub_mb_pred( mb_type ) {	C	記述子
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
<b>sub_mb_type[ mbPartIdx ]</b>	2	ue(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( ( num_ref_idx_l0_active_minus1 > 0    mb_field_decoding_flag ) && mb_type != P_8x8ref0 && sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L1 )		
<b>ref_idx_l0[ mbPartIdx ]</b>	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( ( num_ref_idx_l1_active_minus1 > 0    mb_field_decoding_flag ) && sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L0 )		
<b>ref_idx_l1[ mbPartIdx ]</b>	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L1 )		
for( subMbPartIdx = 0; subMbPartIdx < NumSubMbPart( sub_mb_type[ mbPartIdx ] ); subMbPartIdx++)		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_l0[ mbPartIdx ][ subMbPartIdx ][ compIdx ]</b>	2	se(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L0 )		
for( subMbPartIdx = 0; subMbPartIdx < NumSubMbPart( sub_mb_type[ mbPartIdx ] ); subMbPartIdx++)		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_l1[ mbPartIdx ][ subMbPartIdx ][ compIdx ]</b>	2	se(v)   ae(v)
}		

### 7.3.5.3 残差データシンタックス

	C	記述子
residual( bmFlag, startIdx, endIdx ) {		
if( !entropy_coding_mode_flag )		
residual_block = residual_block_cavlc		
else		
residual_block = residual_block_cabac		
residual_luma( i16x16DClevel, i16x16AClevel, level, level8x8, bmFlag, startIdx, endIdx )		
Intra16x16DCLevel = i16x16DClevel		
Intra16x16ACLevel = i16x16AClevel		
LumaLevel = level		
LumaLevel8x8 = level8x8		
if( ChromaArrayType == 1    ChromaArrayType == 2 ) {		
NumC8x8 = 4 / ( SubWidthC * SubHeightC )		
for( iCbCr = 0; iCbCr < 2; iCbCr++ )		
if( ( CodedBlockPatternChroma & 3 ) && startIdx == 0 ) /* 色差DC残差存在 */		
residual_block( ChromaDCLevel[ iCbCr ], 0, 4 * NumC8x8 - 1, 4 * NumC8x8 )	3   4	
else		
for( i = 0; i < 4 * NumC8x8; i++ )		
ChromaDCLevel[ iCbCr ][ i ] = 0		
for( iCbCr = 0; iCbCr < 2; iCbCr++ )		
for( i8x8 = 0; i8x8 < NumC8x8; i8x8++ )		
for( i4x4 = 0; i4x4 < 4; i4x4++ )		
if( ( CodedBlockPatternChroma & 2 ) && endIdx > 0 ) /* 色差AC残差存在 */		
residual_block( ChromaACLevel[ iCbCr ][ i8x8*4+i4x4 ], max( 0, startIdx - 1 ), endIdx - 1, 15 )	3   4	
else		
for( i = 0; i < 15; i++ )		
ChromaACLevel[ iCbCr ][ i8x8*4+i4x4 ][ i ] = 0		
} else if( ChromaArrayType == 3 ) {		
residual_luma( i16x16DClevel, i16x16AClevel, level, level8x8, bmFlag, startIdx, endIdx )		
CbIntra16x16DCLevel = i16x16DClevel		
CbIntra16x16ACLevel = i16x16AClevel		
CbLevel = level		
CbLevel8x8 = level8x8		
residual_luma( i16x16DClevel, i16x16AClevel, level, level8x8, bmFlag, startIdx, endIdx )		
CrIntra16x16DCLevel = i16x16DClevel		

CrIntra16x16ACLevel = i16x16AClevel		
CrLevel = level		
CrLevel8x8 = level8x8		
}		

### 7.3.5.3.1 残差輝度シンタックス

residual_luma( i16x16DClevel, i16x16AClevel, level, level8x8, bmFlag, startIdx, endIdx ) {	C	記述子
if( !entropy_coding_mode_flag )		
residual_block = residual_block_cavlc		
else		
residual_block = residual_block_cabac		
if( !bmFlag && startIdx == 0 && MbPartPredMode( mb_type, 0 ) == Intra_16x16 )		
residual_block( i16x16DClevel, 0, 15, 16 )	3	
for( i8x8 = 0; i8x8 < 4; i8x8++ )		
if( !transform_size_8x8_flag    !entropy_coding_mode_flag )		
for( i4x4 = 0; i4x4 < 4; i4x4++ ) {		
if( CodedBlockPatternLuma & ( 1 << i8x8 ) )		
if( !bmFlag && endIdx > 0 && MbPartPredMode( mb_type, 0 ) == Intra_16x16 )		
residual_block( i16x16AClevel[ i8x8*4+ i4x4 ], max( 0, startIdx - 1 ), endIdx - 1, 15 )	3	
else		
residual_block( level[ i8x8 * 4 + i4x4 ], startIdx, endIdx, 16 )	3   4	
else if( !bmFlag && MbPartPredMode( mb_type, 0 ) == Intra_16x16 )		
for( i = 0; i < 15; i++ )		
i16x16AClevel[ i8x8 * 4 + i4x4 ][ i ] = 0		
else		
for( i = 0; i < 16; i++ )		
level[ i8x8 * 4 + i4x4 ][ i ] = 0		
if( !entropy_coding_mode_flag && transform_size_8x8_flag )		
for( i = 0; i < 16; i++ )		
level8x8[ i8x8 ][ 4 * i + i4x4 ] = level[ i8x8 * 4 + i4x4 ][ i ]		
}		
else if( CodedBlockPatternLuma & ( 1 << i8x8 ) )		
residual_block( level8x8[ i8x8 ], 4 * startIdx, 4 * endIdx + 3, 64 )	3   4	
else		
for( i = 0; i < 64; i++ )		
level8x8[ i8x8 ][ i ] = 0		
}		

### 7.3.5.3.2 残差ブロックCAVLCシンタックス

residual_block_cavlc( coeffLevel, startIdx, endIdx, maxNumCoeff ) {	C	記述子
for( i = 0; i < maxNumCoeff; i++ )		
coeffLevel[ i ] = 0		
<b>coeff_token</b>	3   4	ce(v)
if( TotalCoeff( coeff_token ) > 0 ) {		
if( TotalCoeff( coeff_token ) > 10 && TrailingOnes( coeff_token ) < 3 )		
suffixLength = 1		
else		
suffixLength = 0		
for( i = 0; i < TotalCoeff( coeff_token ); i++ )		
if( i < TrailingOnes( coeff_token ) ) {		
<b>trailing_ones_sign_flag</b>	3   4	u(1)
level[ i ] = 1 - 2 * trailing_ones_sign_flag		
} else {		
<b>level_prefix</b>	3   4	ce(v)
levelCode = ( Min( 15, level_prefix ) << suffixLength )		
if( suffixLength > 0    level_prefix >= 14 ) {		
<b>level_suffix</b>	3   4	u(v)
levelCode += level_suffix		
}		
if( level_prefix >= 15 && suffixLength == 0 )		
levelCode += 15		
if( level_prefix >= 16 )		
levelCode += ( 1 << ( level_prefix - 3 ) ) - 4096		
if( i == TrailingOnes( coeff_token ) &&		
TrailingOnes( coeff_token ) < 3 )		
levelCode += 2		
if( levelCode % 2 == 0 )		
level[ i ] = ( levelCode + 2 ) >> 1		
else		
level[ i ] = ( -levelCode - 1 ) >> 1		
if( suffixLength == 0 )		
suffixLength = 1		
if( Abs( level[ i ] ) > ( 3 << ( suffixLength - 1 ) ) &&		
suffixLength < 6 )		
suffixLength++		
}		
if( TotalCoeff( coeff_token ) < endIdx - startIdx + 1 ) {		
<b>total_zeros</b>	3   4	ce(v)
zerosLeft = total_zeros		
} else		

zerosLeft = 0		
for( i = 0; i < TotalCoeff( coeff_token ) - 1; i++ ) {		
if( zerosLeft > 0 ) {		
<b>run_before</b>	3   4	ce(v)
run[ i ] = run_before		
} else		
run[ i ] = 0		
zerosLeft = zerosLeft - run[ i ]		
}		
run[ TotalCoeff( coeff_token ) - 1 ] = zerosLeft		
coeffNum = -1		
for( i = TotalCoeff( coeff_token ) - 1; i >= 0; i-- ) {		
coeffNum += run[ i ] + 1		
coeffLevel[ startIdx + coeffNum ] = level[ i ]		
}		
}		
}		

### 7.3.5.3.3 残差ブロックCABACシンタックス

residual_block_cabac( coeffLevel, startIdx, endIdx, maxNumCoeff ) {	C	記述子
if( maxNumCoeff == 64    ( ChromaArrayType == 3 ) )		
<b>coded_block_flag</b>	3   4	ae(v)
for( i = 0; i < maxNumCoeff; i++ )		
coeffLevel[ i ] = 0		
if( coded_block_flag ) {		
numCoeff = endIdx + 1		
i = startIdx		
do {		
<b>significant_coeff_flag[ i ]</b>	3   4	ae(v)
if( significant_coeff_flag[ i ] ) {		
<b>last_significant_coeff_flag[ i ]</b>	3   4	ae(v)
if( last_significant_coeff_flag[ i ] )		
numCoeff = i + 1		
}		
i++		
} while( i < numCoeff - 1 )		
<b>coeff_abs_level_minus1[ numCoeff - 1 ]</b>	3   4	ae(v)
<b>coeff_sign_flag[ numCoeff - 1 ]</b>	3   4	ae(v)
coeffLevel[ numCoeff - 1 ] = ( coeff_abs_level_minus1[ numCoeff - 1 ] + 1 ) * ( 1 - 2 * coeff_sign_flag[ numCoeff - 1 ] )		
for( i = numCoeff - 2; i >= 0; i-- )		
if( significant_coeff_flag[ i ] ) {		
<b>coeff_abs_level_minus1[ i ]</b>	3   4	ae(v)
<b>coeff_sign_flag[ i ]</b>	3   4	ae(v)
coeffLevel[ i ] = ( coeff_abs_level_minus1[ i ] + 1 ) * ( 1 - 2 * coeff_sign_flag[ i ] )		
}		
}		
}		

## 7.4 意味

シンタックス構造やこれらの構造のシンタックス要素に関連付けられた意味はこの節において規定される。シンタックス要素の意味が1つの表あるいは複数の表の組を使用して規定される時、本標準で別に規定されない限り、表において規定されないいかなる値もビットストリームの中に存在してはならない。

### 7.4.1 NALユニットの意味

記1 VCLは、ビデオデータの内容を効率的に表すために規定される。NALはそのデータをフォーマットし、かつ、様々な通信チャネルあるいは記憶メディア上の伝達に適切な方式でヘッダ情報を提供するた

めに規定される。全てのデータは、それぞれが整数バイト数を含むNALユニットに含まれる。NALユニットは、パケット指向とビットストリームの両システムで使用のため、一般的なフォーマットを規定する。パケット指向伝送およびバイトストリームの両者に対するNALユニットフォーマットは、バイトストリームフォーマットにおけるスタートコードプリフィックスおよび余分なパディングバイトが各NALユニットに先行できる以外は同一である。

NumBytesInNALunit は、NAL ユニットのサイズをバイトで規定する。この値は、NAL ユニットの復号のために必要である。何らかの形式の NAL ユニット境界分離が NumBytesInNALunit の推定を可能にするのに必要である。バイトストリームフォーマットに対して、付属資料 B において、そのような分離手法が 1 つ規定される。他の分離方法は本標準の外部で規定されてもよい。

**forbidden\_zero\_bit** は 0 に等しくなければならない。

**nal\_ref\_idc** が 0 でないことは、NAL ユニットの内容が、シーケンスパラメータセット、シーケンスパラメータセット拡張、サブセットシーケンスパラメータセット、ピクチャパラメータセット、参照ピクチャのスライス、参照ピクチャのスライスデータパーティション、あるいは参照ピクチャのスライスに先行するプリフィックス NAL ユニットのいずれかを規定する。

2-9 章に規定される復号処理を用いて復号される付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスに対しては、スライスまたはスライスデータパーティションを含む 1 つの NAL ユニットの **nal\_ref\_idc** が 0 に等しい **nal\_ref\_idc** は、スライスまたはスライスデータパーティションが、非参照ピクチャの部分であることを示す。

シーケンスパラメータセットあるいはシーケンスパラメータセット拡張あるいはサブセットシーケンスパラメータセットあるいはピクチャパラメータセット NAL ユニットの **nal\_ref\_idc** は 0 であってはならない。1 つの特定のピクチャの、1 つの **nal\_unit\_type** が 1 から 4 のそれぞれの値を含む範囲の NAL ユニットの **nal\_ref\_idc** が 0 に等しい場合、それは、そのピクチャの全ての **nal\_unit\_type** が 1 から 4 のそれぞれの値を含む範囲の NAL ユニットの **nal\_ref\_idc** が 0 に等しくなければならない。

**nal\_unit\_type** が 5 に等しい NAL ユニットの **nal\_ref\_idc** は 0 であってはならない。

6、9、10、11、あるいは 12 に等しい **nal\_unit\_type** を持つ全ての NAL ユニットの **nal\_ref\_idc** は 0 に等しくなければならない。

**nal\_unit\_type** は、表 7-1/JT-H264 において規定される、その NAL ユニットの含まれる RBSP データ構造のタイプを規定する。

表 7-1/JT-H264 において “C” とマークされた列は、その NAL ユニット中に存在してもよいシンタックス要素のカテゴリをリストする。さらに、RBSP データ構造のシンタックスおよび意味により決定されるような、シンタックスカテゴリ “All” を持つシンタックス要素が存在してもよい。特定のリストされたカテゴリの任意のシンタックス要素が存在するかしないかは、関連する RBSP データ構造のシンタックスおよび意味から決定される。シンタックス要素カテゴリ値が、**nal\_unit\_type** の値に等しく、かつ、“All” として分類されない、少なくとも 1 個のシンタックス要素が RBSP データ構造内に存在しない限り、**nal\_unit\_type** は 3 または 4 に等しくてはならない。

2-9 章に規定される復号処理を用いて復号される付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスに対しては、VCL および非 VCL NAL ユニットの **nal\_ref\_idc** は表 7-1/JT-H264 において“付属資料 A の NAL ユニットのタイププラス”とラベル付けされた列により規定される。付属資料 G に規定される復号処理を用いて復号される付属資料 G に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスに対しては、VCL および非 VCL NAL ユニットの **nal\_ref\_idc** は表 7-1/JT-H264 において“付属資料 G の NAL ユニットのタイプ”とラベル付けされた列により規定される。

ニットタイプクラス"とラベル付けされた列により規定される。nal\_unit\_type が 14 に等しい場合の"サフィックスに依存"の項目は、次のように規定する。

- もしnal\_unit\_type が14に等しいNALユニットに復号順序で直接に続くNALユニットがnal\_unit\_type が1または5に等しいNALユニットであれば、nal\_unit\_type が14に等しいNALユニットはVCL NALユニットである。
- それ以外、もしnal\_unit\_type が14に等しいNALユニットに復号順序で直接に続くNALユニットがnal\_unit\_type が12に等しいNALユニットであれば、nal\_unit\_type が14に等しいNALユニットは非VCL NALユニットである。
- それ以外(nal\_unit\_type が14に等しいNALユニットに復号順序で直接に続くNALユニットがnal\_unit\_type が1、5または12に等しくないNALユニット)、nal\_unit\_type が14に等しいNALユニットは非VCL NALユニットであり、復号器は、そのnal\_unit\_type が14に等しいNALユニットおよびそのnal\_unit\_type が14に等しいNALユニットに直接に続くNALユニットを無視(ビットストリームより除去して破棄)しなければならない。

表7-1/JT-H264 NALユニットタイプ符号、シンタックス要素カテゴリおよびNALユニットタイプクラス  
(ITU-T H.264)

nal_unit_type	NALユニットの内容およびRBSPシンタックス構造	C	付属資料AのNALユニットタイプクラス	付属資料GのNALユニットタイプクラス
0	無規定		非VCL	非VCL
1	非IDRピクチャの符号化スライス slice_layer_without_partitioning_rbsp()	2,3,4	VCL	VCL
2	符号化スライスデータパーティションA slice_data_partition_a_layer_rbsp()	2	VCL	適用不可
3	符号化スライスデータパーティションB slice_data_partition_b_layer_rbsp()	3	VCL	適用不可
4	符号化スライスデータパーティションC slice_data_partition_c_layer_rbsp()	4	VCL	適用不可
5	IDRピクチャの符号化スライス slice_layer_without_partitioning_rbsp()	2,3	VCL	VCL
6	付加拡張情報 (SEI) sei_rbsp()	5	非VCL	非VCL
7	シーケンスパラメータセット seq_parameter_set_rbsp()	0	非VCL	非VCL
8	ピクチャパラメータセット pic_parameter_set_rbsp()	1	非VCL	非VCL
9	アクセスユニット境界 access_unit_delimiter_rbsp()	6	非VCL	非VCL
10	シーケンス終了	7	非VCL	非VCL

	end_of_seq_rbsp()			
11	ストリーム終了 end_of_stream_rbsp()	8	非VCL	非VCL
12	フィルデータ filler_data_rbsp()	9	非VCL	非VCL
13	シーケンスパラメータセット拡張 seq_parameter_set_extension_rbsp()	10	非VCL	非VCL
14	スケーラブル拡張のプリフィックスNALユニット prefix_nal_unit_rbsp() /* 付属資料Gにて規定 */	2	非VCL	サフィックス に依存
15	サブセットシーケンスパラメータセット subset_seq_parameter_set_rbsp() /* 付属資料Gにて規定 */	0	非VCL	非VCL
16..18	予約		非VCL	非VCL
19	パーティションのない補助符号化ピクチャの符号化スライス slice_layer_without_partitioning_rbsp()	2,3,4	非VCL	非VCL
20	スケーラブル拡張の符号化スライス slice_layer_in_scalable_extension_rbsp() /* 付属資料Gにて規定 */	2,3,4	非VCL	VCL
21..23	予約		非VCL	非VCL
24..31	無規定		非VCL	非VCL

13 または 19 に等しい nal\_unit\_type をもつ NAL ユニットの、13 または 19 に等しくない nal\_unit\_type をもつ NAL ユニットの復号処理に影響することなく、かつ本標準の適合に影響することなく、復号器によって破棄されてもよい。

14、15 または 20 に等しい nal\_unit\_type をもつ NAL ユニットの、14、15 または 20 に等しくない nal\_unit\_type をもつ NAL ユニットの復号処理に影響することなく、かつ付属資料 A に規定されるプロファイルへの適合に影響することなく、復号器によって破棄されてもよい。

0 に等しい、または 24 から 31 のそれぞれの値を含む範囲にある nal\_unit\_type を使用する NAL ユニットの、本標準において規定される復号処理に影響してはならない。

記2 NAL ユニットのタイプ 0 および 24 から 31 は、アプリケーションで決定され使用されてもよい。nal\_unit\_type のこれらの値に対する復号処理は本標準においては規定されない。

復号器は、nal\_unit\_type の予約値を使用する全ての NAL ユニットの内容を見捨て（ビットストリームから除去して破棄）しなければならない。

記3 この要求は本標準の互換拡張の将来定義を許す。

本文中では、符号化されたスライス NAL ユニットの、ある非 IDR ピクチャ NAL ユニットの符号化されたスライス、または、ある IDR ピクチャ NAL ユニットの符号化されたスライスを総称的に呼ぶ。次のフラグが規定される：

$$\text{IdrPicFlag} = (\text{nal\_unit\_type} == 5 ? 1 : 0) \quad (7-1)$$

ある特定のピクチャのスライスを含む NAL ユニットに対する `nal_unit_type` の値が 5 に等しい場合、そのピクチャは `nal_unit_type` が 1 から 4 のそれぞれの値を含む範囲の NAL ユニットを含んではならない。2-9 章に規定される復号処理を用いて復号される付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスに対しては、そのようなピクチャは、IDR ピクチャと呼ばれる。

記4 スライスデータパーティションは、IDR ピクチャに対しては使用されない。

`rbsp_byte[i]` は、RBSP の *i* 番目のバイトである。RBSP は、次のようなバイトの順序づけられたシーケンスとして規定される。

RBSP は、次の通り SODB を含む。

- もし SODB が空 (すなわち長さが 0 ビット) ならば、RBSP もまた空である。
- それ以外、RBSP は次の通り SODB を含む。
  - 1) RBSP の第 1 のバイトは、SODB の (最上位、左端) 8 ビットを含む ; 以降同様に、SODB が残り 8 ビット未満となるまで、RBSP の次のバイトは、SODB の次の 8 ビットを含まなければならない。
  - 2) `rbsp_trailing_bits()` は、以下のように SODB の後に存在する :
    - i) 最後の RBSP バイトの最初の (最上位、左端) ビットは、(もしあれば) SODB に残る (複数の) ビットを含む、
    - ii) 次のビットは、1 に等しい 1 つの `rbsp_stop_one_bit` からなる。そして、
    - iii) `rbsp_stop_one_bit` がバイト整列されたバイトの最後のビットでない場合、1 つ以上の `rbsp_alignment_zero_bit` がバイト整列させるために存在する。
  - 3) いくつかの RBSP 中で、RBSP の終わりの `rbsp_trailing_bits()` の後に、0x0000 に等しい 1 つ以上の、`cabac_zero_word16` ビットシンタックス要素が、存在するかもしれない。

これら RBSP 特性をもつシンタックス構造は、“`_rbsp`” 接尾辞を使用してシンタックス表において表示される。これらの構造は、`rbsp_byte[i]` データバイトの内容として NAL ユニット内で運ばれねばならない。RBSP シンタックス構造の、NAL ユニットへの関連は、表 7-1/JT-H264 で規定されるようなものでなければならない。

記5 RBSP の境界が知られている場合、RBSP のバイトのビット連結と、1 に等しい最後 (最下位、右端) のビットである `rbsp_stop_one_bit` の破棄と、さらに、それに続く全ての 0 に等しい (下位側、さらに右側) ビットの破棄により、復号器は RBSP から SODB を抽出できる。復号処理に必要なデータは、RBSP の SODB 部分に含まれている。

`emulation_prevention_three_byte` は、0x03 に等しいバイトである。`emulation_prevention_three_byte` が NAL ユニットに存在する場合、それは復号処理で破棄されなければならない。

NAL ユニットの最後のバイトは、0x00 に等しくてはならない。

NAL ユニット内では、次の 3 バイトのシーケンスはいかなるバイト整列位置でも生じてはならない :

- 0x000000
- 0x000001
- 0x000002

NAL ユニット内では、次のシーケンス以外で 0x000003 で始まるいかなる 4 バイトのシーケンスも、いかなるバイト整列位置でも生じてはならない：

- 0x00000300
- 0x00000301
- 0x00000302
- 0x00000303

記6 nal\_unit\_typeが0に等しい時、シンタックス要素emulation\_prevention\_three\_byteがNALユニットの3バイト目になれないので、NALユニットシンタックス構造の開始において上記に示された3バイトと4バイトのパターンの存在を避けるために、符号器の設計において特別な注意が用いられなければならない。

#### 7.4.1.1 RBSP内のSODBのカプセル化（参考）

この小節は、本標準の必須部分を形成しない。

RBSP 内の SODB のカプセル化の形式、および NAL ユニット内の RBSP のカプセル化のための emulation\_prevention\_three\_byte の使用は、次の目的のために規定される：

- 任意のSODBがNALユニット内に表されることを可能にしながら、NALユニット内のスタートコードのエミュレーションを防ぐこと、
- rbsp\_stop\_one\_bitに対してRBSPの終わりから始まるRBSPの探索をすることで、NALユニット内のSODBの終了の識別を可能にすること、および
- (1つ以上のcabac\_zero\_wordを使用して) いくつかの状況下で、NALユニットがSODBのそれより大きなサイズを持つことを可能にすること。

符号器は下記手続きによって RBSP から NAL ユニットを生成することができる：

RBSP データは、次に従うバイナリパターンのバイト整列されたビットに対して探索される：

'00000000 00000000 000000xx' (ここで xx は、いずれかの 2 ビットパターンを示す：00、01、10 または 11)

そして、0x03 に等しいバイトが、次のパターンを持つビットパターンに置き換えるために挿入される。

'00000000 00000000 00000011 000000xx'

そして最後に、RBSP データの最後のバイトが 0x00 と等しい場合 (RBSP が cabac\_zero\_word で終わる場合にのみ、生じる)、0x03 と等しい最終バイトが、データの終わりへ追加される。

結果生じるバイトシーケンスは、それが含む RBSP データ構造のタイプの指示を含む NAL ユニットの第 1 のバイトが前に付けられる。これが、NAL ユニット全体の構築になる。

この処理は、任意の SODB が以下を保証しながら NAL ユニットで表されることを許す。

- バイト整列されたスタートコードプリフィックスは、NALユニット内にエミュレートされない、および
- スタートコードプリフィックスが後に続く8つの0値のビットシーケンスは、バイト整列にかかわらず、NALユニット内にエミュレートされない。

#### 7.4.1.2 NALユニットの順序および符号化ピクチャ、アクセスユニットとビデオシーケンスへの関連

この小節は、ビットストリームにおける NAL ユニットの順序の制約を規定する。ビットストリームにおいて、これらの制約に従う NAL ユニットの任意の順序が、文書中では、NAL ユニットの復号順序と呼ばれる。NAL ユニット内では、7.3、D.1 および E.1 節のシンタックスが、シンタックス要素の復号順序を規定する。本標準に適合する復号器は、復号順序になっている NAL ユニットおよびそれらのシンタックス要素を受け取ることが可能でなければならない。

##### 7.4.1.2.1 シーケンスおよびピクチャパラメータセット RBSP の順序およびそれらのアクティブ化

この小節は、2-9 章に規定される復号処理を用いて復号される付属資料 A に規定される 1 つ以上のプロフィールに適合する符号化ビデオシーケンスに対する、ピクチャおよびシーケンスパラメータセットのアクティブ化処理を規定する。

記1 シーケンスおよびピクチャパラメータセット機構は、まれに変更する情報の伝送を、符号化マクロブロックデータの伝送から分離する。シーケンスおよびピクチャパラメータセットは、いくつかのアプリケーションにおいて、信頼できる伝送機構を使用した“帯域外”で伝達されてもよい。

ピクチャパラメータセット RBSP は、1 つ以上の符号化ピクチャの符号化スライス NAL ユニットあるいは符号化スライスデータパーティション A NAL ユニットによって参照されるパラメータを含む。各々のピクチャパラメータセット RBSP は、復号処理操作の開始時点では、最初はアクティブでないとみなされる。復号処理操作の間の任意の与えられた瞬間で、多くとも 1 つのピクチャパラメータセット RBSP がアクティブとみなされ、そして、任意の特定ピクチャパラメータセット RBSP のアクティブ化は、(もしあれば) 直前のアクティブピクチャパラメータセットを非アクティブ化する結果になる。

ピクチャパラメータセット RBSP (`pic_parameter_set_id` の特定の値を持つ) がアクティブではなく、それが符号化スライス NAL ユニットまたは符号化スライスデータパーティション A NAL ユニットによって (`pic_parameter_set_id` のその値を使用して) 参照される時、それがアクティブ化される。別のピクチャパラメータセット RBSP のアクティブ化によりそれが非アクティブ化されるまで、このピクチャパラメータセット RBSP は、アクティブなピクチャパラメータセット RBSP と呼ばれる。`pic_parameter_set_id` の特定の値をもつピクチャパラメータセット RBSP は、そのアクティブ化の前に復号処理で利用可能にならねばならない。

そのアクティブなピクチャパラメータセット RBSP に対する `pic_parameter_set_id` の値を含むどんなピクチャパラメータセット NAL ユニットも、それが 1 つの符号化ピクチャの最後の VCL NAL ユニットに続きかつ別の符号化ピクチャの最初の VCL NAL ユニットに先行するのではない限り、そのアクティブなピクチャパラメータセット RBSP のそれと同じ内容を持たなければならない。

シーケンスパラメータセット RBSP は、1 つ以上のピクチャパラメータセット RBSP、または、1 つ以上のバッファリング期間 SEI メッセージを含む SEI NAL ユニットによって参照されるパラメータを含む。各々のシーケンスパラメータセット RBSP は、復号処理操作の開始時点では、最初はアクティブでないとみなされる。復号処理操作の間の任意の与えられた瞬間で、多くとも 1 つのシーケンスパラメータセット RBSP がアクティブとみなされ、そして、任意の特定シーケンスパラメータセット RBSP のアクティブ化は、(もしあれば) 直前のアクティブシーケンスパラメータセットを非アクティブ化する結果になる。

シーケンスパラメータセット RBSP (`seq_parameter_set_id` の特定の値を持つ) が今アクティブではなく、それがピクチャパラメータセット RBSP のアクティブ化によって (`seq_parameter_set_id` のその値を使用して) 参照されるか、バッファリング期間 SEI メッセージを含む SEI NAL ユニットによって (`seq_parameter_set_id` のその値を使用して) 参照される時、それがアクティブ化される。別のシーケンスパラメータセット RBSP のアクティブ化によりそれが非アクティブ化されるまで、このシーケンスパラメータセット RBSP はアクテ

ィブなシーケンスパラメータセット RBSP と呼ばれる。seq\_parameter\_set\_id の特定の値をもつシーケンスパラメータセット RBSP は、そのアクティブ化の前に、復号処理で利用可能にならねばならない。アクティブ化されたシーケンスパラメータセット RBSP は、符号化ビデオシーケンス全体に対してアクティブなままでなければならない。

記2 IDRアクセスユニットが新しい符号化ビデオシーケンスを始め、そしてアクティブ化されたシーケンスパラメータセットRBSPは符号化ビデオシーケンス全体に対してアクティブのままではなければならないので、シーケンスパラメータセットRBSPは、あるバッファリング期間SEIメッセージがIDRアクセスユニットの一部である時、そのバッファリング期間SEIメッセージのみによりアクティブ化される。

そのアクティブなシーケンスパラメータセット RBSP に対する seq\_parameter\_set\_id の値を含むどんなシーケンスパラメータセット NAL ユニットも、それが符号化ビデオシーケンスの最後のアクセスユニットに続きかつ別の符号化ビデオシーケンスの最初の VCL NAL ユニットとバッファリング期間 SEI メッセージ（存在する時）を含む最初の SEI NAL ユニットに先行するのではない限り、そのアクティブなシーケンスパラメータセット RBSP のそれと同じ内容を持たなければならない。

記3 ピクチャパラメータセットRBSPあるいはシーケンスパラメータセットRBSPがビットストリームの内部で伝達される場合、これらの制約は、ピクチャパラメータセットRBSPあるいはシーケンスパラメータセットRBSPを含むNALユニットに順序の制約を、それぞれ課す。それ以外（ピクチャパラメータセットRBSPあるいはシーケンスパラメータセットRBSPが、本標準で規定されない他の手段によって伝達される）の場合は、それらは、これらの制約に従えるような適時の方法で復号処理に利用可能でなければならない。

存在する場合、シーケンスパラメータセット拡張 RBSP はシーケンスパラメータセット RBSP のそれらに似た機能を持つパラメータを含む。シーケンスパラメータセット拡張 RBSP のシンタックス要素の制約を確立する目的と、シーケンスパラメータセット拡張 RBSP のアクティブ化決定の目的のために、シーケンスパラメータセット拡張 RBSP は、同じ値の seq\_parameter\_set\_id を持つ先行するシーケンスパラメータセット RBSP の一部として見なされなければならない。シーケンスパラメータセット RBSP のアクティブ化よりも前に、同じ値の seq\_parameter\_set\_id を持つシーケンスパラメータセット拡張 RBSP に後続されないシーケンスパラメータセット RBSP が存在する場合、シーケンスパラメータセット拡張 RBSP とそのシンタックス要素は、そのアクティブなシーケンスパラメータセット RBSP に対して存在しないと見なされなければならない。

シーケンスパラメータセットおよびピクチャパラメータセットのシンタックス要素の値（および、それらのシンタックス要素から導出される変数の値）と、その他のシンタックス要素の値の間関係として表現される全ての制約は、アクティブなシーケンスパラメータセットとアクティブなピクチャパラメータセットにのみ適用される制約の表現である。もし任意のシーケンスパラメータセット RBSP がビットストリームでアクティブ化されずに存在するならば、そのシンタックス要素は、仮にそれが別の適合ビットストリームの参照によってアクティブ化されたなら、規定された制約に準拠する値を持たなければならない。もし任意のピクチャパラメータセット RBSP がビットストリームでこれまでにアクティブ化されずに存在するならば、そのシンタックス要素は、仮にそれが別の適合ビットストリームの参照によってアクティブ化されたなら、規定された制約に準拠する値を持たなければならない。

復号処理の動作中（8章を参照）に、アクティブなピクチャパラメータセットおよびアクティブなシーケンスパラメータセットのパラメータ値は、有効とみなされなければならない。SEI メッセージの解釈において、同じアクセスユニットの主符号化ピクチャの VCL NAL ユニットにおける復号処理の動作に対してアクティブであるピクチャパラメータセットおよびシーケンスパラメータセットのパラメータの値は、SEI メッセージの意味に別に規定されない限り有効とみなされなければならない。

#### 7.4.1.2.2 アクセスユニットの順序および符号化ビデオシーケンスへの関連

本標準に適合するビットストリームは、1つ以上の符号化ビデオシーケンスから成る。

符号化ビデオシーケンスは、1つ以上のアクセスユニットから成る。2-9章に規定される復号処理を用いて復号される付属資料 A に規定される 1つ以上のプロファイルに適合する符号化ビデオシーケンスに対する、NAL ユニットおよび符号化ピクチャの順序、およびそれらのアクセスユニットへの関連は、7.4.1.2.3 小小小節において記述される。

それぞれの符号化ビデオシーケンスの最初のアクセスユニットは、IDR アクセスユニットである。符号化ビデオシーケンスのその後続く全てのアクセスユニットは、非 IDR アクセスユニットである。

非参照ピクチャを含む復号順序で連続するアクセスユニットの符号化ピクチャに対するピクチャ順序カウンタの値は非減少でなければならない。

存在するなら、シーケンス終了 NAL ユニットを含むアクセスユニットに続くアクセスユニットは IDR アクセスユニットでなければならない。

SEI NAL ユニットが 1つより多くのアクセスユニットに関連するデータを含む場合(例えば SEI NAL ユニットがその範囲として符号化ビデオシーケンスを持つ場合)、それは、それが適用される最初のアクセスユニットに含まれていなければならない。

ストリーム終了 NAL ユニットがアクセスユニットに存在する場合、このアクセスユニットはビットストリームにおける最後のアクセスユニットでなければならない、ストリーム終了 NAL ユニットはアクセスユニットの最後の NAL ユニットでなければならない。

#### 7.4.1.2.3 NALユニットおよび符号化ピクチャの順序およびアクセスユニットへの関連

この小小小節は、2-9章に規定される復号処理を用いて復号される付属資料 A に規定される 1つ以上のプロファイルに適合する符号化ビデオシーケンスに対する、NAL ユニットおよび符号化ピクチャの順序およびアクセスユニットへの関連を規定する。

アクセスユニットは 1つの主符号化ピクチャ、0以上の対応する冗長符号化ピクチャ、および 0以上の非 VCL NAL ユニットから成る。主あるいは冗長符号化ピクチャへの VCL NAL ユニットの関連は、7.4.1.2.5 小小小節において記述される。

ビットストリーム中の第 1 番目のアクセスユニットは、ビットストリームの第 1 番目の NAL ユニットで開始する。

ある主符号化ピクチャの最後の VCL NAL ユニット後の、以下の任意の NAL ユニットの第 1 番目は、新しいアクセスユニットの開始を規定する。

- アクセスユニット境界 NAL ユニット (存在する時)
- シーケンスパラメータセット NAL ユニット (存在する時)
- ピクチャパラメータセット NAL ユニット (存在する時)
- SEI NAL ユニット (存在する時)
- 14 から 18 のそれぞれの値を含む範囲の nal\_unit\_type を持つ NAL ユニット

- 主符号化ピクチャの最初の VCL NAL ユニット (常に存在)

主符号化ピクチャの最初の VCL NAL ユニットの検出に対する制約は、7.4.1.2.4 小小小節において規定される。

アクセスユニット内の符号化ピクチャおよび非 VCL NAL ユニットの順序は、次の制約に従わなければならない。

- アクセスユニット境界 NAL ユニットが存在する場合、それは最初の NAL ユニットでなければならない。任意のアクセスユニットにおいて、アクセスユニット境界 NAL ユニットは高々1つでなければならない。
- いくつかの SEI NAL ユニットが存在する場合、それらは主符号化ピクチャに先行しなければならない。
- バッファリング期間 SEI メッセージを含む SEI NAL ユニットが存在する場合、そのバッファリング期間 SEI メッセージは、そのアクセスユニットの最初の SEI NAL ユニットの最初の SEI メッセージペイロードでなければならない。
- 主符号化ピクチャは、対応する冗長符号化ピクチャに先行しなければならない。
- 冗長符号化ピクチャが存在する場合、それらは、`redundant_pic_cnt` 値の昇順に順序づけられなければならない。
- シーケンスパラメータセット拡張 NAL ユニットが存在する場合、そのシーケンスパラメータセット拡張 NAL ユニット中の `seq_parameter_set_id` と同じ値を持つシーケンスパラメータセット NAL ユニットに続く次の NAL ユニットでなければならない。
- 1つ以上のパーティションのない補助符号化ピクチャの符号化スライス NAL ユニットが存在する場合、それらは主符号化ピクチャおよび全ての冗長符号化ピクチャ (あれば) に続かなければならない。
- シーケンス終了 NAL ユニットが存在する場合、それは主符号化ピクチャ、および全ての冗長符号化ピクチャ (あれば)、および全てのパーティションのない補助符号化ピクチャ NAL ユニット(あれば)に続かなければならない。
- ストリーム終了 NAL ユニットが存在する場合、それは最後の NAL ユニットでなければならない。
- 0、12、または 20 から 31 のそれぞれの値を含む範囲に等しい `nal_unit_type` を持つ NAL ユニットは、主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

記1 シーケンスパラメータセットNALユニットまたはピクチャパラメータセットNALユニットは、あるアクセスユニットに存在してもよいが、この状態は新しいアクセスユニットの開始を規定するので、アクセスユニットの主符号化ピクチャの最後のVCL NALユニットに続くことはできない。

記2 7または8に等しい`nal_unit_type`を持つNALユニットがアクセスユニットに存在する場合、それは存在するアクセスユニットの符号化ピクチャ内で参照されてもされなくてもよく、後に続くアクセスユニットの符号化ピクチャ内で参照されてもよい。

0、7、8あるいは12から18のそれぞれの値を含む範囲あるいは20から31のそれぞれの値を含む範囲に等しい `nal_unit_type` を持ついかなる NAL ユニットも含まないアクセスユニットの構造は、図 7-1/JT-H264 において示される。

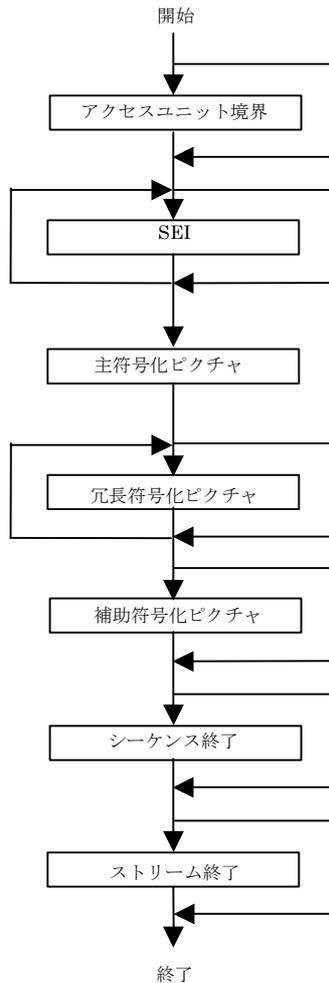


図7-1/JT-H264 0、7、8、あるいは12から18のそれぞれの値を含む範囲あるいは20から31のそれぞれの値を含む範囲に等しいnal\_unit\_typeを持ついかなるNALユニットも含まないアクセスユニットの構造 (ITU-T H.264)

#### 7.4.1.2.4 主符号化ピクチャの最初のVCL NALユニットの検出

この小節は、2-9章に規定される復号処理を用いて復号される付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスに対する、各々の主符号化ピクチャの最初のVCL NALユニットの検出を可能にするのに十分であるVCL NALユニットシンタックスの制約を規定する。

現アクセスユニットの主符号化ピクチャの任意の符号化スライスNALユニットあるいは符号化スライスデータパーティションNALユニットも、1つ以上の下記の点において、以前のアクセスユニットの主符号化ピクチャの任意の符号化スライスNALユニットあるいは符号化スライスデータパーティションNALユニットとも異ならなければならない。

- frame\_num は値が異なる。この条件を調べるために使用される frame\_num の値は、5 に等しい memory\_management\_control\_operation の存在のため、後段の復号処理中に、その値が 0 に等しかったと推定されるかどうかにかかわらず、スライスヘッダのシンタックスに現れる frame\_num の値である。

記1 上の文の結果として、1に等しいframe\_numをもつ主符号化ピクチャは、以下にリストされる他のある条件が、その後続く次の主符号化ピクチャ（もしあるなら）に対して満たされないかぎり、5に等しいmemory\_management\_control\_operationを含むことができない。

- pic\_parameter\_set\_id は値が異なる。
- field\_pic\_flag は値が異なる。
- bottom\_field\_flag が両方に存在し、値が異なる。
- nal\_ref\_idc は値が異なり、その1つは0である。
- pic\_order\_cnt\_type は両方とも0に等しく、および、pic\_order\_cnt\_lsb が値で異なるか、またはdelta\_pic\_order\_cnt\_bottom が値が異なる。
- pic\_order\_cnt\_type は両方とも1に等しく、および delta\_pic\_order\_cnt[0]が値で異なるか、またはdelta\_pic\_order\_cnt[1]が値が異なる。
- IdrPicFlag は値が異なる。
- IdrPicFlag は両方に対して1に等しく、idr\_pic\_id は値が異なる。

記2 冗長符号化ピクチャ内のVCL NALユニットのいくつか、あるいは非VCL NALユニット（例えばアクセスユニット境界NALユニット）のいくつかは、アクセスユニット間の境界検出のために使用されてもよく、それゆえに新しい主符号化ピクチャの開始の検出を助けてもよい。

#### 7.4.1.2.5 VCL NALユニットの順序および符号化ピクチャへの関連

この小小小節は、2-9章に規定される復号処理を用いて復号される付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスに対する、VCL NAL ユニットの順序および符号化ピクチャへの関連を規定する。

それぞれのVCL NAL ユニットの、符号化ピクチャの一部である。

符号化IDRピクチャ内のVCL NAL ユニットの順序は、次の通り制約される。

- もし任意スライス順序が、付属資料Aにおいて規定されるように許可されるならば、IDRピクチャNALユニットの符号化スライスは、互いに相対的な任意の順序を持つてもよい。
- それ以外（任意スライス順序は許可されない）、次が適用される。
  - もし separate\_colour\_plane\_flag が0に等しいならば、スライスグループのIDRピクチャNALユニットの符号化スライスは他のスライスグループのIDRピクチャNALユニットの符号化スライスとインターリーブされてはならず、スライスグループ内のIDRピクチャNALユニットの符号化スライス順序は、その特定のスライスグループのIDRピクチャNALユニットの各符号化スライスの最初のマクロブロックが昇順のマクロブロックアドレスになければならない。
  - それ以外（separate\_colour\_plane\_flagが1に等しい）、特定の値のcolour\_plane\_idに対するスライスグループのIDRピクチャNALユニットの符号化スライスは同じ値のcolour\_plane\_idに対する他のスライスグループのIDRピクチャNALユニットの符号化スライスとインターリーブされてはならず、特定の値のcolour\_plane\_idに対するスライスグループ内のIDRピクチャNALユニットの符号化スライス順序は、その特定の値のcolour\_plane\_idを持つその特定のスライスグループのIDRピクチャNALユニットの各符号化スライスの最初のマクロブロックが昇順のマクロブロックアドレス

になければならない。

記1 `separate_colour_plane_flag`が1に等しい時、異なる`colour_plane_id`の値を持つ符号化スライスの相対的な順序は制約されない。

符号化非 IDR ピクチャ内の VCL NAL ユニットの順序は、次の通り制約される。

- もし任意スライス順序が付属資料 A において規定されるように許可されるならば、非 IDR ピクチャ NAL ユニット、あるいは符号化スライスデータパーティション A NAL ユニットの符号化スライスは、互いに相対的な任意の順序を持ってよい。`slice_id` の特定の値を持つ符号化スライスデータパーティション A NAL ユニットは、`slice_id` の同じ値を持ち存在するどんな符号化スライスデータパーティション B NAL ユニットにも先行しなければならない。`slice_id` の特定の値を持つ符号化スライスデータパーティション A NAL ユニットは、`slice_id` の同じ値を持ち存在するどんな符号化スライスデータパーティション C NAL ユニットにも先行しなければならない。`slice_id` の特定の値を持つ符号化スライスデータパーティション B NAL ユニットが存在する場合、それは `slice_id` の同じ値を持ち存在するどんな符号化スライスデータパーティション C NAL ユニットにも先行しなければならない。
- それ以外（任意スライス順序は許可されない）、次が適用される。
  - もし `separate_colour_plane_flag` が 0 に等しいならば、スライスグループの非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション NAL ユニットの符号化スライスは、他のスライスグループの非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション NAL ユニットの符号化スライスとインターリーブされてはならず、スライスグループ内の非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション A NAL ユニットの符号化スライスの順序は、その特定のスライスグループの非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション A NAL ユニットのそれぞれの符号化スライスの最初のマクロブロックが昇順のマクロブロックアドレスになければならない。`slice_id` の特定の値を持つ符号化スライスデータパーティション A NAL ユニットは、`slice_id` の同じ値を持ち存在するどんな符号化スライスデータパーティション B NAL ユニットにも直ちに先行しなければならない。`slice_id` の同じ値を持つ符号化スライスデータパーティション B NAL ユニットが存在しない場合、`slice_id` の特定の値を持つ符号化スライスデータパーティション A NAL ユニットは、`slice_id` の同じ値を持ち存在するどんな符号化スライスデータパーティション C NAL ユニットにも直ちに先行しなければならない。`slice_id` の特定の値を持つ符号化スライスデータパーティション B NAL ユニットが存在する場合、それは `slice_id` の同じ値を持ち存在するどんな符号化スライスデータパーティション C NAL ユニットにも直ちに先行しなければならない。
  - それ以外（`separate_colour_plane_flag` が 1 に等しい）、特定の値の `colour_plane_id` に対するスライスグループの非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション NAL ユニットの符号化スライスは、同じ値の `colour_plane_id` に対する他のスライスグループの非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション NAL ユニットの符号化スライスとインターリーブされてはならず、特定の値の `colour_plane_id` に対するスライスグループ内の非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション A NAL ユニットの符号化スライスの順序は、その特定の値の `colour_plane_id` を持つその特定のスライスグループの非 IDR ピクチャ NAL ユニットあるいは符号化スライスデータパーティション A NAL ユニットのそれぞれの符号化スライスの最初のマクロブロックが昇順のマクロブロックアドレスになければならない。`slice_id` および `colour_plane_id` の特定の値を持つ符号化スライスデータパーティション A NAL ユニットは、`slice_id` および `colour_plane_id` の同じ値を持ち存在するどんな符号化スライスデータパーテ

イション B NAL ユニットにも直ちに先行しなければならない。slice\_id および colour\_plane\_id の同じ値を持つ符号化スライスデータパーティション B NAL ユニットが存在しない場合、slice\_id および colour\_plane\_id の特定の値を持つ符号化スライスデータパーティション A NAL ユニットは、slice\_id および colour\_plane\_id の同じ値を持ち存在するどんな符号化スライスデータパーティション C NAL ユニットにも直ちに先行しなければならない。slice\_id および colour\_plane\_id の特定な値を持つ符号化スライスデータパーティション B NAL ユニットが存在する場合、それは slice\_id および colour\_plane\_id の同じ値を持ち存在するどんな符号化スライスデータパーティション C NAL ユニットにも直ちに先行しなければならない。

記2 separate\_colour\_plane\_flagが1に等しい時、異なるcolour\_plane\_idの値を持つ符号化スライスの相対的な順序は制約されない。

12 に等しい nal\_unit\_type を持つ NAL ユニットは、アクセスユニット内に存在しても良いが、アクセスユニット内の主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

0、または 24 から 31 のそれぞれの値を含む範囲に等しい nal\_unit\_type を持つ NAL ユニットは、無規定であり、アクセスユニット内に存在しても良いが、アクセスユニット内の主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

20 から 23 のそれぞれの値を含む範囲の nal\_unit\_type を持つ NAL ユニットは、アクセスユニット内の主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

## 7.4.2 未加工のバイトシーケンスペイロードおよびRBSP追従ビットの意味

### 7.4.2.1 シーケンスパラメータセットRBSPの意味

#### 7.4.2.1.1 シーケンスパラメータセットデータの意味

**profile\_idc** および **level\_idc** は、符号化ビデオシーケンスが適合するプロファイルおよびレベルを示す。

**constraint\_set0\_flag** が 1 に等しいことは、符号化ビデオシーケンスが A.2.1 小節において規定された全ての制約に従うことを示す。**constraint\_set0\_flag** が 0 に等しいことは、符号化ビデオシーケンスが A.2.1 小節において規定された制約全てに従ってもよいし、従わなくてもよいことを示す。

**constraint\_set1\_flag** が 1 に等しいことは、符号化ビデオシーケンスが A.2.2 小節において規定された全ての制約に従うことを示す。**constraint\_set1\_flag** が 0 に等しいことは、符号化ビデオシーケンスが A.2.2 小節において規定された制約全てに従ってもよいし、従わなくてもよいことを示す。

**constraint\_set2\_flag** が 1 に等しいことは、符号化ビデオシーケンスが A.2.3 小節において規定された全ての制約に従うことを示す。**constraint\_set2\_flag** が 0 に等しいことは、符号化ビデオシーケンスが A.2.3 小節において規定された制約全てに従ってもよいし、従わなくてもよいことを示す。

記1 **constraint\_set0\_flag**、**constraint\_set1\_flag**あるいは**constraint\_set2\_flag**のうちの1つあるいは1つより多くが1に等しい場合、符号化ビデオシーケンスはA.2節の指示される小節の全ての制約に従わなくてはならない。**profile\_idc**が44、100、110、122あるいは244に等しい場合、**constraint\_set0\_flag**、**constraint\_set1\_flag**および**constraint\_set2\_flag**の値は全て0に等しくなければならない。

**constraint\_set3\_flag** は次の通りを示す。

- もし **profile\_idc** が 66、77 あるいは 88 に等しくかつ **level\_idc** が 11 に等しいならば、**constraint\_set3\_flag** が 1 に等しいことは、符号化ビデオシーケンスが付属資料 A のレベル 1b において規定された全ての制約に従うことを示し、また **constraint\_set3\_flag** が 0 に等しいことは、符号化ビデオシーケンスが付

属資料Aのレベル1bにおいて規定された制約全てに従ってもよいし、従わなくてもよいことを示す。

- それ以外、もし `profile_idc` が 100 あるいは 110 に等しいならば、`constraint_set3_flag` が 1 に等しいことは、符号化ビデオシーケンスが付属資料 A のハイ 10 INTRA プロファイルにおいて規定された全ての制約に従うことを示し、また `constraint_set3_flag` が 0 に等しいことは、符号化ビデオシーケンスがこれらの対応する制約に従ってもよいし、従わなくてもよいことを示す。
- それ以外、もし `profile_idc` が 122 に等しいならば、`constraint_set3_flag` が 1 に等しいことは、符号化ビデオシーケンスが付属資料 A のハイ 4:2:2 INTRA プロファイルにおいて規定された全ての制約に従うことを示し、また `constraint_set3_flag` が 0 に等しいことは、符号化ビデオシーケンスがこれらの対応する制約に従ってもよいし、従わなくてもよいことを示す。
- それ以外、もし `profile_idc` が 44 に等しいならば、`constraint_set3_flag` は 1 に等しくなければならない。`profile_idc` が 44 に等しい時、`constraint_set3_flag` が 0 の値であることは禁止される。
- それ以外、もし `profile_idc` が 244 に等しいならば、`constraint_set3_flag` が 1 に等しいことは、符号化ビデオシーケンスが付属資料 A のハイ 4:4:4 INTRA プロファイルにおいて規定された全ての制約に従うことを示し、また `constraint_set3_flag` が 0 に等しいことは、符号化ビデオシーケンスがこれらの対応する制約に従ってもよいし、従わなくてもよいことを示す。
- それ以外 (`profile_idc` が 66、77 あるいは 88 に等しくかつ `level_idc` が 11 に等しくない)、`constraint_set3_flag` における値 1 は、将来の使用のために TTC によって予約される。`profile_idc` が 66、77 あるいは 88 に等しくかつ `level_idc` が 11 に等しくない場合、`constraint_set3_flag` は本標準に適合するビットストリームにおいて 0 に等しくなくてはならない。`profile_idc` が 66、77 あるいは 88 に等しくかつ `level_idc` が 11 に等しくない場合、本標準に適合する復号器は `constraint_set3_flag` の値を無視しなければならない。

`reserved_zero_4bits` は 0 に等しくなくてはならない。`reserved_zero_4bits` の他の値は、TTC によって、今後規定されてもよい。復号器は、`reserved_zero_4bits` の値を無視しなければならない。

`seq_parameter_set_id` は、ピクチャパラメータセットによって参照されるシーケンスパラメータセットを識別する。`seq_parameter_set_id` の値は、0 から 31 のそれぞれの値を含む範囲でなければならない。

記2 符号器は、それが容易ならば、他のシーケンスパラメータセットシンタックス要素の値が異なる時、特定の値の `seq_parameter_set_id` に関係付けられたシンタックス要素の値を変更するよりも、異なる値の `seq_parameter_set_id` を使用すべきである。

`chroma_format_idc` は、6.2 節において規定された輝度サンプリングに対する色差サンプリングを規定する。`chroma_format_idc` の値は 0 から 3 のそれぞれの値を含む範囲でなければならない。`chroma_format_idc` が存在しない場合、それは 1 に等しい(4:2:0 色差フォーマット)と推定されなければならない。

`separate_colour_plane_flag` が 1 に等しいことは、4:4:4 色差フォーマットの 3 つの色成分が分離して符号化されることを規定する。`separate_colour_plane_flag` が 0 に等しいことは、色成分が分離して符号化されないことを規定する。`separate_colour_plane_flag` が存在しない場合、それは 0 に等しいと推定されなければならない。`separate_colour_plane_flag` が 1 に等しい時は、主符号化ピクチャは、それぞれがモノクロームの符号化シンタックスを用いる色プレーン (Y、Cb または Cr) の符号化サンプルからなる 3 つの分離した成分からなる。この場合、各色プレーンは特定の `colour_plane_id` 値に関連づけられる。

記3 復号処理において、異なる `colour_plane_id` 値を持つ色プレーンの中で依存性はない。例えば、ある値の `colour_plane_id` に対するモノクロームピクチャの復号処理は、INTER 予測において、異なる `colour_plane_id` 値を持つモノクロームピクチャからの何らのデータも使用しない。

separate\_colour\_plane\_flag の値によって、次のように ChromaArrayType 変数の値を割り当てる。

- もし、separate\_colour\_plane\_flag が 0 であれば、ChromaArrayType は chroma\_format\_idc に設定される。
- それ以外 (separate\_colour\_plane\_flag は 1 に等しい)、ChromaArrayType は 0 に設定される。

**bit\_depth\_luma\_minus8** は、輝度配列のサンプルのビット深度と輝度量子化パラメータ範囲オフセット  $QpBdOffset_Y$  を次で規定されるように規定する。

$$BitDepth_Y = 8 + bit\_depth\_luma\_minus8 \quad (7-2)$$

$$QpBdOffset_Y = 6 * bit\_depth\_luma\_minus8 \quad (7-3)$$

**bit\_depth\_luma\_minus8** が存在しない場合、それは 0 に等しいと推定されなければならない。**bit\_depth\_luma\_minus8** は 0 から 6 のそれぞれの値を含む範囲になければならない。

**bit\_depth\_chroma\_minus8** は、色差配列のサンプルのビット深度を規定と色差量子化パラメータ範囲オフセット  $QpBdOffset_C$  を次で規定されるように規定する。

$$BitDepth_C = 8 + bit\_depth\_chroma\_minus8 \quad (7-4)$$

$$QpBdOffset_C = 6 * (bit\_depth\_chroma\_minus8 + residual\_colour\_transform\_flag) \quad (7-5)$$

**bit\_depth\_chroma\_minus8** が存在しない場合、それは 0 に等しいと推定されなければならない。**bit\_depth\_chroma\_minus8** は 0 から 6 のそれぞれの値を含む範囲になければならない。

記4 **bit\_depth\_chroma\_minus8**の値はChromaArrayTypeが0の時は、復号処理に使用されない。特に、separate\_colour\_plane\_flagが1の時、輝度成分の復号処理を使用して（スケーリング行列の選択は除く）各色プレーンは別個のモノクロームピクチャとして復号され、輝度ビット深度が3つの色成分の全てに使用される。

変数 RawMbBits は次のように導出される。

$$RawMbBits = 256 * BitDepth_Y + 2 * MbWidth_C * MbHeight_C * BitDepth_C \quad (7-6)$$

**qprime\_y\_zero\_transform\_bypass\_flag** が 1 に等しいことは、 $QP'_Y$  が 0 に等しい場合、8.5 節で規定される変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理の変換バイパス操作が適用されなければならないことを規定する。**qprime\_y\_zero\_transform\_bypass\_flag** が 0 に等しいことは、変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理が変換バイパス操作を使用してはならないことを規定する。**qprime\_y\_zero\_transform\_bypass\_flag** が存在しない場合、0 に等しいと推定されなければならない。

**seq\_scaling\_matrix\_present\_flag** が 1 に等しいことは、seq\_scaling\_list\_present\_flag[i]、 $i=0..7$  あるいは  $i=0..11$  が存在することを規定する。**seq\_scaling\_matrix\_present\_flag** が 0 に等しいことは、これらのフラグは存在せず、 $i=0..5$  においては Flat\_4x4\_16 で規定されるシーケンスレベルのスケーリングリストが推定されなければならない、 $i=6..11$  においては Flat\_8x8\_16 で規定されるシーケンスレベルのスケーリングリストが推定されなければならない。**seq\_scaling\_matrix\_present\_flag** が存在しない場合、0 に等しいと推定されなければならない。

スケーリングリスト Flat\_4x4\_16 と Flat\_8x8\_16 は次の通り規定される。

$$Flat\_4x4\_16[i]=16, i=0..15 \quad (7-7)$$

$$Flat\_8x8\_16[i]=16, i=0..63 \quad (7-8)$$

seq\_scaling\_list\_present\_flag[i]が1に等しいことは、スケーリングリストiにおけるシンタックス構造がシーケンスパラメータセットの中に存在することを規定する。seq\_scaling\_list\_present\_flag[i]が0に等しいことは、スケーリングリストiにおけるシンタックス構造がシーケンスパラメータセットの中に存在せず、インデックスiにおけるシーケンスレベルのスケーリングリストを推定するのに表7-2/JT-H264で規定されるスケーリングリストフォールバック規則集合Aが使用されなければならないことを規定する。

表7-2/JT-H264 ニーモニック名のスケーリングリストインデックスへの割り当てとフォールバック規則の規定  
(ITU-T H.264)

スケーリングリストインデックスの値	ニーモニック名	ブロックサイズ	MB予測タイプ	成分	スケーリングリストフォールバック規則集合A	スケーリングリストフォールバック規則集合B	デフォルトのスケーリングリスト
0	SI_4x4_Intra_Y	4×4	Intra	Y	デフォルトのスケーリングリスト	シーケンスレベルのスケーリングリスト	Default_4x4_Intra
1	SI_4x4_Intra_Cb	4×4	Intra	Cb	スケーリングリスト、i=0	スケーリングリスト、i=0	Default_4x4_Intra
2	SI_4x4_Intra_Cr	4×4	Intra	Cr	スケーリングリスト、i=1	スケーリングリスト、i=1	Default_4x4_Intra
3	SI_4x4_Inter_Y	4×4	Inter	Y	デフォルトのスケーリングリスト	シーケンスレベルのスケーリングリスト	Default_4x4_Inter
4	SI_4x4_Inter_Cb	4×4	Inter	Cb	スケーリングリスト、i=3	スケーリングリスト、i=3	Default_4x4_Inter
5	SI_4x4_Inter_Cr	4×4	Inter	Cr	スケーリングリスト、i=4	スケーリングリスト、i=4	Default_4x4_Inter
6	SI_8x8_Intra_Y	8×8	Intra	Y	デフォルトのスケーリングリスト	シーケンスレベルのスケーリングリスト	Default_8x8_Intra
7	SI_8x8_Inter_Y	8×8	Inter	Y	デフォルトのスケーリングリスト	シーケンスレベルのスケーリングリスト	Default_8x8_Inter
8	SI_8x8_Intra_Cb	8×8	Intra	Cb	i=6に対するスケーリングリスト	i=6に対するスケーリングリスト	Default_8x8_Intra

9	SI_8x8_Inter_Cb	8×8	Inter	Cb	i=7に対するスケールリングリスト	i=7に対するスケールリングリスト	Default_8x8_Inter
10	SI_8x8_Intra_Cr	8×8	Intra	Cr	i=8に対するスケールリングリスト	i=8に対するスケールリングリスト	Default_8x8_Intra
11	SI_8x8_Inter_Cr	8×8	Inter	Cr	i=9に対するスケールリングリスト	i=9に対するスケールリングリスト	Default_8x8_Inter

表 7-3/JT-H264 はデフォルトのスケールリングリスト Default\_4x4\_Intra と Default\_4x4\_Inter を規定する。表 7-4/JT-H264 はデフォルトのスケールリングリスト Default\_8x8\_Intra と Default\_8x8\_Inter を規定する。

表7-3/JT-H264 デフォルトのスケールリングリストDefault\_4x4\_IntraとDefault\_4x4\_Interの規定  
(ITU-T H.264)

idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Default_4x4_Intra[ idx ]	6	13	13	20	20	20	28	28	28	28	32	32	32	37	37	42
Default_4x4_Inter[ idx ]	10	14	14	20	20	20	24	24	24	24	27	27	27	30	30	34

表7-4/JT-H264 デフォルトのスケールリングリストDefault\_8x8\_IntraとDefault\_8x8\_Interの規定  
(ITU-T H.264)

idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Default_8x8_Intra[ idx ]	6	10	10	13	11	13	16	16	16	16	18	18	18	18	18	23
Default_8x8_Inter[ idx ]	9	13	13	15	13	15	17	17	17	17	19	19	19	19	19	21

表7-4/JT-H264 (つづき) デフォルトのスケールリングリストDefault\_8x8\_IntraとDefault\_8x8\_Interの規定  
(ITU-T H.264)

idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Default_8x8_Intra[ idx ]	23	23	23	23	23	25	25	25	25	25	25	25	27	27	27	27
Default_8x8_Inter[ idx ]	21	21	21	21	21	22	22	22	22	22	22	22	24	24	24	24

表7-4/JT-H264 (つづき) デフォルトのスケーリングリストDefault\_8x8\_IntraとDefault\_8x8\_Interの規定  
(ITU-T H.264)

idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
Default_8x8_Intra[ idx ]	27	27	27	27	29	29	29	29	29	29	29	31	31	31	31	31
Default_8x8_Inter[ idx ]	24	24	24	24	25	25	25	25	25	25	25	27	27	27	27	27

表7-4/JT-H264 (完) デフォルトのスケーリングリストDefault\_8x8\_IntraとDefault\_8x8\_Interの規定  
(ITU-T H.264)

idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Default_8x8_Intra[ idx ]	31	33	33	33	33	33	36	36	36	36	38	38	38	40	40	42
Default_8x8_Inter[ idx ]	27	28	28	28	28	28	30	30	30	30	32	32	32	33	33	35

log2\_max\_frame\_num\_minus4 は、frame\_num に関連した導出の中で使用される変数 MaxFrameNum の値を、以下のように規定する：

$$\text{MaxFrameNum} = 2^{(\text{log2\_max\_frame\_num\_minus4})} \quad (7-9)$$

log2\_max\_frame\_num\_minus4 の値は、0 から 12 のそれぞれの値を含む範囲になければならない。

pic\_order\_cnt\_type は、(8.2.1 小節において規定されるような) ピクチャ順序カウントを復号する手法を規定する。pic\_order\_cnt\_type の値は、0 から 2 のそれぞれの値を含む範囲になければならない。

pic\_order\_cnt\_type は、下記のうちのいずれかを含む符号化ビデオシーケンス内では 2 に等しくはならない。

- 非参照ピクチャを含むアクセスユニットが直後に続く、非参照フレームを含むアクセスユニット
- 非参照ピクチャを含むアクセスユニットが直後に続く、各々フィールドを含み、それら 2 つのフィールドがともに相補的非参照フィールドペアを形成するような 2 つのアクセスユニット
- 2 つのアクセスユニットの 1 番目とともに、相補的非参照フィールドペアを形成しない、2 番目の非参照ピクチャを含むアクセスユニットが直後に続く、1 番目の非参照フィールドを含むアクセスユニット

log2\_max\_pic\_order\_cnt\_lsb\_minus4 は、8.2.1 小節で規定されるようなピクチャ順序カウントに対する復号処理で使用される変数 MaxPicOrderCntLsb の値を以下のように規定する：

$$\text{MaxPicOrderCntLsb} = 2^{(\text{log2\_max\_pic\_order\_cnt\_lsb\_minus4})} \quad (7-10)$$

log2\_max\_pic\_order\_cnt\_lsb\_minus4 の値は、0 から 12 のそれぞれの値を含む範囲になければならない。

**delta\_pic\_order\_always\_zero\_flag** が 1 に等しいことは、**delta\_pic\_order\_cnt[0]**および **delta\_pic\_order\_cnt[1]** がシーケンスのスライスヘッダ内に存在せず、0 に等しいと推定されなければならないことを規定する。**delta\_pic\_order\_always\_zero\_flag** が 0 に等しいことは、**delta\_pic\_order\_cnt[0]**がシーケンスのスライスヘッダ内に存在すること、および **delta\_pic\_order\_cnt[1]**がシーケンスのスライスヘッダ内に存在するかもしれないことを規定する。

**offset\_for\_non\_ref\_pic** は、8.2.1 小節に規定されるような非参照ピクチャのピクチャ順序カウンタを計算するために使用される。**offset\_for\_non\_ref\_pic** の値は、 $-2^{31}$  から  $2^{31}-1$  のそれぞれの値を含む範囲になければならない。

**offset\_for\_top\_to\_bottom\_field** は、8.2.1 小節に規定されるようなボトムフィールドのピクチャ順序カウンタを計算するために使用される。**offset\_for\_top\_to\_bottom\_field** の値は、 $-2^{31}$  から  $2^{31}-1$  のそれぞれの値を含む範囲になければならない。

**num\_ref\_frames\_in\_pic\_order\_cnt\_cycle** は、8.2.1 小節に規定されるようなピクチャ順序カウンタのために、復号処理の中で使用される。**num\_ref\_frames\_in\_pic\_order\_cnt\_cycle** の値は、0 から 255 のそれぞれの値を含む範囲になければならない。

**offset\_for\_ref\_frame[i]** は、8.2.1 小節に規定されるようなピクチャ順序カウンタのために、復号処理の中で使用される **num\_ref\_frames\_in\_pic\_order\_cnt\_cycle** 個のリストの要素である。**offset\_for\_ref\_frame[i]** の値は、 $-2^{31}$  から  $2^{31}-1$  のそれぞれの値を含む範囲になければならない。

**num\_ref\_frames** は、シーケンスにおける任意のピクチャの INTER 予測のために復号処理で使用されるかもしれない短期間と長期間参照フレーム、相補的参照フィールドペア、および、非ペア参照フィールドの最大数を規定する。**num\_ref\_frames** はまた、8.2.5.3 小節において規定されるようなスライディングウィンドウ動作のサイズを決定する。**num\_ref\_frames** の値は、0 から **MaxDpbSize** (A.3.1 あるいは A.3.2 小節で規定されるような) のそれぞれの値を含む範囲になければならない。

**gaps\_in\_frame\_num\_value\_allowed\_flag** は、7.4.3 小節で規定される **frame\_num** の許容値と、8.2.5.2 小節で規定される **frame\_num** 値の間の推定されるギャップが存在するときの復号処理を規定する。

**pic\_width\_in\_mbs\_minus1** に 1 を加えたものは、マクロブロック単位でそれぞれの復号ピクチャの幅を規定する。

マクロブロック単位のピクチャ幅に対する変数は、以下のように導出される：

$$\text{PicWidthInMbs} = \text{pic\_width\_in\_mbs\_minus1} + 1 \quad (7-11)$$

輝度成分に対するピクチャ幅のための変数は、以下のように導出される：

$$\text{PicWidthInSamples}_l = \text{PicWidthInMbs} * 16 \quad (7-12)$$

色差成分に対するピクチャ幅のための変数は、以下のように導出される：

$$\text{PicWidthInSamples}_c = \text{PicWidthInMbs} * \text{MbWidthC} \quad (7-13)$$

**pic\_height\_in\_map\_units\_minus1** に 1 を加えたものは、復号フレームまたはフィールドのスライスグループマップユニット単位における高さを規定する。

変数 **PicHeightInMapUnits** および **PicSizeInMapUnits** は、以下のように導出される：

$$\text{PicHeightInMapUnits}=\text{pic\_height\_in\_map\_units\_minus1}+1 \quad (7-14)$$

$$\text{PicSizeInMapUnits}=\text{PicWidthInMbs}*\text{PicHeightInMapUnits} \quad (7-15)$$

**frame\_mbs\_only\_flag** が 0 に等しいことは、符号化ビデオシーケンスの符号化ピクチャは、符号化フィールドまたは符号化フレームのどちらでもいいことを規定する。**frame\_mbs\_only\_flag** が 1 に等しいことは、符号化ビデオシーケンスの全ての符号化ピクチャは、フレームマクロブロックのみを含む符号化フレームであることを規定する。

**pic\_width\_in\_mbs\_minus1**、**pic\_height\_in\_map\_units\_minus1** および **frame\_mbs\_only\_flag** に対する値の許容範囲は、付属資料 A における制約によって規定される。

**frame\_mbs\_only\_flag** に従って、次の通り、意味が **pic\_height\_in\_map\_units\_minus1** に割り当てられる：

- もし **frame\_mbs\_only\_flag** が 0 に等しいならば、**pic\_height\_in\_map\_units\_minus1** 足す 1 はマクロブロック単位でフィールドの高さである。
- それ以外 (**frame\_mbs\_only\_flag** は 1 に等しい)、**pic\_height\_in\_map\_units\_minus1** 足す 1 はマクロブロック単位でフレームの高さである。

変数 **FrameHeightInMbs** は、次のように導出される：

$$\text{FrameHeightInMbs}=(2-\text{frame\_mbs\_only\_flag})*\text{PicHeightInMapUnits} \quad (7-16)$$

**mb\_adaptive\_frame\_field\_flag** が 0 に等しいことは、ピクチャ内でフレームおよびフィールドマクロブロックの切り替えなしを規定する。**mb\_adaptive\_frame\_field\_flag** が 1 に等しいことは、フレーム内でフレームおよびフィールドマクロブロックの切り替え使用可能を規定する。**mb\_adaptive\_frame\_field\_flag** が存在しない場合、0 に等しいと推定されなければならない。

**direct\_8x8\_inference\_flag** は、8.4.1.2 小節において規定されるような **B\_Skip**、**B\_Direct\_16x16** および **B\_Direct\_8x8** に対する輝度動きベクトルのための導出処理において使用される手法を規定する。**frame\_mbs\_only\_flag** が 0 に等しい場合、**direct\_8x8\_inference\_flag** は 1 に等しくなければならない。

**frame\_cropping\_flag** が 1 に等しいことは、フレーム切り出しオフセットパラメータが、シーケンスパラメータセットで次に続くことを規定する。**frame\_cropping\_flag** が 0 に等しいことは、フレーム切り出しオフセットパラメータが存在しないことを規定する。

**frame\_crop\_left\_offset**、**frame\_crop\_right\_offset**、**frame\_crop\_top\_offset**、**frame\_crop\_bottom\_offset** は、出力のためにフレーム座標で規定される矩形領域によって、復号処理から出力される符号化ビデオシーケンスのピクチャのサンプルを規定する。

変数 **CropUnitX** および **CropUnitY** は次の通り導出される：

- もし **ChromaArrayType** が 0 に等しいならば、**CropUnitX** および **CropUnitY** は次のように導出される。

$$\text{CropUnitX}=1 \quad (7-17)$$

$$\text{CropUnitY}=2-\text{frame\_mbs\_only\_flag} \quad (7-18)$$

- それ以外 (**ChromaArrayType** は 1、2 あるいは 3 に等しい)、**CropUnitX** および **CropUnitY** は次のように導出される。

$$\text{CropUnitX}=\text{SubWidthC} \quad (7-19)$$

$$\text{CropUnitY}=\text{SubHeightC}*(2-\text{frame\_mbs\_only\_flag}) \quad (7-20)$$

フレーム切り出し矩形は、 $\text{CropUnitX}*\text{frame\_crop\_left\_offset}$  から  $\text{PicWidthInSamples}_L-(\text{CropUnitX}*\text{frame\_crop\_right\_offset}+1)$  のそれぞれの値を含む水平フレーム座標、および  $\text{CropUnitY}*\text{frame\_crop\_top\_offset}$  から  $(16*\text{FrameHeightInMbs})-(\text{CropUnitY}*\text{frame\_crop\_bottom\_offset}+1)$  のそれぞれの値を含む垂直フレーム座標を持つ輝度サンプルを含む。 $\text{frame\_crop\_left\_offset}$  の値は、0 から  $(\text{PicWidthInSamples}_L/\text{CropUnitX})-(\text{frame\_crop\_right\_offset}+1)$  のそれぞれの値を含む範囲になければならず、また  $\text{frame\_crop\_top\_offset}$  の値は、0 から  $(16*\text{FrameHeightInMbs}/\text{CropUnitY})-(\text{frame\_crop\_bottom\_offset}+1)$  のそれぞれの値を含む範囲になければならない。

$\text{frame\_cropping\_flag}$  が 0 に等しい場合、 $\text{frame\_crop\_left\_offset}$ 、 $\text{frame\_crop\_right\_offset}$ 、 $\text{frame\_crop\_top\_offset}$ 、および  $\text{frame\_crop\_bottom\_offset}$  の値は 0 に等しいと推定されなければならない。

$\text{ChromaArrayType}$  が 0 に等しくない場合、2つの色差配列の対応する規定されるサンプルは、フレーム座標  $(x/\text{SubWidthC}, y/\text{SubHeightC})$  を持つサンプルである。ここで  $(x, y)$  は、規定される輝度サンプルのフレーム座標である。

2つの色差配列の規定サンプルは、フレーム座標  $(x/2, y/2)$  を持つサンプルである。ここで  $(x, y)$  は、規定輝度サンプルのフレーム座標である。

復号フィールドに対して、復号フィールドの規定サンプルは、フレーム座標で規定された矩形内に収まるサンプルである。

$\text{vui\_parameters\_present\_flag}$  が 1 に等しいことは、付属資料 E において規定される  $\text{vui\_parameters}()$  シンタックス構造が存在することを規定する。 $\text{vui\_parameters\_present\_flag}$  が 0 に等しいことは、付属資料 E において規定される  $\text{vui\_parameters}()$  シンタックス構造が存在しないことを規定する。

#### 7.4.2.1.1.1 スケーリングリストの意味

$\text{delta\_scale}$  は、0 から  $\text{sizeofScalingList}-1$  のそれぞれの値を含む範囲にある  $j$  について、スケーリングリストの  $j$  番目の要素を導出するため使用される。 $\text{delta\_scale}$  の値は、-128 から +127 のそれぞれの値を含む範囲になければならない。

$\text{useDefaultScalingMatrixFlag}$  が 1 に等しいと導出される場合、スケーリングリストは表 7-2/JT-H264 で規定されるデフォルトスケーリングリストに等しいと推定されなければならない。

#### 7.4.2.1.2 シーケンスパラメータセット拡張RBSPシンタックスの意味

$\text{seq\_parameter\_set\_id}$  は、シーケンスパラメータセット拡張に関連付けられたシーケンスパラメータセットを識別する。 $\text{seq\_parameter\_set\_id}$  の値は、0 から 31 のそれぞれの値を含む範囲になければならない。

$\text{aux\_format\_idc}$  が 0 に等しいことは、符号化ビデオシーケンス中に補助符号化ピクチャが存在しないことを示す。 $\text{aux\_format\_idc}$  が 1 に等しいことは、符号化ビデオシーケンスの各々のアクセスユニット中にちょうど 1 つの補助符号化ピクチャが存在し、アルファブレンディングの目的の為、各々のアクセスユニットで関連付けられた主符号化ピクチャの復号サンプルは、そのアクセスユニットで復号処理から出力された後の表示処理で、補助符号化ピクチャの解釈サンプル値が乗算されるべきであることを示す。 $\text{aux\_format\_idc}$  が 2 に等しいことは、符号化ビデオシーケンスの各々のアクセスユニット中にちょうど 1 つの補助符号化ピクチャが存在し、アルファブレンディングの目的の為、各々のアクセスユニットで関連付けられた主符号化ピク

チャの復号サンプルは、そのアクセスユニットで復号処理から出力された後の表示処理で、補助符号化ピクチャの解釈サンプル値が乗算されるべきでないことを示す。aux\_format\_idc が 3 に等しいことは、符号化ビデオシーケンスの各々のアクセスユニット中にちょうど 1 つの補助符号化ピクチャが存在し、補助符号化ピクチャの使用法は無規定であることを示す。aux\_format\_idc の値は、0 から 3 のそれぞれの値を含む範囲になければならない。aux\_format\_idc の 3 より大きい値は、将来 TTC により規定される目的に対して、符号化ビデオシーケンスの各々のアクセスユニット中にちょうど 1 つの補助符号化ピクチャが存在することを示すために予約される。aux\_format\_idc が存在しない場合、それは 0 に等しいと推定されなければならない。

記1 本標準に適合するビデオ復号器は、補助符号化ピクチャを復号することを要求されない。

bit\_depth\_aux\_minus8 は、補助符号化ピクチャのサンプル配列のサンプルのビット深度を規定する。bit\_depth\_aux\_minus8 は、0 から 4 のそれぞれの値を含む範囲になければならない。

alpha\_incr\_flag が 0 に等しいことは、アルファブレンディングの目的に対して、各々の復号された補助符号化ピクチャサンプル値に対する解釈サンプル値は、復号された補助符号化ピクチャサンプル値に等しいことを示す。alpha\_incr\_flag が 1 に等しいことは、アルファブレンディングの目的に対して、補助符号化ピクチャサンプルの復号後に、Min(alpha\_opaque\_value,alpha\_transparent\_value)より大きい任意の補助符号化ピクチャサンプル値は、補助符号化ピクチャサンプルに対して解釈サンプル値を得るために 1 が加算されるべきであり、また Min(alpha\_opaque\_value,alpha\_transparent\_value)より小さいかまたは等しい任意の補助符号化ピクチャサンプル値は、その復号された補助符号化ピクチャサンプル値に対する解釈サンプル値として変更無しで使用されるべきである。

alpha\_opaque\_value は、アルファブレンディングの目的のため同じアクセスユニットの関連付けられる輝度と色差のサンプルが不透明であると見なされる補助符号化ピクチャサンプルの解釈サンプル値を規定する。シンタックス要素 alpha\_opaque\_value を表すのに使用されるビットの数は bit\_depth\_aux\_minus8+9 ビットである。

alpha\_transparent\_value は、アルファブレンディングの目的のため同じアクセスユニットの関連付けられる輝度と色差のサンプルが透明であると見なされる補助符号化ピクチャサンプルの解釈サンプル値を規定する。シンタックス要素 alpha\_transparent\_value を表すのに使用されるビットの数は bit\_depth\_aux\_minus8+9 ビットである。

alpha\_incr\_flag が 1 に等しい場合、alpha\_transparent\_value は alpha\_opaque\_value に等しくはならず、また  $\text{Log}_2(\text{Abs}(\text{alpha\_opaque\_value}-\text{alpha\_transparent\_value}))$  は整数の値を持たなくてはならない。alpha\_opaque\_value に等しい alpha\_transparent\_value の値は、補助符号化ピクチャがアルファブレンディング目的を意図されないことを示す。

記2 アルファブレンディング目的において、alpha\_opaque\_valueはalpha\_transparent\_valueよりも大きいかもしれないし、あるいはalpha\_transparent\_valueより小さいかもしれない。解釈サンプル値は、alpha\_opaque\_valueからalpha\_transparent\_valueのそれぞれの値を含む範囲にクリップされるべきである。

シーケンスパラメータセット拡張の復号と補助符号化ピクチャの復号は、本 TTC 標準の適合では要求されない。

補助符号化ピクチャの各々の符号化スライスのシンタックスは、次に示す制約の違いとともに、冗長ピクチャの符号化スライスと同様の制約に従わなければならない。

- 主符号化ピクチャが IDR ピクチャかどうかによって、次が適用される：
  - もし主符号化ピクチャが IDR ピクチャならば、補助符号化スライスシンタックスは 5 に等しい

nal\_unit\_type をもつスライス (IDR ピクチャのスライス) のそれに対応しなければならない。

- それ以外(主符号化ピクチャが IDR ピクチャではない)、補助符号化スライスシンタックスは 1 に等しい nal\_unit\_type をもつスライス(非 IDR ピクチャのスライス)のそれに対応しなければならない。
- 補助符号化ピクチャのスライス(存在する時)は、主符号化ピクチャのそれらに対応する全てのマクロブロックを含まなければならない。
- redundant\_pic\_cnt は、全ての補助符号化スライスにおいて 0 に等しくなければならない。

補助符号化ピクチャの復号における(オプションの)復号処理は、補助符号化ピクチャが、現在の符号化ビデオストリームの主符号化ピクチャと以下に示す方法において異なる、分離された符号化ビデオストリームの主符号化ピクチャであるかのように同じである。

- 各々の補助符号化ピクチャの IDR あるいは非 IDR 状態は、nal\_ref\_idc の値から推定されるのではなく、同じアクセスユニット中の主符号化ピクチャの IDR あるいは非 IDR 状態と同じであると推定されなければならない。
- chroma\_format\_idc の値および ChromaArrayType の値は、補助符号化ピクチャの復号において 0 に等しいと推定されなければならない。
- bit\_depth\_luma\_minus8 の値は、補助符号化ピクチャの復号において bit\_depth\_aux\_minus8 に等しいと推定されなければならない。

記3 アルファブレンディング合成は、普通は全て同じ大きさのバックグラウンドピクチャB、フォアグラウンドピクチャF、そして復号された補助符号化ピクチャAで実行される。例の説明目的で、BとFの色差の解像度は、輝度と同等の解像度にアップサンプリングされると仮定する。B、FとAの対応するサンプルをそれぞれb、fとaで記す。輝度と色差のサンプルは下付きのY、CbとCrで記す。

変数alphaRange、alphaFwtそしてalphaBwtを次の通り定義する。

$$\text{alphaRange}=\text{Abs}(\text{alpha\_opaque\_value}-\text{alpha\_transparent\_value})$$

$$\text{alphaFwt}=\text{Abs}(a-\text{alpha\_transparent\_value})$$

$$\text{alphaBwt}=\text{Abs}(a-\text{alpha\_opaque\_value})$$

次に、アルファブレンディング合成において、表示されるピクチャDのサンプルdは次のように計算されてもよい。

$$d_Y=(\text{alphaFwt}*f_Y+\text{alphaBwt}*b_Y+\text{alphaRange}/2)/\text{alphaRange}$$

$$d_{CB}=(\text{alphaFwt}*f_{CB}+\text{alphaBwt}*b_{CB}+\text{alphaRange}/2)/\text{alphaRange}$$

$$d_{CR}=(\text{alphaFwt}*f_{CR}+\text{alphaBwt}*b_{CR}+\text{alphaRange}/2)/\text{alphaRange}$$

ピクチャD、FそしてBのサンプルはまた、赤、緑そして青の成分値 (E.2.1小節参照) を表現することも可能ではある。ここではY、Cb、Cr成分の値を仮定した。上記例の説明目的では、各々の成分、例えばYは各々のピクチャD、FそしてBで同じビット深度を持つことが仮定される。しかしながらこの例では、異なった成分、例えばYとCbは同じビット深度を持つ必要はない。

aux\_format\_idcが1に等しい場合、Fとは復号された輝度と色差から得られる復号ピクチャであろう。そしてAとは復号された補助符号化ピクチャから得られる復号ピクチャであろう。この場合、示されたアルファブレンディング合成例は、FのサンプルにAのサンプルから得られた要素を掛けることを意味する。

編集あるいはダイレクトビューに有用で一般に使用されるあるピクチャフォーマットは、プリマルチプライドブラックビデオと呼ばれる。もしフォアグラウンドピクチャがFならば、プリマルチプライドブラックビデオSは次のように与えられる。

$$s_Y=(\text{alphaFwt}*f_Y)/\text{alphaRange}$$

$$s_{CB}=(\text{alphaFwt}*f_{CB})/\text{alphaRange}$$

$$s_{CR}=(\alpha Fwt*f_{CR})/\alpha Range$$

プリマルチプライドブラックビデオは、黒のバックグラウンドに対して表示されるなら、ピクチャSが正しく見えるという特徴をもつ。非黒バックグラウンドBに対しては、表示されるピクチャDの合成は以下のように計算されてもよい。

$$d_Y=s_Y+(\alpha Bwt*b_Y+\alpha Range/2)/\alpha Range$$

$$d_{CB}=s_{CB}+(\alpha Bwt*b_{CB}+\alpha Range/2)/\alpha Range$$

$$d_{CR}=s_{CR}+(\alpha Bwt*b_{CR}+\alpha Range/2)/\alpha Range$$

aux\_format\_idcが2に等しい場合、Sとは復号された輝度と色差から得られる復号ピクチャであろう。そしてAとは再び、復号された補助符号化ピクチャから得られる復号ピクチャであろう。この場合、アルファブレンディング合成は、SのサンプルにAのサンプルから得られた要素を掛けることにはならない。

**additional\_extension\_flag** が0に等しいことは、シーケンスパラメータセット拡張シンタックス構造の中で、RBSP 追従ビットより前に、追加データが後に続かないことを示す。additional\_extension\_flag の値は0に等しくなければならない。additional\_extension\_flag における1の値は、TTCによる将来の使用のため予約される。本標準に適合する復号器は、シーケンスパラメータセット拡張 NAL ユニットの additional\_extension\_flag に対する値1の後に続く全てのデータを無視しなければならない。

#### 7.4.2.2 ピクチャパラメータセットRBSPの意味

**pic\_parameter\_set\_id** は、スライスヘッダ内で参照されるピクチャパラメータセットを識別する。pic\_parameter\_set\_id の値は、0 から 255 のそれぞれの値を含む範囲になければならない。

**seq\_parameter\_set\_id** はアクティブなシーケンスパラメータセットを参照する。seq\_parameter\_set\_id の値は、0 から 31 のそれぞれの値を含む範囲になければならない。

**entropy\_coding\_mode\_flag** は、2つの記述子がシンタックス表に現われるシンタックス要素に対して適用されるべきエントロピ復号手法を次の通り選択する。

- もし entropy\_coding\_mode\_flag が0に等しいならば、シンタックス表における左の記述子によって規定される手法が適用される（指数 Golomb 符号、9.1節を参照、あるいは CAVLC、9.2節を参照）。
- それ以外（entropy\_coding\_mode\_flag は1に等しい）、シンタックス表において右の記述子によって規定される手法が適用される（CABAC、9.3節を参照）。

**pic\_order\_present\_flag** が1に等しいことは、ピクチャ順序カウントに関連するシンタックス要素が、7.3.3小節において規定されるようなスライスヘッダ内に存在することを規定する。pic\_order\_present\_flag が0に等しいことは、ピクチャ順序カウントに関連するシンタックス要素がスライスヘッダ内に存在しないことを規定する。

**num\_slice\_groups\_minus1** に1を加えたものはピクチャに対するスライスグループの数を規定する。num\_slice\_groups\_minus1 が0に等しい場合、ピクチャの全てのスライスが同じスライスグループに属す。num\_slice\_groups\_minus1 の許容範囲は、付属資料Aにおいて規定される。

**slice\_group\_map\_type** は、スライスグループへのスライスグループマップユニットのマップがどのように符号化されるのかを規定する。slice\_group\_map\_type の値は、0 から 6 のそれぞれの値を含む範囲になければならない。

0に等しい slice\_group\_map\_type は、インタリーブされたスライスグループを規定する。

1に等しい slice\_group\_map\_type は、分散したスライスグループマップを規定する。

2 に等しい `slice_group_map_type` は、1 つ以上の“前景”スライスグループおよび“残存”スライスグループを規定する。

3、4 および 5 に等しい `slice_group_map_type` 値は、変化するスライスグループを規定する。`num_slice_groups_minus1` が 1 に等しくない場合、`slice_group_map_type` は 3、4、あるいは 5 に等しくてはならない。

6 に等しい `slice_group_map_type` は、各スライスグループマップユニットへのスライスグループの明示的な割り当てを規定する。

スライスグループマップユニットは、次の通り規定される：

- もし `frame_mbs_only_flag` が 0 に等しく、`mb_adaptive_frame_field_flag` が 1 に等しく、また、符号化ピクチャがフレームであるならば、スライスグループマップユニットはマクロブロックペアを単位とする。
- それ以外、もし `frame_mbs_only_flag` が 1 に等しいか、符号化ピクチャがフィールドであるならば、スライスグループマップユニットはマクロブロックの単位とする。
- それ以外 (`frame_mbs_only_flag` は 0 に等しく、`mb_adaptive_frame_field_flag` が 0 に等しく、また、符号化ピクチャがフレームである)、スライスグループマップユニットは MBAFF フレームのフレームマクロブロックペアのように、垂直に接触する 2 つのマクロブロックを単位とする。

`run_length_minus1[i]` は、スライスグループマップユニットのラスタ走査順で  $i$  番目のスライスグループに割り当てられる、連続するスライスグループマップユニットの数を規定するために使用される。`run_length_minus1[i]` の値は、0 から `PicSizeInMapUnits-1` のそれぞれの値を含む範囲になければならない。

`top_left[i]` および `bottom_right[i]` は、矩形の左上隅と右下隅をそれぞれ規定する。`top_left[i]` および `bottom_right[i]` は、そのスライスグループマップユニットたちに対するピクチャのラスタ走査におけるスライスグループマップユニット位置である。各矩形  $i$  に対して、シンタックス要素 `top_left[i]` および `bottom_right[i]` の値は、次の制約全てに従わなければならない。

- `top_left[i]` は `bottom_right[i]` より小さいかあるいは等しくなければならず、`bottom_right[i]` は `PicSizeInMapUnits` より小さくなければならない。
- $(\text{top\_left}[i]\% \text{PicWidthInMbs})$  は、 $(\text{bottom\_right}[i]\% \text{PicWidthInMbs})$  の値よりも小さい、あるいは等しくなければならない。

`slice_group_change_direction_flag` は、`slice_group_map_type` が 3、4、あるいは 5 である場合に、マップタイプを細かく規定するために `slice_group_map_type` と共に使用される。

`slice_group_change_rate_minus1` は、変数 `SliceGroupChangeRate` を規定するために使用される。`SliceGroupChangeRate` は、スライスグループのサイズが 1 つのピクチャから次へ変わることができるスライスグループマップユニットの数における倍数を規定する。`slice_group_change_rate_minus1` の値は、0 から `PicSizeInMapUnits-1` のそれぞれの値を含む範囲になければならない。`SliceGroupChangeRate` 変数は、次のように規定される：

$$\text{SliceGroupChangeRate} = \text{slice\_group\_change\_rate\_minus1} + 1 \quad (7-21)$$

`pic_size_in_map_units_minus1` は、ピクチャにおけるスライスグループマップユニットの数を規定するために使用される。`pic_size_in_map_units_minus1` は、`PicSizeInMapUnits` から 1 を引いたものと等しくなければならない。

ならない。

**slice\_group\_id[i]**は、ラスト走査順における  $i$  番目のスライスグループマップユニットのスライスグループを識別する。**slice\_group\_id[i]**シンタックス要素のサイズは、 $\text{Ceil}(\text{Log}_2(\text{num\_slice\_groups\_minus1}+1))$  ビットである。**slice\_group\_id[i]**の値は、0 から **num\_slice\_groups\_minus1** のそれぞれの値を含む範囲になければならない。

**num\_ref\_idx\_l0\_active\_minus1** は、**num\_ref\_idx\_active\_override\_flag** がそのスライスで 0 に等しい場合に、その中でリスト 0 予測が使われる際に、そのピクチャのそれぞれのスライスを復号するために使用されなければならない参照ピクチャリスト 0 に対する最大の参照インデックスを規定する。**MbaffFrameFlag** が 1 に等しい場合、**num\_ref\_idx\_l0\_active\_minus1** はフレームマクロブロックの復号に対する最大のインデックス値であり、また、 $2 * \text{num\_ref\_idx\_l0\_active\_minus1} + 1$  はフィールドマクロブロックの復号に対する最大のインデックス値である。**num\_ref\_idx\_l0\_active\_minus1** の値は、0 から 31 のそれぞれの値を含む範囲になければならない。

**num\_ref\_idx\_l1\_active\_minus1** は、**num\_ref\_idx\_l0\_active\_minus1** の l0 およびリスト 0 を l1 とリスト 1 によってそれぞれ置き換えたものと同じ意味を持つ。

**weighted\_pred\_flag** が 0 に等しいことは、重み付け予測が P および SP スライスに適用されてはならないことを規定する。**weighted\_pred\_flag** が 1 に等しいことは、重み付け予測が P および SP スライスに適用されなければならないことを規定する。

**weighted\_bipred\_idc** が 0 に等しいことは、デフォルトの重み付け予測が B スライスに適用されなければならないことを規定する。**weighted\_bipred\_idc** が 1 に等しいことは、明示的重み付け予測が B スライスに適用されなければならないことを規定する。2 に等しい **weighted\_bipred\_idc** は、暗黙的重み付け予測が B スライスに適用されなければならないことを規定する。**weighted\_bipred\_idc** の値は、0 から 2 のそれぞれの値を含む範囲になければならない。

**pic\_init\_qp\_minus26** は、それぞれのスライスに対し、 $\text{SliceQP}_Y$  の初期値引く 26 を規定する。**slice\_qp\_delta** の非ゼロ値が復号される場合、その初期値はスライスレイヤにおいて修正され、また **mb\_qp\_delta** の非ゼロ値が復号される場合、さらにマクロブロックレイヤにおいて修正される。**pic\_init\_qp\_minus26** の値は、 $-(26 + \text{QpBdOffset}_Y)$  から +25 のそれぞれの値を含む範囲になければならない。

**pic\_init\_qs\_minus26** は、SP あるいは SI スライスにおける全てのマクロブロックに対し、 $\text{SliceQS}_Y$  の初期値引く 26 を規定する。**slice\_qs\_delta** の非ゼロ値が復号される場合、その初期値はスライスレイヤにおいて修正される。**pic\_init\_qs\_minus26** の値は、-26 から +25 のそれぞれの値を含む範囲になければならない。

**chroma\_qp\_index\_offset** は、Cb 色差成分に対する  $\text{QP}_C$  値の表のアドレッシングのために、 $\text{QP}_Y$  と  $\text{QS}_Y$  に加えられなければならないオフセットを規定する。**chroma\_qp\_index\_offset** の値は、-12 から +12 のそれぞれの値を含む範囲になければならない。

**deblocking\_filter\_control\_present\_flag** が 1 に等しいことは、デブロックフィルタの特性を制御するシンタックス要素の組が、スライスヘッダ内に存在することを規定する。**deblocking\_filter\_control\_present\_flag** が 0 に等しいことは、デブロックフィルタの特性を制御するシンタックス要素の組がスライスヘッダ内に存在せず、それらの推定された値が有効であることを規定する。

**constrained\_intra\_pred\_flag** が 0 に等しいことは、INTRA 予測が、INTRA マクロブロック予測モードを使用して符号化されるマクロブロックの予測に対して、残差データおよび、INTER マクロブロック予測モード

を使用して符号化される隣接マクロブロックの復号サンプルの使用を許可することを規定する。**constrained\_intra\_pred\_flag** が 1 に等しいことは、制約された INTRA 予測を規定し、その場合 INTRA マクロブロック予測モードを使用して符号化されるマクロブロックの予測は、残差データおよび、I または SI マクロブロックタイプから復号されたサンプルのみを使う。

**redundant\_pic\_cnt\_present\_flag** が 0 に等しいことは、**redundant\_pic\_cnt** シンタックス要素が、そのピクチャパラメータセットを (直接、あるいは対応するデータパーティション A との関係によるいずれかによって) 参照するスライスヘッダ、データパーティション B およびデータパーティション C の中に存在しないことを規定する。**redundant\_pic\_cnt\_present\_flag** が 1 に等しいことは、**redundant\_pic\_cnt** シンタックス要素が、そのピクチャパラメータセットを (直接あるいは対応するデータパーティション A との関係によるいずれかによって) 参照する全てのスライスヘッダ、データパーティション B およびデータパーティション C の中に存在することを規定する。

**transform\_8x8\_mode\_flag** が 1 に等しいことは、 $8 \times 8$  変換復号処理 (8.5 節参照) が使用されるかもしれないことを規定する。**transform\_8x8\_mode\_flag** が 0 に等しいことは、 $8 \times 8$  変換復号処理が使用されないことを規定する。**transform\_8x8\_mode\_flag** が存在しない場合、それは 0 であると推定されなければならない。

**pic\_scaling\_matrix\_present\_flag** が 1 に等しいことは、シーケンスパラメータセットで規定されたスケーリングリストを修正するためにパラメータが存在することを規定する。**pic\_scaling\_matrix\_present\_flag** が 0 に等しいことは、ピクチャに使用されるスケーリングリストが、シーケンスパラメータセットによって規定されたものと等しいと推定されなければならないことを規定する。**pic\_scaling\_matrix\_present\_flag** が存在しない場合、それは 0 に等しいと推定されなければならない。

**pic\_scaling\_list\_present\_flag[i]** が 1 に等しいことは、インデックス  $i$  に対するスケーリングリストを規定するためにスケーリングリストシンタックス構造が存在することを規定する。**pic\_scaling\_list\_present\_flag[i]** が 0 に等しいことは、スケーリングリスト  $i$  に対するシンタックス構造がピクチャパラメータセット内に存在しないことを規定し、**seq\_scaling\_matrix\_present\_flag** の値に依存して次が適用される。

- もし、**seq\_scaling\_matrix\_present\_flag** が 0 に等しいならば、表 7-2/JT-H264 で規定されるスケーリングリストフォールバック規則集合 A がインデックス  $i$  におけるピクチャレベルスケーリングリストを導出するために使用されなければならない。
- それ以外 (**seq\_scaling\_matrix\_present\_flag** が 1 に等しい)、表 7-2/JT-H264 で規定されるスケーリングリストフォールバック規則集合 B がインデックス  $i$  におけるピクチャレベルスケーリングリストを導出するために使用されなければならない。

**second\_chroma\_qp\_index\_offset** は、Cr 色差成分に対する  $QP_c$  値の表のアドレッシングのために、 $QP_y$  と  $QS_y$  に加えられなければならないオフセットを規定する。**chroma\_qp\_index\_offset** の値は、-12 から +12 のそれぞれの値を含む範囲にななければならない。

**second\_chroma\_qp\_index\_offset** が存在しない場合、それは **chroma\_qp\_index\_offset** に等しいと推定されなければならない。

記 ChromaArrayType が 0 に等しい時、**bit\_depth\_chroma\_minus8**、**chroma\_qp\_index\_offset** および **second\_chroma\_qp\_index\_offset** の値は復号処理に使用されない。特に、**separate\_colour\_plane\_flag** が 1 に等しい時、**colour\_plane\_id** が 0 に等しくないピクチャの復号に用いるオフセットを適用しない輝度量子化パラメータ導出処理の適用を含む輝度成分の復号処理を使用して (スケーリング行列の選択は除く)、各色プレーンは別個のモノクロームピクチャとして復号される。

### 7.4.2.3 付加拡張情報RBSPの意味

付加拡張情報(SEI)は、VCL NAL ユニットから符号化ピクチャのサンプルを復号するのに必要ではない情報を含む。

#### 7.4.2.3.1 付加拡張情報メッセージの意味

SEI NAL ユニットは1つまたは複数の SEI メッセージを含む。各 SEI メッセージは、SEI ペイロードのタイプを規定する変数 `payloadType` およびサイズを規定する変数 `payloadSize` から構成される。SEI ペイロードは付属資料 D で規定される。導出された SEI ペイロードのサイズ `payloadSize` は、バイト単位で規定され、SEI ペイロードのバイト数と等しくなければならない。

`ff_byte` は、これが使用されるシンタックス構造でより長い表現が必要であることを識別する `0xFF` に等しい1バイトである。

`last_payload_type_byte` は、SEI メッセージのペイロードタイプの最終バイトである。

`last_payload_size_byte` は、SEI メッセージのサイズの最終バイトである。

### 7.4.2.4 アクセスユニット境界RBSPの意味

アクセスユニット境界は、主符号化ピクチャの中に存在するスライスのタイプを指示するために、およびアクセスユニット間の境界の検出を単純化するために使用されても良い。アクセスユニット境界に関連した必須復号処理はない。

`primary_pic_type` は、`primary_pic_type` の与えられた値に対して、主符号化ピクチャの全てのスライスにおける `slice_type` 値が表 7-5/JT-H264 に列挙される集合に属することを指示している。

表7-5/JT-H264 `primary_pic_type`の意味  
(ITU-T H.264)

<code>primary_pic_type</code>	主符号化ピクチャの中に存在しても良い <code>slice_type</code> の値
0	2, 7
1	0, 2, 5, 7
2	0, 1, 2, 5, 6, 7
3	4, 9
4	3, 4, 8, 9
5	2, 4, 7, 9
6	0, 2, 3, 4, 5, 7, 8, 9
7	0, 1, 2, 3, 4, 5, 6, 7, 8, 9

### 7.4.2.5 シーケンス終了RBSPの意味

シーケンス終了 RBSP は復号順序でビットストリームの中で次に続くアクセスユニット (あれば) が IDR

アクセスユニットでなければならないことを規定する。シーケンス終了 RBSP に対する SODB と RBSP のシンタックス内容は、空である。シーケンス終了 RBSP に対して規定された必須復号処理はない。

#### 7.4.2.6 ストリーム終了RBSPの意味

ストリーム終了 RBSP は、復号順序でストリーム終了 RBSP に続くいかなる追加 NAL ユニットもビットストリームの中に存在してはならないことを示す。ストリーム終了 RBSP に対する SODB と RBSP のシンタックス内容は、空である。ストリーム終了 RBSP に対して規定された必須復号処理はない。

#### 7.4.2.7 フィルデータRBSPの意味

フィルデータ RBSP は、その値が 0xFF に等しくなければならないいくつかのバイトを含む。フィルデータ RBSP に対して規定された必須復号処理はない。

**ff\_byte** は、0xFF に等しい 1 バイトである。

#### 7.4.2.8 パーティションなしスライスレイヤRBSPの意味

パーティションなしスライスレイヤ RBSP はスライスヘッダとスライスデータから構成される。

#### 7.4.2.9 スライスデータパーティションRBSPの意味

##### 7.4.2.9.1 スライスデータパーティションA RBSPの意味

スライスデータパーティションが使用される場合、ある 1 つのスライスに対して符号化されたデータは、3 個の分離したパーティションに分割される。パーティション A はカテゴリ 2 の全てのシンタックス要素を含む。

カテゴリ 2 シンタックス要素は、**residual()**シンタックス構造中のシンタックス要素以外の、スライスヘッダおよびスライスデータのシンタックス構造中の全てのシンタックス要素を含む。

**slice\_id** はデータパーティションに関連付けられたスライスを識別する。**slice\_id** の値は次のように制約される。

- もし **separate\_colour\_plane\_flag** が 0 に等しいならば、次が適用される。
  - もし付属資料 A で規定される任意スライス順序が許されていないならば、符号化ピクチャの復号順序で最初のスライスは、0 に等しい **slice\_id** を持たなければならない。 **slice\_id** 値は符号化ピクチャの復号順序で続く各々のスライスに対して 1 ずつ増加されなければならない。
  - それ以外（任意スライス順序が許されている）、各スライスは符号化ピクチャのスライス集合内で唯一の **slice\_id** 値を持たなければならない。
- それ以外（**separate\_colour\_plane\_flag** が 1 に等しい）、次が適用される。
  - もし付属資料 A で規定される任意スライス順序が許されていないならば、符号化ピクチャの各 **colour\_plane\_id** 値を持つ復号順序で最初のスライスは、0 に等しい **slice\_id** を持たなければならない。 **slice\_id** 値は符号化ピクチャの同じ **colour\_plane\_id** 値を持つ復号順序で続く各々のスライスに対して 1 ずつ増加されなければならない。
  - それ以外（任意スライス順序が許されている）、各スライスは符号化ピクチャの同じ

colour\_plane\_id 値を持つ各スライス集合内で唯一の slice\_id 値を持たなければならない。

slice\_id の範囲は、次の通り規定される。

- もし MbaffFrameFlag が 0 に等しいならば、slice\_id は、0 から PicSizeInMbs-1 のそれぞれの値を含む範囲にななければならない。
- それ以外 (MbaffFrameFlag が 1 に等しい)、slice\_id は、0 から PicSizeInMbs/2-1 のそれぞれの値を含む範囲にななければならない。

#### 7.4.2.9.2 スライスデータパーティション B RBSP の意味

スライスデータパーティションが使用される場合、ある 1 つのスライスに対して符号化されたデータは、1 から 3 個の分離したパーティションに分割される。スライスデータパーティション B は、カテゴリ 3 の全てのシンタックス要素を含む。

カテゴリ 3 シンタックス要素は residual( ) シンタックス構造中、および、表 7-10/JT-H264 において規定されるような、総称的マクロブロックタイプ I および SI に対するシンタックス構造の内部で用いられるシンタックス構造中の全てのシンタックス要素を含む。

slice\_id は、7.4.2.9.1 小小節で規定されるのと同じ意味である。

colour\_plane\_id は、separate\_colour\_plane\_flag が 1 の時、現在のスライス RBSP に関連づけられる色プレーンを規定する。colour\_plane\_id の値は、0..2 のそれぞれの値を含む範囲にななければならない。colour\_plane\_id の 0、1 および 2 は、それぞれ Y、Cb および Cr プレーンに対応する。

記 異なる colour\_plane\_id 値を持つピクチャの復号処理の間には依存性はない。

redundant\_pic\_cnt は、主符号化ピクチャに属するスライスとスライスデータパーティションに対しては、0 に等しくなければならない。redundant\_pic\_cnt は、冗長符号化ピクチャ中の符号化スライスと符号化スライスデータパーティションに対しては 0 より大きくななければならない。redundant\_pic\_cnt が存在しない場合、その値は 0 に等しいと推定されなければならない。redundant\_pic\_cnt の値は、0 から 127 のそれぞれの値を含む範囲にななければならない。

スライスデータパーティション B RBSP の存在は次の通り規定される。

- もしスライスデータパーティション A RBSP のシンタックス要素が、あるスライスに対するスライスデータ中に何らかのカテゴリ 3 のシンタックス要素の存在を指示するならば、スライスデータパーティション A RBSP 中におけるそれらと同一値の slice\_id および redundant\_pic\_cnt を持つスライスデータパーティション B RBSP が存在しなければならない。
- それ以外 (スライスデータパーティション A RBSP のシンタックス要素が、あるスライスに対するスライスデータ中にいかなるカテゴリ 3 のシンタックス要素の存在も指示しない)、スライスデータパーティション A RBSP 中におけるそれらと同一値の slice\_id および redundant\_pic\_cnt を持つスライスデータパーティション B RBSP が存在してはならない。

#### 7.4.2.9.3 スライスデータパーティション C RBSP の意味

スライスデータパーティションが使用される場合、ある 1 つのスライスに対して符号化されたデータは、3 個の分離したパーティションに分割される。スライスデータパーティション C は、カテゴリ 4 の全てのシンタックス要素を含む。

カテゴリ 4 シンタックス要素は `residual()` シンタックス構造中、および、表 7-10/JT-H264 において規定されるような、総称的マクロブロックタイプ P および B に対するシンタックス構造の内部で用いられるシンタックス構造中の全てのシンタックス要素を含む。

`slice_id` は、7.4.2.9.1 小小小節で規定されるのと同じ意味である。

`colour_plane_id` は、7.4.2.9.2 小小小節で規定されるのと同じ意味である。

`redundant_pic_cnt` は、7.4.2.9.2 小小小節で規定されるのと同じ意味である。

スライスデータパーティション C RBSP の存在は次の通り規定される。

- もしスライスデータパーティション A RBSP のシンタックス要素が、あるスライスに対するスライスデータ中に何らかのカテゴリ 4 のシンタックス要素の存在を指示するならば、スライスデータパーティション A RBSP 中におけるそれらと同一値の `slice_id` および `redundant_pic_cnt` を持つスライスデータパーティション C RBSP が存在しなければならない。
- それ以外(スライスデータパーティション A RBSP のシンタックス要素が、あるスライスに対するスライスデータ中にいかなるカテゴリ 4 のシンタックス要素の存在も指示しない)、スライスデータパーティション A RBSP 中におけるそれらと同一値の `slice_id` および `redundant_pic_cnt` を持つスライスデータパーティション C RBSP が存在してはならない。

#### 7.4.2.10 RBSPスライス追従ビットの意味

`cabac_zero_word` は、0x0000 に等しい 2 バイトの、バイト整列されたシーケンスである。

`NumBytesInVclNALunits` は 1 つの符号化ピクチャの全ての VCL NAL ユニットに対する `NumBytesInNALunit` 値の和であるとする。

`BinCountsInNALunits` を、1 つの符号化ピクチャの全ての VCL NAL ユニットの内容を復号するために、9.3.3.2 小小小節で規定される構文解析処理関数 `DecodeBin()` が起動される回数とする。`entropy_coding_mode_flag` が 1 に等しい場合、`BinCountsInNALunits` は  $(32 \div 3) * \text{NumBytesInVclNALunits} + (\text{RawMbBits} * \text{PicSizeInMbs}) \div 32$  を超えてはならない。

記 スライスレイヤNALユニットの内容に対する復号による結果である bin の最大個数の制約は、`NumBytesInVclNALunits` の値を増加させるためのいくつかの `cabac_zero_word` シンタックス要素の挿入によって満たされ得る。各々の `cabac_zero_word` は NAL ユニットの中では(各々の `cabac_zero_word` に対して `emulation_prevention_three_byte` の包含を要求した結果として生じる NAL ユニット内容への制約結果としての)3 バイトのシーケンス 0x000003 によって表現される。

#### 7.4.2.11 RBSP追従ビットの意味

`rbsp_stop_one_bit` は 1 に等しくなければならない。

`rbsp_alignment_zero_bit` は 0 に等しくなければならない。

#### 7.4.3 スライスヘッダの意味

スライスヘッダシンタックス要素 `pic_parameter_set_id`、`frame_num`、`field_pic_flag`、`bottom_field_flag`、`idr_pic_id`、`pic_order_cnt_lsb`、`delta_pic_order_cnt_bottom`、`delta_pic_order_cnt[0]`、`delta_pic_order_cnt[1]`、`sp_for_switch_flag` および `slice_group_change_cycle` の値は、存在する場合、1 つの符号化ピクチャの全てのスライスヘッダの中で同一でなければならない。

**first\_mb\_in\_slice** はスライス中の最初のマクロブロックのアドレスを規定する。付属資料 A で規定されているような任意スライス順序が許されていない時、**first\_mb\_in\_slice** の値は、次のように制約される。

- もし **separate\_colour\_plane\_flag** が 0 ならば、**first\_mb\_in\_slice** の値は、復号順序で現スライスに先行する現ピクチャの他の全てのスライスに対する **first\_mb\_in\_slice** の値より小さくあってはならない。
- それ以外 (**separate\_colour\_plane\_flag** が 1)、**first\_mb\_in\_slice** の値は、復号順序で現スライスに先行し、同じ **colour\_plane\_id** の値を持つ現ピクチャの他の全てのスライスに対する **first\_mb\_in\_slice** の値より小さくあってはならない。

スライスの最初のマクロブロックのアドレスは、次の通り導出される。

- もし **MbaffFrameFlag** が 0 に等しいならば、**first\_mb\_in\_slice** は、スライス中の最初のマクロブロックのマクロブロックアドレスであり、**first\_mb\_in\_slice** は 0 から **PicSizeInMbs-1** のそれぞれの値を含む範囲になければならない。
- それ以外 (**MbaffFrameFlag** が 1 に等しい)、**first\_mb\_in\_slice\*2** は、スライス中の最初のマクロブロックペアのトップマクロブロックである、そのスライス中の最初のマクロブロックのマクロブロックアドレスであり、**first\_mb\_in\_slice** は、0 から **PicSizeInMbs/2-1** のそれぞれの値を含む範囲になければならない。

**slice\_type** は、表 7-6/JT-H264 によって、スライスの符号化タイプを規定する。

表 7-6/JT-H264 **slice\_type** への名前に関連付け  
(ITU-T H.264)

<b>slice_type</b>	<b>slice_type</b> の名前
0	P(Pスライス)
1	B(Bスライス)
2	I(Iスライス)
3	SP(SPスライス)
4	SI(SIスライス)
5	P(Pスライス)
6	B(Bスライス)
7	I(Iスライス)
8	SP(SPスライス)
9	SI(SIスライス)

範囲 5.9 の **slice\_type** 値は、現スライスの符号化タイプに加えて、現符号化ピクチャの他の全てのスライスが **slice\_type** の現在の値に等しい **slice\_type** 値を持つか、**slice\_type** の現在の値-5 に等しい値を持たなければならないことを規定する。

**nal\_unit\_type** が 5 に等しい (IDR ピクチャ) 時、**slice\_type** は 2、4、7 あるいは 9 のいずれかに等しくなければならない。

**num\_ref\_frames** が 0 に等しい時、**slice\_type** は 2、4、7 あるいは 9 のいずれかに等しくなければならない。

**pic\_parameter\_set\_id** は使用中のピクチャパラメータセットを規定する。**pic\_parameter\_set\_id** 値は、0 から 255 のそれぞれの値を含む範囲になければならない。

**colour\_plane\_id** は、**separate\_colour\_plane\_flag** が 1 の時、現在のスライス RBSP に関連づけられる色プレーンを規定する。**colour\_plane\_id** の値は、0 から 2 のそれぞれの値を含む範囲になければならない。**colour\_plane\_id** の 0、1 および 2 は、それぞれ Y、Cb および Cr プレーンに対応する。

記1 異なる**colour\_plane\_id**値を持つピクチャの復号処理の間には依存性はない。

**frame\_num** はピクチャに対する識別子として使用され、ビットストリームの中で  $\log_2 \max\_frame\_num\_minus4 + 4$  ビットで表現されなければならない。**frame\_num** は、次の通り制約される。

変数 **PrevRefFrameNum** は次の通り導出される。

- もし現ピクチャが IDR ピクチャであるならば、**PrevRefFrameNum** は 0 に設定される。
- それ以外(現ピクチャが、IDR ピクチャでない)、**PrevRefFrameNum** は次の通り設定される。
  - もし、8.2.5.2 小小節で規定される **frame\_num** のギャップの復号処理が、参照ピクチャを含んだ復号順序で直前のアクセスユニットに続く非参照ピクチャを含んだアクセスユニットの復号処理によって起動されたならば、**PrevRefFrameNum** は 8.2.5.2 小小節で規定される **frame\_num** のギャップの復号処理によって推定される一番最後の”非存在”な参照フレームにおける **frame\_num** の値に等しく設定される。
  - それ以外、**PrevRefFrameNum** は、参照ピクチャを含んだ復号順序で直前のアクセスユニットにおける **frame\_num** の値に等しく設定される。

**frame\_num** の値は、次の通り制約される。

- もし現ピクチャが IDR ピクチャであるならば、**frame\_num** は 0 に等しくなければならない。
- それ以外(現ピクチャが IDR ピクチャでない)、参照ピクチャを含む復号順序でその直前のアクセスユニット中の主符号化ピクチャを先行参照ピクチャと呼ぶ時、現ピクチャに対する **frame\_num** の値は下記の 3 つの全ての条件が成立しない限り **PrevRefFrameNum** に等しくあってはならない。
  - 現ピクチャと先行参照ピクチャは、復号順序で連続するアクセスユニットに属する。
  - 現ピクチャと先行参照ピクチャは、逆パリティを持つ参照フィールドである。
  - 1 つ以上の次の条件が成り立つ。
    - 先行参照ピクチャは、IDR ピクチャである。
    - 先行参照ピクチャは 5 に等しい **memory\_management\_control\_operation** シンタックス要素を含む。

記2 先行参照ピクチャが 5 に等しい **memory\_management\_control\_operation** シンタックス要素を含む場合、**PrevRefFrameNum** は 0 に等しい。

- 先行参照ピクチャに先行する主符号化ピクチャが存在し、その先行参照ピクチャに先行する主符号化ピクチャは **PrevRefFrameNum** に等しい **frame\_num** を持たない。
- 先行参照ピクチャに先行する主符号化ピクチャが存在し、その先行参照ピクチャに先行する主符号化ピクチャは参照ピクチャではない。

frame\_num の値が PrevRefFrameNum に等しくない時、以下が適用される。

- 現在“短期間参照使用”にマークされており、下記における変数 UnusedShortTermFrameNum によって得られる、いずれかの値と等しい frame\_num 値を持つ復号順序で以前の、いかなるフィールドあるいはフレームもあってはならない。

```
UnusedShortTermFrameNum=(PrevRefFrameNum+1)%MaxFrameNum
while(UnusedShortTermFrameNum!=frame_num)
    UnusedShortTermFrameNum=(UnusedShortTermFrameNum+1)%MaxFrameNum
```

(7-22)

- frame\_num の値は、次の通り制約される。
  - もし gaps\_in\_frame\_num\_value\_allowed\_flag が 0 に等しいならば、現ピクチャに対する frame\_num の値は (PrevRefFrameNum+1)%MaxFrameNum に等しくなければならない。
  - それ以外 (gaps\_in\_frame\_num\_value\_allowed\_flag が 1 に等しい)、次が適用される。
    - もし frame\_num が PrevRefFrameNum より大きいならば、以下の条件のどちらかが正しい、復号順序で以前の参照ピクチャに後続しかつ現ピクチャに先行する、任意の非参照ピクチャがビットストリームにあってはならない。
      - 非参照ピクチャに対する frame\_num の値は PrevRefFrameNum より小さい。
      - 非参照ピクチャに対する frame\_num の値は現ピクチャに対する frame\_num の値より大きい。
    - それ以外 (frame\_num が PrevRefFrameNum より小さい)、以下の条件のどちらも正しい、復号順序で以前の参照ピクチャに後続しかつ現ピクチャに先行する、任意の非参照ピクチャがビットストリームにあってはならない。
      - 非参照ピクチャに対する frame\_num の値は PrevRefFrameNum より小さい。
      - 非参照ピクチャに対する frame\_num の値は現ピクチャに対する frame\_num の値より大きい。

5 に等しい memory\_management\_control\_operation を持ったピクチャは上述のような制約された frame\_num を持たなくてはならない、そして、現ピクチャの復号とメモリ管理制御操作処理の後、7.4.1.2.4 小小小節で規定されるようなものを除いて、ピクチャは復号処理における続く全ての使用に対して 0 に等しい frame\_num を持っていたと推定されなければならない。

記3 主符号化ピクチャが IDR ピクチャでないと同時に 5 に等しい memory\_management\_control\_operation シンタックス要素を持たない場合、対応する冗長符号化ピクチャの frame\_num の値は主符号化ピクチャにおける frame\_num の値と同一である。あるいは、その冗長符号化ピクチャは 5 に等しい memory\_management\_control\_operation シンタックス要素を含み、その対応する主符号化ピクチャは IDR ピクチャである。

field\_pic\_flag が 1 に等しいことはスライスが符号化フィールドのスライスであることを規定する。field\_pic\_flag が 0 に等しいことはスライスが符号化フレームのスライスであることを規定する。field\_pic\_flag が存在しない場合、0 に等しいと推定されなければならない。

変数 MbaffFrameFlag は下記のように導出される。

```
MbaffFrameFlag=(mb_adaptive_frame_field_flag && !field_pic_flag)
```

(7-23)

マクロブロックを単位とするピクチャ高に対する変数は下記のように導出される。

$$\text{PicHeightInMbs} = \text{FrameHeightInMbs} / (1 + \text{field\_pic\_flag}) \quad (7-24)$$

輝度成分のピクチャ高に対する変数は下記のように導出される。

$$\text{PicHeightInSamples}_l = \text{PicHeightInMbs} * 16 \quad (7-25)$$

色差成分のピクチャ高に対する変数は下記のように導出される。

$$\text{PicHeightInSamples}_c = \text{PicHeightInMbs} * \text{MbHeightC} \quad (7-26)$$

現ピクチャの変数  $\text{PicSizeInMbs}$  は下記に従って導出される：

$$\text{PicSizeInMbs} = \text{PicWidthInMbs} * \text{PicHeightInMbs} \quad (7-27)$$

変数  $\text{MaxPicNum}$  は、次の通り導出される。

- $\text{field\_pic\_flag}$  が 0 に等しいならば、 $\text{MaxPicNum}$  は  $\text{MaxFrameNum}$  に等しく設定される。
- それ以外( $\text{field\_pic\_flag}$  が 1 に等しい)、 $\text{MaxPicNum}$  は  $2 * \text{MaxFrameNum}$  に等しく設定される。

変数  $\text{CurrPicNum}$  は、次の通り導出される。

- $\text{field\_pic\_flag}$  が 0 に等しいならば、 $\text{CurrPicNum}$  は  $\text{frame\_num}$  に等しく設定される。
- それ以外( $\text{field\_pic\_flag}$  が 1 に等しい)、 $\text{CurrPicNum}$  は  $2 * \text{frame\_num} + 1$  に等しく設定される。

$\text{bottom\_field\_flag}$  が 1 に等しいことは、そのスライスが符号化ボトムフィールドの一部分であることを規定する。 $\text{bottom\_field\_flag}$  が 0 に等しいことは、そのピクチャが符号化トップフィールドであることを規定する。現スライスに対してこのシンタックス要素が存在しない場合は、0 に等しいと推定されなければならない。

$\text{idr\_pic\_id}$  は、IDR ピクチャを識別する。ある IDR ピクチャの全てのスライス中の  $\text{idr\_pic\_id}$  の値は、変更されないままになければならない。復号順序に従って 2 つの連続したアクセスユニットが双方とも IDR アクセスユニットの時、最初のそのような IDR アクセスユニットのスライス内の  $\text{idr\_pic\_id}$  値は、2 番目の IDR アクセスユニットの  $\text{idr\_pic\_id}$  値と異なっていなければならない。 $\text{idr\_pic\_id}$  の値は、0 から 65535 で、それぞれの値を含む範囲になければならない。

$\text{pic\_order\_cnt\_lsb}$  は符号化フレームのトップフィールドあるいは符号化フィールドに対するモジュロ  $\text{MaxPicOrderCntLsb}$  計算でピクチャ順序カウントを規定する。 $\text{pic\_order\_cnt\_lsb}$  シンタックス要素のサイズは、 $\log_2 \text{max\_pic\_order\_cnt\_lsb\_minus4} + 4$  ビットである。 $\text{pic\_order\_cnt\_lsb}$  の値は、0 から  $\text{MaxPicOrderCntLsb} - 1$  のそれぞれの値を含む範囲になければならない。

$\text{delta\_pic\_order\_cnt\_bottom}$  は、符号化フレームのボトムフィールドとトップフィールドとの間のピクチャ順序カウントの差分を次の通り規定する。

- もし現ピクチャが 5 に等しい  $\text{memory\_management\_control\_operation}$  を含むならば、 $\text{delta\_pic\_order\_cnt\_bottom}$  の値は、 $(1 - \text{MaxPicOrderCntLsb})$  から  $2^{31} - 1$  でそれぞれの値を含む範囲になければならない。
- それ以外 (現ピクチャが 5 に等しい  $\text{memory\_management\_control\_operation}$  を含まない)、

`delta_pic_order_cnt_bottom` の値は、 $-2^{31}$  から  $2^{31}-1$  でそれぞれの値を含む範囲になければならない。

このシンタックス要素が現スライスに対するビットストリーム中に存在しない場合、0 に等しいと推定されなければならない。

`delta_pic_order_cnt[0]` は 8.2.1 小節で規定されるように、符号化フレームのトップフィールドに対して、あるいは、符号化フィールドに対して、予測されたピクチャ順序カウントからのピクチャ順序カウント差分を規定する。`delta_pic_order_cnt[0]` の値は、 $-2^{31}$  から  $2^{31}-1$  のそれぞれの値を含む範囲になければならない。現スライスに対するビットストリーム中にこのシンタックス要素が存在しない場合、0 に等しいと推定されなければならない。

`delta_pic_order_cnt[1]` は 8.2.1 小節で規定されるように、符号化フレームのボトムフィールドに対して、予測されたピクチャ順序カウントからのピクチャ順序カウント差分を規定する。`delta_pic_order_cnt[1]` の値は、 $-2^{31}$  から  $2^{31}-1$  のそれぞれの値を含む範囲になければならない。現スライスに対するビットストリーム中にこのシンタックス要素が存在しない場合、0 に等しいと推定されなければならない。

`redundant_pic_cnt` は、主符号化ピクチャに属しているスライスとスライスデータパーティションでは、0 に等しくなければならない。`redundant_pic_cnt` の値は、冗長符号化ピクチャの符号化スライスや符号化スライスデータパーティションでは、0 より大きくなければならない。ビットストリーム中に `redundant_pic_cnt` が存在しない場合、その値は 0 に等しいと推定されなければならない。`redundant_pic_cnt` の値は、0 から 127 のそれぞれの値を含む範囲になければならない。

記4 復号された主ピクチャの任意の領域と、同じアクセスユニット内の任意の冗長ピクチャに対する 8 章で規定される復号処理のアプリケーションからの結果である対応する領域は、視覚的に似た様子になるべきである。

冗長符号化ピクチャの符号化スライスあるいは符号化スライスデータパーティション中の `pic_parameter_set_id` の値は、冗長符号化ピクチャの中で使用されるピクチャパラメータセット中の `pic_order_present_flag` 値が、対応する主符号化ピクチャの中で使用されるピクチャパラメータセット中の `pic_order_present_flag` の値と等しいようなものでなければならない。

主符号化ピクチャ、および、任意の冗長符号化ピクチャ中に存在する場合、下記のシンタックス要素は同一の値を持たなければならない：`field_pic_flag`、`bottom_field_flag` と `idr_pic_id`。

アクセスユニット中のある 1 つの VCL NAL ユニットにおいて、`nal_ref_idc` の値が 0 に等しい時、同じアクセスユニット中のその他全ての VCL NAL ユニットにおいて、`nal_ref_idc` の値は 0 に等しくなければならない。

記5 上記制約は次の暗黙的な意味をも持つ。主符号化ピクチャの VCL NAL ユニットの `nal_ref_idc` の値が 0 に等しい時、任意の対応する冗長符号化ピクチャの VCL NAL ユニットに対する `nal_ref_idc` の値も 0 に等しい。それ以外(主符号化ピクチャの VCL NAL ユニットに対する `nal_ref_idc` の値が 0 より大きい)の場合、任意の対応する冗長ピクチャの VCL NAL ユニットに対する `nal_ref_idc` の値もまた 0 より大きい。

主符号化ピクチャまたは同じアクセスユニットの任意の冗長符号化ピクチャに対して起動される、8.2.5 小節において規定されるような、復号参照ピクチャマーキング処理後の参照ピクチャのマーキングステータスと `frame_num` の値は、そのアクセスユニットの主符号化ピクチャ、あるいは(主符号化ピクチャの代りに)任意の冗長符号化ピクチャが復号されたかにかかわらず同一でなくてはならない。

記6 上記制約は次の暗黙的な意味をも持つ。

もし、主符号化ピクチャが IDR ピクチャでないならば、`dec_ref_pic_marking( )` シンタックス構造の内容は、主符号化ピクチャおよび主符号化ピクチャに対応する全ての冗長符号化ピクチャの、全てのスライスヘッダ内において同一でなければならない。

それ以外(主符号化ピクチャがIDRピクチャ)、次が適用される。

もし、主符号化ピクチャに対応する冗長符号化ピクチャがIDRピクチャならば、`dec_ref_pic_marking()` シンタックス構造の内容は、主符号化ピクチャおよび主符号化ピクチャに対応する冗長符号化ピクチャの全てのスライスヘッダ内において同一でなければならない。

それ以外(主符号化ピクチャに対応する冗長ピクチャがIDRピクチャでない)、冗長ピクチャの全てのスライスヘッダは、5に等しい`memory_management_control_operation` シンタックス要素を含む`dec_ref_pic_marking()` シンタックス構造を含まなければならない、次が適用される。

もし、主符号化ピクチャの`long_term_reference_flag`の値が0に等しいならば、冗長符号化ピクチャの`dec_ref_pic_marking`シンタックス構造は、6に等しい`memory_management_control_operation`シンタックス要素を含んではならない。

それ以外(主符号化ピクチャ内における`long_term_reference_flag`の値が1に等しいならば)、冗長符号化ピクチャの`dec_ref_pic_marking`シンタックス構造は、復号順序で5、4、および6に等しい`memory_management_control_operation`シンタックス要素を含まなければならない、`max_long_term_frame_idx_plus1`の値は1に等しくなければならず、`long_term_frame_idx`の値は0に等しくなければならない。

同じアクセスユニットの任意の冗長符号化ピクチャまたは主符号化ピクチャに対する復号処理完了の後に結果として生じる `TopFieldOrderCnt` と `BottomFieldOrderCnt`(適用可能なら)の値は、アクセスユニットの主符号化ピクチャが復号されるか、または任意の(主符号化ピクチャの代わりに)冗長符号化ピクチャが復号されるかに関わらず同一でなければならない。

冗長符号化ピクチャの符号化スライスと符号化スライスデータパーティションのために、要求される復号処理はない。符号化スライスのスライスヘッダ中の `redundant_pic_cnt` が0より大きい時、復号器は、符号化スライスを破棄してもよい。しかし、任意の冗長符号化ピクチャの符号化スライスや符号化スライスデータパーティションは、主ピクチャの符号化スライスや符号化スライスデータパーティションと同じ制約に従わなければならない。

記7 復号主ピクチャ中のいくつかのサンプルが、そのシーケンス転送中のエラーやロスにより正しく復号されず、ある符号化冗長スライスが正しく復号できる時、復号器は復号された主ピクチャのサンプルを復号された冗長スライスの対応するサンプルに置き換えるべきである。1つよりも多い冗長スライスが主ピクチャの関連領域をカバーする時、もっとも低い`redundant_pic_cnt`値を持つ冗長スライスが用いられるべきである。

`redundant_pic_cnt` の同じ値を持つ冗長スライスおよびスライスデータパーティションは、同じ冗長ピクチャに属している。同じ冗長ピクチャ内の復号スライスは、ピクチャ領域全体を覆う必要はなく、そして、重なってはいけない。

`direct_spatial_mv_pred_flag` は、INTER 予測による動きベクトルと参照インデックスを導出する復号処理において使用される方法を次の通り規定する。

- もし `direct_spatial_mv_pred_flag` が1に等しいならば、8.4.1.2 小小節における `B_Skip`、`B_Direct_16x16` および `B_Direct_8x8` に対する輝度動きベクトル導出処理は、8.4.1.2.2 小小節において規定される空間方向直接モード予測を用いなければならない。
- それ以外(`direct_spatial_mv_pred_flag` が0に等しい)、8.4.1.2 小小節における `B_Skip`、`B_Direct_16x16` および `B_Direct_8x8` に対する輝度動きベクトル導出処理は8.4.1.2.3 小小節において規定される時間方向直接モード予測を用いなければならない。

`num_ref_idx_active_override_flag` が0に等しいことは、参照されるピクチャパラメータセット内で規定されるシンタックス要素 `num_ref_idx_l0_active_minus1` および `num_ref_idx_l1_active_minus1` の値が有効であることを規定する。`num_ref_idx_active_override_flag` が1に等しいことは、参照されるピクチャパラメータセット内で規定される `num_ref_idx_l0_active_minus1` および `num_ref_idx_l1_active_minus1` が、現スライスに対し

て(現スライスに対してのみ)スライスヘッダ中の次の値で上書きされることを規定する。

現スライスが P、SP、もしくは B スライスの時、かつ、**field\_pic\_flag** が 0 に等しい時、かつ、ピクチャパラメータセット中の **num\_ref\_idx\_l0\_active\_minus1** の値が 15 を超える時、**num\_ref\_idx\_active\_override\_flag** は 1 に等しくなければならない。

現スライスが B スライスの時、かつ、**field\_pic\_flag** が 0 に等しい時、かつ、ピクチャパラメータセット中の **num\_ref\_idx\_l1\_active\_minus1** の値が 15 を超える時、**num\_ref\_idx\_active\_override\_flag** は 1 に等しくなければならない。

**num\_ref\_idx\_l0\_active\_minus1** はそのスライスを復号するのに使用されなければならない参照ピクチャリスト 0 の最大参照インデックスを規定する。

**num\_ref\_idx\_l0\_active\_minus1** の範囲は、次の通り規定される。

- もし、**field\_pic\_flag**が0に等しいならば、**num\_ref\_idx\_l0\_active\_minus1**は、0から15のそれぞれの値を含む範囲になければならない。**MbaffFrameFlag**が1に等しい時、**num\_ref\_idx\_l0\_active\_minus1**は、フレームマクロブロックの復号に対する最大インデックス値であり、 $2 * \text{num\_ref\_idx\_l0\_active\_minus1} + 1$ は、フィールドマクロブロックの復号に対する最大インデックス値である。
- それ以外(**field\_pic\_flag**が1に等しい)、**num\_ref\_idx\_l0\_active\_minus1**は、0から31のそれぞれの値を含む範囲になければならない。

**num\_ref\_idx\_l1\_active\_minus1** は、**num\_ref\_idx\_l0\_active\_minus1** の説明から、10 とリスト 0 をそれぞれ 11 とリスト 1 に置き換えたものと同じ意味を持つ。

**cabac\_init\_idc** は、コンテキスト変数のための初期化処理で使われる初期化表の決定のためのインデックスを規定する。**cabac\_init\_idc** の値は、0 から 2 のそれぞれの値を含む範囲になければならない。

**slice\_qp\_delta** は、マクロブロックレイヤの **mb\_qp\_delta** 値によって変更されるまで、そのスライスの全てのマクロブロックで使用される  $QP_Y$  の初期値を規定する。そのスライスの初期  $QP_Y$  量子化パラメータは次のように計算される：

$$\text{SliceQP}_Y = 26 + \text{pic\_init\_qp\_minus26} + \text{slice\_qp\_delta} \quad (7-28)$$

**slice\_qp\_delta** の値は、 $\text{SliceQP}_Y$  が  $-\text{QpBdOffset}_Y$  から +51 のそれぞれの値を含む範囲にあるように制限されなければならない。

**sp\_for\_switch\_flag** は、SP スライス中の P マクロブロックの復号に使用される復号処理を次の通り規定する。

- もし、**sp\_for\_switch\_flag**が0に等しいならば、SPスライス中のPマクロブロックは、8.6.1小節に規定される非スイッチングピクチャに対するSP復号処理を使って、復号されなければならない。
- それ以外(**sp\_for\_switch\_flag**が1に等しい)、SPスライス中のPマクロブロックは、8.6.2小節に規定されるスイッチングピクチャに対するSPとSIの復号処理を使って、復号されなければならない。

**slice\_qs\_delta** は、SP と SI スライス中の全てのマクロブロックの  $QS_Y$  値を規定する。そのスライスの  $QS_Y$  量子化パラメータは、次のように計算される：

$$\text{QS}_Y = 26 + \text{pic\_init\_qs\_minus26} + \text{slice\_qs\_delta} \quad (7-29)$$

slice\_qs\_delta の値は、QS<sub>Y</sub> が 0 から 51 のそれぞれの値を含む範囲にあるように制限されなければならない。この QS<sub>Y</sub> の値は、SI に等しい mb\_type を持つ SI スライス中の全てのマクロブロックと、INTER に等しい予測モードをもつ SP スライス中の全てのマクロブロックの復号のために用いられる。

disable\_deblocking\_filter\_idc は、デブロックフィルタの操作がスライスのいくつかのブロックエッジ交差に対して無効化されなければならないかどうかを規定し、どのエッジに対してフィルタが無効化されるかを規定する。disable\_deblocking\_filter\_idc がそのスライスヘッダ中で存在しない時、disable\_deblocking\_filter\_idc の値は、0 に等しいと推定されなければならない。

disable\_deblocking\_filter\_idc の値は、0 から 2 のそれぞれの値を含む範囲にななければならない。

slice\_alpha\_c0\_offset\_div2 は、スライス内のマクロブロックで制御されるフィルタ操作に対する  $\alpha$  と  $t_{c0}$  デブロックフィルタ表へのアクセスに使用されるオフセットを規定する。この値から、これらの表へアドレッシングする時に適用されなくてはならないオフセットは次のように計算されなくてはならない：

$$\text{FilterOffsetA} = \text{slice\_alpha\_c0\_offset\_div2} \ll 1 \quad (7-30)$$

slice\_alpha\_c0\_offset\_div2 の値は、-6 から +6 のそれぞれの値を含む範囲になくてはならない。スライスヘッダ中に slice\_alpha\_c0\_offset\_div2 が無い時、slice\_alpha\_c0\_offset\_div2 の値は 0 に等しいと推定されなくてはならない。

slice\_beta\_offset\_div2 は、スライス内のマクロブロックで制御されるフィルタ操作に対する  $\beta$  デブロックフィルタ表へのアクセスに使用されるオフセットを規定する。この値から、デブロックフィルタの  $\beta$  表へアドレッシングする時に適用されるオフセットは次のように計算されなくてはならない：

$$\text{FilterOffsetB} = \text{slice\_beta\_offset\_div2} \ll 1 \quad (7-31)$$

slice\_beta\_offset\_div2 の値は、-6 から +6 のそれぞれの値を含む範囲になくてはならない。スライスヘッダ中に slice\_beta\_offset\_div2 が無い時、slice\_beta\_offset\_div2 の値は 0 に等しいと推定されなくてはならない。

slice\_group\_change\_cycle は、slice\_group\_map\_type が 3、4 もしくは 5 に等しい時に、スライスグループ 0 中のスライスグループマップユニットの数を導出するために使用され、次のように規定される。

$$\text{MapUnitsInSliceGroup0} = \text{Min}(\text{slice\_group\_change\_cycle} * \text{SliceGroupChangeRate}, \text{PicSizeInMapUnits}) \quad (7-32)$$

slice\_group\_change\_cycle の値は、ビットストリームにおいて下記のビット数で表される。

$$\text{Ceil}(\text{Log2}(\text{PicSizeInMapUnits} \div \text{SliceGroupChangeRate} + 1)) \quad (7-33)$$

slice\_group\_change\_cycle の値は、0 から  $\text{Ceil}(\text{PicSizeInMapUnits} \div \text{SliceGroupChangeRate})$  のそれぞれの値を含む範囲になくてはならない。

#### 7.4.3.1 参照ピクチャリスト並替の意味

シンタックス要素 reordering\_of\_pic\_nums\_idc、abs\_diff\_pic\_num\_minus1 および long\_term\_pic\_num は、初期参照ピクチャリストから、そのスライスの復号に対して使用される参照ピクチャリストへの変更を規定する。

ref\_pic\_list\_reordering\_flag\_10 が 1 に等しいことは、シンタックス要素 reordering\_of\_pic\_nums\_idc が参照

ピクチャリスト 0 を規定するために存在することを規定する。ref\_pic\_list\_reordering\_flag\_l0 が 0 に等しいことは、このシンタックス要素が存在しないことを規定する。

ref\_pic\_list\_reordering\_flag\_l0 が 1 に等しい時、ref\_pic\_list\_reordering\_flag\_l0 に続く reordering\_of\_pic\_nums\_idc が 3 に等しくない回数は num\_ref\_idx\_l0\_active\_minus1+1 を超えてはならない。

8.2.4.2 小節で規定されるように作成された初期参照ピクチャリスト中の RefPicList0[num\_ref\_idx\_l0\_active\_minus1]が“参照ピクチャ無し”に等しい場合、8.2.4.3 小節で規定されるように作成された並替されたリスト中の RefPicList0[num\_ref\_idx\_l0\_active\_minus1]が“参照ピクチャ無し”に等しくなくなるまでは ref\_pic\_list\_reordering\_flag\_l0 は 1 に等しくなければならず、reordering\_of\_pic\_nums\_idc は 3 に等しくあってはならない。

ref\_pic\_list\_reordering\_flag\_l1 が 1 に等しいことは、シンタックス要素 reordering\_of\_pic\_nums\_idc が参照ピクチャリスト 1 を規定するために存在することを規定する。ref\_pic\_list\_reordering\_flag\_l1 が 0 に等しいことは、このシンタックス要素が存在しないことを規定する。

ref\_pic\_list\_reordering\_flag\_l1 が 1 に等しい時、ref\_pic\_list\_reordering\_flag\_l1 に続く reordering\_of\_pic\_nums\_idc が 3 に等しくない回数は num\_ref\_idx\_l1\_active\_minus1+1 を超えてはならない。

slice\_type が 1 または 6 に等しいスライスの復号を行っており、8.2.4.2 小節で規定されるように作成された初期参照ピクチャリスト中の RefPicList1[num\_ref\_idx\_l1\_active\_minus1]が“参照ピクチャ無し”に等しい場合、8.2.4.3 小節で規定されるように作成された並替されたリスト中の RefPicList1[num\_ref\_idx\_l1\_active\_minus1]が“参照ピクチャ無し”に等しくなくなるまでは ref\_pic\_list\_reordering\_flag\_l1 は 1 に等しくなければならず、reordering\_of\_pic\_nums\_idc は 3 に等しくあってはならない。

reordering\_of\_pic\_nums\_idc は、abs\_diff\_pic\_num\_minus1 もしくは、long\_term\_pic\_num とともに、どの参照ピクチャが再マップされるのかを規定する。reordering\_of\_pic\_nums\_idc の値は、表 7-7/JT-H264 で規定される。ref\_pic\_list\_reordering\_flag\_l0 もしくは ref\_pic\_list\_reordering\_flag\_l1 の直後に続く最初の reordering\_of\_pic\_nums\_idc の値は、3 に等しくあってはならない。

表7-7/JT-H264 参照ピクチャリストの並替に対するreordering\_of\_pic\_nums\_idc操作  
(ITU-T H.264)

reordering_of_pic_nums_idc	並替規定
0	abs_diff_pic_num_minus1が存在し、それはピクチャ番号予測値から減じるための差分に対応する
1	abs_diff_pic_num_minus1が存在し、それはピクチャ番号予測値に加えるための差分に対応する
2	long_term_pic_numが存在し、それは参照ピクチャに対する長期間ピクチャ番号を規定する
3	初期参照ピクチャリストの並替に対するループ終了

**abs\_diff\_pic\_num\_minus1** に 1 加えたものは、リスト中の現インデックスへ移動されるピクチャのピクチャ番号とそのピクチャ番号の予測値との間の差分の絶対値を規定する。**abs\_diff\_pic\_num\_minus1** は 0 から MaxPicNum-1 の範囲になければならない。**abs\_diff\_pic\_num\_minus1** に許される値は、8.2.4.3.1 小小節で規定されるように、さらに制限される。

**long\_term\_pic\_num** は、そのリスト中の現インデックスへ移動されるピクチャの長期間ピクチャ番号を規定する。符号化フレームを復号する時、**long\_term\_pic\_num** は、“長期間参照使用”とマークされた参照フレームもしくは相補的参照フィールドペアのうちの 1 つに割り当てられた LongTermPicNum に等しくなければならない。符号化フィールドを復号する時、**long\_term\_pic\_num** は、“長期間参照使用”とマークされた参照フィールドのうちの 1 つに割り当てられた LongTermPicNum に等しくなければならない。

#### 7.4.3.2 予測重み表の意味

**luma\_log2\_weight\_denom** は、全ての輝度重み因子に対する分母の 2 を底とする対数である。**luma\_log2\_weight\_denom** の値は、0 から 7 のそれぞれの値を含む範囲になければならない。

**chroma\_log2\_weight\_denom** は、全ての色差重み因子に対する分母の 2 を底とする対数である。**chroma\_log2\_weight\_denom** の値は、0 から 7 のそれぞれの値を含む範囲になければならない。

**luma\_weight\_10\_flag** が 1 に等しいことは、リスト 0 予測の輝度成分重み因子が存在することを規定する。**luma\_weight\_10\_flag** が 0 に等しいことは、これらの重み因子が存在しないことを規定する。

**luma\_weight\_10[i]** は、RefPicList0[i]を使用するリスト 0 予測に対する輝度予測値に適用される重み因子である。**luma\_weight\_10\_flag** が 1 に等しい時、**luma\_weight\_10[i]**の値は、-128 から 127 のそれぞれの値を含む範囲になければならない。**luma\_weight\_10\_flag** が 0 に等しい時、**luma\_weight\_10[i]**は、RefPicList0[i]に対して  $2^{\text{luma\_log2\_weight\_denom}}$  に等しいと推定されなければならない。

**luma\_offset\_10[i]**は、RefPicList0[i]を使用するリスト 0 予測に対する輝度予測値に適用される付加オフセットである。**luma\_offset\_10[i]**の値は、-128 から 127 のそれぞれの値を含む範囲になければならない。**luma\_weight\_10\_flag** が 0 に等しい時、**luma\_offset\_10[i]**は、RefPicList0[i]に対して 0 に等しいと推定されなければならない。

**chroma\_weight\_10\_flag** が 1 に等しいことは、リスト 0 予測の色差予測値重み因子が存在することを規定する。**chroma\_weight\_10\_flag** が 0 に等しいことは、これらの重み因子が存在しないことを規定する。

**chroma\_weight\_10[i][j]**はCbに対して0に等しいj、および、Crに対して1に等しいjを用いるRefPicList0[i]を使用するリスト0予測に対する色差予測値に適用される重み因子である。**chroma\_weight\_10\_flag**が1に等しい時、**chroma\_weight\_10[i][j]**の値は、-128 から 127 のそれぞれの値を含む範囲になければならない。**chroma\_weight\_10\_flag**が0に等しい時、**chroma\_weight\_10[i][j]**は、RefPicList0[i]に対して  $2^{\text{chroma\_log2\_weight\_denom}}$  に等しいと推定されなければならない。

**chroma\_offset\_10[i][j]**はCbに対して0に等しいj、および、Crに対して1に等しいjを用いるRefPicList0[i]を使用するリスト0予測に対する色差予測値に適用される付加オフセットである。**chroma\_offset\_10[i][j]**の値は、-128 から 127 のそれぞれの値を含む範囲になければならない。**chroma\_weight\_10\_flag**が0に等しい時、**chroma\_offset\_10[i][j]**は、RefPicList0[i]に対して0に等しいと推定されなければならない。

**luma\_weight\_11\_flag**、**luma\_weight\_11**、**luma\_offset\_11**、**chroma\_weight\_11\_flag**、**chroma\_weight\_11**、**chroma\_offset\_11**は、それぞれ**luma\_weight\_10\_flag**、**luma\_weight\_10**、**luma\_offset\_10**、**chroma\_weight\_10\_flag**、**chroma\_weight\_10**、**chroma\_offset\_10**において、それぞれ10、リスト0、List0を11、リスト1、List1に置き換えたものと同じ意味を持つ。

### 7.4.3.3 復号参照ピクチャマーキングの意味

シンタックス要素**no\_output\_of\_prior\_pics\_flag**、**long\_term\_reference\_flag**、**adaptive\_ref\_pic\_marking\_mode\_flag**、**memory\_management\_control\_operation**、**difference\_of\_pic\_nums\_minus1**、**long\_term\_frame\_idx**、**long\_term\_pic\_num**と**max\_long\_term\_frame\_idx\_plus1**は、参照ピクチャのマーキングを規定する。

参照ピクチャのマーキングは、“参照不使用”、“短期間参照使用”、“長期間参照使用”の、3つのうちの1つのみで可能である。参照ピクチャが“参照使用”としてマークされると呼ばれる時、これは“短期間参照使用”もしくは“長期間参照使用”（両者同時ではない）としてマークされるピクチャを総称的に呼ぶ。“短期間参照使用”としてマークされる参照ピクチャは短期間参照ピクチャと呼ばれる。“長期間参照使用”としてマークされる参照ピクチャは長期間参照ピクチャと呼ばれる。

シンタックス要素**adaptive\_ref\_pic\_marking\_mode\_flag**と復号参照ピクチャマーキングのシンタックス構造の内容は、1つの符号化ピクチャの全ての符号化スライスにおいて同一でなければならない。

復号参照ピクチャマーキングのシンタックス構造のシンタックスカテゴリは次の通り推定されなければならない。

- もし復号参照ピクチャマーキングのシンタックス構造がスライスヘッダ中にあるならば、復号参照ピクチャマーキングのシンタックス構造のシンタックスカテゴリは2に等しいと推定されなければならない。
- それ以外(その復号参照ピクチャマーキングのシンタックス構造が、付属資料 D において規定される、ある復号参照ピクチャマーキング繰り返し SEI メッセージにある)、復号参照ピクチャマーキングのシンタックス構造のシンタックスカテゴリは5に等しいと推定されなければならない。

**no\_output\_of\_prior\_pics\_flag**は、復号ピクチャバッファ内の復号済みピクチャが、あるIDRピクチャの復号後どのように扱われるかを規定する。付属資料 C を見る。そのIDRピクチャがそのビットストリーム中の最初のIDRピクチャの時、**no\_output\_of\_prior\_pics\_flag**の値は、復号処理に影響を及ぼさない。そのIDRピクチャがそのビットストリーム中の最初のIDRピクチャでない時、かつ、アクティブなシーケンスパラメータセットから導出された**PicWidthInMbs**、**FrameHeightInMbs**もしくは**max\_dec\_frame\_buffering**の値が、直前のピクチャに対するアクティブなシーケンスパラメータセットから導出された**PicWidthInMbs**、**FrameHeightInMbs**もしくは**max\_dec\_frame\_buffering**と異なる時、**no\_output\_of\_prior\_pics\_flag**の実際の値に

は関係なく、no\_output\_prior\_pics\_flag が 1 に等しいと復号器によって推定されてもよい。

**long\_term\_reference\_flag** が 0 に等しいことは、MaxLongTermFrameIdx 変数が“長期間フレームインデックス無し”に等しく設定され、かつ、その IDR ピクチャが“短期間参照使用”とマークされることを規定する。long\_term\_reference\_flag が 1 に等しいことは、MaxLongTermFrameIdx 変数が 0 に等しく設定され、かつ、現 IDR ピクチャが“長期間参照使用”とマークされ、0 に等しい LongTermFrameIdx に割り当てられることを規定する。num\_ref\_frames が 0 に等しい時、long\_term\_reference\_flag は 0 に等しくなければならない。

**adaptive\_ref\_pic\_marking\_mode\_flag** は、表 7-8/JT-H264 で規定されるような現復号ピクチャの参照ピクチャマーキングモードを選択する。現在“長期間参照使用”とマークされたフレーム、相補的フィールドペア、そして非ペアフィールドの数が Max(num\_ref\_frames,1)に等しい時、adaptive\_ref\_pic\_marking\_mode\_flag は 1 に等しくなければならない。

表7-8/JT-H264 adaptive\_ref\_pic\_marking\_mode\_flagの解釈  
(ITU-T H.264)

adaptive_ref_pic_marking_mode_flag	参照ピクチャマーキングモードの規定
0	スライディングウィンドウ参照ピクチャマーキングモード：短期間参照ピクチャに対して先入れ先出し機構を提供するマーキングモード
1	適応参照ピクチャマーキングモード：“参照不使用”として参照ピクチャのマーキングを規定するため、および、長期間フレームインデックスを割り当てるためのシンタックス要素を提供する参照ピクチャマーキングモード

**memory\_management\_control\_operation** は、参照ピクチャマーキングに影響を与えるために適用される制御操作を規定する。memory\_management\_control\_operation シンタックス要素の後ろに memory\_management\_control\_operation の値によって規定された操作に対して必要なデータが続く。memory\_management\_control\_operation に関連した値と制御操作は、表 7-9/JT-H264 で規定される。memory\_management\_control\_operation シンタックス要素は、スライスヘッダにそれらが現れる順序で復号処理によって処理され、各々の memory\_management\_control\_operation に対して表現される意味制約は、その個別の memory\_management\_control\_operation が処理されるその順序内の特定位置において適用される。

memory\_management\_control\_operation の解釈のために、用語参照ピクチャは次の通り解釈される。

- もし現在のピクチャがフレームならば、用語参照ピクチャは参照フレームか、あるいは相補的参照フィールドペアを呼ぶ。
- それ以外(現在のピクチャがフィールド)、用語参照ピクチャは参照フィールドか、あるいは参照フレームのフィールドを呼ぶ。

memory\_management\_control\_operation は、memory\_management\_control\_operation が復号処理によって処理される時、規定された参照ピクチャが“短期間参照使用”とマークされない限り、スライスヘッダ中で 1 に等しくあってはならない。

memory\_management\_control\_operation は、memory\_management\_control\_operation が復号処理によって処理される時、規定された長期間ピクチャ番号が、“長期間参照使用”とマークされた参照ピクチャを参照しな

い限り、スライスヘッダ中で2に等しくあってはならない。

`memory_management_control_operation` は、`memory_management_control_operation` が復号処理によって処理される時、規定された参照ピクチャが“短期間参照使用”とマークされない限り、スライスヘッダ中で3に等しくあってはならない。

`memory_management_control_operation` は、`memory_management_control_operation` が復号処理によって処理される時、もし変数 `MaxLongTermFrameIdx` の値が“長期間フレームインデックス無し”に等しいならば、3または6に等しくあってはならない。

スライスヘッダ中に1つより多くの4に等しい `memory_management_control_operation` が存在してはいけない。

スライスヘッダ中に1つより多くの5に等しい `memory_management_control_operation` が存在してはいけない。

スライスヘッダ中に1つより多くの6に等しい `memory_management_control_operation` が存在してはいけない。

`memory_management_control_operation` は同じ復号参照ピクチャマーキングのシンタックス構造に1から3の範囲の `memory_management_control_operation` が存在しない、という場合を除いてスライスヘッダ中で5に等しくあってはならない。

5に等しい `memory_management_control_operation` は、同じスライスヘッダで6に等しい `memory_management_control_operation` に続いてはいけない。

6に等しい `memory_management_control_operation` が存在する時、同じスライスヘッダで6に等しい `memory_management_control_operation` に続く2、3または4に等しい任意の `memory_management_control_operation` は、現ピクチャを“参照不使用”としてマークされるよう規定してはならない。

記1 これらの制約は、現ピクチャが“参照不使用”としてマークされるよう規定する複数の `memory_management_control_operation` シンタックス要素の任意の組み合わせを禁止する。しかしながら、同じスライスヘッダで1度より多く他の参照ピクチャのマーク状態へ影響する `memory_management_control_operation` シンタックス要素のいくつかの別の組み合わせは許される。特に、後に同じ参照ピクチャが“参照不使用”としてマークされるよう規定する2、3、4または6に等しい `memory_management_control_operation` が同じスライスヘッダで続く、特定の短期間参照ピクチャに割り当てられる長期間フレームインデックスを規定する3に等しい `memory_management_control_operation` に対しては許される。

表7-9/JT-H264 メモリ管理制御操作値(memory\_management\_control\_operation)  
(ITU-T H.264)

memory_management_control_operation	メモリ管理制御操作
0	memory_management_control_operationシンタックス要素ループの終了
1	短期間参照ピクチャに“参照不使用”とマークする
2	長期間参照ピクチャに“参照不使用”とマークする
3	短期間参照ピクチャに“長期間参照使用”とマークし、かつ長期間フレームインデックスをそれに割り当てる
4	最大長期間フレームインデックスを規定し、かつ最大値よりも大きな長期間フレームインデックスを持つ全ての長期間参照ピクチャに“参照不使用”とマークする
5	全ての参照ピクチャに“参照不使用”とマークし、かつMaxLongTermFrameIdx変数を“長期間フレームインデックス無し”に設定
6	現ピクチャに“長期間参照使用”とマークし、かつ長期間フレームインデックスをそれに割り当てる

フィールドを復号する時、かつ、長期間フレームインデックスに、短期間参照フレームの一部もしくは短期間相補的参照フィールドペアの一部のフィールドを割り当てる 3 に等しい memory\_management\_control\_operation コマンドが存在する時、同じ長期間フレームインデックスに、同じフレームもしくは相補的参照フィールドペアのもう片方のフィールドを割り当てる他の memory\_management\_control\_operation コマンドが、同じ復号参照ピクチャマーキングのシンタックス構造に存在しなければならない。

記2 上記の要求は、3に等しいmemory\_management\_control\_operationによって参照されているフィールドが、その後“参照不使用”とマークされる場合でさえも満たされなければならない(例えば、フィールドが“参照不使用”とマークされる原因となる2に等しいmemory\_management\_control\_operationが同じスライスヘッダに存在する場合)。

相補的参照フィールドペアの最初のフィールド(復号順序に従って)が、1に等しい long\_term\_reference\_flag、もしくは6に等しい memory\_management\_control\_operation コマンドを含む時、相補的参照フィールドペアのもう片方のフィールドに対する復号参照ピクチャマーキングのシンタックス構造は、同じ長期間フレームインデックスにもう片方のフィールドを割り当てる6に等しい memory\_management\_control\_operation コマンドを含まなければならない。

記3 上記の要求は、相補的参照フィールドペアの最初のフィールドがその後“参照不使用”とマークされる場合でさえも満たされなければならない(例えば、最初のフィールドが“参照不使用”とマークされる原因となる2に等しいmemory\_management\_control\_operationが2番目のフィールドのスライスヘッダに存在する場合)。

**difference\_of\_pic\_nums\_minus1** は、短期間参照ピクチャに長期間フレームインデックスを割り当てるために、あるいは“参照不使用”として短期間参照ピクチャをマークするために (**memory\_management\_control\_operation** が 3 もしくは 1 に等しい時)使用される。関連付けられた **memory\_management\_control\_operation** が復号処理で処理される時、**difference\_of\_pic\_nums\_minus1** から導出され、結果として得られるピクチャ番号は、“参照使用”としてマークされ、以前に長期間フレームインデックスが割り当てられていない参照ピクチャの 1 つに割り当てられたピクチャ番号でなければならない。

結果として得られるピクチャ番号は次の通り制約される。

- もし **field\_pic\_flag** が 0 に等しいならば、結果として得られるピクチャ番号は、参照フレームあるいは相補的参照フィールドペアに割り当てられたピクチャ番号集合の 1 つでなければならない。

記4 **field\_pic\_flag**が0に等しいならば、結果として得られるピクチャ番号は、両フィールドが“参照使用”としてマーク付けされている相補的参照フィールドペア、または両フィールドが“参照使用”としてマーク付けされているフレームへ割り当てられたピクチャ番号でなければならない。特に、**field\_pic\_flag**が0に等しいならば、単一フィールドが“参照使用”としてマーク付けされた非ペアフィールドまたはフレームのマーク付けは、1に等しい**memory\_management\_control\_operation**により影響を及ぼされえない。

- それ以外(**field\_pic\_flag** が 1 に等しい)、結果として得られるピクチャ番号は、参照フィールドに割り当てられたピクチャ番号集合の 1 つでなければならない。

**long\_term\_pic\_num** は、長期間参照ピクチャに“参照不使用”とマークするために使用 (**memory\_management\_control\_operation** が 2 に等しい時)される。関連付けられた **memory\_management\_control\_operation** が復号処理で処理される時、**long\_term\_pic\_num** は、“長期間参照使用”と現在マークされた参照ピクチャのうちの 1 つに割り当てられた長期間ピクチャ番号に等しくなければならない。

結果として得られる長期間ピクチャ番号は次の通り制約される。

- もし、**field\_pic\_flag** が 0 に等しいならば、結果として得られる長期間ピクチャ番号は、参照フレームもしくは相補的参照フィールドペアを割り当てられた長期間ピクチャ番号集合の 1 つでなければならない。

記5 **field\_pic\_flag**が0に等しいならば、結果として得られる長期間ピクチャ番号は、両フィールドが“参照使用”としてマーク付けされている相補的参照フィールドペア、または両フィールドが“参照使用”としてマーク付けされているフレームへ割り当てられた長期間ピクチャ番号でなければならない。特に、**field\_pic\_flag**が0に等しいならば、単一フィールドが“参照使用”としてマーク付けされた非ペアフィールドまたはフレームのマーク付けは、2に等しい**memory\_management\_control\_operation**により影響を及ぼされえない。

- それ以外(**field\_pic\_flag** が 1 に等しい)、結果として得られる長期間ピクチャ番号は、参照フィールドを割り当てられた長期間ピクチャ番号集合の 1 つでなければならない。

**long\_term\_frame\_idx** は、ピクチャに長期間フレームインデックスを割り当てるために使用 (**memory\_management\_control\_operation** が 3 もしくは 6 に等しい時)される。関連付けられた **memory\_management\_control\_operation** が復号処理で処理される時、**long\_term\_frame\_idx** の値は、0 から **MaxLongTermFrameIdx** のそれぞれの値を含む範囲になければならない。

**max\_long\_term\_frame\_idx\_plus1** から 1 引いたものは、(**max\_long\_term\_frame\_idx\_plus1** の他の値の受領まで)長期間参照ピクチャに対して許される長期間フレームインデックスの最大値を規定する。**max\_long\_term\_frame\_idx\_plus1** の値は、0 から **num\_ref\_frames** のそれぞれの値を含む範囲になければならない。

#### 7.4.4 スライスデータの意味

**cabac\_alignment\_one\_bit** は、1 に等しい 1 ビットである。

**mb\_skip\_run** は、P あるいは SP スライスを復号する時、**mb\_type** は P\_Skip であると推定されなければならない、そのマクロブロックタイプは総称的に P マクロブロックタイプとして参照され、また B スライスを復号する時、**mb\_type** は B\_Skip であると推定されなければならない、そのマクロブロックタイプは総称的に B マクロブロックタイプとして参照される、連続してスキップされるマクロブロックの数を規定する。**mb\_skip\_run** の値は、0 から **PicSizeInMbs-CurrMbAddr** のそれぞれの値を含む範囲になければならない。

**mb\_skip\_flag** が 1 に等しいことは、現マクロブロックが、P あるいは SP スライスを復号する時、**mb\_type** は P\_Skip であると推定されなければならない、そのマクロブロックタイプは総称的に P マクロブロックタイプとして参照され、また、B スライスを復号する時は、**mb\_type** は、B\_Skip であると推定されなければならない、そのマクロブロックタイプは総称的に B マクロブロックタイプとして参照されることを規定する。**mb\_skip\_flag** が 0 に等しいことは、現マクロブロックがスキップされないことを規定する。

**mb\_field\_decoding\_flag** が 0 に等しいことは、現マクロブロックペアがフレームマクロブロックペアであることを規定する。**mb\_field\_decoding\_flag** が 1 に等しいことは、そのマクロブロックペアがフィールドマクロブロックペアであることを規定する。フレームマクロブロックペアの両方のマクロブロックは、フレームマクロブロックと文書中で呼ばれ、フィールドマクロブロックペアの両方のマクロブロックは、フィールドマクロブロックと文書中で呼ばれる。

マクロブロックペアのいずれのマクロブロックに対しても **mb\_field\_decoding\_flag** が存在しない場合、**mb\_field\_decoding\_flag** の値は次の通り導出される。

- もし、同じスライス内で、現マクロブロックペアのすぐ左隣にマクロブロックペアがあるならば、**mb\_field\_decoding\_flag** の値は、現マクロブロックペアのすぐ左隣のマクロブロックペアの **mb\_field\_decoding\_flag** の値に等しいと推定されなければならない。
- それ以外、もし、同じスライス内で、現マクロブロックペアのすぐ左隣にマクロブロックペアがなく、かつ、同じスライス内で、現マクロブロックペアのすぐ上隣にマクロブロックペアがあるならば、**mb\_field\_decoding\_flag** の値は、現マクロブロックのすぐ上隣のマクロブロックペアの **mb\_field\_decoding\_flag** の値に等しいと推定されなければならない。
- それ以外(同じスライス内で、現マクロブロックペアのすぐ左隣にもすぐ上隣にもマクロブロックペアがない)、**mb\_field\_decoding\_flag** の値は 0 に等しいと推定されなければならない。

**end\_of\_slice\_flag** が 0 に等しいことは、そのスライス中に別のマクロブロックが後続していることを規定する。**end\_of\_slice\_flag** が 1 に等しいことは、スライスの最後であり、それ以降マクロブロックが後続しないことを規定する。

スライスデータシンタックス表内で使用される **NextMbAddress()** 関数は、8.2.2 小節で規定される。

#### 7.4.5 マクロブロックレイヤの意味

**mb\_type** は、マクロブロックタイプを規定する。**mb\_type** の意味は、そのスライスタイプに依存する。

I, SI, P, SP そして B スライスにおける様々なマクロブロックタイプに対して表および意味が規定される。各々の表は **mb\_type** の値、**mb\_type** の名前、使用されるマクロブロックパーティション数 (**NumMbPart(mb\_type)** 関数で与えられる)、マクロブロック (パーティションされない場合) あるいは第 1 のパーティションの予測モード (**MbPartPredMode(mb\_type,0)** 関数によって与えられる)、および第 2 のパーティションの予測モード (**MbPartPredMode(mb\_type,1)** 関数によって与えられる) を提示する。値が適用可能

でない場合、それは“na”によって明示される。文書中では、mb\_type の値は、マクロブロックタイプと呼ばれるかもしれず、また、MbPartPredMode( )の値 X は、文書中では、“X マクロブロック(パーティション)予測モード”によって、もしくは“X 予測マクロブロック”と呼ばれるかもしれない。

表 7-10/JT-H264 に、各 slice\_type に対して許可される総称的マクロブロックタイプを示す。

記1 Bマクロブロックタイプとして分類されるPred\_L0予測モードを持ついくつかのマクロブロックタイプがある。

**表7-10/JT-H264 各スライスタイプに許可される総称的マクロブロックタイプ  
(ITU-T H.264)**

slice_type	許可される総称的マクロブロックタイプ
I(スライス)	I(表 7-11/JT-H264 を見ること)(マクロブロックタイプ)
P(スライス)	P(表 7-13/JT-H264 を見ること)およびI(表 7-11/JT-H264 を見ること)(マクロブロックタイプ)
B(スライス)	B(表 7-14/JT-H264 を見ること)およびI(表 7-11/JT-H264 を見ること)(マクロブロックタイプ)
SI(スライス)	SI(表 7-12/JT-H264 を見ること)およびI(表 7-11/JT-H264 を見ること)(マクロブロックタイプ)
SP(スライス)	P(表 7-13/JT-H264 を見ること)およびI(表 7-11/JT-H264 を見ること)(マクロブロックタイプ)

transform\_size\_8x8\_flag が 1 に等しいことは、現マクロブロックにおいて残差 8×8 ブロックにおける変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理が、輝度サンプルについて、そして ChromaArrayType==3 の時は Cb および Cr サンプルについても、起動されなければならないことを規定する。transform\_size\_8x8\_flag が 0 に等しいことは、現マクロブロックにおいて残差 4×4 ブロックにおける変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理が、輝度サンプルについて、そして ChromaArrayType==3 の時は Cb および Cr サンプルについても、起動されなければならないことを規定する。transform\_size\_8x8\_flag がビットストリーム中に存在しない場合、それは 0 に等しいと推定されなければならない。

記2 現マクロブロック予測モード MbPartPredMode(mb\_type,0) が Intra\_16x16 に等しい場合、transform\_size\_8x8\_flag はビットストリーム中には存在せず、したがって 0 に等しいと推定される。

mbPartIdx = 0..3 と索引された全ての 8×8 ブロックについて sub\_mb\_type[mbPartIdx] (7.4.5.2 小節参照) がビットストリーム中に存在する場合、変数 noSubMbPartSizeLessThan8x8Flag は、4つの 8×8 ブロックの各々において対応する SubMbPartWidth(sub\_mb\_type[mbPartIdx])と SubMbPartHeight(sub\_mb\_type[mbPartIdx])がどちらも 8 に等しいかどうかを示す。

記3 noSubMbPartSizeLessThan8x8Flag が 0 に等しく、かつ現マクロブロックタイプが I\_NxN と等しくない場合、transform\_size\_8x8\_flag はビットストリーム中には存在せず、したがって 0 に等しいと推定される。

I マクロブロックタイプとして総称的に呼ばれることがあるマクロブロックタイプは、表 7-11/JT-H264 で規定される。

I スライスのためのマクロブロックタイプは、全て I マクロブロックタイプである。

表7-11/JT-H264 Iスライスのためのマクロブロックタイプ  
(ITU-T H.264)

mb_type	mb_typeの名前	transform_size_8x8_flag	MbPartPredMode (mb_type, 0)	Intra16x16PredMode	CodedBlockPatternChroma	CodedBlockPatternLuma
0	I_NxN	0	Intra_4x4	na	等式7-34	等式7-34
0	I_NxN	1	Intra_8x8	na	等式7-34	等式7-34
1	I_16x16_0_0_0	na	Intra_16x16	0	0	0
2	I_16x16_1_0_0	na	Intra_16x16	1	0	0
3	I_16x16_2_0_0	na	Intra_16x16	2	0	0
4	I_16x16_3_0_0	na	Intra_16x16	3	0	0
5	I_16x16_0_1_0	na	Intra_16x16	0	1	0
6	I_16x16_1_1_0	na	Intra_16x16	1	1	0
7	I_16x16_2_1_0	na	Intra_16x16	2	1	0
8	I_16x16_3_1_0	na	Intra_16x16	3	1	0
9	I_16x16_0_2_0	na	Intra_16x16	0	2	0
10	I_16x16_1_2_0	na	Intra_16x16	1	2	0
11	I_16x16_2_2_0	na	Intra_16x16	2	2	0
12	I_16x16_3_2_0	na	Intra_16x16	3	2	0
13	I_16x16_0_0_1	na	Intra_16x16	0	0	15
14	I_16x16_1_0_1	na	Intra_16x16	1	0	15
15	I_16x16_2_0_1	na	Intra_16x16	2	0	15
16	I_16x16_3_0_1	na	Intra_16x16	3	0	15
17	I_16x16_0_1_1	na	Intra_16x16	0	1	15
18	I_16x16_1_1_1	na	Intra_16x16	1	1	15
19	I_16x16_2_1_1	na	Intra_16x16	2	1	15
20	I_16x16_3_1_1	na	Intra_16x16	3	1	15
21	I_16x16_0_2_1	na	Intra_16x16	0	2	15
22	I_16x16_1_2_1	na	Intra_16x16	1	2	15
23	I_16x16_2_2_1	na	Intra_16x16	2	2	15

24	I_16x16_3_2_1	na	Intra_16x16	3	2	15
25	I_PCM	na	na	na	na	na

表 7-11/JT-H264 中のマクロブロックタイプには、次の意味があてはまる。

I\_NxN : Intra\_4x4 あるいは Intra\_8x8 に等しい MbPartPredMode(mb\_type,0)を持つ mb\_type が 0 に等しいニ  
ーモニック名。

I\_16x16\_0\_0\_0、I\_16x16\_1\_0\_0、I\_16x16\_2\_0\_0、I\_16x16\_3\_0\_0、I\_16x16\_0\_1\_0、I\_16x16\_1\_1\_0、  
I\_16x16\_2\_1\_0、I\_16x16\_3\_1\_0、I\_16x16\_0\_2\_0、I\_16x16\_1\_2\_0、I\_16x16\_2\_2\_0、I\_16x16\_3\_2\_0、I\_16x16\_0\_0\_1、  
I\_16x16\_1\_0\_1、I\_16x16\_2\_0\_1、I\_16x16\_3\_0\_1、I\_16x16\_0\_1\_1、I\_16x16\_1\_1\_1、I\_16x16\_2\_1\_1、I\_16x16\_3\_1\_1、  
I\_16x16\_0\_2\_1、I\_16x16\_1\_2\_1、I\_16x16\_2\_2\_1、I\_16x16\_3\_2\_1 : マクロブロックは Intra\_16x16 予測モード  
マクロブロックとして符号化される。

それぞれの Intra\_16x16 予測マクロブロックに対して、Intra16x16 予測モードを規定する、  
Intra16x16PredMode が割り当てられ、CodedBlockPatternLuma および CodedBlockPatternChroma の値は表 7-11  
/JT-H264 に規定されるように割り当てられる。

Intra\_4x4 は、マクロブロック予測モードを規定し、8.3.1 小節で規定される Intra\_4x4 予測処理が起動され  
ることを規定する。Intra\_4x4 は、INTRA マクロブロック予測モードである。

Intra\_8x8 は、マクロブロック予測モードを規定し、8.3.2 小節で規定される Intra\_8x8 予測処理が起動され  
ることを規定する。Intra\_8x8 は、INTRA マクロブロック予測モードである。

Intra\_16x16 は、マクロブロック予測モードを規定し、8.3.3 小節で規定される Intra\_16x16 予測処理が起動  
されることを規定する。Intra\_16x16 は、INTRA マクロブロック予測モードである。

I\_PCM に等しい mb\_type で符号化されたマクロブロックに対しては、INTRA マクロブロック予測モード  
が推定されなければならない。

SI マクロブロックタイプと呼ばれることがあるマクロブロックタイプは、表 7-12/JT-H264 で規定される。

SI スライスのためのマクロブロックタイプは、表 7-12/JT-H264 と 7-11/JT-H264 で規定される。mb\_type  
の値 0 は表 7-12/JT-H264 で規定され、mb\_type の値 1 から 26 は、その mb\_type の値から 1 を減じたもので  
索引される表 7-11/JT-H264 によって規定される。

表7-12/JT-H264 SIスライスに対する値0をもつマクロブロックタイプ  
(ITU-T H.264)

mb_type	mb_typeの名前	MbPartPredMode (mb_type,0)	Intra16x16PredMode	CodedBlockPatternChroma	CodedBlockPatternLuma
0	SI	Intra_4x4	na	等式 7-34	等式 7-34

表 7-12/JT-H264 中のマクロブロックタイプには、次の意味があてはまる。SI マクロブロックは、Intra\_4x4 予測マクロブロックとして符号化される。

P マクロブロックタイプとして総称的に呼ばれることがあるマクロブロックタイプは、表 7-13/JT-H264 で規定される。

P と SP スライスのためのマクロブロックタイプは表 7-13/JT-H264 と 7-11/JT-H264 で規定される。mb\_type の値 0 から 4 は表 7-13/JT-H264 で規定され、mb\_type の値 5 から 30 は、その mb\_type の値から 5 を減じたもので索引される表 7-11/JT-H264 によって規定される。

表7-13/JT-H264 PとSPスライスにおけるマクロブロックタイプ値0から4  
(ITU-T H.264)

mb_type	mb_typeの名前	NumMbPart (mb_type)	MbPartPredMode (mb_type,0)	MbPartPredMode (mb_type,1)	MbPartWidth (mb_type)	MbPartHeight (mb_type)
0	P_L0_16x16	1	Pred_L0	na	16	16
1	P_L0_L0_16x8	2	Pred_L0	Pred_L0	16	8
2	P_L0_L0_8x16	2	Pred_L0	Pred_L0	8	16
3	P_8x8	4	na	na	8	8
4	P_8x8ref0	4	na	na	8	8
推定	P_Skip	1	Pred_L0	na	16	16

表 7-13/JT-H264 中のマクロブロックタイプには、次の意味があてはまる。

- P\_L0\_16x16 : そのマクロブロックのサンプルは、サイズ16×16の輝度サンプルからなる1つの輝度マクロブロックパーティションおよび関連づけられた色差サンプルによって予測される。
- P\_L0\_L0\_MxN(MxNは16x8もしくは8x16に置き換えられる) : そのマクロブロックのサンプルは、16×8に等しいサイズM×Nの輝度パーティション2つ、または、8×16に等しいサイズM×Nの輝度パーティション2つと、それぞれに関連づけられた色差サンプルを使用して予測される。
- P\_8x8 : それぞれのサブマクロブロックのために、付加的なシンタックス要素(sub\_mb\_type)が、その対応するサブマクロブロックのタイプ(7.4.5.2小節を見ること)を規定するビットストリーム中に存在する。
- P\_8x8ref0:P\_8x8と同じ意味を持つ。しかし、そのビットストリーム中には、参照インデックス(ref\_idx\_10)のためのシンタックス要素は存在せず、ref\_idx\_10[mbPartIdx]は、そのマクロブロック(インデックスmbPartIdxが0...3に等しい時)の全てのサブマクロブロックに対して、0に等しいと推定されなければならない。
- P\_Skip : そのビットストリーム中に、そのマクロブロックに対するそれより先のデータはない。

表 7-13/JT-H264 中のマクロブロック予測モード(MbPartPredMode())には、次の意味があてはまる。

- Pred\_L0 : INTER予測処理は、リスト0予測を使って起動されることを規定する。Pred\_L0は、INTERマクロブロック予測モードである。

B マクロブロックタイプとして総称的に呼ばれることがあるマクロブロックタイプは、表 7-14/JT-H264 で規定される。

B スライスのためのマクロブロックタイプは、表 7-14/JT-H264 と 7-11/JT-H264 で規定される。mb\_type の値 0 から 22 は表 7-14/JT-H264 で規定され、mb\_type の値 23 から 48 は、その mb\_type の値から 23 を減じたもので索引される表 7-11/JT-H264 によって規定される。

表7-14/JT-H264 Bスライスにおけるマクロブロックタイプ値0から22  
(ITU-T H.264)

mb_type	mb_typeの名前	NumMbPart (mb_type)	MbPartPredMode (mb_type,0)	MbPartPredMode (mb_type,1)	MbPartWidth (mb_type)	MbPartHeight (mb_type)
0	B_Direct_16x16	na	Direct	na	8	8
1	B_L0_16x16	1	Pred_L0	na	16	16
2	B_L1_16x16	1	Pred_L1	na	16	16
3	B_Bi_16x16	1	BiPred	na	16	16
4	B_L0_L0_16x8	2	Pred_L0	Pred_L0	16	8
5	B_L0_L0_8x16	2	Pred_L0	Pred_L0	8	16
6	B_L1_L1_16x8	2	Pred_L1	Pred_L1	16	8
7	B_L1_L1_8x16	2	Pred_L1	Pred_L1	8	16
8	B_L0_L1_16x8	2	Pred_L0	Pred_L1	16	8
9	B_L0_L1_8x16	2	Pred_L0	Pred_L1	8	16
10	B_L1_L0_16x8	2	Pred_L1	Pred_L0	16	8
11	B_L1_L0_8x16	2	Pred_L1	Pred_L0	8	16
12	B_L0_Bi_16x8	2	Pred_L0	BiPred	16	8
13	B_L0_Bi_8x16	2	Pred_L0	BiPred	8	16
14	B_L1_Bi_16x8	2	Pred_L1	BiPred	16	8
15	B_L1_Bi_8x16	2	Pred_L1	BiPred	8	16
16	B_Bi_L0_16x8	2	BiPred	Pred_L0	16	8
17	B_Bi_L0_8x16	2	BiPred	Pred_L0	8	16
18	B_Bi_L1_16x8	2	BiPred	Pred_L1	16	8
19	B_Bi_L1_8x16	2	BiPred	Pred_L1	8	16
20	B_Bi_Bi_16x8	2	BiPred	BiPred	16	8
21	B_Bi_Bi_8x16	2	BiPred	BiPred	8	16
22	B_8x8	4	na	na	8	8
推定	B_Skip	na	Direct	na	8	8

表 7-14/JT-H264 中のマクロブロックタイプには、次の意味があてはまる。

- **B\_Direct\_16x16** : ビットストリーム中にマクロブロックのための、動きベクトル差分も参照インデックスも存在しない。関数MbPartWidth(B\_Direct\_16x16)とMbPartHeight(B\_Direct\_16x16)は、直接モード予測において、8.4.1小節中の動きベクトルと参照フレームインデックスの導出処理で使用される。
- **B\_X\_16x16**(XはL0、L1もしくはBiに置き換えられる) : マクロブロックのサンプルは、サイズ16×16輝度サンプルの1つの輝度マクロブロックパーティションおよび関連づけられた色差サンプルによって予測される。XがL0もしくはL1に置き換えられるXをもつタイプB\_X\_16x16のマクロブロックにおいては、1つの動きベクトル差分と1つの参照インデックスが、そのマクロブロックのビットストリーム中に存在する。XがBiに置き換えられるXをもつタイプB\_X\_16x16のマクロブロックにおいては、2つの動きベクトル差分と2つの参照インデックスが、そのマクロブロックのビットストリーム中に存在する。
- **B\_X0\_X1\_MxN**(第1と第2のマクロブロックパーティションをさすX0、X1は、L0、L1もしくはBiに置き換えられ、MxNは16x8もしくは8x16に置き換えられる) : マクロブロックのサンプルは、16×8に等しいサイズM×Nの輝度パーティション2つ、または、8×16に等しいサイズM×Nの輝度パーティション2つとそれぞれに関連づけられた色差サンプルを使用して予測される。X0またはX1がL0もしくはL1に置き換えられるX0またはX1をもつマクロブロックパーティションにおいては、1つの動きベクトル差分と1つの参照インデックスが、ビットストリーム中に存在する。X0またはX1がBiに置き換えられるX0またはX1をもつマクロブロックパーティションにおいては、2つの動きベクトル差分と2つの参照インデックスが、そのマクロブロックパーティションのためにビットストリーム中に存在する。
- **B\_8x8** : それぞれのサブマクロブロックのために、その対応するサブマクロブロックのタイプ(7.4.5.2小節を見ること)を規定する付加的なシンタックス要素(sub\_mb\_type)が、ビットストリーム中に存在する。
- **B\_Skip** : そのビットストリーム中に、そのマクロブロックに対するそれより先のデータはない。関数MbPartWidth(B\_Skip)とMbPartHeight(B\_Skip)は、直接モード予測において、8.4.1小節中の動きベクトルと参照フレームインデックスの導出処理で使用される。

表 7-14/JT-H264 中のマクロブロック予測モード(MbPartPredMode())には、次の意味があてはまる。

- **Direct** : 動きベクトル差分と参照インデックスが、そのマクロブロック (B\_SkipあるいはB\_Direct\_16x16の場合) のためにビットストリーム中に存在しない。DirectはINTERマクロブロック予測モードである。
- **Pred\_L0** : 表7-13/JT-H264の意味を見ること。
- **Pred\_L1** : INTER予測処理は、リスト1予測を使って起動されることを規定する。Pred\_L1は、INTERマクロブロック予測モードである。
- **BiPred** : INTER予測処理は、リスト0とリスト1予測を使って起動されることを規定する。BiPredは、INTERマクロブロック予測モードである。

**pcm\_alignment\_zero\_bit** は、0 に等しい1ビットである。

**pcm\_sample\_luma[i]**はサンプル値である。最初にある **pcm\_sample\_luma[i]**値は、マクロブロック内部をラスタ走査で輝度サンプル値を表現する。これらのサンプルの各々を表現するのに必要なビットの数はBitDepth<sub>Y</sub>である。profile\_idc が 44、100、110、122 あるいは 244 に等しくない場合、pcm\_sample\_luma[i]は0に等しくあてはまらない。

**pcm\_sample\_chroma[i]**はサンプル値である。最初の MbWidthC \* MbHeightC 個の pcm\_sample\_chroma[i]値は、マクロブロック内部をラスタ走査で Cb サンプル値を表現し、残りの MbWidthC \* MbHeightC 個の pcm\_sample\_chroma[i]値は、マクロブロック内部をラスタ走査で Cr サンプル値を表現する。これらのサンプルの各々を表現するのに必要なビットの数は BitDepth<sub>C</sub> である。profile\_idc が 44、100、110、122 あるいは 244 に等しくない場合、pcm\_sample\_chroma[i]は 0 に等しくあってはならない。

**coded\_block\_pattern** は、マクロブロックの 4 個の 8×8 輝度ブロックと関連する色差ブロックのなかでどのブロックが非ゼロ変換係数レベルを含んでも良いかを規定する。coded\_block\_pattern がビットストリーム中に存在する時、変数 CodedBlockPatternLuma および CodedBlockPatternChroma が以下のごとく導出される。

$$\begin{aligned} \text{CodedBlockPatternLuma} &= \text{coded\_block\_pattern} \% 16 \\ \text{CodedBlockPatternChroma} &= \text{coded\_block\_pattern} / 16 \end{aligned} \quad (7-34)$$

マクロブロックタイプが I\_PCM に等しくない時、次が適用される。

- もしマクロブロック予測モードが Intra\_16x16 に等しいならば、次が適用される。
  - もし ChromaArrayType が 3 に等しくないならば、CodedBlockPatternLuma の値が次の場合を区別する。
    - もし CodedBlockPatternLuma が 0 に等しいならば、16x16 輝度ブロックの 4x4 ブロックの 16 個全てに対して、マクロブロックの輝度成分の全ての AC 変換係数レベルは 0 に等しい。
    - それ以外 (CodedBlockPatternLuma が 0 に等しくない)、CodedBlockPatternLuma は 15 に等しく、マクロブロックの輝度成分の少なくとも 1 つの AC 変換係数レベルは非ゼロでなければならず、16x16 輝度ブロックの 4x4 ブロックの 16 個全てに対して AC 変換係数レベルが走査される。
  - それ以外 (ChromaArrayType が 3 に等しい)、CodedBlockPatternLuma の値が次の場合を区別する。
    - もし CodedBlockPatternLuma が 0 に等しいならば、マクロブロックの輝度、Cb および Cr 成分の 4x4 ブロックの 16 個全てに対して、マクロブロックの輝度、Cb および Cr 成分の全ての AC 変換係数レベルは 0 に等しい。
    - それ以外 (CodedBlockPatternLuma が 0 に等しくない)、CodedBlockPatternLuma は 15 に等しく、マクロブロックの輝度、Cb または Cr 成分の少なくとも 1 つの AC 変換係数レベルは非ゼロでなければならず、マクロブロックの輝度、Cb および Cr 成分の 4x4 ブロックの 16 個全てに対して AC 変換係数レベルが走査される。
- それ以外 (マクロブロック予測モードが Intra\_16x16 に等しくない)、coded\_block\_pattern がビットストリーム中に存在し、次が適用される。
  - もし ChromaArrayType が 3 に等しくないならば、マクロブロックの 4 つの 8x8 輝度ブロックの 1 つに対して、CodedBlockPatternLuma の 4 つの LSB の各々が、次の場合を区別する。
    - もし CodedBlockPatternLuma の対応するビットが 0 に等しいならば、8x8 輝度ブロックの輝度変換ブロックの全ての変換係数レベルはゼロに等しい。
    - それ以外 (CodedBlockPatternLuma の対応するビットが 1 に等しい)、8x8 輝度ブロックの 1 つ以上の輝度変換ブロックの 1 つ以上の変換係数レベルは非ゼロ値でなければならず、対応する変換ブロックの変換係数レベルが走査される。

- それ以外 (ChromaArrayType が 3 に等しい)、マクロブロックの 4 つの 8x8 輝度ブロックの 1 つに対して、CodedBlockPatternLuma の 4 つの LSB の各々が、次の場合を区別する。
  - もし CodedBlockPatternLuma の対応するビットが 0 に等しいならば、8x8 輝度ブロックの輝度、Cb および Cr 変換ブロックの全ての変換係数レベルはゼロに等しい。
  - それ以外 (CodedBlockPatternLuma の対応するビットが 1 に等しい)、8x8 輝度ブロックの 1 つ以上の輝度、Cb または Cr 変換ブロックの 1 つ以上の変換係数レベルは非ゼロ値でなければならず、対応する変換ブロックの変換係数レベルが走査される。

マクロブロックタイプが I\_PCM に等しくない時、CodedBlockPatternChroma は次のように解釈される。

- もし ChromaArrayType が 0 または 3 に等しくないならば、CodedBlockPatternChroma は表 7-15 / JT-H264 で規定される。
- それ以外 (ChromaArrayType が 0 または 3 に等しい)、ビットストリームは CodedBlockPatternChroma の導出された値が 0 に等しくならぬ結果となるデータを含んではならない。

表 7-15 / JT-H264 CodedBlockPatternChroma 値の規定  
(ITU-T H.264)

CodedBlockPatternChroma	記述
0	全ての色差変換係数レベルが 0 に等しい。
1	1 つ以上の色差 DC 変換係数レベルが非ゼロ値でなければならない。 全ての色差 AC 変換係数レベルが 0 である。
2	0 以上の色差 DC 変換係数レベルが非ゼロ値である。 1 つ以上の色差 AC 変換係数レベルが非ゼロ値でなければならない。

**mb\_qp\_delta** はマクロブロックレイヤにおいて  $QP_Y$  の値を変更し得る。**mb\_qp\_delta** の復号値は  $-(26+QpBdOffset_Y/2)$  から  $+(25+QpBdOffset_Y/2)$  のそれぞれの値を含む範囲になければならない。**mb\_qp\_delta** は (P\_Skip および B\_Skip マクロブロックタイプを含め) どのマクロブロックに対しても、存在しないならば 0 に等しいと推定されなければならない。

$QP_Y$  の値は次によって導出される。

$$QP_Y = ((QP_{Y,PREV} + mb\_qp\_delta + 52 + 2 * QpBdOffset_Y) \% (52 + QpBdOffset_Y)) - QpBdOffset_Y \quad (7-35)$$

ここで、 $QP_{Y,PREV}$  は、現スライスにおける復号順序で直前のマクロブロックの輝度量子化パラメータ  $QP_Y$  である。スライスの最初のマクロブロックに対して  $QP_{Y,PREV}$  は、各々のスライスの始めに等式 7-28 によって導出される  $SliceQP_Y$  に等しく初期設定される。

$QP'_Y$  の値は次によって導出される。

$$QP'_Y = QP_Y + QpBdOffset_Y \quad (7-36)$$

#### 7.4.5.1 マクロブロック予測の意味

マクロブロックの全てのサンプルは予測される。予測モードは下記のシンタックス要素を用いて導出される。

**prev\_intra4x4\_pred\_mode\_flag**[luma4x4BlkIdx]および**rem\_intra4x4\_pred\_mode**[luma4x4BlkIdx]は4×4輝度ブロックの Intra\_4x4 予測をインデックス luma4x4BlkIdx=0..15 を使って規定する。ChromaArrayType が 3 の時、prev\_intra4x4\_pred\_mode\_flag[luma4x4BlkIdx]および rem\_intra4x4\_pred\_mode[luma4x4BlkIdx]はまた、4×4 Cb ブロックの Intra\_4x4 予測を cb4x4BlkIdx = 0..15 に等しいインデックス luma4x4BlkIdx を使い、そして 4×4 Cr ブロックの Intra\_4x4 予測を cr4x4BlkIdx = 0..15 に等しいインデックス luma4x4BlkIdx を使って規定する。

**prev\_intra8x8\_pred\_mode\_flag**[luma8x8BlkIdx]および**rem\_intra8x8\_pred\_mode**[luma8x8BlkIdx]は8×8輝度ブロックの Intra\_8x8 予測をインデックス luma8x8BlkIdx=0..3 を使って規定する。ChromaArrayType が 3 の時、prev\_intra8x8\_pred\_mode\_flag[luma8x8BlkIdx]および rem\_intra8x8\_pred\_mode[luma8x8BlkIdx]はまた、8×8 Cb ブロックの Intra\_8x8 予測を cb8x8BlkIdx = 0..3 に等しいインデックス luma8x8BlkIdx を使い、そして 8×8 Cr ブロックの Intra\_8x8 予測を cr8x8BlkIdx = 0..3 に等しいインデックス luma8x8BlkIdx を使って規定する。

**intra\_chroma\_pred\_mode** は、表 7-16/JT-H264 に示されるように、Intra\_4x4 または Intra\_16x16 予測を使用するマクロブロックで、色差に対して用いられる空間予測のタイプを規定する。intra\_chroma\_pred\_mode の値は、0 から 3 までのそれぞれの値を含む範囲になければならない。

表7-16/JT-H264 intra\_chroma\_pred\_modeと空間予測モードとの関係  
(ITU-T H.264)

intra_chroma_pred_mode	INTRA色差予測モード
0	DC
1	水平(Horizontal)
2	垂直(Vertical)
3	平面(Plane)

**ref\_idx\_10**[mbPartIdx]は、存在する場合、予測のために用いられるべき参照ピクチャの参照ピクチャリスト 0 におけるインデックスを規定する。

参照ピクチャのリスト 0 におけるインデックス ref\_idx\_10[mbPartIdx]の範囲、および、もし適用可能ならば、予測のために用いられる参照ピクチャの中のフィールドのパリティは次の通り規定される。

- もし MbaffFrameFlag が 0 に等しいかあるいは mb\_field\_decoding\_flag が 0 に等しいならば、ref\_idx\_10[mbPartIdx]の値は 0 から num\_ref\_idx\_10\_active\_minus1 の、それぞれの値を含む範囲になければならない。
- それ以外(MbaffFrameFlag が 1 に等しくかつ mb\_field\_decoding\_flag が 1 に等しい)、ref\_idx\_10[mbPartIdx]の値は 0 から 2\*num\_ref\_idx\_10\_active\_minus1+1 の、それぞれの値を含む範囲になければならない。

INTER 予測のために唯一の参照ピクチャが用いられるならば、ref\_idx\_10[mbPartIdx]の値は 0 に等しいと推定されなければならない。

**ref\_idx\_11**[mbPartIdx]は 10 およびリスト 0 をそれぞれ 11 およびリスト 1 へと置き換えることで ref\_idx\_10 と同一の意味を持つ。

**mvd\_10**[mbPartIdx][0][compIdx]は用いられるベクトル成分とその予測との差分を規定する。インデックス mbPartIdx はどのマクロブロックパーティションに mvd\_10 が割り当てられるのかを規定する。マクロブロッ

クのパーティションは `mb_type` によって規定される。水平動きベクトル成分差分は復号順序で最初に復号され、`compIdx=0` に割り当てられる。垂直動きベクトル成分差分は復号順序で 2 番目に復号され、`compIdx=1` に割り当てられる。`mvd_l0[mbPartIdx][0][compIdx]` の成分の範囲は、付属資料 A において規定される、それから導出される動きベクトル変数値への制約によって規定される。

`mvd_l1[mbPartIdx][0][compIdx]` は 10 および L0 をそれぞれ 11 および L1 へと置き換えることで `mvd_l0` と同一の意味を持つ。

#### 7.4.5.2 サブマクロブロック予測の意味

`sub_mb_type[mbPartIdx]` はサブマクロブロックタイプを規定する。

表および意味は P および B スライスに対する種々のサブマクロブロックタイプに対して規定される。それぞれの表は `sub_mb_type` の値、`sub_mb_type` の名前、(`NumSubMbPart(sub_mb_type)` 関数で与えられる) 用いられるサブマクロブロックパーティション数、および (`SubMbPredMode(sub_mb_type)` 関数で与えられる) サブマクロブロックの予測モードを提示する。文書中では、`sub_mb_type` の値は、“サブマクロブロックタイプ” と呼ばれることがある。文書中では、`SubMbPredMode()` の値は“サブマクロブロック予測モード” と呼ばれることがある。

P マクロブロックタイプに対する `sub_mb_type[mbPartIdx]` の解釈は表 7-17/JT-H264 において規定される。ここで、“推定” の行は、`sub_mb_type[mbPartIdx]` が存在しない場合に推定値を規定する。

表7-17/JT-H264 Pマクロブロックにおけるサブマクロブロックタイプ  
(ITU-T H.264)

<code>sub_mb_type[mbPartIdx]</code>	<code>sub_mb_type[mbPartIdx]</code> の 名前	<code>NumSubMbPart</code> ( <code>sub_mb_type[mbPartIdx]</code> )	<code>SubMbPredMode</code> ( <code>sub_mb_type[mbPartIdx]</code> )	<code>SubMbPartWidth</code> ( <code>sub_mb_type[mbPartIdx]</code> )	<code>SubMbPartHeight</code> ( <code>sub_mb_type[mbPartIdx]</code> )
推定	na	na	na	na	na
0	P_L0_8x8	1	Pred_L0	8	8
1	P_L0_8x4	2	Pred_L0	8	4
2	P_L0_4x8	2	Pred_L0	4	8
3	P_L0_4x4	4	Pred_L0	4	4

下記の意味は、表 7-17/JT-H264 におけるサブマクロブロックタイプに割り当てられる。

- P\_L0\_MxN、MxNが8x8、8x4、4x8あるいは4x4に置き換える：サブマクロブロックのサンプルはサイズ M×Nが8×8である1つの輝度パーティション、サイズM×Nが8×4である2つの輝度パーティション、あるいはサイズM×Nが4×8である2つの輝度パーティション、あるいはサイズM×Nが4×4である4つの輝度パーティション、および関連づけられた色差サンプルによってそれぞれ予測される。

下記の意味は、表 7-17/JT-H264 におけるサブマクロブロック予測モード (`SubMbPredMode()`) に割り当

てられる。

- Pred\_L0 : 表7-13/JT-H264に対する意味を見よ。

B マクロブロックタイプに対する sub\_mb\_type[mbPartIdx]の解釈は表 7-18/JT-H264 において規定される。ここで、“推定”の行は sub\_mb\_type[mbPartIdx]が存在しない場合に推定値を規定し、そして推定値“mb\_type”は、この場合 sub\_mb\_type[mbPartIdx]の名前が mb\_type の名前と同じであることを規定する。

表7-18/JT-H264 Bマクロブロックにおけるサブマクロブロックタイプ  
(ITU-T H.264)

sub_mb_type[mbPartIdx]	sub_mb_type[mbPartIdx] の 名前	NumSubMbPart (sub_mb_type[mbPartIdx])	SubMbPredMode (sub_mb_type[mbPartIdx])	SubMbPartWidth (sub_mb_type[mbPartIdx])	SubMbPartHeight (sub_mb_type[mbPartIdx])
推定	mb_type	4	Direct	4	4
0	B_Direct_8x8	4	Direct	4	4
1	B_L0_8x8	1	Pred_L0	8	8
2	B_L1_8x8	1	Pred_L1	8	8
3	B_Bi_8x8	1	BiPred	8	8
4	B_L0_8x4	2	Pred_L0	8	4
5	B_L0_4x8	2	Pred_L0	4	8
6	B_L1_8x4	2	Pred_L1	8	4
7	B_L1_4x8	2	Pred_L1	4	8
8	B_Bi_8x4	2	BiPred	8	4
9	B_Bi_4x8	2	BiPred	4	8
10	B_L0_4x4	4	Pred_L0	4	4
11	B_L1_4x4	4	Pred_L1	4	4
12	B_Bi_4x4	4	BiPred	4	4

下記の意味は表 7-18/JT-H264 におけるサブマクロブロックタイプに割り当てられる。

- B\_SkipとB\_Direct\_16x16 : サブマクロブロックに対する動きベクトル差分および参照インデックスがビットストリーム中には存在しない。関数SubMbPartWidth()およびSubMbPartHeight()は8.4.1小節で直接モード予測に対する動きベクトルおよび参照フレームインデックスの導出処理の中で用いられる。
- B\_Direct\_8x8 : サブマクロブロックに対する動きベクトル差分および参照インデックスがビットストリーム中には存在しない。関数SubMbPartWidth(B\_Direct\_8x8)およびSubMbPartHeight(B\_Direct\_8x8)は8.4.1

小節で直接モード予測に対する動きベクトルおよび参照フレームインデックスの導出処理の中で用いられる。

- $B\_X\_M \times N$ 、ただし $X$ を $L0$ 、 $L1$ あるいは $Bi$ へ置き換え、 $M \times N$ を $8 \times 8$ 、 $8 \times 4$ 、 $4 \times 8$ あるいは $4 \times 4$ へ置き換える：サブマクロブロックのサンプルはサイズ $M \times N$ が $8 \times 8$ である1つの輝度パーティションにより、あるいはサイズ $M \times N$ が $8 \times 4$ である2つの輝度パーティションにより、あるいはサイズ $M \times N$ が $4 \times 8$ である2つの輝度パーティションにより、あるいはサイズ $M \times N$ が $4 \times 4$ である4つの輝度パーティションにより、および関連付けられた色差サンプルによってそれぞれ予測される。全てのサブマクロブロックパーティションは同一の参照インデックスを共有する。 $X$ を $L0$ あるいは $L1$ へ置き換えた $B\_X\_M \times N$ を $sub\_mb\_type$ にもつサブマクロブロック中の1つの $M \times N$ サブマクロブロックパーティションに対して、1つの動きベクトル差分がビットストリーム中に存在する。 $B\_Bi\_M \times N$ を $sub\_mb\_type$ にもつサブマクロブロック中の1つの $M \times N$ サブマクロブロックパーティションに対して、2つの動きベクトル差分がビットストリーム中に存在する。

下記の意味は、表 7-18/JT-H264 におけるサブマクロブロック予測モード ( $SubMbPredMode()$ ) に割り当てられる。

- Direct : 表7-14/JT-H264に対する意味を見よ。
- Pred\_L0 : 表7-13/JT-H264に対する意味を見よ。
- Pred\_L1 : 表7-14/JT-H264に対する意味を見よ。
- BiPred : 表7-14/JT-H264に対する意味を見よ。

$ref\_idx\_10[mbPartIdx]$ は 7.4.5.1 小小節における  $ref\_idx\_10$  と同一の意味を持つ。

$ref\_idx\_11[mbPartIdx]$ は 7.4.5.1 小小節における  $ref\_idx\_11$  と同一の意味を持つ。

$mvd\_10[mbPartIdx][subMbPartIdx][compIdx]$ は  $subMbPartIdx$  をもつサブマクロブロックパーティションインデックスに適用されることを除いて、7.4.5.1 小小節における  $mvd\_10$  と同一の意味を持つ。インデックス  $mbPartIdx$  および  $subMbPartIdx$  はどのマクロブロックパーティションおよびサブマクロブロックパーティションに  $mvd\_10$  が割り当てられるのかを規定する。

$mvd\_11[mbPartIdx][subMbPartIdx][compIdx]$ は 7.4.5.1 小小節における  $mvd\_11$  と同一の意味を持つ。

### 7.4.5.3 残差データの意味

変換係数レベル構文解析のために用いられるシンタックス構造  $residual\_block()$ を次の通り割り当てる。

- もし  $entropy\_coding\_mode\_flag$  が 0 に等しいならば、 $residual\_block$  は変換係数レベルに対するシンタックス要素を構文解析するために用いられる  $residual\_block\_cavlc$  に等しく設定される。
- それ以外 ( $entropy\_coding\_mode\_flag$  が 1 に等しい)、 $residual\_block$  は変換係数レベルに対するシンタックス要素を構文解析するために用いられる  $residual\_block\_cabac$  に等しく設定される。

シンタックス構造  $residual\_luma(i16x16DClevel, i16x16AClevel, level, level8x8)$ が、その出力であり次のように割り当てられる括弧内の変数とともに用いられる。

Intra16x16DCLevel は i16x16DClevel に等しく設定され、Intra16x16ACLevel は i16x16AClevel に等しく設定され、LumaLevel は level に等しく設定され、そして LumaLevel8x8 は level8x8 に等しく設定される。

ChromaArrayType が 1 あるいは 2 に等しい時、以下が適用される。

- iCbCr=0.1 により索引される各色差成分について、 $4 * \text{NumC8x8}$  個の 4x4 色差ブロックの DC 変換係数レベルが iCbCr 番目のリスト ChromaDCLevel[iCbCr] に構文解析される。
- iCbCr=0.1 により索引される各色差成分の、 $i4x4=0.3$  および  $i8x8=0.. \text{NumC8x8}-1$  により索引される各 4x4 色差ブロックについて、15 個の AC 変換係数レベルが iCbCr 番目の色差成分の  $(i8x8*4+i4x4)$  番目のリスト ChromaACLevel[iCbCr][ $i8x8*4+i4x4$ ] に構文解析される。

ChromaArrayType が 3 に等しい時、以下が適用される。

- シンタックス構造 residual\_luma(i16x16DClevel, i16x16AClevel, level, level8x8) が、その出力であり次のように割り当てられる括弧内の変数とともに Cb 成分に対して用いられる。CbIntra16x16DCLevel は i16x16DClevel に等しく設定され、CbIntra16x16ACLevel は i16x16AClevel に等しく設定され、CbLevel は level に等しく設定され、そして CbLevel8x8 は level8x8 に等しく設定される。
- シンタックス構造 residual\_luma(i16x16DClevel, i16x16AClevel, level, level8x8) が、その出力であり次のように割り当てられる括弧内の変数とともに Cr 成分に対して用いられる。CrIntra16x16DCLevel は i16x16DClevel に等しく設定され、CrIntra16x16ACLevel は i16x16AClevel に等しく設定され、CrLevel は level に等しく設定され、そして CrLevel8x8 は level8x8 に等しく設定される。

#### 7.4.5.3.1 残差輝度データの意味

このシンタックス構造の出力は、変数 i16x16DClevel、i16x16AClevel、level および level8x8 である。

変換係数レベルの構文解析に用いられる、シンタックス構造 residual\_block() は次のように割り当てられる。

- もし entropy\_coding\_mode\_flag が 0 に等しいならば、residual\_block は residual\_block\_cavlc に等しく設定され、変換係数レベルのシンタックス要素の構文解析に用いられる。
- それ以外 (entropy\_coding\_mode\_flag が 1 に等しい)、residual\_block は residual\_block\_cabac に等しく設定され、変換係数レベルのシンタックス要素の構文解析に用いられる。

mb\_type に依存して、シンタックス構造 residual\_block(coeffLevel,maxNumCoeff) は residual\_block() の中で構文解析される maxNumCoeff 個の変換係数レベルを含むリストである引数 coeffLevel、および、引数 maxNumCoeff と共に次の通り用いられる。

- MbPartPredMode(mb\_type,0) に依存して、次が適用される。
  - もし MbPartPredMode(mb\_type,0) が Intra\_16x16 に等しいならば、変換係数レベルはリスト i16x16DCLevel へ構文解析され、さらに 16 個のリスト i16x16ACLevel[i] へ構文解析される。i16x16DCLevel は各々の 4x4 輝度ブロックに対する DC 変換係数レベルの 16 個の変換係数レベルを含む。i=0..15 によって索引される 16 個の 4x4 輝度ブロックの各々に対して、i 番目のブロックの 15 個の AC 変換係数レベルが i 番目のリスト i16x16ACLevel[i] へ構文解析される。
  - それ以外 (MbPartPredMode(mb\_type,0) が Intra\_16x16 に等しくない)、次が適用される。
    - もし transform\_size\_8x8\_flag が 0 に等しいならば、i=0..15 によって索引される 16 個の 4x4

輝度ブロックの各々に対して、 $i$  番目のブロックの 16 個の変換係数レベルが  $i$  番目のリスト  $level[i]$  へ構文解析される。

- それ以外 ( $transform\_size\_8x8\_flag$  が 1 に等しい)、 $i8x8=0..3$  によって索引される 4 個の  $8 \times 8$  輝度ブロックの各々に対して、次が適用される。
  - もし、 $entropy\_coding\_mode\_flag$  が 0 に等しいならば、 $i4x4=0..3$  によって索引される 4 個の  $4 \times 4$  輝度ブロックの各々に対して、 $i4x4$  番目のブロックの 16 個の変換係数レベルが  $(i8x8*4+i4x4)$  番目のリスト  $level[i8x8*4+i4x4]$  へ構文解析される。それから、 $4*i+i4x4$  によって索引される  $i8x8$  番目の  $8 \times 8$  輝度ブロックの 64 個の変換係数レベルは、 $level8x8[i8x8][4*i+i4x4]=level[i8x8*4+i4x4][i]$  から導出される。ここで  $i=0..15$  であり  $i4x4=0..3$  である。

記 オフセット  $i4x4$  を用いて対応する  $i8x8$  番目の  $8 \times 8$  輝度ブロックの 4 つ目ごとの変換係数レベルを含む  $luma4x4BlkIdx=i8x8*4+i4x4$  を持つ  $4 \times 4$  輝度ブロックは、6.4.3 小節の逆  $4 \times 4$  輝度ブロック走査処理によって得られる空間位置を表現すると仮定される。
  - それ以外 ( $entropy\_coding\_mode\_flag$  が 1 に等しい)、 $i8x8$  番目のブロックの 64 個の変換係数レベルが  $i8x8$  番目のリスト  $level8x8[i8x8]$  へ構文解析される。

#### 7.4.5.3.2 残差ブロック CAVLC の意味

7.3.5.3.2 小小小節で使用される関数  $TotalCoeff(coeff\_token)$  は  $coeff\_token$  から導出される非ゼロ変換係数レベルの数を返す。

7.3.5.3.2 小小小節で使用される関数  $TrailingOnes(coeff\_token)$  は  $coeff\_token$  から導出される追従 1 の数を返す。

**coeff\_token** は変換係数レベルの走査の中での非ゼロ変換係数レベルの総数および追従 1 変換係数レベルの数を規定する。追従 1 変換係数レベルは、非ゼロ変換係数レベルの走査の終端に位置する 1 に等しい絶対値を持つ最大 3 個の連続する非ゼロ変換係数レベルのうちの 1 つである。**coeff\_token** の範囲は 9.2.1 小節で規定される。

**trailing\_ones\_sign\_flag** は、追従 1 変換係数レベルの符号を次の通り規定する。

- もし **trailing\_ones\_sign\_flag** が 0 に等しいならば、対応する変換係数レベルは +1 として復号される。
- それ以外 (**trailing\_ones\_sign\_flag** が 1 に等しい)、対応する変換係数レベルは -1 として復号される。

**level\_prefix** および **level\_suffix** は非ゼロ変換係数レベルの値を規定する。**level\_prefix** および **level\_suffix** の範囲は 9.2.2 小節で規定される。

**total\_zeros** は変換係数レベルの走査内で最後の非ゼロ変換係数レベルの前に位置するゼロ値変換係数レベルの総数を規定する。**total\_zeros** の範囲は 9.2.3 小節で規定される。

**run\_before** は走査の中で非ゼロ変換係数レベルの前に位置する値 0 をもつ連続する変換係数レベルの個数を規定する。**run\_before** の範囲は 9.2.3 小節で規定される。

**coeffLevel** は変換係数レベルの現リストに対する  $maxNumCoeff$  個の変換係数レベルを含む。

### 7.4.5.3.3 残差ブロックCABACの意味

**coded\_block\_flag** はブロックが非ゼロ変換係数レベルを含むかどうかを次の通り規定する。

- もし **coded\_block\_flag** が 0 に等しいならば、ブロックは 1 つも非ゼロ変換係数レベルを含まない。
- それ以外 (**coded\_block\_flag** が 1 に等しい)、ブロックは少なくとも 1 つの非ゼロ変換係数レベルを含む。

**coded\_block\_flag** が存在しない時、それは 1 に等しいと推定される。

**significant\_coeff\_flag[i]** は走査位置 *i* の変換係数レベルが非ゼロであるかどうかを次の通り規定する。

- もし **significant\_coeff\_flag[i]** が 0 に等しいならば、走査位置 *i* の変換係数レベルは 0 に設定される。
- それ以外 (**significant\_coeff\_flag[i]** が 1 に等しい)、走査位置 *i* の変換係数レベルは非ゼロの値を有する。

**last\_significant\_coeff\_flag[i]** は走査位置 *i* に対して、後続する *i*+1 から **maxNumCoeff**-1 に至るまでの走査位置において非ゼロ変換係数レベルが存在するかどうかを次の通り規定する。

- もし **last\_significant\_coeff\_flag[i]** が 1 に等しいならば、ブロックの(走査順序で)後続する全ての変換係数レベルは 0 に等しい値を有する。
- それ以外 (**last\_significant\_coeff\_flag[i]** が 0 に等しい)、走査経路に沿ってさらに非ゼロ変換係数レベルがある。

**coeff\_abs\_level\_minus1[i]** は変換係数レベルの絶対値から 1 を減じたものである。**coeff\_abs\_level\_minus1** の値は 8.5 節で記述される限度によって制限される。

**coeff\_sign\_flag[i]** は変換係数レベルの符号を次の通り規定する。

- もし **coeff\_sign\_flag** が 0 に等しいならば、対応する変換係数レベルは正の値を有する。
- それ以外 (**coeff\_sign\_flag** が 1 に等しい)、対応する変換係数レベルは負の値を有する。

**coeffLevel** は変換係数レベルの現リストに対する **maxNumCoeff** 個の変換係数レベルを含む。

## 8 復号処理

この処理の出力は (変数 **CurrPic** で参照される時もある) 現ピクチャの復号されたサンプルである。

**chroma\_format\_idc** の値に依存して、現ピクチャのサンプル配列の数は次の通りである。

- **chroma\_format\_idc** が 0 に等しい時、現ピクチャは 1 つのサンプル配列  $S_L$  からなる。
- それ以外 (**chroma\_format\_idc** が 0 に等しくない)、現ピクチャは 3 つのサンプル配列  $S_L$ ,  $S_{Cb}$ ,  $S_{Cr}$  からなる。

この章は 7 章から与えられたシンタックス要素および大文字変数で復号処理を記述する。

復号処理は全ての復号器が数値的に同一の結果を生み出さなければならないように規定される。ここで記述される処理と同一の結果を生み出すいかなる復号処理も本標準の復号処理の要件に適合する。

この章の中で言及される各々のピクチャは主符号化ピクチャの全体または一部である。この章の中で言及される各々のスライスも主符号化ピクチャのスライスである。この章の中で言及される各々のスライスデータパーティションは主符号化ピクチャのスライスデータパーティションである。

separate\_colour\_plane\_flag の値に依存して、復号処理は次のように構造化される。

- もし separate\_colour\_plane\_flag が 0 に等しいならば、復号処理は、現ピクチャを出力として 1 度だけ起動される。
- それ以外 (separate\_colour\_plane\_flag が 1 に等しい)、復号処理は 3 度起動される。復号処理への入力は、colour\_plane\_id の同一値を持つ主符号化ピクチャの全 NAL ユニットである。ある特定の colour\_plane\_id の値を持つ NAL ユニットの復号処理は、あたかもその特定の colour\_plane\_id の値を持つモノクローム色フォーマットの符号化ビデオシーケンスだけがビットストリームに存在するかのように規定される。3 度の復号処理のそれぞれの出力は、colour\_plane\_id が 0 の NAL ユニットが  $S_L$  に、colour\_plane\_id が 1 の NAL ユニットが  $S_{Cb}$  に、colour\_plane\_id が 2 の NAL ユニットが  $S_{Cr}$  に割り当てられた現ピクチャの 3 つのサンプル配列に割り当てられる。

記 変数 ChromaArrayType は、separate\_colour\_plane\_flag が 1 に等しく、chroma\_format\_idc が 3 に等しい時、0 として導出される。復号処理において、この変数の値は chroma\_format\_idc が 0 に等しいモノクロームピクチャの操作と同一の操作を結果とするように評価される。

復号処理の概要は次の通り与えられる。

- NAL ユニットの復号は 8.1 節で規定される。
- 8.2 節に記載される処理は、スライスレイヤおよびその上位レイヤにあるシンタックス要素を用いた復号処理を規定する。
  - ピクチャ順序カウントに関する変数および関数は 8.2.1 小節において導出される。(ピクチャの 1 つのスライスに対してのみ起動される必要がある。)
  - マクロブロックからスライスグループへのマップに関する変数および関数は 8.2.2 小節において導出される。(ピクチャの 1 つのスライスに対してのみ起動される必要がある。)
  - スライスデータパーティションが使用される時、様々なパーティションを統合する手法が 8.2.3 小節に記述される。
  - 現ピクチャの frame\_num が PrevRefFrameNum に等しくなく、かつ  $(\text{PrevRefFrameNum}+1)\% \text{MaxFrameNum}$  に等しくない時、現ピクチャの任意のスライスの復号の前に、frame\_num のギャップの復号処理が、8.2.5.2 小節に従って実行される。
  - 各々の P、SP あるいは B スライスにおける復号処理の開始で、8.2.4 小節で規定される参照ピクチャリスト構築の復号処理が、参照ピクチャリスト 0 (RefPicList0)、そして B スライスを復号する時、参照ピクチャリスト 1 (RefPicList1) の導出に対して実行される。
  - 現ピクチャが参照ピクチャである場合、そして、現ピクチャの全てのスライスが復号された後に、8.2.5 小節にある復号参照ピクチャマーキング処理は、後の復号ピクチャ中の INTER 予測の復号処理で、現ピクチャがどのように用いられるのかを規定する。
- 8.3、8.4、8.5、8.6 および 8.7 節に記載される処理はマクロブロックレイヤおよびその上位にあるシンタックス要素を用いた復号処理を規定する。
  - 8.3 節において規定される I\_PCM マクロブロックを除いた I および SI マクロブロックに対する INTRA 予測処理は、その出力として INTRA 予測サンプルをもつ。I\_PCM マクロブロックに対して 8.3 節は直接的にピクチャ構築処理を規定する。出力はデブロックフィルタ処理前の構築されたサ

ンプルである。

- P および B マクロブロックに対する INTER 予測処理は、その出力である INTER 予測サンプルとともに 8.4 節において規定される。
- 変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理は 8.5 節において規定される。その処理は I および B マクロブロックに対して、および、P スライス中の P マクロブロックに対してサンプルを導出する。出力はデブロックフィルタ処理前の構築されたサンプルである。
- SP スライス内の P マクロブロックあるいは SI マクロブロックに対する復号処理は 8.6 節において規定される。その処理は SP スライス中の P マクロブロックに対して、および、SI マクロブロックに対してサンプルを導出する。出力はデブロックフィルタ処理前の構築されたサンプルである。
- ブロックとマクロブロックのエッジに隣接したデブロックフィルタ処理前の構築されたサンプルは、8.7 節で規定されるように、その出力が復号サンプルになるデブロックフィルタにより処理される。

## 8.1 NAL ユニット復号処理

この処理の入力は NAL ユニットである。

この処理の出力は NAL ユニットの内部にカプセル化された RBSP シンタックス構造である。

各々の NAL ユニットに対する復号処理は NAL ユニットから RBSP シンタックス構造を取り出し、そして下記のように NAL ユニット内の RBSP シンタックス構造に対して規定された復号処理を操作する。

8.2 節は `nal_unit_type` が 1 から 5 に等しい NAL ユニットに対する復号処理を記述する。

8.3 節は `nal_unit_type` が 1、2 および 5 に等しい NAL ユニット内で符号化されるマクロブロックあるいはマクロブロックの一部に対する復号処理を記述する。

8.4 節は `nal_unit_type` が 1 および 2 に等しい NAL ユニット内で符号化されるマクロブロックあるいはマクロブロックの一部に対する復号処理を記述する。

8.5 節は `nal_unit_type` が 1 および 3 から 5 に等しい NAL ユニット内で符号化されるマクロブロックあるいはマクロブロックの一部に対する復号処理を記述する。

8.6 節は `nal_unit_type` が 1 および 3 から 5 に等しい NAL ユニット内で符号化されるマクロブロックあるいはマクロブロックの一部に対する復号処理を記述する。

8.7 節は `nal_unit_type` が 1 から 5 に等しい NAL ユニット内で符号化されるマクロブロックあるいはマクロブロックの一部に対する復号処理を記述する。

`nal_unit_type` が 7 および 8 に等しい NAL ユニットはシーケンスパラメータセットおよびピクチャパラメータセットをそれぞれ含む。ピクチャパラメータセットは、各々のピクチャのスライスヘッダ内部でのピクチャパラメータセットへの参照で決定されるようにして、他の NAL ユニットの復号処理中で使用される。シーケンスパラメータセットは、各々のシーケンスのピクチャパラメータセット内部でのシーケンスパラメータセットへの参照で決定されるようにして、他の NAL ユニットの復号処理中で使用される。

`nal_unit_type` が 6、9、10、11 および 12 に等しい NAL ユニットに対しては必須復号処理は規定されない。

## 8.2 スライス復号処理

### 8.2.1 ピクチャ順序カウントに対する復号処理

この処理の出力は `TopFieldOrderCnt` (適用可能ならば) および `BottomFieldOrderCnt` (適用可能ならば) である。

ピクチャ順序カウントは、B スライスの復号で参照ピクチャに対する初期ピクチャ順序付けを決定するため (8.2.4.2.3 小小小節および 8.2.4.2.4 小小小節を見よ)、時間方向直接モードにおける動きベクトル導出のためのフレームあるいはフィールド間のピクチャ順序差分を表現するため (8.4.1.2.3 小小小節を見よ)、B スライスにおける暗黙的モードの重み付け予測のため (8.4.2.3.2 小小小節を見よ)、および、復号器適合検査のため (C.4 節を見よ) に用いられる。

ピクチャ順序カウント情報は、毎フレーム、毎フィールド (符号化フィールドから復号された、あるいは、復号されたフレームの一部としてのいずれか)、あるいは毎相補的フィールドペアに対して、次の通り導出される。

- 各々の符号化フレームは、そのトップフィールドおよびボトムフィールドに対するそれぞれ `TopFieldOrderCnt` および `BottomFieldOrderCnt` と呼ばれる 2 つのピクチャ順序カウントに関連付けられる。
- 各々の符号化フィールドは、符号化されたトップフィールドに対する `TopFieldOrderCnt`、および、ボトムフィールドに対する `BottomFieldOrderCnt` と呼ばれるピクチャ順序カウントに関連付けられる。
- 各々の相補的フィールドペアは、その符号化されたトップフィールドに対する `TopFieldOrderCnt`、および、その符号化されたボトムフィールドに対する `BottomFieldOrderCnt` である 2 つのピクチャ順序カウントにそれぞれ関連付けられる。

`TopFieldOrderCnt` および `BottomFieldOrderCnt` は、復号順序において、直前の IDR ピクチャあるいは 5 に等しい `memory_management_control_operation` を含む直前の参照ピクチャの最初の出力フィールド相対で対応するトップフィールドあるいはボトムフィールドのピクチャ順序を指示する。

`TopFieldOrderCnt` および `BottomFieldOrderCnt` は、それぞれ 8.2.1.1、8.2.1.2 および 8.2.1.3 小小小節内のピクチャ順序カウントタイプ 0、1 および 2 に対する復号処理の 1 つを起動することで導出される。現ピクチャが 5 に等しいメモリ管理制御操作を含む場合は、現ピクチャの復号の後に、`tempPicOrderCnt` は `PicOrderCnt(CurrPic)` に等しく設定され、現ピクチャの `TopFieldOrderCnt` (もしあれば) は `TopFieldOrderCnt-tempPicOrderCnt` に等しく設定され、現ピクチャの `BottomFieldOrderCnt` (もしあれば) は `BottomFieldOrderCnt-tempPicOrderCnt` に等しく設定される。

ビットストリームは、符号化された IDR フレームに対して  $\text{Min}(\text{TopFieldOrderCnt}, \text{BottomFieldOrderCnt})$  の結果が 0 に等しくないデータ、符号化された IDR トップフィールドに対して `TopFieldOrderCnt` が 0 に等しくないデータ、あるいは符号化された IDR ボトムフィールドに対して `BottomFieldOrderCnt` が 0 に等しくないデータを含んではならない。すなわち、符号化された IDR フレームのフィールドに対して少なくとも `TopFieldOrderCnt` および `BottomFieldOrderCnt` のうちの 1 つは 0 でなければならない。

現ピクチャが IDR ピクチャでない場合、次が適用される。

- 下記の全てを含むピクチャのリストに関連付けられる `TopFieldOrderCnt` および `BottomFieldOrderCnt` の値を要素として含むリスト変数 `listD` があるものとする。
  - リスト中の最初のピクチャは、直前の、下記タイプのいずれかのピクチャである。

- IDR ピクチャ
- 5 に等しい `memory_management_control_operation` を含むピクチャ
- 後に続く追加ピクチャ
  - もし `pic_order_cnt_type` が 0 に等しいならば、リスト中での最初のピクチャに復号順序で後続し、かつ、8.2.5.2 小小節で規定される `frame_num` のギャップの復号処理で推定される“非存在”フレームではなく、かつ、復号順序で現ピクチャに先行するか現ピクチャであるその他全てのピクチャ。`pic_order_cnt_type` が 0 に等しく、現ピクチャが 8.2.5.2 小小節で規定される `frame_num` のギャップの復号処理で推定される“非存在”フレームではないならば、現ピクチャは復号参照ピクチャマーキング処理の起動に先行して `listD` に含まれる。
  - それ以外 (`pic_order_cnt_type` が 0 に等しくない)、リスト中での最初のピクチャに復号順序で後続し、かつ、復号順序で現ピクチャに先行するか現ピクチャであるその他全てのピクチャ。`pic_order_cnt_type` が 0 に等しくないならば、現ピクチャは復号参照ピクチャマーキング処理の起動に先行して `listD` に含まれる。
- 昇順でソートされた `listD` の要素を含むリスト変数 `listO` があるものとする。`listO` は下記のいかなるものも含んではならない。
  - `listO` の中で連続的な位置にない、フレームあるいは相補的フィールドペアに対する `TopFieldOrderCnt` および `BottomFieldOrderCnt` の対。
  - もう 1 つの `TopFieldOrderCnt` に等しい値を有する `TopFieldOrderCnt`。
  - もう 1 つの `BottomFieldOrderCnt` に等しい値を有する `BottomFieldOrderCnt`。
  - `BottomFieldOrderCnt` および `TopFieldOrderCnt` が同一の符号化フレームあるいは相補的フィールドペアに属している場合を除いて、`TopFieldOrderCnt` に等しい値を有する `BottomFieldOrderCnt`。

ビットストリームは、8.2.1.1 小小節から 8.2.1.3 小小節において規定されるような復号処理の中で用いられる `TopFieldOrderCnt`、`BottomFiledOrderCnt`、`PicOrderCntMsb` または `FrameNumOffset` の値が  $-2^{31}$  から  $2^{31}-1$  のそれぞれの値を含む範囲の値を超える結果となるようなデータを含んではならない。

関数 `PicOrderCnt(picX)` は下記のように規定される。

```

if(picXはフレームまたは相補的フィールドペア)
  PicOrderCnt(picX)=(フレームまたは相補的フィールドペアpicXの
                    Min(TopFieldOrderCnt,BottomFieldOrderCnt))
else if(picXはトップフィールド)
  PicOrderCnt(picX)=(フィールドpicXのTopFieldOrderCnt)
else if(picXはボトムフィールド)
  PicOrderCnt(picX)=(フィールドpicXのBottomFieldOrderCnt)

```

(8-1)

そして `DiffPicOrderCnt(picA,picB)` が以下のように規定される。

```

DiffPicOrderCnt(picA,picB)=PicOrderCnt(picA)-PicOrderCnt(picB)

```

(8-2)

ビットストリームは復号処理で用いられる `DiffPicOrderCnt(picA,picB)` の値が  $-2^{15}$  から  $2^{15}-1$  までのそれぞれ

の値を含む範囲を超える結果となるようなデータを含んではならない。

記1 Xを現ピクチャに、そしてYおよびZを同一シーケンス内の2つの別のピクチャとし、DiffPicOrderCnt(X,Y)およびDiffPicOrderCnt(X,Z)の両方が正值である場合、あるいは両方が負値である場合に、YおよびZがXからの同一出力順序方向にあるものと考えられる。

記2 多くのアプリケーションはPicOrderCnt(X)を、IDRピクチャのサンプリング時刻相対の、ピクチャXのサンプリング時刻に比例するように割り当てる。

現ピクチャが5に等しい memory\_management\_control\_operation を含む時、PicOrderCnt(CurrPic)は PicOrderCnt(listD のそれ以外の任意のピクチャ)より大きくなくてはならない。

### 8.2.1.1 ピクチャ順序カウントタイプ0に対する復号処理

この処理は pic\_order\_cnt\_type が0に等しい場合に起動される。

この処理の入力はこの小節において規定される復号順序における直前の参照ピクチャの PicOrderCntMsb である。

この処理の出力は TopFieldOrderCnt あるいは BottomFieldOrderCnt のいずれか、あるいはその両方である。

変数 prevPicOrderCntMsb と prevPicOrderCntLsb は次の通り導出される。

- もし現ピクチャが IDR ピクチャであるならば、prevPicOrderCntMsb は0に設定され、prevPicOrderCntLsb は0に設定される。
- それ以外（現ピクチャが IDR ピクチャではない）、次が適用される。
  - もし復号順序において直前の参照ピクチャが5に等しい memory\_management\_control\_operation を含んでいるならば、次が適用される。
    - もし復号順序において直前の参照ピクチャがボトムフィールドでないならば、prevPicOrderCntMsb は0に設定され、prevPicOrderCntLsb は復号順序において直前の参照ピクチャに対する TopFieldOrderCnt の値に設定される。
    - それ以外（復号順序において直前の参照ピクチャがボトムフィールドである）、prevPicOrderCntMsb は0に設定され、prevPicOrderCntLsb は0に設定される。
  - それ以外（復号順序において直前の参照ピクチャが5に等しい memory\_management\_control\_operation を含んでいなかった）、prevPicOrderCntMsb は復号順序において直前の参照ピクチャの PicOrderCntMsb に等しく設定され、prevPicOrderCntLsb は復号順序において直前の参照ピクチャの pic\_order\_cnt\_lsb に等しく設定される。

現ピクチャの PicOrderCntMsb は次の通り導出される。

```
if((pic_order_cnt_lsb<prevPicOrderCntLsb) &&
  ((prevPicOrderCntLsb-pic_order_cnt_lsb)>=(MaxPicOrderCntLsb/2)))
  PicOrderCntMsb=prevPicOrderCntMsb+MaxPicOrderCntLsb           (8-3)
else if((pic_order_cnt_lsb>prevPicOrderCntLsb) &&
  ((pic_order_cnt_lsb-prevPicOrderCntLsb)>(MaxPicOrderCntLsb/2)))
  PicOrderCntMsb=prevPicOrderCntMsb-MaxPicOrderCntLsb
else
  PicOrderCntMsb=prevPicOrderCntMsb
```

現ピクチャがボトムフィールドでない場合、TopFieldOrderCnt は次の通り導出される。

```
if(!field_pic_flag || !bottom_field_flag)
    TopFieldOrderCnt=PicOrderCntMsb+pic_order_cnt_lsb
```

 (8-4)

現ピクチャがトップフィールドでない場合、BottomFieldOrderCnt は次の通り導出される。

```
if(!field_pic_flag)
    BottomFieldOrderCnt=TopFieldOrderCnt+delta_pic_order_cnt_bottom
else if(bottom_field_flag)
    BottomFieldOrderCnt=PicOrderCntMsb+pic_order_cnt_lsb
```

 (8-5)

### 8.2.1.2 ピクチャ順序カウントタイプ 1 に対する復号処理

この処理は pic\_order\_cnt\_type が 1 に等しい場合に起動される。

この処理の入力はこの小小節において規定される復号順序における直前のピクチャの FrameNumOffset である。

この処理の出力は TopFieldOrderCnt あるいは BottomFieldOrderCnt のいずれか、あるいはその両方である。

TopFieldOrderCnt および BottomFieldOrderCnt の値はこの小小節において規定されるように導出される。prevFrameNum は、復号順序において直前のピクチャの frame\_num に等しくする。

現ピクチャが IDR ピクチャでない場合、変数 prevFrameNumOffset は次の通り導出される。

- もし復号順序における直前のピクチャが 5 に等しい memory\_management\_control\_operation を含んでいたならば、prevFrameNumOffset は 0 に設定される。
- それ以外（復号順序における直前のピクチャが 5 に等しい memory\_management\_control\_operation を含んでいなかった）、prevFrameNumOffset は復号順序における直前のピクチャの FrameNumOffset の値に等しく設定される。

記 gaps\_in\_frame\_num\_value\_allowed\_flag が 1 に等しい時、復号順序において直前のピクチャは、8.2.5.2 小小節で規定される frame\_num のギャップに対する復号処理により推定される“非存在”フレームとなるかもしれない。

導出が下記の順序付けられたステップに従って進む。

1. 変数 FrameNumOffset が下記のように導出される。

```
if(IdxPicFlag==1)
    FrameNumOffset=0
else if(prevFrameNum>frame_num)
    FrameNumOffset=prevFrameNumOffset+MaxFrameNum
else
    FrameNumOffset=prevFrameNumOffset
```

 (8-6)

2. 変数 absFrameNum が下記のように導出される。

```
if(num_ref_frames_in_pic_order_cnt_cycle!=0)
    absFrameNum=FrameNumOffset+frame_num
```

```

else
    absFrameNum=0
if(nal_ref_idc==0 && absFrameNum>0)
    absFrameNum=absFrameNum-1

```

(8-7)

3. absFrameNum>0 である場合、picOrderCntCycleCnt および frameNumInPicOrderCntCycle が下記のように導出される。

```

if(absFrameNum>0){
    picOrderCntCycleCnt=(absFrameNum-1)/num_ref_frames_in_pic_order_cnt_cycle
    frameNumInPicOrderCntCycle=(absFrameNum-1)%num_ref_frames_in_pic_order_cnt_cycle
}

```

(8-8)

4. 変数 expectedDeltaPerPicOrderCntCycle が下記のように導出される。

```

expectedDeltaPerPicOrderCntCycle=0
for(i=0;i<num_ref_frames_in_pic_order_cnt_cycle;i++)
    expectedDeltaPerPicOrderCntCycle+=offset_for_ref_frame[i]

```

(8-9)

5. 変数 expectedPicOrderCnt が下記のように導出される。

```

if(absFrameNum>0){
    expectedPicOrderCnt=picOrderCntCycleCnt*expectedDeltaPerPicOrderCntCycle
    for(i=0;i<=frameNumInPicOrderCntCycle;i++)
        expectedPicOrderCnt=expectedPicOrderCnt+offset_for_ref_frame[i]
}
else
    expectedPicOrderCnt=0
if(nal_ref_idc==0)
    expectedPicOrderCnt=expectedPicOrderCnt+offset_for_non_ref_pic

```

(8-10)

6. 変数 TopFieldOrderCnt あるいは BottomFieldOrderCnt が下記のように導出される。

```

if(!field_pic_flag){
    TopFieldOrderCnt=expectedPicOrderCnt+delta_pic_order_cnt[0]
    BottomFieldOrderCnt=TopFieldOrderCnt+
        offset_for_top_to_bottom_field+delta_pic_order_cnt[1]
}
else if(!bottom_field_flag)
    TopFieldOrderCnt=expectedPicOrderCnt+delta_pic_order_cnt[0]
else
    BottomFieldOrderCnt=expectedPicOrderCnt+offset_for_top_to_bottom_field+delta_pic_order_cnt[0]

```

(8-11)

### 8.2.1.3 ピクチャ順序カウントタイプ 2 に対する復号処理

この処理は pic\_order\_cnt\_type が 2 に等しい場合に起動される。

この処理の出力は TopFieldOrderCnt あるいは BottomFieldOrderCnt のいずれか、あるいはその両方である。

prevFrameNum は、復号順序において直前のピクチャの frame\_num に等しくする。

現ピクチャが IDR ピクチャでない場合、変数 prevFrameNumOffset は次の通り導出される。

- もし復号順序における直前のピクチャが 5 に等しい memory\_management\_control\_operation を含んでいたならば、prevFrameNumOffset は 0 に設定される。
- それ以外（復号順序における直前のピクチャが 5 に等しい memory\_management\_control\_operation を含んでいなかった）、prevFrameNumOffset は復号順序における直前のピクチャの FrameNumOffset の値に等しく設定される。

記1 gaps\_in\_frame\_num\_value\_allowed\_flagが1に等しい時、復号順序において直前のピクチャは、8.2.5.2 小節で規定されるframe\_numのギャップに対する復号処理により推定される“非存在”フレームとなるかもしれない。

変数 FrameNumOffset は下記のように導出される。

```
if(IdxPicFlag==1)
    FrameNumOffset=0
else if(prevFrameNum>frame_num)
    FrameNumOffset=prevFrameNumOffset+MaxFrameNum
else
    FrameNumOffset=prevFrameNumOffset
```

(8-12)

変数 tempPicOrderCnt は下記のように導出される。

```
if(IdxPicFlag==1)
    tempPicOrderCnt=0
else if(nal_ref_idc==0)
    tempPicOrderCnt=2*(FrameNumOffset+frame_num)-1
else
    tempPicOrderCnt=2*(FrameNumOffset+frame_num)
```

(8-13)

変数 TopFieldOrderCnt あるいは BottomFieldOrderCnt は下記のように導出される。

```
if(!field_pic_flag){
    TopFieldOrderCnt=tempPicOrderCnt
    BottomFieldOrderCnt=tempPicOrderCnt
}else if(bottom_field_flag)
    BottomFieldOrderCnt=tempPicOrderCnt
else
    TopFieldOrderCnt=tempPicOrderCnt
```

(8-14)

記2 ピクチャ順序カウントタイプ2は、それらのピクチャの1つより多くがTopFieldOrderCntの同じ値をもつ、またはそれらのピクチャの1つより多くがBottomFieldOrderCntの同じ値をもつ結果になる連続する非参照ピクチャを含む符号化ビデオシーケンス内では使用され得ない。

記3 ピクチャ順序カウントタイプ2は復号順序と同一の出力順序の結果となる。

## 8.2.2 マクロブロックからスライスグループへのマップの復号処理

この処理の入力は、アクティブなピクチャパラメータセットと、復号されるべきスライスのスライスヘッダである。

この処理の出力は、マクロブロックからスライスグループへのマップ MbToSliceGroupMap である。

この処理は、全てのスライスの開始で起動される。

記 この処理の出力は、ピクチャの全てのスライスに対して等しい。

num\_slice\_groups\_minus1 が 1 に等しく、そして slice\_group\_map\_type が 3、4 あるいは 5 に等しい時、スライスグループ 0 と 1 は、表 8-1/JT-H264 で示され 8.2.2.4 から 8.2.2.6 小節までで規定される slice\_group\_change\_direction\_flag で決定されるサイズと形を持つ。

表8-1/JT-H264 詳細なスライスグループマップタイプ  
(ITU-T H.264)

slice_group_map_type	slice_group_change_direction_flag	詳細なスライスグループマップタイプ
3	0	時計回りボックスアウト
3	1	反時計回りボックスアウト
4	0	ラスト走査
4	1	逆ラスト走査
5	0	右ワイプ
5	1	左ワイプ

このような場合において、規定された成長順序で MapUnitsInSliceGroup0 個のスライスグループマップユニットはスライスグループ 0 に割り当てられ、そのピクチャの残りの PicSizeInMapUnits-MapUnitsInSliceGroup0 個のスライスグループマップユニットはスライスグループ 1 に割り当てられる。

num\_slice\_groups\_minus1 が 1 に等しく、そして slice\_group\_map\_type が 4 あるいは 5 に等しい時、変数 sizeOfUpperLeftGroup は次の通り定義される。

$$\text{sizeOfUpperLeftGroup} = (\text{slice\_group\_change\_direction\_flag} ? (\text{PicSizeInMapUnits} - \text{MapUnitsInSliceGroup0}) : \text{MapUnitsInSliceGroup0}) \quad (8-15)$$

変数 mapUnitToSliceGroupMap は次の通り導出される。

- もし num\_slice\_groups\_minus1 が 0 に等しいならば、マップユニットからスライスグループへのマップは、0 から PicSizeInMapUnits-1 までのそれぞれの値を含む範囲にある全ての i に対して、次式で規定されるように生成される。

$$\text{mapUnitToSliceGroupMap}[i] = 0 \quad (8-16)$$

- それ以外 (num\_slice\_groups\_minus1 が 0 に等しくない)、mapUnitToSliceGroupMap は次の通り導出される。

- もし slice\_group\_map\_type が 0 に等しいならば、8.2.2.1 小節で規定される mapUnitToSliceGroupMap の導出を適用する。

- それ以外、もし `slice_group_map_type` が 1 に等しいならば、8.2.2.2 小小節で規定される `mapUnitToSliceGroupMap` の導出を適用する。
- それ以外、もし `slice_group_map_type` が 2 に等しいならば、8.2.2.3 小小節で規定される `mapUnitToSliceGroupMap` の導出を適用する。
- それ以外、もし `slice_group_map_type` が 3 に等しいならば、8.2.2.4 小小節で規定される `mapUnitToSliceGroupMap` の導出を適用する。
- それ以外、もし `slice_group_map_type` が 4 に等しいならば、8.2.2.5 小小節で規定される `mapUnitToSliceGroupMap` の導出を適用する。
- それ以外、もし `slice_group_map_type` が 5 に等しいならば、8.2.2.6 小小節で規定される `mapUnitToSliceGroupMap` の導出を適用する。
- それ以外（`slice_group_map_type` が 6 に等しい）、8.2.2.7 小小節で規定される `mapUnitToSliceGroupMap` の導出を適用する。

`mapUnitToSliceGroupMap` の導出の後、8.2.2.8 小小節で規定される処理は、マップユニットからスライスグループへのマップ `mapUnitToSliceGroupMap` を、マクロブロックからスライスグループへのマップ `MbToSliceGroupMap` に変換するために起動される。8.2.2.8 小小節で規定されるマクロブロックからスライスグループへのマップの導出の後、関数 `NextMbAddress(n)` は次式により規定されるように導出される変数 `nextMbAddress` の値で定義される。

```

i=n+1
while(i<PicSizeInMbs && MbToSliceGroupMap[i]!=MbToSliceGroupMap[n])
    i++;
nextMbAddress=i

```

(8-17)

#### 8.2.2.1 インターリーブスライスグループマップタイプの規定

`slice_group_map_type` が 0 に等しい時、この小小節での規定が適用される。

マップユニットからスライスグループへのマップは、次式で規定されるように生成される。

```

i=0
do
    for(iGroup=0;iGroup<=num_slice_groups_minus1 && i<PicSizeInMapUnits;
        i+=run_length_minus1[iGroup++]+1)
        for(j=0;j<=run_length_minus1[iGroup] && i+j<PicSizeInMapUnits;j++)
            mapUnitToSliceGroupMap[i+j]=iGroup
while( i<PicSizeInMapUnits)

```

(8-18)

#### 8.2.2.2 分散スライスグループマップタイプの規定

`slice_group_map_type` が 1 に等しい時、この小小節での規定が適用される。

マップユニットからスライスグループへのマップは、次式で規定されるように生成される。

```

for(i=0;i<PicSizeInMapUnits;i++)
    mapUnitToSliceGroupMap[i]= ((i%PicWidthInMbs)+
                                (((i/PicWidthInMbs)*(num_slice_groups_minus1+1))/2)
                                %(num_slice_groups_minus1+1)
                                (8-19)

```

### 8.2.2.3 残存を持つ前景スライスグループマップタイプの規定

slice\_group\_map\_type が 2 に等しい時、この小節での規定が適用される。

マップユニットからスライスグループへのマップは、次式で規定されるように生成される。

```

for(i=0;i<PicSizeInMapUnits;i++)
    mapUnitToSliceGroupMap[i]=num_slice_groups_minus1
for(iGroup=num_slice_groups_minus1-1;iGroup>=0;iGroup--){
    yTopLeft=top_left[iGroup]/PicWidthInMbs
    xTopLeft=top_left[iGroup]%PicWidthInMbs
    yBottomRight=bottom_right[iGroup]/PicWidthInMbs
    xBottomRight=bottom_right[iGroup]%PicWidthInMbs
    for(y=yTopLeft;y<=yBottomRight;y++)
        for(x=xTopLeft;x<=xBottomRight;x++)
            mapUnitToSliceGroupMap[y*PicWidthInMbs+x]=iGroup
    }
(8-20)

```

記 矩形は重なるかもしれない。スライスグループ0は、top\_left[0]とbottom\_right[0]で規定される矩形内にあるマクロブロックを含んでいる。0より大きくnum\_slice\_groups\_minus1より小さいスライスグループIDを持つあるスライスグループは、そのスライスグループのために規定された矩形内にあり、より小さいスライスグループIDを持つ全てのスライスグループのために規定された矩形内にはない、マクロブロックを含んでいる。num\_slice\_groups\_minus1に等しいスライスグループIDを持つスライスグループは、その他のスライスグループにはないマクロブロックを含んでいる。

### 8.2.2.4 ボックスアウトスライスグループマップタイプの規定

slice\_group\_map\_type が 3 に等しい時、この小節での規定が適用される。

マップユニットからスライスグループへのマップは、次式で規定されるように生成される。

```

for(i=0;i<PicSizeInMapUnits;i++)
    mapUnitToSliceGroupMap[i]=1
x=(PicWidthInMbs-slice_group_change_direction_flag)/2
y=(PicHeightInMapUnits-slice_group_change_direction_flag)/2
(leftBound,topBound)=(x,y)
(rightBound,bottomBound)=(x,y)
(xDir,yDir)=(slice_group_change_direction_flag-1,slice_group_change_direction_flag)
for(k=0;k<MapUnitsInSliceGroup0;k+=mapUnitVacant){
    mapUnitVacant=(mapUnitToSliceGroupMap[y*PicWidthInMbs+x]==1)
    if(mapUnitVacant)
        mapUnitToSliceGroupMap[y*PicWidthInMbs+x]=0
    if(xDir==-1 && x==leftBound){
        leftBound=Max(leftBound-1,0)
(8-21)

```

```

    x=leftBound
    (xDir,yDir)=(0,2*slice_group_change_direction_flag-1)
} else if(xDir==1 && x==rightBound){
    rightBound=Min(rightBound+1,PicWidthInMbs-1)
    x=rightBound
    (xDir,yDir)=(0,1-2*slice_group_change_direction_flag)
} else if(yDir==-1 && y==topBound){
    topBound=Max(topBound-1,0)
    y=topBound
    (xDir,yDir)=(1-2*slice_group_change_direction_flag,0)
} else if(yDir==1 && y==bottomBound){
    bottomBound=Min(bottomBound+1,PicHeightInMapUnits-1)
    y=bottomBound
    (xDir,yDir)=(2*slice_group_change_direction_flag-1,0)
} else
    (x,y)=(x+xDir,y+yDir)
}

```

#### 8.2.2.5 ラスタ走査スライスグループマップタイプの規定

slice\_group\_map\_type が 4 に等しい時、この小小節での規定が適用される。

マップユニットからスライスグループへのマップは、次式で規定されるように生成される。

```

for(i=0;i<PicSizeInMapUnits;i++)
    if(i<sizeOfUpperLeftGroup)
        mapUnitToSliceGroupMap[i]=slice_group_change_direction_flag
    else
        mapUnitToSliceGroupMap[i]=1-slice_group_change_direction_flag

```

(8-22)

#### 8.2.2.6 ワイプスライスグループマップタイプの規定

slice\_group\_map\_type が 5 に等しい時、この小小節での規定が適用される。

マップユニットからスライスグループへのマップは、次式で規定されるように生成される。

```

k=0;
for(j=0;j<PicWidthInMbs;j++)
    for(i=0;i<PicHeightInMapUnits;i++)
        if(k++<sizeOfUpperLeftGroup)
            mapUnitToSliceGroupMap[i*PicWidthInMbs+j]=slice_group_change_direction_flag
        else
            mapUnitToSliceGroupMap[i*PicWidthInMbs+j]=1-slice_group_change_direction_flag

```

(8-23)

#### 8.2.2.7 明示的なスライスグループマップタイプの規定

slice\_group\_map\_type が 6 に等しい時、この小小節での規定が適用される。

マップユニットからスライスグループへのマップは、0 から PicSizeInMapUnits-1 のそれぞれの値を含む範囲にある全ての i に対して、次式で規定されるように生成される。

$$\text{mapUnitToSliceGroupMap}[i]=\text{slice\_group\_id}[i] \quad (8-24)$$

### 8.2.2.8 マップユニットからスライスグループへのマップから、マクロブロックからスライスグループへのマップへの変換のための規定

マクロブロックからスライスグループへのマップは、0 から PicSizeInMbs-1 のそれぞれの値を含む範囲にある i の値それぞれに対して、次の通り規定される。

- もし frame\_mbs\_only\_flag が 1 に等しい、あるいは field\_pic\_flag が 1 に等しいならば、マクロブロックからスライスグループへのマップは次のように規定される。

$$\text{MbToSliceGroupMap}[i]=\text{mapUnitToSliceGroupMap}[i] \quad (8-25)$$

- それ以外、もし MbaffFrameFlag が 1 に等しいならば、マクロブロックからスライスグループへのマップは次のように規定される。

$$\text{MbToSliceGroupMap}[i]=\text{mapUnitToSliceGroupMap}[i/2] \quad (8-26)$$

- それ以外 (frame\_mbs\_only\_flag が 0 に等しく、そして mb\_adaptive\_frame\_field\_flag が 0 に等しく、そして field\_pic\_flag が 0 に等しい)、マクロブロックからスライスグループへのマップは次のように規定される。

$$\begin{aligned} \text{MbToSliceGroupMap}[i]=\text{mapUnitToSliceGroupMap}[(i/(2*\text{PicWidthInMbs}))*\text{PicWidthInMbs} \\ + (i\% \text{PicWidthInMbs})] \end{aligned} \quad (8-27)$$

### 8.2.3 スライスデータパーティションの復号処理

この処理の入力は以下の通り。

- スライスデータパーティション A レイヤ RBSP、
- カテゴリ 3 のシンタックス要素がスライスデータに存在する時、スライスデータパーティション A レイヤ RBSP と同じ slice\_id を持つスライスデータパーティション B レイヤ RBSP、および、
- カテゴリ 4 のシンタックス要素がスライスデータに存在する時、スライスデータパーティション A レイヤ RBSP と同じ slice\_id を持つスライスデータパーティション C レイヤ RBSP。

記1 スライスデータパーティション B レイヤ RBSP とスライスデータパーティション C レイヤ RBSP は存在する必要はない。

この処理の出力は符号化スライスである。

スライスデータパーティションが使用されない時、符号化スライスは、スライスヘッダとそれに続くそのスライスのマクロブロックに対するマクロブロックデータのカテゴリ 2、3 および 4 (7.3 節のカテゴリ列を参照) の全てのシンタックス要素を含むスライスデータシンタックス構造を含む、パーティションのないスライスレイヤ RBSP で表現される。

スライスデータパーティションが使用される時、スライスのマクロブロックデータは、分離された NAL ユニットに含まれる 1 つから 3 つのパーティションに分割される。パーティション A は、スライスデータパーティション A ヘッダ、そしてカテゴリ 2 の全てのシンタックス要素を含んでいる。パーティション B が

存在する時、スライスデータパーティションBヘッダとカテゴリ3の全てのシンタックス要素を含んでいる。パーティションCが存在する時、スライスデータパーティションCヘッダとカテゴリ4の全てのシンタックス要素を含んでいる。

スライスデータパーティションが使用される時、各々のカテゴリのシンタックス要素は、分離されたNALユニットから構文解析される。そのNALユニットは、それぞれのカテゴリのシンボルが存在しない時、存在する必要がない。復号処理は、スライスデータパーティションの中でシンタックス要素が7.3節のシンタックス表のスライスデータパーティション割り当てに依存して現れる、スライスデータパーティションから各々のシンタックス要素を抽出することによって、対応するパーティションのないスライスレイヤRBSPの処理と同等の方法で、符号化スライスのスライスデータパーティションを処理しなくてはならない。

記2 カテゴリ3のシンタックス要素は、IおよびSIマクロブロックタイプの残差データの復号に関連がある。カテゴリ4のシンタックス要素は、PおよびBマクロブロックタイプの残差データの復号に関連がある。カテゴリ2は、マクロブロックの復号に関連づけられる全ての他のシンタックス要素を包含し、それらの情報はヘッダ情報としてしばしば示される。スライスデータパーティションAヘッダは、スライスヘッダの全てのシンタックス要素を含んでいる。そして、さらにスライスデータパーティションBとCをスライスデータパーティションAに関連付けるために使われるslice\_idを含んでいる。スライスデータパーティションBとCヘッダは、そのスライスのスライスデータパーティションAとの関連付けを確立するslice\_idシンタックス要素を含んでいる。

#### 8.2.4 参照ピクチャリスト構築の復号処理

この処理はP、SPあるいはBスライス各々の復号の開始で起動される。

復号参照ピクチャは、ビットストリームで規定され8.2.5小節で規定されるように“短期間参照使用”あるいは“長期間参照使用”としてマークされる。短期間参照ピクチャはframe\_numの値で識別される。長期間参照ピクチャは、ビットストリームで規定され8.2.5小節で規定されるように長期間フレームインデックスが割り当てられる。

8.2.4.1 小小節は、以下を規定するために起動される。

- 短期間参照ピクチャ各々への変数FrameNum、FrameNumWrapとPicNumの割り当て、そして
- 長期間参照ピクチャ各々への変数LongTermPicNumの割り当てを規定する。

参照ピクチャは、8.4.2.1小小節で規定されるように参照インデックスを通してアドレスされる。参照インデックスは、参照ピクチャリストへのインデックスである。PあるいはSPスライスを復号する時、1つの参照ピクチャリストRefPicList0がある。Bスライスを復号する時、RefPicList0に加えて第2の独立した参照ピクチャリストRefPicList1がある。

各々のスライスの復号の開始で、参照ピクチャリストRefPicList0、そしてBスライスのためのRefPicList1は次のように導出される。

- 初期の参照ピクチャリストRefPicList0とBスライスのためのRefPicList1は、8.2.4.2小小節で規定されるように導出される。
- 初期の参照ピクチャリストRefPicList0とBスライスのためのRefPicList1は、8.2.4.3小小節で規定されるように修正される。

記 8.2.4.3小小節で規定される参照ピクチャリストの並替処理は、RefPicList0とBスライスに対するRefPicList1の内容が柔軟な方法で修正されることを許容する。特に“参照使用”と現在マークされたピクチャは、たとえ、そのピクチャが8.2.4.2小小節で規定される導出された初期参照ピクチャリストに無い時でも、RefPicList0とBスライスに対するRefPicList1に挿入できる。

修正された参照ピクチャリスト RefPicList0 のエン트리数は、num\_ref\_idx\_l0\_active\_minus1+1 である。そして、B スライスのための修正された参照ピクチャリスト RefPicList1 のエン트리数は、num\_ref\_idx\_l1\_active\_minus1+1 である。1つの参照ピクチャは、修正された参照ピクチャリスト RefPicList0 あるいは RefPicList1 において1つより多くのインデックスに現れるかもしれない。

#### 8.2.4.1 ピクチャ番号の復号処理

この処理は、8.2.4 小節で規定される参照ピクチャリスト構築の復号処理、または 8.2.5 小節で規定される復号参照ピクチャマーキング処理が起動される時に起動される。

変数 FrameNum、FrameNumWrap、PicNum、LongTermFrameIdx および LongTermPicNum は、8.2.4.2 小節の参照ピクチャリストの初期化処理、8.2.4.3 小節の参照ピクチャリストの修正処理、そして 8.2.5 小節の復号参照ピクチャマーキング処理のために使用される。

各々の短期間参照ピクチャに対する変数 FrameNum と FrameNumWrap は、次のように割り当てられる。最初に、FrameNum は対応する短期間参照ピクチャのスライスヘッダで、すでに復号されているシンタックス要素 frame\_num に等しく設定される。そして、変数 FrameNumWrap は、次のように導出される。

```

if(FrameNum>frame_num)
    FrameNumWrap=FrameNum-MaxFrameNum
else
    FrameNumWrap=FrameNum
    
```

(8-28)

ここで、等式 8-28 で使用される frame\_num の値は、現ピクチャのスライスヘッダの frame\_num である。

各々の長期間参照ピクチャは、関連付けられた LongTermFrameIdx の値 (8.2.5 小節で規定されるようにそれに割り当てられる) をもつ。

各々の短期間参照ピクチャに対する変数 PicNum が割り当てられ、そして、各々の長期間参照ピクチャに対する変数 LongTermPicNum が割り当てられる。これらの変数の値は、現ピクチャの field\_pic\_flag と bottom\_field\_flag の値に依存し、それらは次のように設定される。

- もし field\_pic\_flag が 0 に等しいならば、次が適用される。

- 各々の短期間参照フレームあるいは相補的参照フィールドペアに対しては、

$$\text{PicNum} = \text{FrameNumWrap} \tag{8-29}$$

- 各々の長期間参照フレームあるいは長期間相補的参照フィールドペアに対しては、

$$\text{LongTermPicNum} = \text{LongTermFrameIdx} \tag{8-30}$$

記 フレームを復号する時、MbaffFrameFlag の値は 8.2.4.2、8.2.4.3 小節および 8.2.5 小節の導出に影響を及ぼさない。

- それ以外 (field\_pic\_flag が 1 に等しい)、次が適用される。

- 各々の短期間参照フィールドに対しては、次が適用される。

- もし参照フィールドが現フィールドと同じパリティを持つならば、

$$\text{PicNum} = 2 * \text{FrameNumWrap} + 1 \tag{8-31}$$

- それ以外（参照フィールドが現フィールドの逆パリティを持つ）、

$$\text{PicNum}=2*\text{FrameNumWrap} \quad (8-32)$$

- 各々の長期間参照フィールドに対しては、次が適用される。
- もし参照フィールドが現フィールドと同じパリティを持つならば、

$$\text{LongTermPicNum}=2*\text{LongTermFrameIdx}+1 \quad (8-33)$$

- それ以外（参照フィールドが現フィールドの逆パリティを持つ）、

$$\text{LongTermPicNum}=2*\text{LongTermFrameIdx} \quad (8-34)$$

#### 8.2.4.2 参照ピクチャリストの初期化処理

この初期化処理は、P、SP あるいは B スライスヘッダを復号する時、起動される。

RefPicList0 と RefPicList1 は、8.2.4.2.1 小小小節から 8.2.4.2.5 小小小節で規定されるように初期のエントリを持つ。

8.2.4.2.1 小小小節から 8.2.4.2.5 小小小節で規定されるように取り出された初期の RefPicList0 あるいは RefPicList1 の エントリ 数 が 、 それ ぞ れ num\_ref\_idx\_l0\_active\_minus1+1 あるいは num\_ref\_idx\_l1\_active\_minus1+1 よりも大きい時、num\_ref\_idx\_l0\_active\_minus1 あるいは num\_ref\_idx\_l1\_active\_minus1 より後ろの位置の余分なエントリは、初期の参照ピクチャリストから捨てられる。

8.2.4.2.1 小小小節から 8.2.4.2.5 小小小節で規定されるように取り出された初期の RefPicList0 あるいは RefPicList1 の エントリ 数 が 、 それ ぞ れ num\_ref\_idx\_l0\_active\_minus1+1 あるいは num\_ref\_idx\_l1\_active\_minus1+1 よりも小さい時、初期の参照ピクチャリストの残りのエントリは、“参照ピクチャ無し”に等しく設定される。

##### 8.2.4.2.1 フレームにおけるPおよびSPスライスの参照ピクチャリストの初期化処理

この初期化処理は、符号化フレームの P あるいは SP スライスを復号する時、起動される。

この処理が起動される時、現在“短期間参照使用”または“長期間参照使用”とマーク付けされた、少なくとも 1 つの参照フレームまたは相補的参照フィールドペアがなければならない。

参照ピクチャリスト RefPicList0 は、短期間参照フレームと短期間相補的参照フィールドペアが、長期間参照フレームと長期間相補的参照フィールドペアよりも低いインデックスを持つように順序付けられる。

短期間参照フレームと相補的参照フィールドペアは、最も高い PicNum 値のフレームあるいは相補的フィールドペアで開始し、最も低い PicNum 値のフレームあるいは相補的フィールドペアに降順で続くように順序付けられる。

長期間参照フレームと相補的参照フィールドペアは、最も低い LongTermPicNum 値のフレームあるいは相補的フィールドペアで開始し、最も高い LongTermPicNum 値のフレームあるいは相補的フィールドペアに昇順で続くように順序付けられる。

記 非ペア参照フィールドは、MbaffFrameFlagの値に関係なく、フレームを復号するためのINTER予測に使用されない。

例えば、3つの参照フレームが300、302そして303に等しいPicNumで“短期間参照使用”としてマークされ、そして、2つの参照フレームが0と3に等しいLongTermPicNumで“長期間参照使用”としてマークされる時、その初期のインデックス順序は

- RefPicList0[0]が、PicNum=303である短期間参照ピクチャに等しく設定され、
- RefPicList0[1]が、PicNum=302である短期間参照ピクチャに等しく設定され、
- RefPicList0[2]が、PicNum=300である短期間参照ピクチャに等しく設定され、
- RefPicList0[3]が、LongTermPicNum=0である長期間参照ピクチャに等しく設定され、そして
- RefPicList0[4]が、LongTermPicNum=3である長期間参照ピクチャに等しく設定される。

#### 8.2.4.2.2 フィールドにおけるPおよびSPスライスの参照ピクチャリストの初期化処理

この初期化処理は、符号化フィールドのPあるいはSPスライスを復号する時、起動される。

参照ピクチャリスト RefPicList0に含まれる各々のフィールドは、参照ピクチャリスト RefPicList0で別々のインデックスを持つ。

記 フィールドを復号する時、復号順序で同じ位置にあるフレームを復号した時の参照に利用可能なピクチャの、実際に少なくとも2倍参照に利用可能なピクチャがある。

参照フレームの2つの順序付けられたリスト refFrameList0ShortTerm と refFrameList0LongTerm は、次のように導出される。このフレームリストの構成目的のため、復号参照フレーム、相補的参照フィールドペア、非ペア参照フィールドおよび1つのフィールドが“短期間参照使用”あるいは“長期間参照使用”とマークされる参照フレームは全て、参照フレームとみなされる。

- 1つ以上の“短期間参照使用”とマークされるフィールドを持つ全てのフレームは、短期間参照フレームのリスト refFrameList0ShortTerm に含まれる。現フィールドが相補的参照フィールドペアの2番目のフィールド（復号順序で）であり、最初のフィールドが“短期間参照使用”とマークされる時、最初のフィールドは、短期間参照フレームのリスト refFrameList0ShortTerm に含まれる。refFrameList0ShortTerm は最も高い FrameNumWrap 値の参照フレームで開始し、最も低い FrameNumWrap 値の参照フレームに降順で続くように順序付けられる。
- 1つ以上の“長期間参照使用”とマークされるフィールドを持つ全てのフレームは、長期間参照フレームのリスト refFrameList0LongTerm に含まれる。現フィールドが相補的参照フィールドペアの2番目のフィールド（復号順序で）であり、最初のフィールドが“長期間参照使用”とマークされる時、最初のフィールドは、長期間参照フレームのリスト refFrameList0LongTerm に含まれる。refFrameList0LongTerm は最も低い LongTermFrameIdx 値の参照フレームで開始し、最も高い LongTermFrameIdx 値の参照フレームに昇順で続くように順序付けられる。

8.2.4.2.5 小小小節で規定される処理が、refFrameList0ShortTerm と refFrameList0LongTerm を入力として与えられて起動され、出力は RefPicList0 に割り当てられる。

#### 8.2.4.2.3 フレームにおけるBスライスの参照ピクチャリストの初期化処理

この初期化処理は、符号化フレームのBスライスを復号する時に起動される。

参照ピクチャリスト RefPicList0 と RefPicList1 の構成目的の為、以下において、参照エントリという用語

は、復号された参照フレームまたは相補的参照フィールドペアを呼ぶ。

この処理が起動される時、現在“短期間参照使用”または“長期間参照使用”とマーク付けされた、少なくとも1つの参照エントリがなければならない。

B スライスにおいて、参照ピクチャリスト RefPicList0 と RefPicList1 の短期間参照エントリの順序は、PicOrderCnt()で与えられるような出力順序に依存する。pic\_order\_cnt\_type が 0 に等しい時、8.2.5.2 小節で規定されるような“非存在”としてマークされる参照ピクチャは、RefPicList0 または RefPicList1 のどちらにも含まれない。

記1 gaps\_in\_frame\_num\_value\_allowed\_flagが1に等しい時、符号器は復号処理（特に、pic\_order\_cnt\_type が0に等しく、その場合にPicOrderCnt()が“非存在”フレームに対して推定されていない時）の適切な操作を保証するために、参照ピクチャリスト並替を使用すべきである。

参照ピクチャリスト RefPicList0 は、短期間参照エントリが、長期間参照エントリより低いインデックスを持つように順序付けられる。それは次の通り順序付けられる。

- entryShortTerm を、現在“短期間参照使用”としてマークされる全ての参照エントリに及ぶ変数とする。entryShortTerm のいくつかの値が、PicOrderCnt(CurrPic)より小さい PicOrderCnt(entryShortTerm)を持ち存在する時、これらの entryShortTerm の値は refPicList0 の始まりに、PicOrderCnt(entryShortTerm)の降順で置かれる。(存在する時) entryShortTerm の残りの値の全ては、それから PicOrderCnt(entryShortTerm)の昇順で refPicList0 に付け加えられる。
- 長期間参照エントリは、LongTermPicNum の最小値を持つ長期間参照エントリで開始し、LongTermPicNum の最大値を持つ長期間参照エントリへ昇順で続くように順序付けられる。

参照ピクチャリスト RefPicList1 は、短期間参照エントリが、長期間参照エントリより低いインデックスを持つように順序付けられる。それは次の通り順序付けられる。

- entryShortTerm を、現在“短期間参照使用”としてマークされる全ての参照エントリに及ぶ変数とする。entryShortTerm のいくつかの値が、PicOrderCnt(CurrPic)より大きい PicOrderCnt(entryShortTerm)を持ち存在する時、これらの entryShortTerm の値は refPicList1 の始まりに、PicOrderCnt(entryShortTerm)の昇順で置かれる。(存在する時) entryShortTerm の残りの値の全ては、それから PicOrderCnt(entryShortTerm)の降順で refPicList1 に付け加えられる。
- 長期間参照エントリは、LongTermPicNum の最小値を持つ長期間参照エントリで開始し、LongTermPicNum の最大値を持つ長期間参照エントリへ昇順で続くように順序付けられる。
- 参照ピクチャリスト RefPicList1 が 1 つより多いエントリを持ち、そして RefPicList1 が参照ピクチャリスト RefPicList0 と同一の時、最初の 2 つのエントリ RefPicList1[0]と RefPicList1[1]は交換される。

記2 非ペア参照フィールドは、(MbaffFrameFlagの値と無関係に) フレームのINTER予測のために使用されない。

#### 8.2.4.2.4 フィールドにおけるBスライスの参照ピクチャリストの初期化処理

この初期化処理は、符号化フィールドの B スライスを復号する時に起動される。

フィールドを復号する時、蓄積された参照フレームのそれぞれのフィールドは、唯一のインデックスを持つ別々の参照ピクチャとして識別される。参照ピクチャリスト RefPicList0 と RefPicList1 の短期間参照ピクチャの順序は、PicOrderCnt()で与えられるような出力順序に依存する。pic\_order\_cnt\_type が 0 に等しい時、8.2.5.2 小節で規定されるような“非存在”としてマークされる参照ピクチャは、RefPicList0 または

RefPicList1 のどちらにも含まれない。

記1 gaps\_in\_frame\_num\_value\_allowed\_flagが1に等しい時、符号器は復号処理（特に、pic\_order\_cnt\_typeが0に等しく、その場合にPicOrderCnt()が“非存在”フレームに対して推定されていない時）の適切な操作を保証するために、参照ピクチャリスト並替を使用すべきである。

記2 フィールドを復号する時、復号順序で同じ位置でフレームを復号した時の参照に利用可能なピクチャの、実際に少なくとも2倍参照に利用可能なピクチャがある。

参照フレームの 3 つの順序付けられたリスト refFrameList0ShortTerm、refFrameList1ShortTerm そして refFrameListLongTerm は、次のように導出される。これらフレームリストの構成目的のため、以下において、参照エントリという用語は、復号された参照フレーム、相補的参照フィールドペア、あるいは非ペア参照フィールドを呼ぶ。pic\_order\_cnt\_type が 0 に等しい時、参照エントリという用語は、8.2.5.2 小小節で規定されるような“非存在”としてマークされるフレームを参照しない。

- entryShortTerm を、現在“短期間参照使用”としてマークされる全ての参照エントリに及ぶ変数とする。entryShortTerm のいくつかの値が、PicOrderCnt(CurrPic)以下の PicOrderCnt(entryShortTerm)を持ち存在する時、これらの entryShortTerm の値は refFrameList0ShortTerm の始まりに、PicOrderCnt(entryShortTerm)の降順で置かれる。（存在する時）entryShortTerm の残りの値の全ては、それから PicOrderCnt(entryShortTerm)の昇順で refFrameList0ShortTerm に付け加えられる。

記3 現フィールドが、相補的参照フィールドペアと一緒に形づくる符号化フィールドfldPrevに復号順序で続く時、fldPrevはPicOrderCnt(fldPrev)を使用してリストrefFrameList0ShortTermに含まれる。そして、前の文で記述された順序付け方法が適用される。

- entryShortTerm を、現在“短期間参照使用”としてマークされる全ての参照エントリに及ぶ変数とする。entryShortTerm のいくつかの値が、PicOrderCnt(CurrPic)より大きい PicOrderCnt(entryShortTerm)を持ち存在する時、これらの entryShortTerm の値は refFrameList1ShortTerm の始まりに、PicOrderCnt(entryShortTerm)の昇順で置かれる。（存在する時）entryShortTerm の残りの値の全ては、それから PicOrderCnt(entryShortTerm)の降順で refFrameList1ShortTerm に付け加えられる。

記4 現フィールドが、相補的参照フィールドペアと一緒に形づくる符号化フィールドfldPrevに復号順序で続く時、fldPrevはPicOrderCnt(fldPrev)を使用してリストrefFrameList1ShortTermに含まれる。そして、前の文で記述された順序付け方法が適用される。

- refFrameListLongTerm は、最も低い LongTermFrameIdx 値を持つ参照エントリで開始し、最も高い LongTermFrameIdx 値を持つ参照エントリに昇順で続くように順序付けられる。

記5 現ピクチャの相補的フィールドが、“長期間参照使用”とマークされる時、それはリストrefFrameListLongTermに含まれる。1つのフィールドのみ“長期間参照使用”としてマークされる参照エントリは、リストrefFrameListLongTermに含まれる。

8.2.4.2.5 小小節で規定される処理が、refFrameList0ShortTerm と refFrameListLongTerm を入力として与えられて起動され、出力は RefPicList0 に割り当てられる。

8.2.4.2.5 小小節で規定される処理が、refFrameList1ShortTerm と refFrameListLongTerm を入力として与えられて起動され、出力は RefPicList1 に割り当てられる。

参照ピクチャリスト RefPicList1 が 1 つより多いエントリを持ち、そして RefPicList1 が参照ピクチャリスト RefPicList0 と同一の時、最初の 2 つのエントリ RefPicList1[0]と RefPicList1[1]は交換される。

#### 8.2.4.2.5 フィールドにおける参照ピクチャリストの初期化処理

この処理の入力は、参照フレームリスト refFrameListXShortTerm (X は 0 または 1 でもよい) と refFrameListLongTerm である。

参照ピクチャリスト RefPicListX は、短期間参照フィールドが長期間参照フィールドより低いインデックスを持つように順序付けられたリストである。参照フレームリスト refFrameListXShortTerm と refFrameListLongTerm が与えられると、それは次の通り導出される。

- 短期間参照フィールドは、フレームの順序付けられたリスト refFrameListXShortTerm から、現フィールドと同じパリティを持つフィールド（存在する時）で開始する、パリティの違うフィールド間を交互に繰り返す参照フィールドの選択によって順序付けられる。参照フレームの1つのフィールドが復号されなかった、あるいは“短期間参照使用”としてマークされない時、欠いたフィールドは無視され、そしてその代わりに、フレームの順序付けられたリスト refFrameListXShortTerm から選択されたパリティを持つ次の利用可能な蓄積された参照フィールドが、RefPicListX に挿入される。順序付けられたフレームのリスト refFrameListXShortTerm に、交互に繰り返すパリティを持つ短期間参照フィールドがこれ以上存在しない場合は、次に、利用可能なパリティを持つまだインデックスされていないフィールドが、順序付けられたフレームのリスト refFrameListXShortTerm に、それらが現れる順序で、RefPicListX に挿入される。
- 長期間参照フィールドは、フレームの順序付けられたリスト refFrameListLongTerm から、現フィールドと同じパリティを持つフィールド（存在する時）で開始する、パリティの違うフィールド間を交互に繰り返す参照フィールドの選択によって順序付けられる。参照フレームの1つのフィールドが復号されなかった、あるいは“長期間参照使用”としてマークされない時、欠いたフィールドは無視され、そしてその代わりに、フレームの順序付けられたリスト refFrameListLongTerm から選択されたパリティを持つ次の利用可能な蓄積された参照フィールドが、RefPicListX に挿入される。順序付けられたフレームのリスト refFrameListLongTerm に、交互に繰り返すパリティを持つ長期間参照フィールドがこれ以上存在しない場合は、次に、利用可能なパリティを持つまだインデックスされていないフィールドが、順序付けられたフレームのリスト refFrameListLongTerm に、それらが現れる順序で、RefPicListX に挿入される。

#### 8.2.4.3 参照ピクチャリストの並替処理

ref\_pic\_list\_reordering\_flag\_l0 が 1 に等しい時、次が適用される。

- refIdxL0 を参照ピクチャリスト RefPicList0 へのインデックスとする。それは、0 に等しく初期設定される。
- 対応するシンタックス要素 reordering\_of\_pic\_nums\_idc は、それらがビットストリームで発生する順序で処理される。これらのシンタックス要素のそれぞれについては、次が適用される。
  - もし reordering\_of\_pic\_nums\_idc が 0 に等しい、あるいは 1 に等しいならば、8.2.4.3.1 小小小節で規定される処理が、refIdxL0 を入力として起動される。そして、出力は refIdxL0 に割り当てられる。
  - それ以外、もし reordering\_of\_pic\_nums\_idc が 2 に等しいならば、8.2.4.3.2 小小小節で規定される処理が、refIdxL0 を入力として起動される。そして、出力は refIdxL0 に割り当てられる。
  - それ以外 (reordering\_of\_pic\_nums\_idc が 3 に等しい)、参照ピクチャリスト RefPicList0 の並替処理が終了される。

ref\_pic\_list\_reordering\_flag\_l1 が 1 に等しい時、次が適用される。

- refIdxL1 を参照ピクチャリスト RefPicList1 へのインデックスと仮定する。それは、0 に等しく初期設定される。

- 対応するシンタックス要素 `reordering_of_pic_nums_idc` は、それらがビットストリームで発生する順序で処理される。これらのシンタックス要素のそれぞれについては、次が適用される。
  - もし `reordering_of_pic_nums_idc` が 0 に等しい、あるいは 1 に等しいならば、8.2.4.3.1 小小小節で規定される処理が `refIdxL1` を入力として起動される。出力は `refIdxL1` に割り当てられる。
  - それ以外、もし `reordering_of_pic_nums_idc` が 2 に等しいならば、8.2.4.3.2 小小小節で規定される処理が `refIdxL1` を入力として起動される。出力は `refIdxL1` に割り当てられる。
  - それ以外 (`reordering_of_pic_nums_idc` が 3 に等しい)、参照ピクチャリスト `RefPicList1` の並替処理が終了される。

#### 8.2.4.3.1 短期間参照ピクチャの参照ピクチャリストの並替処理

この処理の入力は、インデックス `refIdxLX` ( $X$  は 0 あるいは 1) である。

この処理の出力は、増加されたインデックス `refIdxLX` である。

変数 `picNumLXNoWrap` は、次の通り導出される。

- もし `reordering_of_pic_nums_idc` が 0 に等しいならば

```

if(picNumLXPred-(abs_diff_pic_num_minus1+1)<0)
    picNumLXNoWrap=picNumLXPred-(abs_diff_pic_num_minus1+1)+MaxPicNum          (8-35)
else
    picNumLXNoWrap=picNumLXPred-(abs_diff_pic_num_minus1+1)

```

- それ以外 (`reordering_of_pic_nums_idc` が 1 に等しい)、

```

if(picNumLXPred+(abs_diff_pic_num_minus1+1)>=MaxPicNum)
    picNumLXNoWrap=picNumLXPred+(abs_diff_pic_num_minus1+1)-MaxPicNum        (8-36)
else
    picNumLXNoWrap=picNumLXPred+(abs_diff_pic_num_minus1+1)

```

`picNumLXPred` は、変数 `picNumLXNoWrap` の予測値である。この小小小節で規定される処理が、1 つのライスの最初に起動される時 (すなわち、`ref_pic_list_reordering()` シンタックスで、0 あるいは 1 に等しい `reordering_of_pic_nums_idc` の最初の発生に対して)、`picNumL0Pred` と `picNumL1Pred` は `CurrPicNum` に等しく初期設定される。それぞれの `picNumLXNoWrap` の割り当ての後、`picNumLXNoWrap` の値は `picNumLXPred` に割り当てられる。

変数 `picNumLX` は次の通り導出される。

```

if(picNumLXNoWrap>CurrPicNum)
    picNumLX=picNumLXNoWrap-MaxPicNum          (8-37)
else
    picNumLX=picNumLXNoWrap

```

`picNumLX` は、“短期間参照使用”としてマークされる参照ピクチャの `PicNum` と等しくなければならない。そして、“非存在”としてマークされる短期間参照ピクチャの `PicNum` と等しくてはならない。

以下の手続きが、短期間ピクチャ番号 `picNumLX` のピクチャをインデックス位置 `refIdxLX` へ配置し、他の全ての残りのピクチャの位置をそのリストの後ろにシフトし、そして、`refIdxLX` の値を増加するように行われる。

```

for(cIdx=num_ref_idx_IX_active_minus1+1;cIdx>refIdxLX;cIdx--)
    RefPicListX[cIdx]=RefPicListX[cIdx-1]
RefPicListX[refIdxLX++]=picNumLXに等しいPicNumをもつ短期間参照ピクチャ
nIdx=refIdxLX
for(cIdx=refIdxLX;cIdx<=num_ref_idx_IX_active_minus1+1;cIdx++)
    if(PicNumF(RefPicListX[cIdx])!=picNumLX)
        RefPicListX[nIdx++]=RefPicListX[cIdx]

```

(8-38)

ここで関数 `PicNumF(RefPicListX[cIdx])` は次の通り導出される。

- もしピクチャ `RefPicListX[cIdx]` が “短期間参照使用” としてマーク付けされているならば、`PicNumF(RefPicListX[cIdx])` はピクチャ `RefPicListX[cIdx]` の `PicNum` である。
- それ以外（ピクチャ `RefPicListX[cIdx]` が “短期間参照使用” としてマーク付けされていない）、`PicNumF(RefPicListX[cIdx])` は `MaxPicNum` に等しい。

記1 `MaxPicNum` の値は `picNumLX` に等しくなりえない。

記2 この擬似コードの手続き中で、リスト `RefPicListX` の長さは、最終リストに必要とされる長さより1要素長く一時的に構成される。この手続き実行の後で、そのリストの0から `num_ref_idx_IX_active_minus1` の要素だけが保持される必要がある。

#### 8.2.4.3.2 長期間参照ピクチャの参照ピクチャリストの並替処理

この処理の入力は、インデックス `refIdxLX` (`X` は 0 あるいは 1) である。

この処理の出力は、増加されたインデックス `refIdxLX` である。

以下の手続きが、長期間ピクチャ番号 `long_term_pic_num` のピクチャをインデックス位置 `refIdxLX` へ配置し、他の全ての残りのピクチャの位置をそのリストの後ろにシフトし、そして、`refIdxLX` の値を増加するように行われる。

```

for(cIdx=num_ref_idx_IX_active_minus1+1;cIdx>refIdxLX;cIdx--)
    RefPicListX[cIdx]=RefPicListX[cIdx-1]
RefPicListX[refIdxLX++]=long_term_pic_numに等しいLongTermPicNumをもつ長期間参照ピクチャ
nIdx=refIdxLX
for(cIdx=refIdxLX;cIdx<=num_ref_idx_IX_active_minus1+1;cIdx++)
    if(LongTermPicNumF(RefPicListX[cIdx])!=long_term_pic_num)
        RefPicListX[nIdx++]=RefPicListX[cIdx]

```

(8-39)

ここで関数 `LongTermPicNumF(RefPicListX[cIdx])` は次の通り導出される。

- もしピクチャ `RefPicListX[cIdx]` が “長期間参照使用” としてマーク付けされているならば、`LongTermPicNumF(RefPicListX[cIdx])` はピクチャ `RefPicListX[cIdx]` の `LongTermPicNum` である。
- それ以外（ピクチャ `RefPicListX[cIdx]` が “長期間参照使用” としてマーク付けされていない）、`LongTermPicNumF(RefPicListX[cIdx])` は  $2 * (\text{MaxLongTermFrameIdx} + 1)$  に等しい。

記1  $2*(\text{MaxLongTermFrameIdx}+1)$ の値は`long_term_pic_num`に等しくなりえない。

記2 この擬似コードの手続きの中で、リスト`RefPicListX`の長さは、最終リストに必要とされる長さより1要素長く一時的に構成される。この手続きの実行の後で、そのリストの0から`num_ref_idx_lX_active_minus1`の要素だけが保持される必要がある。

## 8.2.5 復号参照ピクチャマーキング処理

この処理は、`nal_ref_idc`が0に等しくない時、復号ピクチャに対して起動される。

記 8.2.5.2 小小節で規定される`frame_num`のギャップに対する復号処理は、8章に規定されるように、`nal_ref_idc`が0に等しい時にも起動されてよい。

参照ピクチャとして参照される、0に等しくない`nal_ref_idc`を持つ復号ピクチャは、“短期間参照使用”あるいは“長期間参照使用”としてマークされる。復号参照フレームについては、その両フィールドは、そのフレームと同じようにマークされる。相補的参照フィールドペアについては、そのペアは、その両フィールドと同じようにマークされる。“短期間参照使用”としてマークされるピクチャは、その`FrameNum`と、フィールドの時、そのパリティで識別される。“長期間参照使用”としてマークされるピクチャは、その`LongTermFrameIdx`と、フィールドの時、そのパリティで識別される。

“短期間参照使用”あるいは“長期間参照使用”としてマークされるフレームあるいは相補的フィールドペアは、そのフレーム、その相補的フィールドペア、あるいは、その構成フィールドの1つが“参照不使用”としてマークされるまで、フレームを復号する時、INTER 予測の参照として使用できる。“短期間参照使用”あるいは“長期間参照使用”としてマークされるフィールドは、“参照不使用”としてマークされるまで、フィールドを復号する時、INTER 予測の参照として使用できる。

ピクチャは、8.2.5.3 小小節で規定される先入れ先出し機構のスライディングウィンドウ参照ピクチャマーキング処理、または、8.2.5.4 小小節で規定されるカスタマイズされた適応マーキング操作の適応メモリ制御参照ピクチャマーキング処理によって“参照不使用”としてマークできる。

短期間参照ピクチャは、復号処理内の使用では、その変数`FrameNum`と`FrameNumWrap`、そしてそのピクチャ番号`PicNum`によって識別される。そして、長期間参照ピクチャは、復号処理内の使用では、その長期間ピクチャ番号`LongTermPicNum`によって識別される。現ピクチャがIDRピクチャでない時、8.2.4.1 小小節は、変数`FrameNum`、`FrameNumWrap`、`PicNum`と`LongTermPicNum`の割り当てを規定するため起動される。

### 8.2.5.1 復号参照ピクチャマーキング処理の操作シーケンス

復号参照ピクチャマーキングは、次の順序付けられたステップで進行する。

1. 現ピクチャの全てのスライスが復号される。
2. 現ピクチャがIDRピクチャであるかどうかによって依存して、次が適用される。
  - もし現ピクチャがIDRピクチャであるならば、次が適用される。
  - 全ての参照ピクチャは“参照不使用”としてマークされる。
  - `long_term_reference_flag`に依存して、次が適用される。
  - もし`long_term_reference_flag`が0に等しいならば、そのIDRピクチャは“短期間参照使用”としてマークされる。そして、`MaxLongTermFrameIdx`は“長期間フレームインデックス無し”に等しく設定される。

- それ以外 (`long_term_reference_flag` が 1 に等しい)、その IDR ピクチャは“長期間参照使用”としてマークされる。IDR ピクチャの `LongTermFrameIdx` は 0 に等しく設定される。そして、`MaxLongTermFrameIdx` は 0 に等しく設定される。
- それ以外 (現ピクチャが IDR ピクチャではない)、次が適用される。
  - もし `adaptive_ref_pic_marking_mode_flag` が 0 に等しいならば、8.2.5.3 小小節で規定される処理が起動される。
  - それ以外 (`adaptive_ref_pic_marking_mode_flag` が 1 に等しい)、8.2.5.4 小小節で規定される処理が起動される。

3. 現ピクチャが IDR ピクチャでなく、6 に等しい `memory_management_control_operation` で“長期間参照使用”としてマークされなかった時、それは“短期間参照使用”としてマークされる。

現復号参照ピクチャのマーキングの後、“参照使用”としてマークされた少なくとも 1 つのフィールドを持つフレームの総数と、“参照使用”としてマークされた少なくとも 1 つのフィールドを持つ相補的フィールドペアの数と、“参照使用”としてマークされた非ペアフィールドの数との総和は、 $\text{Max}(\text{num\_ref\_frames}, 1)$  よりも大きくなってはいけない。

#### 8.2.5.2 `frame_num`のギャップの復号処理

この処理は、`frame_num` が `PrevRefFrameNum` に等しくない時、そして  $(\text{PrevRefFrameNum} + 1) \% \text{MaxFrameNum}$  に等しくない時、起動される。

記1 この処理は、(`nal_ref_idc`が0に等しくない時のみに起動される処理を定義する) 8.2.5 小節のある小小節として規定されているが、(8章に規定されるように) この処理は `nal_ref_idc` が 0 に等しい時にも起動されてよい。本標準の構造におけるこの小小節の位置に関する理由は歴史的なものである。

記2 この処理は、`gaps_in_frame_num_value_allowed_flag` が 1 に等しい時だけ、適合するビットストリームに対して起動される。`gaps_in_frame_num_value_allowed_flag` が 0 に等しく、かつ `frame_num` が `PrevRefFrameNum` に等しくなく、 $(\text{PrevRefFrameNum} + 1) \% \text{MaxFrameNum}$  にも等しくない時、復号処理は意図しないピクチャロスと推定すべきである。

この処理が起動される時、“非存在”ピクチャに属する `frame_num` 値の集合は、現ピクチャの `frame_num` 値を除いて、等式 7-22 の `UnusedShortTermFrameNum` で取り出される全ての値として導出される。

復号処理は、8.2.5.3 小小節で規定される“スライディングウィンドウ”ピクチャマーキング処理を使用して、等式 7-22 で生成される `UnusedShortTermFrameNum` 値の順序で、“非存在”ピクチャに属する、それぞれの `frame_num` 値に対するフレームを生成しマークする。生成されるフレームもまた、“非存在”そして“短期間参照使用”としてマークされる。生成されるフレームのサンプル値は、どんな値に設定されてもよい。ビットストリームは、“非存在”としてマークされるこれら生成されるフレームへの INTER 予測での参照、短期間参照ピクチャの参照ピクチャリストの並替コマンドでのこれらフレームへの参照 (8.2.4.3.1 小小節)、または短期間参照ピクチャへの `LongTermFrameIdx` の割り当て処理でのこれらフレームへの参照 (8.2.5.4.3 小小節) を結果として生じるデータを含んではならない。

`pic_order_cnt_type` が 0 に等しくない時、`TopFieldOrderCnt` と `BottomFieldOrderCnt` は、8.2.1 小節のピクチャ順序カウントに対する復号処理の起動によって“非存在”フレームの各々に対して導出される。特定の“非存在”フレームに対して 8.2.1 小節の処理を起動する時、現ピクチャは、`UnusedShortTermFrameNum` に等しいと推定される `frame_num`、0 に等しくないと推定される `nal_ref_idc`、5 に等しくないと推定される `nal_unit_type`、0 に等しいと推定される `IdrPicFlag`、0 に等しいと推定される `field_pic_flag`、0 に等しいと推定

される `adaptive_ref_pic_marking_mode_flag`、(もし必要なら) 0 に等しいと推定される `delta_pic_order_cnt[0]`、そして (もし必要なら) 0 に等しいと推定される `delta_pic_order_cnt[1]`、をもつと考えられるピクチャであるとみなされる。

記3 “非存在”ピクチャに属する、これらの`frame_num`値のいずれかが、INTER予測処理で参照される、短期間参照ピクチャに対する参照ピクチャリストにおける並替コマンド(8.2.4.3.1小小小節)で参照される、あるいは短期間参照ピクチャへの`LongTermFrameIdx`の割り当て処理(8.2.5.4.3小小小節)で参照される時、復号処理は意図しないピクチャロスと推定すべきである。3に等しくないメモリ管理制御操作が“非存在”としてマークされるフレームに適用される時、復号処理は意図しないピクチャロスと推定すべきではない。

### 8.2.5.3 スライディングウィンドウ復号参照ピクチャマーキング処理

この処理は `adaptive_ref_pic_marking_mode_flag` が 0 に等しい時、起動される。

以下に規定されるような現ピクチャの特性に依存して、次が適用される。

- もし現ピクチャが、相補的参照フィールドペアの復号順序で2番目のフィールドであり、かつその最初のフィールドが“短期間参照使用”としてマークされた符号化フィールドであるならば、現ピクチャもまた“短期間参照使用”としてマークされる。
- それ以外、次が適用される。
  - `numShortTerm` を、少なくとも1つのフィールドが“短期間参照使用”としてマークされる、参照フレーム、相補的参照フィールドペア、そして、非ペア参照フィールドの総数とする。`numLongTerm` を、少なくとも1つのフィールドが“長期間参照使用”としてマークされる、参照フレーム、相補的参照フィールドペア、そして、非ペア参照フィールドの総数とする。
  - `numShortTerm+numLongTerm` が  $\text{Max}(\text{num\_ref\_frames}, 1)$  に等しい時、`numShortTerm` が 0 より大きいという条件が満たされなくてはいけない。そして、`FrameNumWrap` の最小値を持つ、その短期間参照フレーム、相補的参照フィールドペア、あるいは非ペア参照フィールドは、“参照不使用”としてマークされる。それがフレームあるいは相補的フィールドペアである時、その両フィールドもまた“参照不使用”としてマークされる。

### 8.2.5.4 適応メモリ制御復号参照ピクチャマーキング処理

この処理は、`adaptive_ref_pic_marking_mode_flag` が 1 に等しい時、起動される。

1 から 6 の値の `memory_management_control_operation` コマンドは、現ピクチャが復号された後、それらがビットストリームで発生する順序で処理される。これらの `memory_management_control_operation` コマンドのそれぞれについては、8.2.5.4.1 小小小節から 8.2.5.4.5 小小小節で規定される処理の1つが、`memory_management_control_operation` の値に依存して起動される。0 の値の `memory_management_control_operation` コマンドは、`memory_management_control_operation` コマンドの終わりを規定する。

メモリ管理制御操作は、ピクチャに次の通り適用される。

- もし `field_pic_flag` が 0 に等しいならば、`memory_management_control_operation` コマンドは、規定されたフレームあるいは相補的参照フィールドペアに適用される。
- それ以外 (`field_pic_flag` が 1)、`memory_management_control_operation` コマンドは、規定された個々の参照フィールドに適用される。

#### 8.2.5.4.1 “参照不使用”としての短期間参照ピクチャマーキング処理

この処理は、memory\_management\_control\_operation が 1 に等しい時、起動される。

picNumX を次式によって規定されるとする。

$$\text{picNumX} = \text{CurrPicNum} - (\text{difference\_of\_pic\_nums\_minus1} + 1) \quad (8-40)$$

field\_pic\_flag に依存して、picNumX 値は“参照不使用”として短期間参照ピクチャをマークするために次の通り使用される。

- もしfield\_pic\_flagが0に等しいならば、picNumXで規定される短期間参照フレームあるいは短期間相補的参照フィールドペア、そしてその両フィールドは、“参照不使用”としてマークされる。
- それ以外 (field\_pic\_flagが1に等しい)、picNumXで規定される短期間参照フィールドは、“参照不使用”としてマークされる。その参照フィールドが参照フレーム、相補的参照フィールドペアの一部である時、そのフレーム、あるいは相補的フィールドペアもまた“参照不使用”としてマークされるが、もう一方のフィールドのマーキングは変更されない。

#### 8.2.5.4.2 “参照不使用”としての長期間参照ピクチャマーキング処理

この処理は、memory\_management\_control\_operation が 2 に等しい時、起動される。

field\_pic\_flag に依存して、LongTermPicNum 値は“参照不使用”として長期間参照ピクチャをマークするために次の通り使用される。

- もしfield\_pic\_flagが0に等しいならば、long\_term\_pic\_numに等しいLongTermPicNumを持つ長期間参照フレームあるいは、長期間相補的参照フィールドペア、そしてその両フィールドは、“参照不使用”としてマークされる。
- それ以外 (field\_pic\_flagが1に等しい)、long\_term\_pic\_numに等しいLongTermPicNumで規定される長期間参照フィールドは、“参照不使用”としてマークされる。その参照フィールドが、参照フレームあるいは相補的参照フィールドペアの一部である時、そのフレームあるいは相補的フィールドペアもまた“参照不使用”としてマークされるが、もう一方のフィールドのマーキングは変更されない。

#### 8.2.5.4.3 短期間参照ピクチャへのLongTermFrameIdxの割り当て処理

この処理は、memory\_management\_control\_operation が 3 に等しい時、起動される。

シンタックス要素 difference\_of\_pic\_nums\_minus1 が与えられると、変数 picNumX は 8.2.5.4.1 小小小節で規定されるように獲得される。picNumX は“短期間参照使用”としてマークされ“非存在”としてマークされないフレーム、あるいは相補的参照フィールドペア、あるいは非ペア参照フィールドを参照しなくてはならない。

long\_term\_frame\_idx に等しい LongTermFrameIdx が、長期間参照フレームあるいは長期間相補的参照フィールドペアに既に割り当てられている時、そのフレームあるいは相補的フィールドペアとその両フィールドは、“参照不使用”としてマークされる。LongTermFrameIdx が非ペア参照フィールドに既に割り当てられ、そしてそのフィールドが picNumX で規定されるピクチャの相補的フィールドでない時、そのフィールドは“参照不使用”としてマークされる。

field\_pic\_flag に依存して、LongTermFrameIdx 値は“短期間参照使用”から“長期間参照使用”へ、ピクチ

ャをマークするために次の通り使用される。

- もしfield\_pic\_flagが0に等しいならば、picNumXで規定される短期間参照フレームあるいは、短期間相補的参照フィールドペアとその両フィールドのマーキングは、“短期間参照使用”から“長期間参照使用”へ変更され、long\_term\_frame\_idxに等しいLongTermFrameIdxが割り当てられる。
- それ以外(field\_pic\_flagが1に等しい)、picNumXで規定される短期間参照フィールドのマーキングは、“短期間参照使用”から“長期間参照使用”へ変更され、long\_term\_frame\_idxに等しいLongTermFrameIdxが割り当てられる。そのフィールドが参照フレーム、相補的参照フィールドペアの一部であり、同じ参照フレーム、相補的参照フィールドペアの他方のフィールドも“長期間参照使用”としてマークされる時、そのフレーム、あるいは相補的フィールドペアもまた“長期間参照使用”としてマークされ、long\_term\_frame\_idxに等しいLongTermFrameIdxが割り当てられる。

#### 8.2.5.4.4 MaxLongTermFrameIdxの復号処理

この処理はmemory\_management\_control\_operationが4に等しい時、起動される。

LongTermFrameIdxがmax\_long\_term\_frame\_idx\_plus1-1より大きく、かつ“長期間参照使用”としてマークされている全てのピクチャは、“参照不使用”としてマークされる。

変数MaxLongTermFrameIdxは次の通り導出される。

- もしmax\_long\_term\_frame\_idx\_plus1が0に等しいならば、MaxLongTermFrameIdxは“長期間フレームインデックス無し”に等しく設定される。
- それ以外(max\_long\_term\_frame\_idx\_plus1が0より大きい)、MaxLongTermFrameIdxはmax\_long\_term\_frame\_idx\_plus1-1に等しく設定される。

記4に等しいmemory\_management\_control\_operationコマンドは、長期間参照ピクチャを“参照不使用”としてマークするために使用することができる。max\_long\_term\_frame\_idx\_plus1を転送する頻度は、本標準では規定されない。しかしながら、INTRA更新要求メッセージのようなエラーメッセージの受信下では、符号器は4に等しいmemory\_management\_control\_operationコマンドを送信すべきである。

##### 8.2.5.4.4.1 全参照ピクチャを“参照不使用”としてマークし、MaxLongTermFrameIdxを“長期間フレームインデックス無し”に設定する処理

この処理は、memory\_management\_control\_operationが5に等しい時、起動される。

全ての参照ピクチャは“参照不使用”としてマークされ、変数MaxLongTermFrameIdxは“長期間フレームインデックス無し”に等しく設定される。

##### 8.2.5.4.5 長期間フレームインデックスを現ピクチャに割り当てる処理

この処理は、memory\_management\_control\_operationが6に等しい時、起動される。

long\_term\_frame\_idxに等しい変数LongTermFrameIdxが既に長期間参照フレームあるいは長期間相補的参照フィールドペアに割り当てられている時は、そのフレームあるいは相補的フィールドペアとその両方のフィールドは“参照不使用”としてマークされる。LongTermFrameIdxが既に非ペア参照フィールドに割り当てられており、かつそのフィールドが現ピクチャの相補的フィールドでない時は、そのフィールドは“参照不使用”としてマークされる。

現ピクチャは“長期間参照使用”としてマークされ、long\_term\_frame\_idxに等しいLongTermFrameIdxが

割り当てられる。

`field_pic_flag` が 0 に等しい時、その両方のフィールドもまた“長期間参照使用”としてマークされ、かつ `long_term_frame_idx` に等しい `LongTermFrameIdx` が割り当てられる。

`field_pic_flag` が 1 に等しく、かつ現ピクチャが相補的参照フィールドペアの(復号順序で)2番目のフィールドであり、かつ相補的参照フィールドペアの1番目のフィールドも現在“長期間参照使用”としてマークされている時、その相補的参照フィールドペアもまた“長期間参照使用”としてマークされ、かつ `long_term_frame_idx` に等しい `LongTermFrameIdx` が割り当てられる。

現復号参照ピクチャのマーク付けの後、少なくとも1フィールドが“参照使用”としてマーク付けされたフレームの総数、足す少なくとも1フィールドが“参照使用”としてマーク付けされた相補的フィールドペアの数、足す“参照使用”としてマーク付けされた非ペアフィールドの数は  $\text{Max}(\text{num\_ref\_frames}, 1)$  より大きくてはならない。

記 いくつかの状況下では、上の文は、6に等しい `memory_management_control_operation` シンタックス要素が、1、2または4に等しい `memory_management_control_operation` シンタックス要素に対して、復号参照ピクチャマーキングシンタックスで現れる事のできる順序に制約を課すことがある。

### 8.3 INTRA 予測処理

この処理は I および SI マクロブロックタイプに対して起動される。

この処理への入力は、デブロックフィルタ処理前の構築サンプルであり、さらに `Intra_NxN` 予測モード (`NxN` は `4x4` または `8x8` に等しい) では、隣接マクロブロックからの `IntraNxNPredMode` の値も入力となる。

この処理の出力は、次の通り規定される。

- もしマクロブロック予測モードが `Intra_4x4` または `Intra_8x8` ならば、出力はデブロックフィルタ処理前の構築輝度サンプルと、(`ChromaArrayType` が 0 に等しくない時は) マクロブロックの色差予測サンプル `predC` である。ここで `C` は `Cb` と `Cr` に等しい。
- それ以外、もし `mb_type` が `I_PCM` に等しくないならば、出力はマクロブロックの輝度予測サンプル `predL` と、(`ChromaArrayType` が 0 に等しくない時は) マクロブロックの色差予測サンプル `predC` である。ここで `C` は `Cb` と `Cr` に等しい。
- それ以外 (`mb_type` が `I_PCM` に等しい)、出力はデブロックフィルタ処理前の構築された輝度と (`ChromaArrayType` が 0 に等しくない時は) 色差サンプルである。

変数 `MvCnt` は 0 に等しく設定される。

`mb_type` の値に従って、次が適用される。

- もし `mb_type` が `I_PCM` に等しいならば、8.3.5 小節に規定される `I_PCM` マクロブロックに対するサンプル構築処理が起動される。
- それ以外 (`mb_type` が `I_PCM` でない)、次が適用される。
  - 輝度成分の INTRA 予測モードの復号処理は、次の通り記載される。
    - もしマクロブロック予測モードが `Intra_4x4` ならば、8.3.1 小節に規定される輝度サンプルの `Intra_4x4` 予測処理が起動される。

- それ以外、もしマクロブロック予測モードが Intra8x8 ならば、8.3.2 小節に規定される Intra\_8x8 予測処理が起動される。
- それ以外(マクロブロック予測モードが Intra\_16x16)、8.3.3 小節に規定される Intra\_16x16 予測処理が S<sub>L</sub> を入力として起動され、出力はマクロブロック pred<sub>L</sub> の輝度予測サンプルである。
- ChromaArrayType が 0 に等しくない時、8.3.4 小節に規定される色差サンプルの INTRA 予測処理が S'<sub>Cb</sub> および S'<sub>Cr</sub> を入力として起動され、出力はマクロブロック pred<sub>Cb</sub> および pred<sub>Cr</sub> の色差予測サンプルである。

INTRA 予測処理に使われるサンプルは、いかなるデブロックフィルタ操作による変更よりも前のサンプル値である。

### 8.3.1 輝度サンプルの Intra\_4x4 予測処理

この処理はマクロブロック予測モードが Intra\_4x4 の時に起動される。

この処理への入力は、隣接マクロブロックあるいは隣接マクロブロックペアからの (利用可能であれば) Intra4x4PredMode あるいは、(利用可能であれば) Intra8x8PredMode の値である。

マクロブロックの輝度成分は 16 個の 4×4 輝度サンプルのブロックから構成される。これらのブロックは、6.4.3 小節に規定された逆 4×4 輝度ブロック走査処理を使って逆走査される。

luma4x4BlkIdx=0..15 であるマクロブロック輝度成分の 4×4 輝度ブロック全てに対して、8.3.1.1 小節に規定される Intra4x4PredMode の導出処理は、luma4x4BlkIdx と、さらに隣接マクロブロックで (復号順序で) 先に導かれた Intra4x4PredMode および Intra8x8PredMode を入力とし、変数 Intra4x4PredMode[luma4x4BlkIdx] を出力として起動される。

luma4x4BlkIdx=0..15 を使って索引される 4×4 サンプルの輝度ブロック各々に対して、

1. 8.3.1.2 小節の Intra\_4x4 サンプル予測処理が、luma4x4BlkIdx と、隣接輝度ブロックからのデブロックフィルタ処理前の構築された輝度サンプルを含む配列 S<sub>L</sub> を入力として起動され、出力は Intra\_4x4 輝度予測サンプル pred4x4<sub>L</sub>[x,y]、x,y=0..3 である。
2. 現マクロブロック内でインデックス luma4x4BlkIdx をもつ 4×4 輝度ブロックの左上のサンプルの位置が、luma4x4BlkIdx を入力として、6.4.3 小節の逆 4×4 輝度ブロック走査処理を起動することで導出され、出力は(xO,yO)に割り当てられ、x,y=0..3 で

$$\text{pred}_{L}[xO+x,yO+y]=\text{pred}4x4_{L}[x,y] \quad (8-41)$$

である。

3. 8.5 節の変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理が、pred<sub>L</sub> と luma4x4BlkIdx を入力とし、現 4×4 輝度ブロック S'<sub>L</sub> に対する構築サンプルを出力として起動される。

#### 8.3.1.1 Intra4x4PredModeの導出処理

この処理への入力は、4×4 輝度ブロックのインデックス luma4x4BlkIdx と、隣接マクロブロックで(復号順序で)先に導かれた変数配列 (利用可能であれば) Intra4x4PredMode と (利用可能であれば) Intra8x8PredMode である。

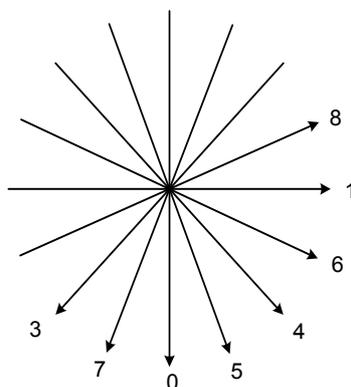
この処理の出力は、変数 Intra4x4PredMode[luma4x4BlkIdx]である。

表 8-2/JT-H264 は Intra4x4PredMode[luma4x4BlkIdx]の値と、各々に関連する名前を規定する。

**表8-2/JT-H264 Intra4x4PredMode[luma4x4BlkIdx]と関連する名前の規定  
(ITU-T H.264)**

Intra4x4PredMode[luma4x4BlkIdx]	Intra4x4PredMode[luma4x4BlkIdx]の名前
0	Intra_4x4_Vertical(予測モード)
1	Intra_4x4_Horizontal(予測モード)
2	Intra_4x4_DC(予測モード)
3	Intra_4x4_Diagonal_Down_Left(予測モード)
4	Intra_4x4_Diagonal_Down_Right(予測モード)
5	Intra_4x4_Vertical_Right(予測モード)
6	Intra_4x4_Horizontal_Down(予測モード)
7	Intra_4x4_Vertical_Left(予測モード)
8	Intra_4x4_Horizontal_Up(予測モード)

0、1、3、4、5、6、7と8で表された Intra4x4PredMode[luma4x4BlkIdx]は、図 8-1/JT-H264 に示されるような予測の方向を表す。



**図8-1/JT-H264 Intra\_4x4予測モードの方向 (参考)  
(ITU-T H.264)**

Intra4x4PredMode[luma4x4BlkIdx]は次の通り導出される。

- 6.4.10.4小節で規定された処理が、luma4x4BlkIdxを入力として与えられて起動され、出力はmbAddrA、luma4x4BlkIdxA、mbAddrBとluma4x4BlkIdxBに割り当てられる。
- 変数dcPredModePredictedFlagは次の通り導出される。

- もし以下の条件のいずれかが真ならば、dcPredModePredictedFlag は 1 に設定される。
  - アドレス mbAddrA のマクロブロックが利用不可である。
  - アドレス mbAddrB のマクロブロックが利用不可である。
  - アドレス mbAddrA のマクロブロックが利用可能、かつ INTER 予測モードで符号化され、かつ constrained\_intra\_pred\_flag が 1 である。
  - アドレス mbAddrB のマクロブロックが利用可能、かつ INTER 予測モードで符号化され、かつ constrained\_intra\_pred\_flag が 1 である。
- それ以外、dcPredModePredictedFlag は 0 に設定される。
- NがAかBのどちらかに置き換えられるものとする、変数intraMxMPredModeNは次の通り導出される。
  - もし dcPredModePredictedFlag が 1、あるいはアドレス mbAddrN のマクロブロックが Intra\_4x4 または Intra\_8x8 マクロブロック予測モードで符号化されていないならば、intraMxMPredModeN は 2(Intra\_4x4\_DC 予測モード)に設定される。
  - それ以外(dcPredModePredictedFlag が 0、かつ (アドレス mbAddrN のマクロブロックが Intra\_4x4 マクロブロック予測モードで符号化されている、あるいはアドレス mbAddrN のマクロブロックが Intra\_8x8 マクロブロック予測モードで符号化されている) )、次が適用される。
    - もしアドレス mbAddrN のマクロブロックが Intra\_4x4 マクロブロック予測モードで符号化されているならば、intraMxMPredModeN は Intra4x4PredMode[luma4x4BlkIdxN]に設定される。ここで Intra4x4PredMode はマクロブロック mbAddrN に割り当てられた配列変数である。
    - それ以外(アドレス mbAddrN のマクロブロックが Intra\_8x8 マクロブロック予測モードで符号化されているならば)、intraMxMPredModeN は Intra8x8PredMode[luma4x4BlkIdxN>>2]に設定される。ここで Intra8x8PredMode はマクロブロック mbAddrN に割り当てられた配列変数である。
- Intra4x4PredMode[luma4x4BlkIdx]は下記の手続きを適用して導出される。

```

predIntra4x4PredMode=Min(intraMxMPredModeA,intraMxMPredModeB)
if(prev_intra4x4_pred_mode_flag[luma4x4BlkIdx])
    Intra4x4PredMode[luma4x4BlkIdx]=predIntra4x4PredMode
else
    if(rem_intra4x4_pred_mode[luma4x4BlkIdx]<predIntra4x4PredMode)
        Intra4x4PredMode[luma4x4BlkIdx]=rem_intra4x4_pred_mode[luma4x4BlkIdx]
    else
        Intra4x4PredMode[luma4x4BlkIdx]=rem_intra4x4_pred_mode[luma4x4BlkIdx]+1

```

(8-42)

### 8.3.1.2 Intra\_4x4 サンプル予測

この処理は、Intra\_4x4 に等しい予測モードを持つマクロブロックの各 4×4 輝度ブロックに対して起動され、さらに、各 4×4 輝度ブロックの変換復号処理とデブロック前のピクチャ構築処理が後に続く。

この処理への入力は、

- 4×4 輝度ブロックのインデックス luma4x4BlkIdx

- 隣接マクロブロックのデブロックフィルタ処理前の構築された輝度サンプルを含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列  $cS_L$

である。

この処理の出力は、インデックス  $\text{luma4x4BlkIdx}$  をもつ  $4 \times 4$  輝度ブロックにおける予測サンプル  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  である。

現マクロブロックの中でインデックス  $\text{luma4x4BlkIdx}$  をもつ  $4 \times 4$  輝度ブロックの左上のサンプル位置は、 $\text{luma4x4BlkIdx}$  を入力として 6.4.3 小節の逆  $4 \times 4$  輝度ブロック走査処理を起動することで導かれ、出力は  $(xO,yO)$  に割り当てられる。

デブロックフィルタ処理前の構築輝度サンプルである 13 個の隣接サンプル  $p[x,y]$ 、 $x=-1,y=-1..3$  と  $x=0..7,y=-1$  は次の通り導出される。

- 輝度位置  $(xN,yN)$  は以下で規定される。

$$xN = xO + x \quad (8-43)$$

$$yN = yO + y \quad (8-44)$$

- 6.4.11 小節の隣接位置の導出処理が、輝度位置に対して  $(xN,yN)$  を入力として、 $\text{mbAddrN}$  と  $(xW,yW)$  を出力として起動される。

- 各々のサンプル  $p[x,y]$ 、 $x=-1,y=-1..3$  と  $x=0..7,y=-1$  は次の通り導出される。

- もし以下の条件のいずれかが真ならば、サンプル  $p[x,y]$  は “Intra\_4x4 予測で利用不可” としてマークされる。

- $\text{mbAddrN}$  が利用不可である。
- マクロブロック  $\text{mbAddrN}$  が INTER 予測モードで符号化され、かつ  $\text{constrained\_intra\_pred\_flag}$  が 1 である。
- マクロブロック  $\text{mbAddrN}$  が SI に等しい  $\text{mb\_type}$  をもち、かつ  $\text{constrained\_intra\_pred\_flag}$  が 1、かつ現マクロブロックは SI に等しい  $\text{mb\_type}$  をもたない。
- $x$  が 3 より大きく、かつ  $\text{luma4x4BlkIdx}$  が 3 あるいは 11 に等しい。

- それ以外、サンプル  $p[x,y]$  は “Intra\_4x4 予測で利用可能” としてマークされ、サンプル  $p[x,y]$  の値は次のように導出される。

- マクロブロック  $\text{mbAddrN}$  の左上輝度サンプル位置が、6.4.1 小節の逆マクロブロック走査処理を  $\text{mbAddrN}$  を入力として起動して導出され、出力は  $(xM,yM)$  に割り当てられる。
- 変数  $\text{MbaffFrameFlag}$  およびマクロブロック  $\text{mbAddrN}$  に依存して、サンプル値  $p[x,y]$  は次のように導出される。

- もし  $\text{MbaffFrameFlag}$  が 1 に等しく、かつマクロブロック  $\text{mbAddrN}$  がフィールドマクロブロックならば、

$$p[x,y] = cS_L[xM + xW, yM + 2 * yW] \quad (8-45)$$

- それ以外 (MbaffFrameFlag が 0 に等しい、あるいはマクロブロック mbAddrN がフレームマクロブロック)、

$$p[x,y]=cS_L[xM+xW,yM+yW] \quad (8-46)$$

サンプル  $p[x,-1]$ 、 $x=4..7$  が “Intra\_4x4 予測で利用不可” としてマークされており、かつサンプル  $p[3,-1]$  が “Intra\_4x4 予測で利用可能” としてマークされている時、 $p[3,-1]$  のサンプル値は  $p[x,-1]$ 、 $x=4..7$  のサンプル値の代わりとして使われ、サンプル  $p[x,-1]$ 、 $x=4..7$  は “Intra\_4x4 予測で利用可能” としてマークされる。

記 各ブロックは次のブロックの復号前に、ピクチャ配列にまで構築されると仮定される。

Intra4x4PredMode[luma4x4BlkIdx]に従って、8.3.1.2.1 小小小節から 8.3.1.2.9 小小小節で規定される Intra\_4x4 予測モードの 1 つが起動される。

### 8.3.1.2.1 Intra\_4x4\_Vertical予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 0 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..3$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $pred4x4_L[x,y]$ 、 $x,y=0..3$  は以下で導出される。

$$pred4x4_L[x,y]=p[x,-1]、x,y=0..3 \quad (8-47)$$

### 8.3.1.2.2 Intra\_4x4\_Horizontal予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 1 の時に起動される。

このモードは、サンプル  $p[-1,y]$ 、 $y=0..3$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $pred4x4_L[x,y]$ 、 $x,y=0..3$  は以下で導出される。

$$pred4x4_L[x,y]=p[-1,y]、x,y=0..3 \quad (8-48)$$

### 8.3.1.2.3 Intra\_4x4\_DC予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 2 の時に起動される。

予測サンプル値  $pred4x4_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし  $p[x,-1]$ 、 $x=0..3$  と  $p[-1,y]$ 、 $y=0..3$  なる全てのサンプルが “Intra\_4x4 予測で利用可能” としてマークされているならば、予測サンプル値  $pred4x4_L[x,y]$ 、 $x,y=0..3$  は以下で導出される。

$$pred4x4_L[x,y]=(p[0,-1]+p[1,-1]+p[2,-1]+p[3,-1]+p[-1,0]+p[-1,1]+p[-1,2]+p[-1,3]+4)>>3 \quad (8-49)$$

- それ以外、もしいくつかのサンプル  $p[x,-1]$ 、 $x=0..3$  が “Intra\_4x4 予測で利用不可” としてマークされていて、かつ全てのサンプル  $p[-1,y]$ 、 $y=0..3$  が “Intra\_4x4 予測で利用可能” としてマークされているならば、予測サンプル値  $pred4x4_L[x,y]$ 、 $x,y=0..3$  は以下で導出される。

$$pred4x4_L[x,y]=(p[-1,0]+p[-1,1]+p[-1,2]+p[-1,3]+2)>>2 \quad (8-50)$$

- それ以外、もしいくつかのサンプル  $p[-1,y]$ 、 $y=0..3$  が “Intra\_4x4 予測で利用不可” としてマークされていて、かつ全てのサンプル  $p[x,-1]$ 、 $x=0..3$  が “Intra\_4x4 予測で利用可能” としてマークされているならば、予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は以下で導出される。

$$\text{pred4x4}_L[x,y]=(p[0,-1]+p[1,-1]+p[2,-1]+p[3,-1]+2)>>2 \quad (8-51)$$

- それ以外(いくつかのサンプル  $p[x,-1]$ 、 $x=0..3$  といくつかのサンプル  $p[-1,y]$ 、 $y=0..3$  が “Intra\_4x4 予測で利用不可” としてマークされている)、予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は以下で導出される。

$$\text{pred4x4}_L[x,y]=(1<<(\text{BitDepth}_Y-1)) \quad (8-52)$$

記 4×4輝度ブロックは常時このモードを使用して予測できる。

#### 8.3.1.2.4 Intra\_4x4\_Diagonal\_Down\_Left予測モードの規定

この Intra\_4x4 予測モードは、 $\text{Intra4x4PredMode}[\text{luma4x4BlkIdx}]$ が3の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..7$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし  $x$  が3かつ  $y$  が3ならば、

$$\text{pred4x4}_L[x,y]=(p[6,-1]+3*p[7,-1]+2)>>2 \quad (8-53)$$

- それ以外( $x$  が3でないか、または  $y$  が3でない)、

$$\text{pred4x4}_L[x,y]=(p[x+y,-1]+2*p[x+y+1,-1]+p[x+y+2,-1]+2)>>2 \quad (8-54)$$

#### 8.3.1.2.5 Intra\_4x4\_Diagonal\_Down\_Right予測モードの規定

この Intra\_4x4 予測モードは、 $\text{Intra4x4PredMode}[\text{luma4x4BlkIdx}]$ が4の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..3$  と  $p[-1,y]$ 、 $y=-1..3$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし  $x$  が  $y$  より大きいならば、

$$\text{pred4x4}_L[x,y]=(p[x-y-2,-1]+2*p[x-y-1,-1]+p[x-y,-1]+2)>>2 \quad (8-55)$$

- それ以外、もし  $x$  が  $y$  より小さいならば、

$$\text{pred4x4}_L[x,y]=(p[-1,y-x-2]+2*p[-1,y-x-1]+p[-1,y-x]+2)>>2 \quad (8-56)$$

- それ以外( $x$  と  $y$  が等しい)、

$$\text{pred4x4}_L[x,y]=(p[0,-1]+2*p[-1,-1]+p[-1,0]+2)>>2 \quad (8-57)$$

### 8.3.1.2.6 Intra\_4x4\_Vertical\_Right予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 5 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..3$  と  $p[-1,y]$ 、 $y=-1..3$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

変数 zVR を  $2*x-y$  に等しく設定する。

予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし zVR が 0、2、4 または 6 ならば、

$$\text{pred4x4}_L[x,y]=(p[x-(y>>1)-1,-1]+p[x-(y>>1),-1]+1)>>1 \quad (8-58)$$

- それ以外、もし zVR が 1、3 または 5 ならば、

$$\text{pred4x4}_L[x,y]=(p[x-(y>>1)-2,-1]+2*p[x-(y>>1)-1,-1]+p[x-(y>>1),-1]+2)>>2 \quad (8-59)$$

- それ以外、もし zVR が -1 ならば、

$$\text{pred4x4}_L[x,y]=(p[-1,0]+2*p[-1,-1]+p[0,-1]+2)>>2 \quad (8-60)$$

- それ以外(zVR が -2 または -3)、

$$\text{pred4x4}_L[x,y]=(p[-1,y-1]+2*p[-1,y-2]+p[-1,y-3]+2)>>2 \quad (8-61)$$

### 8.3.1.2.7 Intra\_4x4\_Horizontal\_Down予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 6 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..3$  と  $p[-1,y]$ 、 $y=-1..3$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

変数 zHD を  $2*y-x$  に等しく設定する。

予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし zHD が 0、2、4 または 6 ならば、

$$\text{pred4x4}_L[x,y]=(p[-1,y-(x>>1)-1]+p[-1,y-(x>>1)+1])>>1 \quad (8-62)$$

- それ以外、もし zHD が 1、3 または 5 ならば、

$$\text{pred4x4}_L[x,y]=(p[-1,y-(x>>1)-2]+2*p[-1,y-(x>>1)-1]+p[-1,y-(x>>1)+2])>>2 \quad (8-63)$$

- それ以外、もし zHD が -1 ならば、

$$\text{pred4x4}_L[x,y]=(p[-1,0]+2*p[-1,-1]+p[0,-1]+2)>>2 \quad (8-64)$$

- それ以外(zHD が -2 または -3)、

$$\text{pred4x4}_L[x,y]=(p[x-1,-1]+2*p[x-2,-1]+p[x-3,-1]+2)>>2 \quad (8-65)$$

### 8.3.1.2.8 Intra\_4x4\_Vertical\_Left予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 7 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..7$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし  $y$  が 0 または 2 ならば、

$$\text{pred4x4}_L[x,y]=(\text{p}[x+(y>>1),-1]+\text{p}[x+(y>>1)+1,-1]+1)>>1 \quad (8-66)$$

- それ以外( $y$  が 1 または 3)、

$$\text{pred4x4}_L[x,y]=(\text{p}[x+(y>>1),-1]+2*\text{p}[x+(y>>1)+1,-1]+\text{p}[x+(y>>1)+2,-1]+2)>>2 \quad (8-67)$$

### 8.3.1.2.9 Intra\_4x4\_Horizontal\_Up予測モードの規定

この Intra\_4x4 予測モードは、Intra4x4PredMode[luma4x4BlkIdx]が 8 の時に起動される。

このモードは、サンプル  $p[-1,y]$ 、 $y=0..3$  が “Intra\_4x4 予測で利用可能” としてマークされている時のみ、使用されなければならない。

変数 zHU を  $x+2*y$  に等しく設定する。

予測サンプル値  $\text{pred4x4}_L[x,y]$ 、 $x,y=0..3$  は次の通り導出される。

- もし zHU が 0、2 または 4 ならば、

$$\text{pred4x4}_L[x,y]=(\text{p}[-1,y+(x>>1)]+\text{p}[-1,y+(x>>1)+1]+1)>>1 \quad (8-68)$$

- それ以外、もし zHU が 1 または 3 ならば、

$$\text{pred4x4}_L[x,y]=(\text{p}[-1,y+(x>>1)]+2*\text{p}[-1,y+(x>>1)+1]+\text{p}[-1,y+(x>>1)+2]+2)>>2 \quad (8-69)$$

- それ以外、もし zHU が 5 ならば、

$$\text{pred4x4}_L[x,y]=(\text{p}[-1,2]+3*\text{p}[-1,3]+2)>>2 \quad (8-70)$$

- それ以外(zHU が 5 より大きい)、

$$\text{pred4x4}_L[x,y]=\text{p}[-1,3] \quad (8-71)$$

## 8.3.2 輝度サンプルのIntra\_8x8 予測処理

この処理はマクロブロック予測モードが Intra\_8x8 の時に起動される。

この処理への入力、隣接マクロブロックあるいはマクロブロックペアからの (利用可能であれば) Intra4x4PredMode あるいは、(利用可能であれば) Intra8x8PredMode の値である。

この処理の出力は、マクロブロックの予測サンプルである  $16 \times 16$  輝度配列  $\text{pred}_L$  の一部分である  $8 \times 8$  輝度サンプル配列である。

マクロブロックの輝度成分は4個の8×8輝度サンプルのブロックから構成される。これらのブロックは、6.4.5小節に規定された逆8×8輝度ブロック走査処理を使って逆走査される。

$\text{luma8x8BlkIdx}=0..3$ であるマクロブロック輝度成分の8×8輝度ブロック全てに対して、8.3.2.1小節に規定される  $\text{Intra8x8PredMode}$  の導出処理は、 $\text{luma8x8BlkIdx}$  と、さらに隣接マクロブロックで（復号順序で）先に導かれた  $\text{Intra4x4PredMode}$  および  $\text{Intra8x8PredMode}$  を入力とし、変数  $\text{Intra8x8PredMode}[\text{luma8x8BlkIdx}]$  を出力として起動される。

$\text{luma8x8BlkIdx}=0..3$  を使って索引される8×8サンプルの輝度ブロック各々に対して、次が適用される。

- 8.3.2.2小節の  $\text{Intra\_8x8}$  サンプル予測処理が、 $\text{luma8x8BlkIdx}$  と、隣接輝度ブロックからのデブロックフィルタ処理前の構築サンプルを含む配列  $S'_L$  を入力として起動され、出力は  $\text{Intra\_8x8}$  輝度予測サンプル  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  である。
- 現マクロブロック内でインデックス  $\text{luma8x8BlkIdx}$  をもつ8×8輝度ブロックの左上のサンプルの位置が、 $\text{luma8x8BlkIdx}$  を入力として、6.4.5小節の逆8×8輝度ブロック走査処理を起動することで導出され、出力は  $(xO,yO)$  に割り当てられ、 $x,y=0..7$  で

$$\text{pred}_L[xO+x,yO+y]=\text{pred8x8}_L[x,y] \quad (8-72)$$

である

- 8.5節の変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理が、 $\text{pred}_L$  と  $\text{luma8x8BlkIdx}$  を入力とし、現8×8輝度ブロック  $S'_L$  に対する構築サンプルを出力として起動される。

### 8.3.2.1 $\text{Intra8x8PredMode}$ の導出処理

この処理への入力は、8×8輝度ブロックのインデックス  $\text{luma8x8BlkIdx}$  と、隣接マクロブロックで（復号順序で）先に導かれた変数配列（利用可能であれば） $\text{Intra4x4PredMode}$  と（利用可能であれば） $\text{Intra8x8PredMode}$  である。

この処理の出力は、変数  $\text{Intra8x8PredMode}[\text{luma8x8BlkIdx}]$  である。

表 8.3/JT-H264 は、 $\text{Intra8x8PredMode}[\text{luma8x8BlkIdx}]$  の値と、各々に関連するニーモニック名を規定する。

表8-3/JT-H264 Intra8x8PredMode[luma8x8BlkIdx]と関連する名前の規定  
(ITU-T H.264)

Intra8x8PredMode[ luma8x8BlkIdx ]	Intra8x8PredMode[ luma8x8BlkIdx ]の名前
0	Intra_8x8_Vertical (予測モード)
1	Intra_8x8_Horizontal (予測モード)
2	Intra_8x8_DC (予測モード)
3	Intra_8x8_Diagonal_Down_Left (予測モード)
4	Intra_8x8_Diagonal_Down_Right (予測モード)
5	Intra_8x8_Vertical_Right (予測モード)
6	Intra_8x8_Horizontal_Down (予測モード)
7	Intra_8x8_Vertical_Left (予測モード)
8	Intra_8x8_Horizontal_Up (予測モード)

Intra8x8PredMode[luma8x8BlkIdx]は次の通り導出される。

- 6.4.10.2 小節で規定された処理が、luma8x8BlkIdx を入力として与えられて起動され、出力は mbAddrA、luma8x8BlkIdxA、mbAddrB、と luma8x8BlkIdxB に割り当てられる
- 変数 dcPredModePredictedFlag は次の通り導出される。
  - もし以下の条件のいずれかが真ならば、dcPredModePredictedFlag は 1 に設定される。
    - アドレス mbAddrA のマクロブロックが利用不可である。
    - アドレス mbAddrB のマクロブロックが利用不可である。
    - アドレス mbAddrA のマクロブロックが利用可能、かつ INTER 予測モードで符号化され、かつ constrained\_intra\_pred\_flag が 1 である。
    - アドレス mbAddrB のマクロブロックが利用可能、かつ INTER 予測モードで符号化され、かつ constrained\_intra\_pred\_flag が 1 である。
  - それ以外、dcPredModePredictedFlag は 0 に設定される。
- N が A か B のどちらかに置き換えられるものとする、変数 intraMxMPredModeN は次の通り導出される。
  - もし dcPredModePredictedFlag が 1、あるいは(アドレス mbAddrN のマクロブロックが Intra\_4x4 マクロブロック予測モードで符号化されていない、かつアドレス mbAddrN のマクロブロックが Intra\_8x8 マクロブロック予測モードで符号化されていない)ならば、intraMxMPredModeN は 2(Intra\_8x8\_DC 予測モード)に設定される。
  - それ以外(dcPredModePredictedFlag が 0、かつ(アドレス mbAddrN のマクロブロックが Intra\_4x4 マクロブロック予測モードで符号化されている、あるいはアドレス mbAddrN のマクロブロックが Intra\_8x8 マクロブロック予測モードで符号化されている))、次が適用される。

- もしアドレス mbAddrN のマクロブロックが Intra\_8x8 マクロブロック予測モードで符号化されているならば、intraMxMPredModeN は Intra8x8PredMode[luma8x8BlkIdxN] に設定される。ここで Intra8x8PredMode はマクロブロック mbAddrN に割り当てられた配列変数である。
- それ以外(アドレス mbAddrN のマクロブロックが Intra\_4x4 マクロブロック予測モードで符号化されているならば)、intraMxMPredModeN は下記の手続きによって導出される。ここで Intra4x4PredMode はマクロブロック mbAddrN に割り当てられた配列変数である。

$$\text{intraMxMPredModeN} = \text{Intra4x4PredMode}[\text{luma8x8BlkIdxN} * 4 + n] \quad (8-73)$$

ここで変数nは以下のように導出される。

- もし N が A と等しければ、変数 MbaffFrameFlag、変数 luma8x8BlkIdx、現マクロブロック、マクロブロック mbAddrN に従って、次が適用される。
  - もし、MbaffFrameFlag が 1、現マクロブロックがフレーム符号化マクロブロック、マクロブロック mbAddrN がフィールド符号化マクロブロック、かつ luma8x8BlkIdx が 2 ならば、n は 3 に設定される。
  - それ以外(MbaffFrameFlag が 0、現マクロブロックがフィールド符号化マクロブロック、マクロブロック mbAddrN がフレーム符号化マクロブロック、あるいは luma8x8BlkIdx が 2 でないならば)、n は 1 に設定される。
- それ以外(N が B と等しければ)、n は 2 に設定される。
- 最後に、intraMxMPredModeA と intraMxMPredModeB が与えられることにより、変数 Intra8x8PredMode[luma8x8BlkIdx] は下記の手続きを適用して導出される。

```

predIntra8x8PredMode = Min(intraMxMPredModeA, intraMxMPredModeB)
if(prev_intra8x8_pred_mode_flag[luma8x8BlkIdx])
  Intra8x8PredMode[luma8x8BlkIdx] = predIntra8x8PredMode
else
  if(rem_intra8x8_pred_mode[luma8x8BlkIdx] < predIntra8x8PredMode)
    Intra8x8PredMode[luma8x8BlkIdx] = rem_intra8x8_pred_mode[luma8x8BlkIdx]
  else
    Intra8x8PredMode[luma8x8BlkIdx] = rem_intra8x8_pred_mode[luma8x8BlkIdx] + 1
  
```

(8-74)

### 8.3.2.2 Intra\_8x8 サンプル予測

この処理は、Intra\_8x8 に等しい予測モードを持つマクロブロックの各 8×8 輝度ブロックに対して起動され、さらに、各 8×8 輝度ブロックの変換復号処理とデブロック前のピクチャ構築処理が後に続く。

この処理への入力は、

- 8×8 輝度ブロックのインデックス luma8x8BlkIdx
- 隣接マクロブロックのデブロックフィルタ処理前の構築された輝度サンプルを含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 cS<sub>L</sub>

である。

この処理の出力は、インデックス `luma8x8BlkIdx` を持つ  $8 \times 8$  輝度ブロックにおける予測サンプル `pred8x8L[x,y]`、 $x,y=0..7$  である。

現マクロブロックの中でインデックス `luma8x8BlkIdx` を持つ  $8 \times 8$  輝度ブロックの左上のサンプル位置は、`luma8x8BlkIdx` を入力として 6.4.5 小節の逆  $8 \times 8$  輝度ブロック走査処理を起動することで導かれ、出力は  $(xO,yO)$  に割り当てられる。

デブロックフィルタ処理前の構築輝度サンプルである 25 個の隣接サンプル `p[x,y]`、 $x=-1,y=-1..7$  および  $x=0..15,y=-1$  は次の通り導出される。

- 輝度位置  $(xN,yN)$  は以下で規定される。

$$xN = xO + x \quad (8-75)$$

$$yN = yO + y \quad (8-76)$$

- 6.4.11 小節の隣接位置の導出処理が、輝度位置に対して  $(xN,yN)$  を入力として、`mbAddrN` と  $(xW,yW)$  を出力として起動される。

- 各サンプル `p[x,y]`、 $x=-1,y=-1..7$  および  $x=0..15,y=-1$  の値は、次の通り導出される。

- もし以下の条件のいずれかが真ならば、サンプル `p[x,y]` は、“Intra\_8x8 予測で利用不可”としてマークされる。

- `mbAddrN` が利用不可である。
- マクロブロック `mbAddrN` が INTER 予測モードで符号化され、かつ `constrained_intra_pred_flag` が 1 である。

- それ以外、サンプル `p[x,y]` は “Intra\_8x8 予測で利用可能”としてマークされ、サンプル値 `p[x,y]` のは次のように導出される。

- マクロブロック `mbAddrN` の左上輝度サンプル位置が、6.4.1 小節の逆マクロブロック走査処理を `mbAddrN` を入力として起動して導出され、出力は  $(xM,yM)$  に割り当てられる。

- 変数 `MbaffFrameFlag` およびマクロブロック `mbAddrN` に依存して、サンプル値 `p[x,y]` は次のように導出される。

- もし `MbaffFrameFlag` が 1 に等しく、かつマクロブロック `mbAddrN` がフィールドマクロブロックならば、

$$p[x,y] = cS_L[xM+xW,yM+2*yW] \quad (8-77)$$

- それ以外 (`MbaffFrameFlag` が 0 に等しい、あるいはマクロブロック `mbAddrN` がフレームマクロブロック)、

$$p[x,y] = cS_L[xM+xW,yM+yW] \quad (8-78)$$

サンプル `p[x,-1]`、 $x=8..15$  が “Intra\_8x8 予測で利用不可”としてマークされており、かつサンプル `p[7,-1]` が “Intra\_8x8 予測で利用可能”としてマークされている時、`p[7,-1]` のサンプル値は、`p[x,-1]`、 $x=8..15$  のサンプル値の代わりとして使われ、サンプル `p[x,-1]`、 $x=8..15$  は、“Intra\_8x8 予測で利用可能”としてマークされ

る。

記 各ブロックは次のブロックの復号前に、ピクチャ配列にまで構築されると仮定される。

8.3.2.2.1 小小小節で示される Intra\_8x8 サンプル予測に対する参照サンプルフィルタ処理は、(利用可能であれば)サンプル  $p[x,y]$ 、 $x=-1,y=-1..7$  および  $x=0..15,y=-1$  を入力とし、サンプル  $p'[x,y]$ 、 $x=-1,y=-1..7$  および  $x=0..15,y=-1$  を出力として起動される。

Intra8x8PredMode[luma8x8BlkIdx]に従って、8.3.2.2.2 小小小節から 8.3.2.2.10 小小小節で規定される Intra\_8x8 予測モードの一つが起動される。

#### 8.3.2.2.1 Intra\_8x8 サンプル予測に対する参照サンプルフィルタ処理

この処理への入力は、Intra\_8x8 サンプル予測に対する(利用可能であれば)参照サンプル  $p[x,y]$ 、 $x=-1,y=-1..7$  および  $x=0..15,y=-1$  である。

この処理の出力は、Intra\_8x8 サンプル予測に対するフィルタ処理された参照サンプル  $p'[x,y]$ 、 $x=-1,y=-1..7$  および  $x=0..15,y=-1$  である。

$p[x,-1]$ 、 $x=0..7$  の全てのサンプルが “Intra\_8x8 予測で利用可能” としてマークされている時、次が適用される。

-  $p'[0,-1]$ の値は、次の通り導出される。

- もし  $p[-1,-1]$ が “Intra\_8x8 予測で利用可能” としてマークされているならば、 $p'[0,-1]$ は以下のよう  
に導出される。

$$p'[0,-1]=(p[-1,-1]+2*p[0,-1]+p[1,-1]+2)>>2 \quad (8-79)$$

- それ以外( $p[-1,-1]$ が “Intra\_8x8 予測で利用不可” としてマークされているならば)、 $p'[0,-1]$ は以下の  
ように導出される。

$$p'[0,-1]=(3*p[0,-1]+p[1,-1]+2)>>2 \quad (8-80)$$

-  $p'[x,-1]$ 、 $x=1..7$  の値は、以下のように導出される。

$$p'[x,-1]=(p[x-1,-1]+2*p[x,-1]+p[x+1,-1]+2)>>2 \quad (8-81)$$

$p[x,-1]$ 、 $x=7..15$  の全てのサンプルが “Intra\_8x8 予測で利用可能” としてマークされている時、次が適用される。

-  $p'[x,-1]$ 、 $x=8..14$  の値は、以下のように導出される。

$$p'[x,-1]=(p[x-1,-1]+2*p[x,-1]+p[x+1,-1]+2)>>2 \quad (8-82)$$

-  $p'[15,-1]$ の値は、以下のように導出される。

$$p'[15,-1]=(p[14,-1]+3*p[15,-1]+2)>>2 \quad (8-83)$$

サンプル  $p[-1,-1]$ が “Intra\_8x8 予測で利用可能” としてマークされている時、 $p'[-1,-1]$ の値は、次の通り導出される。

- もしサンプル  $p[0,-1]$  が “Intra\_8x8 予測で利用不可” としてマークされているか、またはサンプル  $p[-1,0]$  が “Intra\_8x8 予測で利用不可” としてマークされているならば、次が適用される。

- もしサンプル  $p[0,-1]$  が “Intra\_8x8 予測で利用可能” としてマークされているならば、 $p'[-1,-1]$  は以下のように導出される。

$$p'[-1,-1]=(3*p[-1,-1]+p[0,-1]+2)>>2 \quad (8-84)$$

- それ以外(サンプル  $p[0,-1]$  が “Intra\_8x8 予測で利用不可” としてマークされており、かつサンプル  $p[-1,0]$  が “Intra\_8x8 予測で利用可能” としてマークされているならば)、 $p'[-1,-1]$  の値は以下のように導出される。

$$p'[-1,-1]=(3*p[-1,-1]+p[-1,0]+2)>>2 \quad (8-85)$$

- それ以外(サンプル  $p[0,-1]$  が “Intra\_8x8 予測で利用可能” としてマークされており、かつサンプル  $p[-1,0]$  が “Intra\_8x8 予測で利用可能” としてマークされているならば)、 $p'[-1,-1]$  の値は以下のように導出される。

$$p'[-1,-1]=(p[0,-1]+2*p[-1,-1]+p[-1,0]+2)>>2 \quad (8-86)$$

全てのサンプル  $p[-1,y]$ 、 $y=0..7$  が、“Intra\_8x8 予測で利用可能”としてマークされている場合は次が適用される。

- $p'[-1,0]$  の値は次の通り導出される。

- もし、 $p[-1,-1]$  が “Intra\_8x8 予測で利用可能” としてマークされているならば、 $p'[-1,0]$  は以下のように導出される。

$$p'[-1,0]=(p[-1,-1]+2*p[-1,0]+p[-1,1]+2)>>2 \quad (8-87)$$

- それ以外、(もし  $p[-1,-1]$  が “Intra\_8x8 予測で利用不可” としてマークされているならば)、 $p'[-1,0]$  は次式により導出される。

$$p'[-1,0]=(3*p[-1,0]+p[-1,1]+2)>>2 \quad (8-88)$$

- $p'[-1,y]$ 、 $y=1..6$  の値は以下により導出される。

$$p'[-1,y]=(p[-1,y-1]+2*p[-1,y]+p[-1,y+1]+2)>>2 \quad (8-89)$$

- $p'[-1,7]$  の値は以下により導出される。

$$p'[-1,7]=(p[-1,6]+3*p[-1,7]+2)>>2 \quad (8-90)$$

### 8.3.2.2.2 Intra\_8x8\_Vertical 予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx] が 0 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..7$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred8x8}_l[x,y]$ 、 $x,y=0..7$  は以下で導出される。

$$\text{pred8x8}_L[x,y]=p'[x,-1], x,y=0..7 \quad (8-91)$$

### 8.3.2.2.3 Intra\_8x8\_Horizontal予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が 1 の時に起動される。

このモードは、サンプル  $p[-1,y]$ 、 $y=0..7$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は以下で導出される。

$$\text{pred8x8}_L[x,y]=p'[-1,y], x,y=0..7 \quad (8-92)$$

### 8.3.2.2.4 Intra\_8x8\_DC予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が 2 の時に起動される。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし  $p[x,-1]$ 、 $x=0..7$  と  $p[-1,y]$ 、 $y=0..7$  なる全てのサンプルが “Intra\_8x8 予測で利用可能” としてマークされているならば、予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は以下で導出される。

$$\text{pred8x8}_L[x,y]=\left(\sum_{x'=0}^7 p'[x',-1]+\sum_{y'=0}^7 p'[-1,y']\right) \gg 4 \quad (8-93)$$

- それ以外、もしいくつかのサンプル  $p[x,-1]$ 、 $x=0..7$  が “Intra\_8x8 予測で利用不可” としてマークされていて、かつ全てのサンプル  $p[-1,y]$ 、 $y=0..7$  が “Intra\_8x8 予測で利用可能” としてマークされているならば、予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は以下で導出される。

$$\text{pred8x8}_L[x,y]=\left(\sum_{y'=0}^7 p'[-1,y']\right) \gg 3 \quad (8-94)$$

- それ以外、もしいくつかのサンプル  $p[-1,y]$ 、 $y=0..7$  が “Intra\_8x8 予測で利用不可” としてマークされていて、かつ全てのサンプル  $p[x,-1]$ 、 $x=0..7$  が “Intra\_8x8 予測で利用可能” としてマークされているならば、予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は以下で導出される。

$$\text{pred8x8}_L[x,y]=\left(\sum_{x'=0}^7 p'[x',-1]\right) \gg 3 \quad (8-95)$$

- それ以外(いくつかのサンプル  $p[x,-1]$ 、 $x=0..7$  といくつかのサンプル  $p[-1,y]$ 、 $y=0..7$  が “Intra\_8x8 予測で利用不可” としてマークされている)、予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は以下で導出される。

$$\text{pred8x8}_L[x,y]=(1 \ll (\text{BitDepth}_Y - 1)) \quad (8-96)$$

記 8×8輝度ブロックは常時このモードを使用して予測できる。

### 8.3.2.2.5 Intra\_8x8\_Diagonal\_Down\_Left予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が 3 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..15$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし  $x$  が  $7$  かつ  $y$  が  $7$  ならば、

$$\text{pred8x8}_L[x,y]=(p'[14,-1]+3*p'[15,-1]+2)>>2 \quad (8-97)$$

- それ以外( $x$  が  $7$  でないか、または  $y$  が  $7$  でない)、

$$\text{pred8x8}_L[x,y]=(p'[x+y,-1]+2*p'[x+y+1,-1]+p'[x+y+2,-1]+2)>>2 \quad (8-98)$$

### 8.3.2.2.6 Intra\_8x8\_Diagonal\_Down\_Right予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が  $4$  の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..7$  と  $p[-1,y]$ 、 $y=-1..7$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし  $x$  が  $y$  より大きいならば、

$$\text{pred8x8}_L[x,y]=(p'[x-y-2,-1]+2*p'[x-y-1,-1]+p'[x-y,-1]+2)>>2 \quad (8-99)$$

- それ以外、もし  $x$  が  $y$  より小さいならば、

$$\text{pred8x8}_L[x,y]=(p'[-1,y-x-2]+2*p'[-1,y-x-1]+p'[-1,y-x]+2)>>2 \quad (8-100)$$

- それ以外( $x$  と  $y$  が等しい)、

$$\text{pred8x8}_L[x,y]=(p'[0,-1]+2*p'[-1,-1]+p'[-1,0]+2)>>2 \quad (8-101)$$

### 8.3.2.2.7 Intra\_8x8\_Vertical\_Right予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が  $5$  の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..7$  と  $p[-1,y]$ 、 $y=-1..7$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

変数  $zVR$  を  $2*x-y$  に等しく設定する。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし  $zVR$  が  $0, 2, 4, 6, 8, 10, 12$  または  $14$  ならば、

$$\text{pred8x8}_L[x,y]=(p'[x-(y>>1)-1,-1]+p'[x-(y>>1)-1,+1])>>1 \quad (8-102)$$

- それ以外、もし  $zVR$  が  $1, 3, 5, 7, 9, 11$  または  $13$  ならば、

$$\text{pred8x8}_L[x,y]=(p'[x-(y>>1)-2,-1]+2*p'[x-(y>>1)-1,-1]+p'[x-(y>>1)-1,+2])>>2 \quad (8-103)$$

- それ以外、もし  $zVR$  が  $-1$  ならば、

$$\text{pred8x8}_L[x,y]=(p'[-1,0]+2*p'[-1,-1]+p'[0,-1]+2)>>2 \quad (8-104)$$

- それ以外(zVR が-2、-3、-4、-5、-6 または-7)、

$$\text{pred8x8}_L[x,y]=(p'[-1,y-2*x-1]+2*p'[-1,y-2*x-2]+p'[-1,y-2*x-3]+2)>>2 \quad (8-105)$$

### 8.3.2.2.8 Intra\_8x8\_Horizontal\_Down予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が 6 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..7$  と  $p[-1,y]$ 、 $y=-1..7$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

変数 zHD を  $2*y-x$  に等しく設定する。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし zHD が 0、2、4、6、8、10、12 または 14 ならば、

$$\text{pred8x8}_L[x,y]=(p'[-1,y-(x>>1)-1]+p'[-1,y-(x>>1)+1])>>1 \quad (8-106)$$

- それ以外、もし zHD が 1、3、5、7、9、11 または 13 ならば、

$$\text{pred8x8}_L[x,y]=(p'[-1,y-(x>>1)-2]+2*p'[-1,y-(x>>1)-1]+p'[-1,y-(x>>1)+2])>>2 \quad (8-107)$$

- それ以外、もし zHD が-1 ならば、

$$\text{pred8x8}_L[x,y]=(p'[-1,0]+2*p'[-1,-1]+p'[0,-1]+2)>>2 \quad (8-108)$$

- それ以外(zHD が-2、-3、-4、-5、-6、-7)、

$$\text{pred8x8}_L[x,y]=(p'[x-2*y-1,-1]+2*p'[x-2*y-2,-1]+p'[x-2*y-3,-1]+2)>>2 \quad (8-109)$$

### 8.3.2.2.9 Intra\_8x8\_Vertical\_Left予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が 7 の時に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..15$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、使用されなければならない。

予測サンプル値  $\text{pred8x8}_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし y が 0、2、4 または 6 ならば、

$$\text{pred8x8}_L[x,y]=(p'[x+(y>>1),-1]+p'[x+(y>>1)+1,-1]+1)>>1 \quad (8-110)$$

- それ以外(y が 1、3、5、7)、

$$\text{pred8x8}_L[x,y]=(p'[x+(y>>1),-1]+2*p'[x+(y>>1)+1,-1]+p'[x+(y>>1)+2,-1]+2)>>2 \quad (8-111)$$

### 8.3.2.2.10 Intra\_8x8\_Horizontal\_Up予測モードの規定

この Intra\_8x8 予測モードは、Intra8x8PredMode[luma8x8BlkIdx]が 8 の時に起動される。

このモードは、サンプル  $p[-1,y]$ 、 $y=0..7$  が “Intra\_8x8 予測で利用可能” としてマークされている時のみ、

使用されなければならない。

変数  $zHU$  を  $x+2*y$  に等しく設定する。

予測サンプル値  $pred8x8_L[x,y]$ 、 $x,y=0..7$  は次の通り導出される。

- もし  $zHU$  が 0、2、4、6、8、10 または 12 ならば、

$$pred8x8_L[x,y]=(p'[-1,y+(x>>1)]+p'[-1,y+(x>>1)+1]+1)>>1 \quad (8-112)$$

- それ以外、もし  $zHU$  が 1、3、5、7、9 または 11 ならば、

$$pred8x8_L[x,y]=(p'[-1,y+(x>>1)]+2*p'[-1,y+(x>>1)+1]+p'[-1,y+(x>>1)+2]+2)>>2 \quad (8-113)$$

- それ以外、もし  $zHU$  が 13 ならば、

$$pred8x8_L[x,y]=(p'[-1,6]+3*p'[-1,7]+2)>>2 \quad (8-114)$$

- それ以外( $zHU$  が 13 より大きい)、

$$pred8x8_L[x,y]=p'[-1,7] \quad (8-115)$$

### 8.3.3 輝度サンプルのIntra\_16x16 予測処理

この処理は、マクロブロック予測モードが Intra\_16x16 の時に起動される。それは、現マクロブロックに対する INTRA 予測輝度サンプルがどのように導出されるかを規定する。

この処理の入力は、隣接マクロブロックのデブロックフィルタ処理前の構築された輝度サンプルを含む  $(PicWidthInSamples_L) \times (PicHeightInSamples_L)$  配列  $cS_L$  である。

この処理の出力は、現マクロブロックに対する INTRA 予測輝度サンプル  $pred_L[x,y]$  である。

デブロックフィルタ処理前の構築輝度サンプルである 33 個の隣接サンプル  $p[x,y]$ 、 $x=-1,y=-1..15$ 、そして  $x=0..15,y=-1$  は、次の通り導出される。

- 6.4.11 小節における隣接位置の導出処理が、 $(x,y)$  が  $(xN,yN)$  に入力として割り当てられ、出力として  $mbAddrN$  と  $(xW,yW)$  を用いて輝度位置に対して起動される。
- 各々のサンプル  $p[x,y]$ 、 $x=-1,y=-1..15$ 、そして  $x=0..15,y=-1$  は、次の通り導出される。
  - もし以下の条件のいずれかが真ならば、サンプル  $p[x,y]$  は、“Intra\_16x16 予測で利用不可” としてマークされる。
    - $mbAddrN$  が利用不可である。
    - マクロブロック  $mbAddrN$  が INTER 予測モードで符号化されており、かつ  $constrained\_intra\_pred\_flag$  が 1 である。
    - マクロブロック  $mbAddrN$  が、SI に等しい  $mb\_type$  を持っており、かつ  $constrained\_intra\_pred\_flag$  が 1 である。
  - それ以外、サンプル  $p[x,y]$  は “Intra\_16x16 予測で利用可能” としてマークされ、サンプル  $p[x,y]$  の値は次のように導出される。

- マクロブロック mbAddrN の左上輝度サンプル位置が、6.4.1 小節の逆マクロブロック走査処理を mbAddrN を入力として起動して導出され、出力は(xM,yM)に割り当てられる。
- 変数 MbaffFrameFlag およびマクロブロック mbAddrN に依存して、サンプル値 p[x,y]は次のように導出される。

- もし MbaffFrameFlag が 1 に等しく、かつマクロブロック mbAddrN がフィールドマクロブロックならば、

$$p[x,y]=cS_L[xM+xW,yM+2*yW] \quad (8-116)$$

- それ以外 (MbaffFrameFlag が 0 に等しい、あるいはマクロブロック mbAddrN がフレームマクロブロック)、

$$p[x,y]=cS_L[xM+xW,yM+yW] \quad (8-117)$$

pred<sub>L</sub>[x,y]、x,y=0..15 は 16x16 輝度ブロックサンプルの予測サンプルを示すとする。

Intra\_16x16 予測モードは、表 8-4/JT-H264 において規定される。

**表8-4/JT-H264 Intra16x16PredModeと関連する名前の規定  
(ITU-T H.264)**

Intra16x16PredMode	Intra16x16PredModeの名称
0	Intra_16x16_Vertical(予測モード)
1	Intra_16x16_Horizontal(予測モード)
2	Intra_16x16_DC(予測モード)
3	Intra_16x16_Plane(予測モード)

Intra16x16PredMode に従い、8.3.3.1 から 8.3.3.4 小節において規定される Intra\_16x16 予測モードの 1 つが起動される。

### 8.3.3.1 Intra\_16x16\_Vertical予測モードの規定

この Intra\_16x16 予測モードは、サンプル p[x,-1]、x=0..15 が、“Intra\_16x16 予測で利用可能”としてマークされている時のみ、使用されなければならない。

$$pred_L[x,y]=p[x,-1]、x,y=0..15 \quad (8-118)$$

### 8.3.3.2 Intra\_16x16\_Horizontal予測モードの規定

この Intra\_16x16 予測モードは、サンプル p[-1,y]、y=0..15 が、“Intra\_16x16 予測で利用可能”としてマークされている時のみ、使用されなければならない。

$$pred_L[x,y]=p[-1,y]、x,y=0..15 \quad (8-119)$$

### 8.3.3.3 Intra\_16x16\_DC予測モードの規定

この Intra\_16x16 予測モードは、隣接サンプルが “Intra\_16x16 予測で利用可能” としてマークされているかどうかに従い、次の通り動作する。

- もし、全ての隣接サンプル  $p[x,-1]$ 、 $x=0..15$  と  $p[-1,y]$ 、 $y=0..15$  が、“Intra\_16x16 予測で利用可能” としてマークされているならば、マクロブロックの全ての輝度サンプルに対する予測は、次によって与えられる。

$$\text{pred}_L[x,y]=\left(\sum_{x'=0}^{15} p[x',-1] + \sum_{y'=0}^{15} p[-1,y'] + 16\right) \gg 5, x,y=0..15 \quad (8-120)$$

- それ以外、もし隣接サンプル  $p[x,-1]$ 、 $x=0..15$  のいくつかが “Intra\_16x16 予測で利用不可” としてマークされていて、かつ隣接サンプル  $p[-1,y]$ 、 $y=0..15$  の全てが、“Intra\_16x16 予測で利用可能” としてマークされているならば、マクロブロックの全ての輝度サンプルに対する予測は、次式によって与えられる。

$$\text{pred}_L[x,y]=\left(\sum_{y'=0}^{15} p[-1,y'] + 8\right) \gg 4, x,y=0..15 \quad (8-121)$$

- それ以外、もし隣接サンプル  $p[-1,y]$ 、 $y=0..15$  のいくつかが “Intra\_16x16 予測で利用不可” としてマークされていて、かつ隣接サンプル  $p[x,-1]$ 、 $x=0..15$  の全てが、“Intra\_16x16 予測で利用可能” としてマークされているならば、マクロブロックの全ての輝度サンプルに対する予測は、次式によって与えられる。

$$\text{pred}_L[x,y]=\left(\sum_{x'=0}^{15} p[x',-1] + 8\right) \gg 4, x,y=0..15 \quad (8-122)$$

- それ以外（隣接サンプル  $p[x,-1]$ 、 $x=0..15$  のいくつかと隣接サンプル  $p[-1,y]$ 、 $y=0..15$  のいくつかのいずれもが、“Intra\_16x16 予測で利用不可” としてマークされている）、マクロブロックの全ての輝度サンプルに対する予測は、次式によって与えられる。

$$\text{pred}_L[x,y]=(1 \ll (\text{BitDepth}_Y - 1)), x,y=0..15 \quad (8-123)$$

### 8.3.3.4 Intra\_16x16\_Plane予測モードの規定

この Intra\_16x16 予測モードは、サンプル  $p[x,-1]$ 、 $x=-1..15$  と  $p[-1,y]$ 、 $y=0..15$  が、“Intra\_16x16 予測で利用可能” としてマークされている時のみ、使用されなければならない。

$$\text{pred}_L[x,y]=\text{Clip}_{1_Y}((a+b*(x-7)+c*(y-7)+16) \gg 5), x,y=0..15 \quad (8-124)$$

ここで

$$a=16*(p[-1,15]+p[15,-1]) \quad (8-125)$$

$$b=(5*H+32) \gg 6 \quad (8-126)$$

$$c=(5*V+32) \gg 6 \quad (8-127)$$

であり、H と V は、等式 8-128 と 8-129 によって規定される。

$$H = \sum_{x'=0}^7 (x'+1) * (p[8+x',-1] - p[6-x',-1]) \quad (8-128)$$

$$V = \sum_{y=0}^7 (y+1) * (p[-1, 8+y] - p[-1, 6-y]) \quad (8-129)$$

### 8.3.4 色差サンプルのINTRA予測処理

この処理は、I と SI マクロブロックタイプに対して起動される。これは、現マクロブロックに対する INTRA 予測色差サンプルがどのように導出されるかを規定する。

この処理の入力は、隣接マクロブロックのデブロックフィルタ処理前の構築された色差サンプルを含む 2 つの  $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$  配列  $cS_{Cb}$  および  $cS_{Cr}$  である。

この処理の出力は、現マクロブロックに対する INTRA 予測色差サンプル  $\text{pred}_{Cb}[x,y]$  と  $\text{pred}_{Cr}[x,y]$  である。

ChromaArrayType の値に依存して次が適用される。

- もし、ChromaArrayType が 3 に等しいならば、現マクロブロック  $\text{pred}_{Cb}[x,y]$  および  $\text{pred}_{Cr}[x,y]$  のための INTRA 予測色差サンプルは、8.3.4.5 小節に規定される 3 に等しい ChromaArrayType の色差サンプルに対する INTRA 予測処理を用いて導出される。
- それ以外 (ChromaArrayType が 1 または 2)、次の文章が現マクロブロック  $\text{pred}_{Cb}[x,y]$  および  $\text{pred}_{Cr}[x,y]$  に対する INTRA 予測色差サンプルを規定する。

マクロブロックの両色差ブロック (Cb と Cr) は、同じ予測モードを使用する。予測モードは、各色差ブロックに個別に適用される。この小節において規定される処理は、各色差ブロックに対して起動される。この小節の残りの部分では、色差ブロックとは、2 個の色差ブロックのうちの 1 個を指し、下付き添字 C は、下付き添字 Cb または Cr の置き換えとして使用されるものとする。

デブロックフィルタ処理前の構築色差サンプルである隣接サンプル  $p[x,y]$ 、 $x=-1,y=-1..MbHeightC-1$ 、そして  $x=0..MbWidthC-1,y=-1$  は、次の通り導出される。

- 6.4.11 小節における隣接位置の導出処理が、 $(x,y)$  が  $(xN,yN)$  に入力として割り当てられ、出力として  $\text{mbAddrN}$  と  $(xW,yW)$  を用いて色差位置に対して起動される。
- 各々のサンプル  $p[x,y]$  は、次の通り導出される。
  - もし以下の条件のいずれかが真ならば、サンプル  $p[x,y]$  は、“INTRA 色差予測で利用不可” としてマークされる。
    - $\text{mbAddrN}$  が利用不可である。
    - マクロブロック  $\text{mbAddrN}$  が INTER 予測モードで符号化されており、かつ  $\text{constrained\_intra\_pred\_flag}$  が 1 である。
    - マクロブロック  $\text{mbAddrN}$  が、SI に等しい  $\text{mb\_type}$  を持っており、かつ  $\text{constrained\_intra\_pred\_flag}$  が 1 であり、かつ現マクロブロックが、SI に等しい  $\text{mb\_type}$  を持っていない。
  - それ以外、サンプル  $p[x,y]$  は “INTRA 色差予測で利用可能” としてマークされ、サンプル  $p[x,y]$  の値は次のように導出される。
    - マクロブロック  $\text{mbAddrN}$  の左上輝度サンプル位置が、6.4.1 小節の逆マクロブロック走査処理

を mbAddrN を入力として起動して導出され、出力は(xL,yL)に割り当てられる。

- マクロブロック mbAddrN の左上色差サンプル位置(xM,yM)が、次により導出される。

$$xM = (xL \gg 4) * MbWidthC \quad (8-130)$$

$$yM = ((yL \gg 4) * MbHeightC) + (yL \% 2) \quad (8-131)$$

- 変数 MbaffFrameFlag およびマクロブロック mbAddrN に依存して、サンプル値 p[x,y]は次のように導出される。

- もし MbaffFrameFlag が 1 に等しく、かつマクロブロック mbAddrN がフィールドマクロブロックならば、

$$p[x,y] = cSc[xM+xW,yM+2*yW] \quad (8-132)$$

- それ以外 (MbaffFrameFlag が 0 に等しい、あるいはマクロブロック mbAddrN がフレームマクロブロック)、

$$p[x,y] = cSc[xM+xW,yM+yW] \quad (8-133)$$

predc[x,y]、x=0..MbWidthC-1,y=0..MbHeightC-1 は色差ブロックサンプルの予測サンプルを示すとする。

INTRA 色差予測モードは、表 8-5/JT-H264 において規定される。

**表8-5/JT-H264 INTRA 色差予測モードと、関連する名前の規定 (ITU-T H.264)**

intra_chroma_pred_mode	intra_chroma_pred_modeの名前
0	Intra_Chroma_DC(予測モード)
1	Intra_Chroma_Horizontal(予測モード)
2	Intra_Chroma_Vertical(予測モード)
3	Intra_Chroma_Plane(予測モード)

intra\_chroma\_pred\_mode に従い、8.3.4.1 から 8.3.4.4 小節において規定される INTRA 色差予測モードの 1 つが起動される。

#### 8.3.4.1 Intra\_Chroma\_DC 予測モードの規定

この INTRA 色差予測モードは、intra\_chroma\_pred\_mode が 0 に等しい場合に起動される。

chroma4x4BlkIdx=0..(1<<(ChromaArrayType+1))-1 で索引される各々の 4×4 サンプルの色差ブロックに対して、次が適用される。

- インデックス chroma4x4BlkIdx を持つ 4×4 色差ブロックの左上のサンプル位置は次の通り導出される。

$$xO = \text{InverseRasterScan}(\text{chroma4x4BlkIdx}, 4, 4, 8, 0) \quad (8-134)$$

$$yO = \text{InverseRasterScan}(\text{chroma4x4BlkIdx}, 4, 4, 8, 1) \quad (8-135)$$

- $xO$ および $yO$ の値に依存して、次が適用される。
- もし $(xO,yO)$ が $(0,0)$ に等しいか、あるいは $xO$ と $yO$ が0より大きいならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は次の通り導出される。

- もし、全てのサンプル $p[x+xO,-1]$ 、 $x=0..3$ と、全てのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が、“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$\text{pred}_c[x+xO,y+yO]=\left(\sum_{x'=0}^3 p[x'+xO,-1]+\sum_{y'=0}^3 p[-1,y'+yO]+4\right)\gg 3, x,y=0..3 \quad (8-136)$$

- それ以外、もしいくつかのサンプル $p[x+xO,-1]$ 、 $x=0..3$ が“INTRA色差予測で利用不可”としてマークされており、全てのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$\text{pred}_c[x+xO,y+yO]=\left(\sum_{y'=0}^3 p[-1,y'+yO]+2\right)\gg 2, x,y=0..3 \quad (8-137)$$

- それ以外、もしいくつかのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用不可”としてマークされており、全てのサンプル $p[x+xO,-1]$ 、 $x=0..3$ が“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$\text{pred}_c[x+xO,y+yO]=\left(\sum_{x'=0}^3 p[x'+xO,-1]+2\right)\gg 2, x,y=0..3 \quad (8-138)$$

- それ以外(いくつかのサンプル $p[x+xO,-1]$ 、 $x=0..3$ といくつかのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用不可”としてマークされている)、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$\text{pred}_c[x+xO,y+yO]=(1\ll(\text{BitDepth}_c-1)), x,y=0..3 \quad (8-139)$$

- それ以外、もし $xO$ が0より大きく、かつ $yO$ が0に等しいならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は次の通り導出される。

- もし全てのサンプル $p[x+xO,-1]$ 、 $x=0..3$ が“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$\text{pred}_c[x+xO,y+yO]=\left(\sum_{x'=0}^3 p[x'+xO,-1]+2\right)\gg 2, x,y=0..3 \quad (8-140)$$

- それ以外、もし全てのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $\text{pred}_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$\text{pred}_c[x+xO,y+yO]=\left(\sum_{y'=0}^3 p[-1,y'+yO]+2\right)\gg 2, x,y=0..3 \quad (8-141)$$

- それ以外(いくつかのサンプル $p[x+xO,-1]$ 、 $x=0..3$ といくつかのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用不可”としてマークされている)、予測サンプル値 $pred_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$pred_c[x+xO,y+yO]=(1 \ll (\text{BitDepth}_c-1)), x,y=0..3 \quad (8-142)$$

- それ以外( $xO$ が0に等しく、かつ $yO$ が0より大きい)、予測サンプル値 $pred_c[x+xO,y+yO]$ 、 $x,y=0..3$ は次の通り導出される。
  - もし全てのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $pred_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$pred_c[x+xO,y+yO]=\left(\sum_{y'=0}^3 p[-1,y'+yO]+2\right) \gg 2, x,y=0..3 \quad (8-143)$$

- それ以外、もし全てのサンプル $p[x+xO,-1]$ 、 $x=0..3$ が“INTRA色差予測で利用可能”としてマークされているならば、予測サンプル値 $pred_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$pred_c[x+xO,y+yO]=\left(\sum_{x'=0}^3 p[x'+xO,-1]+2\right) \gg 2, x,y=0..3 \quad (8-144)$$

- それ以外(いくつかのサンプル $p[x+xO,-1]$ 、 $x=0..3$ といくつかのサンプル $p[-1,y+yO]$ 、 $y=0..3$ が“INTRA色差予測で利用不可”としてマークされている)、予測サンプル値 $pred_c[x+xO,y+yO]$ 、 $x,y=0..3$ は以下で導出される。

$$pred_c[x+xO,y+yO]=(1 \ll (\text{BitDepth}_c-1)), x,y=0..3 \quad (8-145)$$

#### 8.3.4.2 Intra\_Chroma\_Horizontal予測モードの規定

このINTRA色差予測モードは、`intra_chroma_pred_mode`が1に等しい場合に起動される。

このモードは、サンプル $p[-1,y]$ 、 $y=0..MbHeightC-1$ が、“INTRA色差予測で利用可能”としてマークされている時のみ、使用されなければならない。

予測サンプル $pred_c[x,y]$ の値は、次の通り導出される。

$$pred_c[x,y]=p[-1,y], x=0..MbWidthC-1,y=0..MbHeightC-1 \quad (8-146)$$

#### 8.3.4.3 Intra\_Chroma\_Vertical予測モードの規定

このINTRA色差予測モードは、`intra_chroma_pred_mode`が2に等しい場合に起動される。

このモードは、サンプル $p[x,-1]$ 、 $x=0..MbWidthC-1$ が、“INTRA色差予測で利用可能”としてマークされている時のみ、使用されなければならない。

予測サンプル $pred_c[x,y]$ の値は、次の通り導出される。

$$pred_c[x,y]=p[x,-1], x=0..MbWidthC-1,y=0..MbHeightC-1 \quad (8-147)$$

### 8.3.4.4 Intra\_Chroma\_Plane予測モードの規定

この INTRA 色差予測モードは、intra\_chroma\_pred\_mode が 3 に等しい場合に起動される。

このモードは、サンプル  $p[x,-1]$ 、 $x=0..MbWidthC-1$  と、 $p[-1,y]$ 、 $y=-1..MbHeightC-1$  が、“INTRA 色差予測で利用可能”としてマークされている時のみ、使用されなければならない。

予測サンプル  $pred_c[x,y]$ の値は、次の通り導出される。

変数  $x_{CF}$  を  $4*(ChromaArrayType==3)$ に等しく設定し、変数  $y_{CF}$  を  $4*(ChromaArrayType!=1)$ に等しく設定する。

$$pred_c[x,y]=Clip1_c((a+b*(x-3-x_{CF})+c*(y-3-y_{CF})+16)>>5), x=0..MbWidthC-1,y=0..MbHeightC-1 \quad (8-148)$$

ここで

$$a=16*(p[-1,MbHeightC-1]+p[MbWidthC-1,-1]) \quad (8-149)$$

$$b=((34-29*(ChromaArrayType==3))*H+32)>>6 \quad (8-150)$$

$$c=((34-29*(ChromaArrayType!=1))*V+32)>>6 \quad (8-151)$$

であり、H と V は、以下で規定される。

$$H = \sum_{x'=0}^{3+x_{CF}} (x'+1) * (p[4+x_{CF}+x',-1] - p[2+x_{CF}-x',-1]) \quad (8-152)$$

$$V = \sum_{y'=0}^{3+y_{CF}} (y'+1) * (p[-1,4+y_{CF}+y'] - p[-1,2+y_{CF}-y']) \quad (8-153)$$

### 8.3.4.5 3に等しいChromaArrayTypeの色差サンプルに対するINTRA予測

この処理は ChromaArrayType が 3 に等しい時起動される。この処理は、I および SI マクロブロックタイプに対して起動される。この処理は ChromaArrayType が 3 に等しい時に現マクロブロックの INTRA 予測色差サンプルがどのように導出されるかを規定する。

この処理の入力は、隣接する Cb および Cr ブロックからのデブロックフィルタ処理前の構築されたサンプル、および Intra\_NxN (NxN は 4x4 または 8x8 に等しい) 予測モードについては、隣接マクロブロックからの IntraNxNPredMode に関連する値である。

この処理の出力は、マクロブロックの Cb および Cr 成分の INTRA 予測サンプル、または Intra\_NxN 予測処理の場合は、出力はマクロブロックの予測サンプルの 16x16 Cb 配列の一部としての NxN Cb サンプル配列およびマクロブロックの予測サンプルの 16x16 Cr 配列の一部としての NxN Cr サンプル配列である。

マクロブロックの同じブロックインデックスを持つ Cb,Cr および輝度ブロックの各々は、同じ予測モードを用いる。予測モードは Cb および Cr ブロックの各々に個別に適用される。この小節で規定された処理は Cb および Cr ブロックの各々に起動される。

マクロブロック予測モードに依存して、次が適用される。

- もし、マクロブロック予測モードが Intra\_4x4 に等しいならば、次が適用される。

- 8.3.1小節に記述されたのと同じ処理が、輝度をCbまたはCrに置き換え、luma4x4BlkIdxをcb4x4BlkIdxまたはcr4x4BlkIdxに置き換え、さらにpred4x4<sub>L</sub>をpred4x4<sub>Cb</sub>またはpred4x4<sub>Cr</sub>に置き換え、BitDepth<sub>Y</sub>をBitDepth<sub>C</sub>に置き換えて、CbまたはCrサンプルにも適用される。
- 8.3.1.1小小節に記述された処理からの出力変数Intra4x4PredMode[luma4x4BlkIdx]は、インデックスcb4x4BlkIdxまたはcr4x4BlkIdxと等しいインデックスluma4x4BlkIdxを用いて4×4 Cbまたは4×4 Crブロックにも用いられる。
- 予測CbまたはCrサンプルを導出する処理は、輝度をCbまたはCrに、pred4x4<sub>L</sub>をpred4x4<sub>Cb</sub>またはpred4x4<sub>Cr</sub>に置き換え、BitDepth<sub>Y</sub>をBitDepth<sub>C</sub>に置き換えた時、8.3.1.2小小節および後に続く小小節に記述された処理と同一である。
- それ以外、もしマクロブロック予測モードがIntra\_8x8に等しいならば、次が適用される。
  - 8.3.2小節に記述されたのと同じ処理が、輝度をCbまたはCrに、luma8x8BlkIdxをcb8x8BlkIdxまたはcr8x8BlkIdxに、pred8x8<sub>L</sub>をpred8x8<sub>Cb</sub>またはpred8x8<sub>Cr</sub>に置き換え、さらにBitDepth<sub>Y</sub>をBitDepth<sub>C</sub>に置き換えて、CbまたはCrサンプルにも適用される。
  - 8.3.2.1小小節に記述された処理からの出力変数Intra8x8PredMode[luma8x8BlkIdx]は、インデックスcb8x8BlkIdxまたはcr8x8BlkIdxと等しいインデックスluma8x8BlkIdxを用いて8×8 Cbまたは8×8 Crブロックにも用いられる。
  - 予測CbまたはCrサンプルを導出する処理は、輝度をCbまたはCrに、pred8x8<sub>L</sub>をpred8x8<sub>Cb</sub>またはpred8x8<sub>Cr</sub>に置き換え、BitDepth<sub>Y</sub>をBitDepth<sub>C</sub>に置き換えた時、8.3.2.2小小節および後に続く小小節に記述された処理と同一である。
- それ以外、もしマクロブロック予測モードがIntra\_16x16に等しいならば、次が適用される。
  - 8.3.3小節および後に続く8.3.3.1から8.3.3.4小小節に記述されたのと同じ処理が、輝度をCbまたはCrに、pred<sub>L</sub>をpred<sub>Cb</sub>またはpred<sub>Cr</sub>に置き換え、さらにBitDepth<sub>Y</sub>をBitDepth<sub>C</sub>に置き換えて、CbまたはCrサンプルにも適用される。

### 8.3.5 I\_PCMマクロブロックに対するサンプル構築処理

この処理は、mb\_typeがI\_PCMに等しい時、起動される。

変数dyは、次の通り導出される。

- もしMbaffFrameFlagが1であり、かつ現マクロブロックがフィールドマクロブロックであるならば、dyは2に等しく設定される。
- それ以外 (MbaffFrameFlagが0か、または現マクロブロックがフレームマクロブロック)、dyは1に等しく設定される。

現マクロブロックの左上の輝度サンプル位置は、6.4.1小節の逆マクロブロック走査処理を、CurrMbAddrを入力として起動し、出力を(xP,yP)に割り当てることで得られる。

デブロック処理の前の構築輝度サンプルは、以下で規定されるよう生成される。

```
for(i=0;i<256;i++)
  SL[xP+(i%16),yP+dy*(i/16)]=pcm_sample_luma[i] (8-154)
```

ChromaArrayType が 0 に等しくない時、デブロック処理の前の構築色差サンプルは、以下で規定されるよう生成される。

```

for(i=0;i<MbWidthC*MbHeightC;i++){
    S'Cb[(xP/SubWidthC)+(i%MbWidthC),
        ((yP+SubHeightC-1)/SubHeightC)+dy*(i/MbWidthC)]=
        pcm_sample_chroma[i]
    S'Cr[(xP/SubWidthC)+(i%MbWidthC),
        ((yP+SubHeightC-1)/SubHeightC)+dy*(i/MbWidthC)]=
        pcm_sample_chroma[i+MbWidthC*MbHeightC]
}

```

(8-155)

#### 8.4 INTER 予測処理

P と B マクロブロックタイプを復号する時、この処理は起動される。

この処理の出力は、輝度サンプルの  $16 \times 16$  配列  $\text{pred}_L$  と、ChromaArrayType が 0 に等しくない場合は各々が色差成分 Cb、Cr に対応する色差サンプルの 2 つの  $(\text{MbWidthC}) \times (\text{MbHeightC})$  配列  $\text{pred}_{Cb}$  と  $\text{pred}_{Cr}$  で構成される現マクロブロックの INTER 予測サンプルである。

マクロブロックのパーティションは **mb\_type** で規定される。それぞれのマクロブロックパーティションは **mbPartIdx** によって参照される。マクロブロックパーティションがサブマクロブロックと等しいパーティションから構成される時、それぞれのサブマクロブロックは **sub\_mb\_type** で規定されるように、さらにサブマクロブロックパーティションに分割することができる。それぞれのサブマクロブロックパーティションは **subMbPartIdx** によって参照される。マクロブロックパーティションがサブマクロブロックから構成されていない時は、**subMbPartIdx** は 0 に等しく設定される。

以下のステップが、各マクロブロックパーティションに対して、もしくは各サブマクロブロックパーティションに対して規定される。

マクロブロックパーティションとサブマクロブロックパーティションの幅と高さを記述する関数 **MbPartWidth()**、**MbPartHeight()**、**SubMbPartWidth()**、および **SubMbPartHeight()** が表 7-13、7-14、7-17 および 7-18/JT-H264 で規定される。

マクロブロックパーティションインデックス **mbPartIdx** の範囲は次の通り導出される。

- もし **mb\_type** が **B\_Skip** または **B\_Direct\_16x16** に等しいならば、**mbPartIdx** は値 0..3 上を進行する。
- それ以外 (**mb\_type** が **B\_Skip** または **B\_Direct\_16x16** に等しくない)、**mbPartIdx** は値 0..**NumMbPart(mb\_type)**-1 上を進行する。

各々の **mbPartIdx** の値に対して、マクロブロックでの、各々のマクロブロックパーティションまたはサブマクロブロックパーティションに対する変数 **partWidth** と **partHeight** は次の通り導出される。

- もし **mb\_type** が **P\_8x8**、**P\_8x8ref0**、**B\_Skip**、**B\_Direct\_16x16** または **B\_8x8** に等しくないならば、**subMbPartIdx** は 0 に等しく設定され、**partWidth** と **partHeight** は以下で導出される。

$\text{partWidth} = \text{MbPartWidth}(\text{mb\_type})$  (8-156)

$\text{partHeight} = \text{MbPartHeight}(\text{mb\_type})$  (8-157)

- それ以外、もしmb\_typeがP\_8x8またはP\_8x8ref0に等しい、またはmb\_typeがB\_8x8に等しくかつsub\_mb\_type[mbPartIdx]がB\_Direct\_8x8に等しくないならば、subMbPartIdxは値0..NumSubMbPart(sub\_mb\_type)-1上を進行し、partWidthとpartHeightは以下で導出される。

$$\text{partWidth}=\text{SubMbPartWidth}(\text{sub\_mb\_type}[\text{mbPartIdx}]) \quad (8-158)$$

$$\text{partHeight}=\text{SubMbPartHeight}(\text{sub\_mb\_type}[\text{mbPartIdx}]) \quad (8-159)$$

- それ以外 (mb\_typeがB\_SkipまたはB\_Direct\_16x16に等しい、またはmb\_typeがB\_8x8に等しくかつsub\_mb\_type[mbPartIdx]がB\_Direct\_8x8に等しい)、subMbPartIdxは値0..3上を進行し、partWidthとpartHeightは以下で導出される。

$$\text{partWidth}=4 \quad (8-160)$$

$$\text{partHeight}=4 \quad (8-161)$$

ChromaArrayTypeが0に等しくない場合、変数partWidthCとpartHeightCは以下で導出される。

$$\text{partWidthC}=\text{partWidth}/\text{SubWidthC} \quad (8-162)$$

$$\text{partHeightC}=\text{partHeight}/\text{SubHeightC} \quad (8-163)$$

マクロブロックに対する8.4.1小節の一切の起動よりも前に、変数MvCntは0に等しく初期設定されるものとする。

マクロブロックパーティションmbPartIdxとサブマクロブロックパーティションsubMbPartIdxに対するINTER予測処理は、以下の順序づけられたステップで構成される。

1. 8.4.1小節で規定される動きベクトル成分と参照インデックスの導出処理。

この処理の入力は以下である。

- マクロブロックパーティションmbPartIdx
- サブマクロブロックパーティションsubMbPartIdx

この処理の出力は以下である。

- 輝度動きベクトルmvL0とmvL1、およびChromaArrayTypeが0に等しくない場合には色差動きベクトルmvCL0とmvCL1
- 参照インデックスrefIdxL0とrefIdxL1
- 予測リスト利用フラグpredFlagL0とpredFlagL1
- サブマクロブロックパーティション動きベクトルカウントsubMvCnt

2. 変数MvCntをsubMvCntだけ増加する。

3. 8.4.3小節で規定される予測重みの導出処理。

この処理の入力は以下である。

- 参照インデックスrefIdxL0とrefIdxL1

- 予測リスト利用フラグ `predFlagL0` と `predFlagL1`

この処理の出力は以下である。

- 重み付け予測のための変数 `logWDC`、`w0C`、`w1C`、`o0C`、`o1C` の `C` を `L` に、および `ChromaArrayType` が `0` に等しくない時は `Cb` および `Cr` に置き換えたもの

#### 4. 8.4.2 小節で規定される INTER 予測サンプルの復号処理。

この処理の入力は以下である。

- マクロブロックパーティション `mbPartIdx`
- サブマクロブロックパーティション `subMbPartIdx`
- 輝度に対して、および（利用可能であれば）色差に対してパーティションの幅と高さを規定する変数 `partWidth`、`partHeight`、（利用可能であれば）`partWidthC` と（利用可能であれば）`partHeightC`
- 輝度動きベクトル `mvL0` と `mvL1`、および `ChromaArrayType` が `0` に等しくない場合には色差動きベクトル `mvCL0` と `mvCL1`
- 参照インデックス `refIdxL0` と `refIdxL1`
- 予測リスト利用フラグ `predFlagL0` と `predFlagL1`
- 重み付け予測のための変数 `logWDC`、`w0C`、`w1C`、`o0C`、`o1C` の `C` を `L` に、および `ChromaArrayType` が `0` に等しくない時は `Cb` および `Cr` に置き換えたもの

この処理の出力は以下である。

- INTER 予測サンプル(`pred`)、これは予測輝度サンプルの $(partWidth) \times (partHeight)$ 配列 `predPartL`、および `ChromaArrayType` が `0` に等しくない場合には各々が色差成分 `Cb`、`Cr` に対応する予測色差サンプルの2つの $(partWidthC) \times (partHeightC)$ 配列 `predPartCb` と `predPartCr`

復号処理で後に起動される変数の導出処理で使うために、以下の割り当てがなされる。

$$MvL0[mbPartIdx][subMbPartIdx]=mvL0 \quad (8-164)$$

$$MvL1[mbPartIdx][subMbPartIdx]=mvL1 \quad (8-165)$$

$$RefIdxL0[mbPartIdx]=refIdxL0 \quad (8-166)$$

$$RefIdxL1[mbPartIdx]=refIdxL1 \quad (8-167)$$

$$PredFlagL0[mbPartIdx]=predFlagL0 \quad (8-168)$$

$$PredFlagL1[mbPartIdx]=predFlagL1 \quad (8-169)$$

パーティションの左上サンプルの、マクロブロックの左上サンプルからの相対位置は、`mbPartIdx` を入力とし、 $(xP,yP)$ を出力として 6.4.2.1 小節で記述されるような逆マクロブロックパーティション走査処理を起動することで導出される。

サブマクロブロックパーティションの左上サンプルの、マクロブロックパーティションの左上サンプルか

らの相対位置は、subMbPartIdx を入力とし、(xS,yS) を出力として 6.4.2.2 小小節で記述されるような逆サブマクロブロックパーティション走査処理を起動することで導出される。

マクロブロック予測は、以下のように、パーティションまたはサブマクロブロックパーティション予測サンプルを、そのマクロブロック内の正しい相対位置に配置して形成される。

変数  $pred_L[xP+xS+x,yP+yS+y]$ 、 $x=0..partWidth-1,y=0..partHeight-1$  は以下で導出される。

$$pred_L[xP+xS+x,yP+yS+y]=predPart_L[x,y] \quad (8-170)$$

ChromaArrayType が 0 に等しくない場合には、変数  $pred_C$  は、 $x=0..partWidthC-1,y=0..partHeightC-1$ 、さらに  $pred_C$  と  $predPart_C$  の C が Cb または Cr で置き換えられて用いられ、以下で導出される。

$$pred_C[xP/SubWidthC+xS/SubWidthC+x,yP/SubHeightC+yS/SubHeightC+y]=predPart_C[x,y] \quad (8-171)$$

#### 8.4.1 動きベクトル成分と参照インデックスの導出処理

この処理の入力は以下の通りである。

- マクロブロックパーティション mbPartIdx
- サブマクロブロックパーティション subMbPartIdx

この処理の出力は以下の通りである。

- 輝度動きベクトル mvL0 と mvL1、および、ChromaArrayType が 0 に等しくない時、色差動きベクトル mvCL0 と mvCL1
- 参照インデックス refIdxL0 と refIdxL1
- 予測リスト利用フラグ predFlagL0 と predFlagL1
- サブパーティションマクロブロック動きベクトルカウント変数 subMvCnt

変数 refIdxL0、refIdxL1 ならびに mvL0、mvL1 の導出に次が適用される。

- もし mb\_type が P\_Skip に等しいならば、8.4.1.1 小小節における P、SP スライススキップマクロブロックの輝度動きベクトルの導出処理が、輝度動きベクトル mvL0 と参照インデックス refIdxL0 を出力として起動され、predFlagL0 は 1 に設定される。mvL1 と refIdxL1 は利用不可としてマークされ、predFlagL1 は 0 に設定される。サブパーティション動きベクトルカウント変数 subMvCnt が 1 に等しく設定される。
- それ以外、もし mb\_type が B\_Skip、または B\_Direct\_16x16 に等しいか、もしくは sub\_mb\_type[mbPartIdx] が B\_Direct\_8x8 と等しいならば、8.4.1.2 小小節の B スライスにおける B\_Skip、B\_Direct\_16x16、および B\_Direct\_8x8 の輝度動きベクトルの導出処理が、mbPartIdx、subMbPartIdx を入力とし、輝度動きベクトル mvL0、mvL1、参照インデックス refIdxL0、refIdxL1、サブパーティション動きベクトルカウント subMvCnt、そして予測利用フラグ predFlagL0 と predFlagL1 が出力となるように起動される。
- それ以外、変数 predFlagLX、mvLX、refIdxLX、そして Pred\_LX、およびシンタックス要素 ref\_idx\_IX と mvd\_IX において、0 または 1 のいずれかに置換される X に対して、次が適用される。

1. 変数 refIdxLX、predFlagLX は次の通り導出される。

- もし MbPartPredMode(mb\_type,mbPartIdx) または SubMbPredMode(sub\_mb\_type[mbPartIdx]) が

Pred\_LX または BiPred に等しいならば、

$$\text{refIdxLX}=\text{ref\_idx\_lX}[\text{mbPartIdx}] \quad (8-172)$$

$$\text{predFlagLX}=1 \quad (8-173)$$

- それ以外、変数 refIdxLX、predFlagLX は以下で規定される。

$$\text{refIdxLX}=-1 \quad (8-174)$$

$$\text{predFlagLX}=0 \quad (8-175)$$

2. サブパーティション動きベクトルカウントに対する変数 subMvCnt は predFlagL0+predFlagL1 に等しく設定される。

3. 変数 currSubMbType は次の通り導出される。

- もしマクロブロックタイプが B\_8x8 に等しいならば、currSubMbType は sub\_mb\_type[ mbPartIdx ] に等しく設定される。
- それ以外 (マクロブロックタイプが B\_8x8 に等しくない)、currSubMbType は "na" に等しく設定される。

4. predFlagLX が 1 に等しい時、8.4.1.3 小小節の輝度動きベクトル予測の導出処理が、mbPartIdx、subMbPartIdx、refIdxLX、および currSubMbType を入力とし、mvpLX が出力になるように起動される。輝度動きベクトルは、以下で導出される。

$$\text{mvLX}[0]=\text{mvpLX}[0]+\text{mvd\_lX}[\text{mbPartIdx}][\text{subMbPartIdx}][0] \quad (8-176)$$

$$\text{mvLX}[1]=\text{mvpLX}[1]+\text{mvd\_lX}[\text{mbPartIdx}][\text{subMbPartIdx}][1] \quad (8-177)$$

ChromaArrayType が 0 に等しくなく、かつ predFlagLX(ここで X は 0 か 1)が 1 に等しい時、8.4.1.4 小小節の色差動きベクトルの導出処理が、mvLX、refIdxLX を入力とし、mvCLX が出力となるように起動される。

#### 8.4.1.1 P、SPスライスにおけるスキップマクロブロックの輝度動きベクトル導出処理

この処理は、mb\_type が P\_Skip に等しい時、起動される。

この処理の出力は、動きベクトル mvL0 と参照インデックス refIdxL0 である。

スキップマクロブロックの参照インデックス refIdxL0 は、次の通り導出される。

$$\text{refIdxL0}=0 \quad (8-178)$$

P\_Skip マクロブロックタイプの動きベクトル mvL0 の導出のために、次が適用される。

- 8.4.1.3.2 小小節で規定された処理が、0 に設定された mbPartIdx、0 に設定された subMbPartIdx、"na" に設定された currSubMbType、そして 0 に設定された listSuffixFlag を入力として起動され、出力は mbAddrA、mbAddrB、mvL0A、mvL0B、refIdxL0A、および refIdxL0B が割り当てられる。
- 変数 mvL0 は次の通り規定される。

- もし以下の条件のいずれかが真ならば、動きベクトル `mvL0` の両成分は 0 に設定される。
  - `mbAddrA` は利用不可である。
  - `mbAddrB` は利用不可である。
  - `refIdxL0A` が 0 であり、かつ `mvL0A` の両成分が 0 である。
  - `refIdxL0B` が 0 であり、かつ `mvL0B` の両成分が 0 である。
- それ以外、8.4.1.3 小小節で規定される輝度動きベクトル予測の導出処理が、`mbPartIdx=0`、`subMbPartIdx=0`、`refIdxL0` および `currSubMbType="na"` を入力として起動され、出力には `mvL0` が割り当てられる。

記 予測因子が実際の動きベクトルと等しいので、出力は直接`mvL0`に割り当てられる。

#### 8.4.1.2 B\_Skip、B\_Direct\_16x16、およびB\_Direct\_8x8の輝度動きベクトル導出処理

この処理は、`mb_type` が `B_Skip` か `B_Direct_16x16` に等しいか、または `sub_mb_type[mbPartIdx]` が `B_Direct_8x8` に等しい時に起動される。

この処理の入力は、`mbPartIdx` と `subMbPartIdx` である。

この処理の出力は、参照インデックス `refIdxL0`、`refIdxL1`、動きベクトル `mvL0` と `mvL1`、サブパーティション動きベクトルカウント `subMvCnt`、および予測リスト利用フラグ `predFlagL0` と `predFlagL1` である。

導出処理は 7.3.3 小節で規定されるように、スライスヘッダシンタックスのビットストリームに存在する `direct_spatial_mv_pred_flag` 値に依存し、次の通り規定される。

- もし `direct_spatial_mv_pred_flag` が 1 に等しいならば、この処理の出力が導出されるモードは空間方向直接予測モードと呼ばれる。
- それ以外(`direct_spatial_mv_pred_flag` が 0 に等しい)、この処理の出力が導出されるモードは時間方向直接予測モードと呼ばれる。

空間方向、時間方向、両直接予測モードは、8.4.1.2.1 小小小節で規定されるように、共通位置の動きベクトルと参照インデックスを使用する。

動きベクトルと参照インデックスは次で導出される。

- もし空間方向直接予測モードが使用されるならば、8.4.1.2.2 小小小節で規定される直接動きベクトルと参照インデックス予測モードが使用され、`subMvCnt` が出力になる。
- それ以外(時間方向直接予測モードが使用される)、8.4.1.2.3 小小小節で規定される直接動きベクトルと参照インデックス予測モードが使用され、変数 `subMvCnt` が次の通り導出される。
  - もし `subMbPartIdx` が 0 に等しいならば、`subMvCnt` は 2 に等しく設定される。
  - それ以外 (`subMbPartIdx` が 0 に等しくない)、`subMvCnt` は 0 に等しく設定される。

##### 8.4.1.2.1 共通位置 4x4 サブマクロブロックパーティション導出処理

この処理の入力は、`mbPartIdx` と `subMbPartIdx` である。

この処理の出力は、ピクチャ colPic、共通位置マクロブロック mbAddrCol、動きベクトル mvCol、参照インデックス refIdxCol、および変数 vertMvScale(One\_To\_One、Frm\_To\_Fld または Fld\_To\_Frm を取りうる)である。

RefPicList1[0]がフレームまたは相補的フィールドペアである時、firstRefPicL1Top と firstRefPicL1Bottom をそれぞれ RefPicList1[0]のトップとボトムフィールドとし、以下の変数を規定する。

$$\text{topAbsDiffPOC} = \text{Abs}(\text{DiffPicOrderCnt}(\text{firstRefPicL1Top}, \text{CurrPic})) \quad (8-179)$$

$$\text{bottomAbsDiffPOC} = \text{Abs}(\text{DiffPicOrderCnt}(\text{firstRefPicL1Bottom}, \text{CurrPic})) \quad (8-180)$$

変数 colPic は表 8-6/JT-H264 で規定されるように、共通位置マクロブロックを含むピクチャを規定する。

**表8-6/JT-H264 変数colPicの規定**  
(ITU-T H.264)

field_pic_flag	RefPicList1[0]は...	mb_field_decoding_flag	付加条件	colPic
1	復号フレームのフィールド			RefPicList1[0]を含むフレーム
	復号フィールド			RefPicList1[0]
0	相補的フィールドペア	0	$\text{topAbsDiffPOC} < \text{bottomAbsDiffPOC}$	firstRefPicL1Top
			$\text{topAbsDiffPOC} \geq \text{bottomAbsDiffPOC}$	firstRefPicL1Bottom
		1	$(\text{CurrMbAddr} \& 1) == 0$	firstRefPicL1Top
			$(\text{CurrMbAddr} \& 1) != 0$	firstRefPicL1Bottom

direct\_8x8\_inference\_flag が 1 に等しい時、subMbPartIdx は次の通り設定される。

$$\text{subMbPartIdx} = \text{mbPartIdx} \quad (8-181)$$

PicCodingStruct(X)を、引数 X が CurrPic か colPic のどちらかである関数とする。それは表 8-7/JT-H264 で規定される。

**表8-7/JT-H264 PicCodingStruct(X)の規定**  
(ITU-T H.264)

X は...に等しい field_pic_flag で符号化される	mb_adaptive_frame_field_flag	PicCodingStruct(X)
1		FLD
0	0	FRM
0	1	AFRM

$luma4x4BlkIdx=mbPartIdx*4+subMbPartIdx$  として、6.4.3 小節に規定される逆 4×4 輝度ブロック走査処理が、 $luma4x4BlkIdx$  を入力とし、 $(xCol,yCol)$ に割り当てられる  $(x,y)$  を出力として起動される。

表 8-8/JT-H264 は共通位置マクロブロックアドレス  $mbAddrCol$ 、 $yM$ 、および変数  $vertMvScale$  を 2 つのステップで規定する。

1.  $PicCodingStruct(CurrPic)$ と  $PicCodingStruct(colPic)$ に依存するマクロブロックアドレス  $mbAddrX$  の規定。

記  $CurrPic$ と  $colPic$ のピクチャ符号化タイプが(FRM,AFRM)か(AFRM,FRM)のどちらかである可能性はない。なぜならば、これらのピクチャ符号化タイプはIDRピクチャによって分離されなければならないからである。

2.  $mb\_field\_decoding\_flag$  と次の通り導出される変数  $fieldDecodingFlagX$  に依存する  $mbAddrCol$ 、 $yM$  および  $vertMvScale$  の規定。

- もしピクチャ  $colPic$  のマクロブロック  $mbAddrX$  がフィールドマクロブロックであるならば、 $fieldDecodingFlagX$  は 1 に等しく設定される。
- それ以外(ピクチャ  $colPic$  のマクロブロック  $mbAddrX$  がフレームマクロブロックである)、 $fieldDecodingFlagX$  は 0 に等しく設定される。

表 8-8/JT-H264 の無規定値は、対応する変数の値が現在の表の行に関連しないことを示す。

$mbAddrCol$  には  $CurrMbAddr$ 、または以下の値のいずれかが設定される。

$$mbAddrCol1=2*PicWidthInMbs*(CurrMbAddr/PicWidthInMbs)+ \\ (CurrMbAddr\%PicWidthInMbs)+PicWidthInMbs*(yCol/8) \quad (8-182)$$

$$mbAddrCol2=2*CurrMbAddr+(yCol/8) \quad (8-183)$$

$$mbAddrCol3=2*CurrMbAddr+bottom\_field\_flag \quad (8-184)$$

$$mbAddrCol4=PicWidthInMbs*(CurrMbAddr/(2*PicWidthInMbs))+ \\ (CurrMbAddr\%PicWidthInMbs) \quad (8-185)$$

$$mbAddrCol5=CurrMbAddr/2 \quad (8-186)$$

$$mbAddrCol6=2*(CurrMbAddr/2)+((topAbsDiffPOC<bottomAbsDiffPOC)?0:1) \quad (8-187)$$

$$mbAddrCol7=2*(CurrMbAddr/2)+(yCol/8) \quad (8-188)$$

表8-8/JT-H264 mbAddrCol、yM、vertMvScaleの規定  
(ITU-T H.264)

PicCodingStruct(CurrPic)	PicCodingStruct(colPic)	mbAddrX	mb_field_decoding_flag	FieldDecodingFlagX	mbAddrCol	yM	vertMvScale
FLD	FLD				CurrMbAddr	yCol	One_To_One
	FRM				mbAddrCol1	$(2*yCol)\%16$	Frm_To_Fld
	AFRM	$2*CurrMbAddr$	0		mbAddrCol2	$(2*yCol)\%16$	Frm_To_Fld
			1		mbAddrCol3	yCol	One_To_One
FRM	FLD				mbAddrCol4	$8*((CurrMbAddr/PicWidthInMbs)\%2)+4*(yCol/8)$	Fld_To_Frm
	FRM				CurrMbAddr	yCol	One_To_One
AFRM	FLD		0		mbAddrCol5	$8*(CurrMbAddr\%2)+4*(yCol/8)$	Fld_To_Frm
			1		mbAddrCol5	yCol	One_To_One
	AFRM	CurrMbAddr	0		CurrMbAddr	yCol	One_To_One
			1		mbAddrCol6	$8*(CurrMbAddr\%2)+4*(yCol/8)$	Fld_To_Frm
	AFRM	CurrMbAddr	0		mbAddrCol7	$(2*yCol)\%16$	Frm_To_Fld
			1		CurrMbAddr	yCol	One_To_One

mbPartIdxCol を共通位置パーティションのマクロブロックパーティションインデックスとし、subMbPartIdxCol を共通位置サブマクロブロックパーティションのサブマクロブロックパーティションインデックスとする。サンプル(xCol,yM)を含むピクチャ colPic 内のマクロブロック mbAddrCol 内のパーティションが、mbPartIdxCol に割り当てられる。また、ピクチャ colPic 内のマクロブロック mbAddrCol のサンプル(xCol,yM)を含むパーティション mbPartIdxCol 内のサブマクロブロックパーティションが、subMbPartIdxCol に割り当てられる。

予測利用フラグ predFlagL0Col と predFlagL1Col は各々 PredFlagL0[mbPartIdxCol] と PredFlagL1[mbPartIdxCol]に等しく設定される。ここで、それらはそれぞれピクチャ colPic 内でマクロブロックパーティション mbAddrCol%mbPartIdxCol に割り当てられた予測利用フラグである。

動きベクトル mvCol と参照インデックス refIdxCol は、次の通り導出される。

- もしマクロブロックmbAddrColがINTRAマクロブロック予測モードで符号化されるか、予測利用フラグpredFlagL0ColとpredFlagL1Colの両方が0であるならば、mvColの両成分は0に設定され、refIdxColは-1に等しく設定される。
- それ以外、次が適用される。
  - もし predFlagL0Col が 1 であるならば、動きベクトル mvCol と参照インデックス refIdxCol は各々 MvL0[mbPartIdxCol][subMbPartIdxCol] と RefIdxL0[mbPartIdxCol] に等しく設定される。ここで、それらはピクチャ colPic 内の (サブ) マクロブロックパーティション mbAddrCol%mbPartIdxCol%subMbPartIdxCol に割り当てられた動きベクトル mvL0 と参照インデックス refIdxL0 である。
  - それ以外(predFlagL0Col が 0 で、predFlagL1Col が 1 ならば)、動きベクトル mvCol と参照インデックス refIdxCol は各々 MvL1[mbPartIdxCol][subMbPartIdxCol] と RefIdxL1[mbPartIdxCol] に等しく設定される。ここで、それらはピクチャ colPic 内の (サブ) マクロブロックパーティション mbAddrCol%mbPartIdxCol%subMbPartIdxCol に割り当てられた動きベクトル mvL1 と参照インデックス refIdxL1 である。

#### 8.4.1.2.2 空間方向直接輝度動きベクトルと参照インデックス予測モードの導出処理

この処理は、direct\_spatial\_mv\_pred\_flag が 1 であり、以下の条件のいずれかが真である時、起動される。

- mb\_type は B\_Skip と等しい
- mb\_type は B\_Direct\_16x16 と等しい
- sub\_mb\_type[mbPartIdx]は B\_Direct\_8x8 と等しい

この処理の入力は mbPartIdx、subMbPartIdx である。

この処理の出力は、参照インデックス refIdxL0、refIdxL1、動きベクトル mvL0 と mvL1、サブパーティション動きベクトルカウント subMvCnt、および予測リスト利用フラグ predFlagL0 と predFlagL1 である。

参照インデックス refIdxL0、refIdxL1、および変数 directZeroPredictionFlag は、以下の順序付けられたステップを適用することで導出される。

1. 変数 currSubMbType を sub\_mb\_type[mbPartIdx]に等しく設定されるとする。
2. 8.4.1.3.2 小小小節で規定される処理が mbPartIdx=0、subMbPartIdx=0、currSubMbType および listSuffixFlag=0 を入力として起動され、出力は動きベクトル mvL0N と参照インデックス refIdxL0N に割り当てられる。ここで、N は A、B、または C に置き換えられる。
3. 8.4.1.3.2 小小小節で規定される処理が mbPartIdx=0、subMbPartIdx=0、currSubMbType および listSuffixFlag=1 を入力として起動され、出力は動きベクトル mvL1N と参照インデックス refIdxL1N に割り当てられる。ここで、N は A、B、または C に置き換えられる。

記1 動きベクトルmvL0N、mvL1N、および参照インデックスrefIdxL0N、refIdxL1Nは、マクロブロックの全ての4×4サブマクロブロックパーティションにおいて同一である。

4. 参照インデックス refIdxL0、refIdxL1、および directZeroPredictionFlag は以下のように導出される。

$$\text{refIdxL0} = \text{MinPositive}(\text{refIdxL0A}, \text{MinPositive}(\text{refIdxL0B}, \text{refIdxL0C})) \quad (8-189)$$

$$\text{refIdxL1} = \text{MinPositive}(\text{refIdxL1A}, \text{MinPositive}(\text{refIdxL1B}, \text{refIdxL1C})) \quad (8-190)$$

$$\text{directZeroPredictionFlag} = 0 \quad (8-191)$$

ここで

$$\text{MinPositive}(x,y) = \begin{cases} \text{Min}(x,y) & x \geq 0 \text{ かつ } y \geq 0 \\ \text{Max}(x,y) & \text{それ以外} \end{cases} \quad (8-192)$$

5. 参照インデックス refIdxL0 と refIdxL1 の両方が 0 未満である時

$$\text{refIdxL0} = 0 \quad (8-193)$$

$$\text{refIdxL1} = 0 \quad (8-194)$$

$$\text{directZeroPredictionFlag} = 1 \quad (8-195)$$

8.4.1.2.1 小小小節で規定される処理が mbPartIdx、subMbPartIdx を入力として与えられて起動され、出力は refIdxCol と mvCol に割り当てられる。

変数 colZeroFlag は次の通り導出される。

- もし以下の条件の全てが真であるならば、colZeroFlag は 1 に設定される。
  - RefPicList1[0]は現在“短期間参照使用”としてマークされている。
  - refIdxCol は 0 である。
  - 動きベクトル成分 mvCol[0]と mvCol[1]の両方が、次の通り規定される単位で-1 から 1 の範囲にある。
    - もし共通位置マクロブロックがフレームマクロブロックならば、mvCol[0]と mvCol[1]の単位は 4 分の 1 輝度フレームサンプルの単位である。
    - それ以外(共通位置マクロブロックがフィールドマクロブロック)、mvCol[0]と mvCol[1]の単位は 4 分の 1 輝度フィールドサンプルの単位である。

記2 上記の条件を決定する目的のため、現マクロブロックがフレームマクロブロックで共通位置マクロブロックがフィールドマクロブロックの場合、あるいは、現マクロブロックがフィールドマクロブロックで共通位置マクロブロックがフレームマクロブロックの場合、現マクロブロックに対する動きベクトルの単位を使用するためにmvCol[1]の値はスケールリングされない。この様子は、それらの場合に等式8-198または等式8-199を使用する、現マクロブロックに対する動きベクトルの単位として同じ単位を使用するために共通位置マクロブロックの動きベクトルにスケールリングを適用する、8.4.1.2.3小小小節で規定される時間方向直接モードのmvCol[1]の使用と異なる。

- それ以外、colZeroFlag は 0 に設定される。

動きベクトル mvLX(X は 0 か 1)は次の通り導出される。

- もし以下の条件のいずれかが真であるならば、動きベクトル mvLX の両成分は 0 に設定される。
  - directZeroPredictionFlag は 1 である。
  - refIdxLX は 0 未満である。
  - refIdxLX は 0 であり、colZeroFlag は 1 である。

- それ以外、8.4.1.3 小小節で規定された処理が、mbPartIdx=0、subMbPartIdx=0、refIdxLX、および currSubMbType を入力として起動され、出力は mvLX に割り当てられる。

記3 8.4.1.3 小小節から返された動きベクトル mvLX は、その処理が起動されたマクロブロックの全ての4×4サブマクロブロックパーティションにおいて同一である。

予測利用フラグ predFlagL0 と predFlagL1 は、表 8-9/JT-H264 を用いて規定されるように導出される。

表8-9/JT-H264 予測利用フラグの割り当て  
(ITU-T H.264)

refIdxL0	refIdxL1	predFlagL0	predFlagL1
>=0	>=0	1	1
>=0	<0	1	0
<0	>=0	0	1

変数 subMvCnt は次の通り導出される。

- もし subMbPartIdx が 0 に等しくないならば、subMvCnt は 0 に等しく設定される。
- それ以外 (subMbPartIdx が 0 に等しい)、subMvCnt は predFlagL0+predFlagL1 に等しく設定される。

#### 8.4.1.2.3 時間方向直接輝度動きベクトルと参照インデックス予測モードの導出処理

この処理は、direct\_spatial\_mv\_pred\_flag が 0 であり、以下の条件のいずれかが真である時、起動される。

- mb\_type は B\_Skip と等しい。
- mb\_type は B\_Direct\_16x16 と等しい。
- sub\_mb\_type[mbPartIdx] は B\_Direct\_8x8 と等しい。

この処理の入力は、mbPartIdx と subMbPartIdx である。

この処理の出力は、動きベクトル mvL0 と mvL1、参照インデックス refIdxL0 と refIdxL1、および予測リスト利用フラグ predFlagL0 と predFlagL1 である。

8.4.1.2.1 小小節で規定される処理が、mbPartIdx、subMbPartIdx を入力として与えられて起動され、出力は colPic、mbAddrCol、mvCol、refIdxCol、および vertMvScale に割り当てられる。

参照インデックス refIdxL0 と refIdxL1 は、以下の通り導出される。

$$\text{refIdxL0} = ((\text{refIdxCol} < 0) ? 0 : \text{MapColToList0}(\text{refIdxCol})) \quad (8-196)$$

$$\text{refIdxL1} = 0 \quad (8-197)$$

記1 現マクロブロックがフィールドマクロブロックなら、refIdxL0 と refIdxL1 はフィールドリストを索引する。それ以外は(現マクロブロックがフレームマクロブロック)、refIdxL0 と refIdxL1 はフレームか相補的参照フィールドペアのリストを索引する。

refPicCol を、ピクチャ colPic 内の共通位置マクロブロック mbAddrCol を復号する時に、参照インデックス refIdxCol で参照されたフレーム、フィールド、または相補的フィールドペアとする。関数 MapColToList0(refIdxCol) は、次の通り規定される。

- もし `vertMvScale` が `One_To_One` と等しいならば、次が適用される。
- もし `field_pic_flag` が 0 に等しく、現マクロブロックがフィールドマクロブロックならば、次が適用される
  - `refIdxL0Frm` を、フィールド `refPicCol` を含む、フレームまたは相補的フィールドペアを参照する現参照ピクチャリスト `RefPicList0` の最小の参照インデックス値とする。`RefPicList0` は、フィールド `refPicCol` を含むフレームまたは相補的フィールドペアを含まなければならない。`MapColToList0()` の戻り値は次の通り規定される。
    - もし `refIdxCol` で参照されるフィールドが現マクロブロックと同じパリティをもつならば、`MapColToList0(refIdxCol)` は参照インデックス(`refIdxL0Frm<<1`)を返す。
    - それ以外 (`refIdxCol` で参照されるフィールドが現マクロブロックと逆パリティをもつ)、`MapColToList0(refIdxCol)` は参照インデックス(`(refIdxL0Frm<<1)+1`)を返す。
  - それ以外 (`field_pic_flag` が 1 に等しいか、現マクロブロックがフレームマクロブロック)、`MapColToList0(refIdxCol)` は、`refPicCol` を参照する現参照ピクチャリスト `RefPicList0` の最小の参照インデックス値 `refIdxL0` を返す。`RefPicList0` は、`refPicCol` を含まなければならない。
- それ以外、もし `vertMvScale` が `Frm_To_Fld` と等しいならば、次が適用される。
  - もし `field_pic_flag` が 0 に等しいならば、`refIdxL0Frm` を、`refPicCol` を参照する現参照ピクチャリスト `RefPicList0` の最小の参照インデックス値とする。`MapColToList0(refIdxCol)` は参照インデックス (`refIdxL0Frm<<1`)を返す。`RefPicList0` は、`refPicCol` を含まなければならない。
  - それ以外 (`field_pic_flag` が 1 に等しい)、`MapColToList0(refIdxCol)` は、現ピクチャ `CurrPic` と同じパリティをもつ `refPicCol` のフィールドを参照する、現参照ピクチャリスト `RefPicList0` の最小の参照インデックス値 `refIdxL0` を返す。`RefPicList0` は、現ピクチャ `CurrPic` と同じパリティをもつ `refPicCol` のフィールドを含まなければならない。
  - それ以外(`vertMvScale` は `Fld_To_Frm` と等しい)、`MapColToList0(refIdxCol)` は、`refPicCol` を含むフレームまたは相補的フィールドペアを参照する、現参照ピクチャリスト `RefPicList0` の最小の参照インデックス値 `refIdxL0` を返す。`RefPicList0` は、フィールド `refPicCol` を含むフレームまたは相補的フィールドペアを含まなければならない。

記2 共通位置マクロブロックを含むピクチャの復号処理で参照された時に“短期間参照使用”とマークされた復号参照ピクチャは、現マクロブロックの直接予測モードを使用するINTER予測の参照のために使用される前に、“長期間参照使用”とマークされるよう変更されていることがある。

`vertMvScale` の値により、`mvCol` の垂直成分は次の通り変更される。

- もし `vertMvScale` が `Frm_To_Fld` と等しいならば
 
$$mvCol[1]=mvCol[1]/2 \quad (8-198)$$
- それ以外、もし `vertMvScale` が `Fld_To_Frm` と等しいならば
 
$$mvCol[1]=mvCol[1]*2 \quad (8-199)$$
- それ以外(`vertMvScale` が `One_To_One` と等しい)、`mvCol[1]` は変化しないままである。

変数 currPicOrField、pic0、そして pic1 は次の通り導出される。

- もし field\_pic\_flag が 0 に等しく現マクロブロックがフィールドマクロブロックならば、次が適用される。
  - currPicOrField は、現マクロブロックと同じパリティを持つ、現ピクチャ CurrPic のフィールドである。
  - pic1 は、現マクロブロックと同じパリティを持つ RefPicList1[0]のフィールドである。
  - 変数 pic0 は次の通り導出される。
    - もし refIdxL0%2 が 0 に等しいならば、pic0 は現マクロブロックと同じパリティを持つ、RefPicList0[refIdxL0/2]のフィールドである。
    - それ以外 (refIdxL0%2 は 0 に等しくない)、pic0 は現マクロブロックと逆パリティを持つ、RefPicList0[refIdxL0/2]のフィールドである。
- それ以外 (field\_pic\_flag が 1 に等しい、または現マクロブロックがフレームマクロブロック)、currPicOrField は現ピクチャ CurrPic であり、pic1 は復号参照ピクチャ RefPicList1[0]であり、そして pic0 は復号参照ピクチャ RefPicList0[refIdxL0]である。

現マクロブロックの 4×4 サブマクロブロックパーティションそれぞれの 2 つの動きベクトル mvL0 と mvL1 は、次の通り導出される：

記3 4×4サブマクロブロックパーティションの多くが、同じ動きベクトルと参照ピクチャを共有することがしばしばある。これらの場合、時間方向直接モード動き補償は、4×4輝度サンプルブロックよりも大きい単位でINTER予測サンプル値を計算することができる。例えば、direct\_8x8\_inference\_flagが1である時、少なくともそれぞれの、マクロブロック4分割、8×8輝度サンプルでは、同じ動きベクトルと参照ピクチャを共有する。

- もし参照インデックスrefIdxL0が長期間参照ピクチャを参照するか、またはDiffPicOrderCnt(pic1,pic0)が0に等しいならば、直接モードパーティションの動きベクトルmvL0、mvL1は以下のように導出される。

$$mvL0=mvCol \quad (8-200)$$

$$mvL1=0 \quad (8-201)$$

- それ以外、動きベクトルmvL0、mvL1は共通位置サブマクロブロックパーティションの動きベクトルmvColをスケーリングしたものとして、以下で規定される通り導出される。(図8-2/JT-H264参照)

$$tx=(16384+Abs(td/2))/td \quad (8-202)$$

$$DistScaleFactor=Clip3(-1024,1023,(tb*tx+32)>>6) \quad (8-203)$$

$$mvL0=(DistScaleFactor*mvCol+128)>>8 \quad (8-204)$$

$$mvL1=mvL0-mvCol \quad (8-205)$$

ここでtbとtdは、次の通り導出される。

$$tb=Clip3(-128,127,DiffPicOrderCnt(currPicOrField,pic0)) \quad (8-206)$$

$$td=Clip3(-128,127,DiffPicOrderCnt(pic1,pic0)) \quad (8-207)$$

記4 mvL0とmvL1は付属資料Aで規定される範囲を超えることはできない。

予測利用フラグ  $\text{predFlagL0}$  と  $\text{predFlagL1}$  は、ともに 1 に設定される。

図 8-2/JT-H264 は、現ピクチャが時間的に参照ピクチャリスト 0 からの参照ピクチャと参照ピクチャリスト 1 からの参照ピクチャ間にある時、時間方向直接モードの動きベクトル推定を示す。

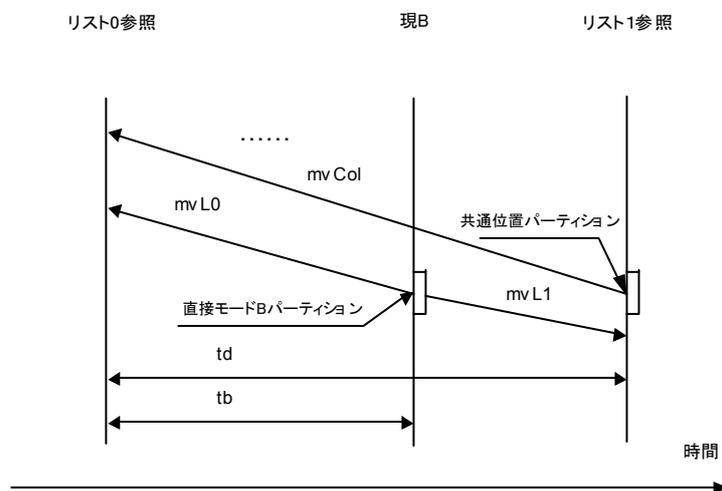


図8-2/JT-H264 時間方向直接モードの動きベクトル推定の例(参考)  
(ITU-T H.264)

#### 8.4.1.3 輝度動きベクトル予測の導出処理

この処理への入力は、以下の通りである。

- マクロブロックパーティションインデックス  $\text{mbPartIdx}$
- サブマクロブロックパーティションインデックス  $\text{subMbPartIdx}$
- 現パーティションの参照インデックス  $\text{refIdxLX}$  ( $X$  は 0 または 1)
- 変数  $\text{currSubMbType}$

この処理の出力は動きベクトル  $\text{mvLX}$  の予測  $\text{mvpLX}$  ( $X$  は 0 または 1) である。

8.4.1.3.2 小小節で、隣接ブロックの動きデータの導出処理が、 $\text{mbPartIdx}$ 、 $\text{subMbPartIdx}$ 、 $\text{currSubMbType}$  と  $\text{listSuffixFlag=X}$  ( $\text{refIdxLX}$  が各々  $\text{refIdxL0}$  または  $\text{refIdxL1}$  になることに対し、 $X$  は 0 または 1) を入力とし、 $N$  が A、B または C で置き換えられる  $\text{mbAddrN}\neq\text{mbPartIdxN}\neq\text{subMbPartIdxN}$ 、参照インデックス  $\text{refIdxLXN}$  と動きベクトル  $\text{mvLXN}$  を出力として起動される。

8.4.1.3.1 小小節のメディア輝度動きベクトル予測の導出処理が、以下の 1 つが真の場合を除いて、 $N$  が A、B または C で置き換えられる  $\text{mbAddrN}\neq\text{mbPartIdxN}\neq\text{subMbPartIdxN}$ 、 $\text{mvLXN}$ 、 $\text{refIdxLXN}$  と  $\text{refIdxLX}$  を入力とし、 $\text{mvpLX}$  を出力として起動される。

- $\text{MbPartWidth}(\text{mb\_type})$  が 16、 $\text{MbPartHeight}(\text{mb\_type})$  が 8、 $\text{mbPartIdx}$  が 0、かつ  $\text{refIdxLXB}$  が  $\text{refIdxLX}$  と等しい。

$$\text{mvpLX}=\text{mvLXB} \quad (8-208)$$

- MbPartWidth(mb\_type)が 16、MbPartHeight(mb\_type)が 8、mbPartIdx が 1、かつ refldxLXA が refldxLX と等しい。

$$\text{mvpLX}=\text{mvLXA} \quad (8-209)$$

- MbPartWidth(mb\_type)が 8、MbPartHeight(mb\_type)が 16、mbPartIdx が 0、かつ refldxLXA が refldxLX と等しい。

$$\text{mvpLX}=\text{mvLXA} \quad (8-210)$$

- MbPartWidth(mb\_type)が 8、MbPartHeight(mb\_type)が 16、mbPartIdx が 1、かつ refldxLXC が refldxLX と等しい。

$$\text{mvpLX}=\text{mvLXC} \quad (8-211)$$

図 8-3/JT-H264 は上の 4 つの箇条書き項目で規定される非メディアン予測を示す。

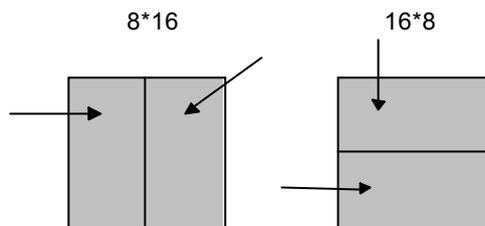


図8-3/JT-H264 方向性分割予測 (参考)  
(ITU-T H.264)

#### 8.4.1.3.1 メディアン輝度動きベクトル予測の導出処理

この処理への入力は以下の通りである。

- 隣接パーティション  $\text{mbAddrN}\nabla\text{mbPartIdxN}\nabla\text{subMbPartIdxN}$  (N は A、B または C で置換)
- 隣接パーティションの動きベクトル  $\text{mvLXN}$  (N は A、B または C で置換)
- 隣接パーティションの参照インデックス  $\text{refIdxLXN}$  (N は A、B または C で置換)
- 現パーティションの参照インデックス  $\text{refIdxLX}$

この処理の出力は動きベクトル予測  $\text{mvpLX}$  である。

変数  $\text{mvpLX}$  は次の通り導出される。

- パーティション  $\text{mbAddrB}\nabla\text{mbPartIdxB}\nabla\text{subMbPartIdxB}$  と  $\text{mbAddrC}\nabla\text{mbPartIdxC}\nabla\text{subMbPartIdxC}$  の両方が利用不可であり、 $\text{mbAddrA}\nabla\text{mbPartIdxA}\nabla\text{subMbPartIdxA}$  が利用可能である時、

$$\text{mvLXB}=\text{mvLXA} \quad (8-212)$$

$$\text{mvLXC}=\text{mvLXA} \quad (8-213)$$

$$\text{refIdxLXB}=\text{refIdxLXA} \quad (8-214)$$

$$\text{refIdxLXC}=\text{refIdxLXA} \quad (8-215)$$

- 参照インデックス  $\text{refIdxLXA}$ 、 $\text{refIdxLXB}$  または  $\text{refIdxLXC}$  に従って、次が適用される。
  - もし参照インデックス  $\text{refIdxLXA}$ 、 $\text{refIdxLXB}$  または  $\text{refIdxLXC}$  の 1 つだけが現パーティションの参照インデックス  $\text{refIdxLX}$  と等しいならば、以下が適用される。 $\text{refIdxLXN}$  を  $\text{refIdxLX}$  と等しい参照インデックスとする場合、動きベクトル  $\text{mvLXN}$  が動きベクトル予測  $\text{mvpLX}$  に割り当てられる。

$$\text{mvpLX}=\text{mvLXN} \quad (8-216)$$

- それ以外、動きベクトル予測  $\text{mvpLX}$  の各成分は、動きベクトル  $\text{mvLXA}$ 、 $\text{mvLXB}$  と  $\text{mvLXC}$  の対応するベクトル成分のメディアンによって与えられる。

$$\text{mvpLX}[0]=\text{Median}(\text{mvLXA}[0],\text{mvLXB}[0],\text{mvLXC}[0]) \quad (8-217)$$

$$\text{mvpLX}[1]=\text{Median}(\text{mvLXA}[1],\text{mvLXB}[1],\text{mvLXC}[1]) \quad (8-218)$$

#### 8.4.1.3.2 隣接パーティションの動きデータの導出処理

この処理の入力は以下の通りである。

- マクロブロックパーティションインデックス  $\text{mbPartIdx}$
- サブマクロブロックパーティションインデックス  $\text{subMbPartIdx}$
- 現サブマクロブロックタイプ  $\text{currSubMbType}$
- リスト接尾辞フラグ  $\text{listSuffixFlag}$

この処理の出力は以下の通りである(N は A、B または C で置換)。

- 隣接パーティションを規定する  $\text{mbAddrN}\nabla\text{mbPartIdxN}\nabla\text{subMbPartIdxN}$
- 隣接パーティションの動きベクトル  $\text{mvLXN}$
- 隣接パーティションの参照インデックス  $\text{refIdxLXN}$

文字列“LX”を含む変数名は、X が  $\text{listSuffixFlag}$  に等しくなると解釈される。

N が A、B または C のいずれかであるパーティション  $\text{mbAddrN}\nabla\text{mbPartIdxN}\nabla\text{subMbPartIdxN}$  は、以下の順序付けられたステップで導出される。

1.  $\text{mbAddrD}\nabla\text{mbPartIdxD}\nabla\text{subMbPartIdxD}$  を、追加隣接パーティションを規定する変数とする。
  - 6.4.10.7 小節の処理が、 $\text{mbPartIdx}$ 、 $\text{currSubMbType}$  と  $\text{subMbPartIdx}$  を入力として起動され、出力には N が A、B、C または D で置き換えられる  $\text{mbAddrN}\nabla\text{mbPartIdxN}\nabla\text{subMbPartIdxN}$  が割り当てられる。
2. パーティション  $\text{mbAddrC}\nabla\text{mbPartIdxC}\nabla\text{subMbPartIdxC}$  が利用不可である時、次が適用される。

$$\text{mbAddrC}=\text{mbAddrD} \quad (8-219)$$

$$\text{mbPartIdxC}=\text{mbPartIdxD} \quad (8-220)$$

$$\text{subMbPartIdxC}=\text{subMbPartIdxD} \quad (8-221)$$

動きベクトル  $\text{mvLXN}$  と参照インデックス  $\text{refIdxLXN}$  ( $N$  は A, B または C である) は、次の通り導出される。

- もしマクロブロックパーティションまたはサブマクロブロックパーティション  $\text{mbAddrN} \neq \text{mbPartIdxN} \neq \text{subMbPartIdxN}$  が利用不可である、あるいは  $\text{mbAddrN}$  が INTRA 予測モードで符号化された、あるいは  $\text{mbAddrN} \neq \text{mbPartIdxN} \neq \text{subMbPartIdxN}$  の  $\text{predFlagLX}$  が 0 ならば、 $\text{mvLXN}$  の両方の成分は 0 に設定され、 $\text{refIdxLXN}$  は -1 に設定される。
- それ以外、次が適用される。
  - 動きベクトル  $\text{mvLXN}$  と参照インデックス  $\text{refIdxLXN}$  は各々  $\text{MvLX}[\text{mbPartIdxN}][\text{subMbPartIdxN}]$  と  $\text{RefIdxLX}[\text{mbPartIdxN}]$  に等しく設定される。これらは(サブ)マクロブロックパーティション  $\text{mbAddrN} \neq \text{mbPartIdxN} \neq \text{subMbPartIdxN}$  に割り当てられていた動きベクトル  $\text{mvLX}$  と参照インデックス  $\text{refIdxLX}$  である。
  - 変数  $\text{mvLXN}[1]$  と  $\text{refIdxLXN}$  は、次の通りさらに処理される。

- もし現マクロブロックがフィールドマクロブロックで、かつマクロブロック  $\text{mbAddrN}$  がフレームマクロブロックならば、

$$\text{mvLXN}[1]=\text{mvLXN}[1]/2 \quad (8-222)$$

$$\text{refIdxLXN}=\text{refIdxLXN} * 2 \quad (8-223)$$

- それ以外、もし現マクロブロックがフレームマクロブロックで、かつマクロブロック  $\text{mbAddrN}$  がフィールドマクロブロックならば、

$$\text{mvLXN}[1]=\text{mvLXN}[1] * 2 \quad (8-224)$$

$$\text{refIdxLXN}=\text{refIdxLXN} / 2 \quad (8-225)$$

- それ以外、垂直動きベクトル成分  $\text{mvLXN}[1]$  と参照インデックス  $\text{refIdxLXN}$  は変化しない。

#### 8.4.1.4 色差動きベクトルの導出処理

この処理は、 $\text{ChromaArrayType}$  が 0 に等しくない場合にのみ起動される。

この処理への入力輝度動きベクトル  $\text{mvLX}$  と参照インデックス  $\text{refIdxLX}$  である。

この処理の出力は色差動きベクトル  $\text{mvCLX}$  である。

色差動きベクトルは対応する輝度動きベクトルから導出される。

色差動きベクトル成分の精度は、水平に  $1 \div (4 * \text{SubWidthC})$  であり垂直に  $1 \div (4 * \text{SubHeightC})$  である。

記 例えば、4:2:0 色差フォーマットを使用する時、輝度動きベクトルの単位は4分の1輝度サンプル単位であり、色差は輝度に比べて半分の水平と垂直の解像度をもつので、色差動きベクトルの単位は8分の1色差サンプル単位である。すなわち、色差動きベクトルにおける1の値は、8分の1色差サンプル変位を言う。例えば輝度ベクトルが  $8 \times 16$  輝度サンプルに適用される時、対応する4:2:0 色差フォーマットの色差ベ

クトルは4×8色差サンプルに適用され、輝度ベクトルが4×4輝度サンプルに適用される時、対応する4:2:0色差フォーマットの色差ベクトルは2×2色差サンプルに適用される。

動きベクトル mvCLX の導出において、次が適用される。

- もし ChromaArrayType が 1 に等しくない、または現マクロブロックがフレームマクロブロックならば、色差動きベクトル mvCLX の水平および垂直成分は、以下で導出される。

$$\text{mvCLX}[0]=\text{mvLX}[0] \quad (8-226)$$

$$\text{mvCLX}[1]=\text{mvLX}[1] \quad (8-227)$$

- それ以外(ChromaArrayType が 1 に等しい、かつ現マクロブロックがフィールドマクロブロック)、色差動きベクトルの水平成分 mvCLX[0]のみが等式 8-218 を使って導出される。色差動きベクトルの垂直成分 mvCLX[1]は、現フィールドあるいは現マクロブロックのパリティと、参照インデックス refIdxLX で参照される参照ピクチャに依存する。mvCLX[1]は表 8-10/JT-H264 に従って mvLX[1]から導出される。

表8-10/JT-H264 フィールド符号化モードにおける色差ベクトル垂直成分の導出  
(ITU-T H.264)

パリティの状態		mvCLX[1]
参照ピクチャ(refIdxLX)	現フィールド(ピクチャ/マクロブロック)	
トップフィールド	ボトムフィールド	mvLX[1]+2
ボトムフィールド	トップフィールド	mvLX[1]-2
その他		mvLX[1]

#### 8.4.2 INTER予測サンプルの復号処理

この処理への入力は、以下の通りである。

- マクロブロックパーティション mbPartIdx
- サブマクロブロックパーティション subMbPartIdx
- パーティションの輝度と（利用可能であれば）色差に対する幅と高さを規定する変数 partWidth、partHeight、（利用可能であれば）partWidthC と（利用可能であれば）partHeightC
- 輝度動きベクトル mvL0、mvL1、および ChromaArrayType が 0 に等しくない場合には色差動きベクトル mvCL0、mvCL1
- 参照インデックス refIdxL0 と refIdxL1
- 予測リスト利用フラグ predFlagL0 と preeFlagL1
- 重み付け予測のための変数 logWDC、w0C、w1C、o0C、o1C の C を L に、および ChromaArrayType が 0 に等しくない時は Cb および Cr に置き換えたもの

この処理の出力は、以下の通りである。

- INTER 予測サンプル  $\text{predPart}$ 、すなわち予測輝度サンプルの  $(\text{partWidth}) \times (\text{partHeight})$  配列  $\text{predPart}_L$ 、および ChromaArrayType が 0 に等しくない場合には色差成分 Cb、Cr の各々に対する予測色差サンプルの 2 つの  $(\text{partWidthC}) \times (\text{partHeightC})$  配列  $\text{predPart}_{Cb}$  と  $\text{predPart}_{Cr}$ 。

ここで、 $\text{predPartL0}_L$  と  $\text{predPartL1}_L$  を予測輝度サンプル値の  $(\text{partWidth}) \times (\text{partHeight})$  配列、および ChromaArrayType が 0 に等しくない場合には  $\text{predPartL0}_{Cb}$ 、 $\text{predPartL1}_{Cb}$ 、 $\text{predPartL0}_{Cr}$  と  $\text{predPartL1}_{Cr}$  を予測色差サンプル値の  $(\text{partWidthC}) \times (\text{partHeightC})$  配列とする。

LX が L0 または L1 のいずれかに置き換えられた、変数  $\text{predFlagLX}$ 、 $\text{RefPicListX}$ 、 $\text{refIdxLX}$ 、 $\text{refPicLX}$ 、 $\text{predPartLX}$  において、以下が規定される。

$\text{predFlagLX}$  が 1 の時、次が適用される。

- 輝度サンプルの順序付けられた 2 次元配列  $\text{refPicLX}_L$  と ChromaArrayType が 0 に等しくない場合には色差サンプルの順序付けられた 2 つの 2 次元配列  $\text{refPicLX}_{Cb}$ 、 $\text{refPicLX}_{Cr}$  からなる参照ピクチャは、 $\text{refIdxLX}$ 、 $\text{RefPicListX}$  を入力として与えられ、8.4.2.1 小小節で規定された処理を起動することで導出される。
- 配列  $\text{predPartLX}_L$  と、ChromaArrayType が 0 に等しくない場合に配列  $\text{predPartLX}_{Cb}$  と  $\text{predPartLX}_{Cr}$  は、 $\text{mbPartIdx}$ 、 $\text{subMbPartIdx}$  で規定された現パーティション、動きベクトル  $\text{mvLX}$ 、(利用可能であれば)  $\text{mvCLX}$ 、および参照配列  $\text{refPicLX}_L$ 、(利用可能であれば)  $\text{refPicLX}_{Cb}$  および (利用可能であれば)  $\text{refPicLX}_{Cr}$  を入力として与えられ、8.4.2.2 小小節で規定された処理を起動することで導出される。

L、(利用可能であれば) Cb または (利用可能であれば) Cr で置き換えられる C に対して、成分 C の予測サンプルの配列  $\text{predPart}_C$  は、 $\text{mbPartIdx}$ 、 $\text{subMbPartIdx}$  で規定された現パーティション、予測利用フラグ  $\text{predFlagL0}$  と  $\text{predFlagL1}$ 、配列  $\text{predPartL0}_C$  と  $\text{predPartL1}_C$ 、および重み付け予測のための変数  $\log\text{WDC}$ 、 $w_{0C}$ 、 $w_{1C}$ 、 $o_{0C}$ 、 $o_{1C}$  を入力として与えられ、8.4.2.3 小小節で規定された処理を起動することで導出される。

#### 8.4.2.1 参照ピクチャ選択処理

この処理への入力は参照インデックス  $\text{refIdxLX}$  である。

この処理の出力は輝度サンプルの 2 次元配列  $\text{refPicLX}_L$  と、ChromaArrayType が 0 に等しくない時、色差サンプルの 2 つの 2 次元配列  $\text{refPicLX}_{Cb}$  と  $\text{refPicLX}_{Cr}$  からなる参照ピクチャである。

$\text{field\_pic\_flag}$  に依存して、(8.2.4 小節で規定されるように導出された) 参照ピクチャリスト  $\text{RefPicListX}$  は、以下からなる。

- もし  $\text{field\_pic\_flag}$  が 1 ならば、 $\text{RefPicListX}$  の各々のエントリは、参照フィールドまたは参照フレームのフィールドである。
- それ以外 ( $\text{field\_pic\_flag}$  が 0)、 $\text{RefPicListX}$  の各々のエントリは、参照フレームまたは相補的参照フィールドペアである。

参照ピクチャの導出において、次が適用される。

- もし  $\text{field\_pic\_flag}$  が 1 ならば、参照フィールドあるいは参照フレームのフィールド  $\text{RefPicListX}[\text{refIdxLX}]$  が出力である。出力する参照フィールドあるいは参照フレームのフィールドは、輝度サンプルの  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列  $\text{refPicLX}_L$  と、ChromaArrayType が 0 に等しくない場合

には色差サンプルの2つの $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$ 配列  $\text{refPicLX}_{Cb}$ 、 $\text{refPicLX}_{Cr}$  からなる。

- それ以外( $\text{field\_pic\_flag}$  が 0)、次が適用される。
  - もし現マクロブロックがフレームマクロブロックならば、参照フレームあるいは相補的参照フィールドペア  $\text{RefPicListX}[\text{refIdxLX}]$ が出力である。出力する参照フレームあるいは相補的参照フィールドペアは、輝度サンプルの $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$ 配列  $\text{refPicLX}_L$  と、 $\text{ChromaArrayType}$  が 0 に等しくない場合には色差サンプルの2つの $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$ 配列  $\text{refPicLX}_{Cb}$ 、 $\text{refPicLX}_{Cr}$  からなる。
  - それ以外(現マクロブロックがフィールドマクロブロック)、次が適用される。
    - $\text{refFrame}$  を参照フレームあるいは相補的参照フィールドペア  $\text{RefPicListX}[\text{refIdxLX}/2]$ とする。
    - $\text{refFrame}$  のフィールドは次の通り選択される。
      - もし  $\text{refIdxLX}\%2$  が 0 ならば、現マクロブロックと同じパリティをもつ  $\text{refFrame}$  のフィールドが出力である。
      - それ以外( $\text{refIdxLX}\%2$  が 1)、現マクロブロックの逆パリティをもつ  $\text{refFrame}$  のフィールドが出力である。
    - 出力する参照フィールドあるいは参照フレームのフィールドは、輝度サンプルの $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L/2)$ 配列  $\text{refPicLX}_L$  と、 $\text{ChromaArrayType}$  が 0 に等しくない場合には色差サンプルの2つの $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C/2)$ 配列  $\text{refPicLX}_{Cb}$ 、 $\text{refPicLX}_{Cr}$  からなる。

$\text{separate\_colour\_plane\_flag}$  に依存して、次が適用される。

- もし  $\text{separate\_colour\_plane\_flag}$  が 0 に等しいならば、参照ピクチャサンプル配列  $\text{refPicLX}_L$ 、(利用可能であれば)  $\text{refPicLX}_{Cb}$ 、(利用可能であれば)  $\text{refPicLX}_{Cr}$  は、以前に復号された参照フィールド、または参照フレーム、または相補的参照フィールドペア、または参照フレームのフィールドに対して、8.7 節で導出された復号サンプル配列  $S_L$ 、(利用可能であれば)  $S_{Cb}$ 、(利用可能であれば)  $S_{Cr}$  に対応する。

それ以外 ( $\text{separate\_colour\_plane\_flag}$  が 1 に等しい)、次が適用される。

- もし  $\text{colour\_plane\_id}$  が 0 に等しいならば、参照ピクチャサンプル配列  $\text{refPicLX}_L$  は、以前に復号された参照フィールド、または参照フレーム、または相補的参照フィールドペア、または参照フレームのフィールドに対して、8.7 節で導出された復号サンプル配列  $S_L$  に対応する。
- それ以外、もし  $\text{colour\_plane\_id}$  が 1 に等しいならば、参照ピクチャサンプル配列  $\text{refPicLX}_L$  は、以前に復号された参照フィールド、または参照フレーム、または相補的参照フィールドペア、または参照フレームのフィールドに対して、8.7 節で導出された復号サンプル配列  $S_{Cb}$  に対応する。
- それ以外 ( $\text{colour\_plane\_id}$  が 2 に等しい)、参照ピクチャサンプル配列  $\text{refPicLX}_L$  は、以前に復号された参照フィールド、または参照フレーム、または相補的参照フィールドペア、または参照フレームのフィールドに対して、8.7 節で導出された復号サンプル配列  $S_{Cr}$  に対応する。

### 8.4.2.2 分数サンプル補間処理

この処理への入力は、以下の通りである。

- そのパーティションインデックス  $mbPartIdx$  とそのサブマクロブロックパーティションインデックス  $subMbPartIdx$  で定義される現パーティション
- 輝度サンプル単位で表されたこのパーティションの幅と高さ  $partWidth$ 、 $partHeight$
- 4分の1輝度サンプル単位で与えられた輝度動きベクトル  $mvLX$
- $ChromaArrayType$  が 0 に等しくない時、水平方向に $(4*SubWidthC)$ 分の1色差サンプル単位および垂直方向に $(4*SubHeightC)$ 分の1色差サンプル単位の精度をもつ色差動きベクトル  $mvCLX$ 、そして
- 選択された参照ピクチャサンプル配列  $refPicLX_L$  および、 $ChromaArrayType$  が 0 に等しくない時、 $refPicLX_{Cb}$  と  $refPicLX_{Cr}$

この処理の出力は、以下の通りである。

- 予測輝度サンプル値の $(partWidth) \times (partHeight)$ 配列  $predPartLX_L$
- $ChromaArrayType$  が 0 に等しくない場合には予測色差サンプル値の2つの $(partWidthC) \times (partHeightC)$ 配列  $predPartLX_{Cb}$  と  $predPartLX_{Cr}$

$(xA_L, yA_L)$ を、与えられた輝度サンプルの2次元配列の左上の輝度サンプルからの相対位置として、 $mbPartIdx \neq subMbPartIdx$  で定義される現パーティションの左上の輝度サンプル位置を、整数サンプル単位で表した位置とする。

$(xInt_L, yInt_L)$ を整数サンプル単位で表される輝度位置、 $(xFrac_L, yFrac_L)$ を4分の1サンプル単位で表されるオフセットとする。これらの変数は、参照サンプル配列  $refPicLX_L$ 、(利用可能であれば)  $refPicLX_{Cb}$  と (利用可能であれば)  $refPicLX_{Cr}$  中の一般の分数サンプル位置を規定するために、この小小節の中でのみ使われる。

予測輝度サンプル配列  $predPartLX_L$  中の各輝度サンプル位置 $(0 \leq x_L < partWidth, 0 \leq y_L < partHeight)$ に対して、対応する予測輝度サンプル値  $predPartLX_L[x_L, y_L]$ は次の通り導出される。

- 変数  $xInt_L$ 、 $yInt_L$ 、 $xFrac_L$  と  $yFrac_L$  は以下で導出される。

$$xInt_L = xA_L + (mvLX[0] \gg 2) + x_L \quad (8-228)$$

$$yInt_L = yA_L + (mvLX[1] \gg 2) + y_L \quad (8-229)$$

$$xFrac_L = mvLX[0] \& 3 \quad (8-230)$$

$$yFrac_L = mvLX[1] \& 3 \quad (8-231)$$

- 予測輝度サンプル値  $predPartLX_L[x_L, y_L]$ は、 $(xInt_L, yInt_L)$ 、 $(xFrac_L, yFrac_L)$ と  $refPicLX_L$  を入力として与えられ、8.4.2.2.1 小小節で規定された処理を起動することで導出される。

$ChromaArrayType$  が 0 に等しくない場合は、次が適用される。

$(xInt_C, yInt_C)$ を整数サンプル単位で表された色差位置、 $(xFrac_C, yFrac_C)$ を水平方向に $(4*SubWidthC)$ 分の1色差サンプル単位および垂直方向に $(4*SubHeightC)$ 分の1色差サンプル単位で表されたオフセットとする。これらの変数は、参照サンプル配列  $refPicLX_{Cb}$  と  $refPicLX_{Cr}$  中の一般の分数サンプル位置を規定するために、この小小節の中でのみ使われる。

予測色差サンプル配列  $\text{predPartLXC}_b$ 、 $\text{predPartLXC}_r$  中の各色差サンプル位置 ( $0 \leq x_c < \text{partWidthC}$ ,  $0 \leq y_c < \text{partHeightC}$ ) に対して、対応する予測色差サンプル値  $\text{predPartLXC}_b[x_c, y_c]$ 、 $\text{predPartLXC}_r[x_c, y_c]$  は以下のように導出される。

- ChromaArrayType に依存して、変数  $x\text{Int}_c$ 、 $y\text{Int}_c$ 、 $x\text{Frac}_c$  と  $y\text{Frac}_c$  は次の通り導出される。

- ChromaArrayType が 1 に等しいならば、

$$x\text{Int}_c = (xA_L / \text{SubWidthC}) + (\text{mvCLX}[0] \gg 3) + x_c \quad (8-232)$$

$$y\text{Int}_c = (yA_L / \text{SubHeightC}) + (\text{mvCLX}[1] \gg 3) + y_c \quad (8-233)$$

$$x\text{Frac}_c = \text{mvCLX}[0] \& 7 \quad (8-234)$$

$$y\text{Frac}_c = \text{mvCLX}[1] \& 7 \quad (8-235)$$

- それ以外、ChromaArrayType が 2 に等しいならば、

$$x\text{Int}_c = (xA_L / \text{SubWidthC}) + (\text{mvCLX}[0] \gg 3) + x_c \quad (8-236)$$

$$y\text{Int}_c = (yA_L / \text{SubHeightC}) + (\text{mvCLX}[1] \gg 2) + y_c \quad (8-237)$$

$$x\text{Frac}_c = \text{mvCLX}[0] \& 7 \quad (8-238)$$

$$y\text{Frac}_c = (\text{mvCLX}[1] \& 3) \ll 1 \quad (8-239)$$

- それ以外 (ChromaArrayType が 3 に等しい)、

$$x\text{Int}_c = xA_L + (\text{mvLX}[0] \gg 2) + x_c \quad (8-240)$$

$$y\text{Int}_c = yA_L + (\text{mvLX}[1] \gg 2) + y_c \quad (8-241)$$

$$x\text{Frac}_c = (\text{mvCX}[0] \& 3) \quad (8-242)$$

$$y\text{Frac}_c = (\text{mvCX}[1] \& 3) \quad (8-243)$$

- ChromaArrayType に依存して、次が適用される。

- もし ChromaArrayType が 3 に等しくないならば、次が適用される。

- 予測サンプル値  $\text{predPartLXC}_b[x_c, y_c]$  は  $(x\text{Int}_c, y\text{Int}_c)$ 、 $(x\text{Frac}_c, y\text{Frac}_c)$  と  $\text{refPicLXC}_b$  を入力として与えられ、8.4.2.2.2 小小小節で規定された処理を起動することで導出される。

- 予測サンプル値  $\text{predPartLXC}_r[x_c, y_c]$  は  $(x\text{Int}_c, y\text{Int}_c)$ 、 $(x\text{Frac}_c, y\text{Frac}_c)$  と  $\text{refPicLXC}_r$  を入力として与えられ、8.4.2.2.2 小小小節で規定された処理を起動することで導出される。

- それ以外 (ChromaArrayType が 3 に等しい)、次が適用される。

- 予測サンプル値  $\text{predPartLXC}_b[x_c, y_c]$  は  $(x\text{Int}_c, y\text{Int}_c)$ 、 $(x\text{Frac}_c, y\text{Frac}_c)$  と  $\text{refPicLXC}_b$  を入力として与えられ、8.4.2.2.1 小小小節で規定された処理を起動することで導出される。

- 予測サンプル値  $\text{predPartLXC}_r[x_c, y_c]$  は  $(x\text{Int}_c, y\text{Int}_c)$ 、 $(x\text{Frac}_c, y\text{Frac}_c)$  と  $\text{refPicLXC}_r$  を入力として与えられ、8.4.2.2.1 小小小節で規定された処理を起動することで導出される。

#### 8.4.2.2.1 輝度サンプル補間処理

この処理への入力は、以下の通りである。

- 整数サンプル単位の輝度位置( $x_{Int_L}, y_{Int_L}$ )
- 分数サンプル単位の輝度位置のオフセット( $x_{Frac_L}, y_{Frac_L}$ )
- 選択された参照ピクチャの輝度サンプル配列  $refPicLX_L$

この処理の出力は予測輝度サンプル値  $predPartLX_L[x_L, y_L]$  である。

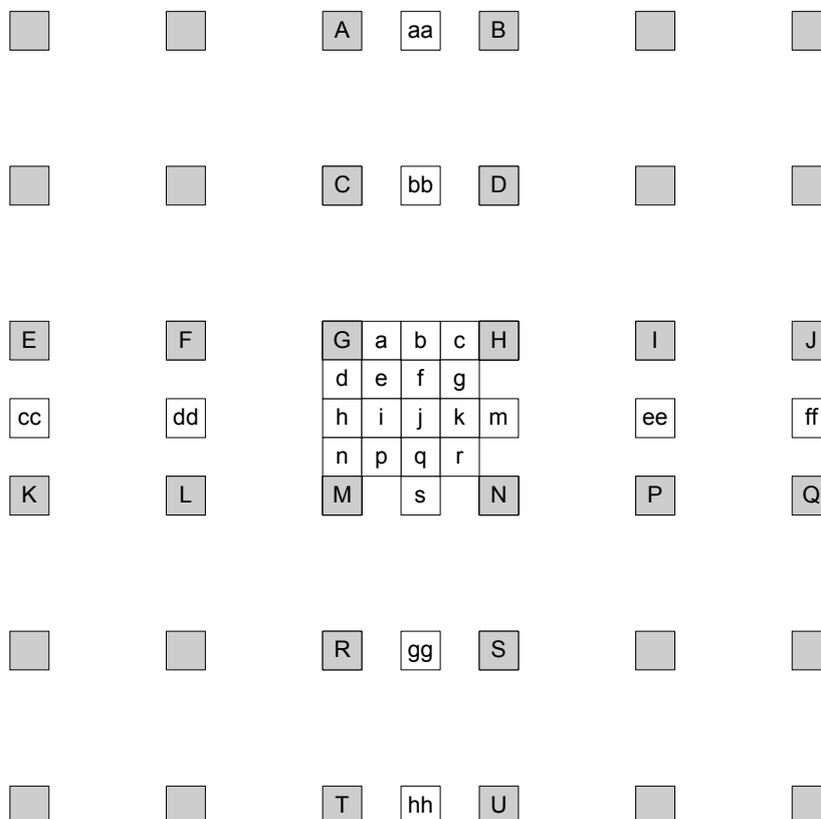


図8-4/JT-H264 4分の1サンプル輝度補間における整数サンプル(大文字の網掛けブロック)と分数サンプル位置(小文字の網掛けされていないブロック)  
(ITU-T H.264)

有効な参照ピクチャ輝度配列の高さである変数  $refPicHeightEffective_L$  は次の通り導出される。

- もし  $MbaffFrameFlag$  が 0 に等しい、または  $mb\_field\_decoding\_flag$  が 0 に等しいならば、 $refPicHeightEffective_L$  は  $PicHeightInSamples_L$  に等しく設定される。
- それ以外 ( $MbaffFrameFlag$  が 1 に等しい、かつ  $mb\_field\_decoding\_flag$  が 1 に等しい)、 $refPicHeightEffective_L$  は  $PicHeightInSamples_L/2$  に等しく設定される。

図 8-4/JT-H264 では、網掛けブロックの大文字で表された位置は、与えられた輝度サンプルの 2 次元配列  $refPicLX_L$  中の整数サンプル位置での輝度サンプルを表す。これらのサンプルは、予測輝度サンプル値  $predPartLX_L[x_L, y_L]$  の生成に用いることができる。与えられた輝度サンプルの配列  $refPicLX_L$  中の、輝度サンプル  $Z$  の各々に対応する位置 ( $x_{Z_L}, y_{Z_L}$ ) は、以下のように導出される。ここで、 $Z$  は A、B、C、D、E、F、G、H、I、J、K、L、M、N、P、Q、R、S、T または U である。

$$xZ_L = \text{Clip3}(0, \text{PicWidthInSamples}_L - 1, x\text{Int}_L + xDZ_L)$$

$$yZ_L = \text{Clip3}(0, \text{refPicHeightEffective}_L - 1, y\text{Int}_L + yDZ_L) \quad (8-244)$$

表 8-11/JT-H264 は Z の個々の置き換えに対する(xDZ<sub>L</sub>,yDZ<sub>L</sub>)を規定する。

表8-11/JT-H264 差分で表した整数輝度サンプル位置  
(ITU-T H.264)

Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	P	Q	R	S	T	U
xDZ <sub>L</sub>	0	1	0	1	-2	-1	0	1	2	3	-2	-1	0	1	2	3	0	1	0	1
yDZ <sub>L</sub>	-2	-2	-1	-1	0	0	0	0	0	0	1	1	1	1	1	1	2	2	3	3

整数サンプル位置(xA<sub>L</sub>,yA<sub>L</sub>)から(xU<sub>L</sub>,yU<sub>L</sub>)までの輝度サンプル ‘A’ から ‘U’ が与えられると、分数サンプル位置での輝度サンプル ‘a’ から ‘s’ は以下の規則で導出される。2 分の 1 サンプル位置での輝度予測値は、タップ値が(1,-5,20,20,-5,1)の 6 タップフィルタを適用することで導出される。4 分の 1 サンプル位置での輝度予測値は、整数サンプルと 2 分の 1 サンプル位置のサンプルを平均することで導出される。各分数位置の処理は、以下で記載される。

- b で表される 2 分の 1 サンプル位置でのサンプルは、まず水平方向の最近傍整数位置サンプルに 6 タップのフィルタをかけることで、b<sub>1</sub> で示される中間値を計算することによって、導出される。h で表される 2 分の 1 サンプル位置でのサンプルは、まず垂直方向の最近傍整数位置サンプルに 6 タップのフィルタをかけることで、h<sub>1</sub> で示される中間値を計算することによって、導出される。

$$b_1 = (E - 5 * F + 20 * G + 20 * H - 5 * I + J) \quad (8-245)$$

$$h_1 = (A - 5 * C + 20 * G + 20 * M - 5 * R + T) \quad (8-246)$$

最終的な予測値 b と h は以下を使って導出される。

$$b = \text{Clip1}_Y((b_1 + 16) \gg 5) \quad (8-247)$$

$$h = \text{Clip1}_Y((h_1 + 16) \gg 5) \quad (8-248)$$

- j で表された 2 分の 1 サンプル位置でのサンプルは、いずれの式を使っても同じ結果を生じるから、まず水平方向あるいは垂直方向いずれかの最近傍の 2 分の 1 サンプル位置の中間値に 6 タップのフィルタをかけることで、j<sub>1</sub> で示される中間値を計算することによって、導出される。

$$j_1 = cc - 5 * dd + 20 * h_1 + 20 * m_1 - 5 * ee + ff \quad (8-249)$$

または

$$j_1 = aa - 5 * bb + 20 * b_1 + 20 * s_1 - 5 * gg + hh \quad (8-250)$$

ここで、aa、bb、gg、s<sub>1</sub> と hh で示される中間値は、b<sub>1</sub> の導出と等価な方法で水平方向に 6 タップフィルタをかけることで導出され、cc、dd、ee、m<sub>1</sub> と ff で示された中間値は、h<sub>1</sub> の導出と等価な方法で垂直方向に 6 タップフィルタをかけることで導出される。最終的な予測値 j は以下を使って導出される。

$$j = \text{Clip1}_Y((j_1 + 512) \gg 10) \quad (8-251)$$

- 最終的な予測値 s と m は、b と h の導出と等価な方法で、以下で与えられるように、s<sub>1</sub> と m<sub>1</sub> から導出される。

$$s=\text{Clip1}_Y((s_1+16)\gg 5) \quad (8-252)$$

$$m=\text{Clip1}_Y((m_1+16)\gg 5) \quad (8-253)$$

- a、c、d、n、f、i、k と q で表される 4 分の 1 サンプル位置でのサンプルは、以下を使って整数サンプル位置と 2 分の 1 サンプル位置での 2 つの最近傍サンプルを平均し、切り上げることによって導出される。

$$a=(G+b+1)\gg 1 \quad (8-254)$$

$$c=(H+b+1)\gg 1 \quad (8-255)$$

$$d=(G+h+1)\gg 1 \quad (8-256)$$

$$n=(M+h+1)\gg 1 \quad (8-257)$$

$$f=(b+j+1)\gg 1 \quad (8-258)$$

$$i=(h+j+1)\gg 1 \quad (8-259)$$

$$k=(j+m+1)\gg 1 \quad (8-260)$$

$$q=(j+s+1)\gg 1 \quad (8-261)$$

- e、g、p と r で表される 4 分の 1 サンプル位置でのサンプルは、以下を使って対角方向の 2 分の 1 サンプル位置での 2 つの最近傍サンプルを平均し、切り上げることによって導出される。

$$e=(b+h+1)\gg 1 \quad (8-262)$$

$$g=(b+m+1)\gg 1 \quad (8-263)$$

$$p=(h+s+1)\gg 1 \quad (8-264)$$

$$r=(m+s+1)\gg 1 \quad (8-265)$$

分数サンプル単位の輝度位置のオフセット(xFrac<sub>L</sub>,yFrac<sub>L</sub>)は、整数サンプル位置と分数サンプル位置で作成されたどの輝度サンプルが、予測輝度サンプル値 predPartLX<sub>L</sub>[x<sub>L</sub>,y<sub>L</sub>]に割り当てられるかを規定する。この割り当ては表 8-12/JT-H264 に従って行われる。出力は predPartLX<sub>L</sub>[x<sub>L</sub>,y<sub>L</sub>]の値である。

表8-12/JT-H264 輝度予測サンプルpredPartLX<sub>L</sub>[x<sub>L</sub>,y<sub>L</sub>]の割り当て  
(ITU-T H.264)

xFrac <sub>L</sub>	0	0	0	0	1	1	1	1	2	2	2	2	3	3	3	3
yFrac <sub>L</sub>	0	1	2	3	0	1	2	3	0	1	2	3	0	1	2	3
predPartLX <sub>L</sub> [x <sub>L</sub> ,y <sub>L</sub> ]	G	d	h	n	a	e	i	p	b	f	j	q	c	g	k	r

#### 8.4.2.2.2 色差サンプル補間処理

この処理は、ChromaArrayType が 1 または 2 に等しい時のみ起動される。

この処理への入力は以下の通りである。

- 整数サンプル単位の色差位置(xInt<sub>C</sub>,yInt<sub>C</sub>)
- 分数サンプル単位の色差位置のオフセット(xFrac<sub>C</sub>,yFrac<sub>C</sub>)
- 選択された参照ピクチャの色差成分サンプル refPicLX<sub>C</sub>

この処理の出力は予測色差サンプル値  $\text{predPartLXC}[x_C, y_C]$  である。

図 8-5/JT-H264 では、A、B、C と D で表された位置は、与えられた色差サンプルの 2 次元配列  $\text{refPicLXC}$  中の整数サンプル位置での色差サンプルを表す。

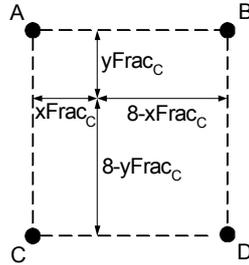


図8-5/JT-H264 色差補間における分数サンプル位置に從属する変数と周囲の整数位置のサンプルA、B、C、およびD  
(ITU-T H.264)

有効な参照ピクチャ色差配列の高さである変数  $\text{refPicHeightEffective}_C$  は次の通り導出される。

- もし  $\text{MbaffFrameFlag}$  が 0 に等しい、または  $\text{mb\_field\_decoding\_flag}$  が 0 に等しいならば、 $\text{refPicHeightEffective}_C$  は  $\text{PicHeightInSamples}_C$  に等しく設定される。
- それ以外 ( $\text{MbaffFrameFlag}$  が 1 に等しい、かつ  $\text{mb\_field\_decoding\_flag}$  が 1 に等しい)、 $\text{refPicHeightEffective}_C$  は  $\text{PicHeightInSamples}_C/2$  に等しく設定される。

等式 8-266 から 8-273 で規定されるサンプル座標は、予測色差サンプル値  $\text{predPartLXC}[x_C, y_C]$  を生成するために使用される。

$$x_{A_C} = \text{Clip3}(0, \text{PicWidthInSamples}_C - 1, x_{\text{Int}_C}) \quad (8-266)$$

$$x_{B_C} = \text{Clip3}(0, \text{PicWidthInSamples}_C - 1, x_{\text{Int}_C} + 1) \quad (8-267)$$

$$x_{C_C} = \text{Clip3}(0, \text{PicWidthInSamples}_C - 1, x_{\text{Int}_C}) \quad (8-268)$$

$$x_{D_C} = \text{Clip3}(0, \text{PicWidthInSamples}_C - 1, x_{\text{Int}_C} + 1) \quad (8-269)$$

$$y_{A_C} = \text{Clip3}(0, \text{refPicHeightEffective}_C - 1, y_{\text{Int}_C}) \quad (8-270)$$

$$y_{B_C} = \text{Clip3}(0, \text{refPicHeightEffective}_C - 1, y_{\text{Int}_C}) \quad (8-271)$$

$$y_{C_C} = \text{Clip3}(0, \text{refPicHeightEffective}_C - 1, y_{\text{Int}_C} + 1) \quad (8-272)$$

$$y_{D_C} = \text{Clip3}(0, \text{refPicHeightEffective}_C - 1, y_{\text{Int}_C} + 1) \quad (8-273)$$

等式 8-266 から 8-273 で規定される整数サンプル位置で色差サンプル A、B、C と D が与えられると、予測色差サンプル値  $\text{predPartLXC}[x_C, y_C]$  が以下のように導出される。

$$\begin{aligned} \text{predPartLXC}[x_C, y_C] = & ((8-x_{\text{Frac}_C}) * (8-y_{\text{Frac}_C}) * A + x_{\text{Frac}_C} * (8-y_{\text{Frac}_C}) * B + \\ & (8-x_{\text{Frac}_C}) * y_{\text{Frac}_C} * C + x_{\text{Frac}_C} * y_{\text{Frac}_C} * D + 32) \gg 6 \end{aligned} \quad (8-274)$$

#### 8.4.2.3 重み付けサンプル予測処理

この処理の入力は、以下の通りである。

- mbPartIdx : パーティションインデックスによって与えられる現パーティション
- subMbPartIdx : サブマクロブロックパーティションインデックス
- predFlagL0 と predFlagL1 : 予測リスト利用フラグ
- predPartLX<sub>L</sub> : 予測輝度サンプルの(partWidth)×(partHeight)配列(ここで、LX は predFlagL0 と predFlagL1 に依存して L0 か L1 に置き換えられる)
- ChromaArrayType が 0 に等しくない場合は、predPartLX<sub>Cb</sub> と predPartLX<sub>Cr</sub> : 色差成分 Cb と Cr の各々に対する、予測色差サンプルの(partWidthC)×(partHeightC)配列(ここで、LX は predFlagL0 と predFlagL1 に依存して L0 か L1 に置き換えられる)
- 重み付け予測のための変数 logWD<sub>C</sub>、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>0C</sub>、o<sub>1C</sub> の C を L に、および ChromaArrayType が 0 に等しくない時は Cb および Cr に置き換えたもの

この処理の出力は、以下の通りである。

- predPart<sub>L</sub> : 予測輝度サンプルの(partWidth)×(partHeight)配列と、
- ChromaArrayType が 0 に等しくない場合は、predPart<sub>Cb</sub> と predPart<sub>Cr</sub> : 色差成分 Cb と Cr の各々に対する、予測色差サンプルの(partWidthC)×(partHeightC)配列

P および SP スライスにおける、predFlagL0 が 1 であるマクロブロックかパーティションにおいて、次が適用される。

- もし weighted\_pred\_flag が 0 ならば、8.4.2.3.1 小小小節で記述されるデフォルト重み付けサンプル予測処理が、この小小小節で記述される処理と同じ入出力で起動される。
- それ以外(weighted\_pred\_flag が 1)、8.4.2.3.2 小小小節で記述される明示的重み付けサンプル予測処理が、この小小小節で記述される処理と同じ入出力で起動される。

B スライスにおける、predFlagL0 か predFlagL1 が 1 であるマクロブロックかパーティションにおいて、次が適用される。

- もし weighted\_bipred\_idc が 0 ならば、8.4.2.3.1 小小小節で記述されるデフォルト重み付けサンプル予測処理が、この小小小節で記述される処理と同じ入出力で起動される。
- それ以外、もし weighted\_bipred\_idc が 1 ならば、8.4.2.3.2 小小小節で記述される明示的重み付けサンプル予測処理が、この小小小節で記述される処理と同じ入出力で起動される。
- それ以外(weighted\_bipred\_idc が 2)、次が適用される。
  - もし predFlagL0 が 1 であり、predFlagL1 が 1 ならば、8.4.2.3.2 小小小節で記述される暗黙的重み付けサンプル予測処理が、この小小小節で記述される処理と同じ入出力で起動される。
  - それ以外(predFlagL0 か predFlagL1 のどちらかが 1 であるが、両方ではない)、8.4.2.3.1 小小小節で記述されるデフォルト重み付けサンプル予測処理が、この小小小節で記述される処理と同じ入出力で起動される。

#### 8.4.2.3.1 デフォルト重み付けサンプル予測処理

この処理の入力は 8.4.2.3 小小節で規定されるものと同じである。

この処理の出力は 8.4.2.3 小小節で規定されるものと同じである。

導出される予測ブロックの利用可能な成分に依存して、次が適用される。

- もし輝度サンプル予測値  $\text{predPart}_l[x,y]$  が導出されるならば、C を L に等しく設定し、x を  $0.\text{partWidth}-1$  に等しく設定し、さらに y を  $0.\text{partHeight}-1$  に等しく設定して次が適用される。
- それ以外、もし色差 Cb 成分サンプル予測値  $\text{predPart}_{cb}[x,y]$  が導出されるならば、C を Cb に等しく設定し、x を  $0.\text{partWidthC}-1$  に等しく設定し、さらに y を  $0.\text{partHeightC}-1$  に等しく設定して次が適用される。
- それ以外(色差 Cr 成分サンプル予測値  $\text{predPart}_{cr}[x,y]$  が導出される)、C を Cr に等しく設定し、x を  $0.\text{partWidthC}-1$  に等しく設定し、さらに y を  $0.\text{partHeightC}-1$  に等しく設定して次が適用される。

予測サンプル値は次の通り導出される。

- もし、 $\text{predFlagL0}$  が 1 かつ  $\text{predFlagL1}$  が 0 ならば

$$\text{predPart}_c[x,y]=\text{predPartL0}_c[x,y] \quad (8-275)$$

- それ以外、もし  $\text{predFlagL0}$  が 0 かつ  $\text{predFlagL1}$  が 1 ならば

$$\text{predPart}_c[x,y]=\text{predPartL1}_c[x,y] \quad (8-276)$$

- それ以外( $\text{predFlagL0}$  と  $\text{predFlagL1}$  が 1)

$$\text{predPart}_c[x,y]=(\text{predPartL0}_c[x,y]+\text{predPartL1}_c[x,y]+1)>>1. \quad (8-277)$$

#### 8.4.2.3.2 重み付けサンプル予測処理

この処理の入力は 8.4.2.3 小小節で規定されるものと同じである。

この処理の出力は 8.4.2.3 小小節で規定されるものと同じである。

導出される予測ブロックの利用可能な成分に依存して、次が適用される。

- もし、輝度サンプル予測値  $\text{predPart}_l[x,y]$  が導出されるならば、C を L に等しく設定し、x を  $0.\text{partWidth}-1$  に等しく設定し、さらに y を  $0.\text{partHeight}-1$  に等しく設定して次が適用される。
- それ以外、もし色差 Cb 成分サンプル予測値  $\text{predPart}_{cb}[x,y]$  が導出されるならば、C を Cb に等しく設定し、x を  $0.\text{partWidthC}-1$  に等しく設定し、さらに y を  $0.\text{partHeightC}-1$  に等しく設定して次が適用される。
- それ以外(色差 Cr 成分サンプル予測値  $\text{predPart}_{cr}[x,y]$  が導出される)、C を Cr に等しく設定し、x を  $0.\text{partWidthC}-1$  に等しく設定し、さらに y を  $0.\text{partHeightC}-1$  に等しく設定して次が適用される。

予測サンプル値は次の通り導出される。

- もし  $\text{predFlagL0}$  が 1 に等しくかつ  $\text{predFlagL1}$  が 0 に等しいならば、最終的な予測サンプル値  $\text{predPart}_c[x,y]$  は以下で導出される。

```

if(logWDC>=1)
    predPartC[x,y]=Clip1C((predPartL0C[x,y]*w0C+2logWDC-1)>>logWDC)+o0C)
else
    predPartC[x,y]=Clip1C(predPartL0C[x,y]*w0C+o0C)

```

(8-278)

- それ以外、もし predFlagL0 が 0 に等しくかつ predFlagL1 が 1 に等しいならば、最終的な予測サンプル値 predPart<sub>C</sub>[x,y]は以下で導出される。

```

if(logWDC>=1)
    predPartC[x,y]=Clip1C((predPartL1C[x,y]*w1C+2logWDC-1)>>logWDC)+o1C)
else
    predPartC[x,y]=Clip1C(predPartL1C[x,y]*w1C+o1C)

```

(8-279)

- それ以外(predFlagL0 と predFlagL1 がどちらも 1 に等しい)、最終的な予測サンプル値 predPart<sub>C</sub>[x,y]は以下で導出される。

$$\text{predPart}_C[x,y]=\text{Clip1}_C(((\text{predPartL0}_C[x,y]*w_{0C}+\text{predPartL1}_C[x,y]*w_{1C}+2^{\log\text{WD}_C})\gg(\log\text{WD}_C+1))+((o_{0C}+o_{1C}+1)\gg 1))$$

(8-280)

### 8.4.3 予測重みの導出処理

この処理への入力は、以下の通りである。

- 参照インデックス refIdxL0 と refIdxL1
- 予測利用フラグ predFlagL0 と prefFlagL1

この処理の出力は、以下の通りである。

- 重み付け予測のための変数 logWD<sub>C</sub>、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>0C</sub>、o<sub>1C</sub>の C を L に、および ChromaArrayType が 0 に等しくない時は Cb および Cr に置き換えたもの

L および、ChromaArrayType が 0 に等しくない時は Cb および Cr に置き換えられる C に対して、変数 logWD<sub>C</sub>、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>0C</sub>、o<sub>1C</sub> は次の通り導出される。

- もし weighted\_bipred\_idc が 2 であり、かつ (slice\_type%5) が 1 に等しいならば、暗黙的モードの重み付け予測が次の通り使用される。

$$\log\text{WD}_C=5$$

(8-281)

$$o_{0C}=0$$

(8-282)

$$o_{1C}=0$$

(8-283)

そして w<sub>0C</sub> と w<sub>1C</sub> は次の通り導出される。

- 変数 currPicOrField、pic0 そして pic1 は次の通り導出される。
  - もし field\_pic\_flag が 0 に等しく、かつ現マクロブロックがフィールドマクロブロックならば、次が適用される。

- currPicOrField は現マクロブロックと同じパリティをもつ現ピクチャ CurrPic のフィールドである。
- 変数 pic0 は次の通り導出される。
  - もし refldxL0%2 が 0 に等しいならば、pic0 は現マクロブロックと同じパリティをもつ RefPicList0[refldxL0/2]のフィールドである。
  - それ以外 (refldxL0%2 が 0 に等しくない)、pic0 は現マクロブロックと逆パリティをもつ RefPicList0[refldxL0/2]のフィールドである。
- 変数 pic1 は次の通り導出される。
  - もし refldxL1%2 が 0 に等しいならば、pic1 は現マクロブロックと同じパリティをもつ RefPicList1[refldxL1/2]のフィールドである。
  - それ以外 (refldxL1%2 が 0 に等しくない)、pic1 は現マクロブロックと逆パリティをもつ RefPicList1[refldxL1/2]のフィールドである。
- それ以外 (field\_pic\_flag が 1 に等しい、または現マクロブロックがフレームマクロブロック)、currPicOrField は現ピクチャ CurrPic であり、pic1 は RefPicList1[refldxL1]であり、pic0 は RefPicList0[refldxL0]である。
- 変数 tb、td、tx そして DistScaleFactor は currPicOrField、pic0、pic1 の値から等式 8-206、8-207、8-202 そして 8-203 をそれぞれ使用して導出される。
- もし DiffPicOrderCnt(pic1,pic0)が 0 に等しいか、または pic1 と pic0 の片方もしくは両方が“長期間参照使用”としてマークされているか、または (DistScaleFactor>>2)<64 または (DistScaleFactor>>2)>128 であるならば、w<sub>0c</sub> と w<sub>1c</sub> は以下で導出される。

$$w_{0c}=32 \quad (8-284)$$

$$w_{1c}=32 \quad (8-285)$$

- それ以外、

$$w_{0c}=64-(DistScaleFactor>>2) \quad (8-286)$$

$$w_{1c}=DistScaleFactor>>2 \quad (8-287)$$

- それ以外(0 または 3 に等しい(slice\_type%5)をもつスライスにおいて weighted\_pred\_flag が 1 であるか、または 1 に等しい(slice\_type%5)をもつスライスにおいて weighted\_bipred\_idc が 1 であるならば)、明示的モードの重み付け予測が以下の通り使用される。
- 変数 refldxL0WP と refldxL1WP は次の通り導出される。

$$refldxL0WP=refldxL0>>1 \quad (8-288)$$

$$refldxL1WP=refldxL1>>1 \quad (8-289)$$

- それ以外(MbaffFrameFlag が 0、または、現マクロブロックがフレームマクロブロック)、

$$\text{refIdxL0WP}=\text{refIdxL0} \quad (8-290)$$

$$\text{refIdxL1WP}=\text{refIdxL1} \quad (8-291)$$

- 変数  $\log\text{WD}_C$ 、 $w_{0C}$ 、 $w_{1C}$ 、 $o_{0C}$ 、および  $o_{1C}$  は次の通り導出される。

- もし C が輝度サンプルに対する L に等しいならば、

$$\log\text{WD}_C=\text{luma\_log2\_weight\_denom} \quad (8-292)$$

$$w_{0C}=\text{luma\_weight\_l0}[\text{refIdxL0WP}] \quad (8-293)$$

$$w_{1C}=\text{luma\_weight\_l1}[\text{refIdxL1WP}] \quad (8-294)$$

$$o_{0C}=\text{luma\_offset\_l0}[\text{refIdxL0WP}]*(\text{1}<<(\text{BitDepth}_Y-8)) \quad (8-295)$$

$$o_{1C}=\text{luma\_offset\_l1}[\text{refIdxL1WP}]*(\text{1}<<(\text{BitDepth}_Y-8)) \quad (8-296)$$

- それ以外(C が色差サンプルに対する Cb または Cr に等しい。Cb に対し  $iCbCr=0$ 、Cr に対し  $iCbCr=1$  を用いて)、

$$\log\text{WD}_C=\text{chroma\_log2\_weight\_denom} \quad (8-297)$$

$$w_{0C}=\text{chroma\_weight\_l0}[\text{refIdxL0WP}][iCbCr] \quad (8-298)$$

$$w_{1C}=\text{chroma\_weight\_l1}[\text{refIdxL1WP}][iCbCr] \quad (8-299)$$

$$o_{0C}=\text{chroma\_offset\_l0}[\text{refIdxL0WP}][iCbCr]*(\text{1}<<(\text{BitDepth}_C-8)) \quad (8-300)$$

$$o_{1C}=\text{chroma\_offset\_l1}[\text{refIdxL1WP}][iCbCr]*(\text{1}<<(\text{BitDepth}_C-8)) \quad (8-301)$$

明示的モードの重み付け予測が使用され、 $\text{predFlagL0}$  と  $\text{predFlagL1}$  が 1 に等しい時、L および、 $\text{ChromaArrayType}$  が 0 に等しくない時は Cb および Cr に置き換えられる C について、以下の制約に従わなければならない。

$$-128 \leq w_{0C} + w_{1C} \leq ((\log\text{WD}_C == 7) ? 127 : 128) \quad (8-302)$$

記 暗黙的モードの重み付け予測においては、重み  $w_{0C}$  と  $w_{1C}$  は各々  $-64..128$  の範囲にあることが保証され、かつ等式8-302で表現される制約は、明示的に課されなくとも、いつも満たされる。 $\log\text{WD}_C$  が 7 に等しい明示的モードの重み付け予測においては、2つの重み  $w_{0C}$  または  $w_{1C}$  の1つが 128 に等しいと (0 に等しい  $\text{luma\_weight\_l0\_flag}$ 、 $\text{luma\_weight\_l1\_flag}$ 、 $\text{chroma\_weight\_l0\_flag}$  または  $\text{chroma\_weight\_l1\_flag}$  の結果として) 推定される時、もう片方の重み ( $w_{0C}$  または  $w_{1C}$ ) は等式8-302で表現される制約を保持するために負の値を持たなければならない (そして、それゆえ、もう片方のフラグ  $\text{luma\_weight\_l0\_flag}$ 、 $\text{luma\_weight\_l1\_flag}$ 、 $\text{chroma\_weight\_l0\_flag}$  または  $\text{chroma\_weight\_l1\_flag}$  は 1 に等しくなければならない)。

## 8.5 変換係数復号処理とデブロックフィルタ処理前のピクチャ構築処理

この処理への入力は、(利用可能であれば)  $\text{Intra16x16DCLevel}$ 、(利用可能であれば)  $\text{Intra16x16ACLevel}$ 、(利用可能であれば)  $\text{CbIntra16x16DCLevel}$ 、(利用可能であれば)  $\text{CbIntra16x16ACLevel}$ 、(利用可能であれば)  $\text{CrIntra16x16DCLevel}$ 、(利用可能であれば)  $\text{CrIntra16x16ACLevel}$ 、(利用可能であれば)  $\text{LumaLevel}$ 、(利用可能

であれば)LumaLevel8x8、(利用可能であれば)ChromaDCLevel、(利用可能であれば)ChromaACLevel、(利用可能であれば)CbLevel、(利用可能であれば)CrLevel、(利用可能であれば)CbLevel8x8、(利用可能であれば)CrLevel8x8、および、適用可能な成分に対する現マクロブロックにおいて利用可能な INTER または INTRA 予測サンプル配列  $pred_L$ 、 $pred_{Cb}$ 、あるいは  $pred_{Cr}$  である。

記1 Intra\_4x4(またはIntra\_8x8)予測モードにおけるマクロブロックの復号時、各4×4(または8×8)輝度ブロックに対して、8.3.1(または8.3.2)小節に規定されるような輝度サンプルのIntra\_4x4(またはIntra\_8x8)予測処理と、この節で規定される処理は繰り返されるので、マクロブロック予測配列の輝度成分は完成していないかもしれない。ChromaArrayTypeが3に等しい時、マクロブロック予測配列のCbまたはCr成分は、同じ理由で、完成していないかもしれない。

この処理の出力は、適用可能な成分に対するデブロックフィルタ処理前の、構築サンプル配列  $S'_L$ 、 $S'_{Cb}$ 、あるいは  $S'_{Cr}$  である。

記2 Intra\_4x4(またはIntra\_8x8)予測モードにおけるマクロブロックの復号時、各4×4(または8×8)輝度ブロックに対して、8.3.1(または8.3.2)小節に規定されるような輝度サンプルのIntra\_4x4(またはIntra\_8x8)予測処理とこの節で規定される処理は繰り返されるので、デブロックフィルタ処理前のマクロブロック構築サンプル配列の輝度成分は完成していないかもしれない。ChromaArrayTypeが3に等しい時、デブロックフィルタ処理前のマクロブロック構築サンプル配列のCbまたはCr成分は、同じ理由で、完成していないかもしれない。

この節は変換係数復号と、デブロックフィルタ処理前のピクチャ構築を規定する。

現マクロブロックが P\_Skip、または B\_Skip として符号化されている場合、現マクロブロックに対して、LumaLevel、LumaLevel8x8、CbLevel、CbLevel8x8、CrLevel、CrLevel8x8、ChromaDCLevel、ChromaACLevel の全ての値は 0 に等しく設定される。

### 8.5.1 4×4 輝度残差ブロックに対する変換復号処理の規定

この規定は、transform\_size\_8x8\_flag が 0 に等しい場合に適用される。

現マクロブロックの予測モードが Intra\_16x16 に等しくない場合、変数 LumaLevel は輝度変換係数に対するレベルを含む。luma4x4BlkIdx=0..15 で索引される 4×4 輝度ブロックに対して、以下の順序付けられたステップが規定される。

1. 8.5.6 小節で記載される変換係数に対する逆走査処理が、LumaLevel[luma4x4BlkIdx]を入力として、2次元配列  $c$  を出力として起動される。
2. 8.5.11 小節で規定される残差 4×4 ブロックに対するスケーリングと変換処理が、 $c$  を入力として、 $r$  を出力として起動される。
3. qp\_prime\_y\_zero\_transform\_bypass\_flag が 1 に等しく、QP'Y が 0 に等しく、マクロブロック予測モードが Intra\_4x4 に等しく、かつ Intra4x4PredMode[luma4x4BlkIdx]が 0 または 1 に等しい時、8.5.14 小節で規定される INTRA 残差変換バイパス復号処理が、4 に等しく設定された nW、4 に等しく設定された nH、Intra4x4PredMode[luma4x4BlkIdx]に等しく設定された horPredFlag、および 4x4 配列  $r$  を入力として起動され、出力は 4x4 配列  $r$  の修正版である。
4. マクロブロック内の luma4x4BlkIdx のインデックスを持つ 4×4 輝度ブロックの左上サンプルの位置は、6.4.3 小節における逆 4×4 輝度ブロック走査処理が、luma4x4BlkIdx を入力として起動され、出力が(xO,yO)に割り当てられることで導出される。
5. 要素  $u_{ij}$ 、 $i,j=0..3$  をもつ 4×4 配列  $u$  は以下より導出される。

$$u_{ij} = \text{Clip}_{1\gamma}(\text{pred}_L[xO+j,yO+i]+r_{ij}) \quad (8-303)$$

qpprime\_y\_zero\_transform\_bypass\_flag が 1 に等しくかつ  $QP'_Y$  が 0 に等しい場合、ビットストリームは、等式 8-303 で計算される  $u_{ij}$  の値が結果として  $\text{pred}_{L[xO+j,yO+i]+r_{ij}}$  と等しくならないデータを含んではならない。

6. 8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、 $u$  および  $\text{luma4x4BlkIdx}$  を入力として起動される。

## 8.5.2 Intra\_16x16 マクロブロック予測モードの輝度サンプルに対する変換復号処理の規定

現マクロブロックの予測モードが Intra\_16x16 に等しい場合、変数  $\text{Intra16x16DCLevel}$  と  $\text{Intra16x16ACLevel}$  は輝度変換係数に対するレベルを含む。変換係数の復号は、以下の順序付けられたステップで進行する。

1. マクロブロックの全ての  $4 \times 4$  輝度ブロックの  $4 \times 4$  輝度 DC 変換係数が復号される。
  - a. 8.5.6 小節に記載される変換係数に対する逆走査処理が、 $\text{Intra16x16DCLevel}$  を入力として、2次元配列  $c$  を出力として起動される。
  - b. 8.5.9 小節で規定される Intra\_16x16 マクロブロックタイプに対する輝度 DC 変換係数のスケールングと変換処理が、 $\text{BitDepth}_Y$ 、 $QP'_Y$  および  $c$  を入力として、 $\text{dcY}$  を出力として起動される。
2.  $\text{luma4x4BlkIdx}=0..15$  で索引される  $4 \times 4$  輝度ブロックを処理することにより  $16 \times 16$  配列  $\text{rMb}$  が導出され、各  $4 \times 4$  輝度ブロックに対して、以下の順序付けられたステップが規定される。
  - a. 16 エントリのリストである変数  $\text{lumaList}$  が導出される。 $\text{lumaList}$  の最初のエントリは配列  $\text{dcY}$  からの対応した値である。図 8-6/JT-H264 は、配列  $\text{dcY}$  のインデックスから  $\text{luma4x4BlkIdx}$  への割り当てを示す。小さい正方形中にある 2 個の数字は  $\text{dcY}_{ij}$  中の  $i$  と  $j$  のインデックスを示しており、大きい正方形中にある数字は  $\text{luma4x4BlkIdx}$  を示している。

<sup>00</sup> 0	<sup>01</sup> 1	<sup>02</sup> 4	<sup>03</sup> 5
<sup>10</sup> 2	<sup>11</sup> 3	<sup>12</sup> 6	<sup>13</sup> 7
<sup>20</sup> 8	<sup>21</sup> 9	<sup>22</sup> 12	<sup>23</sup> 13
<sup>30</sup> 10	<sup>31</sup> 11	<sup>32</sup> 14	<sup>33</sup> 15

図8-6/JT-H264  $\text{dcY}$ のインデックスの $\text{luma4x4BlkIdx}$ への割り当て  
(ITU-T H.264)

インデックス  $k=1..15$  を用いて、 $\text{lumaList}$  中の要素は次式で規定される。

$$\text{lumaList}[k]=\text{Intra16x16ACLevel}[\text{luma4x4BlkIdx}][k-1] \quad (8-304)$$

- b. 8.5.6 小節に記載される変換係数に対する逆走査処理が、 $\text{lumaList}$  を入力として、2次元配列  $c$  を出力として起動される。
- c. 8.5.11 小節で規定される残差  $4 \times 4$  ブロックに対するスケールングと変換処理が、 $c$  を入力として、

r を出力として起動される。

- d. マクロブロック内のインデックス luma4x4BlkIdx をもつ 4×4 輝度ブロックの左上サンプルの位置は、6.4.3 小節の逆 4×4 輝度ブロック走査処理を、luma4x4BlkIdx を入力として起動し、出力に(xO,yO) が割り当てられることで導出される。
- e. 16x16 配列 rMb の  $x=xO..xO+3$  および  $y=yO..yO+3$  を用いた要素 rMb[x,y]は次により導出される。

$$rMb[xO+j,yO+i]=r_{ij} \quad (8-305)$$

3. qpprime\_y\_zero\_transform\_bypass\_flag が 1 に等しく、QP'Y が 0 に等しく、かつ Intra16x16PredMode が 0 または 1 に等しい時、8.5.14 小節で規定される INTRA 残差変換バイパス復号処理が、16 に等しく設定された nW、16 に等しく設定された nH、Intra16x16PredMode に等しく設定された horPredFlag、および 16x16 配列 rMb を入力として起動され、出力は 16x16 配列 rMb の修正版である。
4. 要素  $u_{ij}$ 、 $i,j=0..15$  をもつ 16×16 配列 u は以下より導出される。

$$u_{ij}=\text{Clip1}_Y(\text{pred}_L[j,i]+rMb[j,i]) \quad (8-306)$$

qpprime\_y\_zero\_transform\_bypass\_flag が 1 に等しくかつ QP'Y が 0 に等しい場合、ビットストリームは、等式 8-306 で計算される  $u_{ij}$  の値が結果として  $\text{pred}_L[j,i]+rMb[j,i]$  と等しくならないデータを含んではならない。

5. 8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、u を入力として起動される。

### 8.5.3 8×8 輝度残差ブロックに対する変換復号処理の規定

この規定は、transform\_size\_8x8\_flag が 1 に等しい場合に適用される。

luma8x8BlkIdx=0..3 をもつ変数 LumaLevel8x8[luma8x8BlkIdx]は、インデックス luma8x8BlkIdx をもつ輝度 8×8 ブロックの輝度変換係数に対するレベルを含む。

luma8x8BlkIdx=0..3 で索引される 8×8 輝度ブロックに対して、以下の順序付けられたステップが規定される。

1. 8.5.7 小節で記載される 8×8 輝度変換係数に対する逆走査処理が、LumaLevel[luma8x8BlkIdx]を入力として、2次元配列 c を出力として起動される。
2. 8.5.12 小節で規定される残差 8×8 ブロックに対するスケーリングと変換処理が、c を入力として、r を出力として起動される。
3. qpprime\_y\_zero\_transform\_bypass\_flag が 1 に等しく、QP'Y が 0 に等しく、マクロブロック予測モードが Intra\_8x8 に等しく、かつ Intra8x8PredMode[luma8x8BlkIdx]が 0 または 1 に等しい時、8.5.14 小節で規定される INTRA 残差変換バイパス復号処理が、8 に等しく設定された nW、8 に等しく設定された nH、Intra8x8PredMode[luma8x8BlkIdx]に等しく設定された horPredFlag、および 8x8 配列 r を入力として起動され、出力は 8x8 配列 r の修正版である。
4. マクロブロック内のインデックス luma8x8BlkIdx を持つ 8×8 輝度ブロックの左上サンプルの位置は、6.4.5 小節における逆 8×8 輝度ブロック走査処理が、luma8x8BlkIdx を入力として起動され、出力が (xO,yO)に割り当てられることで導出される。

5. 要素  $u_{ij}$ 、 $i,j=0..7$  をもつ  $8 \times 8$  配列  $u$  は以下より導出される。

$$u_{ij} = \text{Clip1}_Y(\text{pred}_L[xO+j, yO+i] + r_{ij}) \quad (8-307)$$

$\text{qpprime\_y\_zero\_transform\_bypass\_flag}$  が 1 に等しくかつ  $\text{QP}'_Y$  が 0 に等しい場合、ビットストリームは、等式 8-307 で計算される  $u_{ij}$  の値が結果として  $\text{pred}_L[xO+j, yO+i] + r_{ij}$  と等しくならないデータを含んではならない。

6. 8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、 $u$  および  $\text{luma8x8BlkIdx}$  を入力として起動される。

#### 8.5.4 色差サンプルに対する変換復号処理の規定

この処理は  $\text{ChromaArrayType}$  が 0 に等しくない時、各々の色差成分 Cb と Cr に対して分離して起動される。

$\text{ChromaArrayType}$  に依存して、以下が適用される。

- もし  $\text{ChromaArrayType}$  が 3 に等しいならば、8.5.5 小節に規定される 3 に等しい  $\text{ChromaArrayType}$  の色差サンプルに対する変換復号処理が起動される。
- それ以外 ( $\text{ChromaArrayType}$  が 3 に等しくない)、以下の文章が色差サンプルに対する変換復号処理を規定する。

各色差成分に対する、Cb で  $iCbCr$  が 0 に設定され Cr で  $iCbCr$  が 1 に設定される変数  $\text{ChromaDCLevel}[iCbCr]$  と  $\text{ChromaACLevel}[iCbCr]$  は、色差変換係数の両成分に対するレベルを含む。

変数  $\text{numChroma4x4Blks}$  を  $(\text{MbWidthC}/4) * (\text{MbHeightC}/4)$  に等しく設定する。

各色差成分に対してそれぞれ別々に、変換復号は以下の順序付けられたステップで進行する。

1. そのマクロブロックの  $iCbCr$  により索引された成分の  $4 \times 4$  色差ブロックの  $\text{numChroma4x4Blks}$  個の色差 DC 変換係数が復号される。
  - a. 変数  $\text{ChromaArrayType}$  により、次が適用される。
    - もし  $\text{ChromaArrayType}$  が 1 に等しいならば、 $2 \times 2$  配列  $c$  は以下に示す様に、 $\text{ChromaDCLevel}$  に適用された逆ラスタ走査処理を用いて導出される。

$$c = \begin{bmatrix} \text{ChromaDCLevel}[iCbCr][0] & \text{ChromaDCLevel}[iCbCr][1] \\ \text{ChromaDCLevel}[iCbCr][2] & \text{ChromaDCLevel}[iCbCr][3] \end{bmatrix} \quad (8-308)$$

- それ以外 ( $\text{ChromaArrayType}$  が 2 に等しい)、 $2 \times 4$  配列  $c$  は以下に示す様に、 $\text{ChromaDCLevel}$  に適用された逆ラスタ走査処理を用いて導出される。

$$c = \begin{bmatrix} \text{ChromaDCLevel}[iCbCr][0] & \text{ChromaDCLevel}[iCbCr][2] \\ \text{ChromaDCLevel}[iCbCr][1] & \text{ChromaDCLevel}[iCbCr][5] \\ \text{ChromaDCLevel}[iCbCr][3] & \text{ChromaDCLevel}[iCbCr][6] \\ \text{ChromaDCLevel}[iCbCr][4] & \text{ChromaDCLevel}[iCbCr][7] \end{bmatrix} \quad (8-309)$$

- b. 8.5.10 小節で規定される色差 DC 変換係数に対するスケーリングと変換処理が、 $c$  を入力として、 $dcC$  を出力として起動される。

2. iCbCr で索引される成分の、chroma4x4BlkIdx=0..numChroma4x4Blks-1 により索引される 4×4 色差ブロックを処理することにより (MbWidthC)×(MbHeightC) 配列 rMb が導出され、各 4×4 色差ブロックに対して、以下の順序付けられたステップが規定される。

- a. 16 エントリのリストである変数 chromaList が導出される。chromaList の最初のエンタリは配列 dcC からの対応した値である。図 8-7/JT-H264 は、配列 dcC のインデックスから chroma4x4BlkIdx への割り当てを示す。小さい正方形中にある 2 個の数字は dcC<sub>ij</sub> 中の i と j のインデックスを示しており、大きい正方形中にある数字は chroma4x4BlkIdx を示している。

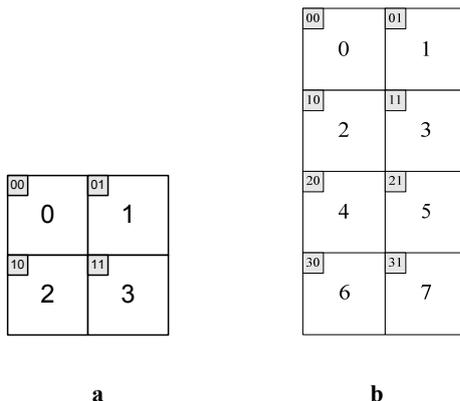


図8-7/JT-H264 dcCのインデックスのchroma4x4BlkIdxへの割り当て:  
 (a)ChromaArrayTypeが1に等しい、(b)ChromaArrayTypeが2に等しい  
 (ITU-T H.264)

インデックス k=1..15 を用いて、chromaList 中の要素は次式で規定される。

$$\text{chromaList}[k]=\text{ChromaACLevel}[\text{chroma4x4BlkIdx}][k-1] \quad (8-310)$$

- b. 8.5.10 小節で規定される変換係数に対する逆走査処理が、chromaList を入力として、2次元配列 c を出力として起動される。
- c. 8.5.11 小節で規定される残差 4×4 ブロックに対するスケーリングと変換処理が、c を入力として、r を出力として起動される。
- d. マクロブロック内のインデックス chroma4x4BlkIdx をもつ 4×4 色差ブロックの左上サンプルの位置は、以下により導出される。
- もし chroma\_format\_idc が 1 あるいは 2 に等しいならば、次が適用される。

$$xO=\text{InverseRasterScan}(\text{chroma4x4BlkIdx},4,4,8,0) \quad (8-311)$$

$$yO=\text{InverseRasterScan}(\text{chroma4x4BlkIdx},4,4,8,1) \quad (8-312)$$

- e. (MbWidthC)×(MbHeightC) 配列 rMb の x=xO..xO+3 および y=yO..yO+3 を用いた要素 rMb[x,y]は次により導出される。

$$rMb[xO+j,yO+i]=r_{ij} \quad (8-313)$$

3. `qpprime_y_zero_transform_bypass_flag` が 1 に等しく、 $QP'_Y$  が 0 に等しく、マクロブロック予測モードが `Intra_4x4`、`Intra_8x8`、または `Intra_16x16` に等しく、かつ `intra_chroma_pred_mode` が 1 または 2 に等しい時、8.5.14 小節で規定される INTRA 残差変換バイパス復号処理が、`MbWidthC` に等しく設定された `nW`、`MbHeightC` に等しく設定された `nH`、(`intra_chroma_pred_mode-1`)に等しく設定された `horPredFlag`、および  $(MbWidthC) \times (MbHeightC)$  配列 `rMb` を入力として起動され、出力は  $(MbWidthC) \times (MbHeightC)$  配列 `rMb` の修正版である。
4. 要素  $u_{ij}$ 、 $i=0..MbHeight-1$ 、 $j=0..MbWidth-1$  をつ  $(MbWidthC) \times (MbHeightC)$  配列 `u` は以下より導出される。

$$u_{ij} = \text{Clip1}_C(\text{pred}_C[j,i] + rMb[j,i]) \quad (8-314)$$

`qpprime_y_zero_transform_bypass_flag` が 1 に等しくかつ  $QP'_Y$  が 0 に等しい場合、ビットストリームは、等式 8-314 で計算される  $u_{ij}$  の値が結果として  $\text{pred}_C[j,i] + rMb[j,i]$  と等しくならないデータを含んではならない。

5. 8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、`u` を入力として起動される。

### 8.5.5 3 に等しい `ChromaArrayType` の色差サンプルに対する変換復号処理の規定

この処理は、`ChromaArrayType` が 3 の時、各色差成分 `Cb` および `Cr` に対して分離して起動される。

マクロブロック予測モードおよび `transform_size_8x8_flag` に依存して、次が適用される。

- もしマクロブロック予測モードが `Intra_16x16` ならば、`Cb` または `Cr` 残差ブロックの変換復号処理は、輝度を `Cb` または `Cr` に置き換え、`Intra16x16DCLevel` を `CbIntra16x16DCLevel` または `CrIntra16x16DCLevel` に置き換え、`Intra16x16ACLevel` を `CbIntra16x16ACLevel` または `CrIntra16x16ACLevel` に置き換え、そして  $\text{pred}_L$  を  $\text{pred}_{Cb}$  または  $\text{pred}_{Cr}$  に置き換え、`luma4x4BlkIdx` を `cb4x4BlkIdx` または `cr4x4BlkIdx` に置き換え、`lumaList` を `CbList` または `CrList` に置き換え、`BitDepthY` を `BitDepthC` に置き換え、 $QP'_Y$  を  $QP'_C$  に置き換え、そして  $\text{Clip1}_Y$  を  $\text{Clip1}_C$  に置き換えた時、8.5.2 小節に記述された処理と同一でなければならない。
- それ以外、もし `transform_size_8x8_flag` が 1 ならば、 $8 \times 8$  `Cb` または  $8 \times 8$  `Cr` 残差ブロックの変換復号処理は、輝度を `Cb` または `Cr` に置き換え、`LumaLevel8x8` を `CbLevel8x8` または `CrLevel8x8` に置き換え、 $\text{pred}_L$  を  $\text{pred}_{Cb}$  または  $\text{pred}_{Cr}$  に置き換え、`luma8x8BlkIdx` を `cb8x8BlkIdx` または `cr8x8BlkIdx` に置き換え、そして  $\text{Clip1}_Y$  を  $\text{Clip1}_C$  に置き換えた時、8.5.3 小節に記述された処理と同一でなければならない。
- それ以外 (マクロブロック予測モードが `Intra_16x16` ではなく、かつ `transform_size_8x8_flag` が 0)、 $4 \times 4$  `Cb` または  $4 \times 4$  `Cr` 残差ブロックの変換復号処理は、輝度を `Cb` または `Cr` に置き換え、`LumaLevel` を `CbLevel` または `CrLevel` に置き換え、 $\text{pred}_L$  を  $\text{pred}_{Cb}$  または  $\text{pred}_{Cr}$  に置き換え、`luma4x4BlkIdx` を `cb4x4BlkIdx` または `cr4x4BlkIdx` に置き換え、そして  $\text{Clip1}_Y$  を  $\text{Clip1}_C$  に置き換えた時、8.5.1 小節に記述された処理と同一でなければならない。8.5.1 小節に規定された処理の一部として起動される、8.5.11 小節内に規定される  $4 \times 4$  ブロックの変換係数レベルのスケーリング中、入力  $4 \times 4$  配列 `c` は、`Intra_16x16` 予測モードを用いて符号化された輝度残差ブロックに関連するものではなく、かつ色差残差ブロックに関連するものではないとして扱われる。

### 8.5.6 変換係数に対する逆走査処理

この処理への入力 は 16 個の値のリストである。

この処理の出力は、4×4 値の 2 次元配列を含む変数  $c$  である。変換係数の場合、これら 4×4 値は変換ブロックの位置に割り当てられたレベルを表す。逆走査処理をスケーリングリストへ適用する場合、出力変数  $c$  は 4×4 スケーリング行列を表す 2 次元配列を含む。

変換係数に対する逆走査処理は変換係数レベル列から変換係数レベル位置へのマッピングを行う。表 8-13/JT-H264 は、2 種類のマッピング、逆ジグザグ走査および逆フィールド走査を規定する。逆ジグザグ走査はフレームマクロブロック内の変換係数に用いられ、また逆フィールド走査はフィールドマクロブロック内の変換係数に用いられる。

スケーリングリストに対する逆走査処理は、スケーリングリストエントリ列から対応するスケーリング行列の位置へのマッピングを行う。このマッピングのために、逆ジグザグ走査が使用される。

図 8-8/JT-H264 はその走査を示す。

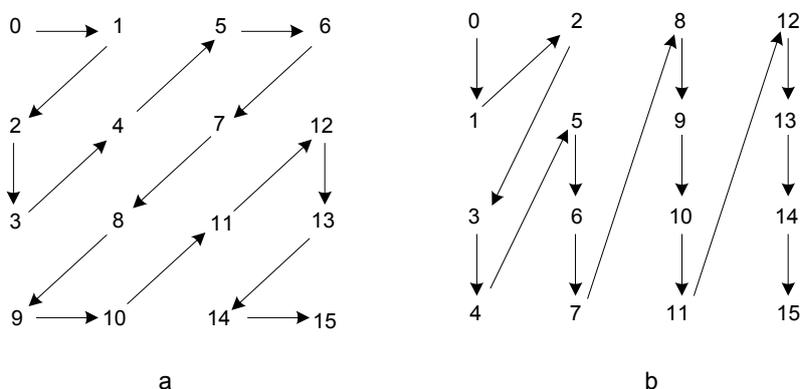


図 8-8/JT-H264 4×4ブロック走査(a)ジグザグ走査(b)フィールド走査(参考)  
(ITU-T H.264)

表 8-13/JT-H264 は、16 要素の入力リストのインデックス  $idx$  から、2 次元配列  $c$  のインデックス  $i$  と  $j$  へのマッピングを与える。

表 8-13/JT-H264 ジグザグとフィールド走査における  $idx$  から  $c_{ij}$  へのマッピング規定  
(ITU-T H.264)

idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ジグザグ	$c_{00}$	$c_{01}$	$c_{10}$	$c_{20}$	$c_{11}$	$c_{02}$	$c_{03}$	$c_{12}$	$c_{21}$	$c_{30}$	$c_{31}$	$c_{22}$	$c_{13}$	$c_{23}$	$c_{32}$	$c_{33}$
フィールド	$c_{00}$	$c_{10}$	$c_{01}$	$c_{20}$	$c_{30}$	$c_{11}$	$c_{21}$	$c_{31}$	$c_{02}$	$c_{12}$	$c_{22}$	$c_{32}$	$c_{03}$	$c_{13}$	$c_{23}$	$c_{33}$

### 8.5.7 8×8 変換係数に対する逆走査処理

この処理への入力 は 64 個の値のリストである。

この処理の出力は、8×8 値の 2 次元配列を含む変数  $c$  である。変換係数の場合、これら 8×8 値は変換ブロックの位置に割り当てられたレベルを表す。逆走査処理をスケーリングリストへ適用する場合、出力変数  $c$  は 8×8 スケーリング行列を表す 2 次元配列を含む。

変換係数に対する逆走査処理は変換係数レベル列から変換係数レベル位置へのマッピングを行う。表 8-14

／JT-H264 は、2 種類のマッピング、逆 8×8 ジグザグ走査および逆 8×8 フィールド走査を規定する。逆 8×8 ジグザグ走査はフレームマクロブロック内の変換係数に用いられ、また逆 8×8 フィールド走査はフィールドマクロブロック内の変換係数に用いられる。

スケーリングリストに対する逆走査処理は、スケーリングリストエントリ列から対応するスケーリング行列の位置へのマッピングを行う。このマッピングのために、逆ジグザグ走査が使用される。

図 8-9／JT-H264 はその走査を示す。

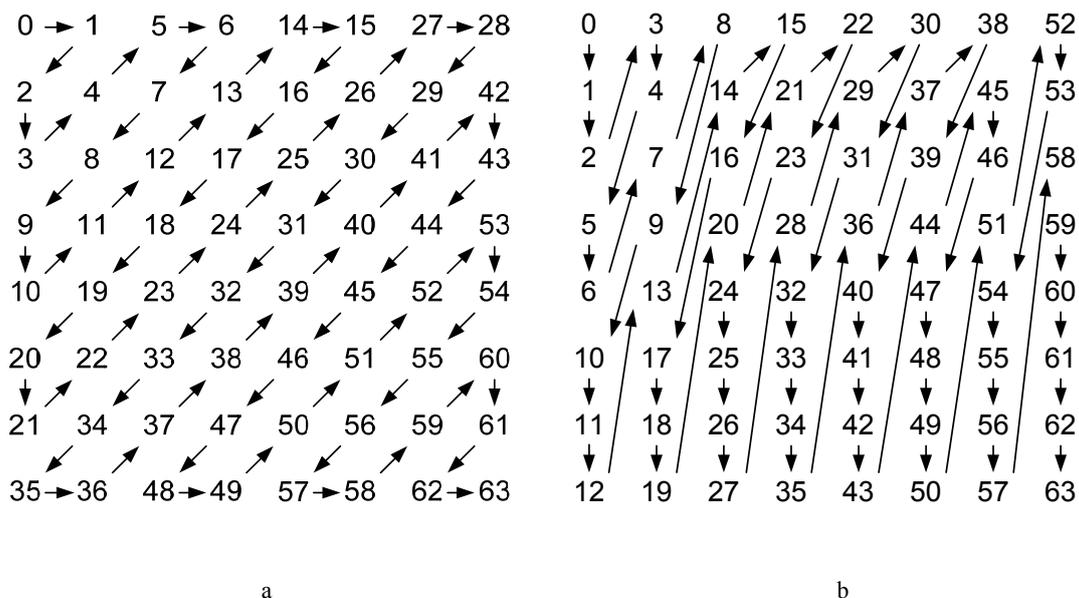


図 8-9／JT-H264 8×8ブロック走査(a) 8×8ジグザグ走査(b) 8×8フィールド走査(参考)  
(ITU-T H.264)

表 8-14／JT-H264 は、64 要素の入力リストのインデックス idx から、2 次元配列 c のインデックス i と j へのマッピングを与える。

表8-14／JT-H264 8×8ジグザグと8×8フィールド走査におけるidxからc<sub>ij</sub>へのマッピング規定  
(ITU-T H.264)

idx	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
ジグザグ	c <sub>00</sub>	c <sub>01</sub>	c <sub>10</sub>	c <sub>20</sub>	c <sub>11</sub>	c <sub>02</sub>	c <sub>03</sub>	c <sub>12</sub>	c <sub>21</sub>	c <sub>30</sub>	c <sub>40</sub>	c <sub>31</sub>	c <sub>22</sub>	c <sub>13</sub>	c <sub>04</sub>	c <sub>05</sub>
フィールド	c <sub>00</sub>	c <sub>10</sub>	c <sub>20</sub>	c <sub>01</sub>	c <sub>11</sub>	c <sub>30</sub>	c <sub>40</sub>	c <sub>21</sub>	c <sub>02</sub>	c <sub>31</sub>	c <sub>50</sub>	c <sub>60</sub>	c <sub>70</sub>	c <sub>41</sub>	c <sub>12</sub>	c <sub>03</sub>

表8-14／JT-H264(つづき) 8×8ジグザグと8×8フィールド走査におけるidxからc<sub>ij</sub>へのマッピング規定  
(ITU-T H.264)

idx	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ジグザグ	c <sub>14</sub>	c <sub>23</sub>	c <sub>32</sub>	c <sub>41</sub>	c <sub>50</sub>	c <sub>60</sub>	c <sub>51</sub>	c <sub>42</sub>	c <sub>33</sub>	c <sub>24</sub>	c <sub>15</sub>	c <sub>06</sub>	c <sub>07</sub>	c <sub>16</sub>	c <sub>25</sub>	c <sub>34</sub>
フィールド	c <sub>22</sub>	c <sub>51</sub>	c <sub>61</sub>	c <sub>71</sub>	c <sub>32</sub>	c <sub>13</sub>	c <sub>04</sub>	c <sub>23</sub>	c <sub>42</sub>	c <sub>52</sub>	c <sub>62</sub>	c <sub>72</sub>	c <sub>33</sub>	c <sub>14</sub>	c <sub>05</sub>	c <sub>24</sub>

表8-14/JT-H264(つづき) 8×8ジグザグと8×8フィールド走査におけるidxからc<sub>ij</sub>へのマッピング規定  
(ITU-T H.264)

idx	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
ジグザグ	c <sub>43</sub>	c <sub>52</sub>	c <sub>61</sub>	c <sub>70</sub>	c <sub>71</sub>	c <sub>62</sub>	c <sub>53</sub>	c <sub>44</sub>	c <sub>35</sub>	c <sub>26</sub>	c <sub>17</sub>	c <sub>27</sub>	c <sub>36</sub>	c <sub>45</sub>	c <sub>54</sub>	c <sub>63</sub>
フィールド	c <sub>43</sub>	c <sub>53</sub>	c <sub>63</sub>	c <sub>73</sub>	c <sub>34</sub>	c <sub>15</sub>	c <sub>06</sub>	c <sub>25</sub>	c <sub>44</sub>	c <sub>54</sub>	c <sub>64</sub>	c <sub>74</sub>	c <sub>35</sub>	c <sub>16</sub>	c <sub>26</sub>	c <sub>45</sub>

表8-14/JT-H264(完) 8×8ジグザグと8×8フィールド走査におけるidxからc<sub>ij</sub>へのマッピング規定  
(ITU-T H.264)

idx	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
ジグザグ	c <sub>72</sub>	c <sub>73</sub>	c <sub>64</sub>	c <sub>55</sub>	c <sub>46</sub>	c <sub>37</sub>	c <sub>47</sub>	c <sub>56</sub>	c <sub>65</sub>	c <sub>74</sub>	c <sub>75</sub>	c <sub>66</sub>	c <sub>57</sub>	c <sub>67</sub>	c <sub>76</sub>	c <sub>77</sub>
フィールド	c <sub>55</sub>	c <sub>65</sub>	c <sub>75</sub>	c <sub>36</sub>	c <sub>07</sub>	c <sub>17</sub>	c <sub>46</sub>	c <sub>56</sub>	c <sub>66</sub>	c <sub>76</sub>	c <sub>27</sub>	c <sub>37</sub>	c <sub>47</sub>	c <sub>57</sub>	c <sub>67</sub>	c <sub>77</sub>

### 8.5.8 色差量子化パラメータとスケーリング関数のための導出処理

この処理の出力は以下である。

- QP<sub>C</sub> : 各色差成分CbおよびCrに対する色差量子化パラメータ
- QS<sub>C</sub> : SPとSIスライス(適用可能なら)の復号に要求される各色差成分CbおよびCrに対する追加色差量子化パラメータ

記1 QP量子化パラメータ値、QP<sub>Y</sub>とQS<sub>Y</sub>は-QpBdOffset<sub>Y</sub>から51のそれぞれの値を含んだ範囲に常にある。QP量子化パラメータ値、QP<sub>C</sub>とQS<sub>C</sub>は-QpBdOffset<sub>C</sub>から51のそれぞれの値を含んだ範囲に常にある。

色差成分に対するQP<sub>C</sub>値は、現在のQP<sub>Y</sub>値と(Cbに対する)chroma\_qp\_index\_offsetあるいは(Crに対する)second\_chroma\_qp\_index\_offsetの値とから決定される。

記2 スケーリング等式は、QP<sub>Y</sub>の各6の増加に対して等価変換係数レベルスケーリング因子が2倍となる様に規定される。従って、スケーリングで使用される因子には、QP<sub>Y</sub>値の各1の増加に対して約12%の増加がある。

各色差成分に対するQP<sub>C</sub>の値はqp<sub>I</sub>と記される索引に基づき表8-15/JT-H264で規定されるように決定される。

各色差成分に対する変数qp<sub>Offset</sub>は、次の通り導出される。

- もし色差成分がCb成分ならば、qp<sub>Offset</sub>は次式で規定される。

$$qp_{Offset} = \text{chroma\_qp\_index\_offset} \quad (8-315)$$

- それ以外(色差成分がCr成分)、qp<sub>Offset</sub>は次式で規定される。

$$qp_{Offset} = \text{second\_chroma\_qp\_index\_offset} \quad (8-316)$$

各色差成分に対する変数qp<sub>I</sub>の値は次式で導出される。

$$qp_I = \text{Clip3}(-QpBdOffset_C, 51, QP_Y + qp_{Offset}) \quad (8-317)$$

色差成分に対するQP'<sub>C</sub>の値は次式で導出される。

$$QP'_C = QP_C + QpBdOffset_C \quad (8-318)$$

表8-15/JT-H264  $qP_1$ の関数としての $QP_C$ の規定  
(ITU-T H.264)

$qP_1$	<30	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
$QP_C$	= $qP_1$	29	30	31	32	32	33	34	34	35	35	36	36	37	37	37	38	38	38	39	39	39	39

現スライスがSPまたはSIスライスである場合、 $QS_C$ は、 $QP_Y$ を $QS_Y$ に、 $QP_C$ を $QS_C$ に置き換えた上記の処理を使用して導出される。

4×4 行列  $weightScale(i,j)$ は、次の通り規定される。

- 変数 $mbIsInterFlag$ は、次の通り導出される。
  - もし現マクロブロックがINTERマクロブロック予測モードを用いて符号化されるならば、 $mbIsInterFlag$ は1に等しく設定される。
  - それ以外(現マクロブロックがINTRAマクロブロック予測モードを用いて符号化される)、 $mbIsInterFlag$ は0に等しく設定される。
- 変数 $iYCbCr$ は、次の通り導出される。
  - もし $separate\_colour\_plane\_flag$ が1に等しいならば、 $iYCbCr$ は $colour\_plane\_id$ に等しく設定される。
  - それ以外 ( $separate\_colour\_plane\_flag$ が0に等しい) 、次が適用される。
    - もし入力配列 $c$ が輝度残差ブロックに関係するならば、 $iYCbCr$ は0に等しく設定される。
    - それ以外、もし入力配列 $c$ が色差残差ブロックに関係し、かつ色差成分がCbに等しいならば、 $iYCbCr$ は1に等しく設定される。
    - それ以外(入力配列 $c$ が色差残差ブロックに関係し、かつ色差成分がCrに等しい)、 $iYCbCr$ は2に等しく設定される。
- 8.5.6小節で規定される変換係数に対する逆走査処理が、 $ScalingList4x4[iYCbCr+((mbIsInterFlag==1)?3:0)]$ を入力として起動され、出力が4×4行列 $weightScale$ に割り当てられる。

$LevelScale(m,i,j)$ は、次により規定される。

$$LevelScale(m,i,j)=weightScale(i,j)*normAdjust(m,i,j) \quad (8-319)$$

ここで

$$normAdjust(m,i,j)=\begin{cases} v_{m0} & (i \% 2, j \% 2) \text{ に対して } (0,0) \text{ に等しい} \\ v_{m1} & (i \% 2, j \% 2) \text{ に対して } (1,1) \text{ に等しい} \\ v_{m2} & \text{それ以外} \end{cases} \quad (8-320)$$

ここで  $v$  の最初と 2 番目の下付き添字はそれぞれ次に規定される行列の行と列インデックスである。

$$v = \begin{bmatrix} 10 & 16 & 13 \\ 11 & 18 & 14 \\ 13 & 20 & 16 \\ 14 & 23 & 18 \\ 16 & 25 & 20 \\ 18 & 29 & 23 \end{bmatrix} \quad (8-321)$$

8×8 行列  $\text{weightScale8x8}(i,j)$  は、次の通り規定される。

- 変数  $\text{mbIsInterFlag}$  は、次の通り導出される。
  - もし現マクロブロックが INTER マクロブロック予測モードを用いて符号化されるならば、 $\text{mbIsInterFlag}$  は 1 に等しく設定される。
  - それ以外(現マクロブロックが INTRA マクロブロック予測モードを用いて符号化される)、 $\text{mbIsInterFlag}$  は 0 に等しく設定される。
- 変数  $\text{iYCbCr}$  は、次の通り導出される。
  - もし  $\text{separate\_colour\_plane\_flag}$  が 1 に等しいならば、 $\text{iYCbCr}$  は  $\text{colour\_plane\_id}$  に等しく設定される。
  - それ以外 ( $\text{separate\_colour\_plane\_flag}$  が 0 に等しい) 、次が適用される。
    - もし入力配列  $c$  が輝度残差ブロックに関係するならば、 $\text{iYCbCr}$  は 0 に等しく設定される。
    - それ以外、もし入力配列  $c$  が色差残差ブロックに関係し、かつ色差成分が Cb に等しいならば、 $\text{iYCbCr}$  は 1 に等しく設定される。
    - それ以外(入力配列  $c$  が色差残差ブロックに関係し、かつ色差成分が Cr に等しい)、 $\text{iYCbCr}$  は 2 に等しく設定される。
- 8.5.7 小節で規定される 8×8 変換係数に対する逆走査処理が、 $\text{ScalingList8x8}[\text{mbIsInterFlag}]$  を入力として起動され、出力が 8×8 行列  $\text{weightScale8x8}$  に割り当てられる。

$\text{LevelScale}(m,i,j)$  は、次により規定される。

$$\text{LevelScale8x8}(m,i,j) = \text{weightScale8x8}(i,j) * \text{normAdjust8x8}(m,i,j) \quad (8-322)$$

ここで

$$\text{normAdjust8x8}(m,i,j) = \begin{cases} v_{m0} & (i\%4, j\%4) \text{ に対して } (0,0) \text{ に等しい} \\ v_{m1} & (i\%2, j\%2) \text{ に対して } (1,1) \text{ に等しい} \\ v_{m2} & (i\%4, j\%4) \text{ に対して } (2,2) \text{ に等しい} \\ v_{m3} & (i\%4, j\%2) \text{ に対して } (0,1) \text{ に等しい、または } (i\%2, j\%4) \text{ に対して } (1,0) \text{ に等しい} \\ v_{m4} & (i\%4, j\%4) \text{ に対して } (0,2) \text{ に等しい、または } (i\%4, j\%4) \text{ に対して } (2,0) \text{ に等しい} \\ v_{m5} & \text{それ以外} \end{cases} \quad (8-323)$$

ここで  $v$  の最初と 2 番目の下付き添字はそれぞれ次に規定される行列の行と列インデックスである。

$$v = \begin{bmatrix} 20 & 18 & 32 & 19 & 25 & 24 \\ 22 & 19 & 35 & 21 & 28 & 26 \\ 26 & 23 & 42 & 24 & 33 & 31 \\ 28 & 25 & 45 & 26 & 35 & 33 \\ 32 & 28 & 51 & 30 & 40 & 38 \\ 36 & 32 & 58 & 34 & 46 & 43 \end{bmatrix} \quad (8-324)$$

### 8.5.9 Intra\_16x16 マクロブロックタイプに対するDC変換係数のスケーリングと変換処理

この処理への入力は、次の通りである。

- 変数bitDepthおよびqP
- $i$ と $j$ が2次元周波数インデックスである要素 $c_{ij}$ をもつ4×4配列 $c$ でのIntra\_16x16マクロブロックのDC変換係数に対する変換係数レベル値

この処理の出力は、要素  $dcY_{ij}$  をもつ 4×4 配列  $dcY$  での Intra\_16x16 マクロブロックの 4×4 ブロックに対する 16 個のスケーリングされた DC 値である。

qprime\_y\_zero\_transform\_bypass\_flag および  $QP'_Y$  の値により、次が適用される。

- もしqprime\_y\_zero\_transform\_bypass\_flagが1に等しくかつ $QP'_Y$ が0に等しいならば、出力 $dcY$ は以下のよう導出される。

$$dcY_{ij}=c_{ij}, i,j=0..3 \quad (8-325)$$

- それ以外(qprime\_y\_zero\_transform\_bypass\_flagが0に等しいあるいは $QP'_Y$ が0に等しくない)、この処理の次の文章が出力を規定する。

4×4 輝度 DC 変換係数に対する逆変換は次式で規定される。

$$f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} & c_{02} & c_{03} \\ c_{10} & c_{11} & c_{12} & c_{13} \\ c_{20} & c_{21} & c_{22} & c_{23} \\ c_{30} & c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \quad (8-326)$$

ビットストリームは、 $f$  の任意の要素  $f_{ij}$ 、 $i,j=0..3$  が  $2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

逆変換後、スケーリングが次の通り実行される。

- もしqPが36より大きいか等しいならば、スケーリングされた結果は次式から導出される。

$$dcY_{ij} = (f_{ij} * LevelScale(qP \% 6, 0, 0)) \ll (qP / 6 - 6), i,j=0..3 \quad (8-327)$$

- それ以外(qPが36より小さい)、スケーリングされた結果は次式から導出される。

$$dcY_{ij} = (f_{ij} * LevelScale(qP \% 6, 0, 0) + (1 \ll (5 - qP / 6))) \gg (6 - qP / 6), i, j = 0..3 \quad (8-328)$$

ビットストリームは、dcY の任意の要素 dcY<sub>ij</sub>、i,j=0..3 が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記1 entropy\_coding\_mode\_flag が 0 に等しく、かつ qP が 10 より小さく、かつ profile\_idc が 66、77 または 88 に等しい時、c の要素 c<sub>ij</sub> で表現される値の範囲は、Intra\_16x16 マクロブロックタイプの使用で、任意の可能な原ピクチャの内容の近い近似を形成するために必要となりうる dcY の要素 dcY<sub>ij</sub> の値の全範囲を表現するために十分ではない。

記2 dcY の要素 dcY<sub>ij</sub> に課された範囲の限界が、等式 8-328 の右シフトの後に課されるので、右シフト以前では、より大きな値の範囲が復号器にサポートされていなければならない。

### 8.5.10 色差 DC 変換係数に対するスケーリングと変換処理

この処理は、ChromaArrayType が 1 または 2 に等しい時のみ起動される。

この処理への入力は、i と j が 2 次元周波数インデックスである要素 c<sub>ij</sub> をもつ (MbWidthC/4) × (MbHeightC/4) 配列 c でのマクロブロックの 1 つの色差成分の色差 DC 変換係数に対する変換係数レベル値である。

この処理の出力は、要素 dcC<sub>ij</sub> をもつ (MbWidthC/4) × (MbHeightC/4) 配列 dcC でのスケーリングされた DC 値である。変数 bitDepth および qP はそれぞれ BitDepth<sub>c</sub> および QP'<sub>c</sub> に等しく設定される。

qprime\_y\_zero\_transform\_bypass\_flag および QP'<sub>y</sub> の値により、次が適用される。

- もし qprime\_y\_zero\_transform\_bypass\_flag が 1 に等しくかつ QP'<sub>y</sub> が 0 に等しいならば、出力 dcC は以下のように導出される。

$$dcC_{ij} = c_{ij}, i = 0..(MbWidthC/4)-1 \text{ かつ } j = 0..(MbHeightC/4)-1 \quad (8-329)$$

- それ以外 (qprime\_y\_zero\_transform\_bypass\_flag が 0 に等しいあるいは QP'<sub>y</sub> が 0 に等しくない)、次が適用される。

- 8.5.10.1 小小節に規定される色差 DC 変換係数に対する変換処理が、bitDepth および c を入力として起動され、出力は f<sub>ij</sub> を要素とする色差 DC 値の (MbWidthC/4) × (MbHeightC/4) 配列 f に割り当てられる。
- 8.5.10.2 小小節に規定される色差 DC 変換係数に対するスケーリング処理が、bitDepth、qP および f を入力として起動され、出力は dcY<sub>ij</sub> を要素とするスケーリングされた色差 DC 値の (MbWidthC/4) × (MbHeightC/4) 配列 dcY に割り当てられる。

#### 8.5.10.1 色差 DC 変換係数に対する変換処理

この処理への入力、i と j が 2 次元周波数インデックスである c<sub>ij</sub> を要素とする (MbWidthC/4) × (MbHeightC/4) 配列 c による、マクロブロックのひとつの色差成分の色差 DC 変換係数に対する変換係数レベル値である。

この処理の出力は、f<sub>ij</sub> を要素とする (MbWidthC/4) × (MbHeightC/4) 配列 f による DC 値である。

ChromaArrayType の値により、逆変換は次の通り規定される。

- もし ChromaArrayType が 1 に等しいならば、2×2 色差 DC 変換係数に対する逆変換は次式で規定される。

$$f = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (8-330)$$

- それ以外、もし ChromaArrayType が 2 に等しいならば、2×4 色差 DC 変換係数に対する逆変換は次式で規定される。

$$f = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00} & c_{01} \\ c_{10} & c_{11} \\ c_{20} & c_{21} \\ c_{30} & c_{31} \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (8-331)$$

### 8.5.10.2 色差DC変換係数に対するスケーリング処理

この処理への入力は、次の通りである。

- 変数 bitDepth および qP
- $f_{ij}$  を要素とする  $(MbWidthC/4) \times (MbHeightC/4)$  配列  $f$  による DC 値

この処理の出力は、 $dcC_{ij}$  を要素とする  $(MbWidthC/4) \times (MbHeightC/4)$  配列  $dcC$  によるスケーリングされた DC 値である。

ビットストリームは、 $f$  の任意の要素  $f_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

スケーリングが変数 ChromaArrayType により次の通り実行される。

- もし ChromaArrayType が 1 に等しいならば、スケーリングされた結果は次式から導出される。

$$dcC_{ij} = ((f_{ij} * LevelScale(qP \% 6, 0, 0)) \ll (qP / 6)) \gg 5, i,j=0,1 \quad (8-332)$$

- それ以外（もし ChromaArrayType が 2 に等しい）、次が適用される。

- 変数  $qP_{DC}$  は次のように導出される。

$$qP_{DC} = qP + 3 \quad (8-333)$$

- $qP_{DC}$  の値により、次が適用される。

- もし  $qP_{DC}$  が 36 より大きいか等しいならば、スケーリングされた結果は次式から導出される。

$$dcC_{ij} = (f_{ij} * LevelScale(qP_{DC} \% 6, 0, 0)) \ll (qP_{DC} / 6 - 6), i=0..3, j=0,1 \quad (8-334)$$

- それ以外 ( $qP_{DC}$  が 36 より小さい)、スケーリングされた結果は次式から導出される。

$$dcC_{ij} = (f_{ij} * LevelScale(qP_{DC} \% 6, 0, 0) + 2^{5-qP_{DC}/6}) \gg (6 - qP_{DC} / 6), i=0..3, j=0,1 \quad (8-335)$$

ビットストリームは、dcC の任意の要素  $dcC_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記1 entropy\_coding\_mode\_flag が 0 に等しく、かつ qP が 4 より小さい、かつ profile\_idc が 66、77 または 88 に等しい時、8.5.10.1 小節における c の要素  $c_{ij}$  で表現される値の範囲は、任意の可能な原ピクチャの内容の近い近似を形成するために必要となりうる dcC の要素  $dcC_{ij}$  の値の全範囲を表現するために十分ではないかもしれない。

記2 dcC の要素  $dcC_{ij}$  に課された範囲の限界が、等式 8-332 あるいは 8-335 の右シフトの後に課されるので、右シフト以前では、より大きな値の範囲が復号器にサポートされていなければならない。

### 8.5.11 残差 4×4 ブロックに対するスケーリングと変換処理

この処理への入力は、その輝度成分の 1 つの残差ブロックに関連した、または、1 つの色差成分の 1 つの残差ブロックに関連した、要素  $c_{ij}$  を持つ 4×4 配列  $c$  である。

この処理の出力は、要素  $r_{ij}$  を持つ 4×4 配列  $r$  での残差サンプル値である。

変数 bitDepth は次の通り導出される。

- もし入力配列  $c$  が輝度残差ブロックに関係するならば、bitDepth は  $BitDepth_Y$  に等しく設定される。
- それ以外(入力配列  $c$  が色差残差ブロックに関係する)、bitDepth は  $BitDepth_C$  に等しく設定される。

変数 sMbFlag は次の通り導出される。

- もし mb\_type が SI、あるいは、SP スライス中でマクロブロック予測モードが INTER ならば、sMbFlag は 1 に等しく設定される。
- それ以外(mb\_type が SI でなく、かつ、SP スライス中でマクロブロック予測モードが INTER でもない)、sMbFlag は 0 に等しく設定される。

変数 qP は次の通り導出される。

- もし入力配列  $c$  が輝度残差ブロックに関連しており、かつ sMbFlag が 0 に等しいならば、

$$qP = QP'_Y \quad (8-336)$$

- それ以外、もし入力配列  $c$  が輝度残差ブロックに関連しており、かつ sMbFlag が 1 に等しいならば、

$$qP = QS_Y \quad (8-337)$$

- それ以外、もし入力配列  $c$  が色差残差ブロックに関連しており、かつ sMbFlag が 0 に等しいならば、

$$qP = QP'_C \quad (8-338)$$

- それ以外(入力配列  $c$  が色差残差ブロックに関連しており、かつ sMbFlag が 1 に等しい)、

$$qP = QS_C \quad (8-339)$$

qprime\_y\_zero\_transform\_bypass\_flag および  $QP'_Y$  の値により、次が適用される。

- もし qprime\_y\_zero\_transform\_bypass\_flag が 1 に等しくかつ  $QP'_Y$  が 0 に等しいならば、出力  $r$  は以下で導出される。

$$r_{ij} = c_{ij}, i,j=0..3 \quad (8-340)$$

- それ以外(qpprime\_y\_zero\_transform\_bypass\_flag が 0 に等しいあるいは  $QP'_Y$  が 0 に等しくない)、次が適用される。
- 8.5.11.1 小小節に規定される残差 4x4 ブロックに対するスケーリング処理が、bitDepth、qP および c を入力として起動され、出力は  $d_{ij}$  を要素とするスケーリングされた変換係数の 4x4 配列 d に割り当てられる。
- 8.5.11.2 小小節に規定される残差 4x4 ブロックに対する変換処理が、bitDepth および d を入力として起動され、出力は  $r_{ij}$  を要素とする残差サンプル値の 4x4 配列 r に割り当てられる。

#### 8.5.11.1 残差 4x4 ブロックに対するスケーリング処理

この処理への入力は、次の通りである。

- 変数bitDepthおよびqP
- 輝度成分の残差ブロックに関連する配列か、色差成分の残差ブロックに関連する配列かのいずれかである、 $c_{ij}$  を要素とする4x4配列c

この処理の出力は、 $d_{ij}$  を要素とするスケーリングされた変換係数の 4x4 配列 d である。

ビットストリームは、c の任意の要素  $c_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

4x4 ブロック変換係数レベル  $c_{ij}$  のスケーリング処理は次の通り進行する。

- もし以下の条件の全てが真ならば、
  - i が 0 に等しい
  - j が 0 に等しい
  - c が、Intra\_16x16 予測モードを用いて符号化された輝度残差ブロックか、色差残差ブロックに関係している

変数 $d_{00}$ は以下で導出される

$$d_{00}=c_{00} \tag{8-341}$$

- それ以外、次が適用される。
  - もし qP が 24 より大きいか等しいならば、スケーリングされた結果は次の通り導出される。

$$d_{ij} = (c_{ij} * \text{LevelScale}(qP \% 6, i, j)) \ll (qP / 6 - 4), i, j=0..3 \text{ (上記を除く)} \tag{8-342}$$

- それ以外(qP が 24 より小さい)、スケーリングされた結果は次の通り導出される。

$$d_{ij} = (c_{ij} * \text{LevelScale}(qP \% 6, i, j) + 2^{3-qP/6}) \gg (4 - qP / 6), i, j=0..3 \text{ (上記を除く)} \tag{8-343}$$

ビットストリームは、d の任意の要素  $d_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

### 8.5.11.2 残差 4x4 ブロックに対する変換処理

この処理への入力は、次の通りである。

- 変数bitDepth
- $d_{ij}$ を要素とするスケーリングされた変換係数の4x4配列d

この処理の出力は、 $r_{ij}$ を要素とする 4x4 配列 r による残差サンプル値である。

ビットストリームは、d の任意の要素  $d_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

変換処理は、以下と数学的に等価な方法で、スケーリングされた変換係数のブロックを出力サンプルのブロックへ変換しなくてはならない。

最初に、スケーリングされた変換係数の各(水平)行が、以下のような 1 次元逆変換を用いて変換される。

中間値の組が以下のように計算される。

$$e_{i0}=d_{i0}+d_{i2}, i=0..3 \quad (8-344)$$

$$e_{i1}=d_{i0}-d_{i2}, i=0..3 \quad (8-345)$$

$$e_{i2}=(d_{i1} \gg 1) - d_{i3}, i=0..3 \quad (8-346)$$

$$e_{i3}=d_{i1} + (d_{i3} \gg 1), i=0..3 \quad (8-347)$$

ビットストリームは、e の任意の要素  $e_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

次に、変換された結果は、これら中間値から以下のように計算される。

$$f_{i0}=e_{i0}+e_{i3}, i=0..3 \quad (8-348)$$

$$f_{i1}=e_{i1}+e_{i2}, i=0..3 \quad (8-349)$$

$$f_{i2}=e_{i1}-e_{i2}, i=0..3 \quad (8-350)$$

$$f_{i3}=e_{i0}-e_{i3}, i=0..3 \quad (8-351)$$

ビットストリームは、f の任意の要素  $f_{ij}$ 、 $i,j=0..3$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

次に、得られた結果の行列の各(垂直)列が、同じ 1 次元逆変換を用いて以下のように変換される。

中間値の組が以下のように計算される。

$$g_{0j}=f_{0j}+f_{2j}, j=0..3 \quad (8-352)$$

$$g_{1j}=f_{0j}-f_{2j}, j=0..3 \quad (8-353)$$

$$g_{2j}=(f_{1j} \gg 1) - f_{3j}, j=0..3 \quad (8-354)$$

$$g_{3j}=f_{1j}+(f_{3j}>>1), j=0..3 \quad (8-355)$$

ビットストリームは、 $g$ の任意の要素 $g_{ij}$ 、 $i,j=0..3$ が $-2^{(7+\text{bitDepth})}$ から $2^{(7+\text{bitDepth})}-1$ でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

次に、変換された結果は、これら中間値から以下のように計算される。

$$h_{0j}=g_{0j}+g_{3j}, j=0..3 \quad (8-356)$$

$$h_{1j}=g_{1j}+g_{2j}, j=0..3 \quad (8-357)$$

$$h_{2j}=g_{1j}-g_{2j}, j=0..3 \quad (8-358)$$

$$h_{3j}=g_{0j}-g_{3j}, j=0..3 \quad (8-359)$$

ビットストリームは、 $h$ の任意の要素 $h_{ij}$ 、 $i,j=0..3$ が $-2^{(7+\text{bitDepth})}$ から $2^{(7+\text{bitDepth})}-33$ でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

変換されたサンプルの配列を生成するための1次元水平と1次元垂直の両逆変換実行後、最終構築残差サンプル値は以下で導出される。

$$r_{ij}=(h_{ij}+2^5)>>6, i,j=0..3 \quad (8-360)$$

#### 8.5.12 残差 8×8 ブロックに対するスケーリングと変換処理

この処理への入力は、輝度成分の 8×8 残差ブロックに関連する配列か、または、ChromaArrayType が 3 に等しい時、色差成分の 8×8 残差ブロックに関連する配列である、要素  $c_{ij}$  を持つ 8×8 配列  $c$  である。

記1 separate\_colour\_plane\_flagが1に等しい時、各符号化ピクチャの復号処理 (colour\_plane\_idの値に従う特定の輝度あるいは色差ピクチャ配列への復号ピクチャの最終割り当ての前)の目的においては、全ての残差ブロックは、輝度成分に関連付けられているものとみなされる。

この処理の出力は、要素  $r_{ij}$  を持つ 8×8 配列  $r$  による残差サンプル値である。

変数 bitDepth および qP は、次の通り導出される。

- もし入力配列  $c$  が輝度残差ブロックに関連するならば、bitDepth は  $\text{BitDepth}_Y$  に、QP は  $\text{QP}'_Y$  に等しく設定される。
- それ以外 (入力配列  $c$  が色差残差ブロックに関連する)、bitDepth は  $\text{BitDepth}_C$  に、QP は  $\text{QP}'_C$  に等しく設定される。

記2 separate\_colour\_plane\_flagが1に等しい時、ピクチャの各色成分の復号処理の目的においては、全ての残差ブロックは、輝度成分に関連付けられているものとみなされる。

qpprime\_y\_zero\_transform\_bypass\_flag および  $\text{QP}'_Y$  の値により、次が適用される。

- もし qpprime\_y\_zero\_transform\_bypass\_flag が 1 に等しくかつ  $\text{QP}'_Y$  が 0 に等しいならば、出力  $r$  は以下で導出される。

$$r_{ij}=c_{ij}, i,j=0..7 \quad (8-361)$$

- それ以外(qpprime\_y\_zero\_transform\_bypass\_flag が 0 に等しいあるいは  $QP'_Y$  が 0 に等しくない)、次が適用される。
  - 8.5.12.1 小小節に規定される残差 8x8 ブロックに対するスケーリング処理が、bitDepth、qP および c を入力として起動され、出力は  $d_{ij}$  を要素とするスケーリングされた変換係数の 8x8 配列 d に割り当てられる。
  - 8.5.12.2 小小節に規定される残差 8x8 ブロックに対する変換処理が、bitDepth および d を入力として起動され、出力は  $r_{ij}$  を要素とする残差サンプル値の 8x8 配列 r に割り当てられる。

#### 8.5.12.1 残差 8x8 ブロックに対するスケーリング処理

この処理への入力は、次の通りである。

- 変数bitDepthおよびqP
- 輝度成分の残差ブロックに関連する配列か、色差成分の残差ブロックに関連する配列かのいずれかである、 $c_{ij}$ を要素とする8x8配列c

この処理の出力は、 $d_{ij}$ を要素とするスケーリングされた変換係数の 8x8 配列 d である。

ビットストリームは、c の任意の要素  $c_{ij}$ 、 $i,j=0..7$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

8x8 ブロック変換係数レベル  $c_{ij}$  のスケーリング処理は次の通り進行する。

- もし qP が 36 より大きいか等しいならば、スケーリングされた結果は次式から導出される。

$$d_{ij}=(c_{ij}*\text{LevelScale8x8}(qP\%6,i,j))\ll(qP/6-6), i,j=0..7 \quad (8-362)$$

- それ以外(qP が 36 より小さい)、スケーリングされた結果は次式から導出される。

$$d_{ij}=(c_{ij}*\text{LevelScale8x8}(qP\%6,i,j))+2^{5-QP'_Y/6}\gg(6-qP/6), i,j=0..7 \quad (8-363)$$

ビットストリームは、d の任意の要素  $d_{ij}$ 、 $i,j=0..7$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

#### 8.5.12.2 残差 8x8 ブロックに対する変換処理

この処理への入力は、次の通りである。

- 変数bitDepth
- $d_{ij}$ を要素とするスケーリングされた変換係数の8x8配列d

この処理の出力は、 $r_{ij}$ を要素とする 8x8 配列 r による残差サンプル値である。

ビットストリームは、d の任意の要素  $d_{ij}$ 、 $i,j=0..7$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

変換処理は、以下と数学的に等価な方法で、スケーリングされた変換係数のブロックを出力サンプルのブロックへ変換しなくてはならない。

最初に、スケーリングされた変換係数の各(水平)行が、以下のような1次元逆変換を用いて変換される。

- 中間値  $e_{ij}$  の組が以下のように導出される。

$$e_{i0}=d_{i0}+d_{i4}, \quad i=0..7 \quad (8-364)$$

$$e_{i1}=-d_{i3}+d_{i5}-d_{i7}-(d_{i7} \gg 1), \quad i=0..7 \quad (8-365)$$

$$e_{i2}=d_{i0}-d_{i4}, \quad i=0..7 \quad (8-366)$$

$$e_{i3}=d_{i1}+d_{i7}-d_{i3}-(d_{i3} \gg 1), \quad i=0..7 \quad (8-367)$$

$$e_{i4}=(d_{i2} \gg 1)-d_{i6}, \quad i=0..7 \quad (8-368)$$

$$e_{i5}=-d_{i1}+d_{i7}+d_{i5}+(d_{i5} \gg 1), \quad i=0..7 \quad (8-369)$$

$$e_{i6}=d_{i2}+(d_{i6} \gg 1), \quad i=0..7 \quad (8-370)$$

$$e_{i7}=d_{i3}+d_{i5}+d_{i1}+(d_{i1} \gg 1), \quad i=0..7 \quad (8-371)$$

- 中間値  $f_{ij}$  の第2の組は、中間値  $e_{ij}$  から以下のように計算される。

$$f_{i0}=e_{i0}+e_{i6}, \quad i=0..7 \quad (8-372)$$

$$f_{i1}=e_{i1}+(e_{i7} \gg 2), \quad i=0..7 \quad (8-373)$$

$$f_{i2}=e_{i2}+e_{i4}, \quad i=0..7 \quad (8-374)$$

$$f_{i3}=e_{i3}+(e_{i5} \gg 2), \quad i=0..7 \quad (8-375)$$

$$f_{i4}=e_{i2}-e_{i4}, \quad i=0..7 \quad (8-376)$$

$$f_{i5}=(e_{i3} \gg 2)-e_{i5}, \quad i=0..7 \quad (8-377)$$

$$f_{i6}=e_{i0}-e_{i6}, \quad i=0..7 \quad (8-378)$$

$$f_{i7}=e_{i7}-(e_{i1} \gg 2), \quad i=0..7 \quad (8-379)$$

- 次に、変換された結果  $g_{ij}$  は、これら中間値  $f_{ij}$  から以下のように計算される。

$$g_{i0}=f_{i0}+f_{i7}, \quad i=0..7 \quad (8-380)$$

$$g_{i1}=f_{i2}+f_{i5}, \quad i=0..7 \quad (8-381)$$

$$g_{i2}=f_{i4}+f_{i3}, \quad i=0..7 \quad (8-382)$$

$$g_{i3}=f_{i6}+f_{i1}, \quad i=0..7 \quad (8-383)$$

$$g_{i4}=f_{i6}-f_{i1}, \quad i=0..7 \quad (8-384)$$

$$g_{i5}=f_{i4}-f_{i3}, \quad i=0..7 \quad (8-385)$$

$$g_{i6}=f_{i2}-f_{i5}, \quad i=0..7 \quad (8-386)$$

$$g_{i7}=f_{i0}-f_{i7}, \quad i=0..7 \quad (8-387)$$

次に、得られた結果の行列の各(垂直)列が、同じ1次元逆変換を用いて以下のように変換される。

- 中間値  $h_{ij}$  の組が、水平に変換された値  $g_{ij}$  から以下のように計算される。

$$h_{0j}=g_{0j}+g_{4j}, \quad j=0..7 \quad (8-388)$$

$$h_{1j}=-g_{3j}+g_{5j}-g_{7j}-(g_{7j} \gg 1), \quad j=0..7 \quad (8-389)$$

$$h_{2j}=g_{0j}-g_{4j}, \quad j=0..7 \quad (8-390)$$

$$h_{3j}=g_{1j}+g_{7j}-g_{3j}-(g_{3j} \gg 1), \quad j=0..7 \quad (8-391)$$

$$h_{4j}=(g_{2j} \gg 1)-g_{6j}, \quad j=0..7 \quad (8-392)$$

$$h_{5j}=-g_{1j}+g_{7j}+g_{5j}+(g_{5j} \gg 1), \quad j=0..7 \quad (8-393)$$

$$h_{6j}=g_{2j}+(g_{6j} \gg 1), \quad j=0..7 \quad (8-394)$$

$$h_{7j}=g_{3j}+g_{5j}+g_{1j}+(g_{1j} \gg 1), \quad j=0..7 \quad (8-395)$$

- 中間値  $k_{ij}$  の第2の組は、中間値  $h_{ij}$  から以下のように計算される。

$$k_{0j}=h_{0j}+h_{6j}, \quad j=0..7 \quad (8-396)$$

$$k_{1j}=h_{1j}+(h_{7j} \gg 2), \quad j=0..7 \quad (8-397)$$

$$k_{2j}=h_{2j}+h_{4j}, \quad j=0..7 \quad (8-398)$$

$$k_{3j}=h_{3j}+(h_{5j} \gg 2), \quad j=0..7 \quad (8-399)$$

$$k_{4j}=h_{2j}-h_{4j}, \quad j=0..7 \quad (8-400)$$

$$k_{5j}=(h_{3j} \gg 2)-h_{5j}, \quad j=0..7 \quad (8-401)$$

$$k_{6j}=h_{0j}-h_{6j}, \quad j=0..7 \quad (8-402)$$

$$k_{7j}=h_{7j}-(h_{1j} \gg 2), \quad j=0..7 \quad (8-403)$$

- 次に、変換された結果  $m_{ij}$  は、これら中間値  $k_{ij}$  から以下のように計算される。

$$m_{0j}=k_{0j}+k_{7j}, \quad j=0..7 \quad (8-404)$$

$$m_{1j}=k_{2j}+k_{5j}, \quad j=0..7 \quad (8-405)$$

$$m_{2j}=k_{4j}+k_{3j}, \quad j=0..7 \quad (8-406)$$

$$m_{3j}=k_{6j}+k_{1j}, \quad j=0..7 \quad (8-407)$$

$$m_{4j}=k_{6j}-k_{1j}, j=0..7 \quad (8-408)$$

$$m_{5j}=k_{4j}-k_{3j}, j=0..7 \quad (8-409)$$

$$m_{6j}=k_{2j}-k_{5j}, j=0..7 \quad (8-410)$$

$$m_{7j}=k_{0j}-k_{7j}, j=0..7 \quad (8-411)$$

ビットストリームは、 $e_{ij}$ 、 $f_{ij}$ 、 $g_{ij}$ 、 $h_{ij}$  または  $k_{ij}$ 、 $i,j=0..7$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

ビットストリームは、 $m_{ij}$ 、 $i,j=0..7$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-33$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

変換されたサンプルの配列を生成するための1次元水平と1次元垂直の両逆変換実行後、最終構築残差サンプル値は以下で導出される。

$$r_{ij}=(m_{ij}+2^5)>>6, i,j=0..7 \quad (8-412)$$

### 8.5.13 デブロックフィルタ処理前のピクチャ構築処理

この処理への入力には以下である。

- 16×16輝度ブロックあるいは(MbWidthC)×(MbHeightC)色差ブロックあるいは4×4輝度ブロックあるいは4×4色差ブロックあるいは8×8輝度ブロックあるいは、ChromaArrayTypeが3に等しい時、8×8色差ブロックで、 $u_{ij}$ を要素に持つサンプル配列 $u$
- $u$ が16×16輝度ブロックあるいは(MbWidthC)×(MbHeightC)色差ブロックでない時、ブロックインデックスluma4x4BlkIdxまたはchroma4x4BlkIdxまたはluma8x8BlkIdxまたはcb4x4BlkIdxまたはcr4x4BlkIdxまたはcb8x8BlkIdxまたはcr8x8BlkIdx

現マクロブロックの左上輝度サンプルの位置は、6.4.1 小節の逆マクロブロック走査処理を、CurrMbAddrを入力として起動し、出力に(xP,yP)が割り当てられることで導出される。

$u$ が輝度ブロックの場合、輝度ブロックの各サンプル $u_{ij}$ に対して、次が適用される。

- ブロック $u$ のサイズにより、次が適用される。
  - もし $u$ が16×16輝度ブロックならば、マクロブロック内の16×16輝度ブロックの左上サンプルの位置(xO,yO)は(0,0)に等しく設定され、変数nEは16に等しく設定される。
  - それ以外、もし $u$ が4×4輝度ブロックならば、マクロブロック内のインデックスluma4x4BlkIdxをもつ4×4輝度ブロックの左上サンプルの位置は、6.4.3小節の逆4×4輝度ブロック走査処理を、luma4x4BlkIdxを入力として起動し、出力に(xO,yO)が割り当てられることで導出され、変数nEは4に等しく設定される。
  - それ以外( $u$ が8×8輝度ブロック)、マクロブロック内のインデックスluma8x8BlkIdxをもつ8×8輝度ブロックの左上サンプルの位置は、6.4.5小節の逆8×8輝度ブロック走査処理を、luma8x8BlkIdxを入力として起動し、出力に(xO,yO)が割り当てられることで導出され、変数nEは8に等しく設定される。
- 変数MbaffFrameFlagおよび現マクロブロックにより、次が適用される。

- もしMbaffFrameFlagが1に等しく、かつ現マクロブロックがフィールドマクロブロックならば、

$$S'_l[xP+xO+j,yP+2*(yO+i)]=u_{ij}, i,j=0..nE-1 \quad (8-413)$$

- それ以外(MbaffFrameFlagが0に等しいか、現マクロブロックがフレームマクロブロック)、

$$S'_l[xP+xO+j,yP+yO+i]=u_{ij}, i,j=0..nE-1 \quad (8-414)$$

u が色差ブロックの場合、色差ブロックの各サンプル  $u_{ij}$  に対して、次が適用される。

- 変数  $S'_c$  の下付き添字 C を、Cb 色差成分に対しては Cb に置き換え、Cr 色差成分に対しては Cr に置き換える。
- ブロック u のサイズにより、次が適用される。

- もし u が  $(MbWidthC) \times (MbHeightC)$  Cb または Cr ブロックならば、変数 nW は MbWidthC に等しく設定され、変数 nH は MbHeightC に等しく設定され、そして、マクロブロック内の  $(nW) \times (nH)$  Cb または Cr ブロックの左上サンプルの位置  $(xO,yO)$  は  $(0,0)$  に等しく設定される。

- それ以外、もし u が  $4 \times 4$  Cb または Cr ブロックならば、変数 nW および nH は 4 に等しく設定され、変数 ChromaArrayType により、マクロブロック内のインデックス chroma4x4BlkIdx をもつ  $4 \times 4$  Cb または Cr ブロックの左上サンプルの位置は、次の通り導出される。

- もし ChromaArrayType が 1 または 2 に等しいならば、次が適用される。

$$xO=InverseRasterScan(chroma4x4BlkIdx,4,4,8,0) \quad (8-415)$$

$$yO=InverseRasterScan(chroma4x4BlkIdx,4,4,8,1) \quad (8-416)$$

- それ以外(ChromaArrayType が 3 に等しい)、マクロブロック内のインデックス cb4x4BlkIdx をもつ  $4 \times 4$  Cb またはインデックス cr4x4BlkIdx をもつ  $4 \times 4$  Cr ブロックの左上サンプルの位置は、6.4.4 小節の逆  $4 \times 4$  Cb または Cr ブロック走査処理を、cb4x4BlkIdx または cr4x4BlkIdx を入力として、出力が  $(xO,yO)$  に割り当てられるように起動することにより導出される。

- それ以外(ChromaArrayType が 3 に等しい時に、u が  $8 \times 8$  Cb または Cr ブロック)、変数 nW および nH は 8 に等しく設定され、マクロブロック内のインデックス cb8x8BlkIdx をもつ  $8 \times 8$  Cb またはインデックス cr8x8BlkIdx をもつ  $8 \times 8$  Cr ブロックの左上サンプルの位置は、6.4.6 小節の逆  $8 \times 8$  Cb または Cr ブロック走査処理を、cb8x8BlkIdx または cr8x8BlkIdx を入力として、出力が  $(xO,yO)$  に割り当てられるように起動することにより導出される。

- 変数 MbaffFrameFlag および現マクロブロックにより、次が適用される。

- もし MbaffFrameFlag が 1 に等しく、かつ現マクロブロックがフィールドマクロブロックならば、

$$S'_c[(xP/subWidthC)+xO+j,(yP+SubHeightC-1)/SubHeightC+2*(yO+i)]=u_{ij}, \\ i=0..nH-1, j=0..nW-1 \quad (8-417)$$

- それ以外(MbaffFrameFlag が 0 に等しいか、現マクロブロックがフレームマクロブロック)、

$$S'_c[(xP/subWidthC)+xO+j,(yP/SubHeightC)+yO+i]=u_{ij}, \\ i=0..nH-1, j=0..nW-1 \quad (8-418)$$

#### 8.5.14 INTRA残差変換バイパス復号処理

この処理は、`qpprime_y_zero_transform_bypass_flag` が 1 に等しく、`QP'y` が 0 に等しく、マクロブロック予測モードが `Intra_4x4`、`Intra_8x8` または `Intra_16x16` に等しく、そして適用可能な INTRA 予測モードが垂直または水平モードに等しい場合に起動される。`Cb` および `Cr` 成分に対する処理は輝度 (L または Y) 成分に対するものと同様に適用される。

この処理の入力は次の通り。

- `nW` と `nH` の 2 つの変数
- 変数 `horPredFlag`
- 輝度成分の残差変換バイパスブロックに関連する配列、あるいは `Cb` および `Cr` 成分の残差変換バイパスブロックに関連する配列である、要素  $r_{ij}$  を持つ  $(nW) \times (nH)$  配列  $r$

この処理の出力は、INTRA 残差変換バイパス復号処理の結果を含む要素  $r_{ij}$  を持つ  $(nW) \times (nH)$  配列  $r$  の修正版である。

$f$  を、次の通り導出される要素  $f_{ij}$  を持つ一時的な  $(nW) \times (nH)$  配列とする。

$$f_{ij} = r_{ij}, \quad i=0..nH-1, \quad j=0..nW-1 \quad (8-419)$$

`horPredFlag` に依存して、次が適用される。

- `horPredFlag` が 0 に等しい時、修正された配列  $r$  は次の通り導出される。

$$r_{ij} = \sum_{k=0}^i f_{kj}, \quad i=0..nH-1, \quad j=0..nW-1 \quad (8-420)$$

それ以外 (`horPredFlag` が 1 に等しい)、修正された配列  $r$  は次の通り導出される。

$$r_{ij} = \sum_{k=0}^j f_{ik}, \quad i=0..nH-1, \quad j=0..nW-1 \quad (8-421)$$

#### 8.6 SPスライス中のPマクロブロックまたはSIマクロブロックに対する復号処理

この処理は SP スライスタイプ中の P マクロブロックタイプ、または SI スライス中の SI マクロブロックタイプを復号する時に起動される。

この処理への入力は予測残差変換係数レベル、および現マクロブロックに対する予測サンプルである。

この処理の出力はデブロックフィルタ処理前の現マクロブロックに対する復号されたサンプルである。

この節では SP スライスの P マクロブロックタイプと SI スライスの SI マクロブロックタイプに対する、変換係数の復号処理とピクチャ構築処理を規定する。

記 SPスライスでは、Pスライス符号化に類似の方法を用いて、シーケンス中の時間的冗長度を利用するためにINTER予測符号化を使用している。しかしPスライス符号化とは異なり、SPスライス符号化では異なった参照ピクチャが用いられた時でさえもスライスの同一の再構築を許容している。SIスライスではIスライスに類似の方法を用いて、空間予測を使用している。SIスライス符号化では対応するSPスライス

と同一の再構築を許容している。SPとSIスライスの特性はビットストリームの切り替え、スプラインシグ、ランダムアクセス、高速再生、高速逆再生、および誤り耐性/回復の機能を支援する。

SP スライスにはIマクロブロックタイプ、あるいはPマクロブロックタイプのどちらかで符号化されたマクロブロックより成る。

SI スライスにはIマクロブロックタイプ、あるいはSIマクロブロックタイプのどちらかで符号化されたマクロブロックより成る。

SI スライスのIマクロブロックタイプに対する、変換係数の復号処理とデブロックフィルタ処理前のピクチャ構築処理は、8.5 節に規定されるように起動される。SI マクロブロックタイプは以下で示されるように復号される。

現マクロブロックが P\_Skip として符号化されている場合、現マクロブロックに対する LumaLevel、ChromaDCLevel、ChromaACLevel の全ての値は 0 に等しく設定される。

### 8.6.1 非スイッチングピクチャに対するSP復号処理

この処理は、sp\_for\_switch\_flag が 0 に等しい SP スライスの P マクロブロックタイプの復号時に起動される。

この処理への入力は、8.4 節からの現マクロブロックに対する INTER 予測サンプルと予測残差変換係数レベルである。

この処理の出力はデブロックフィルタ処理前の現マクロブロックの復号されたサンプルである。

この小節は、sp\_for\_switch\_flag が 0 に等しい SP スライスの、マクロブロック予測モードが Intra\_4x4 または Intra\_16x16 に等しい場合を除いた全てのマクロブロックに対して適用される。この処理は SI スライスには適用されない。

#### 8.6.1.1 輝度変換係数の復号処理

この処理への入力は、8.4 節からの現マクロブロックに対する INTER 予測輝度サンプル  $pred_L$  と、予測残差変換係数レベル LumaLevel、および 4x4 輝度ブロックのインデックス luma4x4BlkIdx である。

現マクロブロック内の luma4x4BlkIdx のインデックスをもつ 4x4 輝度ブロックの左上サンプルの位置は、6.4.3 小節の逆 4x4 輝度ブロック走査処理を luma4x4BlkIdx を入力として起動し、出力が(x,y)に割り当てられることで導出される。

変数  $p$  を以下で導出される要素  $p_{ij}$  を持つ予測サンプルの 4x4 配列とする。

$$p_{ij} = pred_L[x+j, y+i], \quad i, j = 0..3 \quad (8-422)$$

変数  $c^p$  は以下に従い、変換されて変換係数  $c^p$  を生成する。

$$c^p = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \begin{bmatrix} p_{00} & p_{01} & p_{02} & p_{03} \\ p_{10} & p_{11} & p_{12} & p_{13} \\ p_{20} & p_{21} & p_{22} & p_{23} \\ p_{30} & p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix} \quad (8-423)$$

8.5.6 小節で記載される変換係数に対する逆走査処理が、LumaLevel[luma4x4BlkIdx]を入力として、要素  $c_{ij}^f$  をもつ 2 次元配列  $c^f$  を出力として起動される。

予測残差変換係数  $c^f$  は、以下で  $i,j=0..3$  として示される通り、量子化パラメータ  $QP_Y$  を用いてスケールングされ予測ブロックの変換係数  $c^p$  に加算される。

$$c_{ij}^s = c_{ij}^p + (((c_{ij}^f * \text{LevelScale}(QP_Y \% 6, i, j) * A_{ij}) << ((QP_Y / 6))) >> 10) \quad (8-424)$$

ここで  $\text{LevelScale}(m, i, j)$  は等式 8-319 で規定され、また  $A_{ij}$  は次式で規定される。

$$A_{ij} = \begin{cases} 16 & (i, j) \in \{(0,0), (0,2), (2,0), (2,2)\}, \\ 25 & (i, j) \in \{(1,1), (1,3), (3,1), (3,3)\}, \\ 20 & \text{それ以外;} \end{cases} \quad (8-425)$$

以下の式で使われる関数  $\text{LevelScale2}(m, i, j)$  は次式で規定される。

$$\text{LevelScale2}(m, i, j) = \begin{cases} w_{m0} & (i, j) \in \{(0,0), (0,2), (2,0), (2,2)\}, \\ w_{m1} & (i, j) \in \{(1,1), (1,3), (3,1), (3,3)\}, \\ w_{m2} & \text{それ以外;} \end{cases} \quad (8-426)$$

ここで、 $w$  の最初と 2 番目の下付き添字はそれぞれ以下で規定される行列の行と列のインデックスである。

$$w = \begin{bmatrix} 13107 & 5243 & 8066 \\ 11916 & 4660 & 7490 \\ 10082 & 4194 & 6554 \\ 9362 & 3647 & 5825 \\ 8192 & 3355 & 5243 \\ 7282 & 2893 & 4559 \end{bmatrix} \quad (8-427)$$

演算結果の和  $c^s$  は、次式で  $i,j=0..3$  として示される通り、量子化パラメータ  $QS_Y$  により量子化される。

$$c_{ij} = \text{Sign}(c_{ij}^s) * ((\text{Abs}(c_{ij}^s) * \text{LevelScale2}(QS_Y \% 6, i, j) + (1 << ((14 + QS_Y / 6)))) >> (15 + QS_Y / 6)) \quad (8-428)$$

8.5.11 小節で規定される残差 4x4 ブロックに対するスケールングと変換処理が、 $c$  を入力として、 $r$  を出力として起動される。

要素  $u_{ij}$  をもつ 4x4 配列  $u$  は以下より導出される。

$$u_{ij} = \text{Clip1}_Y(r_{ij}), \quad i, j = 0..3 \quad (8-429)$$

8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、luma4x4BlkIdx と  $u$  を入力として起動される。

### 8.6.1.2 色差変換係数の復号処理

この処理への入力は、8.4 節からの現マクロブロックに対する INTER 予測色差サンプルと、予測残差変換係数レベル ChromaDCLevel、および ChromaACLevel である。

この処理は2度起動される。1つはCb成分に対するもので、もう1つはCr成分に対するものである。Cb成分に対してはCをCbに置き換え、Cr成分に対してはCをCrに置き換えることにより成分が参照される。iCbCrが現色差成分を選択するとする。

chroma4x4BlkIdxを0..3に等しくし、chroma4x4BlkIdxを用いて索引される現色差成分の各4×4ブロックに対して次が適用される。

- マクロブロック内のchroma4x4BlkIdxのインデックスをもつ4×4色差ブロックの左上サンプルの位置は、次の通り導出される。

$$x = \text{InverseRasterScan}(\text{chroma4x4BlkIdx}, 4, 4, 8, 0) \quad (8-430)$$

$$y = \text{InverseRasterScan}(\text{chroma4x4BlkIdx}, 4, 4, 8, 1) \quad (8-431)$$

- pを以下で導出される要素 $p_{ij}$ を持つ予測サンプルの4×4配列とする。

$$p_{ij} = \text{pred}_c[x+j, y+i], \quad i, j = 0..3 \quad (8-432)$$

- 4×4配列pは、等式8-423を用いて変換され、変換係数 $c^p(\text{chroma4x4BlkIdx})$ を生成する。
- 16エントリのリストである変数chromaListが導出される。chromaList[0]は0に等しく設定される。chromaList[k]、インデックス $k=1..15$ は、以下で規定される。

$$\text{chromaList}[k] = \text{ChromaACLevel}[iCbCr][\text{chroma4x4BlkIdx}][k-1] \quad (8-433)$$

- 8.5.6小節で記載される変換係数に対する逆走査処理が、chromaListを入力として、4×4配列 $c^f$ を出力として起動される。
- 予測残差変換係数 $c^f$ は、 $i=0, j=0$ の組み合わせを除いた $i, j=0..3$ で以下で示される通り、量子化パラメータ $QP_C$ を用いてスケーリングされ予測ブロックの変換係数 $c^p$ に加算される。

$$c_{ij}^s = c_{ij}^p(\text{chroma4x4BlkIdx}) + (((c_{ij}^f * \text{LevelScale}(QP_C \% 6, i, j) * A_{ij}) << (QP_C / 6)) >> 10) \quad (8-434)$$

- 結果の和 $c^s$ は、 $i=0, j=0$ の組み合わせを除いた $i, j=0..3$ で次式で示される通り、量子化パラメータ $QS_C$ により量子化される。 $c_{00}(\text{chroma4x4BlkIdx})$ の導出はこの小小節で以後記載される。

$$c_{ij}(\text{chroma4x4BlkIdx}) = (\text{Sign}(c_{ij}^s) * (\text{Abs}(c_{ij}^s) * \text{LevelScale2}(QS_C \% 6, i, j) + (1 << (14 + QS_C / 6)))) >> (15 + QS_C / 6) \quad (8-435)$$

- 8.5.11小節で規定される残差4×4ブロックに対するスケーリングと変換処理が、 $c(\text{chroma4x4BlkIdx})$ を入力として、rを出力として起動される。
- 要素 $u_{ij}$ をもつ4×4配列uが以下より導出される。

$$u_{ij} = \text{Clip1}_c(r_{ij}), \quad i, j = 0..3 \quad (8-436)$$

- 8.5.13小節のデブロックフィルタ処理前のピクチャ構築処理が、chroma4x4BlkIdxとuを入力として起動される。

DC変換係数レベル $c_{00}(\text{chroma4x4BlkIdx})$ の導出が以下に規定される。マクロブロックの現要素の4つの予測色差4×4ブロックのDC変換係数は、要素 $c_{00}^p(\text{chroma4x4BlkIdx})$ を有する2×2行列に組み立てられ、その

DC 変換係数に 2×2 変換が次の通り適用される。

$$dc^p = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} c_{00}^p(0) & c_{00}^p(1) \\ c_{00}^p(2) & c_{00}^p(3) \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (8-437)$$

色差 DC 予測残差変換係数レベル ChromaDCLevel[iCbCr][k]、k=0..3 は次の通り、量子化パラメータ QP を用いてスケールングされ、予測 DC 変換係数に加算される。

$$dc_{ij}^s = dc_{ij}^p + (((\text{ChromaDCLevel}[iCbCr][j*2+i]*\text{LevelScale}(\text{QP}_c\%6,0,0)*A_{00}) \ll ((\text{QP}_c/6)) \gg 9), i,j=0,1 \quad (8-438)$$

2×2 配列  $dc^s$  は次の通り量子化パラメータ  $QSc$  を用いて量子化される。

$$dc_{ij}^r = (\text{Sign}(dc_{ij}^s) * (\text{Abs}(dc_{ij}^s) * \text{LevelScale2}(QSc\%6,0,0) + (1 \ll ((15+QSc/6)))) \gg (16+QSc/6), i,j=0,1 \quad (8-439)$$

要素  $f_{ij}$ 、 $i,j=0..1$  を有する 2×2 配列  $f$  は、次の通り導出される。

$$f = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} dc_{00}^r & dc_{01}^r \\ dc_{10}^r & dc_{11}^r \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \quad (8-440)$$

$f$  の要素  $f_{ij}$  のスケールングが次の通り実行される。

$$c_{00}(j*2+i) = (f_{ij} * \text{LevelScale}(QSc\%6,0,0)) \ll ((QSc/6)) \gg 5, i,j=0,1 \quad (8-441)$$

## 8.6.2 スイッチングピクチャに対するSPとSIスライスの復号処理

この処理は、sp\_for\_switch\_flag が 1 に等しい SP スライスの P マクロブロックタイプの復号時、および SI スライス内の SI マクロブロックの復号時に起動される。

この処理への入力には現マクロブロックに対する、予測残差変換係数レベル、および予測サンプル配列  $pred_L$ 、 $pred_{Cb}$  および  $pred_{Cr}$  である。

### 8.6.2.1 輝度変換係数の復号処理

この処理への入力には予測輝度サンプル  $pred_L$  と輝度予測残差変換係数レベル LumaLevel である。

要素  $p_{ij}$ 、 $i,j=0..3$  を有する 4×4 配列  $p$  は 8.6.1.1 小小節の通りに導出され、変換係数  $c^p$  を生成するために等式 8-423 に基づいて変換される。これらの変換係数は、次に以下に示す通り量子化パラメータ  $QSY$  を用いて量子化される。

$$c_{ij}^s = \text{Sign}(c_{ij}^p) * ((\text{Abs}(c_{ij}^p) * \text{LevelScale2}(QSY\%6,i,j) + (1 \ll ((14+QSY/6)))) \gg (15+QSY/6)), i,j=0..3 \quad (8-442)$$

8.5.6 小節で記載される変換係数に対する逆走査処理が、LumaLevel[luma4x4BlkIdx] を入力として、要素  $c_{ij}^r$  をもつ 2 次元配列  $c^r$  を出力として起動される。

要素  $c_{ij}$ 、 $i,j=0..3$  をもつ 4×4 配列  $c$  は以下より導出される。

$$c_{ij} = c_{ij}^r + c_{ij}^s, i,j=0..3 \quad (8-443)$$

8.5.11 小節で規定される残差 4×4 ブロックに対するスケーリングと変換処理が、c を入力として、r を出力として起動される。

要素  $u_{ij}$  をもつ 4×4 配列  $u$  は以下より導出される。

$$u_{ij} = \text{Clip1}_Y(r_{ij}), i, j = 0..3 \quad (8-444)$$

8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、luma4x4BlkIdx と  $u$  を入力として起動される。

### 8.6.2.2 色差変換係数の復号処理

この処理への入力は、8.4 節からの現マクロブロックに対する予測色差サンプルと、予測残差変換係数レベル、ChromaDCLevel、および ChromaACLevel である。

この処理は 2 度起動される。1 つは Cb 成分に対するもので、もう 1 つは Cr 成分に対するものである。Cb 成分に対しては C を Cb に置き換え、Cr 成分に対しては C を Cr に置き換えることにより成分が参照される。iCbCr が現色差成分を選択するとする。

chroma4x4BlkIdx を 0..3 に等しくした chroma4x4BlkIdx を用いて索引される現色差成分の各 4×4 ブロックに対して次が適用される。

1. 要素  $p_{ij}$ 、 $i, j = 0..3$  を有する 4×4 配列  $p$  は 8.6.1.2 小節の通りに導出され、変換係数  $c^p(\text{chroma4x4BlkIdx})$  を生成するために等式 8-423 に基づいて変換される。次に、これらの変換係数は、 $i=0, j=0$  の組み合わせを除いた  $i, j = 0..3$  で、以下に示す通り量子化パラメータ  $QS_C$  を用いて量子化される。 $c_{00}^p(\text{chroma4x4BlkIdx})$  の処理はこの小節で以後記載される。

$$c_{ij}^s = (\text{Sign}(c_{ij}^p(\text{chroma4x4BlkIdx})) * (\text{Abs}(c_{ij}^p(\text{chroma4x4BlkIdx})) * \text{LevelScale2}(QS_C \% 6, i, j) + (1 << (14 + QS_C / 6)))) >> (15 + QS_C / 6) \quad (8-445)$$

- 16 エントリのリストである変数 chromaList が導出される。chromaList[0] は 0 に等しく設定される。chromaList[k]、インデックス  $k=1..15$  は、以下で規定される。

$$\text{chromaList}[k] = \text{ChromaACLevel}[iCbCr][\text{chroma4x4BlkIdx}][k-1] \quad (8-446)$$

- 8.5.6 小節に記載される変換係数に対する逆走査処理が、chromaList を入力として、要素に  $c_{ij}^r(\text{chroma4x4BlkIdx})$  を有する 2 次元配列  $c^r(\text{chroma4x4BlkIdx})$  を出力として起動される。
- $i=0, j=0$  の組み合わせを除いた  $i, j = 0..3$  で示される要素  $c_{ij}(\text{chroma4x4BlkIdx})$  を有する 4×4 配列  $c(\text{chroma4x4BlkIdx})$  は以下で導出される。 $c_{00}(\text{chroma4x4BlkIdx})$  の導出は以後記載される。

$$c_{ij}(\text{chroma4x4BlkIdx}) = c_{ij}^r(\text{chroma4x4BlkIdx}) + c_{ij}^s \quad (8-447)$$

- 8.5.11 小節で規定される残差 4×4 ブロックに対するスケーリングと変換処理が、 $c(\text{chroma4x4BlkIdx})$  を入力として、r を出力として起動される。
- 要素  $u_{ij}$  をもつ 4×4 配列  $u$  は以下より導出される。

$$u_{ij} = \text{Clip1}_C(r_{ij}), i, j = 0..3 \quad (8-448)$$

- 8.5.13 小節のデブロックフィルタ処理前のピクチャ構築処理が、chroma4x4BlkIdx と  $u$  を入力として起動

される。

DC 変換係数レベル  $c_{00}(\text{chroma4x4BlkIdx})$  の導出が以下に規定される。マクロブロックの現成分の 4 つの予測 4x4 色差ブロックの DC 変換係数  $c_{00}^p(\text{chroma4x4BlkIdx})$  は、2x2 行列に組み立てられ、これらのブロックの DC 変換係数に対して、DC 変換係数  $dc_{ij}^p$  を結果とする等式 8-437 に基づいた 2x2 変換が適用される。

次に、これらの DC 変換係数は以下で与えられる通り、量子化パラメータ  $QS_C$  を用いて量子化される。

$$dc_{ij}^s = (\text{Sign}(dc_{ij}^p) * (\text{Abs}(dc_{ij}^p) * \text{LevelScale2}(QS_C \% 6, 0, 0) + (1 << (15 + QS_C / 6)))) >> (16 + QS_C / 6), i, j = 0, 1 \quad (8-449)$$

構文解析された色差 DC 予測残差変換係数  $\text{ChromaDCLevel}[i\text{CbCr}][k]$ 、 $k=0..3$  は以下で与えられる通り、予測ブロックのこれら量子化された DC 変換係数に加算される。

$$dc_{ij}^f = dc_{ij}^s + \text{ChromaDCLevel}[i\text{CbCr}][j*2+i], i, j = 0, 1 \quad (8-450)$$

要素  $f_{ij}$ 、 $i, j = 0..1$  を有する 2x2 配列  $f$  は等式 8-440 を用いて導出される。

要素  $f_{ij}$ 、 $i, j = 0..1$  を有する 2x2 配列  $f$  は以下の通り複製される。

$$c_{00}(j*2+i) = f_{ij}, i, j = 0, 1 \quad (8-451)$$

## 8.7 デブロックフィルタ処理

条件付きフィルタ処理は、基本、拡張、メイン、ハイ、ハイ 10、ハイ 4:2:2 およびハイ 4:4:4 予測プロファイルに適合する復号器により適用されなければならない復号処理の必須部分であるこの節において規定される。ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA および CAVLC 4:4:4 INTRA プロファイルに適合する復号器については、この節に規定されるフィルタ処理、またはそれに類似したものは、適用されるべきだが要求されない。

条件付きフィルタ処理が、ピクチャ境界のエッジと `disable_deblocking_filter_idc` によってデブロックフィルタ処理が無効となるエッジを除き、ピクチャの全ての  $N \times N$  ブロックエッジ（ここで、輝度においては  $N=4$  または  $N=8$ 、`ChromaArrayType` が 1 または 2 に等しい時、色差においては  $N=4$ 、`ChromaArrayType` が 3 に等しい時、色差においては  $N=4$  または  $N=8$ ）に対して以下で規定されるように適用される。このフィルタ処理は、復号ピクチャ全体に対する(8.5 と 8.6 節で規定されるような)デブロックフィルタ処理前のピクチャ構築処理の終了の後、マクロブロックアドレスの増加する順番で、処理されたピクチャ内の全てのマクロブロックに対して、マクロブロック単位に基づき実施される。

記1 各マクロブロックに対するデブロックフィルタ処理を操作する前に、現マクロブロックの、上側のマクロブロックあるいはマクロブロックペア(あれば)、および左側のマクロブロックあるいはマクロブロックペア(あれば)のデブロックされるサンプルは、復号ピクチャ全体に対するデブロックフィルタ処理前のピクチャ構築処理の終了の後、デブロックフィルタ処理が行われるため、常に利用可能である。しかしながら、`disable_deblocking_filter_idc` が 2 に等しい場合、どのエッジがフィルタされるのかを決定する目的のために、異なるスライスにあるマクロブロックは、デブロックフィルタ処理演算の規定されたステップの間は利用可能ではないとみなされる。

デブロックフィルタ処理は、輝度と色差成分で別々に起動される。各マクロブロックと各成分に対して、最初に垂直エッジが、マクロブロックの左手側のエッジで始まり、それらの幾何的な順序でマクロブロックの右手側に向かってエッジを次々に進みながらフィルタされ、次に水平エッジが、マクロブロックの最上部のエッジで始まり、それらの幾何的な順序でマクロブロックの最下部に向かってエッジを次々に進みながらフィルタされる。図 8-10/JT-H264 は、輝度または色差エッジとして解釈されうるマクロブロックのエッジ

を示す。

図 8-10/JT-H264 のエッジを輝度エッジとして解釈する場合、`transform_size_8x8_flag` に依存して、次が適用される。

- もし `transform_size_8x8_flag` が 0 に等しいならば、太実線と太破線の輝度エッジがフィルタされる。
- それ以外 (`transform_size_8x8_flag` が 1 に等しい)、太実線の輝度エッジのみがフィルタされる。

図 8-10/JT-H264 のエッジを色差エッジとして解釈する場合、`chroma_format_idc` に依存して、次が適用される。

- もし `ChromaArrayType` が 1(4:2:0 フォーマット)に等しいならば、太実線の色差エッジのみがフィルタされる。
- それ以外、もし `ChromaArrayType` が 2(4:2:2 フォーマット)に等しいならば、太実線の垂直色差エッジがフィルタされ、太実線と太破線両方の水平色差エッジがフィルタされる。
- それ以外、もし `ChromaArrayType` が 3(4:4:4 フォーマット)に等しいならば、次が適用される。
  - もし `transform_size_8x8_flag` が 0 に等しいならば、太実線と太破線両方の色差エッジがフィルタされる。
  - それ以外 (`transform_size_8x8_flag` が 1 に等しい)、太実線の色差エッジのみがフィルタされる。
- それ以外 (`ChromaArrayType` が 0)、フィルタされる色差エッジは何も無い。

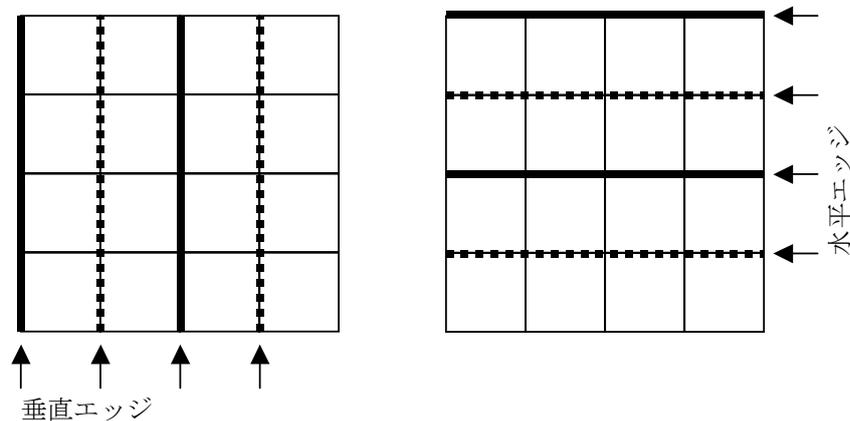


図8-10/JT-H264 フィルタされるマクロブロック内の境界  
(ITU-T H.264)

0 から `PicSizeInMbs - 1` まで進む現マクロブロックアドレス `CurrMbAddr` に対して、次が適用される。

1. 6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、出力は `mbAddrA` と `mbAddrB` に割り当てられる。

2. 変数 `fieldModeMbFlag`、`filterInternalEdgesFlag`、`filterLeftMbEdgeFlag` および `filterTopMbEdgeFlag` は次の通り導出される。
- 変数 `fieldModeMbFlag` は次の通り導出される。
    - もし以下の条件のいずれかが真ならば、`fieldModeMbFlag` は 1 に等しく設定される。
      - `field_pic_flag` が 1 に等しい。
      - `MbaffFrameFlag` が 1 に等しく、かつマクロブロック `CurrMbAddr` がフィールドマクロブロックである。
    - それ以外、`fieldModeMbFlag` は 0 に等しく設定される。
  - 変数 `filterInternalEdgesFlag` は次の通り導出される。
    - もしマクロブロック `CurrMbAddr` を含んだスライスに対する `disable_deblocking_filter_idc` が 1 に等しいならば、変数 `filterInternalEdgesFlag` は 0 に等しく設定される。
    - それ以外(マクロブロック `CurrMbAddr` を含んだスライスに対する `disable_deblocking_filter_idc` が 1 に等しくない)、変数 `filterInternalEdgesFlag` は 1 に等しく設定される。
  - 変数 `filterLeftMbEdgeFlag` は次の通り導出される。
    - もし以下の条件のいずれかが真ならば、変数 `filterLeftMbEdgeFlag` は 0 に等しく設定される。
      - `MbaffFrameFlag` が 0 に等しく、かつ `CurrMbAddr%PicWidthInMbs` が 0 に等しい。
      - `MbaffFrameFlag` が 1 に等しく、かつ  $(CurrMbAddr \gg 1) \% PicWidthInMbs$  が 0 に等しい。
      - マクロブロック `CurrMbAddr` を含んだスライスに対する `disable_deblocking_filter_idc` が 1 に等しい。
      - マクロブロック `CurrMbAddr` を含んだスライスに対する `disable_deblocking_filter_idc` が 2 に等しく、かつマクロブロック `mbAddrA` が利用可能でない。
    - それ以外、変数 `filterLeftMbEdgeFlag` は 1 に等しく設定される。
  - 変数 `filterTopMbEdgeFlag` は次の通り導出される。
    - もし以下の条件のいずれかが真ならば、変数 `filterTopMbEdgeFlag` は 0 に等しく設定される。
      - `MbaffFrameFlag` が 0 に等しく、かつ `CurrMbAddr` が `PicWidthInMbs` より小さい。
      - `MbaffFrameFlag` が 1 に等しく、 $(CurrMbAddr \gg 1)$  が `PicWidthInMbs` より小さく、かつマクロブロック `CurrMbAddr` がフィールドマクロブロックである。
      - `MbaffFrameFlag` が 1 に等しく、 $(CurrMbAddr \gg 1)$  が `PicWidthInMbs` より小さく、かつマクロブロック `CurrMbAddr` がフレームマクロブロックであり、かつ `CurrMbAddr%2` が 0 に等しい。
      - マクロブロック `CurrMbAddr` を含んだスライスに対する `disable_deblocking_filter_idc` が 1 に等しい。

- マクロブロック CurrMbAddr を含んだスライスに対する disable\_deblocking\_filter\_idc が 2 に等しく、かつマクロブロック mbAddrB が利用可能でない。
  - それ以外、変数 filterTopMbEdgeFlag は 1 に等しく設定される。
3. 与えられた変数 fieldModeMbFlag、filterInternalEdgesFlag、filterLeftMbEdgeFlag および filterTopMbEdgeFlag により、デブロックフィルタは次の通り制御される。
- filterLeftMbEdgeFlag が 1 に等しい場合、左部垂直輝度エッジのフィルタは次の通り規定される。
    - 8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=1、fieldModeFilteringFlag=fieldModeMbFlag、および $(xE_k, yE_k)=(0, k)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
  - filterInternalEdgesFlag が 1 に等しい場合、内部垂直輝度エッジのフィルタは次の通り規定される。
    - transform\_size\_8x8\_flag が 0 に等しい場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=1、fieldModeFilteringFlag=fieldModeMbFlag、および $(xE_k, yE_k)=(4, k)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
    - 8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=1、fieldModeFilteringFlag=fieldModeMbFlag、および $(xE_k, yE_k)=(8, k)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
    - transform\_size\_8x8\_flag が 0 に等しい場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=1、fieldModeFilteringFlag=fieldModeMbFlag、および $(xE_k, yE_k)=(12, k)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
  - filterTopMbEdgeFlag が 1 に等しい場合、上部水平輝度エッジのフィルタは次の通り規定される。
    - もし MbaffFrameFlag が 1 に等しく、 $(CurrMbAddr\%2)$  が 0 に等しく、CurrMbAddr が  $2*PicWidthInMbs$  より大きいかに等しく、マクロブロック CurrMbAddr がフレームマクロブロックで、かつマクロブロック  $(CurrMbAddr-2*PicWidthInMbs+1)$  がフィールドマクロブロックであるならば、次が適用される。
      - 8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=0、fieldModeFilteringFlag=1、および $(xE_k, yE_k)=(k, 0)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
      - 8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=0、fieldModeFilteringFlag=1、および $(xE_k, yE_k)=(k, 1)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
    - それ以外、8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および $(xE_k, yE_k)=(k, 0)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
  - filterInternalEdgesFlag が 1 に等しい場合、内部水平輝度エッジのフィルタは次の通り規定される。
    - transform\_size\_8x8\_flag が 0 に等しい場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=0、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および $(xE_k, yE_k)=(k, 4)$ 、 $k=0..15$ を

入力として、 $S'_L$ を出力として起動される。

- 8.7.1 小節で規定される処理が、 $\text{chromaEdgeFlag}=0$ 、 $\text{verticalEdgeFlag}=0$ 、 $\text{fieldModeFilteringFlag}=\text{fieldModeMbFlag}$ 、および $(xE_k, yE_k)=(k, 8)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
- $\text{transform\_size\_8x8\_flag}$ が0に等しい場合、8.7.1小節で規定される処理が、 $\text{chromaEdgeFlag}=0$ 、 $\text{verticalEdgeFlag}=0$ 、 $\text{fieldModeFilteringFlag}=\text{fieldModeMbFlag}$ 、および $(xE_k, yE_k)=(k, 12)$ 、 $k=0..15$ を入力として、 $S'_L$ を出力として起動される。
- ChromaArrayTypeが0に等しくない時、Cbに対する $\text{iCbCr}=0$ とCrに対する $\text{iCbCr}=1$ の両色差成分のフィルタに対して、次が適用される。
  - $\text{filterLeftMbEdgeFlag}$ が1に等しい場合、左部垂直色差エッジのフィルタは次の通り規定される。
  - 8.7.1小節で規定される処理が、 $\text{chromaEdgeFlag}=1$ 、 $\text{iCbCr}$ 、 $\text{verticalEdgeFlag}=1$ 、 $\text{fieldModeFilteringFlag}=\text{fieldModeMbFlag}$ 、および $(xE_k, yE_k)=(0, k)$ 、 $k=0..MbHeightC-1$ を入力として、 $\text{iCbCr}=0$ に対してはCがCbに置き換えられる、 $\text{iCbCr}=1$ に対してはCがCrに置き換えられる、 $S'_c$ を出力として起動される。
  - $\text{filterInternalEdgesFlag}$ が1に等しい場合、内部垂直色差エッジのフィルタは次の通り規定される。
    - ChromaArrayTypeが3に等しくないか $\text{transform\_size\_8x8\_flag}$ が0に等しい場合、8.7.1小節で規定される処理が、 $\text{chromaEdgeFlag}=1$ 、 $\text{iCbCr}$ 、 $\text{verticalEdgeFlag}=1$ 、 $\text{fieldModeFilteringFlag}=\text{fieldModeMbFlag}$ 、および $(xE_k, yE_k)=(4, k)$ 、 $k=0..MbHeightC-1$ を入力として、 $\text{iCbCr}=0$ に対してはCがCbに置き換えられる、 $\text{iCbCr}=1$ に対してはCがCrに置き換えられる、 $S'_c$ を出力として起動される。
    - ChromaArrayTypeが3に等しい場合、8.7.1小節で規定される処理が、 $\text{chromaEdgeFlag}=1$ 、 $\text{iCbCr}$ 、 $\text{verticalEdgeFlag}=1$ 、 $\text{fieldModeFilteringFlag}=\text{fieldModeMbFlag}$ 、および $(xE_k, yE_k)=(8, k)$ 、 $k=0..MbHeightC-1$ を入力として、 $\text{iCbCr}=0$ に対してはCがCbに置き換えられる、 $\text{iCbCr}=1$ に対してはCがCrに置き換えられる、 $S'_c$ を出力として起動される。
    - ChromaArrayTypeが3に等しいか $\text{transform\_size\_8x8\_flag}$ が0に等しい場合、8.7.1小節で規定される処理が、 $\text{chromaEdgeFlag}=1$ 、 $\text{iCbCr}$ 、 $\text{verticalEdgeFlag}=1$ 、 $\text{fieldModeFilteringFlag}=\text{fieldModeMbFlag}$ 、および $(xE_k, yE_k)=(12, k)$ 、 $k=0..MbHeightC-1$ を入力として、 $\text{iCbCr}=0$ に対してはCがCbに置き換えられる、 $\text{iCbCr}=1$ に対してはCがCrに置き換えられる、 $S'_c$ を出力として起動される。
- $\text{filterTopMbEdgeFlag}$ が1に等しい場合、上部水平色差エッジのフィルタは次の通り規定される。
  - もし $\text{MbaffFrameFlag}$ が1に等しく、 $(\text{CurrMbAddr}\%2)$ が0に等しく、 $\text{CurrMbAddr}$ が $2*\text{PicWidthInMbs}$ より大きいか等しく、マクロブロック $\text{CurrMbAddr}$ がフレームマクロブロックで、かつマクロブロック $(\text{CurrMbAddr}-2*\text{PicWidthInMbs}+1)$ がフィールドマクロブロックであるならば、次が適用される。
    - 8.7.1小節で規定される処理が、 $\text{chromaEdgeFlag}=1$ 、 $\text{iCbCr}$ 、 $\text{verticalEdgeFlag}=0$ 、 $\text{fieldModeFilteringFlag}=1$ 、および $(xE_k, yE_k)=(k, 0)$ 、 $k=0..MbWidthC-1$ を入力として、

iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。

- 8.7.1 小節で規定される処理が、chromaEdgeFlag=1、iCbCr、verticalEdgeFlag=0、fieldModeFilteringFlag=1、および(xE<sub>k</sub>,yE<sub>k</sub>)=(k,1)、k=0..MbWidthC-1 を入力として、iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。
- それ以外、8.7.1 小節で規定される処理が、chromaEdgeFlag=1、iCbCr、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および(xE<sub>k</sub>,yE<sub>k</sub>)=(k,0)、k=0..MbWidthC-1 を入力として、iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。
- filterInternalEdgesFlag が 1 に等しい場合、内部水平色差エッジのフィルタは次の通り規定される。
  - ChromaArrayType が 3 に等しくないか transform\_size\_8x8\_flag が 0 に等しい場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=1、iCbCr、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および(xE<sub>k</sub>,yE<sub>k</sub>)=(k,4)、k=0..MbWidthC-1 を入力として、iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。
  - ChromaArrayType が 1 に等しくない場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=1、iCbCr、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および(xE<sub>k</sub>,yE<sub>k</sub>)=(k,8)、k=0..MbWidthC-1 を入力として、iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。
  - ChromaArrayType が 2 に等しい場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=1、iCbCr、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および(xE<sub>k</sub>,yE<sub>k</sub>)=(k,12)、k=0..MbWidthC-1 を入力として、iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。
  - ChromaArrayType が 3 に等しいか transform\_size\_8x8\_flag が 0 に等しい場合、8.7.1 小節で規定される処理が、chromaEdgeFlag=1、iCbCr、verticalEdgeFlag=0、fieldModeFilteringFlag=fieldModeMbFlag、および(xE<sub>k</sub>,yE<sub>k</sub>)=(k,12)、k=0..MbWidthC-1 を入力として、iCbCr=0 に対しては C が Cb に置き換えられる、iCbCr=1 に対しては C が Cr に置き換えられる、S'<sub>c</sub> を出力として起動される。

記2 フィールドモードフィルタ(fieldModeFilteringFlagが1に等しい)が、フレームマクロブロックの上部水平エッジを交差して適用される場合、そのトップまたはボトムマクロブロック境界を交差するこの垂直フィルタは、フレームモードで内部でフィルタされる内部ブロックエッジを交差して拡がるいくつかのサンプルを必要とするかもしれない。

記3 例えば、transform\_size\_8x8\_flagが0に等しい時の4:2:0色差フォーマットの場合において、次が適用される。全ての場合において、3つの水平輝度エッジ、Cbに対する1つの水平色差エッジ、およびCrに対する1つの水平色差エッジはマクロブロックの内部でフィルタされる。フィールドモードフィルタ(fieldModeFilteringFlagが1に等しい)がフレームマクロブロックの上部エッジに適用される場合、フレームマクロブロックと上のマクロブロックペアの間の2つの水平輝度エッジ、Cbに対する2つの水平色差エッジ、およびCrに対する2つの水平色差エッジは、フレームマクロブロックにより制御されると見なされてフィルタされる全部で5つまでの水平輝度エッジ、Cbに対する3つまでの水平色差エッジ、およびCrに対する3つまでの水平色差エッジに対して、フィールドモードフィルタを用いてフィ

ルタされる。他の全ての場合、多くとも、4つの水平輝度エッジ、Cbに対する2つの水平色差エッジ、およびCrに対する2つの水平色差エッジが、特定のマクロブロックにより制御されると見なされてフィルタされる。

separate\_colour\_plane\_flag に依存して、次が適用される。

- もし separate\_colour\_plane\_flag が 0 に等しいならば、配列  $S'_L$ 、 $S'_{Cb}$ 、 $S'_{Cr}$  が配列  $S_L$ 、 $S_{Cb}$ 、 $S_{Cr}$ (復号ピクチャを表現する)にそれぞれ割り当てられる。
- それ以外 (separate\_colour\_plane\_flag が 1 に等しい)、次が適用される。
  - もし colour\_plane\_id が 0 に等しいならば、配列  $S'_L$  は配列  $S_L$  (復号ピクチャの輝度成分を表す) に割り当てられる。
  - それ以外、もし colour\_plane\_id が 1 に等しいならば、配列  $S'_L$  は配列  $S_{Cb}$  (復号ピクチャの Cb 成分を表す) に割り当てられる。
  - それ以外 (colour\_plane\_id が 2 に等しい)、配列  $S'_L$  は配列  $S_{Cr}$  (復号ピクチャの Cr 成分を表す) に割り当てられる。

#### 8.7.1 ブロックエッジにおけるフィルタ処理

この処理への入力 は chromaEdgeFlag、色差成分インデックス iCbCr(chromaEdgeFlag が 1 に等しい場合)、verticalEdgeFlag、fieldModeFilteringFlag、および nE の、マクロブロック CurrMbAddr の左上隅からの相対で表現されたサンプル位置  $(xE_k, yE_k)$ 、 $k=0..nE-1$  の組である。サンプル位置  $(xE_k, yE_k)$  の組は、垂直エッジ (verticalEdgeFlag が 1 に等しい場合)のすぐ右側のサンプル位置、あるいは水平エッジ(verticalEdgeFlag が 0 に等しい場合)のすぐ下側のサンプル位置を示している。

変数 nE は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、nE は 16 に等しく設定される。
- それ以外(chromaEdgeFlag が 1 に等しい)、nE は  $(verticalEdgeFlag == 1) ? MbHeightC : MbWidthC$  に等しく設定される。

s'を輝度または色差サンプル配列を規定する変数とし次の通り導出されるとする。

- もし chromaEdgeFlag が 0 に等しいならば、s'は現ピクチャの輝度サンプル配列  $S'_L$  を示す。
- それ以外、もし chromaEdgeFlag が 1 に等しく、かつ iCbCr が 0 に等しいならば、s'は現ピクチャの色差成分 Cb の色差サンプル配列  $S'_{Cb}$  を示す。
- それ以外(chromaEdgeFlag が 1 に等しく、かつ iCbCr が 1 に等しい)、s'は現ピクチャの色差成分 Cr の色差サンプル配列  $S'_{Cr}$  を示す。

変数 dy は次の通り導出される。

- もし fieldModeFilteringFlag が 1 に等しく、かつ MbaffFrameFlag が 1 に等しいならば、dy は 2 に等しく設定される。
- それ以外(fieldModeFilteringFlag が 0 に等しいか、または MbaffFrameFlag が 0 に等しい)、dy は 1 に等しく設定される。

マクロブロック CurrMbAddr の左上輝度サンプルの位置は、6.4.1 小節の逆マクロブロック走査処理を mbAddr=CurrMbAddr を入力として起動し、出力が(xI,yI)に割り当てられることで導出される。

変数 xP と yP は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、xP は xI に等しく設定され、かつ yP は yI に等しく設定される。
- それ以外(chromaEdgeFlag が 1 に等しい)、xP は xI/SubWidthC に等しく設定され、かつ yP は (yI+SubHeightC-1)/SubHeightC に等しく設定される。

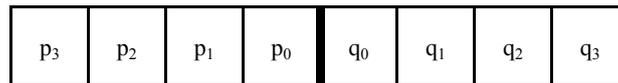


図8-11/JT-H264 4×4ブロック水平または垂直境界を交差するサンプルの記述における約束  
(ITU-T H.264)

各サンプル位置(xE<sub>k</sub>,yE<sub>k</sub>)、k=0..nE-1 に対して以下が適用される。

- p<sub>0</sub> と q<sub>0</sub> の間にエッジがある図 8-11/JT-H264 に示される様な p<sub>i</sub> と q<sub>i</sub>、i=0..3 で示される 4×4 ブロックの水平または垂直エッジを交差する 8 個のサンプルの組にフィルタ処理が適用される。p<sub>i</sub> と q<sub>i</sub>、i=0..3 は次の通り規定される。

- もし verticalEdgeFlag が 1 に等しいならば、

$$q_i = s'[xP + xE_k + i, yP + dy * yE_k] \quad (8-452)$$

$$p_i = s'[xP + xE_k - i - 1, yP + dy * yE_k] \quad (8-453)$$

- それ以外(verticalEdgeFlag が 0 に等しい)、

$$q_i = s'[xP + xE_k, yP + dy * (yE_k + i) - (yE_k \% 2)] \quad (8-454)$$

$$p_i = s'[xP + xE_k, yP + dy * (yE_k - i - 1) - (yE_k \% 2)] \quad (8-455)$$

- 8.7.2 小節で規定される処理が、サンプル値 p<sub>i</sub> と q<sub>i</sub>(i=0..3)、chromaEdgeFlag、verticalEdgeFlag、および fieldModeFilteringFlag を入力として起動され、出力がフィルタされた結果のサンプル値 p'<sub>i</sub> と q'<sub>i</sub>、i=0..2 に割り当てられる。

- 入力サンプル値 p<sub>i</sub> と q<sub>i</sub>、i=0..2 は、サンプル配列 s'内で対応したフィルタ結果のサンプル値 p'<sub>i</sub> と q'<sub>i</sub>、i=0..2 に次の通り置き換えられる。

- もし verticalEdgeFlag が 1 に等しいならば、

$$s'[xP + xE_k + i, yP + dy * yE_k] = q'_i \quad (8-456)$$

$$s'[xP + xE_k - i - 1, yP + dy * yE_k] = p'_i \quad (8-457)$$

- それ以外(verticalEdgeFlag が 0 に等しい)、

$$s'[xP+xE_k,yP+dy*(yE_k+i)-(yE_k\%2)]=q'_i \quad (8-458)$$

$$s'[xP+xE_k,yP+dy*(yE_k-i-1)-(yE_k\%2)]=p'_i \quad (8-459)$$

### 8.7.2 水平または垂直ブロックエッジを交差するサンプルの組に対するフィルタ処理

この処理への入力、フィルタされるエッジを交差するサンプルの1つの組である  $i$  を 0.3 の範囲とした入力サンプル値  $p_i$  と  $q_i$ 、また、chromaEdgeFlag、verticalEdgeFlag、および fieldModeFilteringFlag である。

この処理の出力は、フィルタされた結果のサンプル値  $p'_i$  と  $q'_i$  であり、 $i$  は 0.2 の範囲である。

内容依存境界フィルタ強度変数  $bS$  は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、8.7.2.1 小小節で規定される内容依存境界フィルタ強度に対する導出処理が、 $p_0$ 、 $q_0$ 、および verticalEdgeFlag を入力として起動され、出力が  $bS$  に割り当てられる。
- それ以外(chromaEdgeFlag が 1 に等しい)、水平または垂直の色差エッジのサンプルの組をフィルタするために用いられる  $bS$  は、それぞれ、その同じフィールドの輝度配列内の(SubWidthC\*x,SubHeightC\*y)位置での輝度サンプルを含む水平または垂直の輝度エッジのサンプルの組をフィルタするために用いられる  $bS$  の値に等しく設定される。ここで(x,y)はそのフィールドでの色差配列内の色差サンプル  $q_0$  の位置を示す。

filterOffsetA および filterOffsetB を、サンプル  $q_0$  を含むマクロブロックを含むスライスに対する、7.4.3 小節に規定された FilterOffsetA および FilterOffsetB の値とする。

$qP_p$  および  $qP_q$  を、各々サンプル  $p_0$  および  $q_0$  を含むマクロブロックに対する量子化パラメータ値を規定する変数とする。変数  $qP_z$  (但し  $z$  は  $p$  または  $q$  に置き換えられる) は次のように導出される。

- もし chromaEdgeFlag が 0 に等しいならば、次が適用される。
  - もしサンプル  $z_0$  を含むマクロブロックが I\_PCM マクロブロックならば、 $qP_z$  は 0 に設定される。
  - それ以外 (サンプル  $z_0$  を含むマクロブロックが I\_PCM マクロブロックでない)、 $qP_z$  はサンプル  $z_0$  を含むマクロブロックの  $QP_Y$  の値に設定される。
- それ以外 (chromaEdgeFlag が 1 に等しい)、次が適用される。
  - もしサンプル  $z_0$  を含むマクロブロックが I\_PCM マクロブロックならば、 $qP_z$  は 8.5.8 小節に規定されるような  $QP_Y$  の 0 値に対応する  $QP_C$  の値に等しく設定される。
  - それ以外 (サンプル  $z_0$  を含むマクロブロックが I\_PCM マクロブロックでない)、 $qP_z$  は 8.5.8 小節に規定されるようなサンプル  $z_0$  を含むマクロブロックの  $QP_Y$  値に対応する  $QP_C$  の値に等しく設定される。

8.7.2.2 小小節で規定される処理が、 $p_0$ 、 $q_0$ 、 $p_1$ 、 $q_1$ 、chromaEdgeFlag、 $bS$ 、filterOffsetA、filterOffsetB、 $qP_p$ 、および  $qP_q$  を入力として起動され、出力が filterSamplesFlag、indexA、 $\alpha$ 、および  $\beta$  に割り当てられる。

変数 chromaStyleFiltering が次により設定される。

$$\text{chromaStyleFiltering}=\text{chromaEdgeFlag}\&\&(\text{ChromaArrayType}\neq 3) \quad (8-460)$$

変数 filterSamplesFlag に依存して、次が適用される。

- もし filterSamplesFlag が 1 に等しいならば、次が適用される。

- もし  $bS$  が 4 より小さいならば、8.7.2.3 小小節で規定される処理が、入力として与えられた  $p_i$  と  $q_i(i=0..2)$ 、`chromaEdgeFlag`、`chromaStyleFiltering`、 $bS$ 、 $\beta$ 、および `indexA` で起動され、出力が  $p'_i$  と  $q'_i(i=0..2)$  に割り当てられる。
- それ以外( $bS$  が 4 に等しい)、8.7.2.4 小小節で規定される処理が、入力として与えられた  $p_i$  と  $q_i(i=0..3)$ 、`chromaEdgeFlag`、`chromaStyleFiltering`、 $\alpha$ 、および  $\beta$  で起動され、出力が  $p'_i$  と  $q'_i(i=0..2)$  に割り当てられる。
- それ以外(`filterSamplesFlag` が 0 に等しい)、フィルタされた結果のサンプル  $p'_i$  と  $q'_i(i=0..2)$  は対応した入力サンプル  $p_i$  と  $q_i$  に置き換えられる。

$$i=0..2 \text{ に対して、} \quad p'_i = p_i \quad (8-461)$$

$$i=0..2 \text{ に対して、} \quad q'_i = q_i \quad (8-462)$$

### 8.7.2.1 輝度の内容依存境界フィルタ強度に対する導出処理

この処理への入力は、フィルタされるエッジを交差するサンプルの 1 つの組である入力サンプル値  $p_0$  と  $q_0$ 、および `verticalEdgeFlag` である。

この処理の出力は変数  $bS$  である。

変数 `mixedModeEdgeFlag` が次で導出されるとする。

- もし `MbaffFrameFlag` が 1 に等しく、かつサンプル  $p_0$  と  $q_0$  が異なったマクロブロックペアにあり、どちらか一方がフィールドマクロブロックペアで、他方がフレームマクロブロックペアならば、`mixedModeEdgeFlag` は 1 に等しく設定される。
- それ以外、`mixedModeEdgeFlag` は 0 に等しく設定される。

変数  $bS$  が次の通り導出される。

- もしブロックエッジがマクロブロックエッジでもあり、かつ以下の条件のいずれかが真ならば、4 に等しい  $bS$  の値が出力になる：
  - サンプル  $p_0$  と  $q_0$  が共にフレームマクロブロックにあり、かつ、サンプル  $p_0$  か  $q_0$  のどちらかあるいは両者が INTRA マクロブロック予測モードを用いて符号化されるマクロブロックにある。
  - サンプル  $p_0$  と  $q_0$  が共にフレームマクロブロックにあり、かつ、サンプル  $p_0$  か  $q_0$  のどちらかあるいは両者が SP あるいは SI に等しい `slice_type` を持つスライスのマクロブロックにある。
  - `MbaffFrameFlag` が 1 に等しいか `field_pic_flag` が 1 に等しく、かつ、`verticalEdgeFlag` が 1 に等しく、かつサンプル  $p_0$  か  $q_0$  のどちらかあるいは両者が INTRA マクロブロック予測モードを用いて符号化されるマクロブロックにある。
  - `MbaffFrameFlag` が 1 に等しいか `field_pic_flag` が 1 に等しく、かつ、`verticalEdgeFlag` が 1 に等しく、かつサンプル  $p_0$  か  $q_0$  のどちらかあるいは両者が SP あるいは SI に等しい `slice_type` を持つスライスのマクロブロックにある。
- それ以外、もし以下の条件のいずれかが真ならば、3 に等しい  $bS$  の値が出力になる：

- mixedModeEdgeFlagが0に等しく、かつ、サンプル $p_0$ か $q_0$ のどちらかあるいは両者がINTRAマクロブロック予測モードを用いて符号化されるマクロブロックにある。
- mixedModeEdgeFlagが0に等しく、かつ、サンプル $p_0$ か $q_0$ のどちらかあるいは両者がSPあるいはSIに等しいslice\_typeを持つスライスのマクロブロックにある。
- mixedModeEdgeFlagが1に等しく、verticalEdgeFlagが0に等しく、かつ、サンプル $p_0$ か $q_0$ のどちらかあるいは両者がINTRAマクロマクロブロック予測モードを用いて符号化されるマクロブロックにある。
- mixedModeEdgeFlagが1に等しく、verticalEdgeFlagが0に等しく、かつ、サンプル $p_0$ か $q_0$ のどちらかあるいは両者がSPあるいはSIに等しいslice\_typeを持つスライスのマクロブロックにある。
- それ以外、もし以下の条件が真ならば、2に等しいbSの値が出力になる：
  - サンプル $p_0$ を含む輝度ブロック、あるいはサンプル $q_0$ を含む輝度ブロックが、非ゼロ変換係数レベルを含む
- それ以外、もし以下の条件のいずれかが真ならば、1に等しいbSの値が出力になる：
  - mixedModeEdgeFlagが1に等しい
  - mixedModeEdgeFlagが0に等しく、かつ、サンプル $p_0$ を含んだマクロブロック/サブマクロブロックパーティションの予測において、サンプル $q_0$ を含んだマクロブロック/サブマクロブロックパーティションの予測時とは異なった参照ピクチャ、あるいは異なった数の動きベクトルが用いられる。

記1 2つのマクロブロック/サブマクロブロックパーティションに対して使用される参照ピクチャが同じか異なるかの決定は、予測が参照ピクチャリスト0へのインデックスまたは参照ピクチャリスト1へのインデックスを使用して形成されたかには関係なく、かつ、参照ピクチャリストのインデックス位置が異なるかにも関係なく、どのピクチャが参照されるかのみに基づく。

  - mixedModeEdgeFlagが0に等しく、かつ、1つの動きベクトルがサンプル $p_0$ を含んだマクロブロック/サブマクロブロックのパーティションを予測するために用いられ、かつ、1つの動きベクトルがサンプル $q_0$ を含んだマクロブロック/サブマクロブロックのパーティションを予測するために用いられ、さらに、その用いられる動きベクトルの水平あるいは垂直成分の絶対差分が4分の1輝度フレームサンプル単位で4よりも大きいか等しい。
  - mixedModeEdgeFlagが0に等しく、かつ、2つの動きベクトルおよび2つの異なった参照ピクチャがサンプル $p_0$ を含んだマクロブロック/サブマクロブロックのパーティションを予測するために用いられ、かつ、同一の2つの参照ピクチャに対する2つの動きベクトルがサンプル $q_0$ を含んだマクロブロック/サブマクロブロックのパーティションを予測するために用いられ、さらに、同一参照ピクチャに対する2つのマクロブロック/サブマクロブロックのパーティションの予測において用いられる2つの動きベクトルの水平あるいは垂直成分の絶対差分が4分の1輝度フレームサンプル単位で4に大きいか等しい。
  - mixedModeEdgeFlagが0に等しく、かつ、同一参照ピクチャに対する2つの動きベクトルがサンプル $p_0$ を含んだマクロブロック/サブマクロブロックのパーティションを予測するために用いられ、かつ、同一の参照ピクチャに対する2つの動きベクトルが、サンプル $q_0$ を含んだマクロブロック/サブマクロブロックのパーティションを予測するために用いられ、さらに、次の条件が両方とも真である。

- 2つのマクロブロック／サブマクロブロックのパーティションの予測において用いられるリスト0の動きベクトルの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい、あるいは、2つのマクロブロック／サブマクロブロックのパーティションの予測において用いられるリスト1の動きベクトルの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい
- サンプル $p_0$ を含むマクロブロック／サブマクロブロックのパーティションの予測において用いられるリスト0の動きベクトルと、サンプル $q_0$ を含むマクロブロック／サブマクロブロックのパーティションの予測において用いられるリスト1の動きベクトルの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい、あるいはサンプル $p_0$ を含むマクロブロック／サブマクロブロックのパーティションの予測において用いられるリスト1の動きベクトルと、サンプル $q_0$ を含むマクロブロック／サブマクロブロックのパーティションの予測において用いられるリスト0の動きベクトルの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい。

記2 4分の1輝度フレームサンプル単位で4の垂直差は、4分の1輝度フィールドサンプル単位で2の差である。

- それ以外、0に等しい $bS$ の値が出力になる。

#### 8.7.2.2 各ブロックエッジの閾値に対する導出処理

この処理への入力はこの通り。

- フィルタされるエッジを交差するサンプルの1つの組の入力サンプル値 $p_0$ 、 $q_0$ 、 $p_1$ 、および $q_1$
- 8.7.2小節で規定される、入力サンプルの組に対する変数 $chromaEdgeFlag$ および $bS$
- 変数 $filterOffsetA$ 、 $filterOffsetB$ 、 $qP_p$ 、および $qP_q$

この処理の出力は、入力サンプルがフィルタされるかどうかを示す変数 $filterSamplesFlag$ 、 $indexA$ の値、および、閾値変数 $\alpha$ と $\beta$ の値である。

$qP_{av}$ を平均量子化パラメータを規定する変数とする。これは以下の通り導出される。

$$qP_{av}=(qP_p+qP_q+1)>>1 \quad (8-463)$$

記 SPとSIスライスでは、 $qP_{av}$ は他のスライスタイプと同様の方法にて導出される。等式7-29の $QS_Y$ はデブロックフィルタでは用いられない。

$indexA$ を、 $\alpha$ 表(表 8-16/JT-H264)ならびに、8.7.2.3小節で規定される4より小さい $bS$ のエッジに対するフィルタに用いられる $t_{c0}$ 表(表 8-17/JT-H264)にアクセスするために用いられる変数とする。また $indexB$ を、 $\beta$ 表(表 8-16/JT-H264)にアクセスするために用いられる変数とする。変数 $indexA$ と $indexB$ は以下の通り導出される。

$$indexA=Clip3(0,51,qP_{av}+filterOffsetA) \quad (8-464)$$

$$indexB=Clip3(0,51,qP_{av}+filterOffsetB) \quad (8-465)$$

変数 $\alpha'$ と $\beta'$ は、 $indexA$ と $indexB$ の値に依存して、表 8-16/JT-H264において規定される。 $chromaEdgeFlag$ によって対応する閾値変数 $\alpha$ と $\beta$ は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、

$$\alpha = \alpha' * (1 \ll (\text{BitDepth}_Y - 8)) \quad (8-466)$$

$$\beta = \beta' * (1 \ll (\text{BitDepth}_Y - 8)) \quad (8-467)$$

- それ以外(chromaEdgeFlag が 1 に等しい)、

$$\alpha = \alpha' * (1 \ll (\text{BitDepth}_C - 8)) \quad (8-468)$$

$$\beta = \beta' * (1 \ll (\text{BitDepth}_C - 8)) \quad (8-469)$$

変数 filterSamplesFlag は次の通り導出される。

$$\text{filterSamplesFlag} = (\text{bS} \neq 0 \ \&\& \ \text{Abs}(p_0 - q_0) < \alpha \ \&\& \ \text{Abs}(p_1 - p_0) < \beta \ \&\& \ \text{Abs}(q_1 - q_0) < \beta) \quad (8-470)$$

表8-16/JT-H264 indexAとindexBからのオフセット依存閾値変数 $\alpha'$ と $\beta'$ の導出  
(ITU-T H.264)

		indexA( $\alpha'$ に対する)またはindexB( $\beta'$ に対する)																									
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
$\alpha'$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	4	5	6	7	8	9	10	12	13	
$\beta'$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	2	2	3	3	3	3	4	4	4	

表8-16/JT-H264(完) indexAとindexBからのオフセット依存閾値変数 $\alpha'$ と $\beta'$ の導出  
(ITU-T H.264)

		indexA( $\alpha'$ に対する)またはindexB( $\beta'$ に対する)																													
		26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51				
$\alpha'$	15	17	20	22	25	28	32	36	40	45	50	56	63	71	80	90	101	113	127	144	162	182	203	226	255	255					
$\beta'$	6	6	7	7	8	8	9	9	10	10	11	11	12	12	13	13	14	14	15	15	16	16	17	17	18	18					

### 8.7.2.3 bSが4より小さいエッジに対するフィルタ処理

この処理への入力、8.7.2小節で規定される通り、フィルタされるエッジを交差するサンプルの1つの組の入力サンプル値  $p_i$  と  $q_i(i=0..2)$ 、そして、入力サンプルの組に対する、chromaEdgeFlag、chromaStyleFiltering、bS、 $\beta$ および indexA である。

この処理の出力は、入力サンプル値の組に対してフィルタされた結果のサンプル値  $p'_i$  および  $q'_i(i=0..2)$  である。

Clip1()を chromaStyleFiltering が 0 に等しい時 Clip1<sub>Y</sub>()で、chromaStyleFiltering が 1 に等しい時 Clip1<sub>C</sub>()で置き換えられる関数とする。

フィルタされた結果のサンプル値  $p'_0$  および  $q'_0$  は、以下によって導出される。

$$\Delta = \text{Clip3}(-t_c, t_c, (((q_0 - p_0) < 2) + (p_1 - q_1) + 4) >> 3)) \quad (8-471)$$

$$p'_0 = \text{Clip1}(p_0 + \Delta) \quad (8-472)$$

$$q'_0 = \text{Clip1}(q_0 - \Delta) \quad (8-473)$$

ここで閾値  $t_c$  は次の通り決定される。

- もし chromaStyleFiltering が 0 に等しいならば、

$$t_c = t_{c0} + ((a_p < \beta) ? 1 : 0) + ((a_q < \beta) ? 1 : 0) \quad (8-474)$$

- それ以外 (chromaStyleFiltering が 1 に等しい)、

$$t_c = t_{c0} + 1 \quad (8-475)$$

変数  $t_{c0}$  は、indexA および bS の値に従って、表 8-17/JT-H264 で規定される。chromaEdgeFlag によって対応する閾値変数  $t_{c0}$  は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、

$$t_{c0} = t'_{c0} * (1 << (\text{BitDepth}_Y - 8)) \quad (8-476)$$

- それ以外 (chromaEdgeFlag が 1 に等しい)、

$$t_{c0} = t'_{c0} * (1 << (\text{BitDepth}_C - 8)) \quad (8-477)$$

$a_p$  および  $a_q$  を以下で規定される 2 個の閾値変数とする。

$$a_p = \text{Abs}(p_2 - p_0) \quad (8-478)$$

$$a_q = \text{Abs}(q_2 - q_0) \quad (8-479)$$

フィルタされた結果のサンプル  $p'_1$  は、次の通り導出される。

- もし chromaStyleFiltering が 0 に等しく、かつ  $a_p$  が  $\beta$  より小さいならば、

$$p'_1 = p_1 + \text{Clip3}(-t_{c0}, t_{c0}, (p_2 + ((p_0 + q_0 + 1) >> 1) - (p_1 < 1)) >> 1) \quad (8-480)$$

- それ以外 (chromaStyleFiltering が 1 に等しい、または  $a_p$  が  $\beta$  以上)、

$$p'_1 = p_1 \quad (8-481)$$

フィルタされた結果のサンプル  $q'_1$  は、次の通り導出される。

- もし chromaStyleFiltering が 0 に等しく、かつ  $a_q$  が  $\beta$  よりも小さいならば、

$$q'_1 = q_1 + \text{Clip3}(-t_{c0}, t_{c0}, (q_2 + ((p_0 + q_0 + 1) >> 1) - (q_1 < 1)) >> 1) \quad (8-482)$$

- それ以外 (chromaStyleFiltering が 1 に等しい、または  $a_q$  が  $\beta$  以上)、

$$q'_1 = q_1 \quad (8-483)$$

フィルタされた結果のサンプル  $p'_2$  および  $q'_2$  は、常に入力サンプル  $p_2$  と  $q_2$  とに等しく設定される。

$$p'_2 = p_2 \quad (8-484)$$

$$q'_2 = q_2 \quad (8-485)$$

表8-17/JT-H264 indexAとbSの関数である変数 $t'_{c0}$ の値  
(ITU-T H.264)

	indexA																									
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
bS=1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	
bS=2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
bS=3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1

表8-17/JT-H264 (完) indexAとbSの関数である変数 $t'_{c0}$ の値  
(ITU-T H.264)

	indexA																									
	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51
bS=1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	4	4	4	5	6	6	7	8	9	10	11	13
bS=2	1	1	1	1	1	2	2	2	2	3	3	3	4	4	5	5	6	7	8	8	10	11	12	13	15	17
bS=3	1	2	2	2	2	3	3	3	4	4	4	5	6	6	7	8	9	10	11	13	14	16	18	20	23	25

#### 8.7.2.4 bSが4に等しいエッジに対するフィルタ処理

この処理への入力、8.7.2小節で規定される通り、フィルタされるエッジを交差するサンプルの1つの組の入力サンプル値  $p_i$  と  $q_i(i=0..3)$ 、そして、入力サンプルの組に対する、chromaEdgeFlag、chromaStyleFiltering、および、閾値変数  $\alpha$  と  $\beta$  の値である。

この処理の出力は、入力サンプル値の組に対してフィルタされた結果のサンプル値  $p'_i$  および  $q'_i(i=0..2)$  である。

$a_p$  および  $a_q$  を 8.7.2.3 小節の等式 8-478 および 8-479 で各々規定される 2 個の閾値変数とする。

フィルタされた結果のサンプル  $p'_i(i=0..2)$  は、次の通り導出される。

- もし chromaStyleFiltering が 0 に等しく、かつ次の条件が成立するならば、

$$a_p < \beta \ \&\& \ Abs(p_0 - q_0) < ((\alpha >> 2) + 2) \quad (8-486)$$

変数  $p'_0$ 、 $p'_1$  および  $p'_2$  は以下で導出される。

$$p'_0 = (p_2 + 2 * p_1 + 2 * p_0 + 2 * q_0 + q_1 + 4) >> 3 \quad (8-487)$$

$$p'_1=(p_2+p_1+p_0+q_0+2)>>2 \quad (8-488)$$

$$p'_2=(2*p_3+3*p_2+p_1+p_0+q_0+4)>>3 \quad (8-489)$$

- それ以外 (chromaStyleFiltering が 1 に等しいか、または、等式 8-486 の条件が成立しない)、変数  $p'_0$ 、 $p'_1$  および  $p'_2$  は以下で導出される。

$$p'_0=(2*p_1+p_0+q_1+2)>>2 \quad (8-490)$$

$$p'_1=p_1 \quad (8-491)$$

$$p'_2=p_2 \quad (8-492)$$

フィルタされた結果のサンプル  $q'_i(i=0..2)$ は、次の通り導出される。

- もし chromaStyleFiltering が 0 に等しく、かつ次の条件が成立するならば、

$$a_q < \beta \ \&\& \ Abs(p_0-q_0) < ((\alpha >> 2) + 2) \quad (8-493)$$

変数  $q'_0$ 、 $q'_1$  および  $q'_2$  は以下で導出される。

$$q'_0=(p_1+2*p_0+2*q_0+2*q_1+q_2+4)>>3 \quad (8-494)$$

$$q'_1=(p_0+q_0+q_1+q_2+2)>>2 \quad (8-495)$$

$$q'_2=(2*q_3+3*q_2+q_1+q_0+p_0+4)>>3 \quad (8-496)$$

- それ以外 (chromaStyleFiltering が 1 に等しいか、または、等式 8-493 の条件が成立しない)、変数  $q'_0$ 、 $q'_1$  および  $q'_2$  は以下で導出される。

$$q'_0=(2*q_1+q_0+p_1+2)>>2 \quad (8-497)$$

$$q'_1=q_1 \quad (8-498)$$

$$q'_2=q_2 \quad (8-499)$$

## 9 構文解析処理

この処理への入力は、RBSP からのビットである。

この処理の出力は、シンタックス要素値である。

この処理は、7.3 節のシンタックス表におけるシンタックス要素の記述子が  $ue(v)$ 、 $me(v)$ 、 $se(v)$ 、 $te(v)$  (9.1 節参照)、 $ce(v)$  (9.2 節参照) または  $ae(v)$  (9.3 節参照) に等しい時に起動される。

### 9.1 指数Golomb符号に対する構文解析処理

この処理は、7.3 節のシンタックス表のシンタックス要素の記述子が  $ue(v)$ 、 $me(v)$ 、 $se(v)$  または  $te(v)$  に等しい時に起動される。7.3.4 小節および 7.3.5 小節のシンタックス要素に対して、この処理は  $entropy\_coding\_mode\_flag$  が 0 に等しい時のみ起動される。

この処理への入力は、RBSP からのビットである。

この処理の出力は、シンタックス要素値である。

ue(v)、me(v)または se(v)として符号化されたシンタックス要素は、指数 Golomb 符号である。te(v)として符号化されたシンタックス要素は、切り詰め指数 Golomb 符号である。これらのシンタックス要素に対する構文解析処理は、ビットストリームの現位置から最初の非ゼロビットを含むその位置までビットを読み出し、それまでの 0 に等しい先行ビットの数を数えることから始める。この処理は以下で規定される：

```

leadingZeroBits=-1;
for(b=0;!b;leadingZeroBits++)
    b=read_bits(1)
    
```

変数 codeNum は、その後以下のように割り当てられる：

```
codeNum=2leadingZeroBits-1+read_bits(leadingZeroBits)
```

ここで read\_bits(leadingZeroBits)からの戻り値は、最上位ビットが最初に書かれる符号無し整数のバイナリ表現として解釈される。

表 9-1/JT-H264 はビット記号列を“プリフィックス”および“サフィックス”ビットに分離して指数 Golomb 符号の構造を示している。“プリフィックス”ビットは、leadingZeroBits の計算のために上記の擬似コードにおいて構文解析されたビットであり、表 9-1/JT-H264 におけるビット記号列の欄で 0 か 1 で示される。“サフィックス”ビットは codeNum の計算の中で、構文解析されたビットであり、表 9-1/JT-H264 の中では  $x_i$  と示される。ここで  $i$  は 0 から leadingZeroBits-1 までのそれぞれの値を含む範囲内である。各  $x_i$  は 0 または 1 の値を取ることができる。

**表9-1/JT-H264 “プリフィックス”および“サフィックス”ビットを持つビット記号列とcodeNumの範囲への割り当て (参考)**

(ITU-T H.264)

ビット記号列形式	codeNumの範囲
1	0
0 1 $x_0$	1-2
0 0 1 $x_1 x_0$	3-6
0 0 0 1 $x_2 x_1 x_0$	7-14
0 0 0 0 1 $x_3 x_2 x_1 x_0$	15-30
0 0 0 0 0 1 $x_4 x_3 x_2 x_1 x_0$	31-62
...	...

表 9-2/JT-H264 は、codeNum 値へのビット記号列の割り当てを明示的に示す。

**表9-2/JT-H264 明示的形式の指数Golombビット記号列とcodeNum、そしてue(v)としての使用 (参考)**

(ITU-T H.264)

ビット記号列	codeNum
--------	---------

1	0
010	1
011	2
00100	3
00101	4
00110	5
00111	6
0001000	7
0001001	8
0001010	9
...	...

記述子に従い、シンタックス要素の値は次の通り導出される。

- もしシンタックス要素が  $ue(v)$  として符号化されるならば、シンタックス要素値は `codeNum` に等しい。
- それ以外、もしシンタックス要素が  $se(v)$  として符号化されるならば、`codeNum` を入力として 9.1.1 小節に規定される符号付き指数 Golomb 符号に対するマッピング処理を起動することにより、シンタックス要素値が導出される。
- それ以外、もしシンタックス要素が  $me(v)$  として符号化されるならば、`codeNum` を入力として 9.1.2 小節に規定される符号化ブロックパターンに対するマッピング処理を起動することにより、シンタックス要素値が導出される。
- それ以外（シンタックス要素が  $te(v)$  として符号化される）、シンタックス要素に対する可能な値の範囲が最初に定められる。このシンタックス要素の範囲は 0 から 1 以上の値をもつ  $x$  の間であってよく、その範囲はそのシンタックス要素値の値の導出に次の通り使われる。
  - もし  $x$  が 1 よりも大きいならば、`codeNum` およびシンタックス要素値は、 $ue(v)$  として符号化されたシンタックス要素と同様に導出される。
  - それ以外 ( $x$  が 1 に等しい)、シンタックス要素値に等しい `codeNum` に対する構文解析処理は、以下と等価の処理によって与えられる。

```
b=read_bits(1)
codeNum=!b
```

#### 9.1.1 符号付き指数Golomb符号に対するマッピング処理

この処理への入力、9.1 節で規定された `codeNum` である。

この処理の出力は、 $se(v)$  として符号化されたシンタックス要素値である。

シンタックス要素は、その絶対値の昇順でかつある与えられた絶対値に対して低い `codeNum` で正値を表

現してシンタックス要素を順序付けることで、codeNum に割り当てられる。表 9-3/JT-H264 は割り当て規則を示す。

**表9-3/JT-H264 符号付き指数Golomb符号化シンタックス要素se(v)に対するcodeNumへのシンタックス要素の割り当て (ITU-T H.264)**

codeNum	シンタックス要素値
0	0
1	1
2	-1
3	2
4	-2
5	3
6	-3
k	$(-1)^{k+1} \text{Ceil}(k \div 2)$

#### 9.1.2 符号化ブロックパターンに対するマッピング処理

この処理への入力、9.1 節で規定された codeNum である。

この処理の出力は、me(v)として符号化されたシンタックス要素 coded\_block\_pattern の値である。

表 9-4/JT-H264 は、マクロブロック予測モードが Intra\_4x4、Intra\_8x8 または Inter のいずれであるかに従った、codeNum への coded\_block\_pattern の割り当てを示す。

**表9-4/JT-H264 マクロブロック予測モードに対するcoded\_block\_pattern値へのcodeNumの割り当て (ITU-T H.264)**

(a) ChromaArrayTypeが1または2に等しい

codeNum	coded_block_pattern	
	Intra_4x4、 Intra_8x8	Inter
0	47	0
1	31	16
2	15	1
3	0	2
4	23	4
5	27	8
6	29	32

7	30	3
8	7	5
9	11	10
10	13	12
11	14	15
12	39	47
13	43	7
14	45	11
15	46	13
16	16	14
17	3	6
18	5	9
19	10	31
20	12	35
21	19	37
22	21	42
23	26	44
24	28	33
25	35	34
26	37	36
27	42	40
28	44	39
29	1	43
30	2	45
31	4	46
32	8	17
33	17	18
34	18	20
35	20	24
36	24	19
37	6	21
38	9	26
39	22	28

40	25	23
41	32	27
42	33	29
43	34	30
44	36	22
45	40	25
46	38	38
47	41	41

(b) ChromaArrayTypeが0または3に等しい

codeNum	coded_block_pattern	
	Intra_4x4, Intra_8x8	Inter
0	15	0
1	0	1
2	7	2
3	11	4
4	13	8
5	14	3
6	3	5
7	5	10
8	10	12
9	12	15
10	1	7
11	2	11
12	4	13
13	8	14
14	6	6
15	9	9

## 9.2 変換係数レベルに対するCAVLC構文解析処理

この処理は、構文解析するシンタックス要素の記述子が、7.3.5.3.2 小小小節の  $ce(v)$  と等しく、かつ `entropy_coding_mode_flag` が 0 に等しい時に、起動される。

この処理への入力、スライスデータからのビット、非ゼロ変換係数レベルの最大数 `maxNumCoeff`、変換係数レベルの現ブロックの輝度ブロックインデックス `luma4x4BlkIdx` または色差ブロックインデックス `chroma4x4BlkIdx`、`cb4x4BlkIdx`、または `cr4x4BlkIdx` である。

この処理の出力は、ブロックインデックス `luma4x4BlkIdx` を持つ輝度ブロックまたは、ブロックインデックス `chroma4x4BlkIdx`、`cb4x4BlkIdx`、または `cr4x4BlkIdx` を持つ色差ブロックの変換係数レベルを含む `coeffLevel` リストである。

処理は以下の順序付けられたステップで規定される：

1. `coeffLevel` リストにある 0 から `maxNumCoeff-1` までのインデックスを持つ全ての変換係数レベルが 0 に等しく設定される。
2. 非ゼロ変換係数レベルの総数 `TotalCoeff(coeff_token)` および追従 1 変換係数レベルの数 `TrailingOnes(coeff_token)` は、`coeff_token` の構文解析 (9.2.1 小節を参照) により次の通り導出される。
  - もし非ゼロ変換係数レベルの数 `TotalCoeff(coeff_token)` が 0 に等しいならば、0 の値を持つ `coeffLevel` リストが戻され、このステップ以降には進まない。
  - それ以外、以下のステップが実行される。
    - a. 非ゼロ変換係数レベルは `trailing_ones_sign_flag`、`level_prefix` および `level_suffix` の構文解析により導出される (9.2.2 小節参照)。
    - b. 各非ゼロ変換係数レベル前のゼロ変換係数レベルのランが `total_zeros` および `run_before` の構文解析により導出される (9.2.3 小節参照)。
    - c. レベルおよびラン情報は、結合されて `coeffLevel` リストになる (9.2.4 小節参照)。

### 9.2.1 変換係数レベル総数および追従 1 に対する構文解析処理

この処理への入力、スライスデータからのビット、非ゼロ変換係数レベルの最大数 `maxNumCoeff`、変換係数レベルの現ブロックの輝度ブロックインデックス `luma4x4BlkIdx` または色差ブロックインデックス `chroma4x4BlkIdx`、`cb4x4BlkIdx`、または `cr4x4BlkIdx` である。

この処理の出力は、`TotalCoeff(coeff_token)`、`TrailingOnes(coeff_token)`、および変数 `nC` である。

シンタックス要素 `coeff_token` は、表 9-5/JT-H264 の右端 6 列で規定された 6 個の VLC の 1 つを使って復号される。各 VLC は与えられた符号語 `coeff_token` に対する `TotalCoeff(coeff_token)` および `TrailingOnes(coeff_token)` の両方を規定する。VLC の選択は次の通り導出される変数 `nC` に従う。

- もし CAVLC 構文解析処理が `ChromaDCLevel` に対して起動されるならば、`nC` は次の通り導出される。
  - もし `ChromaArrayType` が 1 に等しいならば、`nC` は -1 に等しく設定される。
  - それ以外、もし `ChromaArrayType` が 2 に等しいならば、`nC` は -2 に等しく設定される。
- それ以外、次が適用される。
  - CAVLC 構文解析処理が `Intra16x16DCLevel` に対して起動される時、`luma4x4BlkIdx` は 0 に等しく設

定される。

- CAVLC 構文解析処理が  $CbIntra16x16DCLevel$  に対して起動される時、 $cb4x4BlkIdx$  は 0 に等しく設定される。
- CAVLC 構文解析処理が  $CrIntra16x16DCLevel$  に対して起動される時、 $cr4x4BlkIdx$  は 0 に等しく設定される。
- 変数  $blkA$  および  $blkB$  は、次の通り導出される。
  - もし CAVLC 構文解析処理が、 $Intra16x16DCLevel$ 、 $Intra16x16ACLevel$  または  $LumaLevel$  に対して起動されるならば、6.4.10.4 小小節で規定される処理が入力として  $luma4x4BlkIdx$  を持って起動され、出力は  $mbAddrA$ 、 $mbAddrB$ 、 $luma4x4BlkIdxA$  および  $luma4x4BlkIdxB$  に割り当てられる。 $mbAddrA \neq luma4x4BlkIdxA$  により規定される  $4 \times 4$  輝度ブロックは  $blkA$  に割り当てられ、 $mbAddrB \neq luma4x4BlkIdxB$  により規定される  $4 \times 4$  輝度ブロックは  $blkB$  に割り当てられる。
  - それ以外、もし CAVLC 構文解析処理が、 $CbIntra16x16DCLevel$ 、 $CbIntra16x16ACLevel$  または  $CbLevel$  に対して起動されるならば、6.4.10.6 小小節で規定される処理が入力として  $cb4x4BlkIdx$  を持って起動され、出力は  $mbAddrA$ 、 $mbAddrB$ 、 $cb4x4BlkIdxA$  および  $cb4x4BlkIdxB$  に割り当てられる。 $mbAddrA \neq cb4x4BlkIdxA$  により規定される  $4 \times 4$  Cb ブロックは  $blkA$  に割り当てられ、 $mbAddrB \neq cb4x4BlkIdxB$  により規定される  $4 \times 4$  Cb ブロックは  $blkB$  に割り当てられる。
  - それ以外、もし CAVLC 構文解析処理が、 $CrIntra16x16DCLevel$ 、 $CrIntra16x16ACLevel$  または  $CrLevel$  に対して起動されるならば、6.4.10.6 小小節で規定される処理が入力として  $cr4x4BlkIdx$  を持って起動され、出力は  $mbAddrA$ 、 $mbAddrB$ 、 $cr4x4BlkIdxA$  および  $cr4x4BlkIdxB$  に割り当てられる。 $mbAddrA \neq cr4x4BlkIdxA$  により規定される  $4 \times 4$  Cr ブロックは  $blkA$  に割り当てられ、 $mbAddrB \neq cr4x4BlkIdxB$  により規定される  $4 \times 4$  Cr ブロックは  $blkB$  に割り当てられる。
  - それ以外 (CAVLC 構文解析処理が  $ChromaACLevel$  に対して起動される)、6.4.10.5 小小節で規定される処理が入力として  $chroma4x4BlkIdx$  を持って起動され、出力は  $mbAddrA$ 、 $mbAddrB$ 、 $chroma4x4BlkIdxA$  および  $chroma4x4BlkIdxB$  に割り当てられる。 $mbAddrA \neq CbCr \neq chroma4x4BlkIdxA$  により規定される  $4 \times 4$  色差ブロックは  $blkA$  に割り当てられ、 $mbAddrB \neq CbCr \neq chroma4x4BlkIdxB$  により規定される  $4 \times 4$  色差ブロックは  $blkB$  に割り当てられる。
- 現ブロックの左側にある変換係数レベルのブロック  $blkA$ 、および現ブロックの上側にある変換係数レベルのブロック  $blkB$  にある ( $TotalCoeff(coeff\_token)$ により与えられる)非ゼロ変換係数レベルの数を各々  $nA$  および  $nB$  とする。
- $mbAddrN$ 、 $blkN$  および  $nN$  における  $N$  を  $A$  および  $B$  に置き換え、次が適用される。
  - もし以下の条件のいずれかが真であるならば、 $nN$  は 0 に等しく設定される。
    - $mbAddrN$  が利用不可である。
    - 現マクロブロックが INTRA 予測モードを使って符号化され、 $constrained\_intra\_pred\_flag$  が 1 に等しく、かつ  $mbAddrN$  が INTER 予測モードで符号化され、かつスライスデータパーティションが使われる ( $nal\_unit\_type$  が 2 から 4 のそれぞれの値を含む範囲内)。
    - マクロブロック  $mbAddrN$  が、 $P\_Skip$  または  $B\_Skip$  に等しい  $mb\_type$  を持つ

- CodedBlockPatternLuma または CodedBlockPatternChroma の対応するビットが 0 に等しいことにより、隣接ブロック blkN の全ての AC 残差変換係数レベルが、0 に等しい
- それ以外、もし mbAddrN が I\_PCM マクロブロックであるならば、nN は 16 に等しく設定される。
- それ以外、nN は、隣接ブロック blkN の TotalCoeff(coeff\_token)値に等しく設定される。

記1 TotalCoeff(coeff\_token)を用いて導出されるnAおよびnB値は、Intra\_16x16マクロブロックにおけるDC変換係数レベルまたは、色差ブロックにおけるDC変換係数レベルを含まない。なぜなら、これらの変換係数レベルは別々に復号されるからである。上側または左側のブロックがIntra\_16x16マクロブロックに属するか、色差ブロックである時は、nAおよびnBは復号された非ゼロAC変換係数レベルの数である。

記2 Intra16x16DCLevel、CbIntra16x16DCLevel、またはCrIntra16x16DCLevelを構文解析する時、nAおよびnB値は、隣接4x4ブロック非ゼロ変換係数レベルの数に基づき、隣接16x16ブロック非ゼロDC変換係数レベルの数に基づかない。

- 与えられた nA および nB 値に対して、変数 nC は次の通り導出される。
  - もし mbAddrA および mbAddrB のいずれも利用可能であるならば、変数 nC は  $(nA+nB+1) \gg 1$  に等しく設定される。
  - それ以外 (mbAddrA が利用不可、または mbAddrB が利用不可である)、変数 nC は  $nA+nB$  に等しく設定される。

coeff\_token を復号して得られる TotalCoeff(coeff\_token)値は、0 から maxNumCoeff のそれぞれの値を含む範囲内でなければならない。

表9-5/JT-H264 TotalCoeff(coeff\_token)およびTrailingOnes(coeff\_token)へのcoeff\_tokenのマッピング (ITU-T H.264)

TrailingOnes (coeff_token)	TotalCoeff (coeff_token)	$0 \leq nC < 2$	$2 \leq nC < 4$	$4 \leq nC < 8$	$8 \leq nC$	$nC == -1$	$nC == -2$
0	0	1	11	1111	0000 11	01	1
0	1	0001 01	0010 11	0011 11	0000 00	0001 11	0001 111
1	1	01	10	1110	0000 01	1	01
0	2	0000 0111	0001 11	0010 11	0001 00	0001 00	0001 110
1	2	0001 00	0011 1	0111 1	0001 01	0001 10	0001 101
2	2	001	011	1101	0001 10	001	001
0	3	0000 0011 1	0000 111	0010 00	0010 00	0000 11	0000 0011 1
1	3	0000 0110	0010 10	0110 0	0010 01	0000 011	0001 100
2	3	0000 101	0010 01	0111 0	0010 10	0000 010	0001 011
3	3	0001 1	0101	1100	0010 11	0001 01	0000 1
0	4	0000 0001 11	0000 0111	0001 111	0011 00	0000 10	0000 0011 0
1	4	0000 0011 0	0001 10	0101 0	0011 01	0000 0011	0000 0010 1

TrailingOnes (coeff_token)	TotalCoeff (coeff_token)	$0 \leq nC < 2$	$2 \leq nC < 4$	$4 \leq nC < 8$	$8 \leq nC$	$nC == -1$	$nC == -2$
2	4	0000 0101	0001 01	0101 1	0011 10	0000 0010	0001 010
3	4	0000 11	0100	1011	0011 11	0000 000	0000 01
0	5	0000 0000 111	0000 0100	0001 011	0100 00	-	0000 0001 11
1	5	0000 0001 10	0000 110	0100 0	0100 01	-	0000 0001 10
2	5	0000 0010 1	0000 101	0100 1	0100 10	-	0000 0010 0
3	5	0000 100	0011 0	1010	0100 11	-	0001 001
0	6	0000 0000 0111 1	0000 0011 1	0001 001	0101 00	-	0000 0000 111
1	6	0000 0000 110	0000 0110	0011 10	0101 01	-	0000 0000 110
2	6	0000 0001 01	0000 0101	0011 01	0101 10	-	0000 0001 01
3	6	0000 0100	0010 00	1001	0101 11	-	0001 000
0	7	0000 0000 0101 1	0000 0001 111	0001 000	0110 00	-	0000 0000 0111
1	7	0000 0000 0111 0	0000 0011 0	0010 10	0110 01	-	0000 0000 0110
2	7	0000 0000 101	0000 0010 1	0010 01	0110 10	-	0000 0000 101
3	7	0000 0010 0	0001 00	1000	0110 11	-	0000 0001 00
0	8	0000 0000 0100 0	0000 0001 011	0000 1111	0111 00	-	0000 0000 0011 1
1	8	0000 0000 0101 0	0000 0001 110	0001 110	0111 01	-	0000 0000 0101
2	8	0000 0000 0110 1	0000 0001 101	0001 101	0111 10	-	0000 0000 0100
3	8	0000 0001 00	0000 100	0110 1	0111 11	-	0000 0000 100
0	9	0000 0000 0011 11	0000 0000 1111	0000 1011	1000 00	-	
1	9	0000 0000 0011 10	0000 0001 010	0000 1110	1000 01	-	

TrailingOnes (coeff_token)	TotalCoeff (coeff_token)	$0 \leq nC < 2$	$2 \leq nC < 4$	$4 \leq nC < 8$	$8 \leq nC$	$nC == -1$	$nC == -2$
2	9	0000 0000 0100 1	0000 0001 001	0001 010	1000 10	-	
3	9	0000 0000 100	0000 0010 0	0011 00	1000 11	-	
0	10	0000 0000 0010 11	0000 0000 1011	0000 0111 1	1001 00	-	
1	10	0000 0000 0010 10	0000 0000 1110	0000 1010	1001 01	-	
2	10	0000 0000 0011 01	0000 0000 1101	0000 1101	1001 10	-	
3	10	0000 0000 0110 0	0000 0001 100	0001 100	1001 11	-	
0	11	0000 0000 0001 111	0000 0000 1000	0000 0101 1	1010 00	-	
1	11	0000 0000 0001 110	0000 0000 1010	0000 0111 0	1010 01	-	
2	11	0000 0000 0010 01	0000 0000 1001	0000 1001	1010 10	-	
3	11	0000 0000 0011 00	0000 0001 000	0000 1100	1010 11	-	
0	12	0000 0000 0001 011	0000 0000 0111 1	0000 0100 0	1011 00	-	
1	12	0000 0000 0001 010	0000 0000 0111 0	0000 0101 0	1011 01	-	
2	12	0000 0000 0001 101	0000 0000 0110 1	0000 0110 1	1011 10	-	
3	12	0000 0000 0010 00	0000 0000 1100	0000 1000	1011 11	-	
0	13	0000 0000 0000 1111	0000 0000 0101 1	0000 0011 01	1100 00	-	
1	13	0000 0000 0000 001	0000 0000 0101 0	0000 0011 1	1100 01	-	
2	13	0000 0000 0001 001	0000 0000 0100 1	0000 0100 1	1100 10	-	
3	13	0000 0000 0001 100	0000 0000 0110 0	0000 0110 0	1100 11	-	
0	14	0000 0000 0000 1011	0000 0000 0011 1	0000 0010 01	1101 00	-	
1	14	0000 0000 0000 1110	0000 0000 0010 11	0000 0011 00	1101 01	-	
2	14	0000 0000 0000 1101	0000 0000 0011 0	0000 0010 11	1101 10	-	
3	14	0000 0000 0001 000	0000 0000 0100 0	0000 0010 10	1101 11	-	
0	15	0000 0000 0000 0111	0000 0000 0010 01	0000 0001 01	1110 00	-	
1	15	0000 0000 0000 1010	0000 0000 0010 00	0000 0010 00	1110 01	-	
2	15	0000 0000 0000 1001	0000 0000 0010 10	0000 0001 11	1110 10	-	
3	15	0000 0000 0000 1100	0000 0000 0000 1	0000 0001 10	1110 11	-	
0	16	0000 0000 0000 0100	0000 0000 0001 11	0000 0000 01	1111 00	-	
1	16	0000 0000 0000 0110	0000 0000 0001 10	0000 0001 00	1111 01	-	
2	16	0000 0000 0000 0101	0000 0000 0001 01	0000 0000 11	1111 10	-	
3	16	0000 0000 0000 1000	0000 0000 0001 00	0000 0000 10	1111 11	-	

## 9.2.2 レベル情報に対する構文解析処理

この処理への入力、スライスデータからのビット、非ゼロ変換係数レベルの数  $TotalCoeff(coeff\_token)$ 、および追従1変換係数レベルの数  $TrailingOnes(coeff\_token)$  である。

この処理の出力は、変換係数レベルを含む、名前  $level$  をもつリストである。

最初にインデックス  $i$  は 0 に等しく設定される。次に追従1変換係数レベル (有れば) を復号するために、 $TrailingOnes(coeff\_token)$  回、以下の手続きが繰り返し適用される：

- 1 ビットのシンタックス要素  $trailing\_ones\_sign\_flag$  が復号され、次の通り評価される。
  - もし  $trailing\_ones\_sign\_flag$  が 0 に等しいならば、値+1 が  $level[i]$  に割り当てられる。
  - それ以外 ( $trailing\_ones\_sign\_flag$  が 1 に等しい) は、値-1 が  $level[i]$  に割り当てられる。
- インデックス  $i$  が 1 加算される。

追従1変換係数レベルの復号に続いて、変数  $suffixLength$  が次の通り初期化される。

- もし  $TotalCoeff(coeff\_token)$  が 10 より大きく、かつ  $TrailingOnes(coeff\_token)$  が 3 より小さいならば、 $suffixLength$  は 1 に等しく設定される。
- それ以外 ( $TotalCoeff(coeff\_token)$  が 10 以下、または  $TrailingOnes(coeff\_token)$  が 3 に等しい)、 $suffixLength$  は 0 に等しく設定される。

残りのレベル (有れば) を復号するために、次に以下の手続きが ( $TotalCoeff(coeff\_token) - TrailingOnes(coeff\_token)$ ) 回繰り返し適用される：

- 9.2.2.1 小小節で規定されるようにシンタックス要素  $level\_prefix$  が復号される。
- 以下の2通りの例外を除いて、変数  $levelSuffixSize$  は変数  $suffixLength$  に等しく設定される。
- $level\_prefix$  が 14 に等しく、かつ  $suffixLength$  が 0 に等しい時、 $levelSuffixSize$  は 4 に等しく設定される。
- $level\_prefix$  が 15 以上の時、 $levelSuffixSize$  は  $level\_prefix - 3$  に等しく設定される。
- シンタックス要素  $level\_suffix$  は、次の通り復号される。
  - もし  $levelSuffixSize$  が 0 より大きいならば、シンタックス要素  $level\_suffix$  は、 $levelSuffixSize$  ビットの符号無し整数表現  $u(v)$  として復号される。
  - それ以外 ( $levelSuffixSize$  が 0 に等しい)、シンタックス要素  $level\_suffix$  は 0 と推定される。
- 変数  $levelCode$  は  $(\text{Min}(15, level\_prefix) \ll suffixLength) + level\_suffix$  に等しく設定される。
- $level\_prefix$  が 15 以上で、かつ  $suffixLength$  が 0 に等しい時、 $levelCode$  は 15 加算される。
- $level\_prefix$  が 16 以上の時、 $levelCode$  は  $(1 \ll (level\_prefix - 3)) - 4096$  加算される。
- インデックス  $i$  が  $TrailingOnes(coeff\_token)$  に等しく、かつ  $TrailingOnes(coeff\_token)$  が 3 より小さい時、 $levelCode$  は 2 加算される。
- 変数  $level[i]$  は次の通り導出される。

- もし levelCode が偶数ならば、値 $(\text{levelCode}+2)\gg 1$  が level[i]に割り当てられる。
- それ以外 (levelCode は奇数)、値 $(-\text{levelCode}-1)\gg 1$  が level[i]に割り当てられる。
- suffixLength が 0 に等しい時、suffixLength が 1 に等しく設定される。
- level[i]の絶対値が $(3\ll(\text{suffixLength}-1))$  より大きく、かつ、suffixLength が 6 より小さい時、suffixLength は 1 加算される。
- インデックス i は 1 加算される。

### 9.2.2.1 level\_prefixに対する構文解析処理

この処理への入力、スライスデータからのビットである。

この処理の出力は、level\_prefix である。

このシンタックス要素に対する構文解析処理は、ビットストリームの現在の位置より始めて最初の非ゼロビットに至るまでそれを含んでビットを読み出し、そして 0 に等しい先行ビット数を数えることである。この処理は、次の通り規定される。

```
leadingZeroBits=-1
```

```
for(b=0;!b;leadingZeroBits++)
```

```
  b=read_bits(1)
```

```
level_prefix=leadingZeroBits
```

表 9-6/JT-H264 は、level\_prefix に対する符号語表を示す。

表9-6/JT-H264 level\_prefixに対する符号語表 (参考)  
(ITU-T H.264)

level_prefix	ビット記号列
0	1
1	01
2	001
3	0001
4	0000 1
5	0000 01
6	0000 001
7	0000 0001
8	0000 0000 1
9	0000 0000 01
10	0000 0000 001

11	0000 0000 0001
12	0000 0000 0000 1
13	0000 0000 0000 01
14	0000 0000 0000 001
15	0000 0000 0000 0001
...	...

### 9.2.3 ラン情報に対する構文解析処理

この処理への入力、スライスデータからのビット、非ゼロ変換係数レベルの数  $TotalCoeff(coeff\_token)$ 、および非ゼロ変換係数レベルの最大数  $maxNumCoeff$  である。

この処理の出力は、run と呼ばれる、非ゼロ変換係数レベルに先行するゼロ変換係数レベルのランのリストである。

最初にインデックス  $i$  は 0 に等しく設定される。

変数  $zerosLeft$  は、次の通り導出される。

- もし非ゼロ変換係数レベルの数  $TotalCoeff(coeff\_token)$  が、非ゼロ変換係数レベルの最大数  $maxNumCoeff$  に等しいならば、変数  $zerosLeft$  は 0 に等しく設定される。
- それ以外（非ゼロ変換係数レベルの数  $TotalCoeff(coeff\_token)$  が非ゼロ変換係数レベルの最大数  $maxNumCoeff$  より小さい）、 $total\_zeros$  が復号され、 $zerosLeft$  がその値に等しく設定される。

変数  $tzVlcIndex$  は  $TotalCoeff(coeff\_token)$  に等しいとする。

$total\_zeros$  の復号に使用される VLC は、次の通り導出される：

- もし  $maxNumCoeff$  が 4 に等しいならば、表 9-9(a)/JT-H264 に規定される VLC の 1 つが使用される。
- それ以外、もし  $maxNumCoeff$  が 8 に等しいならば、表 9-9(b)/JT-H264 に規定される VLC の 1 つが使用される。
- それ以外 ( $maxNumCoeff$  が 4 に等しくなく、かつ 8 に等しくない)、表 9-7/JT-H264 および表 9-8/JT-H264 からの VLC が使用される。

次に以下の手続きが  $(TotalCoeff(coeff\_token)-1)$  回、繰り返し適用される。

- 変数  $run[i]$  が次の通り導出される。
  - もし  $zerosLeft$  が 0 より大きいならば、値  $run\_before$  が表 9-10/JT-H264 および  $zerosLeft$  に基づいて復号される。 $run[i]$  は  $run\_before$  に等しく設定される。
  - それ以外 ( $zerosLeft$  は 0 に等しい)、 $run[i]$  は 0 に等しく設定される。
- $zerosLeft$  から値  $run[i]$  が減算され、結果が  $zerosLeft$  に割り当てられる。減算結果は、0 以上でなければならない。
- インデックス  $i$  は 1 加算される。

最後に zerosLeft の値が run[i]に割り当てられる。

表9-7/JT-H264 tzVlcIndexが1から7の4×4ブロックに対するtotal\_zeros表  
(ITU-T H.264)

total_zeros	tzVlcIndex						
	1	2	3	4	5	6	7
0	1	111	0101	0001 1	0101	0000 01	0000 01
1	011	110	111	111	0100	0000 1	0000 1
2	010	101	110	0101	0011	111	101
3	0011	100	101	0100	111	110	100
4	0010	011	0100	110	110	101	011
5	0001 1	0101	0011	101	101	100	11
6	0001 0	0100	100	100	100	011	010
7	0000 11	0011	011	0011	011	010	0001
8	0000 10	0010	0010	011	0010	0001	001
9	0000 011	0001 1	0001 1	0010	0000 1	001	0000 00
10	0000 010	0001 0	0001 0	0001 0	0001	0000 00	
11	0000 0011	0000 11	0000 01	0000 1	0000 0		
12	0000 0010	0000 10	0000 1	0000 0			
13	0000 0001 1	0000 01	0000 00				
14	0000 0001 0	0000 00					
15	0000 0000 1						

表9-8/JT-H264 tzVlcIndexが8から15の4×4ブロックに対するtotal\_zeros表  
(ITU-T H.264)

total_zeros	tzVlcIndex							
	8	9	10	11	12	13	14	15
0	0000 01	0000 01	0000 1	0000	0000	000	00	0
1	0001	0000 00	0000 0	0001	0001	001	01	1
2	0000 1	0001	001	001	01	1	1	
3	011	11	11	010	1	01		
4	11	10	10	1	001			
5	10	001	01	011				

6	010	01	0001					
7	001	0000 1						
8	0000 00							

表9-9/JT-H264 色差DC2×2と2×4ブロックに対するtotal\_zeros表  
(ITU-T H.264)

(a)色差DC2×2ブロック(4:2:0色差サンプリング)

total_zeros	tzVlcIndex		
	1	2	3
0	1	1	1
1	01	01	0
2	001	00	
3	000		

(b)色差DC2×4ブロック(4:2:2色差サンプリング)

total_zeros	tzVlcIndex						
	1	2	3	4	5	6	7
0	1	000	000	110	00	00	0
1	010	01	001	00	01	01	1
2	011	001	01	01	10	1	
3	0010	100	10	10	11		
4	0011	101	110	111			
5	0001	110	111				
6	0000 1	111					
7	0000 0						

表9-10/JT-H264 run\_beforeに対する表  
(ITU-T H.264)

run_before	zerosLeft						
	1	2	3	4	5	6	>6
0	1	1	11	11	11	11	111
1	0	01	10	10	10	000	110
2	-	00	01	01	011	001	101
3	-	-	00	001	010	011	100
4	-	-	-	000	001	010	011
5	-	-	-	-	000	101	010
6	-	-	-	-	-	100	001
7	-	-	-	-	-	-	0001
8		-	-	-	-	-	00001
9	-	-	-	-	-	-	000001
10	-	-	-	-	-	-	0000001
11	-	-	-	-	-	-	00000001
12	-	-	-	-	-	-	000000001
13	-	-	-	-	-	-	0000000001
14	-	-	-	-	-	-	00000000001

#### 9.2.4 レベルおよびラン情報の結合

この処理への入力は、level と呼ばれる変換係数レベルのリスト、run と呼ばれるランのリスト、および非ゼロ変換係数レベルの数 TotalCoeff(coeff\_token)である。

この処理の出力は、変換係数レベルのリスト coeffLevel である。

変数 coeffNum は-1 に等しく設定され、インデックス i は(TotalCoeff(coeff\_token)-1)に等しく設定される。以下の手続きが、TotalCoeff(coeff\_token)回繰り返して適用される：

- coeffNum が、run[i]+1 加算される。
- coeffLevel[coeffNum]が level[i]に等しく設定される。
- インデックス i は 1 減算される。

#### 9.3 スライスデータに対するCABAC構文解析処理

この処理は、entropy\_coding\_mode\_flag が 1 である時に、7.3.4 および 7.3.5 小節の記述子 ae(v)を持つシンタックス要素を構文解析する時に起動される。

この処理の入力は、シンタックス要素の値に対する要求と、先に構文解析されたシンタックス要素の値である。

この処理の出力はシンタックス要素の値である。

7.3.4 小節のスライスデータの構文解析を始める時、CABAC 構文解析処理の初期化処理が、9.3.1 小節で規定されるように起動される。

シンタックス要素の構文解析は以下の通り進む：

シンタックス要素のそれぞれの要求値において、バイナリ化が 9.3.2 小節で記述されるように導出される。

シンタックス要素と構文解析される bin のシーケンスに対するバイナリ化は、9.3.3 小節で記述されるような復号処理フローを決定する。

変数 binIdx で索引されるシンタックス要素のバイナリ化の各 bin に対して、コンテキストインデックス ctxIdx が 9.3.3.1 小節で規定されるように導出される。

各 ctxIdx に関して、算術復号処理が 9.3.3.2 小節で規定されるように起動される。

構文解析された bin の結果のシーケンス ( $b_0..b_{binIdx}$ ) は、それぞれの bin の復号後にバイナリ化処理により与えられる bin 記号列の集合と比較される。シーケンスが、与えられた集合の bin 記号列に一致する時、対応する値がシンタックス要素に割り当てられる。

シンタックス要素値に対する要求がシンタックス要素 mb\_type に対して処理され、mb\_type の復号値が I\_PCM に等しい場合、9.3.1.2 小節で規定されるようにいくつかの pcm\_alignment\_zero\_bit と全ての pcm\_sample\_luma と pcm\_sample\_chroma データの復号後に、復号エンジンは初期化される。

全体の CABAC 構文解析処理は、シンタックス要素を示す略語 SE を用いて図 9-1/JT-H264 のフローチャートで図示される。

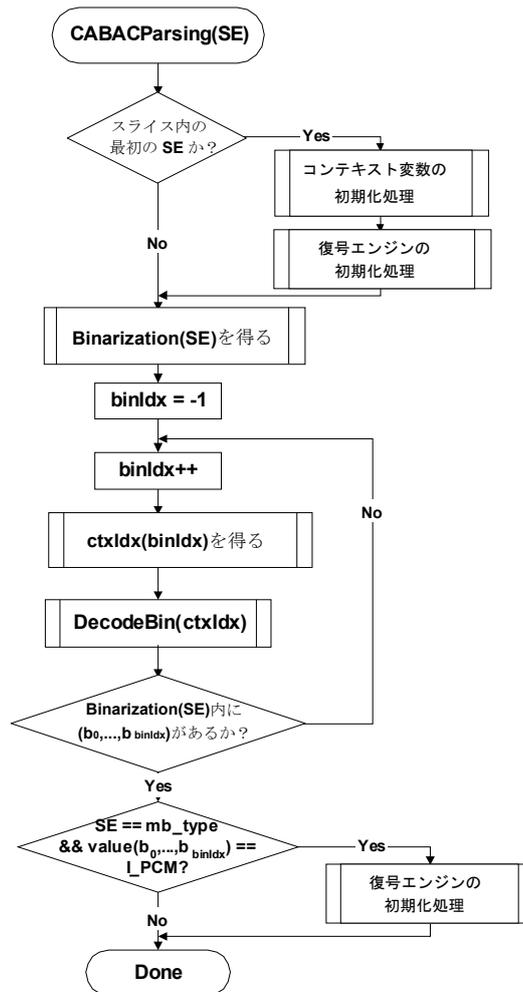


図9-1/JT-H264 シンタックス要素SEに対するCABAC構文解析処理の説明図(参考)  
(ITU-T H.264)

### 9.3.1 初期化処理

この処理の出力は、初期化された CABAC 内部変数である。

9.3.1.1 小小節、9.3.1.2 小小節の処理は、7.3.4 小節のスライスのスライスデータ構文解析を始める時に起動される。

9.3.1.2 小小節の処理はまた、タイプ I\_PCM のマクロブロックにおいて、いくつかの pcm\_alignment\_zero\_bit と全ての pcm\_sample\_luma と pcm\_sample\_chroma データを復号した後にも起動される。

#### 9.3.1.1 コンテキスト変数の初期化処理

この処理の出力は、ctxIdx によって索引される初期化された CABAC コンテキスト変数である。

表 9-12/JT-H264 から表 9-33/JT-H264 は、スライス終了フラグを除く 7.3.4 小節と 7.3.5 小節の全てのシンタックス要素に割り当てられたコンテキスト変数の初期化に用いられる変数 n と m の値を含んでいる。

それぞれのコンテキスト変数において、2 つの変数 pStateIdx と valMPS が初期化される。

記1 9.3.3.2小節でさらに記述されるように変数pStateIdxは確率状態インデックスに対応しており、変数valMPSは優勢シンボルの値に対応している。

初期化のために pStateIdx と valMPS に割り当てられる 2 つの値が、等式 7-28 で得られる SliceQP<sub>Y</sub> から導出される。与えられた 2 つの表のエントリ (m,n) で、以下の通りになる。

1. preCtxState=Clip3(1,126,((m\*Clip3(0,51,SliceQP<sub>Y</sub>))>>4)+n)
2. if(preCtxState<=63){
  - pStateIdx=63-preCtxState
  - valMPS=0
 }else{
  - pStateIdx=preCtxState-64
  - valMPS=1
 }

表 9-11/JT-H264 では、各スライスタイプに対する初期化に必要な ctxIdx が記載される。また、初期化に必要な m と n の値を含む表の番号が記載される。P、SP、および B スライスタイプにおいて、初期化は cabac\_init\_idc シンタックス要素の値にも依存する。シンタックス要素名が初期化処理に影響しないことに注意すること。

**表9-11/JT-H264 初期化処理の各スライスタイプにおけるctxIdxとシンタックス要素の関連 (ITU-T H.264)**

	シンタックス要素	表	スライスタイプ			
			SI	I	P,SP	B
slice_data()	mb_skip_flag	表9-13/JT-H264 表9-14/JT-H264			11-13	24-26
	mb_field_decoding_flag	表9-18/JT-H264	70-72	70-72	70-72	70-72
macroblock_layer()	mb_type	表9-12/JT-H264 表9-13/JT-H264 表9-14/JT-H264	0-10	3-10	14-20	27-35
	transform_size_8x8_flag	表9-16/JT-H264	na	399-401	399-401	399-401
	coded_block_pattern (輝度)	表9-18/JT-H264	73-76	73-76	73-76	73-76
	coded_block_pattern (色差)	表9-18/JT-H264	77-84	77-84	77-84	77-84
	mb_qp_delta	表9-17/JT-H264	60-63	60-63	60-63	60-63
mb_pred()	prev_intra4x4_pred_mode_flag	表9-17/JT-H264	68	68	68	68
	rem_intra4x4_pred_mode	表9-17/JT-H264	69	69	69	69
	prev_intra8x8_pred_mode_flag	表9-17/JT-H264	na	68	68	68
	rem_intra8x8_pred_mode	表9-17/JT-H264	na	69	69	69
	intra_chroma_pred_mode	表9-17/JT-H264	64-67	64-67	64-67	64-67
mb_pred()と	ref_idx_l0	表9-16/JT-H264			54-59	54-59

sub_mb_pred()	ref_idx_l1	表9-16/JT-H264				54-59
	mvd_10[ ][0]	表9-15/JT-H264			40-46	40-46
	mvd_11[ ][0]	表9-15/JT-H264				40-46
	mvd_10[ ][1]	表9-15/JT-H264			47-53	47-53
	mvd_11[ ][1]	表9-15/JT-H264				47-53
sub_mb_pred()	sub_mb_type	表9-13/JT-H264 表9-14/JT-H264			21-23	36-39
residual_block_cabac()	coded_block_flag	表9-18/JT-H264 表9-25/JT-H264 表9-33/JT-H264	85-104 460-483	85-104 460-483 1012-1023	85-104 460-483 1012-1023	85-104 460-483 1012-1023
	significant_coeff_flag[ ]	表9-19/JT-H264 表9-22/JT-H264 表9-24/JT-H264 表9-24/JT-H264 表9-26/JT-H264 表9-30/JT-H264 表9-28/JT-H264 表9-29/JT-H264	105-165 277-337	105-165 277-337 402-416 436-450 484-571 776-863 660-689 718-747	105-165 277-337 402-416 436-450 484-571 776-863 660-689 718-747	105-165 277-337 402-416 436-450 484-571 776-863 660-689 718-747
	last_significant_coeff_flag[ ]	表9-20/JT-H264 表9-23/JT-H264 表9-24/JT-H264 表9-24/JT-H264 表9-27/JT-H264 表9-31/JT-H264 表9-28/JT-H264 表9-29/JT-H264	166-226 338-398	166-226 338-398 417-425 451-459 572-659 864-951 690-707 748-765	166-226 338-398 417-425 451-459 572-659 864-951 690-707 748-765	166-226 338-398 417-425 451-459 572-659 864-951 690-707 748-765
	coeff_abs_level_minus1[ ]	表9-21/JT-H264 表9-24/JT-H264 表9-32/JT-H264 表9-28/JT-H264 表9-29/JT-H264	227-275	227-275 426-435 952-1011 708-717 766-775	227-275 426-435 952-1011 708-717 766-775	227-275 426-435 952-1011 708-717 766-775

記2 276と等しいctxIdxは、end\_of\_slice\_flagとI\_PCMマクロブロックタイプを規定するmb\_typeのbinに関連付けられる。9.3.3.2.4小小小節で規定される復号処理は、276と等しいctxIdxに適用される。しかし、この復号処理もまた、9.3.3.2.1小小小節で規定される復号処理を使用することで実装されてもよい。この場合、276と等しいctxIdxに関連する初期値は、pStateIdx=63とvalMPS=0であると規定される。ここで、pStateIdx=63は非適応確率状態を表す。

表9-12/JT-H264 0から10のctxIdxにおける変数mとnの値  
(ITU-T H.264)

初期化変数	ctxIdx										
	0	1	2	3	4	5	6	7	8	9	10
m	20	2	3	20	2	3	-28	-23	-6	-1	7
n	-15	54	74	-15	54	74	127	104	53	54	51

表9-13/JT-H264 11から23のctxIdxにおける変数mとnの値  
(ITU-T H.264)

cabac_init_idc 値	初期化変数	ctxIdx												
		11	12	13	14	15	16	17	18	19	20	21	22	23
0	m	23	23	21	1	0	-37	5	-13	-11	1	12	-4	17
	n	33	2	0	9	49	118	57	78	65	62	49	73	50
1	m	22	34	16	-2	4	-29	2	-6	-13	5	9	-3	10
	n	25	0	0	9	41	118	65	71	79	52	50	70	54
2	m	29	25	14	-10	-3	-27	26	-4	-24	5	6	-17	14
	n	16	0	0	51	62	99	16	85	102	57	57	73	57

表9-14/JT-H264 24から39のctxIdxにおける変数mとnの値  
(ITU-T H.264)

cabac_init_idc 値	初期化変数	ctxIdx																
		24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	
0	m	18	9	29	26	16	9	-46	-20	1	-13	-11	1	-6	-17	-6	9	
	n	64	43	0	67	90	104	127	104	67	78	65	62	86	95	61	45	
1	m	26	19	40	57	41	26	-45	-15	-4	-6	-13	5	6	-13	0	8	
	n	34	22	0	2	36	69	127	101	76	71	79	52	69	90	52	43	
2	m	20	20	29	54	37	12	-32	-22	-2	-4	-24	5	-6	-14	-6	4	
	n	40	10	0	0	42	97	127	117	74	85	102	57	93	88	44	55	

表9-15/JT-H264 40から53のctxIdxにおける変数mとnの値  
(ITU-T H.264)

cabac_init_idc 値	初期化変数	ctxIdx													
		40	41	42	43	44	45	46	47	48	49	50	51	52	53
0	m	-3	-6	-11	6	7	-5	2	0	-3	-10	5	4	-3	0
	n	69	81	96	55	67	86	88	58	76	94	54	69	81	88
1	m	-2	-5	-10	2	2	-3	-3	1	-3	-6	0	-3	-7	-5
	n	69	82	96	59	75	87	100	56	74	85	59	81	86	95
2	m	-11	-15	-21	19	20	4	6	1	-5	-13	5	6	-3	-1
	n	89	103	116	57	58	84	96	63	85	106	63	75	90	101

表9-16/JT-H264 54から59と399から401のctxIdxにおける変数mとnの値  
(ITU-T H.264)

cabac_init_idc 値	初期化 変数	ctxIdx								
		54	55	56	57	58	59	399	400	401
Iスライス	m	na	na	na	na	na	na	31	31	25
	n	na	na	na	na	na	na	21	31	50
0	m	-7	-5	-4	-5	-7	1	12	11	14
	n	67	74	74	80	72	58	40	51	59
1	m	-1	-1	1	-2	-5	0	25	21	21
	n	66	77	70	86	72	61	32	49	54
2	m	3	-4	-2	-12	-7	1	21	19	17
	n	55	79	75	97	50	60	33	50	61

表9-17/JT-H264 60から69のctxIdxにおける変数mとnの値  
(ITU-T H.264)

初期化変数	ctxIdx									
	60	61	62	63	64	65	66	67	68	69
m	0	0	0	0	-9	4	0	-7	13	3
n	41	63	63	63	83	86	97	72	41	62

表9-18/JT-H264 70から104のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	IとSI スライス		cabac_init_idc値						ctxIdx	IとSI スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
70	0	11	0	45	13	15	7	34	88	-11	115	-13	108	-4	92	5	78
71	1	55	-4	78	7	51	-9	88	89	-12	63	-3	46	0	39	-6	55
72	0	69	-3	96	2	80	-20	127	90	-2	68	-1	65	0	65	4	61
73	-17	127	-27	126	-39	127	-36	127	91	-15	84	-1	57	-15	84	-14	83
74	-13	102	-28	98	-18	91	-17	91	92	-13	104	-9	93	-35	127	-37	127
75	0	82	-25	101	-17	96	-14	95	93	-3	70	-3	74	-2	73	-5	79
76	-7	74	-23	67	-26	81	-25	84	94	-8	93	-9	92	-12	104	-11	104

77	-21	107	-28	82	-35	98	-25	86	95	-10	90	-8	87	-9	91	-11	91
78	-27	127	-20	94	-24	102	-12	89	96	-30	127	-23	126	-31	127	-30	127
79	-31	127	-16	83	-23	97	-17	91	97	-1	74	5	54	3	55	0	65
80	-24	127	-22	110	-27	119	-31	127	98	-6	97	6	60	7	56	-2	79
81	-18	95	-21	91	-24	99	-14	76	99	-7	91	6	59	7	55	0	72
82	-27	127	-18	102	-21	110	-18	103	100	-20	127	6	69	8	61	-4	92
83	-21	114	-13	93	-18	102	-13	90	101	-4	56	-1	48	-3	53	-6	56
84	-30	127	-29	127	-36	127	-37	127	102	-5	82	0	68	0	68	3	68
85	-17	123	-7	92	0	80	11	80	103	-7	76	-4	69	-7	74	-8	71
86	-12	115	-5	89	-5	89	5	76	104	-22	125	-8	88	-9	88	-13	98
87	-16	122	-7	96	-7	94	2	84									

表9-19/JT-H264 105から165のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	IとSI スライス		cabac_init_idc値						ctxIdx	IとSI スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
105	-7	93	-2	85	-13	103	-4	86	136	-13	101	5	53	0	58	-5	75
106	-11	87	-6	78	-13	91	-12	88	137	-13	91	-2	61	-1	60	-8	80
107	-3	77	-1	75	-9	89	-5	82	138	-12	94	0	56	-3	61	-21	83
108	-5	71	-7	77	-14	92	-3	72	139	-10	88	0	56	-8	67	-21	64
109	-4	63	2	54	-8	76	-4	67	140	-16	84	-13	63	-25	84	-13	31
110	-4	68	5	50	-12	87	-8	72	141	-10	86	-5	60	-14	74	-25	64
111	-12	84	-3	68	-23	110	-16	89	142	-7	83	-1	62	-5	65	-29	94
112	-7	62	1	50	-24	105	-9	69	143	-13	87	4	57	5	52	9	75
113	-7	65	6	42	-10	78	-1	59	144	-19	94	-6	69	2	57	17	63
114	8	61	-4	81	-20	112	5	66	145	1	70	4	57	0	61	-8	74
115	5	56	1	63	-17	99	4	57	146	0	72	14	39	-9	69	-5	35
116	-2	66	-4	70	-78	127	-4	71	147	-5	74	4	51	-11	70	-2	27
117	1	64	0	67	-70	127	-2	71	148	18	59	13	68	18	55	13	91
118	0	61	2	57	-50	127	2	58	149	-8	102	3	64	-4	71	3	65
119	-2	78	-2	76	-46	127	-1	74	150	-15	100	1	61	0	58	-7	69
120	1	50	11	35	-4	66	-4	44	151	0	95	9	63	7	61	8	77

121	7	52	4	64	-5	78	-1	69	152	-4	75	7	50	9	41	-10	66
122	10	35	1	61	-4	71	0	62	153	2	72	16	39	18	25	3	62
123	0	44	11	35	-8	72	-7	51	154	-11	75	5	44	9	32	-3	68
124	11	38	18	25	2	59	-4	47	155	-3	71	4	52	5	43	-20	81
125	1	45	12	24	-1	55	-6	42	156	15	46	11	48	9	47	0	30
126	0	46	13	29	-7	70	-3	41	157	-13	69	-5	60	0	44	1	7
127	5	44	13	36	-6	75	-6	53	158	0	62	-1	59	0	51	-3	23
128	31	17	-10	93	-8	89	8	76	159	0	65	0	59	2	46	-21	74
129	1	51	-7	73	-34	119	-9	78	160	21	37	22	33	19	38	16	66
130	7	50	-2	73	-3	75	-11	83	161	-15	72	5	44	-4	66	-23	124
131	28	19	13	46	32	20	9	52	162	9	57	14	43	15	38	17	37
132	16	33	9	49	30	22	0	67	163	16	54	-1	78	12	42	44	-18
133	14	62	-7	100	-44	127	-5	90	164	0	62	0	60	9	34	50	-34
134	-13	108	9	53	0	54	1	67	165	12	72	9	69	0	89	-22	127
135	-15	100	2	53	-5	61	-15	72									

表9-20/JT-H264 166から226のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	IとSI スライス		cabac_init_idc値						ctxIdx	IとSI スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
166	24	0	11	28	4	45	4	39	197	26	-17	28	3	36	-28	28	-3
167	15	9	2	40	10	28	0	42	198	30	-25	28	4	38	-28	24	10
168	8	25	3	44	10	31	7	34	199	28	-20	32	0	38	-27	27	0
169	13	18	0	49	33	-11	11	29	200	33	-23	34	-1	34	-18	34	-14
170	15	9	0	46	52	-43	8	31	201	37	-27	30	6	35	-16	52	-44
171	13	19	2	44	18	15	6	37	202	33	-23	30	6	34	-14	39	-24
172	10	37	2	51	28	0	7	42	203	40	-28	32	9	32	-8	19	17
173	12	18	0	47	35	-22	3	40	204	38	-17	31	19	37	-6	31	25
174	6	29	4	39	38	-25	8	33	205	33	-11	26	27	35	0	36	29
175	20	33	2	62	34	0	13	43	206	40	-15	26	30	30	10	24	33
176	15	30	6	46	39	-18	13	36	207	41	-6	37	20	28	18	34	15

177	4	45	0	54	32	-12	4	47	208	38	1	28	34	26	25	30	20
178	1	58	3	54	102	-94	3	55	209	41	17	17	70	29	41	22	73
179	0	62	2	58	0	0	2	58	210	30	-6	1	67	0	75	20	34
180	7	61	4	63	56	-15	6	60	211	27	3	5	59	2	72	19	31
181	12	38	6	51	33	-4	8	44	212	26	22	9	67	8	77	27	44
182	11	45	6	57	29	10	11	44	213	37	-16	16	30	14	35	19	16
183	15	39	7	53	37	-5	14	42	214	35	-4	18	32	18	31	15	36
184	11	42	6	52	51	-29	7	48	215	38	-8	18	35	17	35	15	36
185	13	44	6	55	39	-9	4	56	216	38	-3	22	29	21	30	21	28
186	16	45	11	45	52	-34	4	52	217	37	3	24	31	17	45	25	21
187	12	41	14	36	69	-58	13	37	218	38	5	23	38	20	42	30	20
188	10	49	8	53	67	-63	9	49	219	42	0	18	43	18	45	31	12
189	30	34	-1	82	44	-5	19	58	220	35	16	20	41	27	26	27	16
190	18	42	7	55	32	7	10	48	221	39	22	11	63	16	54	24	42
191	10	55	-3	78	55	-29	12	45	222	14	48	9	59	7	66	0	93
192	17	51	15	46	32	1	0	69	223	27	37	9	64	16	56	14	56
193	17	46	22	31	0	0	20	33	224	21	60	-1	94	11	73	15	57
194	0	89	-1	84	27	36	8	63	225	12	68	-2	89	10	67	26	38
195	26	-19	25	7	33	-25	35	-18	226	2	97	-9	108	-10	116	-24	127
196	22	-17	30	-7	34	-30	33	-25									

表9-21/JT-H264 227から275のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	IとSI スライス		cabac_init_idc値						ctxIdx	IとSI スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
227	-3	71	-6	76	-23	112	-24	115	252	-12	73	-6	55	-16	72	-14	75
228	-6	42	-2	44	-15	71	-22	82	253	-8	76	0	58	-7	69	-10	79
229	-5	50	0	45	-7	61	-9	62	254	-7	80	0	64	-4	69	-9	83
230	-3	54	0	52	0	53	0	53	255	-9	88	-3	74	-5	74	-12	92
231	-2	62	-3	64	-5	66	0	59	256	-17	110	-10	90	-9	86	-18	108
232	0	58	-2	59	-11	77	-14	85	257	-11	97	0	70	2	66	-4	79
233	1	63	-4	70	-9	80	-13	89	258	-20	84	-4	29	-9	34	-22	69

234	-2	72	-4	75	-9	84	-13	94	259	-11	79	5	31	1	32	-16	75
235	-1	74	-8	82	-10	87	-11	92	260	-6	73	7	42	11	31	-2	58
236	-9	91	-17	102	-34	127	-29	127	261	-4	74	1	59	5	52	1	58
237	-5	67	-9	77	-21	101	-21	100	262	-13	86	-2	58	-2	55	-13	78
238	-5	27	3	24	-3	39	-14	57	263	-13	96	-3	72	-2	67	-9	83
239	-3	39	0	42	-5	53	-12	67	264	-11	97	-3	81	0	73	-4	81
240	-2	44	0	48	-7	61	-11	71	265	-19	117	-11	97	-8	89	-13	99
241	0	46	0	55	-11	75	-10	77	266	-8	78	0	58	3	52	-13	81
242	-16	64	-6	59	-15	77	-21	85	267	-5	33	8	5	7	4	-6	38
243	-8	68	-7	71	-17	91	-16	88	268	-4	48	10	14	10	8	-13	62
244	-10	78	-12	83	-25	107	-23	104	269	-2	53	14	18	17	8	-6	58
245	-6	77	-11	87	-25	111	-15	98	270	-3	62	13	27	16	19	-2	59
246	-10	86	-30	119	-28	122	-37	127	271	-13	71	2	40	3	37	-16	73
247	-12	92	1	58	-11	76	-10	82	272	-10	79	0	58	-1	61	-10	76
248	-15	55	-3	29	-10	44	-8	48	273	-12	86	-3	70	-5	73	-13	86
249	-10	60	-1	36	-10	52	-8	61	274	-13	90	-6	79	-1	70	-9	83
250	-6	62	1	38	-10	57	-8	66	275	-14	97	-8	85	-4	78	-10	87
251	-4	65	2	43	-9	58	-7	70									

表9-22/JT-H264 277から337のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	IとSI スライス		cabac_init_idc値						ctxIdx	IとSI スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
277	-6	93	-13	106	-21	126	-22	127	308	-16	96	-1	51	-16	77	-10	67
278	-6	84	-16	106	-23	124	-25	127	309	-7	88	7	49	-2	64	1	68
279	-8	79	-10	87	-20	110	-25	120	310	-8	85	8	52	2	61	0	77
280	0	66	-21	114	-26	126	-27	127	311	-7	85	9	41	-6	67	2	64
281	-1	71	-18	110	-25	124	-19	114	312	-9	85	6	47	-3	64	0	68
282	0	62	-14	98	-17	105	-23	117	313	-13	88	2	55	2	57	-5	78
283	-2	60	-22	110	-27	121	-25	118	314	4	66	13	41	-3	65	7	55
284	-2	59	-21	106	-27	117	-26	117	315	-3	77	10	44	-3	66	5	59
285	-5	75	-18	103	-17	102	-24	113	316	-3	76	6	50	0	62	2	65

286	-3	62	-21	107	-26	117	-28	118	317	-6	76	5	53	9	51	14	54
287	-4	58	-23	108	-27	116	-31	120	318	10	58	13	49	-1	66	15	44
288	-9	66	-26	112	-33	122	-37	124	319	-1	76	4	63	-2	71	5	60
289	-1	79	-10	96	-10	95	-10	94	320	-1	83	6	64	-2	75	2	70
290	0	71	-12	95	-14	100	-15	102	321	-7	99	-2	69	-1	70	-2	76
291	3	68	-5	91	-8	95	-10	99	322	-14	95	-2	59	-9	72	-18	86
292	10	44	-9	93	-17	111	-13	106	323	2	95	6	70	14	60	12	70
293	-7	62	-22	94	-28	114	-50	127	324	0	76	10	44	16	37	5	64
294	15	36	-5	86	-6	89	-5	92	325	-5	74	9	31	0	47	-12	70
295	14	40	9	67	-2	80	17	57	326	0	70	12	43	18	35	11	55
296	16	27	-4	80	-4	82	-5	86	327	-11	75	3	53	11	37	5	56
297	12	29	-10	85	-9	85	-13	94	328	1	68	14	34	12	41	0	69
298	1	44	-1	70	-8	81	-12	91	329	0	65	10	38	10	41	2	65
299	20	36	7	60	-1	72	-2	77	330	-14	73	-3	52	2	48	-6	74
300	18	32	9	58	5	64	0	71	331	3	62	13	40	12	41	5	54
301	5	42	5	61	1	67	-1	73	332	4	62	17	32	13	41	7	54
302	1	48	12	50	9	56	4	64	333	-1	68	7	44	0	59	-6	76
303	10	62	15	50	0	69	-7	81	334	-13	75	7	38	3	50	-11	82
304	17	46	18	49	1	69	5	64	335	11	55	13	50	19	40	-2	77
305	9	64	17	54	7	69	15	57	336	5	64	10	57	3	66	-2	77
306	-12	104	10	41	-7	69	1	67	337	12	70	26	43	18	50	25	42
307	-11	97	7	46	-6	67	0	68									

表9-23/JT-H264 338から398のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	IとSI スライス		cabac_init_idc値						ctxIdx	IとSI スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
338	15	6	14	11	19	-6	17	-13	369	32	-26	31	-4	40	-37	37	-17
339	6	19	11	14	18	-6	16	-9	370	37	-30	27	6	38	-30	32	1
340	7	16	9	11	14	0	17	-12	371	44	-32	34	8	46	-33	34	15
341	12	14	18	11	26	-12	27	-21	372	34	-18	30	10	42	-30	29	15
342	18	13	21	9	31	-16	37	-30	373	34	-15	24	22	40	-24	24	25

<b>343</b>	13	11	23	-2	33	-25	41	-40	<b>374</b>	40	-15	33	19	49	-29	34	22
<b>344</b>	13	15	32	-15	33	-22	42	-41	<b>375</b>	33	-7	22	32	38	-12	31	16
<b>345</b>	15	16	32	-15	37	-28	48	-47	<b>376</b>	35	-5	26	31	40	-10	35	18
<b>346</b>	12	23	34	-21	39	-30	39	-32	<b>377</b>	33	0	21	41	38	-3	31	28
<b>347</b>	13	23	39	-23	42	-30	46	-40	<b>378</b>	38	2	26	44	46	-5	33	41
<b>348</b>	15	20	42	-33	47	-42	52	-51	<b>379</b>	33	13	23	47	31	20	36	28
<b>349</b>	14	26	41	-31	45	-36	46	-41	<b>380</b>	23	35	16	65	29	30	27	47
<b>350</b>	14	44	46	-28	49	-34	52	-39	<b>381</b>	13	58	14	71	25	44	21	62
<b>351</b>	17	40	38	-12	41	-17	43	-19	<b>382</b>	29	-3	8	60	12	48	18	31
<b>352</b>	17	47	21	29	32	9	32	11	<b>383</b>	26	0	6	63	11	49	19	26
<b>353</b>	24	17	45	-24	69	-71	61	-55	<b>384</b>	22	30	17	65	26	45	36	24
<b>354</b>	21	21	53	-45	63	-63	56	-46	<b>385</b>	31	-7	21	24	22	22	24	23
<b>355</b>	25	22	48	-26	66	-64	62	-50	<b>386</b>	35	-15	23	20	23	22	27	16
<b>356</b>	31	27	65	-43	77	-74	81	-67	<b>387</b>	34	-3	26	23	27	21	24	30
<b>357</b>	22	29	43	-19	54	-39	45	-20	<b>388</b>	34	3	27	32	33	20	31	29
<b>358</b>	19	35	39	-10	52	-35	35	-2	<b>389</b>	36	-1	28	23	26	28	22	41
<b>359</b>	14	50	30	9	41	-10	28	15	<b>390</b>	34	5	28	24	30	24	22	42
<b>360</b>	10	57	18	26	36	0	34	1	<b>391</b>	32	11	23	40	27	34	16	60
<b>361</b>	7	63	20	27	40	-1	39	1	<b>392</b>	35	5	24	32	18	42	15	52
<b>362</b>	-2	77	0	57	30	14	30	17	<b>393</b>	34	12	28	29	25	39	14	60
<b>363</b>	-4	82	-14	82	28	26	20	38	<b>394</b>	39	11	23	42	18	50	3	78
<b>364</b>	-3	94	-5	75	23	37	18	45	<b>395</b>	30	29	19	57	12	70	-16	123
<b>365</b>	9	69	-19	97	12	55	15	54	<b>396</b>	34	26	22	53	21	54	21	53
<b>366</b>	-12	109	-35	125	11	65	0	79	<b>397</b>	29	39	22	61	14	71	22	56
<b>367</b>	36	-35	27	0	37	-33	36	-16	<b>398</b>	19	66	11	86	11	83	25	61
<b>368</b>	36	-34	28	0	39	-36	37	-14									

表9-24/JT-H264 402から459のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I スライス		cabac_init_idc値						ctxIdx	I スライス		cabac_init_idc値					
			0		1		2					0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n		
402	-17	120	-4	79	-5	85	-3	78	431	-2	55	-12	56	-9	57	-12	59
403	-20	112	-7	71	-6	81	-8	74	432	0	61	-6	60	-6	63	-8	63
404	-18	114	-5	69	-10	77	-9	72	433	1	64	-5	62	-4	65	-9	67
405	-11	85	-9	70	-7	81	-10	72	434	0	68	-8	66	-4	67	-6	68
406	-15	92	-8	66	-17	80	-18	75	435	-9	92	-8	76	-7	82	-10	79
407	-14	89	-10	68	-18	73	-12	71	436	-14	106	-5	85	-3	81	-3	78
408	-26	71	-19	73	-4	74	-11	63	437	-13	97	-6	81	-3	76	-8	74
409	-15	81	-12	69	-10	83	-5	70	438	-15	90	-10	77	-7	72	-9	72
410	-14	80	-16	70	-9	71	-17	75	439	-12	90	-7	81	-6	78	-10	72
411	0	68	-15	67	-9	67	-14	72	440	-18	88	-17	80	-12	72	-18	75
412	-14	70	-20	62	-1	61	-16	67	441	-10	73	-18	73	-14	68	-12	71
413	-24	56	-19	70	-8	66	-8	53	442	-9	79	-4	74	-3	70	-11	63
414	-23	68	-16	66	-14	66	-14	59	443	-14	86	-10	83	-6	76	-5	70
415	-24	50	-22	65	0	59	-9	52	444	-10	73	-9	71	-5	66	-17	75
416	-11	74	-20	63	2	59	-11	68	445	-10	70	-9	67	-5	62	-14	72
417	23	-13	9	-2	17	-10	9	-2	446	-10	69	-1	61	0	57	-16	67
418	26	-13	26	-9	32	-13	30	-10	447	-5	66	-8	66	-4	61	-8	53
419	40	-15	33	-9	42	-9	31	-4	448	-9	64	-14	66	-9	60	-14	59
420	49	-14	39	-7	49	-5	33	-1	449	-5	58	0	59	1	54	-9	52
421	44	3	41	-2	53	0	33	7	450	2	59	2	59	2	58	-11	68
422	45	6	45	3	64	3	31	12	451	21	-10	21	-13	17	-10	9	-2
423	44	34	49	9	68	10	37	23	452	24	-11	33	-14	32	-13	30	-10
424	33	54	45	27	66	27	31	38	453	28	-8	39	-7	42	-9	31	-4
425	19	82	36	59	47	57	20	64	454	28	-1	46	-2	49	-5	33	-1
426	-3	75	-6	66	-5	71	-9	71	455	29	3	51	2	53	0	33	7
427	-1	23	-7	35	0	24	-7	37	456	29	9	60	6	64	3	31	12
428	1	34	-7	42	-1	36	-8	44	457	35	20	61	17	68	10	37	23
429	1	43	-8	45	-2	42	-11	49	458	29	36	55	34	66	27	31	38
430	0	54	-5	48	-2	52	-10	56	459	14	67	42	62	47	57	20	64

表9-25/JT-H264 460から483のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
460	-17	123	-7	92	0	80	11	80	472	-17	123	-7	92	0	80	11	80
461	-12	115	-5	89	-5	89	5	76	473	-12	115	-5	89	-5	89	5	76
462	-16	122	-7	96	-7	94	2	84	474	-16	122	-7	96	-7	94	2	84
463	-11	115	-13	108	-4	92	5	78	475	-11	115	-13	108	-4	92	5	78
464	-12	63	-3	46	0	39	-6	55	476	-12	63	-3	46	0	39	-6	55
465	-2	68	-1	65	0	65	4	61	477	-2	68	-1	65	0	65	4	61
466	-15	84	-1	57	-15	84	-14	83	478	-15	84	-1	57	-15	84	-14	83
467	-13	104	-9	93	-35	127	-37	127	479	-13	104	-9	93	-35	127	-37	127
468	-3	70	-3	74	-2	73	-5	79	480	-3	70	-3	74	-2	73	-5	79
469	-8	93	-9	92	-12	104	-11	104	481	-8	93	-9	92	-12	104	-11	104
470	-10	90	-8	87	-9	91	-11	91	482	-10	90	-8	87	-9	91	-11	91
471	-30	127	-23	126	-31	127	-30	127	483	-30	127	-23	126	-31	127	-30	127

表9-26/JT-H264 484から571のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
484	-7	93	-2	85	-13	103	-4	86	528	-7	93	-2	85	-13	103	-4	86
485	-11	87	-6	78	-13	91	-12	88	529	-11	87	-6	78	-13	91	-12	88
486	-3	77	-1	75	-9	89	-5	82	530	-3	77	-1	75	-9	89	-5	82
487	-5	71	-7	77	-14	92	-3	72	531	-5	71	-7	77	-14	92	-3	72
488	-4	63	2	54	-8	76	-4	67	532	-4	63	2	54	-8	76	-4	67
489	-4	68	5	50	-12	87	-8	72	533	-4	68	5	50	-12	87	-8	72
490	-12	84	-3	68	-23	110	-16	89	534	-12	84	-3	68	-23	110	-16	89
491	-7	62	1	50	-24	105	-9	69	535	-7	62	1	50	-24	105	-9	69
492	-7	65	6	42	-10	78	-1	59	536	-7	65	6	42	-10	78	-1	59
493	8	61	-4	81	-20	112	5	66	537	8	61	-4	81	-20	112	5	66
494	5	56	1	63	-17	99	4	57	538	5	56	1	63	-17	99	4	57
495	-2	66	-4	70	-78	127	-4	71	539	-2	66	-4	70	-78	127	-4	71
496	1	64	0	67	-70	127	-2	71	540	1	64	0	67	-70	127	-2	71
497	0	61	2	57	-50	127	2	58	641	0	61	2	57	-50	127	2	58
498	-2	78	-2	76	-46	127	-1	74	542	-2	78	-2	76	-46	127	-1	74
499	1	50	11	35	-4	66	-4	44	543	1	50	11	35	-4	66	-4	44
500	7	52	4	64	-5	78	-1	69	544	7	52	4	64	-5	78	-1	69
501	10	35	1	61	-4	71	0	62	545	10	35	1	61	-4	71	0	62
502	0	44	11	35	-8	72	-7	51	546	0	44	11	35	-8	72	-7	51
503	11	38	18	25	2	59	-4	47	547	11	38	18	25	2	59	-4	47
504	1	45	12	24	-1	55	-6	42	548	1	45	12	24	-1	55	-6	42
505	0	46	13	29	-7	70	-3	41	549	0	46	13	29	-7	70	-3	41
506	5	44	13	36	-6	75	-6	53	550	5	44	13	36	-6	75	-6	53
507	31	17	-10	93	-8	89	8	76	551	31	17	-10	93	-8	89	8	76
508	1	51	-7	73	-34	119	-9	78	552	1	51	-7	73	-34	119	-9	78
509	7	50	-2	73	-3	75	-11	83	553	7	50	-2	73	-3	75	-11	83
510	28	19	13	46	32	20	9	52	554	28	19	13	46	32	20	9	52
511	16	33	9	49	30	22	0	67	555	16	33	9	49	30	22	0	67
512	14	62	-7	100	-44	127	-5	90	556	14	62	-7	100	-44	127	-5	90
513	-13	108	9	53	0	54	1	67	557	-13	108	9	53	0	54	1	67
514	-15	100	2	53	-5	61	-15	72	558	-15	100	2	53	-5	61	-15	72
515	-13	101	5	53	0	58	-5	75	559	-13	101	5	53	0	58	-5	75
516	-13	91	-2	61	-1	60	-8	80	560	-13	91	-2	61	-1	60	-8	80
517	-12	94	0	56	-3	61	-21	83	561	-12	94	0	56	-3	61	-21	83
518	-10	88	0	56	-8	67	-21	64	562	-10	88	0	56	-8	67	-21	64
519	-16	84	-13	63	-25	84	-13	31	563	-16	84	-13	63	-25	84	-13	31

<b>520</b>	-10	86	-5	60	-14	74	-25	64	<b>564</b>	-10	86	-5	60	-14	74	-25	64
<b>521</b>	-7	83	-1	62	-5	65	-29	94	<b>565</b>	-7	83	-1	62	-5	65	-29	94
<b>522</b>	-13	87	4	57	5	52	9	75	<b>566</b>	-13	87	4	57	5	52	9	75
<b>523</b>	-19	94	-6	69	2	57	17	63	<b>567</b>	-19	94	-6	69	2	57	17	63
<b>524</b>	1	70	4	57	0	61	-8	74	<b>568</b>	1	70	4	57	0	61	-8	74
<b>525</b>	0	72	14	39	-9	69	-5	35	<b>569</b>	0	72	14	39	-9	69	-5	35
<b>526</b>	-5	74	4	51	-11	70	-2	27	<b>570</b>	-5	74	4	51	-11	70	-2	27
<b>527</b>	18	59	13	68	18	55	13	91	<b>571</b>	18	59	13	68	18	55	13	91

表9-27/JT-H264 572から659のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
572	24	0	11	28	4	45	4	39	616	24	0	11	28	4	45	4	39
573	15	9	2	40	10	28	0	42	617	15	9	2	40	10	28	0	42
574	8	25	3	44	10	31	7	34	618	8	25	3	44	10	31	7	34
575	13	18	0	49	33	-11	11	29	619	13	18	0	49	33	-11	11	29
576	15	9	0	46	52	-43	8	31	620	15	9	0	46	52	-43	8	31
577	13	19	2	44	18	15	6	37	621	13	19	2	44	18	15	6	37
578	10	37	2	51	28	0	7	42	622	10	37	2	51	28	0	7	42
579	12	18	0	47	35	-22	3	40	623	12	18	0	47	35	-22	3	40
580	6	29	4	39	38	-25	8	33	624	6	29	4	39	38	-25	8	33
581	20	33	2	62	34	0	13	43	625	20	33	2	62	34	0	13	43
582	15	30	6	46	39	-18	13	36	626	15	30	6	46	39	-18	13	36
583	4	45	0	54	32	-12	4	47	627	4	45	0	54	32	-12	4	47
584	1	58	3	54	102	-94	3	55	628	1	58	3	54	102	-94	3	55
585	0	62	2	58	0	0	2	58	629	0	62	2	58	0	0	2	58
586	7	61	4	63	56	-15	6	60	630	7	61	4	63	56	-15	6	60
587	12	38	6	51	33	-4	8	44	631	12	38	6	51	33	-4	8	44
588	11	45	6	57	29	10	11	44	632	11	45	6	57	29	10	11	44
589	15	39	7	53	37	-5	14	42	633	15	39	7	53	37	-5	14	42
590	11	42	6	52	51	-29	7	48	634	11	42	6	52	51	-29	7	48
591	13	44	6	55	39	-9	4	56	635	13	44	6	55	39	-9	4	56
592	16	45	11	45	52	-34	4	52	636	16	45	11	45	52	-34	4	52
593	12	41	14	36	69	-58	13	37	637	12	41	14	36	69	-58	13	37
594	10	49	8	53	67	-63	9	49	638	10	49	8	53	67	-63	9	49
595	30	34	-1	82	44	-5	19	58	639	30	34	-1	82	44	-5	19	58
596	18	42	7	55	32	7	10	48	640	18	42	7	55	32	7	10	48
597	10	55	-3	78	55	-29	12	45	641	10	55	-3	78	55	-29	12	45
598	17	51	15	46	32	1	0	69	642	17	51	15	46	32	1	0	69
599	17	46	22	31	0	0	20	33	643	17	46	22	31	0	0	20	33
600	0	89	-1	84	27	36	8	63	644	0	89	-1	84	27	36	8	63
601	26	-19	25	7	33	-25	35	-18	645	26	-19	25	7	33	-25	35	-18
602	22	-17	30	-7	34	-30	33	-25	646	22	-17	30	-7	34	-30	33	-25
603	26	-17	28	3	36	-28	28	-3	647	26	-17	28	3	36	-28	28	-3
604	30	-25	28	4	38	-28	24	10	648	30	-25	28	4	38	-28	24	10
605	28	-20	32	0	38	-27	27	0	649	28	-20	32	0	38	-27	27	0
606	33	-23	34	-1	34	-18	34	-14	650	33	-23	34	-1	34	-18	34	-14
607	37	-27	30	6	35	-16	52	-44	651	37	-27	30	6	35	-16	52	-44

<b>608</b>	33	-23	30	6	34	-14	39	-24	<b>652</b>	33	-23	30	6	34	-14	39	-24
<b>609</b>	40	-28	32	9	32	-8	19	17	<b>653</b>	40	-28	32	9	32	-8	19	17
<b>610</b>	38	-17	31	19	37	-6	31	25	<b>654</b>	38	-17	31	19	37	-6	31	25
<b>611</b>	33	-11	26	27	35	0	36	29	<b>655</b>	33	-11	26	27	35	0	36	29
<b>612</b>	40	-15	26	30	30	10	24	33	<b>656</b>	40	-15	26	30	30	10	24	33
<b>613</b>	41	-6	37	20	28	18	34	15	<b>657</b>	41	-6	37	20	28	18	34	15
<b>614</b>	38	1	28	34	26	25	30	20	<b>658</b>	38	1	28	34	26	25	30	20
<b>615</b>	41	17	17	70	29	41	22	73	<b>659</b>	41	17	17	70	29	41	22	73

表9-28/JT-H264 660から717のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I		cabac_init_idc 値						ctxIdx	I		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
660	-17	120	-4	79	-5	85	-3	78	689	2	59	2	59	2	58	-11	68
661	-20	112	-7	71	-6	81	-8	74	690	23	-13	9	-2	17	-10	9	-2
662	-18	114	-5	69	-10	77	-9	72	691	26	-13	26	-9	32	-13	30	-10
663	-11	85	-9	70	-7	81	-10	72	692	40	-15	33	-9	42	-9	31	-4
664	-15	92	-8	66	-17	80	-18	75	693	49	-14	39	-7	49	-5	33	-1
665	-14	89	-10	68	-18	73	-12	71	694	44	3	41	-2	53	0	33	7
666	-26	71	-19	73	-4	74	-11	63	695	45	6	45	3	64	3	31	12
667	-15	81	-12	69	-10	83	-5	70	696	44	34	49	9	68	10	37	23
668	-14	80	-16	70	-9	71	-17	75	697	33	54	45	27	66	27	31	38
669	0	68	-15	67	-9	67	-14	72	698	19	82	36	59	47	57	20	64
670	-14	70	-20	62	-1	61	-16	67	699	21	-10	21	-13	17	-10	9	-2
671	-24	56	-19	70	-8	66	-8	53	700	24	-11	33	-14	32	-13	30	-10
672	-23	68	-16	66	-14	66	-14	59	701	28	-8	39	-7	42	-9	31	-4
673	-24	50	-22	65	0	59	-9	52	702	28	-1	46	-2	49	-5	33	-1
674	-11	74	-20	63	2	59	-11	68	703	29	3	51	2	53	0	33	7
675	-14	106	-5	85	-3	81	-3	78	704	29	9	60	6	64	3	31	12
676	-13	97	-6	81	-3	76	-8	74	705	35	20	61	17	68	10	37	23
677	-15	90	-10	77	-7	72	-9	72	706	29	36	55	34	66	27	31	38
678	-12	90	-7	81	-6	78	-10	72	707	14	67	42	62	47	57	20	64
679	-18	88	-17	80	-12	72	-18	75	708	-3	75	-6	66	-5	71	-9	71
680	-10	73	-18	73	-14	68	-12	71	709	-1	23	-7	35	0	24	-7	37
681	-9	79	-4	74	-3	70	-11	63	710	1	34	-7	42	-1	36	-8	44
682	-14	86	-10	83	-6	76	-5	70	711	1	43	-8	45	-2	42	-11	49
683	-10	73	-9	71	-5	66	-17	75	712	0	54	-5	48	-2	52	-10	56
684	-10	70	-9	67	-5	62	-14	72	713	-2	55	-12	56	-9	57	-12	59
685	-10	69	-1	61	0	57	-16	67	714	0	61	-6	60	-6	63	-8	63
686	-5	66	-8	66	-4	61	-8	53	715	1	64	-5	62	-4	65	-9	67
687	-9	64	-14	66	-9	60	-14	59	716	0	68	-8	66	-4	67	-6	68
688	-5	58	0	59	1	54	-9	52	717	-9	92	-8	76	-7	82	-10	79

表9-29/JT-H264 718から775のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I		cabac_init_idc 値						ctxIdx	I		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
718	-17	120	-4	79	-5	85	-3	78	747	2	59	2	59	2	58	-11	68
719	-20	112	-7	71	-6	81	-8	74	748	23	-13	9	-2	17	-10	9	-2
720	-18	114	-5	69	-10	77	-9	72	749	26	-13	26	-9	32	-13	30	-10
721	-11	85	-9	70	-7	81	-10	72	750	40	-15	33	-9	42	-9	31	-4
722	-15	92	-8	66	-17	80	-18	75	751	49	-14	39	-7	49	-5	33	-1
723	-14	89	-10	68	-18	73	-12	71	752	44	3	41	-2	53	0	33	7
724	-26	71	-19	73	-4	74	-11	63	753	45	6	45	3	64	3	31	12
725	-15	81	-12	69	-10	83	-5	70	754	44	34	49	9	68	10	37	23
726	-14	80	-16	70	-9	71	-17	75	755	33	54	45	27	66	27	31	38
727	0	68	-15	67	-9	67	-14	72	756	19	82	36	59	47	57	20	64
728	-14	70	-20	62	-1	61	-16	67	757	21	-10	21	-13	17	-10	9	-2
729	-24	56	-19	70	-8	66	-8	53	758	24	-11	33	-14	32	-13	30	-10
730	-23	68	-16	66	-14	66	-14	59	759	28	-8	39	-7	42	-9	31	-4
731	-24	50	-22	65	0	59	-9	52	760	28	-1	46	-2	49	-5	33	-1
732	-11	74	-20	63	2	59	-11	68	761	29	3	51	2	53	0	33	7
733	-14	106	-5	85	-3	81	-3	78	762	29	9	60	6	64	3	31	12
734	-13	97	-6	81	-3	76	-8	74	763	35	20	61	17	68	10	37	23
735	-15	90	-10	77	-7	72	-9	72	764	29	36	55	34	66	27	31	38
736	-12	90	-7	81	-6	78	-10	72	765	14	67	42	62	47	57	20	64
737	-18	88	-17	80	-12	72	-18	75	766	-3	75	-6	66	-5	71	-9	71
738	-10	73	-18	73	-14	68	-12	71	767	-1	23	-7	35	0	24	-7	37
739	-9	79	-4	74	-3	70	-11	63	768	1	34	-7	42	-1	36	-8	44
740	-14	86	-10	83	-6	76	-5	70	769	1	43	-8	45	-2	42	-11	49
741	-10	73	-9	71	-5	66	-17	75	770	0	54	-5	48	-2	52	-10	56
742	-10	70	-9	67	-5	62	-14	72	771	-2	55	-12	56	-9	57	-12	59
743	-10	69	-1	61	0	57	-16	67	772	0	61	-6	60	-6	63	-8	63
744	-5	66	-8	66	-4	61	-8	53	773	1	64	-5	62	-4	65	-9	67
745	-9	64	-14	66	-9	60	-14	59	774	0	68	-8	66	-4	67	-6	68
746	-5	58	0	59	1	54	-9	52	775	-9	92	-8	76	-7	82	-10	79

表9-30/JT-H264 776から863のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
776	-6	93	-13	106	-21	126	-22	127	820	-6	93	-13	106	-21	126	-22	127
777	-6	84	-16	106	-23	124	-25	127	821	-6	84	-16	106	-23	124	-25	127
778	-8	79	-10	87	-20	110	-25	120	822	-8	79	-10	87	-20	110	-25	120
779	0	66	-21	114	-26	126	-27	127	823	0	66	-21	114	-26	126	-27	127
780	-1	71	-18	110	-25	124	-19	114	824	-1	71	-18	110	-25	124	-19	114
781	0	62	-14	98	-17	105	-23	117	825	0	62	-14	98	-17	105	-23	117
782	-2	60	-22	110	-27	121	-25	118	826	-2	60	-22	110	-27	121	-25	118
783	-2	59	-21	106	-27	117	-26	117	827	-2	59	-21	106	-27	117	-26	117
784	-5	75	-18	103	-17	102	-24	113	828	-5	75	-18	103	-17	102	-24	113
785	-3	62	-21	107	-26	117	-28	118	829	-3	62	-21	107	-26	117	-28	118
786	-4	58	-23	108	-27	116	-31	120	830	-4	58	-23	108	-27	116	-31	120
787	-9	66	-26	112	-33	122	-37	124	831	-9	66	-26	112	-33	122	-37	124
788	-1	79	-10	96	-10	95	-10	94	832	-1	79	-10	96	-10	95	-10	94
789	0	71	-12	95	-14	100	-15	102	833	0	71	-12	95	-14	100	-15	102
790	3	68	-5	91	-8	95	-10	99	834	3	68	-5	91	-8	95	-10	99
791	10	44	-9	93	-17	111	-13	106	835	10	44	-9	93	-17	111	-13	106
792	-7	62	-22	94	-28	114	-50	127	836	-7	62	-22	94	-28	114	-50	127
793	15	36	-5	86	-6	89	-5	92	837	15	36	-5	86	-6	89	-5	92
794	14	40	9	67	-2	80	17	57	838	14	40	9	67	-2	80	17	57
795	16	27	-4	80	-4	82	-5	86	839	16	27	-4	80	-4	82	-5	86
796	12	29	-10	85	-9	85	-13	94	840	12	29	-10	85	-9	85	-13	94
797	1	44	-1	70	-8	81	-12	91	841	1	44	-1	70	-8	81	-12	91
798	20	36	7	60	-1	72	-2	77	842	20	36	7	60	-1	72	-2	77
799	18	32	9	58	5	64	0	71	843	18	32	9	58	5	64	0	71
800	5	42	5	61	1	67	-1	73	844	5	42	5	61	1	67	-1	73
801	1	48	12	50	9	56	4	64	845	1	48	12	50	9	56	4	64
802	10	62	15	50	0	69	-7	81	846	10	62	15	50	0	69	-7	81
803	17	46	18	49	1	69	5	64	847	17	46	18	49	1	69	5	64
804	9	64	17	54	7	69	15	57	848	9	64	17	54	7	69	15	57
805	-12	104	10	41	-7	69	1	67	849	-12	104	10	41	-7	69	1	67
806	-11	97	7	46	-6	67	0	68	850	-11	97	7	46	-6	67	0	68
807	-16	96	-1	51	-16	77	-10	67	851	-16	96	-1	51	-16	77	-10	67
808	-7	88	7	49	-2	64	1	68	852	-7	88	7	49	-2	64	1	68
809	-8	85	8	52	2	61	0	77	853	-8	85	8	52	2	61	0	77
810	-7	85	9	41	-6	67	2	64	854	-7	85	9	41	-6	67	2	64
811	-9	85	6	47	-3	64	0	68	855	-9	85	6	47	-3	64	0	68

<b>812</b>	-13	88	2	55	2	57	-5	78	<b>856</b>	-13	88	2	55	2	57	-5	78
<b>813</b>	4	66	13	41	-3	65	7	55	<b>857</b>	4	66	13	41	-3	65	7	55
<b>814</b>	-3	77	10	44	-3	66	5	59	<b>858</b>	-3	77	10	44	-3	66	5	59
<b>815</b>	-3	76	6	50	0	62	2	65	<b>859</b>	-3	76	6	50	0	62	2	65
<b>816</b>	-6	76	5	53	9	51	14	54	<b>860</b>	-6	76	5	53	9	51	14	54
<b>817</b>	10	58	13	49	-1	66	15	44	<b>861</b>	10	58	13	49	-1	66	15	44
<b>818</b>	-1	76	4	63	-2	71	5	60	<b>862</b>	-1	76	4	63	-2	71	5	60
<b>819</b>	-1	83	6	64	-2	75	2	70	<b>863</b>	-1	83	6	64	-2	75	2	70

表9-31/JT-H264 864から951のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
864	15	6	14	11	19	-6	17	-13	908	15	6	14	11	19	-6	17	-13
865	6	19	11	14	18	-6	16	-9	909	6	19	11	14	18	-6	16	-9
866	7	16	9	11	14	0	17	-12	910	7	16	9	11	14	0	17	-12
867	12	14	18	11	26	-12	27	-21	911	12	14	18	11	26	-12	27	-21
868	18	13	21	9	31	-16	37	-30	912	18	13	21	9	31	-16	37	-30
869	13	11	23	-2	33	-25	41	-40	913	13	11	23	-2	33	-25	41	-40
870	13	15	32	-15	33	-22	42	-41	914	13	15	32	-15	33	-22	42	-41
871	15	16	32	-15	37	-28	48	-47	915	15	16	32	-15	37	-28	48	-47
872	12	23	34	-21	39	-30	39	-32	916	12	23	34	-21	39	-30	39	-32
873	13	23	39	-23	42	-30	46	-40	917	13	23	39	-23	42	-30	46	-40
874	15	20	42	-33	47	-42	52	-51	918	15	20	42	-33	47	-42	52	-51
875	14	26	41	-31	45	-36	46	-41	919	14	26	41	-31	45	-36	46	-41
876	14	44	46	-28	49	-34	52	-39	920	14	44	46	-28	49	-34	52	-39
877	17	40	38	-12	41	-17	43	-19	921	17	40	38	-12	41	-17	43	-19
878	17	47	21	29	32	9	32	11	922	17	47	21	29	32	9	32	11
879	24	17	45	-24	69	-71	61	-55	923	24	17	45	-24	69	-71	61	-55
880	21	21	53	-45	63	-63	56	-46	924	21	21	53	-45	63	-63	56	-46
881	25	22	48	-26	66	-64	62	-50	925	25	22	48	-26	66	-64	62	-50
882	31	27	65	-43	77	-74	81	-67	926	31	27	65	-43	77	-74	81	-67
883	22	29	43	-19	54	-39	45	-20	927	22	29	43	-19	54	-39	45	-20
884	19	35	39	-10	52	-35	35	-2	928	19	35	39	-10	52	-35	35	-2
885	14	50	30	9	41	-10	28	15	929	14	50	30	9	41	-10	28	15
886	10	57	18	26	36	0	34	1	930	10	57	18	26	36	0	34	1
887	7	63	20	27	40	-1	39	1	931	7	63	20	27	40	-1	39	1
888	-2	77	0	57	30	14	30	17	932	-2	77	0	57	30	14	30	17
889	-4	82	-14	82	28	26	20	38	933	-4	82	-14	82	28	26	20	38
890	-3	94	-5	75	23	37	18	45	934	-3	94	-5	75	23	37	18	45
891	9	69	-19	97	12	55	15	54	935	9	69	-19	97	12	55	15	54
892	-12	109	-35	125	11	65	0	79	936	-12	109	-35	125	11	65	0	79
893	36	-35	27	0	37	-33	36	-16	937	36	-35	27	0	37	-33	36	-16
894	36	-34	28	0	39	-36	37	-14	938	36	-34	28	0	39	-36	37	-14
895	32	-26	31	-4	40	-37	37	-17	939	32	-26	31	-4	40	-37	37	-17
896	37	-30	27	6	38	-30	32	1	940	37	-30	27	6	38	-30	32	1
897	44	-32	34	8	46	-33	34	15	941	44	-32	34	8	46	-33	34	15
898	34	-18	30	10	42	-30	29	15	942	34	-18	30	10	42	-30	29	15
899	34	-15	24	22	40	-24	24	25	943	34	-15	24	22	40	-24	24	25

<b>900</b>	40	-15	33	19	49	-29	34	22	<b>944</b>	40	-15	33	19	49	-29	34	22
<b>901</b>	33	-7	22	32	38	-12	31	16	<b>945</b>	33	-7	22	32	38	-12	31	16
<b>902</b>	35	-5	26	31	40	-10	35	18	<b>946</b>	35	-5	26	31	40	-10	35	18
<b>903</b>	33	0	21	41	38	-3	31	28	<b>947</b>	33	0	21	41	38	-3	31	28
<b>904</b>	38	2	26	44	46	-5	33	41	<b>948</b>	38	2	26	44	46	-5	33	41
<b>905</b>	33	13	23	47	31	20	36	28	<b>949</b>	33	13	23	47	31	20	36	28
<b>906</b>	23	35	16	65	29	30	27	47	<b>950</b>	23	35	16	65	29	30	27	47
<b>907</b>	13	58	14	71	25	44	21	62	<b>951</b>	13	58	14	71	25	44	21	62

表9-32/JT-H264 952から1011のctxIdxにおける変数mとnの値

(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
952	-3	71	-6	76	-23	112	-24	115	982	-3	71	-6	76	-23	112	-24	115
953	-6	42	-2	44	-15	71	-22	82	983	-6	42	-2	44	-15	71	-22	82
954	-5	50	0	45	-7	61	-9	62	984	-5	50	0	45	-7	61	-9	62
955	-3	54	0	52	0	53	0	53	985	-3	54	0	52	0	53	0	53
956	-2	62	-3	64	-5	66	0	59	986	-2	62	-3	64	-5	66	0	59
957	0	58	-2	59	-11	77	-14	85	987	0	58	-2	59	-11	77	-14	85
958	1	63	-4	70	-9	80	-13	89	988	1	63	-4	70	-9	80	-13	89
959	-2	72	-4	75	-9	84	-13	94	989	-2	72	-4	75	-9	84	-13	94
960	-1	74	-8	82	-10	87	-11	92	990	-1	74	-8	82	-10	87	-11	92
961	-9	91	-17	102	-34	127	-29	127	991	-9	91	-17	102	-34	127	-29	127
962	-5	67	-9	77	-21	101	-21	100	992	-5	67	-9	77	-21	101	-21	100
963	-5	27	3	24	-3	39	-14	57	993	-5	27	3	24	-3	39	-14	57
964	-3	39	0	42	-5	53	-12	67	994	-3	39	0	42	-5	53	-12	67
965	-2	44	0	48	-7	61	-11	71	995	-2	44	0	48	-7	61	-11	71
966	0	46	0	55	-11	75	-10	77	996	0	46	0	55	-11	75	-10	77
967	-16	64	-6	59	-15	77	-21	85	997	-16	64	-6	59	-15	77	-21	85
968	-8	68	-7	71	-17	91	-16	88	998	-8	68	-7	71	-17	91	-16	88
969	-10	78	-12	83	-25	107	-23	104	999	-10	78	-12	83	-25	107	-23	104
970	-6	77	-11	87	-25	111	-15	98	1000	-6	77	-11	87	-25	111	-15	98
971	-10	86	-30	119	-28	122	-37	127	1001	-10	86	-30	119	-28	122	-37	127
972	-12	92	1	58	-11	76	-10	82	1002	-12	92	1	58	-11	76	-10	82
973	-15	55	-3	29	-10	44	-8	48	1003	-15	55	-3	29	-10	44	-8	48
974	-10	60	-1	36	-10	52	-8	61	1004	-10	60	-1	36	-10	52	-8	61
975	-6	62	1	38	-10	57	-8	66	1005	-6	62	1	38	-10	57	-8	66
976	-4	65	2	43	-9	58	-7	70	1006	-4	65	2	43	-9	58	-7	70
977	-12	73	-6	55	-16	72	-14	75	1007	-12	73	-6	55	-16	72	-14	75
978	-8	76	0	58	-7	69	-10	79	1008	-8	76	0	58	-7	69	-10	79
979	-7	80	0	64	-4	69	-9	83	1009	-7	80	0	64	-4	69	-9	83
980	-9	88	-3	74	-5	74	-12	92	1010	-9	88	-3	74	-5	74	-12	92
981	-17	110	-10	90	-9	86	-18	108	1011	-17	110	-10	90	-9	86	-18	108

表9-33/JT-H264 1012から1023のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	I と SI		cabac_init_idc 値						ctxIdx	I と SI		cabac_init_idc 値					
	スライス		0		1		2			スライス		0		1		2	
	m	n	m	n	m	n	m	n		m	n	m	n	m	n	m	n
1012	-3	70	-3	74	-2	73	-5	79	1018	-10	90	-8	87	-9	91	-11	91
1013	-8	93	-9	92	-12	104	-11	104	1019	-30	127	-23	126	-31	127	-30	127
1014	-10	90	-8	87	-9	91	-11	91	1020	-3	70	-3	74	-2	73	-5	79
1015	-30	127	-23	126	-31	127	-30	127	1021	-8	93	-9	92	-12	104	-11	104
1016	-3	70	-3	74	-2	73	-5	79	1022	-10	90	-8	87	-9	91	-11	91
1017	-8	93	-9	92	-12	104	-11	104	1023	-30	127	-23	126	-31	127	-30	127

### 9.3.1.2 算術復号エンジンの初期化処理

この処理は、スライスの最初のマクロブロックを復号する前か、タイプ I\_PCM のマクロブロックにおけるいくつかの pcm\_alignment\_zero\_bit と全ての pcm\_sample\_luma と pcm\_sample\_chroma データを復号した後に起動される。

この処理の出力は、両者とも 16 ビットのレジスタ精度をもつ、初期化された復号エンジンレジスタ codIRange と codIOffset である。

算術復号エンジンの状態は変数 codIRange と codIOffset によって表される。算術復号処理の初期化手続きにおいて、codIRange は 0x01FE に等しく設定され、codIOffset は最上位ビットが最初に書かれた符号なし整数の 9 ビットのバイナリ表現として解釈される read\_bits(9)からの戻り値に等しく設定される。

ビットストリームは、結果として codIOffset の値が 0x01FE または 0x01FF に等しくなるデータを含んではならない。

記 本標準の算術復号エンジンの記述は、16ビットのレジスタ精度を利用する。しかしながら、変数 codIRange と codIOffset の最小レジスタ精度は9ビットである。

### 9.3.2 バイナリ化処理

この処理への入力、シンタックス要素に対する要求である。

この処理の出力は、シンタックス要素のバイナリ化、maxBinIdxCtx、ctxIdxOffset および bypassFlag である。

表 9-34/JT-H264 は各シンタックス要素に関連付けられたバイナリ化処理のタイプ、maxBinIdxCtx および ctxIdxOffset を規定する。

ユーナリ (U) バイナリ化処理の規定、切り詰めユーナリ (TU) バイナリ化処理、連結ユーナリ/k 次指数 Golomb (UEGk) バイナリ化処理および固定長 (FL) バイナリ化処理は、9.3.2.1 小小節から 9.3.2.4 小小節までで各々与えられる。他のバイナリ化は 9.3.2.5 小小節から 9.3.2.7 小小節までに規定される。

I スライスを除いて、9.3.2.5 小小節に規定されるシンタックス要素 mb\_type のバイナリ化は、プリフィックスおよびサフィックスのビット記号列の連結により与えられる bin 記号列から構成される。シンタックス

要素 `mvd_IX(X=0,1)` と `coeff_abs_level_minus1` のバイナリ化に使用される 9.3.2.3 小小節で規定される UEGk バイナリ化、および `coded_block_pattern` のバイナリ化は、同様にプリフィックスおよびサフィックスのビット記号列の連結で構成される。これらのバイナリ化処理において、プリフィックスおよびサフィックスのビット記号列は、9.3.3 小節でさらに規定されるように `binIdx` 変数を用いて別々に索引される。プリフィックスのビット記号列およびサフィックスのビット記号列の 2 組は、バイナリ化プリフィックス部分およびバイナリ化サフィックス部分と各々呼ばれる。

表 9-34/JT-H264 で与えられる、コンテキストインデックスオフセット(`ctxIdxOffset`)変数の特定の値および `maxBinIdxCtx` 変数の特定の値は、シンタックス要素の各バイナリ化またはバイナリ化部分に関連する。表 9-34/JT-H264 における単一のシンタックス要素に対して、それらの変数それぞれに 2 つの値が規定される時、対応するシンタックス要素のバイナリ化の上の行の値がプリフィックス部分に関連し下の行の値はサフィックス部分に関連する。

DecodeBypass 処理の使用および変数 `bypassFlag` は、次の通り導出される。

- もし、表 9-34/JT-H264 で”na”とラベル付けされ、対応するバイナリ化またはバイナリ化部分に対して、`ctxIdxOffset` へ値が割り当てられないならば、対応するバイナリ化またはバイナリ化プリフィックス/サフィックス部分のビット記号列の全ての `bin` は、9.3.3.2.3 小小節で規定されるような DecodeBypass 処理を起動することで復号される。このような場合、`bypassFlag` は 1 に設定される。ここで `bypassFlag` は、ビットストリームから `bin` 値を構文解析するために、DecodeBypass 処理が適用されることを示すために使用される。
- それ以外、表 9-34/JT-H264 に与えられる `maxBinIdxCtx` の規定値までの各々可能な `binIdx` の値に対して、変数 `ctxIdx` の特定の値は、9.3.3 小節でさらに規定される。`bypassFlag` は 0 に等しく設定される。

コンテキストインデックス `ctxIdx` の可能な値は、0 から 1023 のそれぞれの値を含む範囲内である。`ctxIdxOffset` に割り当てられる値は、シンタックス要素の対応するバイナリ化またはバイナリ化部分に割り当てられた `ctxIdx` の範囲の下側の値を規定する。

`ctxIdx=ctxIdxOffset=276` は、シンタックス要素 `end_of_slice_flag`、および 9.3.3.1 小小節でさらに規定される I\_PCM マクロブロックタイプを規定する `mb_type` の `bin` に割り当てられる。ビットストリームから対応する `bin` 値を構文解析するために、9.3.3.2.4 小小節で規定される終端前の判定のための算術復号処理 (DecodeTerminate)が適用される。

記 I スライスの `mb_type` の `bin` と、同一の `binIdx` 値に対応する SI スライスの `mb_type` のサフィックスの `bin` は同一の `ctxIdx` を共有する。P、SP および B スライスの、`mb_type` のプリフィックスの最後の `bin` と `mb_type` のサフィックスの最初の `bin` は同一の `ctxIdx` を共有してもよい。

表9-34/JT-H264 シンタックス要素と関連付けられたバイナリ化のタイプ、maxBinIdxCtxおよび  
ctxIdxOffset  
(ITU-T H.264)

シンタックス要素	バイナリ化のタイプ	maxBinIdxCtx	ctxIdxOffset
mb_type (SIスライスのみ)	9.3.2.5小節で規定される プリフィックスおよびサフィックス	プリフィックス:0 サフィックス:6	プリフィックス:0 サフィックス:3
mb_type (Iスライスのみ)	9.3.2.5小節で規定される	6	3
mb_skip_flag (P,SPスライスのみ)	FL,cMax=1	0	11
mb_type (P,SPスライスのみ)	9.3.2.5小節で規定される プリフィックスおよびサフィックス	プリフィックス:2 サフィックス:5	プリフィックス:14 サフィックス:17
sub_mb_type (P,SPスライスのみ)	9.3.2.5小節で規定される	2	21
mb_skip_flag (Bスライスのみ)	FL,cMax=1	0	24
mb_type (Bスライスのみ)	9.3.2.5小節で規定される プリフィックスおよびサフィックス	プリフィックス:3 サフィックス:5	プリフィックス:27 サフィックス:32
sub_mb_type (Bスライスのみ)	9.3.2.5小節で規定される	3	36
mvd_10[ ][ ][0],mvd_11[ ][ ][0]	signedValFlag=1、uCoff=9である UEG3で与えられる プリフィックスおよびサフィックス	プリフィックス:4 サフィックス:na	プリフィックス:40 サフィックス:na (DecodeBypassを使用)

mvd_l0[ ][1],mvd_l1[ ][1]		プリフィックス:4 サフィックス:na	プリフィックス:47 サフィックス:na (DecodeBypassを使用)
ref_idx_l0,ref_idx_l1	U	2	54
mb_qp_delta	9.3.2.7小節で規定される	2	60
intra_chroma_pred_mode	TU,cMax=3	1	64
prev_intra4x4_pred_mode_flag, prev_intra8x8_pred_mode_flag	FL,cMax=1	0	68
rem_intra4x4_pred_mode, rem_intra8x8_pred_mode	FL,cMax=7	0	69
mb_field_decoding_flag	FL,cMax=1	0	70
coded_block_pattern	9.3.2.6小節で規定される プリフィックスおよびサフィックス	プリフィックス:3 サフィックス:1	プリフィックス:73 サフィックス:77
coded_block_flag	FL,cMax=1	0	85
significant_coeff_flag (ctxBlockCat<5であるフレーム符号化されたブロック)	FL,cMax=1	0	105
last_significant_coeff_flag (ctxBlockCat<5であるフレーム符号化されたブロック)	FL,cMax=1	0	166
coeff_abs_level_minus1 (ctxBlockCat<5であるブロック)	signedValFlag=0、uCoff=14である UEG0で与えられる プリフィックスおよびサフィックス	プリフィックス:1 サフィックス:na	プリフィックス:227 サフィックス:na (DecodeBypassを使用)
coeff_sign_flag	FL,cMax=1	0	na(DecodeBypassを使用)
end_of_slice_flag	FL,cMax=1	0	276
significant_coeff_flag (ctxBlockCat<5であるフィールド符号化されたブロック)	FL,cMax=1	0	277
last_significant_coeff_flag (ctxBlockCat<5であるフィールド符号化されたブロック)	FL,cMax=1	0	338
transform_size_8x8_flag	FL,cMax=1	0	399
significant_coeff_flag (ctxBlockCat==5であるフレーム符号化されたブロック)	FL,cMax=1	0	402

last_significant_coeff_flag (ctxBlockCat == 5であるフレーム符号化されたブロック)	FL,cMax=1	0	417
coeff_abs_level_minus1 (ctxBlockCat == 5であるブロック)	signedValFlag=0、uCoff=14であるUEG0で与えられる プリフィックスおよびサフィックス	プリフィックス:1 サフィックス:na	プリフィックス:426 サフィックス:na (DecodeBypassを使用)
significant_coeff_flag (ctxBlockCat == 5であるフィールド符号化されたブロック)	FL,cMax=1	0	436
last_significant_coeff_flag (ctxBlockCat == 5であるフィールド符号化されたブロック)	FL,cMax=1	0	451
coded_block_flag (5 < ctxBlockCat < 9)	FL,cMax=1	0	460
coded_block_flag (9 < ctxBlockCat < 13)	FL,cMax=1	0	472
coded_block_flag (ctxBlockCat == 5, 9, または13)	FL,cMax=1	0	1012
significant_coeff_flag (5 < ctxBlockCat < 9であるフレーム符号化されたブロック)	FL,cMax=1	0	484
significant_coeff_flag (9 < ctxBlockCat < 13であるフレーム符号化されたブロック)	FL,cMax=1	0	528
last_significant_coeff_flag (5 < ctxBlockCat < 9であるフレーム符号化されたブロック)	FL,cMax=1	0	572
last_significant_coeff_flag (9 < ctxBlockCat < 13であるフレーム符号化されたブロック)	FL,cMax=1	0	616
coeff_abs_level_minus1 (5 < ctxBlockCat < 9であるブロック)	signedValFlag=0、uCoff=14であるUEG0で与えられるプリフィックス およびサフィックス	プリフィックス:1 サフィックス:na	プリフィックス:952 サフィックス:na (DecodeBypassを使用)
coeff_abs_level_minus1 (9 < ctxBlockCat < 13であるブロック)	signedValFlag=0、uCoff=14であるUEG0で与えられるプリフィックス およびサフィックス	プリフィックス:1 サフィックス:na	プリフィックス:982 サフィックス:na (DecodeBypassを使用)
significant_coeff_flag (5 < ctxBlockCat < 9であるフィールド符号化されたブロック)	FL,cMax=1	0	776

significant_coeff_flag (9 < ctxBlockCat < 13である フィールド符号化されたブ ロック)	FL,cMax=1	0	820
last_significant_coeff_flag (5 < ctxBlockCat < 9であるフ ィールド符号化されたブロ ック)	FL,cMax=1	0	864
last_significant_coeff_flag (9 < ctxBlockCat < 13である フィールド符号化されたブ ロック)	FL,cMax=1	0	908
significant_coeff_flag (f ctxBlockCat == 9であるフ レーム符号化されたブロ ック)	FL,cMax=1	0	660
significant_coeff_flag (ctxBlockCat == 13であるフ レーム符号化されたブロ ック)	FL,cMax=1	0	718
last_significant_coeff_flag (ctxBlockCat == 9であるフ レーム符号化されたブロ ック)	FL,cMax=1	0	690
last_significant_coeff_flag (ctxBlockCat == 13であるフ レーム符号化されたブロ ック)	FL,cMax=1	0	748
coeff_abs_level_minus1 (ctxBlockCat == 9であるブ ロック)	signedValFlag=0、uCoff=14である UEG0で与えられるプリフィックス およびサフィックス	プリフィックス:1 サフィックス:na	プリフィックス:708 サフィックス:na (DecodeBypassを使用)
coeff_abs_level_minus1 (ctxBlockCat == 13であるブ ロック)	signedValFlag=0、uCoff=14である UEG0で与えられるプリフィックス およびサフィックス	プリフィックス:1 サフィックス:na	プリフィックス:766 サフィックス:na (DecodeBypassを使用)
significant_coeff_flag (ctxBlockCat == 9であるフ ィールド符号化されたブロ ック)	FL,cMax=1	0	675
significant_coeff_flag (ctxBlockCat == 13であるフ ィールド符号化されたブロ ック)	FL,cMax=1	0	733
last_significant_coeff_flag (ctxBlockCat == 9であるフ ィールド符号化されたブロ ック)	FL,cMax=1	0	699
last_significant_coeff_flag (ctxBlockCat == 13であるフ ィールド符号化されたブロ ック)	FL,cMax=1	0	757

### 9.3.2.1 ユーナリ (U) バイナリ化処理

この処理への入力は、シンタックス要素に対する U バイナリ化の要求である。

この処理の出力は、シンタックス要素の U バイナリ化である。

(符号無し整数) 値  $\text{synElVal}$  を持つシンタックス要素の bin 記号列は、 $\text{binIdx}$  によって索引される長さ  $\text{synElVal}+1$  のビット記号列である。 $\text{synElVal}$  未満の  $\text{binIdx}$  に対する bin は 1 に等しい。 $\text{synElVal}$  に等しい  $\text{binIdx}$  を持つ bin は 0 に等しい。

表 9-35/JT-H264 はシンタックス要素に対するユーナリバイナリ化の bin 記号列を示す。

表9-35/JT-H264 ユーナリバイナリ化のbin記号列 (参考)  
(ITU-T H.264)

シンタックス要素値	bin記号列					
0(I_NxN)	0					
1	1	0				
2	1	1	0			
3	1	1	1	0		
4	1	1	1	1	0	
5	1	1	1	1	1	0
...						
binIdx	0	1	2	3	4	5

### 9.3.2.2 切り詰めユーナリ (TU) バイナリ化処理

この処理への入力は、シンタックス要素に対する TU バイナリ化要求および  $\text{cMax}$  である。

この処理の出力は、シンタックス要素の TU バイナリ化である。

$\text{cMax}$  より小さいシンタックス要素 (符号無し整数) 値に対して、9.3.2.1 小節で規定される U バイナリ化処理が起動される。 $\text{cMax}$  と等しいシンタックス要素値に対して、bin 記号列は、全ての bin が 1 に等しく、長さが  $\text{cMax}$  のビット記号列である。

記 TU バイナリ化は、復号されるシンタックス要素の最大可能値に等しい  $\text{cMax}$  値を持って、常に起動される。

### 9.3.2.3 連結ユーナリ/k次指数Golomb (UEGk) バイナリ化処理

この処理への入力は、シンタックス要素に対する UEGk バイナリ化要求、 $\text{signedValFlag}$  および  $\text{uCoff}$  である。

この処理の出力は、シンタックス要素の UEGk バイナリ化である。

UEGk の bin 記号列は、プリフィックスビット記号列とサフィックスビット記号列との連結である。バイナリ化のプリフィックスは、シンタックス要素値  $\text{synElVal}$  のプリフィックス部分  $\text{Min}(\text{uCoff}, \text{Abs}(\text{synElVal}))$

に対して、 $cMax=uCoff$  で 9.3.2.2 小小節で規定される TU バイナリ化処理を起動することにより規定される。  
ここで  $uCoff>0$  である。

UEGk の bin 記号列は、次の通り導出される。

- もし以下の 1 つが真であるならば、値  $synElVal$  を持つシンタックス要素の bin 記号列は、プリフィックスビット記号列のみから構成される。
  - $signedValFlag$  が 0 に等しく、かつ、プリフィックスビット記号列は、全てのビットが 1 に等しい長さ  $uCoff$  のビット記号列とは等しくない。
  - $signedValFlag$  が 1 に等しく、かつ、プリフィックスビット記号列は、値が 0 に等しい単一のビットからなるビット記号列に等しい。
- それ以外、シンタックス要素値  $synElVal$  を持つ UEGk サフィックス部分の bin 記号列は、以下の擬似コードと等価な処理により規定される：

```
if(Abs(synElVal)>=uCoff){
    sufS=Abs(synElVal)-uCoff
    stopLoop=0
    do{
        if(sufS>=(1<<k)){
            put(1)
            sufS=sufS-(1<<k)
            k++
        }else{
            put(0)
            while(k--){
                put((sufS>>k)&0x01)
            }
            stopLoop=1
        }
    }while(!stopLoop)
}
if(signedValFlag && synElVal!=0)
    if(synElVal>0)
        put(0)
    else
        put(1)
```

記  $k$  次指数 Golomb (EGk) 符号における規定は、9.1 節で規定される 0 次の指数 Golomb 符号のユニナリ部分に対して、逆の意味で 1 および 0 を用いる。

#### 9.3.2.4 固定長 (FL) バイナリ化処理

この処理への入力は、シンタックス要素に対する FL バイナリ化要求および  $cMax$  である。

この処理の出力は、シンタックス要素に対する FL バイナリ化である。

FL バイナリ化は、シンタックス要素値の `fixedLength` ビット符号無し整数の `bin` 記号列を用いて構築される。ここで `fixedLength=Ceil(Log2(cMax+1))` である。FL バイナリ化における `bin` の索引は、`binIdx=0` が最下位ビットに関連し、`binIdx` 値の増加が最上位ビットに向かうように作られる。

### 9.3.2.5 マクロブロックタイプおよびサブマクロブロックタイプに対するバイナリ化処理

この処理への入力、シンタックス要素 `mb_type` または `sub_mb_type` に対するバイナリ化要求である。

この処理の出力は、シンタックス要素のバイナリ化である。

I スライスにおけるマクロブロックタイプの復号のためのバイナリ化体系は、表 9-36/JT-H264 で規定される。

SI スライスにおけるマクロブロックタイプにおいては、バイナリ化はプリフィックスとサフィックスビット記号列の連結として規定される `bin` 記号列から次の通り構成される。

プリフィックスビット記号列は、 $b_0=(mb\_type==SI)?0:1$  により規定される単一のビットから構成される。 $b_0$  が 0 に等しい場合のシンタックス要素値に対しては、`bin` 記号列はプリフィックスビット記号列のみからなる。 $b_0$  が 1 に等しい場合のシンタックス要素に対しては、プリフィックス  $b_0$  と、SI スライスの `mb_type` 値から 1 を減じて索引される I スライス中のマクロブロックタイプに対して、表 9-36/JT-H264 で規定されるサフィックスビット記号列を連結してバイナリ化が与えられる。

表9-36/JT-H264 Iスライスにおけるマクロブロックタイプのバイナリ化  
(ITU-T H.264)

mb_type値 (名前)	Bin記号列						
0(I_4x4)	0						
1(I_16x16_0_0_0)	1	0	0	0	0	0	
2(I_16x16_1_0_0)	1	0	0	0	0	1	

3(I_16x16_2_0_0)	1	0	0	0	1	0	
4(I_16x16_3_0_0)	1	0	0	0	1	1	
5(I_16x16_0_1_0)	1	0	0	1	0	0	0
6(I_16x16_1_1_0)	1	0	0	1	0	0	1
7(I_16x16_2_1_0)	1	0	0	1	0	1	0
8(I_16x16_3_1_0)	1	0	0	1	0	1	1
9(I_16x16_0_2_0)	1	0	0	1	1	0	0
10(I_16x16_1_2_0)	1	0	0	1	1	0	1
11(I_16x16_2_2_0)	1	0	0	1	1	1	0
12(I_16x16_3_2_0)	1	0	0	1	1	1	1
13(I_16x16_0_0_1)	1	0	1	0	0	0	
14(I_16x16_1_0_1)	1	0	1	0	0	1	
15(I_16x16_2_0_1)	1	0	1	0	1	0	
16(I_16x16_3_0_1)	1	0	1	0	1	1	
17(I_16x16_0_1_1)	1	0	1	1	0	0	0
18(I_16x16_1_1_1)	1	0	1	1	0	0	1
19(I_16x16_2_1_1)	1	0	1	1	0	1	0
20(I_16x16_3_1_1)	1	0	1	1	0	1	1
21(I_16x16_0_2_1)	1	0	1	1	1	0	0
22(I_16x16_1_2_1)	1	0	1	1	1	0	1
23(I_16x16_2_2_1)	1	0	1	1	1	1	0
24(I_16x16_3_2_1)	1	0	1	1	1	1	1
25(I_PCM)	1	1					
binIdx	0	1	2	3	4	5	6

P および SP スライスの P マクロブロックタイプおよび、B スライスの B マクロブロックに対するバイナリ化体系は、表 9-37/JT-H264 で規定される。

mb\_type 値が 5 から 30 に対応する P および SP スライスの I マクロブロックタイプに対する bin 記号列は、表 9-37/JT-H264 で規定される値が 1 に等しい単一のビットから構成されるプリフィックス、および mb\_type 値から 5 を減じて索引される表 9-36/JT-H264 で規定されるサフィックスの連結からなる。

4 に等しい mb\_type (P\_8x8ref0) は、許容されない。

B スライスの I マクロブロックタイプ (mb\_type 値が 23 から 48) に対して、バイナリ化は、表 9-37/JT-H264 で規定されるプリフィックスビット記号列、および mb\_type 値から 23 を減じて索引される表 9-36/JT-H264 で規定されるサフィックスビット記号列の連結として規定される bin 記号列から構成される。

表9-37/JT-H264 P、SPおよびBスライスのマクロブロックタイプにおけるバイナリ化  
(ITU-T H.264)

スライスタイプ	mb_typeの値 (名前)	Bin記号列							
P,SPスライス	0(P_L0_16x16)	0	0	0					
	1(P_L0_L0_16x8)	0	1	1					
	2(P_L0_L0_8x16)	0	1	0					
	3(P_8x8)	0	0	1					
	4(P_8x8ref0)	na							
	5から30 (INTRA、プリフィックスのみ)	1							
Bスライス	0(B_Direct_16x16)	0							
	1(B_L0_16x16)	1	0	0					
	2(B_L1_16x16)	1	0	1					
	3(B_Bi_16x16)	1	1	0	0	0	0		
	4(B_L0_L0_16x8)	1	1	0	0	0	1		
	5(B_L0_L0_8x16)	1	1	0	0	1	0		
	6(B_L1_L1_16x8)	1	1	0	0	1	1		
	7(B_L1_L1_8x16)	1	1	0	1	0	0		
	8(B_L0_L1_16x8)	1	1	0	1	0	1		
	9(B_L0_L1_8x16)	1	1	0	1	1	0		
	10(B_L1_L0_16x8)	1	1	0	1	1	1		
	11(B_L1_L0_8x16)	1	1	1	1	1	0		
	12(B_L0_Bi_16x8)	1	1	1	0	0	0	0	
	13(B_L0_Bi_8x16)	1	1	1	0	0	0	1	
	14(B_L1_Bi_16x8)	1	1	1	0	0	1	0	
	15(B_L1_Bi_8x16)	1	1	1	0	0	1	1	
	16(B_Bi_L0_16x8)	1	1	1	0	1	0	0	
	17(B_Bi_L0_8x16)	1	1	1	0	1	0	1	
	18(B_Bi_L1_16x8)	1	1	1	0	1	1	0	
	19(B_Bi_L1_8x16)	1	1	1	0	1	1	1	
	20(B_Bi_Bi_16x8)	1	1	1	1	0	0	0	
	21(B_Bi_Bi_8x16)	1	1	1	1	0	0	1	
	22(B_8x8)	1	1	1	1	1	1		
23から48 (INTRA、プリフィックスのみ)	1	1	1	1	0	1			

binIdx	0	1	2	3	4	5	6
--------	---	---	---	---	---	---	---

P、SP および B スライスに対する sub\_mb\_type におけるバイナリ化の規定は、表 9-38/JT-H264 で与えられる。

表9-38/JT-H264 P、SPおよびBスライスのサブマクロブロックタイプにおけるバイナリ化 (ITU-T H.264)

スライスタイプ	sub_mb_type値 (名前)	Bin記号列					
P,SPスライス	0(P_L0_8x8)	1					
	1(P_L0_8x4)	0	0				
	2(P_L0_4x8)	0	1	1			
	3(P_L0_4x4)	0	1	0			
Bスライス	0(B_Direct_8x8)	0					
	1(B_L0_8x8)	1	0	0			
	2(B_L1_8x8)	1	0	1			
	3(B_Bi_8x8)	1	1	0	0	0	
	4(B_L0_8x4)	1	1	0	0	1	
	5(B_L0_4x8)	1	1	0	1	0	
	6(B_L1_8x4)	1	1	0	1	1	
	7(B_L1_4x8)	1	1	1	0	0	0
	8(B_Bi_8x4)	1	1	1	0	0	1
	9(B_Bi_4x8)	1	1	1	0	1	0
	10(B_L0_4x4)	1	1	1	0	1	1
	11(B_L1_4x4)	1	1	1	1	0	
12(B_Bi_4x4)	1	1	1	1	1		
binIdx		0	1	2	3	4	5

### 9.3.2.6 符号化ブロックパターンに対するバイナリ化処理

この処理への入力、シンタックス要素 coded\_block\_pattern に対するバイナリ化要求である。

この処理の出力は、シンタックス要素のバイナリ化である。

coded\_block\_pattern のバイナリ化は、プリフィックス部分と（存在する時）サフィックス部分から構成される。バイナリ化のプリフィックス部分は、cMax=15 を用いる CodedBlockPatternLuma の FL バイナリ化により与えられる。ChromaArrayType が 0 または 3 に等しくない時、サフィックス部分が存在し、かつ cMax=2 を用いる CodedBlockPatternChroma の TU バイナリ化から構成される。シンタックス要素 coded\_block\_pattern

の値と、CodedBlockPatternLuma および CodedBlockPatternChroma の値との間の関係は 7.4.5 小節で規定されるように与えられる。

### 9.3.2.7 mb\_qp\_deltaに対するバイナリ化処理

この処理への入力、シンタックス要素 mb\_qp\_delta に対するバイナリ化要求である。

この処理の出力は、シンタックス要素のバイナリ化である。

mb\_qp\_delta の bin 記号列は、シンタックス要素 mb\_qp\_delta のマップされた値の U バイナリ化により導出される。ここで、mb\_qp\_delta の符号付きの値とそのマップされた値との間の割り当て規則は表 9-3/JT-H264 で規定されるように与えられる。

### 9.3.3 復号処理フロー

この処理への入力、9.3.2 小節で規定されるような、要求されたシンタックス要素のバイナリ化、maxBinIdxCtx、bypassFlag および ctxIdxOffset である。

この処理の出力は、シンタックス要素値である。

この処理は、各シンタックス要素に対して、ビット記号列中の各ビットがどのように構文解析されるかを規定する。

各ビットを構文解析後、結果であるビット記号列は、シンタックス要素のバイナリ化の全ての bin 記号列と比較され、次が適用される。

- もしビット記号列が bin 記号列の 1 つに等しいならば、対応するシンタックス要素値が出力である。
- それ以外（ビット記号列が bin 記号列の 1 つに等しくない）、次のビットが構文解析される。

各 bin を構文解析する間、変数 binIdx は、最初の bin に対して 0 に等しく設定されて始まり 1 ずつ加算される。

対応するシンタックス要素のバイナリ化がプリフィックスとサフィックスバイナリ化部分から構成される時、bin 記号列の各部分（プリフィックス部分またはサフィックス部分）の最初の bin に対して、変数 binIdx は 0 に等しく設定される。この場合、プリフィックスビット記号列を構文解析した後で、9.3.2.3 および 9.3.2.5 小節で規定されるバイナリ化に関連するサフィックスビット記号列の構文解析処理が、9.3.2.3 および 9.3.2.5 小節で規定されるプリフィックスビット記号列の結果に従って起動される。シンタックス要素 coded\_block\_pattern のバイナリ化においては、9.3.2.6 小節で規定されるように、長さ 4 のプリフィックスビット記号列に関わらずサフィックスビット記号列が現れることに注意せよ。

変数 bypassFlag に従って、次が適用される。

- もし bypassFlag が 1 に等しいならば、9.3.3.2.3 小節で規定されるようなバイパス復号処理が、ビットストリームから bin の値を構文解析するために適用される。
- それ以外（bypassFlag が 0 に等しい）、各 bin の構文解析は次の 2 つの順序付けられたステップで規定される：
  1. 与えられた binIdx、maxBinIdxCtx および ctxIdxOffset で、9.3.3.1 小節で規定されるように ctxIdx が導出される。

2. 与えられた `ctxIdx` で、9.3.3.2 小小節で規定されるようにビットストリームから `bin` の値が復号される。

### 9.3.3.1 `ctxIdx`の導出処理

この処理への入力は、`binIdx`、`maxBinIdxCtx` および `ctxIdxOffset` である。

この処理の出力は、`ctxIdx` である。

表 9-39/JT-H264 は、シンタックス要素 `coded_block_flag`、`significant_coeff_flag`、`last_significant_coeff_flag` および `coeff_abs_level_minus1` に関連するものを除き、全ての `ctxIdxOffset` 値に対する `ctxIdx` 増分(`ctxIdxInc`)の `binIdx` への割り当てを示す。

特定の `binIdx` で使用される `ctxIdx` は、与えられた `bin` 記号列、またはそれについての部分に関連した `ctxIdxOffset` を最初に決定することで規定される。`ctxIdx` は次の通り決定される。

- もし `ctxIdxOffset` が表 9-39/JT-H264 にリストされているならば、`binIdx` に対する `ctxIdx` は、`ctxIdxOffset` と `ctxIdxInc` の和であり、それは表 9-39/JT-H264 で見つけられる。`binIdx` に対して表 9-39/JT-H264 に 1 より多くの値がリストされている時は、`binIdx` に対する `ctxIdxInc` の割り当て処理が対応する表エントリの括弧内に与えられる節でさらに規定される。
- それ以外 (`ctxIdxOffset` が表 9-39/JT-H264 にリストされていない)、`ctxIdx` は次の項の和で規定される：`ctxIdxOffset` と、表 9-40/JT-H264 で規定される `ctxIdxBlockCatOffset`(`ctxBlockCat`)、および `ctxIdxInc`(`ctxBlockCat`)。9.3.3.1.3 小小節は、どの `ctxBlockCat` が使われるかを規定する。9.3.3.1.1.9 小小節は、`coded_block_flag` に対する `ctxIdxInc`(`ctxBlockCat`)の割り当てを規定し、9.3.3.1.3 小小節は `significant_coeff_flag`、`last_significant_coeff_flag` と `coeff_abs_level_minus1` に対する `ctxIdxInc`(`ctxBlockCat`)の割り当てを規定する。

`maxBinIdxCtx` より大きい `binIdx` を持つ全ての `bin` は、`maxBinIdxCtx` に等しい `binIdx` に割り当てられる `ctxIdx` の値を用いて構文解析される。

表 9-39/JT-H264 で“na”とラベル付けされた全てのエントリは、対応する `ctxIdxOffset` に対して生起しない `binIdx` 値に対応する。

`ctxIdx=276` が `I_PCM` モードを示す `mb_type` の `binIdx` に割り当てられる。ビットストリームから対応する `bin` 値の構文解析のために、9.3.3.2.4 小小節で規定されるような終端前の判定に対する算術復号処理が適用される。

表9-39/JT-H264 シンタックス要素coded\_block\_flag、significant\_coeff\_flag、last\_significant\_coeff\_flag  
 およびcoeff\_abs\_level\_minus1に関連するものを除く全てのctxIdxOffset値に対するctxIdxIncのbinIdxへの割  
 り当て  
 (ITU-T H.264)

ctxIdxOffset	binIdx						
	0	1	2	3	4	5	>=6
0	0,1,2 (9.3.3.1.1.3 小小小小節)	na	na	na	na	na	na
3	0,1,2 (9.3.3.1.1.3 小小小小節)	ctxIdx=276	3	4	5,6 (9.3.3.1.2 小小小節)	6,7 (9.3.3.1.2 小小小節)	7
11	0,1,2 (9.3.3.1.1.1 小小小小節)	na	na	na	na	na	na
14	0	1	2,3 (9.3.3.1.2 小小小節)	na	na	na	na
17	0	ctxIdx=276	1	2	2,3 (9.3.3.1.2 小小小節)	3	3
21	0	1	2	na	na	na	na
24	0,1,2 (9.3.3.1.1.1 小小小小節)	na	na	na	na	na	na
27	0,1,2 (9.3.3.1.1.3 小小小小節)	3	4,5 (9.3.3.1.2 小小小節)	5	5	5	5
32	0	ctxIdx=276	1	2	2,3 (9.3.3.1.2 小小小節)	3	3
36	0	1	2,3 (9.3.3.1.2 小小小節)	3	3	3	na
40	0,1,2 (9.3.3.1.1.7 小小小小節)	3	4	5	6	6	6
47	0,1,2 (9.3.3.1.1.7 小小小小節)	3	4	5	6	6	6
54	0,1,2,3 (9.3.3.1.1.6 小小小小節)	4	5	5	5	5	5

60	0,1 (9.3.3.1.1.5 小小小小節)	2	3	3	3	3	3
64	0,1,2 (9.3.3.1.1.8 小小小小節)	3	3	na	na	na	na
68	0	na	na	na	na	na	na
69	0	0	0	na	na	na	na
70	0,1,2 (9.3.3.1.1.2 小小小小節)	na	na	na	na	na	na
73	0,1,2,3 (9.3.3.1.1.4 小小小小節)	0,1,2,3 (9.3.3.1.1.4 小小小小節)	0,1,2,3 (9.3.3.1.1.4 小小小小節)	0,1,2,3 (9.3.3.1.1.4 小小小小節)	na	na	na
77	0,1,2,3 (9.3.3.1.1.4 小小小小節)	4,5,6,7 (9.3.3.1.1.4 小小小小節)	na	na	na	na	na
276	0	na	na	na	na	na	na
399	0,1,2 (9.3.3.1.1.10 小小小小節)	na	na	na	na	na	na

表 9-40/JT-H264 は、シンタックス要素 coded\_block\_flag、significant\_coeff\_flag、last\_significant\_coeff\_flag および coeff\_abs\_level\_minus1 に対して、ctxBlockCat に従う ctxIdxBlockCatOffset の値を示す。ctxBlockCat の規定は、表 9-42/JT-H264 で与えられる。

**表9-40/JT-H264 シンタックス要素coded\_block\_flag、significant\_coeff\_flag、last\_significant\_coeff\_flag  
およびcoeff\_abs\_level\_minus1に対するctxIdxBlockCatOffsetのctxBlockCatへの割り当て  
(ITU-T H.264)**

シンタックス要素	ctxBlockCat(表9-42/JT-H264で規定される)													
	0	1	2	3	4	5	6	7	8	9	10	11	12	13
coded_block_flag	0	4	8	12	16	0	0	4	8	4	0	4	8	8
significant_coeff_flag	0	15	29	44	47	0	0	15	29	0	0	15	29	0
last_significant_coeff_flag	0	15	29	44	47	0	0	15	29	0	0	15	29	0
coeff_abs_level_minus1	0	10	20	30	39	0	0	10	20	0	0	10	20	0

### 9.3.3.1.1 隣接のシンタックス要素を用いるctxIdxIncの割り当て処理

9.3.3.1.1.1 小小小節は、シンタックス要素 mb\_skip\_flag に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.2 小小小節は、シンタックス要素 mb\_field\_decoding\_flag に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.3 小小小節は、シンタックス要素 mb\_type に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.4 小小小節は、シンタックス要素 coded\_block\_pattern に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.5 小小小節は、シンタックス要素 mb\_qp\_delta に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.6 小小小節は、シンタックス要素 ref\_idx\_l0 と ref\_idx\_l1 に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.7 小小小節は、シンタックス要素 mvd\_l0 と mvd\_l1 に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.8 小小小節は、シンタックス要素 intra\_chroma\_pred\_mode に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.9 小小小節は、シンタックス要素 coded\_block\_flag に対する ctxIdxInc の導出処理を規定する。

9.3.3.1.1.10 小小小節は、シンタックス要素 transform\_size\_8x8\_flag に対する ctxIdxInc の導出処理を規定する。

#### 9.3.3.1.1.1 シンタックス要素mb\_skip\_flagに対するctxIdxIncの導出処理

この処理の出力は、ctxIdxInc である。

MbaffFrameFlag が 1 に等しく、かつトップマクロブロックアドレス  $2*(CurrMbAddr/2)$  を持つ現マクロブロックペアに対して（まだ）mb\_field\_decoding\_flag が復号されていない時、7.4.4 小節で規定されるようなシンタックス要素 mb\_field\_decoding\_flag に対する推定規則が適用される。

6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が mbAddrA と mbAddrB に割り当てられる。

変数 condTermFlagN（N は A または B）が、次の通り導出されるとする。

- もし mbAddrN が利用不可、またはマクロブロック mbAddrN に対する mb\_skip\_flag が 1 に等しいならば、condTermFlagN は 0 に等しく設定される。
- それ以外（mbAddrN が利用可能であり、かつマクロブロック mbAddrN に対する mb\_skip\_flag が 0 に等しい）、condTermFlagN は 1 に等しく設定される。

変数 ctxIdxInc は、以下で導出される。

$$ctxIdxInc = condTermFlagA + condTermFlagB \quad (9-1)$$

#### 9.3.3.1.1.2 シンタックス要素mb\_field\_decoding\_flagに対するctxIdxIncの導出処理

この処理の出力は、ctxIdxInc である。

6.4.9 小節に規定されるような MBAFF フレームにおける隣接マクロブロックアドレスおよびそれらの利用

可能性の導出処理が起動され、その出力が mbAddrA と mbAddrB に割り当てられる。

マクロブロック mbAddrN と mbAddrN+1 の両方が P\_Skip または B\_Skip に等しい mb\_type を持つ時、7.4.4 小節で規定されるようなシンタックス要素 mb\_field\_decoding\_flag に対する推定規則が、マクロブロック mbAddrN に対して適用される。

変数 condTermFlagN (N は A または B) が次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、condTermFlagN は 0 に等しく設定される。
  - mbAddrN が利用不可である。
  - マクロブロック mbAddrN がフレームマクロブロックである。
- それ以外、condTermFlagN は 1 に等しく設定される。

変数 ctxIdxInc は、以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + \text{condTermFlagB} \quad (9-2)$$

#### 9.3.3.1.1.3 シンタックス要素 mb\_type に対する ctxIdxInc の導出処理

この処理への入力、ctxIdxOffset である。

この処理の出力は、ctxIdxInc である。

6.4.10.1 小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が mbAddrA と mbAddrB に割り当てられる。

変数 condTermFlagN (N は A または B) が次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、condTermFlagN は 0 に等しく設定される。
  - mbAddrN が利用不可である
  - ctxIdxOffset が 0 に等しく、かつマクロブロック mbAddrN に対する mb\_type が SI に等しい
  - ctxIdxOffset が 3 に等しく、かつマクロブロック mbAddrN に対する mb\_type が I\_NxN に等しい
  - ctxIdxOffset が 27 に等しく、かつマクロブロック mbAddrN に対する mb\_type が P\_Skip、B\_Skip、または B\_Direct\_16x16 に等しい
- それ以外、condTermFlagN は 1 に等しく設定される。

変数 ctxIdxInc は、以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + \text{condTermFlagB} \quad (9-3)$$

#### 9.3.3.1.1.4 シンタックス要素 coded\_block\_pattern に対する ctxIdxInc の導出処理

この処理への入力、ctxIdxOffset と binIdx である。

この処理の出力は、ctxIdxInc である。

変数 ctxIdxOffset の値に従って、次が適用される。

- もし `ctxIdxOffset` が 73 に等しいならば、次が適用される。
  - 6.4.10.2 小小節で規定された隣接 8×8 輝度ブロックのための導出処理が、`luma8x8BlkIdx=binIdx` を入力として起動され、その出力が `mbAddrA`、`mbAddrB`、`luma8x8BlkIdxA` および `luma8x8BlkIdxB` に割り当てられる。
  - 変数 `condTermFlagN` (N は A または B) が次の通り導出されるとする。
    - もし以下の条件のいずれかが真であるならば、`condTermFlagN` は 0 に等しく設定される。
      - `mbAddrN` が利用不可である
      - マクロブロック `mbAddrN` に対する `mb_type` が `I_PCM` に等しい
      - マクロブロック `mbAddrN` が現マクロブロック `CurrMbAddr` ではなく、かつマクロブロック `mbAddrN` が `P_Skip` または `B_Skip` に等しい `mb_type` を持たず、かつマクロブロック `mbAddrN` に対する `CodedBlockPatternLuma` の値において、 $((\text{CodedBlockPatternLuma} \gg \text{luma8x8BlkIdxN}) \& 1)$  が 0 に等しくない
      - マクロブロック `mbAddrN` が現マクロブロック `CurrMbAddr` であり、かつ  $k = \text{luma8x8BlkIdxN}$  を持つ `coded_block_pattern` の先に復号された bin 値  $b_k$  が 0 ではない。
    - それ以外、`condTermFlagN` は 1 に等しく設定される。
  - 変数 `ctxIdxInc` は、以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + 2 * \text{condTermFlagB} \quad (9-4)$$

- それ以外 (`ctxIdxOffset` は 77 に等しい)、次が適用される。
  - 6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が `mbAddrA` および `mbAddrB` に割り当てられる。
  - 変数 `condTermFlagN` (N は A または B) が次の通り導出されるとする。
    - もし `mbAddrN` が利用可能であり、かつマクロブロック `mbAddrN` に対する `mb_type` が `I_PCM` に等しいならば、`condTermFlagN` は 1 に等しく設定される。
    - それ以外、もし以下の条件のいずれかが真であるならば、`condTermFlagN` は 0 に設定される。
      - `mbAddrN` が利用不可である、またはマクロブロック `mbAddrN` が `P_Skip` または `B_Skip` に等しい `mb_type` を持つ
      - `binIdx` が 0 に等しく、かつマクロブロック `mbAddrN` に対する `CodedBlockPatternChroma` が 0 に等しい
      - `binIdx` が 1 に等しく、かつマクロブロック `mbAddrN` に対する `CodedBlockPatternChroma` が 2 に等しくない
    - それ以外、`condTermFlagN` は 1 に等しく設定される。
  - 変数 `ctxIdxInc` は、以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + 2 * \text{condTermFlagB} + ((\text{binIdx} == 1) ? 4 : 0) \quad (9-5)$$

記 マクロブロックが Intra\_16x16 予測モードを使用する時、マクロブロックに対する CodedBlockPatternLuma と CodedBlockPatternChroma の値が表7-11/JT-H264で規定されるように mb\_type から導出される。

#### 9.3.3.1.1.5 シンタックス要素 mb\_qp\_delta に対する ctxIdxInc の導出処理

この処理の出力は、ctxIdxInc である。

prevMbAddr を復号順序で、現マクロブロックに先行するマクロブロックのマクロブロックアドレスとする。現マクロブロックがスライスの最初のマクロブロックである時は、prevMbAddr は利用不可としてマークされる。

変数 ctxIdxInc は次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、ctxIdxInc は 0 に等しく設定される。
  - prevMbAddr が利用不可である、またはマクロブロック prevMbAddr が P\_Skip または B\_Skip に等しい mb\_type を持つ
  - マクロブロック prevMbAddr の mb\_type が I\_PCM に等しい
  - Intra\_16x16 予測モードで、マクロブロック prevMbAddr が符号化されず、かつマクロブロック prevMbAddr に対する CodedBlockPatternLuma と CodedBlockPatternChroma の両方が 0 に等しい
  - マクロブロック prevMbAddr に対する mb\_qp\_delta が 0 に等しい
- それ以外、ctxIdxInc は 1 に設定される。

#### 9.3.3.1.1.6 シンタックス要素 ref\_idx\_10 および ref\_idx\_11 に対する ctxIdxInc の導出処理

この処理への入力は、mbPartIdx である。

この処理の出力は、ctxIdxInc である。

この小小小節の中の ref\_idx\_IX と Pred\_LX の解釈は次の通り規定される。

- もしこの処理が ref\_idx\_10 の導出に対して起動されるならば、ref\_idx\_IX は ref\_idx\_10 と解釈され、Pred\_LX は Pred\_L0 と解釈される。
- それ以外(この処理が ref\_idx\_11 の導出に対して起動される)、ref\_idx\_IX は ref\_idx\_11 と解釈され、Pred\_LX は Pred\_L1 と解釈される。

6.4.10.7 小小節に規定される隣接パーティションのための導出処理が、mbPartIdx、currSubMbType および subMbPartIdx=0 を入力として起動され、その出力が mbAddrA≠mbPartIdxA および mbAddrB≠mbPartIdxB に割り当てられる。

マクロブロック mbAddrN に対するシンタックス要素を規定する ref\_idx\_IX[mbPartIdxN] (N は A または B) を用いて、変数 refIdxZeroFlagN は次の通り導出されるとする。

- もし MbaffFrameFlag が 1 に等しく、現マクロブロックはフレームマクロブロックであり、そしてマクロブロック mbAddrN はフィールドマクロブロックであるならば、

$$\text{refIdxZeroFlagN} = ((\text{ref\_idx\_IX}[\text{mbPartIdxN}] > 1) ? 0 : 1) \quad (9-6)$$

- それ以外、

$$\text{refIdxZeroFlagN} = ((\text{ref\_idx\_IX}[\text{mbPartIdxN}] > 0) ? 0 : 1) \quad (9-7)$$

変数 `predModeEqualFlagN` は次の通り規定されるとする。

- もしマクロブロック `mbAddrN` が `P_8x8` または `B_8x8` に等しい `mb_type` を持つならば、次が適用される。
  - もし `SubMbPredMode(sub_mb_type[mbPartIdxN])` が `Pred_LX` に等しくなく、かつ `BiPred` に等しくないならば、`predModeEqualFlagN` は 0 に等しく設定される。ここで `sub_mb_type` は、マクロブロック `mbAddrN` に対するシンタックス要素を規定する。
  - それ以外、`predModeEqualFlagN` は 1 に等しく設定される。
- それ以外、次が適用される。
  - もし `MbPartPredMode(mb_type,mbPartIdxN)` が `Pred_LX` に等しくなく、かつ `BiPred` に等しくないならば、`predModeEqualFlagN` は 0 に等しく設定される。ここで `mb_type` は、マクロブロック `mbAddrN` に対するシンタックス要素を規定する。
  - それ以外、`predModeEqualFlagN` は 1 に等しく設定される。

変数 `condTermFlagN` (`N` は `A` または `B`) が、次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、`condTermFlagN` が 0 に等しく設定される。
  - `mbAddrN` が利用不可である
  - マクロブロック `mbAddrN` が `P_Skip` または `B_Skip` に等しい `mb_type` を持つ
  - マクロブロック `mbAddrN` が `INTRA` 予測モードで符号化される
  - `predModeEqualFlagN` が 0 に等しい
  - `refIdxZeroFlagN` が 1 に等しい
- それ以外、`condTermFlagN` は 1 に等しく設定される。

変数 `ctxIdxInc` は以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + 2 * \text{condTermFlagB} \quad (9-8)$$

### 9.3.3.1.1.7 シンタックス要素 `mvd_I0` および `mvd_I1` に対する `ctxIdxInc` の導出処理

この処理への入力は、`mbPartIdx`、`subMbPartIdx`、および `ctxIdxOffset` である。

この処理の出力は、`ctxIdxInc` である。

この小節の中の `mvd_IX` と `Pred_LX` の解釈は次の通り規定される。

- もしこの処理が `mvd_I0` の導出に対して起動されるならば、`mvd_IX` は `mvd_I0` と解釈され、`Pred_LX` は `Pred_L0` と解釈される。
- それ以外(この処理が `mvd_I1` の導出に対して起動される)、`mvd_IX` は `mvd_I1` と解釈され、`Pred_LX` は `Pred_L1` と解釈される。

currSubMbType は sub\_mb\_type[mbPartIdx] に等しく設定されるとする。

6.4.10.7 小小節で規定される隣接パーティションのための導出処理が、mbPartIdx、currSubMbType および subMbPartIdx を入力として起動され、その出力は mbAddrA¥mbPartIdxA¥subMbPartIdxA と mbAddrB¥mbPartIdxB¥subMbPartIdxB に割り当てられる。

変数 compIdx が次の通り導出されるとする。

- もし ctxIdxOffset が 40 に等しいならば、compIdx は 0 に等しく設定される。
- それ以外 (ctxIdxOffset が 47 に等しい)、compIdx は 1 に等しく設定される。

変数 predModeEqualFlagN は次の通り規定されるとする。

- もしマクロブロック mbAddrN が P\_8x8 または B\_8x8 に等しい mb\_type を持つならば、次が適用される。
  - もし SubMbPredMode(sub\_mb\_type[mbPartIdxN]) が Pred\_LX に等しくなく、かつ BiPred に等しくないならば、predModeEqualFlagN は 0 に等しく設定される。ここで sub\_mb\_type は、マクロブロック mbAddrN に対するシンタックス要素を規定する。
  - それ以外、predModeEqualFlagN は 1 に等しく設定される。
- それ以外、次が適用される。
  - もし MbPartPredMode(mb\_type,mbPartIdxN) が Pred\_LX に等しくなく、かつ BiPred に等しくないならば、predModeEqualFlagN は 0 に等しく設定される。ここで mb\_type は、マクロブロック mbAddrN に対するシンタックス要素を規定する。
  - それ以外、predModeEqualFlagN は 1 に等しく設定される。

変数 absMvdCompN (N は A または B) が次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、absMvdCompN は 0 に等しく設定される。
  - mbAddrN が利用不可である
  - マクロブロック mbAddrN が P\_Skip または B\_Skip に等しい mb\_type を持つ
  - マクロブロック mbAddrN が INTRA 予測モードで符号化される
  - predModeEqualFlagN が 0 に等しい
- それ以外、次が適用される。

- もし compIdx が 1 に等しく、MbaffFrameFlag が 1 に等しく、現マクロブロックがフレームマクロブロックであり、かつマクロブロック mbAddrN がフィールドマクロブロックであるならば、

$$\text{absMvdCompN} = \text{Abs}(\text{mvd\_IX}[\text{mbPartIdxN}][\text{subMbPartIdxN}][\text{compIdx}]) * 2 \quad (9-9)$$

- それ以外、もし compIdx が 1 に等しく、MbaffFrameFlag が 1 に等しく、現マクロブロックがフィールドマクロブロックであり、かつマクロブロック mbAddrN がフレームマクロブロックであるならば、

$$\text{absMvdCompN} = \text{Abs}(\text{mvd\_IX}[\text{mbPartIdxN}][\text{subMbPartIdxN}][\text{compIdx}]) / 2 \quad (9-10)$$

- それ以外、

$$\text{absMvdCompN} = \text{Abs}(\text{mvd\_IX}[\text{mbPartIdxN}][\text{subMbPartIdxN}][\text{compIdx}]) \quad (9-11)$$

変数 `ctxIdxInc` は、次の通り導出される。

- もし  $(\text{absMvdCompA} + \text{absMvdCompB})$  が 3 より小さいならば、`ctxIdxInc` は 0 に等しく設定される。
- それ以外、もし  $(\text{absMvdCompA} + \text{absMvdCompB})$  が 32 より大きいならば、`ctxIdxInc` は 2 に等しく設定される。
- それ以外  $(\text{absMvdCompA} + \text{absMvdCompB})$  が 3 から 32 のそれぞれの値を含む範囲内、`ctxIdxInc` は 1 に等しく設定される。

#### 9.3.3.1.1.8 シンタックス要素 `intra_chroma_pred_mode` に対する `ctxIdxInc` の導出処理

この処理の出力は、`ctxIdxInc` である。

6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が `mbAddrA` および `mbAddrB` に割り当てられる。

変数 `condTermFlagN` (N は A または B に置き換えられる) が次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、`condTermFlagN` は 0 に等しく設定される。
  - `mbAddrN` が利用不可である
  - マクロブロック `mbAddrN` が INTER 予測モードで符号化される
  - マクロブロック `mbAddrN` に対する `mb_type` が `I_PCM` に等しい
  - マクロブロック `mbAddrN` に対する `intra_chroma_pred_mode` が 0 に等しい
- それ以外、`condTermFlagN` は 1 に等しく設定される。

変数 `ctxIdxInc` は以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + \text{condTermFlagB} \quad (9-12)$$

#### 9.3.3.1.1.9 シンタックス要素 `coded_block_flag` に対する `ctxIdxInc` の導出処理

この処理への入力、`ctxBlockCat` であり、さらに追加入力は次の通りである。

- もし `ctxBlockCat` が 0、6、または 10 に等しいならば、追加入力は無し
- それ以外、もし `ctxBlockCat` が 1 または 2 に等しいならば、`luma4x4BlkIdx`
- それ以外、もし `ctxBlockCat` が 3 に等しいならば、色差成分インデックス `iCbCr`
- それ以外、もし `ctxBlockCat` が 4 に等しいならば、`chroma4x4BlkIdx` および色差成分インデックス `iCbCr`
- それ以外、もし `ctxBlockCat` が 5 に等しいならば、`luma8x8BlkIdx`
- それ以外、もし `ctxBlockCat` が 7 または 8 に等しいならば、`cb4x4BlkIdx`
- それ以外、もし `ctxBlockCat` が 9 に等しいならば、`cb8x8BlkIdx`

- それ以外、もし ctxBlockCat が 11 または 12 に等しいならば、cr4x4BlkIdx
- それ以外 (ctxBlockCat が 13 に等しい)、cr8x8BlkIdx

この処理の出力は、ctxIdxInc(ctxBlockCat)である。

変数 transBlockN (N は A または B) は次の通り導出されるとする。

- もし ctxBlockCat が 0、6、または 10 に等しいならば、次が適用される。
  - 6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が mbAddrN (N は A または B) に割り当てられる。
  - 変数 transBlockN が次の通り導出される。
    - もし mbAddrN が利用可能で、かつ Intra\_16x16 予測モードでマクロブロック mbAddrN が符号化されるならば、次が適用される。
      - もし ctxBlockCat が 0 に等しいならば、マクロブロック mbAddrN の輝度 DC ブロックが transBlockN に割り当てられる。
      - それ以外、もし ctxBlockCat が 6 に等しいならば、マクロブロック mbAddrN の Cb DC ブロックが transBlockN に割り当てられる。
      - それ以外、もし ctxBlockCat が 10 に等しいならば、マクロブロック mbAddrN の Cr DC ブロックが transBlockN に割り当てられる。
    - それ以外、transBlockN は利用不可としてマークされる。
- それ以外、もし ctxBlockCat が 1 または 2 に等しいならば、次が適用される。
  - 6.4.10.4 小小節で規定される隣接 4x4 輝度ブロックのための導出処理が、luma4x4BlkIdx を入力として起動され、その出力が mbAddrN と luma4x4BlkIdxN (N は A または B) に割り当てられる。
  - 変数 transBlockN が次の通り導出される。
    - もし mbAddrN が利用可能であり、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg (\text{luma4x4BlkIdxN} \gg 2)) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 0 に等しいならば、マクロブロック mbAddrN のインデックス luma4x4BlkIdxN を持つ 4x4 輝度ブロックが、transBlockN に割り当てられる。
    - それ以外、もし mbAddrN が利用可能であり、マクロブロック mbAddrN が P\_Skip または B\_Skip に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg (\text{luma4x4BlkIdxN} \gg 2)) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 1 に等しいならば、マクロブロック mbAddrN のインデックス  $(\text{luma4x4BlkIdxN} \gg 2)$  を持つ 8x8 輝度ブロックが、transBlockN に割り当てられる。
    - それ以外、transBlockN は利用不可としてマークされる。
- それ以外、もし ctxBlockCat が 3 に等しいならば、次が適用される。
  - 6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が

mbAddrN (N は A または B) に割り当てられる。

- 変数 transBlockN が次の通り導出される。
  - もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、かつマクロブロック mbAddrN に対する CodedBlockPatternChroma が 0 に等しくないならば、マクロブロック mbAddrN の色差成分 iCbCr の色差 DC ブロックが、transBlockN に割り当てられる。
  - それ以外、transBlockN は利用不可としてマークされる。
- それ以外、もし ctxBlockCat が 4 に等しいならば、次が適用される。
  - 6.4.10.5 小小節で規定される隣接 4×4 色差ブロックのための導出処理が、chroma4x4BlkIdx を入力として起動され、その出力が mbAddrN、chroma4x4BlkIdxN (N は A または B) に割り当てられる。
  - 変数 transBlockN が次の通り導出される。
    - もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、かつマクロブロック mbAddrN に対する CodedBlockPatternChroma が 2 に等しいならば、マクロブロック mbAddrN の色差成分 iCbCr の chroma4x4BlkIdxN を持つ 4×4 色差ブロックが、transBlockN に割り当てられる。
    - それ以外、transBlockN は利用不可としてマークされる。
  - それ以外、もし ctxBlockCat が 5 に等しいならば、次が適用される。
    - 6.4.10.2 小小節で規定される隣接 8×8 輝度ブロックのための導出処理が、luma8x8BlkIdx を入力として起動され、その出力が mbAddrN、luma8x8BlkIdxN (N は A または B) に割り当てられる。
    - 変数 transBlockN が次の通り導出される。
      - もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、マクロブロック mbAddrN に対する (( CodedBlockPatternLuma >>luma8x8BlkIdx) & 1) が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 1 に等しいならば、マクロブロック mbAddrN のインデックス luma8x8BlkIdxN を持つ 8×8 輝度ブロックが、transBlockN に割り当てられる。
      - それ以外、transBlockN は利用不可としてマークされる。
    - それ以外、もし ctxBlockCat が 7 または 8 に等しいならば、次が適用される。
      - 6.4.10.5 小小節で規定される隣接 4×4 Cb ブロックのための導出処理が、cb4x4BlkIdx を入力として起動され、その出力が mbAddrN、cb4x4BlkIdxN (N は A または B) に割り当てられる。
      - 変数 transBlockN が次の通り導出される。
        - もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、マクロブロック mbAddrN に対する (( CodedBlockPatternLuma >> (cb4x4BlkIdxN >> 2)) & 1) が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 0 に等しいならば、マクロブロック mbAddrN のインデックス cb4x4BlkIdxN を持つ 4×4 Cb ブロックが、transBlockN に割り当てられる。

- それ以外、もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip または B\_Skip に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg (\text{cb4x4BlkIdxN} \gg 2)) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 1 に等しいならば、マクロブロック mbAddrN のインデックス  $(\text{cb4x4BlkIdxN} \gg 2)$  を持つ 8×8 Cb ブロックが、transBlockN に割り当てられる。
- それ以外、transBlockN は利用不可としてマークされる。
- それ以外、もし ctxBlockCat が 9 に等しいならば、次が適用される。
  - 6.4.10.3 小小小節で規定される隣接 8×8 Cb ブロックのための導出処理が、cb8x8BlkIdx を入力として起動され、その出力が mbAddrN、cb8x8BlkIdxN (N は A または B) に割り当てられる。
  - 変数 transBlockN が次の通り導出される。
    - もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg \text{cb8x8BlkIdx}) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 1 に等しいならば、マクロブロック mbAddrN のインデックス cb8x8BlkIdxN を持つ 8×8 Cb ブロックが、transBlockN に割り当てられる。
    - それ以外、transBlockN は利用不可としてマークされる。
- それ以外、もし ctxBlockCat が 11 または 12 に等しいならば、次が適用される。
  - 6.4.10.5 小小小節で規定される隣接 4×4 Cr ブロックのための導出処理が、cr4x4BlkIdx を入力として起動され、その出力が mbAddrN、cr4x4BlkIdxN (N は A または B) に割り当てられる。
  - 変数 transBlockN が次の通り導出される。
    - もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg (\text{cr4x4BlkIdxN} \gg 2)) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 0 に等しいならば、マクロブロック mbAddrN のインデックス cr4x4BlkIdxN を持つ 4×4 Cr ブロックが、transBlockN に割り当てられる。
    - それ以外、もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip または B\_Skip に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg (\text{cr4x4BlkIdxN} \gg 2)) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 1 に等しいならば、マクロブロック mbAddrN のインデックス  $(\text{cr4x4BlkIdxN} \gg 2)$  を持つ 8×8 Cr ブロックが、transBlockN に割り当てられる。
    - それ以外、transBlockN は利用不可としてマークされる。
- それ以外 (ctxBlockCat が 13 に等しい)、次の通り適用される。
  - 6.4.10.3 小小小節で規定される隣接 8×8 Cr ブロックのための導出処理が、cr8x8BlkIdx を入力として起動され、その出力が mbAddrN、cr8x8BlkIdxN (N は A または B) に割り当てられる。
  - 変数 transBlockN が次の通り導出される。

- もし mbAddrN が利用可能で、マクロブロック mbAddrN が P\_Skip、B\_Skip、または I\_PCM に等しい mb\_type を持たず、マクロブロック mbAddrN に対する  $((\text{CodedBlockPatternLuma} \gg \text{cr8x8BlkIdx}) \& 1)$  が 0 に等しくなく、かつマクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 1 に等しいならば、マクロブロック mbAddrN のインデックス cr8x8BlkIdxN を持つ 8×8 Cr ブロックが、transBlockN に割り当てられる。
- それ以外、transBlockN は利用不可としてマークされる。

変数 condTermFlagN (N は A または B) が次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、condTermFlagN は 0 に等しく設定される。
  - mbAddrN が利用不可であり、かつ現マクロブロックが INTER 予測モードで符号化される
  - mbAddrN が利用可能で、transBlockN が利用不可であり、かつ、マクロブロック mbAddrN に対する mb\_type が I\_PCM に等しくない
  - 現マクロブロックが INTRA 予測モードで符号化され、constrained\_intra\_pred\_flag が 1 に等しく、マクロブロック mbAddrN が利用可能であり、かつ INTER 予測モードで符号化され、かつスライスデータパーティションが使われる (nal\_unit\_type が 2 から 4 のそれぞれの値を含む範囲内にある)。
- それ以外、もし以下の条件のいずれかが真であるならば、condTermFlagN は 1 に等しく設定される。
  - mbAddrN が利用不可で、かつ現マクロブロックが INTRA 予測モードで符号化される
  - マクロブロック mbAddrN に対する mb\_type が I\_PCM に等しい。
- それ以外、condTermFlagN はマクロブロック mbAddrN に対して復号された変換ブロック transBlockN の coded\_block\_flag 値に等しく設定される。

変数 ctxIdxInc(ctxBlockCat)は以下で導出される。

$$\text{ctxIdxInc}(\text{ctxBlockCat}) = \text{condTermFlagA} + 2 * \text{condTermFlagB} \quad (9-13)$$

#### 9.3.3.1.1.10 シンタックス要素 transform\_size\_8x8\_flag に対する ctxIdxInc の導出処理

この処理の出力は、ctxIdxInc である。

6.4.10.1 小小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が mbAddrA と mbAddrB に割り当てられる。

変数 condTermFlagN (N は A または B) が次の通り導出されるとする。

- もし以下の条件のいずれかが真であるならば、condTermFlagN は 0 に等しく設定される。
  - mbAddrN が利用不可である
  - マクロブロック mbAddrN に対する transform\_size\_8x8\_flag が 0 に等しい
- それ以外、condTermFlagN は 1 に等しく設定される。

変数 ctxIdxInc は、以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + \text{condTermFlagB} \quad (9-14)$$

### 9.3.3.1.2 先に復号されたbin値を用いるctxIdxIncの割り当て処理

この処理への入力は、ctxIdxOffset および binIdx である。

この処理の出力は、ctxIdxInc である。

表 9-41/JT-H264 は、ctxIdxOffset および binIdx の与えられた値に対する ctxIdxInc の規定を含む。

ctxIdxOffset および binIdx の各値に対して、先に復号された bin 値( $b_0, b_1, b_2, \dots, b_k$ )の値のいくつかを使用して ctxIdxInc が導出される。ここでインデックス  $k$  の値は binIdx 値よりも小さい。

**表9-41/JT-H264 ctxIdxOffsetおよびbinIdxの特定値に対するctxIdxIncの規定  
(ITU-T H.264)**

ctxIdxOffset値 (名前)	binIdx	ctxIdxInc
3	4	$(b_3 \neq 0)?5:6$
	5	$(b_3 \neq 0)?6:7$
14	2	$(b_1 \neq 1)?2:3$
17	4	$(b_3 \neq 0)?2:3$
27	2	$(b_1 \neq 0)?4:5$
32	4	$(b_3 \neq 0)?2:3$
36	2	$(b_1 \neq 0)?2:3$

### 9.3.3.1.3 シンタックス要素 significant\_coeff\_flag、last\_significant\_coeff\_flag および coeff\_abs\_level\_minus1 に対するctxIdxIncの割り当て処理

この処理への入力は、ctxIdxOffset および binIdx である。

この処理の出力は、ctxIdxInc である。

シンタックス要素 significant\_coeff\_flag、last\_significant\_coeff\_flag および coeff\_abs\_level\_minus1、並びに coded\_block\_flag に対する ctxIdxInc の割り当て処理は、変数 ctxBlockCat により示される異なるブロックのカテゴリに依存する。これらブロックのカテゴリ規定が表 9-42/JT-H264 で与えられる。

表9-42/JT-H264 異なるブロックに対するctxBlockCatの規定  
(ITU-T H.264)

ブロックの記述	maxNumCoeff	ctxBlockCat
輝度DC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストIntra16x16DCLevel)	16	0
輝度AC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストIntra16x16ACLevel[i])	15	1
16個の輝度変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストLumaLevel[i])	16	2
ChromaArrayTypeが1または2に等しい時の色差DC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストChromaDCLevel)	4*NumC8x8	3
ChromaArrayTypeが1または2に等しい時の色差AC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストChromaACLevel)	15	4
64個の輝度変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストLumaLevel8x8[i])	64	5
ChromaArrayTypeが3に等しい時のCb DC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCbIntra16x16DCLevel)	16	6
ChromaArrayTypeが3に等しい時のCb AC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCbIntra16x16ACLevel[i])	15	7
ChromaArrayTypeが3に等しい時の16個のCb変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCbLevel[i])	16	8
ChromaArrayTypeが3に等しい時の64個のCb変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCbLevel8x8[i])	64	9
ChromaArrayTypeが3に等しい時のCr DC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCrIntra16x16DCLevel)	16	10
ChromaArrayTypeが3に等しい時のCr AC変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCrIntra16x16ACLevel[i])	15	11
ChromaArrayTypeが3に等しい時の16個のCr変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCrLevel[i])	16	12
ChromaArrayTypeが3に等しい時の64個のCr変換係数レベルのブロック (即ち7.4.5.3小小節で記述されるリストCrLevel8x8[i])	64	13

変数 levelListIdx が 7.4.5.3 小小節で規定される変換係数レベルのリストのインデックスと等しく設定されるとする。

ctxBlockCat が 3、5、9、および 13 に等しくないブロックのシンタックス要素 significant\_coeff\_flag と last\_significant\_coeff\_flag に対して、変数 ctxIdxInc は以下で導出される。

$$\text{ctxIdxInc} = \text{levelListIdx} \quad (9-15)$$

ここで、levelListIdx は 0 から maxNumCoeff-2 のそれぞれの値を含む範囲にある。

ctxBlockCat==3 であるブロックのシンタックス要素 significant\_coeff\_flag と last\_significant\_coeff\_flag に対して、変数 ctxIdxInc は以下で導出される。

$$\text{ctxIdxInc} = \text{Min}(\text{levelListIdx} / \text{NumC8x8}, 2)$$

(9-16)

ここで、levelListIdx は 0 から  $4 * \text{NumC8x8} - 2$  のそれぞれの値を含む範囲にある。

ctxBlockCat=5、9、または 13 である  $8 \times 8$  輝度、Cb または Cr ブロックのシンタックス要素 significant\_coeff\_flag と last\_significant\_coeff\_flag に対して、表 9-43/JT-H264 は levelListIdx の値で与えられた ctxIdxInc の規定を含んでいる。ここで levelListIdx は 0 から 62 のそれぞれの値を含む範囲にある。

表9-43/JT-H264 ctxBlockCat=5、9、または13における走査位置のctxIdxIncへのマッピング  
(ITU-T H.264)

levelListIdx	significant_coeff_flagに対するctxIdxInc (フレーム符号化マクロブロック)	significant_coeff_flagに対するctxIdxInc (フィールド符号化マクロブロック)	last_significant_coeff_flagに対する ctxIdxInc	levelListIdx	significant_coeff_flagに対するctxIdxInc (フレーム符号化マクロブロック)	significant_coeff_flagに対するctxIdxInc (フィールド符号化マクロブロック)	last_significant_coeff_flagに対する ctxIdxInc
0	0	0	0	32	7	9	3
1	1	1	1	33	6	9	3
2	2	1	1	34	11	10	3
3	3	2	1	35	12	10	3
4	4	2	1	36	13	8	3
5	5	3	1	37	11	11	3
6	5	3	1	38	6	12	3
7	4	4	1	39	7	11	3
8	4	5	1	40	8	9	4
9	3	6	1	41	9	9	4
10	3	7	1	42	14	10	4
11	4	7	1	43	10	10	4
12	4	7	1	44	9	8	4
13	4	8	1	45	8	13	4
14	5	4	1	46	6	13	4
15	5	5	1	47	11	9	4
16	4	6	2	48	12	9	5
17	4	9	2	49	13	10	5

levelListIdx	significant_coeff_flagに対するctxIdxInc (フレーム符号化マクロブロック)	significant_coeff_flagに対するctxIdxInc (フィールド符号化マクロブロック)	last_significant_coeff_flagに対する ctxIdxInc	levelListIdx	significant_coeff_flagに対するctxIdxInc (フレーム符号化マクロブロック)	significant_coeff_flagに対するctxIdxInc (フィールド符号化マクロブロック)	last_significant_coeff_flagに対する ctxIdxInc
18	4	10	2	50	11	10	5
19	4	10	2	51	6	8	5
20	3	8	2	52	9	13	6
21	3	11	2	53	14	13	6
22	6	12	2	54	10	9	6
23	7	11	2	55	9	9	6
24	7	9	2	56	11	10	7
25	7	9	2	57	12	10	7
26	8	10	2	58	13	14	7
27	9	10	2	59	11	14	7
28	10	8	2	60	14	14	8
29	9	11	2	61	10	14	8
30	8	12	2	62	12	14	8
31	7	11	2				

numDecodAbsLevelEq1 を絶対値が 1 に等しい復号変換係数レベルの累積総数を示すとし、numDecodAbsLevelGt1 を絶対値が 1 より大きい復号変換係数レベルの累積総数を示すとする。両方の数は現復号処理が行われる同一の変換係数ブロックに関連付けられる。そして、coeff\_abs\_level\_minus1 の復号のために coeff\_abs\_level\_minus1 に対する ctxIdxInc が binIdx に従って次の通り規定される。

- もし binIdx が 0 に等しいならば、ctxIdxInc は以下で導出される。

$$\text{ctxIdxInc} = ((\text{numDecodAbsLevelGt1} \neq 0) ? 0 : \text{Min}(4, 1 + \text{numDecodAbsLevelEq1})) \quad (9-17)$$

- それ以外 (binIdx が 0 より大きい)、ctxIdxInc は以下で導出される。

$$\text{ctxIdxInc} = 5 + \text{Min}(4 - (\text{ctxBlockCat} == 3), \text{numDecodAbsLevelGt1}) \quad (9-18)$$

### 9.3.3.2 算術復号処理

この処理への入力は、bypassFlag、9.3.3.1 小小節で導出される ctxIdx、そして算術復号エンジンの状態変数 codIRange と codIOffset である。

この処理の出力は bin の値である。

図 9-2/JT-H264 は単一の bin に対する算術復号処理全体を図示している。bin 値の復号において、コンテキストインデックス ctxIdx は以下で規定される算術復号処理 DecodeBin(ctxIdx)に渡される。

- もし bypassFlag が 1 ならば、9.3.3.2.3 小小節で規定される DecodeBypass( )が起動される。
- それ以外、もし bypassFlag が 0 かつ ctxIdx が 276 ならば、9.3.3.2.4 小小節で規定される DecodeTerminate( )が起動される。
- それ以外(bypassFlag が 0 かつ ctxIdx が 276 でない)、9.3.3.2.1 小小節で規定される、DecodeDecision( )が適用される。

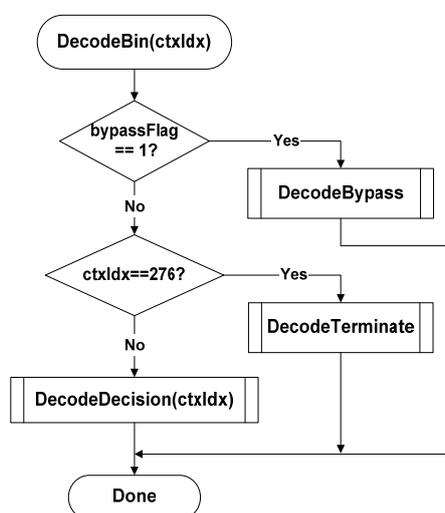


図9-2/JT-H264 単一のbinに対する算術復号処理概要 (参考)  
(ITU-T H.264)

記 算術符号化は再帰的な区間再分割の原理に基づく。バイナリ判定(0,1)の確率推定  $p(0)$  と  $p(1)=1-p(0)$  が与えられると、範囲codIRangeをもつ初めに与えられた符号部分区間はそれぞれ範囲  $p(0)*codIRange$  と  $codIRange-p(0)*codIRange$  を持つ2つの部分区間に再分割される。観測された判定に従って、対応する部分区間が新しい符号区間として選ばれ、その区間内を指すバイナリ符号記号列が、観測されるバイナリ判定の列を表して行く。優勢シンボル (MPS) と劣勢シンボル (LPS) を識別することは役に立つので、バイナリ判定は0か1よりむしろMPSかLPSのどちらかとして識別されなければならない。与えられたこの語法によって、それぞれのコンテキストはLPSの確率 $p_{LPS}$ とMPSの値(valMPS)により規定される。ここでMPSの値は0か1である。

本標準において、算術コア・エンジンは3つの独特な特性をもつ。

- 確率推定は、LPS確率 $p_{LPS}$ に対する64の異なる代表確率状態  $\{p_{LPS}(pStateIdx) | 0 \leq pStateIdx < 64\}$  間の、表ベース遷移過程をもつ有限状態機械手段により実行される。状態の番号付けは、インデックス  $pStateIdx=0$  をもつ確率状態がLPS確率値0.5に対応し、LPSの確率の減少とともにより高い状態インデックスに向かうような方法で調節される。
- 符号化エンジンの状態を表す範囲codIRangeは新しい区間範囲計算の前に、あらかじめ設定された量子化値の小集合  $\{Q_1, \dots, Q_4\}$  に量子化される。  $Q_i * p_{LPS}(pStateIdx)$  の  $64 \times 4$  全ての予め計算された積を含む表を格納していることが、積  $codIRange * p_{LPS}(pStateIdx)$  の乗算無し近似を可能にする。

- ほぼ一様確率分布が与えられたと仮定されるシンタックス要素あるいは部分については、個別の単純化された符号化、復号バイパス処理が使用される。

### 9.3.3.2.1 バイナリ判定に対する算術復号処理

この処理への入力、ctxIdx、codIRange、および codIOffset である。

この処理の出力は、復号値 binVal、更新された変数 codIRange と codIOffset である。

図 9-3/JT-H264 は 1 つの判定の復号(DecodeDecision)に対するフローチャートを示す。

1. 変数 codIRangeLPS の値は次の通り導出される。

- 与えられる現在の codIRange 値で、変数 qCodIRangeIdx が以下で導出される。

$$qCodIRangeIdx=(codIRange>>6)\&0x03 \quad (9-19)$$

- ctxIdx に関連して与えられる qCodIRangeIdx と pStateIdx で、表 9-44/JT-H264 で規定される変数 rangeTabLPS の値が codIRangeLPS に割り当てられる：

$$codIRangeLPS=rangeTabLPS[pStateIdx][qCodIRangeIdx] \quad (9-20)$$

2. 変数 codIRange が codIRange-codIRangeLPS の値に等しく設定され、次が適用される。

- もし codIOffset が codIRange 以上ならば、変数 binVal は 1-valMPS に等しく設定され、codIOffset は codIRange だけ減らされ、そして codIRange は codIRangeLPS と等しく設定される。
- それ以外、変数 binVal は valMPS に等しく設定される。

与えられた binVal の値で、9.3.3.2.1.1 小小小節で規定されるように状態遷移が行われる。codIRange の現在値により、9.3.3.2.2 小小小節で規定されるように再正規化が実行される。

#### 9.3.3.2.1.1 状態遷移処理

この処理への入力、ctxIdx に関連するコンテキスト変数の、現在の pStateIdx、復号値 binVal と、valMPS の値である。

この処理の出力は、ctxIdx に関連するコンテキスト変数の更新された pStateIdx と valMPS である。

復号値 binVal により、ctxIdx に関連する 2 つの変数 pStateIdx と valMPS の更新は次の通り導出される。

```

if(binVal==valMPS)
    pStateIdx=transIdxMPS(pStateIdx)
else{
    if(pStateIdx==0)
        valMPS=1-valMPS
    pStateIdx=transIdxLPS(pStateIdx)
}

```

(9-21)

表 9-45/JT-H264 は、それぞれ valMPS と 1-valMPS の値を復号した後の遷移規則 transIdxMPS( )と transIdxLPS( )を規定する。

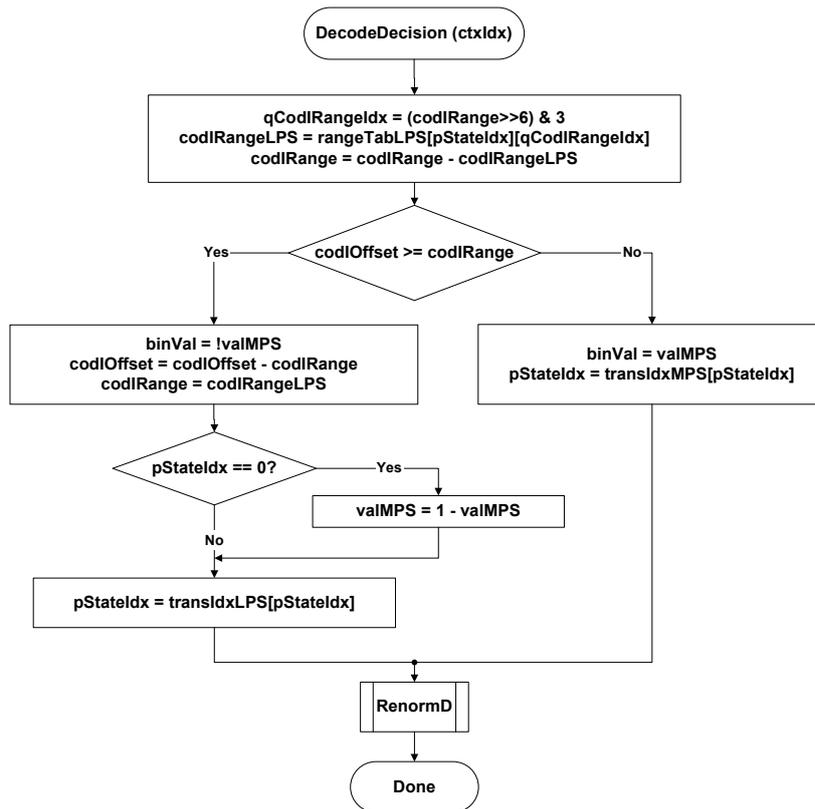


図9-3/JT-H264 判定の復号に対するフローチャート  
(ITU-T H.264)

表9-44/JT-H264 pStateIdxとqCodIRangeIdxによるrangeTabLPSの規定  
(ITU-T H.264)

pStateIdx	qCodIRangeIdx				pStateIdx	qCodIRangeIdx			
	0	1	2	3		0	1	2	3
0	128	176	208	240	32	27	33	39	45
1	128	167	197	227	33	26	31	37	43
2	128	158	187	216	34	24	30	35	41
3	123	150	178	205	35	23	28	33	39
4	116	142	169	195	36	22	27	32	37
5	111	135	160	185	37	21	26	30	35
6	105	128	152	175	38	20	24	29	33
7	100	122	144	166	39	19	23	27	31
8	95	116	137	158	40	18	22	26	30
9	90	110	130	150	41	17	21	25	28

<b>10</b>	85	104	123	142	<b>42</b>	16	20	23	27
<b>11</b>	81	99	117	135	<b>43</b>	15	19	22	25
<b>12</b>	77	94	111	128	<b>44</b>	14	18	21	24
<b>13</b>	73	89	105	122	<b>45</b>	14	17	20	23
<b>14</b>	69	85	100	116	<b>46</b>	13	16	19	22
<b>15</b>	66	80	95	110	<b>47</b>	12	15	18	21
<b>16</b>	62	76	90	104	<b>48</b>	12	14	17	20
<b>17</b>	59	72	86	99	<b>49</b>	11	14	16	19
<b>18</b>	56	69	81	94	<b>50</b>	11	13	15	18
<b>19</b>	53	65	77	89	<b>51</b>	10	12	15	17
<b>20</b>	51	62	73	85	<b>52</b>	10	12	14	16
<b>21</b>	48	59	69	80	<b>53</b>	9	11	13	15
<b>22</b>	46	56	66	76	<b>54</b>	9	11	12	14
<b>23</b>	43	53	63	72	<b>55</b>	8	10	12	14
<b>24</b>	41	50	59	69	<b>56</b>	8	9	11	13
<b>25</b>	39	48	56	65	<b>57</b>	7	9	11	12
<b>26</b>	37	45	54	62	<b>58</b>	7	9	10	12
<b>27</b>	35	43	51	59	<b>59</b>	7	8	10	11
<b>28</b>	33	41	48	56	<b>60</b>	6	8	9	11
<b>29</b>	32	39	46	53	<b>61</b>	6	7	9	10
<b>30</b>	30	37	43	50	<b>62</b>	6	7	8	9
<b>31</b>	29	35	41	48	<b>63</b>	2	2	2	2

表9-45/JT-H264 状態遷移表  
(ITU-T H.264)

<b>pStateIdx</b>	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>	<b>15</b>
<b>transIdxLPS</b>	0	0	1	2	2	4	4	5	6	7	8	9	9	11	11	12
<b>transIdxMPS</b>	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
<b>pStateIdx</b>	<b>16</b>	<b>17</b>	<b>18</b>	<b>19</b>	<b>20</b>	<b>21</b>	<b>22</b>	<b>23</b>	<b>24</b>	<b>25</b>	<b>26</b>	<b>27</b>	<b>28</b>	<b>29</b>	<b>30</b>	<b>31</b>
<b>transIdxLPS</b>	13	13	15	15	16	16	18	18	19	19	21	21	22	22	23	24
<b>transIdxMPS</b>	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
<b>pStateIdx</b>	<b>32</b>	<b>33</b>	<b>34</b>	<b>35</b>	<b>36</b>	<b>37</b>	<b>38</b>	<b>39</b>	<b>40</b>	<b>41</b>	<b>42</b>	<b>43</b>	<b>44</b>	<b>45</b>	<b>46</b>	<b>47</b>
<b>transIdxLPS</b>	24	25	26	26	27	27	28	29	29	30	30	30	31	32	32	33
<b>transIdxMPS</b>	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48
<b>pStateIdx</b>	<b>48</b>	<b>49</b>	<b>50</b>	<b>51</b>	<b>52</b>	<b>53</b>	<b>54</b>	<b>55</b>	<b>56</b>	<b>57</b>	<b>58</b>	<b>59</b>	<b>60</b>	<b>61</b>	<b>62</b>	<b>63</b>
<b>transIdxLPS</b>	33	33	34	34	35	35	35	36	36	36	37	37	37	38	38	63
<b>transIdxMPS</b>	49	50	51	52	53	54	55	56	57	58	59	60	61	62	62	63

### 9.3.3.2.2 算術復号エンジンの再正規化処理

この処理への入力、スライスデータからのビットと変数 `codIRange`、`codIOffset` である。

この処理の出力は、更新された変数 `codIRange`、`codIOffset` である。

再正規化のフローチャートが図 9-4/JT-H264 で示される。`codIRange` の現在値は最初に `0x0100` と比較され、更なるステップが次の通り規定される。

- もし `codIRange` が `0x0100` 以上ならば、再正規化は必要なく、`RenormD` 処理は終了する。
- それ以外(`codIRange` が `0x0100` 未満)、再正規化ループが開始される。このループ内で `codIRange` 値は 2 倍される、すなわち 1 だけ左にシフトされ、`codIOffset` には `read_bits(1)` を使用して 1 ビットが繰り入れられる。

ビットストリームは、この処理の終了での結果として `codIOffset` の値が `codIRange` 以上になるデータを含んではならない。

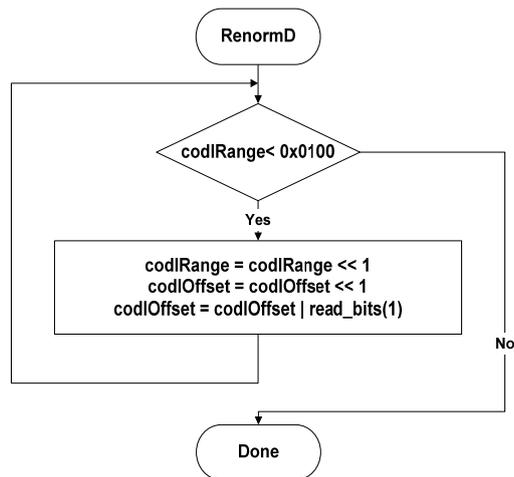


図9-4/JT-H264 再正規化のフローチャート  
(ITU-T H.264)

### 9.3.3.2.3 バイナリ判定に対するバイパス復号処理

この処理への入力、スライスデータからのビットと変数 `codIRange`、`codIOffset` である。

この処理の出力は、更新された変数 `codIOffset` と復号値 `binVal` である。

`bypassFlag` が 1 の時、バイパス復号処理は起動される。図 9-5/JT-H264 は対応する処理のフローチャートを示す。

初めに、`codIOffset` の値は 2 倍される、すなわち 1 だけ左にシフトされ、`read_bits(1)` を使って `codIOffset` に 1 ビットが繰り返し入れられる。次に `codIOffset` の値が `codIRange` の値と比較される。そして、更なるステップが次の通り規定される。

- もし `codIOffset` が `codIRange` 以上ならば、変数 `binVal` が 1 に等しく設定され、`codIOffset` は `codIRange` だけ減らされる。
- それ以外(`codIOffset` が `codIRange` 未満)、変数 `binVal` は 0 に等しく設定される。

ビットストリームは、この処理の終了での結果として `codIOffset` の値が `codIRange` 以上になるデータを含んではならない。

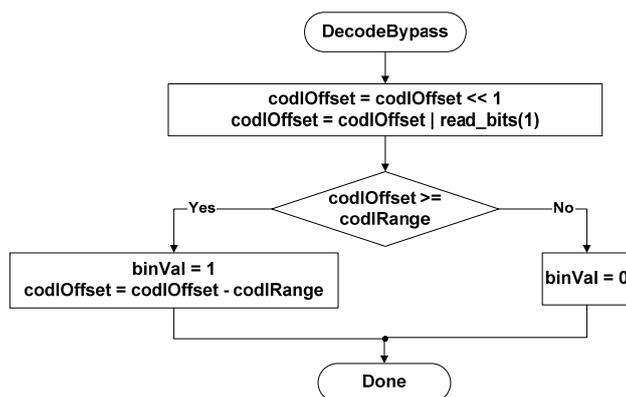


図9-5/JT-H264 パイパス復号処理のフローチャート  
(ITU-T H.264)

#### 9.3.3.2.4 終端前のバイナリ判定の復号処理

この処理への入力、スライスデータからのビットと変数 `codlRange`、`codlOffset` である。

この処理の出力は、更新された変数 `codlRange`、`codlOffset`、復号値 `binVal` である。

この特別な復号ルーチンは、276 に等しい `ctxIdx` に対応する `end_of_slice_flag` と I\_PCM モードを示す `bin` を復号するのに適用される。図 9-6/JT-H264 は対応する復号処理のフローチャートを示し、それは次の通り規定される。

初めに `codlRange` 値は 2 だけ減らされる。次に `codlOffset` 値が `codlRange` 値と比較され、そして更なるステップが次の通り規定される。

- もし `codlOffset` が `codlRange` 以上ならば、変数 `binVal` は 1 に等しく設定され、再正規化は実行されずに CABAC 復号は終了する。レジスタ `codlOffset` に挿入された最後のビットは 1 に等しい。`end_of_slice_flag` を復号する時、レジスタ `codlOffset` に挿入されたこの最後のビットは `rbsp_stop_one_bit` として解釈される。
- それ以外(`codlOffset` が `codlRange` 未満)、変数 `binVal` は 0 に等しく設定され、9.3.3.2.2 小小節で規定されるような再正規化が実行される。

記 この手続きはまた、`ctxIdx=276`である`DecodeDecision(ctxIdx)`を使用して実装されてもよい。復号値が1と等しい場合、さらに7ビットが`DecodeDecision(ctxIdx)`により読みこまれ、復号処理は、続くシンタックス要素を適切に復号できるように、そのビットストリームポインタを調整しなければならない。

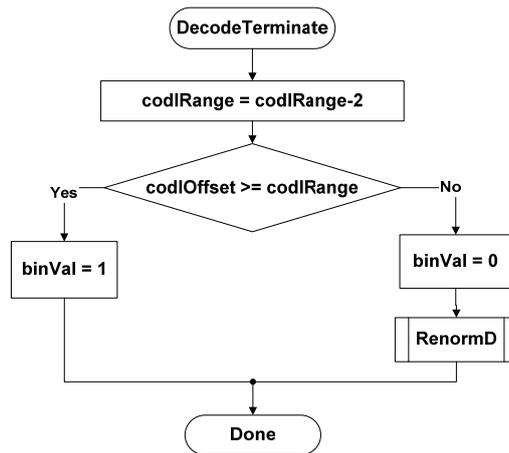


図9-6/JT-H264 終端前の判定復号のフローチャート  
(ITU-T H.264)

### 9.3.4 算術符号化処理 (参考)

この小節は、本標準の必須部分を形成しない。

この処理への入力は符号化され書かれる判定である。

この処理の出力は RBSP に書かれるビットである。

この参考小節は、9.3.3.2 小節で記述される算術復号エンジンと適合する算術符号化エンジンを記述する。符号化エンジンは復号エンジンと本質的に対称である、すなわち手続きは同じ順番で呼ばれる。以下の手続きがこの小節で記述される：InitEncoder、EncodeDecision、EncodeBypass、EncodeTerminate。これらはそれぞれ InitDecoder、DecodeDecision、DecodeBypass、DecodeTerminate に対応する。算術符号化エンジンの状態は、部分区間の下端を示す変数 `codILow` の値と、その部分区間の対応する範囲を規定する変数 `codIRange` の値によって表される。

#### 9.3.4.1 算術符号化エンジンの初期化処理 (参考)

この小節は、本標準の必須部分を形成しない。

この処理は、スライスの最初のマクロブロックを符号化する前と、タイプ I\_PCM マクロブロックに対するいくつかの `pcm_alignment_zero_bit` と全ての `pcm_sample_luma` と `pcm_sample_chroma` データを符号化した後に起動される。

この処理の出力は、算術符号化エンジンの値 `codILow`、`codIRange`、`firstBitFlag`、`bitsOutstanding`、および `symCnt` である。

符号器の初期化手続きで、`codILow` は 0 に設定され、`codIRange` は 0x01FE に設定される。さらに `firstBitFlag` は 1 に設定され、`bitsOutstanding` と `symCnt` カウンタは 0 に設定される。

記 `codILow` について必要な最小レジスタ精度は10ビットであり、`codIRange` については9ビットである。カウンタ `bitsOutstanding` と `symCnt` に必要な精度は、関連するレジスタのオーバーフローを防ぐために十分大きくすべきである。`MaxBinCountInSlice` が1スライス中で符号化するバイナリ判定の最大総数を示す時、変数 `bitsOutstanding` と `symCnt` に必要な最小レジスタ精度は  $\text{Ceil}(\text{Log}_2(\text{MaxBinCountInSlice}+1))$  によって与えられる。

#### 9.3.4.2 バイナリ判定の符号化処理(参考)

この小小節は、本標準の必須部分を形成しない。

この処理への入力、コンテキストインデックス `ctxIdx`、符号化される `binVal` 値と、変数 `codIRange`、`codILow` および `symCnt` である。

この処理の出力は、変数 `codIRange`、`codILow`、および `symCnt` である。

図 9-7/JT-H264 は 1 つの判定を符号化するためのフローチャートを示す。第 1 のステップでは、変数 `codIRangeLPS` は次の通り導出される。

与えられた `codIRange` の現在値で、`codIRange` は等式 9-19 を使用して、`codIRange` の量子化値のインデックス `qCodIRangeIdx` にマップされる。`ctxIdx` に関連付けられた `qCodIRangeIdx` 値と `pStateIdx` 値は、表 9-44/JT-H264 で規定されるように変数 `rangeTabLPS` の値を決定するために使用され、それは `codIRangeLPS` に割り当てられる。`codIRange-codIRangeLPS` の値は `codIRange` に割り当てられる。

第 2 のステップでは、`binVal` 値は `ctxIdx` に関連付けられた `valMPS` と比較される。`binVal` が `valMPS` と異なる時、`codIRange` は `codILow` に加えられ、`codIRange` は `codIRangeLPS` の値と等しく設定される。与えられた符号化判定で、状態遷移は 9.3.3.2.1.1 小小小節で規定されるように実行される。`codIRange` の現在値に従い、再正規化が 9.3.4.3 小小節で規定されるように実行される。最後に、変数 `symCnt` は 1 増加される。

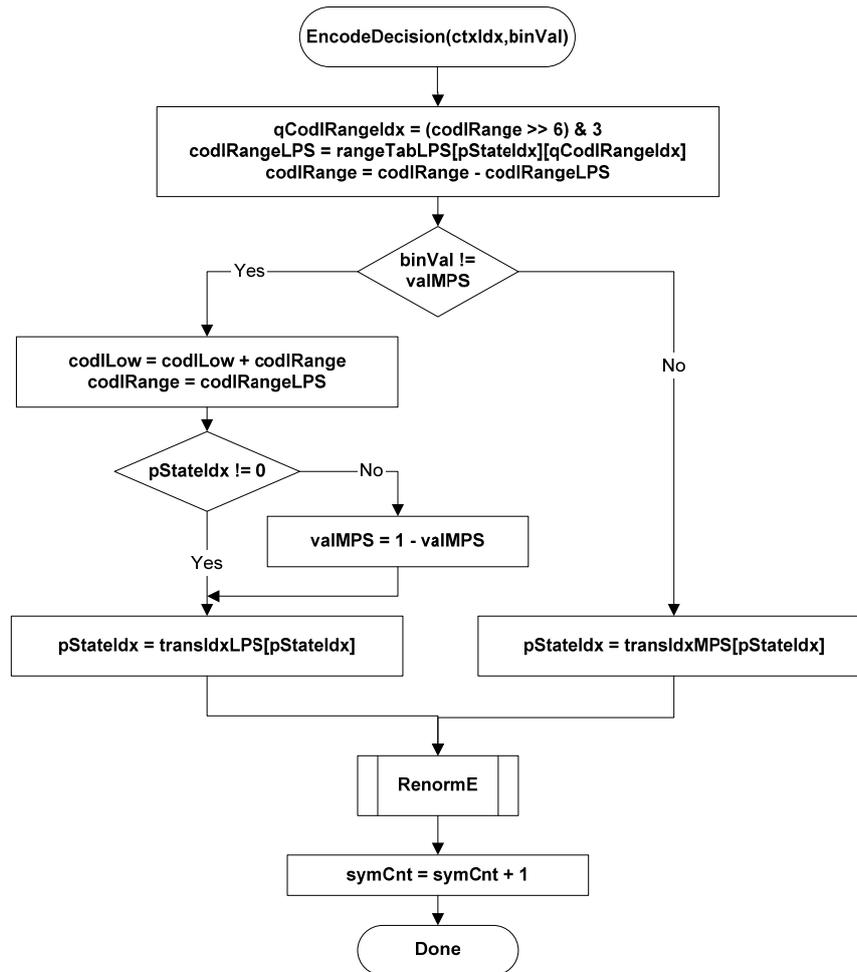


図9-7/JT-H264 判定を符号化するためのフローチャート  
(ITU-T H.264)

### 9.3.4.3 算術符号化エンジンの再正規化処理（参考）

この小節は、本標準の必須部分を形成しない。

この処理への入力、変数 `codIRange`、`codILow` と、`firstBitFlag`、および `bitsOutstanding` である。

この処理の出力は、RBSP に書かれる 0 個以上のビットと、更新された変数 `codIRange`、`codILow`、`firstBitFlag`、および `bitsOutstanding` である。

再正規化は図 9-8/JT-H264 で図示される。

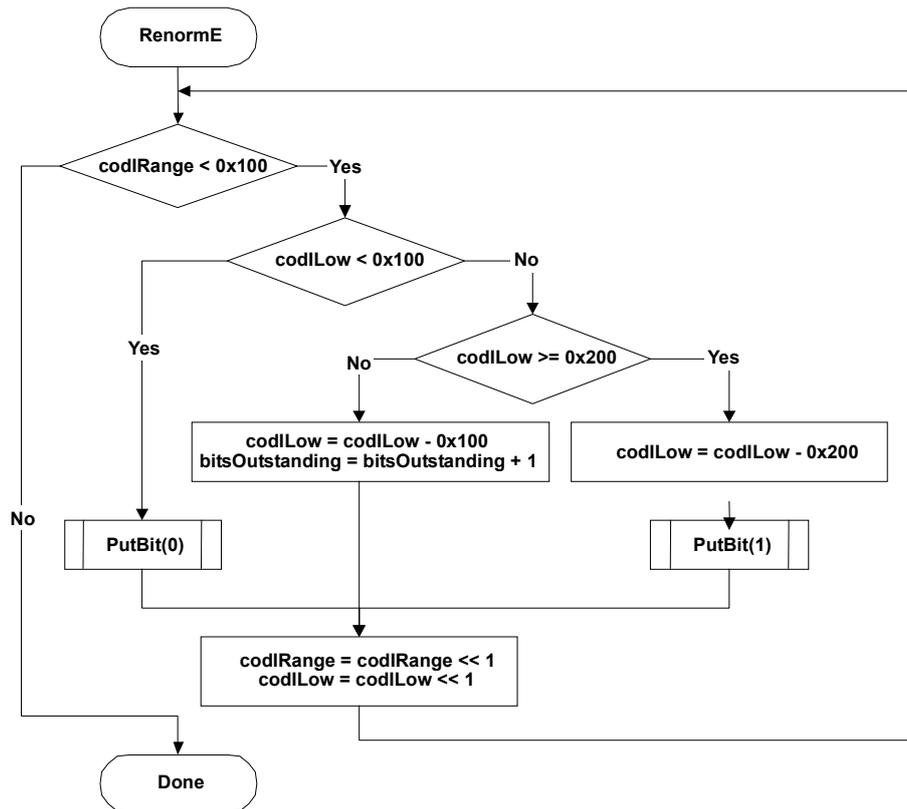


図9-8/JT-H264 符号器の再正規化のフローチャート  
(ITU-T H.264)

図 9-9/JT-H264 で記述される PutBit( )手続きは桁上がり制御を与える。それはビットストリームに値 B をもつ N ビットを書き込み、N ビット位置分だけビットストリームポインタを進める関数 WriteBits(B,N)を使用する。この関数は、符号化処理でビットストリームに書かれるべき次のビット位置を示すビットストリームポインタの存在を仮定する。

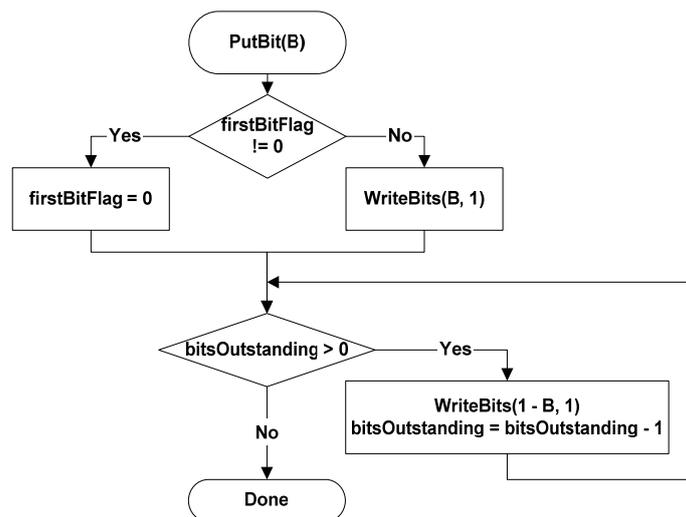


図9-9/JT-H264 PutBit(B)フローチャート  
(ITU-T H.264)

#### 9.3.4.4 バイナリ判定のためのバイパス符号化処理(参考)

この小節は、本標準の必須部分を形成しない。

この処理への入力、変数 binVal、codILow、codIRange、bitsOutstanding、および symCnt である。

この処理の出力は RBSP に書かれるビット、更新された変数 codILow、bitsOutstanding、および symCnt である。

この符号化処理は bypassFlag が 1 である全てのバイナリ判定に適用される。再正規化は図 9-10/JT-H264 で与えられるように、この処理の規定に含まれる。

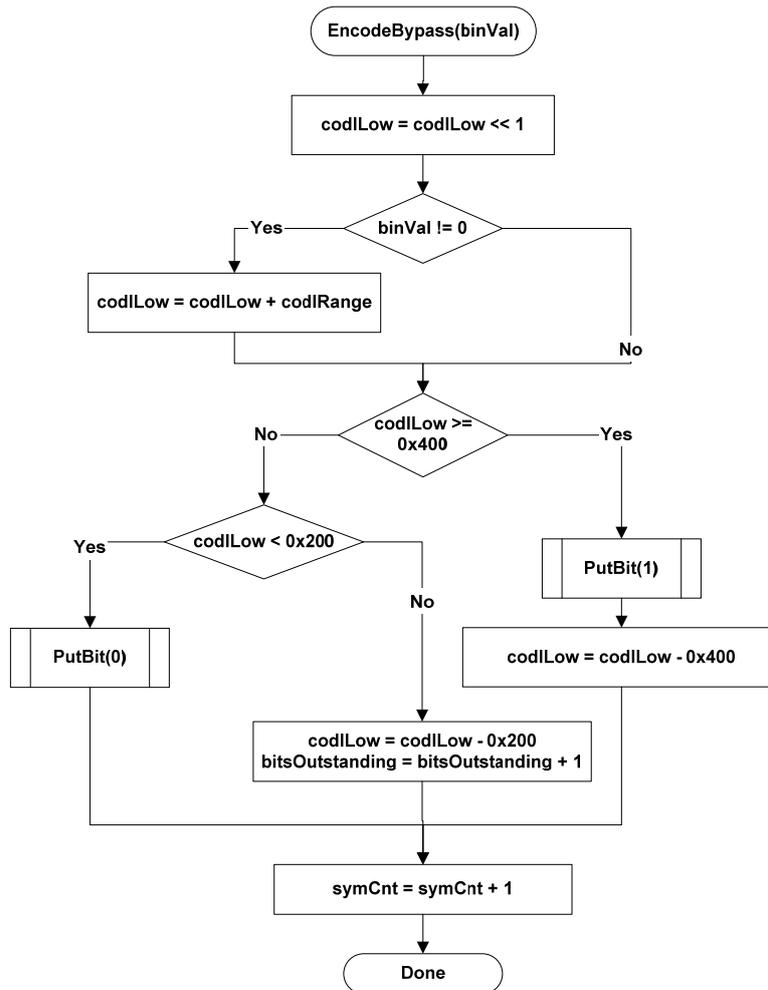


図9-10/JT-H264 バイパス符号化のフローチャート  
(ITU-T H.264)

#### 9.3.4.5 終端前のバイナリ判定のための符号化処理(参考)

この小節は、本標準の必須部分を形成しない。

この処理への入力は、変数 binVal、codIRange、codILow、bitsOutstanding、および symCnt である。

この処理の出力は、RBSP に書かれる 0 個以上のビットと、更新された変数 codILow、codIRange、bitsOutstanding、および symCnt である。

図9-11/JT-H264 で示されるこの符号化ルーチンは、276 に等しい ctxIdx に関連付けられた end\_of\_slice\_flag と I\_PCM の mb\_type を示す bin の両方の符号化に適用される。

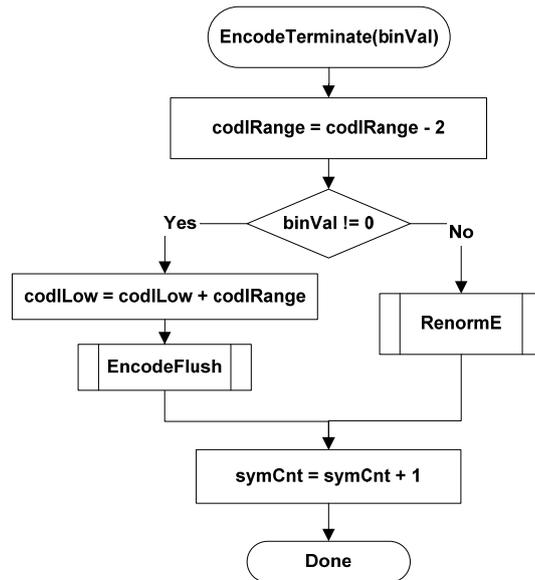


図9-11/JT-H264 終端前の判定を符号化するためのフローチャート  
(ITU-T H.264)

符号化する binVal 値が 1 である時、CABAC 符号化は終了され、図 9-12/JT-H264 で示されるフラッシュ手続きが適用される。このフラッシュ手続きでは、WriteBits(B,N)によって書かれる最後のビットは 1 に等しい。end\_of\_slice\_flag を符号化する場合、この最後のビットは rbsp\_stop\_one\_bit として解釈される。

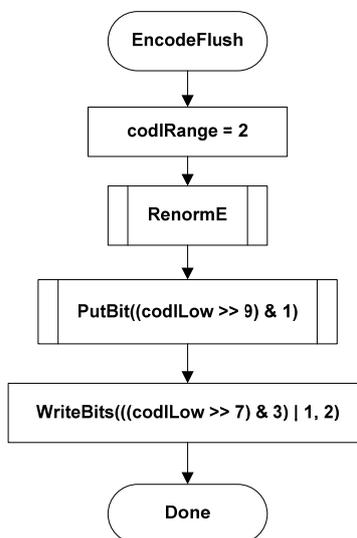


図9-12/JT-H264 終端時のフラッシュのフローチャート  
(ITU-T H.264)

#### 9.3.4.6 バイトスタッフィング処理(参考)

この小節は、本標準の必須部分を形成しない。

この処理は、ピクチャの最終スライスの最終マクロブロックを符号化してカプセル化した後に起動される。

この処理への入力、1つのピクチャの全 VCL NAL ユニットのバイト数 NumBytesInVclNALunits、そのピクチャのマクロブロック数 PicSizeInMbs、そしてそのピクチャの全 VCL NAL ユニットの内容を符号化した結果であるバイナリシンボル数 BinCountsInNALunits である。

この処理の出力は NAL ユニットの追加される 0 個以上のバイトである。

変数  $k$  を  $\text{Ceil}(\text{Ceil}(3 \cdot (32 \cdot \text{BinCountsInNALunits} - \text{RawMbBits} \cdot \text{PicSizeInMbs}) \div 1024) - \text{NumBytesInVclNALunits}) \div 3$  と等しく設定するとする。変数  $k$  に従い次が適用される。

- もし  $k$  が 0 以下ならば、cabac\_zero\_word は NAL ユニットの追加されない。
- それ以外( $k$  は 0 より大きい)、3 バイトシーケンス 0x000003 がカプセル化の後に NAL ユニットの  $k$  回追加される。ここで、最初の 2 バイト 0x0000 は cabac\_zero\_word を表し、3 番目のバイトの 0x03 が emulation\_prevention\_three\_byte を表す。

## 付属資料A プロフィールとレベル

(この付属資料は、本標準の必須部分を形成する)

プロフィールとレベルは、ビットストリームの制限と、それによってビットストリームを復号するために必要とされる能力の限定を規定する。プロフィールとレベルは、個々の復号器の実装における相互接続性のポイントを示すために使用されても良い。

記1 相互接続性の難しさを増加させるので、本標準は復号器における個々の選択可能“オプション”を含まない。

各プロフィールは、そのプロフィールに適合する全ての復号器でサポートされなければならない、アルゴリズム機能および限定のサブセットを規定する。

記2 符号器は、あるプロフィールでサポートされる機能の任意の特定サブセットを全て使用することを要求されない。

各レベルは、本標準のシンタックス要素で使用できる値の限定のセットを規定する。同じレベル定義のセットが全てのプロフィールで使用されるが、個々の実装はサポートされたプロフィール毎に異なるレベルをサポートしても良い。任意の与えられたプロフィールにおいて、レベルは一般的に復号器の処理負荷とメモリ容量に対応する。

### A.1 ビデオ復号器能力の要件

本標準に適合するビデオ復号器の能力は、この付属資料で規定されるプロフィールおよびレベルの制約に適合するビデオストリームを復号するための能力で規定される。このような個々のプロフィールに対しては、そのプロフィールでサポートされるレベルもまた表現されなければならない。

シンタックス要素 profile\_idc および level\_idc に対して、特定の値がこの付属資料の中で規定される。profile\_idc および level\_idc の他の全ての値は、将来の使用のために TTC によって予約される。

記 復号器は、将来の予約値使用において TTC で選択される方法に制限がないため、profile\_idc または level\_idc の予約値が本標準で規定された値の間に入る場合、これが規定されたプロフィールまたはレベルの中間の能力を示すと推定すべきではない。

## A.2 プロファイル

### A.2.1 基本プロファイル (Baseline)

基本プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- IおよびPのスライスタイプだけが存在してよい。
- NALユニットストリームは、2から4のそれぞれの値を含む範囲のnal\_unit\_type値を含んではならない。
- シーケンスパラメータセットは、1に等しいframe\_mbs\_only\_flagを有さなければならない。
- シンタックス要素 chroma\_format\_idc 、 bit\_depth\_luma\_minus8 、 bit\_depth\_chroma\_minus8 、 qp\_prime\_y\_zero\_transform\_bypass\_flag、 および seq\_scaling\_matrix\_present\_flagは、シーケンスパラメータセットに存在してはならない。
- ピクチャパラメータセットは、両方が0に等しいweighted\_pred\_flagとweighted\_bipred\_idcを有さなければならない。
- ピクチャパラメータセットは、0に等しいentropy\_coding\_mode\_flagを有さなければならない。
- ピクチャパラメータセットは、0から7のそれぞれの値を含む範囲のnum\_slice\_groups\_minus1を有さなければならない。
- シンタックス要素 transform\_8x8\_mode\_flag 、 pic\_scaling\_matrix\_present\_flag 、 および second\_chroma\_qp\_index\_offsetは、ピクチャパラメータセットに存在してはならない。
- シンタックス要素level\_prefixは、15よりも大きくてはならない。
- 基本プロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

基本プロファイルへのビットストリームの適合は、profile\_idc が 66 であることによって規定される。

特定のレベルで基本プロファイルに適合する復号器は、profile\_idc が 66 であるか constraint\_set0\_flag が 1 であり、かつ level\_idc と constraint\_set3\_flag がその指定されたレベルより小さいか等しいレベルを表す、全てのビットストリームを復号できなければならない。

### A.2.2 メインプロファイル (Main)

メインプロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- I、PおよびBのスライスタイプのみが存在してよい。
- NALユニットストリームは、2から4のそれぞれの値を含む範囲のnal\_unit\_type値を含んではならない。
- 任意スライス順序は許可されない。
- シンタックス要素 chroma\_format\_idc 、 bit\_depth\_luma\_minus8 、 bit\_depth\_chroma\_minus8 、 qp\_prime\_y\_zero\_transform\_bypass\_flag、 および seq\_scaling\_matrix\_present\_flagは、シーケンスパラメータセットに存在してはならない。
- ピクチャパラメータセットは、0のみに等しいnum\_slice\_groups\_minus1を有さなければならない。
- ピクチャパラメータセットは、0のみに等しいredundant\_pic\_cnt\_present\_flagを有さなければならない。

- シンタックス要素 `transform_8x8_mode_flag`、`pic_scaling_matrix_present_flag`、および `second_chroma_qp_index_offset`は、ピクチャパラメータセットに存在してはならない。
- シンタックス要素 `level_prefix`は、（存在する時）15よりも大きくてはならない。
- メインプロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

メインプロファイルへのビットストリームの適合は、`profile_idc` が 77 であることによって規定される。

特定のレベルでメインプロファイルに適合する復号器は、`profile_idc` が 77 であるか `constraint_set1_flag` が 1 であり、かつ `level_idc` と `constraint_set3_flag` がその指定されたレベルより小さいか等しいレベルを表す、全てのビットストリームを復号できなければならない。

### A.2.3 拡張プロファイル (Extended)

拡張プロファイルに適合するビットストリームは、次の制約に従わなければならない。

- シーケンスパラメータセットは、1に等しい`direct_8x8_inference_flag`を有さなければならない。
- シンタックス要素 `chroma_format_idc`、`bit_depth_luma_minus8`、`bit_depth_chroma_minus8`、`qprime_y_zero_transform_bypass_flag`、および`seq_scaling_matrix_present_flag`は、シーケンスパラメータセットに存在してはならない。
- ピクチャパラメータセットは、0に等しい`entropy_coding_mode_flag`を有さなければならない。
- ピクチャパラメータセットは、0から7のそれぞれの値を含む範囲の`num_slice_groups_minus1`を有さなければならない。
- シンタックス要素 `transform_8x8_mode_flag`、`pic_scaling_matrix_present_flag`、および `second_chroma_qp_index_offset`は、ピクチャパラメータセットに存在してはならない。
- シンタックス要素 `level_prefix`は、（存在する時）15よりも大きくてはならない。
- 拡張プロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

拡張プロファイルへのビットストリームの適合は、`profile_idc` が 88 であることによって規定される。

特定のレベルで拡張プロファイルに適合する復号器は、`profile_idc` が 88 であるか `constraint_set2_flag` が 1 であり、かつ `level_idc` がその指定されたレベルより小さいか等しいレベルを表す、全てのビットストリームを復号できなければならない。

特定のレベルで拡張プロファイルに適合する復号器は、`profile_idc` が 66 であるか `constraint_set0_flag` が 1 であり、かつ `level_idc` と `constraint_set3_flag` がその指定されたレベルより小さいか等しいレベルを表す、全てのビットストリームも復号できなければならない。

### A.2.4 ハイプロファイル (High)

ハイプロファイルに適合するビットストリームは、次の制約に従わなければならない。

- I、P、およびBのスライスタイプのみが存在してよい。
- NALユニットストリームは、2から4のそれぞれの値を含む範囲の`nal_unit_type`値を含んではならない。

- 任意スライス順序は許可されない。
- ピクチャパラメータセットは、0のみに等しいnum\_slice\_groups\_minus1を有さなければならない。
- ピクチャパラメータセットは、0のみに等しいredundant\_pic\_cnt\_present\_flagを有さなければならない。
- シーケンスパラメータセットは、0から1のそれぞれの値を含む範囲のchroma\_format\_idcを有さなければならない。
- シーケンスパラメータセットは、0のみに等しいbit\_depth\_luma\_minus8を有さなければならない。
- シーケンスパラメータセットは、0のみに等しいbit\_depth\_chroma\_minus8を有さなければならない。
- シーケンスパラメータセットは、0のみに等しいqprime\_y\_zero\_transform\_bypass\_flagを有さなければならない。
- ハイプロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイプロファイルへのビットストリームの適合は、profile\_idc が 100 であることによって規定される。特定のレベルでハイプロファイルに適合する復号器は、次の条件のいずれかあるいは両方が真である全てのビットストリームを復号できなければならない。

- profile\_idcが77であるかconstraint\_set1\_flagが1であり、かつlevel\_idcおよびconstraint\_set3\_flagの組合せがその指定されたレベルより小さいか等しいレベルを表す。
- profile\_idcが100に等しく、かつlevel\_idcがその指定されたレベルより小さいか等しいレベルを表す。

記 profile\_idcの値100は、ビットストリームがこの小節で規定されたハイプロファイルに適合することを示す。profile\_idcが100であり、かつconstraint\_set3\_flagが1に等しい時、この事はビットストリームがハイプロファイルに適合し、さらにハイ10 INTRAプロファイルに対してA.2.8小節で規定された制約に適合することを示す。例えば、そのようなビットストリームは、0に等しいbit\_depth\_luma\_minus8を持ち、0に等しいbit\_depth\_chroma\_minus8を持ち、ハイプロファイルのMinCR、MaxBR、およびMaxCPBの制約に従い、IDRピクチャのみを含み、0に等しいnum\_ref\_framesを持ち、0に等しいdpb\_output\_delayを持ち、ハイ10 INTRAプロファイルの最大スライスサイズの制約に従う。

#### A.2.5 ハイ 10 プロファイル (High 10)

ハイ 10 プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- I、P、およびBのスライスタイプのみが存在してよい。
- NALユニットストリームは、2から4のそれぞれの値を含む範囲のnal\_unit\_type値を含んではならない。
- 任意スライス順序は許可されない。
- ピクチャパラメータセットは、0のみに等しいnum\_slice\_groups\_minus1を有さなければならない。
- ピクチャパラメータセットは、0のみに等しいredundant\_pic\_cnt\_present\_flagを有さなければならない。
- シーケンスパラメータセットは、0から1のそれぞれの値を含む範囲のchroma\_format\_idcを有さなければならない。
- シーケンスパラメータセットは、0から2のそれぞれの値を含む範囲のbit\_depth\_luma\_minus8を有さなければならない。

- シーケンスパラメータセットは、0から2のそれぞれの値を含む範囲のbit\_depth\_chroma\_minus8を有さなければならない。
- シーケンスパラメータセットは、0のみに等しいqprime\_y\_zero\_transform\_bypass\_flagを有さなければならない。
- ハイ10プロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイ 10 プロファイルへのビットストリームの適合は、profile\_idc が 110 であることによって規定される。特定のレベルでハイ 10 プロファイルに適合する復号器は、次の条件のいずれかあるいは両方が真である全てのビットストリームを復号できなければならない。

- profile\_idcが77であるかconstraint\_set1\_flagが1であり、かつlevel\_idcおよびconstraint\_set3\_flagの組合せがその指定されたレベルより小さいか等しいレベルを表す。
- profile\_idcが100または110に等しく、かつlevel\_idcがその指定されたレベルより小さいか等しいレベルを表す。

#### A.2.6 ハイ 4:2:2 プロファイル (High 4:2:2)

ハイ 4:2:2 プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- I、P、およびBのスライスタイプのみが存在してよい。
- NALユニットストリームは、2から4のそれぞれの値を含む範囲のnal\_unit\_type値を含んではならない。
- 任意スライス順序は許可されない。
- ピクチャパラメータセットは、0のみに等しいnum\_slice\_groups\_minus1を有さなければならない。
- ピクチャパラメータセットは、0のみに等しいredundant\_pic\_cnt\_present\_flagを有さなければならない。
- シーケンスパラメータセットは、0から2のそれぞれの値を含む範囲のchroma\_format\_idcを有さなければならない。
- シーケンスパラメータセットは、0から2のそれぞれの値を含む範囲のbit\_depth\_luma\_minus8を有さなければならない。
- シーケンスパラメータセットは、0から2のそれぞれの値を含む範囲のbit\_depth\_chroma\_minus8を有さなければならない。
- シーケンスパラメータセットは、0のみに等しいqprime\_y\_zero\_transform\_bypass\_flagを有さなければならない。
- ハイ4:2:2プロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイ 4:2:2 プロファイルへのビットストリームの適合は、profile\_idc が 122 であることによって規定される。特定のレベルでハイ 4:2:2 プロファイルに適合する復号器は、次の条件のいずれかあるいは両方が真である全てのビットストリームを復号できなければならない。

- profile\_idcが77であるかconstraint\_set1\_flagが1であり、かつlevel\_idcおよびconstraint\_set3\_flagの組合せがその指定されたレベルより小さいか等しいレベルを表す。

- profile\_idcが100、110、または122に等しく、かつlevel\_idcがその指定されたレベルより小さいか等しいレベルを表す。

#### A.2.7 ハイ 4:4:4 予測プロファイル (High 4:4:4 Predictive)

ハイ 4:4:4 予測プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- I、P、およびBのスライスタイプのみが存在してよい。
- NALユニットストリームは、2から4のそれぞれの値を含む範囲のnal\_unit\_type値を含んではならない。
- 任意スライス順序は許可されない。
- ピクチャパラメータセットは、0のみに等しいnum\_slice\_groups\_minus1を有さなければならない。
- ピクチャパラメータセットは、0のみに等しいredundant\_pic\_cnt\_present\_flagを有さなければならない。
- シーケンスパラメータセットは、0から6のそれぞれの値を含む範囲のbit\_depth\_luma\_minus8を有さなければならない。
- シーケンスパラメータセットは、0から6のそれぞれの値を含む範囲のbit\_depth\_chroma\_minus8を有さなければならない。
- ハイ4:4:4予測プロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイ 4:4:4 予測プロファイルへのビットストリームの適合は、profile\_idc が 244 であることによって規定される。特定のレベルでハイ 4:4:4 予測プロファイルに適合する復号器は、次の条件のいずれかあるいは両方が真である全てのビットストリームを復号できなければならない。

- profile\_idcが77であるかconstraint\_set1\_flagが1であり、かつlevel\_idcおよびconstraint\_set3\_flagの組合せがその指定されたレベルより小さいか等しいレベルを表す。
- profile\_idcが44、100、110、122、または244に等しく、かつlevel\_idcがその指定されたレベルより小さいか等しいレベルを表す。

#### A.2.8 ハイ 10 INTRAプロファイル (High 10 Intra)

ハイ 10 INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- ハイ10プロファイルに対してA.2.5節で規定された全ての制約に従わなければならない。
- 全てのピクチャはIDRピクチャでなければならない。
- シーケンスパラメータセットは、0に等しいnum\_ref\_framesを有さなければならない。
- vui\_parameters\_present\_flagが1、かつ、bitstream\_restriction\_flagが1の時、シーケンスパラメータセットは、0に等しいnum\_reorder\_framesを有さなければならない。
- vui\_parameters\_present\_flagが1、かつ、bitstream\_restriction\_flagが1の時、シーケンスパラメータセットは、0に等しいmax\_dec\_frame\_bufferingを有さなければならない。

- ピクチャタイミングSEIメッセージは、(非VCL NALユニットにより)ビットストリーム中に存在するか、あるいは、本標準に規定されていない他の手段により同等に伝達されているかに関わらず、0に等しい `dpb_output_delay`を有さなければならない。
- ハイ10 INTRAプロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイ 10 INTRA プロファイルへのビットストリームの適合は、110 に等しい `profile_idc` を有し `constraint_set3_flag` が1に等しいことによって規定される。特定のレベルでハイ 10 INTRA プロファイルに適合する復号器は、次の条件の全てが真である全てのビットストリームを復号できなければならない。

- `profile_idc`が100または110に等しい。
- `constraint_set3_flag`が1に等しい。
- `level_idc`がその指定されたレベルより小さいか等しいレベルを表す。

記 `profile_idc`の値100は、ビットストリームがA.2.4小節で規定されたハイプロファイルに適合することを示す。`profile_idc`が100であり、かつ`constraint_set3_flag`が1に等しい時、この事はビットストリームがハイプロファイルに適合し、ハイ10 INTRAプロファイルに対して本小節で規定された制約に適合することを示す。例えば、そのようなビットストリームは、0に等しい`bit_depth_luma_minus8`を有し、0に等しい`bit_depth_chroma_minus8`を有し、ハイプロファイルのMinCR、MaxBRおよびMaxCPBの制約に従い、IDRピクチャのみを含み、0に等しい`num_ref_frames`を有し、0に等しい`dpb_output_delay`を有し、ハイ10 INTRAプロファイルの最大スライスサイズの制約に従う。

8.7節で規定されたデブロックフィルタ処理の操作は、ハイ 10 INTRA プロファイルへの復号器適合には要求されない。

記 これはハイ10 INTRAプロファイルへの復号器適合の要件ではないのだが、8.7節で規定されたデブロックフィルタ処理、もしくは、何らかの類似の事後処理フィルタが実行されるべきである。8.7節で規定されるデブロックフィルタ処理の制御のために符号器により送信されるシンタックス要素は、ハイ10 INTRAプロファイルへの復号器適合においては、補助的な情報とみなされるのみである。しかしながら、ハイ10プロファイル、ハイ4:2:2プロファイルおよびハイ4:4:4予測プロファイルへの復号器適合においては、ハイ10 INTRAプロファイルに適合するビットストリームを復号する時、8.7節で規定されるデブロックフィルタ処理の適用は要求される。

#### A.2.9 ハイ 4:2:2 INTRAプロファイル (High 4:2:2 Intra)

ハイ 4:2:2 INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- ハイ4:2:2プロファイルに対してA.2.6節で規定された全ての制約に従わなければならない。
- 全てのピクチャはIDRピクチャでなければならない。
- シーケンスパラメータセットは、0に等しい`num_ref_frames`を有さなければならない。
- `vui_parameters_present_flag`が1、かつ、`bitstream_restriction_flag`が1の時、シーケンスパラメータセットは、0に等しい`num_reorder_frames`を有さなければならない。
- `vui_parameters_present_flag`が1、かつ、`bitstream_restriction_flag`が1の時、シーケンスパラメータセットは、0に等しい`max_dec_frame_buffering`を有さなければならない。
- ピクチャタイミングSEIメッセージは、(非VCL NALユニットにより)ビットストリーム中に存在するか、あるいは、本標準に規定されていない他の手段により同等に伝達されているかに関わらず、0に等しい `dpb_output_delay`を有しなければならない。
- ハイ4:2:2 INTRAプロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイ 4:2:2 INTRA プロファイルへのビットストリームの適合は、122 に等しい `profile_idc` を有し `constraint_set3_flag` が 1 に等しいことによって規定される。特定のレベルでハイ 4:2:2 INTRA プロファイルに適合する復号器は、次の条件の全てが真である全てのビットストリームを復号できなければならない。

- `profile_idc`が100、110または122に等しい。
- `constraint_set3_flag`が1に等しい。
- `level_idc`がその指定されたレベルより小さいか等しいレベルを表す。

8.7節で規定されたデブロックフィルタ処理の操作は、ハイ 4:2:2 INTRA プロファイルへの復号器適合には要求されない。

記 これはハイ 4:2:2 INTRA プロファイルへの復号器適合の要件ではないのだが、8.7節で規定されたデブロックフィルタ処理、もしくは、何らかの類似の事後処理フィルタが実行されるべきである。8.7節で規定されるデブロックフィルタ処理の制御のために符号器により送信されるシンタックス要素は、ハイ 4:2:2 INTRA プロファイルへの復号器適合においては、補助的な情報とみなされるのみである。しかしながら、ハイ 4:2:2 プロファイルおよびハイ 4:4:4 予測プロファイルへの復号器適合においては、ハイ 4:2:2 INTRA プロファイルに適合するビットストリームを復号する時、8.7節で規定されるデブロックフィルタ処理の適用は要求される。

#### A.2.10 ハイ 4:4:4 INTRA プロファイル (High 4:4:4 Intra)

ハイ 4:4:4 INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- ハイ 4:4:4 予測プロファイルに対してA.2.7節で規定された全ての制約に従わなければならない。
- 全てのピクチャはIDRピクチャでなければならない。
- シーケンスパラメータセットは、0に等しい`num_ref_frames`を有さなければならない。
- `vui_parameters_present_flag`が1、かつ、`bitstream_restriction_flag`が1の時、シーケンスパラメータセットは、0に等しい`num_reorder_frames`を有さなければならない。
- `vui_parameters_present_flag`が1、かつ、`bitstream_restriction_flag`が1の時、シーケンスパラメータセットは、0に等しい`max_dec_frame_buffering`を有さなければならない。
- ピクチャタイミングSEIメッセージは、(非VCL NALユニットにより)ビットストリーム中に存在するか、あるいは、本標準に規定されていない他の手段により同等に伝達されているかに関わらず、0に等しい`dpb_output_delay`を有しなければならない。
- ハイ 4:4:4 INTRA プロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

ハイ 4:4:4 INTRA プロファイルへのビットストリームの適合は、244 に等しい `profile_idc` を有し `constraint_set3_flag` が 1 に等しいことによって規定される。特定のレベルでハイ 4:4:4 INTRA プロファイルに適合する復号器は、次の条件の全てが真である全てのビットストリームを復号できなければならない。

- `profile_idc`が44、100、110、122または244に等しい。
- `constraint_set3_flag`が1に等しい。
- `level_idc`がその指定されたレベルより小さいか等しいレベルを表す。

8.7節で規定されたデブロックフィルタ処理の操作は、ハイ 4:4:4 INTRA プロファイルへの復号器適合には要求されない。

記 これはハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルへの復号器適合の要件ではないのだが、8.7節で規定されたデブロックフィルタ処理、もしくは、何らかの類似の事後処理フィルタが実行されるべきである。8.7節で規定されるデブロックフィルタ処理の制御のために符号器により送信されるシンタックス要素は、ハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルへの復号器適合においては、補助的な情報とみなされるのみである。しかしながら、ハイ4:4:4予測プロファイルへの復号器適合においては、ハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルに適合するビットストリームを復号する時、8.7節で規定されるデブロックフィルタ処理の適用は要求される。

#### A.2.11 CAVLC 4:4:4 INTRAプロファイル (CAVLC 4:4:4 Intra)

CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- ハイ4:4:4 INTRAプロファイルに対してA.2.10小節で規定された全ての制約に従わなければならない。
- ピクチャパラメータセットは、0に等しいentropy\_coding\_mode\_flagを有さなければならない。
- CAVLC 4:4:4 INTRAプロファイルに対してA.3節で規定されたレベルの制約が、満たされなければならない。

CAVLC 4:4:4 INTRA プロファイルへのビットストリームの適合は、profile\_idc が 44 に等しいことによって規定される。特定のレベルで CAVLC 4:4:4 INTRA プロファイルに適合する復号器は、次の条件の全てが真である全てのビットストリームを復号できなければならない。

- profile\_idcが44に等しい。
- level\_idcがその指定されたレベルより小さいか等しいレベルを表す。

8.7節で規定されたデブロックフィルタ処理の操作は、CAVLC 4:4:4 INTRA プロファイルへの復号器適合には要求されない。

記 これはハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルへの復号器適合の要件ではないのだが、8.7節で規定されたデブロックフィルタ処理、もしくは、何らかの類似の事後処理フィルタが実行されるべきである。8.7節で規定されるデブロックフィルタ処理の制御のために符号器により送信されるシンタックス要素は、ハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルへの復号器適合においては、補助的な情報とみなされるのみである。しかしながら、ハイ4:4:4予測プロファイルへの復号器適合においては、ハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルに適合するビットストリームを復号する時、8.7節で規定されるデブロックフィルタ処理の適用は要求される。

### A.3 レベル

この付属資料の制約を表現するために、次が規定される。

- 初めのアクセスユニットをアクセスユニット 0 として、アクセスユニット n を復号順序で n 番目のアクセスユニットとする。
- ピクチャ n を、アクセスユニット n の主符号化ピクチャ、あるいは対応した復号ピクチャとする。

#### A.3.1 基本、メイン、および拡張プロファイルに対する共通のレベル限定

変数 fR は次の通り導出されるものとする。

- もしピクチャ n がフレームであるならば、fR は  $1 \div 172$  に設定される。
- それ以外 (ピクチャ n がフィールドである)、fR は  $1 \div (172 * 2)$  に設定される。

特定のレベルにおいて、基本、メイン、あるいは拡張プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- a) C.1.2 小節で規定されるような、CPB からのアクセスユニット  $n$  ( $n>0$  である) の名目上の除去時刻は、 $t_{r,n}(n)-t_r(n-1)$  が  $\text{Max}(\text{PicSizeInMbs}+\text{MaxMBPS},fR)$  より大きいかまたは等しいという制約を満たす。ここで、 $\text{MaxMBPS}$  はピクチャ  $n-1$  に適用される付表 A-1/JT-H264 で規定された値であり、 $\text{PicSizeInMbs}$  はピクチャ  $n-1$  内のマクロブロック数である。
- b) C.2.2 小節で規定されるような、DPB からの連続したピクチャの出力時刻の差分は、 $\Delta t_{o,dpb}(n) \geq \text{Max}(\text{PicSizeInMbs}+\text{MaxMBPS},fR)$  という制約を満たす。ここで、 $\text{MaxMBPS}$  はピクチャ  $n$  に対する付表 A-1/JT-H264 で規定された値であり、 $\text{PicSizeInMbs}$  はピクチャ  $n$  内のマクロブロック数である。ピクチャ  $n$  は出力ピクチャであり、そのビットストリームの最後の出力ピクチャではないものと仮定される。
- c) アクセスユニット 0 に対する  $\text{NumBytesInNALunit}$  変数の和は、 $384 * (\text{PicSizeInMbs} + \text{MaxMBPS} * (t_r(0) - t_r(n(0)))) \div \text{MinCR}$  より小さいかまたは等しい。ここで、 $\text{MaxMBPS}$  と  $\text{MinCR}$  はピクチャ 0 に適用される付表 A-1/JT-H264 で規定された値であり、 $\text{PicSizeInMbs}$  はピクチャ 0 内のマクロブロック数である。
- d) アクセスユニット  $n$  ( $n>0$  である) に対する  $\text{NumBytesInNALunit}$  変数の和は、 $384 * \text{MaxMBPS} * (t_r(n) - t_r(n-1)) \div \text{MinCR}$  より小さいかまたは等しい。ここで、 $\text{MaxMBPS}$  と  $\text{MinCR}$  はピクチャ  $n$  に適用される付表 A-1/JT-H264 で規定された値である。
- e)  $\text{PicWidthInMbs} * \text{FrameHeightInMbs} \leq \text{MaxFS}$  である。ここで、 $\text{MaxFS}$  は付表 A-1/JT-H264 で規定される。
- f)  $\text{PicWidthInMbs} \leq \text{Sqrt}(\text{MaxFS} * 8)$  である。
- g)  $\text{FrameHeightInMbs} \leq \text{Sqrt}(\text{MaxFS} * 8)$  である。
- h)  $\text{max\_dec\_frame\_buffering} \leq \text{MaxDpbSize}$  である。ここで、 $\text{MaxDpbSize}$  は  $\text{Min}(1024 * \text{MaxDPB} / (\text{PicWidthInMbs} * \text{FrameHeightInMbs} * 384), 16)$  に等しく、 $\text{MaxDPB}$  は付表 A-1/JT-H264 で 1024 バイト単位で与えられる。
- i) VCL HRD パラメータに関しては、 $\text{SchedSelIdx}$  の少なくとも 1 つの値に対して、 $\text{BitRate}[\text{SchedSelIdx}] \leq 1000 * \text{MaxBR}$  かつ  $\text{CpbSize}[\text{SchedSelIdx}] \leq 1000 * \text{MaxCPB}$  である。ここで、 $\text{vcl\_hrd\_parameters\_present\_flag}$  が 1 に等しい場合、 $\text{BitRate}[\text{SchedSelIdx}]$  は等式 E-37 で与えられ、 $\text{CpbSize}[\text{SchedSelIdx}]$  は等式 E-38 で与えられる。 $\text{MaxBR}$  と  $\text{MaxCPB}$  は、付表 A-1/JT-H264 でそれぞれ 1000 ビット/秒および 1000 ビット単位で規定される。ビットストリームは、0 から  $\text{cpb\_cnt\_minus1}$  のそれぞれの値を含む範囲の  $\text{SchedSelIdx}$  の少なくとも 1 つの値に対して、これらの条件を満たさなければならない。
- j) NAL HRD パラメータに関しては、 $\text{SchedSelIdx}$  の少なくとも 1 つの値に対して、 $\text{BitRate}[\text{SchedSelIdx}] \leq 1200 * \text{MaxBR}$  かつ  $\text{CpbSize}[\text{SchedSelIdx}] \leq 1200 * \text{MaxCPB}$  である。ここで、 $\text{nal\_hrd\_parameters\_present\_flag}$  が 1 に等しい場合、 $\text{BitRate}[\text{SchedSelIdx}]$  は等式 E-37 で与えられ、 $\text{CpbSize}[\text{SchedSelIdx}]$  は等式 E-38 で与えられる。 $\text{MaxBR}$  と  $\text{MaxCPB}$  は、付表 A-1/JT-H264 でそれぞれ 1200 ビット/秒および 1200 ビット単位で規定される。ビットストリームは、0 から  $\text{cpb\_cnt\_minus1}$  の範囲の  $\text{SchedSelIdx}$  の少なくとも 1 つの値に対して、これらの条件を満たさなければならない。
- k) 輝度動きベクトルに対する垂直方向の動きベクトル成分の範囲は、輝度フレームサンプル単位で

MaxVmvR を超えない。ここで、MaxVmvR は付表 A-1/JT-H264 で規定される。

記1 chroma\_format\_idcが1に等しく、かつ現マクロブロックがフィールドマクロブロックである時、色差動きベクトルに対する動きベクトル成分の範囲は、8.4.1.4小節で規定される色差動きベクトルを導出する方法のため、輝度フレームサンプル単位でMaxVmvRを超えても良い。

- l) 水平方向の動きベクトルの範囲は、輝度サンプル単位で-2048 から 2047.75 のそれぞれの値を含む範囲を超えない。
- m) 復号順序で連続する2つのマクロブロックあたりの動きベクトルの数（あるスライスの最終マクロブロック、および復号順序で次のスライスの先頭マクロブロックからの合計にも適用され、そして特に、あるピクチャの最終スライスの最終マクロブロック、および復号順序で次のピクチャの先頭スライスの先頭マクロブロックからの合計にも適用される）は、MaxMvsPer2Mbを超えない。ここで、MaxMvsPer2Mb は付表 A-1/JT-H264 で規定される。各々のマクロブロックに対する動きベクトルの数は、そのマクロブロックに対する INTRA または INTER 予測処理終了の後の変数 MvCnt の値である。
- n) 任意のマクロブロックに対する macroblock\_layer( )データのビット数は、3200 よりも大きくない。entropy\_coding\_mode\_flag によって、macroblock\_layer( )データのビットは、次の通り数えられる。
  - もし entropy\_coding\_mode\_flag が 0 に等しいならば、macroblock\_layer( )データのビット数は、あるマクロブロックに対する macroblock\_layer( )シンタックス構造内のビット数によって与えられる。
  - それ以外（entropy\_coding\_mode\_flag が 1 に等しい）、あるマクロブロックに対する macroblock\_layer( )データのビット数は、そのマクロブロックに関連する macroblock\_layer( )を構文解析する時に read\_bits(1)が9.3.3.2.2 小節および9.3.3.2.3 小節で呼ばれる回数によって与えられる。

付表 A-1/JT-H264 は、各レベルの限定を規定する。付表 A-1/JT-H264 中の“-”でマークされたエントリは、対応する限定の欠如を表している。レベル能力比較の目的においては、他のレベルよりも付表 A-1 の上端（下端）の行に近い位置にあれば、他のレベルより低い（高い）レベルであるとみなされなければならない。

ビットストリームが適合するレベルは、シンタックス要素 level\_idc および constraint\_set3\_flag によって次の通り指示されなければならない。

- もし level\_idc が 11 に等しく、かつ constraint\_set3\_flag が 1 に等しいならば、指示されるレベルはレベル 1b である。
- それ以外（level\_idc が 11 に等しくない、または constraint\_set3\_flag が 1 に等しくない）、level\_idc は付表 A-1/JT-H264 で規定されるレベル番号の 10 倍の値に等しく設定されていなければならない、かつ constraint\_set3\_flag は 0 に等しく設定されていなければならない。

付表A-1/JT-H264 レベルの限定  
(ITU-T H.264)

レベル番号	最大マクロブロック処理レート MaxMBPS(MB/秒)	最大フレームサイズ MaxFS(MB)	最大復号ピクチャバッファサイズ MaxDPB(4:2:0 において 1024 バイト)	最大ビデオビットレート MaxBR(1000 ビット/秒、1200 ビット/秒、cpbBrVclFactor ビット/秒、または cpbBrNalFactor ビット/秒)	最大 CPB サイズ MaxCPB(1000 ビット、1200 ビット、cpbBrVclFactor ビ ット、または cpbBrNalFactor ビット)	垂直 MV 成分範囲 MaxVmvR(輝度フレームサンプル)	最小圧縮比 MinCR	2つの連続した MB あたりの最大動きベクトル数 MaxMvsPer2Mb
1	1 485	99	148.5	64	175	[-64,+63.75]	2	-
1b	1 485	99	148.5	128	350	[-64,+63.75]	2	-
1.1	3 000	396	337.5	192	500	[-128,+127.7 5]	2	-
1.2	6 000	396	891.0	384	1 000	[-128,+127.7 5]	2	-
1.3	11 880	396	891.0	768	2 000	[-128,+127.7 5]	2	-
2	11 880	396	891.0	2 000	2 000	[-128,+127.7 5]	2	-
2.1	19 800	792	1 782.0	4 000	4 000	[-256,+255.7 5]	2	-
2.2	20 250	1 620	3 037.5	4 000	4 000	[-256,+255.7 5]	2	-
3	40 500	1 620	3 037.5	10 000	10 000	[-256,+255.7 5]	2	32
3.1	108 000	3 600	6 750.0	14 000	14 000	[-512,+511.7 5]	4	16
3.2	216 000	5 120	7 680.0	20 000	20 000	[-512,+511.7 5]	4	16

4	245 760	8 192	12 288.0	20 000	25 000	[-512,+511.7 5]	4	16
4.1	245 760	8 192	12 288.0	50 000	62 500	[-512,+511.7 5]	2	16
4.2	522 240	8 704	13 056.0	50 000	62 500	[-512,+511.7 5]	2	16
5	589 824	22 080	41 400.0	135 000	135 000	[-512,+511.7 5]	2	16
5.1	983 040	36 864	69 120.0	240 000	240 000	[-512,+511.7 5]	2	16

付表 A-1/JT-H264 の中の非整数レベル番号を備えたレベルは“中間レベル”と呼ばれる。

記2 全てのレベルは同じステータスを持っているが、いくつかのアプリケーションは整数に番号付けられたレベルだけを使用することを選択しても良い。

A.3.4 参考小節では、いくつかのピクチャフォーマット例に対して、これらの限定のフレームレートへの影響を示す。

#### A.3.2 ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、およびCAVLC 4:4:4 INTRAプロファイルに対する共通のレベル限定

変数  $fR$  は次の通り導出されるものとする。

- もしピクチャ  $n$  がフレームであるならば、 $fR$  は  $1 \div 172$  に設定される。
- それ以外（ピクチャ  $n$  がフィールドである）、 $fR$  は  $1 \div (172 * 2)$  に設定される。

特定のレベルにおいて、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- a) C.1.2 小節で規定されるような、CPB からのアクセスユニット  $n$  ( $n > 0$  である) の名目上の除去時刻は、 $t_{r,n}(n) - t_r(n-1)$  が  $\text{Max}(\text{PicSizeInMbs} \div \text{MaxMBPS}, fR)$  より大きいかまたは等しいという制約を満たす。ここで、 $\text{MaxMBPS}$  はピクチャ  $n-1$  に適用される付表 A-1/JT-H264 で規定された値であり、 $\text{PicSizeInMbs}$  はピクチャ  $n-1$  内のマクロブロック数である。
- b) C.2.2 小節で規定されるような、DPB からの連続したピクチャの出力時刻の差分は、 $\Delta t_{o,dpb}(n) \geq \text{Max}(\text{PicSizeInMbs} \div \text{MaxMBPS}, fR)$  という制約を満たす。ここで、 $\text{MaxMBPS}$  はピクチャ  $n$  に対する付表 A-1/JT-H264 で規定された値であり、 $\text{PicSizeInMbs}$  はピクチャ  $n$  内のマクロブロック数である。ピクチャ  $n$  は出力ピクチャであり、そのビットストリームの最後の出力ピクチャではないものと仮定される。
- c)  $\text{PicWidthInMbs} * \text{FrameHeightInMbs} \leq \text{MaxFS}$  である。ここで、 $\text{MaxFS}$  は付表 A-1/JT-H264 で規定される。

- d)  $\text{PicWidthInMbs} \leq \text{Sqrt}(\text{MaxFS} * 8)$ である。
- e)  $\text{FrameHeightInMbs} \leq \text{Sqrt}(\text{MaxFS} * 8)$ である。
- f)  $\text{max\_dec\_frame\_buffering} \leq \text{MaxDpbSize}$  である。ここで、 $\text{MaxDpbSize}$  は  $\text{Min}(1024 * \text{MaxDPB} / (\text{PicWidthInMbs} * \text{FrameHeightInMbs} * 384), 16)$  に等しく、 $\text{MaxDPB}$  は付表 A-1/JT-H264 で規定される。
- g) 垂直方向の動きベクトル成分の範囲は、輝度フレームサンプル単位で  $\text{MaxVmvR}$  を超えない。ここで、 $\text{MaxVmvR}$  は付表 A-1/JT-H264 で規定される。
- h) 水平方向の動きベクトルの範囲は、輝度サンプル単位で -2048 から 2047.75 のそれぞれの値を含む範囲を超えない。
- i) 復号順序で連続する 2 つのマクロブロックあたりの動きベクトルの数 (あるスライスの最終マクロブロック、および復号順序で次のスライスの先頭マクロブロックからの合計にも適用される) は、 $\text{MaxMvsPer2Mb}$  を超えない。ここで、 $\text{MaxMvsPer2Mb}$  は付表 A-1/JT-H264 で規定される。各々のマクロブロックに対する動きベクトルの数は、そのマクロブロックに対する INTRA または INTER 予測処理終了の後の変数  $\text{MvCnt}$  の値である。
- j) 任意のマクロブロックに対する  $\text{macroblock\_layer}()$  データのビット数は、 $128 + \text{RawMbBits}$  よりも大きくない。 $\text{entropy\_coding\_mode\_flag}$  によって、 $\text{macroblock\_layer}()$  データのビットは、次の通り数えられる。
  - もし  $\text{entropy\_coding\_mode\_flag}$  が 0 に等しいならば、 $\text{macroblock\_layer}()$  データのビット数は、あるマクロブロックに対する  $\text{macroblock\_layer}()$  シンタックス構造内のビット数によって与えられる。
  - それ以外 ( $\text{entropy\_coding\_mode\_flag}$  が 1 に等しい)、あるマクロブロックに対する  $\text{macroblock\_layer}()$  データのビット数は、そのマクロブロックに関連する  $\text{macroblock\_layer}()$  を構文解析する時に  $\text{read\_bits}(1)$  が 9.3.3.2.2 小小小節および 9.3.3.2.3 小小小節で呼ばれる回数によって与えられる。

付表 A-1/JT-H264 は、各レベルの限定を規定する。付表 A-1/JT-H264 中の“-”でマークされたエントリは、対応する限定の欠如を表している。これらのプロファイルに対する付表 A-1/JT-H264 の  $\text{MinCR}$  パラメータ列の使用は、A.3.3 小節に規定される。

ビットストリームが適合するレベルは、シンタックス要素  $\text{level\_idc}$  によって次の通り指示されなければならない。

- もし  $\text{level\_idc}$  が 9 に等しいならば、指示されるレベルはレベル 1b である。
- それ以外 ( $\text{level\_idc}$  が 9 に等しくない)、 $\text{level\_idc}$  は付表 A-1/JT-H264 で規定されるレベル番号の 10 倍の値に等しく設定されていなければならない。

### A.3.3 プロファイルに特有のレベル限定

- a) メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームにおいては、アクセスユニット 0 の除去時刻は、ピクチャ 0 内のスライス数は

$(PicSizeInMbs+MaxMBPS*(t_r(0)-t_{r,n}(0)))\div SliceRate$  より小さいかまたは等しいという制約を満たさなければならない。ここで、SliceRate はピクチャ 0 に適用される付表 A-4/JT-H264 で規定された値である。

- b) メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームにおいては、アクセスユニット n と n-1 (n>0 である) との連続する除去時刻の間の差分は、ピクチャ n 内のスライス数が  $MaxMBPS*(t_r(n)-t_r(n-1))\div SliceRate$  より小さいかまたは等しいという制約を満たさなければならない。ここで、SliceRate はピクチャ n に適用される付表 A-4/JT-H264 で規定された値である。
- c) メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームにおいては、付表 A-4/JT-H264 で規定されたレベルで、シーケンスパラメータセットは 1 に等しい direct\_8x8\_inference\_flag を有さなければならない。

記1 基本プロファイル (A.2.1小節で規定される) はBスライスタイプを許可しないので、direct\_8x8\_inference\_flagは基本プロファイルには関連がない。また、拡張プロファイル (A.2.3小節で規定される) の全てのレベルで、direct\_8x8\_inference\_flagは1に等しい。

- d) メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、CAVLC 4:4:4 INTRA、または拡張プロファイルに適合するビットストリームにおいては、メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに対して付表 A-4/JT-H264 で規定されたレベル、および拡張プロファイルに対して付表 A-5/JT-H264 で規定されたレベルで、シーケンスパラメータセットは 1 に等しい frame\_mbs\_only\_flag を有さなければならない。

記2 基本プロファイル (A.2.1小節で規定される) の全てのレベルに対して、frame\_mbs\_only\_flag は1と等しい。

- e) メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、CAVLC 4:4:4 INTRA、または拡張プロファイルに適合するビットストリームにおいては、メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに対する付表 A-4/JT-H264、および拡張プロファイルに対する付表 A-5/JT-H264 内の  $8\times 8$  として示された MinLumaBiPredSize に対応するレベルで、B マクロブロック内の sub\_mb\_type の値は B\_Bi\_8x4、B\_Bi\_4x8 あるいは B\_Bi\_4x4 と等しくなっていない。
- f) 基本または拡張プロファイルに適合するビットストリームにおいては、P\_8x8、P\_8x8ref0 または B\_8x8 に等しい mb\_type で符号化されたマクロブロックでは各  $8\times 8$  サブマクロブロックに対する単一参照ピクチャリスト (参照ピクチャリスト 0 または参照ピクチャリスト 1) で予測された輝度サンプル配列を生成するために使用される 8.4.2.2.1 小小小節で規定された処理の全ての起動に対して  $(xInt_{max}-xInt_{min}+6)*(yInt_{max}-yInt_{min}+6)\leq MaxSubMbRectSize$ 。ここで、 $NumSubMbPart(sub\_mb\_type)>1$  であり、MaxSubMbRectSize は基本プロファイルに対する付表 A-3/JT-H264、および拡張プロファイルに対する付表 A-5/JT-H264 で規定され、さらに、

- $xInt_{min}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $xInt_t$  の最小値
- $xInt_{max}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $xInt_t$  の最大値
- $yInt_{min}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $yInt_t$  の最小値

- $yInt_{\max}$ は、サブマクロブロックの全ての輝度サンプル予測中の、 $yInt_t$ の最大値
- g) ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームにおいては、VCL HRD パラメータに関しては、SchedSelIdx の少なくとも 1 つの値に対して、 $BitRate[SchedSelIdx] \leq cpbBrVclFactor * MaxBR$  かつ  $CpbSize[SchedSelIdx] \leq cpbBrVclFactor * MaxCPB$  である。ここで、vcl\_hrd\_parameters\_present\_flag が 1 に等しい場合、cpbBrVclFactor は付表 A-2/JT-H264 で規定され、 $BitRate[SchedSelIdx]$  は等式 E-37 で規定され、また  $CpbSize[SchedSelIdx]$  は等式 E-38 で規定される。MaxBR および MaxCPB は、それぞれ cpbBrVclFactor ビット/秒および cpbBrVclFactor ビットを単位として付表 A-1/JT-H264 で規定される。ビットストリームは、0 から cpb\_cnt\_minus1 のそれぞれの値を含む範囲の SchedSelIdx の少なくとも 1 つの値に対して、これらの条件を満たさなければならない。
- h) ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームにおいては、NAL HRD パラメータに関しては、SchedSelIdx の少なくとも 1 つの値に対して、 $BitRate[SchedSelIdx] \leq cpbBrNalFactor * MaxBR$  かつ  $CpbSize[SchedSelIdx] \leq cpbBrNalFactor * MaxCPB$  である。ここで、nal\_hrd\_parameters\_present\_flag が 1 に等しい場合、cpbBrNalFactor は付表 A-2/JT-H264 で規定され、 $BitRate[SchedSelIdx]$  は等式 E-37 で規定され、また  $CpbSize[SchedSelIdx]$  は等式 E-38 で規定される。MaxBR および MaxCPB は、それぞれ cpbBrNalFactor ビット/秒および cpbBrNalFactor ビットを単位として付表 A-1/JT-H264 で規定される。ビットストリームは、0 から cpb\_cnt\_minus1 の範囲の SchedSelIdx の少なくとも 1 つの値に対して、これらの条件を満たさなければならない。
- i) ハイプロファイルに適合するビットストリームにおいては、アクセスユニット 0 に対する変数 NumBytesInNALunit の和は、 $384 * (PicSizeInMbs + MaxMBPS * (t_r(0) - t_{rn}(0))) \div MinCR$  より小さいか等しい。ここで、MaxMBPS および MinCR は、ピクチャ 0 に適用される付表 A-1/JT-H264 で規定される値であり、PicSizeInMbs はピクチャ 0 におけるマクロブロックの数である。
- 記2 MinCRに関係するこのような限定は、ハイ10、ハイ 4:2:2、ハイ4:4:4 予測、ハイ10 INTRA、ハイ4:2:2 INTRA、ハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルに適合するビットストリームに対しては課せられない。
- j) ハイプロファイルに適合するビットストリームにおいては、アクセスユニット n (但し n>0) に対する変数 NumBytesInNALunit の和は、 $384 * MaxMBPS * (t_r(n) - t_r(n-1)) \div MinCR$  より小さいか等しい。ここで、MaxMBPS および MinCR は、ピクチャ n に適用される付表 A-1/JT-H264 で規定される値である。
- 記3 MinCRに関係するこのような限定は、ハイ10、ハイ 4:2:2、ハイ4:4:4 予測、ハイ10 INTRA、ハイ4:2:2 INTRA、ハイ4:4:4 INTRAおよびCAVLC 4:4:4 INTRAプロファイルに適合するビットストリームに対しては課せられない。
- k) ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA または CAVLC 4:4:4 に適合するビットストリームにおいては、PicSizeInMbs が 1620 よりも大きい時、どの符号化スライスのマクロブロック数も MaxFS/4 を超えてはならない。ここで、MaxFS は付表 A-1/JT-H264 で規定される。

記4 上述のi)、j)、およびk)項は、ハイ4:4:4 予測、ハイ10 INTRA、ハイ4:2:2 INTRA、ハイ4:4:4 INTRA およびCAVLC 4:4:4 INTRAプロファイルの規定の目的でここに含まれる。これらの項のハイ、ハイ

10、およびハイ 4:2:2プロファイルに関連する側面は、完全性のためにここに含まれ、これらのプロファイルの意図された規定の修正を反映する。

付表A-2/JT-H264 cpbBrVclFactorとcpbBrNalFactorの規定  
(ITU-T H.264)

プロファイル	cpbBrVclFactor	cpbBrNalFactor
ハイ	1 250	1 500
ハイ 10 ハイ 10 INTRA	3 000	3 600
ハイ 4:2:2 ハイ 4:2:2 INTRA	4 000	4 800
ハイ 4:4:4 予測 ハイ 4:4:4 INTRA CAVLC 4:4:4 INTRA	4 000	4 800

#### A.3.3.1 基本プロファイルのレベル限定

付表 A-3/JT-H264 は、基本プロファイルに適合するビットストリームに特有のレベル毎の限定を規定する。付表 A-3/JT-H264 中の“-”でマークされたエントリは、対応する限定の欠如を表す。

付表A-3/JT-H264 基本プロファイルのレベル限定  
(ITU-T H.264)

レベル番号	MaxSubMbRectSize
1	576
1b	576
1.1	576
1.2	576
1.3	576
2	576
2.1	576
2.2	576
3	576
3.1	-
3.2	-

4	-
4.1	-
4.2	-
5	-
5.1	-

**A.3.3.2 メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、およびCAVLC 4:4:4 INTRAプロファイルのレベル限定**

付表 A-4/JT-H264 は、メイン、ハイ、ハイ 10、ハイ 4:2:2、ハイ 4:4:4 予測、ハイ 10 INTRA、ハイ 4:2:2 INTRA、ハイ 4:4:4 INTRA、または CAVLC 4:4:4 INTRA プロファイルに適合するビットストリームに特有のレベル毎の限定を規定する。付表 A-4/JT-H264 中の“-”でマークされたエントリは、対応する限定の欠如を表す。

**付表A-4/JT-H264 メイン、ハイ、ハイ10、ハイ4:2:2、ハイ4:4:4予測、ハイ10 INTRA、ハイ4:2:2 INTRA、ハイ4:4:4 INTRA、およびCAVLC 4:4:4 INTRAプロファイルのレベル限定(ITU-T H.264)**

レベル番号	SliceRate	MinLumaBiPredSize	direct_8x8_inference_flag	frame_mbs_only_flag
1	-	-	-	1
1b	-	-	-	1
1.1	-	-	-	1
1.2	-	-	-	1
1.3	-	-	-	1
2	-	-	-	1
2.1	-	-	-	-
2.2	-	-	-	-
3	22	-	1	-
3.1	60	8×8	1	-
3.2	60	8×8	1	-
4	60	8×8	1	-
4.1	24	8×8	1	-
4.2	24	8×8	1	1
5	24	8×8	1	1
5.1	24	8×8	1	1

**A.3.3.3 拡張プロファイルのレベル限定**

付表 A-5/JT-H264 は、拡張プロファイルに適合するビットストリームに特有のレベル毎の限定を規定す

る。付表 A-5/JT-H264 中の “-” でマークされたエントリは、対応する限定の欠如を表す。

付表A-5/JT-H264 拡張プロファイルのレベル限定  
(ITU-T H.264)

レベル番号	MaxSubMbRectSize	MinLumaBiPredSize	frame_mbs_only_flag
1	576	-	1
1b	576	-	1
1.1	576	-	1
1.2	576	-	1
1.3	576	-	1
2	576	-	1
2.1	576	-	-
2.2	576	-	-
3	576	-	-
3.1	-	8×8	-
3.2	-	8×8	-
4	-	8×8	-
4.1	-	8×8	-
4.2	-	8×8	1
5	-	8×8	1
5.1	-	8×8	1

#### A.3.4 フレームレートに対するレベル限定の影響（参考）

この小節は、本標準の必須部分を形成しない。

付表A-6/JT-H264 フレームサイズの例に対する最大フレームレート (フレーム/秒)

(ITU-T H.264)

レベル					1	1b	1.1	1.2	1.3	2	2.1
最大フレームサイズ (マクロブロック):					99	99	396	396	396	396	792
最大マクロブロック/秒:					1 485	1 485	3 000	6 000	11 880	11 880	19 800
最大フレームサイズ (サンプル):					25 344	25 344	101 376	101 376	101 376	101 376	202 752
最大サンプル/秒:					380 160	380 160	768 000	1 536 000	3 041 280	3 041 280	5 068 800
フォーマット	輝度 幅	輝度 高さ	MB 合計	輝度 サンプル							
SQCIF	128	96	48	12 288	30.9	30.9	62.5	125.0	172.0	172.0	172.0
QCIF	176	144	99	25 344	15.0	15.0	30.3	60.6	120.0	120.0	172.0
QVGA	320	240	300	76 800	-	-	10.0	20.0	39.6	39.6	66.0
525 SIF	352	240	330	84 480	-	-	9.1	18.2	36.0	36.0	60.0
CIF	352	288	396	101 376	-	-	7.6	15.2	30.0	30.0	50.0
525 HHR	352	480	660	168 960	-	-	-	-	-	-	30.0
625 HHR	352	576	792	202 752	-	-	-	-	-	-	25.0
VGA	640	480	1 200	307 200	-	-	-	-	-	-	-
525 4SIF	704	480	1 320	337 920	-	-	-	-	-	-	-
525 SD	720	480	1 350	345 600	-	-	-	-	-	-	-
4CIF	704	576	1 584	405 504	-	-	-	-	-	-	-
625 SD	720	576	1 620	414 720	-	-	-	-	-	-	-
SVGA	800	600	1 900	486 400	-	-	-	-	-	-	-
XGA	1024	768	3 072	786 432	-	-	-	-	-	-	-
720p HD	1280	720	3 600	921 600	-	-	-	-	-	-	-
4VGA	1280	960	4 800	1 228 800	-	-	-	-	-	-	-
SXGA	1280	1024	5 120	1 310 720	-	-	-	-	-	-	-
525 16SIF	1408	960	5 280	1 351 680	-	-	-	-	-	-	-
16CIF	1408	1152	6 336	1 622 016	-	-	-	-	-	-	-
4SVGA	1600	1200	7 500	1 920 000	-	-	-	-	-	-	-
1080 HD	1920	1088	8 160	2 088 960	-	-	-	-	-	-	-
2Kx1K	2048	1024	8 192	2 097 152	-	-	-	-	-	-	-
2Kx1080	2048	1088	8 704	2 228 224	-	-	-	-	-	-	-
4XGA	2048	1536	12 288	3 145 728	-	-	-	-	-	-	-
16VGA	2560	1920	19 200	4 915 200	-	-	-	-	-	-	-
3616x1536 (2.35:1)	3616	1536	21 696	5 554 176	-	-	-	-	-	-	-
3672x1536 (2.39:1)	3680	1536	22 080	5 652 480	-	-	-	-	-	-	-
4Kx2K	4096	2048	32 768	8 388 608	-	-	-	-	-	-	-
4096x2304 (16:9)	4096	2304	36 864	9 437 184	-	-	-	-	-	-	-

付表A-6/JT-H264(つづき) フレームサイズの例に対する最大フレームレート (フレーム/秒)

(ITU-T H.264)

レベル					2.2	3	3.1	3.2	4	4.1	4.2
最大フレームサイズ (マクロブロック):					1 620	1 620	3 600	5 120	8 192	8 192	8 704
最大マクロブロック/秒:					20 250	40 500	108 000	216 000	245 760	245 760	522 240
最大フレームサイズ (サンプル):					414 720	414 720	921 600	1 310 720	2 097 152	2 097 152	2 228 224
最大サンプル/秒:					5 184 000	10 368 000	27 648 000	55 296 000	62 914 560	62 914 560	133 693 440
フォーマット	輝度 幅	輝度 高さ	MB 合計	輝度 サンプル							
SQCIF	128	96	48	12 288	172.0	172.0	172.0	172.0	172.0	172.0	172.0
QCIF	176	144	99	25 344	172.0	172.0	172.0	172.0	172.0	172.0	172.0
QVGA	320	240	300	76 800	67.5	135.0	172.0	172.0	172.0	172.0	172.0
525 SIF	352	240	330	84 480	61.4	122.7	172.0	172.0	172.0	172.0	172.0
CIF	352	288	396	101 376	51.1	102.3	172.0	172.0	172.0	172.0	172.0
525 HHR	352	480	660	168 960	30.7	61.4	163.6	172.0	172.0	172.0	172.0
625 HHR	352	576	792	202 752	25.6	51.1	136.4	172.0	172.0	172.0	172.0
VGA	640	480	1 200	307 200	16.9	33.8	90.0	172.0	172.0	172.0	172.0
525 4SIF	704	480	1 320	337 920	15.3	30.7	81.8	163.6	172.0	172.0	172.0
525 SD	720	480	1 350	345 600	15.0	30.0	80.0	160.0	172.0	172.0	172.0
4CIF	704	576	1 584	405 504	12.8	25.6	68.2	136.4	155.2	155.2	172.0
625 SD	720	576	1 620	414 720	12.5	25.0	66.7	133.3	151.7	151.7	172.0
SVGA	800	600	1 900	486 400	-	-	56.8	113.7	129.3	129.3	172.0
XGA	1024	768	3 072	786 432	-	-	35.2	70.3	80.0	80.0	172.0
720p HD	1280	720	3 600	921 600	-	-	30.0	60.0	68.3	68.3	145.1
4VGA	1280	960	4 800	1 228 800	-	-	-	45.0	51.2	51.2	108.8
SXGA	1280	1024	5 120	1 310 720	-	-	-	42.2	48.0	48.0	102.0
525 16SIF	1408	960	5 280	1 351 680	-	-	-	-	46.5	46.5	98.9
16CIF	1408	1152	6 336	1 622 016	-	-	-	-	38.8	38.8	82.4
4SVGA	1600	1200	7 500	1 920 000	-	-	-	-	32.8	32.8	69.6
1080 HD	1920	1088	8 160	2 088 960	-	-	-	-	30.1	30.1	64.0
2Kx1K	2048	1024	8 192	2 097 152	-	-	-	-	30.0	30.0	63.8
2Kx1080	2048	1088	8 704	2 228 224	-	-	-	-	-	-	60.0
4XGA	2048	1536	12 288	3 145 728	-	-	-	-	-	-	-
16VGA	2560	1920	19 200	4 915 200	-	-	-	-	-	-	-
3616x1536 (2.35:1)	3616	1536	21 696	5 554 176	-	-	-	-	-	-	-
3672x1536 (2.39:1)	3680	1536	22 080	5 652 480	-	-	-	-	-	-	-
4Kx2K	4096	2048	32 768	8 388 608	-	-	-	-	-	-	-
4096x2304 (16:9)	4096	2304	36 864	9 437 184	-	-	-	-	-	-	-

付表A-6/JT-H264(完) フレームサイズの例に対する最大フレームレート (フレーム/秒)  
(ITU-T H.264)

レベル					5	5.1
最大フレームサイズ (マクロブロック):					22 080	36 864
最大マクロブロック/秒:					589 824	983 040
最大フレームサイズ (サンプル):					5 652 480	9 437 184
最大サンプル/秒:					150 994 944	251 658 240
フォーマット	輝度 幅	輝度 高さ	MB 合計	輝度 サンプル		
SQCIF	128	96	48	12 288	172.0	172.0
QCIF	176	144	99	25 344	172.0	172.0
QVGA	320	240	300	76 800	172.0	172.0
525 SIF	352	240	330	84 480	172.0	172.0
CIF	352	288	396	101 376	172.0	172.0
525 HHR	352	480	660	168 960	172.0	172.0
625 HHR	352	576	792	202 752	172.0	172.0
VGA	640	480	1 200	307 200	172.0	172.0
525 4SIF	704	480	1 320	337 920	172.0	172.0
525 SD	720	480	1 350	345 600	172.0	172.0
4CIF	704	576	1 584	405 504	172.0	172.0
625 SD	720	576	1 620	414 720	172.0	172.0
SVGA	800	600	1 900	486 400	172.0	172.0
XGA	1024	768	3 072	786 432	172.0	172.0
720p HD	1280	720	3 600	921 600	163.8	172.0
4VGA	1280	960	4 800	1 228 800	122.9	172.0
SXGA	1280	1024	5 120	1 310 720	115.2	172.0
525 16SIF	1408	960	5 280	1 351 680	111.7	172.0
16CIF	1408	1152	6 336	1 622 016	93.1	155.2
4SVGA	1600	1200	7 500	1 920 000	78.6	131.1
1080 HD	1920	1088	8 160	2 088 960	72.3	120.5
2Kx1K	2048	1024	8 192	2 097 152	72.0	120.0
2Kx1080	2048	1088	8 704	2 228 224	67.8	112.9
4XGA	2048	1536	12 288	3 145 728	48.0	80.0
16VGA	2560	1920	19 200	4 915 200	30.7	51.2
3616x1536 (2.35:1)	3616	1536	21 696	5 554 176	27.2	45.3
3672x1536 (2.39:1)	3680	1536	22 080	5 652 480	26.7	44.5
4Kx2K	4096	2048	32 768	8 388 608	-	30.0
4096x2304 (16:9)	4096	2304	36 864	9 437 184	-	26.7

次が注意されるべきである。

- 本標準は可変フレームサイズの規定である。付表 A-6/JT-H264 中の特定のフレームサイズは説明用として例示しただけである。
- 付表 A-6/JT-H264 で使用されるように、“525”は（およそ 480 ラインが可視ピクチャ範囲を含む）525 アナログ走査線を使用した環境に対する典型的な使用について言及する。また、“625”は（およそ 576 ラインが可視ピクチャ領域を含む）625 アナログ走査線を使用した環境について言及する。
- XGA は XVGA、4SVGA は UXGA、16XGA は 4Kx3K、CIF は 625 SIF、625 HHR は 2CIF およびハーフ 625 D-1 およびハーフ 625 ITU-R BT.601、525 SD は 525 D-1 および 525 ITU-R BT.601、625 SD は 625 D-1 および 625 ITU-R BT.601 としても知られる。
- 与えられたフレームレートはプログレッシブ走査モード用に対して正しい。フレームレートはまた、フレームの高さが 32 で割り切れる場合のインタレースビデオ符号用に対しても正しい。

## 付属資料B

### バイトストリームフォーマット

（この付属資料は、本標準の必須部分を形成する）

この付属資料は、NAL ユニットストリームのうちのいくらかあるいは全てを、順序付けられたバイトまたはビットのストリームとして配達するアプリケーションでの使用のため規定されるバイトストリームフォーマットのシンタックスと意味を規定する。この中では、TTC 標準 JT-H222.0/ITU-T 勧告 H.222.0/ISO/IEC 13818-1 システムあるいは TTC 標準 JT-H320/ITU-T 勧告 H.320 システムのように、データ中のパターンから NAL ユニット境界の位置が識別可能である必要がある。ビット指向の配達のために、バイトストリームフォーマットにおけるビット順序が、第 1 バイトの MSB で始まり、第 1 バイトの LSB に移り、第 2 バイトの MSB がこれに続く、等に規定される。

バイトストリームフォーマットは、バイトストリーム NAL ユニットシンタックス構造のシーケンスから成る。各バイトストリーム NAL ユニットシンタックス構造は、1 つの `nal_unit` (`NumBytesInNALunit`) シンタックス構造が後続する 1 つのスタートコードプリフィックスを含んでいる。それは、1 つの追加 `zero_byte` シンタックス要素を含んでも良い（そして、ある状況下では含まなければならない）。それは、1 つ以上の追加 `trailing_zero_8bits` シンタックス要素もまた含んでもよい。ビットストリーム中の最初のバイトストリーム NAL ユニットである場合、それは、1 つ以上の追加 `leading_zero_8bits` シンタックス要素もまた含んでもよい。

#### B.1 バイトストリームNALユニットのシンタックスと意味

##### B.1.1 バイトストリームNALユニットシンタックス

	C	記述子
byte_stream_nal_unit( NumBytesInNALunit ) {		
while( next_bits( 24 ) != 0x000001 && next_bits( 32 ) != 0x00000001 )		
<b>leading_zero_8bits</b> /* 0x00に等しい */		f(8)
if( next_bits( 24 ) != 0x000001 )		
<b>zero_byte</b> /* 0x00に等しい */		f(8)
<b>start_code_prefix_one_3bytes</b> /* 0x000001に等しい */		f(24)
nal_unit( NumBytesInNALunit )		
while( more_data_in_byte_stream() && next_bits( 24 ) != 0x000001 && next_bits( 32 ) != 0x00000001 )		
<b>trailing_zero_8bits</b> /* 0x00に等しい */		f(8)
}		

### B.1.2 バイトストリームNALユニットの意味

バイトストリーム中のバイトストリーム NAL ユニットの順序は、バイトストリーム NAL ユニットに含まれる NAL ユニットの復号順序に従わなければならない (7.4.1.2 小小節参照)。各バイトストリーム NAL ユニットの内容は、バイトストリーム NAL ユニットに含まれる NAL ユニットと同じアクセスユニットに関連付けられる (7.4.1.2.3 小小小節参照)。

**leading\_zero\_8bits**は0x00に等しいバイトである。

記 **leading\_zero\_8bits**シンタックス要素は、ビットストリームの初めのバイトストリームNALユニットにのみ存在できる。なぜならば、(B.1.1小節のシンタックス付図表で示されるように) NALユニットシンタックス構造に続き、(**start\_code\_prefix\_one\_3bytes**が続く**zero\_byte**と解釈される) 4バイトシーケンス0x00000001に先行する0x00に等しい任意のバイトは、先行するバイトストリームNALユニットの一部である**trailing\_zero\_8bits**シンタックス要素であるとみなされるからである。

**zero\_byte**は0x00に等しい単一のバイトである。

以下の条件のいずれかが満たされる場合、**zero\_byte**シンタックス要素が存在しなければならない。

- **nal\_unit()**内の**nal\_unit\_type**が、7 (シーケンスパラメータセット) または8 (ピクチャパラメータセット) に等しい。
- 7.4.1.2.3小小小節で規定されるように、バイトストリームNALユニットのシンタックス構造が、復号順序においてアクセスユニットの最初のNALユニットを含む。

**start\_code\_prefix\_one\_3bytes**は、0x000001に等しい3バイトの固定値シーケンスである。このシンタックス要素はスタートコードプリフィックスと呼ばれる。

**trailing\_zero\_8bits**は0x00に等しいバイトである。

### B.2 バイトストリームNALユニットの復号処理

この処理への入力は、バイトストリーム NAL ユニットのシンタックス構造のシーケンスから成る、順序付けられたバイトストリームから成る。

この処理の出力は、NAL ユニットシンタックス構造のシーケンスから成る。

復号処理の始めに、復号器はバイトストリーム中の現在の位置を、バイトストリームの始めへ初期化する。そして、バイトストリーム中の現在の位置が、ビットストリーム中の次の 4 バイトが 4 バイトシーケンス 0x00000001 を形成する様になるまで、(もし存在すれば) 各々の `leading_zero_8bits` シンタックス要素を抽出し捨て、バイトストリーム中の現在の位置をそのたびに 1 バイトずつ前進させる。

そして復号器は、バイトストリーム中の各 NAL ユニットのシンタックス構造を抽出および復号するために、(無規定の手段によって判定される) バイトストリームの終了に出会い、バイトストリーム中の最後の NAL ユニットが復号されるまで、以下のステップ毎処理を繰り返し実行する。

1. ビットストリーム中の次の 4 バイトが 4 バイトシーケンス 0x00000001 を形成する時、バイトストリーム中の次のバイト (これは `zero_byte` シンタックス要素である) が抽出され捨てられ、そしてバイトストリーム中の現在の位置は、この捨てられたバイトに続くバイト位置に等しく設定される。
2. バイトストリーム中の次の 3 バイトシーケンス (これは `start_code_prefix_one_3bytes` である) は抽出され破棄される。また、バイトストリーム中の現在の位置は、この 3 バイトシーケンスに続くバイト位置に等しく設定される。
3. `NumBytesInNALunit` は、バイトストリーム中の現在の位置におけるバイトから始まり、以下の条件のいずれかの位置に先行する最後のバイトを含むまでの、バイト数に等しく設定される。
  - a. 次のバイト整列された 0x000000 と等しい 3 バイトシーケンス、あるいは
  - b. 次のバイト整列された 0x000001 と等しい 3 バイトシーケンス、あるいは
  - c. 無規定の手段によって決定されたバイトストリームの終了
4. `NumBytesInNALunit` バイトがビットストリームから除去される。また、バイトストリーム中の現在の位置は、`NumBytesInNALunit` バイトだけ進められる。バイトのこのシーケンスは `nal_unit(NumBytesInNALunit)` であり、NAL ユニットの復号処理を使って復号される。
5. バイトストリーム中の現在の位置が (無規定な方法で決められた) バイトストリームの終了ではなく、かつバイトストリーム中の次の複数バイトが 0x000001 に等しい 3 バイトシーケンスで始まらず、かつバイトストリーム中の次の複数バイトが 0x00000001 に等しい 4 バイトシーケンスで始まらない時、復号器は、バイトストリーム中の現在の位置が、バイトストリーム中の次の複数バイトが 4 バイトシーケンス 0x00000001 を形成する様になる、または (無規定の手段によって判定される) バイトストリームの終了に出会うまで、各々の `trailing_zero_8bits` シンタックス要素を抽出し捨て、バイトストリーム中の現在の位置をそのたびに 1 バイトずつ前進させる。

### B.3 復号器のバイト整列の回復 (参考)

この節は、本標準の必須部分を形成しない。

多くのアプリケーションは、本質的にバイト整列された方法で復号器へのデータを提供しており、従ってこの節に記述されているビット指向のバイト整列検出手続きは必要ない。

復号器がビットストリームのデータの位置がバイト整列されているかどうか決定できる時、復号器はビットストリームとのバイト整列をもつと呼ばれる。復号器が符号器のバイトストリームとのバイト整列を持たない場合、復号器は入力ビットストリームに対してバイナリのパターン'00000000 00000000 00000000

00000001(1 と等しいビットが後続する、連続 31 ビットの 0 と等しいビット)を検査しても良い。このパターンの直後のビットが、スタートコードプリフィックスに後続する整列されたバイトの最初のビットである。このパターンを検出することにより、復号器は符号器とバイト整列が取れ、またバイトストリーム中の NAL ユニットの開始位置に置かれる。

符号器とのバイト整列が 1 度取れたら、復号器は入力バイトストリームに対して後続する 3 バイトシーケンス 0x000001 および 0x000003 を検査すれば良い。

3 バイトシーケンス 0x000001 が検出された場合、これはスタートコードプリフィックスである。

3 バイトシーケンス 0x000003 が検出された場合、3 番目のバイト(0x03)は 7.4.1 小節で規定されるように破棄される `emulation_prevention_three_byte` である。

ビットストリームシンタックスに誤りが検出された時（例えば、7.4.1 小節で禁じられた、非ゼロ値の `forbidden_zero_bit` または 3 バイト、4 バイトシーケンスの 1 つ）、復号器は、その検出状態をバイト整列が失われているかもしれない徴候と考えてよく、この節で記述されるようなビットストリーム中の後の位置でのバイト整列の検出まで全てのビットストリームデータを破棄してよい。

## 付属資料C 仮想標準復号器

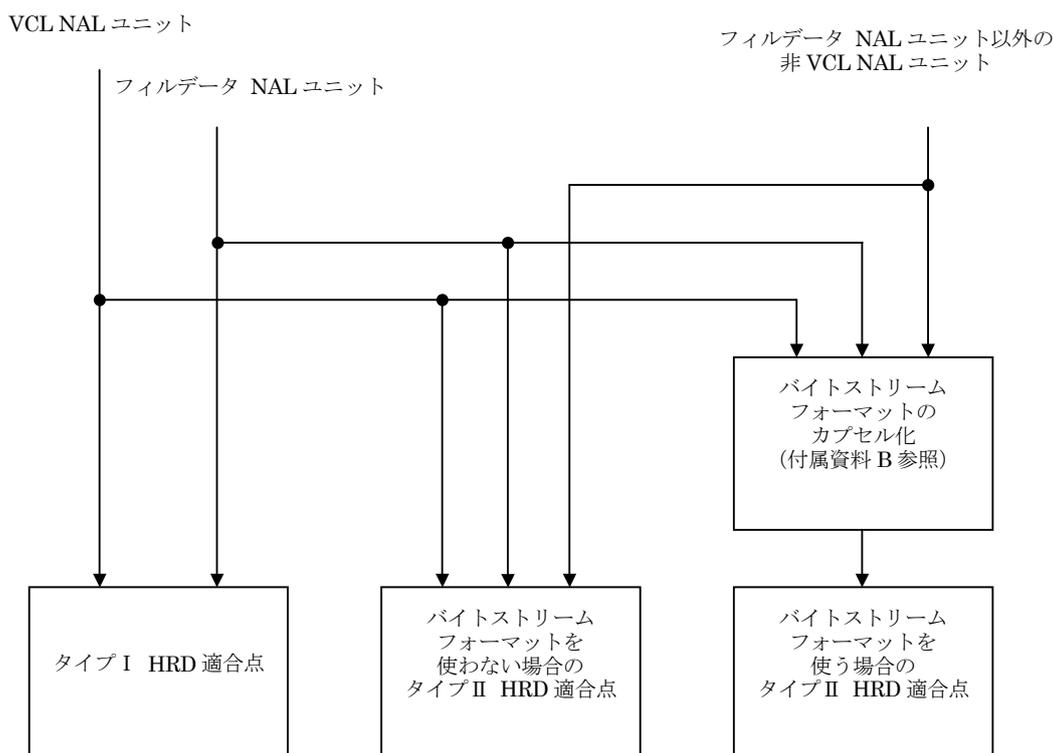
(この付属資料は、本標準の必須部分を形成する)

この付属資料は、仮想標準復号器 (HRD) と、ビットストリームおよび復号器の適合を検証するためのその使用法を規定する。

2 つのタイプのビットストリームが、本標準の HRD 適合検証の対象である。タイプ I ビットストリームと呼ばれる第 1 のタイプのビットストリームは、ビットストリーム中の全てのアクセスユニットで、VCL NAL ユニットおよびフィルデータ NAL ユニットだけを含んでいる NAL ユニットストリームである。タイプ II ビットストリームと呼ばれる第 2 のタイプのビットストリームは、ビットストリーム中の全てのアクセスユニットでの VCL NAL ユニットとフィルデータ NAL ユニットに加えて、下記の少なくとも 1 つを含んでいる。

- フィルデータNALユニット以外の追加の非VCL NALユニット
- (付属資料Bで規定されるように)NALユニットストリームからバイトストリームを形成する、全ての`leading_zero_8bits`、`zero_byte`、`start_code_prefix_one_3bytes`、そして`trailing_zero_8bits`シンタックス要素

付図 C-1/JT-H264 は HRD で検証されるビットストリーム適合点のタイプを示している。



付図C-1/JT-H264 HRD適合検証のためのバイトストリームとNALユニットストリームの構造 (ITU-T H.264)

HRD に必要とされる非 VCL NAL ユニットのシンタックス要素（あるいは、いくつかのシンタックス要素に対するデフォルト値）は、7 章および付属資料 D と E の意味に関する小節で規定される。

2 つのタイプの HRD パラメータセットが使われる。HRD パラメータセットは、E.1 節および E.2 節で規定されるような、シーケンスパラメータセットのシンタックス構造の一部であるビデオ有用性情報によって伝えられる。

HRD を使ってビットストリームの適合を検証するためには、VCL NAL ユニット内で参照される全てのシーケンスパラメータセットおよびピクチャパラメータセット、および対応するバッファリング期間とピクチャタイミング SEI メッセージが、適時の方法で、ビットストリーム内（非 VCL NAL units による）あるいは本標準で規定されない他の手段で、HRD へ伝達されなければならない。

付属資料 C、D および E では、それらの NAL ユニット（あるいは単にそれらの内のいくつか）が本標準で規定されない他の手段によって復号器（あるいは HRD）に伝達される場合でも、非 VCL NAL ユニットの「存在」に対する規定は満たされる。ビットを数える目的では、ビットストリーム中に実際に存在する適切なビットだけが数えられる。

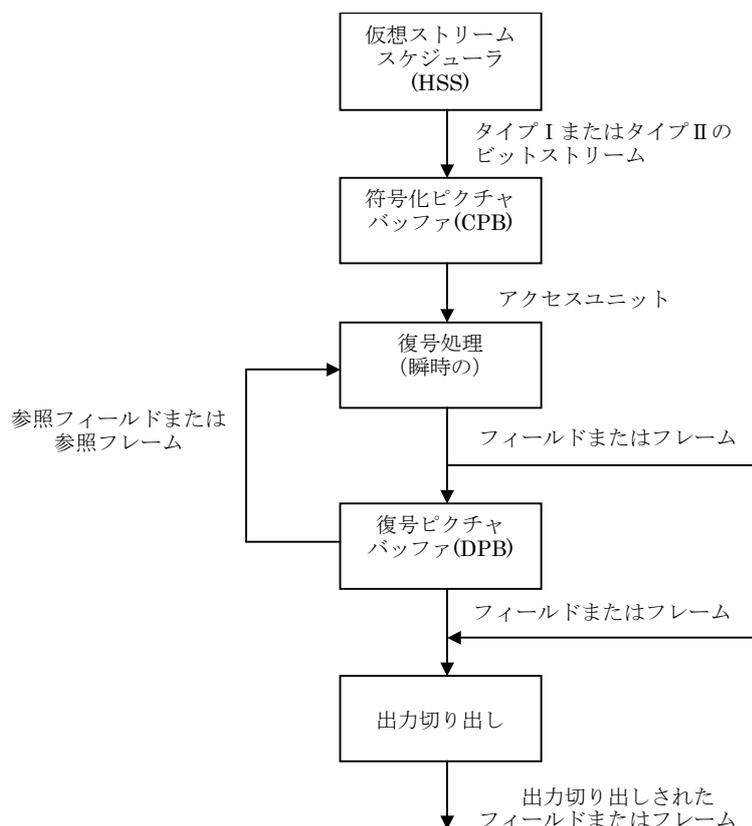
記1 例として、ビットストリーム中の存在以外の手段によって伝達された非VCL NALユニットの、ビットストリーム中に存在するNALユニットとの同期は、符号器がビットストリーム中でそれを伝達することを決定したとするなら、ビットストリーム中でその間に非VCL NALユニットが存在したであろう2点をビットストリーム中に示すことによって達成できる。

非 VCL NAL ユニット内容が、ビットストリーム内の存在以外の手段でアプリケーションに伝達される場

合、非 VCL NAL ユニット内容の表現は、この付属資料で規定された同じシンタックスを使用することは要求されない。

記2 ビットストリーム内にHRD情報が含まれている場合は、ビットストリームに含まれる情報に基づいて単独で、この節の要件へのビットストリームの適合を検証することが可能である。全ての“スタンドアロン”タイプ I ビットストリームの場合と同様に、HRD情報がビットストリームに存在しない場合は、本標準で規定されない他の方法によってHRDデータが提供された場合にのみ適合は検証できる。

付図 C-2/JT-H264 で示されるように、HRD は符号化ピクチャバッファ (CPB)、瞬時復号処理、復号ピクチャバッファ (DPB) と出力の切り出しを含んでいる。



付図C-2/JT-H264 HRDバッファモデル  
(ITU-T H.264)

CPB サイズ (ビット数) は  $CpbSize[SchedSelIdx]$  である。DPB サイズ (フレームバッファ数) は  $Max(1, max\_dec\_frame\_buffering)$  である。

HRD は以下のように動作する。規定された到達スケジュールに従って CPB へ流れるアクセスユニットに関連したデータは、HSS によって配達される。各アクセスユニットに関連したデータは、CPB 除去時刻に瞬時復号処理によって瞬時に除去され復号される。各復号ピクチャは、それが CPB 除去時刻における出力でありかつ非参照ピクチャである場合を除いて、CPB 除去時刻に DPB に置かれる。ピクチャが DPB に置かれる場合、DPB 出力時刻または“参照不使用”としてマークされた時刻のうち遅い方の時刻に、それは DPB から除去される。

ビットストリーム中の各ピクチャに対して、復号ピクチャに対する変数 `OutputFlag` および、適用可能な場

合、参照ベースピクチャが以下の通り設定される。

- もしそのピクチャを含む符号化ビデオシーケンスが付属資料 A に規定される 1 つ以上のプロファイルに適合し、かつ 2-9 章に規定される復号処理が用いられるならば、OutputFlag は 1 に等しく設定される。
- それ以外 (そのピクチャを含む符号化ビデオシーケンスが付属資料 G に規定されるプロファイルに適合し、かつ付属資料 G に規定される復号処理が用いられる)、次が適用される。
  - 参照ベースピクチャに対して、OutputFlag は 0 に等しく設定される。
  - 復号ピクチャに対して、OutputFlag はターゲットレイヤ表現の output\_flag シンタックス要素の値に等しく設定される。

CPB の動作は C.1 節で規定される。瞬時復号器の動作は、8 章および 9 章で規定される。DPB の動作は、C.2 節で規定される。出力切り出しは C.2.2 小節で規定される。

列挙された配達スケジュール数、およびそれらの関連するビットレートとバッファサイズに関する HSS および HRD 情報は、E.1.1 小節、E.1.2 小節、E.2.1 小節と E.2.2 小節で規定される。D.1.1 小節および D.2.1 小節で規定されるバッファリング期間 SEI メッセージによって規定されるように、HRD が初期化される。CPB からのアクセスユニットの除去タイミングおよび DPB からの出力タイミングは、D.1.2 小節および D.2.2 小節で規定されるように、ピクチャタイミング SEI メッセージの中で規定される。特定のアクセスユニットに関する全てのタイミング情報は、アクセスユニットの CPB 除去時刻に先立って到達しなければならない。

C.3 節および C.4 節でそれぞれ規定されるように、HRD はビットストリームおよび復号器の適合を検証するために使用される。

記3 ビットストリームを生成するために使用される全てのフレームレートおよびクロックは、ビットストリームで伝えられた値と正確に一致するという仮定の下で適合が保証されている一方で、現実のシステムでは、これらの各々は伝えられたり規定される値とは異なるかもしれない。

丸め誤差が伝播しないように、この付属資料の計算は全て実数で行われる。例えば、アクセスユニット除去の直前または直後の CPB 中のビット数は、必ずしも整数ではない。

変数  $t_c$  は下記のように導出され、クロックチックと呼ばれる。

$$t_c = \text{num\_units\_in\_tick} \div \text{time\_scale} \quad (\text{C-1})$$

下記は、この付属資料の制約を表現するために規定される。

- 最初のアクセスユニットをアクセスユニット 0 とした、復号順序で n 番目のアクセスユニットをアクセスユニット n とする。
- ピクチャ n をアクセスユニット n の主符号化ピクチャまたは復号主ピクチャとする。

## C.1 符号化ピクチャバッファ(CPB)の動作

この節の規定は、存在する各 CPB パラメータのセットと付図 C-1/JT-H264 で示されるタイプ I とタイプ II の両方の適合点に、独立して適用される。

### C.1.1 ビットストリームの到達タイミング

HRD は、いずれか 1 つのバッファリング期間 SEI メッセージで初期化されて良い。初期化に先立って、

CPB は空である。

記 初期化の後に、HRDはその後のバッファリング期間SEIメッセージによって再び初期化されない。

各々のアクセスユニットはアクセスユニット  $n$  として言及される。ここで数  $n$  は特定のアクセスユニットを識別する。CPB を初期化するバッファリング期間 SEI メッセージに関連付けられるアクセスユニットは、アクセスユニット 0 として言及される。 $n$  の値は、復号順序で各々の後続するアクセスユニットに対して 1 だけ増加される。

アクセスユニット  $n$  の最初のビットが CPB に入り始める時刻が、初期到達時刻  $t_{ai}(n)$  として言及される。

アクセスユニットの初期到達時刻は次の通り導出される。

- もしアクセスユニットがアクセスユニット 0 であるならば、 $t_{ai}(0)=0$ 、
- それ以外 (アクセスユニットが  $n>0$  であるアクセスユニット  $n$ )、次が適用される。
  - もし  $cbr\_flag[SchedSelIdx]$  が 1 に等しいならば、アクセスユニット  $n$  の初期到達時刻はアクセスユニット  $n-1$  の最終到達時刻 (以後導出される) となる。

$$t_{ai}(n)=t_{af}(n-1) \quad (C-2)$$

- それ以外 ( $cbr\_flag[SchedSelIdx]$  が 0 に等しい)、アクセスユニット  $n$  の初期到達時刻は次のように導出される。

$$t_{ai}(n)=\text{Max}(t_{ai}(n-1), t_{ai,earliest}(n)) \quad (C-3)$$

ここで  $t_{ai,earliest}(n)$  は次の通り導出される。

- もしアクセスユニット  $n$  がその後のバッファリング期間の最初のアクセスユニットでないならば、 $t_{ai,earliest}(n)$  は以下のように導出される。

$$t_{ai,earliest}(n)=t_{r,n}(n)-\frac{(\text{initial\_cpb\_removal\_delay}[SchedSelIdx]+\text{initial\_cpb\_removal\_delay\_offset}[SchedSelIdx])}{90000} \quad (C-4)$$

$t_{r,n}(n)$  は、C.1.2 小節で規定されるように CPB からのアクセスユニット  $n$  の名目上の除去時刻である。また  $\text{initial\_cpb\_removal\_delay}[SchedSelIdx]$  と  $\text{initial\_cpb\_removal\_delay\_offset}[SchedSelIdx]$  はこれに先立つバッファリング期間 SEI メッセージで規定される。

- それ以外 (アクセスユニット  $n$  がその後のバッファリング期間の最初のアクセスユニットである)、 $t_{ai,earliest}(n)$  は以下のように導出される。

$$t_{ai,earliest}(n)=t_{r,n}(n)-(\text{initial\_cpb\_removal\_delay}[SchedSelIdx] \div 90000) \quad (C-5)$$

$\text{initial\_cpb\_removal\_delay}[SchedSelIdx]$  は、アクセスユニット  $n$  に関連したバッファリング期間 SEI メッセージで規定される。

アクセスユニット  $n$  の最終到達時刻は、以下のように導出される。

$$t_{af}(n)=t_{ai}(n)+b(n) \div \text{BitRate}[SchedSelIdx] \quad (C-6)$$

ここで、 $b(n)$ はアクセスユニット  $n$  のビットでのサイズであり、タイプ I 適合点のための VCL NAL ユニットとフィルデータ NAL ユニットのビット、またはタイプ II 適合点のためのタイプ II ビットストリームの全てのビットを数える。ここで、タイプ I とタイプ II 適合点は付図 C-1/JT-H264 で示される。

SchedSelIdx、BitRate[SchedSelIdx]と CpbSize[SchedSelIdx]の値は、次の通り制約される。

- もしアクセスユニット  $n$  とアクセスユニット  $n-1$  に対するアクティブシーケンスパラメータセットの内容が異なるならば、HSS は、アクセスユニット  $n$  に対して、アクセスユニット  $n$  に対するアクティブシーケンスパラメータセットで提供される SchedSelIdx の値の中から、BitRate[SchedSelIdx1]または CpbSize[SchedSelIdx1]となる SchedSelIdx の値 SchedSelIdx1 を選択する。BitRate[SchedSelIdx1]または CpbSize[SchedSelIdx1]の値は、アクセスユニット  $n-1$  に対して使用された SchedSelIdx の値 SchedSelIdx0 に対する BitRate[SchedSelIdx0]または CpbSize[SchedSelIdx0]の値と異なってもよい。
- それ以外、HSS は前の SchedSelIdx、BitRate[SchedSelIdx]および CpbSize[SchedSelIdx]の値で動作を続ける。

HSS が前のアクセスユニットとは値が異なる BitRate[SchedSelIdx]あるいは CpbSize[SchedSelIdx]の値を選択した場合、次が適用される。

- 変数 BitRate[SchedSelIdx]は、時刻  $t_{ai}(n)$ で有効となる。
- 変数 CpbSize[SchedSelIdx]は、次の通り有効となる。
  - もし CpbSize[SchedSelIdx]の新しい値が古い CPB サイズを超えるならば、それは時刻  $t_{ai}(n)$ で有効となる。
  - それ以外、CpbSize[SchedSelIdx]の新しい値は、時刻  $t_r(n)$ で有効となる。

### C.1.2 符号化ピクチャ除去のタイミング

アクセスユニット 0 については、CPB からのアクセスユニットの名目上の除去時刻は次のように規定される。

$$t_{r,n}(0)=\text{initial\_cpb\_removal\_delay}[\text{SchedSelIdx}] \div 90000 \quad (\text{C-7})$$

HRD を初期化しないバッファリング期間の最初のアクセスユニットについては、CPB からのアクセスユニットの名目上の除去時刻は次のように規定される。

$$t_{r,n}(n)=t_{r,n}(n_b)+t_c*\text{cpb\_removal\_delay}(n) \quad (\text{C-8})$$

ここで、 $t_{r,n}(n_b)$ はその前のバッファリング期間の最初のアクセスユニットの名目上の除去時刻である。また  $\text{cpb\_removal\_delay}(n)$ はアクセスユニット  $n$  に関連付けられたピクチャタイミング SEI メッセージの中で規定される  $\text{cpb\_removal\_delay}$  の値である。

アクセスユニット  $n$  がバッファリング期間の最初のアクセスユニットである場合、 $n_b$  はアクセスユニット  $n$  の除去時刻において  $n$  と等しく設定される。

バッファリング期間の最初のアクセスユニットではないアクセスユニット  $n$  の名目上の除去時刻  $t_{r,n}(n)$ は、次のように与えられる。

$$t_{r,n}(n)=t_{r,n}(n_b)+t_c*\text{cpb\_removal\_delay}(n) \quad (\text{C-9})$$

ここで、 $t_{rn}(n_b)$ は現バッファリング期間の最初のアクセスユニットの名目上の除去時刻である。また  $cpb\_removal\_delay(n)$ はアクセスユニット  $n$  に関連付けられたピクチャタイミング SEI メッセージの中で規定される  $cpb\_removal\_delay$  の値である。

アクセスユニット  $n$  の除去時刻は次の通り規定される。

- もし  $low\_delay\_hrd\_flag$  が 0 と等しいか、あるいは  $t_{rn}(n) \geq t_{af}(n)$  であるならば、アクセスユニット  $n$  の除去時刻は次のように規定される。

$$t_r(n) = t_{rn}(n) \tag{C-10}$$

- それ以外 ( $low\_delay\_hrd\_flag$  が 1 と等しく、かつ  $t_{rn}(n) < t_{af}(n)$ )、アクセスユニット  $n$  の除去時刻は次のように規定される。

$$t_r(n) = t_{rn}(n) + t_c * Ceil((t_{af}(n) - t_{rn}(n)) \div t_c) \tag{C-11}$$

記 後者のケースは、アクセスユニット  $n$  のサイズ  $b(n)$  は非常に大きいので、名目上の除去時刻に除去を防ぐことを示している。

## C.2 復号ピクチャバッファ(DPB)の動作

復号ピクチャバッファはフレームバッファを含んでいる。2-9 章に規定される復号処理を用いて、付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、各フレームバッファは、“参照使用” (参照ピクチャ) としてマークされるか、あるいは将来の出力 (並替または遅延されたピクチャ) のために保持される、復号フレーム、復号相補的フィールドペア、あるいは単一 (非ペア) 復号フィールドを含んでも良い。付属資料 G に規定される復号処理を用いて付属資料 G に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、各フレームバッファは、“参照使用” (参照ピクチャ) としてマークされるか、あるいは将来の出力 (並替または遅延されたピクチャ) のために保持される、復号フレーム、復号相補的フィールドペア、単一 (非ペア) 復号フィールド、復号参照ベースフレーム、復号参照ベース相補的フィールドペア、あるいは単一 (非ペア) 復号参照ベースフィールドを含んでも良い。初期化に先立って、DPB は空 (DPB 占有量はゼロに設定される) である。この節の小節の以下のステップは全て、記載されるシーケンスで  $t_r(n)$  において瞬時に起こる。

### C.2.1 frame\_numのギャップの復号と“非存在”フレームの保存

適用可能な場合、 $frame\_num$  のギャップは復号処理で検出され、そして生成フレームはマークされ以後規定されるように DPB に挿入される。

$frame\_num$  のギャップは復号処理で検出され、生成フレームは 8.2.5.2 小小節で規定されるようにマークされる。

各生成フレームのマークの後、“スライディングウィンドウ” 処理により “参照不使用” としてマークされる各ピクチャ  $m$  は、それが “非存在” としてもマークされるか、その DPB 出力時刻が現ピクチャ  $n$  の CPB 除去時刻以下、すなわち  $t_{o,dpb}(m) \leq t_r(n)$ 、または 0 に等しい  $OutputFlag$  を持つ時、DPB から除去される。フレームバッファの 1 つのフレームや最後のフィールドが DPB から除去される時、DPB 占有量が 1 減少される。“非存在” 生成フレームは DPB に挿入され、DPB 占有量が 1 増加される。

### C.2.2 ピクチャの復号と出力

ピクチャ  $n$  が復号される。それが 1 に等しい  $OutputFlag$  を持つ時、その DPB 出力時刻  $t_{o,dpb}(n)$  は次のよう

に導出される。

$$t_{o,dpb}(n)=t_r(n)+t_c*dpb\_output\_delay(n) \quad (C-12)$$

現ピクチャの出力は次の通り規定される。

- もし OutputFlag が 1 に等しくかつ  $t_{o,dpb}(n)=t_r(n)$  ならば、現ピクチャが出力される。  
記 現ピクチャが参照ピクチャである場合、それはDPBへ保存されるだろう。
- それ以外、もし OutputFlag が 0 に等しいならば、現ピクチャは出力されないが、C.2.4 小節で規定されるように、DPB に保存されるかもしれない。
- それ以外 (OutputFlag が 1 に等しくかつ  $t_{o,dpb}(n)>t_r(n)$ )、現ピクチャが遅れて出力され、(C.2.4 小節で規定されるように) DPB へ保存されるだろう。そして、 $t_{o,dpb}(n)$  に先立ったある時刻で、no\_output\_of\_prior\_pics\_flag が 1 と等しいという復号あるいは推定によって出力されるべきではないと示されない限り、現ピクチャは時刻  $t_{o,dpb}(n)$  に出力される。

出力される時、ピクチャは、そのピクチャに対するアクティブシーケンスパラメータセットの中で規定される切り出し矩形を使用して切り出されなければならない。

ピクチャ  $n$  が出力されるピクチャであり、ビットストリームの出力される最後のピクチャではない場合、 $\Delta t_{o,dpb}(n)$ の値は次のように定義される。

$$\Delta t_{o,dpb}(n)=t_{o,dpb}(n_n)-t_{o,dpb}(n) \quad (C-13)$$

ここで、 $n_n$  は出力順序でピクチャ  $n$  の後に続き、1 に等しい OutputFlag を持つピクチャを示す。

復号ピクチャは一時的に (DPB の中ではなく) 保存される。

### C.2.3 現ピクチャの挿入が起こりうる前のDPBからのピクチャ除去

現ピクチャの挿入が起こりうる前の DPB からのピクチャの除去は、次の通り進む。

- もし復号ピクチャが IDR ピクチャであるならば、次が適用される。
  - 2-9章に規定される復号処理を用いて、付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、8.2.5.1小節で規定されるように、あるいは、付属資料Gに規定される復号処理を用いて、付属資料Gに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、G.8.2.4小節に規定されるように、DPB内の全ての参照ピクチャは“参照不使用”としてマークされる。
  - IDRピクチャが復号された最初のIDRピクチャではなく、アクティブなシーケンスパラメータセットから導出されるPicWidthInMbs、FrameHeightInMbs、あるいはmax\_dec\_frame\_bufferingの値が、先行ピクチャに対してアクティブなシーケンスパラメータセットから導出されたPicWidthInMbs、FrameHeightInMbs、あるいは max\_dec\_frame\_buffering の値と各々異なる場合、no\_output\_of\_prior\_pics\_flagの実際の値とは無関係にno\_output\_of\_prior\_pics\_flagはHRDによって1に等しいと推定される。

記 復号器の実装は、PicWidthInMbsまたはFrameHeightInMbs内の変更に関してHRDよりも行儀良く、フレームまたはDPBサイズの変更を扱うよう努めるべきである。

- no\_output\_of\_prior\_pics\_flagが1と等しい、あるいは1と等しいと推定される場合、それらが含むピクチャを出力することなくDPB内の全てのフレームバッファは空にされ、DPB占有量は0に設定される。
- それ以外（復号ピクチャがIDRピクチャでない）、次が適用される。
  - もし現ピクチャのスライスヘッダが5と等しいmemory\_management\_control\_operationを含むならば、DPB内の全ての参照ピクチャは“参照不使用”としてマークされる。
  - それ以外（現ピクチャのスライスヘッダが5と等しいmemory\_management\_control\_operationを含まない）、2-9章に規定される復号処理を用いて、付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、8.2.5小節で規定される復号参照ピクチャマーキング処理が起動され、または、付属資料Gに規定される復号処理を用いて、付属資料Gに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、G.8.2.4小節で規定される復号参照ピクチャマーキング処理が起動される。

以下の条件の全てが真であるDPB内の全てのピクチャmは、DPBから除去される。

- ピクチャmが“参照不使用”としてマークされる、あるいはピクチャmが非参照ピクチャである。ピクチャが参照フレームである場合は、その両方のフィールドが“参照不使用”としてマークされていた場合だけ、それは“参照不使用”としてマークされているとみなされる。
- ピクチャmが“非存在”としてマークされる、あるいは0に等しいOutputFlagを持つ、あるいはそのDPB出力時刻が、現ピクチャnのCPB除去時刻よりも小さいかまたは等しい。すなわち、 $t_{o,dpb}(m) \leq t_r(n)$ である。

フレームバッファの1つのフレームや最後のフィールドがDPBから除去される時、DPB占有量が1減少される。

## C.2.4 現復号ピクチャのマーキングと保存

### C.2.4.1 DPBへの参照復号ピクチャのマーキングと保存

現ピクチャが参照ピクチャである場合、それは次の通りDPBへ保存される。

- もし現復号ピクチャが相補的参照フィールドペアの（復号順序で）2番目のフィールドなら、そしてそのペアの1番目のフィールドがまだDPBにあるならば、その現復号ピクチャはそのペアの1番目のフィールドと同じフレームバッファに保存される。
- それ以外、現復号ピクチャは空のフレームバッファに保存され、DPB占有量は1増加される。

### C.2.4.2 DPBへの非参照ピクチャの保存

現ピクチャが非参照ピクチャであり、現ピクチャnが $t_{o,dpb}(n) > t_r(n)$ である場合、それは次の通りDPBへ保存される。

- もし現復号ピクチャが相補的非参照フィールドペアの（復号順序で）2番目のフィールドなら、そしてそのペアの1番目のフィールドがまだDPBにあるならば、その現復号ピクチャはそのペアの1番目のフィールドと同じフレームバッファに保存される。
- それ以外、現復号ピクチャは空のフレームバッファに保存され、DPB占有量は1増加される。

### C.3 ビットストリームの適合

本標準に適合する符号化データのビットストリームは、以下の要件を満たしている。

ビットストリームは、この付属資料以外の本標準で規定されたシンタックス、意味、および制約に従って構築される。

ビットストリームは以下で規定されるように HRD によって試験される。

タイプ I ビットストリームについては、実施される試験数は  $cpb\_cnt\_minus1+1$  と等しい。ここで、 $cpb\_cnt\_minus1$  は、 $vcl\_hrd\_parameters\_present\_flag$  に続く  $hrd\_parameters()$  のシンタックス要素であるか、あるいは本標準で規定されない他の方法によりアプリケーションで決定される。1 つの試験が、 $vcl\_hrd\_parameters\_present\_flag$  に続く  $hrd\_parameters()$  で規定される各ビットレートおよび CPB サイズの組合せに対して実施される。これら試験の各々は付図 C-1/JT-H264 で示されるタイプ I 適合点で行われる。

タイプ II ビットストリームについては、2 組の試験がある。第 1 の組の試験数は  $cpb\_cnt\_minus1+1$  と等しい。ここで、 $cpb\_cnt\_minus1$  は、 $vcl\_hrd\_parameters\_present\_flag$  に続く  $hrd\_parameters()$  のシンタックス要素であるか、あるいは本標準で規定されない他の方法によりアプリケーションで決定される。1 つの試験が、各ビットレートおよび CPB サイズの組合せに対して実施される。これら試験の各々は付図 C-1/JT-H264 で示されるタイプ I 適合点で行われる。これらの試験については、入力ビットレートおよび CPB 保存のために、VCL およびフィルデータ NAL ユニットだけが数えられる。

タイプ II ビットストリームに対する第 2 の組の試験数は、 $cpb\_cnt\_minus1+1$  と等しい。ここで、 $cpb\_cnt\_minus1$  は、 $nal\_hrd\_parameters\_present\_flag$  に続く  $hrd\_parameters()$  のシンタックス要素であるか、あるいは本標準で規定されない他の方法によりアプリケーションで決定される。1 つの試験が、 $nal\_hrd\_parameters\_present\_flag$  に続く  $hrd\_parameters()$  で規定された各ビットレートおよび CPB サイズの組合せに対して実施される。これら試験の各々は付図 C-1/JT-H264 で示されるタイプ II 適合点で行われる。これらの試験については、(タイプ II NAL ユニットストリームの) 全ての NAL ユニットあるいは (バイトストリームの) 全てのバイトが入力ビットレートおよび CPB 保存のために数えられる。

記1 付図C-1/JT-H264で示されるタイプ II 適合点に対する  $SchedSelIdx$  の値により確立された NAL HRD パラメータは、VBR の場合 ( $cbr\_flag[SchedSelIdx]$  が 0 に等しい) における  $initial\_cpb\_removal\_delay[SchedSelIdx]$ 、 $BitRate[SchedSelIdx]$  そして  $CpbSize[SchedSelIdx]$  の同じ値に対して、付図C-1/JT-H264で示されるタイプ I 適合点に対する VCL HRD 適合をも確立するのに十分である。これは、タイプ I 適合点へのデータの流れがタイプ II 適合点へのデータの流れの部分集合だからであり、そして、VBR の場合においては、CPB が空になり、次のピクチャが着き始めることがスケジュールされるまで空であり続けることが許されるからである。例えば、2-9章に規定される復号処理を用いて、付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、NAL HRD パラメータが、A.3.1小節のj項またはA.3.3小節のh項(使用プロファイルによる)のプロファイル適合における NAL HRD パラメータに対する束縛集合に入るだけでなく、A.3.1小節のi項またはA.3.3小節のg項(使用プロファイルによる)のプロファイル適合における VCL HRD パラメータに対する束縛集合にも入るタイプ II 適合点に対して与えられた時、タイプ I 適合点に対する VCL HRD の適合は、A.3.1小節のi項の束縛に入ることも保証される。

適合するビットストリームについては、各試験に対して以下の条件の全てが満たされなければならない。

- バッファリング期間 SEI メッセージに関連付けられた、 $n>0$  である各アクセスユニット  $n$  に対して、以下で規定される  $\Delta t_{g,90}(n)$  を用いて

$$\Delta t_{g,90}(n)=90000*(t_{r,n}(n)-t_{af}(n-1)) \quad (C-14)$$

$initial\_cpb\_removal\_delay[SchedSelIdx]$  の値は次の通り制約される。

- もし  $cbr\_flag[SchedSelIdx]$  が 0 に等しいならば、

$$initial\_cpb\_removal\_delay[SchedSelIdx] \leq Ceil(\Delta t_{g,90}(n)) \quad (C-15)$$

- それ以外 ( $cbr\_flag[SchedSelIdx]$  が 1 に等しい)、

$$Floor(\Delta t_{g,90}(n)) \leq initial\_cpb\_removal\_delay[SchedSelIdx] \leq Ceil(\Delta t_{g,90}(n)) \quad (C-16)$$

記2 各ピクチャの除去時刻でのCPBの正確なビット数は、HRDを初期化するためにどのバッファリング期間SEIメッセージが選択されたかに依存してよい。符号器は、HRDがバッファリング期間SEIメッセージの任意の1つで初期化されるため、HRDを初期化するためにどのバッファリング期間SEIメッセージが選択されたかにかかわらず、全ての規定された制約に従わなくてはならないことを保証するよう配慮しなくてはならない。

- CPB オーバーフローは、CPB の合計ビット数が CPB サイズより大きい状態として規定される。CPB はオーバーフローしてはならない。
- CPB アンダーフローは、 $t_{r,n}(n)$  が  $t_a(n)$  より小さい状態として規定される。 $low\_delay\_hrd\_flag$  が 0 に等しい場合、CPB はアンダーフローしてはならない。
- CPB からの名目上のピクチャの除去時刻は (復号順序で 2 番目のピクチャから開始して)、2-9 章に規定される復号処理を用いて、付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、ビットストリームに規定されたプロファイルとレベルに対して A.3.1 小節から A.3.3 小節までで示された  $t_{r,n}(n)$  および  $t_r(n)$  の制約を満たさなければならず、そしてそれらは、付属資料 G に規定される復号処理を用いて、付属資料 G に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、ビットストリームに規定されたプロファイルとレベルに対して G.10.2.1 小節および G.10.2.2 小節で示された  $t_{r,n}(n)$  および  $t_r(n)$  の制約を満たさなければならない。
- ビットストリームに規定されたプロファイルとレベルに対して付属資料 A、D、および E で制限されるように、ある復号ピクチャが DPB に加えられた直後は、DPB の占有量は DPB サイズよりも小さいかまたは等しくなければならない。
- 予測のために必要な場合、全ての参照ピクチャは DPB の中になければならない。各ピクチャは、C.2 節で規定された処理の 1 つによって、そもそも DPB の中に保存されていないかあるいは出力時刻よりも前に DPB から除去されたのでない限り、DPB 出力時刻には DPB の中になければならない。
- 等式 C-13 で与えられる  $\Delta t_{o,dpb}(n)$  の値、すなわちあるピクチャと、出力順序で続く最初の 1 に等しい OutputFlag を持つピクチャとの出力時刻の差分は、2-9 章に規定される復号処理を用いて、付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、ビットストリームに規定されたプロファイルとレベルに関して A.3.1 小節で示された制約を満たさなければならず、そしてそれは、付属資料 G に規定される復号処理を用いて、付属資料 G に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、ビットストリームに規定されたプロファイルとレベルに対して G.10.2.1 小節で示された制約を満たさなければならない。

#### C.4 復号器の適合

本標準に従った復号器は、以下の要件を満たしている。

特定のプロファイルとレベルへの適合をクレームする復号器は、VCL NAL ユニットで参照される全てのシーケンスパラメータセットおよびピクチャパラメータセットと、適切なバッファリング期間 SEI メッセージおよびピクチャタイミング SEI メッセージが、適時な手段で、ビットストリーム中 (非 VCL NAL ユニッ

トによって) または本標準で規定されない外部手段によって、復号器へ伝達されるという条件で、復号器の適合のために C.3 節で規定される全ての適合ビットストリームを正常に復号することができなければならない。

復号器によってクレームされる 2 つのタイプの適合、すなわち出力タイミングの適合と出力順序の適合がある。

復号器の適合を確認するために、C.3 節で規定されたようなクレームされるプロファイルとレベルに適合した試験ビットストリームが、仮想ストリームスケジューラ (HSS) によって HRD と試験される復号器 (DUT) の両方に配達される。HRD による全ての出力ピクチャは DUT でも出力されなければならない、HRD による各出力ピクチャに対して、対応するピクチャの DUT によって出力される全てのサンプル値は HRD によって出力されるサンプル値と等しくなければならない。

出力タイミングの復号器適合については、SchedSelIdx の値の部分集合からのみ選択され、規定されたプロファイルとレベルに対してビットレートと CPB サイズが付属資料 A で規定されるように制限された配達スケジュールで、あるいは以下で規定される、ビットレートと CPB サイズが付属資料 A で規定されるように制限された“補間された”配達スケジュールで、HSS は上述のように動作する。同じ配達スケジュールが HRD と DUT の両方に使用される。

cpb\_cnt\_minus1 が 0 より大きい、HRD パラメータとバッファリング期間 SEI メッセージが存在する場合、復号器は、次の通りピークビットレート  $r$ 、CPB サイズ  $c(r)$  と初期 CPB 除去遅延  $f(r) \div r$  を持つように規定される“補間された”配達スケジュールを使う動作をする HSS から配達されたビットストリームの復号が可能でなければならない。

$$\alpha = (r - \text{BitRate}[\text{SchedSelIdx} - 1]) \div (\text{BitRate}[\text{SchedSelIdx}] - \text{BitRate}[\text{SchedSelIdx} - 1]) \quad (\text{C-17})$$

$$c(r) = \alpha * \text{CpbSize}[\text{SchedSelIdx}] + (1 - \alpha) * \text{CpbSize}[\text{SchedSelIdx} - 1] \quad (\text{C-18})$$

$$f(r) = \alpha * \text{initial\_cpb\_removal\_delay}[\text{SchedSelIdx}] * \text{BitRate}[\text{SchedSelIdx}] + (1 - \alpha) * \text{initial\_cpb\_removal\_delay}[\text{SchedSelIdx} - 1] * \text{BitRate}[\text{SchedSelIdx} - 1] \quad (\text{C-19})$$

上式は任意の SchedSelIdx > 0 および、 $r$  と  $c(r)$  が規定されたプロファイルとレベルに対する最大ビットレートおよびバッファサイズに関する付属資料 A で規定された限定を満たすような、 $\text{BitRate}[\text{SchedSelIdx} - 1] < r \leq \text{BitRate}[\text{SchedSelIdx}]$  なる任意の  $r$  に対するものである。

記1 initial\_cpb\_removal\_delay[SchedSelIdx] はあるバッファリング期間と別のバッファリング期間とで異なっても良く、再計算されなくてはならない。

出力タイミングの復号器適合については、上記のような HRD が使用され、出力ピクチャのタイミング (最初のビットの配達時刻からの相対) は HRD と DUT の両方に対して固定遅延を除き同じである。

出力順序の復号器適合については、HSS は DUT からの“要求によって”、DUT へビットストリームを配達する。これは、DUT が処理を進めるためにより多くのビットを要求した場合にだけ、HSS が (復号順序で) ビットを配達することを意味している。

記2 これは、この試験において、DUT の符号化ピクチャバッファは最も大きなアクセスユニットのサイズと同じだけ小さくできることを意味する。

以後に示す修正 HRD が使用され、ビットレートや CPB サイズが付属資料 A で規定されるように制限されたビットストリーム内に規定されるスケジュールの 1 つによって、HSS は HRD へビットストリームを配達

する。出力ピクチャの順序は HRD と DUT の両方で同じでなければならない。

出力順序の復号器適合については、HRD CPB サイズは選択されたスケジュールのための CpbSize[SchedSelIdx]と等しく、そして DPB サイズは MaxDpbSize に等しい。HRD に対する CPB からの除去時刻は最終ビット到達時刻と等しく、復号は即時である。この HRD の DPB の動作は以後の通りである。

#### C.4.1 出力順序DPBの動作

復号ピクチャバッファはフレームバッファを含んでいる。2-9 章に規定される復号処理を用いて、付属資料 A に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、各フレームバッファは、“参照使用”としてマークされたか、(並替されるピクチャの) 将来の出力のために保持された復号フレーム、復号相補的フィールドペアまたは単一の (非ペア) 復号フィールドを含んでも良い。付属資料 G に規定される復号処理を用いて、付属資料 G に規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、各フレームバッファは、“参照使用”としてマークされたか、(並替されるピクチャの) 将来の出力のために保持された復号フレーム、復号相補的フィールドペア、単一 (非ペア) 復号フィールド、復号参照ベースフレーム、復号参照ベース相補的フィールドペア、あるいは単一 (非ペア) 復号参照ベースフィールドを含んでも良い。HRD の初期化において、フレームで計測される DPB 占有量は 0 に設定される。以下のステップは全て記載された順番で、アクセスユニットが CPB から除去される時、瞬時に起こる。

#### C.4.2 frame\_numのギャップの復号と“非存在”ピクチャの保存

適用可能な場合、frame\_num のギャップは復号処理によって検出され、そして必要な数の“非存在”フレームが等式 7-22 の UnusedShortTermFrameNum 値の生成で規定された順序で推定され、そして 8.2.5.2 小節で規定されるようにマーク付けされる。“出力用に不要”かつ“参照不使用”としてマークされたフレームまたは相補的フィールドペアまたは非ペアフィールドを含むフレームバッファは (出力なしで) 空にされ、そして DPB 占有量は空にされたフレームバッファ数だけ減少される。各々の“非存在”フレームは DPB に次の通り保存される。

- 空フレームバッファがない時(すなわち、DPB 占有量が DPB サイズと等しい)、C.4.5.3 小節で規定される“バンピング”処理が、“非存在”フレームをその中に保存する空フレームバッファができるまで繰り返し起動される。
- “非存在”フレームが、空フレームバッファに保存され、“出力用に不要”としてマーク付けされ、そして DPB 占有量が 1 増加される。

#### C.4.3 ピクチャの復号

主符号化ピクチャ n は復号され、そして一時的に (DPB の中ではなく) 保存される。

#### C.4.4 現ピクチャの挿入が起こりうる前のDPBからのピクチャ除去

現ピクチャの挿入が起こりうる前の DPB からのピクチャ除去は、次の通り進む。

- もし復号ピクチャが IDR ピクチャであるならば、次が適用される。
  - 2-9章に規定される復号処理を用いて、付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、8.2.5小節で規定されるように、または、付属資料Gに規定される復号処理を用いて、付属資料Gに規定される1つ以上のプロファイルに適合する符号化ビデオ

シーケンスを復号する時、G.8.2.4小節で規定されるように、DPB内の全ての参照ピクチャは“参照不使用”としてマークされる。

- IDRピクチャが復号された最初のIDRピクチャではなく、アクティブなシーケンスパラメータセットから導出されるPicWidthInMbs、FrameHeightInMbs、あるいはmax\_dec\_frame\_bufferingの値が、先行ピクチャに対してアクティブなシーケンスパラメータセットから導出されるPicWidthInMbs、FrameHeightInMbs、あるいはmax\_dec\_frame\_bufferingの値と各々異なる場合、no\_output\_of\_prior\_pics\_flagの実際の値とは無関係にno\_output\_of\_prior\_pics\_flagはHRDによって1に等しいと推定される。

記 復号器の実装は、PicWidthInMbsまたはFrameHeightInMbsまたはmax\_dec\_frame\_bufferingの値の変更を、HRDよりも行儀良く扱うよう努めるべきである。

- no\_output\_of\_prior\_pics\_flagが1と等しい、あるいは1と等しいと推定される場合、それらが含むピクチャを出力することなくDPB内の全てのフレームバッファは空にされ、DPB占有量は0に設定される。
- それ以外（復号ピクチャがIDRピクチャでない）、2-9章に規定される復号処理を用いて、付属資料Aに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、8.2.5小節で規定されるように、または、付属資料Gに規定される復号処理を用いて、付属資料Gに規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスを復号する時、G.8.2.4小節で規定されるように、復号参照ピクチャマーキング処理が起動される。“出力用に不要”かつ“参照不使用”としてマークされたフレームまたは相補的フィールドペアまたは非ペアフィールドを含むフレームバッファは（出力なしで）空にされ、そしてDPB占有量は空にされたフレームバッファ数だけ減少される。

現ピクチャが、5に等しいmemory\_management\_control\_operationを持つか、no\_output\_of\_prior\_pics\_flagが1に等しくなく1に等しいとも推定されないIDRピクチャである場合、以下に続く2ステップが実行される。

1. “出力用に不要”かつ“参照不使用”としてマークされたフレームまたは相補的フィールドペアまたは非ペアフィールドを含むフレームバッファは（出力なしで）空にされ、そしてDPB占有量は空にされたフレームバッファ数だけ減少される。
2. DPB内の全ての空でないフレームバッファは、C.4.5.3小節で規定される“バンピング”処理を繰り返し起動して空にされ、DPB占有量は0に等しく設定される。

## C.4.5 現復号ピクチャのマーキングと保存

### C.4.5.1 DPBへの参照復号ピクチャの保存とマーキング

現復号ピクチャが参照ピクチャの時、それはDPBに次の通り保存される。

- もし現復号ピクチャが相補的参照フィールドペアの（復号順序で）2番目のフィールドなら、そしてそのペアの1番目のフィールドがまだDPBにあるならば、その現ピクチャはそのペアの1番目のフィールドと同じフレームバッファに保存され、そして次が適用される。
  - もし現復号ピクチャが1に等しいOutputFlagを持つならば、“出力用に必要”としてマーク付けされる。
  - それ以外（現復号ピクチャが0に等しいOutputFlagを持つ）、“出力用に不要”としてマーク付けされる。

- それ以外、次の操作が実施される。
  - 空のフレームバッファがない場合（すなわち、DPB 占有量が DPB サイズに等しい）、C.4.5.3 小小節で規定される“バンピング”処理が、現復号ピクチャをその中に保存する空フレームバッファができるまで繰り返し起動される。
  - 現復号ピクチャが、空のフレームバッファに保存され、DPB 占有量が 1 増加され、そして次が適用される。
    - もし現復号ピクチャが1に等しいOutputFlagを持つならば、“出力用に必要”としてマーク付けされる。
    - それ以外（現復号ピクチャが0に等しいOutputFlagを持つ）、“出力用に不要”としてマーク付けされる。

#### C.4.5.2 DPBへの非参照復号ピクチャの保存とマーキング

現ピクチャが非参照ピクチャの時、次の操作が実施される。

- もし現復号ピクチャが相補的非参照フィールドペアの（復号順序で）2番目のフィールドなら、そしてそのペアの1番目のフィールドがまだDPBにあるならば、その現ピクチャはそのペアの1番目のフィールドと同じフレームバッファに保存され、そして次が適用される。
  - もし現復号ピクチャが 1 に等しい OutputFlag を持つならば、“出力用に必要”としてマーク付けされる。
  - それ以外（現復号ピクチャが0に等しいOutputFlagを持つ）、“出力用に不要”としてマーク付けされる。
- それ以外、もし現ピクチャが0に等しいOutputFlagを持つならば、DPBは変更されず、現ピクチャは出力されない。
- それ以外、現復号ピクチャが切り出され出力され終える、またはDPBに保存され終えるまで、次の操作が繰り返し実施される。
  - もし空のフレームバッファがない（すなわち、DPB 占有量が DPB サイズと等しい）ならば、次が適用される。
    - もし現ピクチャが、“出力用に必要”とマーク付けされたDPB内の全てのピクチャに対して低いPicOrderCnt( )値を持たないならば、C.4.5.3小小節で述べられる“バンピング”処理が実施される。
    - それ以外（現ピクチャが、“出力用に必要”とマーク付けされたDPB内の全てのピクチャに対して低いPicOrderCnt( )値を持つ）、現ピクチャはそのピクチャに対するアクティブシーケンスパラメータセットで規定される切り出し矩形を使って切り出され、切り出されたピクチャが出力される。
  - それ以外（空のフレームバッファがある、すなわち、DPB 占有量が DPB サイズより少ない）、現復号ピクチャが、空のフレームバッファに保存され、“出力用に必要”としてマーク付けされ、そして DPB 占有量が 1 増加される。

### C.4.5.3 “バンピング” 処理

“バンピング” 処理は次の場合に起動される。

- 空フレームバッファがなく（すなわち、DPB占有量がDPBサイズと等しい）、C.4.2小節で規定されるように、空フレームバッファが推定される“非存在”フレーム保存のため必要になる。
- 現ピクチャがIDRピクチャであり、C.4.4小節で規定されるように、no\_output\_of\_prior\_pics\_flagが1に等しくはなく、かつ1に等しいと推定されない。
- 現ピクチャが、C.4.4小節で規定されるように、5に等しいmemory\_management\_operationを持っている。
- 空フレームバッファがなく（すなわち、DPB占有量がDPBサイズと等しい）、C.4.5.1小節で規定されるように、空フレームバッファが(非IDR)復号参照ピクチャ保存のため必要になる。
- 空フレームバッファがなく（すなわち、DPB占有量がDPBサイズと等しい）、C.4.5.2小節で規定されるように、現ピクチャが、相補的非参照フィールドペアの第2フィールドではない非参照ピクチャであり、そして現ピクチャが1に等しいOutputFlagを持ち、そして出力順序で現非参照ピクチャに先行する“出力用に必要”としてマーク付けされたピクチャがDPBにあり、そのため、空フレームバッファが現ピクチャ保存のため必要になる。

“バンピング” 処理は次からなる。

- 出力における最初のピクチャまたは相補的参照フィールドペアが次の通り選択される。
  - “出力用に必要”としてマーク付けされたDPB内の全てのピクチャで最も小さいPicOrderCnt()の値をもつピクチャを含むフレームバッファが選択される。
  - もし、このフレームバッファが両フィールドとも“出力用に必要”としてマーク付けられた相補的非参照フィールドペアを含み、かつ両フィールドが同じPicOrderCnt()をもつならば、復号順序でこれら2フィールドの1番目が出力における最初だと考えられる。
  - それ以外、もし、このフレームバッファが両フィールドとも“出力用に必要”としてマーク付けられた相補的参照フィールドペアを含み、かつ両フィールドが同じPicOrderCnt()をもつならば、相補的参照フィールドペア全体が出力における最初だと考えられる。
  - それ以外、このフレームバッファでPicOrderCnt()の最も小さい値をもつピクチャが出力における最初だと考えられる。
- もし単一ピクチャが出力における最初だと考えられるならば、そのピクチャは、そのピクチャに対するアクティブシーケンスパラメータセットで規定される切り出し矩形を使って切り出され、その切り出されたピクチャが出力され、そしてそのピクチャは“出力用に不要”としてマークされる。
- それ以外（相補的参照フィールドペアが出力における最初だと考えられる）、相補的参照フィールドペアの2フィールドが両方ともそのピクチャに対するアクティブシーケンスパラメータセットで規定される切り出し矩形を使って切り出され、相補的参照フィールドペアの2フィールドがともに出力され、そして相補的参照フィールドペアの両フィールドは“出力用に不要”としてマークされる。
- 切り出され出力されたピクチャまたは相補的参照フィールドペアを含んでいたフレームバッファが検査され、もし以下の条件のいずれかが満たされるならば、そのフレームバッファは空にされ、DPB占有量は1減少される。

- フレームバッファが、非ペア非参照フィールドを含んでいる。
- フレームバッファが、非参照フレームを含んでいる。
- フレームバッファが、両方のフィールドが“出力用に不要”とマークされた相補的非参照フィールドペアを含んでいる。
- フレームバッファが、“参照不使用”としてマークされた非ペア参照フィールドを含んでいる。
- フレームバッファが、両方のフィールドが“参照不使用”としてマークされた参照フレームを含んでいる。
- フレームバッファが、両方のフィールドが“参照不使用”および“出力用に不要”としてマークされた相補的参照フィールドペアを含んでいる。

## 付属資料D 付加拡張情報

(この付属資料は、本標準の必須部分を形成する)

この付属資料では SEI メッセージペイロードに関するシンタックスと意味を規定する。

SEI メッセージは復号、表示、または他の目的に関する処理を支援する。しかし、SEI メッセージは復号処理で輝度や色差サンプルを構築するために必要とはされない。適合した復号器で、本標準への出力順序適合(適合性に関する規定については付属資料 C を参照)のためにこの情報を処理することは必要とされない。いくつかの SEI メッセージ情報はビットストリーム適合性や出力タイミングに対する復号器適合性を確認するために必要とされる。

付属資料 D において、SEI メッセージの存在に関する規定は、これらのメッセージ(またはそれらのいくつかのサブセット)が本標準によって規定されない他の手段によって、復号器(または HRD)に伝達される場合にも、満たされる。ビットストリームに存在する場合、SEI メッセージは 7.3.2.3 小小節と 7.4.2.3 小小節とこの付属資料において規定されたシンタックスと意味に従わなければならない。SEI メッセージの内容が、ビットストリーム内に存在すること以外のある手段により、アプリケーションのために伝達される場合、その SEI メッセージ内容の表現は、この付属資料において規定される同じシンタックスの使用は要求されない。ビットをカウントする目的では、実際にビットストリーム中に存在する適切なビットのみがカウントされる。

### D.1 SEIペイロードシンタックス

sei_payload( payloadType, payloadSize ) {	C	記述子
if( payloadType == 0 )		
buffering_period( payloadSize )	5	
else if( payloadType == 1 )		
pic_timing( payloadSize )	5	
else if( payloadType == 2 )		
pan_scan_rect( payloadSize )	5	
else if( payloadType == 3 )		
filler_payload( payloadSize )	5	
else if( payloadType == 4 )		
user_data_registered_itu_t_t35( payloadSize )	5	
else if( payloadType == 5 )		
user_data_unregistered( payloadSize )	5	
else if( payloadType == 6 )		
recovery_point( payloadSize )	5	
else if( payloadType == 7 )		
dec_ref_pic_marking_repetition( payloadSize )	5	
else if( payloadType == 8 )		
spare_pic( payloadSize )	5	
else if( payloadType == 9 )		
scene_info( payloadSize )	5	

else if( payloadType == 10 )		
sub_seq_info( payloadSize )	5	
else if( payloadType == 11 )		
sub_seq_layer_characteristics( payloadSize )	5	
else if( payloadType == 12 )		
sub_seq_characteristics( payloadSize )	5	
else if( payloadType == 13 )		
full_frame_freeze( payloadSize )	5	
else if( payloadType == 14 )		
full_frame_freeze_release( payloadSize )	5	
else if( payloadType == 15 )		
full_frame_snapshot( payloadSize )	5	
else if( payloadType == 16 )		
progressive_refinement_segment_start( payloadSize )	5	
else if( payloadType == 17 )		
progressive_refinement_segment_end( payloadSize )	5	
else if( payloadType == 18 )		
motion_constrained_slice_group_set( payloadSize )	5	
else if( payloadType == 19 )		
film_grain_characteristics( payloadSize )	5	
else if( payloadType == 20 )		
deblocking_filter_display_preference( payloadSize )	5	
else if( payloadType == 21 )		
stereo_video_info( payloadSize )	5	
else if( payloadType == 22 )		
post_filter_hint( payloadSize )	5	
else if( payloadType == 23 )		
tone_mapping_info( payloadSize )	5	
else if( payloadType == 24 )		
scalability_info( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 25 )		
sub_pic_scalable_layer( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 26 )		
non_required_layer_rep( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 27 )		
priority_layer_info( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 28 )		
layers_not_present( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 29 )		
layer_dependency_change( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 30 )		
scalable_nesting( payloadSize ) /* 付属資料Gにて規定 */	5	

else if( payloadType == 31 )		
base_layer_temporal_hrd( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 32 )		
quality_layer_integrity_check( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 33 )		
redundant_pic_property( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 34 )		
tl0_picture_index( payloadSize ) /* 付属資料Gにて規定 */	5	
else if( payloadType == 35 )		
tl_switching_point( payloadSize ) /* 付属資料Gにて規定 */	5	
else		
reserved_sei_message( payloadSize )	5	
if( !byte_aligned() ) {		
<b>bit_equal_to_one</b> /* 1に等しい */	5	f(1)
while( !byte_aligned() )		
<b>bit_equal_to_zero</b> /* 0に等しい */	5	f(1)
}		
}		

#### D.1.1 バッファリング期間SEIメッセージシンタックス

buffering_period( payloadSize ) {	<b>C</b>	記述子
<b>seq_parameter_set_id</b>	5	ue(v)
if( NalHrdBpPresentFlag ) {		
for( SchedSelIdx = 0; SchedSelIdx <=cpb_cnt_minus1; SchedSelIdx++ ) {		
<b>initial_cpb_removal_delay</b> [ SchedSelIdx ]	5	u(v)
<b>initial_cpb_removal_delay_offset</b> [ SchedSelIdx ]	5	u(v)
}		
}		
if( VclHrdBpPresentFlag ) {		
for( SchedSelIdx = 0; SchedSelIdx <=cpb_cnt_minus1; SchedSelIdx++ ) {		
<b>initial_cpb_removal_delay</b> [ SchedSelIdx ]	5	u(v)
<b>initial_cpb_removal_delay_offset</b> [ SchedSelIdx ]	5	u(v)
}		
}		
}		

#### D.1.2 ピクチャタイミングSEIメッセージシンタックス

pic_timing( payloadSize ) {	<b>C</b>	記述子
-----------------------------	----------	-----

if( CpbDpbDelaysPresentFlag ) {		
<b>cpb_removal_delay</b>	5	u(v)
<b>dpb_output_delay</b>	5	u(v)
}		
if( pic_struct_present_flag ) {		
<b>pic_struct</b>	5	u(4)
for( i = 0; i < NumClockTS ; i++ ) {		
<b>clock_timestamp_flag[ i ]</b>	5	u(1)
if( clock_timestamp_flag[i] ) {		
<b>ct_type</b>	5	u(2)
<b>nuit_field_based_flag</b>	5	u(1)
<b>counting_type</b>	5	u(5)
<b>full_timestamp_flag</b>	5	u(1)
<b>discontinuity_flag</b>	5	u(1)
<b>cnt_droppeded_flag</b>	5	u(1)
<b>n_frames</b>	5	u(8)
if( full_timestamp_flag ) {		
<b>seconds_value</b> /* 0..59 */	5	u(6)
<b>minutes_value</b> /* 0..59 */	5	u(6)
<b>hours_value</b> /* 0..23 */	5	u(5)
} else {		
<b>seconds_flag</b>	5	u(1)
if( seconds_flag ) {		
<b>seconds_value</b> /* 範圍0..59 */	5	u(6)
<b>minutes_flag</b>	5	u(1)
if( minutes_flag ) {		
<b>minutes_value</b> /* 0..59 */	5	u(6)
<b>hours_flag</b>	5	u(1)
if( hours_flag )		
<b>hours_value</b> /* 0..23 */	5	u(5)
}		
}		
}		
}		
if( time_offset_length > 0 )		
<b>time_offset</b>	5	i(v)
}		
}		
}		
}		

### D.1.3 パン・スキャン矩形SEIメッセージシンタックス

pan_scan_rect( payloadSize ) {	<b>C</b>	記述子
<b>pan_scan_rect_id</b>	5	ue(v)
<b>pan_scan_rect_cancel_flag</b>	5	u(1)
if( !pan_scan_rect_cancel_flag ) {		
<b>pan_scan_cnt_minus1</b>	5	ue(v)
for( i = 0; i <= pan_scan_cnt_minus1; i++ ) {		
<b>pan_scan_rect_left_offset[ i ]</b>	5	se(v)
<b>pan_scan_rect_right_offset[ i ]</b>	5	se(v)
<b>pan_scan_rect_top_offset[ i ]</b>	5	se(v)
<b>pan_scan_rect_bottom_offset[ i ]</b>	5	se(v)
}		
<b>pan_scan_rect_repetition_period</b>	5	ue(v)
}		
}		

### D.1.4 フィルペイロードSEIメッセージシンタックス

filler_payload( payloadSize ) {	<b>C</b>	記述子
for( k = 0; k < payloadSize; k++ )		
<b>ff_byte</b> /* 0xFFに等しい*/	5	f(8)
}		

### D.1.5 ITU-T勧告T.35による登録ユーザデータSEIメッセージシンタックス

user_data_registered_itu_t_t35( payloadSize ) {	<b>C</b>	記述子
<b>itu_t_t35_country_code</b>	5	b(8)
if( itu_t_t35_country_code != 0xFF )		
i = 1		
else {		
<b>itu_t_t35_country_code_extension_byte</b>	5	b(8)
i = 2		
}		
do {		
<b>itu_t_t35_payload_byte</b>	5	b(8)
i++		
} while( i < payloadSize )		
}		

D.1.6 非登録ユーザデータSEIメッセージシンタックス

user_data_unregistered( payloadSize ) {	C	記述子
<b>uuid_iso_iec_11578</b>	5	u(128)
for( i = 16; i < payloadSize; i++ )		
<b>user_data_payload_byte</b>	5	b(8)
}		

D.1.7 復旧点SEIメッセージシンタックス

recovery_point( payloadSize ) {	C	記述子
<b>recovery_frame_cnt</b>	5	ue(v)
<b>exact_match_flag</b>	5	u(1)
<b>broken_link_flag</b>	5	u(1)
<b>changing_slice_group_idc</b>	5	u(2)
}		

D.1.8 復号参照ピクチャマーキング繰り返しSEIメッセージシンタックス

dec_ref_pic_marking_repetition( payloadSize ) {	C	記述子
<b>original_idr_flag</b>	5	u(1)
<b>original_frame_num</b>	5	ue(v)
if( !frame_mbs_only_flag ) {		
<b>original_field_pic_flag</b>	5	u(1)
if( original_field_pic_flag )		
<b>original_bottom_field_flag</b>	5	u(1)
}		
dec_ref_pic_marking( )	5	
}		

#### D.1.9 予備ピクチャSEIメッセージシンタックス

spare_pic( payloadSize ) {	C	記述子
<b>target_frame_num</b>	5	ue(v)
<b>spare_field_flag</b>	5	u(1)
if( spare_field_flag )		
<b>target_bottom_field_flag</b>	5	u(1)
<b>num_spare_pics_minus1</b>	5	ue(v)
for( i = 0; i < num_spare_pics_minus1 + 1; i++ ) {		
<b>delta_spare_frame_num[ i ]</b>	5	ue(v)
if( spare_field_flag )		
<b>spare_bottom_field_flag[ i ]</b>	5	u(1)
<b>spare_area_idc[ i ]</b>	5	ue(v)
if( spare_area_idc[ i ] == 1 )		
for( j = 0; j < PicSizeInMapUnits; j++ )		
<b>spare_unit_flag[ i ][ j ]</b>	5	u(1)
else if( spare_area_idc[ i ] == 2 ) {		
mapUnitCnt = 0		
for( j=0; mapUnitCnt < PicSizeInMapUnits; j++ ) {		
<b>zero_run_length[ i ][ j ]</b>	5	ue(v)
mapUnitCnt += zero_run_length[ i ][ j ] + 1		
}		
}		
}		
}		

#### D.1.10 シーン情報SEIメッセージシンタックス

scene_info( payloadSize ) {	C	記述子
<b>scene_info_present_flag</b>	5	u(1)
if( scene_info_present_flag ) {		
<b>scene_id</b>	5	ue(v)
<b>scene_transition_type</b>	5	ue(v)
if( scene_transition_type > 3 )		
<b>second_scene_id</b>	5	ue(v)
}		
}		

D.1.11 サブシーケンス情報SEIメッセージシンタックス

sub_seq_info( payloadSize ) {	<b>C</b>	記述子
<b>sub_seq_layer_num</b>	5	ue(v)
<b>sub_seq_id</b>	5	ue(v)
<b>first_ref_pic_flag</b>	5	u(1)
<b>leading_non_ref_pic_flag</b>	5	u(1)
<b>last_pic_flag</b>	5	u(1)
<b>sub_seq_frame_num_flag</b>	5	u(1)
if( sub_seq_frame_num_flag )		
<b>sub_seq_frame_num</b>	5	ue(v)
}		

D.1.12 サブシーケンスレイヤ特性SEIメッセージシンタックス

sub_seq_layer_characteristics( payloadSize ) {	<b>C</b>	記述子
<b>num_sub_seq_layers_minus1</b>	5	ue(v)
for( layer = 0; layer <= num_sub_seq_layers_minus1; layer++ ) {		
<b>accurate_statistics_flag</b>	5	u(1)
<b>average_bit_rate</b>	5	u(16)
<b>average_frame_rate</b>	5	u(16)
}		
}		

D.1.13 サブシーケンス特性SEIメッセージシンタックス

sub_seq_characteristics( payloadSize ) {	<b>C</b>	記述子
<b>sub_seq_layer_num</b>	5	ue(v)
<b>sub_seq_id</b>	5	ue(v)
<b>duration_flag</b>	5	u(1)
if( duration_flag)		
<b>sub_seq_duration</b>	5	u(32)
<b>average_rate_flag</b>	5	u(1)
if( average_rate_flag ) {		
<b>accurate_statistics_flag</b>	5	u(1)
<b>average_bit_rate</b>	5	u(16)
<b>average_frame_rate</b>	5	u(16)
}		
<b>num_referenced_subseqs</b>	5	ue(v)
for( n = 0; n < num_referenced_subseqs; n++ ) {		
<b>ref_sub_seq_layer_num</b>	5	ue(v)
<b>ref_sub_seq_id</b>	5	ue(v)
<b>ref_sub_seq_direction</b>	5	u(1)
}		
}		

D.1.14 フルフレーム凍結SEIメッセージシンタックス

full_frame_freeze( payloadSize ) {	<b>C</b>	記述子
<b>full_frame_freeze_repetition_period</b>	5	ue(v)
}		

D.1.15 フルフレーム凍結解除SEIメッセージシンタックス

full_frame_freeze_release( payloadSize ) {	<b>C</b>	記述子
}		

D.1.16 フルフレームスナップショットSEIメッセージシンタックス

full_frame_snapshot( payloadSize ) {	<b>C</b>	記述子
<b>snapshot_id</b>	5	ue(v)
}		

D.1.17 プログレッシブリファインメント区分開始SEIメッセージシンタックス

progressive_refinement_segment_start( payloadSize ) {	C	記述子
<b>progressive_refinement_id</b>	5	ue(v)
<b>num_refinement_steps_minus1</b>	5	ue(v)
}		

D.1.18 プログレッシブリファインメント区分終了SEIメッセージシンタックス

progressive_refinement_segment_end( payloadSize ) {	C	記述子
<b>progressive_refinement_id</b>	5	ue(v)
}		

D.1.19 動き制約スライスグループセットSEIメッセージシンタックス

motion_constrained_slice_group_set( payloadSize ) {	C	記述子
<b>num_slice_groups_in_set_minus1</b>	5	ue(v)
for( i = 0; i <= num_slice_groups_in_set_minus1; i++)		
<b>slice_group_id[ i ]</b>	5	u(v)
<b>exact_sample_value_match_flag</b>	5	u(1)
<b>pan_scan_rect_flag</b>	5	u(1)
if( pan_scan_rect_flag )		
<b>pan_scan_rect_id</b>	5	ue(v)
}		

D.1.20 フィルムグレイン特性SEIメッセージシンタックス

film_grain_characteristics( payloadSize ) {	C	記述子
<b>film_grain_characteristics_cancel_flag</b>	5	u(1)
if( !film_grain_characteristics_cancel_flag ) {		
<b>model_id</b>	5	u(2)
<b>separate_colour_description_present_flag</b>	5	u(1)
if( separate_colour_description_present_flag ) {		
<b>film_grain_bit_depth_luma_minus8</b>	5	u(3)
<b>film_grain_bit_depth_chroma_minus8</b>	5	u(3)
<b>film_grain_full_range_flag</b>	5	u(1)
<b>film_grain_colour primaries</b>	5	u(8)
<b>film_grain_transfer_characteristics</b>	5	u(8)
<b>film_grain_matrix_coefficients</b>	5	u(8)
}		
}		

<b>blending_mode_id</b>	5	u(2)
<b>log2_scale_factor</b>	5	u(4)
for( c = 0; c < 3; c++ )		
<b>comp_model_present_flag[ c ]</b>	5	u(1)
for( c = 0; c < 3; c++ )		
if( comp_model_present_flag[ c ] ) {		
<b>num_intensity_intervals_minus1[ c ]</b>	5	u(8)
<b>num_model_values_minus1[ c ]</b>	5	u(3)
for( i = 0; i <= num_intensity_intervals_minus1[ c ]; i++ ) {		
<b>intensity_interval_lower_bound[ c ][ i ]</b>	5	u(8)
<b>intensity_interval_upper_bound[ c ][ i ]</b>	5	u(8)
for( j = 0; j <= num_model_values_minus1[ c ]; j++ )		
<b>comp_model_value[ c ][ i ][ j ]</b>	5	se(v)
}		
}		
<b>film_grain_characteristics_repetition_period</b>	5	ue(v)
}		
}		

#### D.1.21 デブロッキングフィルタ表示選好SEIメッセージシンタックス

deblocking_filter_display_preference( payloadSize ) {	<b>C</b>	記述子
<b>deblocking_display_preference_cancel_flag</b>	5	u(1)
if( !deblocking_display_preference_cancel_flag ) {		
<b>display_prior_to_deblocking_preferred_flag</b>	5	u(1)
<b>dec_frame_buffering_constraint_flag</b>	5	u(1)
<b>deblocking_display_preference_repetition_period</b>	5	ue(v)
}		
}		

D.1.22 ステレオビデオ情報SEIメッセージシンタックス

stereo_video_info( payloadSize ) {	<b>C</b>	<b>記述子</b>
<b>field_views_flag</b>	5	u(1)
if( field_views_flag )		
<b>top_field_is_left_view_flag</b>	5	u(1)
else {		
<b>current_frame_is_left_view_flag</b>	5	u(1)
<b>next_frame_is_second_view_flag</b>	5	u(1)
}		
<b>left_view_self_contained_flag</b>	5	u(1)
<b>right_view_self_contained_flag</b>	5	u(1)
}		

D.1.23 ポストフィルタヒントSEIメッセージシンタックス

post_filter_hint( payloadSize ) {	<b>C</b>	<b>記述子</b>
<b>filter_hint_size_y</b>	5	ue(v)
<b>filter_hint_size_x</b>	5	ue(v)
<b>filter_hint_type</b>	5	u(2)
for( colour_component = 0; colour_component < 3; colour_component ++ )		
for( cy = 0; cy < filter_hint_size_y; cy ++ )		
for( cx = 0; cx < filter_hint_size_x; cx ++ )		
<b>filter_hint[ colour_component ][ cy ][ cx ]</b>	5	se(v)
<b>additional_extension_flag</b>	5	u(1)
}		

D.1.24 トーンマッピング情報SEIメッセージシンタックス

tone_mapping_info( payloadSize ) {	<b>C</b>	記述子
<b>tone_map_id</b>	5	ue(v)
<b>tone_map_cancel_flag</b>	5	u(1)
if( !tone_map_cancel_flag ) {		
<b>tone_map_repetition_period</b>	5	ue(v)
<b>coded_data_bit_depth</b>	5	u(8)
<b>target_bit_depth</b>	5	u(8)
<b>model_id</b>	5	ue(v)
if( model_id == 0 ) {		
<b>min_value</b>	5	u(32)
<b>max_value</b>	5	u(32)
}		
if( model_id == 1 ) {		
<b>sigmoid_midpoint</b>	5	u(32)
<b>sigmoid_width</b>	5	u(32)
}		
if( model_id == 2 )		
for( i = 0; i < ( 1 << target_bit_depth ); i++ )		
<b>start_of_coded_interval[ i ]</b>	5	u(v)
if( model_id == 3 ) {		
<b>num_pivots</b>	5	u(16)
for( i=0; i < num_pivots; i++ ) {		
<b>coded_pivot_value[ i ]</b>	5	u(v)
<b>target_pivot_value[ i ]</b>	5	u(v)
}		
}		
}		
}		
}		

#### D.1.25 予約SEIメッセージシンタックス

reserved_sei_message( payloadSize ) {	<b>C</b>	記述子
for( i = 0; i < payloadSize; i++ )		
<b>reserved_sei_message_payload_byte</b>	5	b(8)
}		

## D.2 SEIペイロードの意味

### D.2.1 バッファリング期間SEIメッセージの意味

NalHrdBpPresentFlag または VclHrdBpPresentFlag が 1 に等しい時、バッファリング期間 SEI メッセージは、ビットストリーム中の任意のアクセスユニットに関連付けされることができ、かつ、バッファリング期間 SEI メッセージは、それぞれの IDR アクセスユニットと、復旧点 SEI メッセージに関連付けられたそれぞれのアクセスユニットに関連付けされなければならない。

記 いくつかのアプリケーションにおいては、バッファリング期間SEIメッセージの頻繁な存在が望ましいかもしれない。

バッファリング期間は、復号順序において、バッファリング期間 SEI メッセージの 2 個の実体の間のアクセスユニットの集合として規定される。

**seq\_parameter\_set\_id** は、シーケンス HRD 属性を含むシーケンスパラメータセットを規定する。seq\_parameter\_set\_id の値はバッファリング期間 SEI メッセージに関連付けられた主符号化ピクチャによって参照されるピクチャパラメータセット中の seq\_parameter\_set\_id の値と等しくなければならない。seq\_parameter\_set\_id の値は、0 から 31 のそれぞれを含む範囲でなければならない。

**initial\_cpb\_removal\_delay**[SchedSelIdx]は、HRD 初期化後最初のバッファリング期間に対する、SchedSelIdx 番目の CPB の、バッファリング期間 SEI メッセージに関連付けられたアクセスユニットに関連付けられた符号化データの最初のビットの CPB への到達時刻と、同じアクセスユニットに関連付けられた符号化データの CPB からの除去時刻との間の遅延を規定する。このシンタックス要素は、initial\_cpb\_removal\_delay\_length\_minus1+1 で与えられるビット長を有する。これは 90kHz クロック単位である。initial\_cpb\_removal\_delay[SchedSelIdx]は 0 に等しくはならず、90kHz クロック単位で CPB サイズと等価な時間  $90000 * (\text{CpbSize}[\text{SchedSelIdx}] \div \text{BitRate}[\text{SchedSelIdx}])$  を越えてもならない。

**initial\_cpb\_removal\_delay\_offset**[SchedSelIdx]は、SchedSelIdx 番目の CPB に対し、符号化アクセスユニットの CPB への初期配達時刻を規定するために、cpb\_removal\_delay と組み合わせて使用される。initial\_cpb\_removal\_delay\_offset[SchedSelIdx] は、90kHz クロック単位である。initial\_cpb\_removal\_delay\_offset[SchedSelIdx] シンタックス要素は、ビット長が initial\_cpb\_removal\_delay\_length\_minus1+1 で与えられる固定長符号である。このシンタックス要素は、復号器によって使用されるのではなく、付属資料 C において規定されている配達スケジューラ(HSS)のみで必要とされる。

符号化ビデオシーケンス全体を通して、それぞれの SchedSelIdx 値において initial\_cpb\_removal\_delay[SchedSelIdx]と initial\_cpb\_removal\_delay\_offset[SchedSelIdx]の和は一定でなければならない。

### D.2.2 ピクチャタイミングSEIメッセージの意味

ピクチャタイミング SEI メッセージのビットストリーム中での存在は次の通り規定される。

- もし CpbDpbDelaysPresentFlag が 1 に等しい、または pic\_struct\_present\_flag が 1 に等しいならば、符号化ビデオシーケンスのアクセスユニット毎に 1 つのピクチャタイミング SEI メッセージが存在しなくてはならない。
- それ以外 (CpbDpbDelaysPresentFlag が 0 に等しい、かつ pic\_struct\_present\_flag が 0 に等しい)、符号化ビデオシーケンスの任意のアクセスユニットにピクチャタイミング SEI メッセージが存在してはならな

い。

**cpb\_removal\_delay** は、最も新しいバッファリング期間 SEI メッセージに関連付けられたアクセスユニットの CPB からの除去の後、そのピクチャタイミング SEI メッセージに関連付けられたアクセスユニットデータをバッファから除去する前に、何クロックチック (E.2.1 小節参照) 待つのかを規定する。付属資料 C に規定されるように、この値は、HSS に対する、アクセスユニットデータの CPB への最も早い到達可能時刻を計算するためにも用いられる。このシンタックス要素は、**cpb\_removal\_delay\_length\_minus1+1** によって与えられるビット長の固定長符号である。**cpb\_removal\_delay** は、 $2^{(\text{cpb\_removal\_delay\_length\_minus1}+1)}$  カウンタの剰余である。

ビットストリーム中で最初のピクチャに対する **cpb\_removal\_delay** の値は 0 でなければならない。

**dpb\_output\_delay** は、ピクチャの DPB 出力時刻を計算するために使用される。これは、アクセスユニットの CPB からの除去後、復号ピクチャが DPB から出力されるまでに、何クロックチック待つのかを規定する (C.2 節参照)。

記1 “短期間参照使用” または “長期間参照使用” としてまだマークされているならば、そのピクチャは出力時刻になっても DPB からは除去されない。

記2 1個の復号ピクチャに対し、ただ1個の **dpb\_output\_delay** が規定される。

シンタックス要素 **dpb\_output\_delay** のサイズは、ビットで **dpb\_output\_delay\_length\_minus1+1** で与えられる。**max\_dec\_frame\_buffering** が 0 に等しい時、**dpb\_output\_delay** は 0 に等しくなければならない。

C.2 節で規定されるような出力タイミング適合復号器から出力される任意のピクチャの **dpb\_output\_delay** から導出される出力時刻は、復号順序で続く任意の符号化ビデオシーケンスの全てのピクチャの **dpb\_output\_delay** から導出される出力時刻に先行しなくてはならない。

相補的非参照フィールドペアの復号順序で 2 番目のフィールドの **dpb\_output\_delay** から導出される出力時刻は、同じ相補的非参照フィールドペアの初めのフィールドの **dpb\_output\_delay** から導出される出力時刻を超えなくてはならない。

このシンタックス要素の値で確立されるピクチャ出力順序は、相補的参照フィールドペアの 2 つのフィールドが同じ **PicOrderCnt()** の値を持ち、その 2 つのフィールドが異なった出力時刻を持つ場合を除いて、C.4.1 から C.4.5 小節で規定されるような **PicOrderCnt()** の値から確立される順序と同じでなければならない。

復号順序で **no\_output\_of\_prior\_pics\_flag** が 1 に等しいあるいは 1 に等しいと推定されるある IDR ピクチャに先行するため C.4.5 小節の “バンピング” 処理により出力されないピクチャに対して、**dpb\_output\_delay** から導出される出力時刻は、5 に等しい **memory\_management\_control\_operation** を持つ任意のピクチャに続く同じ符号化ビデオシーケンス内の全てのピクチャに対して **PicOrderCnt()** の値の増加に伴って増加してはならない。

**pic\_struct** は、ピクチャが、フレームまたは 1 つ以上のフィールドとして表示されるべきかを付表 D-1/JT-H264 に従って示す。フレーム 2 重化(7 に等しい **pic\_struct**)は、そのフレームは 2 度続けて表示されなければならないことを示し、フレーム 3 重化(8 に等しい **pic\_struct**)は、そのフレームは 3 度続けて表示されなければならないことを示す。

記3 フレーム2重化は、例えば50p表示装置上の25pビデオや59.94p表示装置上の29.97pビデオの表示を容易にできる。フレーム1つおきに、フレーム2重化とフレーム3重化の組み合わせを用いることは、59.94p表示装置上の23.98pビデオの表示を容易にできる。

付表D-1/JT-H264 pic\_structの解釈  
(ITU-T H.264)

値	指示されたピクチャの表示	制限	NumClockTS
0	フレーム	field_pic_flag は 0 でなければならない	1
1	トップフィールド	field_pic_flag は 1 でなければならない bottom_field_flag は 0 でなければならない	1
2	ボトムフィールド	field_pic_flag は 1 でなければならない bottom_field_flag は 1 でなければならない	1
3	トップフィールド、ボトムフィールドの順序	field_pic_flag は 0 でなければならない	2
4	ボトムフィールド、トップフィールドの順序	field_pic_flag は 0 でなければならない	2
5	トップフィールド、ボトムフィールド、繰り返しトップフィールドの順序	field_pic_flag は 0 でなければならない	3
6	ボトムフィールド、トップフィールド、繰り返しボトムフィールドの順序	field_pic_flag は 0 でなければならない	3
7	フレーム 2 重化	field_pic_flag は 0 でなければならない fixed_frame_rate_flag は 1 でなければならない	2
8	フレーム 3 重化	field_pic_flag は 0 でなければならない fixed_frame_rate_flag は 1 でなければならない	3
9..15	予約		

NumClockTS は、付表 D-1/JT-H264 において規定されるように、pic\_struct によって決定される。それぞれのセットに対して clock\_timestamp\_flag[i] によって規定されるように、1つのピクチャに対して NumClockTS 個までのクロックタイムスタンプ情報のセットが存在する。クロックタイムスタンプ情報のセットは、pic\_struct によってピクチャに関連付けられたフィールドやフレームに適用される。

クロックタイムスタンプシンタックス要素の内容は、元の、またはキャプチャの、または代わりとなる理想表示の時刻を示す。この示された時刻は、クロック周波数が time\_scale Hz に等しいクロックのクロックチック単位で、clockTimestamp が 0 に等しい無規定のある時刻との相対で、以下のように計算される。

$$\text{clockTimestamp} = ((\text{hH} * 60 + \text{mM}) * 60 + \text{sS}) * \text{time\_scale} + \text{nFrames} * (\text{num\_units\_in\_tick} * (1 + \text{nuit\_field\_based\_flag})) + \text{tOffset} \quad (\text{D-1})$$

出力順序と DPB 出力タイミングは clockTimestamp の値によって影響を受けない。pic\_struct が 0 に等しい 2 個以上のフレームが出力順序で連続していて、かつ等しい clockTimestamp の値を持つ場合、その複数フレームは同じ内容を表し、出力順序でそのような最後のフレームが、より好ましい表現であることを示す。

記4 clockTimestamp時刻指示は、DPB出力時刻に良く合致した以外のリフレッシュレートをもつ装置上の表示を支援するかもしれない。

clock\_timestamp\_flag[i] が 1 であることは、いくつかのクロックタイムスタンプシンタックス要素が存在し、

直後に続くことを示す。clock\_timestamp\_flag[i]が0であることは、関連付けられたクロックスタンプシンタックス要素が存在しないことを示す。NumClockTS が1よりも大きく、かつ、1よりも大きいiの値に対しclock\_timestamp\_flag[i]が1である場合、iの値の増加に対してclockTimestampの値は非減少でなければならない。

ct\_typeは、付表D-2/JT-H264に規定されるように、原素材の走査タイプ（インタレースまたはプログレッシブ）を示す。

符号化フレームの2個のフィールドは異なるct\_typeの値を持つてもよい。

両方のct\_typeが0（プログレッシブ）またはct\_typeが2（不明）であり、出力順序で連続した、逆パリティの2個のフィールドに対するclockTimestampが等しい場合、その2フィールドは同じ元のプログレッシブフレームから来たことを示している。どちらかのフィールドに対するct\_typeの値が1（インタレース）であるならば、出力順序で連続した2個のフィールドは異なるclockTimestampの値を持たなければならない。

付表D-2/JT-H264 ct\_typeの原ピクチャ走査への割り当て  
(ITU-T H.264)

値	原ピクチャ走査
0	プログレッシブ
1	インタレース
2	不明
3	予約

nuit\_field\_based\_flag は、等式 D-1 で規定されるように、clockTimestamp を計算する際に使用される。

counting\_type は、付表 D-3/JT-H264 で規定されるように、n\_frames の値のドロップ方法を規定する。

付表D-3/JT-H264 counting\_type値の定義  
(ITU-T H.264)

値	解釈
0	n_frames カウント値をドロップせず、time_offset を使用しない
1	n_frames カウント値をドロップしない
2	n_frames カウントの個々の 0 値をドロップする
3	n_frames カウントの個々の MaxFPS-1 値をドロップする
4	seconds_value が 0 であり、minutes_value が 10 の整数倍でない場合、最も小さい 2 個の n_frames カウント (0 と 1) をドロップする
5	個々の無規定 n_frames カウント値をドロップする
6	無規定数の無規定 n_frames カウント値をドロップする
7..31	予約

full\_timestamp\_flag が 1 であることは、n\_frames シンタックス要素の後に seconds\_value と minutes\_value と hours\_value が続くことを規定する。full\_timestamp\_flag が 0 であることは、n\_frames シンタックス要素の後に seconds\_flag が続くことを規定する。

discontinuity\_flag が 0 であることは、現 clockTimestamp の値と、出力順序で前のクロックタイムスタンプから計算された clockTimestamp の値との差が、関連付けられたフレームやフィールドの、元の、またはキャプチャの時刻の時間差であると解釈できることを示す。discontinuity\_flag が 1 であることは、現 clockTimestamp の値と、出力順序で前のクロックタイムスタンプから計算された clockTimestamp の値との差が、関連付けられたフレームやフィールドの、元の、またはキャプチャの時刻の時間差であると解釈されるべきではないことを示す。discontinuity\_flag が 0 である場合、clockTimestamp の値は DPB 出力順序で先行するピクチャに対して存在する clockTimestamp 全ての値よりも大きいか等しくなくてはならない。

cnt\_dropped\_flag は、counting\_type により規定されるカウント方法を用いて、1 個以上の n\_frames の値のスキップを規定する。

n\_frames は、clockTimestamp を計算するために使用される nFrames の値を規定する。n\_frames は、以下よりも小さくなければならない。

$$\text{MaxFPS}=\text{Ceil}(\text{time\_scale}\div\text{num\_units\_in\_tick}) \quad (\text{D-2})$$

記5 `n_frames`はフレーム単位のカウンタである。フィールド固有のタイミング指示に対しては、それぞれのフィールドに対する異なる`clockTimestamp`を指示するために`time_offset`が使用されるべきである。

`counting_type`が2で、かつ`cnt_dropped_flag`が1の場合、`discontinuity_flag`が1でない限り、`n_frames`は1でなければならない、かつ出力順序で前のピクチャにおける`n_frames`の値は0であってはならない。

記6 `counting_type`が2の時、固定の非整数フレームレート（例えば`time_scale`が25、`num_units_in_tick`が2、`nuit_field_based_flag`が0で、12.5フレーム毎秒）を使用する場合に、等式D-1における`tOffset`の大きさを漸増的に大きくする必要は、カウント時に`n_frames`が0となる値を時々スキップすることにより避けることが可能である（例えば、`n_frames`を0から12までカウントし、それから`seconds_value`を増加し、そして`n_frames`を1から12までカウントし、それから`seconds_value`を増加し、そして`n_frames`を0から12までカウントする、等して行く）。

`counting_type`が3で、かつ`cnt_dropped_flag`が1の場合、`discontinuity_flag`が1でない限り、`n_frames`は0でなければならない、かつ出力順序で前のピクチャにおける`n_frames`の値は`MaxFPS-1`であってはならない。

記7 `counting_type`が3の時、固定の非整数フレームレート（例えば`time_scale`が25、`num_units_in_tick`が2、`nuit_field_based_flag`が0で、12.5フレーム毎秒）を使用する場合に、等式D-1における`tOffset`の大きさを漸増的に大きくする必要は、カウント時に`n_frames`が`MaxFPS`となる値を時々スキップすることにより避けることが可能である（例えば、`n_frames`を0から12までカウントし、それから`seconds_value`を増加し、そして`n_frames`を0から11までカウントし、それから`seconds_value`を増加し、そして`n_frames`を0から12までカウントする、等して行く）。

`counting_type`が4で、かつ`cnt_dropped_flag`が1の場合、`discontinuity_flag`が1でない限り、`n_frames`は2でなければならない、`sS`の指定値は0でなければならない、`mM`の指定値は10の整数倍であってはならず、かつ出力順序で前のピクチャにおける`n_frames`は0または1であってはならない。

記8 `counting_type`が4の時、固定の非整数フレームレート（例えば`time_scale`が60000、`num_units_in_tick`が1001、`nuit_field_based_flag`が1で、30000÷1001フレーム毎秒）を使用する場合に、等式D-1における`tOffset`の大きさを漸増的に大きくする必要は、カウント時に`n_frames`が`MaxFPS`となる値を時々スキップすることにより削減することが可能である（例えば、`seconds_value`が0でありかつ`minutes_value`が10の整数倍でなくなるまで、`n_frames`を0から29までカウントし、それから`seconds_value`を増加し、そして`n_frames`を0から29までカウントする、等して行く。それから`n_frames`を2から29までカウントし、それから`seconds_value`を増加し、そして`n_frames`を0から29までカウントする、等して行く）。このカウント方法は、産業界においてよく知られており、しばしば、“NTSCドロップフレーム”カウントと呼ばれる。

`counting_type`が5または6で、かつ`cnt_dropped_flag`が1の場合、`discontinuity_flag`が1でない限り、`n_frames`は、1足す出力順序で前のピクチャに対する`n_frames`値のモジュロ`MaxFPS`であってはならない。

記9 `counting_type`が5または6の時、固定の非整数フレームレートを使用する場合に、等式D-1における`tOffset`の大きさを漸増的に大きくする必要は、カウント時に`n_frames`のいくつかの値を時々スキップすることにより避けることが可能である。`counting_type`が5または6の場合、スキップされる、`n_frames`の具体的な値は規定されない。

`seconds_flag`が1であることは、`full_timestamp_flag`が0である場合に、`seconds_value`と`minutes_flag`が存在することを規定する。`seconds_flag`が0であることは、`seconds_value`と`minutes_flag`が存在しないことを規定する。

`seconds_value`は、`clockTimestamp`を計算するために使用される`sS`の値を規定する。`seconds_value`の値は0から59の、それぞれを含む範囲内でなければならない。`seconds_value`が存在しない場合、復号順序で前の`seconds_value`が、`clockTimestamp`を計算するための`sS`として使用されなければならない。

`minutes_flag`が1であることは、`full_timestamp_flag`が0で、かつ`seconds_flag`が1である時に、`minutes_value`と`hours_flag`が存在することを規定する。`minutes_flag`が0であることは、`minutes_value`と`hours_flag`が存在しないことを規定する。

**minutes\_value** は、clockTimestamp を計算するために使用される mM の値を規定する。minutes\_value の値は 0 から 59 の、それぞれを含む範囲でなければならない。minutes\_value が存在しない場合、復号順序で前の minutes\_value が、clockTimestamp を計算するための mM として使用されなければならない。

**hours\_flag** が 1 であることは、full\_timestamp\_flag が 0 で、かつ seconds\_flag が 1 で、かつ minutes\_flag が 1 である時に hours\_value が存在することを規定する。

**hours\_value** は、clockTimestamp を計算するために使用される hH の値を規定する。hours\_value は 0 から 23 の、それぞれを含む範囲でなければならない。hours\_value が存在しない場合、復号順序で前の hours\_value が、clockTimestamp を計算するための hH として使用されなければならない。

**time\_offset** は、clockTimestamp を計算するために使用される tOffset の値を規定する。time\_offset を表すことに使用されるビット数は、time\_offset\_length に等しくなければならない。time\_offset が存在しない場合、clockTimestamp を計算するための tOffset として、値 0 が使用されなければならない。

### D.2.3 パン・スキャン矩形SEIメッセージの意味

パン・スキャン矩形 SEI メッセージシンタックス要素は、シーケンスパラメータセットの切り出し矩形に相対で、矩形の座標を規定する。この矩形の各座標は、輝度サンプルグリッドに対する 16 分の 1 サンプル間隔の単位で規定される。

**pan\_scan\_rect\_id** は、パン・スキャン矩形の目的を識別することに使用できる識別番号を持つ（例えば、特定の表示装置上で表示される領域として、または、その場面で特定人物を含む領域として矩形を識別する）。pan\_scan\_rect\_id の値は 0 から  $2^{32}-1$  までの、それぞれを含む範囲でなければならない。

0 から 255 までと 512 から  $2^{31}-1$  までの pan\_scan\_rect\_id の値はアプリケーションで決定される通りに使用されてもよい。256 から 511 までと  $2^{31}$  から  $2^{32}-1$  までの pan\_scan\_rect\_id の値は TTC により将来の使用のために予約される。復号器が、256 から 511 までの範囲か、 $2^{31}$  から  $2^{32}-1$  までの範囲の pan\_scan\_rect\_id の値に遭遇した際は、それを無視（ビットストリームから除去して破棄）しなければならない。

**pan\_scan\_rect\_cancel\_flag** が 1 であることは、その SEI メッセージが、出力順序で以前の任意のパン・スキャン矩形 SEI メッセージの持続を取消することを示す。pan\_scan\_rect\_cancel\_flag が 0 であることは、パン・スキャン矩形情報が次に続くことを示す。

**pan\_scan\_cnt\_minus1** は、その SEI メッセージに存在するパン・スキャン矩形の数を規定する。pan\_scan\_cnt\_minus1 は 0 から 2 までの、それぞれを含む範囲でなければならない。pan\_scan\_cnt\_minus1 が 0 であることは、復号ピクチャの全てのフィールドに適用される、ただ 1 個のパン・スキャン矩形が存在することを示す。現ピクチャがフィールドである場合、pan\_scan\_cnt\_minus1 は 0 でなければならない。pan\_scan\_cnt\_minus1 が 1 であることは、2 個のパン・スキャン矩形が存在することを示し、その最初のは出力順序で最初のフィールドに適用され、2 番目のものは出力順序で 2 番目のフィールドに適用される。pan\_scan\_cnt\_minus1 が 2 であることは、3 個のパン・スキャン矩形が存在し、その最初のは出力順序で最初のフィールドに適用され、2 番目のものは出力順序で 2 番目のフィールドに適用され、3 番目のものは出力順序で 3 番目のフィールドとしての、1 番目のフィールドの繰り返しに対して適用される。

**pan\_scan\_rect\_left\_offset[i]**、**pan\_scan\_rect\_right\_offset[i]**、**pan\_scan\_rect\_top\_offset[i]**、そして **pan\_scan\_rect\_bottom\_offset[i]** は、パン・スキャン矩形の位置を、輝度サンプルグリッドに対して 16 分の 1 サンプル間隔単位で、符号付き整数として規定する。これらの 4 個のシンタックス要素それぞれの値は、 $-2^{31}$  から  $2^{31}-1$  までのそれぞれを含む範囲でなければならない。

パン・スキャン矩形は、輝度フレームサンプルグリッドに対して 16 分の 1 サンプル間隔単位で、 $16 * \text{CropUnitX} * \text{frame\_crop\_left\_offset} + \text{pan\_scan\_rect\_left\_offset}[i]$  から、 $16 * (16 * \text{PicWidthInMbs} - \text{CropUnitX} * \text{frame\_crop\_right\_offset}) + \text{pan\_scan\_rect\_right\_offset}[i] - 1$  までのフレーム水平座標と、 $16 * \text{CropUnitY} * \text{frame\_crop\_top\_offset} + \text{pan\_scan\_rect\_top\_offset}[i]$  から、 $16 * (16 * \text{PicHeightInMbs} - \text{CropUnitY} * \text{frame\_crop\_bottom\_offset}) + \text{pan\_scan\_rect\_bottom\_offset}[i] - 1$  までの垂直座標を、両端値を含んで持つ座標の領域として規定される。 $16 * \text{CropUnitX} * \text{frame\_crop\_left\_offset} + \text{pan\_scan\_rect\_left\_offset}[i]$  の値は  $16 * (16 * \text{PicWidthInMbs} - \text{CropUnitX} * \text{frame\_crop\_right\_offset}) + \text{pan\_scan\_rect\_right\_offset}[i] - 1$  よりも小さいか等しくなくてはならず、 $16 * \text{CropUnitY} * \text{frame\_crop\_top\_offset} + \text{pan\_scan\_rect\_top\_offset}[i]$  の値は、 $16 * (16 * \text{PicHeightInMbs} - \text{CropUnitY} * \text{frame\_crop\_bottom\_offset}) + \text{pan\_scan\_rect\_bottom\_offset}[i] - 1$  よりも小さいか等しくなくてはならない。

パン・スキャン矩形領域が、切り出し矩形の外側のサンプルを含む場合、その切り出し矩形の外側の領域は、表示に際し、合成内容(黒のビデオ内容や中間である灰色のビデオ内容等)で埋められてもよい。

**pan\_scan\_rect\_repetition\_period** は、パン・スキャン矩形 SEI メッセージの持続性を規定し、さらに同じ **pan\_scan\_rect\_id** の値を持つもう 1 つのパン・スキャン矩形 SEI メッセージ、または符号化ビデオシーケンスの終了がビットストリーム中に存在しなければならないピクチャ順序カウント間隔を規定してもよい。**pan\_scan\_rect\_repetition\_period** の値は 0 から 16384 までの、それぞれを含む範囲でなければならない。**pan\_scan\_cnt\_minus1** が 0 よりも大きい場合、**pan\_scan\_rect\_repetition\_period** は 1 よりも大きくてはならない。

**pan\_scan\_rect\_repetition\_period** が 0 であることは、パン・スキャン矩形情報が現復号ピクチャのみに適用されることを規定する。

**pan\_scan\_rect\_repetition\_period** が 1 であることは、以下の条件のいずれかが真となるまで、出力順序でパン・スキャン矩形情報が持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる。
- 同じ **pan\_scan\_rect\_id** の値を持つパン・スキャン矩形 SEI メッセージを含むアクセスユニット中のピクチャが、**PicOrderCnt(CurrPic)** よりも大きい **PicOrderCnt()** を持って出力される。

**pan\_scan\_rect\_repetition\_period** が 0、または 1 であることは、同じ **pan\_scan\_rect\_id** の値を持つもう 1 つのパン・スキャン矩形 SEI メッセージが存在してもよく、または存在しなくてもよいことを示す。

**pan\_scan\_rect\_repetition\_period** が 1 よりも大きいことは、以下の条件のいずれかが真となるまで、パン・スキャン矩形情報が持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる。
- 同じ **pan\_scan\_rect\_id** の値を持つパン・スキャン矩形 SEI メッセージを含むアクセスユニット中のピクチャが、**PicOrderCnt(CurrPic)** よりも大きく、かつ **PicOrderCnt(CurrPic) + pan\_scan\_rect\_repetition\_period** よりも小さいかもしくは等しい **PicOrderCnt()** を持って出力される。

**pan\_scan\_rect\_repetition\_period** が 1 よりも大きいことは、同じ **pan\_scan\_rect\_id** の値を持つもう 1 つのパン・スキャン矩形 SEI メッセージが、**PicOrderCnt(CurrPic)** よりも大きく、かつ **PicOrderCnt(CurrPic) + pan\_scan\_rect\_repetition\_period** よりも小さいかもしくは等しい **PicOrderCnt()** を持って出力されるあるアクセスユニット中のあるピクチャにおいて、そのようなピクチャの出力なしにビットストリ

ームが終了するか新しい符号化ビデオシーケンスが開始しない限り、存在しなければならないことを示す。

#### D.2.4 フィルペイロードSEIメッセージの意味

このメッセージは、捨てることのできる、値 0xFF の payloadSize バイトの並びを含む。

**ff\_byte** は、値 0xFF を持つ 1 バイトでなければならない。

#### D.2.5 ITU-T勧告T.35による登録ユーザデータSEIメッセージの意味

このメッセージは ITU-T 勧告 T.35 で規定される登録ユーザデータを含み、その内容は本標準によって規定されない。

**itu\_t\_t35\_country\_code** は、ITU-T 勧告 T.35 Annex A により、国番号として規定される値をもつ 1 バイトでなければならない。

**itu\_t\_t35\_country\_code\_extension\_byte** は、ITU-T 勧告 T.35 Annex B により、国番号として規定された値を持つ 1 バイトでなければならない。

**itu\_t\_t35\_payload\_byte** は、ITU-T 勧告 T.35 で規定される登録データを含む 1 バイトでなければならない。

ITU-T T.35 端末提供者番号と端末提供者指向番号は、端末提供者番号を発行した管理機関によって規定された形式で、**itu\_t\_t35\_payload\_byte** の最初の 1 個以上のバイト中に含まなければならない。残りの全ての **itu\_t\_t35\_payload\_byte** データは、ITU-T T.35 国番号と端末提供者番号によって識別された実体で規定されるシンタックスと意味を持つデータでなければならない。

#### D.2.6 非登録ユーザデータSEIメッセージの意味

このメッセージは UUID によって識別された非登録ユーザデータを含み、その内容は本標準によって規定されない。

**uuid\_iso\_iec\_11578** は、ISO/IEC 11578:1996 Annex A の手順に従って、UUID として規定された値を持たなければならない。

**user\_data\_payload\_byte** は、UUID 生成者によって規定されたシンタックスと意味を持つデータを含む 1 バイトでなければならない。

#### D.2.7 復旧点SEIメッセージの意味

復旧点 SEI メッセージは、復号器がランダムアクセスを開始した後、または符号器がシーケンス中でブロークンリンクを示した後、表示に容認できるピクチャを復号処理がいつ生成するかを決定することで復号器を支援する。復号処理が復旧点 SEI メッセージに復号順序に関連付けられたアクセスユニットで開始された時、その SEI メッセージで規定された復旧点と、出力順序でそれに続く全ての復号ピクチャは内容が正しいまたはほぼ正しいことが示される。復旧点 SEI メッセージに関連付けられたピクチャにおける、またはそれ以前における、ランダムアクセスによって生成された復号ピクチャは、示された復旧点まで内容が正しい必要はなく、復旧点 SEI メッセージに関連付けられたピクチャから開始する復号処理の動作は、復号ピクチャバッファ中で利用不可であるピクチャへの参照を含むかもしれない。

加えて、**broken\_link\_flag** の使用により、復旧点 SEI メッセージは、たとえ復号処理が、復号順序で前の IDR

アクセスユニットの位置で開始された場合でも、表示すると深刻な視覚上の歪みを生ずるかもしれないビットストリーム中のいくつかのピクチャの位置を、復号器に示すことができる。

記1 **broken\_link\_flag**は、その点以降で、いくつかのピクチャの復号においてその復号処理が、復号処理で使用できるが、ビットストリームが元々符号化された際に参照に使用されたピクチャではない（例えば、ビットストリーム生成期間にスプライシング操作がなされている）ピクチャへの参照を引き起こすかもしれない点の位置を示すために符号器により使用できる。

復旧点は、その SEI メッセージ位置の現アクセスユニットの **frame\_num** に続く、**frame\_num** 増加単位のカウンタとして規定される。

記2 HRD情報がビットストリーム中に存在する場合、ランダムアクセス後のHRDバッファモデルの初期化を確立するために、バッファリング期間SEIメッセージは復旧点SEIメッセージに関連付けられたアクセスユニットに関連付けられるべきである。

**recovery\_frame\_cnt** は、出力順序で出力ピクチャの復旧点を規定する。現アクセスユニットに対する VCL NAL ユニットの **frame\_num** に **recovery\_frame\_cnt** を加えた値に、モジュロ **MaxFrameNum** 算術で等しい **frame\_num** を持つ参照ピクチャの出力順序位置で開始する、出力順序で全ての復号ピクチャは内容が正しい、またはほぼ正しいことが示される。**recovery\_frame\_cnt** は、0 から **MaxFrameNum-1** までの、それぞれを含む範囲内でなければならない。

**exact\_match\_flag** は、その復旧点 SEI メッセージに関連付けられたアクセスユニットで復号処理を開始することで導出される、規定された復旧点と、出力順序でそれに続く復号ピクチャが、NAL ユニットストリーム中で前の IDR アクセスユニットの位置から復号処理を開始して生成されるピクチャと完全に一致しなければならないかどうかを示す。値 0 は一致が完全である必要がないことを示し、値 1 は一致が完全でなくてはならないことを示す。

**exact\_match\_flag** の値の適合性を決定する目的のために、復旧点 SEI メッセージの位置から復号を開始する場合、利用不可である参照ピクチャへの全ての参照は、INTRA マクロブロック予測モードを使用して符号化されたマクロブロックのみを含み、Y サンプルが 128、Cb サンプルが 128、Cr サンプルが 128（中央レベルの灰色）により与えられるサンプル値をもつピクチャへの参照だと推定されなくてはならない。

記3 ランダムアクセスを行う場合、復号器は**exact\_match\_flag**の値にかかわらず、利用不可の参照ピクチャへの全ての参照は、INTRAマクロブロックのみを含む、Yが128、Cbが128、Crが128（中央レベルの灰色）により与えられるサンプル値をもつピクチャへの参照だと推定するべきである。

**exact\_match\_flag** が 0 の場合、符号化処理によって復旧点における近似の質が選択され、それは本標準では規定されない。

**broken\_link\_flag** は、復旧点 SEI メッセージの位置で、NAL ユニットストリームにブローケンリンクが存在するか存在しないかを示し、さらなる意味が次の通り割り当てられる。

- もし **broken\_link\_flag** が 1 であるならば、前の IDR アクセスユニットの位置から復号処理を開始して生成されたピクチャは、復旧点 SEI メッセージに関連付けられたアクセスユニットの復号ピクチャと、復号順序でそれに続く復号ピクチャが、出力順序で規定された復旧点まで、表示されるべきではないほどの望ましくない視覚上の歪を含むかもしれない。
- それ以外 (**broken\_link\_flag** が 0)、視覚上の歪のいかなる潜在的存在に関しても指示は与えられない。

**broken\_link\_flag** の値にかかわらず、出力順序で、規定された復旧点に続くピクチャは、内容が正しいか、ほぼ正しいものであると規定される。

記4 サブシーケンス情報SEIメッセージが、**broken\_link\_flag**が1となっている復旧点SEIメッセージとともに存在する場合、そして、**sub\_seq\_layer\_num**が0であるならば、**sub\_seq\_id**は、復旧点SEIメッセージの

位置よりも前に復号された、0に等しいsub\_seq\_layer\_numに対する最新のsub\_seq\_idとは異なるべきである。broken\_link\_flagが0である時は、サブシーケンスレイヤ0におけるsub\_seq\_idは変わらぬままであるべきである。

changing\_slice\_group\_idc が 0 であることは、主符号化ピクチャの全てのマクロブロックがスライスグループ変更期間、すなわち復号順序で復旧点 SEI メッセージに関連付けられたアクセスユニット（これを含めて）と規定された復旧点（これを含めて）の期間内に復号されるならば、復旧点と、出力順序でそれに続く復号ピクチャは、内容が正しいかほぼ正しいことを示す。スライスグループ変更期間内で、任意の主符号化ピクチャ中で num\_slice\_groups\_minus1 が 0 であるなら、changing\_slice\_group\_idc は 0 でなければならない。

changing\_slice\_group\_idc が 1 または 2 である場合、num\_slice\_groups\_minus1 は 1 でなければならない、マクロブロックからスライスグループへのマップタイプ 3、4、または 5 はスライスグループ変更期間中のそれぞれの主符号化ピクチャに適用されなければならない。

changing\_slice\_group\_idc が 1 であることは、スライスグループ変更期間内に、スライスグループ 0 により含まれる復号マクロブロックの外側のサンプル値はスライスグループ 0 の中のどんなマクロブロックの INTER 予測のためにも使用されないことを示す。加えて、changing\_slice\_group\_idc が 1 であることは、スライスグループ変更期間内のスライスグループ 0 の全てのマクロブロックが復号された時、スライスグループ変更期間内の、スライスグループ 1 の任意のマクロブロックが復号されるかどうかにかかわらず、規定された復旧点か、出力順序でそれに続く復号ピクチャは内容が正しいかほぼ正しいであろうことを示す。

changing\_slice\_group\_idc が 2 であることは、スライスグループ変更期間内で、スライスグループ 1 により含まれる復号マクロブロックの外側のサンプル値はスライスグループ 1 の中のどんなマクロブロックの INTER 予測のためにも使用されないことを示す。加えて、changing\_slice\_group\_idc が 2 であることは、スライスグループ変更期間内のスライスグループ 1 の全てのマクロブロックが復号された時、スライスグループ変更期間内の、スライスグループ 0 の任意のマクロブロックが復号されるかどうかにかかわらず、規定された復旧点か、出力順序でそれに続く復号ピクチャは内容が正しいかほぼ正しいであろうことを示す。

changing\_slice\_group\_idc は 0 から 2 の、それぞれを含む範囲内でなければならない。

#### D.2.8 復号参照ピクチャマーキング繰り返しSEIメッセージの意味

復号参照ピクチャマーキング繰り返し SEI メッセージは、復号順序でシーケンス内のより早いピクチャのスライスヘッダに位置した復号参照ピクチャマーキングシンタックス構造を繰り返すために用いられる。

original\_idr\_flag は、復号参照ピクチャマーキングシンタックス構造が元々 IDR ピクチャの中で生じていた時、1 に等しくなければならない。original\_idr\_flag は、繰り返される復号参照ピクチャマーキングシンタックス構造が元々 IDR ピクチャの中で生じていなかった時、0 に等しくなければならない。

original\_frame\_num は、繰り返される復号参照ピクチャマーキングシンタックス構造が元々生じていたピクチャの frame\_num に等しくなければならない。original\_frame\_num で示されるピクチャは、規定された frame\_num の値をもつ前に符号化されたピクチャである。5 に等しい memory\_management\_control\_operation をもつピクチャを参照することに使用される original\_frame\_num の値は 0 でなければならない。

original\_field\_pic\_flag は、繰り返される復号参照ピクチャマーキングシンタックス構造が元々生じていたピクチャの field\_pic\_flag に等しくなければならない。

original\_bottom\_field\_flag は、繰り返される復号参照ピクチャマーキングシンタックス構造が元々生じていたピクチャの bottom\_field\_flag に等しくなければならない。

`dec_ref_pic_marking()`は、その `frame_num` が `original_frame_num` であったピクチャの復号参照ピクチャマージングシンタックス構造の複写を含まなければならない。繰り返される `dec_ref_pic_marking()` シンタックス構造の規定のために用いられる `IdrPicFlag` は、その `frame_num` が `original_frame_num` であったピクチャの(複数の) スライスヘッダの `IdrPicFlag` でなければならない(すなわち、7.3.3.3 小小節で用いられるような `IdrPicFlag` は、`original_idr_flag` に等しいとみなされなければならない)。

## D.2.9 予備ピクチャSEIメッセージの意味

この SEI メッセージは、1 つ以上の復号参照ピクチャの中の予備スライスグループマップユニットと呼ばれる、あるいくつかのスライスグループマップユニットが、ターゲットピクチャと呼ばれる規定された復号ピクチャ中の共通位置スライスグループマップユニットに似ていることを示す。予備スライスグループマップユニットは、ターゲットピクチャの共通位置にある誤って復号されたスライスグループマップユニットを置き換えるために用いられるかもしれない。予備スライスグループマップユニットを含む復号ピクチャは予備ピクチャと呼ばれる。

予備ピクチャ SEI メッセージの中で識別される全ての予備ピクチャに対して、`frame_mbs_only_flag` の値は、同じ SEI メッセージ中のターゲットピクチャの `frame_mbs_only_flag` の値に等しくなければならない。SEI メッセージの予備ピクチャは次の通り制約される。

- もしターゲットピクチャが復号フィールドであるならば、同じ SEI メッセージの中で識別される全ての予備ピクチャは復号フィールドでなければならない。
- それ以外(ターゲットピクチャが復号フレームである)、同じ SEI メッセージの中で識別される全ての予備ピクチャは復号フレームでなければならない。

予備ピクチャ SEI メッセージの中で識別される全ての予備ピクチャに対して、`pic_width_in_mbs_minus1` および `pic_height_in_map_units_minus1` の値は、同じ SEI メッセージ中のターゲットピクチャの `pic_width_in_mbs_minus1` および `pic_height_in_map_units_minus1` の値にそれぞれ等しくなければならない。このメッセージに(7.4.1.2.3 小小節で規定されるように) 関連付けられるピクチャは、復号順序でターゲットピクチャの後に出現しなければならない。

`target_frame_num` はターゲットピクチャの `frame_num` を示す。

`spare_field_flag` が 0 に等しいことは、ターゲットピクチャおよび予備ピクチャが復号フレームであることを示す。`spare_field_flag` が 1 に等しいことは、ターゲットピクチャおよび予備ピクチャが復号フィールドであることを示す。

`target_bottom_field_flag` が 0 に等しいことは、ターゲットピクチャがトップフィールドであることを示す。`target_bottom_field_flag` が 1 に等しいことは、ターゲットピクチャがボトムフィールドであることを示す。

ターゲットピクチャは、その対応する主符号化ピクチャが復号順序で現ピクチャに先行し、そのピクチャの中の `frame_num`、`field_pic_flag` (存在する時) および `bottom_field_flag` (存在する時) が `target_frame_num`、`spare_field_flag` および `target_bottom_field_flag` にそれぞれ等しい復号参照ピクチャである。

`num_spare_pics_minus1` は、規定されるターゲットピクチャに対する予備ピクチャの数を示す。予備ピクチャの数は `num_spare_pics_minus1+1` に等しい。`num_spare_pics_minus1` の値は 0 から 15 のそれぞれの値を含む範囲でなければならない。

`delta_spare_frame_num[i]`は、下記で規定されるように、以下 `i` 番目の予備ピクチャと呼ばれる、予備スラ

イスグループマップユニットの  $i$  番目の組を含む予備ピクチャを識別するために用いられる。 $\text{delta\_spare\_frame\_num}[i]$ の値は 0 から  $\text{MaxFrameNum}-1-\text{spare\_field\_flag}$  のそれぞれの値を含む範囲でなければならない。

$i$  番目の予備ピクチャの  $\text{frame\_num}$  である  $\text{spareFrameNum}[i]$ は、0 から  $\text{num\_spare\_pics\_minus1}$  のそれぞれの値を含む全ての  $i$  の値に対して次のように導出される。

```
candidateSpareFrameNum=target_frame_num-!spare_field_flag
for(i=0;i<=num_spare_pics_minus1;i++){
    if(candidateSpareFrameNum<0)
        candidateSpareFrameNum=MaxFrameNum-1
    spareFrameNum[i]=candidateSpareFrameNum-delta_spare_frame_num[i]
    if(spareFrameNum[i]<0)
        spareFrameNum[i]=MaxFrameNum+spareFrameNum[i]
    candidateSpareFrameNum=spareFrameNum[i]-!spare_field_flag
}
```

(D-3)

$\text{spare\_bottom\_field\_flag}[i]$ が 0 に等しいことは、 $i$  番目の予備ピクチャがトップフィールドであることを示す。 $\text{spare\_bottom\_field\_flag}[i]$ が 1 に等しいことは、 $i$  番目の予備ピクチャがボトムフィールドであることを示す。

0 番目の予備ピクチャは、その対応する主符号化ピクチャが復号順序でターゲットピクチャに先行し、そしてその  $\text{frame\_num}$ 、 $\text{field\_pic\_flag}$  (存在する時)、および  $\text{bottom\_field\_flag}$  (存在する時)が  $\text{spareFrameNum}[0]$ 、 $\text{spare\_field\_flag}$ 、および  $\text{spare\_bottom\_field\_flag}[0]$ にそれぞれ等しい復号参照ピクチャである。 $i$  番目の予備ピクチャは、その対応する主符号化ピクチャが復号順序で $(i-1)$ 番目の予備ピクチャに先行し、そしてその  $\text{frame\_num}$ 、 $\text{field\_pic\_flag}$  (存在する時)、および  $\text{bottom\_field\_flag}$  (存在する時)が  $\text{spareFrameNum}[i]$ 、 $\text{spare\_field\_flag}$ 、および  $\text{spare\_bottom\_field\_flag}[i]$ にそれぞれ等しい復号参照ピクチャである。

$\text{spare\_area\_idc}[i]$ は、 $i$  番目の予備ピクチャの中にある予備スライスグループマップユニットを識別するために用いられる方法を示す。 $\text{spare\_area\_idc}[i]$ は 0 から 2 のそれぞれの値を含む範囲でなければならない。 $\text{spare\_area\_idc}[i]$ が 0 に等しいことは、 $i$  番目の予備ピクチャの中にある全てのスライスグループマップユニットが予備ユニットであることを示す。 $\text{spare\_area\_idc}[i]$ が 1 に等しいことは、シンタックス要素  $\text{spare\_unit\_flag}[i][j]$ が予備スライスグループマップユニットを識別するために用いられることを示す。 $\text{spare\_area\_idc}[i]$ が 2 に等しいことは、以下に記述されるように  $\text{zero\_run\_length}[i][j]$ シンタックス要素が  $\text{spareUnitFlagInBoxOutOrder}[i][j]$ の値を導出するために用いられることを示す。

$\text{spare\_unit\_flag}[i][j]$ が 0 に等しいことは、 $i$  番目の予備ピクチャ中のラスタ走査順序で  $j$  番目のスライスグループマップユニットが予備ユニットであることを示す。 $\text{spare\_unit\_flag}[i][j]$ が 1 に等しいことは、 $i$  番目の予備ピクチャ中のラスタ走査順序で  $j$  番目のスライスグループマップユニットが予備ユニットではないことを示す。

$\text{zero\_run\_length}[i][j]$ は、 $\text{spare\_area\_idc}[i]$ が 2 に等しい時に  $\text{spareUnitFlagInBoxOutOrder}[i][j]$ の値を導出するために用いられる。この場合、 $\text{spareUnitFlagInBoxOutOrder}[i][j]$ で識別される予備スライスグループマップユニットは、それぞれの予備ピクチャに対して 8.2.2.4 小節で規定されるように反時計回りボックスアウトの順序で出現する。 $\text{spareUnitFlagInBoxOutOrder}[i][j]$ が 0 に等しいことは、 $i$  番目の予備ピクチャ中の反時計回りボックスアウトの順序で  $j$  番目のスライスグループマップユニットが予備ユニットであることを示す。

spareUnitFlagInBoxOutOrder[i][j]が 1 に等しいことは、i 番目の予備ピクチャ中の反時計回りボックスアウトの順序で j 番目のスライスグループマップユニットが予備ユニットでないことを示す。

spare\_area\_idc[0]が 2 に等しい時、spareUnitFlagInBoxOutOrder[0][j]は次のように導出される。

```
for(j=0,loop=0;j<PicSizeInMapUnits;loop++){
    for(k=0;k<zero_run_length[0][loop];k++)
        spareUnitFlagInBoxOutOrder[0][j++] = 0
        spareUnitFlagInBoxOutOrder[0][j++] = 1
}
```

(D-4)

spare\_area\_idc[i]が 2 に等しく、かつ i の値が 0 より大きい時、spareUnitFlagInBoxOutOrder[i][j]は次のように導出される。

```
for(j=0,loop=0;j<PicSizeInMapUnits;loop++){
    for(k=0;k<zero_run_length[i][loop];k++)
        spareUnitFlagInBoxOutOrder[i][j] = spareUnitFlagInBoxOutOrder[i-1][j++]
        spareUnitFlagInBoxOutOrder[i][j] = !spareUnitFlagInBoxOutOrder[i-1][j++]
}
```

(D-5)

## D.2.10 シーン情報SEIメッセージの意味

シーンおよびシーン遷移は、ここでは出力順序で連続するピクチャの組として定義される。

記1 1つのシーンの中の復号ピクチャは一般的に類似の内容をもつ。シーン情報SEIメッセージは、ピクチャをシーン識別子でラベル付けするために、そしてシーンチェンジを示すために用いられる。そのメッセージは、ラベル付けされたピクチャに対する原ピクチャがどのように作成されたのかを規定する。復号器は、伝送誤りをコンシールメントするために適切なアルゴリズムを選択することにその情報を用いても良い。例えば、ある特定のアルゴリズムが、漸進的なシーン遷移に属するピクチャの中で発生した伝送誤りをコンシールメントするために用いられても良い。さらに、シーン情報SEIメッセージは、符号化シーケンスのシーンを索引付けするためにアプリケーションで決められる方法で用いられても良い。

シーン情報 SEI メッセージは復号順序で、7.4.1.2.3 小小小節で規定されるように SEI メッセージが関連付けられる主符号化ピクチャ（を含む）から、（もし存在するならば）復号順序で次のシーン情報 SEI メッセージが関連付けられる主符号化ピクチャ（含まない）まで、または（それ以外）ビットストリームの最後のアクセスユニット（を含む）までの全てのピクチャをラベル付けする。これらのピクチャは、ここではターゲットピクチャと呼ばれる。

**scene\_info\_present\_flag** が 0 に等しいことは、ターゲットピクチャが属するシーンあるいはシーン遷移が無規定であることを示す。scene\_info\_present\_flag が 1 に等しいことは、ターゲットピクチャが同じシーンあるいはシーン遷移に属することを示す。

**scene\_id** はターゲットピクチャが属するシーンを識別する。ターゲットピクチャの scene\_transition\_type の値が 4 より小さく、かつ出力順序で前のピクチャが 4 より小さい scene\_transition\_type の値をマークされ、かつ scene\_id の値が出力順序で前のピクチャの scene\_id の値に等しい場合、これはそのターゲットピクチャに対する原シーンと（出力順序で）前ピクチャに対する原シーンが、符号器で同じシーンであったと考えられることを示す。ターゲットピクチャの scene\_transition\_type の値が 3 より大きく、かつ出力順序で前のピクチャが 4 より小さい scene\_transition\_type の値をマークされ、かつ scene\_id の値が出力順序で前のピクチャの scene\_id の値に等しい場合、これはそのターゲットピクチャに対する原シーンの 1 つと（出力順序で）前ピク

クチャに対する原シーンが、符号器で同じシーンであったと考えられることを示す。scene\_id の値が出力順序で前のピクチャの scene\_id の値と等しくないならば、これはそのターゲットピクチャと（出力順序で）前ピクチャが、符号器で異なった原シーンからであったと考えられることを示す。

scene\_id の値は、0 から  $2^{32}-1$  のそれぞれの値を含む範囲でなければならない。0 から 255 のそれぞれの値を含む範囲および 512 から  $2^{31}-1$  のそれぞれの値を含む範囲の scene\_id の値は、アプリケーションによって決められるものとして用いられても良い。256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の scene\_id の値は、TTC によって将来の使用のために予約される。256 から 511 のそれぞれの値を含む範囲、あるいは  $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の scene\_id の値に遭遇した復号器は、それを無視（ビットストリームから除去して破棄）しなければならない。

scene\_transition\_type は、（シーン遷移があるならば）ターゲットピクチャが関わるシーン遷移のタイプを規定する。scene\_transition\_type の有効な値は付表 D-4/JT-H264 で規定される。

付表D-4/JT-H264 scene\_transition\_type値  
(ITU-T H.264)

値	記述
0	遷移なし
1	黒へのフェード
2	黒からのフェード
3	一定な色からまたは一定な色への無規定遷移
4	ディゾルブ
5	ワイプ
6	無規定である2つのシーンの混合

scene\_transition\_type が 3 より大きい時、ターゲットピクチャは、その scene\_id によってラベル付けされるシーン、および second\_scene\_id（下記参照）によってラベル付けされる出力順序で次のシーンの両方からの内容を含む。“現シーン”という用語は、scene\_id によってラベル付けされるシーンを示すために用いられる。“次のシーン”という用語は、second\_scene\_id によってラベル付けされるシーンを示すために用いられる。出力順序で次のどんなピクチャに対しても、現 SEI メッセージの second\_scene\_id に等しい scene\_id でラベル付けされることは必要とされない。

シーン遷移タイプは、以下のように規定される。

“遷移無し”は、ターゲットピクチャが漸進的なシーン遷移に関わらないことを規定する。

記2 出力順序で2つの連続するピクチャが、0に等しいscene\_transition\_type、および異なるscene\_idの値をもつ時、シーンカットが2つのピクチャ間で生じた。

“黒へのフェード”は、ターゲットピクチャが黒へのフェードシーン遷移、すなわち、シーンの輝度サンプルが徐々に0に近づき、シーンの色差サンプルが徐々に128に近づくシーン遷移に関わる出力順序でのピクチャ列の一部であることを示す。

記3 2つのピクチャが同じシーン遷移に属するようにラベル付けされ、それらのscene\_transition\_typeが“黒へのフェード”である時、出力順序で後のピクチャは前のピクチャよりも暗い。

“黒からのフェード”は、ターゲットピクチャが黒からのフェードシーン遷移、すなわち、シーンの輝度

サンプルが徐々に 0 から発散し、シーンの色差サンプルが徐々に 128 から発散するシーン遷移に関わる出力順序でのピクチャ列の一部であることを示す。

記4 2つのピクチャが同じシーン遷移に属するようにラベル付けされ、それらの `scene_transition_type` が“黒からのフェード”である時、出力順序で後のピクチャは前のピクチャよりも明るい。

“ディゾルブ”は、(符号化前の)各ターゲットピクチャのサンプル値が、現シーンからのピクチャと次のシーンからのピクチャの共通位置の重み付けされたサンプル値の和を計算することで生成されたことを示す。現シーンの重みはフルレベルからゼロレベルへと徐々に減少する一方で、次のシーンの重みはゼロレベルからフルレベルへと徐々に増大する。2つのピクチャが同じシーン遷移に属するようにラベル付けされ、それらの `scene_transition_type` が“ディゾルブ”である時、出力順序でより後のピクチャに対する現シーンの重みは、前のピクチャに対する現シーンの重みより小さく、そして出力順序でより後のピクチャに対する次のシーンの重みは、前のピクチャに対する次のシーンの重みより大きい。

“ワイプ”は、(符号化前の)各ターゲットピクチャのサンプル値のいくつかが現シーンのピクチャの共通位置サンプル値の複写によって生成され、そして、(符号化前の)各ターゲットピクチャの残ったサンプル値が次のシーンのピクチャの共通位置サンプル値の複写で生成されたことを示す。2つのピクチャが同じシーン遷移に属するようにラベル付けされ、それらの `scene_transition_type` が“ワイプ”である時、出力順序でより後のピクチャに対する次のシーンから複写されたサンプルの数は、前のピクチャに対する次のシーンから複写されたサンプルの数よりも大きい。

`second_scene_id` は、ターゲットピクチャに関わる漸進的なシーン遷移の次のシーンを識別する。`second_scene_id` の値は、`scene_id` の値と等しくはならない。`second_scene_id` の値は出力順序で前のピクチャの `scene_id` の値と等しくはならない。出力順序で次のピクチャが、4より小さい `scene_transition_type` の値でマークされ、かつ `second_scene_id` の値が出力順序で次のピクチャの `scene_id` の値と同じであるならば、これは符号器がそのターゲットピクチャに対する原シーンの1つと(出力順序で)次のピクチャに対する原シーンが同じシーンであったとみなすことを示す。`second_scene_id` の値と、出力順序で次のピクチャの `scene_id` の値、または(存在すれば) `second_scene_id` の値が等しくないならば、これは符号器がそのターゲットピクチャと(出力順序で)次のピクチャが異なった原シーンからであったとみなすことを示す。

ピクチャの `scene_id` の値が出力順序で続くピクチャの `scene_id` の値に等しく、かつこれらピクチャの両方で `scene_transition_type` の値が4より小さい時、これは符号器がその2つのピクチャは同じ原シーンからであったとみなすことを示す。あるピクチャの `scene_id`、`scene_transition_type` そして(存在すれば) `second_scene_id` の値が出力順序で続くピクチャの `scene_id`、`scene_transition_type` そして `second_scene_id` の値と(それぞれ)等しく、かつ `scene_transition_type` の値が0より大きい時、これは符号器がその2つのピクチャは同じ原漸進的シーン遷移からであったとみなすことを示す。

`second_scene_id` の値は、0から  $2^{32}-1$  のそれぞれの値を含む範囲でなければならない。0から255のそれぞれの値を含む範囲および、512から  $2^{31}-1$  のそれぞれの値を含む範囲の `second_scene_id` の値は、アプリケーションで決められるものとして用いられても良い。256から511のそれぞれの値を含む範囲および、 $2^{31}$ から  $2^{32}-1$  のそれぞれの値を含む範囲の `second_scene_id` の値は、TTCによって将来の使用のために予約される。256から511のそれぞれの値を含む範囲、あるいは  $2^{31}$ から  $2^{32}-1$  のそれぞれの値を含む範囲の `second_scene_id` の値に遭遇した復号器は、それを無視(ビットストリームから除去して破棄)しなければならない。

#### D.2.11 サブシーケンス情報SEIメッセージの意味

サブシーケンス情報SEIメッセージは、サブシーケンスレイヤおよびサブシーケンスから構成されるデータ依存階層でのピクチャ位置を示すために用いられる。

サブシーケンスレイヤはシーケンス内の符号化ピクチャの部分集合を含む。サブシーケンスレイヤは非負の整数で番号付けされる。より大きいレイヤ番号をもつレイヤは、より小さいレイヤ番号をもつレイヤより高いレイヤである。レイヤは、あるレイヤの任意のピクチャが、任意のより高いレイヤの任意のピクチャから予測されてはならないようにするために、互いの依存性に基づいて階層的に順序付けられる。

記1 言い換えると、レイヤ0の任意のピクチャはレイヤ1以上のレイヤの任意のピクチャから予測されてはならず、レイヤ1のピクチャはレイヤ0から予測されても良く、レイヤ2のピクチャはレイヤ0および1から予測されても良い、など。

記2 主観的画質は復号されるレイヤの数が増加するに従って向上すると期待される。

あるサブシーケンスは、あるサブシーケンスレイヤ内の符号化ピクチャの集合である。あるピクチャは、1つのサブシーケンスレイヤ、および1つのサブシーケンスのみに属さなければならない。あるサブシーケンス内の任意のピクチャは、同じあるいはより高いサブシーケンスレイヤの別サブシーケンス内のいかなるピクチャからも予測されてはならない。レイヤ0内のサブシーケンスは、そのサブシーケンスに属さないいかなるピクチャとも独立に復号され得る。

サブシーケンス情報 SEI メッセージは、現アクセスユニットに関わる。ここでは、そのアクセスユニット内の主符号化ピクチャは現ピクチャと呼ばれる。

サブシーケンス SEI メッセージに関連付けられるピクチャによって参照されるシーケンスパラメータセット内の `gaps_in_frame_num_value_allowed_flag` が 1 に等しくない限り、サブシーケンス情報 SEI メッセージは存在してはならない。

`sub_seq_layer_num` は現ピクチャのサブシーケンスレイヤ番号を規定する。`sub_seq_layer_num` が 0 より大きい時、メモリ管理制御操作は現ピクチャのいかなるスライスヘッダ中でも用いられてはならない。現ピクチャが、復号順序でその最初のピクチャが IDR ピクチャであるサブシーケンスに属する時、`sub_seq_layer_num` の値は 0 に等しくなければならない。非ペパ参照フィールドに対して、`sub_seq_layer_num` の値は 0 に等しくなければならない。`sub_seq_layer_num` は 0 から 255 のそれぞれの値を含む範囲でなければならない。

`sub_seq_id` はあるレイヤの内部のサブシーケンスを識別する。現ピクチャが、復号順序でその最初のピクチャが IDR ピクチャであるサブシーケンスに属する時、`sub_seq_id` の値は IDR ピクチャの `idr_pic_id` の値と同じでなければならない。`sub_seq_id` は 0 から 65535 のそれぞれの値を含む範囲でなければならない。

`first_ref_pic_flag` が 1 に等しいことは、現ピクチャがサブシーケンスの復号順序で最初の参照ピクチャであることを規定する。現ピクチャがサブシーケンスの復号順序で最初のピクチャでない時、`first_ref_pic_flag` は 0 に等しくなければならない。

`leading_non_ref_pic_flag` が 1 に等しいことは、現ピクチャがサブシーケンス内で、任意の参照ピクチャに復号順序で先行する非参照ピクチャであること、あるいはサブシーケンスが参照ピクチャを含まないことを規定する。現ピクチャが参照ピクチャであるか、または現ピクチャがサブシーケンス内で復号順序で少なくとも1つの参照ピクチャに続く非参照ピクチャである時、`leading_non_ref_pic_flag` は 0 に等しくなければならない。

`last_pic_flag` が 1 に等しいことは、現ピクチャがサブシーケンスの全ての参照ピクチャおよび非参照ピクチャを含むサブシーケンスの（復号順序で）最後のピクチャであることを示す。現ピクチャがサブシーケンスの（復号順序で）最後のピクチャでないならば、`last_pic_flag` は 0 に等しくなければならない。

現ピクチャはサブシーケンスに次の通り割り当てられる。

- もし1つ以上の下記条件が真であるならば、現ピクチャは復号順序でサブシーケンスの最初のピクチャである。
  - 復号順序でより早いピクチャは現ピクチャと同じ `sub_seq_id` および `sub_seq_layer_num` の値でラベル付けされない
  - `leading_non_ref_pic_flag` の値が 1 に等しく、かつ現ピクチャと同じ `sub_seq_id` と `sub_seq_layer_num` の値をもつ復号順序で直前のピクチャにおける `leading_non_ref_pic_flag` の値が 0 に等しい
  - `first_ref_pic_flag` の値が 1 に等しく、かつ現ピクチャと同じ `sub_seq_id` および `sub_seq_layer_num` の値をもつ復号順序で直前のピクチャにおける `leading_non_ref_pic_flag` の値が 0 に等しい
  - 現ピクチャと同じ `sub_seq_id` と `sub_seq_layer_num` の値をもつ復号順序で直前のピクチャにおける `last_pic_flag` の値が 1 に等しい
- それ以外、現ピクチャは現ピクチャと同じ `sub_seq_id` および `sub_seq_layer_num` の値をもつ復号順序で直前のピクチャと同じサブシーケンスに属する。

`sub_seq_frame_num_flag` が 0 に等しいことは、`sub_seq_frame_num` が存在しないことを規定する。  
`sub_seq_frame_num_flag` が 1 に等しいことは、`sub_seq_frame_num` が存在することを規定する。

`sub_seq_frame_num` は、サブシーケンスの最初の参照ピクチャに対して、および復号順序でサブシーケンスの最初の参照ピクチャに先行する任意の非参照ピクチャに対して、0 に等しくなければならない。  
`sub_seq_frame_num` はさらに次の通り制約される。

- もし現ピクチャが相補的フィールドペアの第2フィールドでないならば、`sub_seq_frame_num` の値はサブシーケンスに属する復号順序で前の参照ピクチャ相対で `MaxFrameNum` によるモジュロ演算において 1 増加されなければならない。
- それ以外（現ピクチャが相補的フィールドペアの第2フィールドである）、`sub_seq_frame_num` の値は相補的フィールドペアの第1フィールドに対する `sub_seq_frame_num` の値と同じでなければならない。

`sub_seq_frame_num` は 0 から `MaxFrameNum-1` のそれぞれの値を含む範囲にななければならない。

現ピクチャが IDR ピクチャである時、それはサブシーケンスレイヤ 0 内で新規のサブシーケンスを開始しなければならない。従って、その `sub_seq_layer_num` は 0 でなければならない、その `sub_seq_id` はサブシーケンスレイヤ 0 内で以前のサブシーケンスと異ならなければならない、その `first_ref_pic_flag` は 1 でなければならない、そしてその `leading_non_ref_pic_flag` は 0 でなければならない。

サブシーケンス情報 SEI メッセージが相補的フィールドペアの両方の符号化フィールドに存在する時、`sub_seq_layer_num`、`sub_seq_id`、`leading_non_ref_pic_flag`、および `sub_seq_frame_num` の値は、存在する時は、これら両方のピクチャで同一でなければならない。

サブシーケンス情報 SEI メッセージが相補的フィールドペアの1つの符号化フィールドに対してのみ存在する時、`sub_seq_layer_num`、`sub_seq_id`、`leading_non_ref_pic_flag`、および `sub_seq_frame_num` の値は、存在する時は、その相補的フィールドペアのもう一方の符号化フィールドにも適用できる。

#### D.2.12 サブシーケンスレイヤ特性SEIメッセージの意味

サブシーケンスレイヤ特性 SEI メッセージは、サブシーケンスレイヤの特性を規定する。

**num\_sub\_seq\_layers\_minus1** に 1 を加えたものは、シーケンス内のサブシーケンスレイヤの数を規定する。**num\_sub\_seq\_layers\_minus1** は 0 から 255 のそれぞれの値を含む範囲でなければならない。

**average\_bit\_rate** と **average\_frame\_rate** のペアは、各々のサブシーケンスレイヤを特徴付ける。**average\_bit\_rate** と **average\_frame\_rate** の最初のペアは、サブシーケンスレイヤ 0 の特性を規定する。2 番目のペアが存在する時、サブシーケンスレイヤ 0 と 1 をあわせた特性を規定する。復号順序で各々のペアは、レイヤ番号 0 から、そのレイヤループカウンタで指定されるレイヤ番号までのサブシーケンスレイヤの範囲に対する特性を規定する。その値は、その値が復号されてからその値の更新が復号されるまで有効である。

**accurate\_statistics\_flag** が 1 に等しいことは、**average\_bit\_rate** および **average\_frame\_rate** の値が統計的に正しい値から丸められていることを示す。**accurate\_statistics\_flag** が 0 に等しいことは、**average\_bit\_rate** および **average\_frame\_rate** の値が推定であり、正しい値から幾分はずれるかもしれないことを示す。

**accurate\_statistics\_flag** が 0 に等しい時、**average\_bit\_rate** および **average\_frame\_rate** の値の計算で用いられる近似の質は、符号化処理によって選択され、本標準によっては規定されない。

**average\_bit\_rate** は 1 秒あたり 1000 ビットの単位で平均ビットレートを示す。上記で規定されたサブシーケンスレイヤの範囲にある全ての NAL ユニットの計算において考慮される。平均ビットレートは、本標準の付属資料 C で規定されるアクセスユニット除去時刻により導出される。下記において、**bTotal** は（復号順序で）（現アクセスユニットの NAL ユニットのビットを含む）サブシーケンスレイヤ特性 SEI メッセージに後続し、（存在すれば）サブシーケンスレイヤ特性 SEI メッセージを含む次のアクセスユニットあるいは（それ以外）ストリームの終了に先行する、全ての NAL ユニット中のビット数である。 $t_1$  は現アクセスユニットの（秒単位の）除去時刻であり、 $t_2$  は（存在すれば）次のサブシーケンスレイヤ特性 SEI メッセージあるいは（それ以外）ストリーム終了の前の（復号順序で）最後のアクセスユニットの（秒単位の）除去時刻である。

**accurate\_statistics\_flag** が 1 に等しい時、次の通り条件が満たされなければならない。

- もし  $t_1$  が  $t_2$  に等しくないならば、以下の条件が真でなければならない。

$$\text{average\_bit\_rate} == \text{Round}(\text{bTotal} \div ((t_2 - t_1) * 1000)) \quad (\text{D-6})$$

- それ以外 ( $t_1$  が  $t_2$  に等しい)、以下の条件が真でなければならない。

$$\text{average\_bit\_rate} == 0 \quad (\text{D-7})$$

**average\_frame\_rate** はフレーム/ (256 秒) 単位の平均フレームレートを示す。上記で規定されたサブシーケンスレイヤの範囲にある全ての NAL ユニットの計算において考慮される。下記において、**fTotal** は現ピクチャ（を含む）と（存在すれば）次のサブシーケンスレイヤ特性 SEI メッセージあるいは（それ以外）ストリーム終了との間のフレーム、相補的フィールドペア、そして非ペアフィールドの数である。 $t_1$  は現アクセスユニットの（秒単位の）除去時刻であり、 $t_2$  は（存在すれば）次のサブシーケンスレイヤ特性 SEI メッセージあるいは（それ以外）ストリーム終了の前の（復号順序で）最後のアクセスユニットの（秒単位の）除去時刻である。

**accurate\_statistics\_flag** が 1 に等しい時、次の通り条件が満たされなければならない。

- もし  $t_1$  が  $t_2$  に等しくないならば、以下の条件が真でなければならない。

$$\text{average\_frame\_rate} == \text{Round}(\text{fTotal} * 256 \div (t_2 - t_1)) \quad (\text{D-8})$$

- それ以外 ( $t_1$  が  $t_2$  に等しい) 、以下の条件が真でなければならない。

$$\text{average\_frame\_rate}==0 \quad (\text{D-9})$$

### D.2.13 サブシーケンス特性SEIメッセージの意味

サブシーケンス特性 SEI メッセージは、サブシーケンスの特性を示す。このメッセージはサブシーケンス間の INTER 予測の依存関係をも示す。このメッセージは、サブシーケンス特性 SEI メッセージが適用されるサブシーケンスの復号順序で最初のアクセスユニットに含まれなくてはならない。このサブシーケンスはここではターゲットサブシーケンスと呼ばれる。

**sub\_seq\_layer\_num** は、ターゲットサブシーケンスのサブシーケンスレイヤ番号を識別する。sub\_seq\_layer\_num は 0 から 255 のそれぞれの値を含む範囲でなければならない。

**sub\_seq\_id** はターゲットサブシーケンスを識別する。sub\_seq\_id は 0 から 65535 のそれぞれの値を含む範囲でなければならない。

**duration\_flag** が 0 に等しいことは、ターゲットサブシーケンスの有効期間が規定されないことを示す。

**sub\_seq\_duration** は、90kHz クロックのクロックチックでターゲットサブシーケンスの有効期間を規定する。

**average\_rate\_flag** が 0 に等しいことは、ターゲットサブシーケンスの平均ビットレートおよび平均フレームレートが無規定を示す。

**accurate\_statistics\_flag** は、どのくらい average\_bit\_rate および average\_frame\_rate の値が信頼できるかを示す。accurate\_statistics\_flag が 1 に等しいことは average\_bit\_rate および average\_frame\_rate が統計的に正しい値から丸められていることを示す。accurate\_statistics\_flag が 0 に等しいことは、average\_bit\_rate および average\_frame\_rate の値が推定であり、統計的に正しい値からはずれるかもしれないことを指示する。

**average\_bit\_rate** はターゲットサブシーケンスの、(1000 ビット)/秒の単位で平均ビットレートを示す。ターゲットサブシーケンスの全ての NAL ユニットの計算において考慮される。平均ビットレートは、C.1.2 小節で規定されるアクセスユニット除去時刻に従って導出される。下記において、nB はサブシーケンス内の全ての NAL ユニットのビット数である。 $t_1$  はサブシーケンスの (復号順序で) 最初のアクセスユニットの (秒単位の) 除去時刻で、 $t_2$  はサブシーケンスの (復号順序で) 最後のアクセスユニットの (秒単位の) 除去時刻である。

accurate\_statistics\_flag が 1 に等しい時、次の通り条件が満たされなければならない。

- もし  $t_1$  が  $t_2$  に等しくないならば、以下の条件が真でなければならない。

$$\text{average\_bit\_rate}==\text{Round}(nB\div((t_2-t_1)*1000)) \quad (\text{D-10})$$

- それ以外 ( $t_1$  が  $t_2$  に等しい) 、以下の条件が真でなければならない。

$$\text{average\_bit\_rate}==0 \quad (\text{D-11})$$

**average\_frame\_rate** は、ターゲットサブシーケンスの、フレーム/(256 秒)単位で平均フレームレートを示す。ターゲットサブシーケンスの全ての NAL ユニットの計算において考慮される。平均フレームレートは、C.1.2 小節で規定されるアクセスユニット除去時刻に従って導出される。下記において、fC はサブシーケン

ス内のフレーム、相補的フィールドペア、そして非ペアフィールドの数である。 $t_1$  はサブシーケンスの（復号順序で）最初のアクセスユニットの（秒単位の）除去時刻で、 $t_2$  はサブシーケンスの（復号順序で）最後のアクセスユニットの（秒単位の）除去時刻である。

`accurate_statistics_flag` が 1 に等しい時、次の通り条件が満たされなければならない。

- もし  $t_1$  が  $t_2$  に等しくないならば、以下の条件が真でなければならない。

$$\text{average\_frame\_rate} == \text{Round}(fC * 256 \div (t_2 - t_1)) \quad (\text{D-12})$$

- それ以外 ( $t_1$  が  $t_2$  に等しい) 、以下の条件が真でなければならない。

$$\text{average\_frame\_rate} == 0 \quad (\text{D-13})$$

`num_referenced_subseqs` はターゲットサブシーケンスのピクチャの中で、INTER 予測の参照ピクチャとして用いられるピクチャを含むサブシーケンスの数を規定する。`num_referenced_subseqs` は 0 から 255 のそれぞれの値を含む範囲でなければならない。

`ref_sub_seq_layer_num`、`ref_sub_seq_id` および `ref_sub_seq_direction` は、ターゲットサブシーケンスのピクチャの中で INTER 予測の参照ピクチャとして用いられるピクチャを含むサブシーケンスを識別する。`ref_sub_seq_direction` に依存して次が適用される。

- もし `ref_sub_seq_direction` が 0 に等しいならば、候補サブシーケンスの集合は、その `sub_seq_id` が `ref_sub_seq_id` に等しく、`ref_sub_seq_layer_num` に等しい `sub_seq_layer_num` をもつサブシーケンスレイヤに属し、そして復号順序でその最初のピクチャが、復号順序でターゲットサブシーケンスの最初のピクチャに先行する、サブシーケンスから構成される。
- それ以外 (`ref_sub_seq_direction` が 1 に等しい) 、候補サブシーケンスの集合は、その `sub_seq_id` が `ref_sub_seq_id` に等しく、`ref_sub_seq_layer_num` に等しい `sub_seq_layer_num` をもつサブシーケンスレイヤに属し、そして復号順序でその最初のピクチャが、復号順序でターゲットサブシーケンスの最初のピクチャに後続する、サブシーケンスから構成される。

ターゲットサブシーケンスの参照として用いられるサブシーケンスは、候補サブシーケンス集合の中で、その最初のピクチャが復号順序でターゲットサブシーケンスの最初のピクチャに最も近接する、サブシーケンスである。

#### D.2.14 フルフレーム凍結SEIメッセージの意味

フルフレーム凍結 SEI メッセージは、規定された条件を満たす、現ピクチャと出力順序で後続する任意のピクチャが表示の内容に影響すべきでないことを示す。任意のアクセスユニットに 1 より多いフルフレーム凍結 SEI メッセージは存在してはならない。

`full_frame_freeze_repetition_period` は、フルフレーム凍結 SEI メッセージの持続性を規定し、もう 1 つのフルフレーム凍結 SEI メッセージ、またはフルフレーム凍結解除 SEI、または符号化ビデオシーケンスの終了がビットストリーム中に存在しなければならないピクチャ順序カウント間隔を規定してもよい。`full_frame_freeze_repetition_period` の値は 0 から 16384 のそれぞれの値を含む範囲でなければならない。

`full_frame_freeze_repetition_period` が 0 に等しいことは、フルフレーム凍結 SEI メッセージが現復号ピクチャだけに適用されることを規定する。

full\_frame\_freeze\_repetition\_period が 1 に等しいことは、フルフレーム凍結 SEI メッセージが、以下の条件のいずれかが真となるまで出力順序で持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる。
- フルフレーム凍結 SEI メッセージ、またはフルフレーム凍結解除 SEI メッセージを含むアクセスユニット中のピクチャが PicOrderCnt(CurrPic) より大きい PicOrderCnt( ) をもち出力される。

full\_frame\_freeze\_repetition\_period が 1 より大きいことは、フルフレーム凍結 SEI メッセージが、以下の条件のいずれかが真となるまで持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる。
- フルフレーム凍結 SEI メッセージ、またはフルフレーム凍結解除 SEI メッセージを含むアクセスユニット中のピクチャが、PicOrderCnt(CurrPic) よりも大きく、かつ PicOrderCnt(CurrPic)+full\_frame\_freeze\_repetition\_period よりも小さいかもしくは等しい PicOrderCnt( ) を持って出力される。

full\_frame\_freeze\_repetition\_period が 1 より大きいことは、もう 1 つのフルフレーム凍結 SEI メッセージ、あるいはフルフレーム凍結解除 SEI メッセージが、PicOrderCnt(CurrPic) よりも大きく、かつ PicOrderCnt(CurrPic)+full\_frame\_freeze\_repetition\_period よりも小さいかもしくは等しい PicOrderCnt( ) を持って出力されるあるアクセスユニット中のあるピクチャにおいて、そのようなピクチャの出力なしにビットストリームが終了するか新しい符号化ビデオシーケンスを開始しない限り、存在しなければならないことを示す。

#### D.2.15 フルフレーム凍結解除SEIメッセージの意味

フルフレーム凍結解除 SEI メッセージは、出力順序で現ピクチャに先行するピクチャとともに送られた任意のフルフレーム凍結 SEI メッセージの効果を打ち消す。フルフレーム凍結解除 SEI メッセージは、現ピクチャおよび出力順序で後続するピクチャが表示の内容に影響を及ぼすべきであることを示す。

いかなるアクセスユニット内においても、1 つを超えるフルフレーム凍結解除 SEI メッセージが存在してはならない。フルフレーム凍結 SEI メッセージを含むアクセスユニット内にはフルフレーム凍結解除 SEI メッセージが存在してはならない。相補的フィールドペアの 2 つのフィールドに対する PicOrderCnt(CurrPic) の値が互いに一致する相補的フィールドペアの一方のフィールドを含むアクセスユニット内にフルフレーム凍結 SEI メッセージが存在する時、フルフレーム凍結解除 SEI メッセージは 2 つのいずれのアクセスユニット内にも存在してはならない。

#### D.2.16 フルフレームスナップショットSEIメッセージの意味

フルフレームスナップショット SEI メッセージは、現フレームがビデオ内容の静止画スナップショットとしてアプリケーションで決められるような使用に対してラベル付けされることを示す。

snapshot\_id はスナップショット識別番号を規定する。snapshot\_id は 0 から  $2^{32}-1$  のそれぞれの値を含む範囲でなければならない。

0 から 255 のそれぞれの値を含む範囲および、512 から  $2^{31}-1$  のそれぞれの値を含む範囲の snapshot\_id の値は、アプリケーションで決められるように使用されて良い。256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の snapshot\_id の値は、TTC によって将来使用のために予約される。

256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の `snapshot_id` の値に遭遇した復号器は、それを無視（ビットストリームから除去して破棄）しなければならない。

#### D.2.17 プログレッシブリファインメント区分開始SEIメッセージの意味

プログレッシブリファインメント区分開始 SEI メッセージは、継続的に動きのあるシーンの表現としてより、むしろ現ピクチャの質改善である 1 つ以上のピクチャの列が後続する現ピクチャとしてラベル付けされる、連続する符号化ピクチャの組の開始を規定する。

以下の条件の 1 つが真となるまで、連続符号化ピクチャのタグ付けされた集合が続かなければならない。以下のある条件が真になる時、復号されるべき次のスライスは、連続符号化ピクチャのタグ付けされた集合に属さない。

1. 次に復号されるべきスライスが IDR ピクチャに属する。
2. `num_refinement_steps_minus1` が 0 より大きく、かつ復号されるべき次のスライスの `frame_num` が  $(\text{currFrameNum} + \text{num\_refinement\_steps\_minus1} + 1) \% \text{MaxFrameNum}$  である。ここで `currFrameNum` は SEI メッセージを含むアクセスユニット中のピクチャの `frame_num` 値である。
3. `num_refinement_steps_minus1` が 0 で、かつこの SEI メッセージのものと同一の `progressive_refinement_id` を伴うプログレッシブリファインメント区分終了 SEI メッセージが復号される。

連続ピクチャのタグ付けされた集合の中にあるピクチャ復号順序は、それらの出力順序と同じであるべきである。

`progressive_refinement_id` はプログレッシブリファインメント操作に対する識別番号を規定する。`progressive_refinement_id` は 0 から  $2^{32}-1$  のそれぞれの値を含む範囲でなければならない。

0 から 255 のそれぞれの値を含む範囲および、512 から  $2^{31}-1$  のそれぞれの値を含む範囲の `progressive_refinement_id` の値は、アプリケーションで決められるものとして用いられても良い。256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の `progressive_refinement_id` の値は、TTC によって将来使用のために予約される。256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の `progressive_refinement_id` の値に遭遇した復号器は、それを無視（ビットストリームから除去して破棄）しなければならない。

`num_refinement_steps_minus1` は連続符号化ピクチャのタグ付けされた集合の中にある参照フレームの数を次の通り規定する。

- もし `num_refinement_steps_minus1` が 0 に等しいならば、連続符号化ピクチャのタグ付けされた集合の中にある参照フレームの数が未知であることを規定する。
- それ以外、連続符号化ピクチャのタグ付けされた集合の中にある参照フレームの数は `num_refinement_steps_minus1 + 1` に等しい。

`num_refinement_steps_minus1` は 0 から `MaxFrameNum-1` のそれぞれの値を含む範囲でなければならない。

#### D.2.18 プログレッシブリファインメント区分終了SEIメッセージの意味

プログレッシブリファインメント区分終了 SEI メッセージは、初期ピクチャとしてプログレッシブリファインメント区分開始 SEI メッセージの使用でラベル付けされている、初期ピクチャの質改善である 1 つ以上

のピクチャの列が後続する連続する符号化ピクチャの組の終了および現ピクチャで終了することを規定する。

**progressive\_refinement\_id** はプログレッシブリファインメント操作に対する識別番号を規定する。**progressive\_refinement\_id** は 0 から  $2^{32}-1$  のそれぞれの値を含む範囲でなければならない。

プログレッシブリファインメント区分終了 SEI メッセージは、同一の **progressive\_refinement\_id** の値を伴うプログレッシブリファインメント区分開始 SEI メッセージを用いて以前に開始した、任意のプログレッシブリファインメント区分の終了を規定する。

0 から 255 のそれぞれの値を含む範囲および、512 から  $2^{31}-1$  のそれぞれの値を含む範囲の **progressive\_refinement\_id** の値は、アプリケーションで決められるものとして用いられても良い。256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の **progressive\_refinement\_id** の値は、TTC によって将来使用のために予約される。256 から 511 のそれぞれの値を含む範囲および、 $2^{31}$  から  $2^{32}-1$  のそれぞれの値を含む範囲の **progressive\_refinement\_id** の値に遭遇した復号器は、それを無視（ビットストリームから除去して破棄）しなければならない。

#### D.2.19 動き制約スライスグループセットSEIメッセージの意味

この SEI メッセージはスライスグループ境界を越える INTER 予測が以下に規定されるように制約されることを示す。存在する時、メッセージは、それが 7.4.1.2.3 小小小節で規定されるように IDR アクセスユニットに関連付けられる場所でのみ現れなければならない。

この SEI メッセージに対するターゲットピクチャセットは、関連付けられた主符号化 IDR ピクチャ（を含む）で始まり、後に続く主符号化 IDR ピクチャ（を含まない）あるいは、後に続く主符号化 IDR ピクチャが無い時、復号順序でビットストリームの最後の主符号化ピクチャ（を含む）で終わる、復号順序で全ての連続する主符号化ピクチャを含む。スライスグループセットは、**slice\_group\_id[i]** シンタックス要素で識別される、1 つ以上のスライスグループの集まりである。**separate\_colour\_plane\_flag** が 1 に等しい時、用語“主符号化ピクチャ”は同じ **colour\_plane\_id** を持つ NAL ユニットに対応する主符号化ピクチャの一部を表す。

この SEI メッセージは、ターゲットピクチャセット内のそれぞれのピクチャに対して、INTER 予測処理が次のように制約されることを示す。スライスグループセット外のサンプル値と、スライスグループセット外の 1 つ以上のサンプル値を用いて導出される分数サンプル位置にあるサンプル値が、スライスグループセット内部のいかなるサンプルの INTER 予測にも用いられることはない。

**num\_slice\_groups\_in\_set\_minus1+1** は、スライスグループセット内のスライスグループの数を規定する。**num\_slice\_groups\_in\_set\_minus1** の許容範囲は、0 から **num\_slice\_groups\_minus1** でそれぞれの値を含む。**num\_slice\_groups\_minus1** の許容範囲は、付属資料 A で規定される。

**slice\_group\_id[i]** は、スライスグループセット内に含まれる（複数の）スライスグループを識別する。その許容範囲は 0 から **num\_slice\_groups\_in\_set\_minus1** のそれぞれの値を含む。**slice\_group\_id[i]** シンタックス要素のサイズは  $\text{Ceil}(\text{Log}_2(\text{num\_slice\_groups\_minus1}+1))$  ビットである。

**exact\_sample\_value\_match\_flag** が 0 に等しいことは、ターゲットピクチャセット内で、スライスグループセットに属さないマクロブロックが復号されない時、スライスグループセット内のそれぞれのサンプル値が、全てのマクロブロックが復号される時の同じサンプルの値として完全に同じである必要がないことを示す。**exact\_sample\_value\_match\_flag** が 1 に等しいことは、ターゲットピクチャセット内で、スライスグループセットに属さないマクロブロックが復号されない時、スライスグループセット内のそれぞれのサンプル値が、タ

ターゲットピクチャセット内の全てのマクロブロックが復号される時の同じサンプルの値として完全に同じでなければならないことを示す。

記1 ターゲットピクチャセット中の全てのスライスで、`disable_deblocking_filter_idc`が2に等しい時、`exact_sample_value_match_flag`は1となるべきである。

`pan_scan_rect_flag` が 0 に等しいことは、`pan_scan_rect_id` が存在しないことを規定する。`pan_scan_rect_flag` が 1 に等しいことは、`pan_scan_rect_id` が存在することを規定する。

`pan_scan_rect_id` は、規定されたスライスグループセットが少なくともターゲットピクチャセット内で `pan_scan_rect_id` で識別されるパン・スキャン矩形を覆うことを示す。

記2 複数の `motion_constrained_slice_group_set` SEIメッセージは、同じIDRピクチャに関連付けられても良い。従って、1つより多くのスライスグループセットは、ターゲットピクチャセット内でアクティブであっても良い。

記3 スライスグループセット内のスライスグループのサイズ、形状および位置はターゲットピクチャセット内で変化しても良い。

#### D.2.20 フィルムグレイン特性SEIメッセージの意味

この SEI メッセージは、復号器にフィルムグレイン合成のパラメータ化されたモデルを提供する。例えば、符号器は、元の原ビデオ素材内に存在しておりかつ前処理フィルタ技術によって除去されたフィルムグレインを特徴付けるためにフィルムグレイン特性 SEI メッセージを使っても良い。表示処理における復号画像上へのシミュレートされたフィルムグレインの合成はオプションであり、かつ本標準で規定される復号処理に影響を与えることはない。もし表示処理において復号画像上へのシミュレートされたフィルムグレインの合成が実施されるならば、合成を実施する手法がフィルムグレイン特性 SEI メッセージ内で提供されるようなフィルムグレインに対するパラメータ化されたモデルと同じであることの要求はない。

記1 表示処理は本標準では規定されない。

`film_grain_characteristics_cancel_flag` が 1 に等しいことは、SEI メッセージが出力順序で以前の任意のフィルムグレイン特性 SEI メッセージの持続を取り消すことを示す。`film_grain_characteristics_cancel_flag` が 0 に等しいことは、フィルムグレインモデリング情報が後続することを示す。

`model_id` は付表 D-5/JT-H264 で規定されるようなフィルムグレインシミュレーションモデルを識別する。`model_id` の値は、0 から 1 のそれぞれの値を含む範囲でなければならない。

付表D-5/JT-H264 `model_id`値  
(ITU-T H.264)

値	記述
0	周波数フィルタ
1	自己回帰
2	予約
3	予約

`separate_colour_description_present_flag` が 1 に等しいことは、SEI メッセージ内で規定されるフィルムグレイン特性に対する異なる色空間記述がフィルムグレイン特性 SEI メッセージシンタックス内に存在するこ

とを示す。 `separate_colour_description_present_flag` が 0 に等しいことは、SEI メッセージ内で規定されるフィルムグレイン特性に対する色空間記述が、E.2.1 小節で規定されるような符号化ビデオシーケンスに対するものと同じであることを示す。

記2 `separate_colour_description_present_flag`が1に等しい時、SEIメッセージ内で規定されるフィルムグレイン特性に対して規定される色空間は、E.2.1小節で規定されるような符号化ビデオに対して規定される色空間と異なっても良い。

`film_grain_bit_depth_luma_minus8` に 8 を加えたものは、SEI メッセージ内で規定されるフィルムグレイン特性の輝度コンポーネントに対して用いられるビット深度を規定する。`film_grain_bit_depth_luma_minus8` がフィルムグレイン特性 SEI メッセージ内に存在しない場合、`film_grain_bit_depth_luma_minus8` の値は `bit_depth_luma_minus8` に等しいものと推定されなければならない。

`filmGrainBitDepth[0]`の値は以下で導出される。

$$\text{filmGrainBitDepth}[0]=\text{film\_grain\_bit\_depth\_luma\_minus8}+8 \quad (\text{D-14})$$

`film_grain_bit_depth_chroma_minus8` に 8 を加えたものは、SEI メッセージ内で規定されるフィルムグレイン特性の Cb および Cr コンポーネントに対して用いられるビット深度を規定する。`film_grain_bit_depth_chroma_minus8` がフィルムグレイン特性 SEI メッセージ内に存在しない場合、`film_grain_bit_depth_chroma_minus8` の値は `bit_depth_chroma_minus8` に等しいものと推定されなければならない。

`c=1` および `2` に対する `filmGrainBitDepth[c]`の値は以下で導出される。

$$\text{filmGrainBitDepth}[c]=\text{film\_grain\_bit\_depth\_chroma\_minus8}+8, \text{ c}=1,2 \quad (\text{D-15})$$

`film_grain_full_range_flag` は、以下のものを除いて、E.2.1 小節で規定される `video_full_range_flag` シンタックス要素に対するものと同じ意味を有する。

- `film_grain_full_range_flag` は符号化ビデオシーケンスに対して用いられる色空間とは異なり、SEI メッセージ内で規定されるフィルムグレイン特性の色空間を規定する。
- `film_grain_full_range_flag` がフィルムグレイン特性 SEI メッセージ内に存在しない場合、`film_grain_full_range_flag` の値は `video_full_range_flag` に等しいものと推定されなければならない。

`film_grain_colour_primaries` は、以下のものを除いて、E.2.1 小節で規定される `colour_primaries` シンタックス要素に対するものと同じ意味を有する。

- `film_grain_colour_primaries` は符号化ビデオシーケンスに対して用いられる色空間とは異なり、SEI メッセージ内で規定されるフィルムグレイン特性の色空間を規定する。
- `film_grain_colour_primaries` がフィルムグレイン特性 SEI メッセージ内に存在しない場合、`film_grain_colour_primaries` の値は `colour_primaries` に等しいものと推定されなければならない。

`film_grain_transfer_characteristics` は、以下のものを除いて、E.2.1 小節で規定される `transfer_characteristics` シンタックス要素に対するものと同じ意味を有する。

- `film_grain_transfer_characteristics` は符号化ビデオシーケンスに対して用いられる色空間とは異なり、SEI メッセージ内で規定されるフィルムグレイン特性の色空間を規定する。
- `film_grain_transfer_characteristics` がフィルムグレイン特性 SEI メッセージ内に存在しない場合、

film\_grain\_transfer\_characteristics の値は transfer\_characteristics に等しいものと推定されなければならない。

film\_grain\_matrix\_coefficients は、以下のものを除いて、E.2.1 小節で規定される matrix\_coefficients シンタックス要素に対するものと同じ意味を有する。

- film\_grain\_matrix\_coefficients は符号化ビデオシーケンスに対して用いられる色空間とは異なり、SEI メッセージ内で規定されるフィルムグレイン特性の色空間を規定する。
- film\_grain\_matrix\_coefficients がフィルムグレイン特性 SEI メッセージ内に存在しない場合、film\_grain\_matrix\_coefficients の値は matrix\_coefficients に等しいものと推定されなければならない。
- film\_grain\_matrix\_coefficients に対して許される値は、chroma\_format\_idc の値によって制限されない。

フィルムグレイン特性SEIメッセージ内で規定されるフィルムグレイン特性のchroma\_format\_idcは3(4:4:4)に等しいものと推定されなければならない。

記3 表示処理で用いられるフィルムグレイン生成機能の実行に対して特定手法の使用は要求されないため、復号器は、もし望まれるならば、その他の色差フォーマット(4:2:0あるいは4:2:2)に対するフィルムグレインをシミュレートするために、フィルムグレイン生成実行前の復号されたビデオを(本標準で規定されない手法を用いて)アップコンバートするよりはむしろ、色差に対するモデル情報をダウンコンバートしても良い。

blending\_mode\_id は復号された画像にシミュレートされたフィルムグレインを付表 D-6/JT-H264 で規定されるようにブレンドするために用いられるブレンドモードを識別する。blending\_mode\_id は、0 から 1 のそれぞれを含む範囲になければならない。

付表D-6/JT-H264 blending\_model\_id値  
(ITU-T H.264)

値	記述
0	加算
1	乗算
2	予約
3	予約

blending\_mode\_id に依存して、ブレンドモードが次のように規定される。

- もし blending\_mode\_id が 0 に等しいならば、ブレンドモードは以下で規定される加算である。

$$I_{\text{grain}}[x,y,c]=\text{Clip3}(0,(1\ll\text{filmGrainBitDepth}[c])-1,I_{\text{decoded}}[x,y,c]+G[x,y,c]) \quad (\text{D-16})$$

- それ以外 (blending\_mode\_id が 1 に等しい) 、ブレンドモードは以下で規定される乗算である。

$$I_{\text{grain}}[x,y,c]=\text{Clip3}(0,(1\ll\text{filmGrainBitDepth}[c])-1,I_{\text{decoded}}[x,y,c]*(1+G[x,y,c])) \quad (\text{D-17})$$

ここで、 $I_{\text{decoded}}[x,y,c]$  は復号画像  $I_{\text{decoded}}$  の色コンポーネント  $c$  の座標  $x,y$  におけるサンプル値を表わし、 $G[x,y,c]$  は同じ位置と色コンポーネントでのシミュレートされたフィルムグレイン値であり、そして  $\text{filmGrainBitDepth}[c]$  は、配列  $I_{\text{grain}}[x,y,c]$  の各サンプルの固定長符号無し二進表現で用いられるビット数である。

log2\_scale\_factor はフィルムグレイン特性等式の中で用いられるスケール因子を規定する。

comp\_model\_present\_flag[c] が 0 に等しいことは、フィルムグレインが  $c$  番目の色コンポーネント上でモデ

ル化されていないことを示す。ここで、0 に等しい  $c$  は輝度コンポーネントを参照し、1 に等しい  $c$  は  $C_b$  コンポーネントを参照し、そして 2 に等しい  $c$  は  $C_r$  コンポーネントを参照する。`comp_model_present_flag[c]` が 1 に等しいことは、色コンポーネント  $c$  上のフィルムグレインのモデリングを規定するシンタックス要素が SEI メッセージ内に存在することを示す。

`num_intensity_intervals_minus1[c]` に 1 を加えたものは、それに対する個々のモデル値の組が見積もられている強度区間の数を規定する。

記4 強度区間は多世代フィルムグレインをシミュレートするために部分的に重なり合っても良い。

`num_model_values_minus1[c]` に 1 を加えたものは、その中でフィルムグレインがモデル化されている各々の強度区間に対して存在するモデル値の数を規定する。`num_model_values_minus1[c]` の値は 0 から 5 のそれぞれの値を含む範囲になければならない。

`intensity_interval_lower_bound[c][i]` は、それに対してモデル値の組が適用される強度レベル区間  $i$  の下界を規定する。

`intensity_interval_upper_bound[c][i]` は、それに対してモデル値の組が適用される強度レベル区間  $i$  の上界を規定する。

`model_id` に依存して、モデル値の組の選択が次の通り規定される。

- もし `model_id` が 0 に等しいならば、 $b_{avg}$  と呼ばれる、 $I_{decoded}$  内の  $16 \times 16$  サンプルからなる各々のブロック  $b$  の平均値が、ブロック内の全てのサンプルに適用されるインデックス  $s[j]$  をもつモデル値の組を選択するために用いられる。

```
for(i=0,j=0;i<=num_intensity_intervals_minus1;i++)
    if( $b_{avg} \geq intensity\_interval\_lower\_bound[c][i]$  &&  $b_{avg} \leq intensity\_interval\_upper\_bound[c][i]$ ){
         $s[j]=i$ 
         $j++$ 
    }
```

(D-18)

- それ以外 (`model_id` が 1 に等しい)、フィルムグレインを生成するために用いられるモデル値の組が  $I_{decoded}$  内の各々のサンプル値に対して次の通り選択される。

```
for(i=0,j=0;i<=num_intensity_intervals_minus1;i++)
    if( $I_{decoded}[x,y,c] \geq intensity\_interval\_lower\_bound[c][i]$  &&
         $I_{decoded}[x,y,c] \leq intensity\_interval\_upper\_bound[c][i]$ ){
         $s[j]=i$ 
         $j++$ 
    }
```

(D-19)

定義されたどの区間にも入らないサンプルは、グレイン生成機能によって変更されない。1 つより多くの区間に入るサンプルは、多世代グレインを起こすであろう。多世代グレインは、各々の強度区間に対して独立に計算されたグレインを加えることによって得られる。

`comp_model_value[c][i][j]` は、色コンポーネント  $c$  および強度区間  $i$  に対して存在するモデル値の各々を表現する。モデル値の組は `model_id` の値に依存して異なる意味を持つ。`comp_model_value[c][i][j]` の値は次の通り制約されなければならない、さらに本小節の他の場所で付加的に制約されるかも知れない。

- もし model\_id が 0 に等しいならば、comp\_model\_value[c][i][j] は 0 から  $2^{\text{filmGrainBitDepth}[c]-1}$  のそれぞれの値を含む範囲になければならない。
- それ以外 (model\_id が 1 に等しい)、comp\_model\_value[c][i][j] は  $-2^{(\text{filmGrainBitDepth}[c]-1)}$  から  $2^{(\text{filmGrainBitDepth}[c]-1)-1}$  のそれぞれの値を含む範囲になければならない。

model\_id に依存して、フィルムグレインの合成が次の通りモデル化される。

- もし model\_id が 0 に等しいならば、周波数フィルタモデルが  $c=0..2$ 、 $x=0..PicWidthInSamples_L$ 、そして  $y=0..PicHeightInSamples_L$  に対して以下で規定されるように元のフィルムグレインをシミュレートすることを有効にする。

$$G[x,y,c]=(\text{comp\_model\_value}[c][s][0]*Q[x,y,c]+\text{comp\_model\_value}[c][s][5]*G[x,y,c-1])>>\log2\_scale\_factor \quad (D-20)$$

ここで、 $Q[c]$  は正規化ガウス分布で生成されるランダム値要素  $\text{gaussRv}_{ij}$  (平均値が 0 で単位分散を備えた独立同分布のガウスランダム変数サンプル) をもつ  $16 \times 16$  ブロック  $\text{gaussRv}$  をフィルタすることで生成される 2 次元ランダム過程であり、そして、等式の右辺で用いられる要素  $G[x,y,c-1]$  の値は、 $c-1$  が 0 よりも小さい時には 0 に等しいものと推定される。

記5 正規化ガウスランダム値は、 $uRv_0$  および  $uRv_1$  として記される、2つの独立で 0 から 1 の区間上で一様分布のランダム値 (0 に等しくない) から、以下で規定されるボックス-ミュラー変換を用いて生成される。

$$\text{gaussRv}_{ij} = \sqrt{-2 * \text{Ln}(uRv_0)} * \text{Cos}(2 * \pi * uRv_1) \quad (D-21)$$

ここで、 $\text{Ln}(x)$  は  $x$  の自然対数 ( $e$  を底とする対数、ここで  $e$  は自然対数の底である定数 2.718 281 828...) であり、 $\text{Cos}(x)$  はラジアン単位の引数  $x$  に作用するコサイン三角関数であり、そして  $\pi$  はアルキメデスの定数 3.141 592 653... である。

ブロック  $\text{gaussRv}$  の離散コサイン変換 (DCT) 領域における帯域通過フィルタが次の通り実施されても良い。

```
for(y=0;y<16;y++)
  for(x=0;x<16;x++)
    if((x<comp_model_value[c][s][3] && y<comp_model_value[c][s][4]) ||
       x>comp_model_value[c][s][1] || y>comp_model_value[c][s][2])
      gaussRv[x,y]=0
    filteredRv=IDCT16x16(gaussRv)
```

(D-22)

ここで、 $\text{IDCT16x16}(z)$  は、以下で規定される  $16 \times 16$  行列引数  $z$  に作用するユニタリ逆離散コサイン変換 (IDCT) を参照する。

$$\text{IDCT16x16}(z)=r*z*r^T \quad (D-23)$$

ここで、上付き文字  $T$  は行列転置を示し、そして  $r$  は以下で規定される要素  $r_{ij}$  をもつ  $16 \times 16$  行列である。

$$r_{ij} = \frac{((i == 0) ? 1 : \sqrt{2})}{4} \text{Cos}\left(\frac{i * (2 * j + 1) * \pi}{32}\right) \quad (D-24)$$

ここで、 $\text{Cos}(x)$ はラジアン単位の引数  $x$  に作用するコサイン三角関数であり、そして  $\pi$  はアルキメデスの定数 3.141 592 653... である。

$Q[c]$ は周波数フィルタされたブロック  $\text{filteredRv}$  によって形成される。

記6 符号化されたモデル値は $16 \times 16$ のブロックに基づいているが、復号器の実装は他のブロックサイズを用いても良い。例えば、 $8 \times 8$ ブロックでIDCTを実装する復号器は、1.4に等しい  $i$  に対する符号化されたモデル値の組  $\text{comp\_model\_value}[c][s][i]$  を因子2でダウンコンバートすべきである。

記7 周波数フィルタされたブロック  $\text{filteredRv}$  のモザイク結果として生まれる視認できるブロックの度合いを低減するために、復号器は周波数フィルタされたブロックの境界へ低域通過フィルタを適用しても良い。

- それ以外 ( $\text{model\_id}$  が 1 に等しい)、自己回帰モデルが  $c=0.2$ 、 $x=0..PicWidthInSamples_L$ 、そして  $y=0..PicHeightInSamples_L$  に対して以下で規定されるように元のフィルムグレインをシミュレートすることを有効にする。

$$\begin{aligned}
 G[x,y,c] = & (\text{comp\_model\_value}[c][s][0] * n[x,y,c] + \\
 & \text{comp\_model\_value}[c][s][1] * (G[x-1,y,c] + ((\text{comp\_model\_value}[c][s][4] * G[x,y-1,c]) \gg \\
 & \quad \log_2\_scale\_factor)) + \\
 & \text{comp\_model\_value}[c][s][3] * (((\text{comp\_model\_value}[c][s][4] * G[x-1,y-1,c]) \gg \\
 & \quad \log_2\_scale\_factor) + G[x+1,y-1,c]) + \\
 & \text{comp\_model\_value}[c][s][5] * (G[x-2,y,c] + \\
 & \quad ((\text{comp\_model\_value}[c][s][4] * \text{comp\_model\_value}[c][s][4] * G[x,y-2,c]) \gg \\
 & \quad (2 * \log_2\_scale\_factor)))) + \\
 & \text{comp\_model\_value}[c][s][2] * G[x,y,c-1]) \gg \log_2\_scale\_factor
 \end{aligned}
 \tag{D-25}$$

ここで、 $n[x,y,c]$ は正規化ガウス分布を持つランダム値 ( $x$ 、 $y$  および  $c$  のそれぞれの値に対する、平均値が 0 で単位分散を備えた独立同分布のガウシアンランダム変数サンプル) であり、そして、等式の右辺で用いられる要素  $G[x,y,c-1]$ の値は、以下の条件のいずれかが真の時には 0 に等しいものと推定される。

- $x$  は 0 より小さい。
- $y$  は 0 より小さい。
- $x$  は  $PicWidthInSamples_L$  より大きい、あるいは等しい。
- $c$  は 0 より小さい。

$\text{comp\_model\_value}[c][i][0]$ は、 $\text{model\_id}$  によって規定されるモデルに対する第 1 のモデル値を供給する。 $\text{comp\_model\_value}[c][i][0]$ は、等式 D-20 から D-23 で規定される生成関数において、ガウシアンノイズ項の標準偏差に対応する。

$\text{comp\_model\_value}[c][i][1]$ は、 $\text{model\_id}$  によって規定されるモデルに対する第 2 のモデル値を供給する。 $\text{comp\_model\_value}[c][i][1]$ は、0 以上かつ 16 未満でなければならない。

フィルムグレイン特性 SEI メッセージの中に存在しない場合、 $\text{comp\_model\_value}[c][i][1]$ は次の通り推定されなければならない。

- もし  $\text{model\_id}$  が 0 に等しいならば、 $\text{comp\_model\_value}[c][i][1]$ は 8 に等しいと推定されなければならない。
- それ以外 ( $\text{model\_id}$  が 1 に等しい)、 $\text{comp\_model\_value}[c][i][1]$ は 0 に等しいと推定されなければならない。

い。

**comp\_model\_value[c][i][1]**は、次の通り解釈される。

- もし **model\_id** が 0 に等しいならば、**comp\_model\_value[c][i][1]**は、16×16 ランダム値ブロックの DCT をフィルタするために使用される水平の高域阻止周波数を示す。
- それ以外 (**model\_id** が 1 に等しい) 、**comp\_model\_value[c][i][1]**は、隣接サンプル(x-1,y)および(x,y-1)に対する 1 次の空間相関を示す。

**comp\_model\_value[c][i][2]**は、**model\_id** によって規定されるモデルに対する第 3 のモデル値を供給する。**comp\_model\_value[c][i][2]**は、0 以上かつ 16 未満でなければならない。

フィルムグレイン特性 SEI メッセージの中に存在しない場合、**comp\_model\_value[c][i][2]**は次の通り推定されなければならない。

- もし **model\_id** が 0 に等しいならば、**comp\_model\_value[c][i][2]**は **comp\_model\_value[c][i][1]**に等しいと推定されなければならない。
- それ以外 (**model\_id** が 1 に等しい) 、**comp\_model\_value[c][i][2]**は 0 に等しいと推定されなければならない。

**comp\_model\_value[c][i][2]**は、次の通り解釈される。

- もし **model\_id** が 0 に等しいならば、**comp\_model\_value[c][i][2]**は、16×16 ランダム値ブロックの DCT をフィルタするために使用される垂直の高域阻止周波数を示す。
- それ以外 (**model\_id** が 1 に等しい) 、**comp\_model\_value[c][i][2]**は、連続した色成分間の色相関を示す。

**comp\_model\_value[c][i][3]**は、**model\_id** によって規定されるモデルに対する第 4 のモデル値を供給する。**comp\_model\_value[c][i][3]**は、0 以上かつ **comp\_model\_value[c][i][1]**以下でなければならない。

フィルムグレイン特性 SEI メッセージの中に存在しない場合、**comp\_model\_value[c][i][3]**は 0 に等しいと推定されなければならない。

**comp\_model\_value[c][i][3]**は、次の通り解釈される。

- もし **model\_id** が 0 に等しいならば、**comp\_model\_value[c][i][3]**は、16×16 ランダム値ブロックの DCT をフィルタするために使用される水平の低域阻止周波数を示す。
- それ以外 (**model\_id** が 1 に等しい) 、**comp\_model\_value[c][i][3]**は、隣接サンプル(x-1,y-1)および(x+1,y-1)に対する 1 次の空間相関を示す。

**comp\_model\_value[c][i][4]**は、**model\_id** によって規定されるモデルに対する第 5 のモデル値を供給する。**comp\_model\_value[c][i][4]**は、0 以上かつ **comp\_model\_value[c][i][2]**以下でなければならない。

フィルムグレイン特性 SEI メッセージの中に存在しない場合、**comp\_model\_value[c][i][4]**は **model\_id** に等しいと推定されなければならない。

**comp\_model\_value[c][i][4]**は、次の通り解釈される。

- もし **model\_id** が 0 に等しいならば、**comp\_model\_value[c][i][4]**は、16×16 ランダム値ブロックの DCT をフィルタするために使用される垂直の低域阻止周波数を示す。

- それ以外 (model\_id が 1 に等しい) 、comp\_model\_value[c][i][4]は、モデル化されたグレインのアスペクト比を示す。

**comp\_model\_value[c][i][5]**は、model\_id によって規定されるモデルに対する第 6 のモデル値を供給する。

フィルムグレイン特性 SEI メッセージの中に存在しない場合、comp\_model\_value[c][i][5]は 0 に等しいと推定されなければならない。

comp\_model\_value[c][i][5]は、次の通り解釈される。

- もし model\_id が 0 に等しいならば、comp\_model\_value[c][i][5]は、連続した色成分間の色相関を示す。
- それ以外 (model\_id が 1 に等しい) 、comp\_model\_value[c][i][5]は、隣接サンプル(x,y-2)および(x-2,y)に対する 2 次の空間相関を示す。

**film\_grain\_characteristics\_repetition\_period** は、フィルムグレイン特性 SEI メッセージの持続性を規定し、さらにもう 1 つのフィルムグレイン特性 SEI メッセージ、または符号化ビデオシーケンスの終了がビットストリーム中に存在しなければならないピクチャ順序カウント間隔を規定してもよい。film\_grain\_characteristics\_repetition\_period の値は 0 から 16384 までの、それぞれを含む範囲でなければならない。

film\_grain\_characteristics\_repetition\_period が 0 であることは、フィルムグレイン特性 SEI メッセージが現復号ピクチャのみに適用されることを規定する。

film\_grain\_characteristics\_repetition\_period が 1 であることは、以下の条件のいずれかが真となるまで、出力順序でフィルムグレイン特性 SEI メッセージが持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる、あるいは
- フィルムグレイン特性 SEI メッセージを含むアクセスユニット中のピクチャが、PicOrderCnt(CurrPic)よりも大きい PicOrderCnt()を持って出力される。

film\_grain\_characteristics\_repetition\_period が 1 よりも大きいことは、以下の条件のいずれかが真となるまで、フィルムグレイン特性 SEI メッセージが持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる、あるいは
- フィルムグレイン特性 SEI メッセージを含むアクセスユニット中のピクチャが、PicOrderCnt(CurrPic)よりも大きく、かつ PicOrderCnt(CurrPic)+film\_grain\_characteristics\_repetition\_period よりも小さいかもしくは等しい PicOrderCnt()を持って出力される。

film\_grain\_characteristics\_repetition\_period が 1 よりも大きいことは、もう 1 つのフィルムグレイン特性 SEI メッセージが、PicOrderCnt(CurrPic) よりも大きく、かつ PicOrderCnt(CurrPic)+film\_grain\_characteristics\_repetition\_period よりも小さいかもしくは等しい PicOrderCnt()を持って出力されるあるアクセスユニット中のあるピクチャにおいて、そのようなピクチャの出力なしにビットストリームが終了するか新しい符号化ビデオシーケンスが開始しない限り、存在しなければならないことを示す。

#### D.2.21 デブロックフィルタ表示選好SEIメッセージシンタックス

この SEI メッセージは、出力される各復号ピクチャの表示に対して、切り出された 8.7 節で規定されるデ

ブロックフィルタ処理の結果、もしくは切り出された 8.5.13 小節で規定されるデブロックフィルタ処理前のピクチャ構築処理の結果、のいずれの表示が符号器によって選好されるかの指示を復号器へ供給する。

記1 本標準で表示処理は規定されない。デブロックフィルタ表示選好SEIメッセージで表される符号器の選好として何を指示するかを符号器が決定するための手段も本標準では規定されず、またデブロックフィルタ表示選好SEIメッセージ中で表される選好の表現は表示処理においていかなる要求をも課さない。

**deblocking\_display\_preference\_cancel\_flag** が 1 に等しいことは、その SEI メッセージが、出力順序で以前の任意のデブロックフィルタ表示選好 SEI メッセージの持続を取り消すことを示す。  
**deblocking\_display\_preference\_cancel\_flag** が 0 に等しいことは、**display\_prior\_to\_deblocking\_preferred\_flag** および **deblocking\_display\_preference\_repetition\_period** が続くことを示す。

記2 デブロックフィルタ表示選好 SEI メッセージの非存在時、または **deblocking\_display\_preference\_cancel\_flag** が 1 に等しいデブロックフィルタ表示選好 SEI メッセージの受信後、復号器は、出力される各復号ピクチャの表示に対して、切り出された 8.5.13 小節で規定されるデブロックフィルタ処理前のピクチャ構築処理の結果の表示よりも、切り出された 8.7 節で規定されるデブロックフィルタ処理の結果の表示が選好されたと推定すべきである。

**display\_prior\_to\_deblocking\_preferred\_flag** が 1 に等しいことは、表示処理（これは本標準で規定されない）において、符号器の選好は、付属資料 C で規定される切り出され出力される各ピクチャに対して、切り出された 8.7 節で規定されるデブロックフィルタ処理の結果よりも、切り出された 8.5.13 小節で規定されるデブロックフィルタ処理前のピクチャ構築処理の結果を表示することであることを示す。

**display\_prior\_to\_deblocking\_preferred\_flag** が 0 に等しいことは、表示処理（これは本標準で規定されない）において、符号器の選好は、付属資料 C で規定される切り出され出力される各ピクチャに対して、切り出された 8.5.13 節で規定されるデブロックフィルタ処理前のピクチャ構築処理の結果よりも、切り出された 8.7 節で規定されるデブロックフィルタ処理の結果を表示することであることを示す。

記3 デブロックフィルタ表示選好 SEI メッセージの存在または非存在、および **display\_prior\_to\_deblocking\_preferred\_flag** の値は、本標準で規定される復号処理の要件に影響を与えない。むしろ、それは復号処理に対して本標準の要件を満たすことに加えて、別の様式で表示処理（本標準で規定されない）を行うことにより、高い表示品質がいつ得られうるかの指示を供給するのみである。デブロックフィルタ表示選好 SEI メッセージを使用する符号器は、使用するプロファイルおよびレベルに対する付属資料 A で規定される DPB 容量の使用を符号器が制限しない限り、いくつかの復号器は、表示のためのピクチャ並替または遅延を行う際、8.7 節で規定されるデブロックフィルタ処理結果の蓄積に加え、8.5.13 小節で規定されるデブロックフィルタ処理前のピクチャ構築処理結果の蓄積のための十分なメモリ容量を持っていないかもしれず、それ故にそのような復号器は選好の指示による恩恵を得られないであろうことを意識して、設計されるべきである。DPB 容量の使用を制限することにより、表示のために好ましいとして指示されているフィルタされないピクチャの蓄積のために、それらのピクチャに対する出力時刻に到達するまで、その余った容量を使用することを復号器に許しつつ、符号器は付属資料 A で規定される DPB 容量の少なくとも半分を使用可能である。

**dec\_frame\_buffering\_constraint\_flag** が 1 に等しいことは、符号化ビデオシーケンスが、ピクチャタイミング SEI メッセージの **dpb\_output\_delay** により規定される出力時刻で、デブロックフィルタ表示選好 SEI メッセージにより指示される、フィルタされた、またはフィルタされない復号ピクチャの出力を可能とするために、 $\text{Max}(1, \text{max\_dec\_frame\_buffering})$  よりも多いフレームバッファを持った復号ピクチャバッファを要求しないように、**max\_dec\_frame\_buffering** により規定される HRD 復号ピクチャバッファ (DPB) のフレームバッファリング容量の使用が制約されていることを示す。**dec\_frame\_buffering\_constraint\_flag** が 0 に等しいことは、HRD におけるフレームバッファリング容量の使用が、1 に等しい **dec\_frame\_buffering\_constraint\_flag** によって示されるような方法で制約されてもよく、されなくてもよいことを示す。

**dec\_frame\_buffering\_constraint\_flag** が 1 に等しい場合に課せられる制約を決定する目的で、任意の与えられた時刻点において、ピクチャを含む DPB の各フレームバッファにより使用されるフレームバッファリング容量の数量は、次の通り導出される。

- もし、フレームバッファに対して以下の評価基準の両方が満たされるならば、そのフレームバッファは、蓄積のための容量として2つのフレームバッファを使用するとみなされる。
  - フレームバッファが、“参照使用”としてマークされる1つのフレーム、もしくは1つ以上のフィールドを含む、かつ
  - フレームバッファが、以下の評価基準の両方が満たされるピクチャを含む。
    - ピクチャのHRD出力時刻が、与えられた時刻点よりも大きい、かつ
    - ピクチャに対する符号器の選好が、表示処理に対して、切り出された8.7節で規定されるデブロックフィルタ処理の結果よりも、切り出された8.5.13小節で規定されるデブロックフィルタ処理前のピクチャ構築処理の結果を表示することであることがデブロックフィルタ表示選好SEIメッセージで指示されている。
- それ以外、フレームバッファは、蓄積のためのDPB容量として1つのフレームバッファを使用するとみなされる。

dec\_frame\_buffering\_constraint\_flagが1に等しい場合、符号化ビデオシーケンスに対するHRDの動作中は、本方法で導出される、ピクチャを含むDPBの全てのフレームバッファによって使用されるフレームバッファリング容量はMax(1,max\_dec\_frame\_buffering)よりも大きくてはならない。

dec\_frame\_buffering\_constraint\_flagの値は、符号化ビデオシーケンスの全てのデブロックフィルタ表示選好SEIメッセージにおいて同じでなければならない。

deblocking\_display\_preference\_repetition\_periodは、デブロックフィルタ表示選好SEIメッセージの持続性を規定し、さらにもう1つのデブロックフィルタ表示選好SEIメッセージ、または符号化ビデオシーケンスの終了がビットストリーム中に存在しなければならないピクチャ順序カウント間隔を規定してもよい。deblocking\_display\_preference\_repetition\_periodの値は0から16384までの、それぞれを含む範囲でなければならない。

deblocking\_display\_preference\_repetition\_periodが0であることは、デブロックフィルタ表示選好SEIメッセージが現復号ピクチャのみに適用されることを規定する。

deblocking\_display\_preference\_repetition\_periodが1であることは、以下の条件のいずれかが真となるまで、出力順序でデブロックフィルタ表示選好SEIメッセージが持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる
- デブロックフィルタ表示選好SEIメッセージを含むアクセスユニット中のピクチャが、PicOrderCnt(CurrPic)よりも大きいPicOrderCnt()を持って出力される。

deblocking\_display\_preference\_repetition\_periodが1よりも大きいことは、以下の条件のいずれかが真となるまで、デブロックフィルタ表示選好SEIメッセージが持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる
- デブロックフィルタ表示選好SEIメッセージを含むアクセスユニット中のピクチャが、PicOrderCnt(CurrPic)よりも大きく、かつPicOrderCnt(CurrPic)+deblocking\_display\_preference\_repetition\_periodよりも小さいかもしくは等しいPicOrderCnt()を持って出力される。

deblocking\_display\_preference\_repetition\_period が 1 よりも大きいことは、もう 1 つのデブロックフィルタ表示 選 好 SEI メ ッ セ ー ジ が 、 PicOrderCnt(CurrPic) より も 大 き く 、 かつ PicOrderCnt(CurrPic)+deblocking\_display\_preference\_repetition\_period より も 小 さ い か も し く は 等 し い PicOrderCnt( ) を 持 っ て 出 力 さ れ る あ る ア ク セ ス ユ ニ ッ ト 中 の あ る ピ ク チ ャ に お い て 、 そ の よ う な ピ ク チ ャ の 出 力 な し に ビ ッ ト ス ト リ ー ム が 終 了 す る か 新 し い 符 号 化 ビ デ オ シ ー ケ ン ス が 開 始 し な い 限 り 、 存 在 し な け れ ば な ら な い こ と を 示 す。

#### D.2.22 ステレオビデオ情報SEIメッセージの意味

この SEI メッセージは、符号化ビデオシーケンス全体がステレオ視点内容を形成するピクチャのペアからなることの指示を復号器へ供給する。

符号化ビデオシーケンスの最初のアクセスユニットにステレオビデオ情報 SEI メッセージが存在しない限り、符号化ビデオシーケンスのいかなるアクセスユニットにおいてもステレオビデオ情報 SEI メッセージは存在してはならない。

**field\_views\_flag** が 1 に等しいことは、現符号化ビデオシーケンスの全てのピクチャがフィールドであり、かつ、ステレオ視点内容に対して、特定パリティの全てのフィールドは左視点であると見なされ、逆パリティの全てのフィールドが右視点であるとみなされることを示す。**field\_views\_flag** が 0 に等しいことは、現符号化ビデオシーケンスの全てのピクチャがフレームであり、かつ出力順序で交互のフレームが、ステレオ視点の視点を表すことを示す。**field\_views\_flag** の値は、符号化ビデオシーケンス内の全てのステレオビデオ情報 SEI メッセージにおいて等しくなければならない。

ステレオビデオ情報 SEI メッセージが存在し、かつ **field\_views\_flag** が 1 に等しい場合、ステレオビデオペアの左視点と右視点は相補的フィールドペアとして符号化されなければならない、かつフィールドペアの出力順序での第 1 フィールドの表示時刻は、フィールドペアの出力順序での第 2 フィールドの表示時刻と一致させるために遅延されるべきであり、かつ表示目的として、各個別フィールド内のサンプルの空間的位置は、図 6-2 で示されるフレーム内の空間的に異なるフィールドとしてよりはむしろ、図 6-1 で示される完全なピクチャの表現として解釈されるべきである。

記 表示処理は本標準で規定されない。

**top\_field\_is\_left\_view\_flag** が 1 に等しいことは、符号化ビデオシーケンスのトップフィールドが左視点を表し、符号化ビデオシーケンスのボトムフィールドが右視点を表すことを示す。**top\_field\_is\_left\_view\_flag** が 0 に等しいことは、符号化ビデオシーケンスのボトムフィールドが左視点を表し、符号化ビデオシーケンスのトップフィールドが右視点を表すことを示す。存在する場合、**top\_field\_is\_left\_view\_flag** の値は、符号化ビデオシーケンス内の全てのステレオビデオ情報 SEI メッセージにおいて同じでなければならない。

**current\_frame\_is\_left\_view\_flag** が 1 に等しいことは、現ピクチャがステレオ視点ペアの左視点であることを示す。**current\_frame\_is\_left\_view\_flag** が 0 に等しいことは、現ピクチャがステレオ視点ペアの右視点であることを示す。

**next\_frame\_is\_second\_view\_flag** が 1 に等しいことは、現ピクチャと出力順序で次のピクチャがステレオ視点ペアを構成し、かつ現ピクチャの表示時刻は出力順序で次のピクチャの表示時刻と一致させるために遅延されるべきであることを示す。**next\_frame\_is\_second\_view\_flag** が 0 に等しいことは、現ピクチャと出力順序で直前のピクチャがステレオ視点ペアを構成する、かつステレオ視点ペアとする目的のために現ピクチャの表示時刻は遅延されるべきでないことを示す。

**left\_view\_self\_contained\_flag** が 1 に等しいことは、符号化ビデオシーケンスの左視点ピクチャに対する復号処理における INTER 予測操作が、右視点ピクチャである参照ピクチャを参照しないことを示す。**left\_view\_self\_contained\_flag** が 0 に等しいことは、符号化ビデオシーケンスの左視点ピクチャに対する復号処理における、いくつかの INTER 予測操作が、右視点ピクチャである参照ピクチャを参照してもよいし、しなくてもよいことを示す。符号化ビデオシーケンス内で、全てのステレオビデオ情報 SEI メッセージ中の **left\_view\_self\_contained\_flag** の値は同じでなければならない。

**right\_view\_self\_contained\_flag** が 1 に等しいことは、符号化ビデオシーケンスの右視点ピクチャに対する復号処理における INTER 予測操作が、左視点ピクチャである参照ピクチャを参照しないことを示す。**right\_view\_self\_contained\_flag** が 0 に等しいことは、符号化ビデオシーケンスの右視点ピクチャに対する復号処理における、いくつかの INTER 予測操作が、左視点ピクチャである参照ピクチャを参照してもよいし、しなくてもよいことを示す。符号化ビデオシーケンス内で、全てのステレオビデオ情報 SEI メッセージ中の **right\_view\_self\_contained\_flag** の値は同じでなければならない。

#### D.2.23 ポストフィルタヒントSEIメッセージの意味

この SEI メッセージは、改善された表示品質を得るための出力復号ピクチャの事後処理における潜在的な使用のために、ポストフィルタの係数またはポストフィルタの設計のための関連情報を提供する。

**filter\_hint\_size\_y** は、フィルタ係数または関連配列の垂直サイズを規定する。**filter\_hint\_size\_y** の値は 1 から 15 のそれぞれを含む範囲内にななければならない。

**filter\_hint\_size\_x** は、フィルタ係数または関連配列の水平サイズを規定する。**filter\_hint\_size\_x** の値は 1 から 15 のそれぞれを含む範囲内にななければならない。

**filter\_hint\_type** は、送出されるフィルタヒントの型を、付表 D-7/JT-H264 に規定されるように識別する。**filter\_hint\_type** の値は 0 から 2 のそれぞれを含む範囲内にななければならない。本標準に適合する復号器は、予約値 3 に等しい **filter\_hint\_type** を持つポストフィルタヒント情報 SEI メッセージを無視しなければならない。

付表D-7/JT-H264 **filter\_hint\_type**の値  
(ITU-T H.264)

値	説明
0	2D FIR フィルタの係数
1	2つの 1D FIR フィルタの係数
2	相互相関行列
3	予約

**filter\_hint[colour\_component][cy][cx]**は、フィルタ係数または原信号と復号信号の間の相互相関行列の要素を 16 ビット精度で規定する。**colour\_component** は関連する色成分を規定する。**cy** は垂直方向のカウンタを表し、**cx** は水平方向のカウンタを表す。**filter\_hint\_type** に依存して、次が適用される。

- もし `filter_hint_type` が 0 に等しいならば、`filter_hint_size_y*filter_hint_size_x` のサイズの 2 次元 FIR フィルタの係数が送出される。
- それ以外、もし `filter_hint_type` が 1 に等しいならば、2 つの 1 次元 FIR フィルタの係数が送出される。この場合、`filter_hint_size_y` は 2 に等しくなければならない。インデックス `cy=0` は水平フィルタのフィルタ係数を規定し、`cy=1` は垂直フィルタのフィルタ係数を規定する。フィルタ処理において、水平フィルタが先に適用されなければならない、その結果が垂直フィルタによりフィルタされなければならない。
- それ以外 (`filter_hint_type` が 2 に等しい)、送出されるヒントは原信号  $s$  と復号信号  $s'$  の間の相互相関行列を規定する。

記1 `filter_hint_size_y*filter_hint_size_x` のサイズの関連する色成分の相互相関行列は次のように定義される。

$$\text{filter\_hint}(cy,cx) = \frac{1}{h \cdot w} \sum_{m=0}^{h-1} \sum_{n=0}^{w-1} s(m,n) \cdot s'(m+cy-\text{offset}_y, n+cx-\text{offset}_x) \quad (\text{D-26})$$

但し、 $s$  は原信号を示し、 $s'$  は復号信号を示し、 $h$  は関連する色成分の垂直の高さを示し、 $w$  は関連する色成分の水平の幅を示し、`offset_y` は (`filter_hint_size_y >> 1`) に等しく、`offset_x` は (`filter_hint_size_x >> 1`) に等しく、 $0 \leq cy < \text{filter\_hint\_size\_y}$  および  $0 \leq cx < \text{filter\_hint\_size\_x}$  である。

記2 復号器は、原信号と復号信号の相互相関行列と復号信号の自己相関行列から Wiener ポストフィルタを導出できる。

`additional_extension_flag` が 0 に等しいことは、ポストフィルタヒント情報 SEI メッセージ内に追加データが続かないことを識別する。`additional_extension_flag` の値は 0 に等しくなければならない。`additional_extension_flag` の 1 の値は将来の使用のために TTC によって予約される。本標準に適合する復号器は、`additional_extension_flag` の 1 の値に続く全てのデータを無視しなければならない。

#### D.2.24 トーンマッピング情報 SEI メッセージの意味

この SEI メッセージは、特定の表示環境へのカスタマイズのために、出力復号ピクチャの色サンプルの再マッピングを可能にするための情報を提供する。この再マッピング処理は、(付属資料 E に規定される) RGB 色空間における符号化サンプル値を目的のサンプル値にマッピングする。全てのマッピングは、RGB 色空間において表現され、各 RGB 成分に分離して適用されるべきである。

`tone_map_id` は、トーンマッピングモデルの目的を識別するために使用されて良い識別番号を含む。`tone_map_id` の値は、0 から  $2^{32}-1$  のそれぞれの値を含む範囲になければならない。

0 から 255 および 512 から  $2^{31}-1$  の `tone_map_id` の値は、アプリケーションにより決定されるように使用されて良い。256 から 511 および  $2^{31}$  から  $2^{32}-1$  の `tone_map_id` の値は、将来の使用のために TTC によって予約される。復号器は、256 から 511 または  $2^{31}$  から  $2^{32}-1$  の範囲内にある `tone_map_id` の値を含む全てのトーンマッピング SEI メッセージを無視 (ビットストリームから除去して破棄) しなければならない、ビットストリームはこれらの値を含んではならない。

記1 `tone_map_id` は、異なる表示シナリオに適したトーンマッピング操作をサポートするために使用することができる。例えば、異なる値の `tone_map_id` は、異なる表示ビット深度に対応するかもしれない。

`tone_map_cancel_flag` が 1 に等しいことは、そのトーンマッピング情報 SEI メッセージが、出力順序で先行する任意のトーンマッピング情報 SEI メッセージの持続を取消することを示す。`tone_map_cancel_flag` が 0 であることは、トーンマッピング情報が次に続くことを示す。

**tone\_map\_repetition\_period** は、トーンマッピング情報 SEI メッセージの持続性を規定し、さらに同じ **tone\_map\_id** の値を持つもう 1 つのトーンマッピング情報 SEI メッセージ、または符号化ビデオシーケンスの終了がビットストリーム中に存在しなければならないピクチャ順序カウンタ間隔を規定してもよい。**tone\_map\_id** の値は 0 から 16384 までの、それぞれを含む範囲でなければならない。

**tone\_map\_repetition\_period** が 0 に等しいことは、トーンマッピング情報が現復号ピクチャのみに適用されることを規定する。

**tone\_map\_repetition\_period** が 1 に等しいことは、以下の条件のいずれかが真となるまで、出力順序でトーンマッピング情報が持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる。
- 同じ **tone\_map\_id** の値を持つトーンマッピング情報 SEI メッセージを含むアクセスユニット中のピクチャが、**PicOrderCnt(CurrPic)**よりも大きい **PicOrderCnt()**を持って出力される。

**tone\_map\_repetition\_period** が 0 に等しい、または 1 に等しいことは、同じ **tone\_map\_id** の値を持つもう 1 つのトーンマッピング情報 SEI メッセージが存在してもよく、または存在しなくてもよいことを示す。

**tone\_map\_repetition\_period** が 1 よりも大きいことは、以下の条件のいずれかが真となるまで、トーンマッピング情報が持続することを規定する。

- 新しい符号化ビデオシーケンスが始まる。
- 同じ **tone\_map\_id** の値を持つトーンマッピング情報 SEI メッセージを含むアクセスユニット中のピクチャが、**PicOrderCnt(CurrPic)**よりも大きく、かつ **PicOrderCnt(CurrPic)+tone\_map\_repetition\_period** よりも小さいかもしくは等しい **PicOrderCnt()**を持って出力される。

**tone\_map\_repetition\_period** が 1 よりも大きいことは、同じ **tone\_map\_id** の値を持つもう 1 つのトーンマッピング情報 SEI メッセージが、**PicOrderCnt(CurrPic)** よりも大きく、かつ **PicOrderCnt(CurrPic)+tone\_map\_repetition\_period** よりも小さいかもしくは等しい **PicOrderCnt()**を持って出力されるあるアクセスユニット中のあるピクチャにおいて、そのようなピクチャの出力なしにビットストリームが終了するか新しい符号化ビデオシーケンスが開始しない限り、存在しなければならないことを示す。

**coded\_data\_bit\_depth** は、符号化ビデオシーケンスの輝度成分の **BitDepth<sub>Y</sub>** を規定する。これは、符号化ビデオシーケンスとともに使用されることが意図されているトーンマッピング情報 SEI メッセージを識別するために使用される。もし、**BitDepth<sub>Y</sub>** と等しくない **coded\_data\_bit\_depth** を持つトーンマッピング情報 SEI メッセージが存在するならば、それらは **coded\_data\_bit\_depth** の値に対応する **BitDepth<sub>Y</sub>** に符号化ビデオを変換するために実行されるトランス符号化操作の仮想的な結果を参照する。

**coded\_data\_bit\_depth** の値は 8 から 14 までの、それぞれを含む範囲でなければならない。0 から 7 および 15 から 255 の範囲内の **coded\_data\_bit\_depth** 値は、将来の使用のために TTC によって予約される。復号器は、0 から 7 および 15 から 255 の値の **coded\_data\_bit\_depth** の値を含む全てのトーンマッピング SEI メッセージを無視(ビットストリームから除去して破棄)しなければならない、ビットストリームはこれらの値を含んではならない。

**target\_bit\_depth** は、トーンマッピング情報 SEI メッセージにより記述されるダイナミックレンジマッピング関数 (あるいはトーンマッピング関数) の出力のビット深度を規定する。特定の **target\_bit\_depth** で規定されるトーンマッピング関数は、その **target\_bit\_depth** に等しいか小さい全ての表示ビット深度に対して妥当

であることが提案されている。

**target\_bit\_depth** の値は 1 から 16 までの、それぞれを含む範囲でなければならない。0 に等しい、および 17 から 255 の範囲内の **target\_bit\_depth** 値は、将来の使用のために TTC によって予約される。復号器は、0 に等しい、および 17 から 255 の範囲内にある **target\_bit\_depth** の値を含む全てのトーンマッピング SEI メッセージを無視（ビットストリームから除去して破棄）しなければならない。ビットストリームはこれらの値を含んではならない。

**model\_id** は、符号化データを **target\_bit\_depth** の範囲にマッピングするために利用されるモデルを規定する。3 より大きい値は、将来の使用のために TTC によって予約される。復号器は、3 より大きい **model\_id** の値を含む全てのトーンマッピング SEI メッセージを無視（ビットストリームから除去して破棄）しなければならない。ビットストリームはこれらの値を含んではならない。

記2 0の**model\_id**はクリッピングつき線形マッピングに対応し、1の**model\_id**はシグモイドマッピングに対応し、2の**model\_id**はユーザ定義テーブルマッピングに対応し、そして3の**model\_id**は区分的線形マッピングに対応する。

**min\_value** は、伝えられた **target\_bit\_depth** 内の最小値にマッピングする、符号化データの RGB サンプル値を規定する。**max\_value** パラメータとの組み合わせで用いられる。符号化データ内の **min\_value** より小さいか等しい全ての値は、**target\_bit\_depth** 表現内のこの最小値にマッピングされる。

**max\_value** は、伝えられた **target\_bit\_depth** 内の最大値にマッピングする、符号化データの RGB サンプル値を規定する。**min\_value** パラメータとの組み合わせで用いられる。符号化データ内の **max\_value** より大きいか等しい全ての値は、**target\_bit\_depth** 表現内のこの最大値にマッピングされる。

**sigmoid\_midpoint** は、**target\_bit\_depth** 表現の中心点にマッピングされる、符号化データの RGB サンプル値を規定する。**sigmoid\_width** パラメータとの組み合わせで用いられる。

**sigmoid\_width** は、**target\_bit\_depth** 表現の値のそれぞれ約 5%および 95%に対応する二つの符号化データ値の間の距離を規定する。**sigmoid\_midpoint** パラメータとの組み合わせで用いられ、次の関数に従って解釈される：

$$f(i) = \text{Round} \left( \frac{2^{\text{target\_bit\_depth}} - 1}{1 + \exp \left( \frac{-6 * (i - \text{sigmoid\_midpoint})}{\text{sigmoid\_width}} \right)} \right), i = 0, \dots, (2^{\text{coded\_bit\_depth}} - 1) \text{ に対して}$$

(D-27)

ここで、 $f(i)$  は、符号化データからの RGB サンプル値  $i$  を、結果となる **target\_bit\_depth** 表現内の RGB サンプル値にマッピングする関数を示す。

**start\_of\_coded\_interval[i]** は、**start\_of\_coded\_interval[i]** より大きいか等しく **start\_of\_coded\_interval[i+1]** より小さい全ての RGB サンプル値がターゲットビット深度表現内の  $i$  にマッピングされるような、符号化データの区間の開始点を規定する。**start\_of\_coded\_interval[2<sup>target\_bit\_depth</sup>]** の値は  $2^{\text{coded\_bit\_depth}}$  に等しい。**start\_of\_coded\_interval** の表現に用いられるビット数は  $((\text{coded\_data\_bit\_depth} + 7) \gg 3) \ll 3$  である。

**num\_pivots** は、2つの既定端点(0,0)および  $(2^{\text{coded\_data\_bit\_depth}-1}, 2^{\text{target\_bit\_depth}-1})$  はカウントせずに、区分的線形マッピング関数のピボット点の数を規定する。

**coded\_pivot\_value[i]**は、i 番目のピボット点に対応する **coded\_bit\_depth** の値を規定する。**coded\_pivot\_value** の表現に用いられるビット数は $((\text{coded\_data\_bit\_depth}+7)\gg 3)\ll 3$  である。

**target\_pivot\_value[i]**は、i 番目のピボット点に対応する参照 **target\_bit\_depth** の値を規定する。**target\_pivot\_value** の表現に用いられるビット数は $((\text{target\_bit\_depth}+7)\gg 3)\ll 3$  である。

#### D.2.25 予約SEIメッセージの意味

このメッセージは、将来の後方互換使用のために TTC によって予約されたデータから構成される。本標準に適合する符号器は、TTC によってこのようなメッセージの使用が規定されるまで、かつ規定されない限りは予約 SEI メッセージを送ってはならない。予約 SEI メッセージに遭遇した本標準に適合する復号器は、TTC によって規定される将来の標準で規定される場合を除いて、復号処理で影響無しにそれらの内容を破棄しなくてはならない。

**reserved\_sei\_message\_payload\_byte** は、将来使用のために TTC によって予約された 1 バイトである。

### 付属資料E

#### ビデオ有用性情報

(この付属資料は、本標準の必須部分を形成する)

この付属資料は、シーケンスパラメータセットの VUI パラメータのシンタックスと意味を規定する。

VUI パラメータは、復号処理で輝度または色差サンプルを構築するために必要とされない。適合復号器は、本標準に適合する出力順序に対して、この情報を処理することは必要とされない（適合の規定に関しては付属資料 C を参照）。VUI パラメータの中にはビットストリーム適合をチェックするためと出力タイミング復号器適合のために必要とされるものもある。

付属資料 E では、VUI パラメータの存在の規定は、それらのパラメータ（あるいはそれらのいくつかのサブセット）が、本標準で規定されない別の手段で復号器（あるいは HRD）に伝達される時にも満たされる。ビットストリーム中に存在する時、VUI パラメータは 7.3.2.1 小小節と 7.4.2.1 小小節とこの付属資料で規定されるシンタックスと意味に従わなければならない。VUI パラメータの内容が、ビットストリーム内で存在するのは別に何らかの手段でアプリケーションに対して伝達される時、VUI パラメータの内容の表現は、この付属資料で規定されるのと同じシンタックスを用いることは必要とされない。ビットを数える目的では、ビットストリーム中に実際に存在する適切なビットだけが数えられる。

## E.1 VUIシンタックス

### E.1.1 VUIパラメータシンタックス

	C	記述子
vui_parameters() {		
<b>aspect_ratio_info_present_flag</b>	0	u(1)
if( aspect_ratio_info_present_flag ) {		
<b>aspect_ratio_idc</b>	0	u(8)
if( aspect_ratio_idc == Extended_SAR ) {		
<b>sar_width</b>	0	u(16)
<b>sar_height</b>	0	u(16)
}		
}		
<b>overscan_info_present_flag</b>	0	u(1)
if( overscan_info_present_flag )		
<b>overscan_appropriate_flag</b>	0	u(1)
<b>video_signal_type_present_flag</b>	0	u(1)
if( video_signal_type_present_flag ) {		
<b>video_format</b>	0	u(3)
<b>video_full_range_flag</b>	0	u(1)
<b>colour_description_present_flag</b>	0	u(1)
if( colour_description_present_flag ) {		
<b>colour_primaries</b>	0	u(8)
<b>transfer_characteristics</b>	0	u(8)
<b>matrix_coefficients</b>	0	u(8)
}		
}		
<b>chroma_loc_info_present_flag</b>	0	u(1)
if( chroma_loc_info_present_flag ) {		
<b>chroma_sample_loc_type_top_field</b>	0	ue(v)
<b>chroma_sample_loc_type_bottom_field</b>	0	ue(v)
}		
<b>timing_info_present_flag</b>	0	u(1)
if( timing_info_present_flag ) {		
<b>num_units_in_tick</b>	0	u(32)
<b>time_scale</b>	0	u(32)
<b>fixed_frame_rate_flag</b>	0	u(1)
}		
<b>nal_hrd_parameters_present_flag</b>	0	u(1)
if( nal_hrd_parameters_present_flag )		
hrd_parameters()		

<b>vcl_hrd_parameters_present_flag</b>	0	u(1)
if( vcl_hrd_parameters_present_flag )		
hrd_parameters( )		
if( nal_hrd_parameters_present_flag    vcl_hrd_parameters_present_flag )		
<b>low_delay_hrd_flag</b>	0	u(1)
<b>pic_struct_present_flag</b>	0	u(1)
<b>bitstream_restriction_flag</b>	0	u(1)
if( bitstream_restriction_flag ) {		
<b>motion_vectors_over_pic_boundaries_flag</b>	0	u(1)
<b>max_bytes_per_pic_denom</b>	0	ue(v)
<b>max_bits_per_mb_denom</b>	0	ue(v)
<b>log2_max_mv_length_horizontal</b>	0	ue(v)
<b>log2_max_mv_length_vertical</b>	0	ue(v)
<b>num_reorder_frames</b>	0	ue(v)
<b>max_dec_frame_buffering</b>	0	ue(v)
}		
}		

#### E.1.2 HRDパラメータシンタックス

hrd_parameters( ) {	C	記述子
<b>cpb_cnt_minus1</b>	0	ue(v)
<b>bit_rate_scale</b>	0	u(4)
<b>cpb_size_scale</b>	0	u(4)
for( SchedSelIdx = 0; SchedSelIdx <= cpb_cnt_minus1; SchedSelIdx++ ) {		
<b>bit_rate_value_minus1</b> [ SchedSelIdx ]	0	ue(v)
<b>cpb_size_value_minus1</b> [ SchedSelIdx ]	0	ue(v)
<b>cbr_flag</b> [ SchedSelIdx ]	0	u(1)
}		
<b>initial_cpb_removal_delay_length_minus1</b>	0	u(5)
<b>cpb_removal_delay_length_minus1</b>	0	u(5)
<b>dpb_output_delay_length_minus1</b>	0	u(5)
<b>time_offset_length</b>	0	u(5)
}		

## E.2 VUIの意味

### E.2.1 VUIパラメータの意味

**aspect\_ratio\_info\_present\_flag** が 1 に等しいことは、**aspect\_ratio\_idc** が存在することを規定する。  
**aspect\_ratio\_info\_present\_flag** が 0 に等しいことは、**aspect\_ratio\_idc** が存在しないことを規定する。

**aspect\_ratio\_idc** は、輝度サンプルのサンプルアスペクト比の値を規定する。付表 E-1/JT-H264 は、その符号の意味を示す。**aspect\_ratio\_idc** が Extended\_SAR を示す時、サンプルアスペクト比は **sar\_width** と **sar\_height** で表される。**aspect\_ratio\_idc** シンタックス要素が存在しない時、**aspect\_ratio\_idc** 値は 0 に等しいと推定されなければならない。

付表E-1/JT-H264 サンプルアスペクト比指示子の意味  
(ITU-T H.264)

<b>aspect_ratio_idc</b>	サンプルアスペクト比	(参考) 使用例
0	無規定	
1	1:1 (“正方形”)	1280×720 16:9 水平方向オーバースキャンなしのフレーム 1920×1080 16:9 水平方向オーバースキャンなしのフレーム (1920×1088 から切り出された) 640×480 4:3 水平方向オーバースキャンなしのフレーム
2	12:11	720×576 4:3 水平方向オーバースキャンありのフレーム 352×288 4:3 水平方向オーバースキャンなしのフレーム
3	10:11	720×480 4:3 水平方向オーバースキャンありのフレーム 352×240 4:3 水平方向オーバースキャンなしのフレーム
4	16:11	720×576 16:9 水平方向オーバースキャンありのフレーム 528×576 4:3 水平方向オーバースキャンなしのフレーム
5	40:33	720×480 16:9 水平方向オーバースキャンありのフレーム 528×480 4:3 水平方向オーバースキャンなしのフレーム
6	24:11	352×576 4:3 水平方向オーバースキャンなしのフレーム 480×576 16:9 水平方向オーバースキャンありのフレーム
7	20:11	352×480 4:3 水平方向オーバースキャンなしのフレーム 480×480 16:9 水平方向オーバースキャンありのフレーム
8	32:11	352×576 16:9 水平方向オーバースキャンなしのフレーム
9	80:33	352×480 16:9 水平方向オーバースキャンなしのフレーム
10	18:11	480×576 4:3 水平方向オーバースキャンありのフレーム
11	15:11	480×480 4:3 水平方向オーバースキャンありのフレーム
12	64:33	528×576 16:9 水平方向オーバースキャンなしのフレーム
13	160:99	528×480 16:9 水平方向オーバースキャンなしのフレーム
14	4:3	1440×1080 16:9 水平方向オーバースキャンなしのフレーム
15	3:2	1280×1080 16:9 水平方向オーバースキャンなしのフレーム
16	2:1	960×1080 16:9 水平方向オーバースキャンなしのフレーム
17..254	予約	
255	Extended_SAR	

**sar\_width** は、(任意の単位で) サンプルアスペクト比の水平方向のサイズを示す。

**sar\_height** は、(sar\_width と同じ任意の単位で) サンプルアスペクト比の垂直方向のサイズを示す。

sar\_width と sar\_height は互いに素あるいは 0 に等しくなければならない。aspect\_ratio\_idc が 0 に等しいかあるいは sar\_width が 0 に等しいかあるいは sar\_height が 0 に等しい時、サンプルアスペクト比は本標準で無規定とみなされなければならない。

**overscan\_info\_present\_flag** が 1 に等しいことは **overscan\_appropriate\_flag** が存在することを規定する。overscan\_info\_present\_flag が 0 に等しいかあるいは存在しない時、ビデオ信号に対する好ましい表示方法が無規定である。

**overscan\_appropriate\_flag** が 1 に等しいことは、切り出された復号ピクチャ出力が、オーバースキャンを用いて表示するのに適することを示す。overscan\_appropriate\_flag が 0 に等しいことは、切り出された復号ピクチャ出力が、それがオーバースキャンを用いて表示されるべきではないような、ピクチャ切り出し矩形エッジ外の全体領域における視覚的に重要な情報を含むことを示す。その代わりに、それらは表示領域と切り出し矩形の完全一致、またはアンダースキャンを用いて表示されるべきである。

記1 例えば、overscan\_appropriate\_flagが1に等しいのは、娯楽テレビ番組あるいはビデオ会議における人々のライブビューに対して用いられるかもしれない、そしてoverscan\_appropriate\_flagが0に等しいのは、コンピュータ画面の取り込みあるいはセキュリティカメラの内容に対して用いられるかもしれない。

**video\_signal\_type\_present\_flag** が 1 に等しいことは、video\_format、video\_full\_range\_flag、および colour\_description\_present\_flag が存在することを規定する。video\_signal\_type\_present\_flag が 0 に等しいことは、video\_format、video\_full\_range\_flag、および colour\_description\_present\_flag が存在しないことを規定する。

**video\_format** は、本標準に従い符号化される前の、付表 E-2/JT-H264 で規定されるようなピクチャの表現を示す。video\_format シンタックス要素が存在しない時、video\_format 値は 5 に等しいと推定されなければならない。

付表E-2/JT-H264 video\_formatの意味  
(ITU-T H.264)

video_format	意味
0	コンポーネント
1	PAL
2	NTSC
3	SECAM
4	MAC
5	無規定ビデオフォーマット
6	予約
7	予約

**video\_full\_range\_flag** は、アナログコンポーネント信号 E'\_Y、E'\_PB および E'\_PR、または E'\_R、E'\_G および E'\_B から導出される黒レベルと輝度、色差信号の範囲を示す。

video\_full\_range\_flag シンタックス要素が存在しない時、video\_full\_range\_flag の値は 0 に等しいと推定され

なければならない。

**colour\_description\_present\_flag** が 1 に等しいことは、**colour primaries**、**transfer characteristics**、および **matrix coefficients** が存在することを規定する。**colour\_description\_present\_flag** が 0 に等しいことは、**colour primaries**、**transfer characteristics**、および **matrix coefficients** が存在しないことを規定する。

**colour primaries** は、ISO/CIE 10527 で規定されるような x と y の CIE 1931 定義に基づいた付表 E-3/JT-H264 で規定されるような情報源原色の色度座標を示す。

**colour primaries** シンタックス要素が存在しない時、**colour primaries** の値は 2 に等しいと推定されなければならない（色度は無規定あるいはアプリケーションにより決められる）。

付表E-3/JT-H264 原色  
(ITU-T H.264)

値	原色			参考注釈
0	予約			TTCで将来使用
1	原色	x	y	ITU-R勧告BT.709-5
	緑	0.300	0.600	ITU-R勧告BT.1361 通常色域系および拡張色域系
	青	0.150	0.060	IEC 61966-2-4
	赤	0.640	0.330	SMPTE RP 177 (1993) Annex B
	白D65	0.3127	0.3290	
2	無規定			画像特性が不明あるいはアプリケーションにより決められる
3	予約			TTCで将来使用
4	原色	x	y	ITU-R勧告BT.470-6 System M（歴史的）
	緑	0.21	0.71	米国NTSC 1953 Recommendation for transmission standards for colour television
	青	0.14	0.08	米国FCC Title 47 Code of Federal Regulations (2003) 73.682 (a) (20)
	赤	0.67	0.33	
	白C	0.310	0.316	
5	原色	x	y	ITU-R勧告BT.470-6 System B, G（歴史的）
	緑	0.29	0.60	ITU-R勧告BT.601-6 625
	青	0.15	0.06	ITU-R勧告BT.1358 625
	赤	0.64	0.33	ITU-R勧告BT.1700 625 PALおよび625 SECAM
	白D65	0.3127	0.3290	

6	原色	x	y	ITU-R勧告BT.601-6 525
	緑	0.310	0.595	ITU-R勧告BT.1358 525
	青	0.155	0.070	ITU-R勧告BT.1700 NTSC
	赤	0.630	0.340	SMPTE 170M (2004)
	白D65	0.3127	0.3290	(機能的には値7と同じ)
7	原色	x	y	SMPTE 240M (1999)
	緑	0.310	0.595	(機能的には値6と同じ)
	青	0.155	0.070	
	赤	0.630	0.340	
	白D65	0.3127	0.3290	
8	原色	x	y	一般フィルム(光源Cを用いる色フィルタ)
	緑	0.243	0.692 ( Wratten 58 )	
	青	0.145	0.049 ( Wratten 47 )	
	赤	0.681	0.319 ( Wratten 25 )	
	白C	0.310	0.316	
9-255	予約			TTCで将来使用

**transfer\_characteristics** は、0 から 1 までのアナログ範囲をもつ線形光強度入力  $L_c$  の関数として、付表 E-4 / JT-H264 で規定されるような原ピクチャの光電子伝達特性を示す。

**transfer\_characteristics** シンタックス要素が存在しない時、**transfer\_characteristics** の値は 2 に等しいと推定されなければならない (伝達特性は無規定あるいはアプリケーションにより決められる)。

付表E-4/JT-H264 伝達特性  
(ITU-T H.264)

値	伝達特性	参考注釈
0	予約	TTCで将来使用
1	$V = 1.099 * L_c^{0.45} - 0.099$ $1 \geq L_c \geq 0.018$ に対して $V = 4.500 * L_c$ $0.018 > L_c > 0$ に対して	ITU-R勧告 BT.709-5 ITU-R勧告BT.1361 通常色域系 (機能的には値6と同じ)
2	無規定	画像特性が不明あるいはアプリケーションにより決められる
3	予約	TTCで将来使用
4	表示ガンマは2.2を仮定	ITU-R勧告BT.470-6 System M (歴史的) 米国NTSC 1953 Recommendation for transmission standards for colour television 米国FCC Title 47 Code of Federal Regulations (2003) 73.682 (a) (20) ITU-R勧告BT.1700 (2007版) 625 PALおよび 625 SECAM

5	表示ガンマは2.8を仮定	ITU-R勧告BT.470-6 System B, G (歴史的)
6	$V = 1.099 * L_c^{0.45} - 0.099$ $1 \geq L_c \geq 0.018$ に対して $V = 4.500 * L_c$ $0.018 > L_c \geq 0$ に対して	ITU-R勧告BT.601-6 525または625 ITU-R勧告BT.1358 525または625 ITU-R勧告BT.1700 NTSC SMPTE 170M (2004) (機能的には値1と同じ)
7	$V = 1.1115 * L_c^{0.45} - 0.1115$ $1 \geq L_c \geq 0.0228$ に対して $V = 4.0 * L_c$ $0.0228 > L_c \geq 0$ に対して	SMPTE 240M (1999)
8	$V = L_c$ $1 > L_c \geq 0$ に対して	線形伝達特性
9	$V = 1.0 - \text{Log}10(L_c) \div 2$ $1 \geq L_c \geq 0.01$ に対して $V = 0.0$ $0.01 > L_c \geq 0$ に対して	対数伝達特性(100:1範囲)
10	$V = 1.0 - \text{Log}10(L_c) \div 2.5$ $1 \geq L_c \geq 0.0031622777$ に対して $V = 0.0$ $0.0031622777 > L_c \geq 0$ に対して	対数伝達特性(316.22777:1範囲)
11	$V = 1.099 * L_c^{0.45} - 0.099$ $L_c \geq 0.018$ に対して $V = 4.500 * L_c$ $0.018 > L_c > -0.018$ に対して $V = -1.099 * (-L_c)^{0.45} + 0.099$ $-0.018 \geq L_c$ に対して	IEC 61966-2-4
12	$V = 1.099 * L_c^{0.45} - 0.099$ $1.33 > L_c \geq 0.018$ に対して $V = 4.500 * L_c$ $0.018 > L_c \geq -0.0045$ に対して $V = -(1.099 * (-4 * L_c)^{0.45} - 0.099) \div 4$ $-0.0045 > L_c \geq -0.25$ に対して	ITU-R勧告BT.1361 拡張色域系
13..255	予約	TTCで将来使用

**matrix\_coefficients** は付表 E-5/JT-H264 で規定されるような緑、青、および赤の原色から輝度、色差信号を導出する時に用いられる行列の係数を記述する。

次の両条件が真でない限り **matrix\_coefficients** は 0 に等しくあってはならない。

- $\text{BitDepth}_c$  が  $\text{BitDepth}_y$  に等しい
- $\text{chroma\_format\_idc}$  が 3(4:4:4)に等しい

他の全ての条件における 0 に等しい **matrix\_coefficients** の使用規定は TTC による将来使用のために予約される。

次の1つの条件が真でない限り **matrix\_coefficients** は 8 に等しくあってはならない。

- $\text{BitDepth}_c$  が  $\text{BitDepth}_y$  に等しい

- $\text{BitDepth}_C$  が  $\text{BitDepth}_Y+1$  に等しく、かつ  $\text{chroma\_format\_idc}$  が 3(4:4:4)に等しい

他の全ての条件における 8 に等しい  $\text{matrix\_coefficients}$  の使用規定は TTC による将来使用のために予約される。

$\text{matrix\_coefficients}$  シンタックス要素が存在しない時、 $\text{matrix\_coefficients}$  の値は 2 (無規定) に等しいと推定されなければならない。

$\text{matrix\_coefficients}$  の解釈は、次の通り規定される。

- もし  $\text{transfer\_characteristics}$  が 11 または 12 に等しくなければ、 $E'_R$ 、 $E'_G$  および  $E'_B$  は 0 から 1 までの範囲の値を用いるアナログ値である。
- それ以外 ( $\text{transfer\_characteristics}$  が 11 (IEC 61966-2-4) または 12 (ITU-R 勧告 BT.1361 拡張色域系) に等しい)、 $E'_R$ 、 $E'_G$  および  $E'_B$  は、本標準で規定されない、より広い範囲を用いるアナログ値である。
- 名目上の白は、1 に等しい  $E'_R$ 、1 に等しい  $E'_G$ 、1 に等しい  $E'_B$  をもつと規定される。
- 名目上の黒は、0 に等しい  $E'_R$ 、0 に等しい  $E'_G$ 、0 に等しい  $E'_B$  をもつと規定される。
- もし  $\text{video\_full\_range\_flag}$  が 0 に等しいならば、次の等式が適用される。

- もし  $\text{matrix\_coefficients}$  が 1、4、5、6、または 7 に等しいならば、次の等式が適用される

$$Y = \text{Clip1}_Y(\text{Round}((1 \ll (\text{BitDepth}_Y - 8)) * (219 * E'_Y + 16))) \quad (\text{E-1})$$

$$Cb = \text{Clip1}_C(\text{Round}((1 \ll (\text{BitDepth}_C - 8)) * (224 * E'_{PB} + 128))) \quad (\text{E-2})$$

$$Cr = \text{Clip1}_C(\text{Round}((1 \ll (\text{BitDepth}_C - 8)) * (224 * E'_{PR} + 128))) \quad (\text{E-3})$$

- それ以外、もし  $\text{matrix\_coefficients}$  が 0 または 8 に等しいならば、次の等式が適用される。

$$R = \text{Clip1}_Y((1 \ll (\text{BitDepth}_Y - 8)) * (219 * E'_R + 16)) \quad (\text{E-4})$$

$$G = \text{Clip1}_Y((1 \ll (\text{BitDepth}_Y - 8)) * (219 * E'_G + 16)) \quad (\text{E-5})$$

$$B = \text{Clip1}_Y((1 \ll (\text{BitDepth}_Y - 8)) * (219 * E'_B + 16)) \quad (\text{E-6})$$

- それ以外、もし  $\text{matrix\_coefficients}$  が 2 に等しいならば、 $\text{matrix\_coefficients}$  シンタックス要素の解釈は不明あるいはアプリケーションにより決められる。

- それ以外 ( $\text{matrix\_coefficients}$  が 0、1、2、4、5、6、7 または 8 に等しくない)、 $\text{matrix\_coefficients}$  シンタックス要素の解釈は TTC による将来定義のために予約される。

- それ以外 ( $\text{video\_full\_range\_flag}$  が 1 に等しい)、次の等式が適用される。

- もし  $\text{matrix\_coefficients}$  が 1、4、5、6、または 7 に等しいならば、次の等式が適用される

$$Y = \text{Clip1}_Y(\text{Round}(((1 \ll \text{BitDepth}_Y) - 1) * E'_Y)) \quad (\text{E-7})$$

$$Cb = \text{Clip1}_C(\text{Round}(((1 \ll \text{BitDepth}_C) - 1) * E'_{PB} + (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-8})$$

$$Cr = \text{Clip1}_C(\text{Round}(((1 \ll \text{BitDepth}_C) - 1) * E'_{PR} + (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-9})$$

- それ以外、もし `matrix_coefficients` が 0 または 8 に等しいならば、次の等式が適用される。

$$R = \text{Clip1}_Y((1 \ll \text{BitDepth}_Y) - 1) * E'_R \quad (\text{E-10})$$

$$G = \text{Clip1}_Y((1 \ll \text{BitDepth}_Y) - 1) * E'_G \quad (\text{E-11})$$

$$B = \text{Clip1}_Y((1 \ll \text{BitDepth}_Y) - 1) * E'_B \quad (\text{E-12})$$

- それ以外、もし `matrix_coefficients` が 2 に等しいならば、`matrix_coefficients` シンタックス要素の解釈は不明あるいはアプリケーションにより決められる。

- それ以外(`matrix_coefficients` が 0、1、2、4、5、6、7 または 8 に等しくない)、`matrix_coefficients` シンタックス要素の解釈は TTC による将来定義のために予約される。

- もし `matrix_coefficients` が 0 または 8 に等しくないならば、次の等式が適用される。

$$E'_Y = K_R * E'_R + (1 - K_R - K_B) * E'_G + K_B * E'_B \quad (\text{E-13})$$

$$E'_{PB} = 0.5 * (E'_B - E'_Y) \div (1 - K_B) \quad (\text{E-14})$$

$$E'_{PR} = 0.5 * (E'_R - E'_Y) \div (1 - K_R) \quad (\text{E-15})$$

記2  $E'_Y$  は名目上の黒に関連付けられる値0と名目上の白に関連付けられる値1を用いるアナログ値である。 $E'_{PB}$ と $E'_{PR}$ は名目上の黒と名目上の白の両方に関連付けられる値0を用いるアナログ値である。`transfer_characteristics`が11または12に等しくない時、 $E'_Y$ は0から1までの範囲の値を用いるアナログ値である。`transfer_characteristics`が11または12に等しくない時、 $E'_{PB}$ と $E'_{PR}$ は-0.5から0.5までの範囲の値を用いるアナログ値である。`transfer_characteristics`が11 (IEC 61966-2-4) または12 (ITU-R勧告BT.1361 拡張色域系) に等しい時、 $E'_Y$ 、 $E'_{PB}$ および $E'_{PR}$ は、本標準で規定されない、より広い範囲を用いるアナログ値である。

- それ以外、`matrix_coefficients` が 0 に等しいならば、次の等式が適用される。

$$Y = \text{Round}(G) \quad (\text{E-16})$$

$$Cb = \text{Round}(B) \quad (\text{E-17})$$

$$Cr = \text{Round}(R) \quad (\text{E-18})$$

- それ以外(`matrix_coefficients` が 8 に等しい)、次が適用される。

- もし `BitDepthC` が `BitDepthY` に等しいならば、次の等式が適用される。

$$Y = \text{Round}(0.5 * G + 0.25 * (R + B)) \quad (\text{E-19})$$

$$Cb = \text{Round}(0.5 * G - 0.25 * (R + B) + (1 \ll (\text{BitDepth}_C - 1))) \quad (\text{E-20})$$

$$Cr = \text{Round}(0.5 * (R - B) + (1 \ll (\text{BitDepth}_C - 1))) \quad (\text{E-21})$$

記3 付表E-5/JT-H264で使用される命名法YCGCoのために、等式E-20とE-21のCbおよびCrはそれぞれCgおよびCoと呼ばれるかもしれない。上記3つの等式に対する逆変換は、以下で計算されるべきである。

$$t = Y - (Cb - (1 \ll (\text{BitDepth}_C - 1))) \quad (\text{E-22})$$

$$G = \text{Clip1}_Y(Y + (Cb - (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-23})$$

$$B = \text{Clip1}_Y(t - (Cr - (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-24})$$

$$R = \text{Clip1}_Y(t + (Cr - (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-25})$$

- それ以外( $\text{BitDepth}_C$ が $\text{BitDepth}_Y$ に等しくない)、次の等式が適用される。

$$Cr = \text{Round}(R) - \text{Round}(B) + (1 \ll (\text{BitDepth}_C - 1)) \quad (\text{E-26})$$

$$t = \text{Round}(B) + ((Cr - (1 \ll (\text{BitDepth}_C - 1))) \gg 1) \quad (\text{E-27})$$

$$Cb = \text{Round}(G) - t + (1 \ll (\text{BitDepth}_C - 1)) \quad (\text{E-28})$$

$$Y = t + ((Cb - (1 \ll (\text{BitDepth}_C - 1))) \gg 1) \quad (\text{E-29})$$

記4 付表E-5/JT-H264で使用される命名法YCgCoのために、等式E-28とE-26のCbおよびCrはそれぞれCgおよびCoと呼ばれるかもしれない。上記4つの等式に対する逆変換は、以下で計算されるべきである。

$$t = Y - (Cb - (1 \ll (\text{BitDepth}_C - 1))) \gg 1 \quad (\text{E-30})$$

$$G = \text{Clip1}_Y(t + (Cb - (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-31})$$

$$B = \text{Clip1}_Y(t - ((Cr - (1 \ll (\text{BitDepth}_C - 1))) \gg 1)) \quad (\text{E-32})$$

$$R = \text{Clip1}_Y(B + (Cr - (1 \ll (\text{BitDepth}_C - 1)))) \quad (\text{E-33})$$

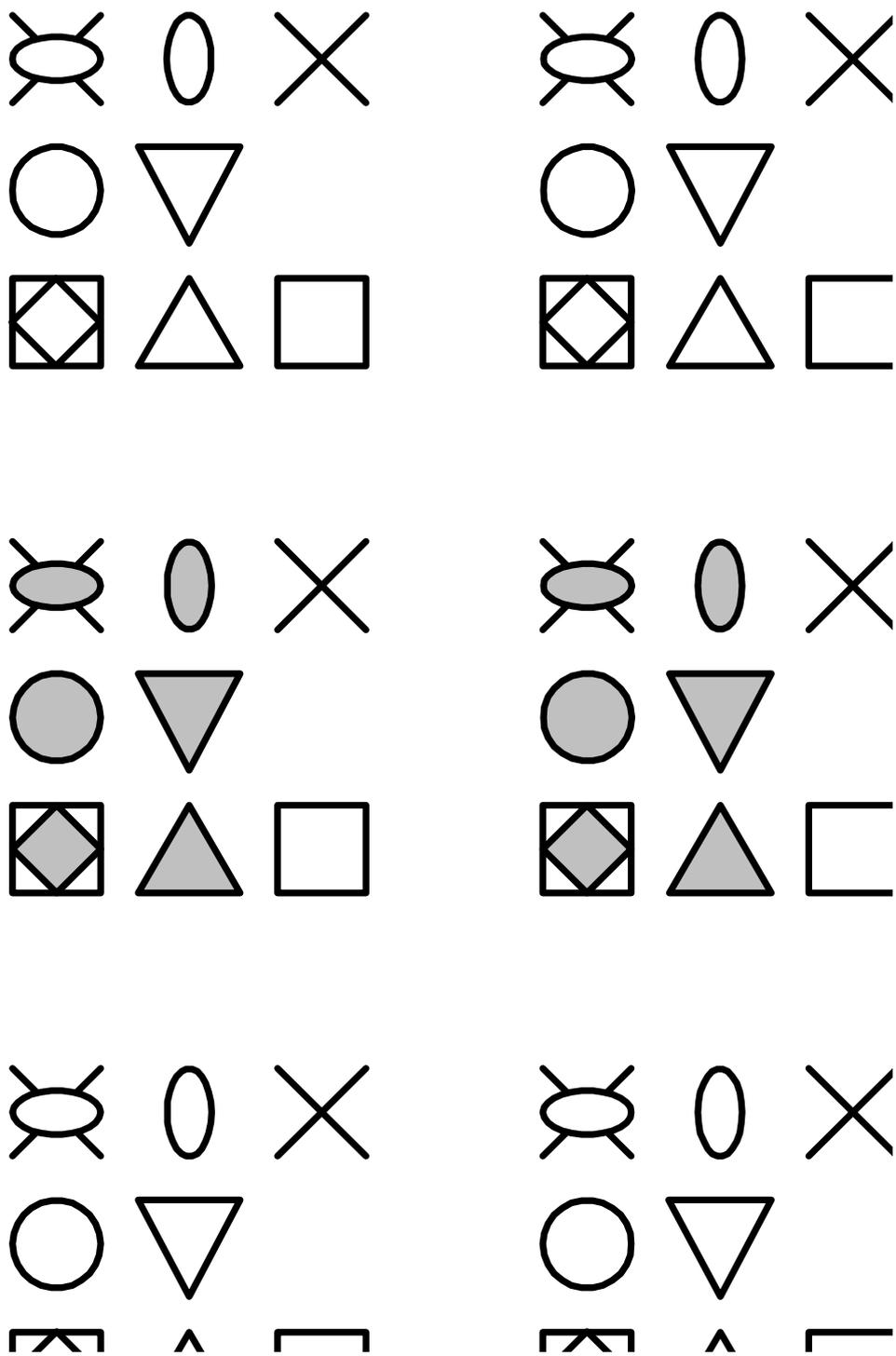
付表E-5/JT-H264 行列の係数  
(ITU-T H.264)

値	行列	参考注釈
0	GBR	典型的にRGBと呼ばれる；等式E-16からE-18を見よ
1	$K_R = 0.2126; K_B = 0.0722$	ITU-R勧告BT.709-5 ITU-R勧告BT.1361 通常色域系および拡張色域系 IEC 61966-2-4 xvYCC <sub>709</sub> SMPTE RP 177 (1993) Annex B
2	無規定	画像特性が不明あるいはアプリケーションにより決められる
3	予約	TTCで将来使用
4	$K_R = 0.30; K_B = 0.11$	US連邦通信委員会Title 47 Code of Federal Regulations (2003) 73.682 (a) (20)
5	$K_R = 0.299; K_B = 0.114$	ITU-R勧告BT.470-6 System B, G (歴史的) ITU-R勧告BT.601-6 625 ITU-R勧告BT.1358 625 ITU-R勧告BT.1700 625 PALおよび625 SECAM IEC 61966-2-4 xvYCC <sub>601</sub> (機能的には値6と同じ)
6	$K_R = 0.299; K_B = 0.114$	ITU-R勧告BT.601-6 525 ITU-R勧告BT.1358 525 ITU-R勧告BT.1700 NTSC SMPTE 170M (2004) (機能的には値5と同じ)
7	$K_R = 0.212; K_B = 0.087$	SMPTE 240M (1999)
8	YCgCo	等式E-19からE-33を見よ
9-255	予約	TTCで将来使用

**chroma\_loc\_info\_present\_flag** が 1 に等しいことは、**chroma\_sample\_loc\_type\_top\_field** と **chroma\_sample\_loc\_type\_bottom\_field** が存在することを規定する。**chroma\_loc\_info\_present\_flag** が 0 に等しいことは、**chroma\_sample\_loc\_type\_top\_field** と **chroma\_sample\_loc\_type\_bottom\_field** が存在しないことを規定する。

**chroma\_sample\_loc\_type\_top\_field**と**chroma\_sample\_loc\_type\_bottom\_field**は付図E-1/JT-H.264で示されるようなトップフィールドとボトムフィールドに対する色差サンプルの位置を規定する。**chroma\_sample\_loc\_type\_top\_field**と**chroma\_sample\_loc\_type\_bottom\_field**の値は0から5のそれぞれの値を含む範囲でなければならない。**chroma\_sample\_loc\_type\_top\_field**と**chroma\_sample\_loc\_type\_bottom\_field**が存在しない時、**chroma\_sample\_loc\_type\_top\_field**と**chroma\_sample\_loc\_type\_bottom\_field**の値は0に等しいと推定されなければならない。

記5 プログレッシブ原素材を符号化する時、**chroma\_sample\_loc\_type\_top\_field**と**chroma\_sample\_loc\_type\_bottom\_field**は同じ値をもつべきである。



付図E-1/JT-H.264 chroma\_sample\_loc\_type\_top\_fieldとchroma\_sample\_loc\_type\_bottom\_fieldの関数としてのトップとボトムフィールドに対する色差サンプルの位置  
(ITU-T H.264)

**timing\_info\_present\_flag** が 1 に等しいことは、**num\_units\_in\_tick**、**time\_scale** と **fixed\_frame\_rate\_flag** がビットストリームに存在することを規定する。**timing\_info\_present\_flag** が 0 に等しいことは、**num\_units\_in\_tick**、**time\_scale** と **fixed\_frame\_rate\_flag** がビットストリームに存在しないことを規定する。

**num\_units\_in\_tick** は、クロックチックカウンタの 1 増加 (1 クロックチックと呼ばれる) に対応する、周

波数  $\text{time\_scale}$  Hz のクロック動作の時間単位の数である。 $\text{num\_units\_in\_tick}$  は 0 より大きくなければならない。1 クロックチックは、符号化データで表現できる時間の最小間隔である。例えば、ビデオ信号のクロック周波数が  $60000 \div 1001$  Hz の時、 $\text{time\_scale}$  は 60000 に等しくてもよく、かつ  $\text{num\_units\_in\_tick}$  は 1001 に等しくてもよい。等式 C-1 参照。

$\text{time\_scale}$  は 1 秒間に経過する時間単位の数である。例えば、27MHz のクロックを用いて時間を計測する時間軸システムは 27000000 の  $\text{time\_scale}$  をもつ。 $\text{time\_scale}$  は 0 より大きくなければならない。

$\text{fixed\_frame\_rate\_flag}$  が 1 に等しいことは、出力順序で連続する任意の 2 つのピクチャの HRD 出力時刻の時間的な距離が以下のように制約されることを示す。 $\text{fixed\_frame\_rate\_flag}$  が 0 に等しいことは、出力順序で連続する任意の 2 つのピクチャの HRD 出力時刻の時間的な距離に適用される、そのような制約がないことを示す。

ここで  $n$  は出力される（出力順序で） $n$  番目のピクチャを示し、ピクチャ  $n$  が（出力順序で）ビットストリームの最後に出力されるピクチャでないとすると、各々のピクチャ  $n$  に対して  $\Delta t_{fi,dpb}(n)$  の値は以下で規定される。

$$\Delta t_{fi,dpb}(n) = \Delta t_{o,dpb}(n) \div \text{DeltaTfiDivisor} \quad (\text{E-34})$$

ここで  $\Delta t_{o,dpb}(n)$  は等式 C-13 により規定され、 $\text{DeltaTfiDivisor}$  はピクチャ  $n$  を含む符号化ビデオシーケンスに対する  $\text{pic\_struct\_present\_flag}$ 、 $\text{field\_pic\_flag}$  および  $\text{pic\_struct}$  の値に基づいて付表 E-6/JT-H264 によって規定される。付表 E-6/JT-H264 で “-” とマークされるエントリは、対応するシンタックス要素が  $\text{DeltaTfiDivisor}$  に影響しないことを示す。

ピクチャ  $n$  を含む符号化ビデオシーケンスに対する  $\text{fixed\_frame\_rate\_flag}$  が 1 に等しい時、等式 C-13 で使用するため規定された続くピクチャ  $n_n$  に対して、次の条件のどちらかもしくは両方が真である場合、 $\Delta t_{fi,dpb}(n)$  に対して算出された値は等式 C-1 で規定されるような  $t_c$  (ピクチャ  $n$  を含む符号化ビデオシーケンスに対する  $t_c$  の値を使用する) に等しくなければならない。

- ピクチャ  $n_n$  がピクチャ  $n$  と同じ符号化ビデオシーケンス内にある。
- ピクチャ  $n_n$  が異なる符号化ビデオシーケンス内にあり、 $\text{fixed\_frame\_rate\_flag}$  がピクチャ  $n_n$  を含む符号化ビデオシーケンス内で 1 に等しく、かつ  $\text{num\_units\_in\_tick} = \text{time\_scale}$  の値が両方の符号化ビデオシーケンスにおいて同じである。

付表E-6/JT-H264  $\Delta t_{fi,dpb}(n)$ 算出のための除数  
(ITU-T H.264)

pic_struct_present_flag	field_pic_flag	pic_struct	DeltaTfiDivisor
0	1	-	1
1	-	1	1
1	-	2	1
0	0	-	2
1	-	0	2
1	-	3	2
1	-	4	2
1	-	5	3
1	-	6	3
1	-	7	4
1	-	8	6

**nal\_hrd\_parameters\_present\_flag** が 1 に等しいことは、(タイプIIビットストリーム適合に属する) NAL HRD パラメータが存在することを規定する。**nal\_hrd\_parameters\_present\_flag** が 0 に等しいことは、NAL HRD パラメータが存在しないことを規定する。

記6 **nal\_hrd\_parameters\_present\_flag**が0に等しい時、ビットストリームの適合は、本標準で規定されない何らかの手段によって、NALシーケンスHRDパラメータ情報と全てのバッファリング期間とピクチャタイミングSEIメッセージを含む、NAL HRDパラメータの供給なしでは検証できない。

**nal\_hrd\_parameters\_present\_flag** が 1 に等しい時、NAL HRD パラメータ (E.1.2 と E.2.2 小節) がそのフラグのすぐ後に続く。

変数 **NalHrdBpPresentFlag** は次の通り導出される。

- もし以下のどれかが真であるならば、**NalHrdBpPresentFlag**の値は1に等しく設定されなければならない。
  - **nal\_hrd\_parameters\_present\_flag** がビットストリームに存在し、そして1に等しい。
  - バッファリング期間 SEI メッセージ内の、ビットストリームに存在すべき NAL HRD 動作に対するバッファリング期間の存在必要性が、本標準で規定されない何らかの手段によって、アプリケーションで決定される。
- それ以外、**NalHrdBpPresentFlag** の値は 0 に等しく設定されなければならない。

**vcl\_hrd\_parameters\_present\_flag** が 1 に等しいことは、(全てのビットストリーム適合に属する) VCL HRD パラメータが存在することを規定する。**vcl\_hrd\_parameters\_present\_flag** が 0 に等しいことは、VCL HRD パラメータが存在しないことを規定する。

記7 **vcl\_hrd\_parameters\_present\_flag**が0に等しい時、ビットストリームの適合は、本標準で規定されない何らかの手段によって、VCL HRDパラメータ、全てのバッファリング期間およびピクチャタイミングSEIメッセージの供給なしでは検証できない。

**vcl\_hrd\_parameters\_present\_flag** が 1 に等しい時、VCL HRD パラメータ (E.1.2 と E.2.2 小節) がそのフラグ

のすぐ後に続く。

変数 `VclHrdBpPresentFlag` は次の通り導出される。

- もし以下のどれかが真であるならば、`VclHrdBpPresentFlag` の値は 1 に等しく設定されなければならない。
  - `vcl_hrd_parameters_present_flag` がビットストリームに存在し、そして 1 に等しい。
  - バッファリング期間 SEI メッセージ内の、ビットストリームに存在すべき VCL HRD 動作に対するバッファリング期間の存在必要性が、本標準で規定されない何らかの手段によって、アプリケーションで決定される。
- それ以外、`VclHrdBpPresentFlag` の値は 0 に等しく設定されなければならない。

変数 `CpbDpbDelaysPresentFlag` は次の通り導出される。

- もし以下のどれかが真であるならば、`CpbDpbDelaysPresentFlag` の値は 1 に等しく設定されなければならない。
  - `nal_hrd_parameters_present_flag` がビットストリームに存在し、そして 1 に等しい。
  - `vcl_hrd_parameters_present_flag` がビットストリームに存在し、そして 1 に等しい。
  - ピクチャタイミング SEI メッセージ内の、ビットストリームに存在すべき CPB と DPB 出力遅延の存在必要性が、本標準で規定されない何らかの手段によって、アプリケーションで決定される。
- それ以外、`CpbDpbDelaysPresentFlag` の値は 0 に等しく設定されなければならない。

`low_delay_hrd_flag` は付属資料 C で規定されるような HRD 動作モードを規定する。`fixed_frame_rate_flag` が 1 に等しい時、`low_delay_hrd_flag` は 0 に等しくなければならない。

記8 `low_delay_hrd_flag` が 1 に等しい時、アクセスユニットで使用されるビット数により、名目上の CPB 除去時刻に違反する“大きいピクチャ”が許容される。このような“大きなピクチャ”が時折のみ起きることは、期待されるが必要とはされない。

`pic_struct_present_flag` が 1 に等しいことは、`pic_struct` シンタックス要素を含むピクチャタイミング SEI メッセージ (D.2.2 小節) が存在することを規定する。`pic_struct_present_flag` が 0 に等しいことは、ピクチャタイミング SEI メッセージの中に `pic_struct` シンタックス要素が存在しないことを規定する。`pic_struct_present_flag` が存在しない場合、その値は 0 に等しいと推定されなければならない。

`bitstream_restriction_flag` が 1 に等しいことは、以下の符号化ビデオシーケンスビットストリーム制限パラメータが存在することを規定する。`bitstream_restriction_flag` が 0 に等しいことは、以下の符号化ビデオシーケンスビットストリーム制限パラメータが存在しないことを規定する。

`motion_vectors_over_pic_boundaries_flag` が 0 に等しいことは、ピクチャ境界外のサンプルや、値がピクチャ境界外の 1 つ以上のサンプルを用いて導出される分数サンプル位置にあるサンプルが、任意のサンプルを INTER 予測することに用いられないことを示す。`motion_vectors_over_pic_boundaries_flag` が 1 に等しいことは、ピクチャ境界外の 1 つ以上のサンプルが、INTER 予測で用いられても良いことを示す。`motion_vectors_over_pic_boundaries_flag` シンタックス要素が存在しない時、`motion_vectors_over_pic_boundaries_flag` 値は 1 に等しいと推定されなければならない。

`max_bytes_per_pic_denom` は、符号化ビデオシーケンス内の任意の符号化ピクチャに関連付けられる VCL NAL ユニットのサイズの和によって超えないバイト数を示す。

NAL ユニットストリームの中にあるピクチャを表現するバイト数は、そのピクチャに対する VCL NAL ユニットデータのバイトの総数（すなわち、VCL NAL ユニットの NumBytesInNALUnit 変数の合計）としてこの目的のために規定される。max\_bytes\_per\_pic\_denom の値は 0 から 16 のそれぞれの値を含む範囲でなければならない。

max\_bytes\_per\_pic\_denom に従って次が適用される。

- もし max\_bytes\_per\_pic\_denom が 0 に等しいならば、制限は示されない。
- それ以外（max\_bytes\_per\_pic\_denom が 0 に等しくない）、その符号化ビデオシーケンス中で以下のバイト数より大きく符号化ピクチャが表されてはならない。

$$(\text{PicSizeInMbs} * \text{RawMbBits}) \div (8 * \text{max\_bytes\_per\_pic\_denom}) \quad (\text{E-35})$$

max\_bytes\_per\_pic\_denom シンタックス要素が存在しない時、max\_bytes\_per\_pic\_denom の値は 2 に等しいと推定されなければならない。

max\_bits\_per\_mb\_denom は、符号化ビデオシーケンスの任意のピクチャ内の任意のマクロブロックに対する macroblock\_layer( )データの符号化ビットの最大数を示す。max\_bits\_per\_mb\_denom の値は 0 から 16 のそれぞれの値を含む範囲でなければならない。

max\_bits\_per\_mb\_denom に従って次が適用される。

- もし max\_bits\_per\_mb\_denom が 0 に等しいならば、制限は示されない。
- それ以外（max\_bits\_per\_mb\_denom が 0 に等しくない）、そのビットストリーム中で以下のビット数より大きく符号化 macroblock\_layer( )が表されてはならない。

$$(128 + \text{RawMbBits}) \div \text{max\_bits\_per\_mb\_denom} \quad (\text{E-36})$$

entropy\_coding\_mode\_flag に従って、macroblock\_layer( )データのビットが次の通り数えられる。

- もし entropy\_coding\_mode\_flag が 0 に等しいならば、macroblock\_layer( )データのビット数は、あるマクロブロックに対する macroblock\_layer( )シンタックス構造内のビット数によって与えられる。
- それ以外（entropy\_coding\_mode\_flag が 1 に等しい）、あるマクロブロックに対する macroblock\_layer( )データのビット数は、そのマクロブロックに関連する macroblock\_layer( )を構文解析する時に read\_bits(1) が 9.3.3.2.2 小小小節および 9.3.3.2.3 小小小節で呼ばれる回数によって与えられる。

max\_bits\_per\_mb\_denom が存在しない時、max\_bits\_per\_mb\_denom の値は 1 に等しいと推定されなければならない。

log2\_max\_mv\_length\_horizontal と log2\_max\_mv\_length\_vertical は、符号化ビデオシーケンス内の全てのピクチャに対して、4 分の 1 輝度サンプル単位で復号された水平および垂直の動きベクトル成分それぞれの最大絶対値を示す。n の値は、動きベクトル成分の値が 4 分の 1 輝度サンプル変位単位で  $-2^n$  から  $2^n - 1$  でそれぞれの値を含む範囲を超えてはならないことを保証する。log2\_max\_mv\_length\_horizontal の値は 0 から 16 のそれぞれの値を含む範囲でなければならない。log2\_max\_mv\_length\_vertical の値は 0 から 16 のそれぞれの値を含む範囲でなければならない。log2\_max\_mv\_length\_horizontal が存在しない時、log2\_max\_mv\_length\_horizontal と log2\_max\_mv\_length\_vertical の値は 16 に等しいと推定されなければならない。

記9 復号された水平または垂直の動きベクトル成分の最大絶対値は付属資料Aで規定されるようなプロファイルとレベルの限定によっても制約される。

**num\_reorder\_frames** は、符号化ビデオシーケンス内の任意のフレーム、相補的フィールドペア、または非ペアフィールドに復号順序で先行し出力順序で後続する、フレーム、相補的フィールドペア、または非ペアフィールドの最大数を示す。num\_reorder\_frames の値は 0 から max\_dec\_frame\_buffering のそれぞれの値を含む範囲でなければならない。num\_reorder\_frames シンタックス要素が存在しない時、num\_reorder\_frames の値は次のように推定されなければならない。

- もし profile\_idc が 44、86、100、110、122、または 244 に等しく、かつ constraint\_set3\_flag が 1 に等しいならば、num\_reorder\_frames の値は 0 に等しいと推定されなければならない。
- それ以外 (profile\_idc が 44、86、100、110、122、または 244 に等しくないか、または constraint\_set3\_flag が 0 に等しい)、num\_reorder\_frames の値は MaxDpbSize に等しいと推定されなければならない。

**max\_dec\_frame\_buffering** は、フレームバッファの単位で HRD 復号ピクチャバッファ(DPB)の必要とされるサイズを規定する。符号化ビデオシーケンスは、ピクチャタイミング SEI メッセージの dpb\_output\_delay によって規定される出力時刻で復号ピクチャの出力を可能とするために、Max(1,max\_dec\_frame\_buffering)より大きいフレームバッファのサイズをもつ復号ピクチャバッファを要求してはならない。max\_dec\_frame\_buffering の値は num\_ref\_frames から (A.3.1 あるいは A.3.2 小節で規定されるような)MaxDpbSize のそれぞれの値を含む範囲でなければならない。max\_dec\_frame\_buffering シンタックス要素が存在しない時、max\_dec\_frame\_buffering の値は次のように推定されなければならない。

- もし profile\_idc が 44 または 244 に等しく、かつ constraint\_set3\_flag が 1 に等しいならば、max\_dec\_frame\_buffering の値は 0 に等しいと推定されなければならない。
- それ以外 (profile\_idc が 44 または 244 に等しくないか、または constraint\_set3\_flag が 0 に等しい)、max\_dec\_frame\_buffering の値は MaxDpbSize に等しいと推定されなければならない。

## E.2.2 HRDパラメータの意味

**cpb\_cnt\_minus1** に 1 を加えたものはビットストリームの代替 CPB 規定の数を規定する。cpb\_cnt\_minus1 の値は 0 から 31 のそれぞれの値を含む範囲でなければならない。low\_delay\_hrd\_flag が 1 に等しい時、cpb\_cnt\_minus1 は 0 に等しくなければならない。cpb\_cnt\_minus1 が存在しない時、それは 0 に等しいと推定されなければならない。

**bit\_rate\_scale** は (bit\_rate\_value\_minus1[SchedSelIdx]とともに) SchedSelIdx 番目の CPB の最大入力ビットレートを規定する。

**cpb\_size\_scale** は (cpb\_size\_value\_minus1[SchedSelIdx]とともに) SchedSelIdx 番目の CPB の CPB サイズを規定する。

**bit\_rate\_value\_minus1[SchedSelIdx]**は (bit\_rate\_scale とともに) SchedSelIdx 番目の CPB に対する最大入力ビットレートを規定する。bit\_rate\_value\_minus1[SchedSelIdx]は 0 から  $2^{32}-2$  のそれぞれの値を含む範囲でなければならない。あるゆる SchedSelIdx>0 に対して、bit\_rate\_value\_minus1[SchedSelIdx] は、bit\_rate\_value\_minus1[SchedSelIdx-1]より大きくななければならない。ビット毎秒単位のビットレートは以下によって導出される。

$$\text{BitRate}[\text{SchedSelIdx}]=(\text{bit\_rate\_value\_minus1}[\text{SchedSelIdx}]+1)*2^{(6+\text{bit\_rate\_scale})} \quad (\text{E-37})$$

bit\_rate\_value\_minus1[SchedSelIdx]シンタックス要素が存在しない時、BitRate[SchedSelIdx]の値は次の通り推定されなければならない。

- もし profile\_idc が 66、77 もしくは 88 に等しいならば、BitRate[SchedSelIdx]は、VCL HRD パラメータに対して  $1000 \cdot \text{MaxBR}$  に等しいと推定されなければならない、かつ NAL HRD パラメータに対して  $1200 \cdot \text{MaxBR}$  に等しいと推定されなければならない。ここで MaxBR は A.3.1 小節で規定される。
- それ以外、BitRate[SchedSelIdx]は VCL HRD パラメータに対して  $\text{cpbBrVclFactor} \cdot \text{MaxBR}$  に等しいと推定されなければならない、かつ NAL HRD パラメータに対して  $\text{cpbBrNalFactor} \cdot \text{MaxBR}$  に等しいと推定されなければならない。ここで cpbBrVclFactor、cpbBrNalFactor および MaxBR は A.3.3.小節で規定される。

cpb\_size\_value\_minus1[SchedSelIdx]は SchedSelIdx 番目の CPB サイズを規定するために cpb\_size\_scale とともに用いられる。cpb\_size\_value\_minus1[SchedSelIdx]は 0 から  $2^{32}-2$  のそれぞれの値を含む範囲でなければならない。あるゆる 0 より大きい SchedSelIdx に対して、cpb\_size\_value\_minus1[SchedSelIdx] は cpb\_size\_value\_minus1[SchedSelIdx-1]より小さいか等しくなければならない。

ビット単位の CPB サイズは以下によって導出される。

$$\text{CpbSize}[\text{SchedSelIdx}] = (\text{cpb\_size\_value\_minus1}[\text{SchedSelIdx}] + 1) \cdot 2^{(4 + \text{cpb\_size\_scale})} \quad (\text{E-38})$$

cpb\_size\_value\_minus1[SchedSelIdx]シンタックス要素が存在しない時、CpbSize[SchedSelIdx]の値は次の通り推定されなければならない。

- もし profile\_idc が 66、77 もしくは 88 に等しいならば、CpbSize[SchedSelIdx]は、VCL HRD パラメータに対して  $1000 \cdot \text{MaxCPB}$  に等しいと推定されなければならない、かつ NAL HRD パラメータに対して  $1200 \cdot \text{MaxCPB}$  に等しいと推定されなければならない。ここで MaxCPB は A.3.1 小節で規定される。
- それ以外、CpbSize[SchedSelIdx]は VCL HRD パラメータに対して  $\text{cpbBrVclFactor} \cdot \text{MaxCPB}$  に等しいと推定されなければならない、かつ NAL HRD パラメータに対して  $\text{cpbBrNalFactor} \cdot \text{MaxCPB}$  に等しいと推定されなければならない。ここで cpbBrVclFactor、cpbBrNalFactor および MaxCPB は A.3.3.小節で規定される。

cbr\_flag[SchedSelIdx]が 0 に等しいことは、SchedSelIdx 番目の CPB 規定を用いる HRD でこのビットストリームを復号するために、仮想ストリーム配達スケジューラ(HSS)が断続的なビットレートモードで動作することを規定する。cbr\_flag[SchedSelIdx]が 1 に等しいことは、HSS が固定ビットレート(CBR)モードで動作することを規定する。cbr\_flag[SchedSelIdx]シンタックス要素が存在しない時、cbr\_flag の値は 0 に等しいと推定されなければならない。

initial\_cpb\_removal\_delay\_length\_minus1 は、バッファリング期間 SEI メッセージの initial\_cpb\_removal\_delay[SchedSelIdx]と initial\_cpb\_removal\_delay\_offset[SchedSelIdx]シンタックス要素のビット長を規定する。initial\_cpb\_removal\_delay[SchedSelIdx]と initial\_cpb\_removal\_delay\_offset[SchedSelIdx]の長さは initial\_cpb\_removal\_delay\_length\_minus1+1 である。initial\_cpb\_removal\_delay\_length\_minus1 シンタックス要素が、VUI パラメータシンタックス構造で 1 つより多くの hrd\_parameters()シンタックス構造に存在する場合、initial\_cpb\_removal\_delay\_length\_minus1 パラメータの値は両方の hrd\_parameters()シンタックス構造で等しくなければならない。initial\_cpb\_removal\_delay\_length\_minus1 シンタックス要素が存在しないならば、それは 23 に等しいと推定されなくてはならない。

cpb\_removal\_delay\_length\_minus1 は、cpb\_removal\_delay シンタックス要素のビット長を規定する。ピクチャタイミング SEI メッセージの cpb\_removal\_delay シンタックス要素の長さは

cpb\_removal\_delay\_length\_minus1+1 である。cpb\_removal\_delay\_length\_minus1 シンタックス要素が、VUI パラメータシンタックス構造で 1 つより多くの hrd\_parameters() シンタックス構造に存在する場合、cpb\_removal\_delay\_length\_minus1 パラメータの値は両方の hrd\_parameters() シンタックス構造で等しくなければならない。cpb\_removal\_delay\_length\_minus1 シンタックス要素が存在しないならば、それは 23 に等しいと推定されなくてはならない。

**dpb\_output\_delay\_length\_minus1** は、dpb\_output\_delay シンタックス要素のビット長を規定する。ピクチャタイミング SEI メッセージの dpb\_output\_delay シンタックス要素の長さは dpb\_output\_delay\_length\_minus1+1 である。dpb\_output\_delay\_length\_minus1 シンタックス要素が、VUI パラメータシンタックス構造で 1 つより多くの hrd\_parameters() シンタックス構造に存在する場合、dpb\_output\_delay\_length\_minus1 パラメータの値は両方の hrd\_parameters() シンタックス構造で等しくなければならない。dpb\_output\_delay\_length\_minus1 シンタックス要素が存在しないならば、それは 23 に等しいと推定されなくてはならない。

**time\_offset\_length** は 0 より大きいならば time\_offset シンタックス要素のビット長を規定する。time\_offset\_length が 0 に等しいことは time\_offset シンタックス要素が存在しないことを規定する。time\_offset\_length シンタックス要素が、VUI パラメータシンタックス構造で 1 つより多くの hrd\_parameters() シンタックス構造に存在する場合、time\_offset\_length パラメータの値は両方の hrd\_parameters() シンタックス構造で等しくなければならない。time\_offset\_length シンタックス要素が存在しないならば、それは 24 に等しいと推定されなくてはならない。

## 付属資料G

### スケーラブルビデオ符号化

(この付属資料は、本標準の必須部分を形成する)

この付属資料は、SVC と呼ばれるスケーラブルビデオ符号化を規定する。

#### G.1 本標準の規定範囲

本付属資料で規定された、1 つ以上のプロファイルの規定に適合するビットストリームと復号器は、2-9 章と付属資料 A-E への参照を用いてこの付属資料で完全に規定される。

#### G.2 標準としての参照

2 章の規定に、以下の追加が適用される。

- ISO/IEC 10646:2003, Information technology – Universal Multiple-Octet Coded Character Set (UCS)
- IETF RFC 3986 (2005), Uniform Resource Identifiers (URI): Generic Syntax

#### G.3 定義

この付属資料の目的のために、3 章での定義に加え以下の定義が適用される。これらの定義は、3 章で存在しないか、3 章での定義を置き換えるかのどちらかである。

**G.3.1 任意スライス順序(ASO)** : あるレイヤ表現の中のスライスグループのあるスライスの先頭マクロブロックのマクロブロックアドレスが、同じレイヤ表現の中の同じスライスグループの他のある先行するスライスの先頭マクロブロックのマクロブロックアドレスよりも小さくてもよく、あるいは、

あるレイヤ表現の中のスライスグループのスライスが、同じレイヤ表現の中の他の1つ以上のスライスグループの中のスライスと、インターリーブされてもよい、スライスの復号順序。

- G.3.2 関連付けされた NAL ユニット:** 復号順序においてプリフィックス NAL ユニットに直接的に後続する NAL ユニット。
- G.3.3 B スライス:** 各々のブロックのサンプル値を予測するために、最大 2 つの動きベクトルと参照インデックスを用いるレイヤ内 INTRA 予測または INTER 予測を用いて復号されてよいスライス。
- G.3.4 ベースレイヤ:** ビットストリームの中の 1 と 5 に等しい nal\_unit\_type シンタックス要素をもつ全ての NAL ユニットを含み、14、15 または 20 に等しい nal\_unit\_type シンタックス要素をもつ NAL ユニットのいずれも含まず、付属資料 A で規定されるプロファイルの 1 つ以上に適合するビットストリームサブセット。
- G.3.5 ベース品質レイヤ表現:** 0 に等しい quality\_id シンタックス要素に関連付けられているアクセスユニットのターゲット依存表現のレイヤ表現。
- G.3.6 ビットストリームサブセット:** 0 個以上の NAL ユニットの破棄することによりサブセットとしてビットストリームから導出されるビットストリーム。ビットストリームサブセットは、サブビットストリームとも呼ばれる。
- G.3.7 (マクロブロックペアの) ボトムマクロブロック:** マクロブロックペアに対するサンプルのボトム行のサンプルを含むマクロブロックペアの中のマクロブロック。フィールドマクロブロックペアでは、ボトムマクロブロックは、マクロブロックペアの空間領域に位置する、フレームまたはレイヤフレームの、各々、ボトムフィールドまたはレイヤボトムフィールドの領域からのサンプルを表す。フレームマクロブロックペアでは、ボトムマクロブロックは、マクロブロックペアの空間領域の下半分に位置するフレームまたはレイヤフレームのサンプルを表す。
- G.3.8 スケーラブル拡張 NAL ユニットの符号化スライス:** EI スライス、EP スライスまたは EB スライスをを含む符号化スライス NAL ユニット。
- G.3.9 相補的参照フィールドペア:** 2 つの符号化フィールドとして復号順序における連続したアクセスユニットに存在する 2 つの参照フィールドであり、アクセスユニットのターゲット依存表現は frame\_num シンタックス要素の同じ値を共有し、復号順序において 2 番目のフィールドは IDR ピクチャではなく、2 番目のアクセスユニットのターゲット依存表現は 5 に等しい memory\_management\_control\_operation シンタックス要素を含まないもの、あるいは相補的参照ベースフィールドペアの総称。
- G.3.10 相補的参照ベースフィールドペア:** 復号順序において連続するアクセスユニットから構築される 2 つの参照ベースフィールドであり、アクセスユニットのターゲット依存表現は frame\_num シンタックス要素の同じ値を共有し、復号順序における 2 番目のアクセスユニットは IDR アクセスユニットではなく、2 番目のアクセスユニットのターゲット依存表現は 5 に等しい memory\_management\_control\_operation シンタックス要素を含まないもの。相補的参照ベースフィールドペアは、相補的参照フィールドペアである。
- G.3.11 依存表現:** NAL ユニットヘッダの一部としてあるいは関連付けされたプリフィックス NAL ユニットにより提供される、dependency\_id シンタックス要素の同じ値に関連付けられたアクセスユニット内の VCL NAL ユニットのサブセット。依存表現は、1 つ以上のレイヤ表現からなる。

- G.3.12** **EB スライス** : *INTRA* 予測または *INTER* 予測または参照レイヤ表現のシンタックス要素および導出された変数からのレイヤ間予測を使って復号されてよいスライス。EB スライスの *INTER* 予測については、最大 2 つの動きベクトルと参照インデックスが各々のブロックのサンプル値を予測するために用いられる。
- G.3.13** **EI スライス** : *INTRA* 予測のみを利用して復号される、*I* スライスもしくは *SI* スライスではないスライス。
- G.3.14** **EP スライス** : *INTRA* 予測または *INTER* 予測または参照レイヤ表現のシンタックス要素および導出された変数からのレイヤ間予測を使って復号されてよいスライス。EP スライスの *INTER* 予測については、最大 1 つの動きベクトルと参照インデックスが各々のブロックのサンプル値を予測するために用いられる。
- G.3.15** **フィールドマクロブロック** : 単一のフィールドまたはレイヤフィールドからのサンプルを含むマクロブロック。
- G.3.16** **フィルプリフィックス NAL ユニット** : フィルデータ NAL ユニットである関連付けされた NAL ユニットをもつプリフィックス NAL ユニット。
- G.3.17** **フレームマクロブロック** : フレームまたはレイヤフレームの、各々、2 つのフィールドまたはレイヤフィールドからのサンプルを含むマクロブロック。
- G.3.18** **I スライス** : レイヤ内 *INTRA* 予測のみを使用し復号されるスライス。
- G.3.19** **瞬時復号更新 (IDR) ピクチャ** : IDR ピクチャを復号直後に、復号処理が全ての参照ピクチャを“参照不使用”としてマークすることになる、アクセスユニット内のターゲット依存表現の全てのスライスが *I* または *EI* スライスである符号化ピクチャ。IDR ピクチャ復号後、復号順序に続く全ての符号化ピクチャは、IDR ピクチャの前に復号されたどんなピクチャからの *INTER* 予測なしで復号が可能である。各符号化ビデオシーケンスの最初のピクチャは IDR ピクチャである。
- G.3.20** **レイヤ間 INTRA 予測** : 参照レイヤ表現の *INTRA* 符号化マクロブロックの復号サンプルから導出されるレイヤ間予測。
- G.3.21** **レイヤ間予測** : 参照レイヤ表現のシンタックス要素、導出された変数、または復号サンプルから導出される予測。
- G.3.22** **レイヤ内 INTRA 予測** : 同じ復号スライスの復号サンプルから導出される予測。
- G.3.23** **INTRA 予測** : レイヤ内 *INTRA* 予測、またはレイヤ間 *INTRA* 予測、またはレイヤ内 *INTRA* 予測と参照レイヤ表現のシンタックス要素と導出された変数からのレイヤ間予測の組み合わせの総称。
- G.3.24** **INTRA スライス** : *I* スライスもしくは *EI* スライスの総称。
- G.3.25** **レイヤボトムフィールド** : レイヤフレームを構成する 2 つのレイヤフィールドのうちの 1 つ。レイヤボトムフィールドの各行は、空間的にレイヤトップフィールドの対応する行の直下に位置する。
- G.3.26** **レイヤフィールド** : レイヤフレームの交互の行を集めたもの。レイヤフレームは、2 つのレイヤフィールド、すなわちレイヤトップフィールドとレイヤボトムフィールドからなる。
- G.3.27** **レイヤフレーム** : レイヤフレームは、モノクロームフォーマットにおいてはフィールドまたはフレームの中間復号結果を表現する 1 つの輝度サンプルの配列を、あるいは 4:2:0、4:2:2、または 4:4:4

色フォーマットにおいてはフィールドまたはフレームの中間復号結果を表現する1つの輝度サンプルの配列と対応する2つの色差サンプルの配列を、含む。1つのレイヤフレームは、2つのレイヤフィールド、すなわちレイヤトップフィールドとレイヤボトムフィールドから構成される。

- G.3.28 レイヤピクチャ**：レイヤフィールドあるいはレイヤフレームの総称。
- G.3.29 レイヤトップフィールド**：レイヤフレームを構成する2つのレイヤフィールドの1つ。レイヤトップフィールドの各行は、空間的にレイヤボトムフィールドの対応する行の直上に位置する。
- G.3.30 レイヤ表現**：VCL NAL ユニットヘッダの一部としてあるいは関連付けされたプリフィックス NAL ユニットにより提供される、dependency\_id および quality\_id シンタックス要素の同じ値に関連付けられたアクセスユニット内の VCL NAL ユニットのサブセット。1個以上のレイヤ表現が、依存表現を表現する。
- G.3.31 レイヤ表現識別子**：これによりアクセスユニット内の特定のレイヤ表現が一意に識別される整数値。
- G.3.32 マクロブロック**：3つのサンプル配列を持つピクチャまたはレイヤピクチャの輝度サンプルの16×16ブロックと2つの対応する色差サンプルのブロック、あるいはモノクロームピクチャまたはレイヤピクチャのサンプルの16×16ブロック。スライスまたはマクロブロックペアのマクロブロックへの分割は、パーティションである。
- G.3.33 マクロブロック適応フレーム/フィールド復号**：いくつかのマクロブロックはフレームマクロブロックとして復号されてよく、他はフィールドマクロブロックに復号されてよい、符号化フレームまたはレイヤ表現の復号処理。
- G.3.34 マクロブロックアドレス**：マクロブロック適応フレーム/フィールド復号が使用されない場合、マクロブロックアドレスは、ピクチャまたはレイヤピクチャのマクロブロックラスタ走査でのマクロブロックのインデックスであり、ピクチャまたはレイヤピクチャの左上のマクロブロックに対するゼロから始まる。マクロブロック適応フレーム/フィールド復号が使用される場合、マクロブロックペアのトップマクロブロックのマクロブロックアドレスは、ピクチャまたはレイヤピクチャのマクロブロックペアラスタ走査でのマクロブロックペアのインデックスの2倍である。そして、マクロブロックペアのボトムマクロブロックのマクロブロックアドレスは、対応するトップマクロブロックのマクロブロックアドレスに1を加算したものである。各マクロブロックペアのトップマクロブロックのマクロブロックアドレスは偶数であり、各マクロブロックペアのボトムマクロブロックのマクロブロックアドレスは奇数である。
- G.3.35 マクロブロック位置**：(x,y) で表記するピクチャまたはレイヤピクチャ中のマクロブロック2次元座標。ピクチャまたはレイヤピクチャの最も左上のマクロブロックは、(x,y) = (0,0) となる。マクロブロック列が左から右に向かう毎、xは1ずつ増加する。マクロブロック適応フレーム/フィールド復号が使用されない場合、各マクロブロック行が上から下へ向かう毎に、yは1ずつ増加する。マクロブロック適応フレーム/フィールド復号が使用される場合、各マクロブロックペア行が上から下へ向かう毎に、yは2ずつ増加し、マクロブロックがボトムマクロブロックの場合には更に1増加する。
- G.3.36 マクロブロックペア**：フレームまたはレイヤフレーム内の垂直に隣接するマクロブロックのペアで、マクロブロック適応フレーム/フィールド復号において、組み合わせて使用される。スライスのマクロブロックペアへの分割は、パーティションである。

- G.3.37** マクロブロックからスライスグループへのマップ：ピクチャまたはレイヤピクチャのマクロブロックからスライスグループへのマッピングの手段。マクロブロックからスライスグループへのマップは、各符号化マクロブロックに対して1つ存在する数値のリストである。それは、各符号化マクロブロックが属するスライスグループを規定する。
- G.3.38** マップユニットからスライスグループへのマップ：ピクチャまたはレイヤピクチャのスライスグループマップユニットからスライスグループへのマッピングの手段。マップユニットからスライスグループへのマップは、各スライスグループマップユニットに対して1つ存在する数値のリストである。それは、各符号化スライスグループマップユニットが属するスライスグループを規定する。
- G.3.39** 非ペア参照ベースフィールド：相補的参照ベースフィールドペアの一部でない参照ベースフィールド。非ペア参照ベースフィールドは、非ペア参照フィールドである。
- G.3.40** P スライス：各ブロックのサンプル値を予測するために、レイヤ内 INTRA 予測、または最大1つの動きベクトルと参照インデックスを使う INTER 予測を用いて復号されたよいスライス。
- G.3.41** パラメータ：SVC シーケンスパラメータセットもしくはピクチャパラメータセットのシンタックス要素。パラメータは、定義語である量子化パラメータの一部としても用いられる。
- G.3.42** ピクチャパラメータセット：各スライスヘッダにある pic\_parameter\_set\_id シンタックス要素によって決定される、0個以上のレイヤ表現に適用されるシンタックス要素を含んだシンタックス構造。
- G.3.43** プリフィックス NAL ユニット：nal\_unit\_type が 1、5、あるいは 12 に等しい NAL ユニットに復号順序で直に先行し、nal\_unit\_type が 14 に等しい NAL ユニット。復号順序でプリフィックス NAL ユニットに直に後続する NAL ユニットは、関連付けされた NAL ユニットと呼ばれる。プリフィックス NAL ユニットは、関連付けされた NAL ユニットに関連付けされるデータを含み、これは、関連付けされた NAL ユニットの一部分であると考えられる。
- G.3.44** 参照ベースフィールド：アクセスユニットの 0 に等しくない nal\_ref\_idc シンタックス要素、1 に等しい store\_ref\_base\_pic\_flag シンタックス要素、および 1 に等しい field\_pic\_flag シンタックス要素をもつベース品質レイヤ表現、およびベース品質レイヤ表現のレイヤ間予測により参照されるそのアクセスユニットの全てのレイヤ表現の復号により得られる、参照フィールド。参照ベースフィールドは、復号処理の出力ではなく、符号化フィールドの P、B、EP、および EB スライスまたは符号化フレームのフィールドマクロブロックが復号される際の INTER 予測に使用されるかもしれない。参照ベースピクチャも参照のこと。
- G.3.45** 参照ベースフレーム：アクセスユニットの 0 に等しくない nal\_ref\_idc シンタックス要素、1 に等しい store\_ref\_base\_pic\_flag、および 0 に等しい field\_pic\_flag シンタックス要素をもつベース品質レイヤ表現、およびベース品質レイヤ表現のレイヤ間予測により参照されるそのアクセスユニットの全てのレイヤ表現の復号により得られる、参照フレーム。参照ベースフレームは、復号処理の出力ではなく、符号化フレームの P、B、EP、および EB スライスが復号される際の INTER 予測に使用されるかもしれない。参照ベースピクチャも参照すること。
- G.3.46** 参照ベースピクチャ：アクセスユニットの 0 に等しくない nal\_ref\_idc シンタックス要素および 1 に等しい store\_ref\_base\_pic\_flag シンタックス要素をもつベース品質レイヤ表現、およびベース品質レイヤ表現のレイヤ間予測により参照されるそのアクセスユニットの全てのレイヤ表現の復号により得られる、参照ピクチャ。参照ベースピクチャは、復号処理の出力ではなく、参照ベースピクチャのサンプルは、復号順序で後続するピクチャの復号処理における INTER 予測に使用されるかもし

れない。参照ベースピクチャは参照ベースフィールドあるいは参照ベースフレームに対する総称である。

- G.3.47 参照フィールド**：参照フィールドは、符号化フィールドの *P*、*B*、*EP*、あるいは *EB* スライスまたは符号化フレームのフィールドマクロブロックが復号される際、*INTER* 予測に使用されるかもしれない。参照ピクチャも参照のこと。
- G.3.48 参照フレーム**：参照フレームは、符号化フレームの *P*、*B*、*EP*、あるいは *EB* スライスが復号される際、*INTER* 予測に使用されるかもしれない。参照ピクチャも参照のこと。
- G.3.49 参照レイヤマクロブロック**：参照レイヤ表現のマクロブロック。
- G.3.50 参照レイヤ表現**：アクセスユニットの特定のレイヤ表現に対する参照レイヤ表現は、その特定のレイヤ表現のレイヤ間予測に使用されるレイヤ表現である。参照レイヤ表現は、レイヤ間予測のために参照レイヤ表現を使用するレイヤ表現と同じアクセスユニットにある。
- G.3.51 参照ピクチャ**：ターゲット依存表現に関連付けられた `nal_ref_idc` シンタックス要素が 0 に等しくないアクセスユニットを復号することで得られる復号ピクチャ、または参照ベースピクチャの総称。参照ピクチャは復号順序における後続ピクチャの復号処理の中で、*INTER* 予測で用いられるかもしれないサンプルを含んでいる。
- G.3.52 参照ピクチャリスト**：*P*、*B*、*EP*、あるいは *EB* スライスの *INTER* 予測に使用される参照ピクチャのリスト。*P* あるいは *EP* スライスの復号処理においては、1 つの参照ピクチャリストが存在する。*B* あるいは *EB* スライスの復号処理においては、2 つの参照ピクチャリストが存在する。
- G.3.53 参照ピクチャリスト 0**：*P*、*B*、*EP*、あるいは *EB* スライスの *INTER* 予測に使用される参照ピクチャリスト。*P* および *EP* スライスに対して使用される全ての *INTER* 予測は、参照ピクチャリスト 0 を使用する。参照ピクチャリスト 0 は、*B* あるいは *EB* スライスに対する *INTER* 予測のために用いられる 2 つの参照ピクチャリストの中の片方であり、他方は参照ピクチャリスト 1 である。
- G.3.54 参照ピクチャリスト 1**：*B* あるいは *EB* スライスの *INTER* 予測に用いられる参照ピクチャリスト。参照ピクチャリスト 1 は、*B* あるいは *EB* スライスに対する *INTER* 予測のために用いられる 2 つの参照ピクチャリストのうちの片方であり、他方は参照ピクチャリスト 0 である。
- G.3.55 スケーラブルビットストリーム**：そのスケーラブルビットストリームと同一でない 1 個以上のビットストリームサブセットが本標準に適合するもうひとつのビットストリームを形成する特質をもつビットストリーム。
- G.3.56 シーケンスパラメータセット**：*I*、*P*、および *B* スライスの各スライスヘッダ中にある `pic_parameter_set_id` シンタックス要素により参照されるピクチャパラメータセット中にある `seq_parameter_set_id` シンタックス要素の内容により決定される、0 に等しい `dependency_id` および 0 に等しい `quality_id` シンタックス要素を持つ、0 個以上の複数レイヤ表現に適用されるシンタックス要素を含むシンタックス構造。
- G.3.57 スライス**：特定のスライスグループ内のラスタ走査で連続した順番の、整数個のマクロブロックまたはマクロブロックペア。ピクチャまたはレイヤピクチャにおいては、各スライスグループからスライスへの分割はパーティションとなる。スライスは、スライスグループの中のラスタ走査で連続した順番のマクロブロックやマクロブロックペアを含んでいるが、そのマクロブロックやマクロブロックペアは、ピクチャまたはレイヤピクチャの内部で連続している必要はない。マクロブロック

アドレスは、(スライスヘッダで表現される)スライスにおける最初のマクロブロックアドレスと、マクロブロックからスライスグループへのマップから導出される。

- G.3.58** **スライスグループ**：ピクチャまたはレイヤピクチャのマクロブロックまたはマクロブロックペアのサブセット。ピクチャまたはレイヤピクチャからスライスグループへの分割は、ピクチャまたはレイヤピクチャのパーティションである。このパーティションは、マクロブロックからスライスグループへのマップによって規定される。
- G.3.59** **空間的 INTRA 予測**：レイヤ内 INTRA 予測を参照のこと。
- G.3.60** **サブビットストリーム**：ビットストリームのサブセット。サブビットストリームは、ビットストリームサブセットとも呼ばれる。
- G.3.61** **サブセット**：サブセット（部分集合）は、サブセットが導出される集合にも含まれる要素だけを含む。サブセットは、それが導出される集合と同一かもしれない。
- G.3.62** **サブセットシーケンスパラメータセット**：EI、EP、およびEB スライスの各スライスヘッダ中にある pic\_parameter\_set\_id シンタックス要素により参照されるピクチャパラメータセット中にある seq\_parameter\_set\_id シンタックス要素の内容により決定される、0 に等しくない dependency\_id または 0 に等しくない quality\_id シンタックス要素を持つ、0 個以上のレイヤ表現に適用されるシンタックス要素を含むシンタックス構造。
- G.3.63** **SVC シーケンスパラメータセット**：シーケンスパラメータセットあるいはサブセットシーケンスパラメータセットの総称。
- G.3.64** **SVC シーケンスパラメータセット RBSP**：シーケンスパラメータセット RBSP あるいはサブセットシーケンスパラメータセット RBSP の総称。
- G.3.65** **ターゲット依存表現**：アクセスユニットの全ての依存表現の中で、dependency\_id シンタックス要素の最大値に関連付けられているアクセスユニットの依存表現。
- G.3.66** **ターゲットレイヤ表現**：アクセスユニットのターゲット依存表現の全てのレイヤ表現の中で、quality\_id シンタックス要素の最大値に関連付けられているアクセスユニットのターゲット依存表現のレイヤ表現。
- G.3.67** **(マクロブロックペアの) トップマクロブロック**：マクロブロックペアに対するサンプルのトップ行のサンプルを含むマクロブロックペアの中のマクロブロック。フィールドマクロブロックペアでは、トップマクロブロックは、マクロブロックペアの空間領域に位置するフレームまたはレイヤフレームのトップフィールドまたはレイヤトップフィールド領域からのサンプルを表す。フレームマクロブロックペアでは、トップマクロブロックは、マクロブロックペアの空間領域の上半分に位置するフレームまたはレイヤフレームのサンプルを表す。
- G.3.68** **VCL NAL ユニット**：符号化スライス NAL ユニットと、VCL プリフィックス NAL ユニットの総称。
- G.3.69** **VCL プリフィックス NAL ユニット**：nal\_unit\_type が 1 あるいは 5 に等しい関連付けされた NAL ユニットのプリフィックス NAL ユニット。

#### G.4 略語

4 章の規定が適用される。

## G.5 規則

5章の規定が適用される。

## G.6 情報源、符号化、復号、および出力データフォーマット、走査処理、隣接および参照レイヤ関係

6節の規定は、シーケンスパラメータセットを SVC シーケンスパラメータセットに置き換えて適用される。加えて、以下の処理が規定される。

### G.6.1 参照レイヤマクロブロックの導出処理

この処理は、no\_inter\_layer\_pred\_flag が 0 に等しい場合のみ起動される。

この処理の入力は、下記である。

- 現マクロブロックの左上輝度サンプル相対の輝度位置(xP,yP)
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックであるか、そして、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerMbType

この処理の出力は下記である。

- 参照レイヤマクロブロックを規定するマクロブロックアドレス mbAddrRefLayer
- 参照レイヤマクロブロックの左上の輝度サンプル相対の輝度位置(xB,yB)

変数 shiftX、shiftY、scaleX、scaleY、offsetX、および offsetY は、下記の順序付けられたステップで規定されるように導出される。

記 1 これらの変数は輝度位置(xP,yP)、変数 fieldMbFlag、あるいは現マクロブロックに依存しない。

1. refW、refH、scaledW、scaledH、offsetX、および offsetY は、下記により導出される。

$$\text{refW} = \text{RefLayerPicWidthInSamples}_L \quad (\text{G-1})$$

$$\text{refH} = \text{RefLayerPicHeightInSamples}_L \quad (\text{G-2})$$

$$\text{scaledW} = \text{ScaledRefLayerPicWidthInSamples}_L \quad (\text{G-3})$$

$$\text{scaledH} = \text{ScaledRefLayerPicHeightInSamples}_L \quad (\text{G-4})$$

$$\text{offsetX} = \text{ScaledRefLayerLeftOffset} \quad (\text{G-5})$$

$$\text{offsetY} = \text{ScaledRefLayerTopOffset}/(1+\text{field\_pic\_flag}) \quad (\text{G-6})$$

2. 変数 shiftX と shiftY は、下記で導出される。

$$\text{shiftX} = ((\text{level\_idc} \leq 30) ? 16 : (31 - \text{Ceil}(\text{Log}_2(\text{refW})))) \quad (\text{G-7})$$

$$\text{shiftY} = ((\text{level\_idc} \leq 30) ? 16 : (31 - \text{Ceil}(\text{Log}_2(\text{refH})))) \quad (\text{G-8})$$

3. 変数  $scaleX$  と  $scaleY$  は、下記で導出される。

$$scaleX = ((refW \ll shiftX) + (scaledW >> 1)) / scaledW \quad (G-9)$$

$$scaleY = ((refH \ll shiftY) + (scaledH >> 1)) / scaledH \quad (G-10)$$

参照レイヤピクチャの左上サンプル相対の参照レイヤ輝度位置( $xRef, yRef$ )は、下記の順序付けられたステップで規定されるように導出される。

1. 6.4.1小節で規定される逆マクロブロック走査処理が、 $CurrMbAddr$ を入力として起動され、出力が( $xM, yM$ )に割り当てられる。この6.4.1小節における処理の起動に対して、現マクロブロックは、 $fieldMbFlag$ が1に等しい時フィールドマクロブロックとして扱われ、 $fieldMbFlag$ が0に等しい時フレームマクロブロックとして扱われる。

2. 輝度位置( $xC, yC$ )は次式で導出される。

$$xC = xM + xP \quad (G-11)$$

$$yC = yM + yP * (1 + fieldMbFlag - field_pic_flag) \quad (G-12)$$

3. 参照レイヤ輝度位置は次式で導出される。

$$xRef = ((xC - offsetX) * scaleX + (1 \ll (shiftX - 1))) \gg shiftX \quad (G-13)$$

$$yRef = ((yC - offsetY) * scaleY + (1 \ll (shiftY - 1))) \gg shiftY \quad (G-14)$$

参照レイヤマクロブロックアドレス $mbAddrRefLayer$ および参照レイヤマクロブロック $mbAddrRefLayer$ の左上サンプル相対の輝度位置( $xB, yB$ )は、下記で導出される。

- もし  $MbaffFrameFlag$  が 0 に等しく、かつ  $RefLayerMbaffFrameFlag$  が 0 に等しいならば、以下の順序付けられたステップが規定される。

1. 参照レイヤマクロブロックアドレス $mbAddrRefLayer$ が、下記で導出される。

- もし下記のいずれかの条件が真ならば、 $mbAddrRefLayer$ は、利用不可とマークされる。

-  $xRef$ が0より小さいか、または $RefLayerPicWidthInSamples_L$ 以上である。

-  $yRef$ が0より小さいか、または $RefLayerPicHeightInSamples_L$ 以上である。

- それ以外、マクロブロックアドレス $mbAddrRefLayer$ は次式で導出される

$$mbAddrRefLayer = (yRef / 16) * RefLayerPicWidthInMbs + (xRef / 16) \quad (G-15)$$

2. 輝度位置( $xB, yB$ )は、以下で導出される。

- もし $mbAddrRefLayer$ が利用不可ならば、( $xB, yB$ )は、利用不可とマークされる。

- それ以外( $mbAddrRefLayer$ が利用可能)、( $xB, yB$ )は、( $xRef \% 16, yRef \% 16$ )に等しく設定される。

- それ以外 ( $MbaffFrameFlag$  が 1 に等しい、もしくは  $RefLayerMbaffFrameFlag$  が 1 に等しい)、下記の順序付けられたステップが規定される

記2 MbaffFrameFlagが1に等しいか、またはRefLayerMbaffFrameFlagが1に等しい場合、field\_pic\_flagとRefLayerFieldPicFlagはどちらも0に等しい（G.7.4.3.4小節参照）。

1. 仮想参照レイヤマクロブロックアドレスvirtMbAddrRefLayerは、下記で導出される。

- もし下記のいずれかの条件が真ならば、virtMbAddrRefLayerは利用不可とマークされる。

- xRefが0より小さいか、またはxRefがRefLayerPicWidthInSamples<sub>L</sub>以上である

- yRefが0より小さいか、またはyRefがRefLayerPicHeightInSamples<sub>L</sub>以上である

- それ以外、以下が適用される。

- もしRefLayerMbaffFrameFlagが1に等しいならば、virtMbAddrRefLayerは次で導出される。

$$\text{virtMbAddrRefLayer} = 2 * ((y\text{Ref}/32) * \text{RefLayerPicWidthInMbs} + (x\text{Ref}/16)) + (y\text{Ref} \% 32) / 16 \quad (\text{G-16})$$

- それ以外(RefLayerMbaffFrameFlagが0に等しい)、virtMbAddrRefLayerは次で導出される。

$$\text{virtMbAddrRefLayer} = (y\text{Ref}/16) * \text{RefLayerPicWidthInMbs} + (x\text{Ref}/16) \quad (\text{G-17})$$

2. 参照レイヤマクロブロックアドレスmbAddrRefLayerと輝度位置(xB,yB)は、下記で導出される。

- もしvirtMbAddrRefLayerが利用不可ならば、参照レイヤマクロブロックアドレスmbAddrRefLayerと輝度位置(xB,yB)は、利用不可とマークされる。

- それ以外、もしfieldMbFlagが0に等しく、かつrefLayerFieldMbRef[virtMbAddrRefLayer]が1に等しいならば、G.6.1.1小節で規定されるフィールド参照レイヤマクロブロックからフレーム参照レイヤマクロブロックへの変換処理が、virtMbAddrRefLayer、(xRef,yRef)、およびrefLayerMbTypeを入力として起動され、出力はmbAddrRefLayerと(xB,yB)に割り当てられる。

- それ以外、もしfieldMbFlagが1に等しく、かつrefLayerFieldMbRef[virtMbAddrRefLayer]が0に等しいならば、G.6.1.2小節で規定されるフレーム参照レイヤマクロブロックからフィールド参照レイヤマクロブロックへの変換処理が、virtMbAddrRefLayerと(xRef,yRef)を入力として起動され、出力はmbAddrRefLayerと(xB,yB)に割り当てられる。

- それ以外(virtMbAddrRefLayerが利用可能、かつfieldMbFlagがrefLayerFieldMbRef[virtMbAddrRefLayer]に等しい)、mbAddrRefLayerおよび(xB,yB)は下記で導出される。

$$\text{mbAddrRefLayer} = ((\text{virtMbAddrRefLayer} \gg \text{fieldMbFlag}) \ll \text{fieldMbFlag}) + (\text{CurrMbAddr} \% 2) * \text{fieldMbFlag} \quad (\text{G-18})$$

$$xB = (x\text{Ref} \% 16) \quad (\text{G-19})$$

$$yB = (y\text{Ref} \% (16 \ll \text{fieldMbFlag})) \gg \text{fieldMbFlag} \quad (\text{G-20})$$

#### G.6.1.1 フィールド参照レイヤマクロブロックからフレーム参照レイヤマクロブロックへの変換処理

この処理への入力は、下記である。

- 仮想参照レイヤマクロブロックアドレス `virtMbAddrRefLayer`
- 参照レイヤピクチャの左上輝度サンプル相対の参照レイヤ輝度位置(`xRef,yRef`)
- 参照レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する `RefLayerPicSizeInMbs` 個の要素を持つ 1 次元配列 `refLayerMbType`

この処理の出力は、下記である。

- 参照レイヤマクロブロックのマクロブロックアドレス `mbAddrRefLayer`
- 参照レイヤマクロブロックの左上輝度サンプル相対の輝度位置(`xB,yB`)

マクロブロックアドレス`mbAddrRefLayerTop`と`mbAddrRefLayerBot`は、次で導出される。

$$\text{mbAddrRefLayerTop} = \text{virtMbAddrRefLayer} - (\text{virtMbAddrRefLayer} \% 2) \quad (\text{G-21})$$

$$\text{mbAddrRefLayerBot} = \text{mbAddrRefLayerTop} + 1 \quad (\text{G-22})$$

参照レイヤマクロブロックアドレス`mbAddrRefLayer`は、下記で導出される。

- もし `refLayerMbType[mbAddrRefLayerTop]` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4` または `I_BL` に等しいならば、`mbAddrRefLayer` は `mbAddrRefLayerBot` に等しく設定される。
- それ以外(`refLayerMbType[mbAddrRefLayerTop]` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、または `I_BL` に等しくない)、`mbAddrRefLayer` は、`mbAddrRefLayerTop` に等しく設定される。

輝度位置(`xB,yB`)は、次で導出される

$$xB = xRef \% 16 \quad (\text{G-23})$$

$$yB = 8 * ((yRef / 16) \% 2) + 4 * ((yRef \% 16) / 8) \quad (\text{G-24})$$

#### G.6.1.2 フレーム参照レイヤマクロブロックからフィールド参照レイヤマクロブロックへの変換処理

この処理への入力は、下記である。

- 仮想参照レイヤマクロブロックアドレス `virtMbAddrRefLayer`
- 参照レイヤピクチャの左上輝度サンプル相対の仮想参照レイヤ輝度位置(`xRef,yRef`)

この処理の出力は、下記である。

- 参照レイヤマクロブロックのマクロブロックアドレス `mbAddrRefLayer`
- 参照レイヤマクロブロックの左上輝度サンプル相対の輝度位置(`xB,yB`)

マクロブロックアドレス`mbAddrRefLayerTop`および`mbAddrRefLayerBot`は、下記で導出される。

- もし `RefLayerMbaffFrameFlag` が 1 に等しいならば、`mbAddrRefLayerTop` と `mbAddrRefLayerBot` は下記で導出される。

$$\text{mbAddrRefLayerTop} = \text{virtMbAddrRefLayer} - (\text{virtMbAddrRefLayer} \% 2) \quad (\text{G-25})$$

$$\text{mbAddrRefLayerBot} = \text{mbAddrRefLayerTop} + 1 \quad (\text{G-26})$$

- それ以外(RefLayerMbaffFrameFlag が 0 に等しい)、mbAddrRefLayerTop と mbAddrRefLayerBot は次で導出される。

$$\text{mbAddrRefLayerTop} = \text{virtMbAddrRefLayer} - ((y\text{Ref}/16)\%2) * \text{RefLayerPicWidthInMbs} \quad (\text{G-27})$$

$$\text{mbAddrRefLayerBot} = \text{mbAddrRefLayerTop} + \text{RefLayerPicWidthInMbs} \quad (\text{G-28})$$

参照レイヤマクロブロックアドレスは、以下で導出される。

- もし(yRef%16)が 8 より小さいならば、mbAddrRefLayer は mbAddrRefLayerTop に等しく設定される。
- それ以外((yRef%16)が 8 より大きいか等しい)、mbAddrRefLayer は、mbAddrRefLayerBot に等しく設定される。

輝度位置(xB,yB)は、次で導出される。

$$xB = x\text{Ref}\%16 \quad (\text{G-29})$$

$$yB = 8 * (((y\text{Ref}\%16)/4)\%2) + 4 * ((y\text{Ref}/16)\%2) \quad (\text{G-30})$$

## G.6.2 参照レイヤパーティションのための導出処理

この処理は、no\_inter\_layer\_pred\_flagが0に等しい場合のみ起動される。

この処理の入力は下記である。

- 現マクロブロックの左上輝度サンプル相対の輝度位置(xP,yP)
- 現マクロブロックが、フィールドマクロブロックかフレームマクロブロックかを規定する fieldMbFlag
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックであるか、そして、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックについて、マクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素による 1 次元配列 refLayerMbType
- 参照レイヤ表現のマクロブロックに対するサブマクロブロックタイプを規定する (RefLayerPicSizeInMbs)×4 配列の refLayerSubMbType

この処理の出力は下記である。

- 参照レイヤマクロブロックを規定するマクロブロックアドレス mbAddrRefLayer
- 参照レイヤマクロブロック mbAddrRefLayer 内の参照レイヤマクロブロックパーティションを規定するマクロブロックパーティションインデックス mbPartIdxRefLayer
- 参照レイヤマクロブロック mbAddrRefLayer のマクロブロックパーティション mbPartIdxRefLayer 内の参照レイヤサブマクロブロックパーティションを規定するサブマクロブロックパーティションインデックス subMbPartIdxRefLayer

G.6.1小節で規定される参照レイヤマクロブロックに対する導出処理が、輝度位置(xP,yP)、fieldMbFlag、refLayerFieldMbFlag、およびrefLayerMbTypeを入力として起動され、出力がmbAddrRefLayerおよび(xB,yB)に割り当てられる。

参照レイヤマクロブロックパーティションインデックスmbPartIdxRefLayerと参照レイヤサブマクロブロックパーティションインデックスsubMbPartIdxRefLayerは、下記のように導出される。

- もし mbAddrRefLayer が利用不可ならば、mbPartIdxRefLayer と subMbPartIdxRefLayer は利用不可とマークされる。
- それ以外、もし refLayerMbType[mbAddrRefLayer]が I\_PCM、I\_16x16、I\_8x8、I\_4x4、または I\_BL に等しいならば、mbPartIdxRefLayer および subMbPartIdxRefLayer は 0 に等しく設定される。
- それ以外、もし refLayerMbType[mbAddrRefLayer]が B\_Skip または B\_Direct\_16x16 に等しいならば、下記が適用される。
  - もし ref\_layer\_dq\_id が 0 より大きいならば、mbPartIdxRefLayer および subMbPartIdxRefLayer は 0 に等しく設定される。
  - それ以外(ref\_layer\_dq\_id が 0 に等しい)、mbPartIdxRefLayer および subMbPartIdxRefLayer は次で導出される。

$$\text{mbPartIdxRefLayer} = 2 * (\text{yB} / 8) + (\text{xB} / 8) \quad (\text{G-31})$$

$$\text{subMbPartIdxRefLayer} = 2 * ((\text{yB} \% 8) / 4) + ((\text{xB} \% 8) / 4) \quad (\text{G-32})$$

- それ以外(mbAddrRefLayer が利用可能であり、かつ refLayerMbType[mbAddrRefLayer]が I\_PCM、I\_16x16、I\_8x8、I\_4x4、I\_BL、B\_Skip、または B\_Direct\_16x16 に等しくない)、下記の順序付けられたステップが規定される。

1. 参照レイヤマクロブロックパーティションインデックスmbPartIdxRefLayerは、輝度位置(xB,yB)を含む参照レイヤマクロブロックmbAddrRefLayer内のマクロブロックパーティションのマクロブロックパーティションインデックスに等しく設定される。参照レイヤマクロブロックmbAddrRefLayerのマクロブロックパーティションは、マクロブロックタイプrefLayerMbType[mbAddrRefLayer]とG.8.4.1小節での規定によって与えられる。

2. 参照レイヤサブマクロブロックパーティションインデックスsubMbPartIdxRefLayerは、下記で導出される。

- もし NumMbPart(refLayerMbType[mbAddrRefLayer]) が 4 より小さいならば、subMbPartIdxRefLayer は 0 に等しく設定される。
- それ以外、(NumMbPart(refLayerMbType[mbAddrRefLayer])が 4 に等しい)、下記が適用される。
  - もし refLayerSubMbType[mbAddrRefLayer][mbPartIdxRefLayer]が B\_Direct\_8x8 に等しいならば、下記が適用される。
  - もし ref\_layer\_dq\_id が 0 より大きいならば、subMbPartIdxRefLayer は 0 に等しく設定される。

- それ以外(ref\_layer\_dq\_idが0に等しい)、subMbPartIdxRefLayerは次で導出される。

$$\text{subMbPartIdxRefLayer} = 2 * ((yB \% 8) / 4) + ((xB \% 8) / 4) \quad (\text{G-33})$$

- それ以外(refLayerSubMbType[mbAddrRefLayer][mbPartIdxRefLayer]が B\_Direct\_8x8 と等しくないとき)、参照レイヤサブマクロブロックパーティションインデックス subMbPartIdxRefLayer は、輝度位置 (xB,yB) を含む参照レイヤマクロブロック mbAddrRefLayer のマクロブロックパーティション mbPartIdxRefLayer 内のサブマクロブロックパーティションのサブマクロブロックパーティションインデックスに等しく設定される。参照レイヤマクロブロック mbAddrRefLayer 内の参照レイヤマクロブロックパーティション mbPartIdxRefLayer のサブマクロブロックパーティションは、サブマクロブロックタイプ refLayerSubMbType[mbAddrRefLayer][mbPartIdxRefLayer]と G.8.4.1 小小節の規定によって与えられる。

### G.6.3 再サンプリングにおける参照レイヤサンプル位置に対する導出処理

この処理への入力は以下である。

- 輝度あるいは色差成分のどちらが再サンプリング処理の対象であるのかを規定する変数 chromaFlag
- 現マクロブロックの左上サンプル相対のサンプル位置(xP,yP)
- 現マクロブロックがフィールドあるいはフレームマクロブロックのいずれであるのかを規定する変数 fieldMbFlag
- (RefLayerFrameMbsOnlyFlag が 0 に等しいあるいは frame\_mbs\_only\_flag が 0 に等しい場合) トップあるいはボトムフィールドのいずれが再サンプリング処理の対象であるのかを規定する変数 botFieldFlag

この処理の出力は、以下を規定する参照レイヤサンプル位置(xRef16,yRef16)である。

- もし RefLayerFrameMbsOnlyFlag が 1 に等しいあるいは RefLayerFieldPicFlag が 1 に等しいならば、(xRef16,yRef16)は参照レイヤピクチャの左上サンプルとの相対で 1/16 サンプルを単位とする参照レイヤサンプル位置を規定する。
- それ以外 (RefLayerFrameMbsOnlyFlag が 0 に等しいかつ RefLayerFieldPicFlag が 0 に等しい)、(xRef16,yRef16)は参照レイヤピクチャの botFieldFlag によって規定されるフィールドの左上サンプルとの相対で 1/16 サンプルを単位とする参照レイヤサンプル位置を規定する。

変数 subW、subH、shiftX、shiftY、scaleX、scaleY、offsetX、offsetY、addX、addY、deltaX、および deltaY は以下の順序付けられたステップにおいて規定されるように導出される。

記 これらの変数は入力サンプル位置(xP,yP)、入力変数fieldMbFlag、あるいは現マクロブロックアドレスCurrMbAddrには依存しない。

1. 0に等しいchromaFlagに対するLおよび1に等しいchromaFlagに対するCによって置換されるZを用いて、変数refW、refH、scaledW、およびscaledHが次によって導出される。

$$\text{refW} = \text{RefLayerPicWidthInSamples}_Z \quad (\text{G-34})$$

$$\text{refH} = \text{RefLayerPicHeightInSamples}_Z * (1 + \text{RefLayerFieldPicFlag}) \quad (\text{G-35})$$

$$\text{scaledW} = \text{ScaledRefLayerPicWidthInSamples}_Z \quad (\text{G-36})$$

$$\text{scaledH} = \text{ScaledRefLayerPicHeightInSamples}_Z * (1 + \text{field\_pic\_flag}) \quad (\text{G-37})$$

2. frame\_mbs\_only\_flagが0に等しくかつRefLayerFrameMbsOnlyFlagが1に等しい場合、変数scaledHは次によって修正される。

$$\text{scaledH}=\text{scaledH}/2 \quad (\text{G-38})$$

3. 変数refPhaseX、refPhaseY、phaseX、phaseY、subW、およびsubHは次によって導出される。

$$\text{refPhaseX}=\left(\text{chromaFlag}=0\right) ? 0 : \left(\text{ref\_layer\_chroma\_phase\_x\_plus1\_flag}-1\right) \quad (\text{G-39})$$

$$\text{refPhaseY}=\left(\text{chromaFlag}=0\right) ? 0 : \left(\text{ref\_layer\_chroma\_phase\_y\_plus1}-1\right) \quad (\text{G-40})$$

$$\text{phaseX}=\left(\text{chromaFlag}=0\right) ? 0 : \left(\text{chroma\_phase\_x\_plus1\_flag}-1\right) \quad (\text{G-41})$$

$$\text{phaseY}=\left(\text{chromaFlag}=0\right) ? 0 : \left(\text{chroma\_phase\_y\_plus1}-1\right) \quad (\text{G-42})$$

$$\text{subW}=\left(\text{chromaFlag}=0\right) ? 1 : \text{SubWidthC} \quad (\text{G-43})$$

$$\text{subH}=\left(\text{chromaFlag}=0\right) ? 1 : \text{SubHeightC} \quad (\text{G-44})$$

4. RefLayerFrameMbsOnlyFlagが0に等しいあるいはframe\_mbs\_only\_flagが0に等しい場合、以下が適用される。

- 変数 phaseY は次によって修正される。

$$\text{phaseY}=\text{phaseY}+4*\text{botFieldFlag} \quad (\text{G-45})$$

- 変数 refPhaseY は次によって修正される。

- もし RefLayerFrameMbsOnlyFlag が 1 に等しいならば、変数 refPhaseY は次によって修正される。

$$\text{refPhaseY}=2*\text{refPhaseY}+2 \quad (\text{G-46})$$

- それ以外 (RefLayerFrameMbsOnlyFlag が 0 に等しい)、変数 refPhaseY は次によって修正される。

$$\text{refPhaseY}=\text{refPhaseY}+4*\text{botFieldFlag} \quad (\text{G-47})$$

5. 変数shiftXおよびshiftYが次によって導出される。

$$\text{shiftX}=\left(\text{level\_idc}\leq 30\right) ? 16 : \left(31-\text{Ceil}\left(\text{Log2}\left(\text{refW}\right)\right)\right) \quad (\text{G-48})$$

$$\text{shiftY}=\left(\text{level\_idc}\leq 30\right) ? 16 : \left(31-\text{Ceil}\left(\text{Log2}\left(\text{refH}\right)\right)\right) \quad (\text{G-49})$$

6. 変数scaleXおよびscaleYが次によって導出される。

$$\text{scaleX}=\left(\left(\text{refW}\ll\text{shiftX}\right)+\left(\text{scaledW}\gg 1\right)\right)/\text{scaledW} \quad (\text{G-50})$$

$$\text{scaleY}=\left(\left(\text{refH}\ll\text{shiftY}\right)+\left(\text{scaledH}\gg 1\right)\right)/\text{scaledH} \quad (\text{G-51})$$

7. 変数offsetX、addX、およびdeltaXは次によって導出される。

$$\text{offsetX}=\text{ScaledRefLayerLeftOffset}/\text{subW} \quad (\text{G-52})$$

$$\text{addX}=\left(\left(\text{refW}*(2+\text{phaseX})\right)\ll\left(\text{shiftX}-2\right)\right)+\left(\text{scaledW}\gg 1\right)/\text{scaledW}+\left(1\ll\left(\text{shiftX}-5\right)\right) \quad (\text{G-53})$$

$$\text{deltaX}=4*(2+\text{refPhaseX}) \quad (\text{G-54})$$

8. 変数offsetY、addY、およびdeltaYは以下のように導出される。

- もし RefLayerFrameMbsOnlyFlag が 1 に等しくかつ frame\_mbs\_only\_flag が 1 に等しいならば、変数 offsetY、addY、および deltaY は次によって導出される。

$$\text{offsetY}=\text{ScaledRefLayerTopOffset}/\text{subH} \quad (\text{G-55})$$

$$\text{addY}=\left(\left(\left(\text{refH}*(2+\text{phaseY})\right)\ll(\text{shiftY}-2)\right)+(\text{scaledH}\gg 1)\right)/\text{scaledH}+(1\ll(\text{shiftY}-5)) \quad (\text{G-56})$$

$$\text{deltaY}=4*(2+\text{refPhaseY}) \quad (\text{G-57})$$

- それ以外 (RefLayerFrameMbsOnlyFlag が 0 に等しいあるいは frame\_mbs\_only\_flag が 0 に等しい)、変数 offsetY、addY、および deltaY は次によって導出される。

$$\text{offsetY}=\text{ScaledRefLayerTopOffset}/(2*\text{subH}) \quad (\text{G-58})$$

$$\text{addY}=\left(\left(\left(\text{refH}*(2+\text{phaseY})\right)\ll(\text{shiftY}-3)\right)+(\text{scaledH}\gg 1)\right)/\text{scaledH}+(1\ll(\text{shiftY}-5)) \quad (\text{G-59})$$

$$\text{deltaY}=2*(2+\text{refPhaseY}) \quad (\text{G-60})$$

以下の順序づけられたステップにおいて規定されるようにサンプル位置(xC,yC)が導出される。

1. 6.4.1小節において規定されている逆マクロブロック走査処理が入力としてCurrMbAddrを用いて起動され、出力は(xM,yM)に割り当てられる。6.4.1小節における処理のこの起動に対して、fieldMbFlagが1に等しい場合は現マクロブロックはフィールドマクロブロックとして扱われ、fieldMbFlagが0に等しい場合は現マクロブロックはフレームマクロブロックとして扱われる。
2. サンプル位置(xC,yC)が次によって導出される。

$$\text{xC}=\text{xP}+(\text{xM}\gg(\text{subW}-1)) \quad (\text{G-61})$$

$$\text{yC}=\text{yP}+(\text{yM}\gg(\text{subH}-1+\text{fieldMbFlag}-\text{field\_pic\_flag})) \quad (\text{G-62})$$

3. RefLayerFrameMbsOnlyFlagが0に等しいあるいはframe\_mbs\_only\_flagが0に等しい場合、サンプル位置(xC,yC)の垂直成分が次によって修正される。

$$\text{yC}=\text{yC}\gg(1-\text{fieldMbFlag}) \quad (\text{G-63})$$

参照レイヤサンプル位置(xRef16,yRef16)が次によって導出される。

$$\text{xRef16}=\left(\left(\left(\text{xC}-\text{offsetX}\right)*\text{scaleX}+\text{addX}\right)\gg(\text{shiftX}-4)\right)-\text{deltaX} \quad (\text{G-64})$$

$$\text{yRef16}=\left(\left(\left(\text{yC}-\text{offsetY}\right)*\text{scaleY}+\text{addY}\right)\gg(\text{shiftY}-4)\right)-\text{deltaY} \quad (\text{G-65})$$

## G.7 シンタックスおよび意味

この節はこの付属資料において規定されている1つ以上のプロファイルに適合する符号化ビデオシーケンスに対するシンタックスと意味を規定する。

### G.7.1 表形式におけるシンタックス規定の手法

7.1 節における規定が適用される。

### G.7.2 シンタックス関数、カテゴリ、記述子の規定

7.2 節における規定が適用される。加えて以下のシンタックス関数が規定される。

in\_crop\_window(mbAddr)は以下の順序付けられたステップによって規定される。

1. 変数mbXは(mbAddr%PicWidthInMbs)に等しく設定される。
2. 変数mbY0およびmbY1が以下のように導出される。
  - もし MbaffFrameFlag が 0 に等しいならば、mbY0 および mbY1 は(mbAddr/PicWidthInMbs)に等しく設定される。

- それ以外 (MbaffFrameFlag が 1 に等しい)、mbY0 は $(2*((mbAddr/PicWidthInMbs)/2))$ に等しく設定され、かつ mbIdxY1 は $(mbIdxY0+1)$ に等しく設定される。
- 3. 変数scalMbHは $(16*(1+field\_pic\_flag))$ に等しく設定される。
- 4. in\_crop\_window(mbAddr)の戻り値は以下のように導出される。
  - もし以下の全ての条件が真であるならば、in\_crop\_window(mbAddr)の戻り値は TRUE に等しい。
    - no\_inter\_layer\_pred\_flag は 0 に等しい。
    - mbX は $((ScaledRefLayerLeftOffset+15)/16)$ より大きいあるいは等しい。
    - mbX は $((ScaledRefLayerLeftOffset+ScaledRefLayerPicWidthInSamples_L)/16)$ より小さい。
    - mbY0 は $((ScaledRefLayerTopOffset+scalMbH-1)/scalMbH)$ より大きいあるいは等しい。
    - mbY1 は $((ScaledRefLayerTopOffset+ScaledRefLayerPicHeightInSamples_L)/scalMbH)$ より小さい。
  - それ以外、in\_crop\_window(mbAddr)の戻り値は FALSE に等しい。

### G.7.3 表形式におけるシンタックス

#### G.7.3.1 NALユニットシンタックス

シンタックス表は 7.3.1 小節において規定される。

##### G.7.3.1.1 NALユニットヘッダSVC拡張シンタックス

nal_unit_header_svc_extension( ) {	C	記述子
reserved_one_bit	All	u(1)
idr_flag	All	u(1)
priority_id	All	u(6)
no_inter_layer_pred_flag	All	u(1)
dependency_id	All	u(3)
quality_id	All	u(4)
temporal_id	All	u(3)
use_ref_base_pic_flag	All	u(1)
discardable_flag	All	u(1)
output_flag	All	u(1)
reserved_three_2bits	All	u(2)
}		

#### G.7.3.2 未加工のバイトシーケンスペイロードおよびRBSP追従ビットシンタックス

##### G.7.3.2.1 シーケンスパラメータセットRBSPシンタックス

シンタックス表は 7.3.2.1 小節において規定される。

### G.7.3.2.1.1 シーケンスパラメータセットデータシンタックス

シンタックス表は 7.3.2.1.1 小小小節において規定される。

#### G.7.3.2.1.1.1 スケーリングリストシンタックス

シンタックス表は 7.3.2.1.1.1 小小小節において規定される。

### G.7.3.2.1.2 シーケンスパラメータセット拡張RBSPシンタックス

シンタックス表は 7.3.2.1.2 小小小節において規定される。

#### G.7.3.2.1.3 サブセットシーケンスパラメータセットRBSPシンタックス

	C	記述子
subset_seq_parameter_set_rbsp( ) {		
seq_parameter_set_data( )	0	
if( profile_idc == 83    profile_idc == 86 ) {		
seq_parameter_set_svc_extension( )	0	
<b>svc_vui_parameters_present_flag</b>	0	u(1)
if( svc_vui_parameters_present_flag == 1 )		
svc_vui_parameters_extension( )	0	
}		
<b>additional_extension2_flag</b>	0	u(1)
if( additional_extension2_flag == 1 )		
while( more_rbsp_data( ) )		
<b>additional_extension2_data_flag</b>	0	u(1)
rbsp_trailing_bits( )	0	
}		

#### G.7.3.2.1.4 シーケンスパラメータセットSVC拡張シンタックス

	C	記述子
seq_parameter_set_svc_extension( ) {		
<b>inter_layer_deblocking_filter_control_present_flag</b>	0	u(1)
<b>extended_spatial_scalability</b>	0	u(2)
if( ChromaArrayType == 1    ChromaArrayType == 2 )		
<b>chroma_phase_x_plus1_flag</b>	0	u(1)
if( ChromaArrayType == 1 )		
<b>chroma_phase_y_plus1</b>	0	u(2)
if( extended_spatial_scalability == 1 ) {		
if( ChromaArrayType > 0 ) {		
<b>seq_ref_layer_chroma_phase_x_plus1_flag</b>	0	u(1)
<b>seq_ref_layer_chroma_phase_y_plus1</b>	0	u(2)
}		

<b>seq_scaled_ref_layer_left_offset</b>	0	se(v)
<b>seq_scaled_ref_layer_top_offset</b>	0	se(v)
<b>seq_scaled_ref_layer_right_offset</b>	0	se(v)
<b>seq_scaled_ref_layer_bottom_offset</b>	0	se(v)
}		
<b>seq_tcoeff_level_prediction_flag</b>	0	u(1)
if( seq_tcoeff_level_prediction_flag ) {		
<b>adaptive_tcoeff_level_prediction_flag</b>	0	u(1)
}		
<b>slice_header_restriction_flag</b>	0	u(1)
}		

#### G.7.3.2.2 ピクチャパラメータセットRBSPシンタックス

シンタックス表は 7.3.2.2 小小節において規定される。

#### G.7.3.2.3 付加拡張情報RBSPシンタックス

シンタックス表は 7.3.2.3 小小節において規定される。

##### G.7.3.2.3.1 付加拡張情報メッセージシンタックス

シンタックス表は 7.3.2.3.1 小小小節において規定される。

#### G.7.3.2.4 アクセユニット境界RBSPシンタックス

シンタックス表は 7.3.2.4 小小節において規定される。

#### G.7.3.2.5 シーケンス終了RBSPシンタックス

シンタックス表は 7.3.2.5 小小節において規定される。

#### G.7.3.2.6 ストリーム終了RBSPシンタックス

シンタックス表は 7.3.2.6 小小節において規定される。

#### G.7.3.2.7 フィルデータRBSPシンタックス

シンタックス表は 7.3.2.7 小小節において規定される。

#### G.7.3.2.8 パーティションのないスライスレイヤRBSPシンタックス

シンタックス表は 7.3.2.8 小小節において規定される。

#### G.7.3.2.9 スライスデータパーティションRBSPシンタックス

スライスデータパーティションシンタックスはこの付属資料で規定される 1 つ以上のプロファイルに適合する符号化ビデオシーケンス中には存在しない。

#### G.7.3.2.10 RBSPスライス追従ビットシンタックス

シンタックス表は 7.3.2.10 小節において規定される。

#### G.7.3.2.11 RBSP追従ビットシンタックス

シンタックス表は 7.3.2.11 小節において規定される。

#### G.7.3.2.12 プリフィックスNALユニットRBSPシンタックス

prefix_nal_unit_rbsp() {	C	記述子
if( nal_ref_idc != 0 ) {		
<b>store_ref_base_pic_flag</b>	2	u(1)
if( ( use_ref_base_pic_flag    store_ref_base_pic_flag ) && !idr_flag )		
dec_ref_base_pic_marking()	2	
<b>prefix_nal_unit_additional_extension_flag</b>	2	u(1)
if( prefix_nal_unit_additional_extension_flag == 1 )		
while( more_rbsp_data() )		
<b>prefix_nal_unit_extension_flag</b>	2	u(1)
rbsp_trailing_bits()	2	
}		
}		

#### G.7.3.2.13 スケーラブル拡張内スライスレイヤRBSPシンタックス

slice_layer_in_scalable_extension_rbsp() {	C	記述子
slice_header_in_scalable_extension()	2	
if( !slice_skip_flag )		
slice_data_in_scalable_extension()	2   3   4	
rbsp_slice_trailing_bits()	2	
}		

#### G.7.3.3 スライスヘッダシンタックス

シンタックス表は 7.3.3 小節において規定される。

##### G.7.3.3.1 参照ピクチャリスト並替シンタックス

シンタックス表は 7.3.3.1 小節において規定される。

### G.7.3.3.2 予測重み表シンタックス

シンタックス表は 7.3.3.2 小小節において規定される。

### G.7.3.3.3 復号参照ピクチャマーキングシンタックス

シンタックス表は 7.3.3.3 小小節において規定される。

### G.7.3.3.4 スケーラブル拡張内スライスヘッダシンタックス

	C	記述子
slice_header_in_scalable_extension() {		
<b>first_mb_in_slice</b>	2	ue(v)
<b>slice_type</b>	2	ue(v)
<b>pic_parameter_set_id</b>	2	ue(v)
if( separate_colour_plane_flag == 1 )		
<b>colour_plane_id</b>	2	u(2)
<b>frame_num</b>	2	u(v)
if( !frame_mbs_only_flag ) {		
<b>field_pic_flag</b>	2	u(1)
if( field_pic_flag )		
<b>bottom_field_flag</b>	2	u(1)
}		
if( idr_flag == 1 )		
<b>idr_pic_id</b>	2	ue(v)
if( pic_order_cnt_type == 0 ) {		
<b>pic_order_cnt_lsb</b>	2	u(v)
if( pic_order_present_flag && !field_pic_flag )		
<b>delta_pic_order_cnt_bottom</b>	2	se(v)
}		
if( pic_order_cnt_type == 1 && !delta_pic_order_always_zero_flag ) {		
<b>delta_pic_order_cnt[ 0 ]</b>	2	se(v)
if( pic_order_present_flag && !field_pic_flag )		
<b>delta_pic_order_cnt[ 1 ]</b>	2	se(v)
}		
if( redundant_pic_cnt_present_flag )		
<b>redundant_pic_cnt</b>	2	ue(v)
if( quality_id == 0 ) {		
if( slice_type == EB )		

<b>direct_spatial_mv_pred_flag</b>	2	u(1)
if( slice_type == EP    slice_type == EB ) {		
<b>num_ref_idx_active_override_flag</b>	2	u(1)
if( num_ref_idx_active_override_flag ) {		
<b>num_ref_idx_l0_active_minus1</b>	2	ue(v)
if( slice_type == EB )		
<b>num_ref_idx_l1_active_minus1</b>	2	ue(v)
}		
}		
ref_pic_list_reordering()	2	
if( ( weighted_pred_flag && slice_type == EP )    ( weighted_bipred_idc == 1 && slice_type == EB ) ) {		
if( !no_inter_layer_pred_flag )		
<b>base_pred_weight_table_flag</b>	2	u(1)
if( no_inter_layer_pred_flag    !base_pred_weight_table_flag )		
pred_weight_table()	2	
}		
if( nal_ref_idc != 0 ) {		
dec_ref_pic_marking()	2	
if( !slice_header_restriction_flag ) {		
<b>store_ref_base_pic_flag</b>	2	u(1)
if( ( use_ref_base_pic_flag    store_ref_base_pic_flag ) && !idr_flag )		
dec_ref_base_pic_marking()	2	
}		
}		
}		
if( entropy_coding_mode_flag && slice_type != EI )		
<b>cabac_init_idc</b>	2	ue(v)
<b>slice_qp_delta</b>	2	se(v)
if( deblocking_filter_control_present_flag ) {		
<b>disable_deblocking_filter_idc</b>	2	ue(v)
if( disable_deblocking_filter_idc != 1 ) {		
<b>slice_alpha_c0_offset_div2</b>	2	se(v)
<b>slice_beta_offset_div2</b>	2	se(v)
}		

}		
if( quality_id == 0 && num_slice_groups_minus1 > 0 && slice_group_map_type >= 3 && slice_group_map_type <= 5 )		
<b>slice_group_change_cycle</b>	2	u(v)
if( !no_inter_layer_pred_flag && quality_id == 0 ) {		
<b>ref_layer_dq_id</b>	2	ue(v)
if( inter_layer_deblocking_filter_control_present_flag ) {		
<b>disable_inter_layer_deblocking_filter_idc</b>	2	ue(v)
if( disable_inter_layer_deblocking_filter_idc != 1 ) {		
<b>inter_layer_slice_alpha_c0_offset_div2</b>	2	se(v)
<b>inter_layer_slice_beta_offset_div2</b>	2	se(v)
}		
}		
<b>constrained_intra_resampling_flag</b>	2	u(1)
if( extended_spatial_scalability == 2 ) {		
if( ChromaArrayType > 0 ) {		
<b>ref_layer_chroma_phase_x_plus1_flag</b>	2	u(1)
<b>ref_layer_chroma_phase_y_plus1</b>	2	u(2)
}		
<b>scaled_ref_layer_left_offset</b>	2	se(v)
<b>scaled_ref_layer_top_offset</b>	2	se(v)
<b>scaled_ref_layer_right_offset</b>	2	se(v)
<b>scaled_ref_layer_bottom_offset</b>	2	se(v)
}		
}		
if( !no_inter_layer_pred_flag ) {		
<b>slice_skip_flag</b>	2	u(1)
if( slice_skip_flag )		
<b>num_mbs_in_slice_minus1</b>	2	ue(v)
else {		
<b>adaptive_base_mode_flag</b>	2	u(1)
if( !adaptive_base_mode_flag )		
<b>default_base_mode_flag</b>	2	u(1)
if( !default_base_mode_flag ) {		
<b>adaptive_motion_prediction_flag</b>	2	u(1)
if( !adaptive_motion_prediction_flag )		
<b>default_motion_prediction_flag</b>	2	u(1)
}		
<b>adaptive_residual_prediction_flag</b>	2	u(1)
if( adaptive_residual_prediction_flag )		
<b>default_residual_prediction_flag</b>	2	u(1)

}		
if( adaptive_tcoeff_level_prediction_flag == 1 )		
<b>tcoeff_level_prediction_flag</b>	2	u(1)
}		
if( !slice_header_restriction_flag ) {		
<b>scan_idx_start</b>	2	u(4)
<b>scan_idx_end</b>	2	u(4)
}		
}		

#### G.7.3.3.5 復号参照ベースピクチャーマーキングシンタックス

	C	記述子
dec_ref_base_pic_marking( ) {		
<b>adaptive_ref_base_pic_marking_mode_flag</b>	2   5	u(1)
if( adaptive_ref_base_pic_marking_mode_flag )		
do {		
<b>memory_management_control_operation</b>	2   5	ue(v)
if( memory_management_control_operation == 1 )		
<b>difference_of_pic_nums_minus1</b>	2   5	ue(v)
if( memory_management_control_operation == 2 )		
<b>long_term_pic_num</b>	2   5	ue(v)
} while( memory_management_control_operation != 0 )		
}		

#### G.7.3.4 スライスデータシンタックス

シンタックス表は 7.3.4 小節において規定される。

##### G.7.3.4.1 スケーラブル拡張内スライスデータシンタックス

	C	記述子
slice_data_in_scalable_extension( ) {		
if( entropy_coding_mode_flag )		
while( !byte_aligned( ) )		
<b>cabac_alignment_one_bit</b>	2	f(1)
CurrMbAddr = first_mb_in_slice * ( 1 + MbaffFrameFlag )		
moreDataFlag = 1		
prevMbSkipped = 0		
do {		

if( slice_type != EI )		
if( !entropy_coding_mode_flag ) {		
<b>mb_skip_run</b>	2	ue(v)
prevMbSkipped = ( mb_skip_run > 0 )		
for( i = 0; i < mb_skip_run; i++ )		
CurrMbAddr = NextMbAddress( CurrMbAddr )		
moreDataFlag = more_rbsp_data( )		
} else {		
<b>mb_skip_flag</b>	2	ae(v)
moreDataFlag = !mb_skip_flag		
}		
if( moreDataFlag ) {		
if( MbaffFrameFlag && (( CurrMbAddr % 2 ) == 0    (( CurrMbAddr % 2 ) == 1 && prevMbSkipped )) )		
<b>mb_field_decoding_flag</b>	2	u(1)   ae(v)
macroblock_layer_in_scalable_extension( )	2   3   4	
}		
if( !entropy_coding_mode_flag )		
moreDataFlag = more_rbsp_data( )		
else {		
if( slice_type != EI )		
prevMbSkipped = mb_skip_flag		
if( MbaffFrameFlag && ( CurrMbAddr % 2 ) == 0 )		
moreDataFlag = 1		
else {		
<b>end_of_slice_flag</b>	2	ae(v)
moreDataFlag = !end_of_slice_flag		
}		
}		
CurrMbAddr = NextMbAddress( CurrMbAddr )		
} while( moreDataFlag )		
}		

### G.7.3.5 マクロブロックレイヤシンタックス

シンタックス表は 7.3.5 小節において規定される。

#### G.7.3.5.1 マクロブロック予測シンタックス

シンタックス表は 7.3.5.1 小節において規定される。

### G.7.3.5.2 サブマクロブロック予測シンタックス

シンタックス表は 7.3.5.2 小小節において規定される。

### G.7.3.5.3 残差データシンタックス

シンタックス表は 7.3.5.3 小小節において規定される。

#### G.7.3.5.3.1 残差輝度シンタックス

シンタックス表は 7.3.5.3.1 小小小節において規定される。

#### G.7.3.5.3.2 残差ブロックCAVLCシンタックス

シンタックス表は 7.3.5.3.2 小小小節において規定される。

#### G.7.3.5.3.3 残差ブロックCABACシンタックス

シンタックス表は 7.3.5.3.3 小小小節において規定される。

### G.7.3.6 スケーラブル拡張内マクロブロックレイヤシンタックス

macroblock_layer_in_scalable_extension() {	C	記述子
if( in_crop_window( CurrMbAddr ) && adaptive_base_mode_flag )		
<b>base_mode_flag</b>	2	u(1)   ae(v)
if( !base_mode_flag )		
<b>mb_type</b>	2	ue(v)   ae(v)
if( !base_mode_flag && mb_type == I_PCM ) {		
while( !byte_aligned() )		
<b>pcm_alignment_zero_bit</b>	3	f(1)
for( i = 0; i < 256; i++ )		
<b>pcm_sample_luma[ i ]</b>	3	u(v)
for( i = 0; i < 2 * MbWidthC * MbHeightC; i++ )		
<b>pcm_sample_chroma[ i ]</b>	3	u(v)
} else {		
if( !base_mode_flag ) {		
noSubMbPartSizeLessThan8x8Flag = 1		
if( mb_type != I_NxN && MbPartPredMode( mb_type, 0 ) != Intra_16x16 && NumMbPart( mb_type ) == 4 ) {		
<b>sub_mb_pred_in_scalable_extension( mb_type )</b>	2	
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( sub_mb_type[ mbPartIdx ] != B_Direct_8x8 ) {		
if( NumSubMbPart( sub_mb_type [ mbPartIdx ] ) > 1 )		
noSubMbPartSizeLessThan8x8Flag = 0		

} else if( !direct_8x8_inference_flag )		
noSubMbPartSizeLessThan8x8Flag = 0		
} else {		
if( transform_8x8_mode_flag && mb_type == I_NxN )		
<b>transform_size_8x8_flag</b>	2	u(1)   ae(v)
mb_pred_in_scalable_extension( mb_type )	2	
}		
}		
if( adaptive_residual_prediction_flag && slice_type != EI && ( base_mode_flag    ( MbPartPredMode( mb_type, 0 ) != Intra_16x16 && MbPartPredMode( mb_type, 0 ) != Intra_8x8 && MbPartPredMode( mb_type, 0 ) != Intra_4x4 && mb_type != I_PCM && in_crop_window( CurrMbAddr ) ) ) )		
<b>residual_prediction_flag</b>	2	u(1)   ae(v)
if( scan_idx_end >= scan_idx_start ) {		
if( base_mode_flag    MbPartPredMode( mb_type, 0 ) != Intra_16x16 ) {		
<b>coded_block_pattern</b>	2	me(v)   ae(v)
if( CodedBlockPatternLuma > 0 && transform_8x8_mode_flag && ( base_mode_flag    ( mb_type != I_NxN && noSubMbPartSizeLessThan8x8Flag && ( mb_type != B_Direct_16x16    direct_8x8_inference_flag ) ) ) )		
<b>transform_size_8x8_flag</b>	2	u(1)   ae(v)
}		
if( CodedBlockPatternLuma > 0    CodedBlockPatternChroma > 0    ( !base_mode_flag && MbPartPredMode( mb_type, 0 ) == Intra_16x16 ) ) {		
<b>mb_qp_delta</b>	2	se(v)   ae(v)
residual( base_mode_flag, scan_idx_start, scan_idx_end )	3   4	
}		
}		
}		
}		
}		

#### G.7.3.6.1 スケーラブル拡張内マクロブロック予測シンタックス

mb_pred_in_scalable_extension( mb_type ) {	C	記述子
if( MbPartPredMode( mb_type, 0 ) == Intra_4x4    MbPartPredMode( mb_type, 0 ) == Intra_8x8    MbPartPredMode( mb_type, 0 ) == Intra_16x16 ) {		
if( MbPartPredMode( mb_type, 0 ) == Intra_4x4 )		
for( luma4x4BlkIdx = 0; luma4x4BlkIdx < 16; luma4x4BlkIdx++ ) {		
<b>prev_intra4x4_pred_mode_flag</b> [ luma4x4BlkIdx ]	2	u(1)   ae(v)
if( !prev_intra4x4_pred_mode_flag[ luma4x4BlkIdx ] )		
<b>rem_intra4x4_pred_mode</b> [ luma4x4BlkIdx ]	2	u(3)   ae(v)
}		
if( MbPartPredMode( mb_type, 0 ) == Intra_8x8 )		
for( luma8x8BlkIdx = 0; luma8x8BlkIdx < 4; luma8x8BlkIdx++ ) {		
<b>prev_intra8x8_pred_mode_flag</b> [ luma8x8BlkIdx ]	2	u(1)   ae(v)
if( !prev_intra8x8_pred_mode_flag[ luma8x8BlkIdx ] )		
<b>rem_intra8x8_pred_mode</b> [ luma8x8BlkIdx ]	2	u(3)   ae(v)
}		
if( ChromaArrayType != 0 )		
<b>intra_chroma_pred_mode</b>	2	ue(v)   ae(v)
} else if( MbPartPredMode( mb_type, 0 ) != Direct ) {		
if( in_crop_window( CurrMbAddr ) && adaptive_motion_prediction_flag ) {		
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++)		
if( MbPartPredMode( mb_type, mbPartIdx ) != Pred_L1 )		
<b>motion_prediction_flag_I0</b> [ mbPartIdx ]	2	u(1)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++)		
if( MbPartPredMode( mb_type, mbPartIdx ) != Pred_L0 )		
<b>motion_prediction_flag_I1</b> [ mbPartIdx ]	2	u(1)   ae(v)
}		
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++)		
if( ( num_ref_idx_I0_active_minus1 > 0    mb_field_decoding_flag ) && MbPartPredMode( mb_type, mbPartIdx ) != Pred_L1 && !motion_prediction_flag_I0[ mbPartIdx ] )		
<b>ref_idx_I0</b> [ mbPartIdx ]	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++)		
if( ( num_ref_idx_I1_active_minus1 > 0    mb_field_decoding_flag ) && MbPartPredMode( mb_type, mbPartIdx ) != Pred_L0 && !motion_prediction_flag_I1[ mbPartIdx ] )		
<b>ref_idx_I1</b> [ mbPartIdx ]	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++)		
if( MbPartPredMode( mb_type, mbPartIdx ) != Pred_L1 )		

for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_10</b> [ mbPartIdx ][ 0 ][ compIdx ]	2	se(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < NumMbPart( mb_type ); mbPartIdx++)		
if( MbPartPredMode( mb_type, mbPartIdx ) != Pred_L0 )		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_11</b> [ mbPartIdx ][ 0 ][ compIdx ]	2	se(v)   ae(v)
}		
}		

#### G.7.3.6.2 スケーラブル拡張内サブマクロブロック予測シンタックス

sub_mb_pred_in_scalable_extension( mb_type ) {	C	記述子
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
<b>sub_mb_type[ mbPartIdx ]</b>	2	ue(v)   ae(v)
if( in_crop_window( CurrMbAddr ) && adaptive_motion_prediction_flag ) {		
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Direct && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L1 )		
<b>motion_prediction_flag_10[ mbPartIdx ]</b>	2	u(1)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Direct && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L0 )		
<b>motion_prediction_flag_11[ mbPartIdx ]</b>	2	u(1)   ae(v)
}		
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( ( num_ref_idx_10_active_minus1 > 0    mb_field_decoding_flag ) && mb_type != P_8x8ref0 && sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L1 && !motion_prediction_flag_10[ mbPartIdx ] )		
<b>ref_idx_10[ mbPartIdx ]</b>	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( ( num_ref_idx_11_active_minus1 > 0    mb_field_decoding_flag ) && sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L0 && !motion_prediction_flag_11[ mbPartIdx ] )		
<b>ref_idx_11[ mbPartIdx ]</b>	2	te(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L1 )		
for( subMbPartIdx = 0; subMbPartIdx < NumSubMbPart( sub_mb_type[ mbPartIdx ] ); subMbPartIdx++)		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_10[ mbPartIdx ][ subMbPartIdx ][ compIdx ]</b>	2	se(v)   ae(v)
for( mbPartIdx = 0; mbPartIdx < 4; mbPartIdx++ )		
if( sub_mb_type[ mbPartIdx ] != B_Direct_8x8 && SubMbPredMode( sub_mb_type[ mbPartIdx ] ) != Pred_L0 )		
for( subMbPartIdx = 0; subMbPartIdx < NumSubMbPart( sub_mb_type[ mbPartIdx ] ); subMbPartIdx++)		
for( compIdx = 0; compIdx < 2; compIdx++ )		
<b>mvd_11[ mbPartIdx ][ subMbPartIdx ][ compIdx ]</b>	2	se(v)   ae(v)

}		
---	--	--

#### G.7.4 意味

シンタックス構造やこれらの構造(G.7.3小節内およびG.7.3小節により参照される7.3節内)のシンタックス要素に関連付けられた意味は、この節において、または7.4節への参照により、規定される。シンタックス要素の意味が1つの表あるいは複数の表の組を使用して規定される時、本標準で別に規定されない限り、表において規定されないいかなる値もビットストリームの中に存在してはならない。

1つ以上のサブビットストリームがこの付属資料で規定される1つ以上のプロファイルに適合するビットストリームから導出されても良い。G.8.8.1小節により規定される処理に従って導出されるサブビットストリームは付属資料Aにより規定される1つ以上のプロファイルまたはこの付属資料により規定される1つ以上のプロファイルに適合する。

1つ以上のサブビットストリームは付属資料Aにより規定されるプロファイルに適合する。これらのサブビットストリームのための復号は2-9章および付属資料B-Eにより規定される。

1つ以上の本付属資料により規定されるプロファイルに適合するビットストリームのための復号は、2-9章と付属資料B-Eを参照する本付属資料により完全に規定される。

2-9章と付属資料B-Eにおける規定または処理はそのまま、または特定部分の割り当てまたは代替の意味を規定することにより使用されてよい。

##### G.7.4.1 NALユニットの意味

G.7.3.1小節のシンタックス要素のための意味は、7.4.1小節において規定される。さらに以下の規定が適用される。

`nal_unit_type` が 14 に等しい NAL ユニットに関しては、`nal_ref_idc` は、復号順序において `nal_unit_type` が 14 に等しい NAL ユニットの後に続く関連 NAL ユニットの `nal_ref_idc` と等しくなければならない。

`nal_ref_idc` の値は、依存表現におけるあらゆる VCL NAL ユニットにおいて等しくなければならない。

変数 `refNalRefIdc` は、以下のように導出される。

- `nal_unit_type` が 20 と等しくないか、または `dependency_id` がアクセスユニットのあらゆる VCL NAL ユニットにおける `dependency_id` の最小値と等しいとき、`refNalRefIdc` は 0 に等しく設定される。
- それ以外、`refNalRefIdc` は、アクセスユニット内で現在の `dependency_id` よりも小さい `dependency_id` を持つ全ての VCL NAL ユニットにおける `nal_ref_idc` の最大値に等しく設定される。

`refNalRefIdc` が 0 よりも大きいとき、`nal_ref_idc` の値は 0 と等しくてはならない。

スライスを含み、かつ、アクセスユニット内の `dependency_id` の最大値に等しい `dependency_id` の値を有する NAL ユニットにおいて `nal_ref_idc` が 0 に等しいことは、アクセスユニットの全ての符号化スライス NAL ユニットは非参照ピクチャの符号化スライス NAL ユニットであることを示す。

スライスを含み、かつ、アクセスユニット内の `dependency_id` の最大値に等しい `dependency_id` の値を有する NAL ユニットにおいて `nal_ref_idc` が 0 よりも大きいことは、アクセスユニットの全ての符号化スライス

NAL ユニットは非参照ピクチャの符号化スライス NAL ユニットであることを示す。

#### G.7.4.1.1 NALユニットヘッダSVC拡張の意味

シンタックス要素 `idr_flag`、`priority_id`、`no_inter_layer_pred_flag`、`dependency_id`、`quality_id`、`temporal_id`、`use_ref_base_pic_flag`、`discardable_flag`、`output_flag` は、プリフィックス NAL ユニットに存在するときは、関連 NAL ユニットに適用されるとみなされる。

`reserved_one_bit` は 1 に等しくなければならない。`reserved_one_bit` の値は、本標準の将来の拡張によって規定されるかもしれない。復号器は `reserved_one_bit` の値を無視しなければならない。

`idr_flag` が 1 に等しいことは、NAL ユニットに対する `dependency_id` の値がアクセスユニットに対する `dependency_id` の最大値に等しいときに、現アクセスユニットが IDR アクセスユニットであることを規定する。

`idr_flag` の値は、依存表現の全ての NAL ユニットに対して同じでなければならない。フィルプリフィックス NAL ユニットに対して、`idr_flag` は 0 に等しくなければならない。VCL プリフィックス NAL ユニットに対して `idr_flag` が 1 に等しいときは、関連 NAL ユニットの `nal_unit_type` は 5 に等しくなければならない。VCL プリフィックス NAL ユニットに対して `idr_flag` が 0 に等しいときは、関連 NAL ユニットの `nal_unit_type` は 1 に等しくなければならない。

`nal_ref_idc` が 0 に等しいとき、`idr_flag` の値は 0 に等しくなければならない。

NAL ユニットにおいて、`idr_flag` が存在する場合、7.4.1 小節で導出される変数 `IdrPicFlag` は `idr_flag` と等しく設定し修正される。

`dIdMax` を、アクセスユニットの NAL ユニット内で `dependency_id` の最大値とする。`dependency_id` が `dIdMax` に等しい任意の NAL ユニットにおいて `IdrPicFlag` が 1 に等しい場合、アクセスユニットは IDR アクセスユニットと呼ばれる。

記1 アクセスユニットをIDRアクセスユニットとする分類や、符号化ビデオシーケンスにおける一連のアクセスユニットのパーティションは、関連NALユニット内の`dependency_id`の最大存在値に依存する。NALユニットがビットストリームから除去される時、（例・受信デバイスの性能に合わせてビットストリームを適応する目的）、アクセスユニットをIDRアクセスユニットとする分類や、符号化ビデオシーケンスにおける一連のアクセスユニットのパーティションは変更されるかもしれない。

`priority_id` は NAL ユニットに対する優先識別子を規定する。

記2 シンタックス要素`priority_id`は、本標準の中で規定される復号処理によって要求されない。シンタックス要素`priority_id`は、規定される制約の中でアプリケーションが定めた方法によって利用されるかもしれない。

`no_inter_layer_pred_flag` は、符号化スライス NAL ユニットに存在する場合、レイヤ間予測が符号化スライスの復号に利用されてよいかどうかを規定する。`no_inter_layer_pred_flag` が 1 に等しいときは、レイヤ間予測は符号化スライスの復号に利用されない。`no_inter_layer_pred_flag` が 0 に等しいときは、レイヤ間予測はマクロブロックレイヤで伝送される符号化スライスの復号に利用されてもよいし、利用されなくてもよい。

プリフィックス NAL ユニットに対しては、`no_inter_layer_pred_flag` は 1 に等しくなくてはならない。`nal_unit_type` が 20 に等しく `quality_id` が 0 よりも大きいとき、`no_inter_layer_pred_flag` は 0 に等しくなくてはならない。

`dependency_id` は NAL ユニットに対する依存識別子を規定する。`dependency_id` は VCL プリフィックス

NAL ユニット内で 0 に等しくなくてはならない。

**quality\_id** は NAL ユニットに対する品質識別子を規定する。**quality\_id** は VCL プリフィックス NAL ユニット内で 0 に等しくなくてはならない。

変数 DQId は以下のように導出される。

$$\text{DQId}=(\text{dependency\_id} \ll 4)+\text{quality\_id} \quad (\text{G-66})$$

**temporal\_id** は NAL ユニットに対する時間識別子を規定する。

**temporal\_id** の値は、アクセスユニット内の全てのプリフィックスおよび符号化スライスのスケーラブル拡張 NAL ユニットにおいて同じでなくてはならない。アクセスユニットが、**nal\_unit\_type** が 5 に等しいかもしくは **idr\_flag** が 1 に等しい NAL ユニットを含むとき、**temporal\_id** は 0 に等しくなくてはならない。

G.8.1.1 小小節で規定されるビットストリーム抽出処理を、0 以上 63 以下の範囲内の任意の値と等しい **pIdTarget**、0 以上 7 以下の範囲内の任意の値と等しい **tdTarget**、0 以上 7 以下の範囲内の任意の値と等しい **dIdTarget**、0 以上 15 以下の範囲内の **qIdTarget** を用いて起動することにより得られるビットストリームは本標準に適合しなければならない。

記3 適合ビットストリームは、**priority\_id**は0に等しく、**dependency\_id**は0に等しく、**quality\_id**は0に等しく、**temporal\_id**は0に等しい1つ以上の符号化スライスNALユニットを含む。

**use\_ref\_base\_pic\_flag** が 1 に等しいことは、参照ベースピクチャが INTER 予測処理において参照ピクチャとして利用されることを規定する。**use\_ref\_base\_pic\_flag** は 0 に等しいことは、復号ピクチャが INTER 予測処理の間参照ピクチャとして利用されることを規定する。

**use\_ref\_base\_pic\_flag** の値は依存表現内のあらゆる NAL ユニットに対して同じでなくてはならない。**use\_ref\_base\_pic\_flag** はフィルプリフィックス NAL ユニット内で 0 に等しくなくてはならない。

**discardable\_flag** が 1 に等しいことは、現 NAL ユニットは、現 NAL ユニットよりも大きな値の **dependency\_id** を有する現アクセスユニットおよび全ての後続アクセスユニットの NAL ユニットの復号に利用されないことを規定する。**discardable\_flag** が 0 に等しいことは、現 NAL ユニットは、現 NAL ユニットよりも大きな値の **dependency\_id** を有する現アクセスユニットおよび全ての後続アクセスユニットの NAL ユニットの復号に利用されるかもしれないことを規定する。

**output\_flag** は、付属資料 C に規定されている復号ピクチャ出力や除去過程に作用する。**output\_flag** の値は依存表現内のあらゆる NAL ユニットに対して同じでなくてはならない。**output\_flag** の値は相補的フィールドペアの両方のフィールドに対して同じでなくてはならない。

**output\_flag** はフィルプリフィックス NAL ユニット内で 0 に等しくなくてはならない。

**reserved\_three\_2bits** は 3 に等しくなくてはならない。**reserved\_three\_2bits** の値は本標準の将来の拡張によって規定されるかもしれない。復号器は **reserved\_three\_2bits** の値を無視しなくてはならない。

#### G.7.4.1.2 NALユニットの順序および符号化ピクチャ、アクセスユニットとビデオシーケンスへの関連

この小小節は、ビットストリームにおける NAL ユニットの順序の制約を規定する。ビットストリームにおいて、これらの制約に従う NAL ユニットの任意の順序が、文書中では、NAL ユニットの復号順序と呼ばれる。NAL ユニット内では、7.3、D.1、E.1、G.7.3、G.13.1 および G.14.1 小節のシンタックスが、シンタックス要素の復号順序を規定する。本標準に適合する復号器は、復号順序になっている NAL ユニットおよ

びそれらのシンタックス要素を受け取ることが可能でなければならない。

#### G.7.4.1.2.1 SVCシーケンスパラメータセットRBSPおよびピクチャパラメータセットRBSPの順序およびそれらのアクティブ化

記1 シーケンスおよびピクチャパラメータセット機構は、頻繁にはではなく変更する情報の伝送を、符号化マクロブロックデータの伝送から分離する。シーケンスおよびピクチャパラメータセットは、いくつかのアプリケーションにおいて、信頼できる伝送機構を使用した“帯域外”で伝達されてもよい。

ピクチャパラメータセット RBSP は、1 つ以上の符号化ピクチャの 1 つ以上のレイヤ表現の符号化スライス NAL ユニットによって参照されうるパラメータを含む。

各々のピクチャパラメータセット RBSP は、復号処理操作の開始時点では、最初はアクティブでないとなされる。復号処理操作の間の任意の与えられた瞬間で、多くとも 1 つのピクチャパラメータセット RBSP がアクティブとみなされ、そして、任意の特定ピクチャパラメータセット RBSP がアクティブピクチャパラメータセットになる時、(もしあれば) 直前のアクティブピクチャパラメータセットが非アクティブ化される。

アクティブピクチャパラメータセットに加えて、0 個以上のピクチャパラメータセット RBSP が、ターゲットレイヤ表現を復号する時、レイヤ間予測を通して参照される (DQIdMax より小さい DQId の特定の値をもつ) レイヤ表現に対して、個々にアクティブであるかもしれない。そのようなピクチャパラメータセット RBSP は、(DQIdMax より小さい) DQId の特定の値に対するアクティブレイヤピクチャパラメータセット RBSP と呼ばれる。アクティブピクチャパラメータセット RBSP の制約が、DQId の特定の値をもつアクティブレイヤピクチャパラメータセット RBSP にも適用される。

ピクチャパラメータセット RBSP (pic\_parameter\_set\_id の特定の値をもつ) がアクティブピクチャパラメータセット RBSP ではなく、それが DQIdMax と等しい DQId をもつ符号化スライス NAL ユニットによって (pic\_parameter\_set\_id のその値を使用して) 参照される時、それがアクティブ化される。このピクチャパラメータセット RBSP は、別のピクチャパラメータセット RBSP がアクティブピクチャパラメータセット RBSP になる時、それが非アクティブ化されるまで、アクティブピクチャパラメータセット RBSP と呼ばれる。そのような pic\_parameter\_set\_id の特定の値をもつピクチャパラメータセット RBSP は、そのアクティブ化の前に復号処理で利用可能にならねばならない。

ピクチャパラメータセット RBSP (pic\_parameter\_set\_id の特定の値をもつ) が DQIdMax より小さい DQId の特定の値に対するアクティブレイヤピクチャパラメータセットでなく、(pic\_parameter\_set\_id のその値を使用して) DQId の特定の値をもつ符号化スライス NAL ユニットによって参照される時、それが DQId の特定の値をもつレイヤ表現に対してアクティブ化される。このピクチャパラメータセット RBSP は、別のピクチャパラメータセット RBSP が DQId の特定の値に対するアクティブピクチャパラメータセット RBSP になる時、あるいは DQId の特定の値より小さいまたは等しい DQIdMax をもつアクセスユニットを復号する時、それが非アクティブ化されるまで、その DQId の特定の値に対するアクティブピクチャパラメータセット RBSP と呼ばれる。そのような pic\_parameter\_set\_id の特定の値をもつピクチャパラメータセット RBSP は、そのアクティブ化の前に復号処理で利用可能にならねばならない。

アクティブピクチャパラメータセット RBSP に対する pic\_parameter\_set\_id の値を含むどんなピクチャパラメータセット NAL ユニットも、それが 1 つの符号化ピクチャの最後の VCL NAL ユニットに続きかつ別の符号化ピクチャの最初の VCL NAL ユニットに先行するのでない限り、そのアクティブピクチャパラメータセット RBSP のそれと同じ内容を持たなければならない。DQIdMax より小さい DQId の特定の値に対するアクティブレイヤピクチャパラメータセット RBSP に対する pic\_parameter\_set\_id の値を含むどんなピクチャパラ

メータセット NAL ユニットも、それが 1 つの符号化ピクチャの最後の VCL NAL ユニットに続きかつ別の符号化ピクチャの最初の VCL NAL ユニットに先行するのでない限り、DQId の特定の値に対するアクティブレイヤピクチャパラメータセット RBSP のそれと同じ内容を持たなければならない。

SVC シーケンスパラメータセット RBSP は、1 つ以上のピクチャパラメータセット RBSP、または、1 つ以上のバッファリング期間 SEI メッセージを含む SEI NAL ユニットによって参照されるパラメータを含む。

各々の SVC シーケンスパラメータセット RBSP は、復号処理操作の開始時点では、最初はアクティブでないとみなされる。復号処理操作の間の任意の与えられた瞬間で、多くとも 1 つの SVC シーケンスパラメータセット RBSP がアクティブ SVC シーケンスパラメータセット RBSP とみなされ、そして、任意の特定 SVC シーケンスパラメータセット RBSP がアクティブ SVC シーケンスパラメータセット RBSP になる時、(もしあれば) 直前のアクティブな SVC シーケンスパラメータセットが非アクティブ化される。

アクティブ SVC シーケンスパラメータセットに加えて、0 個以上の SVC シーケンスパラメータセット RBSP が、ターゲットレイヤ表現を復号する時、レイヤ間予測を通して参照される (DQIdMax より小さい DQId の特定の値をもつ) レイヤ表現に対して、個々にアクティブであるかもしれない。そのような SVC シーケンスパラメータセット RBSP は、(DQIdMax より小さい) DQId の特定の値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP と呼ばれる。アクティブ SVC シーケンスパラメータセット RBSP の制約が、DQId の特定の値をもつアクティブレイヤ SVC シーケンスパラメータセット RBSP にも適用される。

seq\_parameter\_set\_id の特定の値をもつ (nal\_unit\_type が 7 に等しい) シーケンスパラメータセット RBSP が今アクティブ SVC シーケンスパラメータセット RBSP ではなく、それがピクチャパラメータセット RBSP のアクティブ化によって (seq\_parameter\_set\_id のその値を使用して) 参照され、ピクチャパラメータセット RBSP が DQIdMax と等しい DQId および 1 または 5 に等しい nal\_unit\_type をもつ符号化スライス NAL ユニットによって参照される (ピクチャパラメータセット RBSP がアクティブピクチャパラメータセット RBSP になる) 時、それがアクティブ化される。このシーケンスパラメータセット RBSP は、別の SVC シーケンスパラメータセット RBSP がアクティブ SVC シーケンスパラメータセット RBSP になる時、それが非アクティブ化されるまで、アクティブ SVC シーケンスパラメータセット RBSP と呼ばれる。そのような seq\_parameter\_set\_id の特定の値をもつシーケンスパラメータセット RBSP は、そのアクティブ化の前に復号処理で利用可能にならねばならない。

seq\_parameter\_set\_id の特定の値をもつ (nal\_unit\_type が 7 に等しい) シーケンスパラメータセット RBSP が今アクティブ SVC シーケンスパラメータセット RBSP ではなく、スケーラブルネスティング SEI メッセージに含まれず、そのアクセスユニット内で復号順序で最後のバッファリング期間 SEI メッセージを表現するバッファリング期間 SEI メッセージを含む SEI NAL ユニットによって (seq\_parameter\_set\_id のその値を使用して) 参照され、DQIdMax が 0 に等しい時、それがアクティブ化される。このシーケンスパラメータセット RBSP は、別の SVC シーケンスパラメータセット RBSP がアクティブ SVC シーケンスパラメータセット RBSP になる時、それが非アクティブ化されるまで、アクティブ SVC シーケンスパラメータセット RBSP と呼ばれる。そのような seq\_parameter\_set\_id の特定の値をもつシーケンスパラメータセット RBSP は、そのアクティブ化の前に復号処理で利用可能にならねばならない。

seq\_parameter\_set\_id の特定の値をもつ (nal\_unit\_type が 15 に等しい) サブセットシーケンスパラメータセット RBSP (pic\_parameter\_set\_id の特定の値をもつ) が今アクティブ SVC シーケンスパラメータセット RBSP ではなく、それがピクチャパラメータセット RBSP のアクティブ化によって (seq\_parameter\_set\_id のその値を使用して) 参照され、ピクチャパラメータセット RBSP が DQIdMax と等しい DQId をもつ (nal\_unit\_type

が 20 に等しい) スケーラブル拡張 NAL ユニット内の符号化スライスによってアクティブ化される (ピクチャパラメータセット RBSP がアクティブピクチャパラメータセット RBSP になる) 時、それがアクティブ化される。このサブセットシーケンスパラメータセット RBSP は、別の SVC シーケンスパラメータセット RBSP がアクティブ SVC シーケンスパラメータセット RBSP になる時、それが非アクティブ化されるまで、アクティブ SVC シーケンスパラメータセット RBSP と呼ばれる。そのような seq\_parameter\_set\_id の特定の値をもつサブセットシーケンスパラメータセット RBSP は、そのアクティブ化の前に復号処理で利用可能にならねばならない。

seq\_parameter\_set\_id の特定の値を持つ(nal\_unit\_type が 15 に等しい)サブセットシーケンスパラメータセット RBSP が今アクティブ SVC シーケンスパラメータセット RBSP ではなく、スケーラブルネスティング SEI メッセージの中に含まれ、 $((DQIdMax >> 4) << 4)$  から  $((DQIdMax >> 4) << 4) + 15$  のそれぞれの値を含む範囲内の DQId の値に関連付けられるアクセスユニット内で復号順序で最後のバッファリング期間 SEI メッセージを表現するバッファリング期間 SEI メッセージを含む SEI NAL ユニットによって(seq\_parameter\_set\_id のその値を用いて)参照される時、それはアクティブ化される。このサブセットシーケンスパラメータセット RBSP は、別の SVC シーケンスパラメータセット RBSP がアクティブ SVC シーケンスパラメータセット RBSP になって非アクティブ化されるまで、アクティブ SVC シーケンスパラメータセット RBSP と呼ばれる。そのような seq\_parameter\_set\_id の特定の値を持つサブセットシーケンスパラメータセット RBSP は、そのアクティブ化の前に復号処理にとって利用可能にならねばならない。

記2 アクティブ SVC シーケンスパラメータセット RBSP は、シーケンスパラメータセット RBSP、あるいはサブセットシーケンスパラメータセット RBSP のどちらかである。nal\_unit\_type が 1 または 5 に等しい符号化スライス NAL ユニットによって、あるいはスケーラブルネスティング SEI メッセージの中に含まれないバッファリング期間 SEI メッセージによって、シーケンスパラメータセット RBSP はアクティブ化される。スケーラブル拡張 NAL ユニット(nal\_unit\_type が 20 に等しい)の中での符号化スライスによって、あるいはスケーラブルネスティング SEI メッセージの中に含まれるバッファリング期間 SEI メッセージによって、サブセットシーケンスパラメータセットはアクティブ化される。シーケンスパラメータセット RBSP とサブセットシーケンスパラメータセット RBSP とは、seq\_parameter\_set\_id が同一値を持つかもしれない。

以下の規定に対して、アクティブ化するバッファリング期間 SEI メッセージは次の通り規定される。

- もし DQIdMax が 16 より小さく、かつアクセスユニットが 1 つあるいはそれ以上のバッファリング期間 SEI メッセージを含み、しかもこれらのバッファリング期間 SEI メッセージの復号順序で最後のものがスケーラブルネスティング SEI メッセージの中に含まれないならば、この最後のバッファリング期間 SEI メッセージがアクティブ化するバッファリング期間 SEI メッセージである。
- それ以外、もしアクセスユニットが 1 つあるいはそれ以上の、スケーラブルネスティング SEI メッセージの中に含まれ、かつ  $((DQIdMax >> 4) << 4)$  から  $((DQIdMax >> 4) << 4) + 15$  のそれぞれの値を含む範囲内の DQId の値に関連付けられるバッファリング期間 SEI メッセージを含むならば、これらのバッファリング期間 SEI メッセージの復号順序で最後のものがアクティブ化するバッファリング期間 SEI メッセージである。
- それ以外、そのアクセスユニットはアクティブ化するバッファリング期間 SEI メッセージを含んでいない。

SVC シーケンスパラメータセット RBSP が、特定アクセスユニット内のピクチャパラメータセット RBSP のアクティブ化により参照され、かつこのピクチャパラメータセット RBSP は DQIdMax に等しい DQId を持つ符号化スライス NAL ユニットによりアクティブ化され (このピクチャパラメータセット RBSP がアクティブピクチャパラメータセット RBSP になる)、かつこの特定アクセスユニットはまた、ピクチャパラメータセット RBSP のアクティブ化により参照される SVC シーケンスパラメータセット RBSP とは異なる SVC

シーケンスパラメータセット RBSP を参照する、アクティブ化するバッファリング期間 SEI メッセージを含む時、アクティブ化するバッファリング期間 SEI メッセージにより参照される SVC シーケンスパラメータセット RBSP が、アクティブ SVC シーケンスパラメータセットとなる。

記3 バッファリング期間SEIメッセージは、SVCシーケンスパラメータセットのアクティブ化に対し、符号化スライスNALユニットより高い優先度を持つ。

記4 付属資料Aに規定されるプロファイル規定では、アクティブ化されたシーケンスパラメータセット RBSPは、符号化ビデオシーケンス全体でアクティブであり続けなければならないが、それに比べて本付属資料に規定されるプロファイル規定では異なる。SVCシーケンスパラメータセットRBSPが今アクティブである(アクティブSVCシーケンスパラメータセットRBSPとして)時、別のSVCシーケンスパラメータセットRBSPが、それがアクティブ化するバッファリング期間SEIメッセージ、またはピクチャパラメータセットSBSPのアクティブ化(アクティブピクチャパラメータセットRBSPとして)により参照される時、非IDRアクセスユニット中のアクティブSVCシーケンスパラメータセットRBSPとなる。この場合、非アクティブ化されたのとアクティブ化されたSVCシーケンスパラメータセットRBSPの内容は、相互に以下に示す制約を受ける。そのため、符号化ビデオシーケンスの中で、複数の連続したアクティブ化/非アクティブ化されたSVCシーケンスパラメータセットRBSPが存在可能である。

特定の seq\_parameter\_set\_id の値を持つシーケンスパラメータセット RBSP(nal\_unit\_type が7に等しい)が今、ある DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP でなく、かつそれがピクチャパラメータセット RBSP のアクティブ化により(seq\_parameter\_set\_id のその値を使用して)参照され、かつそのピクチャパラメータセット RBSP は DQId が DQIdMax より小さい符号化スライス NAL ユニットによりアクティブ化され(そのピクチャパラメータセット RBSP がその DQId の特定値に対するアクティブレイヤピクチャパラメータセット RBSP となる)、かつ nal\_unit\_type が 1 または 5 に等しい時、それは DQId の特定値についてのレイヤ表現に対して、アクティブ化される。このシーケンスパラメータセット RBSP は、別の SVC シーケンスパラメータセット RBSP が DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP になるか、あるいは DQIdMax が DQId の特定値より小さいか等しいアクセスユニットを復号して非アクティブ化されるまで、DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP と呼ばれる。そのような seq\_parameter\_set\_id の特定値を持つシーケンスパラメータセット RBSP は、そのアクティブ化の前に復号処理にとって利用可能にならねばならない。

特定の seq\_parameter\_set\_id の値を持つサブセットシーケンスパラメータセット RBSP(nal\_unit\_type が 15 に等しい)が今、ある DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP でなく、かつそれがピクチャパラメータセット RBSP のアクティブ化により(seq\_parameter\_set\_id のその値を使用して)参照され、かつそのピクチャパラメータセット RBSP は DQId が DQIdMax より小さい符号化スライス NAL ユニットによりアクティブ化され(そのピクチャパラメータセット RBSP がその DQId の特定値に対するアクティブレイヤピクチャパラメータセット RBSP となる)、かつ nal\_unit\_type が 20 に等しい時、それは DQId の特定値についてのレイヤ表現に対して、アクティブ化される。このサブセットシーケンスパラメータセット RBSP は、別の SVC シーケンスパラメータセット RBSP が DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP になるか、あるいは DQIdMax が DQId の特定値より小さいか等しいアクセスユニットを復号して非アクティブ化されるまで、DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP と呼ばれる。そのような seq\_parameter\_set\_id の特定値を持つ、サブセットシーケンスパラメータセット RBSP は、そのアクティブ化の前に復号処理にとって利用可能にならねばならない。

付属資料 A においても付属資料 G においても規定されない profile\_idc の値を含むシーケンスパラメータセット RBSP またはサブセットシーケンスパラメータセット RBSP は、(seq\_parameter\_set\_id のその値を用いた)アクティブピクチャパラメータセット RBSP としてまたはアクティブレイヤピクチャパラメータセット RBSP としてのピクチャパラメータセット RBSP のアクティブ化により参照されてはならず、あるいは (seq\_parameter\_set\_id のその値を用いた)バッファリング期間 SEI メッセージを含む SEI NAL ユニットにより

参照されてはならない。付属資料 A においても付属資料 G においても規定されない `profile_idc` の値を含むシーケンスパラメータセット RBSP またはサブセットシーケンスパラメータセット RBSP は、付属資料 A あるいは付属資料 G において規定されるプロファイルに対する復号においては無視される。

非 IDR アクセスユニット内でアクティブ SVC シーケンスパラメータセットとなる SVC シーケンスパラメータセット RBSP (SPSB) は、以下で規定される通り、それが非アクティブ化する SVC シーケンスパラメータセット RBSP (SPSA) と比較して、その内容に関して制限される。

- SPSA と SPSB の `seq_parameter_set_data()` シンタックス構造中のシンタックス要素の値は、シンタックス要素 `profile_idc`、`constraint_setX_flag` (ここで X は 0 から 3、それぞれの値を含む、に等しい)、`level_idc`、`seq_parameter_set_id`、および `hrd_parameters()` シンタックス構造だけが異なっても良く、それ以外は同じでなければならない。
- `seq_parameter_set_svc_extension()` シンタックス構造が SPSA と SPSB の両方の中に存在する時、`seq_parameter_set_svc_extension()` シンタックス構造内の全てのシンタックス要素の値は同じでなければならない。

DQId の各特定値に対し、符号化ビデオシーケンス内の任意のアクセスユニット内のその特定値の DQId に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP である SVC シーケンスパラメータセット RBSP は、その内容に関して制限される。これらの (特定 DQId 値に対する) SVC シーケンスパラメータセット RBSP のシンタックス要素の値は、シンタックス要素 `profile_idc`、`constraint_setX_flag` (X は 0 から 3 までのそれぞれの値を含む)、`level_idc`、および `hrd_parameters()` シンタックス構造についてのみ異なっても良く、それ以外は同じでなければならない。

異なる DQId 値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP が、同じ SVC シーケンスパラメータセット RBSP であっても良い。アクティブ SVC シーケンスパラメータセット RBSP と DQId の特定値に対するアクティブレイヤ SVC シーケンスパラメータセット RBSP が、同じ SVC シーケンスパラメータセット RBSP であっても良い。

アクティブ SVC シーケンスパラメータセット RBSP がシーケンスパラメータセット RBSP である場合には、それが符号化ビデオシーケンスの最後のアクセスユニットに続きもうひとつの符号化ビデオシーケンスの最初の VCL NAL ユニットと最初のバッファリング期間 SEI メッセージを含む SEI NAL ユニット (存在する時) に先行するものでない限り、アクティブ SVC シーケンスパラメータセット RBSP に対する `seq_parameter_set_id` を持つ全てのシーケンスパラメータセット RBSP の内容はアクティブ SVC シーケンスパラメータセット RBSP のものと同じでなければならない。

アクティブ SVC シーケンスパラメータセット RBSP がサブセットシーケンスパラメータセット RBSP である場合には、それが符号化ビデオシーケンスの最後のアクセスユニットに続きもうひとつの符号化ビデオシーケンスの最初の VCL NAL ユニットと最初のバッファリング期間 SEI メッセージを含む SEI NAL ユニット (存在する時) に先行するものでない限り、アクティブ SVC シーケンスパラメータセット RBSP に対する `seq_parameter_set_id` を持つ全てのサブセットシーケンスパラメータセット RBSP の内容はアクティブ SVC シーケンスパラメータセット RBSP のものと同じでなければならない。

個々の DQId 値に対し、以下が適用される。

- アクティブレイヤ SVC シーケンスパラメータセット RBSP がシーケンスパラメータセット RBSP である場合には、それが符号化ビデオシーケンスの最後のアクセスユニットに続きもうひとつの符号化ビデオシーケンスの最初の VCL NAL ユニットに先行するものでない限り、アクティブレイヤ SVC シーケンス

パラメータセット RBSP に対する seq\_parameter\_set\_id を持つ全てのシーケンスパラメータセット RBSP の内容はアクティブレイヤ SVC シーケンスパラメータセット RBSP のものと同じでなければならない。

- アクティブレイヤ SVC シーケンスパラメータセット RBSP がサブセットシーケンスパラメータセット RBSP である場合には、それが符号化ビデオシーケンスの最後のアクセスユニットに続きもうひとつの符号化ビデオシーケンスの最初の VCL NAL ユニットに先行するものでない限り、アクティブレイヤ SVC シーケンスパラメータセット RBSP に対する seq\_parameter\_set\_id を持つ全てのサブセットシーケンスパラメータセット RBSP の内容はアクティブレイヤ SVC シーケンスパラメータセット RBSP のものと同じでなければならない。

記5 もしピクチャパラメータセット RBSP あるいはシーケンスパラメータセット RBSP をビットストリーム内で伝達するならば、これらの制約は、それぞれ、ピクチャパラメータセット RBSP あるいはシーケンスパラメータセット RBSP を含む NAL ユニットについて順序の制約を課す。それ以外（ピクチャパラメータセット RBSP あるいはシーケンスパラメータセット RBSP を本標準では規定されない他の手段で伝達する場合）、それらは、これらの制約に従えるよう復号処理にとって適時の方法で利用可能でなければならない。

存在する場合、シーケンスパラメータセット拡張 RBSP は、シーケンスパラメータセット RBSP のものと類似の機能を持つパラメータを備える。シーケンスパラメータセット拡張 RBSP のシンタックス要素に制約を与える目的、また、各シーケンスパラメータセット拡張 RBSP のアクティブ化を決定する目的において、シーケンスパラメータセット拡張 RBSP は、それに先行する同じ seq\_parameter\_set\_id を持つシーケンスパラメータセット RBSP の一部とみなされなければならない。シーケンスパラメータセット RBSP のアクティブ化以前に、同じ seq\_parameter\_set\_id を持つシーケンスパラメータセット拡張 RBSP が続かないようなシーケンスパラメータセット RBSP が存在する時、アクティブシーケンスパラメータセット RBSP に対しては、シーケンスパラメータセット拡張 RBSP およびそのシンタックス要素は存在しないとみなされなければならない。付属資料 G に規定されたプロファイルの 1 つ以上に適合する符号化ビデオシーケンスの、付属資料 A に規定されたプロファイルの 1 つ以上に適合するベースレイヤが復号される時のみ、シーケンスパラメータセット拡張 RBSP の内容が適用される。サブセットシーケンスパラメータセット RBSP にシーケンスパラメータセット拡張 RBSP が続いてはならない。

記6 シーケンスパラメータセット拡張 RBSP は、サブセットシーケンスパラメータセット RBSP の一部とはみなされず、サブセットシーケンスパラメータセット RBSP にシーケンスパラメータセット拡張 RBSP が続いてはならない。

DQIdMax に等しい DQId を持つレイヤ表現において、SVC シーケンスパラメータセットおよびピクチャパラメータセットのシンタックス要素の値(およびそれらのシンタックス要素から導出された変数の値)と他のシンタックス要素との関係について記述された全ての制約は、アクティブ SVC シーケンスパラメータセットおよびアクティブピクチャパラメータセットのみに適用される制約の記述である。DQIdMax より小さい DQId を持つレイヤ表現において、SVC シーケンスパラメータセットおよびピクチャパラメータセットのシンタックス要素の値(およびそれらのシンタックス要素から導出された変数の値)と他のシンタックス要素との関係について記述された全ての制約は、その特定値の DQId に対するアクティブレイヤ SVC シーケンスパラメータセットおよびアクティブレイヤピクチャパラメータセットのみに適用される制約の記述である。もしビットストリーム中において一度もアクティブ化されない（つまり、一度もアクティブ SVC シーケンスパラメータセットまたはアクティブレイヤ SVC シーケンスパラメータセットにならない）、付属資料 A または付属資料 G に規定される profile\_idc 値の 1 つに等しい profile\_idc を持つ任意の SVC シーケンスパラメータセット RBSP が存在するならば、そのシンタックス要素は、別の適合ビットストリーム中の参照によりアクティブ化されることがあったとしたとき、規定される制約に適合するであろう値を持たなければならない。もしビットストリーム中において一度もアクティブ化されない（つまり、一度もアクティブピクチャパラメータセットまたはアクティブレイヤピクチャパラメータセットにならない）任意のピクチャパラメータセッ

ト RBSP が存在するならば、そのシンタックス要素は、別の適合ビットストリーム中の参照によりアクティブ化されることがあったとしたとき、規定される制約に適合するであろう値を持たなければならない。

復号処理(G.8 節参照)の操作の間、DQIdMax に等しい DQId を持つレイヤ表現においては、アクティブピクチャパラメータセットとアクティブ SVC シーケンスパラメータセットのパラメータ値が効力を持つとみなされなければならない。DQIdMax より小さい DQId の特定値を持つレイヤ表現においては、その DQId の特定値に対するアクティブレイヤピクチャパラメータセットとアクティブレイヤ SVC シーケンスパラメータセットのパラメータ値が効力を持つとみなされなければならない。アクセスユニットあるいは DependencyIdMax に等しい dependency\_id を持つ依存表現または DQIdMax に等しい DQId を持つレイヤ表現に適用される SEI メッセージの解釈においては、それぞれ、同じアクセスユニットの主符号化ピクチャの VCL NAL ユニットの復号処理の操作に対するアクティブピクチャパラメータセットおよびアクティブ SVC シーケンスパラメータセットである、ピクチャパラメータセットおよび SVC シーケンスパラメータセットのパラメータ値が、SEI メッセージの意味において別に規定されない限り、効力を持つとみなされなければならない。DependencyIdMax より小さい dependency\_id の特定値を持つ依存表現に適用される SEI メッセージの解釈においては、それぞれ、同じアクセスユニットの(dependency\_id<<4)に等しい DQId を持つレイヤ表現の VCL NAL ユニットの復号処理の操作に対するアクティブレイヤピクチャパラメータセットおよびアクティブレイヤ SVC シーケンスパラメータセットである、ピクチャパラメータセットおよび SVC シーケンスパラメータセットのパラメータ値が、SEI メッセージの意味において別に規定されない限り、効力を持つとみなされなければならない。DQIdMax より小さい DQId の特定値を持つレイヤ表現に適用される SEI メッセージの解釈においては、同じアクセスユニットのその DQId の特定値を持つレイヤ表現の VCL NAL ユニットの復号処理の操作に対するアクティブレイヤピクチャパラメータセットおよびアクティブレイヤ SVC シーケンスパラメータセットである、ピクチャパラメータセットおよび SVC シーケンスパラメータセットのパラメータ値が、SEI メッセージの意味において別に規定されない限り、効力を持つとみなされなければならない。

#### G.7.4.1.2.2 アクセスユニット順序および符号化ビデオシーケンスへの関連

7.4.1.2.2 小小小節の規定が次の修正とともに適用される。

ビットストリームの最初のアクセスユニットは 5 に等しい nal\_unit\_type か 1 に等しい idr\_flag を持つ符号化スライス NAL ユニットののみを含まなければならない。

NAL ユニットの符号化ピクチャの順序、およびそれらのアクセスユニットへの関連付けは、G.7.4.1.2.3 小小小節において記述される。

#### G.7.4.1.2.3 NALユニットおよび符号化ピクチャの順序およびアクセスユニットへの関連

7.4.1.2.3 小小小節の規定が次の修正とともに適用される。

主あるいは冗長符号化ピクチャへの VCL NAL ユニットの関連は、G.7.4.1.2.5 小小小節で規定される。

主符号化ピクチャの最初の VCL NAL ユニットの検出に対する制約は、G.7.4.1.2.4 小小小節において規定される。

バッファリング期間 SEI の順序において 7.4.1.2.3 小小小節で表現される制約は、次の制約によって置き換えられる。

- バッファリング期間 SEI メッセージを含む SEI NAL ユニットの存在する場合、次が適用される。

- もしバッファリング期間 SEI メッセージがアクセスユニットの中で唯一のバッファリング期間 SEI メッセージであり、それがスケーラブルネスティング SEI メッセージに含まれていないならば、バッファリング期間 SEI メッセージはアクセスユニットの中で最初の SEI NAL ユニットの最初の SEI メッセージペイロードでなければならない。
- それ以外(バッファリング期間 SEI メッセージがアクセスユニットの中で唯一のバッファリング期間 SEI メッセージではない、またはスケーラブルネスティング SEI メッセージに含まれる)、次が適用される。
  - スケーラブルネスティング SEI メッセージの中に含まれていないバッファリング期間 SEI メッセージが存在する時、このバッファリング期間 SEI メッセージはアクセスユニットの中の最初の SEI NAL ユニットの唯一の SEI メッセージペイロードでなければならない。
  - バッファリング期間 SEI メッセージを含むスケーラブルネスティング SEI メッセージは、他のいかなる SEI メッセージも含んではならず、またバッファリング期間 SEI を含むスケーラブルネスティング SEI メッセージは、SEI NAL ユニット内の唯一の SEI メッセージでなければならない。
  - アクセスユニット内のペイロードとしてバッファリング期間 SEI メッセージを含むスケーラブルネスティング SEI メッセージを含む SEI NAL ユニットの先行する全ての SEI NAL ユニットの、バッファリング期間 SEI メッセージか、またはバッファリング期間 SEI メッセージをペイロードとして持つスケーラブルネスティング SEI メッセージのみを含まなければならない。
  - もし存在する時、1 に等しい `all_layer_representations_in_au_flag` およびバッファリング期間 SEI メッセージをペイロードとして持つスケーラブルネスティング SEI メッセージは、アクセスユニットの中で最初のスケーラブルネスティング SEI メッセージでなければならない。
  - `sei_dependency_id[i]` と `sei_quality_id[i]` の特定値およびバッファリング期間 SEI メッセージをペイロードとしてもつ特定のスケーラブルネスティング SEI メッセージに先行する、任意のスケーラブルネスティング SEI メッセージは、全ての存在する `i` に対し、その特定のバッファリング期間 SEI メッセージをペイロードとしてもつスケーラブルネスティング SEI メッセージに対する任意の特定値  $(sei\_dependency\_id[i] \ll 4) + sei\_quality\_id[i]$  より小である値  $(sei\_dependency\_id[i] \ll 4) + sei\_quality\_id[i]$  を持たなければならない。

次の追加制約に従わなければならない。

- 1、5 もしくは 12 に等しい `nal_unit_type` を持つ各 NAL ユニットの、プリフィックス NAL ユニットによって直ちに先行されなければならない。
- 各プリフィックス NAL ユニットの、1、5 もしくは 12 に等しい `nal_unit_type` を持つ NAL ユニットによって直ちに後続されなければならない。

#### G.7.4.1.2.4 主符号化ピクチャの最初の VCL NAL ユニットの検出

この小節は、各々の主符号化ピクチャの最初の VCL NAL ユニットの検出を可能にするのに十分である VCL NAL ユニットシンタックスの制約を規定する。

復号順序で現アクセスユニットの主符号化ピクチャの最初の VCL NAL ユニットの、次の方法の 1 つもしくはそれ以上で復号順序で直前のアクセスユニットの主符号化ピクチャの最後の VCL NAL ユニットの異なる

らなければならない。

- 現アクセスユニットの主符号化ピクチャの最初の VCL NAL ユニットの `dependency_id` は、直前のアクセスユニットの主符号化ピクチャの最後の VCL NAL ユニットの `dependency_id` より小さい。
- 現アクセスユニットの主符号化ピクチャの最初の VCL NAL ユニットの `dependency_id` は、直前のアクセスユニットの主符号化ピクチャの最後の VCL NAL ユニットの `dependency_id` に等しく、かつ次の条件のいずれかが真である。
  - 現アクセスユニットの主符号化ピクチャの最初の VCL NAL ユニットの `quality_id` は、直前のアクセスユニットの主符号化ピクチャの VCL NAL ユニットの `quality_id` より小さい。
  - 現アクセスユニットの主符号化ピクチャの最初の VCL NAL ユニットと直前のアクセスユニットの主符号化ピクチャの最後の VCL NAL ユニットの `quality_id` は 0 に等しく、かつ 7.4.1.2.4 小小小節で規定されるいずれかの状態を満たす。

#### G.7.4.1.2.5 VCL NALユニットの順序および符号化ピクチャへの関連

それぞれの VCL NAL ユニットは、符号化ピクチャの一部である。

`dId` は `dependency_id` の値とし、`qId` は任意の特定の VCL NAL ユニットの `quality_id` の値とする。符号化ピクチャ内の VCL NAL ユニットの順序は、次の通り制約される。

- この特定の VCL NAL ユニットに続く全ての VCL NAL ユニットにおいて、`dependency_id` の値は `dId` より大きいとか等しくなければならない。
- この特定の VCL NAL ユニットに続く `dId` に等しい `dependency_id` の値を持つ全ての VCL NAL ユニットにおいて、`quality_id` の値は `qId` より大きいとか等しくなければならない。

レイヤ表現内の各 VCL NAL ユニット集合において、次が適用される。

- もし付属資料 A もしくは G.10 節で規定された任意スライス順序が許可されるなら、レイヤ表現の符号化スライス NAL ユニットは、互いに対して任意の順序を持ってもよい。
- それ以外(任意スライス順序は許可されない)、スライスグループの符号化スライス NAL ユニットは別のスライスグループの符号化スライス NAL ユニットにインターリーブされてはならず、スライスグループ中の符号化スライス NAL ユニットの順序は、同じスライスグループのそれぞれの符号化スライス NAL ユニットの最初のマクロブロックアドレスが、昇順のマクロブロックアドレスにならなければならない。

12 に等しい `nal_unit_type` を持つ NAL ユニットは、アクセスユニット内に存在しても良いが、アクセスユニット内の主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

0、または 24 から 31 のそれぞれの値を含む範囲に等しい `nal_unit_type` を持つ NAL ユニットは、無規定であり、アクセスユニット内に存在しても良いが、アクセスユニット内の主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

21 から 23 のそれぞれの値を含む範囲の `nal_unit_type` を持つ NAL ユニットは、予約されており、(将来 TTC により規定される時)アクセスユニット内の主符号化ピクチャの最初の VCL NAL ユニットに先行してはならない。

## G.7.4.2 未加工のバイトシーケンスペイロードおよびRBSP追従ビットの意味

### G.7.4.2.1 シーケンスパラメータセットRBSPの意味

7.4.2.1 小小節で規定された意味が適用される。

#### G.7.4.2.1.1 シーケンスパラメータセットデータの意味

シーケンスパラメータセットを SVC シーケンスパラメータセットに置き換えて 7.4.2.1.1 小小小節で規定される意味が適用される。7.4.2.1.1 小小小節で規定される全ての制約は、G.7.4.1.2.1 小小小小節で規定される SVC シーケンスパラメータセットがアクティブシーケンスパラメータセットであるか、またはアクティブレイヤ SVC シーケンスパラメータセットであるレイヤ表現にのみ適用される。加えて、次が適用される。

**constraint\_set0\_flag** は次のように規定する。

- もしシーケンスパラメータセットのデータシンタックス構造がシーケンスパラメータセット RBSP に含まれているならば、7.4.2.1.1 小小小節で規定される意味を適用する。
- それ以外（シーケンスパラメータセットのデータシンタックス構造がサブセットシーケンスパラメータセット RBSP に含まれる）、**constraint\_set0\_flag** は符号化ビデオシーケンスが G.10.1.1 小小節で規定される全ての制約に従うことを規定する。0 に等しい **constraint\_set0\_flag** は符号化ビデオシーケンスが G.10.1.1 小小節で規定される全ての制約に従ってもよいし、従わなくてもよいことを規定する。

**constraint\_set1\_flag** は次のように規定する。

- もしシーケンスパラメータセットのデータシンタックス構造がシーケンスパラメータセット RBSP に含まれているならば、7.4.2.1.1 小小小節で規定される意味を適用する。
- それ以外（シーケンスパラメータセットのデータシンタックス構造がサブセットシーケンスパラメータセット RBSP に含まれる）、**constraint\_set1\_flag** は符号化ビデオシーケンスが G.10.1.2 小小節で規定される全ての制約に従うことを規定する。0 に等しい **constraint\_set1\_flag** は符号化ビデオシーケンスが G.10.1.2 小小節で規定される全ての制約に従ってもよいし、従わなくてもよいことを規定する。

**constraint\_set2\_flag** は次のように規定する。

- もしシーケンスパラメータセットのデータシンタックス構造がシーケンスパラメータセット RBSP に含まれているならば、7.4.2.1.1 小小小節で規定される意味を適用する。
- それ以外（シーケンスパラメータセットのデータシンタックス構造がサブセットシーケンスパラメータセット RBSP に含まれる）、**constraint\_set2\_flag** における 1 の値は、TTC によって将来の使用が予約されている。**constraint\_set2\_flag** は本標準に適合するビットストリームの中で、83 もしくは 86 に等しい **profile\_idc** を持つ符号化ビデオシーケンスにおいて 0 に等しくなければならない。本標準に適合する復号器は、**profile\_idc** が 83 もしくは 86 に等しい時、**constraint\_set2\_flag** の値を無視しなければならない。

**constraint\_set3\_flag** は次のように規定する。

- もしシーケンスパラメータセットのデータシンタックス構造がシーケンスパラメータセット RBSP に含まれているならば、7.4.2.1.1 小小小節で規定された意味を適用する。
- それ以外（シーケンスパラメータセットのデータシンタックス構造がサブセットシーケンスパラメータセット RBSP に含まれる）、次が適用される。

- もし `profile_idc` が 86 に等しいならば、1 に等しい `constraint_set3_flag` は符号化ビデオシーケンスは G.10.1.3 小小節で規定される全ての制約に従うことを規定しており、0 に等しい `constraint_set3_flag` は符号化ビデオシーケンスがこれらの対応する制約に従ってもよいし、従わなくてもよいことを規定する。
- それ以外 (`profile_idc` が 86 に等しくない)、`constraint_set3_flag` において 1 の値は TTC によって将来の使用が予約される。`constraint_set3_flag` は本標準に適合するビットストリームの中で 86 に等しい `profile_idc` を持つ符号化ビデオシーケンスにおいて 0 に等しくなければならない。本標準に適合する復号器は `profile_idc` が 86 に等しい時、`constraint_set3_flag` の値を無視しなければならない。

`separate_colour_plane_flag` の値は 0 に等しくなければならない、`qprime_y_zero_transform_bypass_flag` は 0 に等しくなければならない。

`profile_idc` の値が付属資料 A の中で規定されるいずれのプロファイルへの適合も示さず、かつ `vui_parameters_present_flag` が 1 に等しい時、`timing_info_present_flag` は 0 に等しくなければならない、`nal_hrd_parameters_present_flag` は 0 に等しくなければならない、`vcl_hrd_parameters_present_flag` は 0 に等しくなければならない、`pic_struct_present_flag` は 0 に等しくなければならない。

`profile_idc` の値が付属資料 A で規定された 1 つ以上のプロファイルへの適合を示し、かつ `vui_parameters_present_flag` が 1 に等しい時、`timing_info_present_flag`、`num_units_in_tick`、`time_scale`、`fixed_frame_rate_flag`、`nal_hrd_parameters_present_flag`、`vcl_hrd_parameters_present_flag`、`low_delay_hrd_flag`、`pic_struct_present_flag` の値および `hdr_parameters()` シンタックス構造に含まれるシンタックス要素の値は、存在する時、シーケンスパラメータセットをアクティブ化するビットストリームが付属資料 A で規定される 1 つ以上のプロファイルに適合するようなものでなければならない。

#### G.7.4.2.1.1.1 スケーリングリストの意味

7.4.2.1.1.1 小小小節で規定された意味が適用される。

#### G.7.4.2.1.2 シーケンスパラメータセット拡張RBSPの意味

7.4.2.1.2 小小小節で規定された意味が適用される。加えて、以下が適用される。

シーケンスパラメータセット拡張 RBSP は、復号順序でシーケンスパラメータセット RBSP の後にのみ続くことができる。サブセットシーケンスパラメータセット RBSP は、シーケンスパラメータセット拡張 RBSP の後に続いてはならない。シーケンスパラメータセット拡張 RBSP の内容は、付属資料 A で規定された 1 つ以上のプロファイルに適合する、付属資料 G で規定された 1 つ以上のプロファイルに適合する符号化ビデオシーケンスのベースレイヤが復号される場合のみ、適用される。

#### G.7.4.2.1.3 サブセットシーケンスパラメータセットRBSPの意味

`svc_vui_parameters_present_flag` が 0 に等しいことは、シンタックス構造 `svc_vui_parameters_extensions()` が存在しないことを示す。`svc_vui_parameters_present_flag` が 1 に等しいことは、シンタックス構造 `svc_vui_parameters_extensions()` が存在することを示す。

`additional_extension2_flag` は 0 に等しくなければならない。`additional_extension2_flag` における値 1 は、TTC によって将来の使用のために予約される。

`additional_extension2_data_flag` は、任意の値を取り得る。それは、付属資料 A もしくは付属資料 G で規

定されるプロファイルへの適合に影響を与えてはならない。

#### G.7.4.2.1.4 シーケンスパラメータセットSVC拡張の意味

**inter\_layer\_deblocking\_filter\_control\_present\_flag** が 1 に等しいことは、レイヤ間予測に対するデブロックフィルタの特性を制御するシンタックス要素の組が、スライスヘッダ内に存在することを規定する。**inter\_layer\_deblocking\_filter\_control\_present\_flag** が 0 に等しいことは、レイヤ間予測に対するデブロックフィルタの特性を制御するシンタックス要素の組がスライスヘッダ内に存在せず、それらの推定値が有効であることを規定する。

**extended\_spatial\_scalability** は、再サンプリング処理に対する幾何パラメータに関連するシンタックス要素の存在を規定する。**extended\_spatial\_scalability** の値は 0 から 2 のそれぞれを含む範囲内でなければならない。**extended\_spatial\_scalability** が 0 に等しい場合、サブセットシーケンスパラメータセットおよびこのサブセットシーケンスパラメータセットを参照するスライスヘッダ内に、幾何パラメータが存在しない。**extended\_spatial\_scalability** が 1 に等しい場合、幾何パラメータはサブセットシーケンスパラメータセット内に存在するが、このサブセットシーケンスパラメータセットを参照するスライスヘッダ内には存在しない。**extended\_spatial\_scalability** が 2 に等しい場合、幾何パラメータはサブセットシーケンスパラメータセット内に存在しないが、このサブセットシーケンスパラメータセットを参照するスライスヘッダ内には存在する。

**chroma\_phase\_x\_plus1\_flag** は、フレームもしくはレイヤフレームの 2 分の 1 輝度サンプル単位で、色差成分の水平位相シフトを規定する。**chroma\_phase\_x\_plus1\_flag** が存在しない場合、それは値 1 に等しく推定されなければならない。

**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** が存在する場合、以下が適用される。

- もし **chroma\_phase\_x\_plus1\_flag** が 0 に等しいならば、**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** は、0、2、または 4 に等しくなるべきである。
- それ以外 (**chroma\_phase\_x\_plus1\_flag** が 1 に等しい)、**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** は、1、3、または 5 に等しくなるべきである。

**chroma\_phase\_y\_plus1** は、フレームもしくはレイヤフレームの 2 分の 1 輝度サンプル単位で、色差成分の垂直位相シフトを規定する。**chroma\_phase\_y\_plus1** が存在しない場合、それは 1 に等しく推定されなければならない。**chroma\_phase\_y\_plus1** は、0 から 2 のそれぞれを含む範囲内でなければならない。

**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** が存在する場合、以下が適用される。

- もし **chroma\_phase\_y\_plus1** が 0 に等しいならば、**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** は、2、または 3 に等しくなるべきである。
- それ以外、もし **chroma\_phase\_y\_plus1** が 1 に等しいならば、**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** は、0、または 1 に等しくなるべきである。
- それ以外 (**chroma\_phase\_y\_plus1** が 2 に等しい)、**chroma\_sample\_loc\_type\_top\_field** および **chroma\_sample\_loc\_type\_bottom\_field** は、4、または 5 に等しくなるべきである。

**seq\_ref\_layer\_chroma\_phase\_x\_plus1\_flag** は、レイヤ間予測のために使用されるかもしれないレイヤピク

チャに対するレイヤフレームの 2 分の 1 輝度サンプル単位で、色差成分の水平位相シフトを規定する。`seq_ref_layer_chroma_phase_x_plus1_flag` が存在しない場合、それは `chroma_phase_x_plus1_flag` に等しく推定されなければならない。

`seq_ref_layer_chroma_phase_y_plus1` は、レイヤ間予測のために使用されるかもしれないレイヤピクチャに対するレイヤフレームの 2 分の 1 輝度サンプル単位で、色差成分の垂直位相シフトを規定する。`seq_ref_layer_chroma_phase_y_plus1` が存在しない場合、それは `chroma_phase_y_plus1` に等しく推定されなければならない。`seq_ref_layer_chroma_phase_y_plus1` の値は 0 から 2 のそれぞれを含む範囲内でなければならない。

`seq_scaled_ref_layer_left_offset` は、2 輝度サンプル単位で、レイヤ間予測のために使用される再サンプリングされたレイヤピクチャの左上輝度サンプルと、現ピクチャまたは現レイヤピクチャの左上輝度サンプルとの間の水平オフセットを規定する。`seq_scaled_ref_layer_left_offset` が存在しない場合、それは 0 に等しく推定されなければならない。

`seq_scaled_ref_layer_top_offset` は、レイヤ間予測のために使用される再サンプリングされたレイヤピクチャの左上輝度サンプルと、現ピクチャまたは現レイヤピクチャの左上輝度サンプルとの間の垂直オフセットを規定する。`frame_mbs_only_flag` が 1 に等しい場合、垂直オフセットは 2 輝度サンプル単位で規定され、`frame_mbs_only_flag` が 0 に等しい場合、それは 4 輝度サンプル単位で規定される。`seq_scaled_ref_layer_top_offset` が存在しない場合、それは 0 に等しく推定されなければならない。

`seq_scaled_ref_layer_right_offset` は、2 輝度サンプル単位で、レイヤ間予測のために使用される再サンプリングされたレイヤピクチャの右下輝度サンプルと、現ピクチャまたは現レイヤピクチャの右下輝度サンプルとの間の水平オフセットを規定する。`seq_scaled_ref_layer_right_offset` が存在しない場合、それは 0 に等しく推定されなければならない。

`seq_scaled_ref_layer_bottom_offset` は、レイヤ間予測のために使用される再サンプリングされたレイヤピクチャの右下輝度サンプルと、現ピクチャまたは現レイヤピクチャの右下輝度サンプルとの間の垂直オフセットを規定する。`frame_mbs_only_flag` が 1 に等しい場合、垂直オフセットは 2 輝度サンプル単位で規定され、`frame_mbs_only_flag` が 0 に等しい場合、それは 4 輝度サンプル単位で規定される。`seq_scaled_ref_layer_bottom_offset` が存在しない場合、それは 0 に等しく推定されなければならない。

`seq_tcoeff_level_prediction_flag` は、サブセットシーケンスパラメータセット内でのシンタックス要素 `adaptive_tcoeff_level_prediction_flag` の存在を規定する。

`adaptive_tcoeff_level_prediction_flag` は、サブセットシーケンスパラメータセットを参照するスライスヘッダ内での、`tcoeff_level_prediction_flag` の存在を規定する。`adaptive_tcoeff_level_prediction_flag` が存在しない場合、それは 0 に等しく推定されなければならない。

`slice_header_restriction_flag` は、サブセットシーケンスパラメータセットを参照するスライスヘッダ内での、シンタックス要素の存在を規定する。

#### G.7.4.2.2 ピクチャパラメータセットRBSPの意味

7.4.2.2 小小節で規定される意味が、シーケンスパラメータセットを SVC シーケンスパラメータセットに置き換えて、適用される。7.4.2.2 小小節で規定される全ての制約は、ピクチャパラメータセットが、G.7.4.1.2.1 小小小小節で規定されるアクティブピクチャパラメータセットあるいはアクティブレイヤピクチャパラメータセットであるレイヤ表現のみに適用される。

#### G.7.4.2.3 付加拡張情報RBSPの意味

7.4.2.3 小小節で規定される意味が適用される。

##### G.7.4.2.3.1 付加拡張情報メッセージの意味

7.4.2.3.1 小小節で規定される意味が適用される。

#### G.7.4.2.4 アクセスユニット境界RBSPの意味

7.4.2.4 小小節で規定される意味が適用される。

#### G.7.4.2.5 シーケンス終了RBSPの意味

7.4.2.5 小小節で規定される意味が適用される。

#### G.7.4.2.6 ストリーム終了RBSPの意味

7.4.2.6 小小節で規定される意味が適用される。

#### G.7.4.2.7 フィルデータRBSPの意味

7.4.2.7 小小節で規定される意味が適用される。

#### G.7.4.2.8 パーティションのないスライスレイヤRBSPの意味

7.4.2.8 小小節で規定される意味が適用される。

#### G.7.4.2.9 スライスデータパーティションRBSPの意味

スライスデータパーティションシンタックスは、付属資料 G で規定されるプロファイルの 1 つ以上に適合するビットストリーム内には存在しない。

#### G.7.4.2.10 RBSPスライス追従ビットの意味

7.4.2.10 小小節で規定される意味が適用される。

#### G.7.4.2.11 RBSP追従ビットの意味

7.4.2.11 小小節で規定される意味が適用される。

#### G.7.4.2.12 プリフィックスNALユニットRBSPの意味

**store\_ref\_base\_pic\_flag** が 1 に等しいことは、NAL ユニットヘッダで規定される **dependency\_id** の値が現アクセスユニットの VCL NAL ユニットに対する **dependency\_id** の最大値に等しい場合、復号ピクチャと同一であってもなくてもよい符号化ピクチャの追加の表現が“参照使用”としてマークされることを規定する。この追加の表現は参照ベースピクチャとも呼ばれ、復号順序で後続のピクチャの INTER 予測のために使用されるかも知れないが、それは出力されない。**store\_ref\_base\_pic\_flag** が存在しない場合、それは 0 に等しく推定されなければならない。

シンタックス要素 **store\_ref\_base\_pic\_flag** は、依存表現の全ての VCL NAL ユニットに対して同一の値を持たなければならない。**nal\_ref\_idc** が 0 の場合、**store\_ref\_base\_pic\_flag** の値は 0 に等しくなければならない。

store\_ref\_base\_pic\_flag は、関連付けられた NAL ユニットに適用されるとみなされる。

**prefix\_nal\_unit\_additional\_extension\_flag** は、0 に等しくなければならない。  
prefix\_nal\_unit\_additional\_extension\_flag の値 1 は、TTC によって将来の使用のために予約される。

**prefix\_nal\_unit\_extension\_flag** は、任意の値を持ってよい。

記 シンタックス要素 prefix\_nal\_unit\_additional\_extension\_flag および prefix\_nal\_unit\_extension\_flag は、本標準で規定される復号処理によって要求はされない。

#### G.7.4.2.13 スケーラブル拡張内スライスレイヤRBSPの意味

スケーラブル拡張内スライスレイヤ RBSP は、スケーラブル拡張内スライスヘッダおよびスケーラブル拡張内スライスデータを含む。

#### G.7.4.3 スライスヘッダの意味

7.4.3 小節で規定される意味が、以下の変更を伴って適用される。

7.4.3 小節で規定される全ての制約が、0 に等しい DQId を持つレイヤ表現のみに適用される。

**frame\_num** は、G.7.4.3.4 小小節で規定されるものと同じ意味を持つ。

slice\_header() シンタックス構造は、以下の推定値をもつ以下のシンタックス要素を含むとみなされなければならない。

- ref\_layer\_dq\_id は、-1 に等しく推定される。
- scan\_idx\_start は、0 に等しく推定される。
- scan\_idx\_end は、15 に等しく推定される。

direct\_spatial\_mv\_pred\_flag の値は、1 に等しくなければならない。

変数 SpatialResolutionChangeFlag は、0 に等しく設定される。

#### G.7.4.3.1 参照ピクチャリスト並替の意味

7.4.3.1 小小節で規定される意味が適用される。

#### G.7.4.3.2 予測重み表の意味

7.4.3.2 小小節で規定される意味が適用される。

#### G.7.4.3.3 復号参照ピクチャマーキングの意味

7.4.3.3 小小節で規定される意味がシーケンスパラメータセットを SVC シーケンスパラメータセットに置き換えて適用される。7.4.3.3 小小節で規定される制約は dependency\_id の現在の値に等しい dependency\_id による依存表現に対してのみ適用され、さらに G.8.2 小節で規定される修正 a) および b) が dependency\_id の現在の値に等しい currDependencyId を用いて適用される。

#### G.7.4.3.4 スケーラブル拡張内スライスヘッダの意味

SVC シーケンスパラメータセットのシンタックス要素 bit\_depth\_luma\_minus8、bit\_depth\_chroma\_minus8 お

よび `chroma_format_idc` の値は、アクセスユニットの全ての符号化スライス NAL ユニットの渡って同じでなければならない。

スライスヘッダのシンタックス要素 `field_pic_flag` および `bottom_field_flag` が存在する場合、それらの値は、アクセスユニットの全ての符号化スライス NAL ユニットの渡って同じでなければならない。

スライスヘッダのシンタックス要素 `frame_num`、`idr_pic_id`、`pic_order_cnt_lsb`、`delta_pic_order_cnt_bottom`、`delta_pic_order_cnt[0]`、および `delta_pic_order_cnt[1]` が存在する場合、それらの値は 1、2、あるいは 5 に等しい `nal_unit_type` を伴う NAL ユニットのスライスヘッダを含む依存表現の全てのスライスに渡って同じでなければならない。

存在する場合、次のスライスヘッダのシンタックス要素の値はレイヤ表現の全てのスライスに渡って同じでなければならない：`pic_parameter_set_id`、`ref_layer_dq_id`、`disable_inter_layer_deblocking_filter_idc`、`inter_layer_slice_alpha_c0_offset_div2`、`inter_layer_slice_beta_offset_div2`、`constrained_intra_resampling_flag`、`ref_layer_chroma_phase_x_plus1_flag`、`ref_layer_chroma_phase_y_plus1`、`scaled_ref_layer_left_offset`、`scaled_ref_layer_top_offset`、`scaled_ref_layer_right_offset`、`scaled_ref_layer_bottom_offset`、`slice_group_change_cycle`、`store_ref_base_pic_flag`、`tcoeff_level_prediction_flag`、およびシンタックス構造 `dec_ref_pic_marking()` および `dec_ref_base_pic_marking()` の全てのシンタックス要素。

現スライスのマクロブロックによって覆われる、あるいは部分的に覆われる、`dependency_id` の現在の値に等しい `dependency_id` および 0 に等しい `quality_id` を有するスライスの集合を `setOfRefLayerSlices` とする。

`quality_id` が 0 よりも大きい場合、集合 `setOfRefLayerSlices` 内の全てのスライスに対する `(slice_type%5)` の値は現スライスに対する `(slice_type%5)` の値と同じでなければならない。

`quality_id` が 0 よりも大きく `setOfRefLayerSlices` が 1 つ以上のスライスを含む場合、以下の制約に従わなければならない。

- a) `slice_type` が EP あるいは EB スライスを規定する場合、シンタックス要素 `num_ref_idx_l0_active_minus1` の値 (`num_ref_idx_active_override_flag` が 1 に等しい場合はスライスヘッダの中で伝送される値あるいは `num_ref_idx_active_override_flag` が 0 に等しい場合は参照されるピクチャパラメータセットの値のいずれか) は集合 `setOfRefLayerSlices` の全てのスライスに渡って同じでなければならない。
- b) `slice_type` が EB スライスを規定する場合、シンタックス要素 `num_ref_idx_l1_active_minus1` の値 (`num_ref_idx_active_override_flag` が 1 に等しい場合はスライスヘッダの中で伝送される値あるいは `num_ref_idx_active_override_flag` が 0 に等しい場合は参照されるピクチャパラメータセットの値のいずれか) は集合 `setOfRefLayerSlices` の全てのスライスに渡って同じでなければならない。
- c) シンタックス構造 `ref_pic_list_reordering()` の全ての要素は集合 `setOfRefLayerSlices` の全てのスライスに渡って同じでなければならない。
- d) `slice_type` が EP スライスを規定する場合、シンタックス要素 `weighted_pred_flag` は集合 `setOfRefLayerSlices` の全てのスライスに渡って同じでなければならない、さらに以下が適用される。
  - i. 集合 `setOfRefLayerSlices` の任意のスライスに対して `weighted_pred_flag` が 1 に等しく `no_inter_layer_pred_flag` が 0 に等しい場合、`base_pred_weight_table_flag` は集合 `setOfRefLayerSlices` の全てのスライスに渡って同じでなければならない。

- ii. 集合 setRefLayerSlices の任意のスライスに対して weighted\_pred\_flag が 1 に等しく、集合 setRefLayerSlices の任意のスライス中に pred\_weight\_table( ) が存在する場合、pred\_weight\_table( ) は集合 setOfRefLayerSlices の全てのスライス中に存在しなければならず、さらにシンタックス構造 pred\_weight\_table( ) 内部の全てのシンタックス要素は集合 setOfRefLayerSlices の全てのスライスに渡って同じでなければならない。
- e) slice\_type が EB スライスを規定する場合、シンタックス要素 weighted\_bipred\_idc は集合 setOfRefLayerSlices の全てのスライスに渡って同じでなければならない、さらに以下が適用される。
  - i. 集合 setRefLayerSlices の任意のスライスに対して weighted\_bipred\_idc が 1 に等しくかつ no\_inter\_layer\_pred\_flag が 0 に等しい場合、base\_pred\_weight\_table\_flag は集合 setOfRefLayerSlices の全てのスライスに渡って同じでなければならない。
  - ii. 集合 setRefLayerSlices の任意のスライスに対して weighted\_bipred\_idc が 1 に等しく、集合 setRefLayerSlices の任意のスライス中に pred\_weight\_table( ) が存在する場合、pred\_weight\_table( ) は集合 setOfRefLayerSlices の全てのスライス中に存在しなければならず、さらにシンタックス構造 pred\_weight\_table( ) 内部の全てのシンタックス要素は集合 setOfRefLayerSlices の全てのスライスに渡って同じでなければならない。

quality\_id が 0 よりも大きい場合、次の（存在しない）シンタックス要素は、dependency\_id の現在の値に等しい dependency\_id および 0 に等しい quality\_id を有し、マクロブロックアドレス first\_mb\_in\_slice を有するマクロブロックを覆うスライスのスライスヘッダの対応するシンタックス要素に、このスライスに存在する場合、等しいと推定されなければならない：direct\_spatial\_mv\_pred\_flag、num\_ref\_idx\_l0\_active\_minus1、num\_ref\_idx\_l1\_active\_minus1、シンタックス構造 ref\_pic\_list\_reordering( ) の全てのシンタックス要素、base\_pred\_weight\_table\_flag、シンタックス構造 pred\_weight\_table( ) の全てのシンタックス要素、シンタックス構造 dec\_ref\_pic\_marking( ) の全てのシンタックス要素、シンタックス構造 dec\_ref\_base\_pic\_marking( ) の全てのシンタックス要素、slice\_group\_change\_cycle、および store\_ref\_base\_pic\_flag。

first\_mb\_in\_slice は 7.4.3 小節で規定されるのと同じ意味を持つ。

slice\_type は付表 G-1/JT-H264 に従ってスライスの符号化タイプを規定する。

付表G-1/JT-H264 20に等しいnal\_unit\_typeのNALユニットに対する  
slice\_typeに関連付けられる名前  
(ITU-T H.264)

slice_type	Name of slice_type
0, 5	EP (スケーラブル拡張内Pスライス)
1, 6	EB (スケーラブル拡張内Bスライス)
2, 7	EI (スケーラブル拡張内Iスライス)

5 から 7 までのそれぞれの値を含む範囲内の slice\_type の値は現スライスの符号化タイプに加えて、現レイヤ表現の他の全てのスライスが slice\_type の現在の値あるいは slice\_type の現在の値から 5 を減じた値に等しい slice\_type の値を持たなければならないことを規定する。

idr\_flag が 1 に等しいあるいは num\_ref\_frames が 0 に等しい場合、slice\_type は 2 あるいは 7 に等しくなければならない。

**pic\_parameter\_set\_id** は 7.4.3 小節で規定されるのと同じの意味を持つ。**pic\_parameter\_set\_id** を有するピクチャパラメータセットによって (**seq\_parameter\_set\_id** を用いて) 参照される SVC シーケンスパラメータセットのシンタックス要素 **profile\_idc** は 83 あるいは 86 に等しくなければならない。

**frame\_num** は依存表現に対する識別子として用いられ、ビットストリーム内で  $\log_2 \max\_frame\_num - 4$  ビットで表現されなければならない。

**frame\_num** は小節 7.4.3 で規定されるように制約され、ここでこの制約は **dependency\_id** の現在の値に等しい **dependency\_id** を有する依存表現に適用され、さらに G.8.2 小節で規定される修正 a) および b) は **dependency\_id** の現在の値に等しい **currDependencyId** を用いて適用される。

**field\_pic\_flag** および **bottom\_field\_flag** は小節 7.4.3 で規定されるのと同じの意味を持つ。

**idr\_pic\_id** は **dependency\_id** が現アクセスユニットの VCL NAL ユニット中の **dependency\_id** の存在する値の最大に等しい場合 IDR ピクチャを識別する。**idr\_pic\_id** の値は 0 から 65535 までのそれぞれの値を含む範囲内にななければならない。

**dependency\_id** が現アクセスユニットの全ての VCL NAL ユニット中の **dependency\_id** の最大値に等しく、復号順序で連続する 2 つのアクセスユニットがいずれも IDR アクセスユニットである場合、最初のそのような IDR アクセスユニット中の **dependency\_id** の現在の値に等しい **dependency\_id** を有する依存表現のスライス中の **idr\_pic\_id** の値は 2 番目のそのような IDR アクセスユニット中の **dependency\_id** の現在の値に等しい **dependency\_id** を有する依存表現のスライス中の **idr\_pic\_id** と異ならなければならない。

記1 アクセスユニットの IDR アクセスユニットとしての分類は **dependency\_id** の最大存在値に依存する。NAL ユニットがあるビットストリームから除去された場合、例えばビットストリームを受信デバイスの能力に適合させるために、アクセスユニットの IDR アクセスユニットとしての分類は変化しても良い。スケーラブルビットストリーム中のサポートされる異なる適合点に対する (特に **dependency\_id** の異なる最大値に対する) 全てのビットストリームはこの規定に適合しなければならないので、スケーラブルビットストリーム中に含まれる全ての適合点に対して **idr\_pic\_id** への制約に従わなければならない。

**pic\_order\_cnt\_lsb**、**delta\_pic\_order\_cnt\_bottom**、**delta\_pic\_order\_cnt[0]**、および **delta\_pic\_order\_cnt[1]** は 7.4.3 小節で規定されるのと同じの意味を持つ。

**redundant\_pic\_cnt** は 7.4.3 小節で規定されるのと同じの意味を持つ。

**direct\_spatial\_mv\_pred\_flag** は 7.4.3 小節で規定されるのと同じの意味を持つ。加えて、**direct\_spatial\_mv\_pred\_flag** の値は 1 に等しくなければならない。

**num\_ref\_idx\_active\_override\_flag**、**num\_ref\_idx\_l0\_active\_minus1**、および **num\_ref\_idx\_l1\_active\_minus1** は 7.4.3 小節で規定されるのと同じの意味を持つ。

**base\_pred\_weight\_table\_flag** が 1 に等しいことは重み付け予測に対する変数が推定されることを規定する。**base\_pred\_weight\_table\_flag** が存在しない場合、以下のように推定されなければならない。

- もし **no\_inter\_layer\_pred\_flag** が 1 に等しい、あるいは **quality\_id** が 0 よりも大きいならば、**base\_pred\_weight\_table\_flag** は 0 に等しいと推定される。
- それ以外 (**no\_inter\_layer\_pred\_flag** が 0 に等しく、かつ **quality\_id** が 0 に等しい)、**base\_pred\_weight\_table\_flag** は 1 に等しいと推定される。

**setOfRefLayerSlices** は現アクセスユニット内の (**ref\_layer\_dq\_id** >> 4) に等しい **dependency\_id** および 0 に等しい **quality\_id** を有する VCL NAL ユニットによって表現されるスライスの集合であるとする。

base\_pred\_weight\_table\_flag が 1 に等しい場合、以下の制約に従わなければならない。

- a) setOfRefLayerSlices 中の全ての slice に対して、(slice\_type%5)の値は現スライスの(slice\_type%5)に等しくなければならない。
- b) base\_pred\_weight\_table\_flag が 1 に等しく、かつシンタックス構造 pred\_weight\_table( )が setOfRefLayerSlices 中のいずれかのスライス中に存在する場合、それは setOfRefLayerSlices 中の全てのスライスに対して存在しなければならず、かつシンタックス構造 pred\_weight\_table( )内の全てのシンタックス要素は setOfRefLayerSlice 中の全てのスライスに対して同一でなければならない。
- c) base\_pred\_weight\_table\_flag が setOfRefLayerSlices 中のいずれかのスライスに存在する場合、それは setOfRefLayerSlices 中の全てのスライスにおいて同一の値を持たなければならない。
- d) 現スライスが EP スライスである場合、以下が適用される。
  - i. setOfRefLayerSlices 中の全てのスライスの num\_ref\_idx\_l0\_active\_minus1 の値は現スライスの num\_ref\_idx\_l0\_active\_minus1 の値と同一でなければならない。
  - ii. setOfRefLayerSlices 中の各々のスライスに対してシンタックス構造 ref\_pic\_list\_reordering()内の全てのシンタックス要素は同一でなければならない、setOfRefLayerSlices 中のスライスに対するシンタックス構造 ref\_pic\_list\_reordering( )は、0 および 1 に等しい useRefBasePicFlag に対して、(ref\_layer\_dq\_id>>4)に等しく設定された currDependencyId、useRefLayerRepFlag、および setOfRefLayerSlices の任意のスライスを入力として用いる G.8.2.3 小小節の起動により導出される参照ピクチャリスト refPicList0RefLayer が、dependency\_id に等しく設定された currDependencyId、useRefLayerRepFlag および現スライスを入力として用いて G.8.2.3 小小節を起動することによって導出される参照ピクチャリスト refPicList0 と同一になるような、シンタックス要素を含まなければならない。
  - iii. setOfRefLayerSlices 中の全てのスライスに対してシンタックス要素 weighted\_pred\_flag は 1 に等しくなければならない。
- e) 現スライスが EB スライスである場合、以下が適用される。
  - i. setOfRefLayerSlices 中の全てのスライスの num\_ref\_idx\_l0\_active\_minus1 および num\_ref\_idx\_l1\_active\_minus1 の値は現スライスの num\_ref\_idx\_l0\_active\_minus1 および num\_ref\_idx\_l1\_active\_minus1 の値とそれぞれ同一でなければならない。
  - ii. setOfRefLayerSlices 中の各々のスライスに対してシンタックス構造 ref\_pic\_list\_reordering()内の全てのシンタックス要素は同一でなければならない、setOfRefLayerSlices 中のスライスに対するシンタックス構造 ref\_pic\_list\_reordering( )は、0 および 1 に等しい useRefBasePicFlag に対して、(ref\_layer\_dq\_id>>4)に等しく設定された currDependencyId、useRefLayerRepFlag、および setOfRefLayerSlices の任意のスライスを入力として用いて G.8.2.3 小小節を起動することによって導出される参照ピクチャリスト refPicList0RefLayer および refPicList1RefLayer が、各々、dependency\_id に等しく設定された currDependencyId、useRefLayerRepFlag および現スライスを入力として用いて G.8.2.3 小小節を起動することによって導出される参照ピクチャリスト refPicList0 および refPicList1 と同一になるような、シンタックス要素を含まなければならない。

- iii. `setOfRefLayerSlices` 中の全てのスライスに対してシンタックス要素 `weighted_bipred_idc` は 1 に等しくなければならない。

`quality_id` が 0 よりも大きい場合、全ての `pred_weight_table` 要素はこの小節の冒頭で規定したように推定される。

`store_ref_base_pic_flag` は G.7.4.2.12 小小節で規定されるのと同じの意味を持つ。

`cabac_init_idc` および `slice_qp_delta` は 7.4.3 小小節で規定されるのと同じの意味を持つ。

`disable_deblocking_filter_idc` は、デブロックフィルタの操作がそのスライスのいくつかのブロックエッジ交差に対して無効化されなければならないかどうかを規定し、どのエッジに対してフィルタリングが無効化されるかを規定する。`disable_deblocking_filter_idc` がスライスヘッダ中に存在しない場合、`disable_deblocking_filter_idc` の値は 0 に等しいと推定されなければならない。

`disable_deblocking_filter_idc` の値は 0 から 6 までのそれぞれの値を含む範囲内にななければならない。0 に等しい `disable_deblocking_filter_idc` は現ピクチャの全ての輝度および色差ブロックエッジがフィルタされることを規定する。1 に等しい `disable_deblocking_filter_idc` は現ピクチャの全てのブロックエッジに対してデブロックが無効化されることを規定する。2 に等しい `disable_deblocking_filter_idc` はスライス境界と一致するブロックエッジを除く全ての輝度および色差ブロックエッジがフィルタされることを規定する。3 に等しい `disable_deblocking_filter_idc` は、(`disable_deblocking_filter_idc` が 2 に等しい時のように) スライス境界と一致しない全ての輝度および色差ブロックエッジのフィルタリングの後、スライス境界と一致する輝度および色差ブロックエッジがフィルタされるといった、2つのステージからなるデブロックフィルタ処理を規定する。4 に等しい `disable_deblocking_filter_idc` は現ピクチャの全ての輝度ブロックエッジがフィルタされ、その一方で色差ブロックエッジのデブロックが無効化されることを規定する。5 に等しい `disable_deblocking_filter_idc` は (`disable_deblocking_filter_idc` が 2 に等しい時のように) スライス境界と一致するブロックエッジを除く全ての輝度ブロックエッジがフィルタされ、色差ブロックエッジに対するデブロックが無効化されることを規定する。6 に等しい `disable_deblocking_filter_idc` は色差ブロックエッジに対するデブロックが無効化され、かつ (`disable_deblocking_filter_idc` が 2 に等しい時のように) スライス境界と一致しない全ての輝度ブロックエッジのフィルタリングの後、スライス境界と一致する輝度ブロックエッジがフィルタされるといった、2つのステージからなるデブロックフィルタ処理が輝度ブロックエッジに対して用いられることを規定する。

`slice_alpha_c0_offset_div2`、および `slice_beta_offset_div2` は 7.4.3 小小節で規定されるのと同じの意味を持つ。

`slice_group_change_cycle` は 7.4.3 小小節で規定されるのと同じの意味を持つ。

`ref_layer_dq_id` は現レイヤ表現のレイヤ間予測に対して用いられる現アクセスユニット内部のレイヤ表現を規定する。`ref_layer_dq_id` が存在しない場合、以下のように推定されなければならない。

- もし `quality_id` が 0 よりも大きいならば、`ref_layer_dq_id` は(DQId-1)に等しいと推定される。
- それ以外 (`quality_id` が 0 に等しい)、`ref_layer_dq_id` は-1 に等しいと推定される。

`quality_id` が 0 に等しい場合、`ref_layer_dq_id` に等しい DQId を有する NAL ユニットの `discardable_flag` は 0 に等しいと推定されなければならない。

存在する場合、`ref_layer_dq_id` の値はレイヤ表現の全ての符号化スライス NAL ユニットのに対して同一でなければならない。

ref\_layer\_dq\_id が-1 に等しくない場合、以下の変数が以下のように導出される。

- RefLayerPicSizeInMbs は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicSizeInMbs の値に等しく設定される。
- RefLayerPicWidthInMbs は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicWidthInMbs の値に等しく設定される。
- RefLayerPicHeightInMbs は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicHeightInMbs の値に等しく設定される。
- RefLayerChromaFormatIdc は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対するシンタックス要素 chroma\_format\_idc の値に等しく設定される。
- RefLayerChromaArrayType は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する ChromaArrayType の値に等しく設定される。
- RefLayerPicWidthInSamples<sub>L</sub> は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicWidthInSamples<sub>L</sub> の値に等しく設定される。
- RefLayerPicHeightInSamples<sub>L</sub> は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicHeightInSamples<sub>L</sub> の値に等しく設定される。
- RefLayerPicWidthInSamples<sub>C</sub> は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicWidthInSamples<sub>C</sub> の値に等しく設定される。
- RefLayerPicHeightInSamples<sub>C</sub> は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 PicHeightInSamples<sub>C</sub> の値に等しく設定される。
- RefLayerMbWidthC は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 MbWidthC の値に等しく設定される。
- RefLayerMbHeightC は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 MbHeightC の値に等しく設定される。
- RefLayerFrameMbsOnlyFlag は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対するシンタックス要素 frame\_mbs\_only\_flag の値に等しく設定される。
- RefLayerFieldPicFlag は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対するシンタックス要素 field\_pic\_flag の値に等しく設定される。
- RefLayerBottomFieldFlag は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対するシンタックス要素 bottom\_field\_flag の値に等しく設定される。
- RefLayerMbaffFrameFlag は ref\_layer\_dq\_id の値に等しい DQId を有するレイヤ表現に対する変数 MbaffFrameFlag の値に等しく設定される。

**disable\_inter\_layer\_deblocking\_filter\_idc** は、デブロックフィルタの操作がスライスのいくつかのブロックエッジ交差に対してレイヤ間 INTRA 予測に対する無効化されるかどうかを規定し、どのエッジに対してフィルタリングが無効化されるかを規定する。disable\_inter\_layer\_deblocking\_filter\_idc がスライスヘッダ中に存在しない場合、disable\_inter\_layer\_deblocking\_filter\_idc の値は 0 に等しいと推定されなければならない。

disable\_inter\_layer\_deblocking\_filter\_idc の値は 0 から 6 までのそれぞれの値を含む範囲内になければならない。disable\_inter\_layer\_deblocking\_filter\_idc の 0 から 6 までの値は、再サンプリング前の ref\_layer\_dq\_id によって規定される依存表現の INTRA マクロブロックのデブロック以外は、disable\_deblocking\_filter\_idc の対応する値と同じデブロックフィルタ操作を規定する。

quality\_id が 0 に等しくかつ後続の段落中で規定されるような SpatialResolutionChangeFlag が 0 に等しい場合、disable\_inter\_layer\_deblocking\_filter\_idc は 1 に等しくなければならない。

inter\_layer\_slice\_alpha\_c0\_offset\_div2 はレイヤ間予測に対するスライス内のマクロブロックによって制御されるフィルタリング操作に対する  $\alpha$  および  $t_{c0}$  デブロックフィルタ表をアクセスする際に使用されるオフセットを規定する。この値から、これらの表をアドレッシングする時に適用されるオフセットは次のように算出されなければならない。

$$\text{InterlayerFilterOffsetA} = \text{inter\_layer\_slice\_alpha\_c0\_offset\_div2} \ll 1 \quad (\text{G-67})$$

inter\_layer\_slice\_alpha\_c0\_offset\_div2 の値は -6 から 6 までのそれぞれの値を含む範囲内になければならない。inter\_layer\_slice\_alpha\_c0\_offset\_div2 がスライスヘッダ内に存在しない場合、inter\_layer\_slice\_alpha\_c0\_offset\_div2 の値は 0 に等しいと推定されなければならない。

inter\_layer\_slice\_beta\_offset\_div2 はレイヤ間予測に対するスライス内のマクロブロックによって制御されるフィルタリング操作に対する  $\beta$  デブロックフィルタ表をアクセスする際に用いられるオフセットを規定する。この値から、デブロックフィルタの  $\beta$  表へのアドレッシングする時に適用されるオフセットは次のように算出される。

$$\text{InterlayerFilterOffsetB} = \text{inter\_layer\_slice\_beta\_offset\_div2} \ll 1 \quad (\text{G-68})$$

inter\_layer\_slice\_beta\_offset\_div2 の値は -6 から 6 までのそれぞれの値を含む範囲内になければならない。inter\_layer\_slice\_beta\_offset\_div2 がスライスヘッダ内に存在しない場合、inter\_layer\_slice\_beta\_offset\_div2 の値は 0 に等しいと推定されなければならない。

constrained\_intra\_resampling\_flag は、INTRA 再サンプリング処理において、レイヤ間予測に対して ref\_layer\_dq\_id によって規定される依存表現のスライス境界がレイヤピクチャ境界と似たように扱われるかどうかを規定する。constrained\_intra\_resampling\_flag が 1 に等しい場合、disable\_inter\_layer\_deblocking\_filter\_idc は 1、2、あるいは 4 に等しくなければならない。

constrained\_intra\_resampling\_flag が 1 に等しい場合、マクロブロックはそれが G.8.6.2 小節で規定されるように参照レイヤ中の 1 つ以上のスライスをカバーする場合は Intra\_Base マクロブロック予測モードを用いて符号化され得ない。

constrained\_intra\_resampling\_flag が存在しない場合、それは 0 に等しいと推定されなければならない。

ref\_layer\_chroma\_phase\_x\_plus1\_flag はレイヤ間予測に対して使用されても良いレイヤピクチャに対してレイヤフレームの 2 分の 1 輝度サンプルを単位とする色差成分の水平方向の位相シフトを規定する。ref\_layer\_chroma\_phase\_x\_plus1\_flag が存在しない場合、それは seq\_ref\_layer\_chroma\_phase\_x\_plus1\_flag に等しいと推定されなければならない。

no\_inter\_layer\_pred\_flag が 0 に等しい場合、以下が適用される。

- ref\_layer\_dq\_id が 0 よりも大きい場合、ref\_layer\_chroma\_phase\_x\_plus1\_flag は(ref\_layer\_dq\_id に等しい

DQId を持って)参照レイヤ表現によって参照されるサブセットシーケンスパラメータセット RBSP の chroma\_phase\_x\_plus1\_flag に等しくあるべきである。

- chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field が (ref\_layer\_dq\_id に等しい DQId を持って)参照レイヤ表現によって参照される SVC シーケンスパラメータセット中に存在する場合、以下が適用される。
  - もし ref\_layer\_chroma\_phase\_x\_plus1\_flag が 0 に等しいならば、参照レイヤ表現によって参照される SVC シーケンスパラメータセットの chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field は 0、2、あるいは 4 であるべきである。
  - それ以外(ref\_layer\_chroma\_phase\_x\_plus1\_flag が 1 に等しい)、参照レイヤ表現によって参照される SVC シーケンスパラメータセットの chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field は 1、3、あるいは 5 であるべきである。

**ref\_layer\_chroma\_phase\_y\_plus1** はレイヤ間予測に対して使用されても良いレイヤピクチャに対してレイヤフレームの 2 分の 1 輝度サンプルを単位とする色差成分の垂直方向の位相シフトを規定する。ref\_layer\_chroma\_phase\_y\_plus1 が存在しない場合、それは seq\_ref\_layer\_chroma\_phase\_y\_plus1 に等しいと推定されなければならない。ref\_layer\_chroma\_phase\_y\_plus1 の値は 0 から 2 までのそれぞれの値を含む範囲内になければならない。

no\_inter\_layer\_pred\_flag が 0 に等しい場合、以下が適用される。

- ref\_layer\_dq\_id が 0 よりも大きい場合、ref\_layer\_chroma\_phase\_y\_plus1 は (ref\_layer\_dq\_id に等しい DQId を伴って) 参照レイヤ表現によって参照されるサブセットシーケンスパラメータセット RBSP の chroma\_phase\_y\_plus1 に等しくあるべきである。
- chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field が (ref\_layer\_dq\_id に等しい DQId を伴って)参照レイヤ表現によって参照される SVC シーケンスパラメータセット中に存在する場合、以下が適用される。
  - もし ref\_layer\_chroma\_phase\_y\_plus1 が 0 に等しいならば、参照レイヤ表現によって参照される SVC シーケンスパラメータセットの chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field は 2 あるいは 3 に等しくあるべきである。
  - それ以外 (ref\_layer\_chroma\_phase\_y\_plus1 が 1 に等しい)、参照レイヤ表現によって参照される SVC シーケンスパラメータセットの chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field は 0 あるいは 1 に等しくあるべきである。
  - それ以外 (ref\_layer\_chroma\_phase\_y\_plus1 が 2 に等しい)、参照レイヤ表現によって参照される SVC シーケンスパラメータセットの chroma\_sample\_loc\_type\_top\_field および chroma\_sample\_loc\_type\_bottom\_field は 4 あるいは 5 に等しくあるべきである。

**scaled\_ref\_layer\_left\_offset** はレイヤ間予測に対して使用される再サンプリングされたレイヤピクチャの左上輝度サンプルと現ピクチャあるいは現レイヤピクチャの左上輝度サンプルとの間の水平方向のオフセットを 2 輝度サンプルを単位として規定する。

scaled\_ref\_layer\_left\_offset が存在しない場合、それは以下のように推定されなければならない。

- もし quality\_id が 0 よりも大きいならば、scaled\_ref\_layer\_left\_offset は 0 に等しいと推定される。

- それ以外 (quality\_id が 0 に等しい) 、 scaled\_ref\_layer\_left\_offset は seq\_scaled\_ref\_layer\_left\_offset に等しいと推定される。

**scaled\_ref\_layer\_top\_offset** はレイヤ間予測に対して使用される再サンプリングされたレイヤピクチャの左上輝度サンプルと現ピクチャあるいは現レイヤピクチャの左上輝度サンプルとの間の垂直方向のオフセットを規定する。垂直方向のオフセットは frame\_mbs\_only\_flag が 1 に等しい場合は 2 輝度サンプルを単位として規定され、かつそれは frame\_mbs\_only\_flag が 0 に等しい場合は 4 輝度サンプルを単位として規定される。

scaled\_ref\_layer\_top\_offset が存在しない場合、それは以下のように推定されなければならない。

- もし quality\_id が 0 よりも大きいならば、scaled\_ref\_layer\_top\_offset は 0 に等しいと推定される。
- それ以外 (quality\_id が 0 に等しい) 、 scaled\_ref\_layer\_top\_offset は seq\_scaled\_ref\_layer\_top\_offset に等しいと推定される。

**scaled\_ref\_layer\_right\_offset** はレイヤ間予測に対して使用される再サンプリングされたレイヤピクチャの右下輝度サンプルと現ピクチャあるいは現レイヤピクチャの右下輝度サンプルとの間の水平方向のオフセットを 2 輝度サンプルを単位として規定する。

scaled\_ref\_layer\_right\_offset が存在しない場合、それは以下のように推定されなければならない。

- もし quality\_id が 0 よりも大きいならば、scaled\_ref\_layer\_right\_offset は 0 に等しいと推定される。
- それ以外 (quality\_id が 0 に等しい) 、 scaled\_ref\_layer\_right\_offset は seq\_scaled\_ref\_layer\_right\_offset に等しいと推定される。

**scaled\_ref\_layer\_bottom\_offset** はレイヤ間予測に対して使用される再サンプリングされたレイヤピクチャの右下輝度サンプルと現ピクチャあるいは現レイヤピクチャの右下輝度サンプルとの間の垂直方向のオフセットを規定する。垂直方向のオフセットは frame\_mbs\_only\_flag が 1 に等しい場合は 2 輝度サンプルを単位として規定され、かつそれは frame\_mbs\_only\_flag が 0 に等しい場合は 4 輝度サンプルを単位として規定される。

scaled\_ref\_layer\_bottom\_offset が存在しない場合、それは以下のように推定されなければならない。

- もし quality\_id が 0 よりも大きいならば、scaled\_ref\_layer\_bottom\_offset は 0 に等しいと推定される。
- それ以外 (quality\_id が 0 に等しい) 、 scaled\_ref\_layer\_bottom\_offset は seq\_scaled\_ref\_layer\_bottom\_offset に等しいと推定される。

変数 ScaledRefLayerLeftOffset 、 ScaledRefLayerRightOffset 、 ScaledRefLayerTopOffset 、 ScaledRefLayerBottomOffset、 ScaledRefLayerPicWidthInSamples<sub>L</sub>、および ScaledRefLayerPicHeightInSamples<sub>L</sub> は以下の式によって導出される。

$$\text{ScaledRefLayerLeftOffset} = 2 * \text{scaled\_ref\_layer\_left\_offset} \quad (\text{G-69})$$

$$\text{ScaledRefLayerRightOffset} = 2 * \text{scaled\_ref\_layer\_right\_offset} \quad (\text{G-70})$$

$$\text{ScaledRefLayerTopOffset} = 2 * \text{scaled\_ref\_layer\_top\_offset} * (2 - \text{frame\_mbs\_only\_flag}) \quad (\text{G-71})$$

$$\text{ScaledRefLayerBottomOffset} = 2 * \text{scaled\_ref\_layer\_bottom\_offset} * (2 - \text{frame\_mbs\_only\_flag}) \quad (\text{G-72})$$

$$\begin{aligned} \text{ScaledRefLayerPicWidthInSamples}_L &= \text{PicWidthInMbs} * 16 - \text{ScaledRefLayerLeftOffset} - \\ &\quad \text{ScaledRefLayerRightOffset} \end{aligned} \quad (\text{G-73})$$

$$\text{ScaledRefLayerPicHeightInSamples}_L = \text{PicHeightInMbs} * 16 -$$

$$\frac{(\text{ScaledRefLayerTopOffset}+\text{ScaledRefLayerBottomOffset})}{(1+\text{field\_pic\_flag})} \quad (\text{G-74})$$

no\_inter\_layer\_pred\_flag が 0 に等しい場合、以下が適用される。

- ビットストリームは RefLayerPicWidthInSamples<sub>L</sub> よりも小さい ScaledRefLayerPicWidthInSamples<sub>L</sub> となる結果を生じるデータを含んではならない。
- RefLayerFrameMbsOnlyFlag が 0 に等しいあるいは frame\_mbs\_only\_flag が 1 に等しい場合、ビットストリームは (RefLayerPicHeightInSamples<sub>L</sub>\*(1+RefLayerFieldPicFlag)) よりも小さい (ScaledRefLayerPicHeightInSamples<sub>L</sub>\*(1+field\_pic\_flag)) となる結果を生じるデータを含んではならない。
- RefLayerFrameMbsOnlyFlag が 1 に等しいあるいは frame\_mbs\_only\_flag が 0 に等しい場合、ビットストリームは (2\*RefLayerPicHeightInSamples<sub>L</sub>) よりも小さい (ScaledRefLayerPicHeightInSamples<sub>L</sub>\*(1+field\_pic\_flag)) となる結果を生じるデータを含んではならない。

ChromaArrayType が 0 に等しくない場合、変数 ScaledRefLayerPicWidthInSamples<sub>C</sub> および ScaledRefLayerPicHeightInSamples<sub>C</sub> は以下の式によって導出される。

$$\text{ScaledRefLayerPicWidthInSamples}_C = \text{ScaledRefLayerPicWidthInSamples}_L / \text{SubWidthC} \quad (\text{G-75})$$

$$\text{ScaledRefLayerPicHeightInSamples}_C = \text{ScaledRefLayerPicHeightInSamples}_L / \text{SubHeightC} \quad (\text{G-76})$$

変数 CroppingChangeFlag は以下のように導出される。

- もし quality\_id が 0 に等しくかつ extended\_spatial\_scalability が 2 に等しいならば、CroppingChangeFlag は 1 に等しく設定される。
- それ以外 (quality\_id が 0 よりも大きいあるいは extended\_spatial\_scalability が 2 よりも小さい)、CroppingChangeFlag は 0 に等しく設定される。

変数 SpatialResolutionChangeFlag は以下のように導出される。

- もし no\_inter\_layer\_pred\_flag が 1 に等しく、quality\_id が 0 よりも大きく、あるいは以下の条件の全てが真であるならば、SpatialResolutionChangeFlag は 0 に等しく設定される。
  - CroppingChangeFlag が 0 に等しい
  - ScaledRefLayerPicWidthInSamples<sub>L</sub> が RefLayerPicWidthInSamples<sub>L</sub> に等しい
  - ScaledRefLayerPicHeightInSamples<sub>L</sub> が RefLayerPicHeightInSamples<sub>L</sub> に等しい
  - (ScaledRefLayerLeftOffset%16)が 0 に等しい
  - (ScaledRefLayerTopOffset%(16\*(1+frame\_mbs\_only\_flag)))が 0 に等しい
  - field\_pic\_flag が RefLayerFieldPicFlag に等しい
  - MbaffFrameFlag が RefLayerMbaffFrameFlag に等しい
  - chroma\_format\_idc が RefLayerChromaFormatIdc に等しい
  - chroma\_phase\_x\_plus1\_flag が ref\_layer\_chroma\_phase\_x\_plus1\_flag に等しい

- chroma\_phase\_y\_plus1 が ref\_layer\_chroma\_phase\_y\_plus1 に等しい
- それ以外、SpatialResolutionChangeFlag は 1 に等しく設定される。

変数 RestrictedSpatialResolutionChangeFlag は以下のように導出される。

- もし no\_inter\_layer\_pred\_flag が 1 に等しく、quality\_id が 0 よりも大きく、あるいは以下の条件の全てが真であるならば、RestrictedSpatialResolutionChangeFlag は 1 に等しく設定される。
  - ScaledRefLayerPicWidthInSamples<sub>L</sub> が RefLayerPicWidthInSamples<sub>L</sub> ある い は (2\*RefLayerPicWidthInSamples<sub>L</sub>)に等しい
  - ScaledRefLayerPicHeightInSamples<sub>L</sub> が RefLayerPicHeightInSamples<sub>L</sub> ある い は (2\*RefLayerPicHeightInSamples<sub>L</sub>)に等しい
  - (ScaledRefLayerLeftOffset%16)が 0 に等しい
  - (ScaledRefLayerTopOffset%(16\*(1+frame\_mbs\_only\_flag)))が 0 に等しい
  - MbaffFrameFlag が RefLayerMbaffFrameFlag に等しい
  - field\_pic\_flag が RefLayerFieldPicFlag に等しい
- それ以外、RestrictedSpatialResolutionChangeFlag は 0 に等しく設定される。

slice\_skip\_flag が 0 に等しいことは、スライス中のスライスデータの存在を規定する。1 に等しい slice\_skip\_flag はスライスのマクロブロックレイヤに対するシンタックス要素が以下の処理によって導出されることを規定する。

1. CurrMbAddr は以下の式によって導出される

$$\text{CurrMbAddr} = \text{first\_mb\_in\_slice} * (1 + \text{MbaffFrameFlag}) \quad (\text{G-77})$$

2. 変数 mbIdx が値 0..num\_mbs\_in\_slice\_minus1 に渡って進行し、mbIdx のそれぞれの値に対して、以下の順序付けられたステップが規定される。
  - a. ビットストリームは 0 に等しい in\_crop\_window(CurrMbAddr)となる結果を生じるデータを含んではない。
  - b. アドレス CurrMbAddr を持つマクロブロックに対して、シンタックス要素 mb\_skip\_flag (適用可能な場合)、mb\_skip\_run (適用可能な場合)、mb\_field\_decoding\_flag、base\_mode\_flag、residual\_prediction\_flag および coded\_block\_pattern は以下のように推定されなければならない。
    - mb\_skip\_flag (適用可能な場合) および mb\_skip\_run (適用可能な場合) は 0 に等しいと推定される
    - mb\_field\_decoding\_flag は 0 に等しいと推定される
 

記2 復号に対して使用されるフレーム/フィールドモードはG.8.1.5.1小小節において推定される。
    - base\_mode\_flag は 1 に等しいと推定される
    - residual\_prediction\_flag は 1 に等しいと推定される
    - coded\_block\_pattern は 0 に等しいと推定される

- $QP_Y$  は  $SliceQP_Y$  に等しいと推定される
  - $QP'_Y$  は  $(QP_Y + QpBdOffset_Y)$  に等しいと推定される
- c.  $CurrMbAddr$  は  $NextMbAddress(CurrMbAddr)$  に設定される

$num\_mbs\_in\_slice\_minus1$  に 1 を加えたものは 1 に等しい  $slice\_skip\_flag$  を有するスライスに対するマクロブロック数を規定する。

**adaptive\_base\_mode\_flag** はスライスヘッダおよびスケーラブル拡張におけるマクロブロックレイヤ中のシンタックス要素の存在を規定する。**adaptive\_base\_mode\_flag** が存在しない場合、それは 0 に等しいと推定されなければならない。

**default\_base\_mode\_flag** はそれがスケーラブル拡張におけるマクロブロックレイヤ中に存在しない場合に **base\_mode\_flag** がどのように推定されるかを規定する。**default\_base\_mode\_flag** が存在しない場合、それは 0 に等しいと推定されなければならない。

**adaptive\_motion\_prediction\_flag** はスケーラブル拡張におけるマクロブロックレイヤ中のシンタックス要素の存在を規定する。**adaptive\_motion\_prediction\_flag** が存在しない場合、それは 0 に等しいと推定されなければならない。

**default\_motion\_prediction\_flag** は  $motion\_prediction\_flag\_10[ ]$  および  $motion\_prediction\_flag\_11[ ]$  がスケーラブル拡張におけるマクロブロックレイヤ中に存在しない場合、それらがどのように推定されるかを規定する。**default\_motion\_prediction\_flag** が存在しない場合、それは 0 であると推定される。

**adaptive\_residual\_prediction\_flag** はスケーラブル拡張におけるマクロブロックレイヤ中のシンタックス要素の存在を規定する。**adaptive\_residual\_prediction\_flag** が存在しない場合、それは 0 に等しいと推定される。

**default\_residual\_prediction\_flag** は **residual\_prediction\_flag** がスケーラブル拡張におけるマクロブロックレイヤ中に存在しない場合、それがどのように推定されるかを規定する。**default\_residual\_prediction\_flag** が存在しない場合、それは 0 であると推定される。

**tcoeff\_level\_prediction\_flag** が 1 に等しいことは、代替のレイヤ間予測処理が G8 節で規定されるように適用されることを規定する。**tcoeff\_level\_prediction\_flag** が存在しない場合、それは以下のように推定されなければならない。

- もし **no\_inter\_layer\_pred\_flag** が 1 に等しいならば、**tcoeff\_level\_prediction\_flag** は 0 に等しいと推定される。
- それ以外（**no\_inter\_layer\_pred\_flag** が 0 に等しい）、**tcoeff\_level\_prediction\_flag** は **seq\_tcoeff\_level\_prediction\_flag** の値に等しいと推定される。

**SpatialResolutionChangeFlag** が 1 に等しい場合、**tcoeff\_level\_prediction\_flag** は 0 に等しくなければならない。

**tcoeff\_level\_prediction\_flag** が 1 に等しい場合、以下の制約に従わなければならない。

- a) **ref\_layer\_dq\_id** に等しい DQId を有するレイヤ表現は 1 に等しい **no\_inter\_layer\_pred\_flag** あるいは 1 に等しい **tcoeff\_level\_prediction\_flag** を持たなくてはならない。
- b) **ScalingList4x4** の全ての要素は現レイヤ表現のスライスおよび **ref\_layer\_dq\_id** の値に等しい DQId を有するレイヤ表現の全てのスライスに対して同じでなければならない。
- c) **ScalingList8x8** の全ての要素は現レイヤ表現のスライスおよび **ref\_layer\_dq\_id** の値に等しい DQId を有するレイヤ表現の全てのスライスに対して同じでなければならない。

d) シンタックス要素 `use_ref_base_pic_flag` の値は現レイヤ表現のスライスおよび `ref_layer_dq_id` の値に等しい DQId を有するレイヤ表現の全てのスライスに対して 0 に等しくなければならない。

`scan_idx_start` は現スライス中の変換係数レベルに対する最初の走査位置を規定する。`scan_idx_start` が存在しない場合、それは 0 に等しいと推定されなければならない。

`scan_idx_end` は現スライス中の変換係数レベルに対する最後の走査位置を規定する。`scan_idx_end` が存在しない場合、それは 15 に等しいと推定されなければならない。

#### G.7.4.3.5 復号参照ベースピクチャマーキングの意味

シンタックス要素 `adaptive_ref_base_pic_marking_mode_flag`、`memory_management_control_operation`、`difference_of_pic_nums_minus1` および `long_term_pic_num` は参照ベースピクチャの“参照不使用”としてのマーキングを規定する。

`adaptive_ref_base_pic_marking_mode_flag` は `dec_ref_pic_marking()` シンタックス構造内のシンタックス要素 `adaptive_ref_pic_marking_mode_flag` と同一の意味を持つ。`adaptive_ref_base_pic_marking_mode_flag` が存在しない場合は、それは 0 に等しいと推定されなければならない。

`memory_management_control_operation` は参照ピクチャマーキングに影響を与えるために適用される制御操作を規定する。`memory_management_control_operation` シンタックス要素に `memory_management_control_operation` の値によって規定される操作に対して必要なデータが後続する。`memory_management_control_operation` に関連する値および制御操作は付表 G-2/JT-H264 において規定される。`memory_management_control_operation` シンタックス要素はそれらが現れる順序で復号処理によって処理され、各々の `memory_management_control_operation` に対して表現される意味制約はその個別の `memory_management_control_operation` が処理されるその順序内の特定位置において適用される。

`memory_management_control_operation` は、`memory_management_control_operation` が復号処理によって処理される時、規定された参照ピクチャが“短期間参照使用”および“参照ベースピクチャ”とマークされない限り、1 に等しくあってはならない。

`memory_management_control_operation` は、`memory_management_control_operation` が復号処理によって処理される時、規定された長期参照ピクチャ番号が“長期間参照使用”および“参照ベースピクチャ”とマークされた参照ピクチャを参照しない限り、2 に等しくあってはならない。

付表G-2/JT-H264 メモリ管理制御操作 (`memory_management_control_operation`) の値 (ITU-T H.264)

<code>memory_management_control_operation</code>	メモリ管理制御操作
0	<code>memory_management_control_operation</code> シンタックス要素ループを終了する
1	短期間参照ベースピクチャを“参照不使用”としてマークする
2	長期間参照ベースピクチャを“参照不使用”としてマークする

`difference_of_pic_nums_minus1` は短期間参照ピクチャを“参照不使用”とマークするために (1 に等しい `memory_management_control_operation` と共に) 使用される。関連付けられた `memory_management_control_operation` が復号処理によって処理される時、`difference_of_pic_nums_minus1` より導出される結果として得られるピクチャ番号は“参照使用”および“参照ベースピクチャ”としてマークさ

れた参照ピクチャのうちのひとつに割り当てられたピクチャ番号でなければならない。

結果として得られるピクチャ番号は以下のように制約される。

- もし `field_pic_flag` が 0 に等しいならば、結果として得られるピクチャ番号は参照フレームあるいは相補的参照フィールドペアに割り当てられるピクチャ番号の集合のうちのひとつでなければならない、あるいは結果として得られるピクチャ番号に割り当てられている参照フレームおよび相補的参照フィールドペアは無い。
- それ以外 (`field_pic_flag` が 1 に等しい)、結果として得られるピクチャ番号は参照フィールドに割り当てられるピクチャ番号の集合のうちのひとつでなければならない、あるいは結果として得られるピクチャ番号に割り当てられている参照ペアは無い。

`long_term_pic_num` は長期間参照ピクチャを“参照不使用”とマークするために (2 に等しい `memory_management_control_operation` と共に) 使用される。関連付けられた `memory_management_control_operation` が復号処理によって処理される時、`long_term_pic_num` は“長期間参照使用”および“参照ベースピクチャ”とマークされた参照ピクチャのうちのひとつに割り当てられた長期間ピクチャ番号に等しくなければならない。

結果として得られる長期間ピクチャ番号は以下のように制約される。

- もし `field_pic_flag` が 0 に等しいならば、結果として得られる長期間ピクチャ番号は参照フレームあるいは相補的参照フィールドペアに割り当てられるピクチャ番号の集合のうちのひとつでなければならない、あるいは結果として得られる長期間ピクチャ番号に割り当てられている参照フレームあるいは相補的参照フィールドペアは無い。
- それ以外 (`field_pic_flag` は 1 に等しい)、結果として得られる長期間ピクチャ番号は参照フィールドに割り当てられるピクチャ番号の集合のうちのひとつでなければならない、あるいは結果として得られる長期間ピクチャ番号に割り当てられている参照ペアは無い。

プリフィックス NAL ユニット中に存在する場合、`dec_ref_base_pic_marking()` シンタックス構造のシンタックス要素は関連付けされた NAL ユニットに適用されるものとみなされる。

#### G.7.4.4 スライスデータの意味

7.4.4 小節で規定される意味が適用される。

##### G.7.4.4.1 スケーラブル拡張内スライスデータの意味

7.4.4 小節で規定される意味が以下の変更と共に適用される。

`mb_skip_run` は、EP スライスを復号する時、`mb_type` は `P_Skip` であると推定されなければならない、そのマクロブロックタイプは総称的に `P` マクロブロックタイプとして参照され、あるいは、EB スライスを復号する時、`mb_type` は `B_Skip` であると推定されなければならない、そのマクロブロックタイプは総称的に `B` マクロブロックタイプとして参照される、連続してスキップされるマクロブロックの数を規定する。`mb_skip_run` の値は 0 から `PicSizeInMbs-CurrMbAddr` のそれぞれの値を含む範囲内にななければならない。

`mb_skip_flag` が 1 に等しいことは、現マクロブロックに対して、EP スライスを復号する時、`mb_type` は `P_Skip` であると推定されなければならない、そのマクロブロックタイプは総称的に `P` マクロブロックタイプとして参照され、あるいは、EB スライスを復号する時、`mb_type` は `B_Skip` であると推定されなければならない、

そのマクロブロックタイプは総称的に B マクロブロックタイプとして参照されることを規定する。0 に等しい `mb_skip_flag` は現マクロブロックはスキップされないことを規定する。

#### G.7.4.5 マクロブロックレイヤの意味

7.4.5 小節で規定される意味が適用される。加えて、以下が適用される。

`macroblock_layer( )` シンタックス構造は以下の推定値をもつ以下のシンタックス要素を含むものとみなされなければならない。

- `base_mode_flag` は 0 に等しいと推定される。
- `residual_prediction_flag` は 0 に等しいと推定される。

##### G.7.4.5.1 マクロブロック予測の意味

7.4.5.1 小小節で規定される意味が適用される。加えて、以下が適用される。

`mb_pred( )` シンタックス構造は以下の推定値をもつ以下のシンタックス要素を含むものとみなされなければならない。

- `motion_prediction_flag_10[mbPartIdx]` は 0 から `NumMbPart(mb_type)-1` までのそれぞれの値を含む範囲内の `mbPartId` のそれぞれの値に対して 0 に等しいと推定される。
- `motion_prediction_flag_11[mbPartIdx]` は 0 から `NumMbPart(mb_type)-1` までのそれぞれの値を含む範囲内の `mbPartId` のそれぞれの値に対して 0 に等しいと推定される。

##### G.7.4.5.2 サブマクロブロック予測の意味

7.4.5.2 小小節で規定される意味が適用される。加えて、以下が適用される。

`sub_mb_pred( )` シンタックス構造は以下の推定値をもつ以下のシンタックス要素を含むものとみなされなければならない。

- `motion_prediction_flag_10[mbPartIdx]` は 0 から 3 までのそれぞれの値を含む範囲内の `mbPartId` のそれぞれの値に対して 0 に等しいと推定される。
- `motion_prediction_flag_11[mbPartIdx]` は 0 から 3 までのそれぞれの値を含む範囲内の `mbPartId` のそれぞれの値に対して 0 に等しいと推定される。

##### G.7.4.5.3 残差データの意味

7.4.5.3 小小節で規定される意味が適用される。

###### G.7.4.5.3.1 残差輝度の意味

7.4.5.3.1 小小小節で規定される意味が適用される。

###### G.7.4.5.3.2 残差ブロック CAVLC の意味

7.4.5.3.2 小小小節で規定される意味が適用される。

### G.7.4.5.3.3 残差ブロックCABACの意味

7.4.5.3.3 小小小節で規定される意味が適用される。

### G.7.4.6 スケーラブル拡張内マクロブロックレイヤの意味

7.4.5 小節で規定される意味が適用される。加えて、以下の変更および拡張が規定される。

**base\_mode\_flag** が 1 に等しいことは対応する動きデータ（適用可能な場合）と同様にマクロブロック予測モードも G.8 節において規定されているように推定されることを規定する。0 に等しい **base\_mode\_flag** は **mb\_type** が推定されないことを規定する。

**base\_mode\_flag** が存在しない場合、**base\_mode\_flag** は以下のように推定されなければならない。

- もし **in\_crop\_window(CurrMbAddr)** が 0 に等しいならば、**base\_mode\_flag** の値は 0 に等しいと推定されなければならない。
- それ以外 (**in\_crop\_window(CurrMbAddr)** が 1 に等しい)、**base\_mode\_flag** の値は **default\_base\_mode\_flag** に等しいと推定されなければならない。

**store\_ref\_base\_pic\_flag** が 1 に等しくかつ **quality\_id** が 0 よりも大きい場合は、**base\_mode\_flag** は 1 に等しくなければならない。

**mb\_type** はマクロブロックタイプを規定する。**mb\_type** の意味はスライスタイプに依存する。

EI、EP、および EB スライスにおける様々なマクロブロックタイプに対して表および意味が規定される。各々の表は **mb\_type** の値、**mb\_type** の名前、(**NumMbPart(mb\_type)**関数によって与えられる) 使用されるマクロブロックパーティション数、マクロブロック（パーティションされない場合）あるいは第 1 のパーティションの予測モード (**MbPartPredMode(mb\_type,0)**関数によって与えられる) および第 2 のパーティションの予測モード (**MbPartPredMode(mb\_type,1)**関数によって与えられる) を提示する。値が適用可能でない場合は“na”によって明示される。文書中では、**mb\_type** の値はマクロブロックタイプと呼ばれるかもしれず、**MbPartPredMode()**の値 X は“X マクロブロック（パーティション）予測モード”によって、もしくは“X 予測マクロブロック”として文書中で呼ばれるかもしれない。

付表 G-3/JT-H264 は、各々の **slice\_type** に対して許可される総称的マクロブロックタイプを表す。

付表G-3/JT-H264 **slice\_type**に対して許可される総称的マクロブロックタイプ  
(ITU-T H.264)

<b>slice_type</b>	許可される総称的マクロブロックタイプ
EI (スライス)	I(表7-11/JT-H264および付表G-4/JT-H264を見ること)(マクロブロックタイプ)
EP (スライス)	P(表7-13/JT-H264を見ること)およびI(表7-11/JT-H264および付表G-4/JT-H264を見ること)(マクロブロックタイプ)
EB (スライス)	B(表7-14/JT-H264を見ること)およびI(表7-11/JT-H264および付表G-4/JT-H264を見ること)(マクロブロックタイプ)

Iマクロブロックタイプとして総称的に参照されても良いマクロブロックタイプは付表 G-4/JT-H264 および表 7-11/JT-H264 において規定される。それ以外の **mb\_type** が表 7-11/JT-H264 において規定されるのに対して、値 0 の **mb\_type** は付表 G-4/JT-H264 において規定される。付表 G-4/JT-H264 は推定されるマクロ

ブロックタイプ I<sub>BL</sub> を付加的に規定する。

EI スライスに対するマクロブロックタイプは全て I マクロブロックタイプである。

付表G-4/JT-H264 EIスライスに対するマクロブロックタイプ  
(ITU-T H.264)

mb_type	mb_typeの名前	transform_size_8x8_flag	MbPartPredMode (mb_type, 0)	Intra16x16PredMode	CodedBlockPatternChroma	CodedBlockPatternLuma
0	I_NxN	0	Intra_4x4	na	等式7-34	等式7-34
0	I_NxN	1	Intra_8x8	na	等式7-34	等式7-34
推定	I <sub>BL</sub>	na	Intra_Base	na	等式7-34	等式7-34

P マクロブロックとして総称的に参照されても良いマクロブロックタイプは表 7-13/JT-H264 において規定される。

EP スライスに対するマクロブロックタイプは表 7-13/JT-H264、表 7-11/JT-H264 および付表 G-4/JT-H264 において規定される。値 0 から 4 の mb\_type は表 7-13/JT-H264 において規定され、値 5 の mb\_type は mb\_type から 5 を減じた値によって索引される付表 G-4/JT-H264 において規定され、そして値 6 から 30 の mb\_type は mb\_type から 5 を減じた値によって索引される表 7-11/JT-H264 において規定される。付表 G-4/JT-H264 は推定されるマクロブロックタイプ I<sub>BL</sub> を付加的に規定する。

B マクロブロックとして総称的に参照されても良いマクロブロックタイプは表 7-14/JT-H264 において規定される。

EB スライスに対するマクロブロックタイプは表 7-14/JT-H264、表 7-11/JT-H264 および付表 G-4/JT-H264 において規定される。値 0 から 22 の mb\_type は表 7-14/JT-H264 において規定され、値 23 の mb\_type は mb\_type から 23 を減じた値によって索引される付表 G-4/JT-H264 において規定され、そして値 24 から 48 の mb\_type は mb\_type から 23 を減じた値によって索引される表 7-14/JT-H264 において規定される。付表 G-4/JT-H264 は推定されるマクロブロックタイプ I<sub>BL</sub> を付加的に規定する。

**coded\_block\_pattern** はマクロブロックの 4 個の 8×8 輝度ブロックおよび関連する色差ブロックのいずれが非ゼロ変換係数値を含んでも良いかを規定する。Intra\_16x16 に等しくない予測モードを有するマクロブロックに対して **coded\_block\_pattern** はビットストリーム中に存在し、変数 CodedBlockPatternLuma および CodedBlockPatternChroma は等式 7-34 によって規定されるように導出される。

CodedBlockPatternLuma および CodedBlockPatternChroma の意味は 7.4.5 小節において規定される。

**residual\_prediction\_flag** が 1 に等しいことは、現マクロブロックの残差信号が ref\_layer\_dq\_id によって規定されるレイヤ表現を使用して G.8 節において規定されるように予測されることを規定する。0 に等しい residual\_prediction\_flag は現マクロブロックの残差信号が予測されないことを規定する。

residual\_prediction\_flag シンタックス要素が存在しない場合、residual\_prediction\_flag は以下のように推定されなければならない。

- もし以下の全ての条件が真であるならば、residual\_prediction\_flag は default\_residual\_prediction\_flag に等しいと推定される。
  - slice\_type は EI に等しくない
  - in\_crop\_window(CurrMbAddr)は 1 に等しい
  - base\_mode\_flag は 1 に等しいかあるいは mb\_type は I マクロブロックタイプを規定しない
- それ以外、residual\_prediction\_flag は 0 に等しいと推定される。

配列 LumaLevel、LumaLevel8x8、Intra16x16DCLevel、Intra16x16ACLevel、CbLevel、CbLevel8x8、CbIntra16x16DCLevel、CbIntra16x16ACLevel、CrLevel、CrLevel8x8、CrIntra16x16DCLevel、CrIntra16x16ACLevel、ChromaDCLevel、および ChromaACLevel の全ての要素は residual( )シンタックス構造を構文解析する前に 0 に等しく設定される。

#### G.7.4.6.1 スケーラブル拡張内マクロブロック予測の意味

7.3.5.1 小小節で規定される意味が適用される。加えて、以下の意味が規定される。

**motion\_prediction\_flag\_10[mbPartIdx]**が 1 に等しいことは、G.8 節において規定されるように代替動きベクトル予測処理がマクロブロックパーティション mbPartIdx のリスト 0 動きベクトルの導出のために使用されること、および G.8 節において規定されるようにマクロブロックパーティション mbPartIdx のリスト 0 参照インデックスが推定されることを規定する。

motion\_prediction\_flag\_10[mbPartIdx]が存在しない場合、motion\_prediction\_flag\_10[mbPartIdx]は以下のように推定されなければならない。

- もし、in\_crop\_window(CurrMbAddr)が 0 に等しいならば、motion\_prediction\_flag\_10[mbPartIdx]は 0 に等しいと推定される。
- それ以外 (in\_crop\_window(CurrMbAddr)が 1 に等しい)、motion\_prediction\_flag\_10[mbPartIdx]は default\_motion\_prediction\_flag に等しいと推定される。

**motion\_prediction\_flag\_11[mbPartIdx]**は motion\_prediction\_flag\_10[mbPartIdx]と同じ意味を持ち、10 およびリスト 0 がそれぞれ 11 およびリスト 1 に置換される。

#### G.7.4.6.2 スケーラブル拡張内サブマクロブロック予測の意味

7.3.5.2 小小節で規定される意味が適用される。加えて、以下の意味が規定される。

**motion\_prediction\_flag\_10[mbPartIdx]**および **motion\_prediction\_flag\_11[mbPartIdx]**は G.7.3.6.1 小小節において規定されるのと同じ意味を持つ。

## G.8 SVC復号処理

この節は、ビットストリームから導出される G.7 節から (G.7 節においてなされる 7 章への参照を用いて) 与えられたシンタックス要素および大文字変数により、アクセスユニットに対する復号処理を記述する。

記1 G.7節から参照される全てのシンタックス要素および大文字変数は現アクセスユニット全体に対して利用できる。シンタックス要素あるいは大文字変数がG.7節において同一の名前を伴って出現した場合、それらは固有の識別子を通してここで参照される。

この処理の出力は現在の主符号化ピクチャの復号サンプルである。

復号処理は全ての復号器が数値的に同一の結果を生み出さなければならないように規定される。ここで記述される処理と同一の結果を生み出すいかなる復号処理も本標準の復号処理の要件に適合する。

入力として `priority_id`、`temporal_id`、`dependency_id`、あるいは `quality_id` の任意の組み合わせを用いて G.8.8.1 小小節に規定されるサブビットストリーム抽出処理を用いて導出され得る全てのサブビットストリームが結果として各符号化ビデオシーケンスが付属資料 A および付属資料 G において規定される 1 つ以上のプロファイルに適合するような符号化ビデオシーケンスの集合にならなければならない。

この節はG.10節において規定される1つ以上のプロファイルに適合する符号化ビデオシーケンスのアクセスユニットに対する復号処理を規定する。

別に記述されない限り、この節で規定される復号処理およびこの節で規定される処理から起動される全ての子処理によって参照されるシンタックス要素および導出される大文字変数は、現アクセスユニットに対するシンタックス要素および導出される大文字変数である。

アクセスユニットに対する復号処理の開始時点で、以下が適用される。

1. ピクチャ順序カウンタに関連する変数および関数が G.8.2.1 小小節において規定されるピクチャ順序カウンタに対する SVC 復号処理を起動することによって導出される。
2. G.8.2.5 小小節において規定される `frame_num` のギャップに対する SVC 復号処理が起動される。
3. 上記で規定されるようにこの処理へ入力される NAL ユニット中に存在する `dependency_id` のそれぞれの値に対して以下が適用される。
  - 8.2.2 小節において規定されるようにマクロブロックからスライスグループへのマップに対する復号処理が `dependency_id` をもつ NAL ユニットのシンタックス要素を入力として起動される。
  - 8.2.2 小節において規定される `NextMbAddress()` 関数が、`dependency_id` の対応する値をもつ全ての NAL ユニットのスライスデータシンタックスの構文解析に対して使用される。

G.8.1.1 小小節において規定される復号に要求されるレイヤ表現集合に対する導出処理が起動され、出力はレイヤ表現識別子を規定する整数値のリスト `dqIdList` である。変数 `DQIdMin` および `DQIdMax` はリスト `dqIdList` のエントリの最小値および最大値にそれぞれ等しく設定され、変数 `DependencyIdMax` は  $(DQIdMax \gg 4)$  に等しく設定される。`DependencyIdMax` は符号化ビデオシーケンスの全てのアクセスユニットに対して同一でなければならない。

総称 `currentVars` および `refLayerVars` は利用不可として初期マークされる。

変数 `currDQId` が値 `DQIdMin..DQIdMax` に渡って進行し、`currDQId` の値がリスト `dqIdList` の中に存在する場合、以下の順序付けられたステップが適用される。

1. 変数 `spatResChangeFlag` は `currDQId` に等しい `DQId` を伴うレイヤ表現の変数 `SpatialResolutionChangeFlag` に等しく設定される。
2. `spatResChangeFlag` に依存して、以下が適用される。

- もし spatResChangeFlag が 0 に等しければ、G.8.1.3.1 小小小節において規定される解像度変更を伴わないレイヤ表現に対するベース復号処理が currDQId および currentVars を入力として起動され、出力は currentVars の修正版である。
  - それ以外 (spatResChangeFlag が 1 に等しい)、G.8.1.3.2 小小小節において規定される解像度変更を伴うレイヤ表現に対するベース復号処理が currDQId および currentVars を入力として起動され、出力は総称 refLayerVars に割り当てられた変数および currentVars の修正版である。
3. currDQId が (DependencyIdMax << 4) に等しくかつ DependencyIdMax に等しい dependency\_id を伴う依存表現に対する store\_ref\_base\_pic\_flag が 1 に等しい場合、G.8.1.3.3 小小小節において規定されるターゲットレイヤ表現の復号処理が currDQId、refLayerVars、および currentVars を入力として起動され、出力はサンプル配列 B<sub>L</sub> に、ChromaArrayType が 0 に等しくない場合はサンプル配列 B<sub>Cb</sub> および B<sub>Cr</sub> に割り当てられる。

記2 サンプル配列 B<sub>L</sub>、B<sub>Cb</sub>、および B<sub>Cr</sub> は DependencyIdMax に等しい dependency\_id を伴う依存表現に対して 1 に等しい store\_ref\_base\_pic\_flag を伴うアクセスユニットに対する参照ベースピクチャを表現する。

G.8.1.3.3 小小小節に規定されるターゲットレイヤ表現の復号処理が currDQId、refLayerVars、および currentVars を入力として起動され、出力はサンプル配列 S<sub>L</sub> に、ChromaArrayType が 0 に等しくない場合はサンプル配列 S<sub>Cb</sub> および S<sub>Cr</sub> に割り当てられる。

記3 サンプル配列 S<sub>L</sub>、S<sub>Cb</sub>、および S<sub>Cr</sub> はアクセスユニットに対する復号ピクチャを表現する。

G.8.2.4 小小小節において規定される SVC 復号参照ピクチャマーキング処理が起動される。

## G.8.1 SVC初期化および復号処理

G.8.1.1 小小小節は復号に要求されるレイヤ表現集合に対する導出処理を規定する。

G.8.1.2 小小小節は配列の割り当て、初期化、および再構成処理を規定する。

G.8.1.3 小小小節はレイヤ表現の復号処理を規定する。

G.8.1.4 小小小節はスライスの復号処理を規定する。

G.8.1.5 小小小節はマクロブロックの初期化および復号処理を規定する。

### G.8.1.1 復号に要求されるレイヤ表現集合に対する導出処理

この処理への入力アクセスユニットの符号化スライス NAL ユニットである。

この処理の出力はレイヤ表現識別子を規定する整数値のリスト dqIdList である。

アクセスユニットの全ての符号化スライス NAL ユニットに対する DQId の最大値に等しく設定されている currDQId を用い、また dqId に等しい DQId を伴うアクセスユニットのレイヤ表現の全ての符号化スライス NAL ユニットに対する ref\_layer\_dq\_id の最大値を規定する refLayerDQId(dqId) を用いて、リスト dqIdList は以下の擬似コードによって規定されるように導出される。

```

numEntries=0
dqIdList[numEntries++]=currDQId
while(refLayerDQId(currDQId)>=0){
    dqIdList[numEntries++]=refLayerDQId(currDQId)
}

```

(G-78)

```

currDQId=dqIdList[numEntries-1]
}

```

## G.8.1.2 配列の割り当て、初期化、および再構成処理

G.8.1.2.1 小小小節は配列の割り当ておよび初期化処理を規定する。

G.8.1.2.2 小小小節は配列の再構成処理を規定する。

### G.8.1.2.1 配列の割り当ておよび初期化処理

この処理の出力は総称 `currentVars` へ割り当てられる配列の組である。

以下の配列が `currentVars` として総称的に参照される。

- レイヤ表現のマクロブロックに対するスライス識別を規定する `PicSizeInMbs` 個の要素をもつ 1 次元配列 `sliceIdx`。アドレス `mbAddr` を有するマクロブロックに対するこの配列の要素は `sliceIdx[mbAddr]` として参照される。配列 `sliceIdx` の全ての要素は無規定として初期マークされる。
- レイヤ表現のどのマクロブロックがフィールドマクロブロックであるのか、およびどのマクロブロックがフレームマクロブロックであるのかを規定する `PicSizeInMbs` 個の要素をもつ 1 次元配列 `fieldMbFlag`。アドレス `mbAddr` を有するマクロブロックに対するこの配列の要素は `fieldMbFlag[mbAddr]` として参照される。配列 `fieldMbFlag` の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックに対する輝度および `ChromArrayType` が 3 に等しい場合は色差変換タイプを規定する `PicSizeInMbs` 個の要素をもつ 1 次元配列 `cTrafo`。アドレス `mbAddr` を有するマクロブロックに対するこの配列の要素は `cTrafo[mbAddr]` として参照される。無規定としてマークされていない限り、`cTrafo` の各々の要素は `T_4x4`、`T_8x8`、`T_16x16`、あるいは `T_PCM` と等しい。配列 `cTrafo` の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する `PicSizeInMbs` 個の要素をもつ 1 次元配列 `mbType`。アドレス `mbAddr` を有するマクロブロックに対するこの配列の要素は `mbType[mbAddr]` として参照される。無規定としてマークされていない限り、`mbType` の各々の要素は `I_4x4`、`I_8x8`、`I_16x16`、`I_PCM`、`I_BL`、あるいは表 7-13/JT-H264 および表 7-14/JT-H264 において規定されている `INTER` マクロブロックタイプのうちのひとつに等しい。配列 `mbType` の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックに対するサブマクロブロックタイプを規定する  $(\text{PicSizeInMbs}) \times 4$  配列 `subMbType`。アドレス `mbAddr` およびマクロブロックパーティションインデックス `mbPartIdx` を有するマクロブロックに対するこの配列の要素は `subMbType[mbAddr][mbPartIdx]` として参照される。アドレス `mbAddr` を有するマクロブロックのマクロブロックパーティションに対するサブマクロブロックタイプを規定する 1 次元配列が `subMbType[mbAddr]` として参照される。無規定としてマークされていない限り、`subMbType` の各々の要素は表 7-17/JT-H264 および表 7-18/JT-H264 において規定されているサブマクロブロックタイプのうちのひとつに等しい。配列 `subMbType` の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックに対する動きベクトルの数を規定する `PicSizeInMbs` 個の要素をもつ 1 次元配列 `mvCnt`。アドレス `mbAddr` を有するマクロブロックに対するこの配列の要素は `mvCnt[mbAddr]` として参照される。配列 `mvCnt` の全ての要素は 0 に等しく初期設定される。

- レイヤ表現のマクロブロックに対する輝度量子化パラメータを規定する PicSizeInMbs 個の要素をもつ 1 次元配列 tQP<sub>Y</sub>。アドレス mbAddr を有するマクロブロックに対するこの配列の要素は tQP<sub>Y</sub>[mbAddr]として参照される。配列 tQP<sub>Y</sub>の全ての要素は 0 に等しく初期設定される。
- ChromaArrayType が 0 に等しくない場合、レイヤ表現のマクロブロックに対する Cb および Cr 量子化パラメータをそれぞれ規定する 2 つの PicSizeInMbs 個の要素をもつ 1 次元配列 tQP<sub>Cb</sub> および tQP<sub>Cr</sub>。アドレス mbAddr を有するマクロブロックに対するこれらの配列の要素は CX が Cb あるいは Cr によって置換される tQP<sub>CX</sub>[mbAddr]として参照される。配列 tQP<sub>Cb</sub> および tQP<sub>Cr</sub> の全ての要素は 0 に等しく初期設定される。
- レイヤ表現のマクロブロックに対する Intra\_4x4 予測モードを規定する(PicSizeInMbs)×16 配列 ipred4x4。アドレス mbAddr を有するマクロブロックおよびインデックス c4x4BlkIdx を有する 4×4 ブロックに対するこの配列の要素は ipred4x4[mbAddr][c4x4BlkIdx]として参照される。アドレス mbAddr を有するマクロブロックの 4×4 ブロックに対する Intra\_4x4 予測モードを規定する 1 次元配列が ipred4x4[mbAddr]として参照される。配列 ipred4x4 の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックに対する Intra\_8x8 予測モードを規定する(PicSizeInMbs)×4 配列 ipred8x8。アドレス mbAddr を有するマクロブロックおよびインデックス c8x8BlkIdx を有する 8×8 ブロックに対するこの配列の要素は ipred8x8[mbAddr][c8x8BlkIdx]として参照される。アドレス mbAddr を有するマクロブロックの 8×8 ブロックに対する Intra\_8x8 予測モードを規定する 1 次元配列が ipred8x8[mbAddr]として参照される。配列 ipred8x8 の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックの Intra\_16x16 予測モードを規定する PicSizeInMbs 個の要素をもつ 1 次元配列 ipred16x16。アドレス mbAddr を有するマクロブロックに対するこの配列の要素は ipred16x16[mbAddr]として参照される。配列 ipred16x16 の全ての要素は無規定として初期マークされる。
- ChromaArrayType が 1 あるいは 2 に等しい場合、レイヤ表現のマクロブロックに対する INTRA 色差予測モードを規定する PicSizeInMbs 個の要素をもつ 1 次元配列 ipredChroma。アドレス mbAddr を有するマクロブロックに対するこの配列の要素は ipredChroma[mbAddr]として参照される。配列 ipredChroma の全ての要素は無規定として初期マークされる。
- レイヤ表現のマクロブロックに対する予測利用フラグを規定する 2 つの(PicSizeInMbs)×4 配列 predFlagL0 および predFlagL1。アドレス mbAddr およびマクロブロックパーティションインデックス mbPartIdx を有するマクロブロックに対するこれらの配列の要素は X が 0 あるいは 1 によって置換される predFlagLX[mbAddr][mbPartIdx]として参照される。アドレス mbAddr を有するマクロブロックのマクロブロックパーティションに対する予測利用フラグを規定する 1 次元配列が X が 0 あるいは 1 によって置換される predFlagLX[mbAddr]として参照される。配列 predFlagL0 および predFlagL1 の全ての要素は 0 に等しく初期設定される。
- レイヤ表現のマクロブロックに対する参照インデックスを規定する 2 つの(PicSizeInMbs)×4 配列 refIdxL0 および refIdxL1。アドレス mbAddr およびマクロブロックパーティションインデックス mbPartIdx を有するマクロブロックに対するこれらの配列の要素は X が 0 あるいは 1 によって置換される refIdxLX[mbAddr][mbPartIdx]として参照される。アドレス mbAddr を有するマクロブロックのマクロブロックパーティションに対する参照インデックスを規定する 1 次元配列が X が 0 あるいは 1 によって置換される refIdxLX[mbAddr]として参照される。配列 refIdxL0 および refIdxL1 の全ての要素は-1 に等しく初期設定される。

- レイヤ表現のマクロブロックに対する動きベクトル成分を規定する2つの(PicSizeInMbs)×4×4×2配列 mvL0 および mvL1。アドレス mbAddr、マクロブロックパーティションインデックス mbPartIdx、サブマクロブロックパーティションインデックス subMbPartIdx、および動きベクトル成分インデックス c を有するマクロブロックに対するこれらの配列要素は X が 0 あるいは 1 によって置換される mvLX[mbAddr][mbPartIdx][subMbPartIdx][c]として参照される。マクロブロック mbAddr 内部のマクロブロックパーティション mbPartIdx のサブマクロブロックパーティション subMbPartIdx に対する動きベクトルを表現する 2 個の要素を伴う 1 次元配列が X が 0 あるいは 1 によって置換される mvLX[mbAddr][mbPartIdx][subMbPartIdx]として参照される。マクロブロック mbAddr 内部のマクロブロックパーティション mbPartIdx に対する動きベクトルを表現する 4×2 配列が X が 0 あるいは 1 によって置換される mvLX[mbAddr][mbPartIdx]として参照される。マクロブロック mbAddr に対する動きベクトルを表現する 4×2×2 配列が X が 0 あるいは 1 によって置換される mvLX[mbAddr]として参照される。サブマクロブロックパーティションに分割されていないマクロブロック mbAddr のマクロブロックパーティション mbPartIdx に対する成分インデックス c を有する動きベクトル成分は mvLX[mbAddr][mbPartIdx][0][c] と同一の X が 0 あるいは 1 によって置換される mvLX[mbAddr][mbPartIdx][c]としても参照され得る。サブマクロブロックパーティションに分割されていないマクロブロック mbAddr のマクロブロックパーティション mbPartIdx に対する動きベクトルは mvLX[mbAddr][mbPartIdx][0]と同一の X が 0 あるいは 1 によって置換される mvLX[mbAddr][mbPartIdx]としても参照され得る。配列 mvL0 および mvL1 の全ての要素は 0 に等しく初期設定される。
- レイヤ表現のマクロブロックに対する変換係数レベル値を規定する (PicSizeInMbs) × (256+2\*MbWidthC\*MbHeightC)配列 tCoeffLevel。アドレス mbAddr および変換係数インデックス tCoeffIdx を有するマクロブロックに対するこの配列の要素は tCoeffLevel[mbAddr][tCoeffIdx]として参照される。アドレス mbAddr を有するマクロブロックに対する変換係数レベル値を規定する 1 次元配列が tCoeffLevel[mbAddr]として参照される。配列 tCoeffLevel の全ての要素は 0 に等しく初期設定される。
- レイヤ表現のマクロブロックに対するスケールされた変換係数値を規定する (PicSizeInMbs)×(256+2\*MbWidthC\*MbHeightC)配列 sTCoeff。アドレス mbAddr および変換係数インデックス tCoeffIdx を有するマクロブロックに対するこの配列の要素は sTCoeff[mbAddr][tCoeffIdx]として参照される。アドレス mbAddr を有するマクロブロックに対するスケールされた変換係数を規定する 1 次元配列が sTCoeff[mbAddr]として参照される。配列 sTCoeff の全ての要素は 0 に等しく初期設定される。
- レイヤピクチャに対する残差輝度サンプル値を規定する(PicWidthInSamples<sub>L</sub>)×(PicHeightInSamples<sub>L</sub>)配列 rS<sub>L</sub>。アドレス 0 を有するマクロブロックの左上輝度サンプル相対の輝度位置(x,y)に対するこの配列の要素は rS<sub>L</sub>[x,y]として参照される。配列 rS<sub>L</sub>の全ての要素は 0 に等しく初期設定される。
- ChromaArrayType が 0 に等しくない場合、レイヤピクチャに対する残差色差サンプル値を規定する2つの(PicWidthInSamples<sub>C</sub>)×(PicHeightInSamples<sub>C</sub>)配列 rS<sub>Cb</sub> および rS<sub>Cr</sub>。アドレス 0 を有するマクロブロックの左上色差サンプル相対の色差位置(x,y)に対するこれらの配列の要素は CX が Cb あるいは Cr によって置換される rS<sub>CX</sub>[x,y]として参照される。配列 rS<sub>Cb</sub> および rS<sub>Cr</sub>の全ての要素は 0 に等しく初期設定される。
- レイヤピクチャに対する構成された輝度サンプルを規定する (PicWidthInSamples<sub>L</sub>)×(PicHeightInSamples<sub>L</sub>)配列 cS<sub>L</sub>。アドレス 0 を有するマクロブロックの左上輝度サンプル相対の輝度位置(x,y)に対するこの配列の要素は cS<sub>L</sub>[x,y]として参照される。配列 cS<sub>L</sub>の全ての要素は 0 に等しく初期設定される。

- ChromaArrayType が 0 に等しくない場合、レイヤピクチャに対する構成された色差サンプルを規定する 2つの(PicWidthInSamples<sub>C</sub>)×(PicHeightInSamples<sub>C</sub>)配列 cS<sub>Cb</sub> および cS<sub>Cr</sub>。アドレス 0 を有するマクロブロックの左上色差サンプル相対の色差位置(x,y)に対するこの配列の要素は CX が Cb あるいは Cr によって置換される cS<sub>Cx</sub>[x,y]として参照される。配列 cS<sub>Cb</sub> および cS<sub>Cr</sub> は 0 に等しく初期設定される。

#### G.8.1.2.2 配列の再構成処理

この処理は no\_inter\_layer\_pred\_flag が 0 に等しく、SpatialResolutionChangeFlag が 0 に等しく、かつ変数 ScaledRefLayerLeftOffset、ScaledRefLayerRightOffset、ScaledRefLayerTopOffset、あるいは ScaledRefLayerBottomOffset のいずれかが 0 に等しくない場合にのみ起動される。

この処理への入力 currentVars として総称的に参照される配列の組である。

この処理の出力は配列要素の順序付けとともに配列のサイズに関連する修正を伴う currentVars として総称的に参照される配列の組である。

総称 currentVars に割り当てられている変数は総称 refLayerVars に割り当てられる。

G.8.1.2.1 小小節に規定される配列の割り当ておよび初期化処理が起動され、その出力は currentVars として総称的に参照される配列の組である。

変数 xOffset、yOffset、xOffsetC、および yOffsetC が以下の等式によって導出される。

$$xOffset = ScaledRefLayerLeftOffset \quad (G-79)$$

$$yOffset = ScaledRefLayerTopOffset / (1 + field\_pic\_flag) \quad (G-80)$$

$$xOffsetC = (xOffset >> 4) * MbWidthC \quad (G-81)$$

$$yOffsetC = (yOffset >> 4) * MbHeightC \quad (G-82)$$

マクロブロックアドレス mbAddr が値 0..(PicSizeInMbs-1)に渡って進行し、以下の順序付けられたステップが規定される。

1. (1+MbaffFrameFlag)に等しく設定された eS を用いて、変数 refMbX および refMbY が以下の等式によって導出される。

$$refMbX = ((mbAddr / eS) \% PicWidthInMbs) - (xOffset / 16) \quad (G-83)$$

$$refMbY = ((mbAddr / eS) / PicWidthInMbs) * eS + (mbAddr \% eS) - (yOffset / 16) \quad (G-84)$$

2. 参照レイヤマクロブロックアドレス refMbAddr が以下のように導出される。

- もし以下の条件のいずれかが真であるならば、refMbAddr は利用不可としてマークされる。
  - refMbX が 0 よりも小さい、あるいは refMbX が RefLayerPicWidthInMbs よりも大きいか等しい
  - refMbY が 0 よりも小さい、あるいは refMbY が RefLayerPicHeightInMbs よりも大きいか等しい
- それ以外、(1+RefLayerMbaffFrameFlag)に等しく設定された bS を用いて、refMbAddr が以下の等式によって導出される。

$$refMbAddr = (refMbY / bS) * bS * RefLayerPicWidthInMbs + (refMbY \% bS) + refMbX \quad (G-85)$$

3. refMbAddr が利用可能である場合、sliceIdx、fieldMbFlag、cTrafo、mbType、subMbType、mvCnt、tQP<sub>Y</sub>、tQP<sub>Cb</sub> (ChromaArrayType が 0 に等しくない場合)、tQP<sub>Cb</sub> (ChromaArrayType が 0 に等しくない場合)、ipred4x4、ipred8x8、ipred16x16、ipredChroma (ChromaArrayType が 1 あるいは 2 に等しい場合)、predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1、tCoeffLevel、および sTCoeff によって置換される X に対して、総称 currentVars の配列 X を表現する currArray および総称 refLayerVars の配列 X を表現する refLayerArray を用いて、スカラーあるいは配列であり得る配列要素 currArray[mbAddr] が配列要素 refLayerArray[refMbAddr] に等しく設定される。

rS<sub>Y</sub> および cS<sub>Y</sub> によって置換される X に対して、総称 currentVars の配列 X を表現する currArray および総称 refLayerArray の配列 X を表現する refLayerArray を用いて、配列 currArray が以下の等式によって修正される。

$$\text{currArray}[x,y]=\text{refLayerArray}[x-x\text{Offset},y-y\text{Offset}] \quad (\text{G-86})$$

但し、 $x=\text{Max}(0,x\text{Offset}).. \text{Min}(\text{PicWidthInSamples}_L,\text{RefLayerPicWidthInSamples}_L-x\text{Offset})$ 、 $y=\text{Max}(0,y\text{Offset}).. \text{Min}(\text{PicHeightInSamples}_L,\text{RefLayerPicHeightInSamples}_L-y\text{Offset})$

ChromaArrayType が 0 に等しくない場合、rS<sub>Cb</sub>、rS<sub>Cr</sub>、cS<sub>Cb</sub>、および cS<sub>Cr</sub> によって置換される X に対して、総称 currentVars の配列 X を表現する currArray および総称 refLayerVars の配列 X を表現する refLayerArray を用いて、配列 currArray が以下の等式によって修正される。

$$\text{currArray}[x,y]=\text{refLayerArray}[x-x\text{OffsetC},y-y\text{OffsetC}] \quad (\text{G-87})$$

但し、 $x=\text{Max}(0,x\text{OffsetC}).. \text{Min}(\text{PicWidthInSamples}_C,\text{RefLayerPicWidthInSamples}_C-x\text{OffsetC})$ 、 $y=\text{Max}(0,y\text{OffsetC}).. \text{Min}(\text{PicHeightInSamples}_C,\text{RefLayerPicHeightInSamples}_C-y\text{OffsetC})$

### G.8.1.3 レイヤ表現の復号処理

G.8.1.3.1 小小小節は解像度変更を伴わないレイヤ表現に対するベース復号処理を規定する。

G.8.1.3.2 小小小節は解像度変更を伴うレイヤ表現に対するベース復号処理を規定する。

G.8.1.3.3 小小小節はターゲットレイヤ表現の復号処理を規定する。

#### G.8.1.3.1 解像度変更を伴わないレイヤ表現に対するベース復号処理

以下がこの処理の入力である。

- 現レイヤ表現を規定する変数 currDQId
- currentVars として総称的に参照される配列の組

この処理の出力は修正された currentVars として総称的に参照される配列の組である。

この処理は currDQId に等しい DQId を有する現レイヤ表現に対するシンタックス要素および導出された大文字変数を使用して currentVars に割り当てられた変数を修正する。

別に記述されない限り、この小小小節において規定される処理によって参照されるシンタックス要素および導出された大文字変数およびこの処理から起動される全ての子処理は currDQId に等しい DQId を有する現レイヤ表現に対するシンタックス要素および導出された大文字変数である。

解像度変更を伴わないレイヤ表現に対するベース復号処理が以下の順序付けられたステップを進行する。

1. 変数 `noInterLayerPredFlag` が `currDQId` に等しい `DQId` を有するレイヤ表現のスライスの `no_inter_layer_pred_flag` の最小値に等しく設定される。
2. `noInterLayerPredFlag` に依存して、以下が適用される。
  - もし `noInterLayerPredFlag` が 1 に等しいならば、G.8.1.2.1 小小小節に規定される配列の割り当ておよび初期化処理が起動され、出力は修正された `currentVars` として総称的に参照される配列の組である。
  - それ以外 (`noInterLayerPredFlag` が 0 に等しい)、以下の順序付けられたステップが規定される。
    - a. `tcoeff_level_prediction_flag` が 0 に等しい場合、マクロブロックアドレス `mbAddr` は値 `0..(RefLayerPicSizeInMbs-1)` に渡って進行し、各々のマクロブロックアドレス `mbAddr` に対して、G.8.1.5.4 小小小節に規定される解像度変更を伴わないレイヤ表現と 0 に等しい `tcoeff_level_prediction_flag` の復号前のマクロブロック復号処理が入力として `ref_layer_dq_id`、`mbAddr`、および `currentVars` を用いて起動され、出力は `currentVars` の修正版である。
    - b. 変数 `ScaledRefLayerLeftOffset`、`ScaledRefLayerRightOffset`、`ScaledRefLayerTopOffset`、あるいは `ScaledRefLayerBottomOffset` のいずれかが 0 に等しくない場合、G.8.1.2.2 小小小節に規定される配列の再構成処理が入力として `currentVars` を用いて起動され、出力は `currentVars` の修正版である。
3. `currDQId` に等しい `DQId` を有する現レイヤ表現の全てのスライスの集合を `setOfSlices` とする。集合 `setOfSlices` の各々のスライスに対して、G.8.1.4.1 小小小節に規定される解像度変更を伴わないスライスに対するベース復号処理が入力として現在処理されているスライスを表示する `currSlice`、`currDQId` および `currentVars` を用いて起動され、出力は `currentVars` の修正版である。
4. `currDQId` が (`DependencyIdMax`<<4) よりも小さいかあるいは等しい場合、`sliceIdx` を総称 `currentVars` の配列 `sliceIdx` であるとする、ビットストリームは、`mbAddr=0..(PicSizeInMbs-1)` である (`sliceIdx[mbAddr]&127`) のいずれかの値が `currDQId` に等しくないような結果となるデータを含んではならない。

記 この制約およびG.8.1.3.2小小小節における類似の制約は0に等しい`quality_id`を有する全てのレイヤ表現およびレイヤ間予測のために利用されている全てのレイヤ表現がアクセスユニットのスライスによって完全に覆われなければならないことを規定する。`DependencyIdMax`に等しい`dependency_id`および0よりも大きい`quality_id`を有するレイヤ表現に対する追加の制約がG.8.1.5.1小小小節において規定される。

### G.8.1.3.2 解像度変更を伴うレイヤ表現のためのベース復号処理

この処理への入力は以下の通りである。

- 現レイヤ表現を規定する変数 `currDQId`
- `currentVars` として総称的に参照される配列の組

この処理の出力は以下の通りである。

- `refLayerVars` として総称的に参照される配列の組

- 修正された `currentVars` として総称的に参照される配列の組

この処理は `currDQId` に等しい `DQId` をもつ現レイヤ表現にシンタックス要素と導出される大文字変数を使用することで `currentVars` に割り当てられた変数を修正する。

別に記述されない限り、本小小小節により規定される処理とこの処理により起動される全ての子処理によって参照されるシンタックス要素と導出された大文字変数は `currDQId` に等しい `DQId` をもつ現レイヤ表現のためのシンタックス要素と導出される大文字変数である。

解像度変更を伴うレイヤ表現のためのベース復号処理は下記の順序付けられたステップに従って進む。

1. マクロブロックアドレス `mbAddr` は値  $0..(\text{RefLayerPicSizeInMbs}-1)$  上を進行し、それぞれのマクロブロックアドレス `mbAddr` に対し、G.8.1.5.5 小小小節により規定される解像度変更に先行するマクロブロック復号処理が `ref_layer_dq_id` と `mbAddr` および `currentVars` を入力として起動され、出力は `currentVars` の修正版である。
2. G.8.7.1 小小小節により規定される `Intra_Base` 予測のためのデブロックフィルタ処理が (`ref_layer_dq_id`>>4) に等しく設定された `dependencyIdRefLayer` および `currentVars` を入力として起動され、出力は `currentVars` の修正版である。
3. 総称 `currentVars` に割り当てられている変数は総称 `refLayerVars` に割り当てられる。
4. G.8.1.2.1 小小小節により規定される配列の割り当ておよび初期化処理が起動され、出力は総称 `currentVars` に割り当てられる。
5. `setOfSlices` を `currDQId` に等しい `DQId` をもつ現レイヤ表現の全てのスライスの集合とする。集合 `setOfSlices` のそれぞれのスライスに対し、G.8.1.4.2 小小小節により規定される解像度変更を伴うスライスのベース復号処理が、現在処理されているスライスを表示する `currSlice`、`currDQId`、`refLayerVars` および `currentVars` を入力として起動され、出力は `currentVars` の修正版である。
6. `sliceIdc` を総称 `currentVars` の配列 `sliceIdc` であるとして、ビットストリームは `currDQId` に等しくない任意の (`sliceIdc[mbAddr]&127`) 値、但し `mbAddr=0..(\text{PicSizeInMbs}-1)`、を結果として生じるデータを含んではならない。

記 この制約とG.8.1.3.1小小小節にある類似の制約は、0に等しい`quality_id`をもつ全てのレイヤ表現とレイヤ間予測に用いられる全てのレイヤ表現はアクセスユニットのスライスに完全に含まれなければならないことを規定する。`DependencyIdMax`に等しい`dependency_id`および0より大きい`quality_id`をもつレイヤ表現に対する追加制約はG.8.1.5.1小小小節に規定される。

### G.8.1.3.3 ターゲットレイヤ表現復号処理

この処理への入力以下の通りである。

- 現レイヤ表現を規定する変数 `currDQId`
- `refLayerVars` として総称的に参照される配列の組
- `currentVars` として総称的に参照される配列の組

この処理の出力は以下の通りである。

- 輝度サンプリング値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  の配列 `sL`

- ChromaArrayType が 0 でない時、色差サンプリング値を含む 2 つの  $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$  配列である  $s_{Cb}$  と  $s_{Cr}$

この処理では構成されたサンプリング値の配列  $s_L$  と、ChromaArrayType が 0 でない時の  $s_{Cb}$  と  $s_{Cr}$  は、currentVars に割り当てられる変数により導出される。

別に記述されない限り、本小小節で規定される処理とこの処理により起動される全ての子処理から参照されるシンタックス要素と導出された大文字変数は、currDQId に等しい DQId をもつ現レイヤ表現に対するシンタックス要素と導出された大文字変数である。

ターゲットレイヤ表現の復号処理は下記の順序付けられたステップに従って進む。

1. 総称 currentVars に割り当てられる変数は総称 tempVars に割り当てられ、本小小節の以降の部分で tempVars として総称的に参照される配列は G.8.1.2.1 小小節により規定される名前により参照される。

記1 総称tempVarsに割り当てられる変数のその後のいかなる修正も総称currentVarsに割り当てられる変数に影響を与えない。

2. 参照ピクチャリスト refPicList0 と refPicList1 は利用不可として初期マークされる。
3. マクロブロックアドレス mbAddr は値  $0..(\text{PicSizeInMbs}-1)$  上を進行し、それぞれのマクロブロックアドレス mbAddr に対して、下記の順序付けられたステップが規定される。

a. currSlice が  $((\text{sliceIdc}[\text{mbAddr}] \& 127) \gg 4) \ll 4$  に等しい DQId および  $(\text{sliceIdc}[\text{mbAddr}] \gg 7)$  に等しい first\_mb\_in\_slice をもつスライスを規定するとする。

b. firstMbInSlice と sliceType をスライス currSlice のシンタックス要素 first\_mb\_in\_slice と slice\_type とする。

c. mbAddr が firstMbInSlice に等しい時、参照ピクチャリスト refPicList0 と refPicList1 は利用不可とマークされる。

d. mbAddr が firstMbInSlice に等しく  $(\text{sliceType} \% 5)$  が 2 未満である時、G.8.2.3 小小節で規定される参照ピクチャリスト構築のための SVC 復号処理は dependency\_id、use\_ref\_base\_pic\_flag、および現スライスを入力として起動され、出力は修正された参照ピクチャリスト refPicList0 と、 $(\text{sliceType} \% 5)$  が 1 に等しい時は修正された参照ピクチャリスト refPicList1 である。

記2 参照ピクチャリスト refPicList0 と refPicList1 は DependencyIdMax に等しい dependency\_id および 0 に等しい quality\_id をもつレイヤ表現のスライスのためにだけ構築される。DependencyIdMax に等しい dependency\_id および 0 より大きい quality\_id をもつスライスに対しては、参照ピクチャリストは推定される。

e. G.8.1.5.6 小小節により規定されるターゲットマクロブロック復号処理は currDQId、mbAddr、refLayerVars、tempVars、refPicList0(利用可能な時)、refPicList1(利用可能な時)を入力として起動され、出力は tempVars の修正版である。

記3 ターゲットレイヤ表現復号処理が1に等しいstore\_ref\_base\_pic\_flagをもつピクチャに対して2回起動されたとしても、それぞれのマクロブロックに対して唯1回の動き補償操作が必要とされる。

4. G.8.7.2 小小節により規定されるターゲット表現のためのデブロックフィルタ処理は tempVars を入力として起動され、出力は tempVars の修正版である。

5. 全ての配列  $s_L$  のサンプル値は本小小節の出力である配列  $s_L$  に複写される。

6. ChromaArrayType が 0 に等しくない時、 $S_{Cb}$  と  $S_{Cr}$  の全てのサンプル値は本小小小節の出力である配列  $s_{Cb}$  と  $s_{Cr}$  にそれぞれ複写される。

#### G.8.1.4 スライス復号処理

G.8.1.4.1 小小小節は解像度変更を伴わないスライスのベース復号処理を規定する。

G.8.1.4.2 小小小節は解像度変更を伴うスライスのベース復号処理を規定する。

##### G.8.1.4.1 解像度変更を伴わないスライスのベース復号処理

この処理への入力は以下の通りである。

- 現スライス currSlice
- 現レイヤ表現を規定する変数 currDQId
- currentVars として総称的に参照される配列の組

この処理の出力は修正された currentVars として総称的に参照される配列の組である。

この処理は現スライス currSlice のためのシンタックス要素と導出された大文字変数を用いて currentVars に割り当てられた変数を修正する。

別に記述されない限り、本小小小節で規定される処理とこの処理により起動される全ての子処理により参照されるシンタックス要素と導出された大文字変数は、現スライス currSlice のスライスヘッダ、現スライス currSlice のスライスヘッダ内のシンタックス要素 pic\_parameter\_set\_id により識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 seq\_parameter\_set\_id により識別される現シーケンスパラメータに対するシンタックス要素と導出された大文字変数である。

currDQId が 0 に等しくかつ(slice\_type%5)が 1 に等しい時、G.8.2.3 小小小節で規定される参照ピクチャリスト構築のための SVC 復号処理は、0 に等しい currDependencyId と、use\_ref\_base\_pic\_flag に等しい useRefBasePicFlag と、現スライスを入力として起動され、出力は参照ピクチャリスト refPicList1 である。

現スライス currSlice のマクロブロックはマクロブロックアドレスの昇順で処理される。それぞれのマクロブロックアドレス mbAddr におけるマクロブロックに対して、G.8.1.5.2 小小小節により規定される解像度変更を伴わないスライス内のマクロブロックのベース復号処理は、currDQId、mbAddr、currentVars、および currDQId が 0 に等しくかつ(slice\_type%5)が 1 に等しい時は参照ピクチャリスト PicList1 を入力として起動され、出力は currentVars の修正版である。

##### G.8.1.4.2 解像度変更を伴うスライスのためのベース復号処理

この処理への入力は以下の通りである。

- 現スライス currSlice
- 現レイヤ表現を規定する currDQId
- refLayerVars として総称的に参照される配列の組
- currentVars として総称的に参照される配列の組

この処理の出力は修正された `currentVars` として総称的に参照される配列の組である。

この処理は現スライス `currSlice` のためのシンタックス要素と導出された大文字変数を使用して `currentVars` に割り当てられた変数を修正する。

別に記述されない限り、本小小節で規定される処理とこの処理により起動される全ての子処理により参照されるシンタックス要素と導出された大文字変数は、現スライス `currSlice` のスライスヘッダ、現スライス `currSlice` のスライスヘッダ内のシンタックス要素 `pic_parameter_set_id` により識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 `seq_parameter_set_id` により識別される現シーケンスパラメータに対するシンタックス要素と導出された大文字変数である。

`CroppingChangeFlag` が 1 に等しく (`slice_type%5`) が 2 未満である時、G.8.2.3 小小節で規定される参照ピクチャリスト構築のための SVC 復号処理は、`dependency_id`、`use_ref_base_pic_flag`、および現スライスを入力として起動され、出力は参照ピクチャリスト `refPicList0` と、(`sliceType%5`) が 1 に等しい時は参照ピクチャリスト `refPicList1` である。

現スライス `currSlice` のマクロブロックはマクロブロックアドレスの昇順で処理される。それぞれのマクロブロックアドレス `mbAddr` におけるマクロブロックに対して、G.8.1.5.3 小小節により規定される解像度変更を伴うスライス内のマクロブロックのベース復号処理は、`currDQId`、`mbAddr`、`refLayerVars`、`currentVars`、`refPicList0` (`CroppingChangeFlag` が 1 に等しく (`slice_type%5`) が 2 未満の時)、そして `refPicList1` (`CroppingChangeFlag` が 1 に等しく (`slice_type%5`) が 1 に等しい時) を入力として起動され、出力は `currentVars` の修正版である。

#### G.8.1.5 マクロブロック初期化および復号処理

G.8.1.5.1 小小節はマクロブロック初期化処理を規定する。

G.8.1.5.2 小小節は解像度変更を伴わないスライスのマクロブロックに対するベース復号処理を規定する。

G.8.1.5.3 小小節は解像度変更を伴うスライスのマクロブロックに対するベース復号処理を規定する。

G.8.1.5.4 小小節は解像度変更を伴わないレイヤ表現と 0 に等しい `tcoeff_level_prediction_flag` を復号する前のマクロブロック復号処理を規定する。

G.8.1.5.5 小小節は解像度変更前のマクロブロック復号処理を規定する。

G.8.1.5.6 小小節はターゲットマクロブロック復号処理を規定する。

##### G.8.1.5.1 マクロブロック初期化処理

この処理への入力は以下の通りである。

- `refLayerVars` として総称的に参照される配列の組
- `CroppingChangeFlag` が 1 に等しく (`slice_type%5`) が 2 未満である時、参照ピクチャリスト `refPicList0`
- `CroppingChangeFlag` が 1 に等しく (`slice_type%5`) が 1 である時、参照ピクチャリスト `refPicList1`

この処理の出力は以下の通りである。

- 現マクロブロックに対するスライス識別を規定する変数 `sliceIdx`

- 現マクロブロックがフィールドマクロブロックかフレームマクロブロックかを規定する変数 `fieldMbFlag`
- 現マクロブロックに対する変換タイプを規定する変数 `cTrafo`
- 現マクロブロックのマクロブロックタイプを規定する変数 `mbType`
- 現マクロブロックのサブマクロブロックタイプを規定する 4 つの要素を持つ `subMbType` リスト
- 現マクロブロックの動きベクトルカウントに対する初期値を規定する変数 `mvCnt`
- 現マクロブロックの輝度量量子化パラメータを規定する変数 `tQPY`
- `ChromaArrayType` が 0 に等しくない時、現マクロブロックの色差量子化パラメータを規定する 2 つの変数 `tQPCb` と `tQPCr`
- 現マクロブロックの参照インデックスのためのレイヤ間予測因子を規定する 2 つの  $2 \times 2$  配列 `refIdxILPredL0` と `refIdxILPredL1`
- 現マクロブロックの動きベクトル成分のためのレイヤ間予測因子を規定する 2 つの  $4 \times 4 \times 2$  配列 `mvILPredL0` と `mvILPredL1`

この節の中では、`refLayerVars` として総称的に参照される配列 `sliceIdc`、`fieldMbFlag`、`cTrafo`、`mbType`、`subMbType`、`tQPY`、`predFlagL0`、`predFlagL1`、`refIdxL0`、`refIdxL1`、`mvL0`、`mvL1`、`sTCoeff` はそれぞれ `refLayerSliceIdc`、`refLayerFieldMbFlag`、`refLayerCTrafo`、`refLayerMbType`、`refLayerSubMbType`、`refLayerQPY`、`refLayerPredFlagL0`、`refLayerPredFlagL1`、`refLayerRefIdxL0`、`refLayerRefIdxL1`、`refLayerMvL0`、`refLayerMvL1`、`refLayerSTCcoeff` として参照される。

変数 `sliceIdc` は  $((\text{first\_mb\_in\_slice} \ll 7) + \text{DQId})$  に等しく設定される。

変数 `fieldMbFlag` は下記のように導出される。

- もし `field_pic_flag` が 1 に等しいならば、`fieldMbFlag` は 1 に等しく設定される。
- それ以外、もし `SpatialResolutionChangeFlag` が 0 に等しく `slice_skip_flag` が 1 に等しいならば、`fieldMbFlag` は `refLayerFieldMbFlag[CurrMbAddr]` に等しく設定される。
- それ以外、`fieldMbFlag` は `mb_field_decoding_flag` に等しく設定される。

G.8.1.5.1.1 小小小節で規定されるマクロブロックタイプ、サブマクロブロックタイプ、参照インデックスと動きベクトルのためのレイヤ間予測因子の導出処理は `fieldMbFlag`、`refLayerFieldMbFlag`、`refLayerMbType`、`refLayerSubMbType`、`refLayerPredFlagL0`、`refLayerPredFlagL1`、`refLayerRefIdxL0`、`refLayerRefIdxL1`、`refLayerMvL0`、`refLayerMvL1`、`refPicList0`(利用可能な時)、そして `refPicList1`(利用可能な時)を入力として起動され、出力は変数 `mbType`、リスト `subMbType`、 $2 \times 2$  配列 `refIdxILPredL0` と `refIdxILPredL1`、 $4 \times 4 \times 2$  配列 `mvILPredL0` と `mvILPredL1` である。

G.8.1.5.1.2 小小小節により規定される量子化パラメータと変換タイプの導出処理は、`mbType`、`subMbType`、`refLayerMbType`、`refLayerCTrafo`、`refLayerQPY`、および `refLayerSTCcoeff` を入力として起動され、出力は `cTrafo`、`tQPY`、`ChromaArrayType` が 0 に等しくない時 `tQPCb` と `tQPCr` である。

変数 `mvCnt` は 0 に等しく設定される。

quality\_id が 0 より大きい時、ビットストリームは (DQId-1) に等しくない (sliceIdcRefLayer[CurrMbAddr]&127)が生じるデータを含んではならない。

no\_inter\_layer\_pred\_flag が 0 に等しく、SpatialResolutionChangeFlag が 0 に等しく、fieldMbFlag が refLayerFieldMbRef[CurrMbAddr]に等しくない時、下記の制約に従わなければならない。

- a) ビットストリームは、1 に等しい base\_mode\_flag か、または 0 と 1 で置き換えられる X と mbPartIdx=0..3 を用いて、1 に等しい motion\_prediction\_flag\_1X[mbPartIdx]をひとつでも生じるようなデータを含んではならない。
- b) residual\_prediction\_flag が 1 に等しい時、ビットストリームは、 $i=0..(255+2*MbWidthC*MbHeightC)$  に対して、0 に等しくない要素 refLayerSTCoeff[CurrMbAddr][i]がひとつでも生じるようなデータを含んではならない。

#### G.8.1.5.1.1 マクロブロックタイプ、サブマクロブロックタイプ、参照インデックスと動きベクトルのためのレイヤ間予測因子の導出処理

この処理への入力は以下の通りである。

- 現マクロブロックがフィールドマクロブロックであるかフレームマクロブロックであるかを規定する変数 fieldMbFlag
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックでどのマクロブロックがフレームマクロブロックかを規定する RefLayerPicSizeInMbs 個の要素をもつ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックのためのマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素をもつ 1 次元配列 refLayerMbType
- 参照レイヤ表現のマクロブロックに対するサブマクロブロックタイプを規定する (RefLayerPicSizeInMbs)×4 配列 refLayerSubMbType
- 参照レイヤ表現のマクロブロックに対する予測リスト利用フラグを規定する 2 つの (RefLayerPicSizeInMbs)×4 配列 refLayerPredFlagL0 と refLayerPredFlagL1
- 参照レイヤ表現のマクロブロックに対する参照インデックスを規定する 2 つの (RefLayerPicSizeInMbs)×4 配列 refLayerRefIdxL0 と refLayerRefIdxL1
- 参照レイヤ表現のマクロブロックに対する動きベクトル成分を規定する 2 つの (RefLayerPicSizeInMbs)×4×4×2 配列 refLayerMvL0 と refLayerMvL1
- CroppingChangeFlag が 1 に等しく (slice\_type%5)が 2 未満の時、参照ピクチャリスト refPicList0
- CroppingChangeFlag が 1 に等しく (slice\_type%5)が 1 に等しい時、参照ピクチャリスト refPicList1

この処理の出力は以下の通りである。

- 現マクロブロックのマクロブロックタイプを規定する変数 mbType
- 現マクロブロックのサブマクロブロックタイプを規定する 4 つの要素を含むリスト subMbType
- 現マクロブロックの参照インデックスに対するレイヤ間予測因子を規定する 2 つの 2×2 配列 refIdxILPredL0 と refIdxILPredL1

- 現マクロブロックの動きベクトル成分に対するレイヤ間予測因子を規定する 2 つの  $4 \times 4 \times 2$  配列 mvILPredL0 と mvILPredL1

変数 mbTypeILPred、リスト subMbTypeILPred、 $2 \times 2$  配列 refIdxILPredL0 と refIdxILPredL1、 $4 \times 4 \times 2$  配列 mvILPredL0 と mvILPredL1 は下記のように導出される。

- もし base\_mode\_flag が 1 に等しい、または 0 と 1 で置き換えられる X と mbPartIdx=0..3 を用いたシンタックス要素 motion\_prediction\_flag\_IX[mbPartIdx]のいずれかが 1 に等しいならば、G.8.6.1 小節により規定されるマクロブロックタイプ、サブマクロブロックタイプ、参照インデックス、および動きベクトルに対するレイヤ間予測因子の導出処理は、fieldMbFlag、refLayerFieldMbFlag、refLayerMbType、refLayerSubMbType、refLayerPredFlagL0、refLayerPredFlagL1、refLayerRefIdxL0、refLayerRefIdxL1、refLayerMvL0、refLayerMvL1、refPicList0(利用可能な時)、そして refPicList1(利用可能な時)を入力として起動され、出力は変数 mbTypeILPred、リスト subMbTypeILPred、 $2 \times 2$  配列 refIdxILPredL0 と refIdxILPredL1、そして  $4 \times 4 \times 2$  配列 mvILPredL0 と mvILPredL1 である。
- それ以外(base\_mode\_flag が 0 に等しく、0 と 1 で置き換えられる X と mbPartIdx=0..3 を用いたシンタックス要素 motion\_prediction\_flag\_IX[mbPartIdx]の全てが 0 に等しいならば)、mbTypeILPred は利用不可とマークされ、リスト subMbTypeILPred において mbPartIdx=0..3 である subMbTypeILPred[mbPartIdx]の全ての要素は利用不可とマークされ、 $2 \times 2$  配列 refIdxILPredL0 と refIdxILPredL1 の全ての要素が-1 に等しく設定され、 $4 \times 4 \times 2$  配列 mvILPredL0 と mvILPredL1 の全ての要素は 0 に等しく設定される。

base\_mode\_flag、mb\_type、transform\_size\_8x8\_flag に依存して変数 mbType は下記のように導出される。

- もし base\_mode\_flag が 1 に等しいならば、mbType は mbTypeILPred に等しく設定される。
- それ以外、もし MbPartPredMode(mb\_type,0)が Intra\_4x4 に等しいならば、mbType は I\_4x4 に等しく設定される。
- それ以外、もし MbPartPredMode(mb\_type,0)が Intra\_8x8 に等しいならば、mbType は I\_8x8 に等しく設定される。
- それ以外、もし MbPartPredMode(mb\_type,0)が Intra\_16x16 に等しいならば、mbType は I\_16x16 に等しく設定される。
- それ以外、もし mb\_type が I\_PCM に等しいならば、mbType は I\_PCM に等しく設定される。
- それ以外(base\_mode\_flag が 0 に等しく mb\_type が P マクロブロックタイプか B マクロブロックタイプかを規定する)、mbType は mb\_type に等しく設定される。

mbType、base\_mode\_flag に依存してリスト subMbType は下記のように導出される。

- もし mbType が P\_8x8 と B\_8x8 とも等しくないならば、mbPartIdx=0..3 である subMbType[mbPartIdx]の全ての要素は無規定としてマークされる。
- それ以外、もし base\_mode\_flag が 1 に等しいならば、mbPartIdx=0..3 である subMbType[mbPartIdx]のそれぞれの要素は subMbTypeILPred[mbPartIdx]に等しく設定される。
- それ以外(mbType が P\_8x8 または B\_8x8 に等しく base\_mode\_flag が 0 に等しい)、mbPartIdx=0..3 である subMbType[mbPartIdx]のそれぞれの要素は sub\_mb\_type[mbPartIdx]に等しく設定される。

slice\_type が EP に等しくかつ base\_mode\_flag が 1 に等しい時、ビットストリームは、mbPartIdx=0..3 とし

て、0 より小さい要素 `refIdxILPredL0[mbPartIdx]`を生じるデータを含んではならない。

1 に等しい `residual_prediction_flag` がビットストリームに存在する時、ビットストリームは、`I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、または `I_BL` に等しい `mbType` を生じるようなデータを含んではならない。

#### G.8.1.5.1.2 量子化パラメータと変換タイプの導出処理

この処理への入力は以下の通りである。

- 現マクロブロックに対するマクロブロックタイプを規定する変数 `mbType`
- 現マクロブロックに対するサブマクロブロックタイプを規定する 4 つの要素を含むリスト `subMbType`
- 参照レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する 1 次元配列 `refLayerMbType`
- 参照レイヤ表現のマクロブロックに対する変換タイプを規定する 1 次元配列 `refLayerCTrafo`
- 参照レイヤ表現のマクロブロックに対する輝度量子化パラメータを規定する 1 次元配列 `refLayerQPY`
- 参照レイヤ表現のマクロブロックに対するスケーリング変換係数値を規定する (`RefLayerPicSizeInMbs`)  
× (`256+2*MbWidthC*MbHeightC`)配列 `refLayerSTCoeff`

この処理の出力は以下の通りである。

- 現マクロブロックに対する変換タイプを規定する変数 `cTrafo`
- 現マクロブロックに対する輝度量子化パラメータを規定する変数 `tQPY`
- `ChromaArrayType` が 0 に等しくない時、現マクロブロックに対する色差量子化パラメータを規定する 2 つの変数 `tQPCb` と `tQPCr`

変数 `tQPY` は次のように導出される。

- もし `SpatialResolutionChangeFlag` が 0 に等しく、`CodedBlockPatternLuma` が 0 に等しく、`CodedBlockPatternChroma` が 0 に等しく、かつ次のいずれかの条件が真ならば、`tQPY` は `refLayerQPY[CurrMbAddr]` に等しく設定される。
  - `mbType` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、または `I_BL` に等しく、`base_mode_flag` が 1 に等しい
  - `mbType` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、または `I_BL` に等しくなく、`residual_prediction_flag` が 1 に等しい
- それ以外、`tQPY` は `QPY` に等しく設定される。

`ChromaArrayType` が 0 に等しくない時、`Cb` と `Cr` によって置き換えられた `CX` に対して、変数 `tQPCX` は 8.5.8 小節で規定されるように `QPY` に対して `tQPY` の値に対応する `QPCX` の値に等しく設定される。

変数 `predCoeffFlag` は次のように導出される。

- もし `SpatialResolutionChangeFlag` が 0 に等しく、かつ次のいずれかの条件が真ならば、`predCoeffFlag` は 1 に等しく設定される。
  - `base_mode_flag` が 1 に等しく、`tcoeff_level_prediction_flag` が 0 に等しく、`refLayerMbType[CurrMbAddr]` が `I_BL` に等しい。

- base\_mode\_flag が 1 に等しく、tcoeff\_level\_prediction\_flag が 1 に等しく、mbType が I\_PCM、I\_16x16、I\_8x8、または I\_4x4 に等しい。
- residual\_prediction\_flag が 1 に等しく、mbType が I\_PCM、I\_16x16、I\_8x8、I\_4x4、または I\_BL に等しくない。
- それ以外、predCoeffFlag は 0 に等しく設定される。

変数 cTrafo は次のように導出される。

- もし mbType が I\_PCM に等しいならば、cTrafo は T\_PCM に等しく設定される。
- それ以外、もし mbType が I\_16x16 に等しいならば、cTrafo は T\_16x16 に等しく設定される。
- それ以外、もし mbType が I\_8x8 に等しいか、または transform\_size\_8x8\_flag が 1 に等しいならば、cTrafo は T\_8x8 に等しく設定される。
- それ以外、もし predCoeffFlag が 1 に等しいならば、cTrafo は cTrafoRefLayer に等しく設定される。
- それ以外(predCoeffFlag が 0 に等しく、transform\_size\_8x8\_flag が 0 に等しく、かつ mbType が I\_PCM、I\_16x16、または I\_8x8 に等しくない)、cTrafo は T\_4x4 に等しく設定される。

base\_mode\_flag が 1 に等しい時、次の制約に従わなければならない。

- a) mbType が P\_8x8 または B\_8x8 に等しく、かつ NumSubMbPart(subMbType[mbPartIdx])が任意の mbPartIdx=0..3 に対して 1 に等しくない時、ビットストリームは 1 に等しい transform\_size\_8x8\_flag を含んではならない。
- b) mbType が I\_PCM に等しい時、ビットストリームは 0 に等しくない CodedBlockPatternLuma、または 0 に等しくない CodedBlockPatternChroma をもたらすデータを含んではならない。
- c) mbType が I\_16x16 または I\_4x4 に等しい時、ビットストリームは 1 に等しい transform\_size\_8x8\_flag を含んではならない。
- d) mbType が I\_8x8 に等しく、かつ transform\_size\_8x8\_flag が 0 に等しい時、ビットストリームは 0 に等しくない CodedBlockPatternLuma をもたらすデータを含んではならない。

predCoeffFlag が 1 に等しい時、次の制約に従わなければならない。

- a) refLayerCTrafo[CurrMbAddr]が T\_8x8 に等しく、かつ transform\_size\_8x8\_flag が 0 に等しい時、ビットストリームは 0 に等しくない CodedBlockPatternLuma をもたらすデータを含んではならない。
- b) refLayerCTrafo[CurrMbAddr]が T\_4x4 に等しく、かつ transform\_size\_8x8\_flag が 1 に等しい時、ビットストリームは  $i=0..(\text{ChromaArrayType}!=3?255:767)$  に対して 0 に等しくない任意の要素 refLayerSTCoeff[CurrMbAddr][i]をもたらすデータを含んではならない。

#### G.8.1.5.2 解像度変更を伴わないスライスでのマクロブロックに対するベース復号処理

この処理への入力には以下の通りである。

- 現レイヤ表現を規定する変数 currDQId
- 現レイヤ表現の中の現マクロブロックを規定する変数 mbAddr

- currentVars として総称的に参照される配列の組
- currDQId が 0 に等しく、かつ(slice\_type%5)が 1 に等しい時、参照ピクチャリスト refPicList1

この処理の出力は、修正された currentVars として総称的に参照される配列の組である。

この処理はそれ自身のアドレス mbAddr とレイヤ表現識別子 currDQId によって規定される、現マクロブロックに対するシンタックス要素と導出される大文字変数を使用して currentVars に割り当てられた変数を変更する。

別に記述されない限り、この小小小節で規定される処理とこの処理から起動される全ての子処理によって参照されるシンタックス要素と導出される大文字変数は、currDQId に等しい DQId を持つレイヤ表現の内部にアドレス mbAddr を持つマクロブロックである現マクロブロック、現マクロブロックを含むスライスである現スライスのスライスヘッダ、現スライスのスライスヘッダ内のシンタックス要素 pic\_parameter\_set\_id によって識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 seq\_parameter\_set\_id によって識別される現シーケンスパラメータに対するシンタックス要素と導出される大文字変数である。

この小小小節の中で、currentVars として総称的に参照される配列は G.8.1.2.1 小小小節で規定されるそれらの名前によって参照される。

解像度変更を伴わないスライスでのマクロブロックに対するベース復号処理は次の順序付けられたステップによって規定される。

1. 変数 CurrMbAddr は mbAddr に等しく設定される。
2. tcoeff\_level\_prediction\_flag が 1 に等しい時、変数 refQP<sub>Y</sub> は tQP<sub>Y</sub>[mbAddr] に等しく設定され、ChromaArrayType が 0 に等しくない時、変数 refQP<sub>Cb</sub> と refQP<sub>Cr</sub> はそれぞれ tQP<sub>Cb</sub>[mbAddr] と tQP<sub>Cr</sub>[mbAddr] に等しく設定される。
3. no\_inter\_layer\_pred\_flag が 0 に等しい時、変数 refLayerIntraBLFlag は次のように導出される。
  - もし mbType[mbAddr] が I\_BL に等しいならば、refLayerIntraBLFlag は 1 に等しく設定される。
  - それ以外(mbType[mbAddr] が I\_BL に等しくない)、refLayerIntraBLFlag は 0 に等しく設定される。
4. G.8.1.5.1 小小小節で規定されるマクロブロック初期化処理が、currentVars に等しく設定される refLayerVars を入力として起動され、出力は sliceIdc[mbAddr]、fieldMbFlag[mbAddr]、cTrafo[mbAddr]、mbType[mbAddr]、subMbType[mbAddr]、mvCnt[mbAddr]、tQP<sub>Y</sub>[mbAddr]、tQP<sub>Cb</sub>[mbAddr](ChromaArrayType が 0 に等しくない時)、tQP<sub>Cr</sub>[mbAddr](ChromaArrayType が 0 に等しくない時)、2×2 配列 refIdxILPredL0 と refIdxILPredL1、および 4×4×2 配列 mvILPredL0 と mvILPredL1 に割り当てられる。
5. G.8.4.1 小小小節で規定される動きベクトル成分と参照インデックスの SVC 導出処理が、sliceIdc、fieldMbFlag、mbType、subMbType、predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1、mvCnt、refIdxILPredL0、refIdxILPredL1、mvILPredL0、mvILPredL1 と refPicList1(利用可能な時)を入力として起動され、出力は predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1 と mvCnt の配列の修正版である。
6. mbType[mbAddr] に依存して、次が適用される

- もし  $mbType[mbAddr]$  が  $I\_PCM$ 、 $I\_16x16$ 、 $I\_8x8$ 、 $I\_4x4$  のいずれかに等しいならば、次の順序付けられたステップが規定される。
  - a.  $base\_mode\_flag$  が 0 に等しい時、G.8.3.1 小小節で規定される INTRA 予測モードの SVC 導出処理が、 $sliceIdx$ 、 $fieldMbFlag$ 、 $mbType$ 、 $ipred4x4$ 、 $ipred8x8$ 、 $ipred16x16$ 、と  $ChromaArrayType$  が 1 または 2 に等しい時の  $ipredChroma$  を入力として起動され、出力は  $ipred4x4$ 、 $ipred8x8$ 、 $ipred16x16$ 、と  $ChromaArrayType$  が 1 または 2 に等しい時の  $ipredChroma$  の修正版である。
  - b.  $base\_mode\_flag$  が 1 に等しい時、G.8.5.2 小小節で規定される変換係数改善の前の変換係数レベルスケーリング処理が、 $cTrafo[mbAddr]$ 、 $tCoeffLevel[mbAddr]$ 、 $tQP_Y[mbAddr]$ 、 $refQP_Y$ 、 $ChromaArrayType$  が 0 に等しくない時の  $tQP_{Cb}[mbAddr]$ 、 $tQP_{Cr}[mbAddr]$ 、 $refQP_{Cb}$ 、および  $refQP_{Cr}$  を入力として起動され、出力は  $tCoeffLevel[mbAddr]$  の修正版である。
  - c. G.8.5.1 小小節で規定される変換係数スケーリングおよび改善処理が、 $base\_mode\_flag$  に等しく設定される  $refinementFlag$ 、 $fieldMbFlag[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$  と  $tCoeffLevel[mbAddr]$  を入力として起動され、出力は  $sTCoeff[mbAddr]$  と  $tCoeffLevel[mbAddr]$  の修正版である。
  - d. G.8.5.5 小小節で規定されるサンプル配列再初期化処理が、 $fieldMbFlag[mbAddr]$ 、 $rS_L$  と、 $ChromaArrayType$  が 0 に等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  を入力として起動され、出力は  $rS_L$  の修正版と、 $ChromaArrayType$  が 0 に等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  の修正版である。
  - e. G.8.5.5 小小節で規定されるサンプル配列再初期化処理が、 $fieldMbFlag[mbAddr]$ 、 $cS_L$  と、 $ChromaArrayType$  が 0 に等しくない時の  $cS_{Cb}$  と  $cS_{Cr}$  を入力として起動され、出力は  $cS_L$  の修正版と、 $ChromaArrayType$  が 0 に等しくない時の  $cS_{Cb}$  と  $cS_{Cr}$  の修正版である。
- それ以外、もし  $mbType[mbAddr]$  が  $I\_BL$  に等しいならば、G.8.5.1 小小節で規定される変換係数スケーリングおよび改善処理が  $refLayerIntraBLFlag$  に等しく設定される  $refinementFlag$ 、 $fieldMbFlag[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$ 、と  $tCoeffLevel[mbAddr]$  を入力として起動され、出力は  $sTCoeff[mbAddr]$  と  $tCoeffLevel[mbAddr]$  の修正版である。
- それ以外( $mbType[mbAddr]$  が  $I\_PCM$ 、 $I\_16x16$ 、 $I\_8x8$ 、 $I\_4x4$ 、 $I\_BL$  のいずれにも等しくない)、次の順序付けられたステップが規定される。
  - a.  $residual\_prediction\_flag$  が 1 に等しい時、G.8.5.2 小小節で規定される変換係数改善の前の変換係数レベルスケーリング処理が、 $cTrafo[mbAddr]$ 、 $tCoeffLevel[mbAddr]$ 、 $tQP_Y[mbAddr]$ 、 $refQP_Y$ 、と、 $ChromaArrayType$  が 0 に等しくない  $tQP_{Cb}[mbAddr]$ 、 $tQP_{Cr}[mbAddr]$ 、 $refQP_{Cb}$ 、と  $refQP_{Cr}$  を入力として起動され、出力は  $tCoeffLevel[mbAddr]$  の修正版である。
  - b. G.8.5.1 小小節で規定される変換係数スケーリングおよび改善処理が、 $residual\_prediction\_flag$  に等しく設定される  $refinementFlag$ 、 $fieldMbFlag[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$ 、と  $tCoeffLevel[mbAddr]$  を入力として起動され、出力は  $sTCoeff[mbAddr]$  と  $tCoeffLevel[mbAddr]$  の修正版である。
  - c.  $residual\_prediction\_flag$  が 0 に等しい時、G.8.5.5 小小節で規定されるサンプル配列再初期化処理が、 $fieldMbFlag[mbAddr]$ 、 $rS_L$ 、と、 $ChromaArrayType$  が 0 に等しくない  $rS_{Cb}$  と  $rS_{Cr}$  を入力として起動され、出力は  $rS_L$  の修正版と、 $ChromaArrayType$  が 0 と等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  の修正版である。

- d. `residual_prediction_flag` が 0 に等しい時、G.8.5.5 小小節で規定されるサンプル配列再初期化処理が、`fieldMbFlag[mbAddr]`、`cSL`、と、`ChromaArrayType` が 0 に等しくない `cSCb` と `cSCr` を入力として起動され、出力は `cSL` の修正版と、`ChromaArrayType` が 0 と等しくない時の `cSCb` と `cSCr` の修正版である。

#### G.8.1.5.3 解像度変更を伴うスライスでのマクロブロックに対するベース復号処理

この処理への入力には以下の通りである。

- 現レイヤ表現を規定する変数 `currDQId`
- 現レイヤ表現中の現マクロブロックを規定する変数 `mbAddr`
- `refLayerVars` として総称的に参照される配列の組
- `currentVars` として総称的に参照される配列の組
- `CroppingChangeFlag` が 1 に等しくかつ(`slice_type%5`)が 2 未満の時の参照ピクチャリスト `refPicList0`
- `CroppingChangeFlag` が 1 に等しくかつ(`slice_type%5`)が 1 に等しい時の参照ピクチャリスト `refPicList1`

この処理の出力は修正された `currentVars` として総称的に参照される配列の組である。

この処理は、`refLayerVars` に割り当てられた変数と同じく、それ自身のアドレス `mbAddr` とレイヤ表現識別子 `currDQId` によって規定される、現マクロブロックに対するシンタックス要素と導出される大文字変数を使用して `currentVars` に割り当てられた変数を修正する。

別に記述されない限り、この小小節で規定される処理とこの処理から起動された全ての子処理によって参照されるシンタックス要素と導出される大文字変数は、`currDQId` に等しい `DQId` を持つレイヤ表現の内部にアドレス `mbAddr` を持つマクロブロックである現マクロブロック、現マクロブロックを含むスライスである現スライスのスライスヘッダ、現スライスのスライスヘッダ内のシンタックス要素 `pic_parameter_set_id` によって識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 `seq_parameter_set_id` によって識別される現シーケンスパラメータに対するシンタックス要素と導出される大文字変数である。

この小小節の中で、`currentVars` として総称的に参照される配列は G.8.1.2.1 小小節で規定されるそれらの名前によって参照される。

この小小節の中で、総称 `refLayerVars` の配列 `sliceIdc`、`fieldMbFlag`、`cTrafo`、`mbType`、`cSL`、`cSCb`、`cSCr`、`rSL`、`rSCb`、および `rSCr` はそれぞれ `refLayerSliceIdc`、`refLayerFieldMbFlag`、`refLayerCTrafo`、`refLayerMbType`、`refSL`、`refSCb`、`refSCr`、`refRL`、`refRCb`、および `refRCr` として参照される。

解像度変更を伴うスライスでのマクロブロックに対するベース復号処理は次の順序付けられたステップによって規定される。

1. 変数 `CurrMbAddr` は `mbAddr` に等しく設定される。
2. G.8.1.5.1 小小節で規定されるマクロブロック初期化処理が `refLayerVars`、`refPicList0`(利用可能な時)、と `refPicList1`(利用可能な時)を入力として起動され、出力は `sliceIdc[mbAddr]`、`fieldMbFlag[mbAddr]`、`cTrafo[mbAddr]`、`mbType[mbAddr]`、`subMbType[mbAddr]`、`mvCnt[mbAddr]`、`tQPY[mbAddr]`、`tQPCb[mbAddr]`(`ChromaArrayType` が 0 に等しくない時)、`tQPCr[mbAddr]`(`ChromaArrayType` が 0 に等しくない時)である。

い時)、 $2 \times 2$  配列 `refIdxILPredL0` と `refIdxILPredL1`、および  $4 \times 4 \times 2$  配列 `mvILPredL0` と `mvILPredL1` に割り当てられる。

3. G.8.4.1 小節で規定される動きベクトル成分と参照インデックスの SVC 導出処理が、`sliceIdc`、`fieldMbFlag`、`mbType`、`subMbType`、`predFlagL0`、`predFlagL1`、`refIdxL0`、`refIdxL1`、`mvL0`、`mvL1`、`mvCnt`、`refIdxILPredL0`、`refIdxILPredL1`、`mvILPredL0`、および `mvILPredL1` を入力として起動され、出力は `predFlagL0`、`predFlagL1`、`refIdxL0`、`refIdxL1`、`mvL0`、`mvL1` と `mvCnt` の配列の修正版である。
4. 変数 `intraResamplingFlag` は次のように導出される。
  - もし次のいずれかの条件が真であるならば、`intraResamplingFlag` は 1 に等しく設定される。
    - `RestrictedSpatialResolutionChangeFlag` が 1 に等しく、かつ `mbType[mbAddr]` が `I_BL` に等しい
    - `RestrictedSpatialResolutionChangeFlag` が 0 に等しく、`MbaffFrameFlag` が 0 に等しく、`RefLayerMbaffFrameFlag` が 0 に等しく、かつ `mbType[mbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しくない
  - それ以外、`intraResamplingFlag` は 0 に等しく設定される。
5. `intraResamplingFlag` が 1 に等しい時、G.8.6.2 小節で規定される INTRA サンプルに対する再サンプリング処理が、`fieldMbFlag[mbAddr]`、`refLayerSliceIdc`、`refLayerFieldMbFlag`、`refLayerMbType`、`refSL`、`cSL`、および、`ChromaArrayType` が 0 に等しくない時、`refSCb`、`refSCr`、`cSCb`、および `cSCr` を入力として起動され、出力は、配列 `cSL` の修正版と、`ChromaArrayType` が 0 に等しくない時は配列 `cSCb` と `cSCr` の修正版である。
6. `mbType[mbAddr]` に依存して、次が適用される。
  - もし `mbType[mbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しいならば、G.8.3.1 小節で規定される INTRA 予測モードの SVC 導出処理が `sliceIdc`、`fieldMbFlag`、`mbType`、`ipred4x4`、`ipred8x8`、`ipred16x16`、および、`ChromaArrayType` が 1 または 2 に等しい時、`ipredChroma` を入力として起動され、出力は `ipred4x4`、`ipred8x8`、`ipred16x16`、および、`ChromaArrayType` が 1 または 2 に等しい時、`ipredChroma` の修正版である。
  - それ以外、もし `mbType[mbAddr]` が `I_BL` に等しくない、かつ `residual_prediction_flag` が 1 に等しいならば、G.8.6.3 小節で規定される残差サンプルに対する再サンプリング処理が、`fieldMbFlag[mbAddr]`、`refLayerFieldMbFlag`、`refLayerCTrafo`、`refRL`、`rSL` と、`ChromaArrayType` が 0 に等しくない時、`refRCb`、`refRCr`、`rSCb`、および `rSCr` を入力として起動され、出力は配列 `rSL` の修正版と、`ChromaArrayType` が 0 に等しくない時の `rSCb` と `rSCr` の修正版である。
  - それ以外、総称の配列 `currentVars` は修正されない。
7. G.8.5.1 小節で規定される変換係数スケールと改善処理が、0 に等しく設定された `refinementFlag`、`fieldMbFlag[mbAddr]`、`cTrafo[mbAddr]`、`sTCoeff[mbAddr]`、と `tCoeffLevel[mbAddr]` を入力として起動され、出力は `sTCoeff[mbAddr]` と `tCoeffLevel[mbAddr]` の修正版である。

#### G.8.1.5.4 解像度変更を伴わないレイヤ表現と 0 に等しい `tcoeff_level_prediction_flag` を復号する前のマクロブロック復号処理

この処理への入力は以下の通りである。

- 現レイヤ表現を規定する変数 `currDQId`
- 現レイヤ表現中の現マクロブロックを規定する変数 `mbAddr`
- `currentVars` として総称的に参照される配列の組

この処理の出力は修正された `currentVars` として総称的に参照される配列の組である。

この処理はそれ自身のアドレス `mbAddr` とレイヤ表現識別子 `currDQId` によって規定される、現マクロブロックに対するシンタックス要素と導出される大文字変数を使用して `currentVars` に割り当てられた変数を修正する。

別に記述されない限り、この小小小節で規定される処理とこの処理から起動された全ての子処理によって参照されるシンタックス要素と導出される大文字変数は、`currDQId` に等しい `DQId` を持つレイヤ表現の内部にアドレス `mbAddr` を持つマクロブロックである現マクロブロック、現マクロブロックを含むスライスである現スライスのスライスヘッダ、現スライスのスライスヘッダ内のシンタックス要素 `pic_parameter_set_id` によって識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 `seq_parameter_set_id` によって識別される現シーケンスパラメータに対するシンタックス要素と導出される大文字変数である。

この小小小節の中で、`currentVars` として総称的に参照される配列は G.8.1.2.1 小小小節で規定されるそれらの名前によって参照される。

ターゲットマクロブロック復号処理は次の順序付けられたステップによって規定される。

1. 変数 `CurrMbAddr` は `mbAddr` に等しく設定される。
2. `mbType[mbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しい時、G.8.3.2 小小小節で規定される `SVC INTRA` サンプル予測と構築処理は、`sliceIdc`、`fieldMbFlag`、`mbType`、`ipred4x4[mbAddr]`、`ipred8x8[mbAddr]`、`ipred16x16[mbAddr]`、`ipredChroma[mbAddr]`、`cTrafo[mbAddr]`、`sTCoeff[mbAddr]`、`cSL`、と `ChromaArrayType` が 0 に等しくない時、`cSCb` と `cSCr` を入力として起動され、出力は配列 `cSL` の修正版と、`ChromaArrayType` が 0 に等しくない時、配列 `cSCb` と `cSCr` の修正版である。

#### G.8.1.5.5 解像度変更前のマクロブロック復号処理

この処理への入力には以下の通りである。

- 現レイヤ表現を規定する変数 `currDQId`
- 現レイヤ表現中の現マクロブロックを規定する変数 `mbAddr`
- `currentVars` として総称的に参照される配列の組

この処理の出力は、修正された `currentVars` として総称的に参照される配列の組である。

この処理はそれ自身のアドレス `mbAddr` とレイヤ表現識別子 `currDQId` によって規定される、現マクロブロックに対するシンタックス要素と導出される大文字変数を使用して `currentVars` に割り当てられた変数を修正する。

別に記述されない限り、この小小小節で規定される処理とこの処理から起動された全ての子処理によって参照されるシンタックス要素と導出される大文字変数は、`currDQId` に等しい `DQId` を持つレイヤ表現の内部

にアドレス `mbAddr` を持つマクロブロックである現マクロブロック、現マクロブロックを含むスライスである現スライスのスライスヘッダ、現スライスのスライスヘッダ内のシンタックス要素 `pic_parameter_set_id` によって識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 `seq_parameter_set_id` によって識別される現シーケンスパラメータに対するシンタックス要素と導出される大文字変数である。

この小小小節の中で、`currentVars` として総称的に参照される配列は G8.1.2.1 小小小節で規定されるそれらの名前によって参照される。

ターゲットマクロブロック復号処理は次の順序付けられたステップによって規定される。

1. 変数 `CurrMbAddr` は `mbAddr` に等しく設定される。
2. `mbType[mbAddr]` に依存して、次が適用される。
  - もし `mbType[mbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しいならば、G8.3.2 小小節で規定される SVC INTRA サンプル予測および構築処理は `sliceIdc`、`fieldMbFlag`、`mbType`、`ipred4x4[mbAddr]`、`ipred8x8[mbAddr]`、`ipred16x16[mbAddr]`、`ipredChroma[mbAddr]`、`cTrafo[mbAddr]`、`sTCoeff[mbAddr]`、`cSL`、と `ChromaArrayType` が 0 に等しくない時、`cSCb` と `cSCr` を入力として起動され、出力は配列 `cSL` の修正版と、`ChromaArrayType` が 0 に等しくない時の配列 `cSCb` と `cSCr` の修正版である。
  - それ以外、もし `mbType[mbAddr]` が `I_BL` に等しいならば、次の順序付けられたステップによって規定される。
    - a. G8.5.3 小小節で規定される残差構築および累算処理が、0 に等しく設定された `accumulationFlag`、`fieldMbFlag[mbAddr]`、`cTrafo[mbAddr]`、`sTCoeff[mbAddr]`、`rSL`、と `ChromaArrayType` が 0 に等しくない時の `rSCb` と `rSCr` を入力として起動され、出力は `rSL` の修正版と、`ChromaArrayType` が 0 に等しくない時、`rSCb` と `rSCr` の修正版である。
    - b. G8.5.4 小小節で規定されるサンプル配列の累算処理が、`fieldMbFlag[mbAddr]`、`rSL`、`cSL`、と `ChromaArrayType` が 0 に等しくない時、`rSCb`、`rSCr`、`cSCb`、および `cSCr` を入力として起動され、出力は `cSL` の修正版と、`ChromaArrayType` が 0 に等しくない時、`cSCb` および `cSCr` の修正版である。
  - それ以外(`mbType[mbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、または `I_BL` に等しくない)、次の順序付けられたステップによって規定される。
    - a. G8.5.3 小小節で規定される残差構築および累算処理が、1 に等しく設定された `accumulationFlag`、`fieldMbFlag[mbAddr]`、`cTrafo[mbAddr]`、`sTCoeff[mbAddr]`、`rSL`、と `ChromaArrayType` が 0 に等しくない時、`rSCb` と `rSCr` を入力として起動され、出力は `rSL` の修正版と、`ChromaArrayType` が 0 に等しくない時、`rSCb` と `rSCr` の修正版である。
    - b. G8.5.5 小小節で規定されるサンプル配列の再初期化処理が、`fieldMbFlag[mbAddr]`、`cSL` と `ChromaArrayType` が 0 に等しくない時、`cSCb` と `cSCr` を入力として起動され、出力は `cSL` の修正版と、`ChromaArrayType` が 0 に等しくない時、`cSCb` と `cSCr` の修正版である。

#### G.8.1.5.6 ターゲットマクロブロック復号処理

この処理への入力は以下の通りである。

- 現レイヤ表現を規定する変数 `currDQId`
- 現レイヤ表現中の現マクロブロックを規定する変数 `mbAddr`
- `refLayerVars` として総称的に参照される配列の組
- `currentVars` として総称的に参照される配列の組
- `(slice_type%5)`が2未満の時の参照ピクチャリスト `refPicList0`
- `(slice_type%5)`が1に等しい時の参照ピクチャリスト `refPicList1`

この処理の出力は修正された `currentVars` として総称的に参照される配列の組である。

この処理はそれ自身のアドレス `mbAddr` とレイヤ表現識別子 `currDQId` によって規定される、現マクロブロックに対するシンタックス要素と導出される大文字変数を使用して `currentVars` に割り当てられた変数を修正する。

別に記述されない限り、この小小小節で規定される処理とこの処理から起動された全ての子処理によって参照されるシンタックス要素と導出される大文字変数は、`currDQId` に等しい `DQId` を持つレイヤ表現の内部にアドレス `mbAddr` を持つマクロブロックである現マクロブロック、現マクロブロックを含むスライスである現スライスのスライスヘッダ、現スライスのスライスヘッダ内のシンタックス要素 `pic_parameter_set_id` によって識別される現ピクチャパラメータ、および現ピクチャパラメータセット内のシンタックス要素 `seq_parameter_set_id` によって識別される現シーケンスパラメータに対するシンタックス要素と導出される大文字変数である。

この小小小節の中で、`currentVars` として総称的に参照される配列は G.8.1.2.1 小小小節で規定されるそれらの名前によって参照される。

この小小小節の中で、配列 `fieldMbFlag` と総称 `refLayerVars` の `mbType` はそれぞれ `refLayerFieldMbFlag` と `refLayerMbType` として参照される。

ターゲットマクロブロック復号処理は次の順序付けられたステップによって規定される。

1. 変数 `CurrMbAddr` は `mbAddr` に等しく設定される。
2. `tcoeff_level_prediction` が 1 に等しく、`(sliceIdc[mbAddr]&127)` は `currDQId` に等しくなく、かつ、`ChromaArrayType` が 0 に等しくない時、次の順序付けられたステップによって規定される。

記 次の順序付けられたステップにより、`currDQId` に等しい `DQId` をもつ現レイヤ表現と `(sliceIdc[mbAddr]&127)` に等しい `DQId` を持つレイヤ表現の色差量子化パラメータオフセットが異なる時、色差に対する変換係数レベルだけがスケーリングされる(そして、スケーリングされた変換係数はそれに応じて導出される)。

- a. 変数 `cQPY` が `tQPY[mbAddr]` に等しく設定され、`Cb` と `Cr` によって置き換えられる `CX` に対して、変数 `cQPCX` は 8.5.8 小節で規定される `QPY` に対する `cQPY` の値に対応する `QPCX` の値に等しく設定される。
- b. G.8.5.2 小小小節で規定される変換係数改善の前の変換係数レベルスケーリング処理が `cTrafo[mbAddr]`、`tCoeffLevel[mbAddr]`、`cQPY` に等しく設定される `tQPY`、`tQPY[mbAddr]` に等しく設定される `refQPY` と、

ChromaArrayType が 0 と等しくない時の  $cQP_{Cb}$  に等しく設定される  $tQP_{Cb}$ 、 $cQP_{Cr}$  に等しく設定される  $tQP_{Cr}$ 、 $tQP_{Cb}[mbAddr]$  に等しく設定される  $refQP_{Cb}$ 、 $tQP_{Cr}[mbAddr]$  に等しく設定される  $refQP_{Cr}$  を入力として起動され、出力は  $tCoeffLevel[mbAddr]$  の修正版である。

- c. G.8.5.1 小小節で規定される変換係数スケールと改善処理が、1 に等しい  $refinementFlag$ 、 $fieldMbFlag[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$ 、と  $tCoeffLevel[mbAddr]$  を入力として起動され、出力は  $sTCoeff[mbAddr]$  と  $tCoeffLevel[mbAddr]$  の修正版である。この G.8.5.1 小小節で規定される処理の起動に対して、配列  $LumaLevel$ 、 $LumaLevel8x8$ 、 $Intra16x16DCLevel$ 、 $Intra16x16ACLevel$ 、 $CbLevel$ 、 $CbLevel8x8$ 、 $CbIntra16x16DCLevel$ 、 $CbIntra16x16ACLevel$ 、 $CrLevel$ 、 $CrLevel8x8$ 、 $CrIntra16x16DCLevel$ 、 $CrIntra16x16ACLevel$ 、 $ChromaDCLevel$ 、と  $ChromaACLevel$  の全ての要素は、0 に等しいと推定され、また量子化パラメータ  $QP'_Y$  は  $(tQP_Y[mbAddr] + QpBdOffset_Y)$  に等しく設定される。

### 3. $mbType[mbAddr]$ に依存して、次が適用される

- もし  $mbType[mbAddr]$  が  $I\_PCM$ 、 $I\_16x16$ 、 $I\_8x8$ 、または  $I\_4x4$  に等しいならば、G.8.3.2 小小節で規定される SVC INTRA サンプル予測と構築処理が、 $sliceIdc$ 、 $fieldMbFlag$ 、 $mbType$ 、 $ipred4x4[mbAddr]$ 、 $ipred8x8[mbAddr]$ 、 $ipred16x16[mbAddr]$ 、 $ipredChroma[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$ 、 $cS_L$ 、と ChromaArrayType が 0 に等しくない時の  $cS_{Cb}$  と  $cS_{Cr}$  を入力として起動され、出力は配列  $cS_L$  の修正版と、ChromaArrayType が 0 に等しくない時の配列  $cS_{Cb}$  と  $cS_{Cr}$  の修正版である。
- それ以外、もし  $mbType[mbAddr]$  が  $I\_BL$  に等しいならば、次の順序付けられたステップによって規定される。
  - a. G.8.5.3 小小節で規定される残差構築および累算処理が、0 に等しく設定された  $accumulationFlag$ 、 $fieldMbFlag[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$ 、 $rS_L$ 、と ChromaArrayType が 0 に等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  を入力として起動され、出力は  $rS_L$  の修正版と、ChromaArrayType が 0 に等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  の修正版である。
  - b. G.8.5.4 小小節で規定されるサンプル配列の累算処理が、 $fieldMbFlag[mbAddr]$ 、 $rS_L$ 、 $cS_L$ 、と ChromaArrayType が 0 に等しくない時の  $rS_{Cb}$ 、 $rS_{Cr}$ 、 $cS_{Cb}$ 、および  $cS_{Cr}$  を入力として起動され、出力は  $cS_L$  の修正版と、ChromaArrayType が 0 に等しくない時の  $cS_{Cb}$  と  $cS_{Cr}$  の修正版である。
- それ以外( $mbType[mbAddr]$  が  $I\_PCM$ 、 $I\_16x16$ 、 $I\_8x8$ 、 $I\_4x4$ 、または  $I\_BL$  に等しくない)、次の順序付けられたステップによって規定される。
  - a. G.8.5.3 小小節で規定される残差構築および累算処理が、1 に等しい  $accumulationFlag$ 、 $fieldMbFlag[mbAddr]$ 、 $cTrafo[mbAddr]$ 、 $sTCoeff[mbAddr]$ 、 $rS_L$ 、と ChromaArrayType が 0 に等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  を入力として起動され、出力は  $rS_L$  の修正版と、ChromaArrayType が 0 に等しくない時の  $rS_{Cb}$  と  $rS_{Cr}$  の修正版である。
  - b. G.8.4.2 小小節で規定される INTER 予測サンプルのための SVC 復号処理が、 $fieldMbFlag[mbAddr]$ 、 $mbType[mbAddr]$ 、 $subMbType[mbAddr]$ 、 $predFlagL0[mbAddr]$ 、 $predFlagL1[mbAddr]$ 、 $refIdxL0[mbAddr]$ 、 $refIdxL1[mbAddr]$ 、 $mvL0[mbAddr]$ 、 $mvL1[mbAddr]$ 、 $refLayerFieldMbFlag$ 、 $refLayerMbType$ 、 $refPicList0$ 、 $refPicList1$ (利用可能な時)、 $cS_L$ 、と ChromaArrayType が 0 に等しくない時の  $cS_{Cb}$  と  $cS_{Cr}$  を入力として起動され、出力は  $cS_L$  の修正版と、ChromaArrayType が 0 に等しくない時の  $cS_{Cb}$  と  $cS_{Cr}$  の修正版である。

- c. G.8.5.4 小小節で規定されるサンプル配列の累算処理が、fieldMbFlag[mbAddr]、rS<sub>L</sub>、cS<sub>L</sub>、と ChromaArrayType が 0 に等しくない時の rS<sub>Cb</sub>、rS<sub>Cr</sub>、cS<sub>Cb</sub>、および cS<sub>Cr</sub> を入力として起動され、出力は cS<sub>L</sub> の修正版と、ChromaArrayType が 0 に等しくない時の cS<sub>Cb</sub> と cS<sub>Cr</sub> の修正版である。

4. マクロブロック mbAddr に対する変数 MvCnt は mvCnt[mbAddr] に等しく設定される。

## G.8.2 SVC参照ピクチャリスト構築と復号参照ピクチャマーキング処理

ピクチャ順序カウントに対する SVC 復号処理は G.8.2.1 小小節で規定される。

ピクチャ番号に対する SVC 復号処理は G.8.2.2 小小節で規定される。

参照ピクチャリスト構築の SVC 復号処理は G.8.2.3 小小節で規定される。

SVC 復号参照ピクチャマーキング処理は G.8.2.4 小小節で規定される。

frame\_num のギャップの SVC 復号処理は G.8.2.5 小小節で規定される。

ピクチャ順序カウントに対する復号処理は dependency\_id の異なる値に対して独立して適用される。dependency\_id の特定の値に対するピクチャ順序カウントに関連するシンタックス要素は dependency\_id の他の値に対するピクチャ順序カウントの導出に影響を与えない。

参照ピクチャマーキングは dependency\_id の異なる値に対して独立して適用される。dependency\_id の特定の値に対する参照ピクチャマーキングに関連するシンタックス要素は dependency\_id の他の値に対する参照ピクチャマーキングに影響を与えない。

frame\_num のギャップに対する復号処理は dependency\_id の異なる値に対して独立して適用される。

異なる依存表現に対する参照ピクチャリストは独立して構築される。dependency\_id の特定の値に対する参照ピクチャリスト構築に関連するシンタックス要素は dependency\_id の他の値に対する参照ピクチャリスト構築に影響を与えない。dependency\_id の特定の値に対する参照ピクチャリストは dependency\_id の特定の値に対する参照ピクチャマーキングに基づいて構築される。dependency\_id の特定の値に対する参照ピクチャマーキングは dependency\_id の異なる値に対する参照ピクチャリスト構築に影響を与えない。

DependencyIdMax に等しい dependency\_id に対する参照ピクチャリストの要素のみが復号サンプルに関連付けられる復号ピクチャを表す。DependencyIdMax に等しい dependency\_id に対する参照ピクチャリストのみが G.8.4.2 小小節で規定される INTER 予測サンプルの導出に対して用いられる。DependencyIdMax より小さい dependency\_id を持つ依存表現に対する参照ピクチャリストの要素は復号サンプルに関連付けられないレイヤピクチャを表す。0 に等しい dependency\_id に対する参照ピクチャリストの要素は、対応するアクセスユニットに対して 0 に等しい dependency\_id と 0 に等しい quality\_id を持つレイヤ表現を復号する時に導出される、G.8.1.2.1 小小節で規定される配列 fieldMbFlag、mbType、subMbType、predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、と mvL1 に関連付けられる。これら配列は G.8.4.1.2 小小節で規定される 0 に等しい dependency\_id と 0 に等しい quality\_id を持つレイヤ表現に対する動きベクトルと参照インデックスの導出のために使用される。0 より大きい dependency\_id を持つ全ての依存表現に対する参照ピクチャリストの要素は、変数 ScaledRefLayerLeftOffset、ScaledRefLayerRightOffset、ScaledRefLayerTopOffset、と ScaledRefLayerBottomOffset に関連付けられる。これらの変数は G.8.6.1.2 小小節で規定されるレイヤ間動きベクトル予測を導出するために使用される。

記 各アクセスユニットに対して、復号サンプルはDependencyIdMaxに等しいdependency\_idを持つ依存表現に対してのみ蓄積される必要があり、動きデータ配列は0に等しいdependency\_idを持つ依存表現に対してのみ蓄積される必要がある。

ピクチャ順序カウンタ、参照ピクチャリスト構築、参照ピクチャマーキングと frame\_num のギャップに対する SVC 復号処理は 8 章で規定される処理を用いて規定される。この小節で規定される処理、および、これらの処理から起動される 8 章の処理への以下の修正が、処理が起動される依存表現に対する dependency\_id の値を表す currDependencyId を用いて適用される。

- a) 本処理の中で、または本処理から起動される子処理の中で参照される全てのシンタックス要素と導出される大文字変数は currDependencyId に等しい dependency\_id を持つ依存表現に対するシンタックス要素と大文字変数である。
- b) dependency\_id が DependencyIdMax より小さい時、次が適用される。
  - フレーム、フィールド、トップフィールド、ボトムフィールド、ピクチャと復号ピクチャは、それぞれ、レイヤフレーム、レイヤフィールド、レイヤトップフィールド、レイヤボトムフィールド、レイヤピクチャ、復号レイヤピクチャとして解釈され、currDependencyId に等しい dependency\_id を持つ依存表現に対する中間の復号結果を表す。
  - IDR ピクチャは currDependencyId に等しい dependency\_id を持つ依存表現に対する、1 に等しい IdrPicFlag を持つレイヤピクチャとして解釈される。
  - 参照フレーム、参照フィールドと参照ピクチャは currDependencyId に等しい dependency\_id を持つ依存表現に対する、0 より大きい nal\_ref\_idc を持つレイヤフレーム、レイヤフィールドとレイヤピクチャとして解釈される。
  - 非参照フレーム、非参照フィールドと非参照ピクチャは currDependencyId に等しい dependency\_id を持つ依存表現に対する、0 に等しい nal\_ref\_idc を持つレイヤフレーム、レイヤフィールドとレイヤピクチャとして解釈される。
  - 相補的非参照フィールドペアは currDependencyId に等しい dependency\_id を持つ依存表現に対する相補的非参照フィールドペアとして解釈される。dependency\_id の特定の値に対する相補的非参照レイヤフィールドペアは、それらが 2 つの符号化レイヤフィールドとして dependency\_id の特定の値を持つ依存表現を含む連続したアクセスユニットの中にあり、逆のパリティを持つ、最初のレイヤフィールドが既にペアとなっているレイヤフィールドではないような、dependency\_id の特定の値に対する 0 に等しい nal\_ref\_idc を持つ 2 つのレイヤフィールドのペアである。
  - 相補的参照フィールドペアは currDependencyId に等しい dependency\_id を持つ依存表現に対する相補的参照レイヤフィールドペアとして解釈される。dependency\_id の特定の値に対する相補的参照レイヤフィールドペアは、それらが 2 つの符号化レイヤフィールドとして dependency\_id の特定の値を持つ依存表現を含む連続したアクセスユニットの中にあり、frame\_num の同じ値を共有する、復号順序で 2 目目のレイヤフィールドが 0 に等しい IdrPicFlag を持ち、かつ 5 に等しい memory\_management\_control\_operation シンタックス要素を含まないような、dependency\_id の特定の値に対する 0 より大きい nal\_ref\_idc を持つ 2 つのレイヤフィールドのペアである。
  - 相補的フィールドペアは currDependencyId に等しい dependency\_id を持つ依存表現に対する相補的レイヤフィールドペアとして解釈される。相補的レイヤフィールドペアは相補的参照レイヤフィールドペアと相補的非参照レイヤフィールドペアに対する総称である。

- 非ペア非参照フィールドは相補的非参照レイヤフィールドペアの一部ではない、currDependencyId に等しい dependency\_id を持つ依存表現に対する 0 に等しい nal\_ref\_idc を持つレイヤフィールドとして解釈される。
- 非ペア参照フィールドは相補的参照レイヤフィールドペアの一部ではない、currDependencyId に等しい dependency\_id を持つ依存表現に対する 0 より大きい nal\_ref\_idc を持つレイヤフィールドとして解釈される。
- 非ペア参照フィールドは相補的参照レイヤフィールドペアの一部ではない、currDependencyId に等しい dependency\_id を持つ依存表現に対するレイヤフィールドとして解釈される。

### G.8.2.1 ピクチャ順序カウントに対するSVC復号処理

この処理の出力は、現アクセスユニットの全ての依存表現に対する変数 TopFieldOrderCnt(もし適用可能なら)と BottomFieldOrderCnt(もし適用可能なら)である。

現アクセスユニットの全ての依存表現に対して、変数 TopFieldOrderCnt(もし適用可能なら)と BottomFieldOrderCnt(もし適用可能なら)は8.2.1小節で規定されるピクチャ順序カウントに対する復号処理を起動することによって導出される。8.2.1小節で規定される処理のこれら起動に対して、G.8.2小節で規定される修正 a)と b)は、対応する依存表現の dependency\_id に等しくなる currDependencyId に適用される。

アクセスユニットの全ての依存表現に対して、TopFieldOrderCnt または BottomFieldOrderCnt、または両方のいずれかが導出される。両方がアクセスユニットの2つ以上の依存表現で導出される時、それらの差分はアクセスユニットのこれらの依存表現で同じでなければならない。

TopFieldOrderCnt と BottomFieldOrderCnt の値は次のように制限されている。

- 各アクセスユニットに対して、1次元配列 picOrderCnt は次のように導出される。
  - もし TopFieldOrderCnt がアクセスユニットの全ての依存表現で導出されるならば、変数 picOrderCnt[dId]は dId に等しくなる dependency\_id を持つ依存表現の TopFieldOrderCnt に等しく設定される。
  - それ以外(TopFieldOrderCnt がアクセスユニットの全ての依存表現で導出されない)、変数 picOrderCnt[dId]は dId に等しくなる dependency\_id を持つ依存表現の BottomFieldOrderCnt に等しく設定される。
- au0 と au1 は復号順序で au1 が au0 よりも後にあるようなビットストリーム中のアクセスユニットの任意のペアであるとする。
- フラグ idrConditionFlag はアクセスユニットの各依存表現に対して次のように導出されるとする。
  - もしアクセスユニットの依存表現が、1 に等しい IdrPicFlag、または 5 に等しい memory\_management\_control\_operation シンタックス要素を持つならば、idrConditionFlag は 1 に等しく設定される。
  - それ以外(アクセスユニットの依存表現が 0 に等しい IdrPicFlag を持ち、かつ 5 に等しい memory\_management\_control\_operation シンタックス要素を持たない)、idrConditionFlag は 0 に等しく設定される。

- 集合 dIdSet0 は、au0 の全ての dependency\_id 値の集合であるとする。
- 集合 dIdSet1 は、復号順序で au0 に続くアクセスユニットとアクセスユニット au1 のそれぞれを含む間の任意のアクセスユニット中の idrConditionFlag が 1 に等しくないような au1 の全ての dependency\_id 値の集合であるとする。
- 集合 dIdSet0 と dIdSet1 の両方に存在する dId の全ての値に対して、au0 の値 picOrderCnt[dId]と au1 の値 picOrderCnt[dId]の間の差分は同じでなければならない。

### G.8.2.2 ピクチャ番号に対するSVC復号処理

この処理は G.8.2.3 小小節で規定される参照ピクチャリスト構築に対する SVC 復号処理が起動される時、または G.8.2.4.1 小小節で規定される依存表現に対する SVC 参照ピクチャマーキング処理が起動される時に、起動される。

この処理への入力は以下の通りである。

- 依存表現を規定する変数 currDependencyId
- この処理が参照ベースピクチャまたは復号ピクチャのどちらを参照するかを規定する変数 useRefBasePicFlag

この処理の出力は“短期間参照使用”または“長期間参照使用”としてマークされた全てのピクチャに割り当てられた変数 FrameNum、FrameNumWrap、PicNum、LongTermFrameIdx と LongTermPicNum である。

ここからこの小小節の最後まで、G.8.2 小節で規定される修正 a)と b)が適用される。

変数 maxFrameNum は以下によって導出される。

$$\text{maxFrameNum}=(1\ll(\log_2\text{max\_frame\_num\_minus}4+4)) \quad (\text{G-88})$$

各短期間および長期間参照ピクチャに変数 useFlag は次のように割り当てられる。

- もし useRefBasePicFlag が 0 に等しいならば、次が適用される。
  - もし参照ピクチャが“参照ベースピクチャ”としてマークされていないならば、useFlag は 1 に等しく設定される。
  - それ以外(参照ピクチャが“参照ベースピクチャ”としてマークされる)、useFlag は 0 に等しく設定される。
- それ以外、もし参照ピクチャが短期間参照ピクチャであるならば、次が適用される。
  - もし短期間参照ピクチャが“参照ベースピクチャ”としてマークされる、または“参照ベースピクチャ”としてマークされる currDependencyId に等しい dependency\_id equal を持つスライスヘッダで符号化された同一の値 frame\_num を持つ短期間参照ピクチャが存在しないならば、useFlag は 1 に等しく設定される。
  - それ以外(短期間ピクチャが“参照ベースピクチャ”としてマークされない、かつ“参照ベースピクチャ”としてマークされる currDependencyId に等しい dependency\_id equal を持つスライスヘッダで符号化された同一の値 frame\_num を持つ短期間参照ピクチャが存在する)、useFlag は 0 に等しく設定される。

- それ以外(useRefBasePicFlag が 1 に等しく、かつ参照ピクチャが長期間参照ピクチャである)、次が適用される。
  - もし長期間参照ピクチャが“参照ベースピクチャ”としてマークされる、または“参照ベースピクチャ”としてマークされる LongTermFrameIdx の同一の値を持つ長期間参照ピクチャが存在しないならば、useFlag は 1 に等しく設定される。
  - それ以外(長期間ピクチャが“参照ベースピクチャ”としてマークされない、かつ“参照ベースピクチャ”としてマークされる LongTermFrameIdx の同一の値を持つ長期間参照ピクチャが存在する)、useFlag は 0 に等しく設定される。

各短期間参照ピクチャで変数 FrameNum と FrameNumWrap は次のように割り当てられる。

- もし useFlag が 0 に等しいならば、短期間ピクチャに対する FrameNum と FrameNumWrap は“無規定”としてマークされる。
- それ以外、FrameNum は、対応する短期間参照ピクチャの currDependencyId に等しい dependency\_id を持つスライスヘッダで復号されたシンタックス要素 frame\_num に等しく設定される。そして変数 FrameNumWrap は以下によって導出される。

```

if(FrameNum>frame_num)
    FrameNumWrap=FrameNum-maxFrameNum          (G-89)
else
    FrameNumWrap=FrameNum

```

但し、等式 G-89 で用いられる frame\_num の値は現アクセスユニットに対する currDependencyId に等しい dependency\_id を持つスライスヘッダの frame\_num である。

記 FrameNum と FrameNumWrap の値は、currDependencyId の値に依存している。FrameNum と FrameNumWrap の計算のために、currDependencyId に等しい dependency\_id を持つ NAL ユニットのシンタックス要素のみが使用される。

各長期間参照ピクチャは(8.2.5 小節で規定されるようにそれに割り当てられる)関連付けられた LongTermFrameIdx の値を持つ。長期間ピクチャに対して、useFlag が 0 に等しい時、LongTermFrameIdx “無規定”として一時的にマークされる。このマーキングは、G.8.2.3 小小節で規定される参照ピクチャリスト構築に対する SVC 復号処理の起動、または G.8.2.4 小小節で規定される SVC 復号参照ピクチャマーキング処理の起動の後に削除される。

各短期間参照ピクチャで、変数 PicNum が割り当てられ、そして各長期間参照ピクチャで、変数 LongTermPicNum が割り当てられる。これらの変数の値は、currDependencyId に等しい dependency\_id を持つ現スライスに対する field\_pic\_flag と bottom\_field\_flag の値に依存し、そしてそれらは次のように設定される。

- もし field\_pic\_flag が 0 に等しいならば、次が適用される。
  - 各短期間参照フレームまたは相補的参照フィールドペアに対して、次が適用される。
    - もし FrameNumWrap が“無規定”としてマークされるならば、PicNum は“無規定”としてマークされる。
    - それ以外(FrameNumWrap が“無規定”としてマークされない)、PicNum は以下によって導出される。

$$\text{PicNum}=\text{FrameNumWrap} \quad (\text{G-90})$$

- 各長期間参照フレームまたは長期間相補的参照フィールドペアに対して、次が適用される。
  - もし LongTermFrameIdx が“無規定”としてマークされるならば、LongTermPicNum は“無規定”としてマークされる。
  - それ以外(LongTermFrameIdx が“無規定”としてマークされない)、LongTermPicNum は以下によって導出される。

$$\text{LongTermPicNum}=\text{LongTermFrameIdx} \quad (\text{G-91})$$

- それ以外(field\_pic\_flag が 1 に等しい)、次が適用される。
  - 現フィールドは currDependencyId に等しい dependency\_id を持つ現アクセスユニットのスライスを参照するものとする。
  - 各短期間参照フィールドに対して、次が適用される。
    - もし FrameNumWrap が“無規定”としてマークされるならば、PicNum は“無規定”としてマークされる。
    - それ以外(FrameNumWrap が“無規定”としてマークされない)、次が適用される。

$$\text{PicNum}=2*\text{FrameNumWrap}+1 \quad (\text{G-92})$$

- それ以外(参照フィールドが現フィールドの逆のパリティを持つ)、

$$\text{PicNum}=2*\text{FrameNumWrap} \quad (\text{G-93})$$

- 各長期間参照フィールドに対して、次が適用される。
  - もし LongTermFrameIdx が“無規定”としてマークされるならば、LongTermPicNum は“無規定”としてマークされる。
  - それ以外(LongTermFrameIdx が“無規定”としてマークされない)、次が適用される。

- もし参照フィールドが現フィールドと同じパリティを持っているならば、

$$\text{LongTermPicNum}=2*\text{LongTermFrameIdx}+1 \quad (\text{G-94})$$

- それ以外(参照フィールドが現フィールドの逆のパリティを持っている)、

$$\text{LongTermPicNum}=2*\text{LongTermFrameIdx} \quad (\text{G-95})$$

### G.8.2.3 参照ピクチャリスト構築のSVC復号処理

この処理は各 P、EP、B、または EB スライスを復号する開始時に起動される。

この処理への入力は以下の通りである。

- 変数 currDependencyId

- 変数 useRefLayerRepFlag
- 現スライス currSlice

この処理の出力は以下の通りである。

- 参照ピクチャリスト refPicList0
- B と EB スライスに対する、参照ピクチャリスト refPicList1

この小小節で記述される処理を適用した後で、出力の参照ピクチャリスト refPicList0 と refPicList1(適用可能な場合)は現ピクチャのシンタックス要素 temporal\_id より大きいシンタックス要素 temporal\_id であるいかなるピクチャも含んではならない。

ここからこの小小節の最後まで、G.8.2 小節で規定される修正 a)と b)が適用される。

変数 biPred は次のように導出される。

- もし現スライス currSlice が B または EB スライスであるならば、biPred は 1 に等しく設定される。
- それ以外、biPred は 0 に等しく設定される。

復号参照ピクチャはビットストリームによって規定され、かつ G.8.2.4 小小節で規定される“短期間参照使用”または“長期間参照使用”としてマークされる。短期間参照ピクチャは currDependencyId に等しい dependency\_id を持つスライスヘッダで復号された frame\_num の値によって識別される。長期間参照ピクチャはビットストリームによって規定され、かつ G.8.2.4 小小節で規定される長期間フレームインデックスによって識別される。

以下を規定するために G.8.2.2 小小節が currDependencyId と useRefBasePicFlag を入力として起動される。

- 短期間参照ピクチャの各々への変数 FrameNum、FrameNumWrap および PicNum の割り当て
- 長期間参照ピクチャの各々への変数 LongTermPicNum の割り当て

記 G.8.2.2 小小節の起動中に、変数 FrameNum、FrameNumWrap、PicNum と LongTermPicNum は“無規定”としてマークされ得る。さらに、変数 LongTermPicNum はこの小小節とこの小小節の中の全ての起動に対して“無規定”として一時的にマークされ得る。

参照ピクチャは 8.4.2.1 小小節で規定される参照インデックスを通してアドレスされる。参照インデックスは参照ピクチャリスト中のインデックスである。biPred が 0 に等しい時、1 つの参照ピクチャリスト refPicList0 が構築される。B または EB スライス(biPred が 1 に等しい)で復号する時、2 つ目の独立参照ピクチャリスト refPicList1 が refPicList0 に加えて構築される。

各スライスの復号を開始する時に、参照ピクチャリスト refPicList0 および、1 に等しい biPred に対する refPicList1 は次のように導出される。

- 初期参照ピクチャリスト RefPicList0 および、1 に等しい biPred に対する RefPicList1 は、8.2.4.2 小小節で規定される参照ピクチャリストに対する初期化処理を起動することによって導出される。8.2.4.2 小小節の初期化処理中に、FrameNum、FrameNumWrap と、PicNum が G.8.2.2 小小節の起動によって“無規定”としてマークされた全ての短期間参照ピクチャと、LongTermFrameIdx と LongTermPicNum が G.8.2.2 小小節の起動によって“無規定”としてマークされた全ての長期間参照ピクチャは非存在とみなされる。参照ピクチャリストは、“無規定”としてマークされない FrameNum、FrameNumWrap と PicNum の値を持つ短期間ピクチャと、“無規定”としてマークされない LongTermFrameIdx と LongTermPicNum の値を

持つ長期間ピクチャだけを用いることによって構築される。

- 初期参照ピクチャリスト RefPicList0 および、1 に等しい biPred に対する RefPicList1 は、8.2.4.3 小小節で規定される参照ピクチャリストに対する並替処理を起動することによって修正される。8.2.4.2 小小節の並替処理中、FrameNum、FrameNumWrap と、PicNum が G.8.2.2 小小節の起動によって“無規定”としてマークされた全ての短期間参照ピクチャと、LongTermFrameIdx と LongTermPicNum が G.8.2.2 小小節の起動によって“無規定”としてマークされた全ての長期間参照ピクチャは非存在とみなされる。“無規定”としてマークされた FrameNum、FrameNumWrap と PicNum の値を持つ短期間ピクチャと“無規定”としてマークされた LongTermFrameIdx と LongTermPicNum の値を持つ長期間ピクチャは参照ピクチャリスト並替コマンドによって呼ばれてはならない。
- RefPicList0 は refPicList0 に割り当てられる。
- biPred が 1 に等しい時、RefPicList1 は refPicList1 に割り当てられる。

記 G.8.2.2 小小節の処理の起動によって、いくつかの参照ピクチャのパラメータは“無規定”としてマークされるかもしれない。それゆえ、これらピクチャは参照ピクチャリストに対する構築処理で考慮されず、参照ピクチャリスト RefPicList0 と 1 に等しい biPred に対する refPicList1 は、入力パラメータ currDependencyId の値に依存する。

修正された参照ピクチャリスト RefPicList0 の項目の番号は、num\_ref\_idx\_l0\_active\_minus1+1 であり、1 に等しい biPred に対する修正された参照ピクチャリスト refPicList1 の項目の番号は、num\_ref\_idx\_l1\_active\_minus1+1 である。参照ピクチャは修正された参照ピクチャリスト refPicList0 または refPicList1 の 1 より大きいインデックスで現れてもよい。

G.8.2.2 小小節の処理の起動中に、LongTermFrameIdx が“無規定”としてマークされた全ての長期間参照ピクチャに対して、LongTermFrameIdx のこのマーキングが削除される。

#### G.8.2.4 SVC 復号参照ピクチャマーキング処理

現アクセスユニットの全ての依存表現に対して、依存表現のスライスデータ NAL ユニットの nal\_ref\_idc が 0 より大きい時、G.8.2.4.1 小小節で規定される依存表現に対する SVC 復号参照ピクチャマーキング処理が起動される。G.8.2.4.1 小小節で規定される処理の起動には、G.8.2 小節で規定される修正 a) と b) が、対応する依存表現の dependency\_id と等しい currDependencyId を用いて適用される。

##### G.8.2.4.1 依存表現に対する SVC 参照ピクチャマーキング処理

この処理への入力、変数 currDependencyId である。

この処理の出力は、currDependencyId と等しい dependency\_id を有する依存表現に対する修正された参照ピクチャマーキングである。

この処理は、currDependencyId と等しい dependency\_id を有する依存表現に対して nal\_ref\_idc が 0 に等しくない時に、復号ピクチャに対して起動される。

本小小節で規定される処理と本処理から起動される全ての子処理によって参照される、dec\_ref\_pic\_marking( ) シンタックス構造と dec\_ref\_base\_pic\_marking( ) シンタックス構造の全シンタックス要素は、currDependencyID と等しい dependency\_id と 0 に等しい quality\_id を有するスライスヘッダのシンタックス要素である。

記 0 より大きい `quality_id` を有する符号化スライス NAL ユニットは、`dec_ref_pic_marking()` と `dec_ref_base_pic_marking()` のシンタックス構造を含まない。これらの NAL ユニットにおいては、復号参照ピクチャマーキングを規定するシンタックス要素は、`DependencyIdMax` と等しい `dependency_id` と 0 に等しい `quality_id` を有する符号化スライス NAL ユニットより推定される。0 に等しい `dependency_id` と 0 に等しい `quality_id` を有する符号化 NAL ユニットは、`dec_ref_base_pic_marking()` シンタックス構造を含まない。これらの NAL ユニットに対して、`def_ref_base_pic_marking()` シンタックス構造のシンタックス要素は、復号順序において、符号化されたスライス NAL ユニットの直前に来る関連付けられたプリフィックス NAL ユニットより推定される。

`nal_ref_idc` が 0 に等しくない復号ピクチャは、参照ピクチャと呼ばれ、“短期間参照使用”あるいは“長期間参照使用”としてマークされる。`store_ref_base_pic_flag` が 1 に等しい時、復号ピクチャの 2 つ目の表現は、参照ベースピクチャとも呼ばれ、“短期間参照使用”あるいは“長期間参照使用”としてマークされ、さらに、“参照ベースピクチャ”としてマークされる。“参照ベースピクチャ”としてマークされたピクチャは、`use_ref_base_pic_flag` が 1 に等しい後に続くピクチャの INTER 予測のための参照としてのみ用いられる。これらのピクチャは、`use_ref_base_pic_flag` が 0 に等しいピクチャの INTER 予測には用いられず、これらのピクチャは復号処理の出力を表現しない。

復号参照フレームにおいて、その両フィールドは、フレームと同じにマークされる。相補的参照フィールドペアに対して、そのペアは、その両フィールドと同じにマークされる。“短期間参照使用”としてマークされたピクチャは、その `FrameNum` によって、そして、それがフィールドである時は、パリティによって、それが参照ベースピクチャである時は、“参照ベースピクチャ”のマーキングによって識別される。“長期間参照使用”としてマークされたピクチャは、`LongTermFrameIdx` によって、そして、それがフィールドである時は、パリティによって、それが参照ベースピクチャである時は、“参照ベースピクチャ”のマーキングによって識別される。

復号ピクチャが、サンプル配列  $S_L$ 、そして、`ChromaArrayType` が 0 に等しくない時は、 $S_{Cb}$  と  $S_{Cr}$  で表される一方で、参照ベースピクチャはサンプル配列  $B_L$ 、そして、`ChromaArrayType` が 0 に等しくない時は、 $B_{Cb}$  と  $B_{Cr}$  で表される。参照ベースピクチャが 8.4.2.1 小小節による INTER 予測処理において参照される時、サンプル配列  $B_L$ 、 $B_{Cb}$ 、 $B_{Cr}$  は、各々、 $S_L$ 、 $S_{Cb}$ 、 $S_{Cr}$  と呼ばれる。参照ベースピクチャは、復号参照ピクチャマーキング処理の同じ起動によってマークされた対応する復号ピクチャと同じの、変数 `FrameNum`、`FrameNumWrap`、`PicNum`、`LongTermFrameIdx`、および `LongTermPicNum` などの記述情報に関連付けられる。

“短期間参照使用”あるいは、“長期間参照使用”としてマークされているフレームあるいは相補的フィールドペアは、フレームを復号する時に、フレームあるいは、相補的フィールドペアあるいは、それを構成するフィールドの一つが“参照不使用”としてマークされるまで、INTER 予測の参照として利用できる。“短期間参照使用”あるいは、“長期間参照使用”としてマークされたフィールドは、フィールドを復号する時、“参照不使用”としてマークされるまで、INTER 予測の参照として利用できる。

8.2.5.3 小小節で規定される先入れ先出し機構である、スライディングウィンドウ参照ピクチャマーキング処理、あるいは、8.2.5.4 小小節で規定されるカスタマイズされた適応マーキング操作である、適応メモリ制御復号参照ピクチャマーキング処理によって、ピクチャは“参照不使用”としてマークされることができる。

短期間参照ピクチャは、その変数 `FrameNum` や `FrameNumWrap` やそのピクチャ番号である `PicNum`、そしてそれが参照ベースピクチャであるときは、“参照ベースピクチャ”としてマーキングされているかによって、復号処理で利用されるために識別される。長期間参照ピクチャは、変数 `LongTermFrameIdx` や長期間ピクチャ番号である `LongTermPicNum`、そして、それが参照ベースピクチャであるときは、“参照ベースピク

チャ”としてマーキングされているかによって、復号処理で利用されるために識別される。現ピクチャが IDR ピクチャでない場合は、参照ピクチャでの変数 `FrameNum`、`FrameNumWrap`、`PicNum`、`LongTermFrameIdx`、および `LongTermPicNum` への割り当てが、次の通り規定される。

- 変数 `FrameNum`、`FrameNumWrap`、`PicNum`、`LongTermFrameIdx`、および `LongTermPicNum` は、入力として `currDependencyId` を `DependencyIdMax` に等しく設定し、`useRefBasePicFlag` を 1 に等しく設定して G.8.2.2 小節を起動することで、“参照ベースピクチャ”としてマークされた全参照ピクチャに割り当てられる。

記 この G.8.2.2 小節の一回目の起動により、各変数は“参照ベースピクチャ”としてマークされていない参照ピクチャにも割り当てられるかもしれない。これらのピクチャにおいては、次の G.8.2.2 小節の起動により再度変数が割り当てられる、そして、それら 2 回の G.8.2.2 小節の起動により割り当てられる変数は同一である。

- 変数 `FrameNum`、`FrameNumWrap`、`PicNum`、`LongTermFrameIdx`、および `LongTermPicNum` は、入力として `currDependencyId` を `DependencyIdMax` に等しく設定し、`useRefBasePicFlag` を 0 に等しく設定して G.8.2.2 小節を起動することで、“参照ベースピクチャ”としてマークされていない全参照ピクチャに割り当てられる。

復号参照ピクチャマーキングは、次の順序付けられたステップで進行する。

- 現アクセスユニットの全スライスが復号される。
- 現ピクチャに依存して、次が適用される。
  - もし現ピクチャが IDR ピクチャならば、次が適用される。
    - 全ての参照ピクチャが“参照不使用”としてマークされる。
    - `long_term_reference_flag` に依存して、以下が適用される。
      - もし `long_term_reference_flag` が 0 に等しいならば、次が適用される。
        - IDR ピクチャは“短期間参照使用”としてマークされ、`MaxLongTermFrameIdx` は“長期間フレームインデックス無し”に等しく設定される。
        - `store_ref_base_pic_flag` が 1 に等しい時は、IDR ピクチャの参照ベースピクチャは“短期間参照使用”、そして、“参照ベースピクチャ”としてマークされる。
      - それ以外(`long_term_reference_flag` が 1 に等しい)、次が適用される。
        - IDR ピクチャは、“長期間参照使用”としてマークされ、IDR ピクチャの `LongTermFrameIdx` は 0 に設定され、`MaxLongTermFrameIdx` は 0 に等しく設定される。
        - `store_ref_base_pic_flag` が 1 に等しい時は、IDR ピクチャの参照ベースピクチャは“長期間参照使用”、そして、“参照ベースピクチャ”としてマークされ、IDR ピクチャのための `LongTermFrameIdx` は 0 に等しく設定される。
  - それ以外(現ピクチャが IDR ピクチャでない)、次が適用される。
    - `store_base_pic_flag` が 1 に等しい時、次が適用される。

- もし `adaptive_ref_base_pic_marking_mode_flag` が 1 に等しいならば、`dec_ref_base_pic_marking( )`の シンタックス構造からのシンタックス要素と、そして、`adaptive_ref_base_pic_marking_mode_flag`により `adaptive_ref_pic_marking_mode_flag`を置き換えて、8.2.5.4 小小節で規定される適応メモリ制御復号参照ピクチャマーキング処理が起動される。“参照ベースピクチャ”としてマークされた参照ピクチャのみが、この処理の起動の対象となる。
- それ以外 (`adaptive_ref_base_pic_marking_mode_flag` が 0 に等しい)、8.2.5.3 小小節で規定されたスライディングウィンドウ復号参照ピクチャマーキング処理が、`adaptive_ref_base_pic_marking_mode_flag`により `adaptive_ref_pic_marking_mode_flag`を置き換えて起動される。この処理が起動している間、“参照ベースピクチャ”としてマークされない復号ピクチャと、“参照ベースピクチャ”としてマークされた参照ベースピクチャが存在し、両者が最も小さい `FrameNumWrap` の値を有する場合、“参照ベースピクチャ”としてマークされたピクチャのみが“参照不使用”としてマークされる。
- 現復号ピクチャの参照ベースピクチャは、“短期間参照使用”および“参照ベースピクチャ”として、マークされる。
- `adaptive_ref_pic_marking_mode_flag` に依存して、次が適用される。
  - もし `adaptive_ref_pic_marking_mode_flag` が 0 に等しいならば、次が適用される。
    - 8.2.5.3 小小節で規定されたスライディングウィンドウ復号参照ピクチャマーキング処理が起動される。この処理が起動している間、“参照ベースピクチャ”としてマークされていない復号ピクチャと、“参照ベースピクチャ”としてマークされた参照ベースピクチャが存在し、両者は `FrameNumWrap` のもっとも小さい値を有する場合、参照ベースピクチャのみが“参照不使用”としてマークされる。
    - 現復号ピクチャは、“短期間参照使用”としてマークされる。
  - それ以外 (`adaptive_ref_pic_marking_mode_flag` が 1 に等しい)、次が適用される。
    - `dec_ref_pic_marking( )`シンタックス構造のシンタックス要素で 8.2.5.4 小小節で規定される適応メモリ制御復号参照ピクチャマーキング処理が起動される。“参照ベースピクチャ”としてマークされていない参照ピクチャのみが、処理の起動の対象となる。加えて、次が適用される。
      - `LongTermFrameIdx` が `MaxLongTermFrameIdx` より大きく、そして“長期間参照使用”および“参照ベースピクチャ”としてマークされた全ピクチャは、“参照不使用”としてマークされる。
      - この処理が起動している間、`LongTermFrameIdx` がピクチャに割り当てられ、そして `PicNum` (長期間フレームインデックスの割り当て前)と同じ値を有する“参照ベースピクチャ”が存在し、`LongTermFrameIdx` は参照ベースピクチャにも割り当てられている場合、マークが、“短期間参照使用”から“長期間参照使用”に変更される。
    - 現ピクチャが、6 に等しい `memory_management_control_operation` で、“長期間参照使用”としてマークされなかった時、それは、“短期間参照使用”としてマ

クされる。

記 store\_ref\_base\_pic\_flag が 1 に等しい時は、num\_ref\_frames は、2 以上でなければならない。

8.2.5.4 小小節で規定される適応メモリ制御復号参照ピクチャマーキング処理が、temporal\_id が tL に等しい参照ピクチャの復号の後に参照ピクチャを“参照不使用”としてマークするために起動される場合、temporal\_id が tL 未満である参照ピクチャは、“参照不使用”としてマークされてはならない。

G.8.2.2 小小節で規定された起動の間に“無規定”としてマークされた LongTermFrameIdx を有する全ての長期間参照ピクチャに対して、LongTermFrameIdx のこのマーキングは除去される。

#### G.8.2.5 frame\_numのギャップのSVC復号処理

現アクセスユニットの全ての依存表現に対し、次が適用される。

- シンタックス要素 gaps\_in\_frame\_num\_value\_allowed\_flag と frame\_num と導出された変数 PrevRefFrameNum と MaxFrameNum は、対象の依存表現に対するシンタックス要素と生成された大文字変数である。
- gaps\_in\_frame\_num\_value\_allowed\_flag が 0 に等しい時、ビットストリームは、frame\_num が、PrevRefFrameNum、あるいは、(PrevRefFrameNum+1)%MaxFrameNum に等しくない結果をもたらすデータを含んではならない。

記 gaps\_in\_frame\_num\_value\_allowed\_flag が 0 に等しく、かつ frame\_num が PrevRefFrameNum に等しくなく、(PrevRefFrameNum+1)%MaxFrameNum にも等しくない時、復号処理は意図しないピクチャロスと推定すべきである。

- frame\_num が PrevRefFrameNum に等しくなく、(PrevRefFrameNum+1)%MaxFrameNum にも等しくない時、8.2.5.2 小小節で規定される frame\_num のギャップの復号処理が起動される。8.2.5.2 小小節で規定される処理の起動に対して、G.8.2 小節で規定される修正 a)項と b)項が、対象の依存表現に対して dependency\_id と等しい currDependencyId で適用される。

#### G.8.3 SVC INTRA復号処理

G.8.3.1 小小節は INTRA 予測モードのための SVC 導出処理を規定する。

G.8.3.2 小小節は SVC INTRA サンプル予測および構築処理を規定する。

##### G.8.3.1 INTRA予測モードのためのSVC導出処理

base\_mode\_flag が 0 に等しくかつこの処理への入力に指定された mbType[CurrMbAddr]が I\_PCM、I\_16x16、I\_8x8、または I\_4x4 に等しいときにのみ、この処理は実行される。

この処理への入力は以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する PicSizeInMbs 個の要素を持つ 1 次元配列 sliceIdx
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、そして、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を持つリスト fieldMbFlag

- 現レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する PicSizeInMbs 個の要素を持つリスト mbType
- 現レイヤ表現のマクロブロックに対する Intra\_4x4 予測モードを規定する (PicSizeInMbs) × 16 配列 ipred4x4
- 現レイヤ表現のマクロブロックに対する Intra\_8x8 予測モードを規定する (PicSizeInMbs) × 4 配列 ipred8x8
- 現レイヤ表現のマクロブロックに対する Intra\_16x16 予測モードを規定する PicSizeInMbs 個の要素を持つリスト intra16x16
- ChromaArrayType が 1 または 2 に等しいときに、現レイヤ表現のマクロブロックに対する INTRA 色差予測モードを規定する PicSizeInMbs 個の要素を持つリスト intraChroma

この処理の出力は以下の通りである。

- 配列 ipred4x4 の修正版
- 配列 ipred8x8 の修正版
- 配列 ipred16x16 の修正版
- ChromaArrayType が 1 または 2 に等しい時、配列 ipredChroma の修正版

この小小節で規定される処理またはこの小小節で規定される処理の子処理から起動される、6 章で規定される全ての処理に関しては、以下の変更が適用される。

- a) 6.4.11.2 小小節において、fieldMbFlag[mbAddr]が 1 に等しいときは、アドレス mbAddr を持つマクロブロックはフィールドマクロブロックとして扱われ、fieldMbFlag[mbAddr]が 0 に等しいときは、フレームマクロブロックとして扱われる。特に、fieldMbFlag[CurrMbAddr]が 1 に等しいときは、現マクロブロックはフィールドマクロブロックとして扱われ、fieldMbFlag[CurrMbAddr]が 0 に等しいときは、フレームマクロブロックとして扱われる。
- b) 6.4.7 小節において、sliceIdc[mbAddr]が sliceIdc[CurrMbAddr]に等しくないとき、アドレス mbAddr を持つマクロブロックは現マクロブロック CurrMbAddr と異なったスライスに属するよう扱われる。
- c) 6.4.11.2 小小節において、(mbAddr%2)が 0 に等しいときは、マクロブロック mbAddr はトップマクロブロックとして扱われ、(mbAddr%2)が 1 に等しいときは、ボトムマクロブロックとして扱われる。

mbType[CurrMbAddr]が I\_PCM に等しくないときに、以下が適用される。

- もし mbType[CurrMbAddr]が I\_4x4 に等しいならば、G.8.3.1.1 小小節で規定される Intra\_4x4 予測モードの SVC 導出処理が sliceIdc、fieldMbFlag、mbType、ipred4x4、および ipred8x8 を入力として起動され、出力は配列 ipred4x4 の修正版である。
- それ以外、もし mbType[CurrMbAddr]が I\_8x8 に等しいならば、G.8.3.1.2 小小節で規定される Intra\_8x8 予測モードの SVC 導出処理が、sliceIdc、fieldMbFlag、mbType、ipred4x4、および ipred8x8 を入力として起動され、出力は配列 ipred8x8 の修正版である。
- それ以外、もし mbType[CurrMbAddr]が I\_16x16 に等しいならば、ipred16x16[CurrMbAddr]は Intra16x16PredMode に等しく設定される。

ChromaArrayType が 1 または 2 に等しく、かつ mbType[CurrMbAddr]が I\_PCM に等しくないときに、ipredChroma[CurrMbAddr]は intra\_chroma\_pred\_mode に等しく設定される。

#### G.8.3.1.1 Intra\_4x4 予測モードのSVC導出処理

この処理への入力は以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する PicSizeInMbs 個の要素を持つ 1 次元配列 sliceIdx
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、また、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を持つリスト fieldMbFlag
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素を持つリスト mbType
- 現レイヤ表現のマクロブロックの Intra\_4x4 予測モードを規定する(PicSizeInMbs)×16 配列 ipred4x4
- 現レイヤ表現のマクロブロックの Intra\_8x8 予測モードを規定する(PicSizeInMbs)×4 配列 ipred8x8

この処理の出力は配列 ipred4x4 の修正版である。

c4x4BlkIdx=0..15 で索引される 4×4 ブロックは c4x4BlkIdx の昇順で処理され、かつ、各 4×4 ブロックに対し、以下の順序付けられたステップが規定される。

1. 6.4.10.4 小小節で規定される隣接 4×4 輝度ブロックの導出処理は、c4x4BlkIdx を入力として起動され、出力が mbAddrA、c4x4BlkIdxA、mbAddrB、および c4x4BlkIdxB に割り当てられる。この 6.4.10.4 小小節の処理の起動には、G.8.3.1 小小節の a)から c)項で規定される修正が適用される。
2. A または B に置換される N に対し、変数 availableFlagN は次の通り導出される。
  - もしマクロブロック mbAddrN が利用可能であり、かつ mbType[mbAddrN]が I\_PCM、I\_16x16、I\_8x8、または I\_4x4 に等しいならば、availableFlagN は 1 に等しく設定される。
  - それ以外(マクロブロック mbAddrN が利用不可、または mbType[mbAddrN]が I\_PCM、I\_16x16、I\_8x8、または I\_4x4 に等しくない)、availableFlagN は 0 に等しく設定される。
3. 変数 dcPredModePredictedFlag は次の通り導出される。
  - もし availableFlagA または availableFlagB が 0 に等しいならば、dcPredModePredictedFlag は 1 に等しく設定される。
  - それ以外 (availableFlagA が 1 に等しく、かつ availableFlagB が 1 に等しい)、dcPredModePredictedFlag は 0 に等しく設定される。
4. A または B に置換される N に対し、変数 intraMxMPredModeN は次の通り導出される。
  - もし dcPredModePredictedFlag が 0 に等しく、かつ mbType[mbAddrN]が I\_4x4 に等しいならば、intraMxMPredModeN は ipred4x4[mbAddrN][c4x4BlkIdxN]に等しく設定される。
  - それ以外、もし dcPredModePredictedFlag が 0 に等しく、かつ mbType[mbAddrN]が I\_8x8 に等

しいならば、 $\text{intraMxMPredModeN}$  は  $\text{ipred8x8}[\text{mbAddrN}][\text{c4x4BlkIdxN} \gg 2]$  に等しく設定される。

- それ以外 ( $\text{dcPredModePredictedFlag}$  が 1 に等しい、または ( $\text{mbType}[\text{mbAddrN}]$  が  $I_{4x4}$  に等しくなく、かつ  $\text{mbType}[\text{mbAddrN}]$  が  $I_{8x8}$  に等しくない) )、 $\text{intraMxMPredModeN}$  は 2 に等しく設定される。

5. 配列  $\text{ipred4x4}$  の要素  $\text{ipred4x4}[\text{CurrMbAddr}][\text{c4x4BlkIdx}]$  は次の擬似コードで規定される手続きを適用して導出される。

```
predIntra4x4PredMode=Min(intraMxMPredModeA,intraMxMPredModeB)

if(prev_intra4x4_pred_mode_flag[c4x4BlkIdx])

    ipred4x4[CurrMbAddr][c4x4BlkIdx]=predIntra4x4PredMode

else if(rem_intra4x4_pred_mode[c4x4BlkIdx]<predIntra4x4PredMode)                (G-96)

    ipred4x4[CurrMbAddr][c4x4BlkIdx]=rem_intra4x4_pred_mode[c4x4BlkIdx]

else

    ipred4x4[CurrMbAddr][c4x4BlkIdx]=rem_intra4x4_pred_mode[c4x4BlkIdx]+1
```

#### G.8.3.1.2 Intra\_8x8 予測モードの SVC 導出処理

この処理への入力には以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する  $\text{PicSizeInMbs}$  個の要素を持つ 1 次元配列  $\text{sliceIdx}$
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、また、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する  $\text{PicSizeInMbs}$  個の要素を持つリスト  $\text{fieldMbFlag}$
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する  $\text{PicSizeInMbs}$  個の要素を持つリスト  $\text{mbType}$
- 現レイヤ表現のマクロブロックの  $\text{Intra}_{4x4}$  予測モードを規定する  $(\text{PicSizeInMbs}) \times 16$  配列  $\text{ipred4x4}$
- 現レイヤ表現のマクロブロックの  $\text{Intra}_{8x8}$  予測モードを規定する  $(\text{PicSizeInMbs}) \times 4$  配列  $\text{ipred8x8}$

この処理の出力は配列  $\text{ipred8x8}$  の修正版である。

$\text{c8x8BlkIdx}=0..3$  で索引される  $8 \times 8$  ブロックは  $\text{c8x8BlkIdx}$  の昇順で処理され、かつ、各  $8 \times 8$  ブロックに対し、以下の順序付けられたステップが規定される。

1. 6.4.10.2 小小節で規定される隣接  $8 \times 8$  輝度ブロックの導出処理は、 $\text{c8x8BlkIdx}$  を入力として起動され、出力が  $\text{mbAddrA}$ 、 $\text{c8x8BlkIdxA}$ 、 $\text{mbAddrB}$ 、および  $\text{c8x8BlkIdxB}$  に割り当てられる。この 6.4.10.2 小小節の処理の起動には、G.8.3.1 小小節の a) から c) 項で規定される修正が適用される。
2. A または B に置換される N に対し、変数  $\text{availableFlagN}$  は次の通り導出される。

- もしマクロブロック `mbAddrN` が利用可能であり、かつ `mbType[mbAddrN]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しいならば、`availableFlagN` は 1 に等しく設定される。
  - それ以外(マクロブロック `mbAddrN` が利用不可、または `mbType[mbAddrN]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しくない) 、`availableFlagN` は 0 に等しく設定される。
3. 変数 `dcPredModePredictedFlag` は次の通り導出される。
- もし `availableFlagA` または `availableFlagB` が 0 に等しいならば、`dcPredModePredictedFlag` は 1 に等しく設定される。
  - それ以外 ( `availableFlagA` が 1 に等しく、かつ `availableFlagB` が 1 に等しい ) 、`dcPredModePredictedFlag` は 0 に等しく設定される。
4. A または B に置換される N に対し、変数 `intraMxMPredModeN` は次の通り導出される。
- もし `dcPredModePredictedFlag` が 0 に等しく、かつ `mbType[mbAddrN]` が `I_4x4` に等しいならば、`intraMxMPredModeN` は次のように導出される変数 `c4x4Idx` を用いて `ipred4x4[mbAddrN][c8x8BlkIdxN*4+c4x4Idx]` に等しく設定される。
    - もし N が B に等しいならば、`c4x4Idx` は 2 に等しく設定される。
    - それ以外、もし `fieldMbFlag[CurrMbAddr]` が 0 に等しく、`fieldMbFlag[mbAddrN]` が 1 に等しく、かつ `c8x8BlkIdx` が 2 に等しいならば、`c4x4Idx` は 3 に等しく設定される。
    - それ以外 ( N が A に等しく、かつ ( `fieldMbFlag[CurrMbAddr]` が 1 に等しい、または `fieldMbFlag[mbAddrN]` が 0 に等しい、または `c8x8BlkIdx` が 2 に等しくない ) )、`c4x4Idx` は 1 に等しく設定される。
  - それ以外、もし `dcPredModePredictedFlag` が 0 に等しく、かつ `mbType[mbAddrN]` が `I_8x8` に等しいならば、`intraMxMPredModeN` は `ipred8x8[mbAddrN][c8x8BlkIdxN]` に等しく設定される。
  - それ以外 ( `dcPredModePredictedFlag` が 1 に等しい、または ( `mbType[mbAddrN]` が `I_4x4` に等しくなく、かつ `mbType[mbAddrN]` が `I_8x8` に等しくない ) )、`intraMxMPredModeN` は 2 に等しく設定される。
5. 配列 `ipred8x8` の要素 `ipred8x8[CurrMbAddr][c8x8BlkIdx]` は次の擬似コードで規定される手続きを適用して導出される。

```

predIntra8x8PredMode=Min(intraMxMPredModeA,intraMxMPredModeB)

if(prev_intra8x8_pred_mode_flag[c8x8BlkIdx])

    ipred8x8[CurrMbAddr][c8x8BlkIdx]=predIntra8x8PredMode

else if(rem_intra8x8_pred_mode[c8x8BlkIdx]<predIntra8x8PredMode)                                     (G-97)

    ipred8x8[CurrMbAddr][c8x8BlkIdx]=rem_intra8x8_pred_mode[c8x8BlkIdx]

else

    ipred8x8[CurrMbAddr][c8x8BlkIdx]=rem_intra8x8_pred_mode[c8x8BlkIdx]+1

```

### G.8.3.2 SVC INTRAサンプル予測および構築処理

この処理は、この処理へ入力として規定される `mbType` が `I_PCM`、`I_16x16`、`I_8x8` あるいは `I_4x4` に等しい場合にのみ起動される。

この処理への入力は以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `sliceIdc`
- 現レイヤを表現するどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `fieldMbFlag`
- 現レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `mbType`
- 現マクロブロックの `Intra_4x4` 予測モードを規定する 16 要素を有するリスト `ipred4x4`
- 現マクロブロックの `Intra_8x8` 予測モードを規定する 4 要素を有するリスト `ipred8x8`
- 現マクロブロックの `Intra_16x16` 予測モードを規定する変数 `ipred16x16`
- 現マクロブロックの INTRA 色差予測モードを規定する変数 `ipredChroma`
- 現マクロブロックの変換タイプを規定する変数 `cTrafo`
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 要素を有するスケーリングされた変換係数値のリスト `sTCoeff`
- 現レイヤ表現に対する構築輝度サンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 `picSamples_L`
- `ChromaArrayType` が 0 に等しくない場合、現レイヤ表現のための構築色差サンプル値を含む 2 つの  $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$  配列 `picSamples_Cb` と `picSamples_Cr`

この処理の出力は以下の通りである。

- 配列 `picSamples_L` の修正版
- `ChromaArrayType` が 0 に等しくない場合、配列 `picSamples_Cb` と `picSamples_Cr` の修正版

この小小節に規定される処理またはこの小小節に規定される処理の子処理から起動される、6 章または 8 章に規定された全ての処理に対して、以下の修正を適用する。

- a) 6.4.11.2 小小節において、`mbAddr` に位置するマクロブロックは、`fieldMbFlag[mbAddr]` が 1 に等しい場合、フィールドマクロブロックとして扱われ、`fieldMbFlag[mbAddr]` が 0 に等しい場合、フレームマクロブロックとして扱われる。特に、現マクロブロックは `fieldMbFlag[CurrMbAddr]` が 1 に等しい場合、フィールドマクロブロックとして扱われ、`fieldMbFlag[CurrMbAddr]` が 0 に等しい場合、フレームマクロブロックとして扱われる。
- b) 6.4.7 小節において、`sliceIdc[mbAddr]` が `sliceIdc[CurrMbAddr]` に等しくない場合、アドレス `mbAddr` を有するマクロブロックは、現マクロブロック `CurrMbAddr` と異なるスライスに属するものとして扱われる。

- c) 6.4.11.2 小小節において、マクロブロック `mbAddr` は、 $(mbAddr\%2)$  が 0 に等しい場合、トップマクロブロックとして扱われ、 $(mbAddr\%2)$  が 1 に等しい場合、ボトムマクロブロックとして扱われる。
- d) 8.3.1.2、8.3.2.2 小小節、8.3.3、および 8.3.4 小節において、変数 `Intra4x4PredMode`、`Intra8x8PredMode`、`Intra16x16PredMode`、および `intra_chroma_pred_mode` は、それぞれ `ipred4x4`、`ipred8x8`、`ipred16x16`、および `ipredChroma` によって置き換えられる。
- e) 8.3.1.2、8.3.2.2 小小節、8.3.3、および 8.3.4 小節において、マクロブロックアドレス `mbAddr` に位置するマクロブロックのシンタックス要素 `mb_type` は、`mbType[mbAddr]` に置き換えられる。
- f) 8.3.1.2、8.3.2.2 小小節、8.3.3、および 8.3.4 小節において、`mbAddrN` に位置するマクロブロックは `mbType[mbAddrN]` が `I_PCM`、`I_16x16`、`I_8x8`、または `I_4x4` に等しくない場合、INTER 予測モードで符号化されているものとして扱われる。

SVC INTRA サンプル予測と構築処理は以下の順序付けされたステップで進む。

1. G.8.5.3.1 小小節に規定される輝度残差あるいは 3 に等しい `ChromaArrayType` の色差残差の構築処理が、`cTrafo` と `sTCoeff` を入力として起動され、出力は要素 `mbResL[x,y]` を有する `16x16` 配列 `mbResL` として表現される残差輝度サンプル値である。
2. `ChromaArrayType` が 0 に等しくない場合、G.8.5.3.2 小小節に規定される色差残差の構築処理が、`cTrafo` と `sTCoeff` を入力として起動され、出力はそれぞれ要素 `mbResCb[x,y]` と `mbResCr[x,y]` を有する 2つの  $(MbWidthC) \times (MbHeightC)$  配列 `mbResCb` および `mbResCr` として表現される残差色差サンプル値である。
3. G.8.3.2.1 小小節に規定される輝度サンプルまたは 3 に等しい `ChromaArrayType` の色差サンプルに対する SVC INTRA 予測および構築処理が、`BitDepthY`、`sliceIdc`、`fieldMbFlag`、`mbType`、`ipred4x4`、`ipred8x8`、`ipred16x16`、`mbResL`、および `picSamplesL` を入力として起動され、出力は `picSamplesL` 配列の修正版である。
4. `ChromaArrayType` が 0 に等しくない場合、G.8.3.2.2 小小節に規定される色差サンプルに対する SVC INTRA 予測および構築処理が、`sliceIdc`、`fieldMbFlag`、`mbType`、`ipred4x4`、`ipred8x8`、`ipred16x16`、`ipredChroma`、`mbResCb`、`mbResCr`、`picSamplesCb`、および `picSamplesCr` を入力として起動され、出力は配列 `picSamplesCb` と `picSamplesCr` の修正版である。

#### G.8.3.2.1 輝度サンプルまたは 3 に等しい `ChromaArrayType` の色差サンプルに対する SVC INTRA 予測および構築処理

この処理への入力は以下の通りである。

- ビット深度を規定する変数 `bitDepth`
- 現レイヤ表現のマクロブロックに対するスライス識別を規定する `PicSizeInMbs` 個の要素を有する 1次元配列 `sliceIdc`
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する `PicSizeInMbs` 個の要素を有する 1次元配列 `fieldMbFlag`
- 現レイヤを表現するマクロブロックのマクロブロックタイプを規定する `PicSizeInMbs` 個の要素を有する

#### 1次元配列 mbType

- 現マクロブロックの Intra\_4x4 予測モードを規定する 16 要素のリスト ipred4x4
- 現マクロブロックの Intra\_8x8 予測モードを規定する 4 要素のリスト ipred8x8
- 現マクロブロックの Intra\_16x16 予測モードを規定する変数 ipred16x16
- 現マクロブロックの残差サンプル値を含む 16x16 配列 mbRes
- 現レイヤを表現する構築サンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 picSamples

この処理の出力は、配列 picSamples の修正版である。

mbType[CurrMbAddr]に依存して、以下を適用する。

- もし mbType[CurrMbAddr]が I\_PCM に等しいならば、G.8.3.2.1.1 小小小節に規定された I\_PCM マクロブロックの輝度サンプルおよび ChromaArrayType が 3 の色差サンプルの SVC 構築処理は、fieldMbFlag、mbRes、および picSamples を入力として起動され、出力は配列 picSamples の修正版である。
- それ以外、もし mbType[CurrMbAddr]が I\_4x4 に等しいならば、G.8.3.2.1.2 小小小節に規定された SVC Intra\_4x4 サンプル予測および構築処理が bitDepth、sliceIdc、fieldMbFlag、mbType、ipred4x4、mbRes、および picSamples を入力として起動され、出力は配列 picSamples の修正版である。
- それ以外、もし mbType[CurrMbAddr]が I\_8x8 に等しいならば、G.8.3.2.1.3 小小小節に規定された SVC Intra\_8x8 サンプル予測および構築処理が bitDepth、sliceIdc、fieldMbFlag、mbType、ipred8x8、mbRes、および picSamples を入力として起動され、出力は配列 picSamples の修正版である。
- それ以外 (mbType[CurrMbAddr]が I\_16x16 に等しい)、G.8.3.2.1.4 小小小節に規定される SVC Intra\_16x16 サンプル予測および構築処理が bitDepth、sliceIdc、fieldMbFlag、mbType、ipred16x16、mbRes、および picSamples を入力として起動され、出力は配列 picSamples の修正版である。

#### G.8.3.2.1.1 I\_PCM マクロブロックの輝度サンプルおよび 3 に等しいChromaArrayTypeの色差サンプルの SVC構築処理

この処理への入力以下の通りである。

- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックをフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を有する 1次元配列 fieldMbFlag
- 現マクロブロックの残差サンプル値を含む 16x16 配列 mbRes
- 現レイヤ表現における構築サンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 picSamples

この処理の出力は、配列 picSamples の修正版である。

G.8.5.4.3 小小小節に規定される信号成分に対するピクチャサンプル配列の構築処理は fieldMbFlag[CurrMbAddr]と、16 に等しく設定された mbW と、16 に等しく設定された mbH と、mbRes および picSamples を入力として起動され、出力は配列 picSamples の修正版である。

### G.8.3.2.1.2 SVC Intra\_4x4 サンプル予測および構築処理

この処理への入力は以下の通りである。

- ビット深度を規定する変数 `bitDepth`
- 現レイヤ表現のマクロブロックに対するスライス識別を規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `sliceIdc`
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `fieldMbFlag`
- 現レイヤ表現のマクロブロックにおけるマクロブロックタイプを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `mbType`
- 現マクロブロックの `Intra_4x4` 予測モードを規定する 16 要素を有するリスト `ipred4x4`
- 現マクロブロックの残差サンプル値を含む `16x16` 配列 `mbRes`
- 現レイヤ表現における構築済みサンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 `picSamples`

この処理の出力は、配列 `picSamples` の修正版である。

`mbSamples` を現マクロブロックの構築済み INTRA サンプル値を含む `16x16` 配列とする。全ての `mbSamples` 要素は 0 に等しく初期設定される。

`c4x4BlkIdx=0..15` で示される `4x4` ブロックは、`c4x4BlkIdx` の昇順に処理され、各 `4x4` ブロックについて、以下の順序付けられたステップが規定される。

1. 8.3.1.2 小小節に規定される `Intra_4x4` サンプル予測処理は、`c4x4BlkIdx` と `picSamples` を入力として起動され、出力は `pred4x4[x,y]` 要素を有する `4x4` 配列 `pred4x4` の INTRA 予測サンプル値である。8.3.1.2 小小節の処理の起動に対して、G.8.3.2 小小節の a) から f) に規定される修正を適用する。さらに、8.3.1.2 小小節で規定される処理の一部として起動されるかもしれない 8.3.1.2.3 小小小節において、変数 `BitDepthy` は `bitDepth` に置き換えられる。
2. 6.4.3 小節に規定される逆 `4x4` 輝度ブロック走査処理は、`c4x4BlkIdx` を入力として起動され、出力が `(xP,yP)` に割り当てられる。
3.  $x=xP..(xP+3)$  および  $y=yP..(yP+3)$  に対し `Clip3(0,(1<<bitDepth)-1,a)` と規定される `Clip(a)` を用いて、`16x16` 配列 `mbSamples` の要素 `mbSamples[x,y]` は次式で導出される。

$$\text{mbSamples}[x,y]=\text{Clip}(\text{pred4x4}[x-xP,y-yP]+\text{mbRes}[x,y]) \quad (\text{G-98})$$

4. G.8.5.4.3 小小小節に規定される信号成分のピクチャサンプル配列の構築処理は、`fieldMbFlag[CurrMbAddr]` と、16 に等しく設定された `mbW` と、16 に等しく設定された `mbH` と、`mbSamples` と、`picSamples` を入力として起動され、出力は配列 `picSamples` の修正版である。

記 `c4x4BlkIdx` が 15 未満の場合、配列 `mbSamples` は、`c4x4BlkIdx` の現在値以下の `c4x4BlkIdx` で示される `4x4` ブロックの構築済み INTRA サンプルのみを含む。

### G.8.3.2.1.3 SVC Intra\_8x8 サンプル予測および構築処理

この処理への入力は以下の通りである。

- ビット深度を規定する変数 bitDepth
- 現レイヤ表現のマクロブロックに対するスライス識別を規定する PicSizeInMbs 個の要素を有する 1 次元配列 sliceIdx
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を有する 1 次元配列 fieldMbFlag
- 現レイヤを表現するマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素を有する 1 次元配列 mbType
- 現マクロブロックの Intra\_8x8 予測モードを規定する 4 要素を有するリスト ipred8x8
- 現マクロブロックの残差サンプル値を含む 16x16 配列 mbRes
- 現レイヤ表現における構築済みサンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 picSamples

この処理の出力は picSamples の修正版である。

mbSamples を現マクロブロックの構築済み INTRA サンプル値を含む 16x16 配列とする。全ての mbSamples 要素は 0 に等しく初期設定される。

c8x8BlkIdx=0..3 で示される 8x8 ブロックは c8x8BlkIdx の昇順に処理され、各 8x8 ブロックについて、以下の順序づけられたステップが規定される。

1. 8.3.2.2 小小節に規定された Intra\_8x8 サンプル予測処理は、c8x8BlkIdx と picSamples を入力として起動され、出力は要素 pred8x8[x,y] からなる 8x8 配列 pred8x8 の INTRA 予測サンプル値である。この 8.3.2.2 小小節の処理の起動に対して、G.8.3.2 小小節の a) から f) に規定される修正が適用される。さらに、8.3.2.2 小小節で規定される処理の一部として起動されるかもしれない 8.3.2.2.4 小小節において、変数 BitDepth<sub>y</sub> は bitDepth に置き換えられる。
2. 6.4.5 小節に規定される逆 8x8 輝度ブロック走査処理が c8x8BlkIdx を入力として起動され、出力が (xP,yP) に割り当てられる。
3.  $x=xP..(xP+7)$  および  $y=yP..(yP+7)$  に対し、Clip3(0,(1<<bitDepth)-1,a) と規定される Clip(a) を用いて、16x16 配列 mbSamples の要素 mbSamples[x,y] は次式で導出される。

$$\text{mbSamples}[x,y]=\text{Clip}(\text{pred8x8}[x-xP,y-yP]+\text{mbRes}[x,y]) \quad (\text{G-99})$$

4. G.8.5.4.3 小小節に規定される信号成分のピクチャサンプル配列構築処理は、fieldMbFlag[CurrMbAddr] と、16 に等しく設定された mbW と、16 に等しく設定された mbH と、mbSamples と、picSamples を入力として起動され、出力は配列 picSamples の修正版である。

記 c8x8BlkIdx が 3 未満の場合、配列 mbSamples は、c8x8BlkIdx の現在値以下の c8x8BlkIdx で示される 8x8 ブロックの構築済み INTRA サンプルのみを含む。

#### G.8.3.2.1.4 SVC Intra\_16x16 サンプル予測および構築処理

この処理への入力は以下の通りである。

- ビット深度を規定する変数 `bitDepth`
- 現レイヤ表現のマクロブロックに対するスライス指示を規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `sliceIdc`
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `fieldMbFlag`
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `mbType`
- 現マクロブロックの `Intra_16x16` 予測モードを規定する変数 `ipred16x16`
- 現マクロブロックの残差サンプル値を含む  $16 \times 16$  配列 `mbRes`
- 現レイヤ表現の構築サンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 `picSamples`

この処理の出力は配列 `picSamples` の修正版である。

SVC Intra\_16x16 サンプル予測および構築処理は下記の順序付けられたステップに従って進む。

1. 8.3.3 小節で規定される輝度サンプルの `Intra_16x16` 予測処理が、`picSamples` を入力として起動され、出力は要素 `pred16x16[x,y]` を有する  $16 \times 16$  配列 `pred16x16` としての INTRA 予測サンプル値である。この 8.3.3 小節の処理の起動に対して、G.8.3.2 小小節の a) 項から f) 項で規定される修正が適用される。さらに、8.3.3 小節に規定される処理の一部として実行されるかもしれない 8.3.3.3 小小節において、変数 `BitDepthy` は `bitDepth` に置き換えられる。
2. `Clip3(0,(1<<bitDepth)-1,a)` を規定する `Clip(a)` を用いて、 $16 \times 16$  配列 `mbSamples` は次の式で導出される。

$$\text{mbSamples}[x,y]=\text{Clip}(\text{pred16x16}[x,y]+\text{mbRes}[x,y]) \text{ , } x,y=0..15 \quad (\text{G-100})$$

3. G.8.5.4.3 小小節で規定される信号成分のピクチャサンプル配列構築処理は `fieldMbFlag[CurrMbAddr]`、16 に等しく設定された `mbW`、16 に等しく設定された `mbH`、そして `picSamples` を入力として起動され、出力は配列 `picSamples` の修正版である。

#### G.8.3.2.2 色差サンプルに対する SVC INTRA 予測と構築処理

この処理の入力は以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `sliceIdc`
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する `PicSizeInMbs` 個の要素を有する 1 次元配列 `fieldMbFlag`

- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素を有する 1 次元配列 mbType
- 現マクロブロックの Intra\_4x4 予測モードを規定する 16 要素を有するリスト ipred4x4
- 現マクロブロックの Intra\_8x8 予測モードを規定する 4 要素を有するリスト ipred8x8
- 現マクロブロックの Intra\_16x16 予測モードを規定する変数 ipred16x16
- 現マクロブロックの INTRA 色差予測モードを規定する変数 ipredChroma
- 現マクロブロックの残差色差サンプル値を含む 2 つの (MbWidthC) × (MbHeightC) 配列 mbResCb と mbResCr
- 現レイヤ表現の構築サンプル値を含む 2 つの (PicWidthInSamplesC) × (PicHeightInSamplesC) 配列 picSamplesCb と picSamplesCr

この処理の出力は配列 picSamplesCb と picSamplesCr の修正版である。

ChromaArrayType に依存して、下記が適用される。

- もし ChromaArrayType が 1 または 2 に等しいならば、下記が適用される。
  - もし mbType[CurrMbAddr] が I\_PCM に等しいならば、G.8.3.2.2.1 小小小節で規定される I\_PCM マクロブロックの色差サンプルの SVC 構築処理は fieldMbFlag、sTCoeff、picSamplesCb、および picSamplesCr を入力として起動され、出力は picSamplesCb と picSamplesCr の修正版である。
  - それ以外 (mbType[CurrMbAddr] が I\_PCM と等しくない)、G.8.3.2.2.2 小小小節で規定するように 1 または 2 に等しい ChromaArrayType を用いた色差サンプルの SVC INTRA 予測と構築処理は sliceIdc、fieldMbFlag、mbType、ipredChroma、mbResCb、mbResCr、picSamplesCb、および picSamplesCr を入力として起動され、出力は配列 picSamplesCb と picSamplesCr の修正版である。
- それ以外 (ChromaArrayType が 3 に等しい)、Cb と Cr に置き換えられる CX に対して、G.8.3.2.1 小小小節で規定される輝度サンプルまたは 3 に等しい ChromaArrayType の色差サンプルの SVC INTRA 予測および構築処理は BitDepthC、sliceIdc、fieldMbFlag、mbType、ipred4x4、ipred8x8、ipred16x16、mbResCX、および picSamplesCX を入力として用いて起動され、出力は配列 picSamplesCX の修正版である。

#### G.8.3.2.2.1 I\_PCMマクロブロックの色差サンプルのSVC構築処理

この処理の入力は以下の通りである。

- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を有する 1 次元配列 fieldMbFlag
- 現マクロブロックの残差色差サンプル値を含む 2 つの (MbWidthC) × (MbHeightC) 配列 mbResCb と mbResCr
- 現レイヤ表現の構築色差サンプル値を含む 2 つの (PicWidthInSamplesC) × (PicHeightInSamplesC) 配列 picSamplesCb と picSamplesCr

この処理の出力は配列 picSamplesCb と picSamplesCr の修正版である。

Cb と Cr に置き換えられる CX に対して、G.8.5.4.3 小小小節で規定される信号成分のピクチャサンプル配

列構築処理は fieldMbFlag[CurrMbAddr]、MbWidthC に等しく設定された mbW、MbHeightC に等しく設定された mbH、mbRes<sub>CX</sub>、および picSamples<sub>CX</sub> を入力として起動され、出力は配列 picSamples<sub>CX</sub> の修正版である。

#### G.8.3.2.2.2 1 または 2 に等しい ChromaArrayType の色差サンプルの SVC INTRA 予測と構築処理

この処理は ChromaArrayType が 1 または 2 に等しい時のみ起動される。

この処理の入力は以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する PicSizeInMbs 個の要素を有する 1 次元配列 sliceIdc
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を有する 1 次元配列 fieldMbFlag
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素を有する 1 次元配列 mbType
- 現マクロブロックの INTRA 色差予測モードを規定する変数 ipredChroma
- 現マクロブロックの残差色差サンプル値を含む 2 つの (MbWidthC) × (MbHeightC) 配列 mbRes<sub>Cb</sub> と mbRes<sub>Cr</sub>
- 現レイヤ表現の構築色差サンプル値を含む 2 つの (PicWidthInSamples<sub>C</sub>) × (PicHeightInSamples<sub>C</sub>) 配列 picSamples<sub>Cb</sub> と picSamples<sub>Cr</sub>

この処理の出力は配列 picSamples<sub>Cb</sub> と picSamples<sub>Cr</sub> の修正版である。

1 または 2 に等しい ChromaArrayType の色差サンプルの SVC INTRA 予測と構築処理は下記の順序付けられたステップに従って進む。

1. 8.3.4 小節に規定された色差サンプルの INTRA 予測処理は picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub> を入力として起動され、出力は要素 pred<sub>Cb</sub>[x,y] と pred<sub>Cr</sub>[x,y] をそれぞれ有する 2 つの (MbWidthC) × (MbHeightC) 配列 pred<sub>Cb</sub> および pred<sub>Cr</sub> としての INTRA 予測色差サンプル値である。この 8.3.4 小節の処理の起動に対して、G.8.3.2 小小節の a) から f) 項に規定される修正を適用する。
2. Cb と Cr に置き換えられる CX に対して、(MbWidthC) × (MbHeightC) 配列 mbSamples<sub>CX</sub> は次の式で導出される。

$$\text{mbSamples}_{CX}[x,y] = \text{Clip1}_C(\text{pred}_{CX}[x,y] + \text{mbRes}_{CX}[x,y]), \quad x=0..(\text{MbWidthC}-1), \\ y=0..(\text{MbHeightC}-1) \quad (\text{G-101})$$

3. Cb と Cr に置き換えられる CX に対して、G.8.5.4.3 小小節で規定される信号成分のピクチャサンプル配列構築処理は fieldMbFlag[CurrMbAddr]、MbWidthC に等しく設定された mbW、MbHeightC に等しく設定された mbH、mbSamples<sub>CX</sub>、および picSamples<sub>CX</sub> を入力として用いて起動され、出力は配列 picSamples<sub>CX</sub> の修正版である。

#### G.8.4 SVC INTER 予測処理

G.8.4.1 小小節は動きベクトル成分と参照インデックスの SVC 導出処理を規定する。

G.8.4.2 小節は INTER 予測サンプルの SVC 復号処理を規定する。

#### G.8.4.1 動きベクトル成分と参照インデックスのSVC導出処理

この処理の入力は以下の通りである。

- 現レイヤ表現のマクロブロックに対するスライス識別を規定する PicSizeInMbs 個の要素を有する 1 次元配列 sliceIdc
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を有する 1 次元配列 fieldMbFlag
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素を有する 1 次元配列 mbType
- 現レイヤ表現のマクロブロックのサブマクロブロックタイプを規定する (PicSizeInMbs) × 4 配列 subMbType
- 現レイヤ表現のマクロブロックの予測リスト利用フラグを規定する 2 つの (PicSizeInMbs) × 4 配列 predFlagL0 と predFlagL1
- 現レイヤ表現のマクロブロックの参照インデックスを規定する 2 つの (PicSizeInMbs) × 4 配列 refIdxL0 と refIdxL1
- 現レイヤ表現のマクロブロックの動きベクトル成分を規定する 2 つの (PicSizeInMbs) × 4 × 4 × 2 配列 mvL0 と mvL1
- 現レイヤ表現のマクロブロックの動きベクトルの数を規定する PicSizeInMbs 個の要素を有する 1 次元配列 mvCnt
- 現マクロブロックのレイヤ間参照インデックスの予測因子を規定する 2 つの 2 × 2 配列 refIdxILPredL0 と refIdxILPredL1
- 現マクロブロックのレイヤ間動きベクトルの予測因子を規定する 2 つの 4 × 4 × 2 配列 mvILPredL0 と mvILPredL1
- DQId が 0 に等しく、かつ (slice\_type % 5) が 1 に等しい時、参照リスト refPicList1

この処理の出力は以下の通りである。

- 配列 predFlagL0 と predFlagL1 の修正版
- 配列 refIdxL0 と refIdxL1 の修正版
- 配列 mvL0 と mvL1 の修正版
- 配列 mvCnt の修正版

mbType[CurrMbAddr]に依存して、下記が適用される。

- もし mbType[CurrMbAddr]が I\_PCM、I\_16x16、I\_8x8、I\_4x4、または I\_BL に等しいならば、配列 predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1、および mvCnt は次の式で修正される

$\text{predFlagLX}[\text{CurrMbAddr}][m]=0, X=0..1, m=0..3$  (G-102)

$\text{refIdxLX}[\text{CurrMbAddr}][m] = -1, X=0..1, m=0..3$  (G-103)

$\text{mvLX}[\text{CurrMbAddr}][m][s][c]=0, X=0..1, m=0..3, s=0..3, c=0..1$  (G-104)

$\text{mvCnt}[\text{CurrMbAddr}] = 0$  (G-105)

- それ以外( $\text{mbType}[\text{CurrMbAddr}]$ が  $I\_PCM, I\_16x16, I\_8x8, I\_4x4$ 、および  $I\_BL$  に等しくない)、配列  $\text{predFlagL0}, \text{predFlagL1}, \text{refIdxL0}, \text{refIdxL1}, \text{mvL0}, \text{mvL1}$ 、および  $\text{mvCnt}$  は下記の文に規定されるように修正される。

変数  $\text{isDirectFlag}$  は下記のように導出される。

- もし下記のいずれかの条件が真ならば、 $\text{isDirectFlag}$  は 1 に等しく設定される。
  - $\text{mbType}[\text{CurrMbAddr}]$ が  $B\_Skip$  または  $B\_Direct\_16x16$  に等しい
  - $\text{mbType}[\text{CurrMbAddr}]$ が  $B\_8x8$  に等しくかつ  $\text{subMbType}[\text{CurrMbAddr}][\text{mbPartIdx}]$ が  $B\_Direct\_8x8$  に等しい
- それ以外、 $\text{isDirectFlag}$  は 0 に等しく設定される。

変数  $\text{numMbPart}$  は下記のように導出される。

- もし  $\text{mbType}[\text{CurrMbAddr}]$ が  $B\_Skip$  または  $B\_Direct\_16x16$  に等しく、かつ  $\text{DQId}$  が 0 に等しい ( $\text{nal\_unit\_type}$  が 20 に等しくない)場合、 $\text{numMbPart}$  は 4 に等しく設定される。
- それ以外、もし  $\text{mbType}[\text{CurrMbAddr}]$ が  $B\_Skip$  または  $B\_Direct\_16x16$  に等しい(かつ  $\text{DQId}$  が 0 より大きくかつ  $\text{nal\_unit\_type}$  が 20 に等しい)場合、 $\text{numMbPart}$  は 1 に等しく設定される。
- それ以外( $\text{mbType}[\text{CurrMbAddr}]$ が  $B\_Skip$  および  $B\_Direct\_16x16$  に等しくない)、 $\text{numMbPart}$  は  $\text{NumMbPart}(\text{mbType}[\text{CurrMbAddr}])$ に等しく設定される。

マクロブロックパーティションインデックス  $\text{mbPartIdx}$  は値  $0..(\text{numMbPart}-1)$ 上で進み、各  $\text{mbPartIdx}$  値に対し下記順序付けられたステップが規定される。

1. 変数  $\text{numSubMbPart}$  は下記のように導出される。

- もし  $\text{isDirectFlag}$  が 1 に等しく、かつ  $\text{DQId}$  が 0 に等しい( $\text{nal\_unit\_type}$  が 20 に等しくない)ならば、 $\text{numSubMbPart}$  は 4 に等しく設定される。
- それ以外、もし  $\text{isDirectFlag}$  が 1 に等しい(かつ  $\text{DQId}$  が 0 より大きい、かつ  $\text{nal\_unit\_type}$  が 20 に等しい)ならば、 $\text{numSubMbPart}$  は 1 に等しく設定される。
- それ以外 ( $\text{isDirectFlag}$  が 0 に等しい) 、  $\text{numSubMbPart}$  は  $\text{NumSubMbPart}(\text{subMbType}[\text{CurrMbAddr}][\text{mbPartIdx}])$ に等しく設定される。

2. サブマクロブロックパーティションインデックス  $\text{subMbPartIdx}$  は値  $0..(\text{numSubMbPart}-1)$ 上で進み、各  $\text{subMbPartIdx}$  の値に対し、G.8.4.1.1 小小小節で規定されるマクロブロックまたはサブマクロブロックパーティションの輝度動きベクトル成分および参照インデックスの SVC 導出処理は  $\text{mbPartIdx}, \text{subMbPartIdx}, \text{isDirectFlag}, \text{sliceIdx}, \text{fieldMbFlag}, \text{mbType}, \text{subMbType}, \text{predFlagL0}, \text{predFlagL1}, \text{refIdxL0},$

refIdxL1、mvL0、mvL1、mvCnt、refIdxILPredL0、refIdxILPredL1、mvILPredL0、mvILPredL1、および、DQId が 0 に等しくかつ(slice\_type%5)が 1 に等しい時は、参照ピクチャリスト refPicList1 を入力として起動され、出力は配列 predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1、および mvCnt の修正版である。

#### G.8.4.1.1 マクロブロックまたはサブマクロブロックパーティションの輝度動きベクトル成分および参照インデックスのSVC導出処理

この小小節は、入力として規定される mbType[CurrMbAddr]が I\_PCM、I\_16x16、I\_8x8、I\_4x4、I\_BL のいずれにも等しくない時のみ起動される。

この処理の入力は以下の通りである。

- 現マクロブロックパーティションを規定する変数 mbPartIdx
- 現サブマクロブロックパーティションを規定する変数 subMbPartIdx
- 現マクロブロックパーティションが直接モードで符号化されるかを規定する変数 isDirectFlag
- 現レイヤ表現のマクロブロックに対するスライス識別を規定する PicSizeInMbs 個の要素を持つ 1 次元配列 sliceIdx
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されるか、および、どのマクロブロックがフレームマクロブロックとして符号化されるかを規定する PicSizeInMbs 個の要素を有する 1 次元配列 fieldMbFlag
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素を持つ 1 次元配列 mbType
- 現レイヤ表現のマクロブロックのサブマクロブロックタイプを規定する (PicSizeInMbs) × 4 配列 subMbType
- 現レイヤ表現のマクロブロックの予測リスト利用フラグを規定する 2 つの (PicSizeInMbs) × 4 配列 predFlagL0、predFlagL1
- 現レイヤ表現のマクロブロックの参照インデックスを規定する 2 つの (PicSizeInMbs) × 4 の配列 refIdxL0、refIdxL1
- 現レイヤ表現のマクロブロックの動きベクトル成分を規定する 2 つの (PicSizeInMbs) × 4 × 4 × 2 の配列 mvL0、mvL1
- 現レイヤ表現のマクロブロックの動きベクトルの数を規定する PicSizeInMbs 個の要素を持つ 1 次元配列 mvCnt
- 現マクロブロックのレイヤ間参照インデックス予測因子を規定する 2 つの 2 × 2 配列 refIdxILPredL0、refIdxILPredL1
- 現マクロブロックのレイヤ間動きベクトル予測因子を規定する 2 つの 4 × 4 × 2 配列 mvILPredL0、mvILPredL1
- DQId が 0 に等しく、かつ(slice\_type%5)が 1 に等しい場合、参照ピクチャリスト refPicList1

この処理の出力は以下の通りである。

- 配列 predFlagL0 と predFlagL1 の修正版
- 配列 reflDxL0 と reflDxL1 の修正版
- 配列 mvL0 と mvL1 の修正版
- 配列 mvCnt の修正版

この小小節で規定される処理またはこの小節で規定される処理の子処理より起動される、6 章または 8 章で規定される全ての処理に対して、次の修正が適用される。

- a) 6.4.11.2 小小節と 8.4.1.3.2 小小節において、アドレス mbAddr を持つマクロブロックは fieldMbFlag[mbAddr] が 1 と等しい場合、フィールドマクロブロックとして扱われ、fieldMbFlag[mbAddr] が 0 と等しい場合、フレームマクロブロックとして扱われる。特に、現マクロブロックは、fieldMbFlag[CurrMbAddr] が 1 に等しい場合、フィールドマクロブロックとして扱われ、fieldMbFlag[CurrMbAddr] が 0 に等しい場合、フレームマクロブロックとして扱われる。
- b) 6.4.7 小節において、mbAddr を持つマクロブロックは、sliceIdc[mbAddr] が sliceIdc[CurrMbAddr] と等しくない場合、現マクロブロック CurrMbAddr とは異なるスライスに属するとして扱われる。
- c) 6.4.11.2 小小節において、mbAddr を持つマクロブロックは、(mbAddr%2) が 0 に等しい場合には、トップマクロブロックとして扱われ、(mbAddr%2) が 1 に等しい場合にはボトムマクロブロックとして扱われる。
- d) 6.4.2.1 小小節、6.4.2.2 小小節、6.4.10.7 小小節、8.4.1.1 小小節、8.4.1.3 小小節において現れる全ての mb\_type は、この小小節の入力である配列 mbType を用いた mbType[CurrMbAddr] で置き換えられる。
- e) 6.4.2.2 と 6.4.10.7 小小節において現れる全ての sub\_mb\_type は、この小小節の入力である配列 subMbType を用いた subMbType[CurrMbAddr] で置き換えられる。
- f) 6.4.10.7 小小節において、輝度位置(xW,yW)に基づくマクロブロックパーティションインデックス mbPartIdxN とサブマクロブロックパーティションインデックス subMbPartN の決定に用いられるマクロブロック mbAddrN のマクロブロックパーティションは、mbType[mbAddrN] と G.8.4.1 小小節で与えられるマクロブロックパーティションである。
- g) 6.4.10.7 小小節において、mbAddrN と mbPartIdxN と subMbPartIdxN で与えられるマクロブロックパーティションもしくはサブマクロブロックパーティションは、mbAddrN が CurrMbAddr と等しく、かつ(4\*mbPartIdxN+subMbPartIdxN)が(4\*mbPartIdx+subMbPartIdx)よりも大きい場合、まだ復号されていないとして扱われる。
- h) 8.4.1.3.2 小小節において、マクロブロック mbAddrN は、mbType[mbAddrN] が I\_PCM、I\_16x16、I\_8x8、I\_4x4、I\_BL のいずれかと等しい場合、INTRA 予測モードで符号化されたとして扱われる。
- i) 8.4.1.3.2 小小節において、mbAddrN、mbPartIdxN、subMbPartIdxN で与えられるマクロブロックパーティションまたはサブマクロブロックパーティションの変数 predFlagLX は、この小小節の入力である配列 predFlagLX である predFlagLX[mbAddrN][mbPartIdxN] で置き換えられる。
- j) 8.4.1.3.2 小小節において、mbAddrN、mbPartIdxN、subMbPartIdxN で与えられるマクロブロック

パーティションまたはサブマクロブロックパーティションの動きベクトル  $MvLX[mbPartIdxN][subMbPartIdxN]$  と参照インデックス  $RefIdxLX[mbPartIdxN]$  は、この小小小節の入力である  $mvLX[mbAddrN][mbPartIdxN][subMbPartIdxN]$  である  $refIdxLX[mbAddrN][mbPartIdxN]$  で各々置き換えられる。

- k) 8.4.1.2.1 小小小節において、現れるいずれの  $RefPicList1[0]$  も、この小小小節の入力として規定される参照ピクチャリスト  $refPicList1$  において、 $refPicList1[0]$  が ( $field\_pic\_flag$  が 1 と等しい場合) 最初のレイヤのフィールドである  $refPicList1[0]$  で置き換えられ、( $field\_pic\_flag$  が 0 に等しい場合) 最初のレイヤのフレームまたは、レイヤ相補的フィールドペアで置き換えられる。参照ピクチャリスト  $refPicList1$  は、 $DQId$  が 0 に等しいレイヤ表現に対応する前に復号済みのアクセスユニットにおけるレイヤピクチャの参照リストである。
- l) 8.4.1.2.1 小小小節において、現ピクチャ  $CurrPic$  は  $DQId$  が 0 に等しい現レイヤピクチャを表現し、変数  $colPic$  は、前に復号済みのアクセスユニットにおける  $DQId$  が 0 に等しいレイヤ表現に対するレイヤピクチャを規定する。
- m) 8.4.1.2.1 小小小節において、全てのピクチャ順序カウンタの値は、 $dependency\_id$  が 0 と等しい依存表現に対するピクチャ順序カウンタである。
- n) 8.4.1.2.1 小小小節において、G.8.2 小節で規定された b) の修正が、 $currDependencyId$  を 0 として適用される。
- o) 8.4.1.2.1 小小小節において、値  $fieldDecodingFlagX$  の導出において、マクロブロック  $mbAddrX$  は、 $fieldMbColPicFlag[mbAddrX]$  が 1 と等しい場合、フィールドマクロブロックとして扱われ、 $fieldMbColPicFlag[mbAddrX]$  が 0 と等しい場合、フレームマクロブロックとして扱われる。配列  $fieldMbColPicFlag$  は、 $DQId$  が 0 と等しいレイヤ表現に対する G.8.1.5.1 小小小節の処理によって導出される配列  $fieldMbFlag$  を規定する。
- p) 8.4.1.2.1 小小小節において、 $colPic$  のピクチャ内のマクロブロック  $mbAddrCol$  に対する変数  $PredFlagL0$ 、 $PredFlagL1$ 、 $RefIdxL0$ 、 $RefIdxL1$ 、 $MvL0$ 、 $MvL1$  は、 $DQId$  が 0 と等しい場合に関連する  $colPic$  レイヤピクチャにおいて導出される  $predFlagL0[mbAddrCol]$ 、 $predFlagL1[mbAddrCol]$ 、 $refIdxL0[mbAddrCol]$ 、 $refIdxL1[mbAddrCol]$ 、 $mvL0[mbAddrCol]$ 、 $mvL1[mbAddrCol]$  で各々置き換えられる。
- q) 8.4.1.2.1 小小小節において、マクロブロック  $mbAddrCol$  は、 $DQId$  が 0 に等しい場合に関連する  $colPic$  レイヤピクチャにおいて導出される  $mbType[mbAddrCol]$  が  $I_{16x16}$ 、 $I_{8x8}$ 、 $I_{4x4}$ 、or  $I_{PCM}$  と等しい場合、イントラマクロブロック予測モードで符号化されたとして解釈される。
- r) 8.4.1.2.2 小小小節において、共通位置マクロブロックは  $fieldMbColPicFlag[mbAddrCol]$  が 1 と等しい場合、フィールドマクロブロックとして扱われ、 $fieldMbColPicFlag[mbAddrCol]$  が 0 と等しい場合にはフレームマクロブロックとして扱われる。配列  $fieldMbColPicFlag$  は  $DQId$  が 0 と等しいレイヤ表現に対する G.8.1.5.1 小小小節の処理により導出される配列  $fieldMbFlag$  を規定する。マクロブロックアドレス  $mbAddrCol$  は 8.4.1.2.1 小小小節において導出される共通位置マクロブロックのマクロブロックアドレスである。

参照インデックス予測因子  $refIdxPredL0$ 、 $refIdxPredL1$  と、動きベクトル予測因子  $mvPredL0$ 、 $mvPredL1$  と、変数  $mvCntInc$  は、以下により導出される。

- もし mbType[CurrMbAddr] が P\_Skip と等しいならば、参照インデックス予測因子 refIdxPredL1 は-1 に等しく設定され、動きベクトル予測因子 mvPredL1 の両成分は 0 に等しく設定され、変数 mvCntInc は 1 に等しく設定され、そして 8.4.1.1 小小節に規定される P スライスのスキップマクロブロックの輝度動きベクトルの導出処理が起動され、その出力は動きベクトル予測因子 mvPredL0 と参照インデックス予測因子 refIdxPredL0 に割り当てられる。8.4.1.1 小小節の処理の起動に対し、この小小節の上記 a)から j)の修正が適用される。
- それ以外、もし isDirectFlag が 1 と等しく、かつ DQId が 0 と等しい(nal\_unit\_type が 20 と等しくない)ならば、8.4.1.2.2 小小節において規定される空間方向直接輝度動きベクトルと参照インデックス予測モードの導出処理が、mbPartIdx と subMbPartIdx を入力として起動され、出力変数 refIdxL0、refIdxL1、mvL0、mvL1、subMvCnt は各々、参照インデックス予測因子 refIdxPredL0、refIdxPredL1、動きベクトル予測因子 mvPredL0、mvPredL1、および変数 subMvCnt に割り当てられる。8.4.1.1 小小節の処理の起動に対し、この小小節の上記 a)から j)の修正が適用される。

記 現小小節が起動される場合、direct\_spatial\_mv\_pred\_flag は常に 1 に等しい。

- それ以外、もし isDirectFlag が 1 と等しい(かつ、DQId が 0 より大きく、nal\_unit\_type が 20 と等しい)ならば、G.8.4.1.2 小小節において規定される nal\_unit\_type が 20 と等しい NAL ユニットにおける B\_Skip、B\_Direct\_16x16、および B\_Direct\_8x8 に対する輝度動きベクトルと参照インデックス SVC 導出処理が、mbPartIdx、fieldMbFlag、mbType、subMbType、predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1 を入力として起動され、出力は refIdxPredL0、refIdxPredL1、mvPredL0、mvPredL1、mvCntInc となる。
- それ以外、変数 mvCntInc は 0 に等しく初期設定され、0 と 1 に置き換えられる X に対して、以下が適用される。
  - もし以下の条件のいずれかが真であるならば、refIdxPredLX は-1 に等しく設定され、mvPredLX の両成分は 0 に等しく設定される。
    - mbType[CurrMbAddr] が P\_8x8 、 P\_8x8ref0 、 B\_8x8 と等しくなく、かつ MbPartPredMode(mbType[CurrMbAddr],mbPartIdx)が Pred\_LX、BiPred と等しくない。
    - mbType[CurrMbAddr] が P\_8x8 、 P\_8x8ref0 、 B\_8x8 と等しくなく、かつ SubMbPartPredMode(subMbType[CurrMbAddr][mbPartIdx])が Pred\_LX、BiPred と等しくない。
  - それ以外、もし base\_mode\_flag が 1 と等しい場合、もしくは motion\_prediction\_flag\_LX[mbPartIdx] が 1 と等しい場合、次の順序付けられたステップが規定される。
    1. 6.4.2.1 小小節で規定される逆マクロブロックパーティション走査処理が、mbPartIdx を入力として起動され、出力は(xP,yP)に割り当てられる。6.4.2.1 小小節の処理のこの起動に対して、この小小節の上記 d)で規定される修正が適用される。
    2. 6.4.2.2 小小節で規定される逆サブマクロブロックパーティション走査処理が、mbPartIdx と subMbPartIdx を入力として起動され、出力は(xS,yS)に割り当てられる。6.4.2.2 小小節の処理のこの起動に対して、この小小節の上記 d)と e)で規定される修正が適用される。
    3. 参照インデックス予測因子 refIdxPredLX と動きベクトル予測因子 mvPredLX は、以下により導出される

$$\text{refIdxPredLX}=\text{refIdxILPredLX}[(xP+xS)/8,(xP+xS)/8]$$

$$\text{mvPredLX}[c]=\text{mvILPredLX}[(xP+xS)/4,(xP+xS)/4][c], c=0..1 \quad (\text{G-106})$$

ビットストリームは0より小さいrefIdxPredLXを生じるデータを含んではならない。

4. mvCntInc は(mvCntInc+1)に等しく設定される。
- それ以外、次の順序付けられたステップが規定される。
  1. mbType[CurrMbAddr]に依存して、参照インデックス予測因子 refIdxPredLX は以下により導出される。
    - もし mbType[CurrMbAddr]が P\_8x8ref0 に等しい場合、refIdxPredLX は 0 に等しく設定される。
    - それ以外、(mbType[CurrMbAddr] が P\_8x8ref0 に等しくない)、refIdxPredLX は ref\_idx\_lx[mbPartIdx]に等しく設定される。
  2. 8.4.1.3 小小節で規定される輝度動きベクトル予測の導出処理は mbPartIdx、subMbPartIdx、refIdxPredLX、subMbType[CurrMbAddr][mbPartIdx]に等しく設定された currSubMbType を入力として起動され、出力は mvPredLX に割り当てられる。8.4.1.3 小小節の処理のこの起動に対して、この小小節の上記 a)から j)項で規定される修正が適用される。
  3. mvCntInc は(mvCntInc+1)に等しく設定される。

0 と 1 に置き換えられる X に対して、配列 refIdxLX、predFlagLX、mvLX は、以下の順序付けられたステップの適用により修正される。

1. subMbPartIdx が 0 と等しい場合、配列 refIdxLX と predFlagLX は以下により修正される。

$$\text{refIdxLX}[\text{CurrMbAddr}][\text{mbPartIdx}]=\text{refIdxPredLX} \quad (\text{G-107})$$

$$\text{predFlagLX}[\text{CurrMbAddr}][\text{mbPartIdx}]=(\text{refIdxPredLX} < 0 \ ? \ 0 \ : \ 1) \quad (\text{G-108})$$

2. 配列 mvLX は以下により修正される。

$$\text{mvLX}[\text{CurrMbAddr}][\text{mbPartIdx}][\text{subMbPartIdx}][c]=\text{mvPredLX}[c], c=0..1 \quad (\text{G-109})$$

3. predFlagLX[CurrMbAddr][mbPartIdx]が 1 と等しく、base\_mode\_flag が 0 と等しく、isDirectFlag が 0 と等しく、mbType[CurrMbAddr]が P\_Skip と等しくない場合、配列 mvLX は以下により修正される。

$$\begin{aligned} \text{mvLX}[\text{CurrMbAddr}][\text{mbPartIdx}][\text{subMbPartIdx}][c]+= \\ \text{mvd\_lx}[\text{mbPartIdx}][\text{subMbPartIdx}][c], c=0..1 \end{aligned} \quad (\text{G-110})$$

配列 mvCnt は、以下により修正される。

- もし mbPartIdx が 0 と等しく、subMbPartIdx が 0 と等しいならば、mvCnt[CurrMbAddr]は mvCntInc に等しく設定される。
- それ以外(mbPartIdx が 0 より大きく、もしくは subMbPartIdx が 0 よりも大きい)、配列 mvCnt は以下に

より修正される。

$$mvCnt[CurrMbAddr] += mvCntInc \quad (G-111)$$

#### G.8.4.1.2 20に等しいnal\_unit\_typeのNALユニットにおけるB\_Skip、B\_Direct\_16x16、およびB\_Direct\_8x8の輝度動きベクトルと参照インデックスのSVC導出処理

この処理の入力は、次の通りである。

- 現マクロブロックパーティションを規定する変数 mbPartIdx
- 現レイヤ表現のどのマクロブロックがフィールドマクロブロックとして符号化されているのかとどのマクロブロックがフレームマクロブロックとして符号化されているのかを規定する PicSizeInMbs 個の要素をもつ1次元配列 fieldMbFlag
- 現レイヤ表現のマクロブロックのマクロブロックタイプを規定する PicSizeInMbs 個の要素をもつ1次元配列 mbType
- 現レイヤ表現のマクロブロックのサブマクロブロックタイプを規定する (PicSizeInMbs)×4 配列 subMbType
- 現レイヤ表現のマクロブロックの予測利用フラグを規定する 2つの(PicSizeInMbs)×4 配列 predFlagL0 および predFlagL1
- 現レイヤ表現のマクロブロックの参照インデックスを規定する 2つの(PicSizeInMbs)×4 配列 refIdxL0 および refIdxL1
- 現レイヤ表現のマクロブロックの動きベクトル成分を規定する 2つの(PicSizeInMbs)×4×4×2 配列 mvL0 および mvL1

この処理の出力は、次の通りである。

- 参照インデックス予測因子 refIdxPredL0 および refIdxPredL1
- 動きベクトル予測因子 mvPredL0 および mvPredL1
- 変数 mvCntInc

変数 currSubMbType は次の通り導出される。

- もし mbType[CurrMbAddr] が B\_Skip または B\_Direct\_16x16 に等しいならば、currSubMbType は“無規定”としてマークされる。
- それ以外 (mbType[CurrMbAddr] が B\_8x8 に等しくかつ subMbType[CurrMbAddr][mbPartIdx] が B\_Direct\_8x8 に等しい)、currSubMbType は B\_Bi\_8x8 に等しく設定される。

記 変数 currSubMbType は 6.4.10.7 小小節で変数 predPartWidth の導出に使用されるのみであり、動きベクトル予測のために使用される隣接パーティションを決定するために、現マクロブロックあるいはサブマクロブロックパーティションのパーティション幅を規定する。6.4.10.7 小小節の中では、現マクロブロックが B\_Skip あるいは B\_Direct\_16x16 に等しいマクロブロックタイプで符号化されるか、あるいは現サブマクロブロックが B\_Direct\_8x8 に等しいサブマクロブロックタイプで符号化される時、変数 predPartWidth は 16 に等しく設定される。現小小節は B\_Direct\_8x8 に等しいサブマクロブロックタイプで符号化されるサブマクロブロックに対して起動される時(現小小節は 20 に等しい nal\_unit\_type をもつ NAL ユニットに対して起動されるのみ)、currSubMbType は 6.4.10.7 小小節で 8 に等しい変数 predPartWidth を設定するために B\_Bi\_8x8 に等しく設定される。

0 または 1 のいずれかに置換される X に対して、参照インデックス予測因子  $\text{refIdxPredLX}$  は、次の順序付けられたステップを適用することで導出される。

1. 8.4.1.3.2 小小節で規定される隣接パーティション動きデータの導出処理は、入力として  $\text{mbPartIdx}$  および 0 に等しく設定された  $\text{subMbPartIdx}$ 、 $\text{currSubMbType}$ 、X に等しく設定された  $\text{listSuffixFlag}$  を用いて起動され、その出力は A、B、および C に置換される N をもつ参照インデックス  $\text{refIdxLXN}$  である。8.4.1.3.2 小小節の処理のこの起動に対して、8.4.1.1 小小節の a) から j) 項で規定される修正が適用される。
2. 参照インデックス予測因子  $\text{refIdxPredLX}$  は次のように導出される。

$$\text{refIdxPredLX} = \text{MinPositive}(\text{refIdxLXA}, \text{MinPositive}(\text{refIdxLXB}, \text{refIdxLXC})), \quad (\text{G-112})$$

$$\text{MinPositive}(x, y) = \begin{cases} \text{Min}(x, y) & \text{if } x \geq 0 \text{ and } y \geq 0 \\ \text{Max}(x, y) & \text{otherwise} \end{cases} \quad (\text{G-113})$$

参照インデックス予測因子  $\text{refIdxPredL0}$  および  $\text{refIdxPredL1}$  の両方が 0 より小さい時、 $\text{refIdxPredL0}$  および  $\text{refIdxPredL1}$  は 0 に等しく設定される。

0 または 1 のいずれかに置換される X に対して、動きベクトル予測因子  $\text{mvPredLX}$  は、次のように導出される。

- もし  $\text{refIdxPredLX}$  が 0 より大きいあるいは等しいならば、8.4.1.3 小小節に規定される輝度動きベクトル予測の導出処理が、入力として  $\text{mbPartIdx}$  および 0 に等しく設定された  $\text{subMbPartIdx}$ 、 $\text{refIdxPredLX}$ 、 $\text{currSubMbType}$  を用いて起動され、出力は  $\text{mvPredLX}$  に割り当てられる。8.4.1.3 小小節の処理のこの起動に対して、G.8.4.1.1 小小節の a) から j) 項で規定される修正が適用される。
- それ以外、動きベクトル  $\text{mvPredLX}$  の両成分は 0 に等しく設定される。

変数  $\text{mvCntInc}$  は、次の順序付けられたステップで規定され導出される。

1.  $\text{mvCntInc}$  は 0 に等しく設定される。
2.  $\text{refIdxPredL0}$  が 0 より大きいあるいは等しい時、 $\text{mvCntInc}$  は  $(\text{mvCntInc}+1)$  に等しく設定される。
3.  $\text{refIdxPredL1}$  が 0 より大きいあるいは等しい時、 $\text{mvCntInc}$  は  $(\text{mvCntInc}+1)$  に等しく設定される。

#### G.8.4.2 INTER予測サンプルのSVC復号処理

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数  $\text{fieldMbFlag}$
- 現マクロブロックのマクロブロックタイプを規定する変数  $\text{mbType}$
- 現マクロブロックのサブマクロブロックタイプを規定する 4 要素をもつリスト  $\text{subMbType}$
- 現マクロブロックの予測利用フラグを規定する 4 要素をもつ 2 つのリスト  $\text{predFlagL0}$  および  $\text{predFlagL1}$
- 現マクロブロックの予測インデックスを規定する 4 要素をもつ 2 つのリスト  $\text{refIdxL0}$  および  $\text{refIdxL1}$
- 現マクロブロックの動きベクトル成分を規定する 2 つの  $4 \times 4 \times 2$  配列  $\text{mvL0}$  および  $\text{mvL1}$

- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックであるのかとどのマクロブロックがフレームマクロブロックであるのかを規定する RefLayerPicSizeInMbs 個の要素をもつ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックのマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素をもつ 1 次元配列 refLayerMbType
- 参照ピクチャリスト refPicList0 および (利用可能であれば) refPicList1
- 輝度サンプル値の(PicWidthInSamples<sub>L</sub>)×(PicHeightInSamples<sub>L</sub>)配列 picSamples<sub>L</sub>
- ChromaArrayType が 0 に等しくない時、色差サンプル値の 2 つの (PicWidthInSamples<sub>C</sub>)×(PicHeightInSamples<sub>C</sub>)配列 picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub>

この処理の出力は、次の通りである。

- 輝度サンプル値配列の修正版 picSamples<sub>L</sub>
- ChromaArrayType が 0 に等しくない時、色差サンプル値の 2 つの配列の修正版 picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub>

この小小節で規定される処理またはこの小小節で規定される処理の子処理から起動される 6 または 8 章で規定される全ての処理に対して、次の修正が適用される。

- a) 8.4.3 小節および 8.4.1.4、8.4.2.1 小小節において、現マクロブロックは fieldMbFlag が 1 に等しい時、フィールドマクロブロックとして扱われ、fieldMbFlag が 0 に等しい時、フレームマクロブロックとして扱われる。field\_pic\_flag が 0 に等しく、かつ現マクロブロック CurrMbAddr がフィールドマクロブロックである時、(CurrMbAddr%2)が 0 に等しい時そのパリティはトップに等しく、(CurrMbAddr%2)が 1 に等しい時そのパリティはボトムに等しい。
- b) 8.4.3 小節および 8.4.2.1 小小節において現れる全ての RefPicList0 または RefPicList1 は、各々、この小小節への入力として規定される参照ピクチャリストである refPicList0 および refPicList1 に置き換えられる。
- c) 8.4.1.4 小小節において、refIdxLX によって参照される参照ピクチャは、この小小節への入力として規定される refPicList0 および refPicList1 をもつ refPicListX[refIdxLX]によって規定される。
- d) 8.4.2.2.1 および 8.4.2.2.2 小小節において現れる全ての mb\_field\_decoding\_flag は、fieldMbFlag に置き換えられる。

predMb<sub>L</sub> をマクロブロック mbAddr の輝度予測サンプルの 16×16 配列とする。

ChromaArrayType が 0 に等しくない時、predMb<sub>Cb</sub> および predMb<sub>Cr</sub> をマクロブロック mbAddr の色差予測サンプルの 2 つの (MbWidthC)×(MbHeightC)配列とする。

変数 isDirectFlag は次のように導出される。

- もし次の条件のいずれかが真ならば、isDirectFlag は 1 に等しく設定される。
  - mbType が B\_Skip または B\_Direct\_16x16 に等しい。
  - mbType が B\_8x8 に等しく、かつ subMbType[mbPartIdx]が B\_Direct\_8x8 に等しい。

- それ以外、isDirectFlag は 0 に等しく設定される。

変数 numMbPart は次のように導出される。

- もし mbType が B\_Skip または B\_Direct\_16x16 に等しい、かつ DQId が 0 に等しい (nal\_unit\_type が 20 に等しくない) ならば、numMbPart は 4 に等しく設定される。
- それ以外、もし mbType が B\_Skip または B\_Direct\_16x16 に等しい (かつ DQId が 0 より大きくかつ nal\_unit\_type が 20 に等しい) ならば、numMbPart は 1 に等しく設定される。
- それ以外 (mbType が B\_Skip または B\_Direct\_16x16 に等しくない)、numMbPart は NumMbPart(mbType) に等しく設定される。

マクロブロックパーティションインデックス mbPartIdx は値 0..(numMbPart-1)上を進行し、mbPartIdx のそれぞれの値に対して次の順序付けられたステップが規定される。

1. G.8.4.2.1 小小節で規定される予測重みの SVC 導出処理が fieldMbFlag および refIdxL0[mbPartIdx]、refIdxL1[mbPartIdx]、predFlagL0[mbPartIdx]、predFlagL1[mbPartIdx]、refPicList0、(利用可能であれば) refPicList1 を入力として起動され、その出力は logWDL および w<sub>0L</sub>、w<sub>1L</sub>、o<sub>0L</sub>、o<sub>1L</sub>、さらに ChromaArrayType が 0 に等しくない時、Cb または Cr のいずれかに置換される C をもつ logWDC、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>0C</sub>、o<sub>1C</sub> に割り当てられる。
2. 輝度位置(xP,yP)は次のように導出される。
  - もし mbType が B\_Skip または B\_Direct\_16x16 に等しいならば、xP が(8\*(mbPartIdx%2))に等しく設定され、yP が(8\*(mbPartIdx/2))に等しく設定される。
  - それ以外 (mbType が B\_Skip または B\_Direct\_16x16 に等しくない)、6.4.2.1 小小節で規定される逆マクロブロックパーティション走査処理が mbPartIdx を入力として起動され、その出力は (xP,yP)に割り当てられる。6.4.2.1 小小節におけるこの処理の起動において現れる全ての mb\_type は mbType に置き換えられる。
3. 変数 numSubMbPart は次のように導出される。
  - もし isDirectFlag が 1 に等しい、かつ DQId が 0 に等しい (nal\_unit\_type が 20 に等しくない) ならば、numSubMbPart は 4 に等しく設定される。
  - それ以外、もし isDirectFlag が 1 に等しい (かつ DQId が 0 より大きくかつ nal\_unit\_type が 20 に等しい) ならば、numMbPart は 1 に等しく設定される。
  - それ以外 (isDirectFlag が 0 に等しい)、numMbPart は NumSubMbPart(subMbType[mbPartIdx])に等しく設定される。
4. サブマクロブロックパーティションインデックスは値 0..(numSubMbPart-1)上を進行し、subMbPartIdx のそれぞれの値に対して次の順序付けられたステップが規定される。
  - a. 変数 partWidth および partHeight は次のように導出される。
    - もし isDirectFlag が 1 に等しい、かつ DQId が 0 に等しい (nal\_unit\_type が 20 に等しくない) ならば、partWidth および partHeight は 4 に等しく設定される。
    - それ以外、もし isDirectFlag が 1 に等しい (かつ DQId が 0 より大きくかつ nal\_unit\_type が 20

に等しい) ならば、次が適用される。

- もし mbType が B\_Skip または B\_Direct\_16x16 に等しいならば、partWidth および partHeight は 16 に等しく設定される。
- それ以外 (mbType が B\_8x8 に等しく、かつ subMbType[mbPartIdx]が B\_Direct\_8x8 に等しい) 、 partWidth および partHeight は 8 に等しく設定される。
- それ以外 (isDirectFlag が 0 に等しい)、次が適用される。
  - もし mbType が P\_8x8 または P\_8x8ref0、B\_8x8 に等しくないならば、partWidth および partHeight は次によって導出される。

$$\text{partWidth}=\text{MbPartWidth}(\text{mbType}) \quad (\text{G-114})$$

$$\text{partHeight}=\text{MbPartHeight}(\text{mbType}) \quad (\text{G-115})$$

- それ以外 (mbType が P\_8x8 または P\_8x8ref0、B\_8x8 に等しい) 、partWidth および partHeight は次によって導出される。

$$\text{partWidth}=\text{SubMbPartWidth}(\text{subMbType}[\text{mbPartIdx}]) \quad (\text{G-116})$$

$$\text{partHeight}=\text{SubMbPartHeight}(\text{subMbType}[\text{mbPartIdx}]) \quad (\text{G-117})$$

- b. ChromaArrayType が 0 に等しくない時、変数 partWidthC および partHeightC は次によって導出される。

$$\text{partWidthC}=\text{partWidth}/\text{SubWidthC} \quad (\text{G-118})$$

$$\text{partHeightC}=\text{partHeight}/\text{SubWidthC} \quad (\text{G-119})$$

- c. 0 または 1 のいずれかに置換される X に対して、ChromaArrayType が 0 に等しくなく、かつ predFlagLX[mbPartIdx]が 1 に等しい時、8.4.1.4 小小節で規定される色差動きベクトル導出処理が mvLX[mbPartIdx][subMbPartIdx]および refIdxLX[mbPartIdx]を入力として起動され、その出力は色差動きベクトル mvCLX である。8.4.1.4 小小節の処理のこの起動に対して、この小小節の上の a)および c)項で規定される修正が適用される。
- d. 8.4.2 小節で規定される INTER 予測サンプルの復号処理は、mbPartIdx および、subMbPartIdx、partWidth、partHeight、(利用可能ならば) partWidthC、輝度動きベクトル mvL0[mbPartIdx][subMbPartIdx]および mvL1[mbPartIdx][subMbPartIdx]、(利用可能ならば) 色差動きベクトル mvCL0 および mvCL1、参照インデックス refIdxL0[mbPartIdx]および refIdxL1[mbPartIdx]、予測利用フラグ predFlagL0[mbPartIdx]および predFlagL1[mbPartIdx]、および重みづけされた予測の変数 logWD<sub>L</sub> および w<sub>0L</sub>、w<sub>1L</sub>、o<sub>1L</sub>、o<sub>0L</sub>、さらに ChromaArrayType が 0 に等しくない時、(Cb または Cr のいずれかに置換される C をもつ) logWD<sub>C</sub>、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>1C</sub>、o<sub>0C</sub> を入力として起動され、その出力は輝度予測サンプルの (partWidth)×(partHeight)配列 predPartL さらに ChromaArrayType が 0 に等しくない時、色差予測サンプルの 2 つの (partWidthC)×(partHeightC)配列 predPart<sub>Cb</sub> および predPart<sub>Cr</sub> である。8.4.2 小節の処理のこの起動に対して、この小小節の上の a)、b)、および d)項で規定される修正が適用される。

- e. 輝度位置(xS,yS)が次のように導出される。
- もし mbType が B\_8x8 に等しくないか、または subMbType[mbPartIdx]が B\_Direct\_8x8 に等しいならば、xS は(4\*(subMbPartIdx%2))に等しく設定され、かつ yS は(4\*(subMbPartIdx/2))に等しく設定される。
  - それ以外 (mbType が B\_8x8 に等しく、かつ subMbType[mbPartIdx]が B\_Direct\_8x8 に等しくない)、6.4.2.2 小小節で規定される逆サブマクロブロックパーティション走査処理が mbPartIdx および subMbPartIdx を入力として起動され、その出力は(xS,yS)に割り当てられる。6.4.2.2 小小節の処理のこの起動において現れる全ての mb\_type は mbType に置き換えられ、現れる全ての sub\_mb\_type は subMbType に置き換えられる。
- f.  $x=0..(\text{partWidth}-1)$ および  $y=0..(\text{partHeight}-1)$ に対して、 $16 \times 16$  配列  $\text{predMb}_L$  は次によって修正される。

$$\text{predMb}_L[xP+xS+x,yP+yS+y]=\text{predPart}_L[x,y] \quad (\text{G-120})$$

- g. ChromaArrayType が 0 に等しくない時、 $x=0..(\text{partWidthC}-1)$ および  $y=0..(\text{partHeightC}-1)$ に対して、 $(\text{MbWidthC}) \times (\text{MbHeightC})$ 配列  $\text{predMb}_{Cb}$  および  $\text{predMb}_{Cr}$  は次によって修正される。

$$\text{predMb}_{Cb}[(xP+xS)/\text{SubWidthC}+x,(yP+yS)/\text{SubHeightC}+y]=\text{predPart}_{Cb}[x,y] \quad (\text{G-121})$$

$$\text{predMb}_{Cr}[(xP+xS)/\text{SubWidthC}+x,(yP+yS)/\text{SubHeightC}+y]=\text{predPart}_{Cr}[x,y] \quad (\text{G-122})$$

base\_mode\_flag が 1 に等しい、かつ quality\_id が 0 に等しい、かつ MbaffFrameFlag が 0 に等しい、かつ RefLayerMbaffFrameFlag が 0 に等しい、かつ RestrictedSpatialResolutionChangeFlag が 0 に等しいとき、G.8.4.2.2 小小小節で規定される INTRA-INTER 予測組み合わせ処理が fieldMbFlag および refLayerFieldMbFlag、refLayerMbType、predMb<sub>L</sub>、picSamples<sub>L</sub>、さらに ChromaArrayType が 0 に等しくない時、predMb<sub>Cb</sub>、predMb<sub>Cr</sub>、picSamples<sub>Cb</sub>、picSamples<sub>Cr</sub> で起動され、その出力は picSamples<sub>L</sub> の修正版、さらに ChromaArrayType が 0 に等しくない時、picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub> の修正版である。

#### G.8.4.2.1 予測重みのSVC導出処理

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 現マクロブロックパーティションの参照インデックス refIdxL0 および refIdxL1
- 現マクロブロックパーティションの予測リスト利用フラグ predFlagL0 および predFlagL1
- 参照ピクチャリスト refPicList0 および (利用可能ならば) refPicList1

この処理の出力は、次の通りである。

- 輝度サンプルの重みづけされた予測の変数 logWD<sub>L</sub> および w<sub>0L</sub>、w<sub>1L</sub>、o<sub>0L</sub>、o<sub>1L</sub>
- ChromaArrayType が 0 に等しくない時 (モノクローム)、Cb または Cr のいずれかに置換される C をもつ色差サンプルの重みづけされた予測の変数 logWD<sub>C</sub>、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>0C</sub>、o<sub>1C</sub>

base\_pred\_weight\_table\_flag に依存して、次が適用される。

- もし base\_pred\_weight\_table\_flag が 0 ならば、8.4.3 小節で規定される予測重みの導出処理が refIdxL0、refIdxL1、predFlagL0、predFlagL1 を入力として起動され、その出力は logWD<sub>L</sub> および w<sub>0L</sub>、w<sub>1L</sub>、o<sub>0L</sub>、o<sub>1L</sub>、さらに ChromaArrayType が 0 に等しくない時、Cb または Cr のいずれかに置換される C をもつ logWD<sub>C</sub>、w<sub>0C</sub>、w<sub>1C</sub>、o<sub>0C</sub>、o<sub>1C</sub> である。8.4.3 小節におけるこの処理の起動に対して、G.8.4.2 小節の a) および b) 項で規定される修正が適用される。
- それ以外(base\_pred\_weight\_table\_flag が 1 に等しい)、0 または 1 のいずれかに置換される X に対して、次の順序付けられたステップが規定される。

1. baseSlice を ref\_layer\_dq\_id の値と等しい DQId をもつレイヤ表現の任意のスライスとする。
2. refLayerLumaLogWD および aRefLayerLumaWeightLX[]、aRefLayerLumaOffsetLX[] をそれぞれ baseSlice のシンタックス要素 luma\_log2\_weight\_denom および luma\_weight\_IX[]、luma\_offset\_IX[] の値と等しく設定される変数とする。
3. ChromaArrayType が 0 に等しくない時、refLayerChromaLogWD および aRefLayerChromaWeightLX[][]、aRefLayerChromaOffsetLX[][] をそれぞれ baseSlice のシンタックス要素 chroma\_log2\_weight\_denom および chroma\_weight\_IX[]、chroma\_offset\_IX[] の値と等しく設定される変数とする。
4. 変数 refIdxLXWP は次のように導出される。

- もし MbaffFrame が 1 に等しい、かつ fieldMbFlag が 1 に等しいならば、

$$\text{refIdxLXWP}=\text{refIdxLX}>>1 \quad (\text{G-123})$$

- それ以外(MbaffFrame が 0 に等しい、または fieldMbFlag が 0 に等しい)、

$$\text{refIdxLXWP}=\text{refIdxLX} \quad (\text{G-124})$$

5. 変数 logWD<sub>Y</sub>、w<sub>XY</sub>、o<sub>XY</sub> は次によって導出される。

$$\text{logWD}_Y=\text{baseLumaLogWD} \quad (\text{G-125})$$

$$w_{XY}=\text{aRefLayerLumaWeightLX}[\text{refIdxLXWP}] \quad (\text{G-126})$$

$$o_{XY}=\text{aRefLayerLumaOffsetLX}[\text{refIdxLXWP}]*(\text{1}<<(\text{BitDepth}_Y-8)) \quad (\text{G-127})$$

6. ChromaArrayType が 0 に等しくない時、(Cb または Cr のいずれかに置換される C、および Cb に対して iCbCr=0、Cr に対して iCbCr=1 を用いて) 変数 logWD<sub>C</sub>、w<sub>XC</sub>、o<sub>XC</sub> は次によって導出される。

$$\text{logWD}_C=\text{baseChromaLogWD} \quad (\text{G-128})$$

$$w_{XC}=\text{aRefLayerChromaWeightLX}[\text{refIdxLXWP}][\text{iCbCr}] \quad (\text{G-129})$$

$$o_{XC}=\text{aRefLayerChromaOffsetLX}[\text{refIdxLXWP}][\text{iCbCr}]*(\text{1}<<(\text{BitDepth}_C-8)) \quad (\text{G-130})$$

7. base\_pred\_weight\_table\_flag が 1 に等しく、かつ predFlagL0 および predFlagL1 が 1 に等しい時、L および ChromaArrayType が 0 に等しくない時は Cb および Cr に等しい C に対して、次の制約に従わなければならない。

$$-128 \leq w_{0c} + w_{1c} \leq ((\log W_{Dc} == 7) ? 127 : 128)$$

(G-131)

#### G.8.4.2.2 INTRA-INTER予測組み合わせ処理

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックであるのかとどのマクロブロックがフレームマクロブロックであるのかを規定する `RefLayerPicSizeInMbs` 個の要素をもつ 1 次元配列 `refLayerFieldMbFlag`
- 参照レイヤ表現のマクロブロックのマクロブロックタイプを規定する `RefLayerPicSizeInMbs` 個の要素をもつ 1 次元配列 `refLayerMbType`
- 現マクロブロックの輝度 INTER 予測サンプルの  $16 \times 16$  配列 `predMb_L`
- 輝度サンプル値の  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 `picSamples_L`
- `ChromaArrayType` が 0 に等しくない時、マクロブロック `mbAddr` の色差予測サンプルの 2 つの  $(\text{MbWidthC}) \times (\text{MbHeightC})$  配列 `predMb_Cb` および `predMb_Cr`
- `ChromaArrayType` が 0 に等しくない時、色差サンプル値の 2 つの  $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$  配列 `picSamples_Cb` および `picSamples_Cr`

この処理の出力は、次の通りである。

- マクロブロック `mbAddr` の輝度予測サンプル配列 `predMb_L` の修正版
- `ChromaArrayType` が 0 に等しくない時、マクロブロック `mbAddr` の色差予測サンプル配列 `predMb_Cb` および `predMb_Cr` の修正版

`predMbTemp_L` を  $16 \times 16$  配列とし、さらに `ChromaArrayType` が 0 に等しくない時、`predMbTemp_Cb` および `predMbTemp_Cr` を 2 つの  $(\text{MbWidthC}) \times (\text{MbHeightC})$  配列とする。G.8.5.4.2 小小小節で規定されるマクロブロックサンプル配列抽出処理は `fieldMbFlag` および `picSamples_L`、さらに `ChromaArrayType` が 0 に等しくない時、`picSamples_Cb`、`picSamples_Cr` を入力として起動され、その出力は `predMbTemp_L` および `ChromaArrayType` が 0 に等しくない時、`predMbTemp_Cb`、`predMbTemp_Cr` に割り当てられる。

値 0..15 上を進行する `x` および値 0..15 上を進行する `y` に対して、次の順序付けられたステップが規定される。

1. G.6.1 小節で規定される参照レイヤマクロブロックの導出処理が  $(x,y)$  および `fieldMbFlag`、`refLayerFieldMbFlag`、`refLayerMbType` を入力として起動され、その出力は `mbAddrRefLayer` および  $(xRef,yRef)$  に割り当てられる。
2. `refLayerMbType[mbAddrRefLayer]` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、または `I_BL` に等しい時、次が適用される。
  - a. 予測輝度サンプル `predMb_L[x,y]` は次によって修正される。

$$\text{predMb}_L[x,y] = \text{predMbTemp}_L[x,y]$$

(G-132)

- b. ChromaArrayType が 0 に等しくない、かつ(x%SubWidthC)が 0 に等しい、かつ(y%SubHeightC)が 0 に等しい時、予測色差サンプル  $\text{predMb}_{Cb}[x,y]$  および  $\text{predMb}_{Cr}[x,y]$  は次によって修正される。

$$\text{predMb}_{Cb}[x,y]=\text{predMbTemp}_{Cb}[x,y] \quad (\text{G-133})$$

$$\text{predMb}_{Cr}[x,y]=\text{predMbTemp}_{Cr}[x,y] \quad (\text{G-134})$$

## G.8.5 SVC変換係数復号処理とサンプル配列構築処理

G.8.5.1 小小節は、変換係数スケーリングおよび改善処理を規定する。

G.8.5.2 小小節は、変換係数改善前の変換係数レベルスケーリング処理を規定する。

G.8.5.3 小小節は、残差構築および累算処理を規定する。

G.8.5.4 小小節は、サンプル配列累算処理を規定する。

G.8.5.5 小小節は、サンプル配列再初期化処理を規定する。

### G.8.5.1 変換係数スケーリングおよび改善処理

この処理の入力は、次の通りである。

- 現マクロブロックの変換係数が、参照レイヤ表現から得られた現マクロブロックの変換係数と組合わされるかを規定する変数 `refinementFlag`
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- 変換タイプを規定する変数 `cTrafo`
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 個の要素をもつ、スケーリングされた変換係数値のリスト `sTCoeff`
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 個の要素をもつ、変換係数レベル値のリスト `tCoeffLevel`

この処理の出力は、次の通りである。

- リスト `sTCoeff` の修正版
- リスト `tCoeffLevel` の修正版

スケーリング関数および、ChromaArrayType が 0 に等しくない時、色差量子化パラメータ  $\text{QP}'_{Cb}$  および  $\text{QP}'_{Cr}$  は 8.5.8 小節で規定される色差量子化パラメータとスケーリング関数の導出処理を起動することで導出される。

`refinementFlag` が 0 に等しいとき、リスト `sTCoeff` および `tCoeffLevel` の全 $(256+2*\text{MbWidthC}*\text{MbHeightC})$ の要素が 0 に等しく設定される。

G.8.5.1.1 小小節で規定される輝度変換係数の改善処理は `fieldMbFlag`、`cTrafo`、`sTCoeff`、および `tCoeffLevel` を入力として起動され、その出力はリスト `sTCoeff` および `tCoeffLevel` の修正版である。

ChromaArrayType が 0 に等しくない時、G.8.5.1.2 小小節で規定される色差変換係数の改善処理は `fieldMbFlag` および `cTrafo`、`sTCoeff`、`tCoeffLevel` を入力として起動され、その出力はリスト `sTCoeff` および

tCoeffLevel の修正版である。

#### G.8.5.1.1 輝度変換係数または 3 に等しいChromaArrayTypeの色差変換係数の改善処理

この処理の入力は、次の通りである。

- (存在する時) 色成分を規定する変数 iYCbCr
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 変換タイプを規定する変数 cTrafo
- $(256+2*MbWidthC*MbHeightC)$ 個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff
- $(256+2*MbWidthC*MbHeightC)$ 個の要素をもつ、変換係数レベル値のリスト tCoeffLevel

この処理の出力は、次の通りである。

- リスト sTCoeff の修正版
- リスト tCoeffLevel の修正版

この小小小節の入力として iYCbCr が存在しない時、それは 0 に等しいと推定される。

iYCbCr に依存して、変数 bitDepth、qP、cO、coeffLevel、coeffLevel8x8、coeffDCLevel、および coeffACLevel は、次のように導出される。

- もし iYCbCr が 0 に等しいならば、bitDepth は  $BitDepth_Y$  に等しく設定され、qP は  $QP'_Y$  に等しく設定され、cO は 0 に等しく設定され、coeffLevel は LumaLevel に等しく設定され、coeffLevel8x8 は LumaLevel8x8 に等しく設定され、coeffDCLevel は Intra16x16DCLevel に等しく設定され、coeffACLevel は Intra16x16ACLevel に等しく設定される。
- それ以外、もし iYCbCr が 1 に等しいならば、bitDepth は  $BitDepth_C$  に等しく設定され、qP は  $QP'_{Cb}$  に等しく設定され、cO は 256 に等しく設定され、coeffLevel は CbLevel に等しく設定され、coeffLevel8x8 は CbLevel8x8 に等しく設定され、coeffDCLevel は CbIntra16x16DCLevel に等しく設定され、coeffACLevel は CbIntra16x16ACLevel に等しく設定される。
- それ以外 (iYCbCr が 2 に等しい)、bitDepth は  $BitDepth_C$  に等しく設定され、qP は  $QP'_{Cr}$  に等しく設定され、cO は  $(256+MbWidthC*MbHeightC)$  に等しく設定され、coeffLevel は CrLevel に等しく設定され、coeffLevel8x8 は CrLevel8x8 に等しく設定され、coeffDCLevel は CrIntra16x16DCLevel に等しく設定され、coeffACLevel は CrIntra16x16ACLevel に等しく設定される。

cTrafo に依存して、次が適用される。

- もし cTrafo が T\_PCM に等しいならば、G.8.5.1.1.1 小小小節で規定される I\_PCM マクロブロックの輝度変換係数値あるいは 3 に等しい ChromaArrayType をもつ色差変換係数値の割り当て処理が iYCbCr、sTCoeff、および tCoeffLevel を入力として起動され、その出力は sTCoeff および tCoeffLevel の修正版である。
- それ以外、もし cTrafo が T\_4x4 に等しいならば、G.8.5.1.1.2 小小小節で規定される残差 4x4 ブロックの変換係数の改善処理が fieldMbFlag、bitDepth、qP、cO、coeffLevel、sTCoeff、および tCoeffLevel を入

力として起動され、その出力はリスト sTCoeff および tCoeffLevel の修正版である。

- それ以外、もし cTrafo が T\_8x8 に等しいならば、G.8.5.1.1.3 小小小節で規定される残差 8×8 ブロックの変換係数の改善処理が fieldMbFlag、bitDepth、qP、cO、coeffLevel8x8、sTCoeff、および tCoeffLevel を入力として起動され、その出力はリスト sTCoeff および tCoeffLevel の修正版である。
- それ以外 (cTrafo が T\_16x16 に等しい)、G.8.5.1.1.4 小小小節で規定される Intra\_16x16 マクロブロックの変換係数の改善処理が fieldMbFlag、bitDepth、qP、cO、coeffDCLevel、coeffACLevel、coeffLevel8x8、sTCoeff、および tCoeffLevel を入力として起動され、その出力はリスト sTCoeff および tCoeffLevel の修正版である。

#### G.8.5.1.1.1 L\_PCM マクロブロックの輝度変換係数値または 3 に等しいChromaArrayTypeの色差変換係数値の割り当て処理

この処理の入力は、次の通りである。

- 色成分を規定する変数 iYCbCr
- (256+2\*MbWidthC\*MbHeightC)個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff
- (256+2\*MbWidthC\*MbHeightC)個の要素をもつ、変換係数レベル値のリスト tCoeffLevel

この処理の出力は、次の通りである。

- リスト sTCoeff の修正版
- リスト tCoeffLevel の修正版

iYCbCr に依存して、変数 cO、cListOffset、および pcmSample は、次によって導出される。

$$cO = iYCbCr * 256$$

$$cListOffset = (iYCbCr == 0 ? 0 : (iYCbCr - 1) * 256) \quad (G-135)$$

$$pcmSample = (iYCbCr == 0 ? pcm_luma\_samples : pcm\_chroma\_samples) \quad (G-136)$$

base\_mode\_flag が 0 に等しいとき、リスト tCoeffLevel および sTCoeff は次によって導出される。

$$tCoeffLevel[cO+k] = 0, \quad k=0..255 \quad (G-137)$$

$$sTCoeff[cO+k] = pcmSample[cListOffset+k], \quad k=0..255 \quad (G-138)$$

#### G.8.5.1.1.2 残差 4×4 ブロックの変換係数の改善処理

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- ビット深度を規定する変数 bitDepth
- 量子化パラメータ値を規定する変数 qP
- スケーリングされた変換係数値のリスト sTCoeff および変換係数値のリスト tCoeffLevel の中で最初の係

数インデックスを規定する変数  $cO$

- LumaLevel または CbLevel、CrLevel を表現する変数  $coeffLevel$
- $(256+2*MbWidthC*MbHeightC)$ 個の要素をもつ、スケーリングされた変換係数値のリスト  $sTCoeff$
- $(256+2*MbWidthC*MbHeightC)$ 個の要素をもつ、変換係数レベル値のリスト  $tCoeffLevel$

この処理の出力は、次の通りである。

- リスト  $sTCoeff$  の修正版
- リスト  $tCoeffLevel$  の修正版

$tcoeff\_level\_prediction\_flag$  に依存して、次が適用される。

- もし  $tcoeff\_level\_prediction\_flag$  が 1 に等しいならば、リスト  $sTCoeff$  は次によって修正される。

$$sTCoeff[cO+k]=0, k=0..255 \quad (G-139)$$

- それ以外 ( $tcoeff\_level\_prediction\_flag$  が 0 に等しい)、リスト  $tCoeffLevel$  は次によって修正される。

$$tCoeffLevel[cO+k]=0, k=0..255 \quad (G-140)$$

$c4x4BlkIdx=0..15$  で索引される残差  $4\times 4$  ブロックそれぞれに対して、次の順序付けられたステップが規定される。

1. 8.5.6 小節で規定される変換係数の逆走査処理は  $coeffLevel[c4x4BlkIdx]$  を入力として起動され、その出力は要素  $c_{ij}$  をもつ  $4\times 4$  配列  $c$  とする変換係数レベル値である。8.5.6 小節で規定されるこの処理の起動に対して、現マクロブロックは  $fieldMbFlag$  が 1 に等しい時フィールドマクロブロックとして扱われ、現マクロブロックは  $fieldMbFlag$  が 0 に等しい時フレームマクロブロックとして扱われる。
2. リスト  $tCoeffLevel$  および  $4\times 4$  配列  $c$  は次によって修正される。

$$tCoeffLevel[cO+16*c4x4BlkIdx+4*i+j]=c_{ij}, i,j=0..3 \quad (G-141)$$

$$c_{ij}=tCoeffLevel[cO+16*c4x4BlkIdx+4*i+j], i,j=0..3 \quad (G-142)$$

3. 8.5.11.1 小小節で規定される残差  $4\times 4$  ブロックのスケーリング処理は  $bitDepth$ 、 $qP$ 、および  $4\times 4$  配列  $c$  を入力として起動され、その出力は要素  $d_{ij}$  をもつ  $4\times 4$  配列  $d$  とするスケーリングされた変換係数値である。8.5.11.1 小小節におけるこの処理の起動に対して、配列  $c$  は Intra\_16x16 予測モードを用いて符号化された輝度残差ブロックに関係ないとして取り扱われ、また色差残差ブロックに関係ないとして取り扱われる。
4. リスト  $sTCoeff$  は次によって修正される。

$$sTCoeff[cO+16*c4x4BlkIdx+4*i+j]=d_{ij}, i,j=0..3 \quad (G-143)$$

ビットストリームは、任意の要素  $sTCoeff[cO+k]$ 、 $k=0..255$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

ビットストリームは、任意の要素  $tCoeffLevel[cO+k]$ 、 $k=0..255$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの

値を含む整数値範囲を超える結果となるデータを含んではならない。

#### G.8.5.1.1.3 残差 8×8 ブロックの変換係数の改善処理

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- ビット深度を規定する変数 `bitDepth`
- 量子化パラメータ値を規定する変数 `qP`
- スケーリングされた変換係数値のリスト `sTCoeff` および変換係数値のリスト `tCoeffLevel` の中で最初の係数インデックスを規定する変数 `cO`
- `LumaLevel8x8`、`CbLevel8x8`、または `CrLevel8x8` を表現する変数 `coeffLevel8x8`
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 個の要素をもつ、スケーリングされた変換係数値のリスト `sTCoeff`
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 個の要素をもつ、変換係数レベル値のリスト `tCoeffLevel`

この処理の出力は、次の通りである。

- リスト `sTCoeff` の修正版
- リスト `tCoeffLevel` の修正版

`tcoeff_level_prediction_flag` に依存して、次が適用される。

- もし `tcoeff_level_prediction_flag` が 1 に等しいならば、リスト `sTCoeff` は次によって修正される。

$$\text{sTCoeff}[\text{cO}+\text{k}]=0, \text{k}=0..255 \quad (\text{G-144})$$

- それ以外 (`tcoeff_level_prediction_flag` が 0 に等しい)、リスト `tCoeffLevel` は次によって修正される。

$$\text{tCoeffLevel}[\text{cO}+\text{k}]=0, \text{k}=0..255 \quad (\text{G-145})$$

`c8x8BlkIdx=0..3` で索引される残差 8×8 ブロックそれぞれに対して、次の順序付けられたステップが規定される。

1. 8.5.7 小節で規定される 8×8 変換係数の逆走査処理は `coeffLevel8x8[c8x8BlkIdx]` を入力として起動され、その出力は要素 `cij` をもつ 8×8 配列 `c` とする変換係数レベル値である。8.5.7 小節で規定されるこの処理の起動に対して、現マクロブロックは `fieldMbFlag` が 1 に等しい時フィールドマクロブロックとして扱われ、現マクロブロックは `fieldMbFlag` が 0 に等しい時フレームマクロブロックとして扱われる。
2. リスト `tCoeffLevel` および 8×8 配列 `c` は次によって修正される。

$$\text{tCoeffLevel}[\text{cO}+64*\text{c8x8BlkIdx}+8*\text{i}+\text{j}]=\text{c}_{\text{ij}}, \text{i}, \text{j}=0..7 \quad (\text{G-146})$$

$$\text{c}_{\text{ij}}=\text{tCoeffLevel}[\text{cO}+64*\text{c8x8BlkIdx}+8*\text{i}+\text{j}], \text{i}, \text{j}=0..7 \quad (\text{G-147})$$

3. 8.5.12.1 小小節で規定される残差 8×8 ブロックのスケーリング処理は bitDepth、qP、および 8×8 配列 c を入力として起動され、その出力は要素  $d_{ij}$  をもつ 8×8 配列 d とするスケーリングされた変換係数値である。
4. リスト sTCoeff は次によって修正される。

$$\text{sTCoeff}[cO+64*c8x8BlkIdx+8*i+j]=d_{ij}, i,j=0..7 \quad (\text{G-148})$$

ビットストリームは、任意の要素  $\text{sTCoeff}[cO+k]$ 、 $k=0..255$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

ビットストリームは、任意の要素  $\text{tCoeffLevel}[cO+k]$ 、 $k=0..255$  が  $-2^{(7+\text{bitDepth})}$  から  $2^{(7+\text{bitDepth})}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

#### G.8.5.1.1.4 Intra\_16x16 マクロブロックの変換係数の改善処理

この処理は、base\_mode\_flag が 0 に等しい、または tcoeff\_level\_prediction\_flag が 1 に等しいときのみ起動される。

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- ビット深度を規定する変数 bitDepth
- 量子化パラメータ値を規定する変数 qP
- スケーリングされた変換係数値のリスト sTCoeff および変換係数値のリスト tCoeffLevel の中で最初の係数インデックスを規定する変数 cO
- Intra16x16DCLevel、CbIntra16x16DCLevel、または CrIntra16x16DCLevel を表現する変数 coeffDCLevel
- Intra16x16ACLevel、CbIntra16x16ACLevel、または CrIntra16x16ACLevel を表現する変数 coeffACLevel
- LumaLevel、CbLevel、または CrLevel を表現する変数 coeffLevel
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff
- $(256+2*\text{MbWidthC}*\text{MbHeightC})$ 個の要素をもつ、変換係数レベル値のリスト tCoeffLevel

この処理の出力は、次の通りである。

- リスト sTCoeff の修正版
- リスト tCoeffLevel の修正版

記1 tcoeff\_level\_prediction\_flag が 0 に等しい時、この小小小小節は 0 に等しい refinementFlag をもつ G.8.5.1 小小節の起動の一部として常に起動される。その場合、リスト tCoeffLevel の全ての要素がこの小小小小節を起動する前に 0 に等しく設定される。

全ての残差 4×4 ブロックの DC 変換係数に対して、次の順序付けられたステップが規定される。

1. base\_mode\_flag に依存して、要素  $c_{ij}$  をもつ 4×4 配列 c が次のように導出される。

- `base_mode_flag` が 0 に等しい時、8.5.6 小節で規定される変換係数の逆走査処理は `coeffDCLevel` を入力として起動され、その出力は要素  $c_{ij}$  をもつ  $4 \times 4$  配列  $c$  とする全ての残差  $4 \times 4$  ブロックの DC 変換係数レベル値である。8.5.6 小節で規定されるこの起動に対して、現マクロブロックは `fieldMbFlag` が 1 に等しい時フィールドマクロブロックとして扱われ、現マクロブロックは `fieldMbFlag` が 0 に等しい時フレームマクロブロックとして扱われる。
- それ以外 (`base_mode_flag` が 1 に等しい)、DC 変換係数レベル値を含む要素  $c_{ij}$  をもつ  $4 \times 4$  配列  $c$  は次によって修正される。

$$c_{ij} = \text{coeffLevel}[8*(i/2)+4*(j/2)+2*(i\%2)+(j\%2)][0], i,j=0..3 \quad (\text{G-149})$$

2. リスト `tCoeffLevel` および  $4 \times 4$  配列  $c$  は次によって修正される。

$$\text{tCoeffLevel}[cO+128*(i/2)+64*(j/2)+32*(i\%2)+16*(j\%2)] += c_{ij}, i,j=0..3 \quad (\text{G-150})$$

$$c_{ij} = \text{tCoeffLevel}[cO+128*(i/2)+64*(j/2)+32*(i\%2)+16*(j\%2)], i,j=0..3 \quad (\text{G-151})$$

3. 8.5.9 小節で規定される `Intra_16x16` マクロブロックタイプに対する DC 変換係数の走査および変換処理は `bitDepth` および `qP`、 $c$  を入力として起動され、その出力は全ての残差  $4 \times 4$  ブロックに対するスケーリングされた DC 変換係数値を表現する要素  $d_{ij}$  をもつ  $4 \times 4$  配列  $d$  である。

4. リスト `sTCoeff` は次によって修正される。

$$\text{sTCoeff}[cO+128*(i/2)+64*(j/2)+32*(i\%2)+16*(j\%2)] = d_{ij}, i,j=0..3 \quad (\text{G-152})$$

`c4x4BlkIdx=0..15` で索引される残差  $4 \times 4$  ブロックそれぞれに対して、次の順序付けられたステップが規定される。

1. `base_mode_flag` に依存して、16 エントリのリストである変数 `c4x4List` が次のように導出される。

- `base_mode_flag` が 0 に等しい時、次が適用される。

$$c4x4List[k] = (k=0 ? 0 : \text{coeffACLevel}[c4x4BlkIdx][k-1]), k=0..15 \quad (\text{G-153})$$

- それ以外 (`base_mode_flag` が 1 に等しい)、次が適用される。

$$c4x4List[k] = (k=0 ? 0 : \text{coeffLevel}[c4x4BlkIdx][k]), k=0..15 \quad (\text{G-154})$$

2. 8.5.6 小節で規定される変換係数の逆走査処理は `c4x4List` を入力として起動され、その出力は要素  $e_{ij}$  をもつ  $4 \times 4$  配列  $e$  とする変換係数レベル値である。8.5.6 小節で規定されるこの起動に対して、現マクロブロックは `fieldMbFlag` が 1 に等しい時フィールドマクロブロックとして扱われ、現マクロブロックは `fieldMbFlag` が 0 に等しい時フレームマクロブロックとして扱われる。

3. リスト `tCoeffLevel` および  $4 \times 4$  配列  $e$  は次によって修正される。

$$\text{tCoeffLevel}[cO+16*c4x4BlkIdx+4*i+j] += e_{ij}, i,j=0..3, i+j>0 \quad (\text{G-155})$$

$$e_{ij} = \text{tCoeffLevel}[cO+16*c4x4BlkIdx+4*i+j], i,j=0..3, i+j>0 \quad (\text{G-156})$$

4. 8.5.11.1 小小節で規定される残差  $4 \times 4$  ブロックの走査処理は `bitDepth` および `qP`、 $4 \times 4$  配列  $e$  を入力として起動され、その出力は要素  $d_{ij}$  をもつ  $4 \times 4$  配列  $d$  とするスケーリングされた変換係数値である。

8.5.11.1 小小節で規定されるこの起動に対して、配列  $e$  は Intra\_16x16 予測モードを用いて符号化された輝度残差ブロックに関係するとして取り扱われる。

5. リスト  $sTCoeff$  は次によって修正される。

$$sTCoeff[cO+16*c4x4BlkIdx+4*i+j]=d_{ij}, i,j=0..3, i+j>0 \quad (G-157)$$

記2 インデックス  $c4x4BlkIdx$  をもつ残差  $4 \times 4$  ブロックに対する処理の間、要素  $tCoeffLevel[cO+16*c4x4BlkIdx]$  および  $sTCoeff[cO+16*c4x4BlkIdx]$  は修正されない。

ビットストリームは、任意の要素  $tCoeffLevel[cO+16*b+k]$ 、 $b=0..15$ 、 $k=1..15$  が  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記3 要素  $tCoeffLevel[cO+16*b]$ 、 $b=0..15$  は  $-2^{(7+bitDepth)}$  から  $2^{(7+bitDepth)}-1$  でそれぞれの値を含む整数値範囲を超えることができる。

### G.8.5.1.2 色差変換係数の改善処理

この処理の入力は、次の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数  $fieldMbFlag$
- 変換タイプを規定する変数  $cTrafo$
- $(256+2*MbWidthC*MbHeightC)$  の要素をもつ、スケーリングされた変換係数値のリスト  $sTCoeff$
- $(256+2*MbWidthC*MbHeightC)$  の要素をもつ、変換係数レベル値のリスト  $tCoeffLevel$

この処理の出力は、次の通りである。

- リスト  $sTCoeff$  の修正版
- リスト  $tCoeffLevel$  の修正版

$iCbCr=0..1$  で索引される両方の色成分に対して、次が適用される。

- もし  $ChromaArrayType$  が 1 または 2 に等しいならば、次が適用される。
  - $cTrafo$  が  $T\_PCM$  に等しいならば、G.8.5.1.2.1 小小小節で規定される  $I\_PCM$  マクロブロックに対する色差変換係数値の割り当て処理は  $iCbCr$ 、 $sTCoeff$ 、および  $tCoeffLevel$  を入力として起動され、その出力は  $sTCoeff$  および  $tCoeffLevel$  の修正版である。
  - それ以外 ( $cTrafo$  が  $T\_PCM$  に等しくない)、G.8.5.1.2.2 小小小節で規定される 1 または 2 に等しい  $ChromaArrayType$  をもつ色差変換係数値の改善処理は  $iCbCr$ 、 $fieldMbFlag$ 、 $sTCoeff$ 、および  $tCoeffLevel$  を入力として起動され、その出力はリスト  $sTCoeff$  および  $tCoeffLevel$  の修正版である。
- それ以外 ( $ChromaArrayType$  が 3 に等しい)、G.8.5.1.1 小小小節で規定される輝度変換係数または 3 に等しい  $ChromaArrayType$  をもつ色差変換係数の改善処理は  $(1+iCbCr)$  に等しく設定される  $iYCbCr$ 、 $fieldMbFlag$ 、 $cTrafo$ 、 $sTCoeff$ 、および  $tCoeffLevel$  を入力として起動され、その出力はリスト  $sTCoeff$  および  $tCoeffLevel$  の修正版である。

#### G.8.5.1.2.1 I\_PCMマクロブロックの色差変換係数の割り当て処理

この処理の入力は、次の通りである。

- 色成分を規定する変数 iCbCr
- $(256+2*MbWidthC*MbHeightC)$ の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff
- $(256+2*MbWidthC*MbHeightC)$ の要素をもつ、変換係数レベル値のリスト tCoeffLevel

この処理の出力は、次の通りである。

- リスト sTCoeff の修正版
- リスト tCoeffLevel の修正版

変数 numC は $(MbWidthC*MbHeightC)$ に等しく設定され、変数 cCO は $(iCbCr*numC)$ に等しく設定される。

base\_mode\_flag が 0 に等しい時、リスト tCoeffLevel および sTCoeff は次によって修正される。

$$tCoeffLevel[256+cCO+k]=0, k=0..(numC-1) \quad (G-158)$$

$$sTCoeff[256+cCO+k]=pcm\_sample\_chroma[cCO+k], k=0..(numC-1) \quad (G-159)$$

#### G.8.5.1.2.2 1 または 2 に等しいChromaArrayTypeをもつ色差変換係数の改善処理

この処理は ChromaArrayType が 1 または 2 に等しい時のみ起動される。

この処理の入力は、次の通りである。

- 色差成分を規定する変数 iCbCr
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- $(256+2*MbWidthC*MbHeightC)$ の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff
- $(256+2*MbWidthC*MbHeightC)$ の要素をもつ、変換係数レベル値のリスト tCoeffLevel

この処理の出力は、次の通りである。

- リスト sTCoeff の修正版
- リスト tCoeffLevel の修正版

変数 nW、nH、numB、cO、および qP は次によって導出される。

$$nW=MbWidthC/4 \quad (G-160)$$

$$nH=MbHeightC/4 \quad (G-161)$$

$$numB=nW*nH \quad (G-162)$$

$$cO=256+(iCbCr*MbWidthC*MbHeightC) \quad (G-163)$$

$$qP=(iCbCr==0 ? QP'_{Cb} : QP'_{Cr}) \quad (G-164)$$

tcoeff\_level\_prediction\_flag に依存して、次が適用される。

- もし tcoeff\_level\_prediction\_flag が 1 に等しいならば、リスト sTCoeff は次によって修正される。

$$sTCoeff[cO+k]=0, k=0..(MbWidthC*MbHeightC-1) \quad (G-165)$$

- それ以外 (tcoeff\_level\_prediction\_flag が 0 に等しい)、リスト tCoeffLevel は次によって修正される。

$$tCoeffLevel[cO+k]=0, k=0..(MbWidthC*MbHeightC-1) \quad (G-166)$$

全ての残差 4×4 色差ブロックの色差 DC 変換係数に対して、次の順序付けられたステップが規定される。

1. ChromaArrayType に依存して、要素  $c_{ij}$  をもつ  $(nW) \times (nH)$  配列  $c$  は次のように導出される。

- もし ChromaArrayType が 1 に等しいならば、

$$c_{ij} = \text{ChromaDCLevel}[iCbCr][2*i+j], i=0..(nH-1), j=0..(nW-1) \quad (G-167)$$

- それ以外 (ChromaArrayType が 2 に等しい)、

$$c_{ij} = \text{ChromaDCLevel}[iCbCr][\text{scan422ChromaDC}[2*i+j]],$$

$$i=0..(nH-1), j=0..(nW-1), \text{scan422ChromaDC} = \{0,2,1,5,3,6,4,7\} \quad (G-168)$$

2. リスト tCoeffLevel および  $(nW) \times (nH)$  配列  $c$  は次によって修正される。

$$tCoeffLevel[cO+32*i+16*j] += c_{ij}, i=0..(nH-1), j=0..(nW-1) \quad (G-169)$$

$$c_{ij} = tCoeffLevel[cO+32*i+16*j], i=0..(nH-1), j=0..(nW-1) \quad (G-170)$$

3. 変数  $qP_{DC}$  は次によって導出される。

$$qP_{DC} = (\text{ChromaArrayType} == 1 ? qP : qP+3) \quad (G-171)$$

4. 全ての残差 4×4 色差ブロックのスケールされた色差 DC 変換係数値を表現する要素  $d_{ij}$  をもつ  $(nW) \times (nH)$  配列  $d$  は次によって導出される。

$$d_{ij} = c_{ij} * (\text{LevelScale}(qP_{DC} \% 6, 0, 0) << (qP_{DC} / 6)), i=0..(nH-1), j=0..(nW-1) \quad (G-172)$$

5. リスト sTCoeff は次によって修正される。

$$sTCoeff[cO+32*i+16*j] += d_{ij}, i=0..(nH-1), j=0..(nW-1) \quad (G-173)$$

$c4x4BlkIdx=0..numB-1$  で索引される残差 4×4 色差ブロックそれぞれに対して、次の順序付けられたステップが規定される。

1. 16 エントリのリストである、変数  $c4x4List$  は次によって導出される。

$$c4x4List[k] = (k == 0 ? 0 : \text{ChromaACLevel}[iCbCr][c4x4BlkIdx][k-1]), k=0..15 \quad (G-174)$$

2. 8.5.6 小節で規定される変換係数の逆走査処理は  $c4x4List$  を入力として起動され、その出力は要素  $e_{ij}$  をもつ 4×4 配列  $e$  とする色差変換係数レベル値である。8.5.6 小節の処理の間、現マクロブロックは fieldMbFlag が 1 に等しい時フィールドマクロブロックとして扱われ、現マクロブロックは

fieldMbFlag が 0 に等しい時フレームマクロブロックとして扱われる。

3. リスト tCoeffLevel および 4×4 配列 e は次によって修正される。

$$tCoeffLevel[cO+16*c4x4BlkIdx+4*i+j]=e_{ij}, i,j=0..3, i+j>0 \quad (G-175)$$

$$e_{ij}=tCoeffLevel[cO+16*c4x4BlkIdx+4*i+j], i,j=0..3, i+j>0 \quad (G-176)$$

4. 8.5.11.1 小小節で規定される残差 4×4 ブロックのスケーリング処理は BitDepth<sub>C</sub> および qP、4×4 配列 e を入力として起動され、その出力は要素 d<sub>ij</sub> をもつ 4×4 配列 d とするスケーリングされた色差変換係数値である。8.5.11.1 小小節の処理の間、配列 e は色差残差ブロックに関係するとして取り扱われる。

5. リスト sTCoeff は次によって修正される。

$$sTCoeff[cO+16*c4x4BlkIdx+4*i+j]=d_{ij}, i,j=0..3, i+j>0 \quad (G-177)$$

記1 インデックス c4x4BlkIdx をもつ残差 4×4 色差ブロックに対する処理の間、要素 tCoeffLevel[cO+16\*c4x4BlkIdx] および sTCoeff[cO+16\*c4x4BlkIdx] は修正されない。

ビットストリームは、任意の要素 sTCoeff[cO+16\*b+k]、b=0..(numB-1)、k=1..15 が  $-2^{(7+BitDepth_C)}$  から  $2^{(7+BitDepth_C)-1}$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

ビットストリームは、任意の要素 tCoeffLevel[cO+16\*b+k]、b=0..(numB-1)、k=1..15 が  $-2^{(7+BitDepth_C)}$  から  $2^{(7+BitDepth_C)-1}$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記2 要素 tCoeffLevel[cO+16\*b] および sTCoeff[cO+16\*b]、b=0..(numB-1) は  $-2^{(7+BitDepth_C)}$  から  $2^{(7+BitDepth_C)-1}$  でそれぞれの値を含む整数値範囲を超えることができる。

#### G.8.5.2 変換係数改善前の変換係数レベルスケーリング処理

この処理の入力は、次の通りである。

- 現マクロブロックの輝度変換タイプを規定する変数 cTrafo
- 現マクロブロックに対する変換係数レベル値を規定する  $(256+2*MbWidth_C*MbHeight_C)$  個の要素をもつリスト tCoeffLevel
- 現マクロブロックの輝度量子化パラメータを規定する変数 tQP<sub>Y</sub>
- 参照レイヤ表現のマクロブロックの量子化パラメータを規定する変数 refQP<sub>Y</sub>
- ChromaArrayType が 0 に等しくない時、現マクロブロックの色差量子化パラメータを規定する 2 つの変数 tQP<sub>Cb</sub> および tQP<sub>Cr</sub>
- ChromaArrayType が 0 に等しくない時、参照レイヤ表現のマクロブロックの色差量子化パラメータを規定する 2 つの変数 refQP<sub>Cb</sub> および refQP<sub>Cr</sub>

この処理の出力は、リスト tCoeffLevel の修正版である。

付表 G-5/JT-H264 は変換係数レベルスケーリングのスケール値 cS を規定する。

付表G-5/JT-H264 - 変換係数レベルスケーリングのスケール値cS  
(ITU-T H.264)

$(\text{refQP} - \text{cQP} + 54) \% 6$	スケール値 cS
0	8
1	9
2	10
3	11
4	13
5	14

変数 iYCbCr は 0 から(ChromaArrayType==0 ? 0 : 2)のそれぞれの値を含む範囲を進行し、iYCbCr のそれぞれの値に対して、次の順序付けられたステップが規定される。

1. 変数 cO、iMax、cQP、および refQP は次によって導出される。

$$\text{cO} = (\text{iYCbCr}==0 ? 0 : 256+(\text{iYCbCr}-1)*\text{MbWidthC}*\text{MbHeightC}) \quad (\text{G-178})$$

$$\text{iMax} = (\text{iYCbCr}==0 ? 255 : \text{MbWidthC}*\text{MbHeightC}-1) \quad (\text{G-179})$$

$$\text{cQP} = (\text{iYCbCr}==0 ? \text{tQP}_Y : (\text{iYCbCr}==1 ? \text{tQP}_{Cb} : \text{tQP}_{Cr})) \quad (\text{G-180})$$

$$\text{refQP} = (\text{iYCbCr}==0 ? \text{refQP}_Y : (\text{iYCbCr}==1 ? \text{refQP}_{Cb} : \text{refQP}_{Cr})) \quad (\text{G-181})$$

2. 変数 cS は、refQP および cQP の値を用いて表 G-5 で規定されるように設定される。
3. 変数 rShift は次によって計算される。

$$\text{rShift}=(\text{refQP}-\text{cQP}+54)/6 \quad (\text{G-182})$$

4. 変換係数レベル値のリスト tCoeffLevel は次によって修正される。

$$\text{tCoeffLevel}[\text{cO}+i]=((\text{cS}*\text{tCoeffLevel}[\text{cO}+i])\ll\text{rShift})\gg 12, i=0..i\text{Max} \quad (\text{G-183})$$

次の制約に従わなければならない。

- a) cTrafo が T\_4x4 または T\_8x8 に等しい時、ビットストリームは、任意の要素 tCoeffLevel[k]、k=0..255 が  $-2^{(7+\text{BitDepth}_Y)}$  から  $2^{(7+\text{BitDepth}_Y)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。
- b) cTrafo が T\_16x16 に等しい時、ビットストリームは、任意の要素 tCoeffLevel[16\*b+k]、b=0..15、k=1..15 が  $-2^{(7+\text{BitDepth}_Y)}$  から  $2^{(7+\text{BitDepth}_Y)}-1$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記1 cTrafoがT\_16x16に等しい時、要素tCoeffLevel[16\*b]、b=0..15は $-2^{(7+\text{BitDepth}_Y)}$ から $2^{(7+\text{BitDepth}_Y)}-1$ でそれぞれの値を含む整数値範囲を超えることができる。

- c) ChromaArrayType が 1 または 2 に等しい、かつ cTrafo が T\_PCM に等しくない時、ビットストリームは、任意の要素 tCoeffLevel[256+16\*b+k]、b=0..(MbWidthC\*MbHeightC/8-1)、k=1..15 が  $-2^{(7+BitDepthC)}$  から  $2^{(7+BitDepthC)-1}$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記2 ChromaArrayType が 1 または 2 に等しい、かつ cTrafo が T\_PCM に等しくない時、要素 tCoeffLevel[256+16\*b]、b=0..(MbWidthC\*MbHeightC/8-1) は  $-2^{(7+BitDepthC)}$  から  $2^{(7+BitDepthC)-1}$  でそれぞれの値を含む整数値範囲を超えることができる。

- d) ChromaArrayType が 3 に等しい、かつ cTrafo が T\_4x4 または T\_8x8 に等しい時、ビットストリームは、任意の要素 tCoeffLevel[256+k]、k=0..511 が  $-2^{(7+BitDepthC)}$  から  $2^{(7+BitDepthC)-1}$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。
- e) ChromaArrayType が 3 に等しい、かつ cTrafo が T\_16x16 に等しい時、ビットストリームは、任意の要素 tCoeffLevel[256+16\*b+k]、b=0..31、k=1..15 が  $-2^{(7+BitDepthC)}$  から  $2^{(7+BitDepthC)-1}$  でそれぞれの値を含む整数値範囲を超える結果となるデータを含んではならない。

記3 ChromaArrayType が 3 に等しく、かつ cTrafo が T\_16x16 に等しい時、要素 tCoeffLevel[256+16\*b]、b=0..(MbWidthC\*MbHeightC/8-1) は  $-2^{(7+BitDepthC)}$  から  $2^{(7+BitDepthC)-1}$  でそれぞれの値を含む整数値範囲を超えることができる。

記4 tQP<sub>Y</sub> が 10 より小さく、かつ cTrafo が T\_16x16 に等しい時、0 に等しい entropy\_coding\_mode\_flag および 66、77、または 88 に等しい profile\_idc をもつビットストリームの別の表現によって表されることができる値の範囲は、任意の可能な原ピクチャの内容の近い近似を形成するのに必要でありうる、要素 tCoeffLevel[16\*b]、b=0..15 の値の全範囲を表すには十分ではないかもしれない。

記5 ChromaArrayType が 1 または 2 に等しく、かつ Cb および Cr のいずれかに置換される CX をもつ tQP<sub>CX</sub> が 4 より小さい時、0 に等しい entropy\_coding\_mode\_flag および 66、77、または 88 に等しい profile\_idc をもつビットストリームの別の表現によって表されることができる値の範囲は、任意の可能な原ピクチャの内容の近い近似を形成するのに必要でありうる、要素 tCoeffLevel[256+16\*b]、b=0..(MbWidthC\*MbHeightC/8-1) の値の全範囲を表すには十分ではないかもしれない。

### G.8.5.3 残差構築および累算処理

この処理の入力は、次の通りである。

- 現マクロブロックの構築された残差サンプル値が、そのマクロブロックに対する存在する残差サンプル値と組み合わせられるかを規定する変数 accumulationFlag
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 変換タイプを規定する変数 cTrafo
- (256+2\*MbWidthC\*MbHeightC)個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff
- 現レイヤ表現の残差輝度サンプル値を含む (PicWidthInSamples<sub>L</sub>) × (PicHeightInSamples<sub>L</sub>) 配列 picRes<sub>L</sub>
- ChromaArrayType が 0 に等しくない時、現レイヤ表現の残差色差サンプル値を含む 2 つの (PicWidthInSamples<sub>C</sub>) × (PicHeightInSamples<sub>C</sub>) 配列 picRes<sub>Cb</sub> および picRes<sub>Cr</sub>

この処理の出力は、次の通りである。

- 配列 picRes<sub>L</sub> の修正版

- ChromaArrayType が 0 に等しくない時、配列 picRes<sub>Cb</sub> および picRes<sub>Cr</sub> の修正版

G.8.5.3.1 小小小節で規定される輝度残差の構築処理が cTrafo および sTCoeff を入力として起動され、その出力は 16×16 配列 mbRes<sub>L</sub> とする残差輝度サンプル値である。

ChromaArrayType が 0 に等しくない時、G.8.5.3.2 小小小節で規定される色差残差の構築処理が cTrafo および sTCoeff を入力として起動され、その出力は 2 つの (MbWidthC)×(MbHeightC) 配列 mbRes<sub>Cb</sub> および mbRes<sub>Cr</sub> とする残差色差サンプル値である。

accumulationFlag が 1 に等しい時、次の順序付けられたステップが規定される。

1. G.8.5.4.2 小小小節で規定されるマクロブロックサンプル配列抽出処理が、fieldMbFlag、picRes<sub>L</sub>、および、ChromaArrayType が 0 に等しくない時、picRes<sub>Cb</sub> および picRes<sub>Cr</sub> を入力として起動され、その出力は 16×16 配列 refLayerMbRes<sub>L</sub> および、ChromaArrayType が 0 に等しくない時、2 つの (MbWidthC)×(MbHeightC) 配列 refLayerMbRes<sub>Cb</sub> および refLayerMbRes<sub>Cr</sub> である。
2. 16×16 配列 mbRes<sub>L</sub>、x,y=0..15 の全ての要素 mbRes<sub>L</sub>[x,y] は次によって修正される。

$$\text{mbRes}_L[x,y]=\text{Clip3}(y\text{Min},y\text{Max},\text{mbRes}_L[x,y]+\text{refLayerMbRes}_L[x,y]) \quad (\text{G-184})$$

ここで、

$$y\text{Min}=(1\ll\text{BitDepth}_Y)+1 \quad (\text{G-185})$$

$$y\text{Max}=(1\ll\text{BitDepth}_Y)-1 \quad (\text{G-186})$$

3. ChromaArrayType が 0 に等しくない時、Cb および Cr のいずれかに置換される CX に対して、(MbWidthC)×(MbHeightC) 配列 mbRes<sub>CX</sub>、x=0..(MbWidthC-1)、y=0..(MbHeightC-1) の全ての要素 mbRes<sub>CX</sub>[x,y] は次によって修正される。

$$\text{mbRes}_{CX}[x,y]=\text{Clip3}(c\text{Min},c\text{Max},\text{mbRes}_{CX}[x,y]+\text{refLayerMbRes}_{CX}[x,y]) \quad (\text{G-187})$$

ここで、

$$c\text{Min}=(1\ll\text{BitDepth}_C)+1 \quad (\text{G-188})$$

$$c\text{Max}=(1\ll\text{BitDepth}_C)-1 \quad (\text{G-189})$$

G.8.5.4.1 小小小節で規定されるピクチャサンプル配列構築処理が fieldMbFlag、mbRes<sub>L</sub>、picRes<sub>L</sub>、および、ChromaArrayType が 0 に等しくない時、mbRes<sub>Cb</sub>、mbRes<sub>Cr</sub>、picRes<sub>Cb</sub>、および picRes<sub>Cr</sub> を入力として起動され、その出力は配列 picRes<sub>L</sub> の修正版および、ChromaArrayType が 0 に等しくない時、配列 picRes<sub>Cb</sub> および picRes<sub>Cr</sub> の修正版である。

### G.8.5.3.1 輝度残差または 3 に等しい ChromaArrayType の色差残差の構築処理

この処理の入力は、次の通りである。

- (存在する時) 色成分を規定する変数 iYCbCr
- 変換タイプを規定する変数 cTrafo

- (256+2\*MbWidthC\*MbHeightC)個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff

この処理の出力は、要素 mbRes[x,y]をもつ 16×16 配列 mbRes とする残差サンプル値である。

この小小小節の入力として iYCbCr が存在しない時、それは 0 に等しいと推定される。

iYCbCr に依存して、変数 bitDepth および cO は次のように導出される。

- もし iYCbCr が 0 に等しいならば、bitDepth は BitDepth<sub>Y</sub> に等しく設定され、cO は 0 に等しく設定される。
- それ以外、もし iYCbCr が 1 に等しいならば、bitDepth は BitDepth<sub>C</sub> に等しく設定され、cO は 256 に等しく設定される。
- それ以外 (iYCbCr が 2 に等しい)、bitDepth は BitDepth<sub>C</sub> に等しく設定され、cO は (256+MbWidthC\*MbHeightC)に等しく設定される。

cTrafo に依存して、16×16 配列 mbRes は次のように導出される。

- もし cTrafo が T\_PCM に等しいならば、G.8.5.3.1.1 小小小節で規定される I\_PCM マクロブロックの輝度残差あるいは 3 に等しい ChromaArrayType の色差残差の構築処理が cO および sTCoeff を入力として起動され、その出力は残差サンプル値の 16×16 配列 mbRes である。
- それ以外、もし cTrafo が T\_4x4 に等しいならば、G.8.5.3.1.2 小小小節で規定される残差 4×4 ブロックの構築処理が bitDepth および cO、sTCoeff を入力として起動され、その出力は残差サンプル値の 16×16 配列 mbRes である。
- それ以外、もし cTrafo が T\_8x8 に等しいならば、G.8.5.3.1.3 小小小節で規定される残差 8×8 ブロックの構築処理が bitDepth および cO、sTCoeff を入力として起動され、その出力は残差サンプル値の 16×16 配列 mbRes である。
- それ以外 (cTrafo が T\_16x16 に等しい)、G.8.5.3.1.4 小小小節で規定される Intra\_16x16 マクロブロックの残差の構築処理が bitDepth および cO、sTCoeff を入力として起動され、その出力は残差サンプル値の 16×16 配列 mbRes である。

#### G.8.5.3.1.1 I\_PCMマクロブロックの輝度残差または 3 に等しいChromaArrayTypeの色差残差の構築処理

この処理の入力は、次の通りである。

- スケーリングされた変換係数値のリスト sTCoeff の中で最初の係数のインデックスを規定する変数 cO
- (256+2\*MbWidthC\*MbHeightC)の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff

この処理の出力は、要素 mbRes[x,y]をもつ 16×16 配列 mbRes とする残差サンプル値である。

16×16 配列 mbRes は次によって導出される。

$$\text{mbRes}[x,y]=\text{sTCoeff}[cO+y*16+x], \quad x,y=0..15 \quad (\text{G-190})$$

#### G.8.5.3.1.2 残差 4×4 ブロックの構築処理

この処理の入力は、次の通りである。

- ビット深度を規定する変数 bitDepth

- スケーリングされた変換係数値のリスト sTCoeff の中で最初の係数のインデックスを規定する変数 cO
- (256+2\*MbWidthC\*MbHeightC)個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff

この処理の出力は、要素 mbRes[x,y]をもつ 16×16 配列 mbRes とする残差サンプル値である。

c4x4BlkIdx=0..15 で索引される残差 4×4 ブロックそれぞれに対して、次の順序付けられたステップが規定される。

1. 要素 d<sub>ij</sub> をもつ 4×4 配列 d は次によって導出される。

$$d_{ij}=sTCoeff[cO+16*c4x4BlkIdx+4*i+j], i,j=0..3 \quad (G-191)$$

2. 8.5.11.2 小節で規定される残差 4×4 ブロックの変換処理は bitDepth および 4×4 配列 d を入力として起動され、その出力は要素 r<sub>ij</sub> をもつ 4×4 配列 r とする残差サンプル値である。
3. 6.4.3 小節で規定される逆 4×4 輝度ブロック走査処理は c4x4BlkIdx を入力として起動され、その出力は(xP,yP)に割り当てられる。
4. x=xP..(xP+3)およびy=yP..(yP+3)をもつ 16×16 配列 mbRes の要素 mbRes[x,y]は次によって導出される。

$$mbRes[xP+j,yP+i]=r_{ij}, i,j=0..3 \quad (G-192)$$

#### G.8.5.3.1.3 残差 8×8 ブロックの構築処理

この処理の入力は、次の通りである。

- ビット深度を規定する変数 bitDepth
- スケーリングされた変換係数値のリスト sTCoeff の中で最初の係数のインデックスを規定する変数 cO
- (256+2\*MbWidthC\*MbHeightC)個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff

この処理の出力は、要素 mbRes[x,y]をもつ 16×16 配列 mbRes とする残差サンプル値である。

c8x8BlkIdx=0..3 で索引される残差 8×8 ブロックのそれぞれに対して、次の順序付けられたステップが規定される。

1. 要素 d<sub>ij</sub> をもつ 8×8 配列 d は次によって導出される。

$$d_{ij}=sTCoeff[cO+64*c8x8BlkIdx+8*i+j], i,j=0..7 \quad (G-193)$$

2. 8.5.12.2 小節で規定される残差 8×8 ブロックの変換処理は bitDepth および 8×8 配列 d を入力として起動され、その出力は要素 r<sub>ij</sub> をもつ 8×8 配列 r とする残差サンプル値である。
3. 6.4.5 小節で規定される逆 8×8 輝度ブロック走査処理は c8x8BlkIdx を入力として起動され、その出力は(xP,yP)に割り当てられる。
4. x=xP..(xP+7)およびy=yP..(yP+7)をもつ 16×16 配列 mbRes の要素 mbRes[x,y]は次によって導出される。

$$mbRes[xP+j,yP+i]=r_{ij}, i,j=0..7 \quad (G-194)$$

#### G.8.5.3.1.4 Intra\_16x16 マクロブロックの残差の構築処理

この処理の入力は、次の通りである。

- ビット深度を規定する変数 bitDepth
- スケーリングされた変換係数値のリスト sTCoeff の中で最初の係数のインデックスを規定する変数 cO
- $(256+2*MbWidthC*MbHeightC)$ 個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff

この処理の出力は、要素 mbRes[x,y]をもつ 16×16 配列 mbRes とする残差サンプル値である。

c4x4BlkIdx=0..15 で索引される残差 4×4 ブロックのそれぞれに対して、次の順序付けられたステップが規定される。

1. 要素  $d_{ij}$  をもつ 4×4 配列 d は次によって導出される。

$$d_{ij}=sTCoeff[cO+16*c4x4BlkIdx+4*i+j], i,j=0..3 \quad (G-195)$$

2. 8.5.11.2 小節で規定される残差 4×4 ブロックの変換処理は bitDepth および 4×4 配列 d を入力として起動され、その出力は要素  $r_{ij}$  をもつ 4×4 配列 r とする残差サンプル値である。
3. 6.4.3 小節で規定される逆 4×4 輝度ブロック走査処理は c4x4BlkIdx を入力として起動され、その出力は(xP,yP)に割り当てられる。
4.  $x=xP..(xP+3)$ および $y=yP..(yP+3)$ をもつ 16×16 配列 mbRes の要素 mbRes[x,y]は次によって導出される。

$$mbRes[xP+j,yP+i]=r_{ij}, i,j=0..3 \quad (G-196)$$

#### G.8.5.3.2 色差残差の構築処理

この処理の入力は、次の通りである。

- 変換タイプを規定する変数 cTrafo
- $(256+2*MbWidthC*MbHeightC)$ 個の要素をもつ、スケーリングされた変換係数値のリスト sTCoeff

この処理の出力は、要素 mbRes<sub>Cb</sub>[x,y]および mbRes<sub>Cr</sub>[x,y]をそれぞれもつ、2つの(MbWidthC)×(MbHeightC)配列 mbRes<sub>Cb</sub> および mbRes<sub>Cr</sub> とする残差色差サンプル値である。

iCbCr=0..1 で索引される両方の色差成分に対して、および 0 に等しい iCbCr に対して Cb および 1 に等しい iCbCr に対して Cr に置換される CX に対して、次が適用される。

- もし ChromaArrayType が 1 または 2 に等しいならば、次が適用される。
  - もし cTrafo が T\_PCM に等しいならば、G.8.5.3.2.1 小節で規定される I\_PCM マクロブロックの色差残差の構築処理が iCbCr および sTCoeff を入力として起動され、その出力は残差色差サンプル値の(MbWidthC)×(MbHeightC)配列 mbRes<sub>CX</sub> である。
  - それ以外 (cTrafo が T\_PCM に等しくない)、G.8.5.3.2.2 小節で規定される 1 または 2 に等しい ChromaArrayType をもつ色差残差の構築処理が iCbCr および sTCoeff を入力として起動され、その出力は残差色差サンプル値の(MbWidthC)×(MbHeightC)配列 mbRes<sub>CX</sub> である。
- それ以外 (ChromaArrayType が 3 に等しい)、G.8.5.3.1 小節で規定される輝度残差または 3 に等しい

ChromaArrayType をもつ色差残差の構築処理は(1+iCbCr)に等しく設定される iYCbCr、cTrafo、および sTCoeff を入力として起動され、その出力は残差色差サンプル値の(MbWidthC)×(MbHeightC)配列 mbRes<sub>CX</sub> である。

#### G.8.5.3.2.1 I\_PCMマクロブロックの色差残差に対する構築処理

本処理への入力は以下の通りである。

- 色差成分を規定する変数 iCbCr
- (256+2\*MbWidthC\*MbHeightC)個の要素を持つスケーリングされた変換係数値 sTCoeff のリスト

本処理に対する出力は、要素mbRes[x,y]を持つ(MbWidthC)×(MbHeightC)配列mbResの残差色差サンプル値である。

(MbWidthC)×(MbHeightC)配列mbResは、次式により導出される。

$$\begin{aligned} \text{mbRes}[x,y] &= \text{sTCoeff}[256+i\text{CbCr}*\text{MbWidthC}*\text{MbHeightC}+y*\text{MbWidthC}+x] \\ x &= 0..\text{(MbWidthC-1)} \text{ および } y = 0..\text{(MbHeightC-1)} \end{aligned} \quad (\text{G-197})$$

#### G.8.5.3.2.2 1 または 2 に等しいChromaArrayTypeの色差残差に対する構築処理

本処理は ChromaArrayType が 1 または 2 に等しい場合にのみ起動される。

本処理に対する入力は以下の通りである。

- 色差成分を規定する変数 iCbCr
- (256+2\*MbWidthC\*MbHeightC)個の要素を持つスケーリングされた変換係数値 sTCoeff のリスト

本処理に対する出力は、mbRes[x,y]要素を持つ(MbWidthC)×(MbHeightC)配列mbResの残差色差サンプル値である。

nW、nH、numB、および cO の各値は、次式より導出される。

$$nW = \text{MbWidthC}/4 \quad (\text{G-198})$$

$$nH = \text{MbHeightC}/4 \quad (\text{G-199})$$

$$\text{numB} = nW*nH \quad (\text{G-200})$$

$$cO = 256+(i\text{CbCr}*\text{MbWidthC}*\text{MbHeightC}) \quad (\text{G-201})$$

全ての残差 4×4 色差ブロックの色差 DC 変換係数に対して、次の順序付けられたステップが規定される。

1. 要素 $c_{ij}$ を持つ(nW)×(nH)配列cは、次式より導出される。

$$c_{ij} = \text{sTCoeff}[cO+32*i+16*j] \quad i = 0..\text{(nH-1)}, j = 0..\text{(nW-1)} \quad (\text{G-202})$$

2. 8.5.10.1 小節で規定される色差DC変換係数に対する変換処理が、BitDepth<sub>c</sub>、および(nW)×(nH)配列 c を入力として起動され、出力は要素 $f_{ij}$ を持つ(nW)×(nH)配列fの全ての残差4×4色差ブロックに対するDC値である。

3. ChromaArrayType に依存して、要素dcC<sub>ij</sub>を持つ(nW)×(nH)配列dcCは、次の通り導出される。

- ChromaArrayType が 1 に等しいならば、

$$dcC_{ij}=f_{ij}>>5 \quad i = 0..(nH - 1), j = 0..(nW - 1) \quad (G-203)$$

- それ以外(ChromaArrayType が 2 に等しい)

$$dcC_{ij}=(f_{ij}+(1<<5))>>6 \quad i = 0..(nH - 1), j = 0..(nW - 1) \quad (G-204)$$

ビットストリームは、dcC の任意の要素  $dcC_{ij}$ 、 $i=0..(nH-1)$ 、 $j=0..(nW-1)$ が、整数値 $-2^{(7+BitDepth_c)}$ から $2^{(7+BitDepth_c)-1}$ のそれぞれの値を含む範囲を超える結果となるデータを含んではならない。

記 dependency\_idが0に等しく、かつquality\_idが0に等しいレイヤ表現に対して、G.8.5.1.2小小節(G.8.5.1小小節の起動の一部として)と本小小節の連続した起動は、配列dcCを生じ、それはG.8.5.10小小節の起動により得られる配列dcCと同一である。しかし、本小小節で導出される中間値 $c_{ij}$ および $f_{ij}$ 、 $i=0..(nH-1)$ 、 $j=0..(nW-1)$ は、整数値 $-2^{(7+BitDepth_c)}$ から $2^{(7+BitDepth_c)-1}$ のそれぞれの値を含む範囲を超えることができる。

$c4x4BlkIdx=0..(numB-1)$ により索引される各残差 $4 \times 4$ 色差ブロックに対して、次の順序付けられたステップが規定される。

1. 要素 $d_{ij}$ を持つ $4 \times 4$ 配列 $d$ は、以下で導出される。

- 要素  $d_{00}$  は次式により導出される。

$$d_{00}=dcY_{kl}, k=c4x4BlkIdx/2, l=c4x4BlkIdx\%2 \quad (G-205)$$

- 要素  $d_{ij}$ 、 $i,j=0..3$  かつ  $i+j>0$  は次式により導出される。

$$d_{ij}=sTCoeff[cO+16*c4x4BlkIdx+4*i+j] \quad (G-206)$$

2. 8.5.11.2小小節で規定される残差 $4 \times 4$ ブロックに対する変換処理が、 $BitDepth_c$ 、および $4 \times 4$ 配列 $d$ を入力として起動され、出力は要素 $r_{ij}$ を持つ $4 \times 4$ 配列 $r$ の残差色差サンプル値である。
3. 色差の位置 $(xP,yP)$ は、次式より導出される。

$$xP=4*(c4x4BlkIdx\%2) \quad (G-207)$$

$$yP=4*(c4x4BlkIdx/2) \quad (G-208)$$

4.  $(MbWidthC) \times (MbHeightC)$ 配列 $mbRes$ 、 $x=xP..(xP+3)$ および $y=yP..(yP+3)$ 、の要素 $mbRes[x,y]$ は次式より導出される。

$$mbRes[xP+j,yP+i]=r_{ij}, i,j=0..3 \quad (G-209)$$

#### G.8.5.4 サンプル配列の累算処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 現レイヤ表現に対する残差輝度サンプル値を含む $(PicWidthInSamples_L) \times (PicHeightInSamples_L)$ 配列 picRes<sub>L</sub>
- 現レイヤ表現に対する構築輝度サンプル値を含む $(PicWidthInSamples_L) \times (PicHeightInSamples_L)$ 配列 picSamples<sub>L</sub>

- ChromaArrayType が 0 に等しくない時、現レイヤ表現に対する残差色差サンプル値を含む 2 つの  $(PicWidthInSamples_C) \times (PicHeightInSamples_C)$  配列  $picRes_{Cb}$ 、および  $picRes_{Cr}$
- ChromaArrayType が 0 に等しくない時、現レイヤ表現に対する構築色差サンプル値を含む 2 つの  $(PicWidthInSamples_C) \times (PicHeightInSamples_C)$  配列  $picSamples_{Cb}$ 、および  $picSamples_{Cr}$

本処理の出力は以下の通りである。

- 配列  $picSamples_L$  の修正版
- ChromaArrayType が 0 に等しくない時、配列  $picSamples_{Cb}$  と  $picSamples_{Cr}$  との修正版

G.8.5.4.2 小小節で規定されるマクロブロックサンプル配列の抽出処理が、 $fieldMbFlag$ 、 $picRes_L$ 、および ChromaArrayType が 0 に等しくない時には  $picRes_{Cb}$  と  $picRes_{Cr}$  を入力として起動され、出力は  $mbRes_L$  および、ChromaArrayType が 0 に等しくない時には  $mbRes_{Cb}$  と  $mbRes_{Cr}$  に割り当てられる。

G.8.5.4.2 小小節で規定されるマクロブロックサンプル配列の抽出処理が、 $fieldMbFlag$ 、 $picSamples_L$ 、および ChromaArrayType が 0 に等しくない時には  $picSamples_{Cb}$  と  $picSamples_{Cr}$  を入力として起動され、出力は  $mbPred_L$  および、ChromaArrayType が 0 に等しくない時には  $mbPred_{Cb}$  と  $mbPred_{Cr}$  に割り当てられる。

16x16 配列  $mbSamples_L$  は次式にて導出される。

$$mbSamples_L[x,y]=Clip1_Y(mbPred_L[x,y]+mbRes_L[x,y]) \quad x, y = 0..15 \quad (G-210)$$

ChromaArrayType が 0 に等しくない時には、CX は Cb と Cr に置き換えられるものとして、 $(MbWidthC) \times (MbHeightC)$  配列  $mbSamples_{CX}$  は次式にて導出される。

$$mbSamples_{CX}[x,y]=Clip1_C(mbPred_{CX}[x,y]+mbRes_{CX}[x,y]) \quad x = 0..(MbWidthC - 1) \\ \text{および } y = 0..(MbHeightC - 1) \quad (G-211)$$

G.8.5.4.1 小小節で規定されるピクチャサンプル配列構築処理が、 $fieldMbFlag$ 、 $mbSamples_L$ 、 $picSamples_L$ 、および、ChromaArrayType が 0 に等しくない時には、 $mbSamples_{Cb}$ 、 $mbSamples_{Cr}$ 、 $picSamples_{Cb}$ 、および  $picSamples_{Cr}$  を入力として起動され、出力は  $picSamples_L$  および、ChromaArrayType が 0 に等しくない時には  $picSamples_{Cb}$  および  $picSamples_{Cr}$  の修正版である。

#### G.8.5.4.1 ピクチャサンプル配列構築処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数  $fieldMbFlag$
- 現マクロブロックに対する輝度サンプル値を含む  $16 \times 16$  配列  $mbArray_L$
- 現レイヤ表現に対する輝度サンプル値を含む  $(PicWidthInSamples_L) \times (PicWidthInHeight_L)$  配列  $picArray_L$
- ChromaArrayType が 0 に等しくない時、現マクロブロックに対する色差サンプル値を含む 2 つの  $(MbWidthC) \times (MbHeightC)$  配列  $mbArray_{Cb}$  および  $mbArray_{Cr}$
- ChromaArrayType が 0 に等しくない時、現レイヤ表現に対する色差サンプル値を含む 2 つの  $(PicWidthInSamples_C) \times (PicHeightInSamples_C)$  配列  $picArray_{Cb}$  および  $picArray_{Cr}$

本処理の出力は以下の通りである。

- 配列  $\text{picArray}_L$  の修正版、
- $\text{ChromaArrayType}$  が 0 に等しくない時、配列  $\text{picArray}_{Cb}$  と  $\text{picArray}_{Cr}$  との修正版

G.8.5.4.3 小小小節で規定される色成分に対するピクチャサンプル配列の構築処理が、 $\text{fieldMbFlag}$ 、 $\text{picRes}_L$ 、16 に等しく設定された  $\text{mbW}$ 、16 に等しく設定された  $\text{mbH}$ 、 $\text{mbArray}_L$  および  $\text{picArray}_L$  を入力として起動され、出力は配列  $\text{picArray}_L$  の修正版である。

$\text{ChromaArrayType}$  が 0 に等しくない時には、Cr および Cb に置換される CX に対して、G.8.5.4.3 小小小節で規定される色成分に対するピクチャサンプル配列の構築処理が、 $\text{fieldMbFlag}$ 、 $\text{MbWidthC}$  に等しく設定された  $\text{mbW}$ 、 $\text{MbHeightC}$  に等しく設定された  $\text{mbH}$ 、 $\text{mbArray}_{CX}$  および  $\text{picArray}_{CX}$  を入力として起動され、出力は配列  $\text{picArray}_{CX}$  の修正版である。

#### G.8.5.4.2 マクロブロックサンプル配列の抽出処理

本処理への入力以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数  $\text{fieldMbFlag}$
- 現レイヤ表現に対する輝度サンプル値を含む  $(\text{PicWidthInSamples}_L) \times (\text{PicWidthInHeight}_L)$  配列  $\text{picArray}_L$
- $\text{ChromaArrayType}$  が 0 に等しくない時、現レイヤ表現に対する色差サンプル値を含む 2 つの  $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$  配列  $\text{picArray}_{Cb}$  および  $\text{picArray}_{Cr}$

本処理の出力は以下の通りである。

- 現マクロブロックに対する輝度サンプル値を含む  $16 \times 16$  配列  $\text{mbArray}_L$ 、
- $\text{ChromaArrayType}$  が 0 に等しくない時、現マクロブロックに対する色差サンプル値を含む 2 つの  $(\text{MbWidthC}) \times (\text{MbHeightC})$  配列  $\text{mbArray}_{Cb}$  および  $\text{mbArray}_{Cr}$

G.8.5.4.4 小小小節で規定される色成分に対するマクロブロックサンプル配列の抽出処理が、 $\text{fieldMbFlag}$ 、16 に等しく設定された  $\text{mbW}$ 、16 に等しく設定された  $\text{mbH}$ 、および  $\text{picArray}_L$  を入力として起動され、出力は  $\text{mbArray}_L$  に割り当てられる。

$\text{ChromaArrayType}$  が 0 に等しくない時には、Cr および Cb に置換される CX に対して、G.8.5.4.4 小小小節で規定される色成分に対するマクロブロックサンプル配列の抽出処理が、 $\text{fieldMbFlag}$ 、 $\text{MbWidthC}$  に等しく設定された  $\text{mbW}$ 、 $\text{MbHeightC}$  に等しく設定された  $\text{mbH}$ 、および  $\text{picArray}_{CX}$  を入力として起動され、出力は  $\text{mbArray}_{CX}$  に割り当てられる。

#### G.8.5.4.3 色成分に対するピクチャサンプル配列の構築処理

本処理への入力以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数  $\text{fieldMbFlag}$
- サンプル単位のマクロブロック色成分の幅を規定する変数  $\text{mbW}$

- サンプル単位のマクロブロック色成分の高さを規定する変数 `mbH`
- 現マクロブロックの色成分に対するサンプル値を含む $(mbW) \times (mbH)$ 配列 `mbArray`
- 現レイヤ表現の色成分に対するサンプル値を含む $(mbW * PicWidthInMbs) \times (mbH * PicHeightInMbs)$ 配列 `picArray`

本処理の出力は配列 `picArray` の修正版である。

6.4.1 小節で規定される逆マクロブロック走査処理が `CurrMbAddr` を入力として起動され、出力は $(xO, yO)$ に割り当てられる。6.4.1 小節の処理を通して現マクロブロックは、`fieldMbFlag` が 1 に等しい時はフィールドマクロブロックとして扱われ、`fieldMbFlag` が 0 に等しい時はフレームマクロブロックとして扱われる。

サンプル位置 $(xP, yP)$ は次式より導出される。

$$xP = (xO \gg 4) * mbW \quad (G-212)$$

$$yP = ((yO \gg 4) * mbH) + (yO \% 2) \quad (G-213)$$

変数`MbaffFrameFlag`と`fieldMbFlag`に依存して、配列`picArray`は次の通りに修正される。

- もし `MbaffFrameFlag` が 1 に等しく、かつ `fieldMbFlag` が 1 に等しいならば、

$$picArray[xP+x, yP+2*y] = mbArray[x, y] \quad x = 0..(mbW - 1), y = 0..(mbH - 1) \quad (G-214)$$

- それ以外(`MbaffFrameFlag` が 0 に等しい、または `fieldMbFlag` が 0 に等しい)、

$$picArray[xP+x, yP+y] = mbArray[x, y] \quad x = 0..(mbW - 1), y = 0..(mbH - 1) \quad (G-215)$$

#### G.8.5.4.4 色成分に対するマクロブロックサンプル配列の抽出処理

本処理への入力には以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- サンプル単位のマクロブロック色成分の幅を規定する変数 `mbW`
- サンプル単位のマクロブロック色成分の高さを規定する変数 `mbH`
- 現レイヤ表現の色成分に対するサンプル値を含む $(mbW * PicWidthInMbs) \times (mbH * PicHeightInMbs)$ 配列 `picArray`

本処理の出力は、現マクロブロックの色成分に対するサンプル値を含む $(mbW) \times (mbH)$ 配列 `mbArray` である。

6.4.1 小節で規定される逆マクロブロック走査処理が `CurrMbAddr` を入力として起動され、出力は $(xO, yO)$ に割り当てられる。6.4.1 小節の処理を通して現マクロブロックは、`fieldMbFlag` が 1 に等しい時はフィールドマクロブロックとして扱われ、`fieldMbFlag` が 0 に等しい時はフレームマクロブロックとして扱われる。

サンプル位置 $(xP, yP)$ は次式より導出される。

$$xP = (xO \gg 4) * mbW \quad (G-216)$$

$$yP = ((yO \gg 4) * mbH) + (yO \% 2) \quad (G-217)$$

変数MbaffFrameFlagとfieldMbFlagに依存して、配列mbArrayのサンプルは次式より導出される。

- もし MbaffFrameFlag が 1 に等しく、かつ fieldMbFlag が 1 に等しいならば、

$$\text{mbArray}[x,y]=\text{picArray}[xP+x,yP+2*y] \quad x = 0..(\text{mbW} - 1), y = 0..(\text{mbH} - 1) \quad (\text{G-218})$$

- それ以外(MbaffFrameFlag が 0 に等しい、または fieldMbFlag が 0 に等しい)、

$$\text{mbArray}[x,y]=\text{picArray}[xP+x,yP+y] \quad x = 0..(\text{mbW} - 1), y = 0..(\text{mbH} - 1) \quad (\text{G-219})$$

#### G.8.5.5 サンプル配列の再初期化処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 現レイヤ表現の輝度サンプル値を含む(PicWidthInSamples<sub>L</sub>)×(PicHeightInSamples<sub>L</sub>)配列 picSamples<sub>L</sub>
- ChromaArrayType が 0 に等しくない時、現レイヤ表現に対する色差サンプル値を含む 2 つの (PicWidthInSamples<sub>C</sub>)×(PicHeightInSamples<sub>C</sub>)配列 picSamples<sub>Cb</sub> と picSamples<sub>Cr</sub>

本処理の出力は以下の通りである。

- 配列 picSamples<sub>L</sub> の修正版
- ChromaArrayType が 0 に等しくない時、配列 picSamples<sub>Cb</sub> と picSamples<sub>Cr</sub> の修正版

16x16配列mbSamples<sub>L</sub>は次式より導出される。

$$\text{mbSamples}_L[x,y]=0 \quad x, y = 0..15 \quad (\text{G-220})$$

ChromaArrayType が 0 に等しくない時には、Cr および Cb に置換される CX に対して、(MbWidthC)×(MbHeightC)配列mbSamples<sub>CX</sub>は次式より導出される。

$$\text{mbSamples}_{CX}[x,y]=0 \quad x = 0..(\text{MbWidthC} - 1), y = 0..(\text{MbHeightC} - 1) \quad (\text{G-221})$$

G.8.5.4.1 小小節で規定されるピクチャサンプル配列の抽出処理が、fieldMbFlag、mbSamples<sub>L</sub>、picSamples<sub>L</sub>、およびChromaArrayTypeが0に等しくない時にはmbSamples<sub>Cb</sub>、mbSamples<sub>Cr</sub>、picSamples<sub>Cb</sub>、およびpicSamples<sub>Cr</sub>が入力として起動され、出力はpicSamples<sub>L</sub>の修正版であり、更にChromaArrayTypeが0に等しくない時にはpicSamples<sub>Cb</sub>およびpicSamples<sub>Cr</sub>の修正版である。

#### G.8.6 予測データ、INTRAサンプル、および残差サンプルに対する再サンプリング処理

G.8.6.1 小小節は、マクロブロックタイプ、サブマクロブロックタイプ、参照インデックス、および動きベクトルに対するレイヤ間予測の導出処理を規定する。

G.8.6.2 小小節は、INTRA サンプルの再サンプリング処理を規定する。

G.8.6.3 小小節は、残差サンプルの再サンプリング処理を規定する。

### G.8.6.1 マクロブロックタイプ、サブマクロブロックタイプ、参照インデックス、および動きベクトルに対するレイヤ間予測因子の導出処理

本処理は、base\_mode\_flag が 1 に等しい時、あるいは、0 と 1 に置換される X と mbPartIdx=0.3 をもつ任意の motion\_prediction\_flag\_IX[mbPartIdx] が 1 に等しい時のみ起動される。

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックで、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックのマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerMbType
- 参照レイヤ表現のマクロブロックのサブマクロブロックタイプを規定する (RefLayerPicSizeInMbs)×4 配列 refLayerSubMbType
- 参照レイヤ表現のマクロブロックの予測利用フラグを規定する 2 つの (RefLayerPicSizeInMbs)×4 配列 refLayerPredFlagL0 と refLayerPredFlagL1
- 参照レイヤ表現のマクロブロックの参照インデックスを規定する 2 つの (RefLayerPicSizeInMbs)×4 配列 refLayerRefIdxL0 と refLayerRefIdxL1
- 参照レイヤ表現のマクロブロックの動きベクトル成分を規定する 2 つの (RefLayerPicSizeInMbs)×4×4×2 配列 refLayerMvL0 と refLayerMvL1
- CroppingChangeFlag が 1 に等しく、かつ (slice\_type%5) が 2 より小さい時は、参照ピクチャリスト refPicList0
- CroppingChangeFlag が 1 に等しく、かつ (slice\_type%5) が 1 に等しい時は、参照ピクチャリスト refPicList1

本処理の出力は以下の通りである。

- 現マクロブロックのマクロブロックタイプに対する予測因子を規定する変数 mbTypeILPred
- 現マクロブロックのサブマクロブロックタイプに対する予測因子を規定する 4 つの要素を持つリスト subMbTypeILPred
- 現マクロブロックの参照インデックスに対するレイヤ間予測因子を規定する 2 つの 2×2 配列 refIdxILPredL0 と refIdxILPredL1
- 現マクロブロックの動きベクトル成分に対するレイヤ間予測因子を規定する 2 つの 4×4×2 配列 mvILPredL0 と mvILPredL1

G.8.6.1.1 小節で規定される参照レイヤパーティション識別の導出処理が、fieldMbFlag、refLayerFieldMbFlag、refLayerMbType、および refLayerSubMbType を入力として起動され、出力は変数 intraILPredFlag、および intraILPredFlag が 0 に等しい時は refLayerPartIdx[x,y] の要素を持つ 4×4 配列 refLayerPartIdx としての参照レイヤパーティション識別である。

slice\_type が EI に等しい時は、ビットストリームは intraILPredFlag が 0 に等しくなるデータを含んではな

らない。

intraILPredFlag に依存して、 $2 \times 2$  配列 refIdxILPredL0 と refIdxILPredL1、および  $4 \times 4 \times 2$  配列 mvILPredL0 と mvILPredL1 が以下の通り導出される。

- もし intraILPredFlag が 1 に等しいならば、 $2 \times 2$  配列 refIdxILPredL0 と refIdxILPredL1 の全ての要素は-1 に等しく設定され、また  $4 \times 4 \times 2$  配列 mvILPredL0 と mvILPredL1 の全ての要素は 0 に等しく設定される。
- それ以外(intraILPredFlag が 0 に等しい)、G.8.6.1.2 小小小節で規定される参照インデックスと動きベクトルのレイヤ間予測因子のための導出処理が、fieldMbFlag、refLayerFieldMbFlag、refLayerPredFlagL0、refLayerPredFlagL1、refLayerRefIdxL0、refLayerRefIdxL1、refLayerMvL0、refLayerMvL1、refLayerPartIdx、refPicList0(利用可能な時)、および refPicList1(利用可能な時)が入力として起動され、出力は配列 refIdxILPredL0、refIdxILPredL1、mvILPredL0、および mvILPredL1 である。

intraILPredFlag に依存して、変数 mbTypeILPred とリスト subMbTypeILPred が以下の通り導出される。

- もし intraILPredFlag が 1 に等しいならば、リスト subMbTypeILPred の subMbTypeILPred[mbPartIdx]、mbPartIdx=0..3、の全ての要素は無規定としてマークされ、変数 mbTypeILPred は以下の通り導出される。
  - もし tcoeff\_level\_prediction\_flag が 1 に等しいならば、mbTypeILPred は refLayerMbType[CurrMbAddr] に等しく設定される。
  - それ以外(tcoeff\_level\_prediction\_flag が 0 に等しい)、mbTypeILPred は I\_BL に等しく設定される。
- それ以外(intraILPredFlag が 0 に等しい)、G.8.6.1.3 小小小節で規定される P と B のマクロブロックとサブマクロブロックタイプに対するレイヤ間予測因子の導出処理が、refIdxILPredL0、refIdxILPredL1、mvILPredL0、および mvILPredL1 を入力として起動され、出力は変数 mbTypeILPred と subMbTypeILPred のリストである。

#### G.8.6.1.1 参照レイヤパーティション識別の導出処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックで、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックに対してマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerMbType
- 参照レイヤ表現のマクロブロックに対してサブマクロブロックタイプを規定する (RefLayerPicSizeInMbs) $\times$ 4 配列 refLayerSubMbType

本処理の出力は以下の通りである。

- 現マクロブロックがレイヤ間 INTRA 予測、あるいは tcoeff\_level\_prediction\_flag が 1 に等しい場合にはレイヤ内 INTRA 予測とレイヤ間予測の組み合わせにより予測され得るかどうかを規定する変数 intraILPredFlag

- `intraLLPredFlag` が 0 に等しい時、要素 `refLayerPartIdc[x,y]` を持つ  $4 \times 4$  配列 `refLayerPartIdc` である現マクロブロックに対する参照レイヤパーティション識別

$4 \times 4$  配列 `refLayerPartIdc` が本処理の出力である場合には、その各要素 `refLayerPartIdc[x,y]` は、 $x$  および  $y$  座標を持つ  $4 \times 4$  ブロックを含む現マクロブロックのマクロブロックまたはサブマクロブロックパーティションのレイヤ間動き予測に使用され得る参照レイヤ表現内のパーティションに対する、マクロブロックアドレス、マクロブロックパーティションインデックス、およびサブマクロブロックパーティションインデックスを規定する。

ブロック座標  $x,y=0..3$  を持つ各  $4 \times 4$  ブロックに対して、 $4 \times 4$  配列 `refLayerPartIdc` の要素 `refLayerPartIdc[x,y]` は次の順序付けられたステップの適用により導出される。

1. G.6.2 小節にて規定される参照レイヤパーティションに対する導出処理が、輝度位置( $4*x+1,4*y+1$ )、`fieldMbFlag`、`refLayerFieldMbFlag`、`refLayerMbType`、および `refLayerSubMbType` を入力として起動され、出力はマクロブロックアドレス `refMbAddr`、マクロブロックパーティションインデックス `refMbPartIdx`、および参照レイヤ表現内のパーティションのサブマクロブロックパーティションインデックス `refSubMbPartIdx` である。

ビットストリームは、`refMbAddr`、`refMbPartIdx`、あるいは `refSubMbPartIdx` が利用不可とマークされる結果となるデータを含んではならない。

2. 配列 `refLayerPartIdc` の要素 `refLayerPartIdc[x,y]` は以下の通り導出される。

- もし `refLayerMbType[refMbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、あるいは `I_BL` に等しいならば、`refLayerPartIdc[x,y]` は -1 に等しく設定される。
- それ以外(`refLayerMbType[refMbAddr]` が `I_PCM`、`I_16x16`、`I_8x8`、`I_4x4`、あるいは `I_BL` に等しくない)、`refLayerPartIdc[x,y]` は次式より導出される。

$$\text{refLayerPartIdc}[x,y]=16*\text{refMbAddr}+4*\text{refMbPartIdx}+\text{refSubMbPartIdx} \quad (\text{G-222})$$

変数 `intraLLPredFlag` は以下の通り導出される。

- もし全ての要素 `refLayerPartIdc[x,y]`、 $x,y=0..3$  が -1 に等しいならば、`intraLLPredFlag` は 1 に等しく設定される。
- それ以外(どれか 1 つの要素 `refLayerPartIdc[x,y]`、 $x,y=0..3$ 、でも -1 に等しくない)、`intraLLPredFlag` は 0 に等しく設定される。

`intraLLPredFlag` が 0 に等しく、かつ `RestrictedSpatialResolutionChangeFlag` が 0 に等しい時、 $4 \times 4$  配列 `refLayerPartIdc` は次の順序付けられたステップに従い修正される。

記 以下の処理において -1 に等しい要素の `refLayerPartIdc[x,y]` は、-1 に等しくない要素の `refLayerPartIdc[x,y]` に置換される。本処理はまた、`RestrictedSpatialResolutionChangeFlag` が 1 に等しい時、あるいは `intraLLPredFlag` が 1 に等しい時にも適用され得る。しかしこの場合には、 $4 \times 4$  配列 `refLayerPartIdc` は修正されない。

1. ブロック座標  $xP,yP=0..1$  の各  $8 \times 8$  ブロックに対して、次の順序付けられたステップが規定される。
  - a. 変数  $xO$  と  $yO$  はそれぞれ  $(2*xP)$  と  $(2*yP)$  に等しく設定される。
  - b.  $2 \times 2$  配列 `procl4x4Blk`、 $xS,yS=0..1$ 、の全ての要素 `procl4x4Blk[xS,yS]` は 0 に等しく設定される。

- c. ブロック座標  $xS, yS=0..1$  を持つ現  $8 \times 8$  ブロックの  $4 \times 4$  ブロックは、 $(2*yS+xS)$  の昇順に処理され、そして、 $4 \times 4$  ブロックに対して  $refLayerPartIdc[xO+xS, yO+yS]$  が  $-1$  に等しい時、配列  $procI4x4Blk$  の要素  $procI4x4Blk[xS, yS]$  は  $1$  に等しく設定され、以下が適用される。

- もし  $procI4x4Blk[1-xS, yS]$  が  $0$  に等しく、かつ  $refLayerPartIdc[xO+1-xS, yO+yS]$  が  $-1$  に等しくないならば、要素  $refLayerPartIdc[xO+xS, yO+yS]$  は次式の通り修正される。

$$refLayerPartIdc[xO+xS, yO+yS] = refLayerPartIdc[xO+1-xS, yO+yS] \quad (G-223)$$

- それ以外、もし  $procI4x4Blk[xS, 1-yS]$  が  $0$  に等しく、かつ  $refLayerPartIdc[xO+xS, yO+1-yS]$  が  $-1$  に等しくないならば、要素  $refLayerPartIdc[xO+xS, yO+yS]$  は次式の通り修正される。

$$refLayerPartIdc[xO+xS, yO+yS] = refLayerPartIdc[xO+xS, yO+1-yS] \quad (G-224)$$

- それ以外、もし  $procI4x4Blk[1-xS, 1-yS]$  が  $0$  に等しく、かつ  $refLayerPartIdc[xO+1-xS, yO+1-yS]$  が  $-1$  に等しくないならば、要素  $refLayerPartIdc[xO+xS, yO+yS]$  は次式の通り修正される。

$$refLayerPartIdc[xO+xS, yO+yS] = refLayerPartIdc[xO+1-xS, yO+1-yS] \quad (G-225)$$

- それ以外、要素  $refLayerPartIdc[xO+xS, yO+yS]$  は修正されない。

2.  $2 \times 2$  配列  $procI8x8Blk$ 、 $xP, yP=0..1$ 、の全ての要素  $procI8x8Blk[xP, yP]$  は、 $0$  に等しく設定される。
3. ブロック座標  $xP, yP=0..1$  を持つ  $8 \times 8$  ブロックは、 $(2*yP+xP)$  の昇順に処理され、そして、 $8 \times 8$  ブロックに対して  $refLayerPartIdc[2*xP, 2*yP]$  が  $-1$  に等しい時、配列  $procI8x8Blk$  の要素  $procI8x8Blk[xP, yP]$  は  $1$  に等しく設定され、以下が適用される。

- もし  $procI8x8Blk[1-xP, yP]$  が  $0$  に等しく、かつ  $refLayerPartIdc[2-xP, 2*yP]$  が  $-1$  に等しくないならば、要素  $refLayerPartIdc[2*xP+xS, 2*yP+yS]$ 、 $xS, yS=0..1$  は次式の通り修正される。

$$refLayerPartIdc[2*xP+xS, 2*yP+yS] = refLayerPartIdc[2-xP, 2*yP+yS] \quad (G-226)$$

- それ以外、もし  $procI8x8Blk[xP, 1-yP]$  が  $0$  に等しく、かつ  $refLayerPartIdc[2*xP, 2-yP]$  が  $-1$  に等しくないならば、要素  $refLayerPartIdc[2*xP+xS, 2*yP+yS]$ 、 $xS, yS=0..1$  は次式の通り修正される。

$$refLayerPartIdc[2*xP+xS, 2*yP+yS] = refLayerPartIdc[2*xP+xS, 2-yP] \quad (G-227)$$

- それ以外、もし  $procI8x8Blk[1-xP, 1-yP]$  が  $0$  に等しく、かつ  $refLayerPartIdc[2-xP, 2-yP]$  が  $-1$  に等しくないならば、要素  $refLayerPartIdc[2*xP+xS, 2*yP+yS]$ 、 $xS, yS=0..1$  は次式の通り修正される。

$$refLayerPartIdc[2*xP+xS, 2*yP+yS] = refLayerPartIdc[2-xP, 2-yP] \quad (G-228)$$

- それ以外、要素  $refLayerPartIdc[2*xP+xS, 2*yP+yS]$ 、 $xS, yS=0..1$  は修正されない。

#### G.8.6.1.2 参照インデックスと動きベクトルのレイヤ間予測因子に対する導出処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数

fieldMbFlag

- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックで、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックの予測利用フラグを規定する 2 つの(RefLayerPicSizeInMbs)×4 配列 refLayerPredFlagL0 と refLayerPredFlagL1
- 参照レイヤ表現のマクロブロックの参照インデックスを規定する 2 つの(RefLayerPicSizeInMbs)×4 配列 refLayerRefIdxL0 と refLayerRefIdxL1
- 参照レイヤ表現のマクロブロックの動きベクトル成分を規定する 2 つの(RefLayerPicSizeInMbs)×4×4×2 配列 refLayerMvL0 と refLayerMvL1
- 現マクロブロックの 4×4 ブロックに対する参照レイヤパーティション識別を規定する 4×4 配列 refLayerPartIdx
- CroppingChangeFlag が 1 に等しく、かつ(slice\_type%5)が 2 より小さい時は、参照ピクチャリスト refPicList0
- CroppingChangeFlag が 1 に等しく、かつ(slice\_type%5)が 1 に等しい時は、参照ピクチャリスト refPicList1

本処理の出力は以下の通りである。

- 現マクロブロックの参照インデックスに対するレイヤ間予測因子を規定する 2 つの 2×2 配列 refIdxILPredL0 と refIdxILPredL1
- 現マクロブロックの動きベクトル成分に対するレイヤ間予測因子を規定する 2 つの 4×4×2 配列 mvILPredL0 と mvILPredL1

tempRefIdxPredL0 と tempRefIdxPredL1 をそれぞれ tempRefIdxPredL0[x,y] と tempRefIdxPredL1[x,y]の要素を持つ 2 つの 4×4 配列とし、それらは参照インデックスの補助レイヤ間予測因子を規定する。

x,y=0..3、および 0 と 1 に置換される X により索引される各 4×4 ブロックインデックスに対して、補助参照インデックス予測因子 tempRefIdxPredLX[x,y]と動きベクトル予測因子 mvILPredLX[x,y]は次の通り導出される。

- もし refLayerPredFlagLX[refLayerPartIdx[x,y]/16][(refLayerPartIdx[x,y]%16)/4]が 0 に等しいならば、参照インデックス予測因子 tempRefIdxPredLX[x,y]と動きベクトル予測因子 mvILPredLX[x,y]は次式により導出される。

$$\text{tempRefIdxPredLX}[x,y]=-1 \quad (\text{G-229})$$

$$\text{mvILPredLX}[x,y][0]=0 \quad (\text{G-230})$$

$$\text{mvILPredLX}[x,y][1]=0 \quad (\text{G-231})$$

- それ以外(refLayerPredFlagLX[refLayerPartIdx[x,y]/16][(refLayerPartIdx[x,y]%16)/4]が 1 に等しい)、次の順序付けられたステップが規定される。

1. 変数 refMbAddr、refMbPartIdx、および refSubMbPartIdx は次式より導出される。

$$\text{refMbAddr}=\text{refLayerPartIdx}[x,y]/16 \quad (\text{G-232})$$

$$\text{refMbPartIdx}=(\text{refLayerPartIdx}[x,y]\%16)/4 \quad (\text{G-233})$$

$$\text{refSubMbPartIdx}=\text{refLayerPartIdx}[x,y]\%4 \quad (\text{G-234})$$

2. 補助参照インデックス予測因子  $\text{tempRefIdxPredLX}[x,y]$  は次式により導出される。

$$\begin{aligned} \text{tempRefIdxPredLX}[x,y] &= \text{refLayerRefIdxLX}[\text{refMbAddr}][\text{refMbPartIdx}] \\ &\quad * (1 + \text{fieldMbFlag} - \text{field\_pic\_flag}) \\ &\quad / (1 + \text{refLayerFieldMbFlag}[\text{refMbAddr}] - \text{RefLayerFieldPicFlag}) \end{aligned} \quad (\text{G-235})$$

3. 動きベクトル  $\text{aMv}$  は  $\text{refLayerMvLX}[\text{refMbAddr}][\text{refMbPartIdx}][\text{refSubMbPartIdx}]$  に等しく設定され、その後垂直成分  $\text{aMv}[1]$  は次式により修正される。

$$\text{aMv}[1] = \text{aMv}[1] * (1 + \text{refLayerFieldMbFlag}[\text{refMbAddr}]) \quad (\text{G-236})$$

4. 変数  $\text{scaledW}$ 、 $\text{scaledH}$ 、 $\text{refLayerW}$ 、および  $\text{refLayerH}$  は次式より導出される。

$$\text{scaledW} = \text{ScaledRefLayerPicWidthInSamples}_L \quad (\text{G-237})$$

$$\text{scaledH} = \text{ScaledRefLayerPicHeightInSamples}_L * (1 + \text{field\_pic\_flag}) \quad (\text{G-238})$$

$$\text{refLayerW} = \text{RefLayerPicWidthInSamples}_L \quad (\text{G-239})$$

$$\text{refLayerH} = \text{RefLayerPicHeightInSamples}_L * (1 + \text{RefLayerFieldPicFlag}) \quad (\text{G-240})$$

5. 変数  $\text{dOX}$ 、 $\text{dOY}$ 、 $\text{dSW}$ 、および  $\text{dSH}$  は以下の通り導出される。

- もし  $\text{CroppingChangeFlag}$  が 0 に等しいか、あるいは参照ピクチャ  $\text{refPicListX}[\text{tempRefIdxPredLX}[x,y]]$  が利用可能でないならば、 $\text{dOX}$ 、 $\text{dOY}$ 、 $\text{dSW}$ 、および  $\text{dSH}$  は 0 に等しく設定される。
- それ以外 ( $\text{CroppingChangeFlag}$  が 1 に等しく、かつ参照ピクチャ  $\text{refPicListX}[\text{tempRefIdxPredLX}[x,y]]$  が利用可能)、変数  $\text{refPicScaledRefLayerLeftOffset}$ 、 $\text{refPicScaledRefLayerRightOffset}$ 、 $\text{refPicScaledRefLayerTopOffset}$ 、および  $\text{refPicScaledRefLayerBottomOffset}$  は、現レイヤ表現の参照ピクチャ  $\text{refPicListX}[\text{tempRefIdxPredLX}[x,y]]$  の  $\text{DQId}$  と同じ値を持つレイヤ表現に関連付けられた、変数  $\text{ScaledRefLayerLeftOffset}$ 、 $\text{ScaledRefLayerRightOffset}$ 、 $\text{ScaledRefLayerTopOffset}$ 、および  $\text{ScaledRefLayerBottomOffset}$  にそれぞれ等しく設定される。また、変数  $\text{dOX}$ 、 $\text{dOY}$ 、 $\text{dSW}$ 、および  $\text{dSH}$  は次式より導出される。

$$\text{dOX} = \text{ScaledRefLayerLeftOffset} - \text{refPicScaledRefLayerLeftOffset} \quad (\text{G-241})$$

$$\text{dOY} = \text{ScaledRefLayerTopOffset} - \text{refPicScaledRefLayerTopOffset} \quad (\text{G-242})$$

$$\text{dSW} = \text{ScaledRefLayerRightOffset} - \text{refPicScaledRefLayerRightOffset} + \text{dOX} \quad (\text{G-243})$$

$$\text{dSH} = \text{ScaledRefLayerBottomOffset} - \text{refPicScaledRefLayerBottomOffset} + \text{dOY} \quad (\text{G-244})$$

6. 変数  $\text{scaleX}$  と  $\text{scaleY}$  は次式より導出される。

$$\text{scaleX} = (((\text{scaledW} + \text{dSW}) << 16) + (\text{refLayerW} >> 1)) / \text{refLayerW} \quad (\text{G-245})$$

$$\text{scaleY} = (((\text{scaledH} + \text{dSH}) << 16) + (\text{refLayerH} >> 1)) / \text{refLayerH} \quad (\text{G-246})$$

7. 動きベクトル  $\text{aMv}$  は次式によりスケーリング処理される。

$$aMv[0]=(aMv[0]*scaleX+32768)>>16 \quad (G-247)$$

$$aMv[1]=(aMv[1]*scaleY+32768)>>16 \quad (G-248)$$

8. CroppingChangeFlag が 1 に等しい時、動きベクトル aMv は次の順序付けられたステップを適用して修正される。

a. 6.4.1 小節で規定される逆マクロブロック走査処理が、CurrMbAddr を入力として起動され、出力は輝度位置(xMbPic,yMbPic)である。6.4.1 小節で規定される本処理の起動において、現マクロブロックは、fieldMbFlag が 1 に等しい時はフィールドマクロブロックとして扱われ、fieldMbFlag が 0 に等しい時はフレームマクロブロックとして扱われる。

b. 輝度位置(xFrm,yFrm)は次式より導出される。

$$xFrm=(xMbPic+(4*x+1)) \quad (G-249)$$

$$yFrm=(yMbPic+(4*y+1)*(1+fieldMbFlag-field_pic_flag))*(1+field_pic_flag) \quad (G-250)$$

c. 変数 scaleX と scaleY は次式により修正される。

$$scaleX=((4*dSW)<<16+(scaledW>>1))/scaledW \quad (G-251)$$

$$scaleY=((4*dSH)<<16+(scaledH>>1))/scaledH \quad (G-252)$$

d. 動きベクトル aMv は次式により修正される。

$$aMv[0]+(((xFrm-ScaledRefLayerLeftOffset)*scaleX+32768)>>16)-4*dOX \quad (G-253)$$

$$aMv[1]+(((yFrm-ScaledRefLayerTopOffset)*scaleY+32768)>>16)-4*dOY \quad (G-254)$$

9. 動きベクトル予測因子 mvILPredLX[x,y]は次式より導出される。

$$mvILPredLX[x,y][0]=aMv[0] \quad (G-255)$$

$$mvILPredLX[x,y][1]=aMv[1]/(1+fieldMbFlag) \quad (G-256)$$

xP,yP=0..1 により索引される各 8×8 ブロックに対して、また、0 または 1 で置換される X に対して、参照インデックス予測因子 refIdxILPredLX[xP,yP]は tempRefIdxPredLX[2\*xP,2\*yP]に等しく設定される。また、RestrictedSpatialResolutionChangeFlag が 0 に等しい時、次の順序付けられたステップが規定される。

記 – 以下の処理はまたRestrictedSpatialResolutionChangeFlagが1に等しい時にも適用され得る。しかし、この場合は参照インデックス予測因子 refIdxILPredLX[xP,yP] と動きベクトル予測因子 mvILPredLX[2\*xP+xS,2\*yP+yS]、xS,yS=0..1は修正されない。

1. 現 8×8 ブロックの、xS,yS=0..1 により索引される 4×4 ブロックは、(2\*yS+xS)の昇順に処理される。また、各 4×4 ブロックに対して、参照インデックス予測因子 refIdxILPredLX[xP,yP]は次式により修正される。

$$\begin{aligned} \text{refIdxILPredLX}[xP,yP]=\text{MinPositive}(\text{refIdxILPredLX}[xP,yP], \\ \text{tempRefIdxPredLX}[2*xP+xS,2*yP+yS]) \end{aligned} \quad (G-257)$$

ここで

$$\text{MinPositive}(a, b) = \begin{cases} \text{Min}(a, b) & \text{もし } a \geq 0 \text{かつ } b \geq 0 \text{ならば} \\ \text{Max}(a, b) & \text{それ以外} \end{cases} \quad (\text{G-258})$$

2. 現  $8 \times 8$  ブロックの、 $xS, yS=0..1$  により索引される  $4 \times 4$  ブロックは、 $(2*yS+xS)$ の昇順に処理される。  
また、各  $4 \times 4$  ブロックに対して、 $\text{tempRefIdxPredLX}[2*xP+xS, 2*yP+yS]$ が参照インデックス予測因子  $\text{refIdxILPredLX}[xP, yP]$ に等しくない時、以下が適用される。

- もし  $\text{tempRefIdxPredLX}[2*xP+1-xS, 2*yP+yS]$ が  $\text{refIdxILPredLX}[xP, yP]$ に等しいならば、動きベクトル予測因子  $\text{mvILPredLX}[2*xP+xS, 2*yP+yS]$ は次式により修正される。

$$\text{mvILPredLX}[2*xP+xS, 2*yP+yS] = \text{mvILPredLX}[2*xP+1-xS, 2*yP+yS] \quad (\text{G-259})$$

- それ以外、もし  $\text{tempRefIdxPredLX}[2*xP+xS, 2*yP+1-yS]$ が  $\text{refIdxILPredLX}[xP, yP]$ に等しいならば、動きベクトル予測因子  $\text{mvILPredLX}[2*xP+xS, 2*yP+yS]$ は次式により修正される。

$$\text{mvILPredLX}[2*xP+xS, 2*yP+yS] = \text{mvILPredLX}[2*xP+xS, 2*yP+1-yS] \quad (\text{G-260})$$

- それ以外、 $(\text{tempRefIdxPredLX}[2*xP+1-xS, 2*yP+1-yS])$ が  $\text{refIdxILPredLX}[xP, yP]$ に等しい)、動きベクトル予測因子  $\text{mvILPredLX}[2*xP+xS, 2*yP+yS]$ は次式により修正される。

$$\text{mvILPredLX}[2*xP+xS, 2*yP+yS] = \text{mvILPredLX}[2*xP+1-xS, 2*yP+1-yS] \quad (\text{G-261})$$

$\text{RestrictedSpatialResolutionChangeFlag}$  が 0 に等しい時、 $xP, yP=0..1$  により索引される各  $8 \times 8$  ブロックに対して、動きベクトル予測因子配列  $\text{mvILPredL0}$  と  $\text{mvILPredL1}$  は次の順序付けられたステップを適用して修正される。

記 以下の処理は $\text{RestrictedSpatialResolutionChangeFlag}$ が1に等しい時は適用されてはならない。

1. 変数  $\text{maxX}$  は以下の通り導出される。

- もし  $\text{slice\_type}$  が EB に等しいならば、 $\text{maxX}$  は 1 に等しく設定される。
- それ以外( $\text{slice\_type}$  が EP に等しい)、 $\text{maxX}$  は 0 に等しく設定される。

2. 変数  $xO$  と  $yO$  はそれぞれ、 $(2*xP)$ と $(2*yP)$ に等しく設定される。

3. 2つの動きベクトル  $\text{mv}_1$  と  $\text{mv}_2$  の関数  $\text{mvDiff}(\text{mv}_1, \text{mv}_2)$ は次式で定義される。

$$\text{mvDiff}(\text{mv}_1, \text{mv}_2) = \text{Abs}(\text{mv}_1[0] - \text{mv}_2[0]) + \text{Abs}(\text{mv}_1[1] - \text{mv}_2[1]) \quad (\text{G-262})$$

4. 変数  $\text{subPartSize}$  は以下の通り導出される。

- もし、 $X=0..\text{maxX}$  に対して、以下の全ての条件が真であるならば、 $\text{subPartSize}$  は  $8 \times 8$  に等しく設定される。
  - $\text{mvDiff}(\text{mvILPredLX}[xO, yO], \text{mvILPredLX}[xO+1, yO])$ は 1 より小さいか等しい。
  - $\text{mvDiff}(\text{mvILPredLX}[xO, yO], \text{mvILPredLX}[xO, yO+1])$ は 1 より小さいか等しい。
  - $\text{mvDiff}(\text{mvILPredLX}[xO, yO], \text{mvILPredLX}[xO+1, yO+1])$ は 1 より小さいか等しい。
- それ以外、もし  $X=0..\text{maxX}$  に対して、以下の全ての条件が真であるならば、 $\text{subPartSize}$  は  $8 \times 4$  に等しく設定される。

- $\text{mvDiff}(\text{mvILPredLX}[xO,yO],\text{mvILPredLX}[xO+1,yO])$ は1より小さいか等しい。
- $\text{mvDiff}(\text{mvILPredLX}[xO,yO+1],\text{mvILPredLX}[xO+1,yO+1])$ は1より小さいか等しい。
- それ以外、もし  $X=0..\text{maxX}$  に対して、以下の全ての条件が真であるならば、 $\text{subPartSize}$  は  $4 \times 8$  に等しく設定される。
  - $\text{mvDiff}(\text{mvILPredLX}[xO,yO],\text{mvILPredLX}[xO,yO+1])$ は1より小さいか等しい。
  - $\text{mvDiff}(\text{mvILPredLX}[xO+1,yO],\text{mvILPredLX}[xO+1,yO+1])$ は1より小さいか等しい。
- それ以外、 $\text{subPartSize}$  は  $4 \times 4$  に等しく設定される。

5.  $X=0..\text{maxX}$  に対して、 $\text{subPartSize}$  が  $4 \times 4$  に等しくない時、動きベクトル  $\text{tempMvALX}$  と ( $\text{subPartSize}$  が  $8 \times 4$  あるいは  $4 \times 8$  に等しい時) $\text{tempMvBLX}$  は、次式より導出される。

- もし  $\text{subPartSize}$  が  $8 \times 8$  に等しいならば、 $\text{tempMvALX}$  は次式より導出される。

$$\begin{aligned} \text{tempMvALX}[c] = & (\text{mvILPredLX}[xO, yO][c] + \\ & \text{mvILPredLX}[xO+1,yO][c] + \\ & \text{mvILPredLX}[xO, yO+1][c] + \\ & \text{mvILPredLX}[xO+1,yO+1][c] + 2) \gg 2, c=0..1 \end{aligned} \quad (\text{G-263})$$

- それ以外、もし  $\text{subPartSize}$  が  $8 \times 4$  に等しいならば、 $\text{tempMvALX}$  と  $\text{tempMvBLX}$  は次式より導出される。

$$\begin{aligned} \text{tempMvALX}[c] = & (\text{mvILPredLX}[xO, yO][c] + \\ & \text{mvILPredLX}[xO+1,yO][c] + 1) \gg 1, c = 0..1 \end{aligned} \quad (\text{G-264})$$

$$\begin{aligned} \text{tempMvBLX}[c] = & (\text{mvILPredLX}[xO, yO+1][c] + \\ & \text{mvILPredLX}[xO+1,yO+1][c] + 1) \gg 1, c = 0..1 \end{aligned} \quad (\text{G-265})$$

- それ以外 ( $\text{subPartSize}$  が  $4 \times 8$  に等しい)、 $\text{tempMvALX}$  と  $\text{tempMvBLX}$  は次式より導出される。

$$\begin{aligned} \text{tempMvALX}[c] = & (\text{mvILPredLX}[xO,yO][c] + \\ & \text{mvILPredLX}[xO,yO+1][c] + 1) \gg 1, c = 0..1 \end{aligned} \quad (\text{G-266})$$

$$\begin{aligned} \text{tempMvBLX}[c] = & (\text{mvILPredLX}[xO+1,yO][c] + \\ & \text{mvILPredLX}[xO+1,yO+1][c] + 1) \gg 1, c = 0..1 \end{aligned} \quad (\text{G-267})$$

6.  $X=0..\text{maxX}$  に対して、 $\text{subPartSize}$  が  $4 \times 4$  に等しくない時、動きベクトル予測因子配列  $\text{mvILPredLX}$  は次式により修正される。

- もし  $\text{subPartSize}$  が  $8 \times 8$  に等しいならば、配列  $\text{mvILPredLX}$  は次式により修正される。

$$\text{mvILPredLX}[xO+xS,yO+yS][c] = \text{tempMvALX}[c], xS,yS=0..1, c=0..1 \quad (\text{G-268})$$

- それ以外、もし  $\text{subPartSize}$  が  $8 \times 4$  に等しいならば、配列  $\text{mvILPredLX}$  は次式により修正される。

$$\text{mvILPredLX}[xO+xS,yO][c] = \text{tempMvALX}[c], xS=0..1, c=0..1 \quad (\text{G-269})$$

$$\text{mvILPredLX}[xO+xS,yO+1][c] = \text{tempMvBLX}[c], xS=0..1, c=0..1 \quad (\text{G-270})$$

- それ以外(subPartSize が 4x8 に等しい)、配列 mvILPredLX は次式により修正される。

$$\text{mvILPredLX}[xO, yO+yS][c]=\text{tempMvALX}[c], yS=0..1, c=0..1 \quad (\text{G-271})$$

$$\text{mvILPredLX}[xO+1,yO+yS][c]=\text{tempMvBLX}[c], yS=0..1, c=0..1 \quad (\text{G-272})$$

### G.8.6.1.3 PおよびBマクロブロックおよびサブマクロブロックタイプに対するレイヤ間予測因子の導出処理

本処理は、slice\_type が EP あるいは EB に等しい時のみ起動される。

本処理への入力は以下の通りである。

- 現マクロブロックの参照インデックスに対する予測因子を規定する 2 つの 2×2 配列 refldxILPredL0 と refldxILPredL1
- 現マクロブロックの動きベクトルに対する予測因子を規定する 2 つの 4×4×2 配列 mvILPredL0 と mvILPredL1

本処理の出力は以下の通りである。

- 現マクロブロックのマクロブロックタイプに対する予測因子を規定する変数 mbTypeILPred
- 現マクロブロックのサブマクロブロックタイプに対する予測因子を規定する、4 つの要素を持つリスト subMbTypeILPred

変数 maxX は以下により導出される。

- もし slice\_type が EB に等しいならば、maxX は 1 に等しく設定される。
- それ以外(slice\_type が EP に等しい)、maxX は 0 に等しく設定される。

マクロブロックタイプ予測因子 mbTypeILPred は次の順序付けられたステップを適用することで導出される。

#### 1. 変数 partitionSize は以下の通り導出される。

- もし  $X=0..maxX$  に対して次の全ての条件が真であるならば、partitionSize は 16x16 に等しく設定される。
  - $x,y=0..1$  である全ての要素 refldxILPredLX[x,y]が同じである。
  - $x,y=0..3$  である全ての要素 mvILPredLX[x,y]が同じである。
- それ以外、もし  $X=0..maxX$  に対して次の全ての条件が真であるならば、partitionSize は 16x8 に等しく設定される。
  - refldxILPredLX[0,0]は refldxILPredLX[1,0]に等しい。
  - refldxILPredLX[0,1]は refldxILPredLX[1,1]に等しい。
  - $x=0..3, y=0..1$  である全ての要素 mvILPredLX[x,y]が同じである。
  - $x=0..3, y=2..3$  である全ての要素 mvILPredLX[x,y]が同じである。
- それ以外、もし  $X=0..maxX$  に対して次の全ての条件が真であるならば、partitionSize は 8x16 に

等しく設定される。

- $\text{refIdxILPredLX}[0,0]$ は  $\text{refIdxILPredLX}[0,1]$ に等しい。
  - $\text{refIdxILPredLX}[1,0]$ は  $\text{refIdxILPredLX}[1,1]$ に等しい。
  - $x=0..1$ 、 $y=0..3$  である全ての要素  $\text{mvILPredLX}[x,y]$ が同じである。
  - $x=2..3$ 、 $y=0..3$  である全ての要素  $\text{mvILPredLX}[x,y]$ が同じである。
- それ以外、 $\text{partitionSize}$  は  $8 \times 8$  に等しく設定される。
2.  $\text{slice\_type}$  が EB に等しく、かつ  $\text{partitionSize}$  が  $8 \times 8$  に等しくない時、変数  $\text{partPredModeA}$  は次式より導出される。

$$\begin{aligned} \text{partPredModeA} = & (\text{refIdxILPredL1}[0,0] \geq 0 ? 2 : 0) + \\ & (\text{refIdxILPredL0}[0,0] \geq 0 ? 1 : 0) \end{aligned} \quad (\text{G-273})$$

3.  $\text{slice\_type}$  が EB に等しく、かつ  $\text{partitionSize}$  が  $16 \times 8$  または  $8 \times 16$  に等しい時、変数  $\text{partPredModeB}$  は次式より導出される。

$$\begin{aligned} \text{partPredModeB} = & (\text{refIdxILPredL1}[1,1] \geq 0 ? 2 : 0) + \\ & (\text{refIdxILPredL0}[1,1] \geq 0 ? 1 : 0) \end{aligned} \quad (\text{G-274})$$

4.  $\text{slice\_type}$ 、 $\text{partitionSize}$ 、 $\text{partPredModeA}$ (適用可能の時)、および  $\text{partPredModeB}$ (適用可能の時)に依存して、マクロブロックタイプ予測因子  $\text{mbTypeILPred}$  は付表 G-6/JT-H264 において規定される通り導出される。

リスト  $\text{subMbTypeILPred}$  の全ての要素  $\text{subMbTypeILPred}[\text{mbPartIdx}]$ 、 $\text{mbPartIdx}=0..3$  は、“無規定”とマークされる。

$\text{mbTypeILPred}$  が  $P\_8 \times 8$ 、あるいは  $B\_8 \times 8$  に等しい時、各要素  $\text{subMbTypeILPred}[\text{mbPartIdx}]$ 、 $\text{mbPartIdx}=0..3$  は、以下の順序付けられたステップを適用することにより修正される。

1. 座標オフセット( $xO,yO$ )は $(2*(\text{mbPartIdx}\%2),2*(\text{mbPartIdx}/2))$ に等しく設定される。
2. 変数  $\text{subPartitionSize}$  は以下の通り導出される。
  - もし  $X=0..\text{maxX}$  に対して、全ての要素  $\text{mvILPredLX}[xO+xS,yO+yS]$ 、 $xS,yS=0..1$  が同一であるならば、 $\text{subPartitionSize}$  は  $8 \times 8$  に等しく設定される。
  - それ以外、もし  $X=0..\text{maxX}$  に対して、 $\text{mvILPredLX}[xO,yO]$ が  $\text{mvILPredLX}[xO+1,yO]$ に等しく、かつ  $\text{mvILPredLX}[xO,yO+1]$ が  $\text{mvILPredLX}[xO+1,yO+1]$ に等しいならば、 $\text{subPartitionSize}$  は  $8 \times 4$  に等しく設定される。
  - それ以外、もし  $X=0..\text{maxX}$  に対して、 $\text{mvILPredLX}[xO,yO]$ が  $\text{mvILPredLX}[xO,yO+1]$ に等しく、かつ  $\text{mvILPredLX}[xO+1,yO]$ が  $\text{mvILPredLX}[xO+1,yO+1]$ に等しいならば、 $\text{subPartitionSize}$  は  $4 \times 8$  に等しく設定される。
  - それ以外、 $\text{subPartitionSize}$  は  $4 \times 4$  に等しく設定される。
3.  $\text{slice\_type}$  が EB に等しい時、変数  $\text{partPredMode}$  は次式で導出される。

$$\begin{aligned} \text{partPredMode} &= (\text{refIdxILPredL1}[\text{xO}/2, \text{yO}/2] \geq 0 ? 2 : 0) + \\ & (\text{refIdxILPredL0}[\text{xO}/2, \text{yO}/2] \geq 0 ? 1 : 0) \end{aligned} \quad (\text{G-275})$$

4. slice\_type、subPartitionSize、および partPredMode(適用可能な時)に依存して、サブマクロブロックタイプ予測因子 subMbTypeILPred[mbPartIdx]は付表 G-7/JT-H264 において規定される通り導出される。

付表 G-6/JT-H264 – マクロブロックタイプ予測因子 mbTypeILPred

(ITU-T H.264)

slice_type	partitionSize	partPredModeA	partPredModeB	mbTypeILPred	slice_type	partitionSize	partPredModeA	partPredModeB	mbTypeILPred
EB	16x16	1	na	B_L0_16x16	EB	16x8	2	3	B_L1_Bi_16x8
EB	16x16	2	na	B_L1_16x16	EB	8x16	2	3	B_L1_Bi_8x16
EB	16x16	3	na	B_Bi_16x16	EB	16x8	3	1	B_Bi_L0_16x8
EB	16x8	1	1	B_L0_L0_16x8	EB	8x16	3	1	B_Bi_L0_8x16
EB	8x16	1	1	B_L0_L0_8x16	EB	16x8	3	2	B_Bi_L1_16x8
EB	16x8	2	2	B_L1_L1_16x8	EB	8x16	3	2	B_Bi_L1_8x16
EB	8x16	2	2	B_L1_L1_8x16	EB	16x8	3	3	B_Bi_Bi_16x8
EB	16x8	1	2	B_L0_L1_16x8	EB	8x16	3	3	B_Bi_Bi_8x16
EB	8x16	1	2	B_L0_L1_8x16	EB	8x8	na	na	B_8x8
EB	16x8	2	1	B_L1_L0_16x8	EP	16x16	na	na	P_L0_16x16
EB	8x16	2	1	B_L1_L0_8x16	EP	16x8	na	na	P_L0_L0_16x8
EB	16x8	1	3	B_L0_Bi_16x8	EP	8x16	na	na	P_L0_L0_8x16
EB	8x16	1	3	B_L0_Bi_8x16	EP	8x8	na	na	P_8x8

付表 G-7/JT-H264 – サブマクロブロックタイプ予測因子 subMbTypeILPred[mbPartIdx]

(ITU-T H.264)

slice_type	subPartitionSize	partPredMode	subMbTypeLPred [ mbPartIdx ]	slice_type	subPartitionSize	partPredMode	subMbTypeLPred [ mbPartIdx ]
EB	8x8	1	B_L0_8x8	EB	4x8	3	B_Bi_4x8
EB	8x8	2	B_L1_8x8	EB	4x4	1	B_L0_4x4
EB	8x8	3	B_Bi_8x8	EB	4x4	2	B_L1_4x4
EB	8x4	1	B_L0_8x4	EB	4x4	3	B_Bi_4x4
EB	8x4	2	B_L1_8x4	EP	8x8	na	P_L0_8x8
EB	8x4	3	B_Bi_8x4	EP	8x4	na	P_L0_8x4
EB	4x8	1	B_L0_4x8	EP	4x8	na	P_L0_4x8
EB	4x8	2	B_L1_4x8	EP	4x4	na	P_L0_4x4

#### G.8.6.2 INTRAサンプルの再サンプリング処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 参照レイヤ表現のマクロブロックに対するスライス識別を規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerSliceIdx
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックで、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerMbType
- 参照レイヤ表現の輝度サンプルの  $(\text{RefLayerPicWidthInSamples}_L) \times (\text{RefLayerPicHeightInSamples}_L)$  配列 refLayerPicSamples<sub>L</sub>
- 輝度サンプルの  $(\text{PicWidthInSamples}_L) \times (\text{PicHeightInSamples}_L)$  配列 picSamples<sub>L</sub>
- ChromaArrayType が 0 に等しくない時、参照レイヤ表現における色差サンプルの 2 つの  $(\text{RefLayerPicWidthInSamples}_C) \times (\text{RefLayerPicHeightInSamples}_C)$  配列 refLayerPicSamples<sub>Cb</sub> と refLayerPicSamples<sub>Cr</sub>
- ChromaArrayType が 0 に等しくない時、色差サンプルの 2 つの  $(\text{PicWidthInSamples}_C) \times (\text{PicHeightInSamples}_C)$  配列 picSamples<sub>Cb</sub> と picSamples<sub>Cr</sub>

本処理の出力は以下の通りである。

- 輝度サンプルの配列 picSamples<sub>L</sub> の修正版

- ChromaArrayType が 0 に等しくない時、色差サンプルの配列 picSamples<sub>Cb</sub> と picSamples<sub>Cr</sub> の修正版

G.8.6.2.1 小小小節で規定されるマクロブロック色成分の INTRA サンプルに対する再サンプリング処理が、0 に等しい chromaFlag、16 に等しい mbW、16 に等しい mbH、fieldMbFlag、refLayerPicSamples<sub>L</sub>、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は現マクロブロックの輝度成分に対する Intra\_Base 予測サンプルの 16x16 配列 mbPred<sub>L</sub> である。

ChromaArrayType が 0 に等しくない時、CX を Cb と Cr に置換して、G.8.6.2.1 小小小節で規定されるマクロブロック色成分の INTRA サンプルに対する再サンプリング処理が、1 に等しい chromaFlag、MbWidthC に等しい mbW、MbHeightC に等しい mbH、fieldMbFlag、refLayerPicSamples<sub>CX</sub>、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は現マクロブロックの CX 成分に対する Intra\_Base 予測サンプルの(MbWidthC)×(MbHeightC)配列 mbPred<sub>CX</sub> である。

G.8.5.4.1 小小小節で規定されるピクチャサンプル配列の構築処理が、fieldMbFlag、mbPred<sub>L</sub>、picSamples<sub>L</sub>、更に ChromaArrayType が 0 に等しくない時には mbPred<sub>Cb</sub>、mbPred<sub>Cr</sub>、picSamples<sub>Cb</sub>、および picSamples<sub>Cr</sub> を入力として起動され、出力は picSamples<sub>L</sub> の修正版と、ChromaArrayType が 0 でない時には、picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub> の修正版である。

#### G.8.6.2.1 マクロブロック色成分のINTRAサンプルに対する再サンプリング処理

本処理への入力は以下の通りである。

- 再サンプリング処理の対象が輝度あるいは色差成分のいずれであるかを規定する変数 chromaFlag
- 対象の色成分のためのマクロブロックの幅と高さをそれぞれ規定する 2 つの変数 mbW と mbH
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- chromaFlag が 0 に等しい時には参照レイヤ表現に対する構築された INTRA 輝度サンプル値を含む (RefLayerPicWidthInSamples<sub>L</sub>)×(RefLayerPicHeightInSamples<sub>L</sub>)配列であり、または chromaFlag が 1 に等しい時には参照レイヤ表現に対する構築された INTRA 色差サンプル値を含む (RefLayerPicWidthInSamples<sub>C</sub>)×(RefLayerPicHeightInSamples<sub>C</sub>)配列である、1 つの配列 refLayerPicSamples
- 参照レイヤ表現のマクロブロックに対するスライス識別を規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerSliceIdc
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックで、どのマクロブロックがフレームマクロブロックであるかを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerMbType

本処理の出力は、Intra\_Base 予測サンプルの(mbW)×(mbH)配列 mbPred である。

変数 botFieldFlag は以下の通り導出される。

- もし RefLayerFrameMbsOnlyFlag が 0 に等しいならば、botFieldFlag は 0 に等しく設定される。
- それ以外、もし field\_pic\_flag が 1 に等しいならば、botFieldFlag は bottom\_field\_flag に等しく設定される。

- それ以外、もし RefLayerFieldPicFlag が 1 に等しいならば、botFieldFlag は RefLayerBottomFieldFlag に等しく設定される。
- それ以外、もし fieldMbFlag が 1 に等しいならば、botFieldFlag は(CurrMbAddr%2)に等しく設定される。
- それ以外、botFieldFlag は 0 に等しく設定される。

変数 frameBasedResamplingFlag は以下の通り導出される。

- もし以下の全ての条件が真であるならば、frameBasedResamplingFlag は 1 に等しく設定される。
  - RefLayerFrameMbsOnlyFlag は 1 に等しい
  - frame\_mbs\_only\_flag は 1 に等しい
- それ以外、frameBasedResamplingFlag は 0 に等しく設定される。

変数 topAndBotResamplingFlag は以下の通り導出される。

- もし以下の全ての条件が真であるならば、topAndBotResamplingFlag は 1 に等しく設定される。
  - RefLayerFrameMbsOnlyFlag は 0 に等しい
  - RefLayerFieldPicFlag は 0 に等しい
  - frame\_mbs\_only\_flag は 0 に等しい
  - fieldMbFlag は 0 に等しい
- それ以外、topAndBotResamplingFlag は 0 に等しく設定される。

変数 filteringModeFlag は以下の通り導出される。

- もし chromaFlag が 0 に等しい、または ChromaArrayType が 3 に等しいならば、filteringModeFlag は 0 に等しく設定される。
- それ以外(chromaFlag が 1 に等しく、かつ ChromaArrayType が 3 に等しくない)、filteringModeFlag は 1 に等しく設定される。

配列 predArray は以下の規定により導出される。

- もし frameBasedResamplingFlag が 1 に等しい、または fieldMbFlag が 1 に等しいならば、次の順序付けられたステップが規定される。
  1. G.8.6.2.2 小小小節において規定される INTRA 再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、refLayerPicSamples、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は変数 refArrayW、refArrayH、参照レイヤサンプル値の配列 refSampleArray、および変数 xOffset と yOffset である。
  2. G.8.6.2.3 小小小節において規定される Intra\_Base 予測に対する補間処理が、filteringModeFlag、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、0 に等しい fldPrdInFrmMbFlag、0 に等しい yBorder、refArrayW、refArrayH、refSampleArray、xOffset、および yOffset を入力として起動され、出力は Intra\_Base 予測サンプルの(mbW)×(mbH)配列 mbPred である。

- それ以外、もし topAndBotResamplingFlag が 0 に等しいならば、以下の順序のステップが規定される。
  1. G.8.6.2.2 小小小節において規定される INTRA 再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、refLayerPicSamples、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は変数 refArrayW、refArrayH、参照レイヤサンプル値の配列 refSampleArray、および変数 xOffset と yOffset である。
  2. 変数 yBorder は(2-chromaFlag)に等しく設定される。
  3. G.8.6.2.3 小小小節において規定される Intra\_Base 予測に対する補間処理が、filteringModeFlag、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、1 に等しい fldPrdInFrmMbFlag、yBorder、refArrayW、refArrayH、refSampleArray、xOffset、および yOffset を入力として起動され、出力はフィールド予測サンプルの(mbW)×(mbH/2+2\*yBorder)配列 fieldPredArray である。
  4. G.8.6.2.4 小小小節において規定される Intra\_Base 予測に対する垂直補間処理が、filteringModeFlag、chromaFlag、mbW、mbH、botFieldFlag、yBorder、および fieldPredArray を入力として起動され、出力は Intra\_Base 予測サンプルの(mbW)×(mbH)配列 mbPred である。
- それ以外(topAndBotResamplingFlag が 1 に等しい)、以下の順序のステップが規定される。
  1. G.8.6.2.2 小小小節において規定される INTRA 再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、0 に等しい botFieldFlag、refLayerPicSamples、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は変数 refArrayTopW、refArrayTopH、トップフィールド参照レイヤサンプル値の配列 refSampleArrayTop、および変数 xOffsetTop と yOffsetTop である。
  2. G.8.6.2.3 小小小節において規定される Intra\_Base 予測に対する補間処理が、filteringModeFlag、chromaFlag、mbW、mbH、fieldMbFlag、0 に等しい botFieldFlag、1 に等しい fldPrdInFrmMbFlag、0 に等しい yBorder、refArrayTopW、refArrayTopH、refSampleArrayTop、xOffsetTop、および yOffsetTop を入力として起動され、出力はトップフィールド予測サンプルの(mbW)×(mbH/2)配列 topFieldPredArray である。
  3. G.8.6.2.2 小小小節において規定される INTRA 再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、1 に等しい botFieldFlag、refLayerPicSamples、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は変数 refArrayBotW、refArrayBotH、ボトムフィールド参照レイヤサンプル値の配列 refSampleArrayBot、および変数 xOffsetBot と yOffsetBot である。
  4. G.8.6.2.3 小小小節において規定される Intra\_Base 予測に対する補間処理が、filteringModeFlag、chromaFlag、mbW、mbH、fieldMbFlag、1 に等しい botFieldFlag、1 に等しい fldPrdInFrmMbFlag、0 に等しい yBorder、refArrayBotW、refArrayBotH、refSampleArrayBot、xOffsetBot、および yOffsetBot を入力として起動され、出力はボトムフィールド予測サンプルの(mbW)×(mbH/2)配列 botFieldPredArray である。
  5. Intra\_Base 予測サンプルの配列 mbPred の各サンプル predArray[x,y]、x=0..(mbW-1)、y=0..(mbH-1) は次式により導出される。

$$\text{mbPred}[x,y]=((y\%2)\neq 0 \text{ ? } \text{topFieldPredArray}[x,y>>1] \text{ : } \text{botFieldPredArray}[x,y>>1]) \quad (\text{G-276})$$

### G.8.6.2.2 INTRA再サンプリング前の参照レイヤサンプル配列構築処理

本処理への入力は以下の通りである。

- 再サンプリング処理の対象が輝度あるいは色差成分のいずれであるかを規定する変数 `chromaFlag`
- 対象の色成分のためのマクロブロックの、幅と高さをそれぞれ規定する2つの変数 `mbW` と `mbH`
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- (`RefLayerFrameMbsOnlyFlag` が 0 に等しいか、または `frame_mbs_only_flag` が 0 に等しい時)再サンプリング処理の対象がトップまたはボトムフィールドのいずれであるかを規定する変数 `botFieldFlag`
- `chromaFlag` が 0 に等しい時には参照レイヤ表現に対する構築された INTRA 輝度サンプル値を含む  $(\text{RefLayerPicWidthInSamples}_L) \times (\text{RefLayerPicHeightInSamples}_L)$  配列であり、または `chromaFlag` が 1 に等しい時には参照レイヤ表現に対する構築された INTRA 色差サンプル値を含む  $(\text{RefLayerPicWidthInSamples}_C) \times (\text{RefLayerPicHeightInSamples}_C)$  配列である、1つの配列 `refLayerPicSamples`
- 参照レイヤ表現のマクロブロックに対するスライス識別を規定する `RefLayerPicSizeInMbs` 個の要素を持つ1次元配列 `refLayerSliceIdx`
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックで、どのマクロブロックがフレームマクロブロックであるかを規定する `RefLayerPicSizeInMbs` 個の要素を持つ1次元配列 `refLayerFieldMbFlag`
- 参照レイヤ表現のマクロブロックに対するマクロブロックタイプを規定する `RefLayerPicSizeInMbs` 個の要素を持つ1次元配列 `refLayerMbType`

本処理の出力は以下の通りである。

- 参照レイヤサンプル値の構築配列の、幅と高さをそれぞれ規定する2つの変数 `refArrayW` と `refArrayH`
- 参照レイヤサンプル値の  $(\text{refArrayW}) \times (\text{refArrayH})$  配列 `refSampleArray`
- 配列 `refSampleArray` のサンプル `refSampleArray[0,0]` に対応した参照レイヤサンプル位置の、それぞれ `x` と `y` 座標を規定する2つの変数 `xOffset` と `yOffset`

変数 `refW`、`refH`、`refMbW`、`refMbH`、`xOffset`、`yOffset`、`refArrayW`、`refArrayH`、`xMin`、`yMin`、`xMax`、`yMax`、`yRefScale`、および `yRefAdd` は次の順序付けられたステップにより規定される。

1. G.6.3 小節にて規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、`chromaFlag`、サンプル位置(0,0)、`fieldMbFlag`、および `botFieldFlag` を入力として起動され、出力は16分の1サンプル単位のサンプル位置(`xRefMin16`,`yRefMin16`)である。
2. G.6.3 小節にて規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、`chromaFlag`、サンプル位置(`mbW-1`,`mbH-1`)、`fieldMbFlag`、および `botFieldFlag` を入力として起動され、出力16分の1サンプル単位のサンプル位置(`xRefMax16`,`yRefMax16`)である。
3. `Z` を、`chromaFlag` が 0 に等しい時は `L` に置換し、`chromaFlag` が 1 に等しい時は `C` に置換することにより、変数 `refW`、`refH`、`refMbW`、および `refMbH` は次式より導出される。

$$\text{refW}=\text{RefLayerPicWidthInSamples}_z \quad (\text{G-277})$$

$$\text{refH}=\text{RefLayerPicHeightInSamples}_z \quad (\text{G-278})$$

$$\text{refMbW}=(\text{chromaFlag}==0) ? 16 : \text{RefLayerMbWidthC} \quad (\text{G-279})$$

$$\text{refMbH}=(\text{chromaFlag}==0) ? 16 : \text{RefLayerMbHeightC} \quad (\text{G-280})$$

4. 変数 xOffset、yOffset、refArrayW、および refArrayH は次式より導出される。

$$\text{xOffset}=(\text{xRefMin}16>>8)<<4-(\text{refMbW}>>1) \quad (\text{G-281})$$

$$\text{yOffset}=(\text{yRefMin}16>>8)<<4-(\text{refMbH}>>1) \quad (\text{G-282})$$

$$\text{refArrayW}(((\text{xRefMax}16+15)>>8)<<4)+(\text{refMbW}>>1)-\text{xOffset}+16 \quad (\text{G-283})$$

$$\text{refArrayH}(((\text{yRefMax}16+15)>>8)<<4)+(\text{refMbH}>>1)-\text{yOffset}+16 \quad (\text{G-284})$$

記1 導出された配列サイズは、G.8.6.2.3 小小小節において規定される Intra\_Base 予測に対する補間処理に要求される配列サイズより大きいかもしれない。

5. 変数 xMin、yMin、xMax、および yMax は次式より導出される。

$$\text{xMin}=(\text{xRefMin}16>>4)-\text{xOffset} \quad (\text{G-285})$$

$$\text{yMin}=(\text{yRefMin}16>>4)-\text{yOffset} \quad (\text{G-286})$$

$$\text{xMax}((\text{xRefMax}16+15)>>4)-\text{xOffset} \quad (\text{G-287})$$

$$\text{yMax}((\text{yRefMax}16+15)>>4)-\text{yOffset} \quad (\text{G-288})$$

6. 変数 yRefScale と yRefAdd は次式の通り導出される。

- もし RefLayerFrameMbsOnlyFlag が 1 に等しい、あるいは RefLayerFieldMbFlag が 1 に等しいならば、yRefScale は 1 に等しく設定され、また yRefAdd は 0 に等しく設定される。
- それ以外(RefLayerFrameMbsOnlyFlag が 0 に等しく、かつ RefLayerFieldMbFlag が 0 に等しい)、yRefScale は 2 に等しく設定され、また yRefAdd は botFieldFlag に等しく設定される。

変数 refSliceIdcMb は“利用不可”としてマークされる。

各サンプル refSampleArray[x,y]、 $x=0..(\text{refArrayW}-1)$ 、 $y=0..(\text{refArrayH}-1)$ は次の順序付けられたステップにより規定される。

1. 参照レイヤサンプル位置(xRef,yRef)は次式より導出される。

$$\text{xRef}=\text{max}(0,\text{min}(\text{refW}-1,\text{x}+\text{xOffset})) \quad (\text{G-289})$$

$$\text{yRef}=\text{yRefScale}*\text{max}(0,\text{min}(\text{refH}/\text{yRefScale}-1,\text{y}+\text{yOffset}))+\text{yRefAdd} \quad (\text{G-290})$$

2. G.8.6.2.2.1 小小小節において規定される参照レイヤスライスおよび INTRA マクロブロック識別の導出処理が、参照レイヤサンプル位置(xRef,yRef)、refMbW、refMbH、refLayerSliceIdc、refLayerFieldMbFlag、および refLayerMbType を入力として起動され、出力は参照レイヤスライス識別 refSliceIdc と変数 refIntraMbFlag である。

3. constrained\_intra\_resampling\_flag が 1 に等しく、refIntraMbFlag が 1 に等しく、x が xMin より大きいあるいは等しく、x が xMax より小さいあるいは等しく、y が yMin より大きいあるいは等しく、かつ y が yMax より小さいあるいは等しい時、以下が適用される。

- もし変数 refSliceIdcMb が“利用不可”としてマークされるならば、refSliceIdcMb は“利用可能”

としてマークされ、またその値は refSliceIdc に等しく設定される。

- それ以外(refSliceIdcMb が“利用可能”としてマークされる)、ビットストリームは、refSliceIdcMb に等しくない参照レイヤスライス識別 refSliceIdc を結果として生じるデータを含んではならない。

記2 この制約は、それが参照レイヤ表現の中の1つより多いスライスのINTRA符号化マクロブロックをカバーし、かつconstrained\_intra\_resampling\_flagが1に等しい時、マクロブロックは、1に等しいbase\_mode\_flagを用いて符号化されることができないことを規定する。

4. refIntraMbFlag、constrained\_intra\_resampling\_flag、および refSliceIdc に依存して以下が適用される。

- もし以下の条件のどれかが真であるならば、サンプル refSampleArray[x,y]は “Intra\_Base 予測に利用不可” とマークされ、その値は0 に等しく設定される。
  - refIntraMbFlag は0 に等しい
  - constrained\_intra\_resampling\_flag は1 に等しく、かつ refSliceIdc は refSliceIdcMb に等しくない
- それ以外、サンプル refSampleArray[x,y]は “Intra\_Base 予測に利用可能” とマークされ、その値は次式より導出される。

$$\text{refSampleArray}[x,y]=\text{refLayerPicSamples}[x\text{Ref},y\text{Ref}] \quad (\text{G-291})$$

G.8.6.2.2.2 小節にて規定される INTRA 再サンプリング前の利用不可サンプル値に対する構築処理が、refMbW、refMbH、refArrayW、refArrayH、refSampleArray、xOffset、および yOffset を入力として起動され、出力はサンプル配列 refSampleArray の修正版である。

#### G.8.6.2.2.1 参照レイヤスライスおよびINTRAマクロブロック識別の導出処理

本処理への入力は以下の通りである。

- 参照レイヤピクチャの、対象の色成分の左上サンプル相対の参照レイヤサンプル位置(xRef,yRef)
- 対象の色成分における参照レイヤマクロブロックの、各々、幅と高さを規定する2つの変数 refMbW と refMbH
- 参照レイヤ表現のマクロブロックのスライス識別を規定する、RefLayerPicSizeInMbs 個の要素を持つ 1次元配列 refLayerSliceIdc
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックでどのマクロブロックがフレームマクロブロックであるかを規定する、RefLayerPicSizeInMbs 個の要素を持つ 1次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックのマクロブロックタイプを規定する、RefLayerPicSizeInMbs 個の要素を持つ 1次元配列 refLayerMbType

本処理からの出力は以下の通りである。

- 入力参照レイヤサンプル位置をカバーするスライスに対する参照レイヤスライス識別 refSliceIdc
- 入力参照レイヤサンプル位置をカバーする参照レイヤマクロブロックが INTRA 符号化であるかどうか

を規定する変数 refIntraMbFlag

参照レイヤマクロブロックアドレス refMbAddr は以下のように導出される。

- もし RefLayerMbaffFrameFlag が 0 に等しいならば、変数 refMbAddr は次で導出される。

$$\text{refMbAddr}=(y\text{Ref}/\text{refMbH})*\text{RefLayerPicWidthInMbs}+(x\text{Ref}/\text{refMbW}) \quad (\text{G-292})$$

- それ以外 (RefLayerMbaffFrameFlag が 1 に等しい)、変数 refMbAddr は以下に規定される順序付けられたステップで導出される。

1. 変数 refMbAddrTop は次で導出される。

$$\begin{aligned} \text{refMbAddrTop} &= 2*((y\text{Ref}/(2*\text{refMbH}))*\text{RefLayerPicWidthInMbs} \\ &\quad +(x\text{Ref}/\text{refMbW})) \end{aligned} \quad (\text{G-293})$$

2. refLayerFieldMbFlag[refMbAddrTop] に依存して、変数 refMbAddr は以下の通り導出される。

- もし refLayerFieldMbFlag[refMbAddrTop] が 0 に等しいならば、変数 refMbAddr は次で導出される。

$$\text{refMbAddr}=\text{refMbAddrTop}+(y\text{Ref}\%(2*\text{refMbH}))/\text{refMbH} \quad (\text{G-294})$$

- それ以外 (refLayerFieldMbFlag[refMbAddrTop] が 1 に等しい)、変数 refMbAddr は次で導出される。

$$\text{refMbAddr}=\text{refMbAddrTop}+(y\text{Ref}\%2) \quad (\text{G-295})$$

参照レイヤスライス識別 refSliceIdc は、refLayerSliceIdc[refMbAddr] に等しく設定される。

refLayerMbType[refMbAddr] に依存して、変数 refIntraMbFlag は以下の通り導出される。

- もし変数 refLayerMbType[refMbAddr] が I\_4x4、I\_8x8、I\_16x16、I\_PCM、または I\_BL に等しいならば、refIntraMbFlag は 1 に等しく設定される。
- それ以外 (refLayerMbType[refMbAddr] が I\_4x4、I\_8x8、I\_16x16、I\_PCM、または I\_BL に等しくない)、refIntraMbFlag は 0 に等しく設定される。

#### G.8.6.2.2.2 INTRA再サンプリング前の利用不可サンプル値の構築処理

本処理への入力には以下の通りである。

- 対象の色成分における参照レイヤマクロブロックの、各々、幅と高さを規定する 2 つの変数 refMbW と refMbH
- 参照レイヤサンプル値配列の、各々、幅と高さを規定する 2 つの変数 refArrayW と refArrayH
- 参照レイヤサンプル値の (refArrayW) × (refArrayH) 配列 refSampleArray
- 配列 refSampleArray のサンプル refSampleArray[0,0] に対応する参照レイヤサンプル位置の、各々、x および y 座標を規定する 2 つの変数 xOffset と yOffset

本処理からの出力は、配列 refSampleArray の修正版である。

$x=(\text{refMbW}/2)..(\text{refArrayW}-\text{refMbW}/2-1)$  および  $y=(\text{refMbH}/2)..(\text{refArrayH}-\text{refMbH}/2-1)$  を用いる各サンプル refSampleArray[x,y] のうち、“Intra\_Base 予測に利用不可” とマークされたものについて、以下の順序付けら

れたステップが規定される。

1. サンプル位置差分(xD,yD)と変数yAは以下の通り導出される。

$$xR=(x+xOffset)\%refMbW \quad (G-296)$$

$$yR=(y+yOffset)\%refMbH \quad (G-297)$$

$$xD=((xR\geq refMbW/2) ? (xR-refMbW) : (xR+1)) \quad (G-298)$$

$$yD=((yR\geq refMbH/2) ? (yR-refMbH) : (yR+1)) \quad (G-299)$$

$$yA=yD-9*Sign(yD) \quad (G-300)$$

2. 以下のいずれかの条件が真である場合、yDはyAに等しく設定される。

- サンプル refSampleArray[x,y-yD]が “Intra\_Base 予測に利用不可” とマークされており、サンプル refSampleArray[x,y-yA]が “Intra\_Base 予測に利用可能” とマークされている
- refSampleArray[x-xD,y]、refSampleArray[x,y-yD]、refSampleArray[x-xD,y-yD]の全てのサンプルが “Intra\_Base 予測に利用不可” とマークされている

3. サンプル値refSampleArray[x,y]は以下の通り導出される：

- もしサンプル refSampleArray[x-xD,y]とサンプル refSampleArray[x,y-yD]が “Intra\_Base 予測に利用可能” とマークされているならば、以下の順序付けられたステップが規定される。
  - a. 変数 cornerSampleAvailableFlag が以下の通り導出される。
    - サンプル refSampleArray[x-xD,y-yD]が “Intra\_Base 予測に利用可能” とマークされていれば、変数 cornerSampleAvailableFlag は 1 に等しく設定される。
    - それ以外 (サンプル refSampleArray[x-xD,y-yD]が “Intra\_Base 予測に利用不可” とマークされている)、変数 cornerSampleAvailable は 0 に等しく設定される。
  - b. G.8.6.2.2.2.1 小小小小節に規定される利用不可サンプルの対角構築処理が refArrayW、refArrayH、refSampleArray、サンプル位置差分(xD,yD)、サンプル位置(x,y)、および変数 cornerSampleAvailableFlag を入力として起動され、出力はサンプル位置(x,y)の修正されたサンプル値を持つサンプル配列 refSampleArray である。
- それ以外 (サンプル refSampleArray[x-xD,y]かサンプル refSampleArray[x,y-yD]が “Intra\_Base 予測に利用不可” とマークされている)、以下が適用される：
  - もしサンプル refSampleArray[x-xD,y]が “Intra\_Base 予測に利用可能” とマークされているならば、サンプル値 refSampleArray[x,y]は refSampleArray[x-xD,y]に等しく設定される。
  - それ以外、もしサンプル refSampleArray[x,y-yD]が “Intra\_Base 予測に利用可能” とマークされているならば、サンプル値 refSampleArray[x,y]は refSampleArray[x,y-yD]に等しく設定される。
  - それ以外(サンプル refSampleArray[x-xD,y]とサンプル refSampleArray[x,y-yD]が “Intra\_Base 予測に利用不可” とマークされている)、サンプル値 refSampleArray[x,y]は

refSampleArray[x-xD,y-yD]に等しく設定される。

x=0..(refArrayW-1)および y=0..(refArrayH-1)を用いる全てのサンプル refSampleArray[x,y]は “Intra\_Base 予測に利用可能”とマークされる。

#### G.8.6.2.2.2.1 利用不可サンプル値の対角構築処理

本処理への入力は以下の通りである。

- 参照レイヤサンプル値配列の、各々、幅と高さを規定する2つの変数 refArrayW と refArrayH
- 参照レイヤサンプル値の(refArrayW)×(refArrayH)配列 p
- サンプル位置の差分(xD,yD)
- 参照レイヤサンプル配列 refSampleArray 中のサンプル位置(x,y)
- 変数 cornerSampleAvailableFlag

本処理の出力は、サンプル位置(x,y)の修正されたサンプル値を持つサンプル配列 p である。

変数 diffHorVer と sgnXY が以下の通り導出される。

$$\text{diffHorVer}=\text{Abs}(xD)-\text{Abs}(yD) \quad (\text{G-301})$$

$$\text{sgnXY} = \text{Sign}(xD*yD) \quad (\text{G-302})$$

変数 cornerSampleAvailableFlag が 1 に等しい時、以下が適用される。

1. 変数cornerSampleはp[x-xD,y-yD]に等しく設定される。
2. サンプル値p[x-xD,y-yD]は以下のように修正される。

$$p[x-xD,y-yD]=(p[x-xD,y]+p[x,y-yD]+1)>>1 \quad (\text{G-324a})$$

サンプル値 p[x,y]は以下の通り導出される。

- もし変数 diffHorVer が 0 より大きいならば、サンプル位置(xC,yC)は(x-sgnXY\*yD,y-yD)に等しく設定され、サンプル値 p[x,y]は以下のように導出される。

$$p[x,y]=(p[xC-1,yC]+2*p[xC,yC]+p[xC+1,yC]+2)>>2 \quad (\text{G-303})$$

- それ以外、もし変数 diffHorVer が 0 より小さいならば、サンプル位置(xC,yC)は(x-xD,y-sgnXY\*xD)に等しく設定され、サンプル値 p[x,y]は以下のように導出される。

$$p[x,y]=(p[xC,yC-1]+2*p[xC,yC]+p[xC,yC+1]+2)>>2 \quad (\text{G-304})$$

- それ以外(変数 diffVerHor が 0 に等しい)、サンプル値 p[x,y]は以下のように導出される。

$$p[x,y]=(p[x,y-yD]+2*p[x-xD,y-yD]+p[x-xD,y]+2)>>2 \quad (\text{G-305})$$

変数 cornerSampleAvailableFlag が 1 に等しい時、サンプル値 p[x-xD,y-yD]は cornerSample に等しく設定される。

### G.8.6.2.3 Intra\_Base予測のための補間処理

本処理への入力は以下の通りである。

- 補間方法を規定する変数 `filteringModeFlag`
- 再サンプリング処理の対象が輝度または色差成分のいずれであるかを規定する変数 `chromaFlag`
- 対象の色成分におけるマクロブロックの、各々、幅と高さを規定する2つの変数 `mbW` と `mbH`
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- (`RefLayerFrameMbsOnlyFlag` が 0 に等しいか、または `frame_mbs_only_flag` が 0 に等しい時) 再サンプリング処理の対象がトップまたはボトムフィールドのいずれであるかを規定する変数 `botFieldFlag`
- フレームマクロブロックに対してフィールド予測が適用されるかを規定する変数 `fldPrdInFrmMbFlag`
- 出力サンプル配列 `predSamples` の垂直境界を規定する変数 `yBorder`
- 参照レイヤサンプル値配列の、各々、幅と高さを規定する2つの変数 `refArrayW` と `refArrayH`
- 参照レイヤサンプル値の(`refArrayW`)×(`refArrayH`)配列 `refSampleArray`
- 配列 `refSampleArray` のサンプル `refSampleArray[0,0]` に対応する参照レイヤサンプル位置の、各々、x および y 座標を規定する2つの変数 `xOffset` と `yOffset`

本処理の出力は、補間サンプル値の(`mbW`)×(`mbH`/(1+`fldPrdInFrmMbFlag`)+2\*`yBorder`)配列 `predArray` である。

付表G-8/JT-H264は、Intra\_Base予測での再サンプリングのための輝度補間フィルタ `eF` のフィルタ係数 `eF[p,x]`、`p=0..15`、`x=0..3`を規定する。

付表G-8/JT-H264 Intra\_Base予測での再サンプリングのための16フェーズの輝度補間フィルタ

(ITU-T H.264)

フェーズ P	補間フィルタ係数			
	eF[ p, 0 ]	eF[ p, 1 ]	eF[ p, 2 ]	eF[ p, 3 ]
0	0	32	0	0
1	-1	32	2	-1
2	-2	31	4	-1
3	-3	30	6	-1
4	-3	28	8	-1
5	-4	26	11	-1
6	-4	24	14	-2

7	-3	22	16	-3
8	-3	19	19	-3
9	-3	16	22	-3
10	-2	14	24	-4
11	-1	11	26	-4
12	-1	8	28	-3
13	-1	6	30	-3
14	-1	4	31	-2
15	-1	2	32	-1

tempArray をサンプルの(refArrayW)×(mbH/(1+fldPrdInFrmMbFlag)+2\*yBorder)配列とする。各サンプル tempArray[x,y]、x=0..(refArrayW-1)、y=0..(mbH/(1+fldPrdInFrmMbFlag)+2\*yBorder-1)は、以下の順序付けられたステップで規定されるように導出される。

1. 変数yPは以下で導出される。

$$yP=y*(1+fldPrdInFrmMbFlag)-yBorder+botFieldFlag \quad (G-306)$$

2. G.6.3小節に規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、chromaFlag、サンプル位置(0,yP)、fieldMbFlag、およびbotFieldFlagを入力として起動され、出力は16分の1サンプルを単位とするサンプル位置(xRef16,yRef16)である。

記1 このG.6.3小節の処理の起動においては、サンプル位置の垂直成分yRef16のみ導出される必要がある。

3. 変数yRefとyPhaseは以下で導出される。

$$yRef=(yRef16>>4)-yOffset \quad (G-307)$$

$$yPhase=(yRef16+16)\%16 \quad (G-308)$$

4. filteringModeFlagに依存して、サンプル値tempArray[x,y]は以下の通り導出される。

- もし filteringModeFlag が 0 に等しいならば、サンプル値 tempArray[x,y]は以下で導出される。

$$\begin{aligned} \text{tempArray}[x,y]= & eF[yPhase,0]*\text{refSampleArray}[x,yRef-1]+ \\ & eF[yPhase,1]*\text{refSampleArray}[x,yRef]+ \\ & eF[yPhase,2]*\text{refSampleArray}[x,yRef+1]+ \\ & eF[yPhase,3]*\text{refSampleArray}[x,yRef+2] \end{aligned} \quad (G-309)$$

- それ以外(filteringModeFlag が 1 に等しい)、サンプル値 tempArray[x,y]は以下で導出される。

$$\begin{aligned} \text{tempArray}[x,y] = & (16-y\text{Phase}) * \text{refSampleArray}[x,y\text{Ref}] + \\ & y\text{Phase} * \text{refSampleArray}[x,y\text{Ref}+1] \end{aligned} \quad (\text{G-310})$$

各サンプル  $\text{predArray}[x,y]$ 、 $x=0..(\text{mbW}-1)$ 、 $y=0..(\text{mbH}/(1+\text{fldPrdInFrmMbFlag})+2*y\text{Border}-1)$ は、以下に規定される順序付けられたステップで導出される。

1. G.6.3 小節で規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、 $\text{chromaFlag}$ 、サンプル位置 $(x,0)$ 、 $\text{fieldMbFlag}$ 、および  $\text{botFieldFlag}$  を入力として起動され、出力は 16 分の 1 サンプルを単位とするサンプル位置 $(x\text{Ref}16,y\text{Ref}16)$ である。

記2 このG.6.3小節の処理の起動においては、サンプル位置の水平成分 $x\text{Ref}16$ のみ導出される必要がある。

2. 変数  $x\text{Ref}$  と  $x\text{Phase}$  は以下で導出される。

$$x\text{Ref} = (x\text{Ref}16 \gg 4) - x\text{Offset} \quad (\text{G-311})$$

$$x\text{Phase} = (x\text{Ref}16 + 16) \% 16 \quad (\text{G-312})$$

3.  $\text{filteringModeFlag}$  に依存して、また 0 に等しい  $\text{chromaFlag}$  に対し  $\text{Clip}1_Y$ 、1 に等しい  $\text{chromaFlag}$  に対し  $\text{Clip}1_C$  に置き換えられる  $\text{Clip}1$  を用いて、サンプル値  $\text{predArray}[x,y]$  は以下の通り導出される。

- もし  $\text{filteringModeFlag}$  が 0 に等しいならば、サンプル値  $\text{tempArray}[x,y]$  は以下の通り導出される。

$$\begin{aligned} \text{predArray}[x,y] = & \text{Clip}1((eF[x\text{Phase},0] * \text{tempArray}[x\text{Ref}-1,y] + \\ & eF[x\text{Phase},1] * \text{tempArray}[x\text{Ref},y] + \\ & eF[x\text{Phase},2] * \text{tempArray}[x\text{Ref}+1,y] + \\ & eF[x\text{Phase},3] * \text{tempArray}[x\text{Ref}+2,y] + 512) \gg 10) \end{aligned} \quad (\text{G-313})$$

- それ以外( $\text{filteringModeFlag}$  が 1 に等しい)、サンプル値  $\text{tempArray}[x,y]$  は以下の通り導出される。

$$\begin{aligned} \text{predArray}[x,y] = & ((16-x\text{Phase}) * \text{tempArray}[x\text{Ref},y] + \\ & x\text{Phase} * \text{tempArray}[x\text{Ref}+1,y] + 128) \gg 8 \end{aligned} \quad (\text{G-314})$$

#### G.8.6.2.4 Intra\_Base予測のための垂直補間処理

本処理への入力は以下の通りである。

- 補間方法を規定する変数  $\text{filteringModeFlag}$
- 再サンプリング処理の対象が輝度または色差成分のいずれであるかを規定する変数  $\text{chromaFlag}$
- 対象の色成分におけるマクロブロックの、各々、幅と高さを規定する 2 つの変数  $\text{mbW}$  と  $\text{mbH}$
- サンプル配列  $\text{fieldPredArray}$  がトップまたはボトムフィールドのいずれに対する補間サンプルを含むかを規定する変数  $\text{botFieldFlag}$
- サンプル配列  $\text{fieldPredArray}$  の垂直境界を規定する変数  $y\text{Border}$
- サンプル値の  $(\text{mbW}) \times (\text{mbH}/2 + 2 * y\text{Border})$  配列  $\text{fieldPredArray}$

本処理の出力は、補間サンプル値の  $(\text{mbW}) \times (\text{mbH})$  配列  $\text{predArray}$  である。

各サンプル  $\text{predArray}[x,y]$ 、 $x=0..(\text{mbW}-1)$ 、 $y=0..(\text{mbH}-1)$  は以下の通り導出される。

- もし(y%2)が botFieldFlag に等しいならば、サンプル値 predArray[x,y]は以下の通り導出される。

$$\text{predArray}[x,y]=\text{fieldPredArray}[x,(y>>1)+y\text{Border}] \quad (\text{G-315})$$

- それ以外 ((y%2)が botFieldFlag に等しくない)、以下の順序付けられたステップが規定される。

1. 変数yFldが以下で導出される。

$$y\text{Fld}=(y>>1)+y\text{Border}-\text{botFieldFlag} \quad (\text{G-316})$$

2. filteringModeFlagに依存して、また0に等しいchromaFlagに対しClip1<sub>y</sub>、1に等しいchromaFlagに対しClip1<sub>C</sub>に置き換えられるClip1を用いて、サンプル値predArray[x,y]が以下の通り導出される。

- もし filteringModeFlag が 0 に等しいならば、predArray[x,y]が以下で導出される。

$$\begin{aligned} \text{predArray}[x,y]=\text{Clip1}((19*(\text{fieldPredArray}[x,y\text{Fld}]+ \\ \text{fieldPredArray}[x,y\text{Fld}+1])- \\ 3*(\text{fieldPredArray}[x,y\text{Fld}-1]+ \\ \text{fieldPredArray}[x,y\text{Fld}+2])+16)>>5) \end{aligned} \quad (\text{G-317})$$

- それ以外(filteringModeFlag が 1 に等しい)、サンプル値 predArray[x,y]は以下で導出される。

$$\begin{aligned} \text{predArray}[x,y]=(\text{fieldPredArray}[x,y\text{Fld}]+ \\ \text{fieldPredArray}[x,y\text{Fld}+1]+1)>>1 \end{aligned} \quad (\text{G-318})$$

### G.8.6.3 残差サンプルのための再サンプリング処理

本処理への入力は以下の通りである。

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックであるか、またどのマクロブロックがフレームマクロブロックであるかを規定する、RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックの輝度変換タイプを規定する、RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerCTrafo
- 参照レイヤ表現の輝度サンプルの(RefLayerPicWidthInSamples<sub>C</sub>)×(RefLayerPicHeightInSamples<sub>C</sub>)配列 refLayerPicSamples<sub>L</sub>
- 輝度サンプルの(PicWidthInSamples<sub>L</sub>)×(PicHeightInSamples<sub>L</sub>)配列 picSamples<sub>L</sub>
- ChromaArrayType が 0 に等しくない時、参照レイヤ表現の色差サンプルの 2 つの (RefLayerPicWidthInSamples<sub>C</sub>)×(RefLayerPicHeightInSamples<sub>C</sub>) 配列 refLayerPicSamples<sub>Cb</sub> および refLayerPicSamples<sub>Cr</sub>
- ChromaArrayType が 0 に等しくない時、色差サンプルの 2 つの (PicWidthInSamples<sub>C</sub>)×(PicHeightInSamples<sub>C</sub>)配列 picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub>

本処理の出力は以下の通りである。

- 輝度サンプルの配列  $\text{picSamples}_L$  の修正版
- ChromaArrayType が 0 に等しくない時、色差サンプルの配列  $\text{picSamples}_{Cb}$  および  $\text{picSamples}_{Cr}$  の修正版

G.8.6.3.1 小小小節に規定されるマクロブロック色成分の残差サンプルのための再サンプリング処理が、0 に等しい chromaFlag、16 に等しい mbW、16 に等しい mbH、fieldMbFlag、refLayerPicSamples<sub>L</sub>、refLayerFieldMbFlag、および、refLayerCTrafo を入力として起動され、出力は現マクロブロックの輝度成分の残差予測サンプルの 16x16 配列 mbPred<sub>L</sub> である。

ChromaArrayType が 0 に等しくない時、Cb および Cr で置換される CX に対して、G.8.6.3.1 小小小節に規定されるマクロブロック色成分の残差サンプルのための再サンプリング処理が、1 に等しい chromaFlag、MbWidthC に等しい mbW、MbHeightC に等しい mbH、fieldMbFlag、refLayerPicSamples<sub>CX</sub>、refLayerFieldMbFlag、および、refLayerCTrafo を入力として起動され、出力は現マクロブロックの CX 成分の残差予測サンプルの (MbWidthC)×(MbHeightC)配列 mbPred<sub>CX</sub> である。

G.8.5.4.1 小小小節に規定されるピクチャサンプル配列構築処理が、fieldMbFlag、mbPred<sub>L</sub>、picSamples<sub>L</sub>、および、ChromaArrayType が 0 に等しくない時は mbPred<sub>Cb</sub>、mbPred<sub>Cr</sub>、picSamples<sub>Cb</sub>、および picSamples<sub>Cr</sub> を入力として起動され、出力は、picSamples<sub>L</sub> の修正版、および、ChromaArrayType が 0 に等しくない時は picSamples<sub>Cb</sub> および picSamples<sub>Cr</sub> の修正版である。

#### G.8.6.3.1 マクロブロック色成分の残差サンプルのための再サンプリング処理

本処理への入力は以下の通りである。

- 再サンプリング処理の対象が輝度または色差成分のいずれであるかを規定する変数 chromaFlag
- 対象の色成分におけるマクロブロックの、各々、幅と高さを規定する 2 つの変数 mbW と mbH
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 fieldMbFlag
- chromaFlag が 0 に等しい時、参照レイヤ表現に対する構築された残差輝度サンプル値を含む  $(\text{RefLayerPicWidthInSamples}_L) \times (\text{RefLayerPicHeightInSamples}_L)$  配列、あるいは chromaFlag が 1 に等しい時、参照レイヤ表現に対する構築された残差色差サンプル値を含む  $(\text{RefLayerPicWidthInSamples}_C) \times (\text{RefLayerPicHeightInSamples}_C)$  配列である、refLayerPicSamples
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックであるか、またどのマクロブロックがフレームマクロブロックであるかを規定する、RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerFieldMbFlag
- 参照レイヤ表現のマクロブロックの輝度変換タイプを規定する、RefLayerPicSizeInMbs 個の要素を持つ 1 次元配列 refLayerCTrafo

本処理からの出力は、残差予測サンプルの  $(\text{mbW}) \times (\text{mbH})$  配列 mbPred である。

変数 botFieldFlag は以下で導出される。

- もし RefLayerFrameMbsOnlyFlag が 0 に等しいならば、botFieldFlag は 0 に等しく設定される。
- それ以外、もし field\_pic\_flag が 1 に等しいならば、botFieldFlag は bottom\_field\_flag に等しく設定される。

- それ以外、もし RefLayerFieldPicFlag が 1 に等しいならば、botFieldFlag は RefLayerBottomFieldFlag に等しく設定される。
- それ以外、もし fieldMbFlag が 1 に等しいならば、botFieldFlag は(CurrMbAddr%2)に等しく設定される。
- それ以外、botFieldFlag は 0 に等しく設定される。

変数 frameBasedResamplingFlag は以下で導出される。

- もし以下の条件が全て真であるならば、frameBasedResamplingFlag は 1 に等しく設定される。
  - RefLayerFrameMbsOnlyFlag が 1 に等しい
  - frame\_mbs\_only\_flag が 1 に等しい
- それ以外、frameBasedResamplingFlag は 0 に等しく設定される。

変数 topAndBotResamplingFlag を以下の通り導出される。

- もし以下の条件が全て真であるならば、topAndBotResamplingFlag は 1 に等しく設定される。
  - RefLayerFrameMbsOnlyFlag が 0 に等しい
  - RefLayerFieldPicFlag が 0 に等しい
  - frame\_mbs\_only\_flag が 0 に等しい
  - fieldMbFlag が 0 に等しい
- それ以外、topAndBotResamplingFlag は 0 に等しく設定される。

配列 predArray は以下の通り導出される。

- もし frameBasedResamplingFlag が 1 に等しいか、または、fieldMbFlag が 1 に等しいならば、以下の順序付けられたステップが規定される。
  1. G.8.6.3.2 小小小節に規定される残差再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、0 に等しい yBorder、refLayerPicSamples、refLayerFieldMbFlag、および refLayerCTrafo を入力として起動され、出力は変数 refArrayW、refArrayH、参照レイヤサンプル値の配列 refSampleArray、参照レイヤ変換ブロック識別の配列 refTransBlkIdc、および変数 xOffset と yOffset である。
  2. G.8.6.3.3 小小小節に規定される残差予測のための補間処理が、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、0 に等しい fldPrdInFrmMbFlag、0 に等しい yBorder、refArrayW、refArrayH、refSampleArray、refTransBlkIdc、xOffset、および yOffset を入力として起動され、出力は残差予測サンプルの (mbW)×(mbH)配列 mbPred である。
- それ以外、もし topAndBotResamplingFlag が 0 に等しいならば、以下の順序付けられたステップが規定される。
  1. G.8.6.3.2 小小小節に規定される残差再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、1 に等しい yBorder、refLayerPicSamples、refLayerFieldMbFlag、および refLayerCTrafo を入力として起動され、出力は変数 refArrayW、refArrayH、

参照レイヤサンプル値の配列 refSampleArray、参照レイヤ変換ブロック識別の配列 refTransBlkIdc、および変数 xOffset と yOffset である。

2. G.8.6.3.3 小小小節に規定される残差予測のための補間処理が、chromaFlag、mbW、mbH、fieldMbFlag、botFieldFlag、1 に等しい fldPrdInFrmMbFlag、1 に等しい yBorder、refArrayW、refArrayH、refSampleArray、refTransBlkIdc、xOffset、および yOffset を入力として起動され、出力はフィールド予測サンプルの (mbW)×(mbH/2+2)配列 fieldPredArray である。
  3. G.8.6.3.4 小小小節に規定される残差予測のための垂直補間処理が、mbW、mbH、botFieldFlag、yBorder、および fieldPredArray を入力として起動され、出力は、残差予測サンプルの (mbW)×(mbH)配列 mbPred である。
- それ以外(topAndBotResamplingFlag が 1 に等しい)、以下の順序付けられたステップが規定される。
1. G.8.6.3.2 小小小節に規定される残差再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、0 に等しい botFieldFlag、0 に等しい yBorder、refLayerPicSamples、refLayerFieldMbFlag、および refLayerCTrafo を入力として起動され、出力は変数 refArrayTopW、refArrayTopH、参照レイヤサンプル値の配列 refSampleArrayTop、参照レイヤ変換ブロック識別の配列 refTransBlkIdcTop、および変数 xOffsetTop と yOffsetTop である。
  2. G.8.6.3.3 小小小節に規定される残差予測のための補間処理が、chromaFlag、mbW、mbH、fieldMbFlag、0 に等しい botFieldFlag、1 に等しい fldPrdInFrmMbFlag、0 に等しい yBorder、refArrayTopW、refArrayTopH、refSampleArrayTop、refTransBlkIdcTop、xOffsetTop、および yOffsetTop を入力として起動され、出力はトップフィールド予測サンプルの (mbW)×(mbH/2)配列 topFieldPredArray である。
  3. G.8.6.3.2 小小小節に規定される残差再サンプリング前の参照レイヤサンプル配列構築処理が、chromaFlag、mbW、mbH、fieldMbFlag、1 に等しい botFieldFlag、0 に等しい yBorder、refLayerPicSamples、refLayerFieldMbFlag、および refLayerCTrafo を入力として起動され、出力は変数 refArrayBotW、refArrayBotH、参照レイヤサンプル値の配列 refSampleArrayBot、参照レイヤ変換ブロック識別の配列 refTransBlkIdcBot、および変数 xOffsetBot と yOffsetBot である。
  4. G.8.6.3.3 小小小節に規定される残差予測のための補間処理が、chromaFlag、mbW、mbH、fieldMbFlag、1 に等しい botFieldFlag、1 に等しい fldPrdInFrmMbFlag、0 に等しい yBorder、refArrayBotW、refArrayBotH、refSampleArrayBot、refTransBlkIdcBot、xOffsetBot、および yOffsetBot を入力として起動され、出力はボトムフィールド予測サンプルの (mbW)×(mbH/2)配列 botFieldPredArray である。
  5. 残差予測サンプルの配列 mbPred の各サンプル predArray[x,y]、x=0..(mbW-1)、y=0..(mbH-1)が以下で導出される。

$$\begin{aligned} \text{mbPred}[x,y] &= (y\%2) == 0 ? \text{topFieldPredArray}[x,y >> 1] \\ & : \text{botFieldPredArray}[x,y >> 1] \end{aligned} \quad (\text{G-319})$$

#### G.8.6.3.2 残差再サンプリング前の参照レイヤサンプル配列構築処理

本処理への入力には以下の通りである。

- 再サンプリング処理の対象が輝度または色差成分のいずれであるかを規定する変数 chromaFlag
- 対象の色成分におけるマクロブロックの、各々、幅と高さを規定する 2 つの変数 mbW と mbH

- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- (`RefLayerFrameMbsOnlyFlag` が 0 に等しいか、または `frame_mbs_only_flag` が 0 に等しい時) 再サンプリング処理の対象がトップまたはボトムフィールドのいずれであるかを規定する変数 `botFieldFlag`
- 出力配列の垂直サイズを決定する垂直境界を規定する変数 `yBorder`
- `chromaFlag` が 0 に等しい時、参照レイヤ表現に対する構築された残差輝度サンプル値を含む  $(\text{RefLayerPicWidthInSamples}_L) \times (\text{RefLayerPicHeightInSamples}_L)$  配列、あるいは `chromaFlag` が 1 に等しい時、参照レイヤ表現に対する構築された残差色差サンプル値を含む  $(\text{RefLayerPicWidthInSamples}_C) \times (\text{RefLayerPicHeightInSamples}_C)$  配列である、`refLayerPicSamples`
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックでどのマクロブロックがフレームマクロブロックであるかを規定する、`RefLayerPicSizeInMbs` 個の要素を持つ 1 次元配列 `refLayerFieldMbFlag`
- 参照レイヤ表現のマクロブロックに対する輝度変換タイプを規定する、`RefLayerPicSizeInMbs` 個の要素を持つ 1 次元配列 `refLayerCTrafo`

本処理からの出力は以下の通りである。

- 参照レイヤサンプル値および参照レイヤ変換ブロック識別の構築された配列の、各々、幅と高さを規定する 2 つの変数 `refArrayW` と `refArrayH`
- 参照レイヤサンプル値の  $(\text{refArrayW}) \times (\text{refArrayH})$  配列 `refSampleArray`
- 参照レイヤ変換ブロック識別の  $(\text{refArrayW}) \times (\text{refArrayH})$  配列 `refTransBlkIdc`
- 配列 `refSampleArray` のサンプル `refSampleArray[0,0]` および配列 `refTransBlkIdc` の変換ブロック識別 `refTransBlkIdc[0,0]` に対応する参照レイヤサンプル位置の、各々、x および y 座標を規定する 2 つの変数 `xOffset` と `yOffset`

以下の順序付けられたステップにより、変数 `refW`、`refH`、`refMbW`、`refMbH`、`xOffset`、`yOffset`、`refArrayW`、`refArrayH`、`yRefScale`、および `yRefAdd` が導出される。

1. G.6.3 小節に規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、`chromaFlag`、サンプル位置  $(0, -yBorder)$ 、`fieldMbFlag`、および `botFieldFlag` を入力として起動され、出力は 16 分の 1 サンプルを単位とするサンプル位置  $(xRefMin16, yRefMin16)$  である。
2. G.6.3 小節に規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、`chromaFlag`、サンプル位置  $(mbW-1, mbH-1+yBorder)$ 、`fieldMbFlag`、および `botFieldFlag` を入力として起動され、出力は 16 分の 1 サンプルを単位とするサンプル位置  $(xRefMax16, yRefMax16)$  である。
3. Z を `chromaFlag` が 0 に等しい時 L に、また `chromaFlag` が 1 に等しい時 C に置き換えて、変数 `refW`、`refH`、`refMbW`、および `refMbH` は、以下の通り導出される。

$$\text{refW} = \text{RefLayerPicWidthInSamples}_Z \quad (\text{G-320})$$

$$\text{refH} = \text{RefLayerPicHeightInSamples}_Z \quad (\text{G-321})$$

$$\text{refMbW} = ((\text{chromaFlag} == 0) ? 16 : \text{RefLayerMbWidthC}) \quad (\text{G-322})$$

$$\text{refMbH} = ((\text{chromaFlag} == 0) ? 16 : \text{RefLayerMbHeightC}) \quad (\text{G-323})$$

4. 変数  $xOffset$ 、 $yOffset$ 、 $refArrayW$ 、および  $refArrayH$  は以下の通り導出される。

$$xOffset=(xRefMin16>>4) \quad (G-324)$$

$$yOffset=(yRefMin16>>4) \quad (G-325)$$

$$refArrayW=((xRefMax16+15)>>4)-xOffset+1 \quad (G-326)$$

$$refArrayH=((yRefMax16+15)>>4)-yOffset+1 \quad (G-327)$$

5. 変数  $yRefScale$  および  $yRefAdd$  は以下の通り導出される。

- もし  $RefLayerFrameMbsOnlyFlag$  が 1 に等しいか  $RefLayerFieldMbFlag$  が 1 に等しいならば、 $yRefScale$  は 1 に等しく設定され、 $yRefAdd$  は 0 に等しく設定される。
- それ以外( $RefLayerFrameMbsOnlyFlag$  が 0 に等しく、かつ  $RefLayerFieldMbFlag$  が 0 に等しい)、 $yRefScale$  は 2 に等しく設定され、 $yRefAdd$  は  $botFieldFlag$  に等しく設定される。

各サンプル  $refSampleArray[x,y]$  および各変換ブロック識別  $refTransBlkIdc[x,y]$ 、 $x=0..(refArrayW-1)$ 、 $y=0..(refArrayH-1)$  は以下の順序付けられたステップで規定される通り導出される。

1. 参照レイヤサンプル位置( $xRef,yRef$ )は以下で導出される。

$$xRef=\max(0,\min(refW-1,x+xOffset)) \quad (G-328)$$

$$yRef=yRefScale*\max(0,\min(refH/yRefScale-1,y+yOffset))+yRefAdd \quad (G-329)$$

2. サンプル  $refSampleArray[x,y]$  は以下で導出される。

$$refSampleArray[x,y]=refLayerPicSamples[xRef,yRef] \quad (G-330)$$

3. 変換ブロック識別  $refTransBlkIdc[x,y]$  は、G.8.6.3.2.1 小小小節に規定される参照レイヤ変換ブロック識別の導出処理を、参照レイヤサンプル位置( $xRef,yRef$ )、 $chromaFlag$ 、 $refMbW$ 、 $refMbH$ 、 $refLayerFieldMbFlag$ 、および  $refLayerCTrafo$  を入力として起動し、その出力を  $refTransBlkIdc[x,y]$  に割り当てることにより導出される。

#### G.8.6.3.2.1 参照レイヤ変換ブロック識別のための導出処理

本処理への入力は以下の通りである。

- 参照レイヤピクチャの、対象の色成分の左上サンプル相対の参照レイヤサンプル位置( $xRef,yRef$ )
- 再サンプリング処理の対象が輝度または色差成分のいずれであるかを規定する変数  $chromaFlag$
- 対象の色成分における参照レイヤマクロブロックの、各々、幅と高さを規定する2つの変数  $refMbW$  と  $refMbH$
- 参照レイヤ表現のどのマクロブロックがフィールドマクロブロックでどのマクロブロックがフレームマクロブロックであるかを規定する、 $RefLayerPicSizeInMbs$  個の要素を持つ1次元配列  $refLayerFieldMbFlag$
- 参照レイヤ表現のマクロブロックに対する輝度変換タイプを規定する、 $RefLayerPicSizeInMbs$  個の要素を持つ1次元配列  $refLayerCTrafo$

本処理からの出力は、位置( $xRef,yRef$ )でのサンプルを含む参照レイヤ変換ブロックに対する識別を規定す

る変数 refTransBlkIdc である。

参照レイヤマクロブロックアドレス refMbAddr と参照レイヤマクロブロック中の参照レイヤサンプル位置 (xM,yM)は以下の通り導出される。

- もし RefLayerMbaffFrameFlag が 0 に等しいならば、変数 refMbAddr とサンプル位置(xM,yM)は以下で導出される。

$$\text{refMbAddr}=(y\text{Ref}/\text{refMbH})*\text{RefLayerPicWidthInMbs}+(x\text{Ref}/\text{refMbW}) \quad (\text{G-331})$$

$$xM = x\text{Ref}\% \text{refMbW} \quad (\text{G-332})$$

$$yM = y\text{Ref}\% \text{refMbH} \quad (\text{G-333})$$

- それ以外(RefLayerMbaffFrameFlag が 1 に等しい)、変数 refMbAddr は、以下に規定される順序付けられたステップにより導出される。

1. 変数refMbAddrTopと水平サンプル位置xMは以下で導出される。

$$\text{refMbAddrTop}=2*((y\text{Ref}/(2*\text{refMbH}))*\text{RefLayerPicWidthInMbs}+(x\text{Ref}/\text{refMbW})) \quad (\text{G-334})$$

$$xM = x\text{Ref}\% \text{refMbW} \quad (\text{G-335})$$

2. refLayerFieldMbFlag[refMbAddrTop]に依存して、変数refMbAddrと垂直サンプル位置yMは以下の通り導出される。

- もし refLayerFieldMbFlag[refMbAddrTop]が 0 に等しいならば、変数 refMbAddr および yM は、以下で導出される。

$$\text{refMbAddr}=\text{refMbAddrTop}+(y\text{Ref}\%(2*\text{refMbH}))/\text{refMbH} \quad (\text{G-336})$$

$$yM = y\text{Ref}\% \text{refMbH} \quad (\text{G-337})$$

- それ以外(refLayerFieldMbFlag[refMbAddrTop]が 1 に等しい)、変数 refMbAddr および yM は、以下で導出される。

$$\text{refMbAddr}=\text{refMbAddrTop}+(y\text{Ref}\%2) \quad (\text{G-338})$$

$$yM = (y\text{Ref}\%(2*\text{refMbH}))>>1 \quad (\text{G-339})$$

chromaFlag、RefLayerChromaArrayType、および refLayerCTrafo[refMbAddr]に依存して、以下が適用される。

- もし(chromaFlag が 0 に等しいか、または RefLayerChromaArrayType が 3 に等しい)かつ refLayerCTrafo[refMbAddr]が T\_8x8 に等しいならば、変数 refTransBlkIdc は以下で導出される。

$$\text{refTransBlkIdc}=1+2*(4*\text{refMbAddr}+2*(yM/8)+(xM/8)) \quad (\text{G-340})$$

- それ以外((chromaFlag が 1 に等しく、かつ RefLayerChromaArrayType が 3 に等しくない)または refLayerCTrafo[refMbAddr]が T\_8x8 に等しくない)、変数 refTransBlkIdc は以下で導出される。

$$\text{refTransBlkIdc}=2*(16*\text{refMbAddr}+4*(yM/4)+(xM/4)) \quad (\text{G-341})$$

### G.8.6.3.3 残差予測のための補間処理

本処理への入力は以下の通りである。

- 再サンプリング処理の対象が輝度または色差成分のいずれであるかを規定する変数 `chromaFlag`
- 対象の色成分におけるマクロブロックの、各々、幅と高さを規定する2つの変数 `mbW` と `mbH`
- 現マクロブロックがフィールドまたはフレームマクロブロックのいずれであるかを規定する変数 `fieldMbFlag`
- (`RefLayerFrameMbsOnlyFlag` が 0 に等しいか、または `frame_mbs_only_flag` が 0 に等しい時) 再サンプリング処理の対象がトップまたはボトムフィールドのいずれであるかを規定する変数 `botFieldFlag`
- フレームマクロブロックに対してフィールド予測が適用されるかを規定する変数 `fldPrdInFrmMbFlag`
- 出力サンプル配列 `predSamples` の垂直境界を規定する変数 `yBorder`
- 参照レイヤサンプル値配列および変換ブロック識別配列の、各々、幅と高さを規定する2つの変数 `refArrayW` と `refArrayH`
- 参照レイヤサンプル値の(`refArrayW`)×(`refArrayH`)配列 `refSampleArray`
- 変換ブロック識別の(`refArrayW`)×(`refArrayH`)配列 `refTransBlkIdc`
- 配列 `refSampleArray` のサンプル `refSampleArray[0,0]`および配列 `refTransBlkIdc` の変換ブロック識別 `refTransBlkIdc[0,0]`に対応する参照レイヤサンプル位置の、各々、`x` および `y` 座標を規定する2つの変数 `xOffset` と `yOffset`

本処理の出力は、補間サンプル値の(`mbW`)×(`mbH`/(1+`fldPrdInFrmMbFlag`)+2\*`yBorder`)配列 `predArray` である。

各サンプル `predArray[x,y]`、`x=0..(mbW-1)`、`y=0..(mbH/(1+fldPrdInFrmMbFlag)+2*yBorder-1)`は、以下の順序付けられたステップで規定される通り導出される。

1. 変数`yP`は以下で導出される。

$$yP=y*(1+fldPrdInFrmMbFlag)-yBorder+botFieldFlag \quad (G-342)$$

2. G.6.3小節に規定される再サンプリングにおける参照レイヤサンプル位置の導出処理が、`chromaFlag`、サンプル位置(`x,yP`)、`fieldMbFlag`、および`botFieldFlag`を入力として起動され、出力は16分の1サンプルを単位とするサンプル位置(`xRef16`、`yRef16`)である。
3. 変数`xRef`、`yRef`、`xPhase`、および`yPhase`は以下で導出される。

$$xRef=(xRef16>>4)-xOffset \quad (G-343)$$

$$yRef=(yRef16>>4)-yOffset \quad (G-344)$$

$$xPhase=(xRef16+16)\%16 \quad (G-345)$$

$$yPhase=(yRef16+16)\%16 \quad (G-346)$$

4. `tempPred`を要素数2の1次元配列とする。各サンプル値`tempPred[dY]`、`dY=0..1`は以下で導出される。
  - もし `refTransBlkIdc[xRef,yRef+dY]`が `refTransBlkIdc[xRef+1,yRef+dY]`に等しいならば、サンプル値 `tempPred[dY]`は以下で導出される。

$$tempPred[dY]=(16-xPhase)*refSampleArray[xRef,yRef+dY]+xPhase*refSampleArray[xRef+1,yRef+dY] \quad (G-347)$$

- それ以外(refTransBlkIdc[xRef,yRef+dY]が refTransBlkIdc[xRef+1,yRef+dY]に等しくない)、サンプル値 tempPred[dY]は以下で導出される。

$$\begin{aligned} \text{tempPred}[dY] &= ((xPhase < 8) ? \text{refSampleArray}[xRef,yRef+dY] \\ &: \text{refSampleArray}[xRef+1,yRef+dY]) \end{aligned} \quad (\text{G-348})$$

5. サンプル値predArray[x,y]は以下のように導出される。

- もし refTransBlkIdc[xRef,yRef]が refTransBlkIdc[xRef,yRef+1]に等しいならば、サンプル値 predArray[x,y]は以下で導出される。

$$\text{predArray}[x,y] = ((16-yPhase) * \text{tempPred}[0] + yPhase * \text{tempPred}[1] + 128) \gg 8 \quad (\text{G-349})$$

- それ以外(refTransBlkIdc[xRef,yRef]が refTransBlkIdc[xRef,yRef+1]に等しくない)、サンプル値 predArray[x,y]は以下で導出される。

$$\text{predArray}[dY] = (((yPhase < 8) ? \text{tempPred}[0] : \text{tempPred}[1]) + 8) \gg 4 \quad (\text{G-350})$$

#### G.8.6.3.4 残差予測のための垂直補間処理

本処理への入力には以下の通りである。

- 対象の色成分におけるマクロブロックの、各々、幅と高さを規定する2つの変数 mbW と mbH
- サンプル配列 fieldPredArray がトップまたはボトムフィールドのいずれに対する補間サンプルを含むかを規定する変数 botFieldFlag
- サンプル配列 fieldPredArray の垂直境界を規定する変数 yBorder
- サンプル値の(mbW)×(mbH/2+2\*yBorder)配列 FieldPredArray

本処理の出力は、補間サンプル値の(mbW)×(mbH)配列 predArray である。

各サンプル predArray[x,y]、x=0..(mbW-1)、y=0..(mbH-1)は以下のように導出される。

- もし(y%2)が botFieldFlag に等しいならば、サンプル値 predArray[x,y]は以下で導出される。

$$\text{predArray}[x,y] = \text{fieldPredArray}[x,(y \gg 1) + yBorder] \quad (\text{G-351})$$

- それ以外((y%2)が botFieldFlag に等しくない)、サンプル値 predArray[x,y]は以下で導出される。

$$\begin{aligned} \text{predArray}[x,y] &= (\text{fieldPredArray}[x,(y \gg 1) + yBorder - \text{botFieldFlag}] + \\ &\text{fieldPredArray}[x,(y \gg 1) + yBorder - \text{botFieldFlag} + 1] + 1) \gg 1 \end{aligned} \quad (\text{G-352})$$

#### G.8.7 SVCデブロックフィルタ処理

G.8.7.1 小節は Intra\_Base 予測のためのデブロックフィルタ処理を規定する。

G.8.7.2 小節はターゲットスライスのためのデブロックフィルタ処理を規定する。

##### G.8.7.1 Intra\_Base予測のためのデブロックフィルタ処理

本処理への入力には以下の通りである。

- 変数 `dependencyIdRefLayer`
- 総称 `currentVars`

本処理からの出力は、`currentVars` の修正版である。

特に記述されない限り、本小小節で示される処理と本処理から起動される全ての子処理で参照される、シンタックス要素と導出された大文字変数は、(`dependencyIdRefLayer<<4`)に等しい `DQId` を持つレイヤ表現に対するシンタックス要素と導出された大文字変数である。

本小小節の中では、`currentVars` として総称的に参照される配列は、G.8.1.2.1 小小節に規定される、それらの名前で参照される。

G.8.7.3 小小節で規定されるデブロックフィルタで使用される量子化パラメータの導出処理が、`mbType`、`tQPY`、および `tCoeffLevel` を入力として起動され、出力は、デブロックフィルタ処理で使用される輝度量子化パラメータを規定するリスト `qpDBY`、および、`ChromaArrayType` が 0 で無い場合にはそれに加えてデブロックフィルタ処理で使用される色差量子化パラメータを規定するリスト `qpDBCb` と `qpDBCr` である。

`disableDeblockingFilterIdc`、`filterOffsetA`、および `filterOffsetB` を、それぞれ、(`dependency_id<<4`)に等しい `DQId` を持つレイヤ表現の `disable_inter_layer_deblocking_filter_idc`、`InterlayerFilterOffsetA`、および `InterlayerFilterOffsetB` に等しい値とする。

0..(`PicSizeInMbs-1`)の範囲をとる現マクロブロックアドレス `CurrMbAddr` に対し、以下が適用される。

- G.8.7.4 小小節で規定されるマクロブロックデブロックフィルタ処理が、`interLayerDeblockingFlag=1`、`disableDeblockingFilterIdc`、`filterOffsetA`、`filterOffsetB`、`sliceBoundariesOnlyFlag=0`、`currentVars`、`qpDBY` および、`ChromaArrayType` が 0 に等しくないときにはこれらに加えて `qpDBCb` と `qpDBCr` を入力として起動され、出力は `currentVars` の修正版である。

`disable_deblocking_filter_idc` が 3 または 6 であるときには、0..(`PicSizeInMbs-1`)の範囲をとる現マクロブロックアドレス `CurrMbAddr` に対し、以下が適用される。

- G.8.7.4 小小節で規定されるマクロブロックデブロックフィルタ処理が、`interLayerDeblockingFlag=1`、`disableDeblockingFilterIdc`、`filterOffsetA`、`filterOffsetB`、`sliceBoundariesOnlyFlag=1`、`currentVars`、`qpDBY` および、`ChromaArrayType` が 0 に等しくないときにはこれらに加えて `qpDBCb` と `qpDBCr` を入力として起動され、出力は `currentVars` の修正版である。

### G.8.7.2 ターゲット表現のためのデブロックフィルタ処理

本処理への入力は、総称 `currentVars` である。

本処理からの出力は、サンプル配列 `SL`、および、`ChromaArrayType` が 0 に等しくない場合には `SCb` および `SCr`、により表現された復号サンプルである。

特に記述されない限り、本小小節で示される処理と本処理から起動される全ての子処理で参照されるシンタックス要素と導出された大文字変数は、(`dependencyId<<4`)に等しい `DQId` を持つレイヤ表現に対するシンタックス要素と導出された大文字変数である。

本小小節の中では、`currentVars` として総称的に参照される配列は、G.8.1.2.1 小小節に規定される、それらの名前で参照される。

G.8.7.3 小小節に規定されるデブロックフィルタ処理に使用される量子化パラメータの導出処理が、mbType、tQP<sub>Y</sub>、および tCoeffLevel を入力として起動され、出力はデブロックフィルタ処理で使用される輝度量子化パラメータを示すリスト qpDB<sub>Y</sub>、および、ChromaArrayType が 0 に等しくない場合にはそれに加えてデブロックフィルタ処理で使用される色差量子化パラメータを示すリスト qpDB<sub>Cb</sub> と qpDB<sub>Cr</sub> である。

0..(PicSizeInMbs-1)の範囲をとる現マクロブロックアドレス CurrMbAddr に対し、以下が適用される。

- disableDeblockingFilterIdc、filterOffsetA、および filterOffsetB を、それぞれ、DQId が (sliceIdc[CurrMbAddr]&127)に等しい DQId および(sliceIdc[CurrMbAddr]>>7)に等しい first\_mb\_in\_slice を持つスライスの disable\_deblocking\_filter\_idc、FilterOffsetA、および FilterOffsetB、に等しい値とする。
- G.8.7.4 小小節に規定されるマクロブロックデブロックフィルタ処理が、interLayerDeblockingFlag=0、disableDeblockingFilterIdc、filterOffsetA、filterOffsetB、sliceBoundariesOnlyFlag=0、currentVars、qpDB<sub>Y</sub> および、ChromaArrayType が 0 に等しくないときにはこれらに加えて qpDB<sub>Cb</sub> と qpDB<sub>Cr</sub> を入力として起動され、出力は currentVars の修正版である。

disable\_deblocking\_filter\_idc が 3 または 6 に等しい時、0..(PicSizeInMbs-1)の範囲をとる現マクロブロックアドレス CurrMbAddr に対し、以下が適用される。

- disableDeblockingFilterIdc、filterOffsetA、および filterOffsetB を、それぞれ、(sliceIdc[CurrMbAddr]&127) に等しい DQId および (sliceIdc[CurrMbAddr]>>7) に等しい first\_mb\_in\_slice を持つスライスの disable\_deblocking\_filter\_idc、FilterOffsetA、および FilterOffsetB、に等しい値とする。
- G.8.7.4 小小節に規定されるマクロブロックデブロックフィルタ処理が、interLayerDeblockingFlag=0、disableDeblockingFilterIdc、filterOffsetA、filterOffsetB、sliceBoundariesOnlyFlag=1、currentVars、qpDB<sub>Y</sub> および、ChromaArrayType が 0 に等しくないときにはこれらに加えて、qpDB<sub>Cb</sub> と qpDB<sub>Cr</sub> を入力として起動され、出力は currentVars の修正版である。

### G.8.7.3 デブロックフィルタで使用される量子化パラメータの導出処理

本処理への入力は以下の通りである。

- 現在の、復号された、または一部復号された依存表現のマクロブロックのマクロブロックタイプを規定する、PicSizeInMbs 個の要素を持つ 1 次元配列 mbType
- 現在の、復号された、または一部復号された依存表現のマクロブロックの輝度量子化パラメータを規定する、PicSizeInMbs 個の要素を持つ 1 次元配列 tQP<sub>Y</sub>
- 現在の、復号された、または一部復号された依存表現のマクロブロックの変換係数レベル値を規定する、(PicSizeInMbs)×(256+2\*MbWidthC\*MbHeightC)配列 tCoeffLevel

本処理からの出力は以下の通りである。

- 現在の、復号された、または一部復号された依存表現のマクロブロックに対するデブロックフィルタ処理で使用される輝度量子化パラメータを規定する、PicSizeInMbs 個の要素を持つ 1 次元配列 qpDB<sub>Y</sub>
- ChromaArrayType が 0 でない時、現在の、復号された、または一部復号された依存表現のマクロブロックに対するデブロックフィルタ処理で使用される色差量子化パラメータを規定する、PicSizeInMbs 個の要素を持つ 2 つの 1 次元配列 qpDB<sub>Cb</sub> と qpDB<sub>Cr</sub>

マクロブロックアドレス  $mbAddr$  は  $0..(PicSizeInMbs-1)$  の範囲をとり、各  $mbAddr$  の値に対し、以下の順序付けられたステップが規定される。

1. 変数  $qpDB_Y[mbAddr]$  は以下のように導出される。
  - もし  $mbType[mbAddr]$  が  $I\_PCM$  に等しければ、 $qpDB_Y[mbAddr]$  は 0 に等しく設定される。
  - それ以外、もし  $tcoeff\_level\_prediction\_flag$  が 1 に等しく、 $mbAddr$  が 0 より大きく、 $mbType[mbAddr]$  が  $I\_16x16$  に等しくなく、 $i=0..(255+2*MbWidthC*MbHeightC)$  の  $tCoeffLevel[mbAddr][i]$  の全ての要素が 0 に等しいならば、 $qpDB_Y[mbAddr]$  は  $qpDB_Y[mbAddr-1]$  に等しく設定される。
  - それ以外、 $qpDB_Y[mbAddr]$  は  $tQP_Y[mbAddr]$  に等しく設定される。
2.  $ChromaArrayType$  が 0 に等しくない時、 $C$  を  $Cb$  および  $Cr$  に置き換えて、変数  $qpDB_C[mbAddr]$  は、8.5.8 小節に規定されるように  $QP_Y$  に対する  $qpDB_Y[mbAddr]$  の値に対応する  $QP_C$  の値に等しく設定される。この 8.5.8 小節の処理の起動の間、 $DQIdMax$  に等しい  $DQId$  を持つレイヤ表現のシンタックス要素  $chroma\_qp\_index\_offset$  および  $second\_chroma\_qp\_index\_offset$  が用いられる。

#### G.8.7.4 マクロブロックデブロックフィルタ処理

本処理への入力は以下の通りである。

- 変数  $interLayerDeblockingFlag$ 、 $disableDeblockingFilterIdc$ 、 $filterOffsetA$ 、 $filterOffsetB$ 、および  $sliceBoundariesOnlyFlag$
- 総称  $currentVars$
- 現在の、復号された、または一部復号された依存表現のマクロブロックのデブロックフィルタ処理で使用される輝度量量子化パラメータを規定する、 $PicSizeInMbs$  個の要素を持つ 1 次元配列  $qpDB_Y$
- $ChromaArrayType$  が 0 に等しくない時、現在の、復号された、または一部復号された依存表現のマクロブロックのデブロックフィルタ処理で使用される色差量子化パラメータを規定する、 $PicSizeInMbs$  個の要素を持つ 2 つの 1 次元配列  $qpDB_{Cb}$  と  $qpDB_{Cr}$

本処理からの出力は、 $currentVars$  の修正版である。

本小節の以降の部分では、総称  $currentVars$  の変数  $sliceIdc$ 、 $fieldMbFlag$ 、 $baseModeFlag$ 、 $mbType$ 、 $lumaTrafo$ 、 $subMbType$ 、 $ipred4x4$ 、 $ipred8x8$ 、 $ipred16x16$ 、 $ipredChroma$ 、 $refIdxL0$ 、 $refIdxL1$ 、 $predFlagL0$ 、 $predFlagL1$ 、 $mvL0$ 、 $mvL1$ 、 $mvCnt$ 、 $sTCoeff$ 、 $tCoeffLevel$ 、 $tQP_Y$ 、 $tQP_{Cb}$ 、 $tQP_{Cr}$ 、 $rS_L$ 、 $rS_{Cb}$ 、 $rS_{Cr}$ 、 $cS_L$ 、 $cS_{Cb}$ 、および  $cS_{Cr}$  は、それぞれ  $sliceIdc$ 、 $fieldMbFlag$ 、 $baseModeFlag$ 、 $mbType$ 、 $lumaTrafo$ 、 $subMbType$ 、 $ipred4x4$ 、 $ipred8x8$ 、 $ipred16x16$ 、 $ipredChroma$ 、 $refIdxL0$ 、 $refIdxL1$ 、 $predFlagL0$ 、 $predFlagL1$ 、 $mvL0$ 、 $mvL1$ 、 $mvCnt$ 、 $sTCoeff$ 、 $tCoeffLevel$ 、 $tQP_Y$ 、 $tQP_{Cb}$ 、 $tQP_{Cr}$ 、 $rS_L$ 、 $rS_{Cb}$ 、 $rS_{Cr}$ 、 $cS_L$ 、 $cS_{Cb}$ 、および  $cS_{Cr}$  として参照される。

6.4.10.1 小節に規定される隣接マクロブロックの導出処理が起動され、出力が  $mbAddrA$  および  $mbAddrB$  に割り当てられる。この 6.4.10.1 小節の処理の起動では、現マクロブロックは、 $fieldMbFlag[CurrMbAddr]$  が 1 に等しい時フィールドマクロブロックとして、 $fieldMbFlag[CurrMbAddr]$  が 0 に等しい時フレームマクロブロックとして扱われる。

記 マクロブロック  $mbAddrA$  および  $mbAddrB$  の利用可能状態は、本小節の中では使用されない。スライス境界は配列  $sliceIdc$  を用いて検出される。

変数 `filterLeftLumaMbEdgeFlag` は以下のように導出される。

- もし以下のいずれかの条件が真ならば、変数 `filterLeftLumaMbEdgeFlag` は 0 に等しく設定される。
  - `MbaffFrameFlag` が 0 に等しく、かつ `CurrMbAddr%PicWidthInMbs` が 0 に等しい
  - `MbaffFrameFlag` が 1 に等しく、かつ `(CurrMbAddr>>1)%PicWidthInMbs` が 0 に等しい
  - `disableDeblockingFilterIdc` が 1 に等しい
  - `disableDeblockingFilterIdc` が 2 または 5 に等しく、かつ `sliceIdc[mbAddrA]` が `sliceIdc[CurrMbAddr]` と異なる
  - `disableDeblockingFilterIdc` が 3 または 6 に等しく、`sliceBoundariesOnlyFlag` が 0 に等しく、かつ `sliceIdc[mbAddrA]` が `sliceIdc[CurrMbAddr]` と異なる
  - `disableDeblockingFilterIdc` が 3 または 6 に等しく、`sliceBoundariesOnlyFlag` が 1 に等しく、かつ `sliceIdc[mbAddrA]` が `sliceIdc[CurrMbAddr]` に等しい
  - `interLayerDeblockingFlag` が 1 に等しく、かつ `mbType[CurrMbAddr]` が INTER マクロブロック予測モードを規定する
  - `interLayerDeblockingFlag` が 1 に等しく、かつ `mbType[mbAddrA]` が INTER マクロブロック予測モードを規定する
- それ以外、変数 `filterLeftLumaMbEdgeFlag` は 1 に等しく設定される。

変数 `filterTopLumaMbEdgeFlag` は以下のように導出される。

- もし以下のいずれかの条件が真ならば、変数 `filterTopLumaMbEdgeFlag` は 0 に等しく設定される。
  - `MbaffFrameFlag` が 0 に等しく、かつ `CurrMbAddr` が `PicWidthInMbs` より小さい
  - `MbaffFrameFlag` が 1 に等しく、`(CurrMbAddr>>1)` が `PicWidthInMbs` より小さく、かつ `fieldMbFlag[CurrMbAddr]` が 1 に等しい
  - `MbaffFrameFlag` が 1 に等しく、`(CurrMbAddr>>1)` が `PicWidthInMbs` より小さく、`fieldMbFlag[CurrMbAddr]` が 0 に等しく、かつ `CurrMbAddr%2` が 0 に等しい
  - `disableDeblockingFilterIdc` が 1 に等しい
  - `disableDeblockingFilterIdc` が 2 または 5 に等しく、かつ `sliceIdc[mbAddrB]` が `sliceIdc[CurrMbAddr]` と異なる
  - `disableDeblockingFilterIdc` が 3 または 6 に等しく、`sliceBoundariesOnlyFlag` が 0 に等しく、かつ `sliceIdc[mbAddrB]` が `sliceIdc[CurrMbAddr]` と異なる
  - `disableDeblockingFilterIdc` が 3 または 6 に等しく、`sliceBoundariesOnlyFlag` が 1 に等しく、かつ `sliceIdc[mbAddrB]` が `sliceIdc[CurrMbAddr]` に等しい
  - `interLayerDeblockingFlag` が 1 に等しく、かつ `mbType[CurrMbAddr]` が INTER マクロブロック予測モードを規定する
  - `interLayerDeblockingFlag` が 1 に等しく、かつ `mbType[mbAddrB]` が INTER マクロブロック予測モードを規定する

ドを規定する

- それ以外、変数 `filterTopLumaMbEdgeFlag` に 1 に等しく設定される。

変数 `filterInternalLumaEdgesFlag` は以下のように導出される。

- もし以下のいずれかの条件が真ならば、`filterInternalLumaEdgesFlag` は 0 に等しく設定される。
  - `disableDeblockingFilterIdc` が 1 に等しい
  - `disableDeblockingFilterIdc` が 3 または 6 に等しく、かつ `sliceBoundariesOnlyFlag` が 1 に等しい
  - `interLayerDeblockingFlag` が 1 に等しく、かつ `mbType[CurrMbAddr]` が INTER マクロブロック予測モードを規定する
- それ以外、変数 `filterInternalLumaEdgesFlag` は 1 に等しく設定される。

変数 `filterLeftChromaMbEdgeFlag`、`filterTopChromaMbEdgeFlag`、および `filterInternalChromaEdgesFlag` は以下のように導出される。

- もし `disableDeblockingFilterIdc` が 3 より大きいならば、`filterLeftChromaMbEdgeFlag`、`filterTopChromaMbEdgeFlag`、および `filterInternalChromaEdgesFlag` は 0 に等しく設定される。
- それ以外 (`disableDeblockingFilterIdc` が 4 より小さい)、`filterLeftChromaMbEdgeFlag`、`filterTopChromaMbEdgeFlag`、および `filterInternalChromaEdgesFlag` は、それぞれ、`filterLeftLumaMbEdgeFlag`、`filterTopLumaMbEdgeFlag`、および `filterInternalLumaEdgesFlag` に等しく設定される。

変数 `fieldModeMbFlag` は以下のように導出される。

- もし以下のいずれかの条件が真ならば、`fieldModeMbFlag` は 1 に等しく設定される。
  - `field_pic_flag` が 1 に等しい
  - `MbaffFrameFlag` が 1 に等しく、かつ `fieldMbFlag[CurrMbAddr]` が 1 に等しい
- それ以外、`fieldModeMbFlag` は 0 に等しく設定される。

`filterLeftLumaMbEdgeFlag` が 1 に等しい場合、左部垂直輝度エッジのフィルタは以下のように規定される。

- G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された `chromaEdgeFlag`、1 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBv[mbAddrA]` に等しく設定された `qPp`、`qpDBv[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..15` として (0,k) に等しく設定された (x<sub>E<sub>k</sub></sub>, y<sub>E<sub>k</sub></sub>)、および `cSL` を入力とし、出力を `cSL` として、起動される。

`filterInternalLumaEdgesFlag` が 1 に等しい場合、内部垂直輝度エッジのフィルタは以下のように規定される。

- `lumaTrafo[CurrMbAddr]` が `T_8x8` と等しくないとき、G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された `chromaEdgeFlag`、1 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBv[CurrMbAddr]` に等しく設定された `qPp`、`qpDBv[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..15` として (4,k) に等しく設定された (x<sub>E<sub>k</sub></sub>, y<sub>E<sub>k</sub></sub>)、および `cSL` を入力とし、出力を `cSL` として、起動される。

- G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、1 に等しく設定された verticalEdgeFlag、fieldModeMbFlag に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、k=0..15 として(8,k) に等しく設定された(xE<sub>k</sub>,yE<sub>k</sub>)、および cS<sub>L</sub> を入力とし、出力を cS<sub>L</sub> として、起動される。
- lumaTrafo[CurrMbAddr] が T\_8x8 と等しくない場合、G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、1 に等しく設定された verticalEdgeFlag、fieldModeMbFlag に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、k=0..15 として(12,k) に等しく設定された(xE<sub>k</sub>,yE<sub>k</sub>)、および cS<sub>L</sub> を入力とし、出力を cS<sub>L</sub> として起動される。

filterTopLumaMbEdgeFlag が 1 に等しい場合、上部水平輝度エッジのフィルタは以下のように規定される。

- MbaffFrameFlag が 1 に等しく、(CurrMbAddr%2) が 0 に等しく、CurrMbAddr が (2\*PicWidthInMbs) 以上であり、fieldMbFlag[CurrMbAddr] が 0 に等しく、かつ fieldMbFlag[CurrMbAddr-2\*PicWidthInMbs+1] が 1 に等しい場合、以下が適用される。
  - G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、0 に等しく設定された verticalEdgeFlag、1 に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[CurrMbAddr-2\*PicWidthInMbs] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、k=0..15 として(k,0) に等しく設定された(xE<sub>k</sub>,yE<sub>k</sub>)、および cS<sub>L</sub> を入力とし、出力を cS<sub>L</sub> として起動される。
  - G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、0 に等しく設定された verticalEdgeFlag、1 に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[CurrMbAddr-2\*PicWidthInMbs+1] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、k=0..15 として(k,1) に等しく設定された(xE<sub>k</sub>,yE<sub>k</sub>)、および cS<sub>L</sub> を入力とし、出力を cS<sub>L</sub> として起動される。
- それ以外、G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、0 に等しく設定された verticalEdgeFlag、fieldModeMbFlag に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[mbAddrB] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、k=0..15 として(k,0) に等しく設定された(xE<sub>k</sub>,yE<sub>k</sub>)、および cS<sub>L</sub> を入力とし、出力を cS<sub>L</sub> として起動される。

filterInternalLumaEdgesFlag が 1 に等しい場合、内部水平輝度エッジのフィルタは以下のように規定される。

- lumaTrafo[CurrMbAddr] が T\_8x8 と等しくない場合、G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、0 に等しく設定された verticalEdgeFlag、fieldModeMbFlag に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、k=0..15 として(k,4) に等しく設定された(xE<sub>k</sub>,yE<sub>k</sub>)、および cS<sub>L</sub> を入力とし、出力を cS<sub>L</sub> として起動される。
- G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された chromaEdgeFlag、0 に等しく設定された verticalEdgeFlag、fieldModeMbFlag に等しく設定された fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>p</sub>、qpDB<sub>V</sub>[CurrMbAddr] に等しく設定された qP<sub>q</sub>、currentVars、

$k=0..15$  として  $(k,8)$  に等しく設定された  $(xE_k, yE_k)$ 、および  $cS_L$  を入力とし、出力を  $cS_L$  として起動される。

- $lumaTrafo[CurrMbAddr]$  が  $T_{8x8}$  と等しくない場合、G.8.7.4.1 小小小節に規定される処理が、0 に等しく設定された  $chromaEdgeFlag$ 、0 に等しく設定された  $verticalEdgeFlag$ 、 $fieldModeMbFlag$  に等しく設定された  $fieldModeFilteringFlag$ 、 $filterOffsetA$ 、 $filterOffsetB$ 、 $qpDB_V[CurrMbAddr]$  に等しく設定された  $qP_p$ 、 $qpDB_V[CurrMbAddr]$  に等しく設定された  $qP_q$ 、 $currentVars$ 、 $k=0..15$  として  $(k,12)$  に等しく設定された  $(xE_k, yE_k)$ 、および  $cS_L$  を入力とし、出力を  $cS_L$  として起動される。

$ChromaArrayType$  が 0 に等しくない場合、 $qpDB_C$  と  $cS_C$  の  $C$  を  $C_b$  と  $C_r$  で置き換え、両方の色差成分のフィルタについて以下が適用される。

- $filterLeftChromaMbEdgeFlag$  が 1 に等しい場合、左部垂直色差エッジのフィルタは以下のように規定される。
  - G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された  $chromaEdgeFlag$ 、1 に等しく設定された  $verticalEdgeFlag$ 、 $fieldModeMbFlag$  に等しく設定された  $fieldModeFilteringFlag$ 、 $filterOffsetA$ 、 $filterOffsetB$ 、 $qpDB_C[mbAddrA]$  に等しく設定された  $qP_p$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_q$ 、 $currentVars$ 、 $k=0..(MbHeightC-1)$  として  $(0,k)$  に等しく設定された  $(xE_k, yE_k)$ 、および  $cS_C$  を入力とし、出力を  $cS_C$  として起動される。
- $filterInternalChromaEdgesFlag$  が 1 に等しい場合、内部垂直色差エッジのフィルタは以下のように規定される。
  - $ChromaArrayType$  が 3 に等しくない、または、 $transform\_size\_8x8\_flag$  が 0 に等しい場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された  $chromaEdgeFlag$ 、1 に等しく設定された  $verticalEdgeFlag$ 、 $fieldModeMbFlag$  に等しく設定された  $fieldModeFilteringFlag$ 、 $filterOffsetA$ 、 $filterOffsetB$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_p$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_q$ 、 $currentVars$ 、 $k=0..(MbHeightC-1)$  として  $(4,k)$  に等しく設定された  $(xE_k, yE_k)$ 、および  $cS_C$  を入力とし、出力を  $cS_C$  として起動される。
  - $ChromaArrayType$  が 3 に等しい場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された  $chromaEdgeFlag$ 、1 に等しく設定された  $verticalEdgeFlag$ 、 $fieldModeMbFlag$  に等しく設定された  $fieldModeFilteringFlag$ 、 $filterOffsetA$ 、 $filterOffsetB$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_p$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_q$ 、 $currentVars$ 、 $k=0..(MbHeightC-1)$  として  $(8,k)$  に等しく設定された  $(xE_k, yE_k)$ 、および  $cS_C$  を入力とし、出力を  $cS_C$  として起動される。
  - $ChromaArrayType$  が 3 に等しい場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された  $chromaEdgeFlag$ 、1 に等しく設定された  $verticalEdgeFlag$ 、 $fieldModeMbFlag$  に等しく設定された  $fieldModeFilteringFlag$ 、 $filterOffsetA$ 、 $filterOffsetB$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_p$ 、 $qpDB_C[CurrMbAddr]$  に等しく設定された  $qP_q$ 、 $currentVars$ 、 $k=0..(MbHeightC-1)$  として  $(12,k)$  に等しく設定された  $(xE_k, yE_k)$ 、および  $cS_C$  を入力とし、出力を  $cS_C$  として起動される。
- $filterTopChromaMbEdgeFlag$  が 1 に等しい場合、上部水平色差エッジのフィルタは以下のように規定される。
  - もし  $MbaffFrameFlag$  が 1 に等しく、 $(CurrMbAddr\%2)$  が 0 に等しく、 $CurrMbAddr$  が  $(2*PicWidthInMbs)$  以上であり、 $fieldMbFlag[CurrMbAddr]$  が 0 に等しく、 $fieldMbFlag[CurrMbAddr-2*PicWidthInMbs+1]$  が 1 に等しいならば、以下が適用される。

- G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、1 に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[CurrMbAddr-2*PicWidthInMbs]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,0)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。
- G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、1 に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[CurrMbAddr-2*PicWidthInMbs+1]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,1)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。
- それ以外、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[mbAddrB]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,0)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。
- `filterInternalChromaEdgesFlag` が 1 に等しい場合、内部水平色差エッジのフィルタは以下のように規定される。
  - `ChromaArrayType` が 3 に等しくない、または、`transform_size_8x8_flag` が 0 に等しい場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[CurrMbAddr]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,4)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。
  - `ChromaArrayType` が 1 に等しくない場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[CurrMbAddr]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,8)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。
  - `ChromaArrayType` が 2 に等しい場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[CurrMbAddr]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,12)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。
  - `ChromaArrayType` が 3 に等しく、または、`transform_size_8x8_flag` が 0 に等しい場合、G.8.7.4.1 小小小節に規定される処理が、1 に等しく設定された `chromaEdgeFlag`、0 に等しく設定された `verticalEdgeFlag`、`fieldModeMbFlag` に等しく設定された `fieldModeFilteringFlag`、`filterOffsetA`、`filterOffsetB`、`qpDBC[CurrMbAddr]` に等しく設定された `qPp`、`qpDBC[CurrMbAddr]` に等しく設定された `qPq`、`currentVars`、`k=0..(MbWidthC-1)` として `(k,12)` に等しく設定された `(xEk,yEk)`、および `cSC` を入力とし、出力を `cSC` として起動される。

記1 フィールドモードフィルタ (fieldModeFilteringFlagが1に等しい) が、フレームマクロブロックの上部水平エッジを交差して適用される場合、この上部または下部マクロブロック境界を交差する垂直フィルタは、フレームモードでも内部的にフィルタされる内部ブロックエッジを越えて拡がるサンプルを使用しても良い。

記2 例えば、transform\_size\_8x8\_flagが0に等しい時の4:2:0色差フォーマットでは、以下が適用される。マクロブロックの内部の、3つの水平輝度エッジ、Cbに対する1つの水平色差エッジ、および、Crに対する1つの水平色差エッジがフィルタされる。フィールドモードフィルタ (fieldModeFilteringFlagが1に等しい) が、フレームマクロブロックの上部エッジに適用される場合、そのフレームマクロブロックと上のマクロブロックペアの間の、2つの水平輝度、Cbに対する2つの水平色差エッジ、および、Crに対する2つの水平色差エッジが、フィールドモードフィルタを用いてフィルタされ、合計最大5つの水平輝度エッジ、Cbに対する3つの水平色差エッジ、およびCrに対する3つの水平色差エッジについて、そのフレームマクロブロックにより制御されるものとみなされてフィルタされる。他の全ての場合においては、最大4つの水平輝度、Cbに対する2つの色差エッジ、およびCrに対する2つの水平色差エッジが、特定のマクロブロックにより制御されるものとみなされてフィルタされる。

#### G.8.7.4.1 ブロックエッジにおけるSVCフィルタ処理

この処理への入力は以下の通りである。

- 変数 chromaEdgeFlag
- 変数 verticalEdgeFlag
- 変数 fieldModeFilteringFlag
- 変数 filterOffsetA と filterOffsetB
- 変数  $qP_p$  と  $qP_q$
- 総称 currentVars
- nE 個の、マクロブロック CurrMbAddr の左上隅からの相対で表現されたサンプル位置( $xE_k, yE_k$ )、 $k=0..nE-1$ の組。サンプル位置( $xE_k, yE_k$ )の組は、垂直エッジ(verticalEdgeFlag が 1 に等しい場合)のすぐ右側のサンプル位置、あるいは水平エッジ(verticalEdgeFlag が 0 に等しい場合)のすぐ下側のサンプル位置を示している。
- サンプル配列  $s'$

この処理の出力は配列  $s'$  の修正版である。

変数 nE は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、nE は 16 に等しく設定される。
- それ以外(chromaEdgeFlag が 1 に等しい)、nE は  $(verticalEdgeFlag == 1) ? MbHeightC : MbWidthC$  に等しく設定される。

この小小小節の内部で、currentVars として総称的に参照される配列は、G.8.1.2.1 小小小節で規定されるそれらの名前を参照される。

変数 dy は次の通り導出される。

- もし fieldModeFilteringFlag が 1 に等しく、かつ MbaffFrameFlag が 1 に等しいならば、dy は 2 に等しく設定される。

- それ以外(fieldModeFilteringFlag が 0 に等しいか、または MbaffFrameFlag が 0 に等しい)、dy は 1 に等しく設定される。

マクロブロック CurrMbAddr の左上輝度サンプルの位置は、6.4.1 小節の逆マクロブロック走査処理を mbAddr=CurrMbAddr を入力として起動し、出力が(xI,yI)に割り当てられることで導出される。6.4.1 小節の処理の間、現マクロブロックは、fieldMbFlag[CurrMbAddr]が 1 に等しい時、フィールドマクロブロックとして扱われ、fieldMbFlag[CurrMbAddr]が 0 に等しい時、フレームマクロブロックとして扱われる。

変数 xP と yP は次の通り導出される。

- もし chromaEdgeFlag が 0 に等しいならば、xP は xI に等しく設定され、かつ yP は yI に等しく設定される。
- それ以外(chromaEdgeFlag が 1 に等しい)、xP は(xI/SubWidthC)に等しく設定され、かつ yP は ((yI+SubHeightC-1)/SubHeightC)に等しく設定される。

各サンプル位置(xE<sub>k</sub>,yE<sub>k</sub>)、k=0..(nE-1)に対して以下が適用される。

- p<sub>0</sub> と q<sub>0</sub> の間にエッジがある図 8-11/JT-H264 に示される様な p<sub>i</sub> と q<sub>i</sub>、i=0..3 で示される 4×4 ブロックの水平または垂直エッジを交差する 8 個のサンプルの組にフィルタ処理が適用される。p<sub>i</sub> と q<sub>i</sub>、i=0..3 は次の通り規定される。

- もし verticalEdgeFlag が 1 に等しいならば、

$$q_i = s'[xP + xE_k + i, yP + dy * yE_k] \quad (G-353)$$

$$p_i = s'[xP + xE_k - i - 1, yP + dy * yE_k] \quad (G-354)$$

- それ以外 (verticalEdgeFlag が 0 に等しい)、

$$q_i = s'[xP + xE_k, yP + dy * (yE_k + i) - (yE_k \% 2)] \quad (G-355)$$

$$p_i = s'[xP + xE_k, yP + dy * (yE_k - i - 1) - (yE_k \% 2)] \quad (G-356)$$

- G.8.7.4.2 小小小節で規定される処理が、サンプル値 p<sub>i</sub> と q<sub>i</sub>(i=0..3)、chromaEdgeFlag、verticalEdgeFlag、fieldModeFilteringFlag、filterOffsetA、filterOffsetB、qP<sub>p</sub>、qP<sub>q</sub>、sliceIdc、fieldMbFlag、mbType、predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1、sTCoeff、および rS<sub>L</sub> を入力として起動され、出力がフィルタされた結果のサンプル値 p'<sub>i</sub> と q'<sub>i</sub>、i=0..2 に割り当てられる。

- 入力サンプル値 p<sub>i</sub> と q<sub>i</sub>、i=0..2 は、サンプル配列 s'内で対応したフィルタ結果のサンプル値 p'<sub>i</sub> と q'<sub>i</sub>、i=0..2 に次の通り置き換えられる。

- もし verticalEdgeFlag が 1 に等しいならば、

$$s'[xP + xE_k + i, yP + dy * yE_k] = q'_i \quad (G-357)$$

$$s'[xP + xE_k - i - 1, yP + dy * yE_k] = p'_i \quad (G-358)$$

- それ以外 (verticalEdgeFlag が 0 に等しい)、

$$s'[xP + xE_k, yP + dy * (yE_k + i) - (yE_k \% 2)] = q'_i \quad (G-359)$$

$$s'[xP+xE_k,yP+dy*(yE_k-i-1)-(yE_k\%2)]=p'_i \quad (\text{G-360})$$

#### G.8.7.4.2 水平または垂直ブロックエッジを交差するサンプルの組に対するSVCフィルタ処理

この処理への入力は以下の通りである。

- フィルタされるエッジを交差するサンプルの 1 つの組である  $i$  を 0..3 の範囲とした入力サンプル値  $p_i$  と  $q_i$
- 変数 `chromaEdgeFlag`
- 変数 `verticalEdgeFlag`
- 変数 `fieldModeFilteringFlag`
- 変数 `filterOffsetA` と `filterOffsetB`
- 変数  $qP_p$  と  $qP_q$
- 配列 `sliceIdc`, `fieldMbFlag`, `mbType`, `predFlagL0`, `predFlagL1`, `refIdxL0`, `refIdxL1`, `mvL0`, `mvL1`, および `sTCoeff`
- 残差サンプル値を含む配列  $rS_L$

この処理の出力は、フィルタされた結果のサンプル値  $p'_i$  と  $q'_i$  であり、 $i$  は 0..2 の範囲である。

内容依存境界フィルタ強度変数  $bS$  は次の通り導出される。

- もし `chromaEdgeFlag` が 0 に等しいならば、G.8.7.4.3 小小節で規定される輝度内容依存独立境界フィルタ強度に対する SVC 導出処理が、 $p_0$ ,  $q_0$ , `verticalEdgeFlag`, `sliceIdc`, `fieldMbFlag`, `mbType`, `predFlagL0`, `predFlagL1`, `refIdxL0`, `refIdxL1`, `mvL0`, `mvL1`, `sTCoeff`, および  $rS_L$  を入力として起動され、出力が  $bS$  に割り当てられる。
- それ以外(`chromaEdgeFlag` が 1 に等しい)、水平または垂直の色差エッジのサンプルの組をフィルタするために用いられる  $bS$  は、それぞれ、その同じフィールドの輝度配列内の(SubWidthC\*x,SubHeightC\*y)位置での輝度サンプルを含む水平または垂直の輝度エッジのサンプルの組をフィルタするために用いられる  $bS$  の値に等しく設定される。ここで(x,y)はそのフィールドでの色差配列内の色差サンプル  $q_0$  の位置を示す。

8.7.2.2 小小節で規定される処理が、 $p_0$ ,  $q_0$ ,  $p_1$ ,  $q_1$ , `chromaEdgeFlag`,  $bS$ , `filterOffsetA`, `filterOffsetB`,  $qP_p$ , および  $qP_q$  を入力として起動され、出力が `filterSamplesFlag`, `indexA`,  $\alpha$ , および  $\beta$  に割り当てられる。

変数 `filterSamplesFlag` に依存して、次が適用される。

- もし `filterSamplesFlag` が 1 に等しいならば、次が適用される。
  - もし  $bS$  が 4 より小さいならば、8.7.2.3 小小節で規定される処理が、入力として与えられた  $p_i$  と  $q_i$ ( $i=0..2$ )、`chromaEdgeFlag`,  $bS$ ,  $\beta$ , および `indexA` で起動され、出力が  $p'_i$  と  $q'_i$ ( $i=0..2$ )に割り当てられる。
  - それ以外( $bS$  が 4 に等しい)、8.7.2.4 小小節で規定される処理が、入力として与えられた  $p_i$  と  $q_i$ ( $i=0..3$ )、`chromaEdgeFlag`,  $\alpha$ , および  $\beta$  で起動され、出力が  $p'_i$  と  $q'_i$ ( $i=0..2$ )に割り当てられる。

- それ以外(filterSamplesFlag が 0 に等しい)、フィルタされた結果のサンプル  $p'_i$  と  $q'_i(i=0..2)$ は対応した入力サンプル  $p_i$  と  $q_i$  に置き換えられる。

$$i=0..2 \text{ に対して、} \quad p'_i = p_i \quad (\text{G-361})$$

$$i=0..2 \text{ に対して、} \quad q'_i = q_i \quad (\text{G-362})$$

#### G.8.7.4.3 輝度の内容依存境界フィルタ強度に対するSVC導出処理

この処理への入力は以下の通りである。

- フィルタされるエッジを交差するサンプルの 1 つの組である入力サンプル値  $p_0$  と  $q_0$
- 変数 verticalEdgeFlag
- 配列 sliceIdc、fieldMbFlag、mbType、predFlagL0、predFlagL1、refIdxL0、refIdxL1、mvL0、mvL1、および sTCoeff
- 残差サンプル値を含む配列 rS<sub>L</sub>

この処理の出力は変数 bS である。

次の変数が、以下で規定されることによって導出される。

- mbAddrP と mbAddrQ は、それぞれサンプル  $p_0$  と  $q_0$  を含むマクロブロックを規定する。
- mbPartIdxP と mbPartIdxQ は、それぞれサンプル  $p_0$  と  $q_0$  を含むマクロブロックパーティションを規定する。
- subMbPartIdxP と subMbPartIdxQ は、それぞれサンプル  $p_0$  と  $q_0$  を含むサブマクロブロックパーティションを規定する。
- X が 0 と 1 に置き換えられる pFLXP と pFLXQ は、それぞれ predFlagLX[mbAddrP][mbPartIdxP] と predFlagLX[mbAddrQ][mbPartIdxQ] に等しくされる。
- X が 0 と 1 に置き換えられる refLXP と refLXQ は、それぞれ refIdxLX[mbAddrP][mbPartIdxP] と refIdxLX[mbAddrQ][mbPartIdxQ] に等しくされる。
- X が 0 と 1 に置き換えられる mvLXP と mvFLXQ は、それぞれ mvLX[mbAddrP][mbPartIdxP][subMbPartP] と mvLX[mbAddrQ][mbPartIdxQ][subMbPartQ] に等しくされる。
- numMvP と numMvQ は、それぞれ (pFL0P+pFL1P) と (pFL0Q+pFL1Q) に等しくされる。
- numMvP と numMvQ が共に 1 に等しい時、X が P と Q に置き換えられる変数 refX と mvX は次のように導出される。
  - もし pFL0X が 1 に等しいならば、refX は refL0X に等しく設定され、mvX は mvL0X に等しく設定される。
  - それ以外(pFL1X が 1 に等しい)、refX は refL1X に等しく設定され、mvX は mvL1X に等しく設定される。
- X が P と Q に置き換えられる sliceX は、(sliceIdc[mbAddrP]&127)に等しい DQId と(sliceIdc[mbAddrP]>>7)

に等しい `first_mb_in_slice` を持つスライスである。

変数 `mixedModeEdgeFlag` が次で導出されるとする。

- もし `MbaffFrameFlag` が 1 に等しく、かつ `fieldMbFlag[mbAddrP]` が `fieldMbFlag[mbAddrQ]` に等しくないならば、`mixedModeEdgeFlag` は 1 に等しく設定される。
- それ以外、`mixedModeEdgeFlag` は 0 に等しく設定される。

変数 `bS` が次の通り導出される。

- もし `SpatialResolutionChangeFlag` が 1 に等しく、かつ `mbType[mbAddrP]` か `mbType[mbAddrQ]` のどちらか、もしくは両方が `I_BL` に等しいならば、次が適用される。
  - もし `mbType[mbAddrP]` か `mbType[mbAddrQ]` のどちらかが `I_BL` 以外の `INTRA` マクロブロック予測モードを規定するならば、`bS` は 4 に設定される。
  - それ以外、もし `mbType[mbAddrP]` が `I_BL` に等しく、かつ `mbType[mbAddrQ]` が `I_BL` に等しいならば、次が適用される。
    - もしサンプル `p0` を含む `sliceP` の輝度変換ブロックもしくはサンプル `q0` を含む `sliceQ` の輝度変換ブロックが非ゼロ変換係数レベルを含んでいるならば、`bS` は 1 に等しく設定される。

記 1 `dependency_id` と `quality_id` を持つレイヤ表現の輝度変換ブロックは、もし非ゼロ変換係数が対象輝度変換ブロックの `dependency_id` と `quality_id` に対してマクロブロックレイヤ内で転送されるならば、かつその場合に限り、非ゼロ変換係数レベルを含むと考えられる。レイヤ間予測で使用されるレイヤに転送される変換係数レベルは考慮されない。

- それ以外、`bS` は 0 に等しく設定される。
- それ以外(`mbType[mbAddrP]` か `mbType[mbAddrQ]` のどちらかが `INTER` マクロブロック予測モードを規定する)、`bS` は次が適用される。
  - もし `mbType[mbAddrP]` が `INTER` マクロブロック予測タイプを規定し、かつ配列 `rSL` がサンプル `p0` を含む輝度ブロックにおいて任意の非ゼロサンプルを含むならば、`bS` は 2 に等しく設定される。
  - それ以外、もし `mbType[mbAddrQ]` が `INTER` マクロブロック予測タイプを規定し、かつ配列 `rSL` がサンプル `q0` を含む輝度ブロックにおいて任意の非ゼロサンプルを含むならば、`bS` は 2 に等しく設定される。

記 2 配列 `rSL` は累算された残差信号のサンプルを含む。レイヤ間予測で使用されるレイヤ表現の変換係数の値は考慮される。

- それ以外、`bS` は 1 に等しく設定される。
- それ以外、もしブロックエッジがマクロブロックエッジでもあり、かつ以下の条件のいずれかが真ならば、`bS` は 4 に等しく設定される。
  - `fieldMbFlag[mbAddrP]` が 0 に等しく、かつ `fieldMbFlag[mbAddrQ]` が 0 に等しく、かつ `mbType[mbAddrP]` か `mbType[mbAddrQ]` のどちらかあるいは両者が `INTRA` マクロブロック予測モードを規定する。

- MbaffFrameFlag が 1 に等しいか field\_pic\_flag が 1 に等しく、かつ、verticalEdgeFlag が 1 に等しく、かつ mbType[mbAddrP]か mbType[mbAddrQ]のどちらかあるいは両者が INTRA マクロブロック予測モードを規定する。
- それ以外、もし以下の条件のいずれかが真ならば、bS は 3 に等しく設定される。
  - mixedModeEdgeFlag が 0 に等しく、かつ mbType[mbAddrP]か mbType[mbAddrQ]のどちらかあるいは両者が INTRA マクロブロック予測モードを規定する。
  - mixedModeEdgeFlag が 1 に等しく、かつ verticalEdgeFlag が 0 に等しく、かつ、mbType[mbAddrP]か mbType[mbAddrQ]のどちらかあるいは両者が INTRA マクロマクロブロック予測モードを規定する。
- それ以外、もし以下の条件のいずれかが真ならば、bS は 2 に等しく設定される。
  - 配列 rS<sub>L</sub> がサンプル p<sub>0</sub> を含む輝度ブロックにおける任意の非ゼロサンプルを含むか、または、配列 sTCoeff[mbAddrP]がサンプル p<sub>0</sub> を含む輝度ブロックにおける任意の非ゼロのスケーリングされた変換係数値を含む。
  - 配列 rS<sub>L</sub> がサンプル q<sub>0</sub> を含む輝度マクロブロックにおける任意の非ゼロサンプルを含むか、または、配列 sTCoeff[mbAddrQ]がサンプル q<sub>0</sub> を含む輝度マクロブロックにおける任意の非ゼロのスケーリングされた変換係数値を含む
 

記 3 配列 rS<sub>L</sub> は累算された残差信号のサンプルを含む。レイヤ間予測で使用されたレイヤ表現の変換係数値は考慮される。
- それ以外、もし以下の条件のいずれかが真ならば、bS は 1 に等しく設定される。
  - mixedModeEdgeFlag が 1 に等しい。
  - mixedModeEdgeFlag が 0 に等しく、かつ、以下の条件のいずれかが真である。
    - numMvPがnumMvQに等しくない。
    - numMvPとnumMvQが共に1に等しく、かつ、以下の条件のいずれかが真である。
      - refPとrefQが異なる参照ピクチャを規定する。
      - 動きベクトルmvPとmvQの間の水平もしくは垂直成分の絶対値差分が、4分の1輝度フレームサンプル単位において4よりも大きいか等しい。
  - numRefPとnumRefQが共に2に等しく、かつ、以下の条件のいずれかが真である。
    - refL0PとrefL1Pが異なる参照ピクチャを規定し、かつ、以下の条件のいずれかが真である。
      - 以下の条件の両方が真である。
        - refL0PとrefL0Qが異なる参照ピクチャを規定する、またはrefL1PとrefL1Qが異なる参照ピクチャを規定する。
        - refL0PとrefL1Qが異なる参照ピクチャを規定する、またはrefL1PとrefL0Qが異なる参照ピクチャを規定する。

- refL0PとrefL0Qが同一参照ピクチャを規定し、かつ、refL1PとrefL1Qが同一参照ピクチャを規定し、かつ、以下の条件のいずれかが真である。
  - 動きベクトルmvL0PとmvL0Qの間の水平もしくは垂直成分の絶対値差分が、4分の1輝度フレームサンプル単位において4よりも大きいか等しい。
  - 動きベクトルmvL1PとmvL1Qの間の水平もしくは垂直成分の絶対値差分が、4分の1輝度フレームサンプル単位において4よりも大きいか等しい。
- refL0PとrefL1Qが同一参照ピクチャを規定し、かつ、refL1PとrefL0Qが同一参照ピクチャを規定し、かつ、以下の条件のいずれかが真である。
  - 動きベクトルmvL0PとmvL1Qの間の水平もしくは垂直成分の絶対値差分が、4分の1輝度フレームサンプル単位において4よりも大きいか等しい。
  - 動きベクトルmvL1PとmvL0Qの間の水平もしくは垂直成分の絶対値差分が、4分の1輝度フレームサンプル単位において4よりも大きいか等しい。
- refL0PとrefL1Pが同一参照ピクチャで規定され、かつ、以下の条件のいずれかが真である。
  - refL0QもしくはrefL1Qが、refL0P(もしくはrefL1P)と異なる参照ピクチャを規定する。
  - refL0QとrefL1QがrefL0P(もしくはrefL1P)と同一参照ピクチャを規定し、かつ、以下の条件のいずれかが真である。
    - 動きベクトルmvL0PとmvL0Qの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい。
    - 動きベクトルmvL1PとmvL1Qの間の水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい。
    - 動きベクトルmvL0PとmvL1Qの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい。
    - 動きベクトルmvL1PとmvL0Qの水平あるいは垂直成分の間の絶対差分が4分の1輝度フレームサンプル単位で4より大きいか等しい。

記4 2つのマクロブロック／サブマクロブロックパーティションに対して使用される参照ピクチャが同じか異なるかの決定は、予測が参照ピクチャリスト0へのインデックスまたは参照ピクチャリスト1へのインデックスを使用して形成されたかには関係なく、かつ、参照ピクチャリストのインデックス位置が異なるかどうかにも関係なく、どのピクチャが参照されるかのみに基づく。

記5 4分の1輝度フレームサンプル単位で4の垂直差は、4分の1輝度フィールドサンプル単位で2の差である。

- それ以外、bSは0に設定される。

## G.8.8 ビットストリームサブセットの規定

G.8.8.1 小節ではサブビットストリーム抽出処理を規定する。

G.8.8.2 小小節ではベースレイヤビットストリームを規定する。

#### G.8.8.1 サブビットストリーム抽出処理

この処理への入力は以下の通りである。

- 変数 pIdTarget (存在する時)
- 変数 tIdTarget (存在する時)
- 変数 dIdTarget (存在する時)
- 変数 qIdTarget (存在する時)

この処理の出力は、サブビットストリームである。

この小小節の入力として pIdTarget が存在しない時、pIdTarget は 63 に等しいと推定される。

この小小節の入力として tIdTarget が存在しない時、tIdTarget は 7 に等しいと推定される。

この小小節の入力として dIdTarget が存在しない時、dIdTarget は 7 に等しいと推定される。

この小小節の入力として qIdTarget が存在しない時、qIdTarget は 15 に等しいと推定される。

サブビットストリームは、シーケンシャルな順序で次の操作が適用されることにより導出される。

1. 次の条件のいずれかが真である全ての VCL、フィルプリフィックス、フィールドデータ NAL ユニットの “ビットストリームから削除される” としてマークする。
  - priority\_id が pIdTarget より大きい。
  - temporal\_id が tIdTarget より大きい。
  - dependency\_id が dIdTarget より大きい。
  - dependency\_id が dIdTarget に等しく、かつ、quality\_id が qIdTarget より大きい。
2. 全ての VCL NAL ユニットに対して “ビットストリームから削除される” とマークされた全てのアクセスユニットを削除する。
3. “ビットストリームから削除される” とマークされた全ての VCL、フィルプリフィックス、フィールドデータ NAL ユニットの削除する。
4. dIdTarget が 0 に等しく、かつ、qIdTarget が 0 に等しい時、次の NAL ユニットの削除する。
  - 14 または 15 に等しい nal\_unit\_type を持つ全ての NAL ユニット
  - 最初の SEI メッセージが 24 から 35 のそれぞれの値を含む範囲にある payloadType を持っている、6 に等しい nal\_unit\_type を持つ全ての NAL ユニット
5. 次の特質のいずれかをもつスケラブルネスティング SEI メッセージの一部である SEI メッセージのみを含む、6 に等しい nal\_unit\_type を持つ全ての NAL ユニットの削除する。
  - sei\_temporal\_id が tIdTarget より大きい。

- $i$  が 0 から  $\text{num\_layer\_representations\_minus1}$  までのそれぞれの値の範囲を持つ ( $\text{sei\_dependency\_id}[i] \ll 4$ ) +  $\text{sei\_quality\_id}[i]$  の最小値が、 $(\text{dIdTarget} \ll 4) + \text{qIdTarget}$  より大きい。

### G.8.8.2 ベースレイヤビットストリームの規定

この規定に適合する各スケーラブルビットストリームは、付属資料 A で規定される 1 つもしくはそれ以上のプロファイルに適合するベースレイヤビットストリームを含まなければならない。このベースレイヤビットストリームは、0 に等しくされた  $\text{dIdTarget}$  と 0 に等しくされた  $\text{qIdTarget}$  を使い、ベースレイヤビットストリームをその出力として G.8.8.1 小小節で規定されるサブビットストリーム抽出処理を起動することによって導出される。

## G.9 構文解析処理

この処理への入力は、RBSP からのビットと、シンタックス要素の値に対する要求と、先に構文解析されたシンタックス要素の値(適用可能なら)である。

この処理の出力はシンタックス要素の値である。

この処理は  $u(v)$ 、 $ue(v)$ 、 $me(v)$ 、 $se(v)$ 、 $te(v)$ 、 $ce(v)$  と  $ae(v)$  に等しい記述子を持つ、G.7.3 小節のシンタックス表の全てのシンタックス要素に対して起動される。

20 に等しい  $\text{nal\_unit\_type}$  を持つ符号化スライス NAL ユニットの構文解析する時、これらの NAL ユニットには存在しないシンタックス要素である  $\text{scan\_idx\_start}$  と  $\text{scan\_idx\_end}$  は、それぞれ 0 と 15 に等しいと推定される。

構文解析処理がスライスデータ中のシンタックス要素値に対する最初の要求に対して起動され、かつ  $\text{entropy\_coding\_mode\_flag}$  が 1 に等しい時、以下の順序づけられたステップが適用される。

1. 9.3.1 小節で規定される初期化処理が、EI に等しい  $\text{slice\_type}$  は I として解釈され、EP に等しい  $\text{slice\_type}$  は P として解釈され、EB に等しい  $\text{slice\_type}$  は B として解釈されることによって起動される。
2. G.9.3.1 小小節で規定される初期化処理が起動される。

$\text{entropy\_coding\_mode\_flag}$  と記述子に従って、シンタックス要素値が次のように導出される。

- もし  $\text{entropy\_coding\_mode\_flag}$  が 0 に等しいならば、次が適用される。
  - $ue(v)$ 、 $se(v)$  または  $te(v)$  として符号化されたシンタックス要素に対する構文解析処理は、9.1 節で規定される。
  - シンタックス要素  $\text{coded\_block\_pattern}$  に対する構文解析処理は、G.9.1 小節で規定される。
  - シンタックス構造  $\text{residual\_block\_cavlc}()$  のシンタックス要素に対する構文解析処理は、G.9.2 小節で規定される。
- それ以外 ( $\text{entropy\_coding\_mode\_flag}$  が 1 に等しい)、シンタックス要素値は次のように導出される。
  - もしシンタックス要素が  $\text{base\_mode\_flag}$ 、 $\text{motion\_prediction\_flag\_10}$ 、 $\text{motion\_prediction\_flag\_11}$ 、または  $\text{residual\_prediction\_flag}$  に等しいならば、以下が適用される。
    1. G.9.3.2 小小節で規定されるバイナリ化処理が起動される。

2. G.9.3.3 小小節で規定される復号処理フローが起動される。
- それ以外(シンタックス要素が `base_mode_flag`, `motion_prediction_flag_l0`, `motion_prediction_flag_l1`, または `residual_prediction_flag` に等しくない)、以下が適用される。
    1. 9.3.2 小節で規定されるバイナリ化処理が、EI に等しい `slice_type` は I として解釈され、EP に等しい `slice_type` は P として解釈され、EB に等しい `slice_type` は B として解釈されることによって起動される。
    2. 9.3.3 小節で規定される復号処理フローが起動される。
    3. シンタックス要素が `mb_type` に等しく、かつ `mb_type` の復号値が I\_PCM に等しい時、算術復号エンジンは、9.3.1.2 小小節で規定される、任意の `pcm_alignment_zero_bit` と全ての `pcm_sample_luma` と `pcm_sample_chroma` データを復号した後に初期化される。

### G.9.1 符号化ブロックパターンに対する代替構文解析処理

この処理は、`entropy_coding_mode_flag` が 0 に等しい時、`me(v)` に等しい記述子を持つ構文解析するシンタックス要素に対して起動される。

この処理への入力、RBSP からのビットである。

この処理の出力は、シンタックス要素 `coded_block_pattern` の値である。

シンタックス要素に対する構文解析処理は、ビットストリームの現位置から最初の非ゼロビットを含むその位置までビットを読み出すことから始める。0 に等しい先行ビットの数を数え、変数 `leadingZeroBits` の値を割り当てることによって、変数 `codeNum` が次のように導出される。

$$\text{codeNum} = 2^{\text{leadingZeroBits}} - 1 + \text{read\_bits}(\text{leadingZeroBits})$$

ここで `read_bits(leadingZeroBits)` からの戻り値は、最上位ビットが最初に書かれる符号無し整数のバイナリ表現として解釈される。

`ref_layer_dq_id` が 0 より大きいかあるいは等しく、かつ `(scan_idx_end - scan_idx_start)` が 15 よりも小さい時、`codeNum` は `(codeNum-1)` に等しく設定される。

`codeNum` に従って、以下を適用する。

- もし `codeNum` が -1 に等しいならば、以下の順序付けられたステップが規定される。
  1. 6.4.10.1 小小節で規定される隣接マクロブロックに対する導出処理が実行され、出力は `mbAddrA` と `mbAddrB` に割り当てられる。
  2. `mbAddrN` が利用可能である時、変数 `codedBlockPatternN` は、マクロブロック `mbAddrN` (N は A または B) における `CodedBlockPatternLuma` と `CodedBlockPatternChroma` の値を表す `cbpChromaN` と `cbpLumaN` を用いて  $(16 * \text{cbpChromaN} + \text{cbpLumaN})$  に設定される。
  3. `mbAddrA` と `mbAddrB` に従って、以下を適用する。
    - もし `mbAddrA` が利用可能ならば、`coded_block_pattern` は `codedBlockPatternA` に等しく設定される。

- それ以外、もし `mbAddrB` が利用可能ならば、`coded_block_pattern` は `codedBlockPatternB` に等しく設定される。
- それ以外 (`mbAddrA` と `mbAddrB` が利用可能でない)、`coded_block_pattern` は 0 に等しく設定される。
- それ以外 (`codeNum` が 0 より大きいか等しい)、9.1.2 小節で規定される符号化ブロックパターンに対するマッピング処理が `codeNum` を入力として起動され、出力はシンタックス要素 `coded_block_pattern` に割り当てられる。

### G.9.2 変換係数レベルに対する代替CAVLC構文解析処理

この処理は、`entropy_coding_mode_flag` が 0 に等しい時、`ce(v)`に等しい記述子を持つ構文解析するシンタックス要素に対して起動される。

この処理への入力、シンタックス要素の値に対する要求、スライスデータからのビット、非ゼロ変換係数レベルの最大数 `maxNumCoeff`、変換係数レベルの現ブロックの輝度ブロックインデックス `luma4x4BlkIdx` または色差ブロックインデックス `chroma4x4BlkIdx`、`cb4x4BlkIdx` または `cr4x4BlkIdx` である。

この処理の出力は、ブロックインデックス `luma4x4BlkIdx` を持つ輝度ブロックまたはブロックインデックス `chroma4x4BlkIdx`、`cb4x4BlkIdx` または `cr4x4BlkIdx` を持つ色差ブロックの変換係数レベルを含む `coeffLevel` リストである。

処理は以下の順序付けられたステップで規定される：

1. `coeffLevel` リストにある 0 から `maxNumCoeff-1` までのインデックスを持つ全ての変換係数レベルが 0 に等しく設定される。
2. 非ゼロ変換係数レベルの総数 `TotalCoeff(coeff_token)` および追従 1 変換係数レベルの数 `TrailingOnes(coeff_token)` は、次のように `coeff_token` の構文解析により導出される。
  - a. 9.2.1 小節で規定される `coeff_token` の構文解析処理が起動され、出力は `TotalCoeff(coeff_token)`、`TrailingOnes(coeff_token)` と `nC` となる。
  - b. CAVLC 構文解析処理が `ChromaDCLevel` に対して起動されず、かつ `(scan_idx_end - scan_idx_start)` が 15 より小さい時、`Min(7, nC)` に等しく設定することにより `nC` が修正され、G.9.2.1 小節で規定される変換係数レベル総数および追従 1 に対する追加構文解析処理が、入力として `nC`、`TotalCoeff(coeff_token)` に等しく設定された `totalCoeffStart`、および `TrailingOnes(coeff_token)` に等しく設定された `trailingOnesStart` を用いて起動され、出力は `TotalCoeff(coeff_token)` と `TrailingOnes(coeff_token)` に割り当てられる。
3. `TotalCoeff(coeff_token)` が 0 より大きい時、以下が適用される。
  - a. 非ゼロ変換係数レベルは 9.2.2 小節で規定される `trailing_ones_sign_flag`、`level_prefix` および `level_suffix` の構文解析により導出される。
  - b. 各非ゼロ変換係数レベル前のゼロ変換係数レベルのランは G.9.2.2 小節で規定された `total_zeros` および `run_before` の構文解析により導出される。
  - c. レベルおよびラン情報は、9.2.4 小節で規定されるように、結合されて `coeffLevel` リストになる。

### G.9.2.1 変換係数レベル総数および追従 1 に対する追加構文解析処理

この処理への入力は、変数  $nC$ 、 $totalCoeffStart$ 、そして  $trailingOnesStart$  である。

この処理の出力は、変数  $totalCoeff$  および  $trailingOnes$  である。

9.2.1 小節で規定される  $coeff\_token$  の復号の結果となる  $totalCoeffStart$  の値は、0 から  $(scan\_idx\_end - scan\_idx\_start + 1)$  のそれぞれの値を含む範囲内でなければならない。

$invTotalCoeff(coeffTokenIdx)$  と  $invTrailingOnes(coeffTokenIdx)$  を、変数  $coeffTokenIdx$  を、変数  $nC$  が与えられる付表 G-9/JT-H264 で規定される変数  $nX$  および  $nY$  にそれぞれマップする関数とする。

変数  $targetCoeffTokenIdx$  は、変数  $nC$ 、 $nX=totalCoeffStart$ 、そして  $nY=trailingOnesStart$  が与えられた付表 G-9/JT-H264 により規定されるように導出される。

変数  $coeffTokenIdx$  は以下の擬似コードにより規定されるように導出される。

```
for(coeffTokenIdx=0,i=0;i<=targetCoeffTokenIdx;coeffTokenIdx++)
{
    if(invTotalCoeff(coeffTokenIdx)<min(4,scan_idx_end - scan_idx_start+2)&&
        invTrailingOnes(coeffTokenIdx)<min(17,scan_idx_end - scan_idx_start+2))
        i++;
}
```

変数  $totalCoeff$  は  $invTotalCoeff(coeffTokenIdx-1)$  に等しく設定され、変数  $trailingOnes$  は  $invTrailingOnes(coeffTokenIdx-1)$  に等しく設定される。

付表G-9/JT-H264 (nX, nY)のcoeffTokenIdxへのマッピング、およびその逆

(ITU-T H.264)

nY	nX	0 ≤ nC < 2	2 ≤ nC < 4	4 ≤ nC < 8
0	0	0	0	0
0	1	4	7	16
1	1	1	1	1
0	2	9	11	20
1	2	5	5	8
2	2	2	2	2
0	3	13	15	23
1	3	10	8	11
2	3	7	9	9
3	3	3	3	3
0	4	17	19	24
1	4	14	12	13
2	4	11	13	12

付表G-9/JT-H264 (nX, nY)のcoeffTokenIdxへのマッピング、およびその逆

(ITU-T H.264)

nY	nX	0 ≤ nC < 2	2 ≤ nC < 4	4 ≤ nC < 8
3	4	6	4	4
0	5	21	22	28
1	5	18	16	15
2	5	15	17	4
3	5	8	6	5
0	6	25	23	30
1	6	22	20	17
2	6	19	21	18
3	6	12	10	6
0	7	29	27	31
1	7	26	24	21
2	7	23	25	22
3	7	16	14	7
0	8	32	31	32
1	8	30	28	25
2	8	27	29	26
3	8	20	18	10
0	9	33	35	36
1	9	34	32	33
2	9	31	33	29
3	9	24	26	19
0	10	37	39	40
1	10	38	36	37
2	10	35	37	34
3	10	28	30	27
0	11	41	42	44
1	11	42	40	41
2	11	39	41	38
3	11	36	34	35
0	12	45	43	47
1	12	46	44	45
2	12	43	45	42
3	12	40	38	39
0	13	50	47	49

付表G-9/JT-H264 (nX, nY)のcoeffTokenIdxへのマッピング、およびその逆

(ITU-T H.264)

nY	nX	0 ≤ nC < 2	2 ≤ nC < 4	4 ≤ nC < 8
1	13	49	48	48
2	13	47	49	46
3	13	44	46	43
0	14	54	51	53
1	14	51	54	50
2	14	52	52	51
3	14	48	50	52
0	15	58	55	57
1	15	55	56	54
2	15	56	57	55
3	15	53	53	56
0	16	61	59	61
1	16	59	60	58
2	16	60	61	59
3	16	57	58	60

#### G.9.2.2 ラン情報に対する代替構文解析処理

この処理への入力は、スライスデータからのビット、非ゼロ変換係数レベルの数 TotalCoeff(coeff\_token)である。

この処理の出力は、run と呼ばれる非ゼロ変換係数レベルに先行するゼロ変換係数レベルのランのリストである。

maxCoeffを(scan\_idx\_end – scan\_idx\_start+1)に等しく設定した変数とする。

最初にインデックス i は 0 に等しく設定される。

変数 zerosLeft は、次の通り導出される。

- もし非ゼロ変換係数レベルの数 TotalCoeff(coeff\_token)が、非ゼロ変換係数レベルの最大数 maxCoeff に等しいならば、変数 zerosLeft は 0 に等しく設定される。
- それ以外(非ゼロ変換係数レベルの数 TotalCoeff(coeff\_token)が非ゼロ変換係数レベルの最大数 maxCoeff より小さい)、total\_zeros が復号され、zerosLeft がその値に等しく設定される。

total\_zeros の復号に使用される VLC は、次の通り導出される :

- もし maxCoeff が 4 に等しいならば、表 9-9(a)/JT-H264 に規定される VLC の 1 つが以下によって導出される tzVlcIndex と共に使用される。

$$tzVlcIndex = \min(3, TotalCoeff(coeff\_token) + 4 - maxCoeff) \quad (G-363)$$

- それ以外、もし  $\text{maxCoeff}$  が 4 より大きく、かつ 8 より小さいか、もしくは等しいならば、表 9-9(b)/JT-H264 に規定される VLC の 1 つが次式によって導出される  $\text{tzVlcIndex}$  と共に使用される。

$$\text{tzVlcIndex} = \min(7, \text{TotalCoeff}(\text{coeff\_token}) + 8 - \text{maxCoeff}) \quad (\text{G-364})$$

- それ以外、もし  $\text{maxCoeff}$  が 8 より大きく、かつ 15 より小さいならば、表 9-7/JT-H264 および表 9-8/JT-H264 からの VLC が次式によって導出される  $\text{tzVlcIndex}$  と共に使用される。

$$\text{tzVlcIndex} = \min(15, \text{TotalCoeff}(\text{coeff\_token}) + 16 - \text{maxCoeff}) \quad (\text{G-365})$$

それ以外 ( $\text{maxCoeff}$  が 15 よりも大きい等しい)、表 9-7/JT-H264 および表 9-8/JT-H264 からの VLC が  $\text{TotalCoeff}(\text{coeff\_token})$  と等しい  $\text{tzVlcIndex}$  と共に使用される。

次に以下の手続きが ( $\text{TotalCoeff}(\text{coeff\_token}) - 1$ ) 回、繰り返して適用される。

1. 変数  $\text{run}[i]$  が次の通り導出される。
  - もし  $\text{zerosLeft}$  がゼロより大きいならば、値  $\text{run\_before}$  が表 9-10/JT-H264 および  $\text{zerosLeft}$  に基づいて復号される。 $\text{run}[i]$  は  $\text{run\_before}$  に等しく設定される。
  - それ以外 ( $\text{zerosLeft}$  は 0 に等しい)、 $\text{run}[i]$  は 0 に等しく設定される。
2.  $\text{zerosLeft}$  から値  $\text{run}[i]$  が減算され、結果が  $\text{zerosLeft}$  に割り当てられる。減算結果は、0 以上でなければならない。
3. インデックス  $i$  は 1 加算される。

最後に  $\text{zerosLeft}$  の値が  $\text{run}[i]$  に割り当てられる。

### G.9.3 スケーラブル拡張内スライスデータに対する代替 CABAC 構文解析処理

G.9.3.1 小小節はスケーラブル拡張のスライスデータに対する代替 CABAC 構文解析処理における初期化処理を規定する。

G.9.3.2 小小節はスケーラブル拡張のスライスデータに対する代替 CABAC 構文解析処理におけるバイナリ化処理を規定する。

G.9.3.3 小小節はスケーラブル拡張のスライスデータに対する代替 CABAC 構文解析処理における復号処理を規定する。

#### G.9.3.1 初期化処理

この処理の出力は、 $\text{ctxIdx}$  によって索引される初期化された CABAC コンテキスト変数である。

付表 G-11/JT-H264 と付表 G-12/JT-H264 は、G.7.3.4.1 小小節と G.7.3.6 小小節のシンタックス要素  $\text{base\_mode\_flag}$ 、 $\text{motion\_prediction\_flag\_l0}$ 、 $\text{motion\_prediction\_flag\_l1}$ 、そして  $\text{residual\_prediction\_flag}$  に割り当てられたコンテキスト変数の初期化に用いられる変数  $n$  と  $m$  の値を含んでいる。G.7.3.4.1 小小節と G.7.3.6 小小節における他の全てのシンタックス要素に対して、9.3.1 小節で規定されるコンテキスト変数の初期化処理が適用される。

それぞれのコンテキスト変数において、2 つの変数  $\text{pStateIdx}$  と  $\text{valMPS}$  が初期化される。初期化のために  $\text{pStateIdx}$  と  $\text{valMPS}$  に割り当てられる 2 つの値が、等式 7-28 で得られる  $\text{SliceQP}_Y$  から導出される。与えられ

た2つの表のエントリ (m,n) で、次が適用される。

1.  $preCtxState = Clip3(1, 126, ((m * Clip3(0, 51, SliceQP_Y)) \gg 4) + n)$
2. `if(preCtxState <= 63){`  
`pStateIdx = 63 - preCtxState`  
`valMPS = 0`  
`}else{`  
`pStateIdx = preCtxState - 64`  
`valMPS = 1`  
`}`

付表 G-10/JT-H264 では、各スライスタイプ EI、EP、および EB に対する初期化に必要な ctxIdx が記載される。また、初期化に必要な m と n の値を含む表の番号が記載される。EP および EB スライスにおいて、初期化は cabac\_init\_idc シンタックス要素の値にも依存する。シンタックス要素名が初期化処理に影響しないことに注意すること。

**付表G-10/JT-H264 初期化処理の各スライスタイプにおけるctxIdxとシンタックス要素の関連 (ITU-T H.264)**

	シンタックス要素	表	スライスタイプ		
			EI	EP	EB
macroblock_layer_in_scalable_extension()	base_mode_flag	付表G-11/JT-H264	1024-1026	1024-1026	1024-1026
mb_pred_in_scalable_extension() and sub_mb_pred_in_scalable_extension()	motion_prediction_flag_l0	付表G-12/JT-H264		1027	1027
	motion_prediction_flag_l1	付表G-12/JT-H264		1027	1027
residual_in_scalable_extension()	residual_prediction_flag	付表G-12/JT-H264		1028-1029	1028-1029

**付表G-11/JT-H264 1024から1026のctxIdxにおける変数mとnの値 (ITU-T H.264)**

ctxIdx	EI スライス		cabac_init_idc値 (EP、EBスライス)					
			0		1		2	
	m	n	m	n	m	n	m	n
1024	-14	138	0	75	0	75	0	75
1025	-22	140	2	65	2	65	2	65
1026	-11	99	2	59	2	59	2	59

付表G-12/JT-H264 1027から1029のctxIdxにおける変数mとnの値  
(ITU-T H.264)

ctxIdx	cabac_init_idc値					
	0		1		2	
	m	n	m	n	m	n
1027	-6	67	-6	67	-6	67
1028	-23	104	-23	104	-23	104
1029	-35	106	-35	106	-35	106

### G.9.3.2 バイナリ化処理

この処理への入力、シンタックス要素に対する要求である。

この処理の出力は、シンタックス要素のバイナリ化、maxBinIdxCtx、ctxIdxOffset および bypassFlag である。

付表 G-13/JT-H264 で与えられる、コンテキストインデックスオフセット(ctxIdxOffset)変数の特定の値および maxBinIdxCtx 変数の特定の値は、シンタックス要素の各バイナリ化またはバイナリ化部分に関連する。

変数 bypassFlag は 0 に等しく設定される。

コンテキストインデックス ctxIdx の可能な値は、1024 から 1029 のそれぞれの値を含む範囲内である。ctxIdxOffset に割り当てられる値は、シンタックス要素の対応するバイナリ化またはバイナリ化部分に割り当てられた ctxIdx の範囲の下側の値を規定する。

付表G-13/JT-H264 シンタックス要素と関連付けられたバイナリ化のタイプ、maxBinIdxCtxおよび  
ctxIdxOffset  
(ITU-T H.264)

シンタックス要素	バイナリ化のタイプ	maxBinIdxCtx	ctxIdxOffset
base_mode_flag	FL, cMax=1	0	1024
motion_prediction_flag_10, motion_prediction_flag_11	FL, cMax=1	0	1027
residual_prediction_flag	FL, cMax=1	0	1028

### G.9.3.3 復号処理フロー

この処理への入力、G.9.3.2 小節で規定されるような、要求されたシンタックス要素のバイナリ化、maxBinIdxCtx、bypassFlag および ctxIdxOffset である。

この処理の出力は、シンタックス要素値である。

この処理は、各シンタックス要素に対して、ビット記号列中の各ビットがどのように構文解析されるかを規定する。

各ビットを構文解析後、結果であるビット記号列は、シンタックス要素のバイナリ化の全ての bin 記号列と比較され、次が適用される。

- もしビット記号列が bin 記号列の 1 つに等しいならば、対応するシンタックス要素値が出力である。
- それ以外（ビット記号列が bin 記号列の 1 つに等しくない）、次のビットが構文解析される。

各 bin を構文解析する間、変数 binIdx は、最初の bin に対して 0 に等しく設定されて始まり 1 ずつ加算される。

各 bin の構文解析は次の 2 つの順序付けられたステップで規定される。

1. 与えられた binIdx、maxBinIdxCtx および ctxIdxOffset で、G.9.3.3.1 小小小節で規定されるように ctxIdx が導出される。
2. 与えられた ctxIdx で、G.9.3.3.2 小小小節で規定されるようにビットストリームから bin の値が復号される。

#### G.9.3.3.1 ctxIdx の導出処理

この処理への入力は、binIdx、maxBinIdxCtx および ctxIdxOffset である。

この処理の出力は、ctxIdx である。

付表 G-14 / JT-H264 は、シンタックス要素 base\_mode\_flag、motion\_prediction\_flag\_10、motion\_prediction\_flag\_11、および residual\_prediction\_flag において全ての ctxIdxOffset 値に対する ctxIdx 増分 (ctxIdxInc) の binIdx への割り当てを示す。

特定の binIdx で使用される ctxIdx は、ctxIdxOffset と ctxIdxInc の和であり、それは付表 G-14 / JT-H264 で見つけられる。binIdx に対して付表 G-14 / JT-H264 または表 9-39 / JT-H264 に 1 より多くの値がリストされている時は、binIdx に対する ctxIdxInc の割り当て処理が対応する表エントリの括弧内に与えられる節でさらに規定される。

付表 G-14 / JT-H264 で “na” とラベル付けられた全てのエントリは、対応する ctxIdxOffset に対して生起しない binIdx 値に対応する。

**付表G-14/JT-H264 シンタックス要素base\_mode\_flagおよびresidual\_prediction\_flagに関連する  
ctxIdxOffset値に対するctxIdxIncのbinIdxへの割り当て  
(ITU-T H.264)**

ctxIdxOff set	binIdx						
	0	1	2	3	4	5	>= 6
<b>1024</b>	0,1,2 (G.9.3.3.2.1小小小節)	na	na	na	na	na	na
<b>1027</b>	0	na	na	na	na	na	na
<b>1028</b>	0,1 (G.9.3.3.2.2小小小節)	na	na	na	na	na	na

### G.9.3.3.2 隣接シンタックス要素を用いるctxIdxIncの割り当て処理

G.9.3.3.2.1 小節は、シンタックス要素 base\_mode\_flag に対する ctxIdxInc の導出処理を規定する。

G.9.3.3.2.2 小節は、シンタックス要素 residual\_prediction\_flag に対する ctxIdxInc の導出処理を規定する。

#### G.9.3.3.2.1 シンタックス要素base\_mode\_flagに対するctxIdxIncの導出処理

この処理の出力は、ctxIdxInc である。

6.4.10.1 小節で規定される隣接マクロブロックのための導出処理が起動され、その出力が mbAddrA と mbAddrB に割り当てられる。

変数 condTermFlagN (N は A または B) が、次の通り導出されるとする。

- もし mbAddrN が利用可能であり、かつマクロブロック mbAddrN に対する base\_mode\_flag が 1 に等しいならば、condTermFlagN は 0 に等しく設定される。
- それ以外 (mbAddrN が利用不可であるか、またはマクロブロック mbAddrN に対する base\_mode\_flag が 0 に等しい)、condTermFlagN は 1 に等しく設定される。

変数 ctxIdxInc は、以下で導出される。

$$\text{ctxIdxInc} = \text{condTermFlagA} + \text{condTermFlagB} \quad (\text{G-366})$$

#### G.9.3.3.2.2 シンタックス要素residual\_prediction\_flagに対するctxIdxIncの導出処理

この処理の出力は、ctxIdxInc である。

base\_mode\_flag に従って、次が適用される。

- もし base\_mode\_flag が 1 に等しいならば、ctxIdxInc は 0 に等しく設定される。
- それ以外 (base\_mode\_flag が 0 に等しい)、ctxIdxInc は 1 に等しく設定される。

## G.10 プロファイルとレベル

### G.10.1 プロファイル

#### G.10.1.1 スケーラブル基本プロファイル (Scalable Baseline)

スケーラブル基本プロファイルに適合するビットストリームは、次の制約に従わなくてはならない：

- a) G.8.8.2 小節で規定されるベースレイヤビットストリームは、次の制約に従わなくてはならない。

記 以下の i) から vi) の項で規定されるベースレイヤ制約に加えて、ベースレイヤストリームのピクチャパラメータセットに対するシンタックス要素 constrained\_intra\_pred\_flag の値は、以下の k) 項で規定されるように制約される。

- i) A.2.1 小節で規定される基本プロファイルの全ての制約に従わなければならない。
- ii) シーケンスパラメータセットは 66 に等しい profile\_idc を有すべきである。シーケンスパラメータセットは 77 もしくは 88 に等しい profile\_idc を有してもよい。シーケンスパラメ

- ータセットは 66、77、または 88 以外の値に等しい `profile_idc` を有してはならない。
- iii) シーケンスパラメータセットは、1 に等しい `constraint_set0_flag`、`constraint_set1_flag`、および `constraint_set2_flag` を有さなければならない。
  - iv) ピクチャパラメータセットは、0 に等しい `num_slice_groups_minus1` を有さなければならない。
  - v) ピクチャパラメータセットは、0 に等しい `redundant_pic_cnt_present_flag` を有さなければならない。
  - vi) 任意スライス順序は許可されない。
- b) I、P、EI、EP、および EB のスライスだけが存在しなければならない。
  - c) SVC シーケンスパラメータセットは 1 に等しい `chroma_format_idc` を有さなければならない。
  - d) SVC シーケンスパラメータセットは 0 に等しい `bit_depth_luma_minus8` を有さなければならない。
  - e) SVC シーケンスパラメータセットは 0 に等しい `bit_depth_chroma_minus8` を有さなければならない。
  - f) SVC シーケンスパラメータセットは 0 に等しい `separate_colour_plane_flag` を有さなければならない。
  - g) SVC シーケンスパラメータセットは 0 に等しい `qpprime_y_zero_transform_bypass_flag` を有さなければならない。
  - h) SVC シーケンスパラメータセットは 1 に等しい `frame_mbs_only_flag` を有さなければならない。
  - i) ピクチャパラメータセットは、0 から 7 のそれぞれの値を含む範囲の `num_slice_groups_minus1` を有さなければならない。
  - j) 存在する時、ピクチャパラメータセットは 2 に等しい `slice_group_map_type` を有さなければならない。
  - k) レイヤ表現識別子で規定される整数値のリストは、リスト `dqIdList` を出力とする G.8.1.1 小節で規定される処理を起動することによって導出される。変数 `numDQEntries` は、リスト `dqIdList` における要素の数に等しく設定される。`numDQEntries` が 1 よりも大きい時、 $i=0..(\text{numDQEntries}-2)$  の任意の要素 `dqIdList[i]` に対して、 $k=i..(\text{numDQEntries}-1)$  の `dqIdList[k]` によって規定される集合の中の `DQId` を持つ任意のレイヤ表現に対して `tcoeff_level_prediction_flag` が 0 に等しい時、`dqIdList[i]` に等しい `DQId` を持つレイヤ表現の符号化スライス NAL ユニットによって参照されるピクチャパラメータセットは、1 に等しい `constrained_intra_pred_flag` を有さなければならない。
  - l) 任意スライス順序は許可されない。
  - m) 0 より大きい `dependency_id` の存在する各々の値に対して、次の制約の内の 1 つに従わなければならない。
    - `ScaledRefLayerPicWidthInSamplesL` は `RefLayerPicWidthInSamplesL` に等しく、かつ `ScaledRefLayerPicHeightInSamplesL` は `RefLayerPicHeightInSamplesL` に等しい。
    - `ScaledRefLayerPicWidthInSamplesL` は  $(1.5 * \text{RefLayerPicWidthInSamples}_L)$  に等しく、かつ `ScaledRefLayerPicHeightInSamplesL` は  $(1.5 * \text{RefLayerPicHeightInSamples}_L)$  に等しい。

- $\text{ScaledRefLayerPicWidthInSamples}_L$  は  $(2 * \text{RefLayerPicWidthInSamples}_L)$  に等しく、かつ  $\text{ScaledRefLayerPicHeightInSamples}_L$  は  $(2 * \text{RefLayerPicHeightInSamples}_L)$  に等しい。
- n) 0 より大きい  $\text{dependency\_id}$  の存在する各々の値に対して、次の全ての制約に従わなければならない。
  - $(\text{ScaledRefLayerLeftOffset}\%16)$  は 0 に等しい。
  - $(\text{ScaledRefLayerTopOffset}\%16)$  は 0 に等しい。
- o) G.10.2 小節で規定されるレベル制約が満たされなければならない。

スケーラブル基本プロファイルへのビットストリームの適合は、 $\text{profile\_idc}$  が 83 であることによって規定される。

特定のレベルでスケーラブル基本プロファイルに適合する復号器は、全てのアクティブ SVC シーケンスパラメータセットが 83 に等しい  $\text{profile\_idc}$ 、あるいは (86 に等しい  $\text{profile\_idc}$  かつ 1 に等しい  $\text{constraint\_set0\_flag}$ )、あるいは (66、77、または 88 に等しい  $\text{profile\_idc}$ 、かつ 1 に等しい  $\text{constraint\_set0\_flag}$ 、かつ 1 に等しい  $\text{constraint\_set1\_flag}$ 、かつ 1 に等しい  $\text{constraint\_set2\_flag}$ ) を有し、かつ全てのアクティブ SVC シーケンスパラメータセットに対する  $\text{level\_idc}$  もしくは ( $\text{level\_idc}$  かつ  $\text{constraint\_set3\_flag}$ ) がその指定されたレベルより小さいか等しいレベルを表す、全てのビットストリームを復号できなければならない。

#### G.10.1.2 スケーラブルハイプロファイル (Scalable High)

スケーラブルハイプロファイルに適合するビットストリームは、次の制約に従わなくてはならない：

- a) G.8.8.2 小節で規定されるベースレイヤビットストリームは、次の制約に従わなくてはならない。
 

記 以下の i) から iii) の項で規定されるベースレイヤ制約に加えて、ベースレイヤストリームのピクチャパラメータセットに対するシンタックス要素  $\text{constrained\_intra\_pred\_flag}$  の値は、以下の j) 項で規定されるように制約される。

  - i) A.2.4 小節で規定されるハイプロファイルの全ての制約に従わなければならない。
  - ii) シーケンスパラメータセットは 100 に等しい  $\text{profile\_idc}$  を有すべきである。シーケンスパラメータセットは 66、77 もしくは 88 に等しい  $\text{profile\_idc}$  および 1 に等しい  $\text{constraint\_set1\_flag}$  を有してもよい。シーケンスパラメータセットは 66、77、88、または 100 以外の値に等しい  $\text{profile\_idc}$  を有してはならない。
  - iii) シンタックス要素  $\text{direct\_spatial\_mv\_pred\_flag}$  は 1 に等しくなければならない。
- b) I、P、B、EI、EP、および EB のスライスだけが存在しなければならない。
- c) SVC シーケンスパラメータセットは 1 に等しい  $\text{chroma\_format\_idc}$  を有さなければならない。
- d) SVC シーケンスパラメータセットは 0 に等しい  $\text{bit\_depth\_luma\_minus8}$  を有さなければならない。
- e) SVC シーケンスパラメータセットは 0 に等しい  $\text{bit\_depth\_chroma\_minus8}$  を有さなければならない。
- f) SVC シーケンスパラメータセットは 0 に等しい  $\text{separate\_colour\_plane\_flag}$  を有さなければならない。
- g) SVC シーケンスパラメータセットは 0 に等しい  $\text{qprime\_y\_zero\_transform\_bypass\_flag}$  を有さなければならない。

- h) ピクチャパラメータセットは 0 に等しい `redundant_pic_cnt_present_flag` を有さなければならない。
- i) ピクチャパラメータセットは 0 に等しい `num_slice_groups_minus1` を有さなければならない。
- j) レイヤ表現識別子で規定される整数値のリストは、リスト `dqIdList` を出力とする G.8.1.1 小節で規定される処理を起動することによって導出される。変数 `numDQEntries` は、リスト `dqIdList` における要素の数に等しく設定される。`numDQEntries` が 1 よりも大きい時、 $i=0..(\text{numDQEntries}-2)$  の任意の要素 `dqIdList[i]` に対して、 $k=i..(\text{numDQEntries}-1)$  の `dqIdList[k]` によって規定される集合の中の `DQId` を持つ任意のレイヤ表現に対して `tcoeff_level_prediction_flag` が 0 に等しい時、`dqIdList[i]` に等しい `DQId` を持つレイヤ表現の符号化スライス NAL ユニットによって参照されるピクチャパラメータセットは、1 に等しい `constrained_intra_pred_flag` を有さなければならない。
- k) 任意スライス順序は許可されない。
- l) G.10.2 小節で規定されるレベル制約が満たされなければならない。

スケーラブルハイプロファイルへのビットストリームの適合は、86 に等しい `profile_idc` によって規定される。

特定のレベルでスケーラブルハイプロファイルに適合する復号器は、全てのアクティブ SVC シーケンスパラメータセットが 86 に等しい `profile_idc`、あるいは (83 に等しい `profile_idc` かつ 1 に等しい `constraint_set1_flag`)、あるいは 77 に等しい `profile_idc`、あるいは 100 に等しい `profile_idc`、あるいは (83 または 86 に等しくない `profile_idc` かつ 1 に等しい `constraint_set1_flag`) を有し、かつ全てのアクティブ SVC シーケンスパラメータセットに対する `level_idc` あるいは (`level_idc` かつ `constraint_set3_flag`) が規定されたレベルよりも小さいか等しいレベルを表す、全てのビットストリームを復号できなければならない。

### G.10.1.3 スケーラブルハイ INTRA プロファイル (Scalable High Intra)

スケーラブルハイ INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない：

- a) G.8.8.2 小節で規定されるベースレイヤビットストリームは、次の制約に従わなくてはならない：
  - i) A.2.4 小節で規定されるハイプロファイルの全ての制約に従わなければならない。
  - ii) シーケンスパラメータセットは 100 に等しい `profile_idc` と 1 に等しい `constraint_set3_flag` を有すべきである。シーケンスパラメータセットは 66、77 もしくは 88 に等しい `profile_idc` および 1 に等しい `constraint_set1_flag` を有してもよい。シーケンスパラメータセットは 66、77、88、または 100 以外の値に等しい `profile_idc` を有してはならない。
- b) G.10.1.2 小節で規定されるスケーラブルハイプロファイルの全ての制約に従わなければならない。
- c) 全てのピクチャは IDR ピクチャでなければならない。
- d) SVC シーケンスパラメータセットは 0 に等しい `num_ref_frames` を有さなければならない。
- e) `vui_parameters_present_flag` が 1 に等しく、かつ `bitstream_restriction_flag` が 1 に等しい時、SVC シーケンスパラメータセットは 0 に等しい `num_reorder_frames` を有さなければならない。
- f) `vui_parameters_present_flag` が 1 に等しく、かつ `bitstream_restriction_flag` が 1 に等しい時、SVC シーケンスパラメータセットは 0 に等しい `max_dec_frame_buffering` を有さなければならない。

- g) ピクチャタイミング SEI メッセージは、(非 VCL NAL ユニットにより) ビットストリーム中に存在するか、あるいは、本標準に規定されていない他の手段によって同等に伝達されているかに関わらず、0 に等しい `dpb_output_delay` を有さなければならない。
- h) G.10.2 小節で規定されるレベル制約が満たされなければならない。

スケーラブルハイ INTRA プロファイルへのビットストリームの適合は、86 に等しい `profile_idc` とともに 1 に等しい `constraint_set3_flag` によって規定される。

特定のレベルでスケーラブルハイ INTRA プロファイルに適合する復号器は、全てのアクティブ SVC シーケンスパラメータセットが 86 もしくは 100 に等しい `profile_idc` かつ 1 に等しい `constraint_set3_flag` を有し、かつ全てのアクティブ SVC シーケンスパラメータセットに対する `level_idc` が規定されたレベルよりも小さいか等しいレベルを表す、全てのビットストリームを復号できなければならない。

## G.10.2 レベル

この小節の制約を表現するために、次が規定される。

- 初めのアクセスユニットをアクセスユニット 0 として、アクセスユニット `n` を復号順序で `n` 番目のアクセスユニットとする。
- ピクチャ `n` を、アクセスユニット `n` の主符号化ピクチャ、あるいは対応する復号ピクチャとする。

### G.10.2.1 スケーラブル基本、スケーラブルハイ、およびスケーラブルハイ INTRA プロファイルに対する共通のレベル限定

変数 `fR` は次の通り導出されるものとする。

- もしピクチャ `n` がフレームであるならば、`fR` は  $1 \div 172$  に設定される。
- それ以外 (ピクチャ `n` がフィールドである)、`fR` は  $1 \div (172 * 2)$  に設定される。

変数 `dqIdMax` は、アクセスユニットのレイヤ表現に対する `DQId` の最大値に等しく設定される。

変数 `refLayerDQId` は、`dqIdMax` に等しい `DQId` を持つレイヤ表現に対する `ref_layer_dq_id` の最大値に等しく設定される。

アクセスユニットに対してレイヤ表現識別子で規定される整数値のリストは、出力をリスト `dqIdList` とする G.8.1.1 小節で規定される処理を起動することにより導出される。変数 `numDQEntries` はリスト `dqIdList` 中の要素の数に等しく設定される。

変数 `dependentDId` は次の擬似コードによって導出される。

```
dependentDId=0
for(i=0;i<numDQEntries;i++)
    if((dqIdList[i]%16)==0)
        dependentDId++
(G-367)
```

変数 `svcPicSizeInMbs` は次のように導出される。

- もし `numDQEntries` が 3 より小さいならば、`dqIdMax` に等しい `DQId` を持つレイヤ表現に対する `svcPicSizeInMbs` は `PicSizeInMbs` に等しく設定される。

- それ以外 (numDQEntries が 2 より大きい)、svcPicSizeInMbs は次の順序付けられた手順を適用することによって導出される。

1. svcPicSizeInMbs は dqIdMax に等しい DQId を持つレイヤ表現に対して PicSizeInMbs に等しく設定される。
2. 変数 refLayerMbs は 0 に等しく設定される。
3.  $i=2..(\text{numDQEntries}-1)$  を持つ dqIdList[i] の各要素に対して、レイヤ表現において変数 PicSizeInMbs に等しくされる refLayerPicSizeInMbs と共に、変数 refLayerMbs は以下のように修正される。

$$\text{refLayerMbs} += \text{refLayerPicSizeInMbs} \quad (\text{G-368})$$

4. svcPicSizeInMbs は以下のように修正される。

$$\text{svcPicSizeInMbs} += (\text{refLayerMbs} + 1) \gg 1 \quad (\text{G-369})$$

特定のレベルにおいて、スケーラブル基本、スケーラブルハイ、あるいはスケーラブルハイ INTRA プロファイルに適合するビットストリームは、次の制約に従わなくてはならない。

- a) C.1.2 小節で規定されるような、CPB からのアクセスユニット  $n$  ( $n > 0$  である) の名目上の除去時刻は、 $t_{r,n}(n) - t_r(n-1)$  が  $\text{Max}(\text{svcPicSizeInMbs} \div \text{MaxMBPS}, \text{fR})$  より大きいかまたは等しいという制約を満たす。ここで、MaxMBPS はピクチャ  $n-1$  に適用される付表 A-1/JT-H264 で規定された値であり、svcPicSizeInMbs はピクチャ  $n-1$  に対して導出される。
- b) C.2.2 小節で規定されるような、DPB からの連続したピクチャの出力時刻の差分は、 $\Delta t_{o,dpb}(n) \geq \text{Max}(\text{svcPicSizeInMbs} \div \text{MaxMBPS}, \text{fR})$  という制約を満たす。ここで、MaxMBPS はピクチャ  $n$  に対する付表 A-1/JT-H264 で規定された値であり、svcPicSizeInMbs はピクチャ  $n$  から導出される。ピクチャ  $n$  は出力ピクチャであり、そのビットストリームの最後の出力ピクチャではないものと仮定される。
- c)  $\text{PicWidthInMbs} * \text{FrameHeightInMbs} \leq \text{MaxFS}$  である。ここで、MaxFS は付表 A-1/JT-H264 で規定される。PicWidthInMbs と FrameHeightInMbs は、dqIdMax に等しい DQId を持つレイヤ表現に対して導出された変数である。
- d)  $\text{PicWidthInMbs} \leq \text{Sqrt}(\text{MaxFS} * 8)$  である。ここで MaxFS は付表 A-1/JT-H264 で規定される。PicWidthInMbs と FrameHeightInMbs は、dqIdMax に等しい DQId を持つレイヤ表現に対して導出された変数である。
- e)  $\text{PicHeightInMbs} \leq \text{Sqrt}(\text{MaxFS} * 8)$  である。ここで MaxFS は付表 A-1/JT-H264 で規定される。PicWidthInMbs と FrameHeightInMbs は、dqIdMax に等しい DQId を持つレイヤ表現に対して導出された変数である。
- f)  $\text{max\_dec\_frame\_buffering} \leq \text{MaxDpbSize}$  である。ここで、MaxDpbSize は  $\text{Min}(1024 * \text{MaxDPB} / (\text{PicWidthInMbs} * \text{FrameHeightInMbs} * 384), 16)$  に等しく、MaxDPB は付表 A-1/JT-H264 で規定される。PicWidthInMbs と FrameHeightInMbs は dqIdMax に等しい DQId を持つレイヤ表現に対して導出された変数である。
- g) 垂直方向の動きベクトル成分の範囲は、輝度フレームサンプル単位で MaxVmvR を超えない。ここで、

MaxMvR は付表 A-1/JT-H264 で規定される。

- h) 水平方向の動きベクトルの範囲は、輝度サンプル単位で-2048 から 2047.75 のそれぞれの値を含む範囲を超えない。
- i) 復号順序で連続する 2 つのマクロブロックあたりの動きベクトルの数(あるスライスの最終マクロブロック、および復号順序で次のスライスの先頭マクロブロックからの合計にも適用される)は、MaxMvsPer2Mb を超えない。ここで、MaxMvsPer2Mb は付表 A-1/JT-H264 で規定される。各々のマクロブロックに対する動きベクトルの数は、G.8.1.5.6 小小小節で規定されるターゲットマクロブロックの復号処理終了の後の変数 MvCnt の値である。
- j) 任意のマクロブロックに対する macroblock\_layer( )と macroblock\_layer\_in\_scalable\_extension( )データのビット数は、128+RawMbBits よりも大きくない。entropy\_coding\_mode\_flag に依存して、macroblock\_layer( )データのビットは、次の通り数えられる。
- もし entropy\_coding\_mode\_flag が 0 に等しいならば、macroblock\_layer( )データのビット数は、あるマクロブロックに対する macroblock\_layer( )シンタックス構造内のビット数によって与えられる。
  - それ以外 (entropy\_coding\_mode\_flag が 1 に等しい)、あるマクロブロックに対する macroblock\_layer( )データのビット数は、そのマクロブロックに関連付けられる macroblock\_layer( )を構文解析する時に read\_bits(1)が 9.3.3.2.2 小小小節および 9.3.3.2.3 小小小節で呼ばれる回数によって与えられる。
- k) この小小節の最初において規定される変数 dependentDId は、3 を越えてはならない。
- l) refLayerDQId が 0 より大きいか等しい時、dqIdMax に等しい DQId を持つレイヤ表現に対して PicSizeInMbs を変数 PicSizeInMbs とし、G.8 節で規定される SVC 復号処理の終了の後の総称 currentVars の配列 mbType で規定する mbType とし、refLayerDQId に等しい DQId を持つレイヤ表現に対して G.8.1.3.1 小小小節と G.8.1.3.2 小小小節で規定されるレイヤ表現におけるベース復号処理の終了の後の総称 currentVars の配列 mbType で規定する refLayerMbType とし、次が適用される。
- もし PicSizeInMbs が 1620 より小さいか等しいならば、numIBLMbs を I\_BL に等しい mbType[mbAddr]をもつマクロブロック mbAddr の数、但し mbAddr=0..(PicSizeInMbs-1)、とし、numRefLayerIntraMbs を I\_PCM、I\_16x16、I\_8x8、I\_4x4、または I\_BL に等しい refLayerMbType[refLayerMbAddr]をもつマクロブロック refLayerMbAddr の数、但し refLayerMbAddr=0..(refLayerPicSizeInMbs-1)、として、次の制約に従わなければならない。  
$$\text{numRefLayerIntraMbs} \leq \text{Ceil}(1.5 * \text{numIBLMbs} * \text{refLayerPicSizeInMbs} / \text{PicSizeInMbs}) \quad (\text{G-370})$$
  - それ以外 (PicSizeInMbs が 1620 より大きい)、numIBLMbs を dqIdMax に等しい DQId を持つレイヤ表現のある特定のスライスにおいて mbType[mbAddr]が I\_BL に等しいマクロブロック mbAddr の数とし、refLayerSlices を、dqIdMax に等しい DQId を持つレイヤ表現の特定スライスの復号に要求される、refLayerDQId に等しい DQId を持つレイヤ表現のある特定のスライスの組とし、さらに numRefLayerIntraMbs を refLayerMbType[refLayerMbAddr]が I\_PCM、I\_16x16、I\_8x8、I\_4x4、または I\_BL に等しい refLayerSlices の組のスライスの中のマクロブロック refLayerMbAddr の数に等しくすると共に、等式 G-370 で規定された制約に従わなければならない。

- m) 特定のレイヤ表現に対する `ref_layer_dq_id` が 0 よりも大きい場合、特定のレイヤ表現によって参照される SVC シーケンスパラメータセットの `level_idc` の値は、`ref_layer_dq_id` に等しい DQId を有するレイヤ表現によって参照される SVC シーケンスパラメータセットに対する `level_idc` の値より大きいか等しくなければならない。

付表 A-1/JT-H264 は、各レベルの限定を規定する。付表 A-1/JT-H264 中の“-”でマークされたエントリは、対応する限定の欠如を表している。

ビットストリームが適合するレベルは、シンタックス要素 `level_idc` によって次の通り指示されなければならない。

- もし `level_idc` が 9 に等しいならば、指示されるレベルはレベル 1b である。
- それ以外 (`level_idc` が 9 に等しくない) 、`level_idc` は付表 A-1/JT-H264 で規定されるレベル番号の 10 倍の値に等しい。

### G.10.2.2 プロファイルに特有のレベル限定

変数 `numSVCslices` はアクセスユニットのレイヤ表現に対して DQId の最大値と等しく設定される。

アクセスユニットに対するレイヤ表現識別子で規定する整数値のリストは、出力をリスト `numDQEntries` とする G.8.1.1 小小節で規定される処理を起動することによって導出される。変数 `numDQEntries` はリスト `numDQEntries` の要素の数と等しく設定される。

変数 `numSVCslices` は次の擬似コードによって規定されるように導出される。

```

numSVCslices=0
for(i=0;i<numDQEntries;i++)
    numSVCslices+=dqIdList[i]に等しいDQIdを有するレイヤ表現内のスライス数

```

(G-371)

変数 `svcPicSizeInMbs` は G.10.2.1 小小節で規定されるように導出される。

以下の制約が規定される。

- a) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、アクセスユニット 0 の除去時刻は、ピクチャ 0 内のスライス数 `numSVCslices` は  $(\text{svcPicSizeInMbs} + \text{MaxMBPS} * (t_r(0) - t_{r,n}(0))) \div \text{SliceRate}$  より小さいかまたは等しいという制約を満たさなければならない。ここで、`SliceRate` はピクチャ 0 に適用される値であり、`MaxMBPS` は付表 A-1/JT-H264 で規定される値である。スケーラブル基本プロファイルにおいては、`SliceRate` は付表 G-15/JT-H264 で規定される。スケーラブルハイとスケーラブルハイ INTRA プロファイルにおいては、`SliceRate` は付表 A-4/JT-H264 で規定される。
- b) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、アクセスユニット `n` と `n-1` (`n > 0` である) との連続する除去時刻の間の差分は、ピクチャ `n` 内のスライス数 `numSVCslices` が  $\text{MaxMBPS} * (t_r(n) - t_r(n-1)) \div \text{SliceRate}$  より小さいかまたは等しいという制約を満たさなければならない。ここで、`SliceRate` はピクチャ `n` に適用される値であり、`MaxMBPS` は付表 A-1/JT-H264 で規定される値である。スケーラブル基本プロファイルにおいては、`SliceRate` は付表 G-15/JT-H264 で規定される。スケーラブルハイとスケーラブルハイ INTRA プロファイルにおいては、`SliceRate` は付表 A-4/JT-H264 で規定される。

- c) スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、付表 A-4/JT-H264 で規定されたレベルで、SVC シーケンスパラメータセットは 1 に等しい `direct_8x8_inference_flag` を有さなければならない。スケーラブル基本プロファイルに適合するビットストリームにおいては、SVC シーケンスパラメータセットは 1 に等しい `direct_8x8_inference_flag` を有さなければならない。
- d) スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、付表 A-4/JT-H264 で規定されたレベルで、SVC シーケンスパラメータセットは 1 に等しい `frame_mbs_only_flag` を有さなければならない。
- e) スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、全てのマクロブロック `mbAddr` とマクロブロックパーティション `mbPartIdx` において、G.8.1.5.1.1 小小小節で規定されることにより導出される `subMbType[mbAddr][mbPartIdx]` の値は、付表 A-4/JT-H264 内の  $8 \times 8$  とし示される `MinLumaBiPredSize` に対応するレベルで、`B_Bi_8x4`、`B_Bi_4x8` あるいは `B_Bi_4x4` と等しくなってはならない。スケーラブル基本プロファイルに適合するビットストリームにおいては、全てのマクロブロック `mbAddr` とマクロブロックパーティション `mbPartIdx` において、G.8.1.5.1.1 小小小節で規定されることにより導出される `subMbType[mbAddr][mbPartIdx]` の値は、`B_Bi_8x4`、`B_Bi_4x8` あるいは `B_Bi_4x4` と等しくなってはならない。
- f) スケーラブル基本プロファイルに適合するビットストリームにおいては、`P_8x8`、`P_8x8ref0` または `B_8x8` に等しいマクロブロックタイプで符号化されたマクロブロックでは各  $8 \times 8$  サブマクロブロックに対する単一参照ピクチャリスト（参照ピクチャリスト 0 または参照ピクチャリスト 1）で予測された輝度サンプル配列を生成するために使用される 8.4.2.2.1 小小小節で規定された処理の全ての起動に対して  $(xInt_{max} - xInt_{min} + 6) * (yInt_{max} - yInt_{min} + 6) \leq \text{MaxSubMbRectSize}$ 。ここで、 $\text{NumSubMbPart}(\text{sub\_mb\_type}) > 1$  であり、`MaxSubMbRectSize` は付表 G-15/JT-H264 で規定され、さらに、
- $xInt_{min}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $xInt_t$  の最小値
  - $xInt_{max}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $xInt_t$  の最大値
  - $yInt_{min}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $yInt_t$  の最小値
  - $yInt_{max}$  は、サブマクロブロックの全ての輝度サンプル予測中の、 $yInt_t$  の最大値
- g) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、VCL HRD パラメータに関しては、`SchedSelIdx` の少なくとも 1 つの値に対して、 $\text{BitRate}[\text{SchedSelIdx}] \leq \text{cpbBrVclFactor} * \text{MaxBR}$  かつ  $\text{CpbSize}[\text{SchedSelIdx}] \leq \text{cpbBrVclFactor} * \text{MaxCPB}$  である。ここで、`vcl_hrd_parameters_present_flag` が 1 に等しい場合、`cpbBrVclFactor` は付表 G-16/JT-H264 で規定され、 $\text{BitRate}[\text{SchedSelIdx}]$  は等式 E-37 で規定され、また  $\text{CpbSize}[\text{SchedSelIdx}]$  は等式 E-38 で規定される。`MaxBR` および `MaxCPB` は、それぞれ `cpbBrVclFactor` ビット/秒および `cpbBrVclFactor` ビットを単位として付表 A-1/JT-H264 で規定される。ビットストリームは、0 から `cpb_cnt_minus1` のそれぞれの値を含む範囲の `SchedSelIdx` の少なくとも 1 つの値に対して、これらの条件を満たさなければならない。
- h) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、NAL HRD パラメータに関しては、`SchedSelIdx` の少なくとも 1 つの

値 に 対 し て 、  $BitRate[SchedSelIdx] \leq cpbBrNalFactor * MaxBR$  か つ  $CpbSize[SchedSelIdx] \leq cpbBrNalFactor * MaxCPB$  である。ここで、`nal_hrd_parameters_present_flag` が 1 に等しい場合、`cpbBrNalFactor` は付表 G-16/JT-H264 で規定され、`BitRate[SchedSelIdx]` は等式 E-37 で規定され、また `CpbSize[SchedSelIdx]` は等式 E-38 で規定される。`MaxBR` および `MaxCPB` は、それぞれ `cpbBrNalFactor` ビット/秒および `cpbBrNalFactor` ビットを単位として付表 A-1/JT-H264 で規定される。ビットストリームは、0 から `cpb_cnt_minus1` の範囲の `SchedSelIdx` の少なくとも 1 つの値に対して、これらの条件を満たさなければならない。

- i) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、アクセスユニット 0 に対する `NumBytesInNALunit` 変数の和は、 $384 * (svcPicSizeInMbs + MaxMBPS * (t_r(0) - t_{r,n}(0))) \div MinCR$  より小さいかまたは等しい。ここで、`MaxMBPS` と `MinCR` はピクチャ 0 に適用される付表 A-1/JT-H264 で規定された値であり、`PicSizeInMbs` はピクチャ 0 内のマクロブロック数である。
- j) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、アクセスユニット  $n$  ( $n > 0$  である) に対する `NumBytesInNALunit` 変数の和は、 $384 * MaxMBPS * (t_r(n) - t_r(n-1)) \div MinCR$  より小さいかまたは等しい。ここで、`MaxMBPS` と `MinCR` はピクチャ  $n$  に適用される付表 A-1/JT-H264 で規定された値である。
- k) スケーラブル基本プロファイルに適合するビットストリームにおいては、2.1 およびそれ以下のレベルにおいては、シーケンスパラメータセットは 0 に等しい `entropy_coding_mode_flag` と 0 に等しい `transform_8x8_mode_flag` を有さなければならない。
- l) スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA プロファイルに適合するビットストリームにおいては、`dqIdMax` に等しい `DQId` における `PicSizeInMbs` が 1620 より大きい時、任意の符号化スライスのマクロブロックの数も `MaxFS/4` を超えてはならない。ここで `MaxFS` は付表 A-1/JT-H264 で規定される。

付表G-15/JT-H264 スケーラブル基本プロファイルのレベル限定  
(ITU-T H.264)

レベル番号	SliceRate	MaxSubMbRectSize
1	-	576
1b	-	576
1.1	-	576
1.2	-	576
1.3	-	576
2	-	576
2.1	22	576
2.2	22	576
3	22	576
3.1	60	1152
3.2	60	1152
4	60	1440
4.1	24	1440
4.2	24	1440
5	24	-
5.1	24	-

付表G-16/JT-H264 cpbBrVclFactorとcpbBrNalFactorの規定  
(ITU-T H.264)

プロファイル	cpbBrVclFactor	cpbBrNalFactor
スケーラブル基本、スケーラブルハイ、またはスケーラブルハイ INTRA	1250	1500

## G.11 バイトストリームフォーマット

付属資料 B における規定が適用される。

## G.12 仮想標準復号器

付属資料 C における規定が、シーケンスパラメータセットを SVC シーケンスパラメータセットと置き換えることにより適用される。

## G.13 付加拡張情報

付属資料 D における規定が、本節で規定される拡張と修正と共に適用される。

### G.13.1 SEIペイロードシンタックス

#### G.13.1.1 スケーラビリティ情報SEIメッセージシンタックス

scalability_info( payloadSize ) {	C	記述子
<b>temporal_id_nesting_flag</b>	5	u(1)
<b>priority_layer_info_present_flag</b>	5	u(1)
<b>priority_id_setting_flag</b>	5	u(1)
<b>num_layers_minus1</b>	5	ue(v)
for( i = 0; i <= num_layers_minus1; i++ ) {		
<b>priority_id[ i ]</b>	5	ue(v)
<b>dependency_id[ i ]</b>	5	u(6)
<b>discardable_flag[ i ]</b>	5	u(1)
<b>dependency_id[ i ]</b>	5	u(3)
<b>quality_id[ i ]</b>	5	u(4)
<b>temporal_id[ i ]</b>	5	u(3)
<b>sub_pic_layer_flag[ i ]</b>	5	u(1)
<b>sub_region_layer_flag[ i ]</b>	5	u(1)
<b>iroi_division_info_present_flag[ i ]</b>	5	u(1)
<b>profile_level_info_present_flag[ i ]</b>	5	u(1)
<b>bitrate_info_present_flag[ i ]</b>	5	u(1)
<b>frm_rate_info_present_flag[ i ]</b>	5	u(1)
<b>frm_size_info_present_flag[ i ]</b>	5	u(1)
<b>layer_dependency_info_present_flag[ i ]</b>	5	u(1)
<b>parameter_sets_info_present_flag[ i ]</b>	5	u(1)
<b>bitstream_restriction_info_present_flag[ i ]</b>	5	u(1)
<b>exact_inter_layer_pred_flag[ i ]</b>	5	u(1)
if( sub_pic_layer_flag[ i ]    iroi_division_info_present_flag[ i ] )		
<b>exact_sample_value_match_flag[ i ]</b>	5	u(1)
<b>layer_conversion_flag[ i ]</b>	5	u(1)
<b>layer_output_flag[ i ]</b>	5	u(1)
if( profile_level_info_present_flag[ i ] )		

<b>layer_profile_level_idc[ i ]</b>	5	u(24)
if( bitrate_info_present_flag[ i ] ) {		
<b>avg_bitrate[ i ]</b>	5	u(16)
<b>max_bitrate_layer[ i ]</b>	5	u(16)
<b>max_bitrate_layer_representation[ i ]</b>	5	u(16)
<b>max_bitrate_calc_window[ i ]</b>	5	u(16)
}		
if( frm_rate_info_present_flag[ i ] ) {		
<b>constant_frm_rate_idc[ i ]</b>	5	u(2)
<b>avg_frm_rate[ i ]</b>	5	u(16)
}		
if( frm_size_info_present_flag[ i ]		
<b>frm_width_in_mbs_minus1[ i ]</b>	5	ue(v)
<b>frm_height_in_mbs_minus1[ i ]</b>	5	ue(v)
}		
if( sub_region_layer_flag[ i ] ) {		
<b>base_region_layer_id[ i ]</b>	5	ue(v)
<b>dynamic_rect_flag[ i ]</b>	5	u(1)
if( !dynamic_rect_flag[ i ] ) {		
<b>horizontal_offset[ i ]</b>	5	u(16)
<b>vertical_offset[ i ]</b>	5	u(16)
<b>region_width[ i ]</b>	5	u(16)
<b>region_height[ i ]</b>	5	u(16)
}		
}		
if( sub_pic_layer_flag[ i ] )		
<b>roi_id[ i ]</b>	5	ue(v)
if( iroi_division_info_present_flag[ i ] ) {		
<b>iroi_grid_flag[ i ]</b>	5	u(1)
if( iroi_grid_flag[ i ] ) {		
<b>grid_width_in_mbs_minus1[ i ]</b>	5	ue(v)
<b>grid_height_in_mbs_minus1[ i ]</b>	5	ue(v)
} else {		
<b>num_rois_minus1[ i ]</b>	5	ue(v)
for( j = 0; j <= num_rois_minus1[ i ]; j++ ) {		
<b>first_mb_in_roi[ i ][ j ]</b>	5	ue(v)
<b>roi_width_in_mbs_minus1[ i ][ j ]</b>	5	ue(v)
<b>roi_height_in_mbs_minus1[ i ][ j ]</b>	5	ue(v)
}		
}		
}		
if( layer_dependency_info_present_flag[ i ] ) {		

<b>num_directly_dependent_layers[ i ]</b>	5	ue(v)
for( j = 0; j < num_directly_dependent_layers[ i ]; j++ )		
<b>directly_dependent_layer_id_delta_minus1[ i ][ j ]</b>	5	ue(v)
} else		
<b>layer_dependency_info_src_layer_id_delta[ i ]</b>	5	ue(v)
if( parameter_sets_info_present_flag[ i ] ) {		
<b>num_seq_parameter_set_minus1[ i ]</b>	5	ue(v)
for( j = 0; j <= num_seq_parameter_set_minus1[ i ]; j++ )		
<b>seq_parameter_set_id_delta[ i ][ j ]</b>	5	ue(v)
<b>num_subset_seq_parameter_set_minus1[ i ]</b>	5	ue(v)
for( j = 0; j <= num_subset_seq_parameter_set_minus1[ i ]; j++ )		
<b>subset_seq_parameter_set_id_delta[ i ][ j ]</b>	5	ue(v)
<b>num_pic_parameter_set_minus1[ i ]</b>	5	ue(v)
for( j = 0; j <= num_pic_parameter_set_minus1[ i ]; j++ )		
<b>pic_parameter_set_id_delta[ i ][ j ]</b>	5	ue(v)
} else		
<b>parameter_sets_info_src_layer_id_delta[ i ]</b>	5	ue(v)
if( bitstream_restriction_info_present_flag[ i ] ) {		
<b>motion_vectors_over_pic_boundaries_flag[ i ]</b>	5	u(1)
<b>max_bytes_per_pic_denom[ i ]</b>	5	ue(v)
<b>max_bits_per_mb_denom[ i ]</b>	5	ue(v)
<b>log2_max_mv_length_horizontal[ i ]</b>	5	ue(v)
<b>log2_max_mv_length_vertical[ i ]</b>	5	ue(v)
<b>num_reorder_frames[ i ]</b>	5	ue(v)
<b>max_dec_frame_buffering[ i ]</b>	5	ue(v)
}		
if( layer_conversion_flag[ i ] ) {		
<b>conversion_type_idc[ i ]</b>	5	ue(v)
for( j=0; j < 2; j++ ) {		
<b>rewriting_info_flag[ i ][ j ]</b>	5	u(1)
if( rewriting_info_flag[ i ][ j ] ) {		
<b>rewriting_profile_level_idc[ i ][ j ]</b>	5	u(24)
<b>rewriting_avg_bitrate[ i ][ j ]</b>	5	u(16)
<b>rewriting_max_bitrate[ i ][ j ]</b>	5	u(16)
}		
}		
}		
}		
if( priority_layer_info_present_flag ) {		
<b>pr_num_dId_minus1</b>	5	ue(v)
for( i = 0; i <= pr_num_dId_minus1; i++ ) {		
<b>pr_dependency_id[ i ]</b>	5	u(3)

<b>pr_num_minus1</b> [ i ]	5	ue(v)
for( j = 0; j <= pr_num_minus1; j++ ) {		
<b>pr_id</b> [ i ][ j ]	5	ue(v)
<b>pr_profile_level_idc</b> [ i ][ j ]	5	u(24)
<b>pr_avg_bitrate</b> [ i ][ j ]	5	u(16)
<b>pr_max_bitrate</b> [ i ][ j ]	5	u(16)
}		
}		
}		
if( priority_id_setting_flag ) {		
PriorityIdSettingUriIdx = 0		
do		
<b>priority_id_setting_uri</b> [ PriorityIdSettingUriIdx ]	5	b(8)
while( priority_id_setting_uri[ PriorityIdSettingUriIdx++ ] != 0 )		
}		
}		

#### G.13.1.2 レイヤ非存在SEIメッセージシンタックス

layers_not_present( payloadSize ) {	<b>C</b>	記述子
<b>num_layers</b>	5	ue(v)
for( i = 0; i < num_layers; i++ ) {		
<b>layer_id</b> [ i ]	5	u(8)
}		
}		

#### G.13.1.3 レイヤ依存変更SEIメッセージシンタックス

layer_dependency_change ( payloadSize ) {	<b>C</b>	記述子
<b>num_layers_minus1</b>	5	ue(v)
for( i = 0; i <= num_layers_minus1; i++ ) {		
<b>layer_id</b> [ i ]	5	u(8)

<b>layer_dependency_info_present_flag[ i ]</b>	5	u(1)
if (layer_dependency_info_present_flag[ i ]) {		
<b>num_directly_dependent_layers[ i ]</b>	5	ue(v)
for ( j = 0; j < num_directly_dependent_layers[ i ]; j++ )		
<b>directly_dependent_layer_id_delta_minus1[ i ][ j ]</b>	5	ue(v)
} else {		
<b>layer_dependency_info_src_layer_id_delta_minus1[ i ]</b>	5	ue(v)
}		
}		
}		

#### G.13.1.4 サブピクチャスケラブルレイヤSEIメッセージシンタックス

sub_pic_scalable_layer( payloadSize ) {	<b>C</b>	記述子
<b>layer_id</b>	5	ue(v)
}		

#### G.13.1.5 非要求レイヤ表現SEIメッセージシンタックス

non_required_layer_rep( payloadSize ) {	<b>C</b>	記述子
<b>num_info_entries_minus1</b>	5	ue(v)
for( i = 0; i <= num_info_entries_minus1; i++ ) {		
<b>entry_dependency_id[ i ]</b>	5	u(3)
<b>num_non_required_layer_rep_minus1[ i ]</b>	5	ue(v)
for(j = 0; j <= num_non_required_layer_rep_minus1[ i ]; j++ ) {		
<b>non_required_layer_rep_dependency_id[ i ][ j ]</b>	5	u(3)
<b>non_required_layer_rep_quality_id[ i ][ j ]</b>	5	u(4)
}		
}		
}		

#### G.13.1.6 優先レイヤ情報SEIメッセージシンタックス

priority_layer_info( payloadSize ) {	C	記述子
<b>pr_dependency_id</b>	All	u(3)
<b>num_priority_ids</b>	5	u(4)
for( i = 0; i < num_priority_ids; i++ ) {		
<b>alt_priority_id[ i ]</b>	5	u(6)
}		
}		

#### G.13.1.7 スケーラブルネスティングSEIメッセージシンタックス

scalable_nesting( payloadSize ) {	C	記述子
<b>all_layer_representations_in_au_flag</b>	5	u(1)
if( all_layer_representations_in_au_flag == 0 ) {		
<b>num_layer_representations_minus1</b>	5	ue(v)
for( i = 0; i <= num_layer_representations_minus1; i++ ) {		
<b>sei_dependency_id[ i ]</b>	5	u(3)
<b>sei_quality_id[ i ]</b>	5	u(4)
}		
<b>sei_temporal_id</b>	5	u(3)
}		
while( !byte_aligned( ) )		
<b>sei_nesting_zero_bit</b> /* 0に等しい */	5	f(1)
do		
sei_message( )	5	
while( more_rbsp_data( ) )		
}		

#### G.13.1.8 ベースレイヤ時間HRD SEIメッセージシンタックス

base_layer_temporal_hrd ( payloadSize ) {	C	記述子
<b>num_of_temporal_layers_in_base_layer_minus1</b>	5	ue(v)
for( i = 0; i < num_of_temporal_layers_in_base_layer_minus1; i++){		
<b>temporal_id[ i ]</b>	5	u(3)
<b>timing_info_present_flag[ i ]</b>	5	u(1)
if( timing_info_present_flag[ i ] ) {		
<b>num_units_in_tick[ i ]</b>	5	u(32)
<b>time_scale[ i ]</b>	5	u(32)
<b>fixed_frame_rate_flag[ i ]</b>	5	u(1)
}		
<b>nal_hrd_parameters_present_flag[ i ]</b>	5	u(1)
if( nal_hrd_parameters_present_flag[ i ] )		
hrd_parameters( )		
<b>vcl_hrd_parameters_present_flag[ i ]</b>	5	u(1)
if( vcl_hrd_parameters_present_flag[ i ] )		
hrd_parameters( )		
if( nal_hrd_parameters_present_flag[ i ]    vcl_hrd_parameters_present_flag[ i ] )		
<b>low_delay_hrd_flag[ i ]</b>	5	u(1)
<b>pic_struct_present_flag[ i ]</b>	5	u(1)
}		
}		

#### G.13.1.9 品質レイヤ整合性検査SEIメッセージシンタックス

quality_layer_integrity_check( payloadSize ) {	C	記述子
<b>num_info_entries_minus1</b>	5	ue(v)
for( i = 0; i <= num_info_entries_minus1; i++ ) {		
<b>entry_dependency_id[ i ]</b>	5	u(3)
<b>quality_layer_crc[ i ]</b>	5	u(16)
}		
}		

#### G.13.1.10 冗長ピクチャプロパティSEIシンタックス

redundant_pic_property( payloadSize ) {	C	記述子
<b>num_dId_minus1</b>	5	ue(v)
for( i = 0; i <= num_dId_minus1; i++ ) {		
<b>dependency_id[ i ]</b>	5	u(3)
<b>num_qId_minus1[ i ]</b>	5	ue(v)
for( j = 0; j <= num_qId_minus1[ i ]; j++ ) {		
<b>quality_id[ i ][ j ]</b>	5	u(4)
<b>num_redundant_pics_minus1[ i ][ j ]</b>	5	ue(v)
for( k = 0; k <= num_redundant_pics_minus1[ i ][ j ]; k++ ) {		
<b>redundant_pic_cnt_minus1[ i ][ j ][ k ]</b>	5	ue(v)
<b>pic_match_flag[ i ][ j ][ k ]</b>	5	u(1)
if( !pic_match_flag[ i ][ j ][ k ] ) {		
<b>mb_type_match_flag[ i ][ j ][ k ]</b>	5	u(1)
<b>motion_match_flag[ i ][ j ][ k ]</b>	5	u(1)
<b>residual_match_flag[ i ][ j ][ k ]</b>	5	u(1)
<b>intra_samples_match_flag[ i ][ j ][ k ]</b>	5	u(1)
}		
}		
}		
}		
}		

#### G.13.1.11 時間レベルゼロ依存表現インデックスSEIメッセージシンタックス

tl0_dep_rep_index( payloadSize ) {	C	記述子
<b>tl0_dep_rep_idx</b>	5	u(8)
<b>effective_idr_pic_id</b>	5	u(16)
}		

#### G.13.1.12 時間レベルスイッチング点SEIメッセージ

tl_switching_point( payloadSize ) {	C	記述子
<b>delta_frame_num</b>	5	se(v)
}		

#### G.13.2 SEIペイロードの意味

payloadType に基づいて、対応する SEI メッセージの意味は以下の通り拡張される。

- もし payloadType が 3、8、19、20 もしくは 22 に等しいならば、以下が適用される。
  - もし SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれないならば、それは 0 に等しい dependency\_id と quality\_id を持つ現アクセスユニットのレイヤ表現に適用される。
  - それ以外 (SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれる)、それは 0 から num\_layer\_representations\_minus1 のそれぞれの値を含む範囲にある任意の i に対して ((sei\_dependency\_id[i]<<4)+sei\_quality\_id[i]) に等しい DQId を持つ現アクセスユニットの全てのレイヤ表現に適用される。
- それ以外、もし payloadType が 2、6、7、9、10、11、12、13、14、15、16、17、18、21、23、25、27、34 もしくは 35 に等しいならば、以下が適用される。
  - もし SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれないならば、それは 0 に等しい dependency\_id を持つ現アクセスユニットの依存表現に適用される。
  - それ以外 (SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれる)、0 から num\_layer\_representations\_minus1 のそれぞれの値を含む範囲にある任意の i に対して sei\_quality\_id[i] の値は 0 でなければならず、それは 0 から num\_layer\_representations\_minus1 のそれぞれの値を含む範囲にある任意の i に対して sei\_quality\_id[i] に等しい dependency\_id を持つ現アクセスユニットの全ての依存表現に適用される。スケーラブルネスティング SEI メッセージ内に含まれている SEI メッセージに対して payloadType が 10 に等しい場合、サブシーケンス情報 SEI メッセージの sub\_seq\_layer\_num の意味は以下の通り修正される。  
**sub\_seq\_layer\_num** は現ピクチャのサブシーケンスレイヤ番号を規定する。現ピクチャが、復号順序で最初のピクチャが IDR ピクチャであるサブシーケンス中に存在する場合、sub\_seq\_layer\_num の値は 0 に等しくなければならない。非ペア参照フィールドに対して sub\_seq\_layer\_num の値は 0 に等しくなければならない。sub\_seq\_layer\_num は 0 から 255 のそれぞれを含む範囲でなければならない。
- それ以外、もし payloadType が 0 もしくは 1 に等しいならば、以下が適用される。
  - もし SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれないならば、以下が適用される。その SEI メッセージと、スケーラブルネスティング SEI メッセージ内に含まれない 0 もしくは 1 に等しい payloadType を持つ他の全ての SEI メッセージが、付属資料 C に従ってビットストリームの適合をチェックするためにバッファリング期間およびピクチャタイミング SEI メッセージとして使用され、2-9 章に規定された復号処理が使用される場合、そのビットストリームは本標準に適合していなければならない。
  - それ以外 (SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれる)、以下が適用される。その SEI メッセージと、sei\_temporal\_id、sei\_dependency\_id[i] および sei\_quality\_id[i] の同一の値を持つスケーラブルネスティング SEI メッセージ内に含まれる 0 もしくは 1 に等しい payloadType を持つ他の全ての SEI メッセージが、付属資料 C に従ってビットストリームの適合をチェックするためにバッファリング期間およびピクチャタイミング SEI メッセージとして使用される場合、sei\_temporal\_id に等しい tIdTarget、sei\_dependency\_id[i] に等しい dIdTarget および sei\_quality\_id[i] に等しい qIdTarget を用いて G.8.8.1 小節で規定されるビットストリーム抽出処理を起動することで得られるビットストリームは、本標準に適合していなければならない。
- それ以外 (残りの全ての payloadType の値)、対応する SEI メッセージの意味は拡張されない。

SEI NALユニットが24から35のそれぞれを含む範囲の payloadType を持った SEI メッセージを含む場合、それは 24 よりも小さい payloadType を持たないか、SEI メッセージも含んでおらず、その SEI NAL ユニットの最初の SEI メッセージは 24 から 35 のそれぞれを含む範囲の payloadType を持たなければならない。

スケーラブルネスティング SEI メッセージ (payloadType が 30 に等しい) が SEI NAL ユニットに存在する場合、その SEI NAL ユニット内にはその SEI メッセージのみが存在しなければならない。

### G.13.2.1 スケーラビリティ情報SEIメッセージの意味

スケーラビリティ情報 SEI メッセージはビットストリームのサブセットに対するスケーラビリティ情報を含む。これらのサブセットのそれぞれはスケーラブルレイヤと呼ばれ、レイヤ識別子に関連付けられる。スケーラブルレイヤは dependency\_id、quality\_id、および temporal\_id の同じ値を持った符号化スライス NAL ユニットの集合と、関連付けられた非 VCL NAL ユニットを表現する。スケーラブルレイヤを復号するために要求されるビットストリームのサブセットはスケーラブルレイヤ表現またはスケーラブルレイヤの表現と呼ばれる。ビットレート、フレームレート、および空間解像度のような情報は、以下で規定されるような各スケーラブルレイヤの表現のために示される。

各スケーラブルレイヤはレイヤ識別子と関連付けられる。レイヤ識別子のより大きな値は、より高いスケーラブルレイヤを示す。0 に等しいレイヤ識別子を持ったスケーラブルレイヤは、最も低いスケーラブルレイヤもしくはスケーラブルレイヤ 0 である。スケーラブルレイヤの復号はより高いか、スケーラブルレイヤとも独立だが、より低いスケーラブルレイヤに依存するかもしれない。それゆえ、最も低いスケーラブルレイヤは独立して復号でき、スケーラブルレイヤ 1 の復号はスケーラブルレイヤ 0 に依存するかもしれない、スケーラブルレイヤ 2 の復号は、スケーラブルレイヤ 0 および 1 に依存するかもしれない、などである。スケーラブルレイヤの復号はそのスケーラブルレイヤそれ自体と、そのスケーラブルレイヤが直接的もしくは間接的に依存する全てのより低いスケーラブルレイヤの存在を要求する。もし、INTER 予測または INTRA 予測に関わらずスケーラブルレイヤ B 内の 1 つ以上の符号化スライスを参照する、少なくとも 1 つの符号化スライスがスケーラブルレイヤ A 内に存在するならば、スケーラブルレイヤ A は直接的にスケーラブルレイヤ B に依存する。もし、スケーラブルレイヤ A が直接的にスケーラブルレイヤ B に依存しないが、スケーラブルレイヤ A の復号にレイヤ B の 1 つ以上の符号化スライスの存在を要求するならば、スケーラブルレイヤ A は間接的にレイヤ B に依存する。

記1 例えば、レイヤ識別子の値は(((dependency\_id<<4)+quality\_id)\*8+temporal\_id)に等しく割り当てられてもよい。

存在する時、スケーラビリティ情報 SEI メッセージは、全ての VCL NAL ユニットが 5 に等しい nal\_unit\_type もしくは 1 に等しい idr\_flag を持ったアクセスユニット内に現れなければならない。変更が 1 つ以上のレイヤ依存変更 SEI メッセージの存在によって示されるかもしれないレイヤ依存情報を例外として、スケーラビリティ情報 SEI メッセージの内容は、次のスケーラビリティ情報 SEI メッセージを含むアクセスユニットまでの、但しそれを除く、全てのアクセスユニットに適用される。現在の SEI メッセージが適用されるアクセスユニットの集合はターゲットアクセスユニット集合と呼ばれる。特定のスケーラブルレイヤに対するレイヤ依存情報は、num\_directly\_dependent\_layers、directly\_dependent\_layer\_id\_delta\_minus1、および layer\_dependency\_info\_src\_layer\_id\_delay シンタックス要素によって規定される。スケーラビリティ情報 SEI メッセージが、スケーラブルレイヤ A が直接的にも間接的にもスケーラブルレイヤ B に依存しないことを規定する場合、この関係はターゲットアクセスユニット集合全体に適用される。スケーラビリティ情報 SEI メッセージが、スケーラブルレイヤ A が直接的または間接的にスケーラブルレイヤ B に依存することを規定する場合、続くレイヤ依存変更 SEI メッセージは、この依存性がターゲットアクセスユニット集合のサブセットに適用されないことを規定するかもしれない。

**temporal\_id\_nesting\_flag** は、ターゲットアクセスユニット集合に対して INTER 予測が追加で制限されるかどうかを規定する。**temporal\_id\_nesting\_flag** に依存して、以下が適用される。

- もし **temporal\_id\_nesting\_flag** が 0 に等しいならば、追加の制約に従わなくて良い。
- それ以外 (**temporal\_id\_nesting\_flag** が 1 に等しい)、ターゲットアクセスユニット集合内に存在する **dependency\_id** と **quality\_id** の値の任意の組み合わせを持つレイヤ表現に対して、以下の制約に従わなければならない。

**temporal\_id** が **tIdA** に等しい各アクセスユニット **auA** に対して、復号順序でアクセスユニット **auB** に後続し、かつアクセスユニット **auA** に先行する、**temporal\_id** が **tIdC** に等しく、かつ **tIdC** が **tIdB** 以下であるアクセスユニット **auC** が存在する場合、**temporal\_id** が **tIdB** に等しく、かつ **tIdB** が **tIdA** 以下であるアクセスユニット **auB** が INTER 予測によって参照されてはならない。

記2 シンタックス要素 **temporal\_id\_nesting\_flag** は、時間方向の上位切り替え、すなわち特定の **temporal\_id tIdN** までの復号から、**temporal\_id tIdM>tIdN** までの復号に切替えること、が常に可能であることを示すために使用される。

**priority\_layer\_info\_present\_flag** が 1 に等しいことは、ターゲットアクセスユニット集合内に優先レイヤ情報 SEI メッセージが存在し、かつ現在のスケーラビリティ情報 SEI メッセージ内に優先レイヤの特性情報が存在することを規定する。**priority\_layer\_info\_present\_flag** が 0 に等しいことは、現在のスケーラビリティ情報 SEI メッセージ内に優先レイヤの特性情報が存在しないことを規定する。

**priority\_id\_setting\_flag** が 1 に等しいことは、シンタックス要素 **priority\_id\_setting\_uri** が存在し、かつ **priority\_id** 値の計算方法記述が分かっていることを規定する。**priority\_id\_setting\_flag** が 0 に等しいことは、シンタックス要素 **priority\_id\_setting\_uri** が存在しないことを規定する。

**num\_layers\_minus1** に 1 を加えたものは、ターゲットアクセスユニット集合内に含まれているスケーラブルレイヤの数を規定する。**num\_layers\_minus1** は、0 から 2047 のそれぞれを含む範囲内でなければならない。

**layer\_id[i]** は、この SEI メッセージで規定される第 *i* 番目のスケーラブルレイヤのレイヤ識別子を規定する。**layer\_id[i]** は、0 から 2047 のそれぞれを含む範囲内でなければならない。

これより本小小節の最後までは、現在の **layer\_id[i]** 値に等しいレイヤ識別子を持ったスケーラブルレイヤは現スケーラブルレイヤと呼ばれ、現スケーラブルレイヤの表現は現スケーラブルレイヤ表現と呼ばれる。

**priority\_id[i]** は、現スケーラブルレイヤ表現の全ての VCL NAL ユニットが、**priority\_id[i]** 以下の **priority\_id** の値を持つことを規定する。

**discardable\_flag[i]** が 1 に等しいことは、現スケーラブルレイヤの全ての VCL NAL ユニットが、1 に等しい **discardable\_flag** を持つことを規定する。**discardable\_flag[i]** が 0 に等しいことは、現スケーラブルレイヤが、0 に等しい **discardable\_flag** を持つ VCL NAL ユニットを含むかもしれないことを規定する。

**dependency\_id[i]**、**quality\_id[i]**、および **temporal\_id[i]** は、それぞれ、現スケーラブルレイヤ中の NAL ユニットの **dependency\_id**、**quality\_id** および **temporal\_id** に等しい。現スケーラブルレイヤの全ての VCL NAL ユニットにおいて **dependency\_id** の値は同一である。現スケーラブルレイヤの全ての VCL NAL ユニットにおいて **quality\_id** の値は同一である。現スケーラブルレイヤの全ての VCL NAL ユニットにおいて **temporal\_id** の値は同一である。

**sub\_pic\_layer\_flag[i]** が 1 に等しいことは、現スケーラブルレイヤがサブピクチャから構成されることを規定する。それぞれのサブピクチャは、レイヤ表現の符号化スライスの真のサブセット（真部分集合）から構

成される。sub\_pic\_layer\_flag[i]が 0 に等しいことは、この SEI メッセージ内の現スケーラブルレイヤに関連付けられたサブピクチャが無いことを規定する。

サブピクチャで構成されているスケーラブルレイヤは、サブピクチャレイヤとも呼ばれる。サブピクチャレイヤのスケーラブルレイヤ表現に含まれないかなる符号化スライスも、サブピクチャレイヤの復号のために INTER 予測またはレイヤ間予測において参照されてはならない。

記3 各サブピクチャからスケーラブルレイヤへのマッピングは、サブピクチャスケーラブルレイヤ情報 SEI メッセージによって示されるかもしれない。

sub\_region\_layer\_flag[i]が 1 に等しいことは、現スケーラブルレイヤが、スケーラビリティ情報 SEI メッセージの意味において後で規定される領域で構成されていることを規定する。sub\_region\_layer\_flag[i]が 0 に等しいことは、現スケーラブルレイヤが、スケーラビリティ情報 SEI メッセージの意味において後で規定される領域を表現しないことを規定する。

iroi\_division\_info\_present\_flag[i]が 1 に等しいことは、現スケーラブルレイヤによって表現される領域がマクロブロック境界に沿ってインタラクティブ着目領域 (IROI) と呼ばれる複数の矩形着目領域に分割され、IROI 分割情報がスケーラビリティ情報 SEI メッセージ内に明示的に示されることを規定する。iroi\_division\_info\_present\_flag[i]が 0 に等しいことは、現スケーラブルレイヤに対する IROI 分割情報がスケーラビリティ情報 SEI メッセージ内に存在しないことを規定する。

IROI の領域のサブセットを覆う全ての符号化スライスが IROI を形成する。IROI のスケーラブルレイヤ表現を含まないかなる符号化スライスも、その IROI の復号のために INTER 予測もしくはレイヤ間予測において参照されてはならない。

profile\_level\_info\_present\_flag[i]が 1 に等しいことは、現スケーラブルレイヤ表現に適用できる profile\_idc、constraint\_set0\_flag、constraint\_set1\_flag、constraint\_set2\_flag、constraint\_set3\_flag、および level\_idc が、その後で規定される layer\_profile\_level\_idc[i]の値により規定されることを規定する。

profile\_level\_info\_present\_flag[i]が 0 に等しい場合、現スケーラブルレイヤ表現に適用できる profile\_idc、constraint\_set0\_flag、constraint\_set1\_flag、constraint\_set2\_flag、constraint\_set3\_flag、および level\_idc は、スケーラビリティ情報 SEI メッセージ内に存在しない。

bitrate\_info\_present\_flag[i]が 1 に等しいことは、現スケーラブルレイヤに対するビットレート情報がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。bitrate\_info\_present\_flag[i]が 0 に等しいことは、現スケーラブルレイヤに対するビットレート情報がスケーラビリティ情報 SEI メッセージ内に存在しないことを規定する。

frm\_rate\_info\_present\_flag[i]が 1 に等しいことは、現スケーラブルレイヤに対するフレームレート情報がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。frm\_rate\_info\_present\_flag[i]が 0 に等しいことは、現スケーラブルレイヤに対するフレームレート情報がスケーラビリティ情報 SEI メッセージ内に存在しないことを規定する。

frm\_size\_info\_present\_flag[i]が 1 に等しいことは、現スケーラブルレイヤに対するフレームサイズ情報がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。frm\_size\_info\_present\_flag[i]が 0 に等しいことは、現スケーラブルレイヤに対するフレームサイズ情報がスケーラビリティ情報 SEI メッセージ内に存在しないことを規定する。

layer\_dependency\_info\_present\_flag[i]が 1 に等しいことは、現スケーラブルレイヤに対するレイヤ依存情

報がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。  
layer\_dependency\_info\_present\_flag[i]が0に等しいことは、layer\_dependency\_info\_src\_layer\_id\_delta[i]がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。

parameter\_sets\_info\_present\_flag[i]が1に等しいことは、現スケーラブルレイヤの VCL NAL ユニットによって参照される、シーケンスパラメータセットおよびサブセットシーケンスパラメータセットの seq\_parameter\_set\_id、およびピクチャパラメータセットの pic\_parameter\_set\_id 値が、スケーラビリティ情報 SEI メッセージ内に存在することを規定する。parameter\_sets\_info\_present\_flag[i]が0に等しいことは、parameter\_sets\_info\_src\_layer\_id\_delta[i]がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。

bitstream\_restriction\_info\_present\_flag[i]が1に等しいことは、現スケーラブルレイヤに対するビットストリーム制限情報がスケーラビリティ情報 SEI メッセージ内に存在することを規定する。bitstream\_restriction\_info\_present\_flag[i]が0に等しいことは、現スケーラブルレイヤに対するビットストリーム制限情報がスケーラビリティ情報 SEI メッセージ内に存在しないことを規定する。

exact\_inter\_layer\_pred\_flag[i]が1に等しいことは、そのスケーラブルレイヤ表現が符号化されたときのレイヤ間予測参照信号と比較して、現スケーラブルレイヤ表現を復号するときのレイヤ間予測参照信号に不一致が無いことを規定する。exact\_inter\_layer\_pred\_flag[i]が0に等しいことは、不一致が存在するかもしれないことを規定する。

記4 符号化と復号に対するレイヤ間予測参照信号の不一致は、例えば、現スケーラブルレイヤ表現中に含まれない、符号化処理のINTER予測のために使用された0より大きいquality\_idを持ったNALユニットが原因で存在するかもしれない。

exact\_sample\_value\_match\_flag[i]は、現スケーラブルレイヤ中のサブピクチャレイヤまたは IROI のサンプル値の正確な一致を規定する。

sub\_pic\_layer\_flag[i]が1に等しい場合、以下が適用される。

- scalLayerRepA を、サブピクチャレイヤの符号化スライスを含むターゲットアクセスユニット集合の全てのレイヤ表現の集合とする。
- exact\_sample\_value\_match\_flag[i]が1に等しいことは、サブピクチャレイヤ表現の復号において再構成されるそれぞれのサンプル値は、scalLayerRepA 内の全ての符号化スライスが復号されるときに再構成される同じ位置のサンプル値と正確に同じであることを規定する。
- exact\_sample\_value\_match\_flag[i]が0に等しいことは、サブピクチャレイヤ表現の復号において再構成されるそれぞれのサンプル値は、scalLayerRepA 内の全ての符号化スライスが復号されるときに再構成される同じ位置のサンプル値と同じである必要がないことを規定する。

各 IROI に対して、iroi\_division\_info\_present\_flag[i]が1に等しい場合、以下が適用される。

- scalLayerRepB を、IROI の符号化スライスを含むターゲットアクセスユニット集合の全てのレイヤ表現の集合とする。
- exact\_sample\_value\_match\_flag[i]が1に等しいことは、IROI の復号において再構成されるそれぞれのサンプル値は、scalLayerRepB 内の全ての符号化スライスが復号されるときに再構成される同じ位置のサンプル値と正確に同じであることを規定する。
- exact\_sample\_value\_match\_flag[i]が0に等しいことは、IROI の復号において再構成されるそれぞれのサンプル値は、scalLayerRepB 内の全ての符号化スライスが復号されるときに再構成される同じ位置のサン

プル値と同じである必要がないことを規定する。

記5 ターゲットアクセスユニット集合内の全てのスライスにおいて `disable_deblocking_filter_idc` が 2 に等しい場合、`exact_sample_value_match_flag` は 1 に等しくなくてはならない。

`layer_conversion_flag[i]` が 1 に等しいことは、現スケーラブルレイヤの表現が、付属資料 A に規定された 1 つ以上のプロファイルに適合するアクセスユニットの代替セットに変換可能であり、かつ現スケーラブルレイヤ表現と正確に同じ複合結果を与え、かつ完全な再構成と再符号化をすることなくこの変換が行えることを規定する。`layer_conversion_flag[i]` が 0 に等しいことは、現スケーラブルレイヤ表現のそのような変換が、可能でないかもしれないことを規定する。

`layer_output_flag[i]` が 1 に等しいことは、現スケーラブルレイヤの表現に対する復号結果が、出力を意図されていることを規定する。`layer_output_flag[i]` が 0 に等しいことは、現スケーラブルレイヤの表現に対する復号結果が、出力を意図されていないことを規定する。

記6 0 に等しい `layer_output_flag[i]` を持つスケーラブルレイヤ表現に対する復号結果は、視覚的品質が低いために不適当かもしれない。

`layer_profile_level_idc[i]` は、現スケーラブルレイヤの表現のプロファイルとレベル適合を規定する。`layer_profile_level_idc[i]` は、もしこれらのシンタックス要素が付属資料 A および付属資料 G で規定される現スケーラブルレイヤの表現のプロファイルおよびレベル適合性を規定するために使用されたなら、`profile_idc`、`constraint_set0_flag`、`constraint_set1_flag`、`constraint_set2_flag`、`constraint_set3_flag`、`reserved_zero_4bits`、および `level_idc` で構成された 3 バイトの正確な複写である。

`avg_bitrate[i]` は、現スケーラブルレイヤの表現の平均ビットレートを規定する。秒あたりのビットを単位として現スケーラブルレイヤの表現の平均ビットレートは、以下で既定される関数 `BitRateBPS()` を用いて `BitRateBPS(avg_bitrate[i])` で与えられる。

$$\text{BitRateBPS}(x) = (x \& 0x3fff) * 10^{(2+(x \gg 14))} \quad (\text{G-372})$$

現スケーラブルレイヤの表現の全ての NAL ユニットの計算において考慮される。平均ビットレートは本標準の付属資料 C で規定されるアクセスユニットの除去時刻に従って導出される。以降、`bTotal` を、スケーラビリティ情報 SEI メッセージに続き（アクセスユニット内の現スケーラブルレイヤの表現の NAL ユニットのビットを含む）、かつ（もし存在するならば）スケーラビリティ情報 SEI メッセージ、もしくは（そうでないならば）ストリームの終了を含む（復号順序で）次のアクセスユニットに先行する、現スケーラブルレイヤの表現の全ての NAL ユニットのビット数とする。`t1` は現アクセスユニットの除去時刻（秒単位）であり、`t2` は（もし存在するならば）次のスケーラビリティ情報 SEI メッセージ、または（それ以外）ストリームの終了の前の、最新のアクセスユニット（復号順序）の除去時刻（秒単位）である。

`avg_bitrate[i]` の値を規定する `x` を用いて、以下の条件が満たされなければならない。

- もし `t1` が `t2` に等しくないならば、以下の条件が真でなければならない。

$$(x \& 0x3fff) == \text{Round}(bTotal \div ((t_2 - t_1) * 10^{(2+(x \gg 14))})) \quad (\text{G-373})$$

- それ以外（`t1` が `t2` に等しい）、以下の条件が真でなければならない。

$$(x \& 0x3fff) == 0 \quad (\text{G-374})$$

`max_bitrate_layer[i]` は、現スケーラブルレイヤの最大ビットレートを規定する。現スケーラブルレイヤの最大ビットレートは、ビット毎秒単位で、等式 G-372 で規定される `BitRateBPS()` を用いて

BitRateBPS(max\_bitrate\_layer[i])によって与えられる。現スケーラブルレイヤの最大ビットレートは、max\_bitrate\_calc\_window[i]によって規定される時間窓に基づいて計算される。

max\_bitrate\_layer\_representation[i]は、現スケーラブルレイヤの表現の最大ビットレートを規定する。現スケーラブルレイヤの表現の最大ビットレートは、ビット毎秒単位で、等式 G-372 で規定される BitRateBPS() を用いて BitRateBPS(max\_bitrate\_layer\_representation[i])によって与えられる。現スケーラブルレイヤの表現の最大ビットレートは、max\_bitrate\_calc\_window[i]によって規定される時間窓に基づいて計算される。

max\_bitrate\_calc\_window[i]は、1/100 秒単位で、時間窓の長さを規定し、これに基づいて max\_bitrate\_layer[i] および max\_bitrate\_layer\_representation[i]が計算される。

constant\_frm\_rate\_idc[i]は、現スケーラブルレイヤの表現の出力フレームレートが固定かどうかを規定する。もし、以下で規定される avg\_frm\_rate[i]の値が、スケーラブルレイヤ表現の全時間区分にわたって固定であるならば、フレームレートは固定である。そうでなければ、フレームレートは固定ではない。constant\_frm\_rate\_idc[i]が 0 に等しいことは、フレームレートが固定でないことを規定する。constant\_frm\_rate\_idc[i]が 1 に等しいことは、フレームレートが固定であることを規定し、constant\_frm\_rate\_idc[i]が 2 に等しいことは、フレームレートが固定かもしれないしそうでないかもしれないことを規定する。constantFrameRate の値は 0 から 2 のそれぞれを含む範囲内でなければならない。

avg\_frm\_rate[i]は、256 秒あたりのフレーム単位で、現スケーラブルレイヤの表現の平均フレームレートを規定する。avg\_frm\_rate[i]の意味は、accurate\_statistics\_flag が 1 に等しい場合、サブシーケンスレイヤの範囲内の NAL ユニット集合が現スケーラブルレイヤの表現の NAL ユニット集合によって置き換えられる点を除き、サブシーケンスレイヤ特性 SEI メッセージ内の average\_frame\_rate の意味と同一である。

frm\_width\_in\_mbs\_minus1[i]および frm\_height\_in\_mbs\_minus1[i]は、それぞれ、現スケーラブルレイヤの表現の幅と高さを規定する。frame\_mbs\_only\_flag が 0 に等しい場合、(frm\_height\_in\_mbs\_minus1[i]+1)%2 は 0 に等しくなくてはならない。

sub\_pic\_layer\_flag[i]が 0 に等しい場合、現スケーラブルレイヤの表現の復号ピクチャの幅は frm\_width\_in\_mbs\_minus1[i]+1 マクロブロックに等しく、現スケーラブルレイヤの表現の復号ピクチャの高さは((frm\_height\_in\_mbs\_minus1[i]+1)/(1+field\_pic\_flag))マクロブロックに等しいことを規定する。

sub\_pic\_layer\_flag[i]が 1 に等しい場合、現スケーラブルレイヤの表現の復号サブピクチャの幅は frm\_width\_in\_mbs\_minus1[i]+1 マクロブロックに等しく、現スケーラブルレイヤの表現の復号サブピクチャの高さは((frm\_height\_in\_mbs\_minus1[i]+1)/(1+field\_pic\_flag))マクロブロックに等しいことを規定する。

変数 FrmWidthInMbs[i]は、(frm\_width\_in\_mbs\_minus1[i]+1)に等しく導出される。変数 FrmHeightInMbs[i]は、frm\_height\_in\_mbs\_minus1[i]+1 に等しく導出される。変数 FrmSizeInMbs[i]は、マクロブロック単位のフレームサイズで FrmWidthInMbs[i]\*FrmHeightInMbs[i]に等しく導出される。

base\_region\_layer\_id[i]は、スケーラブルレイヤ sIA のレイヤ識別子を規定する。現スケーラブルレイヤ表現の復号ピクチャの領域は、スケーラブルレイヤ sIA のスケーラブルレイヤ表現の対応する復号ピクチャの領域のサブセットを表現する。現スケーラブルレイヤは、スケーラブルレイヤ sIA 内の領域を表現し、スケーラブルレイヤ sIA はベース領域と呼ばれる。

dynamic\_rect\_flag[i]が 1 に等しいことは、現スケーラブルレイヤ表現によって表現される領域が、動的に変化するベース領域の矩形サブセットであることを規定する。dynamic\_rect\_flag[i]が 1 に等しいことは、現スケーラブルレイヤによって表現される領域が固定の矩形サブセットであり、horizontal\_offset[i]、

vertical\_offset[i]、region\_width[i]、および vertical\_offset[i]によって規定されることを規定する。

**horizontal\_offset[i]、vertical\_offset[i]、region\_width[i]、および region\_height[i]**は、現スケーラブルレイヤによって表現される領域の位置とサイズを規定する。frame\_mbs\_only\_flag が 0 に等しい場合、(vertical\_offset[i]+1)%2 および(region\_height[i]+1)%2 は、共に 0 に等しくなければならない。

sub\_pic\_layer\_flag[i]が 0 に等しい場合、現スケーラブルレイヤ表現の復号ピクチャによって表現される矩形領域の左上画素の水平および垂直オフセットは、ベース領域の左上画素との相対で、ベース領域内の復号フレームの輝度サンプル単位で、それぞれ horizontal\_offset[i]および vertical\_offset[i]/(1+field\_pic\_flag)に等しく、現スケーラブルレイヤ表現の復号ピクチャによって表現される矩形領域の幅と高さは、ベース領域内の復号フレームの輝度サンプル単位で、それぞれ region\_width[i]および region\_height[i]/(1+field\_pic\_flag)に等しい。

sub\_pic\_layer\_flag[i]が 1 に等しい場合、現スケーラブルレイヤ表現の復号サブピクチャによって表現される矩形領域の左上画素の水平および垂直オフセットは、ベース領域の左上画素との相対で、ベース領域内の復号フレームの輝度サンプル単位で、それぞれ horizontal\_offset[i]および vertical\_offset[i]/(1+field\_pic\_flag)に等しく、現スケーラブルレイヤ表現の復号ピクチャによって表現される矩形領域の幅と高さは、ベース領域内の復号フレームの輝度サンプル単位で、それぞれ region\_width[i]および region\_height[i]/(1+field\_pic\_flag)に等しい。

**roi\_id[i]**は、現スケーラブルレイヤにより表現される領域の着目領域識別子を示す。

layerIdA が layerIdB よりも小さい layerIdA および layerIdB を、それぞれ、スケーラブルレイヤ A および B のレイヤ識別子とし、そのそれぞれは 1 に等しい sub\_pic\_layer\_flag[i]を持ち、roiIdA および roiIdB は、それぞれスケーラブルレイヤ A および B の着目領域識別子とする。roiIdA は roiIdB よりも大きくてはならない。

**iroi\_grid\_flag[i]**は、現スケーラブルレイヤに対して IROI 分割情報がどのようにして符号化されるかを規定する。iroi\_grid\_flag[i]が 1 に等しいことは、現スケーラブルレイヤに対する全ての IROI が同じサイズを持つことを規定する。iroi\_grid\_flag[i]が 0 に等しいことは、現スケーラブルレイヤに対する全ての IROI が異なるサイズを持つかもしれないことを規定する。

**grid\_width\_in\_mbs\_minus1[i]および grid\_height\_in\_mbs\_minus1[i]**は、現スケーラブルレイヤの IROI グリッドのサイズを規定する。frame\_mbs\_only\_flag が 0 に等しい場合、(grid\_height\_in\_mbs\_minus1+1)%2 は 0 に等しくなければならない。

現スケーラブルレイヤの IROI グリッドの復号サブピクチャの幅および高さは、それぞれ、以下は例外として、マクロブロック単位で grid\_width\_in\_mbs\_minus1[i]+1 および (grid\_height\_in\_mbs\_minus1[i]+1)/(1+field\_pic\_flag)に等しくなければならない。

- (FrmWidthInMbs[i]%(grid\_width\_in\_mbs\_minus1[i]+1))が 0 に等しくない時、現スケーラブルレイヤの最右グリッド列の IROI の復号サブピクチャは、(FrmWidthInMbs[i]%(grid\_width\_in\_mbs\_minus1[i]+1))マクロブロックに等しい幅を持っている。
- (FrmHeightInMbs[i]%(grid\_height\_in\_mbs\_minus1[i]+1))が 0 に等しくない時、現スケーラブルレイヤの最下グリッド行の IROI の復号サブピクチャは、(FrmHeightInMbs[i]%(grid\_height\_in\_mbs\_minus1[i]+1))/(1+field\_pic\_flag)マクロブロックに等しい幅を持っている。

**num\_rois\_minus1[i]**に1を加えたものは、現スケーラブルレイヤ中の IROI の数を規定する。

**first\_mb\_in\_roi[i][j]**は、現スケーラブルレイヤの第j番目の IROI の最初のマクロブロックのマクロブロックアドレスを規定する。**first\_mb\_in\_roi[i][j]**の値は、現スケーラブルレイヤに属する符号化スライスの **first\_mb\_in\_slice** シンタックス要素に等しくなければならない、それは第j番目の IROI の左上マクロブロックを含んでいる。jに等しくないkで、i、j、kの各組み合わせに対して、**first\_mb\_in\_roi[i][j]**は**first\_mb\_in\_roi[i][k]**と異ならなければならない。

**roi\_width\_in\_mbs\_minus1[i][j]**および**roi\_height\_in\_mbs\_minus1[i][j]**は、現スケーラブルレイヤの IROI のサイズを規定する。**frame\_mbs\_only\_flag**が0に等しい場合、**(roi\_height\_in\_mbs\_minus1+1)%2**は0に等しくなければならない。

現スケーラブルレイヤの第j番目の IROI の復号サブピクチャの幅および高さは、それぞれ、マクロブロック単位で、**roi\_width\_in\_mbs\_minus1[i][j]+1** および**(roi\_height\_in\_mbs\_minus1[i][j]+1)/(1+field\_pic\_flag)**に等しい。

**num\_directly\_dependent\_layers[i]**は、現スケーラブルレイヤが直接的に依存するスケーラブルレイヤの数を規定する。**num\_directly\_dependent\_layers**の値は、0から255のそれぞれを含む範囲内でなければならない。

**directly\_dependent\_layer\_id\_delta\_minus1[i][j]**に1を加えたものは、**layer\_id[i]**と、現スケーラブルレイヤが直接的に依存する第j番目のスケーラブルレイヤのレイヤ識別子の差を規定する。第j番目の直接的に依存するスケーラブルレイヤのレイヤ識別子は、**(layer\_id[i]-directly\_dependent\_layer\_id\_delta\_minus1[i][j]-1)**に等しい。

**layer\_dependency\_info\_src\_layer\_id\_delta[i]**が0よりも大きいことは、現スケーラブルレイヤが、**(layer\_id[i]-layer\_dependency\_info\_src\_layer\_id\_delta[i])**に等しいレイヤ識別子を持つレイヤと同じレイヤ依存情報を持つことを規定する。**layer\_dependency\_info\_src\_layer\_id\_delta[i]**が0に等しいことは、現スケーラブルレイヤのレイヤ依存情報がその SEI メッセージ内に存在しないことを規定する。

**num\_seq\_parameter\_set\_minus1[i]**に1を加えたものは、現スライスレイヤ表現の VCL NAL ユニットによって参照される異なるシーケンスパラメータセットの数を規定する。

**seq\_parameter\_set\_id\_delta[i][j]**は、もしjが0に等しいならば、現スケーラブルレイヤの表現の復号に対して要求されるあらゆるシーケンスパラメータセットの **seq\_parameter\_set\_id** の最小値を規定する。そうでないならば (jが0よりも大きい)、**seq\_parameter\_set\_id\_delta[i][j]**は、現スケーラブルレイヤの表現の復号に対し、第j番目の要求されたシーケンスパラメータセットの **seq\_parameter\_set\_id** の値と、要求された第(j-1)番目のシーケンスパラメータセットの **seq\_parameter\_set\_id** の値の差を規定する。シーケンスパラメータセットは、**seq\_parameter\_set\_id** の値の昇順で論理的に整列される。

**num\_subset\_seq\_parameter\_set\_minus1[i]**に1を加えたものは、現スケーラブルレイヤ表現の VCL NAL ユニットによって参照される異なるサブセットシーケンスパラメータセットの数を規定する。

**subset\_seq\_parameter\_set\_id\_delta[i][j]**は、もしjが0に等しいならば、現スケーラブルレイヤの表現の復号に対して要求されるあらゆるサブセットシーケンスパラメータセットの **seq\_parameter\_set\_id** の最小値を規定する。そうでないならば (jが0よりも大きい)、**subset\_seq\_parameter\_set\_id\_delta[i][j]**は、現スケーラブルレイヤの表現の復号に対し、第j番目の要求されたサブセットシーケンスパラメータセットの **seq\_parameter\_set\_id** の値と、要求された第(j-1)番目のサブセットシーケンスパラメータセットの **seq\_parameter\_set\_id** の値の差を規定する。サブセットシーケンスパラメータセットは、**seq\_parameter\_set\_id** の値の昇順で論理的に順序付けされる。

**num\_pic\_parameter\_set\_minus1[i]**に1を加えたものは、現スケーラブルレイヤ表現のVCL NALユニットによって参照される異なるピクチャパラメータセットの数を規定する。

**pic\_parameter\_set\_id\_delta[i][j]**は、もしjが0に等しいならば、現スケーラブルレイヤの表現の復号に対して要求されるあらゆるピクチャパラメータセットの**pic\_parameter\_set\_id**の最小値を規定する。そうでないならば(jが0よりも大きい)、**pic\_parameter\_set\_id\_delta[i][j]**は、現スケーラブルレイヤの表現の復号に対し、第j番目の要求されたピクチャパラメータセットの**pic\_parameter\_set\_id**の値と、要求された第(j-1)番目のピクチャパラメータセットの**pic\_parameter\_set\_id**の値の差を規定する。ピクチャパラメータセットは、**pic\_parameter\_set\_id**の値の昇順で論理的に順序付けされる。

**parameter\_sets\_info\_src\_layer\_id\_delta[i]**が0よりも大きいことは、シーケンスパラメータセットとサブシーケンスパラメータセットの**seq\_parameter\_set\_id**の値と、現スケーラブルレイヤ表現のVCL NALユニットによって参照されるピクチャパラメータセットの**pic\_parameter\_set\_id**の値が、(**layer\_id[i]**-**parameter\_sets\_info\_src\_layer\_id\_delta[i]**)に等しいレイヤ識別子を持つスケーラブルレイヤ表現に対するこれらと等しいことを規定する。

**parameter\_sets\_info\_src\_layer\_id\_delta[i]**が0に等しいことは、現スケーラブルレイヤのVCL NALユニットによって参照されるシーケンスパラメータセットおよびサブセットシーケンスパラメータセットの**seq\_parameter\_set\_id**の値とピクチャパラメータセットの**pic\_parameter\_set\_id**の値が、そのSEIメッセージに存在しないことを規定する。

**motion\_vectors\_over\_pic\_boundaries\_flag[i]**は、現スケーラブルレイヤに対し、E.2.1 小節で規定される**motion\_vectors\_over\_pic\_boundaries\_flag**の値を規定する。**motion\_vectors\_over\_pic\_boundaries\_flag[i]**シンタックス要素が存在しない場合、現スケーラブルレイヤに対する**motion\_vectors\_over\_pic\_boundaries\_flag**の値は1に等しく推定されなければならない。

**max\_bytes\_per\_pic\_denom[i]**は、現スケーラブルレイヤに対し、E.2.1 小節で規定される**max\_bytes\_per\_pic\_denom**の値を規定する。**max\_bytes\_per\_pic\_denom[i]**シンタックス要素が存在しない場合、現スケーラブルレイヤに対する**max\_bytes\_per\_pic\_denom**の値は2に等しく推定されなければならない。

**max\_bits\_per\_mb\_denom[i]**は、現スケーラブルレイヤに対し、E.2.1 小節で規定される**max\_bits\_per\_mb\_denom**の値を規定する。**max\_bits\_per\_mb\_denom[i]**が存在しない場合、現スケーラブルレイヤに対する**max\_bits\_per\_mb\_denom**の値は1に等しく推定されなければならない。

**log2\_max\_mv\_length\_horizontal[i]**および**log2\_max\_mv\_length\_vertical[i]**は、現スケーラブルレイヤに対し、E.2.1 小節で規定される**log2\_max\_mv\_length\_horizontal**および**log2\_max\_mv\_length\_vertical**の値を規定する。**log2\_max\_mv\_length\_horizontal[i]**が存在しない場合、現スケーラブルレイヤに対する**log2\_max\_mv\_length\_horizontal**および**log2\_max\_mv\_length\_vertical**の値は16に等しく推定されなければならない。

記7 復号された垂直もしくは水平動きベクトル要素の最大絶対値は、付属資料Aで規定されるプロファイルおよびレベルの制約によってもまた制限される。

**num\_reorder\_frames[i]**は、現スケーラブルレイヤに対し、E.2.1 小節で規定される**num\_reorder\_frames**の値を規定する。**num\_reorder\_frames[i]**の値は0から**max\_dec\_frame\_buffering**のそれぞれを含む範囲内でなければならない。**num\_reorder\_frames[i]**シンタックス要素が存在しない場合、現スケーラブルレイヤに対する**num\_reorder\_frames**の値は、**max\_dec\_frame\_buffering**に等しく推定されなければならない。

**max\_dec\_frame\_buffering[i]**は、現スケーラブルレイヤに対し、E.2.1 小節で規定される

max\_dec\_frame\_buffering の値を規定する。max\_dec\_frame\_buffering[i] の値は、num\_ref\_frames[i] から MaxDpbSize (A.3.1 または A.3.2 小節で規定される) までのそれぞれを含む範囲内でなければならない。max\_dec\_frame\_buffering[i] シンタックス要素が存在しない場合、現スケーラブルレイヤに対する max\_dec\_frame\_buffering の値は MaxDpbSize に等しく推定されなければならない。

conversion\_type\_idc[i] が 0 に等しいことは、現スケーラブルレイヤ表現の、1 に等しい no\_inter\_layer\_pred\_flag を持つものを除いた全ての符号化スライスに対して tcoeff\_level\_prediction\_flag が 1 に等しいことを規定する。

conversion\_type\_idc[i] が 1 に等しいことは、現スケーラブルレイヤによって参照されるシーケンスパラメータセットの slice\_header\_restriction\_flag が 1 に等しく、かつ現スケーラブルレイヤ表現の、0 に等しい no\_inter\_layer\_pred\_flag を持った全ての VCL NAL ユニットの slice\_skip\_flag が 1 に等しいことを規定する。

conversion\_type\_idc[i] が 2 に等しいことは、現スケーラブルレイヤによって参照されるシーケンスパラメータセットの slice\_header\_restriction\_flag が 1 に等しく、かつ現スケーラブルレイヤの全ての VCL NAL ユニットの no\_inter\_layer\_pred\_flag が 1 に等しいことを規定する。

conversion\_type\_idc[i] は、0 から 2 までのそれぞれを含む範囲内でなければならない。

conversion\_type\_idc[i] が 0 に等しいことは、付属資料 A で規定された 1 つ以上のプロファイルに適合し、現スケーラブルレイヤ表現と正確に同じ復号結果を与える、現スケーラブルレイヤ表現をアクセスユニットの代替セットに変換する方法が無規定であるが、シンタックス要素 rewriting\_profile\_level\_idc[i][j]、rewriting\_avg\_bitrate[i][j]、および rewriting\_max\_bitrate[i][j] によって規定される情報が、存在するならば、正しいことを規定する。

conversion\_type\_idc[i] が 1 に等しいことは、現スケーラブルレイヤ表現に、逐次順に以下の操作を適用することによって得られるアクセスユニットの代替セットが、付属資料 A で規定される 1 つ以上のプロファイルに適合することを規定する。

1. 20 に等しい nal\_unit\_type および 1 に等しい skip\_slice\_flag を持つ全ての NAL ユニットの除去する
2. 14 に等しい nal\_unit\_type を持つ全ての NAL ユニットの除去する
3. 20 に等しい nal\_unit\_type を持つ NAL ユニットからの NAL ユニットヘッダ SVC 拡張を除去する
4. 20 に等しい nal\_unit\_type と 1 に等しい idr\_flag とを持つ NAL ユニットに対し、nal\_unit\_type を 5 に設定する
5. 20 に等しい nal\_unit\_type と 0 に等しい idr\_flag とを持つ NAL ユニットに対し、nal\_unit\_type を 1 に設定する
6. 全ての SEI NAL ユニットの除去する
7. nal\_unit\_type が 1 または 5 を持つ NAL ユニットによって参照される全てのピクチャパラメータセット NAL ユニットに対して、83 または 86 に等しい profile\_idc を持つサブセットシーケンスパラメータセット NAL ユニットの seq\_parameter\_set\_id の値を seq\_parameter\_set\_id に等しく変更する
8. 7 に等しい nal\_unit\_type を持つ NAL ユニットの除去する
9. 15 に等しい nal\_unit\_type を持つ全ての NAL ユニットに対し、nal\_unit\_type を 7 に等しく設定し、シンタックス構造 seq\_parameter\_set\_data( ) の後、rbbsp\_trailing\_bits( ) シンタックス構造の前の全ての

シンタックス要素を除去し、存在する場合、`profile_idc` から開始する 3 バイトを `rewriting_profile_level_idc[i][entropy_coding_mode_flag]` によって規定されるように置換し、そして RBSP 追従ビットを適切に変更する。

`conversion_type_idc[i]` が 2 に等しいことは、現スケーラブルレイヤ表現に、逐次順に以下の操作を適用することによって得られる代替ビットストリームが、付属資料 A で規定される 1 つ以上のプロファイルに適合することを示す。

1. 14 に等しい `nal_unit_type` を持つ全ての NAL ユニットの除去する
2. 20 に等しい `nal_unit_type` を持つ NAL ユニットからの NAL ユニットヘッダ SVC 拡張を除去する
3. 20 に等しい `nal_unit_type` と 1 に等しい `idr_flag` とを持つ NAL ユニットに対し、`nal_unit_type` を 5 に設定する
4. 20 に等しい `nal_unit_type` と 0 に等しい `idr_flag` とを持つ NAL ユニットに対し、`nal_unit_type` を 1 に設定する
5. 全ての SEI NAL ユニットの除去する
6. 15 に等しい `nal_unit_type` を持つ全ての NAL ユニットに対し、`nal_unit_type` を 7 に等しく設定し、シンタックス構造 `seq_parameter_set_data()` の後、`rbsp_trailing_bits()` シンタックス構造の前の全てのシンタックス要素を除去し、存在する場合、`profile_idc` から開始する 3 バイトを `rewriting_profile_level_idc[i][entropy_coding_mode_flag]` によって規定されるように置換し、そして RBSP 追従ビットを適切に変更する。

`rewriting_info_flag[i][j]` が 1 に等しいことは、現スケーラブルレイヤ表現を変換することによって得られるアクセスユニットの代替セットに関して情報が存在することを規定する。アクセスユニットの代替セットを構築するための方法が規定されないにも関わらず、指示された情報が正しいことを表している。変数 `j` はアクセスユニットの代替セットに対する `entropy_coding_mode_flag` の値を規定し、`j` の値が 0 または 1 に等しいことは、それぞれ、エントロピー符号化方法に CAVLC を使用するまたは CABAC を使用することを示している。

`rewriting_profile_level_idc[i][j]` は、変換後の現スケーラブルレイヤ表現に対するアクセスユニットの代替セットのプロファイル適合性を規定する。もしこれらのシンタックス要素が、付属資料 A で規定された、スケーラブルレイヤ表現の変換により得られたアクセスユニットの代替セットのプロファイルおよびレベル適合性を規定するために使用されたなら、`rewriting_profile_level_idc[i]` は、`profile_idc`、`constraint_set0_flag`、`constraint_set1_flag`、`constraint_set2_flag`、`constraint_set3_flag`、`reserved_zero_4bits`、`level_idc` で構成される 3 バイトの正確な複写である。

`rewriting_avg_bitrate[i][j]` は、現スケーラブルレイヤの表現の変換によって得られたアクセスユニットの代替セットの平均ビットレートを規定する。ビット毎秒単位で、アクセスユニットの代替セットの平均ビットレートは、等式 G-372 で規定されている関数 `BitRateBPS()` を用いて、`BitRateBPS(rewriting_avg_bitrate[i][j])` で与えられる。

`rewriting_max_bitrate[i][j]` は、付属資料 C で規定されたアクセスユニット除去時刻の任意の 1/2 時刻窓単位で、現スケーラブルレイヤの表現を変換することにより得られるアクセスユニットの代替セットの最大ビットレートを規定する。ビット毎秒単位で、アクセスユニットの代替セットの最大ビットレートは、等式 G-372 で規定されている関数 `BitRateBPS()` を用いて、`BitRateBPS(rewriting_max_bitrate[i][j])` で得られる。

**pr\_num\_dId\_minus1** に 1 を加えたものは、その SEI メッセージに優先レイヤ特性情報が存在している依存レイヤの数を規定する。依存レイヤは、固有の **dependency\_id** 値を持つターゲットアクセスユニット集合の全ての VCL NAL ユニットおよび関連付けられた非 VCL NAL ユニットで構成されている。

**pr\_dependency\_id[i]** は、優先レイヤ特性情報が後続のシンタックス要素によって示される依存レイヤの **dependency\_id** の値を規定する。

**pr\_num\_minus1[i]** に 1 を加えたものは、**pr\_dependency\_id[i]** に等しい **dependency\_id** を持つ依存レイヤに対応する優先レイヤの数を規定する。優先レイヤは、規定された **priority\_id** 値以下の **priority\_id** 値を持つ依存レイヤの符号化スライス NAL ユニットと関連付けられた非 VCL NAL ユニット集合を表現する。優先レイヤを復号するために要求されるビットストリームサブセットは、優先レイヤ表現と呼ばれる。**ql\_num\_minus1** の値は、0 から、ターゲットアクセスユニット集合の全ての優先レイヤ情報 SEI メッセージのシンタックス要素 **element alt\_priority\_id[i]** の最大値の、それぞれを含む範囲内でなければならない。

**pr\_id[i][j]** は、**pr\_dependency\_id[i]** に等しい **dependency\_id** を持つ依存レイヤに対する優先レイヤの識別子を規定する。**pr\_dependency\_id[i]** および **pr\_id[i][j]** により識別された優先レイヤは、優先レイヤ情報 SEI メッセージにより規定された、**alt\_priority\_id[i]** の値を持つ全ての VCL NAL ユニットおよび関連付けられた非 VCL NAL ユニットで構成される。

**pr\_profile\_level\_idc[i][j]** は、**pr\_dependency\_id[i]** および **pr\_id[i][j]** により識別される優先レイヤに対する優先レイヤ表現の、プロファイルおよびレベル適合を規定する。**pr\_profile\_level\_idc[i]** は、もしこれらのシンタックス要素が付属資料 A および付属資料 G で規定される **pr\_dependency\_id[i]** および **pr\_id[i][j]** によって識別された優先レイヤに対する優先レイヤ表現のビットストリームサブセットのプロファイルおよびレベル適合性を規定するために使用されたなら、**profile\_idc**、**constraint\_set0\_flag**、**constraint\_set1\_flag**、**constraint\_set2\_flag**、**constraint\_set3\_flag**、**reserved\_zero\_4bits**、**level\_idc** で構成された 3 バイトの正確な複写である。

**pr\_avg\_bitrate[i][j]** は、**ql\_dependency\_id[i]** および **ql\_id[i][j]** により識別される優先レイヤのビットストリームサブセットの平均ビットレートを規定する。ビット毎秒単位で、**ql\_dependency\_id[i]** および **pr\_id[i][j]** によって識別された優先レイヤのビットストリームサブセットの平均ビットレートは、等式 G-372 で規定された関数 **BitRateBPS()** を用いて、**BitRateBPS(pr\_avg\_bitrate[i][j])** によって与えられる。平均ビットレートは、本標準の付属資料 C で規定されるアクセスユニットの除去時刻に従って導出される。以降、**bTotal** をスケーラビリティ情報 SEI メッセージに続き（アクセスユニットの **pr\_dependency\_id[i]** および **pr\_id[i][j]** によって識別される品質レイヤの NAL ユニットのビットを含む）、かつ（もし存在するならば）スケーラビリティ情報 SEI メッセージ、もしくは（そうでないならば）ストリームの終了を含む（復号順序で）次のアクセスユニットに先行する、**pr\_dependency\_id[i]** および **pr\_id[i][j]** によって識別される優先レイヤの全ての NAL ユニットのビット数とする。 $t_1$  は現アクセスユニットの除去時刻（秒単位）であり、 $t_2$  は（もし存在するならば）次のスケーラビリティ情報 SEI メッセージ、または（それ以外）ストリームの終了の前の、最新のアクセスユニット（復号順序）の除去時刻（秒単位）である。

**pr\_avg\_bitrate[i]** の値を規定する  $x$  を用いて、以下の条件が満たされなければならない。

- もし  $t_1$  が  $t_2$  に等しくないならば、等式 G-373 で規定された条件が満たされなければならない。
- それ以外 ( $t_1$  が  $t_2$  に等しい)、等式 G-374 で規定された条件が満たされなければならない。

**pr\_max\_bitrate[i][j]** は、付属資料 C で規定されるアクセスユニット除去時刻の任意の 1/2 時間窓単位で、**pr\_dependency\_id[i]** および **pr\_id[i][j]** によって識別される優先レイヤのビットストリームサブセットの最大ビットレートを規定する。ビット毎秒単位で、**pr\_dependency\_id[i]** および **pr\_id[i][j]** によって識別される優先レ

イヤの最大ビットレートは、等式 G-372 で規定された関数  $\text{BitRateBPS}(\ )$  を用いて、 $\text{BitRateBPS}(\text{pr\_max\_bitrate}[i][j])$  によって与えられる。

**priority\_id\_setting\_uri**[PriorityIdSettingUriIdx]は、ターゲットアクセスユニット集合に対する NAL ユニットヘッダの **priority\_id** 値を計算するために使用される方法の記述の統一資源識別子 (URI) を規定する、UTF-8 キャラクタで符号化されたヌル終端文字列の第 PriorityIdSettingUriIdx 番目のバイトである。

### G.13.2.2 レイヤ非存在SEIメッセージの意味

存在する場合、この SEI メッセージは、復号順序で、次のレイヤ非存在 SEI メッセージ、次のスケーラビリティ情報 SEI メッセージ、またはビットストリームの終端までの、いずれか復号順序で最も早いものまでの、現アクセスユニットおよび全ての後続するアクセスユニットから構成されるターゲットアクセスユニット集合に適用される。レイヤ非存在 SEI メッセージは、ターゲットアクセスユニット集合内に存在しない先行するスケーラブル情報 SEI メッセージによって示されるスケーラブルレイヤを識別する。レイヤ非存在 SEI メッセージは、復号順序で最新のスケーラビリティ SEI メッセージを常に参照する。

記 レイヤ非存在SEIメッセージは累積的な効果を持たない。

**num\_layers** は、ターゲットアクセスユニット集合内に存在しないスケーラブルレイヤの数を規定する。**num\_layers** が 0 に等しいことは、復号順序で最新のスケーラビリティ情報 SEI メッセージ内で規定される全てのスケーラブルレイヤがターゲットアクセスユニット集合内に存在することを規定する。**num\_layers** の値は、0 から 2047 のそれぞれを含む範囲内でなければならない。

**layer\_id[i]**は、ターゲットアクセスユニット集合内に存在しないスケーラブルレイヤの識別子を規定する。**layer\_id[i]**値は、復号順序で最新のスケーラビリティ情報 SEI メッセージ内に含まれる **layer\_id[i]**値を参照し、かつその中になければならない。**layer\_id[i]**の値は、0 から 2047 のそれぞれを含む範囲内でなければならない。

### G.13.2.3 レイヤ依存変更SEIメッセージの意味

レイヤ依存変更 SEI メッセージは、スケーラブルレイヤ依存関係の変更を規定する。存在する場合、レイヤ依存変更 SEI メッセージは、現アクセスユニットおよび復号順序で後続する、次のレイヤ依存変更 SEI メッセージ、次のレイヤ非存在 SEI メッセージ、次のスケーラビリティ情報 SEI メッセージ、またはビットストリームの終端の、復号順序でいずれか最も早いものまでの全てのアクセスユニットで構成されたターゲットアクセスユニット集合に適用される。レイヤ依存変更 SEI メッセージの存在は、全ての  $i$  に対する **layer\_id[i]** によって識別されるスケーラブルレイヤに対するレイヤ依存関係が、復号順序で最新のスケーラビリティ情報 SEI メッセージによって規定されるレイヤ依存関係に比べて、ターゲットアクセスユニット集合に対して変化することを規定する。任意の  $i$  の値に対する **layer\_id[i]** と等しくないレイヤ識別子を持ったスケーラブルレイヤに対して、レイヤ依存関係は、復号順序で最新のスケーラビリティ情報 SEI メッセージで規定されるものと同じままである。

もし、復号順序で最新のスケーラビリティ SEI メッセージ内で指示されたレイヤ依存情報に従って、スケーラブルレイヤ A が他のスケーラブルレイヤ B に直接的にも間接的にも依存しないならば、レイヤ依存変更 SEI メッセージはレイヤ A と B の間のレイヤ依存関係を規定してはならない。レイヤ依存変更 SEI メッセージは、復号順序で最新のスケーラビリティ情報 SEI メッセージを常に参照する。

記 レイヤ依存変更SEIメッセージは累積的な効果を持たない。

**num\_layers\_minus1** に 1 を加えたものは、復号順序で最新のスケーラビリティ情報 SEI メッセージによっ

て指示されるレイヤ依存情報と比べて、ターゲットアクセスユニット集合に対してレイヤ依存情報が変化するスケーラブルレイヤの数を規定する。`num_layers_minus1` の値は 0 から 2047 のそれぞれを含む範囲内である。

`layer_id[i]` は、レイヤ依存情報が変化するスケーラブルレイヤの識別子を規定する。`layer_id[i]` 値は、復号順序で最新のスケーラビリティ情報 SEI メッセージ内に含まれる `layer_id[i]` 値を参照し、かつその中になければならない。`layer_id[i]` の値は、0 から 2047 のそれぞれを含む範囲内でなければならない。

`layer_dependency_info_present_flag[i]` が 1 に等しいことは、レイヤ依存変更 SEI メッセージ内の `layer_id[i]` に等しいレイヤ識別子を持ったスケーラブルレイヤに対するレイヤ依存情報の存在を規定する。`layer_dependency_info_present_flag[i]` が 0 に等しいことは、レイヤ依存変更 SEI メッセージ内に `layer_id[i]` に等しいレイヤ識別子を持ったスケーラブルレイヤに対するレイヤ依存情報が存在せず、かつそれは `layer_dependency_info_src_layer_id_delta_minus1[i]` により指示されたスケーラブルレイヤに対するレイヤ依存情報と同じであることを規定する。

`num_directly_dependent_layers[i]` は、`layer_id[i]` に等しいレイヤ識別子を持ったスケーラブルレイヤが直接的に依存するスケーラブルレイヤの数を規定する。`num_directly_dependent_layers` の値は、0 から 2047 のそれぞれを含む範囲内でなければならない。

`directly_dependent_layer_id_delta_minus1[i][j]` は、`layer_id[i]` と、`layer_id[i]` に等しいレイヤ識別子を持ったスケーラブルレイヤが直接的に依存する第  $j$  番目のスケーラブルレイヤのレイヤ識別子との差を規定する。直接的に依存するスケーラブルレイヤのレイヤ識別子は、 $(layer\_id[i] - directly\_dependent\_layer\_id\_delta\_minus1 - 1)$  に等しい。

`layer_dependency_info_src_layer_id_delta_minus1[i]` は、`layer_id[i]` に等しいレイヤ識別子を持ったスケーラブルレイヤと同じレイヤ依存情報を持つスケーラブルレイヤのレイヤ識別子を規定する。`layer_id[i]` に等しいレイヤ識別子を持つスケーラブルレイヤと同じレイヤ依存情報を持つスケーラブルレイヤのレイヤ識別子は、 $(layer\_id[i] - layer\_dependency\_info\_src\_layer\_id\_delta\_minus1 - 1)$  に等しい。

#### G.13.2.4 サブピクチャスケーラブルレイヤSEIメッセージの意味

存在する場合、サブピクチャスケーラブルレイヤ SEI メッセージは、復号順序で、動き制約スライスグループセット SEI メッセージの直後に続き、これらの SEI メッセージ両方は、同じ SEI NAL ユニット内に存在しなければならない。動き制約スライスグループセット SEI メッセージによって識別されるスライスグループセットは、サブピクチャスケーラブルレイヤ SEI メッセージの関連付けされたスライスグループセットと呼ばれる。

`layer_id` は、関連付けされたスライスグループセット内の符号化スライス NAL ユニットが属するスケーラブルレイヤのレイヤ識別子を規定する。`layer_id` の値は、0 から 2047 のそれぞれを含む範囲内でなければならない。

#### G.13.2.5 非要求レイヤ表現SEIメッセージの意味

`num_info_entries_minus1` に 1 を加えたものは、このシンタックス要素に続く情報エントリ数を規定する。`num_info_entries_minus1` の値は 0 から 7 のそれぞれを含む範囲内でなければならない。

`entry_dependency_id[i]` は、非要求レイヤ表現の情報が以下のシンタックス要素によって規定される、以降ターゲットレイヤ表現と呼ばれるレイヤ表現の、`dependency_id` 値を規定する。`entry_dependency_id[i]` のイン

スタンスは `entry_dependency_id[i]` の昇順で存在しなければならない。ターゲットレイヤ表現の `quality_id` 値は 0 に等しくなければならない。ターゲットレイヤ表現に対する非要求レイヤ表現は、ターゲットレイヤ表現として同じ `dependency_id` 値および `quality_id` 値を持つ符号化ビデオシーケンスにおける他のいかなるレイヤ表現の復号においても要求されない。

記1 0 よりも大きい `quality_id` を持つレイヤ表現は、同じ `dependency_id` 値および 0 に等しい `quality_id` 値を持つレイヤ表現と同じ非要求レイヤ表現の集合を持つ。

`num_non_required_layer_rep_minus1[i]` に 1 を加えたものは、ターゲットレイヤ表現に対する非要求レイヤ表現の数を規定する。`num_non_required_layer_rep_minus1[i]` の値は、0 から 127 のそれぞれを含む範囲内であらなければならない。

`non_required_layer_rep_dependency_id[i][j]` は、ターゲットレイヤ表現に対する第  $j$  番目の非要求レイヤ表現の `dependency_id` 値を規定する。

`non_required_layer_rep_quality_id[i][j]` は、ターゲットレイヤ表現に対する第  $j$  番目の非要求レイヤ表現の `quality_id` 値を規定する。加えて、`non_required_layer_rep_dependency_id[i][j]` に等しい `dependency_id` および `non_required_layer_rep_quality_id[i][j]` よりも大きい `quality_id` を持つこれらのレイヤ表現もまた、ターゲットレイヤ表現に対する非要求レイヤ表現である。さらには、追加の非要求レイヤ表現が以下の通り導出される。

- A に等しい `dependency_id` を持つレイヤ表現が、A 以上の B である B に等しい `dependency_id` を持つレイヤ表現に対する非要求レイヤ表現でない場合、A に等しい `dependency_id` を持つレイヤ表現に対する全ての非要求レイヤ表現もまた、B に等しい `dependency_id` を持つレイヤ表現に対する非要求レイヤ表現である。
- `dIdMax` が、いかなる  $i$  に対しても `entry_dependency_id[i]` に等しくなく、かつ `dIdMax` が少なくとも 1 つの  $i$  の値に対して `entry_dependency_id[i]` よりも大きい場合、以下が適用される。
  - `dIdC` を、`dIdMax` 未満である `entry_dependency_id[i]` の最大値に等しいものとする。
  - `dIdMax` に等しい `dependency_id` を持つレイヤ表現は、`dIdC` に等しい `dependency_id` および 0 に等しい `quality_id` を持ったレイヤ表現と同じ非要求レイヤ表現の集合を持たなければならない。

記2 `dIdMax` が、全ての  $i$  の値に対して `entry_dependency_id[i]` よりも小さい場合、`dIdMax` に等しい `dependency_id` を持つレイヤ表現に対する関連付けられたアクセスユニット内の非要求レイヤ表現は存在しない。

### G.13.2.6 優先レイヤ情報SEIメッセージの意味

この SEI メッセージにおいて伝達される情報は、現アクセスユニット内の与えられた `dependency_id` の値を持つ NAL ユニットに対する、`priority_id` の代替値を規定する。`priority_id` の代替値は、優先レイヤを規定する。

`pr_dependency_id` は、この SEI メッセージが適用されるアクセスユニット内の NAL ユニットに対する `dependency_id` の値を規定する。

`num_priority_ids` は、代替の `priority_id` の値がこの SEI メッセージによって規定される、`pr_dependency_id` に等しい `dependency_id` を持った依存表現を持つレイヤ表現の数を規定する。

`alt_priority_id[i]` は、現アクセスユニット内の、 $i$  に等しい `quality_id` と `pr_dependency_id` に等しい `dependency_id` を持った NAL ユニットに対する `priority_id` の代替値を規定する。

### G.13.2.7 スケーラブルネスティングSEIメッセージの意味

スケーラブルネスティング SEI メッセージは、30 に等しくない `payLoadType` の 1 つ以上の SEI メッセージを含まなければならない。スケーラブルネスティング SEI メッセージ内に含まれる SEI メッセージは、ネストされた SEI メッセージと呼ばれる。スケーラブルネスティング SEI メッセージ内に含まれない SEI メッセージはネストされない SEI メッセージと呼ばれる。ネストされた SEI メッセージが適用される範囲は、存在する場合、シンタックス要素 `all_layer_representations_in_au_flag`、`num_layer_representations_minus1`、`sei_dependency_id[i]`、`sei_quality_id[i]`、および `sei_temporal_id` シンタックス要素によって指示される。

バッファリング期間 SEI メッセージおよび他のあらゆるタイプの SEI メッセージは、同じスケーラブルネスティング SEI メッセージ内でネストされてはならない。バッファリング期間 SEI メッセージがスケーラブルネスティング SEI メッセージ内に含まれる場合、それは対象となる SEI メッセージと呼ばれ、以下が適用される。

- 同じアクセスユニット内で、対象となる SEI メッセージに復号順序で先行するいかなる SEI メッセージも、バッファリング期間 SEI メッセージ、もしくはペイロードとしてバッファリング期間 SEI メッセージを伴ったスケーラブルネスティング SEI メッセージでなければならない。
- 加えて、以下が適用される。
  - もし、対象となる SEI メッセージ内の `all_layer_representations_in_au_flag` の値が 1 に等しいならば、スケーラブルネスティング SEI メッセージは、同じアクセスユニット内で、対象となる SEI メッセージに復号順序で先行してはならない。
  - それ以外（対象となる SEI メッセージ内の `all_layer_representations_in_au_flag` の値が 0 に等しい）、同じアクセスユニット内の、対象となる SEI メッセージに先行する任意のスケーラブルネスティング SEI メッセージに対して、全ての存在する  $i$  に対する  $(sei\_dependency\_id[i] \ll 4) + sei\_quality\_id[i]$  の値は、その対象となる SEI メッセージに対する  $(sei\_dependency\_id[i] \ll 4) + sei\_quality\_id[i]$  のいかなる値よりも小さくなくてはならない。

ピクチャタイミング SEI メッセージと、他のタイプのあらゆる SEI メッセージは、同じスケーラブルネスティング SEI メッセージ内でネストされてはならない。

`all_layer_representations_in_au_flag` が 1 に等しいことは、ネストされた SEI メッセージがアクセスユニットの全てのレイヤ表現に適用されることを規定する。`all_layer_representations_in_au_flag` が 0 に等しいことは、ネストされた SEI メッセージの適用可能範囲が、シンタックス要素 `num_layer_representations_minus1`、`sei_dependency_id[i]`、`sei_quality_id[i]`、および `sei_temporal_id` によって規定されることを規定する。

`num_layer_representations_minus1` に 1 を加えたものは、ネストされた SEI メッセージが適用されるレイヤ表現の数を規定する。`num_layer_representations_minus1` が存在しない場合、それは、`numSVCLayers` をそのアクセスユニット内に存在するレイヤ表現の数として、 $(numSVCLayers - 1)$  に等しく推定される。

`sei_dependency_id[i]` および `sei_quality_id[i]` は、それぞれ、ネストされた SEI メッセージが適用される第  $i$  番目のレイヤ表現の `dependency_id` および `quality_id` を規定する。`num_layer_representations_minus1` が存在しない場合、0 から `num_layer_representations_minus1` (`num_layer_representations_minus1` を、推定された値とする) のそれぞれを含む範囲内の  $i$  に対して、`sei_dependency_id[i]` および `sei_quality_id[i]` の値は、以下の通り推定される。

- `setDQId` を、アクセスユニット内に存在する全てのレイヤ表現に対する DQId 値とする。

- 0 から `num_layer_representations_minus1` までのそれぞれを含む範囲を進む `i` に対して、以下が適用される。
  - `sei_dependency_id[i]` および `sei_quality_id[i]` は、`minDQId` を `setDQId` における最小値 (DQId の最小値) として、それぞれ (`minDQId >> 4`) および (`minDQId & 15`) に等しく推定される。
  - 集合 `setDQId` の最小値 (DQId の最小値) が `setDQId` から除去され、したがって `setDQId` の要素数が 1 減少される。

`sei_temporal_id` は、ネストされた SEI メッセージが適用されるビットストリームサブセットの `temporal_id` を規定する。

`temporal_id` が存在しない場合、そのアクセスユニットの `temporal_id` に等しく推定されなければならない。

ネストされた SEI メッセージがバッファリング期間 SEI メッセージもしくはピクチャタイミング SEI メッセージである場合、`sei_temporal_id` は、ネストされたバッファリング期間 SEI メッセージもしくはピクチャタイミング SEI メッセージが適用されるビットストリームサブセットを示す。スケーラブルネスティング SEI メッセージ内でネストされるバッファリング期間 SEI メッセージもしくはピクチャタイミング SEI メッセージに対して、`sei_dependency_id[i]`、`sei_quality_id[i]`、および `sei_temporal_id` は、それぞれ、ネストされたバッファリング期間 SEI メッセージもしくはピクチャタイミング SEI メッセージが適用されるビットストリームサブセットの `dependency_id`、`quality_id`、および `temporal_id` の最大値を規定する。

`payloadType` が 2 から 3 のそれぞれを含む範囲内、もしくは 6 から 35 のそれぞれを含む範囲内であるネストされた SEI メッセージに対して、`sei_temporal_id` は、そのアクセスユニットの `temporal_id` に等しくなければならない。

`sei_nesting_zero_bit` は、0 に等しくなければならない。

#### G.13.2.8 ベースレイヤ時間HRD SEIメッセージの意味

存在する場合、この SEI メッセージは、現アクセスユニットおよび、次のベースレイヤ時間 HRD SEI メッセージもしくはビットストリームの終端のいずれか復号順序で早いものまでの、復号順序で後続する全てのアクセスユニットから構成されたターゲットアクセスユニット集合に適用される。

`num_of_temporal_layers_in_base_layer_minus1` に 1 を加えたものは、シンタックス要素が適用されるターゲットアクセスユニット集合内のビットストリームサブセットの数を規定する。

`temporal_id[i]` は、第 `i` 番目のビットストリームサブセットの `temporal_id` 値を規定する。

第 `i` 番目のビットストリームサブセットを、G.8.8.1 小々節で規定されるビットストリーム抽出処理を、`temporal_id[i]` に等しい `tIdTarget`、0 に等しい `dIdTarget`、および 0 に等しい `qIdTarget` を入力として起動することにより得られるターゲットアクセスユニット集合に対するものとする。

`timing_info_present_flag[i]` は、第 `i` 番目のビットストリームサブセットに対して適用される `timing_info_present_flag` の値を規定する。

`num_units_in_tick[i]` は、第 `i` 番目のビットストリームサブセットに対して適用される `num_units_in_tick` の値を規定する。`num_units_in_tick[i]` は 0 より大きくなくてはならない。

`time_scale[i]` は、第 `i` 番目のビットストリームサブセットに対して適用される `time_scale` の値を規定する。`time_scale[i]` は 0 より大きくなくてはならない。

**fixed\_frame\_rate\_flag[i]** は、第 i 番目のビットストリームサブセットに対して適用される **fixed\_frame\_rate\_flag** の値を規定する。

**nal\_hrd\_parameters\_present\_flag[i]** は、第 i 番目のビットストリームサブセットに対して適用される **nal\_hrd\_parameters\_present\_flag** の値を規定する。

**vcl\_hrd\_parameters\_present\_flag[i]** は、第 i 番目のビットストリームサブセットに対して適用される **vcl\_hrd\_parameters\_present\_flag** の値を規定する。

**low\_delay\_hrd\_flag[i]** は、第 i 番目のビットストリームサブセットに対して適用される **low\_delay\_hrd\_flag** の値を規定する。

**pic\_struct\_present\_flag[i]** は、第 i 番目のビットストリームサブセットに対して適用される **pic\_struct\_present\_flag** の値を規定する。

### G.13.2.9 品質レイヤ整合性検査SEIメッセージの意味

**num\_info\_entries\_minus1** に 1 を加えたものは、シンタックス要素 **entry\_dependency\_id[i]** および **quality\_layer\_crc[i]** の数を規定する。**num\_info\_entries\_minus1** の値は 0 から 7 のそれぞれを含む範囲内でなければならない。

**entry\_dependency\_id[i]** は、**quality\_layer\_crc[i]** が指示される依存表現の **dependency\_id** 値を規定する。**entry\_dependency\_id[i]** のインスタンスは、それらの値の昇順で出現しなければならない。

**quality\_layer\_crc[i]** は、依存表現内の 0 より大きい **quality\_id** を持つ全ての VCL NAL ユニットに対する巡回冗長検査を規定する。

**quality\_layer\_crc[i]** は、下記の順序付けられたステップによって得られる値に等しくなくてはならない。

1. 変数 **qNalUnits[ ]** を、復号順序で、**entry\_dependency\_id[i]** に等しい **dependency\_id** を持つ依存表現内の、0 より大きい **quality\_id** を持った全ての VCL NAL ユニットのバイトの、ネットワークバイト順序での連結を含むバイトの 1 次元配列とする。
2. **pLen** を、**qNalUnits[ ]** 内のバイト数とする。
3. **quality\_layer\_crc[i]** の値は、以下の擬似コード処理を実施することにより得られる **crcVal** の値に等しくなくてはならない。

```
qNalUnits[pLen ]=0
qNalUnits[pLen+1]=0
crcVal=0xFFFF
for(bitIdx=0;bitIdx<(pLen+2)*8;bitIdx++){
    crcMsb=(crcVal>>15)&1
    bitVal=(qNalUnits[bitIdx>>3]>>(7-(bitIdx&7)))&1
    crcVal=((crcVal<<1)+bitVal)&0xFFFF^(crcMsb*0x1021)
}
```

### G.13.2.10 冗長ピクチャプロパティSEIメッセージの意味

**num\_dId\_minus1** に 1 を加えたものは、冗長ピクチャプロパティが以下のシンタックス要素によって規定される依存表現の数を規定する。

**dependency\_id[i]**は、冗長ピクチャプロパティが以下のシンタックス要素によって規定される依存表現の **dependency\_id** 値を規定する。

**num\_qId\_minus1[i]**に 1 を加えたものは、冗長ピクチャプロパティが以下のシンタックス要素によって規定される **dependency\_id[i]**に等しい **dependency\_id** を持った依存表現内のレイヤ表現の数を規定する。

**quality\_id[i][j]**は、冗長ピクチャプロパティが以下のシンタックス要素によって規定される **dependency\_id[i]**に等しい **dependency\_id** を持つレイヤ表現の **quality\_id** 値を規定する。

**num\_redundant\_pics\_minus1[i][j]**に 1 を加えたものは、冗長ピクチャプロパティが以下のシンタックス要素によって規定される、**dependency\_id[i]**に等しい **dependency\_id** および **quality\_id[i][j]**に等しい **quality\_id** を持ったレイヤ表現内の冗長符号化ピクチャの数を規定する。

**redundant\_pic\_cnt\_minus1[i][j][k]**に 1 を加えたものは、冗長ピクチャプロパティが以下のシンタックス要素によって規定される、**dependency\_id[i]**に等しい **dependency\_id** および **quality\_id[i][j]**に等しい **quality\_id** を持った冗長符号化ピクチャの **redundant\_pic\_cnt** 値を規定する。

**dependency\_id[i]** に等しい **dependency\_id** 、 **quality\_id[i][j]** に等しい **quality\_id** 、 および **(redundant\_pic\_cnt\_minus1[i][j][k]+1)** に等しい **redundant\_pic\_cnt** を持った冗長符号化ピクチャは、ターゲット冗長符号化ピクチャと呼ばれる。**dependency\_id[i]**に等しい **dependency\_id**、**quality\_id[i][j]**に等しい **quality\_id**、および 0 に等しい **redundant\_pic\_cnt** を持ったレイヤ表現は、主レイヤ表現と呼ばれる。

**pic\_match\_flag[i][j][k]**が 1 に等しいことは、ターゲット冗長符号化ピクチャが、**redundant\_pic\_cnt** の値のみが異なる、主レイヤ表現の正確な複写であることを規定する。

**mb\_type\_match\_flag[i][j][k]**が 1 に等しいことは、ターゲット冗長符号化ピクチャ内の全てのマクロブロックに対するシンタックス要素 **mb\_type** が、主レイヤ表現内の対応するシンタックス要素と同一であることを規定する。

**motion\_match\_flag[i][j][k]**が 1 に等しいことは、冗長符号化ピクチャ内の全てのマクロブロックに対する、全ての INTER 符号化マクロブロックの **mb\_type** および関連付けされた参照インデックスと動きベクトルが、主レイヤ表現内の対応するシンタックス要素と同一であることを規定する。

**residual\_match\_flag[i][j][k]**が 1 に等しいことは、冗長符号化ピクチャ内の全ての INTER 符号化マクロブロックに対する、G.8.5.3 小小説で規定された再構成された残差サンプル値が、主レイヤ表現内の対応するマクロブロックに対する再構成された残差サンプル値と同一もしくは近いことを規定する。

**intra\_samples\_match\_flag[i][j][k]**が 1 に等しいことは、冗長符号化ピクチャ内の全ての INTRA 符号化マクロブロックの再構成されたサンプル値は、主レイヤ表現内の対応するマクロブロックの再構成されたサンプル値と同一もしくは近いことを規定する。

#### G.13.2.11 時間レベルゼロ依存表現インデックスSEIメッセージの意味

時間レベルゼロ依存表現インデックス SEI メッセージは、NAL ユニットのロスが伝送中に予期される場合、現アクセスユニットの復号に必要とされる 0 に等しい **temporal\_id** を持つ依存表現が利用可能かどうかを検出するための機構を提供する。

**setOfDId** を、以下で導出される値の集合とする。

- もしスケラブルネスティング SEI メッセージ内に時間レベルゼロ依存表現インデックス SEI メッセージ

ジが含まれないならば、setOfDId は、0 に等しく設定された正確に 1 つの値で構成される。

- それ以外 (スケーラブルネスティング SEI メッセージ内に時間レベルゼロ依存表現インデックス SEI メッセージが含まれる)、setOfDId は、その時間レベルゼロ依存表現インデックス SEI メッセージに関連付けられた、0 から num\_layer\_representations\_minus1 のそれぞれを含む範囲内の全ての i に対する sei\_dependency\_id[i] 値で構成される。

**tl0\_dep\_rep\_idx** は、もし temporal\_id が 0 に等しいならば、関連付けられた依存表現に対する時間レベルゼロインデックスを規定する。そうでないならば (temporal\_id が 0 よりも大きい)、tl0\_dep\_rep\_idx は、関連付けられた、0 に等しい temporal\_id を持った復号順序で最新のアクセスユニットの依存表現の時間レベルゼロインデックスを規定する。

集合 setOfDId の dId の各値に対して、tl0\_dep\_rep\_idx への以下の制約に従わなければならない。

- もし dId に等しい dependency\_id を持った依存表現が、5 に等しい nal\_unit\_type を持った NAL ユニットあるいは、20 に等しい nal\_unit\_type および 1 に等しい idr\_flag を持った NAL ユニットを含むならば、tl0\_dep\_rep\_idx は 0 に等しくなければならない。
- それ以外 (dId に等しい dependency\_id を持った依存表現が、5 に等しい nal\_unit\_type を持った NAL ユニットあるいは、20 に等しい nal\_unit\_type および 1 に等しい idr\_flag を持った NAL ユニットを含まない)、以下が適用される。
  - prevTL0DepRep を、dId に等しい dependency\_id を持った依存表現を含む、0 に等しい temporal\_id を持った復号順序で最新のアクセスユニットの dId に等しい dependency\_id を持った依存表現であるとする。
  - prevTL0DepRepIdx を、依存表現 prevTL0DepRep の tl0\_dep\_rep\_idx に等しいとする。
  - 現アクセスユニットの temporal\_id に依存して、以下が適用される。
    - もし現アクセスユニットの temporal\_id が 0 に等しいならば、tl0\_dep\_rep\_idx は (prevTL0DepRepIdx+1)%256 に等しくなければならない。
    - それ以外 (現アクセスユニットの temporal\_id が 0 よりも大きい)、tl0\_dep\_rep\_idx は prevTL0DepRepIdx に等しくなければならない。

記 tl0\_dep\_rep\_idx 機構が効果的に使用されるために、伝送機構は、dependency\_id の特定の値に対するデータを搬送する全てのパケット内に、その情報が存在することを保証すべきである。

**effective\_idr\_pic\_id** は、復号順序で最新の idr\_pic\_id の値が、このアクセスユニットもしくは、sei\_dependency\_id[i] によって指示された依存表現に対するアクセスユニットに先行する任意のアクセスユニット内に存在することを規定する。

集合 setOfDId の dId の各値に対して、effective\_idr\_pic\_id への以下の制約に従わなければならない。

- もし dId に等しい dependency\_id を持った依存表現が、5 に等しい nal\_unit\_type を持った NAL ユニットあるいは、20 に等しい nal\_unit\_type および 1 に等しい idr\_flag を持った NAL ユニットを含むならば、effective\_idr\_pic\_id は、dId に等しい dependency\_id を持った依存表現の idr\_pic\_id に等しくなければならない。
- それ以外 (dId に等しい dependency\_id を持った依存表現が、5 に等しい nal\_unit\_type を持った NAL ユ

ニットあるいは、20 に等しい `nal_unit_type` および 1 に等しい `idr_flag` を持った NAL ユニットを含まない)、`effective_idr_pic_id` は、5 に等しい `nal_unit_type` を持った NAL ユニットあるいは、20 に等しい `nal_unit_type` および 1 に等しい `idr_flag` を持った NAL ユニットを含む、`dId` に等しい `dependency_id` を持った復号順序で前の依存表現の `idr_pic_id` に等しくなければならない。

### G.13.2.12 時間レベルスイッチング点SEIメッセージの意味

時間レベルスイッチング点 SEI メッセージは、時間レベルスイッチング点の情報を与える。もしレイヤ表現が時間レベルスイッチング点 SEI メッセージに関連付けられているならば、それは、次で規定される、`delta_frame_num` によって制限される時間レベルスイッチング点である。それ以外、そのレイヤ表現は時間レベルスイッチング点かもしれないし、そうでないかもしれない。

以下において、`tId` は、時間レベルスイッチング点 SEI メッセージが含まれているアクセスユニットに関連付けられた `temporal_id` の値に等しいとする。

記1 `dId`を、ビットストリーム適応処理が`dId`以下の`dependency_id`および、現アクセスユニットまでのそれを含まない`tId`より小さい`temporal_id`を持った依存表現を含むビットストリームサブセット `subBitstreamA`を生成するために使用される、`dependency_id`の値とする。ビットストリーム適応処理は、`subBitstreamA`、および、現アクセスユニットから開始しそれを含む、`dId`以下の`dependency_id`および`tId`以下の`temporal_id`を持った依存表現を含むビットストリームサブセットが、本標準に適合しているかどうかを、時間レベルスイッチング点SEIメッセージから推定できる。

時間レベルスイッチング点 SEI メッセージは、`temporal_id` が 0 に等しいアクセスユニット、もしくは 20 に等しい `nal_unit_type` を持つ NAL ユニットを含まず、かつプリフィックス NAL ユニットが存在しないアクセスユニット内に存在してはならない。

時間レベルスイッチング点 SEI メッセージは、それがスケーラブルネスティング SEI メッセージ内に含まれない限り、付加拡張情報 RBSP 内に存在してはならない。

その意味は、この SEI メッセージを含んだスケーラブルネスティング SEI メッセージの `sei_dependency_id[i]` および `sei_quality_id[i]` の値の各ペアによって指示されたレイヤ表現へ、独立に適用される。

切替え先レイヤ表現を、`sei_dependency_id[i]` に等しい `dependency_id` および、`sei_quality_id[i]` に等しい `quality_id` を持つこの SEI メッセージを含むアクセスユニット内のレイヤ表現とする。

`delta_frame_num` は、切替え先レイヤ表現と、後で規定される要求されるレイヤ表現との間の `frame_num` 値の差分を指示する。`currFrameNum` を切替え先レイヤ表現の `frame_num` とし、`requiredFrameNum` を要求されるレイヤ表現の `frame_num` とする。変数 `requiredFrameNum` を、`currFrameNum-delta_frame_num` に等しく設定する。要求されるレイヤ表現は、`requiredFrameNum` に等しい `frame_num`、`sei_dependency_id[i]` に等しい `dependency_id`、および `sei_quality_id[i]` に等しい `quality_id` を持った、復号順序で前のアクセスユニットのレイヤ表現とする。

記2 切替え先レイヤ表現に対する復号処理は、要求されるレイヤ表現および、復号処理において`tId`より小さい`temporal_id`および`sei_dependency_id[i]`以下の`dependency_id`を持つ、復号順序で後続する任意のレイヤ表現を使用するかも知れない。

切替え先レイヤ表現および、`temporal_id` が `tId` に等しくかつ `dependency_id` が `sei_dependency_id[i]` 以下の、復号順序で後続の任意のレイヤ表現は、復号処理において `temporal_id` が `tId-1` に等しくかつ `dependency_id` が `sei_dependency_id[i]` 以下である要求されるレイヤ表現に復号順序で先行する、いかなるレイヤ表現も使用してはならない。

切替え先レイヤ表現および、`temporal_id` が `tId` に等しくかつ `dependency_id` が `sei_dependency_id[i]` 以下の、

復号順序で後続の任意のレイヤ表現は、復号処理において `temporal_id` が `tId` に等しくかつ `dependency_id` が `sei_dependency_id[i]` 以下である切替え先レイヤ表現を含むアクセスユニットに復号順序で先行する、いかなるレイヤ表現も使用してはならない。

記3 `temporal_id`が`tId`に等しくかつ`dependency_id`が`sei_dependency_id[i]`以下である、復号順序で切替え先レイヤ表現に続く、それを含む全てのレイヤ表現は、`temporal_id`が`tId`より小さくかつ`dependency_id`が`sei_dependency_id[i]`以下である、要求されるレイヤ表現から始まり、復号順序で切替え先レイヤ表現を含むアクセスユニットに先行するアクセスユニット内のレイヤ表現までのそれぞれを含む、全てのレイヤ表現が復号された場合、復号可能である。

#### G.14 SVC ビデオ有用性情報拡張

付属資料 E における規定が、シーケンスパラメータセットを SVC シーケンスパラメータセットに置き換えて適用される。加えて、以下が適用される。

G.14.1 SVC VUIパラメータ拡張シンタックス

svc_vui_parameters_extension() {	C	記述子
<b>num_layers_minus1</b>	0	ue(v)
for( i = 0; i <= num_layers_minus1; i++ ) {		
<b>dependency_id[ i ]</b>	0	u(3)
<b>quality_id[ i ]</b>	0	u(4)
<b>temporal_id[ i ]</b>	0	u(3)
<b>timing_info_present_flag[ i ]</b>	0	u(1)
if( timing_info_present_flag[ i ] ) {		
<b>num_units_in_tick[ i ]</b>	0	u(32)
<b>time_scale[ i ]</b>	0	u(32)
<b>fixed_frame_rate_flag[ i ]</b>	0	u(1)
}		
<b>nal_hrd_parameters_present_flag[ i ]</b>	0	u(1)
if( nal_hrd_parameters_present_flag[ i ] )		
hrd_parameters( )		
<b>vcl_hrd_parameters_present_flag[ i ]</b>	0	u(1)
if( vcl_hrd_parameters_present_flag[ i ] )		
hrd_parameters( )		
if( nal_hrd_parameters_present_flag[ i ]    vcl_hrd_parameters_present_flag[ i ] )		
<b>low_delay_hrd_flag[ i ]</b>	0	u(1)
<b>pic_struct_present_flag[ i ]</b>	0	u(1)
}		
}		

G.14.2 SVC VUIパラメータ拡張の意味

**num\_layer\_minus1** に 1 を加えたものは、タイミング情報、NAL HRD パラメータ、VCL HRD パラメータ および **pic\_struct\_present\_flag** が存在するかもしれない、**temporal\_id**、**dependency\_id**、および **quality\_id** の 3

組の個数を規定する。

**dependency\_id[i]**は、第 i 番目の依存レイヤを指示する。

**quality\_id[i]**は、第 i 番目の品質レイヤを指示する。

**temporal\_id[i]**は、第 i 番目の時間レイヤを指示する。

以下のシンタックス要素は、**temporal\_id[i]**に等しい **tIdTarget**、**dependency\_id[i]**に等しい **dIdTarget**、および **quality\_id[i]**に等しい **qIdTarget** を入力とし、第 i 番目のサブビットストリームを出力として、G.8.8.1 小節で規定されたサブビットストリーム抽出処理によって得られる符号化ビデオシーケンスに適用される。

**timing\_info\_present\_flag[i]**は、第 i 番目のサブビットストリームの **timing\_info\_present\_flag** 値を規定する。

**num\_units\_in\_tick[i]**は、第 i 番目のサブビットストリームの **num\_units\_in\_tick** 値を規定する。

**time\_scale[i]**は、第 i 番目のサブビットストリームの **time\_scale** 値を規定する。

**fixed\_frame\_rate\_flag[i]**は、第 i 番目のサブビットストリームの **fixed\_frame\_rate\_flag** 値を規定する。

**nal\_hrd\_parameters\_present\_flag[i]**は、第 i 番目のサブビットストリームの **nal\_hrd\_parameters\_present\_flag** 値を規定する。

**vcl\_hrd\_parameters\_present\_flag[i]**は、第 i 番目のサブビットストリームの **vcl\_hrd\_parameters\_present\_flag** 値を規定する。

**low\_delay\_hrd\_flag[i]**は、第 i 番目のサブビットストリームの **low\_delay\_hrd\_flag** 値を規定する。

**pic\_struct\_present\_flag[i]**は、第 i 番目のサブビットストリームの **pic\_struct\_present\_flag** 値を規定する。