

**TTC標準**  
Standard

JT-H263

低ビットレート通信用  
ビデオ符号化方式

Video Coding for Low Bitrate Communication

第 3.3 版

2009 年 11 月 17 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。

## 目次

< 参考 > .....	7
要約 .....	9
1. 本標準の規定範囲 .....	9
2. 参照 .....	9
2. 1 標準としての参照 .....	9
2. 2 参考としての参照 .....	10
3. 仕様概要 .....	10
3. 1 ビデオ入出力 .....	10
3. 2 デジタル入出力 .....	10
3. 3 標本化周波数 .....	10
3. 4 情報源符号化アルゴリズム .....	11
3. 5 ビットレート .....	14
3. 6 バッファリング .....	14
3. 7 伝送の対称性 .....	14
3. 8 エラー処理 .....	14
3. 9 マルチポイント操作 .....	15
4. 情報源符号器 .....	15
4. 1 情報源フォーマット .....	15
4. 2 ビデオ情報源符号化アルゴリズム .....	17
4. 3 符号化制御 .....	20
4. 4 強制画面更新 .....	21
4. 5 スタートコードのバイトアライメント .....	21
5. シンタックスと意味 .....	21
5. 1 ピクチャレイヤ .....	22
5. 2 グループオブブロック (GOB) レイヤ .....	39
5. 3 マクロブロック (MB) レイヤ .....	40
5. 4 ブロックレイヤ .....	48
6. 復号処理 .....	51
6. 1 動き補償 .....	51
6. 2 係数復号 .....	53
6. 3 ブロックの再生 .....	54
付属資料A 逆変換精度の仕様 .....	55
付属資料B 仮想標準復号器 .....	57
付属資料C マルチポイントへの考慮 .....	59
C. 1 画面凍結要求 .....	59
C. 2 画面更新要求 .....	59
C. 3 画面凍結解除 .....	59
C. 4 コンティニューアスプレゼンスマルチポイントとビデオ多重化 (CPM) .....	59

付属資料D 無制限動きベクトルモード .....	61
D. 1 ピクチャの境界を越えた動きベクトル .....	61
D. 2 動きベクトルの範囲の拡大 .....	62
付属資料E シンタックス算術符号化モード .....	64
E. 1 序論 .....	64
E. 2 SAC符号器の仕様 .....	64
E. 3 SAC復号器の仕様 .....	65
E. 4 シンタックス .....	66
E. 5 P S C _ F I F O .....	66
E. 6 ヘッダレイヤのシンボル .....	67
E. 7 マクロブロックおよびブロックレイヤのシンボル .....	67
E. 8 SACモデル .....	68
付属資料F 拡張予測モード .....	72
F. 1 序論 .....	72
F. 2 マクロブロック当たり4本の動きベクトル .....	72
F. 3 輝度成分に対するオーバーラップ動き補償 .....	73
付属資料G P Bフレームモード .....	76
G. 1 序論 .....	76
G. 2 P Bフレームと I N T R Aブロック .....	76
G. 3 ブロックレイヤ .....	77
G. 4 P BフレームでのBピクチャのベクトル計算 .....	77
G. 5 P BフレームでのBブロックの予測 .....	77
付属資料H 符号化されたビデオ信号に対する前方誤り訂正 .....	80
H. 1 序論 .....	80
H. 2 誤り訂正フレーム化 .....	80
H. 3 誤り訂正符号 .....	80
H. 4 誤り訂正フレーム化の再ロック時間 .....	80
付属資料I 拡張 I N T R A符号化モード .....	82
I. 1 序論 .....	82
I. 2 シンタックス .....	82
I. 3 復号処理 .....	83
付属資料J デブロッキングフィルタモード .....	89
J. 1 序論 .....	89
J. 2 U M VとA Pモードとの関係（付属資料D、F） .....	89
J. 3 デブロッキングフィルタの定義 .....	90
付属資料K スライス構造モード .....	94
K. 1 序論 .....	94
K. 2 スライスレイヤの構造 .....	95
付属資料L 付加拡張情報仕様 .....	98
L. 1 序論 .....	98
L. 2 P S U P Pフォーマット .....	98
L. 3 “無動作” .....	98
L. 4 全画面凍結要求 .....	99

L. 5	部分画面凍結要求	99
L. 6	サイズ変更部分画面凍結要求	99
L. 7	部分画面凍結解除要求	99
L. 8	全画面スナップショットタグ	100
L. 9	部分画面スナップショットタグ	100
L. 10	ビデオ時間セグメント開始タグ	100
L. 11	ビデオ時間セグメント終了タグ	100
L. 12	プログレッシブ高画質化セグメント開始タグ	100
L. 13	プログレッシブ高画質化セグメント終了タグ	100
L. 14	クロマキー情報	101
L. 15	拡張機能タイプ	102
付属資料M 改良PBフレームモード		104
M. 1	序論	104
M. 2	B <sub>PB</sub> マクロブロック予測モード	104
M. 3	Bマクロブロックの双方向予測におけるベクトルの計算	105
M. 4	MODB表	105
付属資料N 参照ピクチャ選択モード		106
N. 1	序論	106
N. 2	ビデオ情報源符号化アルゴリズム	106
N. 3	逆方向チャネルメッセージ用のチャネル	107
N. 4	シンタックス	108
N. 5	復号処理	111
付属資料O 時間、SNR、空間スケーラビリティモード		112
O. 1	序論	112
O. 2	ピクチャの伝送順序	115
O. 3	ピクチャレイヤのシンタックス	116
O. 4	マクロブロックレイヤのシンタックス	117
O. 5	動きベクトルの復号	121
O. 6	内挿補間フィルタ	121
付属資料P 参照ピクチャ再サンプリング		124
P. 1	序論	124
P. 2	シンタックス	126
P. 3	再サンプリングアルゴリズム	128
P. 4	実装方法の例	131
P. 5	“ファクター4”再サンプリング	134
付属資料Q 縮小解像度更新モード		137
Q. 1	序論	137
Q. 2	復号処理	138
Q. 3	参照ピクチャの拡張	139
Q. 4	動きベクトルの再生	141
Q. 5	輝度信号用の拡大オーバーラップ動き補償	142
Q. 6	縮小解像度再生予測誤差のアップサンプリング	143
Q. 7	ブロック境界フィルタ	145
付属資料R 独立セグメント復号モード		147
R. 1	序論	147

R. 2 動作モード .....	147
R. 3 使用制約 .....	148
付属資料S 代替 I N T E R 可変長符号モード .....	149
S. 1 序論 .....	149
S. 2 係数用の代替 I N T E R 可変長符号 .....	149
S. 3 C B P Y用の代替 I N T E R 可変長符号 .....	149
付属資料T 修正量子化モード .....	151
T. 1 序論 .....	151
T. 2 修正D Q U A N Tの更新 .....	151
T. 3 色差係数用の変更量子化ステップサイズ .....	152
T. 4 修正された係数の範囲 .....	152
T. 5 使用制約 .....	153
付属資料U 拡張参照ピクチャ選択モード .....	154
U. 1 序論 .....	154
U. 2 ビデオ情報源符号化アルゴリズム .....	154
U. 3 順方向チャンネルのシンタックス .....	156
U. 4 復号処理 .....	173
U. 5 逆方向チャンネルメッセージ .....	177
付属資料V データパーティションスライスモード .....	182
V. 1 序論 .....	182
V. 2 データパーティショニングの構造 .....	182
V. 3 他のオプションモードとの相互作用 .....	185
付属資料W 追加付加拡張情報仕様 .....	189
W. 1 序論 .....	189
W. 2 参照 .....	189
W. 3 追加F T Y P E値 .....	189
W. 4 P S U P Pオクテットの推奨される最大値 .....	189
W. 5 固定小数点I D C T .....	189
W. 6 ピクチャメッセージ .....	196
付属資料X プロファイルとレベルの定義 .....	202
X. 1 本標準の規定範囲 .....	202
X. 2 優先モードサポートプロファイル .....	202
X. 3 画像フォーマットと画像クロック周波数 .....	205
X. 4 動作能力のレベル .....	206
X. 5 T T C標準J T - H 2 4 5との組み合わせに関するG E N E R I C C A P A B I L I T Y定義 .....	210
付録I 誤り追跡 .....	214
I. 1 序論 .....	214
I. 2 誤り追跡 .....	214
付録II 推奨されるオプション拡張 .....	215

## 低ビットレート通信用ビデオ符号化方式

## &lt;参考&gt;

## 1. 国際勧告などとの関連

本標準は、低ビットレートでのオーディオビジュアルサービスにおいて動画像を圧縮するために用いられる符号化方式を規定している。本標準第1版は1995年11月のITU-T SG15会合において承認されたITU-T勧告H. 263に準拠している。1998年1月から2月に開催されたITU-T SG16会合において、画質向上、誤り耐性強化のためのITU-T勧告H. 263の改定が承認された。本標準第2版は、これに伴う第1版の改定であり、改定されたITU-T勧告H. 263に準拠している。さらに、2000年11月に開催されたITU-T SG16会合において、更なる誤り耐性機能強化と機能向上のためのITU-T勧告H. 263の改定が承認された。本標準第3版は、これに伴う第2版の改定であり、改定されたITU-T勧告H. 263に準拠している。さらに、プロファイルとレベルの定義のためのITU-T勧告H. 263の改定が、2000年11月に開催されたITU-T SG16会合においてAAPへConsentされ、2001年2月に成立した。本標準第3.1版は、これに伴う第3版の改定であり、改定されたITU-T勧告H. 263に準拠している。さらに、2004年3月15日のITU-T SG16(2001-2004)において、QCIF、サブQCIF解像度で、先に定義済みのレベルで提供されるよりも高いビットレート(2・(64000)=128000bit/s)をサポートする適合点を提供するレベル(レベル45と示される)を追加するために付属資料Xの改定が承認された。本標準第3.2版は、これに伴う第3.1版の改定であり、改定されたITU-T勧告H. 263に準拠している。

ITU-T勧告では、第2版以降の付属資料U、V、W、Xに関する改定はかつて分離されて承認され出版されていたが、2005年1月13日にITU-T SG16(2005-2008)において承認されたITU-T勧告H. 263第3版において本文に統合された。そのITU-T勧告H. 263第3版において、本文も含む文書全体に訂正・明確化が施された。本標準第3.3版は、この改定されたITU-T勧告H. 263第3版に準拠している。

なお、ITU-T勧告H. 263では2001年6月に、参考として符号器・復号器の実装例を解説するAppendix IIIが分離されて出版されており、現在も有効なITU-T文書であるが、本文には統合されていない。Appendix III(付録III)は本標準の必須部分ではなく、TTC文書としては現在発行されていない。

また、2005年8月にITU-T勧告H. Imp263が承認され出版されており、それはITU-T勧告H. 263の実装者のための細かな暫定訂正を示したものである。具体的には付図Q.8/JT-H263及び付図Q.9/JT-H263における整数除算の負の被除数の解釈に関するもので、本標準の実装者もITU-T勧告H. Imp263を参照することを推奨する。

## 2. 上記国際勧告に対する追加項目等

## 2.1 オプション選択項目

なし。

## 2.2 ナショナルマター決定項目

なし。

## 2.3 その他

- (1) 本標準は上記ITU-T勧告に対し、先行している項目はない。
- (2) 本標準は上記ITU-T勧告に対し、追加した項目はない。
- (3) 本標準は上記ITU-T勧告に対し、削減した項目はない。
- (4) 本標準は上記ITU-T勧告の以下の箇所に対し、誤記または誤りと判断し修正を行っている。

修正箇所	修正前	修正後
------	-----	-----

F. 3節 第14行目 (m, n)の定義式	$m=y/8$	$n=y/8$
------------------------------	---------	---------

2. 4 原勧告との章立ての構成比較表  
上記国際勧告との章立て構成の相違はない。

### 3. 改版の履歴

版 数	制 定 日	改 版 内 容
第 1 版	1996年11月27日	制 定
第 2 版	1998年11月26日	画質向上、誤り耐性強化のための付属資料、付録の追加、及びそれに伴う本文修正
第 3 版	2001年11月27日	誤り耐性強化、機能向上のための付属資料の追加、及びそれに伴う本文修正
第 3. 1 版	2002年 5月30日	プロファイルとレベルの定義のための付属資料の追加、及びそれに伴う本文修正
第 3. 2 版	2005年 6月 2日	Q C I F、サブQ C I F解像度で、先に定義済みのレベルで提供されるよりも高いビットレート ( $2 \cdot (64000) = 128000 \text{bit/s}$ ) をサポートする適合点を提供するレベル (レベル45と示される) の追加、及びそれに伴う本文修正
第 3. 3 版	2009年11月17日	I T U - T H. 263 (01/2005) における訂正・明確化への準拠、及び編集上の瑕疵修正

### 4. 工業所有権

本標準に関わる「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。

### 5. その他

(1) 参照している勧告、標準など

TTC標準： JT-H223、JT-H242、JT-H245、JT-H261、  
JT-H262、JT-H320、JT-H324

ITU-R勧告： BT. 601-5

ISO/IEC標準： 10646

IETF RFC： 2396



## 要約

本標準は低ビットレートでのオーディオビジュアルサービスにおいて動画像を圧縮するために用いられる符号化方式を規定する。ビデオ情報源符号化アルゴリズムの基本的な構成はTTC標準JT-H261に基づいており、時間的冗長性を用いたピクチャ間予測と、予測誤差の空間的冗長性を削減するための変換符号化とを用いている。符号器はサブQCIF、QCIF、CIF、4CIF、16CIFの5つの標準化されたビデオ情報源フォーマットにおいて動作し、広い範囲のカスタムビデオフォーマットを利用しても動作する。

復号器は、符号器においてはオプションである動き補償の能力を持つ。TTC標準JT-H261では1画素精度の動き補償とループ内フィルタが用いられるが、本標準では1/2画素精度の動き補償が用いられる。伝送符号には可変長符号化が用いられる。

基本的なビデオ情報源符号化アルゴリズムに加え、圧縮性能改善と付加機能をサポートするため、交渉により選択可能な18の符号化オプションを含んでいる。拡張表示能力と外部使用のためにビット列に追加の付加情報を含んでもよい。

本標準第3.3版は、本標準第3.2版に対し、主に以下のような多くの訂正と明確化を加えたものである（本標準第3.3版はITU-T H.263第3版（01/2005）に対応するが、ITU-T H.263第3版は、ITU-Tにおいてかつて分離されて承認・出版されていた付属資料U、V、W、Xを統合した上で、これらの訂正と明確化を加えたものである）：

- ・ 図8/JT-H263の訂正
- ・ 表1/JT-H263におけるBPPmaxKbテーブル及びピクチャパディングに関する明確化
- ・ 5.3.2小節におけるスタートコードに先行するマクロブロックスタッフ化に関する明確化
- ・ JT-H263付属資料JとIDCT丸め誤差の間の相互作用に関する明確化
- ・ 付属資料NにおけるBCM内のGN/MBAフィールドの文法解析可能性に関する明確化
- ・ 付属資料Oにおける直接双方向予測MB及びピクチャ外挿に関する明確化
- ・ 付属資料OにおけるINTRA参照ピクチャを用いるBピクチャの使用についての明確化
- ・ 付属資料Nを併用する付属資料Pの使用についての明確化
- ・ 付属資料Uにおける付図U.7/JT-H263の訂正

## 1. 本標準の規定範囲

本標準は低ビットレートでのオーディオビジュアルサービスにおいて動画像を圧縮するために用いられる符号化方式を規定する。ビデオ情報源符号化アルゴリズムの基本的な構成はTTC標準JT-H261に基づいている。性能向上と機能性増加のために交渉により選択可能な18の符号化オプションが含まれている。本標準は、最初の本標準第1版と完全に互換性があり、第1版の内容にオプション機能を追加する、本標準第2版の内容を含んでいる。

## 2. 参照

### 2.1 標準としての参照

以下の勧告及びその他の標準は、本標準で参照することにより本標準の規定となる事項を規定している。出版時には以下に示す版が有効であった。すべての勧告/標準及びその他の標準は、改定されることがあり、本標準を使用する場合は、以降に列挙される勧告/標準及びその他の標準の最新版が適用可能かどうかを調べるのが望ましい。現在の有効なITU-T勧告/TTC標準のリストが定期的に出版されている。本標準内の文書への参照は、それに単体文書としての勧告/標準のステータスを与えるものではない。

[1] ITU-R Recommendation BT.601-5 (1995), Studio Encoding Parameters of Digital Television for Standard 4:3 and Wide-Screen 16:9 Aspect Ratios.

ITU-R 勧告 BT.601-5 : 標準 4:3 及びワイドスクリーン 16:9 アスペクト比のデジタルテレビのスタジオ用符号化パラメータ

ここでは、勧告[1]は本標準に従って設計されるビデオコーデックで用いる画像の(Y、C<sub>B</sub>、C<sub>R</sub>)色空間とその8ビット整数表現を定義するために参照されている。(勧告[1]はこの本標準の他のいかなる側面を定義するためにも使われることはない。)

## 2. 2 参考としての参照

以下の付加的な I T U - T 勧告 / T T C 標準は本文で例証目的で言及されている。

- [2] ITU-T Recommendation H.223 (2001), Multiplexing protocol for low bit rate multimedia communication.  
T T C 標準 J T - H 2 2 3 : 低ビットレートマルチメディア通信用多重化プロトコル
- [3] ITU-T Recommendation H.242 (2004), System for establishing communication between audiovisual terminals using digital channels up to 2 Mbit/s.  
T T C 標準 J T - H 2 4 2 : 1920kbit/sまでのデジタルチャネルを利用したオーディオビジュアル端末間の通信を設定する方式
- [4] ITU-T Recommendation H.245 (2005), Control protocol for multimedia communication.  
T T C 標準 J T - H 2 4 5 : マルチメディア通信用制御プロトコル
- [5] ITU-T Recommendation H.261 (1993), Video codec for audiovisual services at  $p \times 64$  kbit/s.  
T T C 標準 J T - H 2 6 1 :  $p \times 64$ kbit/sオーディオビジュアル・サービス用ビデオ符号化方式
- [6] ITU-T Recommendation H.262 (2000) | ISO/IEC 13818-2:2000, Information technology - Generic coding of moving pictures and associated audio information: video.  
T T C 標準 J T - H 2 6 2 : 汎用映像符号化方式(ISO/IEC 13818-2)
- [7] ITU-T Recommendation H.324 (2002), Terminal for low bit-rate multimedia communication.  
T T C 標準 J T - H 3 2 4 : 低ビットレートマルチメディア通信用端末

## 3. 仕様概要

ビデオコーデックの概要のブロック図を図 1 / J T - H 2 6 3 に示す。

### 3. 1 ビデオ入出力

6 2 5 本および 5 2 5 本の標準テレビジョン信号を採用している地域内および地域間で唯一の標準を使用することを可能とするために、情報源符号器が動作する標準化された情報源フォーマットは共通中間フォーマット (C I F) に基づいている。外部の交渉(例えば T T C 標準 J T - H 2 4 5)を用いて、広範囲のオプションのカスタム情報源フォーマットを利用することも可能である。入出力テレビジョン信号が、例えばコンポジットであるかコンポーネントであるか、あるいはアナログであるかデジタルであるかということ、およびそれらのテレビジョン信号と情報源符号化フォーマットとの間で必要な変換方式は本標準の対象外である。

### 3. 2 デジタル入出力

ビデオ符号器は復号に必要な全ての情報を含むデジタルビット列を出力し、これらはさらに(例えば T T C 標準 J T - H 2 2 3 に規定されるような)他の信号と混合される場合もある。ビデオ復号器はビデオ符号器の逆の処理を行う。

### 3. 3 標本化周波数

画像は水平走査周波数の整数倍で標本化される。この標本化クロックとデジタル網クロックとは非同期である。

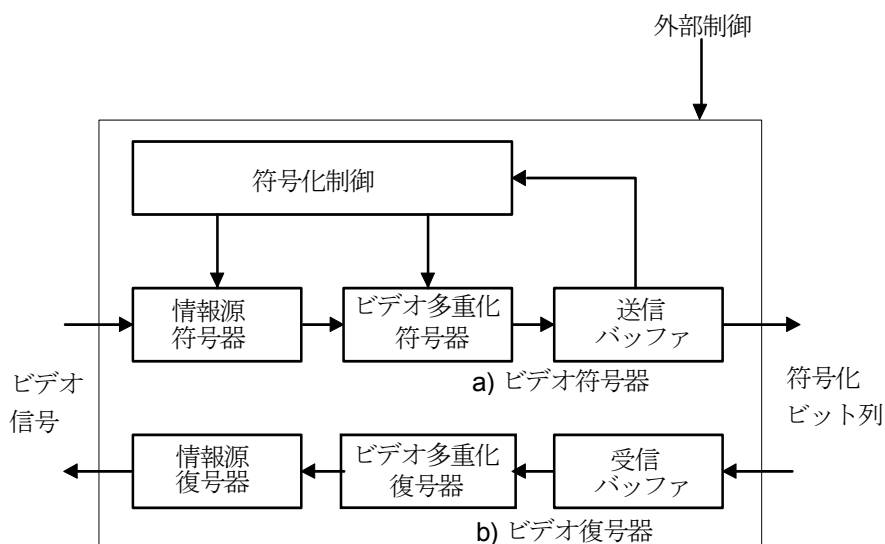


図1 / JT-H263 ビデオコーデックの概要ブロック図  
(ITU-T H. 263)

### 3.4 情報源符号化アルゴリズム

時間的冗長度を用いたピクチャ間予測と、予測誤差の空間的冗長度を削減するための変換符号化とを組み合わせたハイブリッド符号化方式を採用する。情報源復号器は動き補償の能力を持つが、情報源符号器ではオプションである。TTC標準JT-H261では1画素精度の動き補償とループ内フィルタが用いられるのに対して、1/2画素精度での動き補償が用いられる。伝送符号には可変長符号化が用いられる。

TTC標準JT-H263の核となる符号化アルゴリズムに加え、交渉により選択可能な18の符号化のオプションを（適当な制約の下に）一緒にまたは個別に用いることができる。拡張表示能力及び外部使用のために、ビット列に追加の付加情報を含めてもよい。必要時には、発生したビデオビット列に適用するため前方誤り訂正方法が用意されている。交渉により選択可能な符号化オプション、前方誤り訂正及び付加情報の使用については次の小節で述べる。

#### 3.4.1 コンティニューアプレゼンスマルチポイント及びビデオ多重化モード

このオプションのモードでは4つまでの個別のビデオ‘サブビット列’が同じビデオチャンネル内で送信できる。この機能は、コンティニューアプレゼンスマルチポイントのアプリケーション、また他には別々の論理チャンネルは使用できないが複数のビデオビット列が必要とされる他の状況での使用のために設計されている（付属資料C参照）。

#### 3.4.2 無制限動きベクトルモード

このオプションのモードでは動きベクトルがピクチャの外側を指すことが許される。ピクチャの端の画素が‘存在しない’画素に代わって予測に用いられる。ピクチャの端を横切る動きがある場合、特に小さい画像フォーマットのとき、このモードにより大きな効果を得ることが出来る（付属資料D参照）。さらに、より大きい動きベクトルの利用のためこのモードでは動きベクトル範囲の拡大を行っている。これはカメラに動きがある場合、及び大きな画像フォーマットで特に有効である。

#### 3.4.3 シンタックス算術符号化モード

このオプションのモードでは、可変長符号化の代わりに算術符号化が用いられる。SNR及び再生されるピクチャは同じであっても、生成されるビット数は大幅に削減される（付属資料E参照）。

#### 3. 4. 4 拡張予測モード

このオプションのモードでは、Pピクチャの輝度信号に対してオーバーラップブロック動き補償（OBMC）が用いられる（付属資料F参照）。ピクチャの中のいくつかのマクロブロックに対して、1つの16×16ベクトルの代わりに4つの8×8ベクトルが用いられる。符号器はどちらのタイプのベクトルを使用するか決定しなければならない。4つのベクトルはより多くのビットを必要とするが、より良い予測が可能である。一般的にこのモードの使用により大きな画質改善が得られる。OBMCはブロック歪みを軽減することから主観的評価が向上する。

#### 3. 4. 5 PBフレームモード

PBフレームは1つのユニットとして符号化される2つのピクチャから構成される。PBフレームの名前は、TTC標準JT-H262のPピクチャとBピクチャに由来する。このようにPBフレームは、前に復号されたPピクチャから予測された1つのPピクチャと、前に復号されたPピクチャと現在復号しているPピクチャの両方から予測された1つのBピクチャから構成される。Bピクチャは部分的に現在と未来のピクチャから双方向（Bidirectionally）に予測されるため、Bピクチャと名付けられた。この符号化オプションにより、ビットレートを実質上増加させることなくピクチャレートを著しく増加させることが出来る（付属資料G参照）。一方で改良PBフレームモードも提供されている（付属資料M参照）。オリジナルのPBフレームモードは、改良PBフレームモード採用以前に作られたシステムとの互換性の目的のためだけにここに残されている。

#### 3. 4. 6 前方誤り訂正

前方誤り訂正方法はビデオビット列の保護が必要な時に、その使用が適切な場合のために規定されている。前方誤り訂正のために用意された方法は、TTC標準JT-H261においても規定されているBCH符号方法と同じである（付属資料H参照）。

#### 3. 4. 7 拡張INTRA符号化モード

このオプションのモードでは、INTRAブロックはまず先に符号化された隣接するINTRAブロックから予測される（付属資料I参照）。別の可変長符号（VLC）表がINTRAブロック用に定義されている。この技術はINTRAピクチャ内のINTRAマクロブロック及びINTERピクチャ内のINTRAマクロブロックに適用される。このモードはTTC標準JT-H263シンタックスのコア部分のINTRA符号化に対して、圧縮性能を著しく改良する。

#### 3. 4. 8 デブロッキングフィルタモード

このオプションのモードではブロック歪みを削減するために、復号されたIピクチャとPピクチャの8×8ブロックの境界を越えてフィルタを適用する（付属資料J参照）。このフィルタの目的は、復号されたピクチャのブロック境界での歪みの発生を緩和することである。このフィルタは後続のピクチャの予測に使われるピクチャ、即ち動き予測ループ内のピクチャに影響を及ぼす。

#### 3. 4. 9 スライス構造モード

このオプションのモードでは‘スライス’レイヤがビット列シンタックスのGOBレイヤの代わりに用いられる（付属資料K参照）。このモードの目的は、拡張された誤り耐性能力を提供し、ビット列を下位の packets 伝送により適切に利用できるものとし、ビデオの遅延を最小限にすることである。ピクチャレイヤとマクロブロックレイヤ間のシンタックスレイヤである点で、スライスはGOBと似ている。しかし、GOB構造に要求される固定分割と固定転送順序とは対照的に、スライスレイヤの使用はピクチャの柔軟な分割を可能にする。

#### 3. 4. 10 付加拡張情報

追加の付加情報は、拡張表示能力の信号を出したり、外部使用のための情報を提供するために、ビット列に含まれてもよい（付属資料L参照）。この付加情報はサイズ変更の有無に関係なく全画面あるいは部分画面の凍結要求、あるいは凍結解除要求などの信号として使われる。また外部使用のためのビデオ列の特別なピクチャやシーケンスにタグを付けるためにも使われる。さらに、ビデオ合成のためのクロマキー情報を伝達するためにも使われる。復号器が付加情報を使ったり適切に解釈するための拡張能力を提供できない場合でも、付加情報はビット列に存在し

てもよい。要求された能力を提供するための必要条件が外部の手段によって交渉されなければ、復号器で付加情報を単に廃棄してもよい。

#### 3.4.1.1 改良PBフレームモード

このオプションモードではPBフレームモードオプションと比較して改善を示す（付属資料G、M参照）。これら2つのモードの主な違いは、改良PBフレームモードでは各Bブロックは別の動きベクトルを使って前方予測してもよく、またはゼロベクトルを使って後方予測をしてもよい点である。双方向予測において、比例縮小された前方予測の動きベクトルが良好な予測を行えない場合に、この方式は符号化効率を大幅に改善できる。後方予測は前のPピクチャとPBフレームとの間にシーンチェンジがある場合に特に有効である。

#### 3.4.1.2 参照ピクチャ選択モード

最近送った参照ピクチャ以外のピクチャからも時間予測できるようにすることで、誤りの発生しやすいチャンネル上でのリアルタイムのビデオ通信の性能を改善するオプションモードが提供されている（付属資料N参照）。このモードはビット列が正しく受信されているかどうかを知らせるために符号器に送り返す逆方向チャンネル状態メッセージとともに使用できる。誤りの発生しやすいチャンネル環境では、このモードはチャンネルの状態に合わせてビデオ符号化を最適化する符号器を可能にする。

#### 3.4.1.3 時間、SNR、空間スケーラビリティモード

このオプションモードでは時間、SNR、空間スケーラビリティを提供する（付属資料O参照）。スケーラビリティはビット列が1つの基本レイヤとこれに関係する1つ以上の高品質化レイヤから構成されることを意味する。基本レイヤは単独で復号できるビット列である。高品質化レイヤは、ピクチャレートの増加やピクチャの品質向上あるいはピクチャサイズの増加によって知覚される品質を向上させるため、基本レイヤと組合せて復号することができる。SNRスケーラビリティはピクチャの解像度を増やさずに画質を上げるため、高品質化レイヤを参照する。空間スケーラビリティは水平方向、垂直方向あるいは両方向ともピクチャの解像度を増やすことによって画質を上げるため、高品質化レイヤを参照する。Bピクチャを用いた時間スケーラビリティも提供されている。Bピクチャは参照レイヤの2枚のピクチャ、即ち現在のピクチャより時間的に前と後のピクチャから双方向予測できるピクチャを含むスケーラビリティ拡張である。Bピクチャは、表示する拡張ビデオシーケンスのピクチャレートを増加し、知覚される画質を向上するために高品質化レイヤ情報を使うことを許容する。このモードは可変の帯域幅の容量をもつ異種網や、誤り訂正機能との組合せでも有効である。

#### 3.4.1.4 参照ピクチャ再サンプリングモード

ビデオ画像予測に用いる参照ピクチャを、現在の入力ピクチャ用予測器の作成に先立って再サンプリング処理するオプションモードをサポートするためのシンタックスが提供されている（付属資料P参照）。これによってビデオ符号化のための適切な解像度の効率的、動的選択が可能になり、またグローバル動き補償器、あるいは特殊効果生成器で使用するためのピクチャワープも提供できる。

#### 3.4.1.5 縮小解像度更新モード

より高解像度の参照ピクチャに対し、縮小解像度への更新を許すオプションモードが提供されている（付属資料Q参照）。このモードは動きの激しいシーンを符号化する場合に使用されることが期待されており、ピクチャレートを増加させてシーン中の動きのある部分を表現し、一方でシーン中のより動きの少ない領域ではより高い解像度表現を維持することを符号器に許している。

#### 3.4.1.6 独立セグメント復号モード

GOBや複数のGOBからなるビデオ画像セグメント、あるいはスライス境界にまたがるあらゆるデータの依存関係を伴わずにピクチャを作成することを許すオプションモードを提供している（付属資料R参照）。このモードはビデオ画像セグメント領域の境界をまたがるような誤りデータの伝播を禁止することで、誤り耐性を提供する。

### 3.4.17 代替INTER可変長符号モード

ピクチャの変化が顕著な場合にINTERピクチャ符号化の効率を改善するオプションモードを提供している（付属資料S参照）。この効率の改善は、もともとINTRAピクチャ用に設計された可変長符号を、複数のINTERピクチャ係数に対しても同様に使用可能にしたことで得られる。

### 3.4.18 修正量子化モード

符号化のビットレート制御能力を高め、色差成分の量子化誤差を軽減し、表現可能なDCT係数の範囲を拡大し、係数値にはある種の制限を加えるオプションモードを提供している（付属資料T参照）。このモードは規定されるステップサイズの変化の範囲を拡大することによって、ビット列の差分量子化ステップサイズの意味を変えるものである。これはまた、色差成分データに使用される量子化ステップサイズを小さくすることもできる。ステップサイズで許された精度の下であらゆる可能な係数値が符号化できることを保証するため、DCT係数レベルの範囲が広げられている。このモードにおいては誤り検出の性能を向上させ、復号器の複雑さを最小にするため、ある種の制限が係数にも加えられる。

### 3.5 ビットレート

伝送クロックは外部から供給される。ビデオのビットレートは可変である。ビデオのビットレートは、本標準により制約を受けることは無いが端末または網により制限される。

### 3.6 バッファリング

符号器は付属資料Bにおいて定義された仮想標準復号器の要求を満足するように出力ビット列を制御しなければならない。ビデオデータは有効なクロックサイクル毎に供給されなければならない。このことは、MCBPCスタッフ化（表7/JT-H263、表8/JT-H263参照）、および前方誤り訂正が用いられる場合は前方誤り訂正スタッフ化フレーム（付属資料H参照）により実現される。

1つのピクチャを符号化することにより生成されるビット数は、1024ビットをユニットとして測定されたBPPmaxKbパラメータにより指定された最大値を越えてはならない。BPPmaxKbパラメータの許容される最小値は、ビット列で交渉により選択された最大のピクチャサイズに依存する（表1/JT-H263参照）。ピクチャサイズは輝度（Y）成分の幅×高さの画素数で測定される。外部の手段、例えばTTC標準JT-H245により、より大きな値が交渉により最初に選択される場合は、符号器は表1/JT-H263に指定されたBPPmaxKbより大きい値を用いることができる。

時間、SNR、空間スケーラビリティモード（付属資料O参照）が用いられる場合は、各高品質化レイヤのピクチャに送られるビット数はBPPmaxKbで規定された最大値を越えてはならない。

表1/JT-H263 異なる情報源画像フォーマットの最小BPPmaxKb  
(ITU-T H. 263)

画素数単位のYピクチャサイズ (=ピクチャヘッダで与えられる幅×高さ)	最小BPPmaxKb
25344まで (あるいはQCIF)	64
25360から101376 (あるいはCIF)	256
101392から405504 (あるいは4CIF)	512
405520以上	1024

### 3.7 伝送の対称性

コーデックは双方向または片方向のビジュアル通信に用いることができる。

### 3.8 エラー処理

エラー処理は外部の手段（例えばTTC標準JT-H223）により提供されるべきである。もし外部の手段（例えばTTC標準JT-H221）により提供されない場合は、付属資料Hに記述されたオプションの誤り訂正符号とフレーム化が用いられる。

復号器は、バッファがオーバーフローしないような符号化パラメータで、INTRAモードである次のピクチャの1つ以上のGOB（あるいは付属資料Kを用いるならばスライス）を符号化するようにコマンドを符号器に送る

ことができる。また復号器はスライス構造モード（付属資料K参照）が使用されない場合に、空でないGOBのヘッダのみを送るように符号器にコマンドを送ることも出来る。これらの信号は外部の手段（例えばTTC標準JT-H245）により伝送される。

### 3.9 マルチポイント操作

切替型マルチポイント操作を提供するのに必要な機能は付属資料Cに記述されている。

## 4 情報源符号器

### 4.1 情報源フォーマット

情報源符号器は、1) 1ライン当たりの画素数、1ピクチャ当たりのライン数及び画素アスペクト比によって定義される画像フォーマット、2) 画像クロック周波数（PCF）によって定義されるピクチャ間のタイミング、によって定義される情報源フォーマットを持つ順次走査画像において動作する。例えば共通中間フォーマット（CIF）は1ライン当たり352画素、1ピクチャ当たり288ライン、画素アスペクト比12:11、毎秒30000/1001ピクチャの画像クロック周波数を持つ。

情報源符号器はCIF PCFと呼ばれる毎秒30000/1001(約29.97)回の画像クロック周波数（PCF）で発生する順次走査画像において動作する。外部の手段によってオプションのカスタムPCF使用の交渉も可能である。カスタムPCFは1800000/(クロック除数×クロック変換ファクタ)によって与えられる。ここで、クロック除数は1から127の値をとり、クロック変換ファクタは1000か1001のいずれかの値をとり得る。画像クロック周波数の許容誤差は±50ppmである。

各画像は輝度成分と2つの色差成分（Y、C<sub>B</sub>、C<sub>R</sub>）として符号化される。これらの成分およびそれらの標本値を表す符号は、ITU-R勧告BT.601-5に規定されている。

- 黒レベル = 16
- 白レベル = 235
- 色差（ゼロ） = 128
- 色差（ピーク値） = 16、及び240

これらの値は公称値であり、符号化アルゴリズムは1から254の入力値に対して動作する。

本標準では、サブQCIF、QCIF、CIF、4CIFおよび16CIFの5つの画像フォーマットが規定されている。カスタム画像フォーマットの使用を交渉することも可能である。これらの全ての画像フォーマットにおいて、輝度信号は1ライン当たりdx画素、1ピクチャ当たりdyラインで直交格子状に配列される構造で標本化される。2つの色差成分は、1ライン当たりdx/2画素、1ピクチャ当たりdy/2ラインで直交に標本化される。各標準画像フォーマットのdx、dy、dx/2、dy/2の値は、表2/JT-H263で与えられる。

表2/JT-H263 TTC標準JT-H263の各標準画像フォーマットの  
(ITU-T H. 263) 1ライン当たりの画素数とライン数

画像フォーマット	輝度の1ライン当たりの画素数(dx)	輝度のライン数(dy)	色差の1ライン当たりの画素数(dx/2)	色差のライン数(dy/2)
サブQCIF	128	96	64	48
QCIF	176	144	88	72
CIF	352	288	176	144
4CIF	704	576	352	288
16CIF	1408	1152	704	576

全ての画像フォーマットにおいて、図2/JT-H263に示すように、色差信号のブロック境界が輝度信号のブロック境界と一致するように色差信号は標本化される。各標準画像フォーマットの画素アスペクト比は同じであり、TTC標準JT-H261のQCIF、CIFで定義されたものと同じで、(288/3) : (352/4)であり、こ

れは互いに素な12：11に約される。サブQ C I F 画像フォーマットを除く全ての標準画像フォーマットのピクチャ領域は4：3のアスペクト比を持つ。

カスタム画素アスペクト比の使用が、最初に外部の手段によって交渉されるならば、カスタム画像フォーマットは表3/J T-H 2 6 3に示すカスタム画素アスペクト比を持ち得る。ライン数が4で割り切れて[4, ..., 1152]の範囲にあり、1ライン当たりの画素数も4で割り切れて[4, ..., 2048]の範囲にあるならば、カスタム画像フォーマットは任意のライン数と任意の1ライン当たりの画素数とを持ち得る。幅や高さが16で割り切れない画像フォーマットについては幅や高さを拡張し、16で割り切れる最小のサイズであるとみなしてピクチャを復号し、表示目的のためだけであるが、画像の右端と下端を実際の幅や高さになるように切り取る。

表3/J T-H 2 6 3 カスタム画素アスペクト比  
( I T U - T H . 2 6 3 )

画素アスペクト比	画素の幅：画素の高さ
正方	1:1
CIF	12:11
4:3ピクチャ用525タイプ	10:11
16:9ピクチャ用CIF	16:11
16:9ピクチャ用525タイプ	40:33
拡張PAR	m:n ただしmとnは互いに素

全ての復号器及び符号器はC I Fの画像クロック周波数で動作しなければならない。カスタム画像クロック周波数のサポートはオプションである。全ての復号器はサブQ C I F 画像フォーマットかつQ C I F 画像フォーマットで動作しなければならない。C I F、4 C I F、1 6 C I F、あるいはカスタム画像フォーマットでの復号器の動作はオプションである。符号器はサブQ C I FとQ C I Fのいずれかの画像フォーマットで動作しなければならない。符号器はこれら2つのフォーマットのどちらを用いるかを定めるが、両方のフォーマットで動作する必要はない。C I F、4 C I F、1 6 C I F、あるいはカスタム画像フォーマットでの符号器の動作はオプションである。復号器がどのオプションのフォーマットで、どの画像クロック周波数で処理すべきかは、例えばT T C 標準J T-H 2 4 5のような、外部の手段により伝達される。可能な画像フォーマットとビデオ符号化アルゴリズムの完全な概要は、例えばT T C 標準J T-H 3 2 4のような端末記述を参照のこと。

注：C I Fにおいて、1ライン当たりの画素数は、525本または625本の標準テレビジョン信号の輝度および色差信号の有効領域をそれぞれ6.75および3.375MHzで標本化したものと実用上は等しい。これらの周波数はI T U-R 勧告B T . 6 0 1-5の標本化周波数と単純な関係を有している。

伝送されるピクチャ間の伝送されないピクチャの最小の数を規定することによって、最大ピクチャレートを制限する手段を符号器は持たなければならない。この最小の数は、外部の手段（例えばT T C 標準J T-H 2 4 5）によって選択される。P B フレームモードでの伝送しないピクチャの最小数の計算において、P B フレームユニットのPピクチャとBピクチャは2つの別々のピクチャとして計算される。



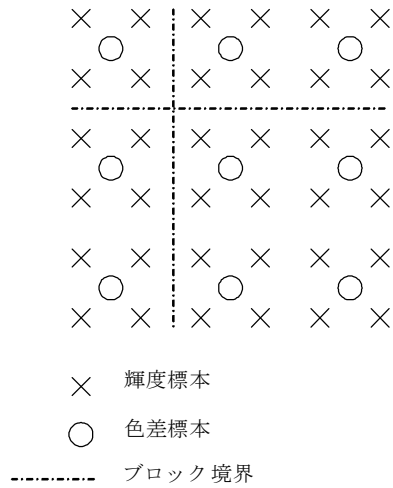
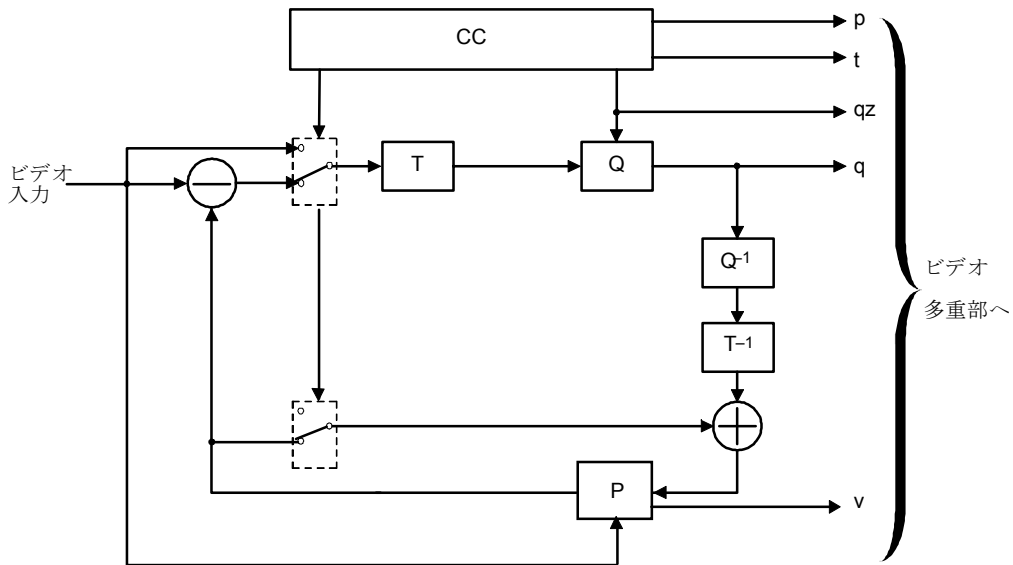


図2 / JT-H 263 輝度信号と色差信号の標本化位置  
(ITU-T H. 263)

#### 4.2 ビデオ情報源符号化アルゴリズム

情報源符号器の一般的な構成を図3 / JT-H 263 に示す。主な構成要素は予測、ブロック変換、量子化である。



- T 変換器
- Q 量子化器
- P 動き補償用可変遅延機能を持つピクチャメモリ
- CC 符号化制御
- p INTRA/INTER識別フラグ
- t 伝送/非伝送識別フラグ
- qz 量子化器表示
- q 変換係数の量子化出力インデックス
- v 動きベクトル

図3 / JT-H 263 情報源符号器  
(ITU-T H. 263)

#### 4. 2. 1 GOB、スライス、マクロブロック、ブロック

各ピクチャは幾つかのグループオブブロック（GOB）あるいはスライスに分割される。

グループオブブロック（GOB）は $k \times 16$ 以下のラインから構成され、 $k$ は画像フォーマットのライン数及びオプションの縮小解像度更新モードが用いられているか否かに依存する（付属資料Q参照）。その依存関係を表4/JT-H263に示す。ライン数が400以下でかつオプションの縮小解像度更新モードが用いられていない場合、 $k=1$ である。ライン数が800以下でかつオプションの縮小解像度更新モードが用いられている場合、あるいはライン数が400より大きい場合、 $k=2$ である。ライン数が800より大きい場合、 $k=4$ である。カスタムピクチャサイズを使用する場合、ピクチャのライン数が $k \times 16$ で割り切れなければ、最後（最下段）のGOB内のライン数は $k \times 16$ 未満となってもよい。しかし各標準画像フォーマットのライン数は $k \times 16$ の整数倍であるから、各標準画像フォーマットの各GOBのライン数は $k \times 16$ ラインである。従って例えばオプションの縮小解像度更新モードを使わない場合、ピクチャ当たりのGOBの数は、サブQCIFのとき6、QCIFのとき9、CIF、4CIF、16CIFのとき18である。GOBの垂直方向の走査によりGOBの番号割り当てが行われ、最上段のGOB（0番）に始まり最下段のGOBで終わる。CIF画像フォーマットのピクチャにおけるGOBの配置を例として図4/JT-H263に示す。各GOBのデータは、GOBヘッダ（空の可能性もある）とそれに続くマクロブロックのデータから構成される。GOBのデータはGOB番号の昇順にGOB毎に伝送される。

表4/JT-H263 GOBサイズを定義するためのパラメータ $k$   
(ITU-T H. 263)

ライン数 $d_y$	RRUモードでないときの $k$ 値	RRUモードのときの $k$ 値
4,...,400	1	2
404,...,800	2	2
804,...,1152	4	4

スライス構造モードは付属資料Kに記述されている。スライスは、マルチマクロブロックレイヤシンタックスにおけるGOBと同様であるがGOBより柔軟な形および使用方法があり、ある状態の下でビット列の中にどのような順序で出現してもよい。

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17

図4/JT-H263 CIFピクチャにおけるGOB配置  
(ITU-T H. 263)

各GOBは幾つかのマクロブロックに分割される。マクロブロック構造は、縮小解像度更新モード（RRU）を使用するかどうかに依存する（付属資料Q参照）。RRUモードでない場合、各マクロブロックは16画素 $\times$ 16ラインのY信号およびこれと空間的に対応する8画素 $\times$ 8ラインの $C_B$ と $C_R$ に相当する。さらに図5/JT-H263に示すように、マクロブロックは4つの輝度信号ブロックと空間的に対応する2つの色差信号ブロックから構成される。各輝度信号、色差信号のブロックは、このようにY、 $C_B$ 、 $C_R$ の8画素 $\times$ 8ラインに相当する。RRU

モードでない場合、GOBはサブQCIF、QCIF、CIFでは1マクロブロック列から、4CIFでは2マクロブロック列から、16CIFでは4マクロブロック列から構成される。

RRUモードの場合、マクロブロックは32画素×32ラインのY信号およびこれと空間的に対応する16画素×16ラインのC<sub>B</sub>とC<sub>R</sub>に相当し、各輝度信号、色差信号のブロックは、Y、C<sub>B</sub>、C<sub>R</sub>の16画素×16ラインに相当し、さらに1GOBは、CIF、4CIFでは1マクロブロック列から、16CIFでは2マクロブロック列から構成される。

各マクロブロック列について左から右への水平走査を、最上段のマクロブロック列から最下段のマクロブロック列まで順番に行うことによりマクロブロックの番号は割り当てられる。マクロブロックのデータはマクロブロック番号の昇順にマクロブロック毎に伝送される。ブロックのデータはブロック番号の昇順にブロック毎に伝送される(図5/JT-H263参照)。

符号化モードとブロックの伝送の選択基準は本標準の範囲外であり、符号化制御の一部として動的に変わってもよい。伝送されるべきブロックは変換され、その結果の変換係数は量子化されエントロピー符号化される。

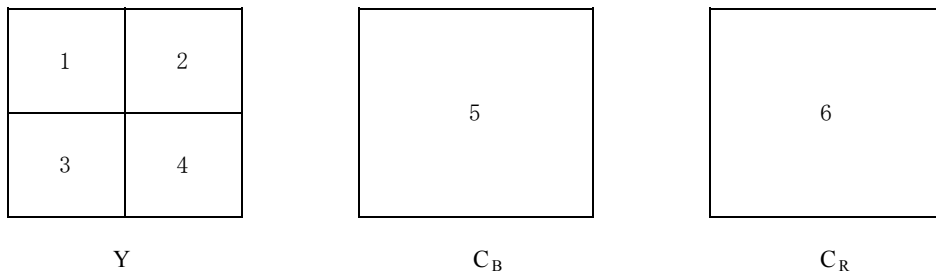


図5/JT-H263 マクロブロックにおけるブロック配置  
(ITU-T H. 263)

#### 4.2.2 予測符号化

予測は基本的にピクチャ間に対して行われ、動き補償を用いてもよい(4.2.3小節参照)。時間方向予測が行われる符号化モードをINTERと呼び、時間方向予測が行われない符号化モードをINTRAと呼ぶ。ピクチャレベル(Iピクチャに対するINTRA、またはPピクチャに対するINTER)にて、もしくはPピクチャのマクロブロックレベルにて、INTRA符号化モードを通知する。オプションのPBフレームモードでは、Bピクチャは常にINTERモードで符号化される。Bピクチャは部分的に双方向予測を用いる(付属資料G参照)。

ITC標準JT-H263は合計で7つの基本的なピクチャタイプ(最初の2つのみは必須)を持つ。これらは主に予測構造の点から以下のように定義される。

- INTRA: ピクチャは予測のための参照ピクチャを持たない(Iピクチャとも呼ぶ)。
- INTER: ピクチャは時間的に前の参照ピクチャを使う(Pピクチャとも呼ぶ)。
- PB: フレームは2つのピクチャを再生し、時間的に前の参照ピクチャを持つ(付属資料G参照)。
- 改良PB: 機能的に同様のフレームであるが、PBフレームよりも通常良い(付属資料M参照)。
- B: ピクチャは2つの参照ピクチャを持ち、そのうちの1つはBピクチャより時間的に前にあり、もう1つはBピクチャより時間的に後ろにあり、同じピクチャサイズを持つ(付属資料O参照)。
- EI: ピクチャは時間的に同時である参照ピクチャを持ち、同じかより小さいピクチャサイズを持つ(付属資料O参照)。
- EP: ピクチャは2つの参照ピクチャを持ち、そのうちの1つはEPピクチャより時間的に前にあり、もう1つは時間的に同時であり、同じかより小さいピクチャサイズを持つ(付属資料O参照)。

ここで使われたように、‘参照’または‘アンカー’ピクチャは、別のピクチャを復号するための基礎として参照することにより使うことができるデータを含むピクチャである。それが時々逆の時間の方向での使用を実際に指示することがあるが、参照によるこの用法は‘予測’としてまた知られている。

#### 4. 2. 3 動き補償

復号器はマクロブロック当たり1ベクトルを、また拡張予測モードまたはデブロッキングフィルタモードが用いられる場合はマクロブロック当たり1つまたは4つのベクトルを受け取る（付属資料F、J参照）。PBフレームが用いられる場合は、Bマクロブロック予測の動きベクトルのため、マクロブロック当たりもう1つのデルタベクトルが伝送される。同様に、改良PBフレームマクロブロック（付属資料M参照）は、付加的前方動きベクトルを含むことができる。Bピクチャマクロブロック（付属資料O参照）は、前方および後方動きベクトルといっしょに伝送でき、EPピクチャは前方動きベクトルといっしょに伝送できる。

動きベクトルの水平・垂直成分は共に整数または0.5単位の値を取り得る。デフォルトの予測モードでは、これらの値は[-16, 15.5]の範囲に限られる（このことはBピクチャの前方および後方動きベクトル成分に対しても同様である）。しかし無制限動きベクトルモードにおいては、ベクトル成分の最大の範囲は増加する。もしPLUSPTYPEが存在しない場合、その範囲は[-31.5, 31.5]である。つまり予測値が[-15.5, 16]の範囲である場合は、各動きベクトル成分は予測値に対して[-16, 15.5]の範囲内の値を取らなければならない。またPLUSPTYPEが存在せず予測値が[-15.5, 16]の範囲外の場合は、予測値と同じ符号を持つ[-31.5, 31.5]の範囲の全ての値及び0の値を取り得る。PLUSPTYPEが存在する場合、動きベクトルの値は制限されない（付属資料D参照）。

縮小解像度更新モードにおいて、動きベクトルの範囲はおよそ2倍に拡大され、各ベクトル成分は0.5単位か0の値だけを持つように制限される。それゆえ各動きベクトル成分の範囲は、デフォルトの縮小解像度更新モードでは[-31.5, 30.5]であり（付属資料Q参照）、もし無制限動きベクトルモードも使われるならば、より広い範囲となる（付属資料D参照）。

水平または垂直成分の動きベクトルの値が正であることは、参照ピクチャにおいて、予測を行う画素から空間的に右または下方にある画素から予測が行われることを意味する。

無制限動きベクトルモード、拡張予測モード、あるいはデブロッキングフィルタモード（付属資料D、F、J参照）が用いられる場合、あるいは時間、SNR、空間スケーラビリティモードのB、EPピクチャの場合（付属資料O参照）を除いて、動きベクトルにより参照される全ての画素は、符号化されたピクチャ領域内に存在するように動きベクトルは制限される。

#### 4. 2. 4 量子化

オプションの拡張INTRA符号化モードあるいは修正量子化モードを使わない場合、INTRAブロックの最初の係数に対して1種類、その他の係数に対しては31種類の量子化器を用いる。1つのマクロブロックの中では、INTRAブロックの最初の係数を除き、全ての係数に対して同じ量子化器が用いられる。この量子化において、入力に対する判別しきい値は定義しない。INTRAブロックの最初の係数は、直流成分でありステップサイズ8で均一に量子化される。他の31種は、0を中心とする不感帯を有し、2から62までの範囲の偶数値のステップサイズを持ち、量子化復号レベルが均等な間隔を持つ量子化器である。正確な定義は6.2節参照のこと。拡張INTRA符号化モードで使う量子化に関しては付属資料I参照。修正量子化モードで使う量子化に関しては付属資料T参照。

注：量子化のステップサイズが小さい場合、もしオプションの修正量子化モードを使わないならば、変換係数の全ダイナミックレンジを完全には表現することはできない。

#### 4. 3 符号化制御

符号化されたビデオデータの発生量を制御するために、いくつかのパラメータは可変となっている。これらは情報源符号器入力前の処理系、量子化器、有意ブロック判定、コマ落としの制御を含む。このような全体的な符号化制御の中でのこれらのパラメータの使用方法は、本標準では規定しない。

コマ落としは、ピクチャ全体をそのまま廃棄することにより実現される。

復号器は、ビデオ信号の空間解像度と時間解像度との間のトレードオフについて、その優先権を送信することができる。符号器は、呼の始めにデフォルトのトレードオフを送信し、このトレードオフを変更する復号器の要求に符号器が応答できるかどうか示さなければならない。これらの信号の伝送方法は外部の手段（例えばTTC標準JTH245）による。

#### 4. 4 強制画面更新

この機能は、符号化アルゴリズムの I N T R Aモードを強制的に使用することによって実現される。更新パターンは定義されない。逆離散コサイン変換のミスマッチによる誤差の蓄積を抑制するために、Pピクチャにおいてマクロブロックの係数を伝送するとき、各々のマクロブロックを少なくとも132回に1回は I N T R Aモードで符号化しなければならない。同様の必要条件は、オプションのE Pピクチャ（付属資料O参照）を使うときにも適用され、その各マクロブロックは、そのマクロブロックに関して係数が伝送される時に少なくとも132回に1回 I N T R Aあるいは上方予測モードで符号化しなければならない。

#### 4. 5 スタートコードのバイトアライメント

スタートコードのバイトアライメントは、スタートコードの第1ビットが1バイトの第1ビット（最上位ビット）となるように、スタートコードの前に8個未満のゼロビットからなるスタンプ符号語を挿入することによって実現できる。従って、もしその最上位ビットの位置がT T C標準J T - H 2 6 3ビット列の第1ビットから数えて8ビットの倍数であるなら、スタートコードはバイトアライメントされている。すべてのピクチャ、スライス、E O S B Sスタートコードはバイトアライメントされなければならない。G O BスタートコードとE O Sはバイトアライメントしてもよい。

注1：あるピクチャに使われるビット数は可変であるが、常に8ビットの倍数である。

注2：ピクチャスタートコードを、アダプテーションレイヤに渡される論理情報単位（A L \_ S D U）の先頭に揃えることを、T T C標準J T - H 3 2 4はT T C標準J T - H 2 6 3符号器に要求している。

### 5. シンタックスと意味

ビデオシンタックスは、4つの基本的なレイヤからなる階層構造状に構成される。各レイヤは上から順に、

- (1) ピクチャ
- (2) グループオブブロック（G O B）あるいはスライス、あるいはビデオピクチャセグメント
- (3) マクロブロック（M B）
- (4) ブロック

である。

ビデオ多重化部の多重化処理系統図を図6 / J T - H 2 6 3に示す。系統図を解釈するためのガイドラインを以下に示す。

- (1) 矢印のパスは、シンタックス要素の可能なフローを示す。ゼロの長さを持ついかなるシンタックス要素も矢印を用いた図示法によって存在しないものとみなされる。（このように、例えば、P S T U Fフィールドが必須であるにもかかわらずP S T U Fフィールドの長さが0であるかもしれないのでP S T U Fをバイパスする矢印のパスがある）。
- (2) 各シンタックス要素に関する略号と意味は、後の節で定義される。
- (3) 太い実線で示されたシンタックス要素と矢印のパスは、オプションの高品質化を使わない‘ベースライン’モードの操作のシンタックスフローを示す。（このシンタックスは本標準の第1版にも存在したが、ともかく変更していない）。
- (4) 太い点線で示されたシンタックス要素と矢印のパスは、本標準の第1版と第2版の両方に存在したオプションの高品質化のシンタックスフローにおける付加的要素を示す。（このシンタックスはともかく変更していない）。
- (5) 細い実線で示されたシンタックス要素と矢印のパスは、第2版において加えられた付加的オプションの機能に特有であるオプションの高品質化のシンタックスフローにおける新しい付加的要素を示す。（このシンタックスは第1版には存在しなかった）。
- (6) 四角い枠で示されたシンタックス要素フィールドは固定長フィールドを示し、丸い枠で示された方は可変長フィールドを示す。あるシンタックス要素（D Q U A N T）は、可変長と固定長のどちらか一方を持つことができるので、両方のタイプの枠で示される。

- (7) 固定長フィールドは、フィールド長がそのフィールド自身の内容に関するデータによらないフィールドとして定義される。このフィールドの長さはいつも同じであるか、シンタックスフローでの前のデータによって決定される。
- (8) ‘レイヤ’の項目は、別個の存在として理解および図示できるシンタックスの部分を参照するために用いられる。

特に規定しない限り、最上位ビットが最初に伝送される。これは第1ビットであり、本標準の符号表の左端のビットである。特に規定しない限り、すべての未使用ビットまたは予約ビットは“1”にセットされる。予約ビットは、その使用法がITUによって規定されるまで使用してはならない。

### 5.1 ピクチャレイヤ

各ピクチャのデータは、ピクチャヘッダとそれに続くGOBまたはスライスデータによって構成される。最終的にオプションのシーケンス終了(EOS)符号とスタッフ化ビットが後に続く。この構成をオプションのPLUSPTYPEデータフィールドを含まないピクチャに関する図7/JT-H263に示す。PSBIは、CPMに指示された場合にのみ現れる。TRBとDBQUANTは、PTYPEが‘PBフレーム’モードの使用を示す時にだけ現れる(PLUSPTYPEフィールドが存在せずDBQUANTの使用がそこに示されていない場合)。

オプションのPLUSPTYPEデータフィールドは、PTYPEの第6～8ビットで示されるとき存在する。存在するとき、付加されたデータセットはビット列の中のPTYPEの直後でかつPQUANTの直前に含まれる。さらに、CPMとPSBIフィールドは、PQUANTの後に配置されるよりむしろPLUSPTYPEの直後に現れるので、PLUSPTYPEが存在するときピクチャヘッダの中で前に移動される。PLUSPTYPEに続く付加的データのフォーマットを図8/JT-H263に示す。PLUSPTYPEの後の付加的ピクチャヘッダデータに関するすべてのフィールドはオプションであり、それが存在するかどうかはPLUSPTYPEの中で示される。スライス構造モード(付属資料K参照)が用いられるとき、スライスは図7/JT-H263で示される配置の中でGOBの代わりに用いられる。

PSUPPとPEIの組合せは現れなくてもよく、現れるときは繰り返してもよい。EOSとEOSBS+ESBIは現れなくてもよく、EOSまたはEOSBSが現れている場合に限りESTUFは現れてもよい。少なくともあるピクチャスタートコードが一对のEOS符号の間に現れなければEOSは繰り返されてはならない。コマ落としされたピクチャのピクチャヘッダは伝送されない。

# ピクチャレイヤ

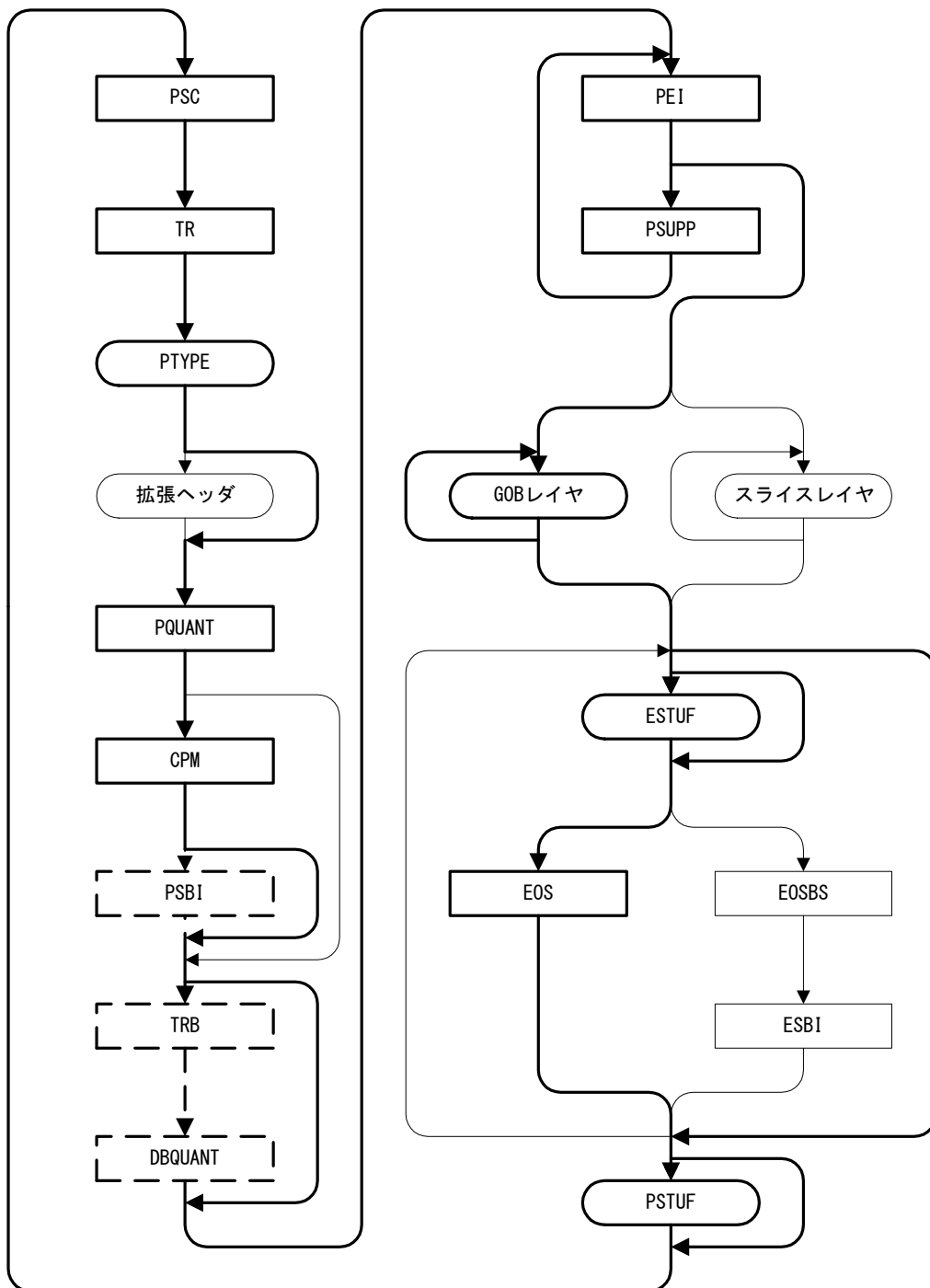


図 6 / JT-H263 ビデオビット列のための多重化処理系統図 (1 / 7)  
(ITU-T H. 263)

拡張ヘッダ

PLUSPTYPE

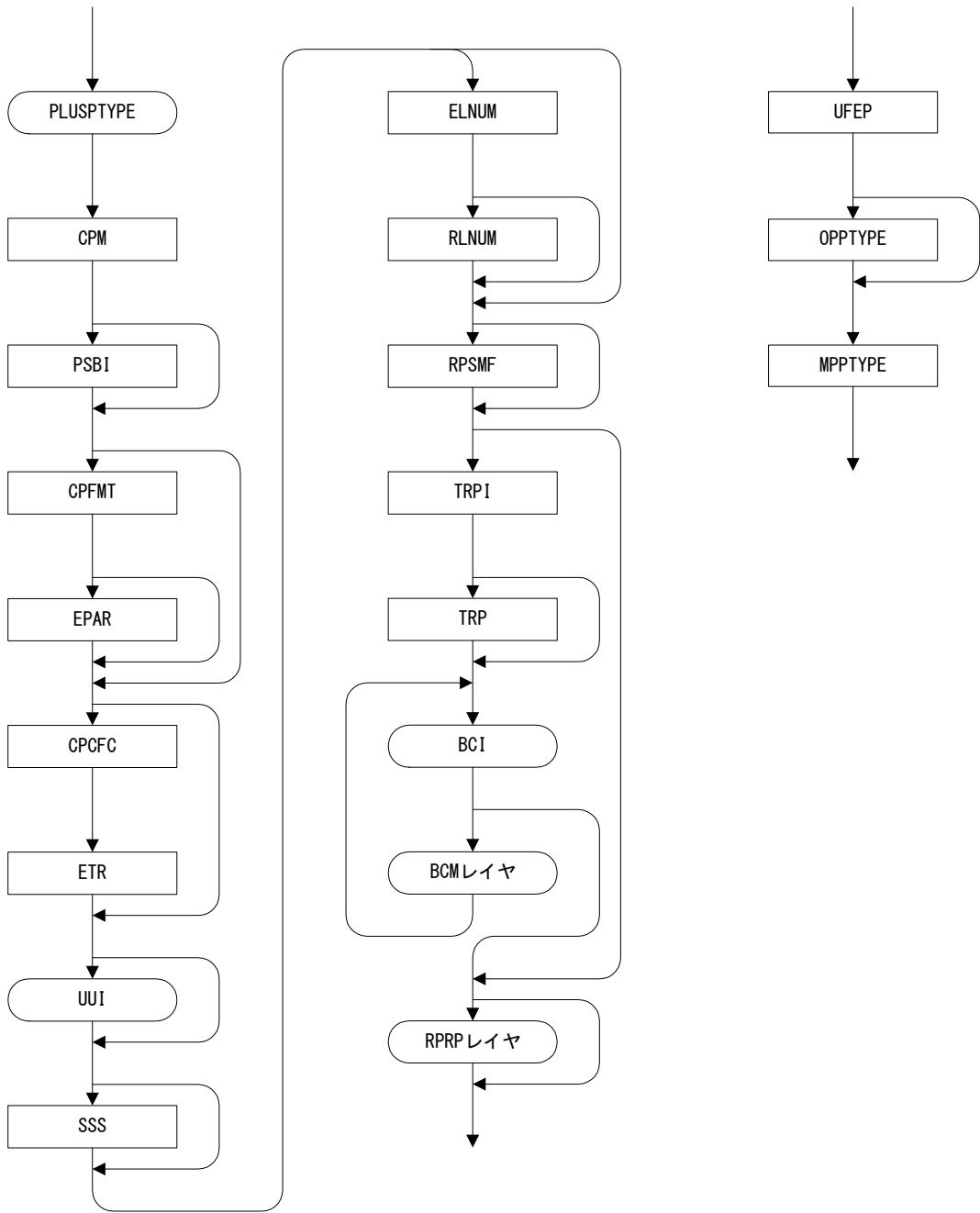


図 6 / JT-H263 ビデオビット列のための多重化処理系統図 (2 / 7)  
(ITU-T H. 263)



GOBレイヤ

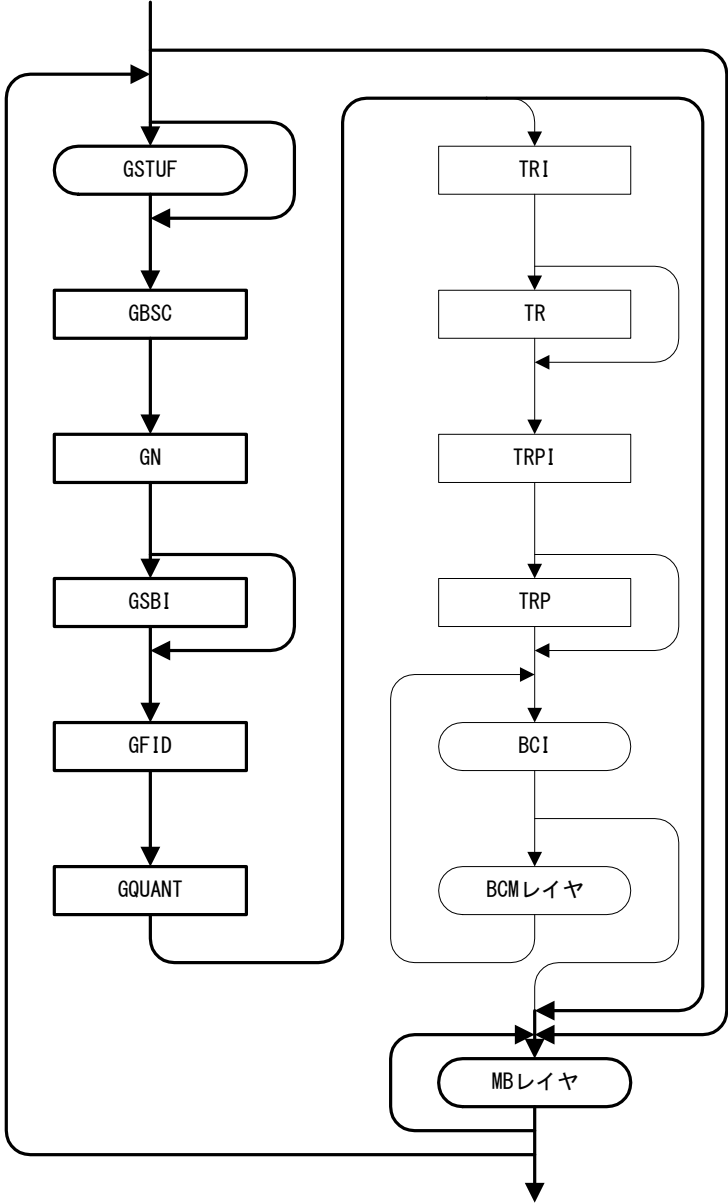


図6 / JT-H263 ビデオビット列のための多重化処理系統図 (3 / 7)  
(ITU-T H. 263)

# スライスレイヤ

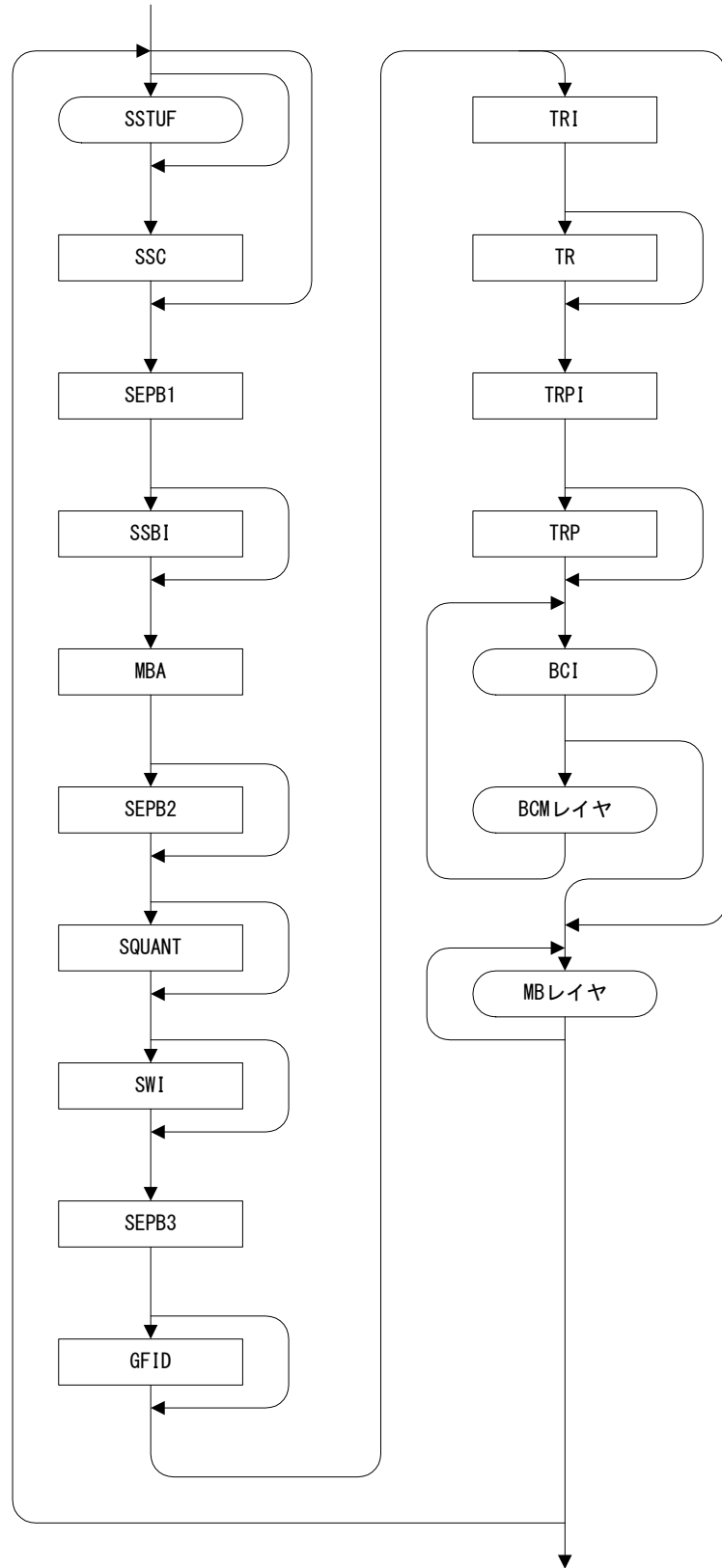
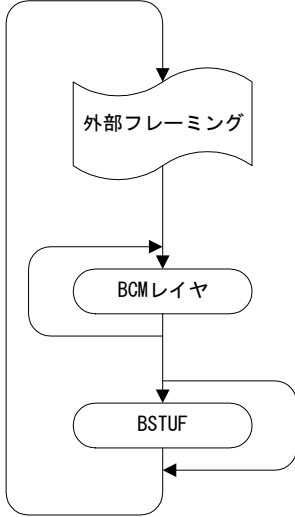


図 6 / JT-H263 ビデオビット列のための多重化処理系統図 (4 / 7)  
(ITU-T H. 263)

**BCM独立論理  
チャンネルレイヤ**



**BCMレイヤ**

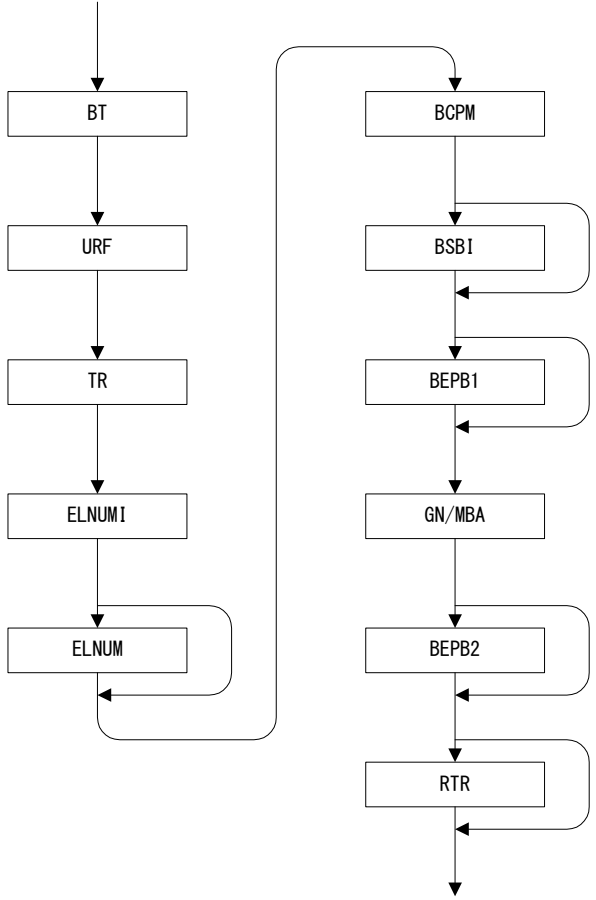


図 6 / JT-H263 ビデオビット列のための多重化処理系統図 (5 / 7)  
(ITU-T H. 263)

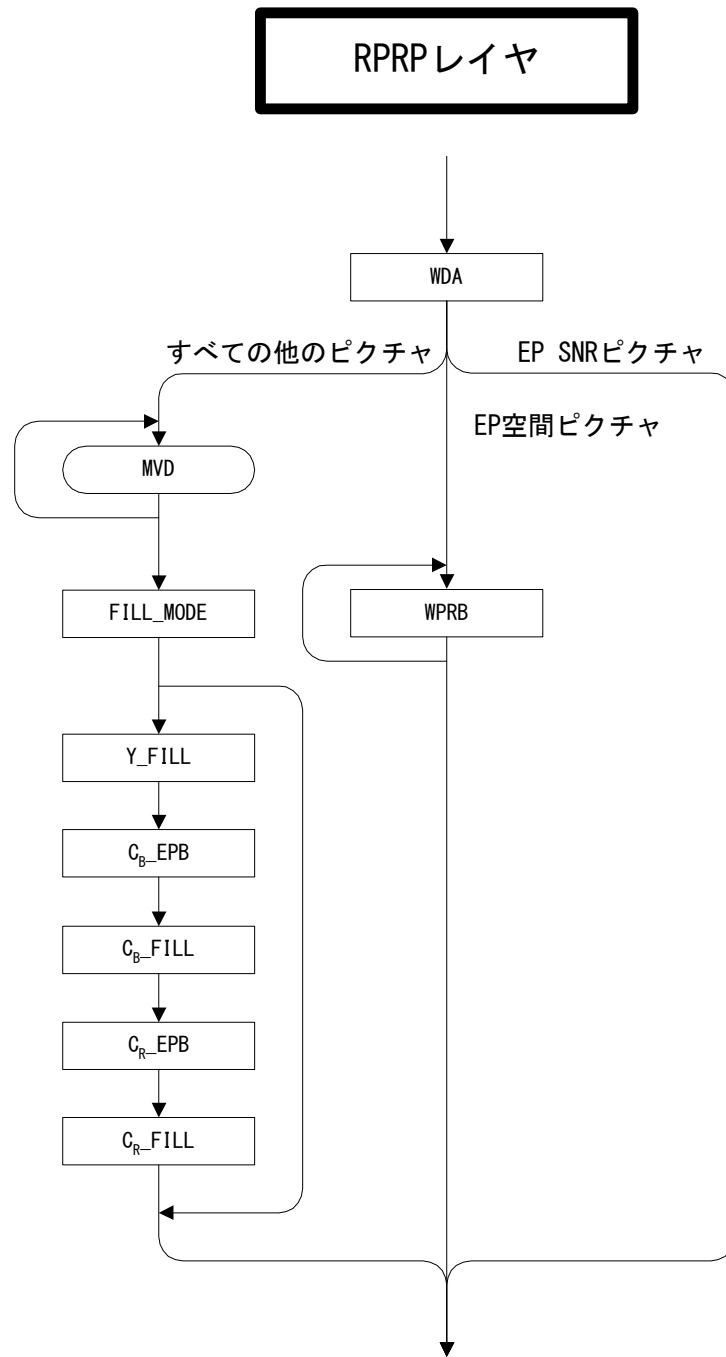


図 6 / JT-H263 ビデオビット列のための多重化処理系統図 (6 / 7)  
(ITU-T H. 263)

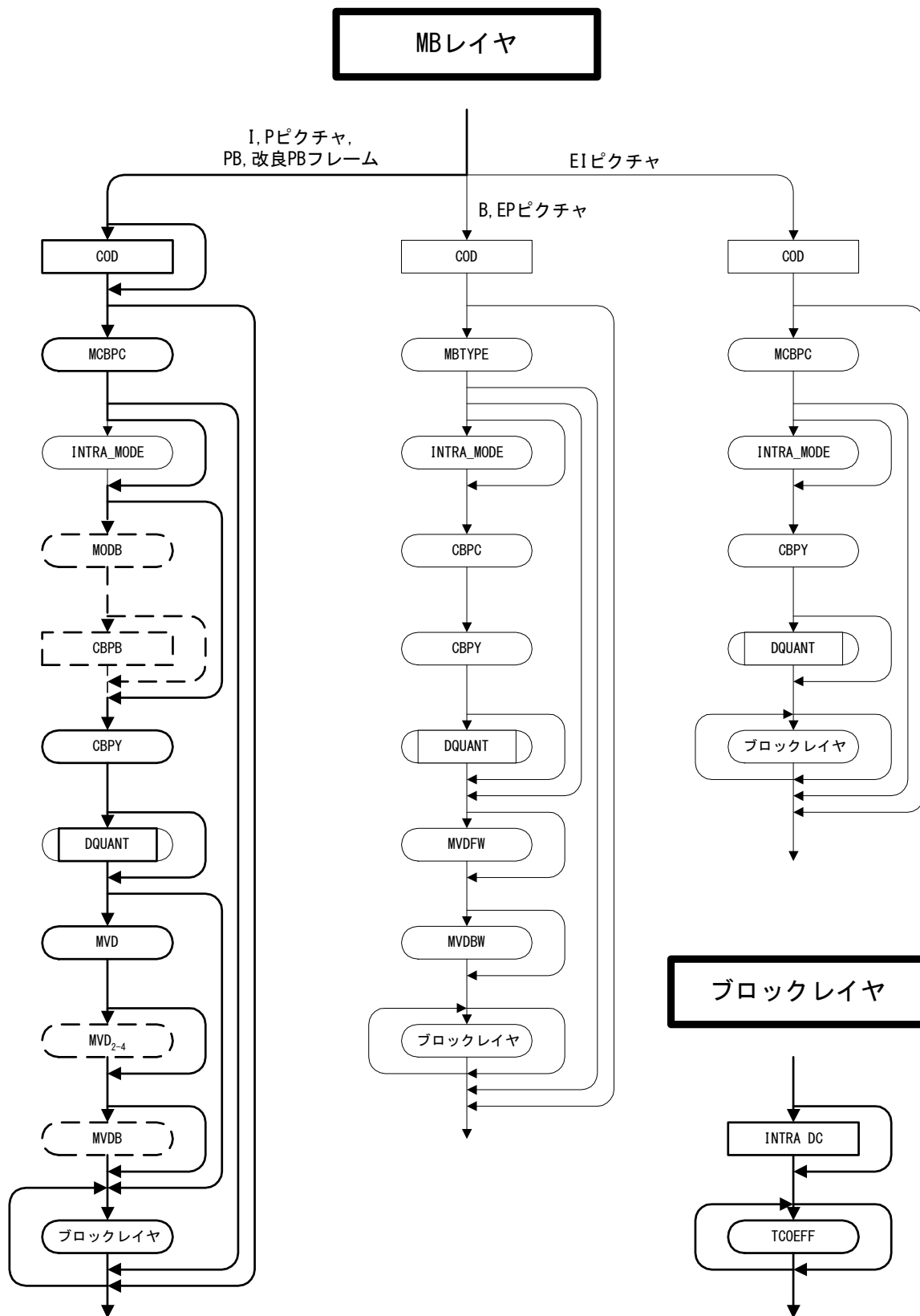


図6 / JT-H263 ビデオビット列のための多重化処理系統図 (7 / 7)  
(ITU-T H. 263)

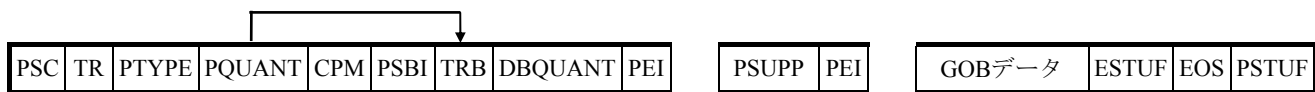


図7 / JT-H263  
(ITU-T H. 263) (オプションのPLUSPTYPE関係のフィールドなし)  
ピクチャレイヤの構成

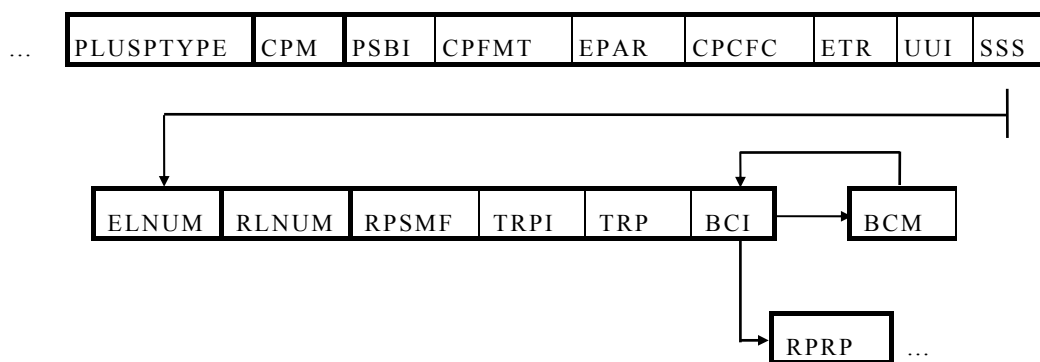


図8 / JT-H263 オプションのPLUSPTYPEに関するフィールドの構成  
(ITU-T H. 263) (存在するときはPTYPE直後に位置する)

### 5. 1. 1 ピクチャスタートコード (PSC) (22ビット)

PSCは22ビットの符号語で、その値は“0000 0000 0000 0000 1 00000”である。すべてのピクチャスタートコードはバイトアライメントされなければならない。これは、スタートコードの第1ビットが1バイトの第1ビット(最上位ビット)になるように、スタートコードの前に必要なビット数のPSTUFを挿入することによって実現しなければならない。

### 5. 1. 2 テンポラルリファレンス (TR) (8ビット)

TRの値は、時間的に前の参照ピクチャヘッダのテンポラルリファレンスに、「画像クロック周波数でスキップあるいは参照されないピクチャ数+1」を加えることによって得られる。TRの解釈は有効な画像クロック周波数に依存する。標準のCIF画像クロック周波数のもとでは、TRは256の可能な値を取り得る8ビットの数である。この演算はLSB8ビットでのみ行うものとする。カスタム画像クロック周波数の使用が示された場合、5.1.8小節の拡張TR(ETR)とTRは、8ビットのLSBを格納したTRと2ビットのMSBを格納したETRにより10ビットの数となる。この場合、この演算はLSB10ビットで行う。オプションのPBフレームまたは改良PBフレームモードでは、TRはPピクチャのみを指す。PBまたは改良PBフレームのBピクチャ部分のテンポラルリファレンスは5.1.2.2小節を参照すること。

### 5. 1. 3 タイプ情報 (PTYPE) (可変長)

ピクチャ全体についての情報を示す。

第1ビット	スタートコードエミュレーションを避けるために、常に“1”。
第2ビット	TTC標準JT-H261との区別のために、常に“0”。
第3ビット	スプリットスクリーンインジケータ。“0”でオフ、“1”でオン。
第4ビット	書画カメラインジケータ。“0”でオフ、“1”でオン。
第5ビット	全画面凍結解除。“0”でオフ、“1”でオン。
第6～8ビット	情報源フォーマット情報。“000”は禁止、“001”はサブQCIF、“010”はQCIF、“011”はCIF、“100”は4CIF、“101”は16CIF、“110”は予約、“111”は拡張PTYPE。

第6～8ビットが拡張P TYPE (PLUS P TYPE)を示す“111”でない場合、以下の5ビットもP TYPEに存在する。

- |        |   |
|--------|---|
| 第9ビット  | ピクチャ符号化タイプ情報。<br>“0”でINTRA (Iピクチャ)、“1”でINTER (Pピクチャ)。 |
| 第10ビット | 無制限動きベクトルモード (付属資料D参照)。<br>“0”でオフ、“1”でオン。             |
| 第11ビット | シンタックス算術符号化モード (付属資料E参照)。<br>“0”でオフ、“1”でオン。           |
| 第12ビット | 拡張予測モード (付属資料F参照)。<br>“0”でオフ、“1”でオン。                  |
| 第13ビット | PBフレームモード (付属資料G参照)。<br>“0”でIまたはPピクチャ、“1”でPBフレーム。     |

スプリットスクリーンインジケータは、復号されたピクチャの上半分と下半分を横並びに表示してもよいことを指示する信号である。このビットはピクチャの符号化、復号には直接影響はない。

全画面凍結解除は、(確認応答がない場合の)パケット再送要求または画面更新要求 (付属資料Cも参照)または画面凍結要求 (付属資料Lも参照)に応答するための符号器からの信号であり、復号器は画面凍結モードから解除され、通常の状態では復号したピクチャを表示することが許される。

もし第6～8ビットが直前のピクチャヘッダと異なる情報源フォーマットを指示するならば、そのピクチャはIピクチャでなければならない。もし拡張P TYPEが第6～8ビットで指示されなければ、オプションの参照ピクチャ再サンプリングモード (付属資料P参照)を使うための能力は外部 (例えばTTC標準J T-H 2 4 5) で交渉されている。

第10～13ビットは、符号器と復号器との交渉の後にだけ使われるオプションモードに関する (それぞれ付属資料D、E、F、Gを参照)。もし第9ビットが“0”にセットされるなら、第13ビットも同じく“0”にセットされなければならない。

第6～8ビットは、外部 (例えばTTC標準J T-H 2 4 5) でカスタム情報源フォーマットあるいは拡張P TYPEを使うことによるのみ可能となる一つまたはそれ以上の他のオプションのモード (付属資料I～KおよびM～T参照)を使うことを許す能力交渉がなされていない場合、拡張P TYPE (PLUS P TYPE)の存在を示す“111”の値を持つてはいけない。第6～8ビットが“111”の値を持たない場合はいつも、拡張P TYPEを使うことによるのみ可能となる付加的モードのすべては、‘オフ’の状態にセットされるように考慮しなければならない。そしてビット列の中で後で明確にスイッチされなければ‘オフ’のままであると推量しなければならない。

#### 5.1.4 拡張P TYPE (PLUS P TYPE) (可変長)

12または30ビットの符号は、P TYPEの第6～8ビットで拡張P TYPEの存在が示される場合に存在する。

PLUS P TYPEは、UFEP、OP P TYPE、MP P TYPEの3つのフィールドからなる。OP P TYPEはUFEPが特別な値の場合のみ存在する。

##### 5.1.4.1 更新全拡張P TYPE (UFEP) (3ビット)

3ビットの固定長符号は、P TYPEの第6～8ビットで‘拡張P TYPE’が指示された場合のみ存在する。“000”にセットされたとき、それぞれのピクチャヘッダに表示する必要のあるその拡張P TYPEフィールド (MP P TYPE)のみが現在のピクチャヘッダに含まれることを示す。“001”にセットされたとき、現在のピクチャヘッダにすべての拡張P TYPEフィールドが含まれることを示す。ピクチャタイプがINTRAまたはEIの場合、このフィールドは“001”にセットしなければならない。

さらに、PLUS P TYPEが連続する各ピクチャシーケンスに存在する場合、少なくとも5秒間または5ピクチャタイムアウト周期のどちらかより長い時間間隔で規定される頻度で、このフィールドは“001”にセットしなければならない。より明確には、(PLUS P TYPEがそれぞれのピクチャ間に存在する場合)タイムアウト

周期は、最後にUFEP = “001”が発生してから5秒以上の時間間隔を示すテンポラルリファレンスを持つ最初のピクチャヘッダかまたは、最後にUFEP = “001”が発生した後の5番目のピクチャのPLUSPTYPEフィールドにUFEP = “001”が出現するようにする（テンポラルリファレンスによって測られるどちらかより長い時間の）周期である。

エンコーダは、エラーの発生しやすい環境ではもっと頻繁にUFEPを“001”にセットすべきである。“000”と“001”以外のUFEPの値は予約されている。

#### 5. 1. 4. 2 PLUSPTYPEのオプション部分 (OPPTYPE) (18ビット)

UFEPが“001”の場合、PLUSPTYPEに以下のビットが存在する。

第1～3ビット	情報源フォーマット、“000”予約、“001”サブQCIF、“010”QCIF、“011”CIF、“100”4CIF、“101”16CIF、“110”カスタム情報源フォーマット、“111”予約。
第4ビット	オプションのカスタムPCF、“0”CIF PCF、“1”カスタムPCF。
第5ビット	オプションの無制限動きベクトル (UMV) モード (付属資料D参照)、“0”オフ、“1”オン。
第6ビット	オプションのシンタックス算術符号化 (SAC) モード (付属資料E参照)、“0”オフ、“1”オン。
第7ビット	オプションの拡張予測 (AP) モード (付属資料F参照)、“0”オフ、“1”オン。
第8ビット	オプションの拡張INTRA符号化 (AIC) モード (付属資料I参照)、“0”オフ、“1”オン。
第9ビット	オプションのデブロッッキングフィルタ (DF) モード (付属資料J参照)、“0”オフ、“1”オン。
第10ビット	オプションのスライス構造 (SS) モード (付属資料K参照)、“0”オフ、“1”オン。
第11ビット	オプションの参照ピクチャ選択 (RPS) モード (付属資料N参照)、“0”オフ、“1”オン。
第12ビット	オプションの独立セグメント復号 (ISD) モード (付属資料R参照)、“0”オフ、“1”オン。
第13ビット	オプションの代替INTER可変長符号 (AIV) モード (付属資料S参照)、“0”オフ、“1”オン。
第14ビット	オプションの修正量子化 (MQ) モード (付属資料T参照)、“0”オフ、“1”オン。
第15ビット	スタートコードエミュレーションを防ぐために“1”にする。
第16ビット	予約、“0”にしなければならない。
第17ビット	予約、“0”にしなければならない。
第18ビット	予約、“0”にしなければならない。

#### 5. 1. 4. 3 PLUSPTYPEが存在するときのPLUSPTYPEの必須部分 (MPPTYPE) (9ビット)

UFEPの値にかかわらず、以下の9ビットはPLUSPTYPEにも存在する。

第1～3ビット	ピクチャタイプコード “000” Iピクチャ (INTRA)、 “001” Pピクチャ (INTER)、 “010” 改良PBフレーム (付属資料M参照)、 “011” Bピクチャ (付属資料O参照)、
---------	---



	“100” EIピクチャ（付属資料O参照）、
	“101” EPピクチャ（付属資料O参照）、
	“110” 予約、
	“111” 予約。
第4ビット	オプションの参照ピクチャ再サンプリング（RPR）モード（付属資料P参照）、“0” オフ、“1” オン。
第5ビット	オプションの縮小解像度更新（RRU）モード（付属資料Q参照）、 “0” オフ、“1” オン。
第6ビット	丸めタイプ（RTYPE）（6.1.2小節参照）。
第7ビット	予約、“0” にしなければならない。
第8ビット	予約、“0” にしなければならない。
第9ビット	スタートコードエミュレーションを防ぐために“1” にする。

Pピクチャ、改良PBフレーム、EPピクチャは第6ビット（Pピクチャの丸めタイプ）に関して動き補償の参照ピクチャと異なる値を持つので、符号器は丸めタイプを制御すべきである。参照ピクチャがIまたはEIピクチャの場合は、第6ビットは任意の値を持つことができる。第6ビットは第1～3ビットがPピクチャまたは改良PBフレームまたはEPピクチャを示しているときだけ“1”にセットできる。他のピクチャタイプに関しては、このビットはいつも“0”にセットしなければならない。

#### 5.1.4.4 PLUSPTYPEの意味

PLUSPTYPEの必須の部分は、ピクチャ毎に変更されるような機能から成る。本来これらは、I、P、改良PB、B、EI、EPの何れかのピクチャタイプを示すビットである。（注：PLUSPTYPEがある場合には付属資料GのPBフレームモードを用いることが出来ず、代わりに付属資料Mの改良PBフレームモードを用いるべきである。）しかし、RPRとRRUモードは同様にピクチャ毎に変わるかも知れないので、これらの使用の指示もここに含まれる。

ピクチャ毎に変更されるのではなく使用され続けるような機能は（5.1.4.5小節で述べられるような明確な方法による場合を除いて）、PLUSPTYPEのオプション部分に置かれている。UFEPが“000”の場合、UFEPが“001”である前のPLUSPTYPEの中で送られているピクチャタイプとモード情報から、欠けているモード情報が推測される。

PLUSPTYPEが存在するが、UFEPが“000”の場合、

- (1) Pピクチャもしくは改良PBフレームの場合（付属資料M参照）、画素アスペクト比、ピクチャ幅、ピクチャ高さは参照ピクチャの場合の値と同じである。
- (2) 高品質化レイヤの時間スケーラビリティBピクチャの場合（付属資料O参照）、もし高品質化レイヤで送られた最後の前のピクチャがEIまたはEPピクチャであるなら、参照レイヤ番号（RLNUM）は高品質化レイヤ番号（ELNUM）と同じである。もし高品質化レイヤで送られた最後のピクチャがBピクチャならば、参照レイヤ番号はそのBピクチャの参照レイヤ番号と同じである。画素アスペクト比、ピクチャ幅、ピクチャ高さは、時間的に後に来る参照レイヤピクチャの値と同じである。
- (3) 時間的に取り囲むEIまたはEPピクチャがBピクチャとして同じ高品質化レイヤに存在する場合、（明示的または暗黙の）RLNUMはELNUMと常に等しくなければならない。（明示的または暗黙の）Bピクチャの画素アスペクト比、ピクチャ幅、ピクチャ高さは、時間的に後に来る参照レイヤピクチャの場合の値と常に等しくなければならない。
- (4) SNR/空間的スケーラビリティEPピクチャの場合（付属資料O参照）、画素アスペクト比、ピクチャ幅、ピクチャ高さは、同じ高品質化レイヤの時間的に前に来る参照ピクチャの場合の値と同じである。

#### 5.1.4.5 ピクチャタイプにおけるモード制限とモード推測規則

特定のモードは特定のピクチャタイプには適用されない。特に以下の制限が適用される。

- (1) 次のモードは I ( I N T R A ) ピクチャの中では適用されない：無制限動きベクトル（付属資料D参照）、拡張予測（付属資料F参照）、代替 I N T E R 可変長符号（付属資料S参照）、参照ピクチャ再サンプリング（付属資料P参照）、縮小解像度更新（付属資料Q参照）。
- (2) 次のモードは B ピクチャの中では適用されない（付属資料O参照）：シンタックス算術符号化（付属資料E参照）、デブロッキングフィルタ（付属資料J参照）、拡張予測（付属資料F参照）。
- (3) 次のモードは E I ピクチャの中では適用されない（付属資料O参照）：無制限動きベクトル（付属資料D参照）、シンタックス算術符号化（付属資料E参照）、拡張予測（付属資料F参照）、参照ピクチャ再サンプリング（付属資料P参照）、縮小解像度更新（付属資料Q参照）、代替 I N T E R 可変長符号（付属資料S参照）。
- (4) 次のモードは E P ピクチャの中では適用されない（付属資料O参照）：シンタックス算術符号化（付属資料E参照）、拡張予測（付属資料F参照）。

上記4項目のリストの1つ以上のモードは、そのモードが禁じられているピクチャタイプ（I、B、E I、E P）の P L U S P T Y P E のオプション部分に、モードフラグ“1”を持ってよい。この条件は許容され、次のパラグラフに述べられるモード推測規則により解釈されなければならない。

モード状態は以下のモード推測規則による。

- (1) 一度モードフラグが P L U S P T Y P E のオプション部分で“1”にセットされると、ビット列中の現在のピクチャと引き続く各ピクチャはそのモードの状態は“オン”に設定されなければならない。
- (2) 現在のピクチャタイプ符号をもつピクチャの中では適用されないどのモードにも、推測された“オフ”状態が設定されなければならない。しかし、（これが、同じ方法で解決されなければならない明らかな矛盾を引き起こさなければ）ビット列の中の引き続く各ピクチャはそのモードで推測された“オン”状態を持たなければならない。階層化スケーラブルビット列の場合（付属資料O参照）、モード状態は同じレイヤのビット列の中からのみ推測されなければならない。
- (3) P L U S P T Y P E のオプション部分を含むまたは P L U S P T Y P E を全く持たない、同じレイヤのピクチャが送られるまで、推測された状態は継続しなければならない。もし P L U S P T Y P E のオプション部分を含む新しいピクチャが送られる場合、新しいメッセージで送られた状態は古い状態に取って代わらなければならない。もし P L U S P T Y P E を持たないピクチャ（P T Y P E の第6～8ビットが“111”でないピクチャ）が送られる場合、P T Y P E に明示的に“オン”とセットされていないすべてのモードに“オフ”状態が割り当てられなければならない。P L U S P T Y P E のオプション部分を含む新しいピクチャが送られるまで、すべてのモードは推測された“オフ”状態を継続しなければならない。
- (4) 次の2つのモードのモードフラグは P L U S P T Y P E の必須部分に現れるので、それらのモード状態推測を必要としない。それらは参照ピクチャ再サンプリングモード（付属資料P参照）、縮小解像度更新モード（付属資料Q参照）である。現在のピクチャがそのモードの使用を許さないならば、これらどちらのモードのモードフラグもセットしてはならない。例えば、縮小解像度更新モードビットを I N T R A ピクチャにおいてはセットしてはならない。

#### 5. 1. 4. 6 モード相互作用制限

特定のモードは、他の特定のモードと組み合わせて使うことはできない。

- (1) シンタックス算術符号化モード（付属資料E参照）は代替 I N T E R 可変長符号モード（付属資料S参照）または修正量子化モード（付属資料T参照）と共に用いてはならない。
- (2) P L U S P T Y P E がある場合は、無制限動きベクトルモード（付属資料D参照）はシンタックス算術符号化モード（付属資料E参照）と共に用いてはならない。
- (3) 独立セグメント復号モード（付属資料R参照）は参照ピクチャ再サンプリングモード（付属資料P参照）と共に用いてはならない。
- (4) 独立セグメント復号モード（付属資料R参照）は、スライス構造モード（付属資料K参照）の矩形スライスサブモードを同時に使用しないならば、スライス構造モードと共に用いてはならない。

#### 5. 1. 4. 7 CPM (1ビット) と PSBI (2ビット) のピクチャヘッダでの位置

ピクチャヘッダでのCPMとPSBIフィールドの位置は、PLUSPTYPEが有るか無いかによって異なる(5. 1. 20小節、5. 1. 21小節参照)。PLUSPTYPEがある場合、CPMはピクチャヘッダのPLUSPTYPEの直後に続く。PLUSPTYPEがない場合、ピクチャヘッダのPQUANTの直後に続く。PSBIは(もしCPM=“1”ならば)常にCPMの直後に続く。

#### 5. 1. 5 カスタム画像フォーマット (CPFMT) (23ビット)

カスタム画像フォーマットの使用がPLUSPTYPEのなかで通知され、UFEPが“001”である場合のみ存在する、23ビットの固定長符号語。CPFMTが存在するとき、これは次のビットから成る。

- 第1～4ビット 画素アスペクト比符号:表5/JT-H263のPAR値に対する4ビットのインデックス。拡張PARに対しては、正確な画素アスペクト比はEPARの中で指定されなければならない(5. 1. 6小節参照)。
- 第5～13ビット ピクチャ幅インジケーション: 範囲[0, ..., 511]、ライン当たりの画素数 =  $(PW I + 1) \times 4$
- 第14ビット スタートコードエミュレーションを防ぐため“1”
- 第15～23ビット ピクチャ高さインジケーション: 範囲[1, ..., 288]、ライン数 =  $PH I \times 4$

表5/JT-H263 PAR符号の定義  
(ITU-T H. 263)

PAR 符号	画素アスペクト比
0000	禁止
0001	1:1 (正方)
0010	12:11 (4:3ピクチャ用CIF)
0011	10:11 (4:3ピクチャ用525タイプ)
0100	16:11 (16:9ピクチャ用伸張CIF)
0101	40:33 (16:9ピクチャ用伸張525タイプ)
0110-1110	予約
1111	拡張 PAR

#### 5. 1. 6 拡張画素アスペクト比 (EPAR) (16ビット)

CPFMTが存在し拡張PARがそこで指示される場合のみ存在する、16ビットの固定長符号語。EPARが存在するとき、これは次のビットから成る。

- 第1～8ビット PAR幅: “0”は禁止。PAR幅の自然2進数表現。
- 第9～16ビット PAR高さ: “0”は禁止。PAR高さの自然2進数表現。

PAR幅とPAR高さは互いに素でなければならない。

#### 5. 1. 7 カスタム画像クロック周波数符号 (CPCFC) (8ビット)

PLUSPTYPEが存在し、UFEPが“001”であり、画像クロック周波数がPLUSPTYPEで通知される場合のみ存在する、8ビットの固定長符号語。CPCFCが存在するとき、これは次のビットから成る。

- 第1ビット クロック変換符号: “0”はクロック変換ファクタが1000、“1”は1001を示す。
- 第2～8ビット クロック除数: “0”は禁止。クロック除数の自然2進数表現の値。

カスタム画像クロック周波数は、 $1800000 / (\text{クロック除数} \times \text{クロック変換ファクタ})$  Hzで与えられる。

テンポラルリファレンスカウンタは、秒を単位とした画像クロック周波数の逆数を単位としてカウントしなければならない。PCFが前のピクチャで指定されたものから変わるときは、現在のピクチャのテンポラルリファレンスは前のPCFで測定され、したがって新しいPCFは未来のピクチャのテンポラルリファレンスの解釈に対してのみ有効である。

#### 5. 1. 8 拡張テンポラルリファレンス (ETR) (2ビット)

カスタム画像クロック周波数が使用される場合のみ (UFEP値に関係なく) 存在する、2ビットの固定長符号語である。これは5.1.2小節に定義された10ビット番号のMSB2ビットである。

#### 5. 1. 9 非限定無制限動きベクトルインジケータ (UUI) (可変長)

オプションである無制限動きベクトルモードがPLUSPTYPEに指示され、UFEPが“001”の場合のみ存在する、1または2ビットの可変長符号語である。UUIが存在するとき、用いられる動きベクトルの範囲の有効な制限を示す。

UUI = “1” : 動きベクトルの範囲は表D. 1 / JT-H263と表D. 2 / JT-H263に従って制限される。

UUI = “01” : 動きベクトルの範囲はピクチャサイズにのみ制限される。

#### 5. 1. 10 スライス構造サブモードビット (SSS) (2ビット)

オプションであるスライス構造モード (付属資料K参照) がPLUSPTYPEに指示され、UFEPが“001”の場合のみ存在する、2ビットの固定長符号語である。スライス構造モードが使用されているがUFEPが“001”でない場合、最後に送られたSSS値は有効で有り続けなければならない。

第1ビット 矩形スライス、“0”はフリーランスライス、“1”は矩形スライスを示す。

第2ビット 任意スライス順序、“0”は順次、“1”は任意の順序を示す。

#### 5. 1. 11 高品質化レイヤ番号 (ELNUM) (4ビット)

(UFEP値に関係なく) オプションである時間、SNR、空間スケーラビリティモードが使用される場合のみ存在する、4ビットの固定長符号語である。特定の高品質化レイヤは高品質化レイヤ番号ELNUMにより識別される。レイヤ間のピクチャの一致はテンポラルリファレンスにより達成される。ピクチャサイズは、既存の情報源フォーマットフィールドを用いて各高品質化レイヤの中に示されるか、または参照レイヤとの関係により推測される。基本レイヤの上の第1高品質化レイヤは高品質化レイヤ番号2とされ、基本レイヤは番号1を持つ。

#### 5. 1. 12 参照レイヤ番号 (RLNUM) (4ビット)

オプションである時間、SNR、空間スケーラビリティモード (付属資料O参照) が使用され、UFEPが“001”の場合のみ存在する、4ビットの固定長符号語である。参照アンカーとして用いられるピクチャのレイヤ番号は、参照レイヤ番号 (RLNUM) として識別される。各レイヤの時間の一致はテンポラルリファレンスにより達成される。

同じ高品質化レイヤにあり、時間的に取り囲んでいるEIまたはEPピクチャをもつ高品質化レイヤのBピクチャの場合、RLNUMはELNUMと等しくなければならない (付属資料O参照)。

#### 5. 1. 13 参照ピクチャ選択モードフラグ (RPSMF) (3ビット)

参照ピクチャ選択モードが使用され、UFEPが“001”の場合のみ存在する、3ビットの固定長符号語である。RPSMFが存在するとき、これは符号器がどのタイプの逆方向チャンネルメッセージを必要とするかを示す。もし参照ピクチャ選択モードが使用されているがRPSMFがない場合、最後に送られたRPSMF値が有効であり続けなければならない。

- 1 0 0 : ACKもNACK信号も必要でない。
- 1 0 1 : ACK信号が返される必要がある。
- 1 1 0 : NACK信号が返される必要がある。
- 1 1 1 : ACKもNACKも返される必要がある。
- 0 0 0 ~ 0 1 1 : 予約

#### 5. 1. 14 予測用テンポラルリファレンスインジケーション (TRPI) (1ビット)

(UFEF値に関係なく) オプションである参照ピクチャ選択モードが使用される場合のみ存在する、1ビットの固定長符号語である。TRPIが存在するとき、これは後に来るTRPフィールドの存在を示す。

- 0 : TRPフィールドは存在しない。
- 1 : TRPフィールドは存在する。

ピクチャヘッダがIまたはEIピクチャを示すときはいつでも、TRPIは“0”でなければならない。

#### 5. 1. 15 予測用テンポラルリファレンス (TRP) (10ビット)

TRPが (TRPIに示されるように) 存在するときは、Bピクチャの場合を除いて、符号器の予測に用いられるテンポラルリファレンスを示す。Bピクチャに対しては、テンポラルリファレンスTRPをもつピクチャが前方予測に使われる。(時間的逆方向の予測は常に時間的に直後に来るピクチャを用いる。) TRPは10ビットの番号である。カスタム画像クロック周波数が参照ピクチャで用いられない場合、TRPのMSB2ビットはゼロで、LSBは参照ピクチャのピクチャヘッダにあるTRの8ビットである。カスタム画像クロック周波数が参照ピクチャで用いられた場合、TRPは参照ピクチャヘッダからのETRとTRの結合から成る10ビットの番号である。

参照ピクチャ選択モードでないときのように、TRPが存在しない場合は、時間的に前の最も最近のアンカーピクチャが予測に用いられなければならない。TRPは、次のPSC、GSCまたはSSCまで有効である。

#### 5. 1. 16 逆方向チャンネルメッセージインジケーション (BCI) (可変長)

オプションである参照ピクチャ選択モードが使われている場合のみ存在する1または2ビットの可変長フィールドである。BCIが“1”であることは、その後にオプションである逆方向チャンネルメッセージ (BCM) フィールドがあることを示す。BCIが“01”であることは、ビデオ逆方向チャンネルメッセージフィールドが無いこと、またはその終わりであることを示す。BCMとBCIの組み合わせは存在しなくてもよく、また存在する場合繰り返されてもよい。オプションである参照ピクチャ選択モードのビデオ多重サブモードが使われない場合、BCIは“01”にセットされなければならない。

#### 5. 1. 17 逆方向チャンネルメッセージ (BCM) (可変長)

N. 4. 2小節に定められるようなシンタックスをもつ逆方向チャンネルメッセージであり、前に来るBCIフィールドが存在し、それが“1”にセットされる場合のみ存在する。

#### 5. 1. 18 参照ピクチャ再サンプリングパラメータ (RPRP) (可変長)

オプションである参照ピクチャ再サンプリングモードビットがPLUSPTYPEにセットされている場合のみ存在する可変長フィールドである。このフィールドは参照ピクチャ再サンプリングモード (付属資料P参照) のパラメータを伝える。また符号化された前のピクチャと異なるピクチャサイズをもつINTER符号化されたピクチャのピクチャヘッダの発生により、参照ピクチャ再サンプリングモードを暗黙的に設定することも可能であることに注意が必要である。その場合RPRPフィールドは存在せず、参照ピクチャ再サンプリングモードビットはセットされない。

#### 5. 1. 19 量子化器情報 (PQUANT) (5ビット)

次のGQUANTあるいはDQUANTによって更新されるまでピクチャで使われる量子化器QUANTを示す、5ビットの固定長符号語である。符号語は、ステップサイズの半分の値で1から31の範囲の値をとるQUANT値の自然2進数表現である。

### 5. 1. 20 コンティニューアスペレzensマルチポイントとビデオ多重 (CPM) (1ビット)

オプションのコンティニューアスペレzensマルチポイントとビデオ多重モード (CPM) の使用を知らせる1ビットの符号語で、“0”でオフ、“1”でオンである。CPMの使用については付属資料Cを参照のこと。CPMは、PLUSPTYPEが存在する場合はPLUSPTYPEの直後に続くが、PLUSPTYPEが存在しない場合はピクチャヘッダのPQUANTに続く。

### 5. 1. 21 ピクチャサブビット列インジケータ (PSBI) (2ビット)

コンティニューアスペレzensマルチポイントとビデオ多重モードがCPMによって指示された場合だけ現れる、2ビットの固定長符号語である。符号語は、ピクチャヘッダ、および次のピクチャスタートコードあるいはGOBスタートコードまで続く全ての情報のサブビット列番号の自然2進数表現である (付属資料Cも参照)。CPMが“1”の場合、PSBIはCPMの直後に続く (PLUSPTYPEが存在するか否かによって、ピクチャヘッダ内でのCPMとPSBIの位置は異なる。)

### 5. 1. 22 PBフレームでのBピクチャのテンポラルリファレンス (TR<sub>B</sub>) (3または5ビット)

TR<sub>B</sub>は、PTYPEまたはPLUSPTYPEが‘PBフレーム’または‘改良PBフレーム’ (付属資料GおよびM参照) を指示した場合に現れる。これは、最後のPピクチャ、Iピクチャ、或いはPBまたは改良PBフレームのP部分から、PBまたは改良PBフレームのBピクチャ部分の前までに、(29.97Hz、またはCPFCに示されるカスタム画像クロック周波数での) 伝送されないまたは参照されないピクチャ数を示す。符号語は「伝送されないピクチャ数+1」の自然2進数表現である。標準のCIF画像クロック周波数に対しては3ビットで、カスタム画像クロック周波数が使われる場合は5ビットに拡張される。伝送されないピクチャの最大数は、標準CIF画像クロック周波数に対しては6、カスタム画像クロック周波数が使われるときは30である。

### 5. 1. 23 PBフレームでのBピクチャの量子化情報 (DBQUANT) (2ビット)

DBQUANTは、PTYPEまたはPLUSPTYPEが‘PBフレーム’または‘改良PBフレーム’ (付属資料GおよびM参照) を指示した場合に現れる。復号処理においては、量子化パラメータQUANTは各マクロブロック毎に与えられる。PBフレームでは、QUANTはPブロックに対して使用され、Bブロックに対しては異なる量子化パラメータBQUANTが使用される。QUANTは1から31までの値をとる。表6/JT-H263に定義されるようなQUANTとBQUANTの関係をDBQUANTは示す。この表で“/”は切捨ての除算を意味する。BQUANTは1から31の範囲の値をとる。つまり表6/JT-H263の演算の結果、BQUANTが31より大きくなった場合、その値は31に制限される。

表6/JT-H263 DBQUANTの符号と、QUANTとBQUANTの関係  
(ITU-T H. 263)

DBQUANT	BQUANT
00	$(5 \times \text{QUANT}) / 4$
01	$(6 \times \text{QUANT}) / 4$
10	$(7 \times \text{QUANT}) / 4$
11	$(8 \times \text{QUANT}) / 4$

### 5. 1. 24 拡張用挿入情報 (PEI) (1ビット)

1ビットの符号語で、“1”にセットされる時、後ろにオプションデータ領域が続く。

### 5. 1. 25 付加拡張情報 (PSUPP) (0/8/16・・・ビット)

PEIが“1”のとき、8ビットのデータ (PSUPP) 及びさらに9ビットの付加拡張情報が続くかどうかを示すPEI1ビットからなる9ビットが続き、以下同様に繰り返す。符号器は、付属資料Lに規定されるようにPSUPPを使用しなければならない。付属資料Lに記述されている拡張能力に対応していない復号器は、PEIが“1”の時にはPSUPPを捨てるように設計されなければならない。このことは拡張能力に利用されるビット列がこれらの能力に対応していない復号器によって変更されることなく用いられ得るように付属資料Lの拡張能力に対しての後方互換性を可能にする。

### 5. 1. 26 スタッフ化 (ESTUF) (可変長)

8 個未満の“0”ビットで構成される可変長符号語。符号器はEOS符号語の前に直接この符号語を挿入してもよい。必須なバイトアライメントを達成するために、符号器はEOSBS符号語の直前に必要に応じてこの符号語を挿入しなければならない。もしESTUFが現れるなら、ESTUFの最終ビットは1バイトの最終ビット(最下位ビット)でなければならない。これによってEOSあるいはEOSBS符号語の先頭がバイトアライメントされる。復号器はESTUFを捨てるように設計されなければならない。EOSBSの記述およびその使用については付属資料Cを参照。

### 5. 1. 27 シーケンス終了情報 (EOS) (22ビット)

22ビットの符号語で、その値を“0000 0000 0000 0000 1 1111”とする。この符号語を挿入するか否かは符号器による。EOSはバイトアライメントされてもよい。これは、EOS符号語の第1ビットが1バイトの第1ビット(最上位ビット)となるようにEOS符号語の前にESTUFを挿入することによって実現できる。EOSは各々の一対のEOS符号の間になくともひとつのピクチャスタートコードが現れない限りは繰り返されてはならない。

### 5. 1. 28 スタッフ化 (PSTUF) (可変長)

8 個未満の“0”ビットで構成される可変長符号語。符号器は次のPSCのバイトアライメントのためにこの符号語を挿入しなければならない。PSTUFを含めたビデオビット列が、TTC標準JT-H263ビット列の第1ビットから8ビットの倍数となるように、PSTUFの最終ビットは1バイトの最終ビット(最下位ビット)でなければならない。復号器はPSTUFを捨てるように設計されなければならない。

もし何らかの理由で符号器がある期間画像の符号化を停止し、その後符号化を再開する場合、前のピクチャの最後の最大7ビットが、符号器が符号化を再開するまで送信されなくなることを防ぐために、PSTUFは符号器が止まる前に伝送されなければならない。

## 5. 2 グループオブブロック (GOB) レイヤ

各GOBデータは、GOBヘッダと後に続くマクロブロックデータによって構成されている。その構成を図9/JT-H263に示す。各GOBは、1つ、あるいはそれ以上のマクロブロック列を含んでいる。各ピクチャの最初のGOB(番号0)では、GOBヘッダが伝送されてはならない。他の全てのGOBについては、符号器の手法によってはGOBヘッダは空でもよい。復号器は外部の手段、例えばTTC標準JT-H245によって相手側の符号器に空でないGOBヘッダだけを送信するように通知することができる。GBSCが現れる時GSTUFが現れてもよい。GBSCが現れる時、GN、GFID、GQUANTが現れる。ピクチャヘッダにおいてCPMが“1”の時、GSBIが現れる。

GSTUF	GBSC	GN	GSBI	GFID	GQUANT	MBデータ
-------	------	----	------	------	--------	-------

図9/JT-H263 GOBレイヤの構成  
(ITU-T H. 263)

### 5. 2. 1 スタッフ化 (GSTUF) (可変長)

8 個未満の“0”ビットから構成される可変長符号語。符号器はGBSC符号語の前に直接この符号語を挿入してもよい。もしGSTUFが現れるならば、GBSC符号語の先頭がバイトアライメントされるように、GSTUFの最終ビットは1バイトの最終ビット(最下位ビット)でなければならない。復号器はGSTUFを捨てるように設計されなければならない。

### 5. 2. 2 GOBスタートコード (GBSC) (17ビット)

17ビットの符号語で、その値は“0000 0000 0000 0000 1”である。GOBスタートコードはバイトアライメントされてもよい。これはスタートコードの第1ビットが1バイトの最初のビット(最上位ビット)になるようにスタートコードの前にGSTUFを挿入することで実現できる。

### 5. 2. 3 グループ番号 (GN) (5ビット)

5ビットの固定長符号語。そのビットはGOB番号の2進数表現である。GOB番号0では、GSTUF、GBSC、GN、GSBI、GFID、そしてGQUANTを含むGOBヘッダは空である。グループ番号0としてはPSCで使用されている。グループ番号1から17は標準画像フォーマットのGOBヘッダの中で使用される。グループ番号1から24はカスタム画像フォーマットのGOBヘッダの中で使用される。CPM="0"の時グループ番号16から28はスライスヘッダの中でエミュレートされ(付属資料K参照)、CPM="1"の時グループ番号25から27および29はスライスヘッダの中でエミュレートされる(付属資料K参照)。グループ番号31はEOS符号で使用され、グループ番号30はEOSBS符号の中で使用される。

### 5. 2. 4 GOBサブビット列インジケータ (GSBI) (2ビット)

2ビットの固定長符号語で、ピクチャヘッダの中でCPMが"1"の時だけ現れる。その符号語は、GOBヘッダおよびそれに続く、次のピクチャスタートコードかGOBスタートコードまでの全ての情報のためのサブビット列の番号の自然2進数表現である(付属資料C参照)。

### 5. 2. 5 GOBフレームID (GFID) (2ビット)

2ビットの固定長符号語。GFIDは、与えられたピクチャの全てのGOB(あるいはスライス)ヘッダで同じ値を持たなければならない。さらに、もしあるピクチャヘッダのPTYPEがその前に伝送されたピクチャのものと同じで、PLUSPTYPEが現れないならば、GFIDはその前のピクチャと同じ値を持たなければならない。しかし、もしあるピクチャヘッダのPTYPEがその前に伝送されたピクチャヘッダのPTYPEと異なるならば、そのピクチャのGFIDの値は前のピクチャの値と異ならなければならない。

もしPLUSPTYPEが現れて、PTYPE、およびPLUSPTYPE、そしてCPFMT、EPAR、CPCFC、SSS、ELNUM、RLNUM、UUI、RPSMF、およびRPRPの中のすべての存在するフィールドが前のピクチャに対して同様に影響し続けるならば、GFIDの値は(同一のレイヤ内の)前のピクチャに対するものと同じの値でなければならない。そうでなければ、GFIDは前のピクチャに対するものと異ならなければならない。

### 5. 2. 6 量子化器情報 (GQUANT) (5ビット)

5ビットの固定長符号語で、次のGQUANTあるいはDQUANTによって更新されるまで、ピクチャの残りの部分で使用される量子化器QUANTを指示する。符号語はQUANTの値の自然2進数表現であり、ステップサイズの半分の値で、1から31の値を取り得る。

## 5. 3 マクロブロック (MB) レイヤ

各マクロブロックのデータは、マクロブロックヘッダと後に続くブロックデータから構成される。その構成を図10/JT-H263に示す。CODは、'INTRA'タイプではないピクチャでだけ、そのピクチャ中の各マクロブロックに対して現れる。MCBPCは、CODによって指示された時、あるいはピクチャが'INTRA'タイプの時に現れる。MODBは、PTYPEが'PBフレーム'を指示する時、MBタイプの0から4を表す。CBPY、DQUANT、MVD、MVD<sub>2,4</sub>は、MCBPCによって指示された時に現れる。CBPBとMVDBは、MODBによって指示された時だけ現れる。ブロックデータは、MCBPCとCBPYによって指示された時に現れる。MVD<sub>2,4</sub>は拡張予測モード(付属資料F参照)あるいはデブロッキングフィルタモード(付属資料J参照)にある時にだけ現れる。MODBとCBPB、MVDBはPBフレームモードの場合だけ現れる(付属資料G参照)。シンタックス算術符号化モードでのシンボルの符号化については付属資料Eを参照すること。B、EI、およびEPピクチャにおけるマクロブロックレイヤの符号化については付属資料Oを参照すること。

COD	MCBPC	MODB	CBPB	CBPY	DQUANT	MVD	MVD <sub>2</sub>	MVD <sub>3</sub>	MVD <sub>4</sub>	MVDB	ブロックデータ
-----	-------	------	------	------	--------	-----	------------------	------------------	------------------	------	---------

図10/JT-H263 マクロブロックレイヤの構成  
(ITU-T H. 263)



### 5. 3. 1 符号化マクロブロックインジケータ (COD) (1ビット)

1ビットの符号で、“0”にセットされている時、そのマクロブロックが符号化されていることを表す。もし“1”にセットされているなら、このマクロブロックについてはそれ以上の情報は伝送されない。この場合復号器は、そのマクロブロックをそのブロック全体に対する動きベクトルがゼロで、係数データを持たないINTERマクロブロックとして扱わなければならない。CODは、‘INTRA’タイプではないピクチャでだけ、そのピクチャ中の各マクロブロックに対して現れる。

注：拡張予測モード（付属資料F参照）において、もしCODが“1”にセットされるなら、オーバーラップブロック動き補償も実行される。デブロッキングフィルタモード（付属資料J参照）においてデブロッキングフィルタは、CODが“1”にセットされているマクロブロックの幾つかの画素の値に影響を与えることもあり得る。

### 5. 3. 2 マクロブロックタイプと色差信号の符号化ブロックパターン (MCBPC) (可変長)

MCBPCはマクロブロックタイプと、色差信号の符号化ブロックパターンについての情報を与えている可変長符号語である。MCBPCの符号語は、表7/JT-H263と表8/JT-H263に与えられる。MCBPCは符号化されたマクロブロックに常に含まれる。

ビットスタッフ化のために表中の特別な符号語が利用可能である。この符号語は復号器によって捨てられるべきである。もし改良PBフレームモードがMPPTYPEの第1～3ビットによって指示され、カスタム情報源フォーマットがOPPTYPEの第1～3ビットによって指示されるならば、MBAは（スタートコードエミュレーションを防ぐために）ピクチャの最初のマクロブロックの前でのスタッフ化を指示してはならない。

注：復号器はビット列内のピクチャ、GOB、またはスライススタートコードの位置直前のビットスタッフ化を示すマクロブロックタイプを許容するように設計されるべきである。しかしながら、符号器はマクロブロックレイヤスタッフ化をこのような方法で用いるべきではない（復号器がこれをサポートする必要性が明確化される以前に設計されたかもしれない復号器との相互接続性のため）。

マクロブロックタイプは、マクロブロックについての情報と、どのデータ要素が現れるかを示す。マクロブロックタイプとそこに含まれたデータ要素を表9/JT-H263と表10/JT-H263に示す。マクロブロックタイプ5（表8/JT-H263のインデックス21から24）は、拡張されたPTYPE（PLUSPTYPE）がピクチャヘッダに現れ、かつ拡張予測モード（付属資料F参照）あるいはデブロッキングフィルタモード（付属資料J参照）のいずれかが使用されていない限りは現れてはならない、またピクチャの最初のマクロブロックに対して現れてはならない。同様に、符号器はスタートコードエミュレーションを防ぐためにビット列の中で7個の連続する“0”（COD=“0”がその後続く特定のINTRADC符号によって引き起こされることがあり得る）のすぐ後にマクロブロックタイプ5に対するMCBPC符号が続くことを許してはならない。マクロブロックタイプ5に対する符号は（ピクチャの最初でないマクロブロックに対しては）この要求を満たすために必要ならば、スタッフ化を先行させることができる。

表7 / JT-H263 MCBPCの可変長符号表 (Iピクチャ用)  
(ITU-T H. 263)

インデックス	MBタイプ	CBPC (56)	ビット数	符号語
0	3	00	1	1
1	3	01	3	001
2	3	10	3	010
3	3	11	3	011
4	4	00	4	0001
5	4	01	6	0000 01
6	4	10	6	0000 10
7	4	11	6	0000 11
8	スタッフ符号	--	9	0000 0000 1

色差信号の符号化ブロックパターンは、オプションの拡張 INTRA 符号化モードが使用されていないならば、少なくとも1つの INTRADC 以外の変換係数が伝送される時の  $C_B$  かつ、または  $C_R$  ブロックを意味する (INTRADC は INTRA ブロックの直流係数である。5.4.1 小節参照)。符号化ブロックパターンの  $CBPC_5$  と  $CBPC_6$  について、もしブロック N において INTRADC 以外の係数が現れるならば  $CBPC_N$  は “1”、それ以外は “0” である。もし拡張 INTRA 符号化が使用されているならば、使用法は似ているが、しかし INTRADC 係数はその他の係数と同じ様に指示される (付属資料 I 参照)。ブロックの番号付けは図 5 / JT-H263 で与えられる。MCBPC がスタッフ符号の時、マクロブロックレイヤの残りの部分は省略される。この場合、前の COD = “0” は、どの符号化マクロブロックあるいは符号化されないマクロブロックにも関係なく、それ故にマクロブロック番号も増加しない。Pピクチャでは、COD = “0” と MCBPC = スタッフ符号の複数の組により複数回のスタッフ化が行われる。表 7 / JT-H263 および表 8 / JT-H263 を参照。

表8 / JT-H263 MCBPCの可変長符号表 (Pピクチャ用)  
(ITU-T H. 263)

インデックス	MBタイプ	CBPC (56)	ビット数	符号語
0	0	00	1	1
1	0	01	4	0011
2	0	10	4	0010
3	0	11	6	0001 01
4	1	00	3	011
5	1	01	7	0000 111
6	1	10	7	0000 110
7	1	11	9	0000 0010 1
8	2	00	3	010
9	2	01	7	0000 101
10	2	10	7	0000 100
11	2	11	8	0000 0101
12	3	00	5	0001 1
13	3	01	8	0000 0100
14	3	10	8	0000 0011
15	3	11	7	0000 011
16	4	00	6	0001 00
17	4	01	9	0000 0010 0
18	4	10	9	0000 0001 1
19	4	11	9	0000 0001 0
20	スタッフ符号	--	9	0000 0000 1
21	5	00	11	0000 0000 010
22	5	01	13	0000 0000 0110 0
23	5	10	13	0000 0000 0111 0
24	5	11	13	0000 0000 0111 1

表9 / JT-H263 標準ピクチャのマクロブロックタイプとデータ要素  
(ITU-T H. 263)

ピクチャタイプ	MBタイプ	名称	COD	MCBPC	CBPY	DQUANT	MVD	MVD <sub>2,4</sub>
INTER	符号化しない	-	X					
INTER	0	INTER	X	X	X		X	
INTER	1	INTER+Q	X	X	X	X	X	
INTER	2	INTER4V	X	X	X		X	X
INTER	3	INTRA	X	X	X			
INTER	4	INTRA+Q	X	X	X	X		
INTER	5	INTER4V+Q	X	X	X	X	X	X
INTER	スタッフ符号	-	X	X				
INTRA	3	INTRA		X	X			
INTRA	4	INTRA+Q		X	X	X		
INTRA	スタッフ符号	-		X				

注：“x”はその項目がマクロブロック中に現れることを意味する。

表 10 / JT-H263 PBフレームでのマクロブロックタイプとデータ要素  
(ITU-T H. 263)

ピクチャタイプ	MBタイプ	名称	COD	MCBPC	MODB	CBPY	CBPB	DQUANT	MVD	MVDB	MVD <sub>2,4</sub>
INTER	符号化しない	-	X								
INTER	0	INTER	X	X	X	X	(X)		X	(X)	
INTER	1	INTER+Q	X	X	X	X	(X)	X	X	(X)	
INTER	2	INTER4V	X	X	X	X	(X)		X	(X)	X
INTER	3	INTRA	X	X	X	X	(X)		X	(X)	
INTER	4	INTRA+Q	X	X	X	X	(X)	X	X	(X)	
INTER	5	INTER4V+Q	X	X	X	X	(X)	X	X	(X)	X
INTER	スタッフ符号	-	X	X							

注 1：“x”はその項目がマクロブロック中に現れることを意味する。

注 2：CBPBとMVDBは、MODBが指示された場合のみ現れる。

注 3：PBマクロブロックのMBタイプがINTRAを指示した場合でも、Bブロックは常にINTERモードで符号化される。

### 5.3.3 Bブロックのマクロブロックモード (MODB) (可変長)

MODBは、PTYPEが‘PBフレーム’を示す場合に、MBタイプの0から4に対して現れ、CBPBが現れ（このマクロブロックのB係数が伝送されることを指示する）かつ、またはMVDBが現れるかどうかを指示する可変長符号語である。表 11 / JT-H263にMODBの符号語を定義する。MODBは改良PBフレームに対しては付属資料Mで定義されるように異なって符号化される。

表 11 / JT-H263 MODBの可変長符号表  
(ITU-T H. 263)

インデックス	CBPB	MVDB	ビット数	符号語
0			1	0
1		x	2	10
2	x	x	2	11

注：“x”はその項目がマクロブロック中に現れることを意味する。

### 5.3.4 Bブロックの符号化ブロックパターン (CBPB) (6ビット)

PBフレームモードにおいて、MODBによって指示された場合だけCBPBが現れる。符号化ブロックパターンのCBPB<sub>N</sub>の各ビットに対して、もしN番目のBブロックに係数が現れるならばCBPB<sub>N</sub>は“1”となり、それ以外は“0”である。ブロックの番号付けは図 5 / JT-H263に与えられており、CBPBの左端のビットはブロック番号1に対応している。

### 5.3.5 輝度信号の符号化ブロックパターン (CBPY) (可変長)

拡張INTRA符号化モードが使用されていないならば、マクロブロック中で、少なくとも1つのINTRADC以外の変換係数が伝送されるYブロックを示すパターン番号を与えている可変長符号語 (INTRADCは、INTRAブロックの直流係数である。5.4.1小節参照)。もし拡張INTRA符号化が使用されているならば、INTRADCは他の係数と同じ方法で指示される (付属資料I参照)。

符号化ブロックパターンのCBPY<sub>N</sub>の各ビットに対して、もしINTRADC以外の係数がブロックNに現れるならばCBPY<sub>N</sub>は“1”、それ以外は“0”である。ブロックの番号付けは図 5 / JT-H263に与えられ、CBPYの左端のビットはブロック番号1に対応している。表 12 / JT-H263で定義されるように、あるパターンCBPY<sub>N</sub>に対し、INTERマクロブロックとINTRAマクロブロックで異なった符号語が使われる。

表 1 2 / J T - H 2 6 3 C B P Y の可変長符号表  
( I T U - T H . 2 6 3 )

インデックス	CBPY(INTRA) (12 34)	CBPY(INTER) (12 34)	ビット数	符号
0	00 00	11 11	4	0011
1	00 01	11 10	5	0010 1
2	00 10	11 01	5	0010 0
3	00 11	11 00	4	1001
4	01 00	10 11	5	0001 1
5	01 01	10 10	4	0111
6	01 10	10 01	6	0000 10
7	01 11	10 00	4	1011
8	10 00	01 11	5	0001 0
9	10 01	01 10	6	0000 11
10	10 10	01 01	4	0101
11	10 11	01 00	4	1010
12	11 00	00 11	4	0100
13	11 01	00 10	4	1000
14	11 10	00 01	4	0110
15	11 11	00 00	2	11

### 5. 3. 6 量子化器情報 (DQUANT) (2ビットまたは可変長)

もし修正量子化モードが使用されていないならば、DQUANTはQUANTの変化を定義する2ビットの符号である。表 1 3 / J T - H 2 6 3 に差分値とその符号語を示す。QUANTは1から31の値を取り得る。もし差分値を加えた後のQUANTの値が1より小さい、もしくは31より大きいならば、それぞれ1と31に制限される。もし修正量子化モードが使用されているならば、DQUANTは付属資料Tで定義されるように可変長の符号語である。

表 1 3 / J T - H 2 6 3 D Q U A N T 符号と Q U A N T の差分値  
( I T U - T H . 2 6 3 )

インデックス	差分値	DQUANT
0	-1	00
1	-2	01
2	1	10
3	2	11

### 5. 3. 7 動きベクトルデータ(MVD)(可変長)

MVDは、全てのINTERマクロブロック（同様に、PBフレームモードにおけるINTRAマクロブロック）に含まれ、水平成分の可変長符号語に垂直成分の可変長符号語が続く形で構成される。可変長符号を表 1 4 / J T - H 2 6 3 に示す。もし無制限動きベクトルモードが使用され、かつPLUSPTYPEが現れたならば、動きベクトルは表 1 4 / J T - H 2 6 3 ではなくて付表D. 3 / J T - H 2 6 3 を用いて符号化される（付属資料D参照）。

表14/JT-H263 MVDの可変長符号表  
(ITU-T H. 263)

インデックス	ベクトル差分値		ビット数	符号
0	-16	16	13	0000 0000 0010 1
1	-15.5	16.5	13	0000 0000 0011 1
2	-15	17	12	0000 0000 0101
3	-14.5	17.5	12	0000 0000 0111
4	-14	18	12	0000 0000 1001
5	-13.5	18.5	12	0000 0000 1011
6	-13	19	12	0000 0000 1101
7	-12.5	19.5	12	0000 0000 1111
8	-12	20	11	0000 0001 001
9	-11.5	20.5	11	0000 0001 011
10	-11	21	11	0000 0001 101
11	-10.5	21.5	11	0000 0001 111
12	-10	22	11	0000 0010 001
13	-9.5	22.5	11	0000 0010 011
14	-9	23	11	0000 0010 101
15	-8.5	23.5	11	0000 0010 111
16	-8	24	11	0000 0011 001
17	-7.5	24.5	11	0000 0011 011
18	-7	25	11	0000 0011 101
19	-6.5	25.5	11	0000 0011 111
20	-6	26	11	0000 0100 001
21	-5.5	26.5	11	0000 0100 011
22	-5	27	10	0000 0100 11
23	-4.5	27.5	10	0000 0101 01
24	-4	28	10	0000 0101 11
25	-3.5	28.5	8	0000 0111
26	-3	29	8	0000 1001
27	-2.5	29.5	8	0000 1011
28	-2	30	7	0000 111
29	-1.5	30.5	5	0001 1
30	-1	31	4	0011
31	-0.5	31.5	3	011
32	0		1	1
33	0.5	-31.5	3	010
34	1	-31	4	0010
35	1.5	-30.5	5	0001 0
36	2	-30	7	0000 110
37	2.5	-29.5	8	0000 1010
38	3	-29	8	0000 1000
39	3.5	-28.5	8	0000 0110
40	4	-28	10	0000 0101 10
41	4.5	-27.5	10	0000 0101 00
42	5	-27	10	0000 0100 10
43	5.5	-26.5	11	0000 0100 010
44	6	-26	11	0000 0100 000
45	6.5	-25.5	11	0000 0011 110
46	7	-25	11	0000 0011 100
47	7.5	-24.5	11	0000 0011 010
48	8	-24	11	0000 0011 000
49	8.5	-23.5	11	0000 0010 110
50	9	-23	11	0000 0010 100
51	9.5	-22.5	11	0000 0010 010
52	10	-22	11	0000 0010 000
53	10.5	-21.5	11	0000 0001 110
54	11	-21	11	0000 0001 100
55	11.5	-20.5	11	0000 0001 010
56	12	-20	11	0000 0001 000
57	12.5	-19.5	12	0000 0000 1110
58	13	-19	12	0000 0000 1100
59	13.5	-18.5	12	0000 0000 1010
60	14	-18	12	0000 0000 1000
61	14.5	-17.5	12	0000 0000 0110
62	15	-17	12	0000 0000 0100
63	15.5	-16.5	13	0000 0000 0011 0

### 5. 3. 8 動きベクトルデータ(MVD<sub>2-4</sub>)(可変長)

P T Y P EおよびM C B P Cによって指示された場合、3つの符号語MVD<sub>2-4</sub>が含まれ、各ベクトルの水平成分の可変長符号語に垂直成分の可変長符号語が続く形で各々構成される。可変長符号は表14/J T-H 263で与えられる。MVD<sub>2-4</sub>は、拡張予測モード(付属資料F参照)あるいはデブロッキングフィルタモード(付属資料J参照)の場合のみ含まれる。

### 5. 3. 9 Bマクロブロックの動きベクトルデータ(MVDB)(可変長)

P Bフレームモードあるいは改良P BフレームモードにおいてMODBによって指示された場合のみ、MVDBは現れ、各ベクトルの水平成分の可変長符号語に垂直成分の可変長符号語が続く形で構成される。可変長符号は表14/J T-H 263で与えられる。MVDBの使用に関しては、付属資料Gおよび付属資料Mを参照のこと。

## 5. 4 ブロックレイヤ

P Bフレームモードでない場合、マクロブロックは、4つの輝度ブロックと2種類の色差ブロックを含む(図5/J T-H 263を参照)。ブロックレイヤの構造を、図11/J T-H 263に示す。M C B P CがM Bタイプ3、または4(表7/J T-H 263、および表8/J T-H 263を参照)を指示する場合、I N T R A D Cは、マクロブロックの全てのブロックに含まれる。M C B P C、または、C B P Yによって示された場合、T C O E Fが含まれる。

P Bフレームモードにおいて、マクロブロックは、12のブロックを含む。最初に、T T C標準J T-H 263のデフォルトのモードで6つのPブロックのデータが伝送され、続いて6つのBブロックのデータが伝送される。M C B P CがM Bタイプ3または4(表7/J T-H 263、および表8/J T-H 263を参照)を指示する場合、I N T R A D Cは、マクロブロックの全てのPブロックに現れる。I N T R A D CはBブロックには現れない。M C B P CまたはC B P Yによって指示された場合に、T C O E FはPブロックに現れ、C B P Bによって指示された場合に、T C O E FはBブロックに現れる。

シンタックス算術符号化モードにおけるシンボルの符号化に関しては、付属資料Eを参照のこと。



図11/J T-H 263 ブロックレイヤの構造  
(I T U-T H. 263)

### 5. 4. 1 I N T R AブロックのDC係数(I N T R A D C)(8ビット)

8ビットの符号語である。符号“0000 0000”は用いられない。また“1000 0000”は使用されず、再生レベルの1024は“1111 1111”で符号化される(表15/J T-H 263を参照)。



表 15 / JT-H263 INTRAモードDC係数の再生レベル  
(ITU-T H. 263)

インデックス	固定長符号	逆変換の再生レベル
0	0000 0001 (1)	8
1	0000 0010 (2)	16
2	0000 0011 (3)	24
.	.	.
.	.	.
126	0111 1111 (127)	1016
127	1111 1111 (255)	1024
128	1000 0001 (129)	1032
.	.	.
.	.	.
252	1111 1101 (253)	2024
253	1111 1110 (254)	2032

#### 5. 4. 2 変換係数(TCOEF)(可変長)

頻繁に発生するイベントは、表 16 / JT-H263 に示す可変長符号によって符号化される。最後についた“s”は、レベルの正負の符号を表し、正では“0”、負では“1”である。

イベントは、最後の非ゼロ係数表示 (LAST:このブロックに更に非ゼロ係数が存在する場合“0”、このブロックの最後の非ゼロ係数の場合“1”)、符号化係数に先行する連続するゼロの数 (RUN)、および非ゼロの符号化された係数の値 (LEVEL) の組み合わせである。

その他の (LAST、RUN、LEVEL) の組み合わせは、7ビットESCAPE、1ビットLAST、6ビットRUN、および、8ビットLEVELの22ビットワードによって符号化される。表 16 / JT-H263 に示された組み合わせの符号化に22ビットワードを使用することは禁止しない。LEVELのための8ビットワードに、符号“0000 0000”は禁止され、また符号“1000 0000”は修正量子化モードが使用されていなければ禁止される (付属資料T参照)。RUN、およびLEVELの符号を表 17 / JT-H263 に示す。

表 16 / JT-H263 TCOEF の可変長符号表  
(ITU-T H. 263)

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
0	0	0	1	3	10s
1	0	0	2	5	1111 s
2	0	0	3	7	0101 01s
3	0	0	4	8	0010 111s
4	0	0	5	9	0001 1111 s
5	0	0	6	10	0001 0010 1s
6	0	0	7	10	0001 0010 0s
7	0	0	8	11	0000 1000 01s
8	0	0	9	11	0000 1000 00s
9	0	0	10	12	0000 0000 111s
10	0	0	11	12	0000 0000 110s
11	0	0	12	12	0000 0100 000s
12	0	1	1	4	110s
13	0	1	2	7	0101 00s
14	0	1	3	9	0001 1110 s
15	0	1	4	11	0000 0011 11s
16	0	1	5	12	0000 0100 001s
17	0	1	6	13	0000 0101 0000s
18	0	2	1	5	1110 s
19	0	2	2	9	0001 1101 s
20	0	2	3	11	0000 0011 10s
21	0	2	4	13	0000 0101 0001s
22	0	3	1	6	0110 1s
23	0	3	2	10	0001 0001 1s
24	0	3	3	11	0000 0011 01s
25	0	4	1	6	0110 0s
26	0	4	2	10	0001 0001 0s
27	0	4	3	13	0000 0101 0010s
28	0	5	1	6	0101 1s
29	0	5	2	11	0000 0011 00s
30	0	5	3	13	0000 0101 0011s
31	0	6	1	7	0100 11s
32	0	6	2	11	0000 0010 11s

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
58	1	0	1	5	0111 s
59	1	0	2	10	0000 1100 1s
60	1	0	3	12	0000 0000 101s
61	1	1	1	7	0011 11s
62	1	1	2	12	0000 0000 100s
63	1	2	1	7	0011 10s
64	1	3	1	7	0011 01s
65	1	4	1	7	0011 00s
66	1	5	1	8	0010 011s
67	1	6	1	8	0010 010s
68	1	7	1	8	0010 001s
69	1	8	1	8	0010 000s
70	1	9	1	9	0001 1010 s
71	1	10	1	9	0001 1001 s
72	1	11	1	9	0001 1000 s
73	1	12	1	9	0001 0111 s
74	1	13	1	9	0001 0110 s
75	1	14	1	9	0001 0101 s
76	1	15	1	9	0001 0100 s
77	1	16	1	9	0001 0011 s
78	1	17	1	10	0000 1100 0s
79	1	18	1	10	0000 1011 1s
80	1	19	1	10	0000 1011 0s
81	1	20	1	10	0000 1010 1s
82	1	21	1	10	0000 1010 0s
83	1	22	1	10	0000 1001 1s
84	1	23	1	10	0000 1001 0s
85	1	24	1	10	0000 1000 1s
86	1	25	1	11	0000 0001 11s
87	1	26	1	11	0000 0001 10s
88	1	27	1	11	0000 0001 01s
89	1	28	1	11	0000 0001 00s
90	1	29	1	12	0000 0100 100s

表16 / JT-H263 TCOEFの可変長符号表 (続き)  
(ITU-T H. 263)

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
33	0	6	3	13	0000 0101 0100s
34	0	7	1	7	0100 10s
35	0	7	2	11	0000 0010 10s
36	0	8	1	7	0100 01s
37	0	8	2	11	0000 0010 01s
38	0	9	1	7	0100 00s
39	0	9	2	11	0000 0010 00s
40	0	10	1	8	0010 110s
41	0	10	2	13	0000 0101 0101s
42	0	11	1	8	0010 101s
43	0	12	1	8	0010 100s
44	0	13	1	9	0001 1100 s
45	0	14	1	9	0001 1011 s
46	0	15	1	10	0001 0000 1s
47	0	16	1	10	0001 0000 0s
48	0	17	1	10	0000 1111 1s
49	0	18	1	10	0000 1111 0s
50	0	19	1	10	0000 1110 1s
51	0	20	1	10	0000 1110 0s
52	0	21	1	10	0000 1101 1s
53	0	22	1	10	0000 1101 0s
54	0	23	1	12	0000 0100 010s
55	0	24	1	12	0000 0100 011s
56	0	25	1	13	0000 0101 0110s
57	0	26	1	13	0000 0101 0111s

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
91	1	30	1	12	0000 0100 101s
92	1	31	1	12	0000 0100 110s
93	1	32	1	12	0000 0100 111s
94	1	33	1	13	0000 0101 1000s
95	1	34	1	13	0000 0101 1001s
96	1	35	1	13	0000 0101 1010s
97	1	36	1	13	0000 0101 1011s
98	1	37	1	13	0000 0101 1100s
99	1	38	1	13	0000 0101 1101s
100	1	39	1	13	0000 0101 1110s
101	1	40	1	13	0000 0101 1111s
102	ESCAPE			7	0000 011

表17 / JT-H263 RUNとLEVELの固定長符号表  
(ITU-T H. 263)

インデックス	RUN	符号
0	0	000 000
1	1	000 001
2	2	000 010
.	.	.
.	.	.
63	63	111 111

インデックス	LEVEL	符号
-	-128	本文参照
0	-127	1000 0001
.	.	.
125	-2	1111 1110
126	-1	1111 1111
-	0	禁止
127	1	0000 0001
128	2	0000 0010
.	.	.
253	127	0111 1111

## 6. 復号処理

### 6.1 動き補償

本節では、TTC標準JT-H263のデフォルトの予測モードの動き補償に関して記述する。無制限動きベクトルモードにおける動き補償の記述に関しては付属資料Dを参照のこと。拡張予測モードにおける動き補償の記述

に関しては付属資料Fを参照のこと。解像度削減更新モードにおける動き補償の記述に関しては付属資料Qを参照のこと。

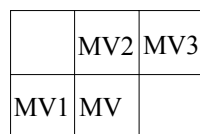
### 6. 1. 1 差分動きベクトル

マクロブロックベクトルは、MVD（表14/JT-H263および付表D.3/JT-H263を参照）で示されたベクトル差分値に予測値を加えることによって得られる。マクロブロック当たり4つのベクトルの差分符号化に関しては、付属資料Fを参照のこと。マクロブロック当たり1つのベクトルの場合、差分符号化のための候補予測値は、図12/JT-H263で示される周囲の3つのマクロブロックから採用される。予測値は、水平成分、垂直成分別々に計算される。

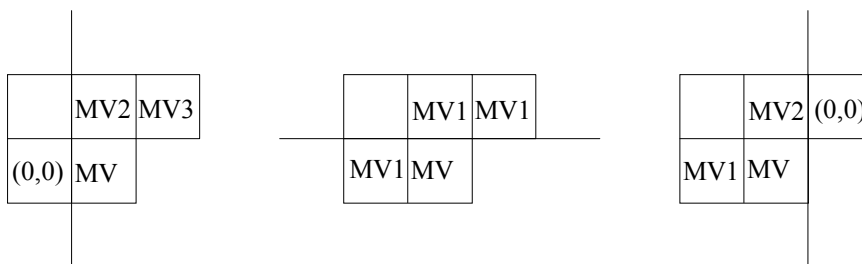
現在のGOB、スライス、またはピクチャの境界の特別な場合に、次の決定規則が順に適用される。

- (1) 予測に用いるマクロブロックが、INTRAモードで符号化された場合（もし双方向予測のPBフレームモードでなければ）、または符号化されなかった場合（COD="1"）、候補予測値はゼロに設定される。
- (2) 予測に用いるマクロブロックがピクチャあるいはスライスの外（左端で）にあるならば、候補予測値MV1は、ゼロに設定される。
- (3) 予測に用いるマクロブロックがピクチャの外（上端で）にある場合、もしくは現在のGOBのGOBヘッダが空でなく、現在のGOBの外（上端で）にあるならば、あるいはスライス構造モードでスライスの外にあるならば、候補予測値MV2およびMV3は、MV1に設定される。
- (4) 予測に用いるマクロブロックがピクチャの外（右端で）にあるならば、候補予測値MV3は、ゼロに設定される。

各ベクトル成分の予測値は、3つの候補予測値の成分の中央値である。



MV : 現時点の動きベクトル  
 MV1 : 左の動きベクトル  
 MV2 : 上の動きベクトル  
 MV3 : 右上の動きベクトル



————— : ピクチャもしくはGOBの境界

図12/JT-H263 動きベクトル予測

(ITU-T H. 263)

動きベクトル成分値の範囲が制限されているという利点が考慮されている。MVDの各可変長符号語は、一对の差分値を表す。一对の片方のみが、許容範囲 [-16, 15.5] 中にあるマクロブロックベクトル成分を生成する。動きベクトルの水平成分、または垂直成分の正の値は、予測が空間的に予測対象の画素の右、または下の画素位置の前ピクチャの画素から形成されることを意味する。もし無制限動きベクトルモード（付属資料D参照）が使用されているならば、動きベクトルの復号はD.2小節で定義されているように実行されなければならない。

動きベクトルは、マクロブロックの4つの輝度ブロックの全ての画素に使用される。両色差ブロックの動きベクトルは、輝度より低い解像度の色差フォーマットのため、マクロブロックベクトルの成分値を2で割ることにより得られる。その結果得られた1/4画素解像度ベクトルの成分値は、表18/JT-H263に示すように、最も近い1/2画素位置側に修正される。

表 1 8 / J T - H 2 6 3 1 / 4 画素解像度色差ベクトル成分の修正  
( I T U - T H . 2 6 3 )

1 / 4 画素位置	0	1/4	1/2	3/4	1
修正位置	0	1/2	1/2	1/2	1

### 6. 1. 2 1 / 2 画素予測の補間

1 / 2 画素位置の画素値は、図 1 3 / J T - H 2 6 3 に示す双線形補間により得られる。“/” は、切捨ての除算を示す。

P T Y P E 中の情報源フォーマットフィールド (第 6 ~ 8 ビット) が“拡張 P T Y P E”を指示する時、R C O N T R O L の値は、M P P T Y P E ( 5 . 1 . 4 . 3 小節参照) 中の丸めタイプ ( R T Y P E ) ビット (第 6 ビット) の値に等しい。それ以外では、R C O N T R O L は暗黙的な値である 0 を持つ。改良 P B フレームの B 部分に対しては、R T Y P E ビットにかかわらず、R C O N T R O L の値は 0 にセットされる。

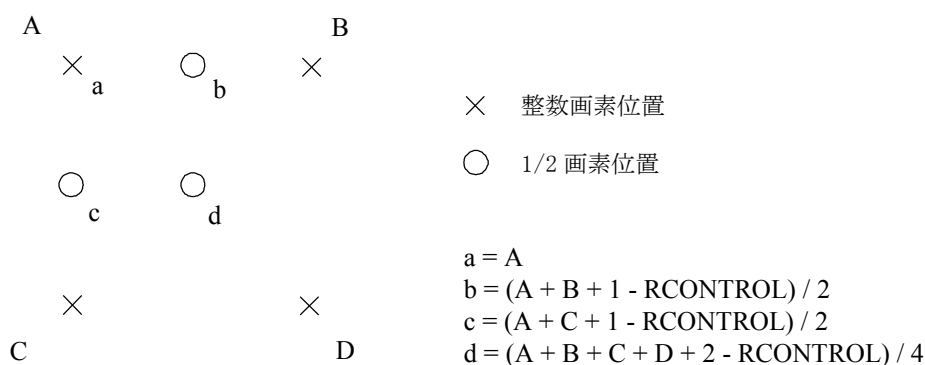


図 1 3 / J T - H 2 6 3 双線形補間による 1 / 2 画素予測  
( I T U - T H . 2 6 3 )

## 6. 2 係数復号

### 6. 2. 1 逆量子化

オプションの拡張 I N T R A 符号化モードが使用されている場合 (付属資料 I 参照) を除いて、逆量子化処理がこの小節で記述される。L E V E L = 0 ならば、再生レベル R E C = 0 である。I N T R A D C の再生レベルは、表 1 5 / J T - H 2 6 3 で与えられる。I N T R A D C 以外の全ての非ゼロ係数の再生レベルは、次式で与えられる。

$$\begin{aligned}
 | R E C | &= Q U A N T \cdot ( 2 \cdot | L E V E L | + 1 ) && Q U A N T \text{ が奇数の場合} \\
 | R E C | &= Q U A N T \cdot ( 2 \cdot | L E V E L | + 1 ) - 1 && Q U A N T \text{ が偶数の場合}
 \end{aligned}$$

この処理が偶数値を許さないことに注目すること。これは、逆離散コサイン変換 mismatch 誤差の累積をさけるためである。| R E C | の計算の後、R E C が得られるように正負の符号が付加される。つまり、

$$R E C = \text{sign} ( L E V E L ) \cdot | R E C |$$

sign ( L E V E L ) は、T C O E F 符号 (表 1 6 / J T - H 2 6 3 を参照) の最後のビット、または、表 1 7 / J T - H 2 6 3 で与えられる。

### 6. 2. 2 再生レベルのクリッピング

逆量子化の後、I N T R A D C 以外の全ての係数の再生レベルは、範囲 [-2048, 2047] に切捨てられる。

### 6. 2. 3 ジグザグ位置

量子化された変換係数は、オプションの拡張 I N T R A 符号化モードが使用されていないならば（付属資料 I 参照）、図 1 4 / J T - H 2 6 3 に示す順番に従って、8 × 8 ブロックに置かれる。係数 1 は、D C 係数である。

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
36	37	49	50	58	59	63	64

図 1 4 / J T - H 2 6 3 量子化された変換係数のジグザグ位置  
( I T U - T H . 2 6 3 )

### 6. 2. 4 逆変換

係数の逆量子化およびジグザグ走査の後、得られた 8 × 8 ブロックは、大きさが 8 × 8 の分離可能な 2 次元逆離散コサイン変換によって処理される。逆変換の出力は 9 ビットで表現できるようにクリッピングされるため、[-256, 255] の範囲の値を取る。逆変換の変換関数を以下に示す。

$$f(x, y) = 1/4 \sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v)F(u, v) \cos[\pi(2x+1)u/16] \cos[\pi(2y+1)v/16]$$

ただし、 $u, v, x, y = 0, 1, 2, \dots, 7$

$x, y$  = 画素領域における空間座標系

$u, v$  = 変換領域における座標系

$C(u) = 1/\sqrt{2}$   $u = 0$  の場合

$C(v) = 1/\sqrt{2}$   $v = 0$  の場合

$C(u), C(v) = 1$  上記以外

注：変換されるブロック内において、 $x = 0$  はブロックの左端、 $y = 0$  はブロックの上端の画素を参照する。

逆変換を計算するための演算処理は定義しない。ただし、付属資料 A で指定された誤差許容範囲を満足しなければならない。

## 6. 3 ブロックの再生

### 6. 3. 1 加算

動き補償、および係数復号（逆変換を含む）の後、各輝度、色差ブロックに対して再生が行われる。I N T R A ブロックでは、再生は逆変換の結果に等しい。I N T E R ブロックでは、再生値は、予測値と逆変換の結果を加算することにより生成される。加算は画素単位で行われる。解像度削減更新モードにおける加算については付属資料 Q を参照のこと。

### 6. 3. 2 クリッピング

符号器および復号器ループでの算術演算オーバーフローを引き起こす変換係数の量子化歪みを防ぐために、クリッピング関数が挿入される。クリッピングにより、予測値と再生された予測誤差値の加算によって得られた画素値が 0 より小さい、または 255 より大きい場合、それぞれ 0 と 255 に修正する。

## 付属資料A

### 逆変換精度の仕様

(この付属資料は本標準の必須部分である)

A. 1 下記の乱数発生 of C 言語プログラムにより、 $-L$  から  $+H$  の範囲の整数画素データを発生させ、 $8 \times 8$  のブロックにまとめる。(  $L = 256$ 、 $H = 255$  )、(  $L = H = 5$  )、(  $L = H = 300$  ) の3通りについて10000ブロック分のデータを発生させる。

A. 2 それぞれの  $8 \times 8$  ブロックについて、64ビット以上の浮動小数点精度を用いて、分離可能な正規直交形のマトリクス演算による離散コサイン変換を行う。

$$F(u, v) = 1/4C(u)C(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos[\pi(2x+1)u/16] \cos[\pi(2y+1)v/16]$$

ただし、 $u, v, x, y = 0, 1, 2, \dots, 7$

$x, y$  = 画素領域における空間座標系

$u, v$  = 変換領域における座標系

$C(u) = 1/\sqrt{2}$   $u = 0$  の場合

$C(v) = 1/\sqrt{2}$   $v = 0$  の場合

$C(u), C(v) = 1$  上記以外の場合

A. 3 それぞれのブロックにおいて求めた64個の変換係数を最も近い整数値に丸め、値を  $[-2048, 2047]$  の範囲に制限する。これを、逆変換への12ビット精度の入力データとする。

A. 4 A. 3節のデータについて、64ビット以上の浮動小数点精度で、分離可能な直交マトリクス演算による逆離散コサイン変換を行う。求められた画素値を最も近い整数に丸めて値を  $[-256, 255]$  の範囲に制限する。これらの  $8 \times 8$  のブロックデータを逆離散コサイン変換出力のデータの基準値とする。

A. 5 A. 3節のデータについて評価対象の逆離散コサイン変換を行い、その出力を  $[-256, 255]$  の範囲に制限する。これらの  $8 \times 8$  のブロックデータを、離散コサイン変換出力の評価値とする。

A. 6 それぞれ64個の逆離散コサイン変換の出力画素と10000ブロック分のデータセットのそれぞれを用いて、基準値と評価値の間の最大、平均、平均二乗誤差を測定する。

A. 7 下記の(1)～(5)を確認する。

- (1) いずれの画素についても、最大誤差が1を超えない。
- (2) いずれの画素についても、平均二乗誤差が0.06を超えない。
- (3) 全画素について求めた平均二乗誤差が0.02を超えない。
- (4) いずれの画素についても、平均誤差が0.015を超えない。
- (5) 全画素について求めた平均誤差が0.0015を超えない。

A. 8 全ての入力値が0の時は、全て0が出力されなければならない。

A. 9 A. 1節の各画素の正負を反転して、再度評価を行う。

乱数発生 of C 言語プログラム

```
/* L and H shall be long, that is 32 bits */
long rand      (L,H)
long           L,H;
{
    static long  randx = 1;                /* long is 32 bits      */
    static double z = (double) 0x7fffffff;
    long        i, j;
    double      x;                        /* double is 64 bits   */
    randx = (randx * 1103515245) + 12345;
    i = randx & 0x7fffffff;               /* keep 30 bits        */
    x = ( (double)i ) / z;                 /* range 0 to 0.99999 ... */
    x *= (L+H+1);                          /* range 0 to < L+H+1 */
    j = x;                                  /* truncate to integer  */
    return(j - L);                          /* range -L to H       */
}
```



## 付属資料B

### 仮想標準復号器

(この付属資料は本標準の必須部分である)

仮想標準復号器 (HRD) は、次のとおりに定義される。

B. 1 HRDと符号器は、画像クロック周波数と同じクロック周波数を持ち、同期して動作する。

B. 2 HRDの受信バッファ容量は、 $B + B P P_{max} K b \times 1 0 2 4$ ビットである。ここで、 $B P P_{max} K b \times 1 0 2 4$ は、ビット列での使用のため交渉された、ピクチャ当たりの最大ビット数である (3. 6節参照)。Bは、次式で定義される。

$$B = 4 \cdot R_{max} / P C F$$

ここで、PCFは有効な画像クロック周波数、 $R_{max}$ は最大ビデオビットレート[bps]である。ピクチャヘッドのCPCFCフィールド内でカスタムPCFが定義されていないならば、有効な画像クロック周波数は標準のCIF画像クロック周波数である。この値はBとして最小である。外部の手段 (例えばTTC標準JT-H245) によって最初に交渉が行われた場合、符号器は、Bとして更に大きい値を使ってもよい。

$R_{max}$ の値は、システム構成 (例えばGSTN (一般交換電話網) またはISDN、単一リンクまたはマルチリンク) に依存し、物理的リンクによって提供された最大ビットレートと等しくてもよい。 $R_{max}$ の交渉は、外部の手段 (例えばTTC標準JT-H245) によって行われる。

B. 3 HRDは、最初は空とする。

B. 4 HRDバッファは、画像クロック間隔 ( $1000/P C F_{ms}$ ) で調べられる。完全な符号化ピクチャが少なくとも1つバッファ内に存在すれば、即座にビット列順で最初のピクチャデータは、はき出される (例えば付図B. 1/JT-H263における  $t_{n+1}$  において)。上記のデータをはき出した直後においては、バッファ占有量はB未満でなければならない。以上が、符号化ピクチャデータ、MCBPC、STUFスタンプ化を含む符号化出力ビット列 (付属資料Hに記述されている、誤り訂正フレーム化ビット、フィルインジケータ ( $F_i$ )、フィルビット、誤り訂正パリティ情報は含まない) の要求条件である。

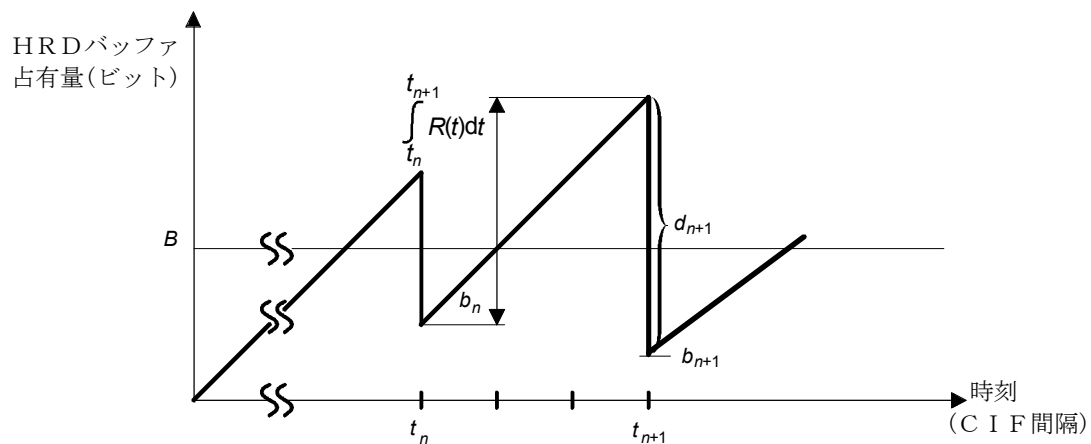
上記の定義において、オプションの時間、SNR、そして空間スケーラビリティモードが使用されていないならば、完全な符号化ピクチャとは、ひとつの標準のIピクチャ、Pピクチャ、またはPBフレーム、あるいは改良PBフレームのことである。

時間、SNR、そして空間スケーラビリティモードが使用されているならば (付属資料O参照)、各々の高品質化レイヤが、完全な符号化ピクチャであるひとつのEI、EPあるいはBピクチャに対して、付加的なHRDに与えられる。基本レイヤバッファはピクチャヘッドのビットが到着した時に、そのピクチャが基本レイヤピクチャかあるいは高品質化レイヤピクチャかを決定するために、また高品質化レイヤの番号を決定するために十分な量のピクチャヘッドが到着するまで、それらを保持しなければならない。到着しているピクチャが高品質化レイヤに属するものと決定し得る時には、そのピクチャに対するすべてのビットは適切な高品質化レイヤHRDに即座に転送されなければならない。またそのビット列が別のHRDバッファに再度転送されるべきであることを決定するために十分な量の新しいピクチャヘッドが到着するまでは、以後到着するすべてのビットは高品質化レイヤHRDに配置され続けられなければならない。高品質化レイヤの識別手順は即時的かつ非同期であり、画像クロック間隔検査回数に独立である。

この要求条件を満たすために ( $n + 1$ ) 番目の符号化ピクチャのビット数  $d_{n+1}$  は、以下の式を満足しなければならない。

$$d_{n+1} \geq b_n + \int_{t_n}^{t_{n+1}} R(t) dt - B$$

ここで、 $b_n$ は、時刻  $t_n$ の直後のバッファ占有量、  
 $t_n$ は、 $n$ 番目の符号化ピクチャがHRDバッファからはき出される時刻、  
 $R(t)$ は、時刻  $t$ におけるビデオビットレートである。



(注) 時間 $(t_{n+1} - t_n)$ はC I Fピクチャ間隔の整数倍 $(1/29.97, 2/29.97, 3/29.97, \dots : \text{秒})$

付図B. 1 / J T - H 2 6 3 HRDバッファ占有量  
 ( I T U - T H . 2 6 3 )

## 付属資料C

### マルチポイントへの考慮

(この付属資料は本標準の必須部分である)

切替型マルチポイント動作を可能とするために、下記の機能が用意されている。

#### C. 1 画面凍結要求

復号器が表示している画面を、画面凍結解除信号が到着するか、または最小6秒のタイムアウトが発生するまで凍結させる。画面凍結要求信号の伝送は外部の手段による（例えばTTC標準JT-H245）。ビデオビット列のピクチャヘッダの中で付加拡張情報を使って、類似のコマンドが送られてもよいことに注意すること（付属資料L、L. 4節参照）。

#### C. 2 画面更新要求

符号器にバッファがオーバーフローしないような符号化パラメータにて、次の画面をINTRAモードで符号化させる。画面更新要求信号は外部の手段による（例えばTTC標準JT-H245）。

#### C. 3 画面凍結解除

符号器は画面更新要求に対応し、復号器に画面凍結モードを解除させ、通常の方法で復号したピクチャを表示させる画面凍結解除信号を送出する。この信号は画面更新要求に対応して、最初の符号化ピクチャのピクチャヘッダに配置されるPTYPE（5. 1. 3小節参照）により通知される。

#### C. 4 コンティニューアプレゼンスマルチポイントとビデオ多重化（CPM）

注1：TTC標準JT-H324では使用されない。

TTC標準JT-H263では、コンティニューアプレゼンスマルチポイントとビデオ多重化モードを交渉により提供出来る。ここでは最大4個の独立したTTC標準JT-H263ビット列をそれぞれ独立の“サブビット列”とみなし、これらを多重化して新たに一つのビデオビット列を形成する。この時PSBI、GSBI、SSBIおよびESBIフィールドを活用する。このモードのための能力情報交換は外部の手段で実施される（例えばTTC標準JT-H242）。独立したサブビット列に対応する参照ピクチャ選択逆方向チャンネルデータがBCPMとBSBIフィールドを使ってサポートされる。

CPMモードでは、それぞれ独立したTTC標準JT-H263サブビット列の中のCPMフィールドは“1”にセットされなければならない。サブビット列は、各TTC標準JT-H263ビット列のピクチャとGOB、またはスライスとEOSBSの各ヘッダ中のサブビット列インジケータ（SBI）で使われているストリーム識別子番号によって識別される。SBIは、そのヘッダとその構成されたビデオビット列中の次のピクチャまたはGOBヘッダ、またはスライスヘッダまでに続くすべての情報が属するべきTTC標準JT-H263ビット列の番号を示す。

各サブビット列は通常のTTC標準JT-H263ビット列と同等に見なされ、それゆえに外部の手段により交換された能力に従わなければならない。異なるTTC標準JT-H263ビット列に関する情報は、事前に定められたいかなる方法にても送信されることは無く、SBIは先行する複数のSBIとは独立にどんな値を持つことができ、異なるTTC標準JT-H263ビット列に対するピクチャレートは異なってもよい。それぞれ個々のビット列中の情報は、他のビット列中の情報とは完全に独立である。例えば、あるサブビット列中のGFID符号語は他のサブビット列中のGFIDやPTYPEの符号語には影響されない。同様に、ピクチャヘッダ中の拡張ピクチャタイプ（PLUSPTYPE）を使った時のモード推論規則と、その他すべての状況でのビデオビット列操作は、各サブビット列に対して独立に、かつ別々に動作しなければならない。

慣例として、若干のリソース要求の衝突が（異なるプライオリティ規則が外部の方法によって確立されないなら）プライオリティの選択を必要としてもよい状態では、（SBIで送られた）最も低いサブビット列識別子番号のサブビット列は最も高いプライオリティを持つと考えられる。

CPMモードでの各サブビット列の終了を表わすために、シンタックスが付図C. 1/JT-H263に示すように規定され、外部の手段によって最初に交渉されるこの付加的なシンタックスを送る能力が規定される（CPM

操作がこの標準の第1版で定義された一方で、サブビット列の終了シンタックスが第2版で追加され、そのためそのシンタックスは第1版におけるCPM操作の一部としてみなされない。サブビット列の終了シンタックス（ESTUF+EOSBS+ESBI）は、EOS符号によってなされるようなストリーム全体の終了というよりむしろ、各サブビット列の終了を表わす。

注2：TTC標準JT-H263のためにここに定義されるようなCPMサブビット列操作を交渉する能力が、この標準の第2版作成の前に、（TTC標準JT-H324のような）端末に対するどんなITU-T勧告/TTC標準の中にも採用されなかった。そのため、将来のTTC標準JT-Hシリーズ端末標準に採用されるCPM操作の外部の交渉は、TTC標準JT-Hシリーズの端末標準で特に指示がない限り、サブビット列の終了シンタックスに対するサポートを含まなければならない。

サブビット列の終了シンタックスには3つの部分がある。ESTUFを使った必須のバイトアライメントに続いて、23ビットのEOSBS符号語が送られる（これはGN=30であるGOBヘッダに相当し、これ以外のシンタックスのなかでは使われない。そして将来の使用のために予約している1ビットの“0”が続く）。そしてEOSBS符号語の後に、どのサブビット列に作用しているかを示す2ビットのESBI符号語が続く。この符号語の組は、関連するサブビット列のデータの送信終了を意味し、同じサブビット列に対して引き続き送られるいかなるデータもEOSBSの前に届いたデータとは完全に独立でなければならないことを意味する。特に、EOSBS符号の後のサブビット列に対する次ピクチャは、INTERピクチャ、または順方向時間予測を使うことができる他のいかなるピクチャタイプであってはならない（Iピクチャ、あるいはEIピクチャは許されるが、Pピクチャ、PBフレーム、改良PBフレーム、Bピクチャ、またはEPピクチャは許されない）。

EOSBSとESBIのシンタックスは次小節で記述される。ESTUFは5.1.26小節で記述される。

ESTUF	EOSBS	ESBI
-------	-------	------

付図C.1/JT-H263 サブビット列の終了インジケータのシンタックスダイアグラム  
(ITU-T H.263)

#### C.4.1 サブビット列終了符号（EOSBS）（23ビット）

EOSBS符号は23ビットの符号語である。その値は“0000 0000 0000 0000 1 11110 0”である。この符号語を挿入するかどうかは符号器による。後に続くESBIフィールドに示される同じサブビット列に対して、少なくとも1つのピクチャヘッダがそれ以前に送られていなければ、EOSBSは送られるべきではない。そしてEOSBSを送るための能力が外部の手段によって交渉されていないならば、EOSBSが送られてはならない。EOSBS符号はバイトアライメントされなければならない。これは、EOSBSスタートコードの先頭ビットが1バイトの先頭ビット（最上位ビット）となるように、EOSBSスタートコードの前にESTUFを挿入することによって達成される（5.1.26小節参照）。

EOSBS符号は、指定されたサブビット列に対するデータの送信が停止したことを示し、そのサブビット列に対する他のピクチャスタートコードの発行によって再開されるまで、サブビット列が終了していることを示す。同じサブビット列識別子番号（ESBI）を持つその後に続くピクチャは、EOSBS符号の前に送られたピクチャと完全に独立でなければならない。いかなる点でもEOSBS符号の前に送られたピクチャに依存してはならない。

（TTC標準JT-H242で送られた画面凍結要求あるいは画面更新要求のような）一般にどのサブビット列に対してこれらの符号を適用させることを意図しているかの指定が無いビデオビット列に関連した制御およびその他の情報は、すべての‘アクティブ’なサブビット列のみに適用されるものと仮定されるべきである。もし少なくとも1つのピクチャスタートコードがサブビット列に対して受信され、最後のデータが送られているならば、サブビット列に適用される最後のデータがEOSあるいはEOSBS+ESBIではなかったなら、アクティブであると考えられる。

#### C.4.2 サブビット列終了インジケータ（ESBI）（2ビット）

ESBIは、EOSBSのすぐ後に続く2ビットの固定長符号語である。それは終了するサブビット列のサブビット列番号を示す。その値はサブビット列番号の2ビットの自然2進数表現である。

## 付属資料D

### 無制限動きベクトルモード

(この付属資料は本標準の必須部分である)

本付属資料は、TTC標準JT-H263のオプション機能である無制限動きベクトルモードを記述する。TTC標準JT-H263のこのモードの能力は外部の手段(例えばTTC標準JT-H245)により通知される。このモードの使用はPTYPE、またはPLUSPTYPE内で指示される。

動きベクトルの範囲と、無制限動きベクトルモードで動きベクトルの差分を符号化するのに使われる可変長符号表は、ピクチャヘッダのPLUSPTYPEフィールドが存在しているかどうかによる。また、PLUSPTYPEが存在する時、動きベクトルの範囲はピクチャサイズとピクチャヘッダ内のUUIフィールドの値に依存する。

#### D. 1 ピクチャの境界を越えた動きベクトル

TTC標準JT-H263のデフォルトの予測モードでは、動きベクトルは予測に用いられる全ての参照画素が符号化されたピクチャ領域にあるように制約される(4.2.3小節参照)。しかし無制限動きベクトルモードでは、この制約は解除され、それゆえ、動きベクトルがピクチャの外側を指し示すことが許容される。動きベクトルにより指し示された画素が符号化されたピクチャの外側にあるときは、境界上の画素が代わりに使用される。この境界上の画素とは、動きベクトルを漸減させた時、符号化されたピクチャ領域の内側に現れる最初の画素である。動きベクトルの漸減は画素単位に実施され、かつ動きベクトルの水平および垂直の各成分毎に独立に実施される。

例えば、もし無制限動きベクトルモードがQCIFピクチャに適用される場合、輝度信号の参照される画素は以下の式で与えられる。

$$R_{umv}(x, y) = R(x', y')$$

ここで、

$x, y, x', y'$  = 画素領域における空間座標

$R_{umv}(x, y)$  = 無制限動きベクトルモードにおける座標  $(x, y)$  の参照画素の値

$R(x', y')$  = 無制限動きベクトルモードにおける座標  $(x', y')$  の参照画素の値

$$\begin{aligned} x' &= 0 && x < 0 \text{ の場合} \\ &= 175 && x > 175 \text{ の場合} \\ &= x && \text{その他の場合} \end{aligned}$$

$$\begin{aligned} y' &= 0 && y < 0 \text{ の場合} \\ &= 143 && y > 143 \text{ の場合} \\ &= y && \text{その他の場合} \end{aligned}$$

また、符号化ピクチャ領域 $R(x', y')$ は $0 \leq x' \leq 175$ 、 $0 \leq y' \leq 143$ である。境界の内側では、 $(x', y')$ は $1/2$ 画素単位の位置も取り得るが、境界上では画素は整数の位置のみ取り得る。

#### D. 1. 1 動きベクトル値の制限

もしピクチャヘッダ中にPLUSPTYPEが存在するなら、動きベクトル値は、選択された $16 \times 16$ (あるいは $8 \times 8$ )の領域のいかなる要素も符号化されるピクチャ領域の外側の水平方向、または垂直方向に $15$ 画素以上の距離を持たないように制限されなければならない。これはPLUSPTYPEが存在しない時より小さい外挿範囲であることを注意すること。

注: PLUSPTYPEが存在しない時、外挿範囲は、無制限動きベクトルモードが使われる時には符号化ピクチャ領域の外側に最大 $31.5$ 画素であり、無制限動きベクトルモード無しで拡張予測モード(付属資料F参照)が使われる時は符号化ピクチャ領域の外側 $16$ 画素である。

注: 拡張予測モード(付属資料F参照)が使われる時には、オーバーラップブロック動き補償のため、各 $16 \times 16$ (または $8 \times 8$ )領域の動きベクトルはより大きな領域に影響する。このことにより、拡張予測モード

の“リモート”動きベクトルの有効な外挿補間範囲を増加させることができる。なせなら、（たとえ各動きベクトル値の許容範囲が、拡張予測モードが使われないときと同様に同じままであっても）オーバーラップする量（4画素、または縮小解像度更新モードも使われる時には8画素）が必要とされる外挿補間の量に加わるからである。

#### D. 2 動きベクトルの範囲の拡大

デフォルトの予測モードでは、水平および垂直両成分の動きベクトル値は  $[-16, 15.5]$  の範囲に制限される（このことはBピクチャにおける前方予測、後方予測動きベクトル成分にも適用される）。しかし無制限動きベクトルモードでは、ベクトル成分の最大範囲は拡大される。

もしPLUSPTYPEフィールドがピクチャヘッダ内に現れなければ、動きベクトルの範囲は、予測値が  $[-15.5, 16]$  の範囲にある場合は、各動きベクトル成分に対する予測値に対して  $[-16, 15.5]$  の範囲の値を取り得るという制限付きで、  $[-31.5, 31.5]$  に拡大される。また予測値が  $[-15.5, 16]$  の間に無い場合には、動きベクトルは予測値と同じ符号の  $[-31.5, 31.5]$  間のすべての値、または0を取り得る。ここで、動きベクトル成分を  $MV_c$  とし、その予測値を  $P_c$  とすると、以下が成立する。

$$\begin{array}{ll}
 -31.5 \leq MV_c \leq 0 & -31.5 \leq P_c \leq -16 \text{ の場合} \\
 -16 + P_c \leq MV_c \leq 15.5 + P_c & -15.5 \leq P_c \leq 16 \text{ の場合} \\
 0 \leq MV_c \leq 31.5 & 16.5 \leq P_c \leq 31.5 \text{ の場合}
 \end{array}$$

無制限動きベクトルモードでは、MVD、MVD<sub>2.4</sub>、MVD<sub>B</sub>に対する表14/JT-H263の解釈は以下の通りである。

- (1) 動きベクトルの各成分の予測値が  $[-15.5, 16]$  の範囲にある場合には、ベクトル差分値の中の最初の列のみを適用する。
- (2) 動きベクトルの各成分の予測値が  $[-15.5, 16]$  の範囲に含まれない場合には、表14/JT-H263のベクトル差分値の中で動きベクトル成分の値が  $[-31.5, 31.5]$  に含まれ、かつ、正負の符号が予測値と一致する（動きベクトル=0を含む）ベクトル差分値のみを用いねばならない。

MVDとMVD<sub>2.4</sub>に対する予測値は6.1.1小節、及びF.2節で規定されるベクトルMV<sub>1</sub>、MV<sub>2</sub>、MV<sub>3</sub>の各成分の中央値で定義される。MVD<sub>B</sub>では、予測値P<sub>c</sub>は  $P_c = (TR_B \times MV) / TR_D$  で定義される。但し、MVはPピクチャ内の輝度信号の8×8の大きさのブロックに対する動きベクトルの各成分を示す（G.4節も参照）。

もしPLUSPTYPEが存在するならば、動きベクトルの範囲は動きベクトル予測値に依存しない。もしUIフィールドが“1”にセットされるなら、動きベクトルの範囲は画像フォーマットに依存する。標準化されたCIFまでの画像フォーマットに対する範囲は  $[-32, 31.5]$  であり、4CIFまでに対するその範囲は  $[-64, 63.5]$ 、そして16CIFまでの範囲は  $[-128, 127.5]$ 、そしてさらに大きいカスタム画像フォーマットに対する範囲は  $[-256, 255.5]$  である。水平および垂直方向の動きベクトル範囲は、カスタム画像フォーマットに対して異なってもよい。水平および垂直範囲は付表D.1/JT-H263と付表D.2/JT-H263で規定される。

付表D.1/JT-H263 PLUSPTYPEが存在し、UI = “1”の時の  
(ITU-T H.263) 水平方向動きベクトル範囲

画像の幅	水平方向動きベクトル範囲
4, ..., 352	$[-32, 31.5]$
356, ..., 704	$[-64, 63.5]$
708, ..., 1408	$[-128, 127.5]$
1412, ..., 2048	$[-256, 255.5]$

付表D. 2 / JT-H263 PLUSPTYPEが存在し、UUI = “1”の時の  
(ITU-T H. 263) 垂直方向動きベクトル範囲

画像の高さ	垂直方向動きベクトル範囲
4, ..., 288	[-32, 31.5]
292, ..., 576	[-64, 63.5]
580, ..., 1152	[-128, 127.5]

縮小解像度更新モードで、指定された範囲は疑似動きベクトルに適用される。これは結果として得られる実際の動きベクトル範囲が約2倍に拡大されていることを意味する（付属資料Qも参照）。

もしUUIが“01”に設定されているなら、動きベクトルは、D. 1. 1小節で説明したように符号化領域境界までの距離を除いて制限されない。同様の制限が（疑似動きベクトルだけではなく）縮小解像度更新モードでの実際の動きベクトル範囲に適用される。

PLUSPTYPEが存在している場合の動きベクトルを符号化するため、付表D. 3 / JT-H263が動きベクトルと動きベクトル予測値の間の差分を符号化するために使われる。付表D. 3 / JT-H263内の全項目が（表14 / JT-H263と対照的に）単一の値を持っている。動きベクトル範囲と、動きベクトルデータを符号化するための付表D. 3 / JT-H263の使用は、PLUSPTYPEが存在している時、すべてのピクチャタイプに適用される。

動きベクトル差分は、常に一組の水平および垂直方向成分の組として符号化される。もしある組が(0.5, 0.5)に等しいなら、6つの連続した“0”が発生する。スタートコードエミュレーションを防ぐために、この発生（6つの連続した“0”）の後には“1”に設定された1ビットが続かなければならない。これは追加のゼロ動きベクトル成分の一つを送信していることに対応する。

付表D. 3 / JT-H263 PLUSPTYPEが存在する時の動きベクトル表  
(ITU-T H. 263)

1 / 2画素単位のベクトル差分の絶対値	ビット数	符号
0	1	1
1	3	0s0
“ $x_0$ ”+2 (2:3)	5	0 $x_0$ 1s0
“ $x_1x_0$ ”+4 (4:7)	7	0 $x_1$ 1 $x_0$ 1s0
“ $x_2x_1x_0$ ”+8 (8:15)	9	0 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_3x_2x_1x_0$ ”+16 (16:31)	11	0 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_4x_3x_2x_1x_0$ ”+32 (32:63)	13	0 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_5x_4x_3x_2x_1x_0$ ”+64 (64:127)	15	0 $x_5$ 1 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_6x_5x_4x_3x_2x_1x_0$ ”+128 (128:255)	17	0 $x_6$ 1 $x_5$ 1 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_7x_6x_5x_4x_3x_2x_1x_0$ ”+256 (256:511)	19	0 $x_7$ 1 $x_6$ 1 $x_5$ 1 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_8x_7x_6x_5x_4x_3x_2x_1x_0$ ”+512 (512:1023)	21	0 $x_8$ 1 $x_7$ 1 $x_6$ 1 $x_5$ 1 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_9x_8x_7x_6x_5x_4x_3x_2x_1x_0$ ”+1024 (1024:2047)	23	0 $x_9$ 1 $x_8$ 1 $x_7$ 1 $x_6$ 1 $x_5$ 1 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0
“ $x_{10}x_9x_8x_7x_6x_5x_4x_3x_2x_1x_0$ ”+2048 (2048:4095)	25	0 $x_{10}$ 1 $x_9$ 1 $x_8$ 1 $x_7$ 1 $x_6$ 1 $x_5$ 1 $x_4$ 1 $x_3$ 1 $x_2$ 1 $x_1$ 1 $x_0$ 1s0

付表D. 3 / JT-H263は規則的に構成されたリバーシブルな表である。各列は1 / 2画素単位の動きベクトル差分の間隔を表す。ビット“ $\dots x_1x_0$ ”は、動きベクトル差分の絶対値の2進数表現での、先頭の“1”に続いているビットを示す。ビット“s”は動きベクトル差分の符号を示し、正に対しては“0”、負に対しては“1”をとる。動きベクトル差分の2進数表現は、符号が継続するか、あるいは終了するかを示すビットでインターリーブされる。例えば、動きベクトル差分-13は、符号s=1と2進数表現1 $x_2x_1x_0$ =“1101”を持つ。これは0 $x_2$ 1 $x_1$ 1 $x_0$ 1s0=“0 11 01 11 10”として符号化される。最後の2ビットの組の2番目の位置である“0”は符号の終了を示す。

## 付属資料E

### シンタックス算術符号化モード

(この付属資料は本標準の必須部分である)

#### E. 1 序論

本標準の5節に記載されているように可変長符号化／復号(VLC/VLD)では、シンボルは符号器のシンタックスに基づく特定の符号表を用いて可変長符号化される。この表は通常可変長符号語の符号語長と符号語を格納している。あるシンボルは表検索により表の内容と比較され、そのシンボルに対応する2進符号語が表から読み出され、通常は受信側へ伝送するための送信側のバッファに入力される。可変長復号においても受信したビット列は符号器のシンタックスに基づいた特定の表により検索される。この表は、現シンボルを送信側で符号化したものと全く同一でなければならない。表内で検索された入力信号は元のシンボルに復元される。復元されたシンボルが可変長復号の最終出力であり、これを用いてピクチャが復元される。この可変長符号化／復号の過程は、各シンボルが固定、かつ、整数のビット長にて符号化されねばならないことを意味している。各シンボルに対する固定、かつ整数のビット長というこの制約を取り除くことにより、ビット数を減少できる。制約の除去は算術符号化により実現できる。

本付属資料では、TTC標準JT-H263のオプションである、シンタックス算術符号化(SAC)モードを説明する。このモードでは、TTC標準JT-H263の中の総ての可変長符号化／復号が対応するSACの可変長符号化／復号に置換される。TTC標準JT-H263のこのモードの能力は外部の手段で通知される(例えばTTC標準JT-H245)。このモードの使用はP T Y P Eの中で示される。

#### E. 2 SAC符号器の仕様

SACモードでは、シンボルは、符号器のシンタックスに基づくある特定の整数列(またはモデル)を用い、かつ、C言語で記述される以下の手順により符号化される。

```
#define q1 16384
#define q2 32768
#define q3 49152
#define top 65535
static long low, high, opposite_bits, length;

void encode_a_symbol(int index, int cumul_freq[ ])
{
    length = high - low + 1;
    high = low - 1 + (length * cumul_freq[index]) / cumul_freq[0];
    low += (length * cumul_freq[index+1]) / cumul_freq[0];
    for ( ; ; ) {
        if (high < q2) {
            send out a bit "0" to PSC_FIFO;
            while (opposite_bits > 0) {
                send out a bit "1" to PSC_FIFO;
                opposite_bits--;
            }
        }
        else if (low >= q2) {
            send out a bit "1" to PSC_FIFO;
            while (opposite_bits > 0) {
                send out a bit "0" to PSC_FIFO;
                opposite_bits--;
            }
        }
        low -= q2;
    }
}
```



```

        high -= q2;
    }
    else if (low >= q1 && high < q3) {
        opposite_bits += 1;
        low -= q1;
        high -= q1;
    }
    else break;

    low *= 2;
    high = 2*high+1;
}
}

```

上記のlow、high及びopposite\_bitsは各々0、top及び0で各々初期化される。P S C \_ F I F Oは算術符号器からの出力ビットをバッファリングするためのF I F Oである。このモデルはcumul\_freq[ ]により規定される。シンボルはモデル内のインデックスを用いて規定される。

### E. 3 S A C復号器の仕様

S A C復号器では、シンボルはシンタックスに基づくある特定のモデルを用い、かつC言語で記述される以下の手順を使用することにより復号される。

```

static long    low, high, code_value, bit, length, index, cum;

int    decode_a_symbol(int cumul_freq[ ])
{
    length = high - low + 1;
    cum = (-1 + (code_value - low + 1) * cumul_freq[0]) / length;
    for (index = 1; cumul_freq[index] > cum; index++);
    high = low - 1 + (length * cumul_freq[index-1]) / cumul_freq[0];
    low += (length * cumul_freq[index]) / cumul_freq[0];
    for ( ; ; ) {
        if (high < q2) ;
        else if (low >= q2) {
            code_value -= q2;
            low -= q2;
            high -= q2;
        }
        else if (low >= q1 && high < q3) {
            code_value -= q1;
            low -= q1;
            high -= q1;
        }
        else break;

        low *= 2;
        high = 2*high + 1;
        get bit from PSC_FIFO;
        code_value = 2*code_value + bit;
    }
}

```

```

return (index-1);
}

```

ここでもモデルはcumul\_freq[ ]により規定される。モデル内のインデックスにより復号されたシンボルが再生される。PSC\_FIFOは入力されるビット列をバッファリングするためのFIFOである。算術符号化されたビット列の復号を開始するために、復号器は以下の手順を使用することにより初期化される。

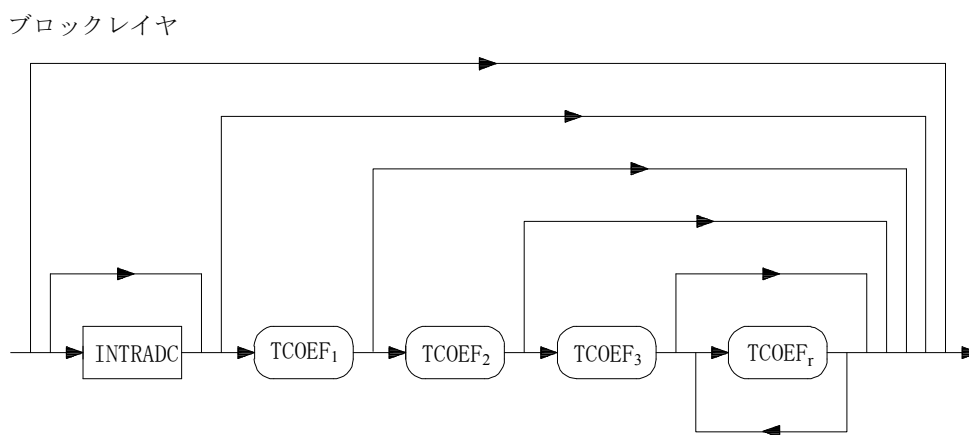
```

void decoder_reset( )
{
    code_value = 0;
    low = 0;
    high = top;
    for (int i = 1; i <= 16; i++) {
        get bit from PSC_FIFO;
        code_value = 2 * code_value + bit;
    }
}

```

#### E. 4 シンタックス

TTC標準JT-H263の変長符号表を用いるモードと同様に、シンボルのシンタックスは4レイヤに分類される。ピクチャ、GOB、マクロブロック及びブロックである。上位から3レイヤのシンタックスは全く同様である。ブロックレイヤのシンタックスもまた極めて類似しており、付図E. 1/JT-H263に示す通りである。



付図E. 1/JT-H263 SACにおけるブロックレイヤの構造  
(ITU-T H. 263)

付図E. 1/JT-H263において、TCOEF1, TCOEF2, TCOEF3及びTCOEFrは5.4.2小節で定義されるLAST-RUN-LEVELを統合化したシンボルであり、各ブロックレイヤの中の第1、第2、第3及び残りのシンボルである。TCOEF1、TCOEF2、TCOEF3及びTCOEFrは各々1個、2個、3個またはそれ以上の係数が各ブロックレイヤに現れた時のみ発生する。

#### E. 5 PSC\_FIFO

符号器または復号器にあるPSC\_FIFOは、18ビット以上の容量を持つFIFOである。符号器のPSC\_FIFOでは、PSCやGBSCの非正規なエミュレーションが検出される。そして、PSCやGBSCではない部分で“0”が14個連続した後は“1”を挿入する事により非正規のPSCやGBSCのエミュレーションは

回避される。復号器のPSC\_FIFOでは“0”が14個連続した後に現れる最初の“1”は削除する。もし“0”の14回連続の後に“0”が現れたなら、それは正規なPSCやGBSCが検出されたと見なす。PSCやGBSCの正確な位置はこれらの“0”の連続に続く次の“1”により決まる。

### E. 6 ヘッダレイヤのシンボル

シンタックスのヘッダレイヤは、ブロックレイヤとマクロブロックレイヤの上にあるそれらの構文上の要素であると考えられる(図6/JT-H263と本文中のシンタックスの記述を参照)。基本となる第1版のシンタックスのヘッダレベルは次に示す3通りの可能なストリングで構成することができる。

- (1) (PSTUF) - PSC - TR - PTYPE - PQUANT - CPM - (PSBI) - (TRB-DBQU  
NAT) - PEI - (PSUPP-PEI-...)
- (2) (GSTUF) - GBSC - GN - (GSBI) - GFID - GQUANT
- (3) (ESTUF) - (EOS) - (PSTUF)

第2版の改定されたシンタックスでは、シンタックスのヘッダレベルは他の構造を持つことができる(図6/JT-H263と本文中のシンタックスの記述を参照)。ヘッダレベルシンタックスのストリングは、符号器側ではTTC標準JT-H263の通常の変長符号表モードと同様にPSC\_FIFOに直接書き込まれる。復号器側では、正規のPSC、GBSC、SSC、EOS、またはEOSBSが検出された後にPSC\_FIFOから直接読み出される。

ビデオセッションの先頭にヘッダが無い場合には、算術符号器はヘッダを送出する前に以下の手順を実行する事により初期化される必要がある。もし(ESTUF) - EOS (または最終ヘッダが送られたサブビット列に対する(ESTUF) - EOSBS)が送られなければ、この手順はビデオセッションの終了時にも同様に実行されなければならない。

```
void encoder_flush( )
{
    opposite_bits++;
    if (low < q1) {
        send out a bit "0" to PSC_FIFO;
        while (opposite_bits > 0) {
            send out a bit "1" to PSC_FIFO;
            opposite_bits--;
        }
    }
    else {
        send out a bit "1" to PSC_FIFO;
        while (opposite_bits > 0) {
            send out a bit "0" to PSC_FIFO;
            opposite_bits--;
        }
    }
    low = 0;
    high = top;
}
```

復号器側では、各々の固定長シンボル列の後で、‘decoder\_reset’手順が実行される。

### E. 7 マクロブロックおよびブロックレイヤのシンボル

マクロブロックおよびブロックレイヤシンボルのモデルは、E. 8節に含まれる。5節の変長符号表に記述された様なインデックスは、このモデルでの整数値のインデックスとして用いられる。

PピクチャにおけるCODのモデルはcumf\_CODと呼ばれる。CODが“0”である場合、インデックスは0、CODが“1”の時はインデックスには1が割り当てられる。PピクチャのMCBPCに対するモデルは、ピクチャヘッダにPLUSPTYPEが存在し、拡張予測モード(付属資料F)、またはデブロッキングフィルタモード(付属資料J)が使われている場合はcumf\_MCBPC\_4MVQと呼ばれ、そうでない場合にはcumf\_MCBPC\_no4MVQと呼ばれる。MCBPCに対するインデックスは、Iピクチャに対しては表7/JT-H263にて定義され、Pピクチャに対しては表8/JT-H263に定義される。IピクチャにおけるMCBPCのモデルはcumf\_MCBPC\_intraと呼ばれる。

MODBのモデルは、付属資料Gが使われていればcumf\_MODB\_G、付属資料Mが使われていればcumf\_MODB\_Mである。MODBのインデックスは表11/JT-H263、または付表M.1/JT-H263にそれぞれ定義される。CBPB<sub>n</sub>(但しn=1, 2, ..., 4)のモデルはcumf\_YCBPBであり、CBPB<sub>n</sub>(但しn=5, 6)のモデルはcumf\_UVCBPBである。ここで、CBPB<sub>n</sub>が“0”の場合のインデックスは0、CBPB<sub>n</sub>が“1”の場合のインデックスは1である。

CBPYのモデルは、INTERマクロブロックではcumf\_CBPY、INTRAマクロブロックではcumf\_CBPY\_intraである。DQUANTに対するモデルはcumf\_DQUANTである。CBPYやDQUANTに対するインデックスは表12/JT-H263と表13/JT-H263に各々定義される。

MVD、MVD<sub>2.4</sub>及びMVD<sub>B</sub>のモデルはcumf\_MVDであり、INTRADCに対するモデルはcumf\_INTRADCである。これらに対するインデックスは表14/JT-H263及び表15/JT-H263に各々定義される。

非エスケープTCOEF係数は、TCOEF1/2/3/rに対するシンボルと、それに引き続くTCOEFの正負の符号を表すシンボルSIGNから構成される。INTERブロックにおけるTCOEF1、TCOEF2、TCOEF3、TCOEF<sub>r</sub>に対するモデルはcumf\_TCOEF1、cumf\_TCOEF2、cumf\_TCOEF3、cumf\_TCOEF<sub>r</sub>である。INTRAブロックにおけるモデルはcumf\_TCOEF1\_intra、cumf\_TCOEF2\_intra、cumf\_TCOEF3\_intra、cumf\_TCOEF<sub>r</sub>\_intraである。総ての係数に対するインデックスは表16/JT-H263に定義される。正負の符号に対するモデルはcumf\_SIGNである。SIGNのインデックスは正の符号では0であり、負の符号では1である。

エスケープ符号後のLAST、RUN、LEVELに対するモデルは、INTER(INTRA)ブロックに対してそれぞれ、cumf\_LAST(cumf\_LAST\_intra)、cumf\_RUN(cumf\_RUN\_intra)、cumf\_LEVEL(cumf\_LEVEL\_intra)である。LASTに対するインデックスはLAST=0に対して0、LAST=1に対して1であり、RUNとLEVELに対するインデックスは表17/JT-H263に定義される。

INTRA\_\_MODEに対するモデルはcumf\_INTRA\_AC\_DCである。そのインデックスは付表I.1/JT-H263に定義される。

## E. 8 SACモデル

```
int cumf_COD[3]={16383, 6849, 0};
```

```
int cumf_MCBPC_no4MVQ[22]={16383, 4105, 3088, 2367, 1988, 1621, 1612, 1609, 1608, 496, 353, 195, 77, 22, 17, 12, 5, 4, 3, 2, 1, 0};
```

```
int cumf_MCBPC_4MVQ[26]={16383, 6880, 6092, 5178, 4916, 3965, 3880, 3795, 3768, 1491, 1190, 889, 655, 442, 416, 390, 360, 337, 334, 331, 327, 326, 88, 57, 26, 0};
```

```
int cumf_MCBPC_intra[10]={16383, 7410, 6549, 5188, 442, 182, 181, 141, 1, 0};
```

```
int cumf_MODB_G[4]={16383, 6062, 2130, 0};
```

```
int cumf_MODB_M[7] = {16383, 6717, 4568, 2784, 1370, 655, 0};
```

```
int cumf_YCBPB[3]={16383, 6062, 0};
```

```
int cumf_UVCBPB[3]={16383, 491, 0};
```

```
int cumf_CBPY[17]={16383, 14481, 13869, 13196, 12568, 11931, 11185, 10814, 9796, 9150, 8781, 7933, 6860, 6116, 4873, 3538, 0};
```

```
int cumf_CBPY_intra[17]={16383, 13619, 13211, 12933, 12562, 12395, 11913, 11783, 11004, 10782, 10689, 9928, 9353, 8945, 8407, 7795, 0};
```

```
int cumf_DQUANT[5]={16383, 12287, 8192, 4095, 0};
```

```
int cumf_MVD[65]={16383, 16380, 16369, 16365, 16361, 16357, 16350, 16343, 16339, 16333, 16326, 16318, 16311, 16306, 16298, 16291, 16283, 16272, 16261, 16249, 16235, 16222, 16207, 16175, 16141, 16094, 16044, 15936, 15764, 15463, 14956, 13924, 11491, 4621, 2264, 1315, 854, 583, 420, 326, 273, 229, 196, 166, 148, 137, 123, 114, 101, 91, 82, 76, 66, 59, 53, 46, 36, 30, 26, 24, 18, 14, 10, 5, 0};
```

```
int cumf_INTRADC[255]={16383, 16380, 16379, 16378, 16377, 16376, 16370, 16361, 16360, 16359, 16358, 16357, 16356, 16355, 16343, 16238, 16237, 16236, 16230, 16221, 16220, 16205, 16190, 16169, 16151, 16130, 16109, 16094, 16070, 16037, 16007, 15962, 15938, 15899, 15854, 15815, 15788, 15743, 15689, 15656, 15617, 15560, 15473, 15404, 15296, 15178, 15106, 14992, 14868, 14738, 14593, 14438, 14283, 14169, 14064, 14004, 13914, 13824, 13752, 13671, 13590, 13515, 13458, 13380, 13305, 13230, 13143, 13025, 12935, 12878, 12794, 12743, 12656, 12596, 12521, 12443, 12359, 12278, 12200, 12131, 12047, 12002, 11948, 11891, 11828, 11744, 11663, 11588, 11495, 11402, 11288, 11204, 11126, 11039, 10961, 10883, 10787, 10679, 10583, 10481, 10360, 10227, 10113, 9961, 9828, 9717, 9584, 9485, 9324, 9112, 9019, 8908, 8766, 8584, 8426, 8211, 7920, 7663, 7406, 7152, 6904, 6677, 6453, 6265, 6101, 5904, 5716, 5489, 5307, 5056, 4850, 4569, 4284, 3966, 3712, 3518, 3342, 3206, 3048, 2909, 2773, 2668, 2596, 2512, 2370, 2295, 2232, 2166, 2103, 2022, 1956, 1887, 1830, 1803, 1770, 1728, 1674, 1635, 1599, 1557, 1500, 1482, 1434, 1389, 1356, 1317, 1284, 1245, 1200, 1179, 1140, 1110, 1092, 1062, 1044, 1035, 1014, 1008, 993, 981, 954, 936, 912, 894, 876, 864, 849, 828, 816, 801, 792, 777, 756, 732, 690, 660, 642, 615, 597, 576, 555, 522, 489, 459, 435, 411, 405, 396, 387, 375, 360, 354, 345, 344, 329, 314, 293, 278, 251, 236, 230, 224, 215, 214, 208, 199, 193, 184, 178, 169, 154, 127, 100, 94, 73, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 20, 19, 18, 17, 16, 15, 9, 0};
```

```
int cumf_TCOEF1[104]={16383, 13455, 12458, 12079, 11885, 11800, 11738, 11700, 11681, 11661, 11651, 11645, 11641, 10572, 10403, 10361, 10346, 10339, 10335, 9554, 9445, 9427, 9419, 9006, 8968, 8964, 8643, 8627, 8624, 8369, 8354, 8352, 8200, 8192, 8191, 8039, 8036, 7920, 7917, 7800, 7793, 7730, 7727, 7674, 7613, 7564, 7513, 7484, 7466, 7439, 7411, 7389, 7373, 7369, 7359, 7348, 7321, 7302, 7294, 5013, 4819, 4789, 4096, 4073, 3373, 3064, 2674, 2357, 2177, 1975, 1798, 1618, 1517, 1421, 1303, 1194, 1087, 1027, 960, 890, 819, 758, 707, 680, 656, 613, 566, 534, 505, 475, 465, 449, 430, 395, 358, 335, 324, 303, 295, 286, 272, 233, 215, 0};
```

```
int cumf_TCOEF2[104]={16383, 13582, 12709, 12402, 12262, 12188, 12150, 12131, 12125, 12117, 12113, 12108, 12104, 10567, 10180, 10070, 10019, 9998, 9987, 9158, 9037, 9010, 9005, 8404, 8323, 8312, 7813, 7743, 7726, 7394, 7366, 7364, 7076, 7062, 7060, 6810, 6797, 6614, 6602, 6459, 6454, 6304, 6303, 6200, 6121, 6059, 6012, 5973, 5928, 5893, 5871, 5847, 5823, 5809, 5796, 5781, 5771, 5763, 5752, 4754, 4654, 4631, 3934, 3873, 3477, 3095, 2758, 2502, 2257, 2054, 1869, 1715, 1599, 1431, 1305, 1174, 1059, 983, 901, 839, 777, 733, 683, 658, 606, 565, 526, 488, 456, 434, 408, 380, 361, 327, 310, 296, 267, 259, 249, 239, 230, 221, 214, 0};
```

```
int cumf_TCOEF3[104]={16383, 13532, 12677, 12342, 12195, 12112, 12059, 12034, 12020, 12008, 12003, 12002, 12001, 10586, 10297, 10224, 10202, 10195, 10191, 9223, 9046, 8999, 8987, 8275, 8148, 8113, 7552, 7483, 7468, 7066, 7003, 6989, 6671, 6642, 6631, 6359, 6327, 6114, 6103, 5929, 5918, 5792, 5785, 5672, 5580, 5507, 5461, 5414, 5382, 5354, 5330, 5312, 5288, 5273, 5261, 5247, 5235, 5227, 5219, 4357, 4277, 4272, 3847, 3819, 3455, 3119, 2829, 2550, 2313, 2104, 1881, 1711, 1565, 1366, 1219, 1068, 932, 866, 799, 750, 701, 662, 605, 559, 513, 471, 432, 403, 365, 336, 312, 290, 276, 266, 254, 240, 228, 223, 216, 206, 199, 192, 189, 0};
```

```
int cumf_TCOEFr[104]={16383, 13216, 12233, 11931, 11822, 11776, 11758, 11748, 11743, 11742, 11741, 11740,
11739, 10203, 9822, 9725, 9691, 9677, 9674, 8759, 8609, 8576, 8566, 7901, 7787, 7770, 7257, 7185, 7168, 6716,
6653, 6639, 6276, 6229, 6220, 5888, 5845, 5600, 5567, 5348, 5327, 5160, 5142, 5004, 4900, 4798, 4743, 4708,
4685, 4658, 4641, 4622, 4610, 4598, 4589, 4582, 4578, 4570, 4566, 3824, 3757, 3748, 3360, 3338, 3068, 2835,
2592, 2359, 2179, 1984, 1804, 1614, 1445, 1234, 1068, 870, 739, 668, 616, 566, 532, 489, 453, 426, 385, 357, 335,
316, 297, 283, 274, 266, 259, 251, 241, 233, 226, 222, 217, 214, 211, 209, 208, 0};
```

```
int cumf_TCOEF1_intra[104]={16383, 13383, 11498, 10201, 9207, 8528, 8099, 7768, 7546, 7368, 7167, 6994,
6869, 6005, 5474, 5220, 5084, 4964, 4862, 4672, 4591, 4570, 4543, 4397, 4337, 4326, 4272, 4240, 4239, 4212,
4196, 4185, 4158, 4157, 4156, 4140, 4139, 4138, 4137, 4136, 4125, 4124, 4123, 4112, 4111, 4110, 4109, 4108,
4107, 4106, 4105, 4104, 4103, 4102, 4101, 4100, 4099, 4098, 4097, 3043, 2897, 2843, 1974, 1790, 1677, 1552,
1416, 1379, 1331, 1288, 1251, 1250, 1249, 1248, 1247, 1236, 1225, 1224, 1223, 1212, 1201, 1200, 1199, 1198,
1197, 1196, 1195, 1194, 1193, 1192, 1191, 1190, 1189, 1188, 1187, 1186, 1185, 1184, 1183, 1182, 1181, 1180,
1179, 0};
```

```
int cumf_TCOEF2_intra[104]={16383, 13242, 11417, 10134, 9254, 8507, 8012, 7556, 7273, 7062, 6924, 6839,
6741, 6108, 5851, 5785, 5719, 5687, 5655, 5028, 4917, 4864, 4845, 4416, 4159, 4074, 3903, 3871, 3870, 3765,
3752, 3751, 3659, 3606, 3580, 3541, 3540, 3514, 3495, 3494, 3493, 3474, 3473, 3441, 3440, 3439, 3438, 3425,
3424, 3423, 3422, 3421, 3420, 3401, 3400, 3399, 3398, 3397, 3396, 2530, 2419, 2360, 2241, 2228, 2017, 1687,
1576, 1478, 1320, 1281, 1242, 1229, 1197, 1178, 1152, 1133, 1114, 1101, 1088, 1087, 1086, 1085, 1072, 1071,
1070, 1069, 1068, 1067, 1066, 1065, 1064, 1063, 1062, 1061, 1060, 1059, 1058, 1057, 1056, 1055, 1054, 1053,
1052, 0};
```

```
int cumf_TCOEF3_intra[104]={16383, 12741, 10950, 10071, 9493, 9008, 8685, 8516, 8385, 8239, 8209, 8179,
8141, 6628, 5980, 5634, 5503, 5396, 5327, 4857, 4642, 4550, 4481, 4235, 4166, 4151, 3967, 3922, 3907, 3676,
3500, 3324, 3247, 3246, 3245, 3183, 3168, 3084, 3069, 3031, 3030, 3029, 3014, 3013, 2990, 2975, 2974, 2973,
2958, 2943, 2928, 2927, 2926, 2925, 2924, 2923, 2922, 2921, 2920, 2397, 2298, 2283, 1891, 1799, 1591, 1445,
1338, 1145, 1068, 1006, 791, 768, 661, 631, 630, 615, 592, 577, 576, 561, 546, 523, 508, 493, 492, 491, 476, 475,
474, 473, 472, 471, 470, 469, 468, 453, 452, 451, 450, 449, 448, 447, 446, 0};
```

```
int cumf_TCOEFr_intra[104]={16383, 12514, 10776, 9969, 9579, 9306, 9168, 9082, 9032, 9000, 8981, 8962,
8952, 7630, 7212, 7053, 6992, 6961, 6940, 6195, 5988, 5948, 5923, 5370, 5244, 5210, 4854, 4762, 4740, 4384,
4300, 4288, 4020, 3968, 3964, 3752, 3668, 3511, 3483, 3354, 3322, 3205, 3183, 3108, 3046, 2999, 2981, 2974,
2968, 2961, 2955, 2949, 2943, 2942, 2939, 2935, 2934, 2933, 2929, 2270, 2178, 2162, 1959, 1946, 1780, 1651,
1524, 1400, 1289, 1133, 1037, 942, 849, 763, 711, 591, 521, 503, 496, 474, 461, 449, 442, 436, 426, 417, 407, 394,
387, 377, 373, 370, 367, 366, 365, 364, 363, 362, 358, 355, 352, 351, 350, 0};
```

```
int cumf_SIGN[3]={16383, 8416, 0};
```

```
int cumf_LAST[3]={16383, 9469, 0};
```

```
int cumf_LAST_intra[3]={16383, 2820, 0};
```

```
int cumf_RUN[65]={16383, 15310, 14702, 13022, 11883, 11234, 10612, 10192, 9516, 9016, 8623, 8366, 7595,
7068, 6730, 6487, 6379, 6285, 6177, 6150, 6083, 5989, 5949, 5922, 5895, 5828, 5774, 5773, 5394, 5164, 5016,
4569, 4366, 4136, 4015, 3867, 3773, 3692, 3611, 3476, 3341, 3301, 2787, 2503, 2219, 1989, 1515, 1095, 934, 799,
691, 583, 435, 300, 246, 206, 125, 124, 97, 57, 30, 3, 2, 1, 0};
```

```
int cumf_RUN_intra[65]={16383, 10884, 8242, 7124, 5173, 4745, 4246, 3984, 3034, 2749, 2607, 2298, 966, 681,
396, 349, 302, 255, 254, 253, 206, 159, 158, 157, 156, 155, 154, 153, 106, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26,
25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0};
```

```
int cumf_LEVEL[255]={16383, 16382, 16381, 16380, 16379, 16378, 16377, 16376, 16375, 16374, 16373, 16372,
16371, 16370, 16369, 16368, 16367, 16366, 16365, 16364, 16363, 16362, 16361, 16360, 16359, 16358, 16357,
16356, 16355, 16354, 16353, 16352, 16351, 16350, 16349, 16348, 16347, 16346, 16345, 16344, 16343, 16342,
16341, 16340, 16339, 16338, 16337, 16336, 16335, 16334, 16333, 16332, 16331, 16330, 16329, 16328, 16327,
16326, 16325, 16324, 16323, 16322, 16321, 16320, 16319, 16318, 16317, 16316, 16315, 16314, 16313, 16312,
16311, 16310, 16309, 16308, 16307, 16306, 16305, 16304, 16303, 16302, 16301, 16300, 16299, 16298, 16297,
16296, 16295, 16294, 16293, 16292, 16291, 16290, 16289, 16288, 16287, 16286, 16285, 16284, 16283, 16282,
16281, 16280, 16279, 16278, 16277, 16250, 16223, 16222, 16195, 16154, 16153, 16071, 15989, 15880, 15879,
15878, 15824, 15756, 15674, 15606, 15538, 15184, 14572, 13960, 10718, 7994, 5379, 2123, 1537, 992, 693, 611,
516, 448, 421, 380, 353, 352, 284, 257, 230, 203, 162, 161, 160, 133, 132, 105, 104, 103, 102, 101, 100, 99, 98,
97, 96, 95, 94, 93, 92, 91, 90, 89, 88, 87, 86, 85, 84, 83, 82, 81, 80, 79, 78, 77, 76, 75, 74, 73, 72, 71, 70, 69, 68,
67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38,
37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8,
7, 6, 5, 4, 3, 2, 1, 0};
```

```
int cumf_LEVEL_intra[255]={16383, 16379, 16378, 16377, 16376, 16375, 16374, 16373, 16372, 16371, 16370,
16369, 16368, 16367, 16366, 16365, 16364, 16363, 16362, 16361, 16360, 16359, 16358, 16357, 16356, 16355,
16354, 16353, 16352, 16351, 16350, 16349, 16348, 16347, 16346, 16345, 16344, 16343, 16342, 16341, 16340,
16339, 16338, 16337, 16336, 16335, 16334, 16333, 16332, 16331, 16330, 16329, 16328, 16327, 16326, 16325,
16324, 16323, 16322, 16321, 16320, 16319, 16318, 16317, 16316, 16315, 16314, 16313, 16312, 16311, 16268,
16267, 16224, 16223, 16180, 16179, 16136, 16135, 16134, 16133, 16132, 16131, 16130, 16129, 16128, 16127,
16126, 16061, 16018, 16017, 16016, 16015, 16014, 15971, 15970, 15969, 15968, 15925, 15837, 15794, 15751,
15750, 15749, 15661, 15618, 15508, 15376, 15288, 15045, 14913, 14781, 14384, 13965, 13502, 13083, 12509,
12289, 12135, 11892, 11738, 11429, 11010, 10812, 10371, 9664, 9113, 8117, 8116, 8028, 6855, 5883, 4710, 4401,
4203, 3740, 3453, 3343, 3189, 2946, 2881, 2661, 2352, 2132, 1867, 1558, 1382, 1250, 1162, 1097, 1032, 967, 835,
681, 549, 439, 351, 350, 307, 306, 305, 304, 303, 302, 301, 300, 299, 298, 255, 212, 211, 210, 167, 166, 165, 164,
163, 162, 161, 160, 159, 158, 115, 114, 113, 112, 111, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54,
53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24,
23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0};
```

```
int cumf_INTRA_AC_DC[4]={16383, 9229, 5461, 0};
```





ロックの番号付けと同じである。2つの色差ブロックに対する動きベクトルMVD<sub>CHR</sub>は、4本の輝度ブロックの動きベクトルを合計し、8で割って求める。結果として得た1/16画素精度のベクトルは、付表F. 1/JT-H263で示したように最も近い1/2画素精度の位置のベクトルに修正される。

付表F. 1/JT-H263 1/16画素精度の色差ベクトル成分の修正  
(ITU-T H. 263)

1/16画素精度における位置	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	/16
修正位置	0	0	0	1	1	1	1	1	1	1	1	1	1	1	2	2	/2

1/2画素の値は6.1.2小節に記述があるように双線形補間で求める。拡張予測モードでは、輝度の予測はF.3節記述のオーバーラップ動き補償により行われる。色差の予測は、(デフォルトの予測モードと同様に)2つの色差ブロックの全画素に対して動きベクトルMVD<sub>CHR</sub>を適用することにより行われる。

**F.3 輝度成分に対するオーバーラップ動き補償**

8×8の輝度予測ブロックの各画素は3つの予測値を重み付け加算し、8で割って(丸めて)求められる。3つの予測値を生成するために3本の動きベクトルを用いる。すなわち現在の輝度成分のブロックに対する動きベクトルと、4本の“リモート”ベクトルの内の2本である。2本のベクトルとは、

- (1) 現在の輝度成分のブロックの左あるいは右側のブロックの動きベクトル
- (2) 現在の輝度成分のブロックの上または下のブロックの動きベクトル

各画素に対して、2つの最も近いブロック境界のブロックのリモート動きベクトルを使う。ブロックの上半分に対しては現在のブロックの上のブロックの動きベクトルを使い、下半分に対しては現在のブロックの下のブロックの動きベクトルを使う(付図F.3/JT-H263参照)。同様に、ブロックの左半分に対しては、現在のブロックの左側のブロックの動きベクトルを使い、右半分に対しては右側のブロックの動きベクトルを使う(付図F.4/JT-H263参照)。

(x,y)は整数画素単位で求めたピクチャ中の位置を表すものとする。

(m,n)はピクチャ中の整数ブロックインデックスで、以下のように与えられる。

$$m = x / 8$$

$$n = y / 8$$

ここで、“/”は切り捨て除算を示す。

(i,j)は8×8ブロック内での整数画素位置を示し、以下のように与えられる。

$$i = x - m \cdot 8$$

$$j = y - n \cdot 8$$

結果として、以下の関係となる。

$$(x,y) = (m \cdot 8 + i, n \cdot 8 + j)$$

$(MV_x^k, MV_y^k)$ は動きベクトル ( $k = 0, 1$ , または  $2$ ) で、1画素、または  $1/2$ 画素のオフセットを含んでもよい。例えば、 $(MV_x^k, MV_y^k)$ は $(-7.0, 13.5)$ とすることができる。ここで、 $(MV_x^0, MV_y^0)$ は現在のブロック  $(m, n)$ の動きベクトルを示し、 $(MV_x^1, MV_y^1)$ は上あるいは下のブロックの動きベクトルを示し、 $(MV_x^2, MV_y^2)$ は、現在のブロック  $(m, n)$ の、左あるいは右の動きベクトルを示す ( $m, n$ は先に定義した)。

ブロックインデックス  $(m, n)$ の  $8 \times 8$ の輝度成分予測ブロックの各画素値  $P(x, y)$ は次の式で決定される。

$$P(x, y) = (q(x, y) \cdot H_0(i, j) + r(x, y) \cdot H_1(i, j) + s(x, y) \cdot H_2(i, j) + 4) / 8$$

ここで、 $q(x, y)$ 、 $r(x, y)$ および  $s(x, y)$ は以下の式で定義される参照ピクチャからの予測値である。

$$q(x, y) = p(x + MV_x^0, y + MV_y^0)$$

$$r(x, y) = p(x + MV_x^1, y + MV_y^1)$$

$$s(x, y) = p(x + MV_x^2, y + MV_y^2)$$

ここで、 $p(x + MV_x^k, y + MV_y^k)$ は参照ピクチャ中の位置  $(x + MV_x^k, y + MV_y^k)$ における予測値である。注意として、 $(x + MV_x^k, y + MV_y^k)$ はピクチャの外側でもよく、1画素または、 $1/2$ 画素位置を取ることができる。 $1/2$ 画素の動きベクトルを使っている場合、 $p(x + MV_x^k, y + MV_y^k)$ は6.1.2小節に記載された内挿処理を適用して得られた値を参照する。

行列  $H_0(i, j)$ 、 $H_1(i, j)$ 、 $H_2(i, j)$ は付図F. 2/J T-H 2 6 3、付図F. 3/J T-H 2 6 3、付図F. 4/J T-H 2 6 3で定義される。ここで  $(i, j)$ はそれぞれ行列の列と行を表す。

スライス構造モード (付属資料K参照)も独立セグメント復号モード (付属資料R参照)もどちらも使われていない時、他のビデオピクチャセグメントからのリモート動きベクトルが、現在のGOB内のリモート動きベクトルと同じように使われる。もし、スライス構造モードか独立セグメント復号モードのいずれかが使われているならば、他のビデオピクチャセグメントからの対応するブロックのリモート動きベクトルが、次の段落で記述された他の状態にかかわらず、現在のブロックの動きベクトルに設定される (ビデオピクチャセグメントの定義は付属資料R参照のこと)。

周囲のマクロブロックの一つが符号化されていない場合、対応するリモート動きベクトルはゼロとする。周囲のブロックの一つがINTRAモードの場合、対応するリモート動きベクトルは、PBフレーム以外では、現在のブロックの動きベクトルで置き換えられる。PBフレームのINTRAモードの場合、INTRAブロックの動きベクトルを用いる (付属資料G参照)。現在のブロックがピクチャの境界にあり周囲のブロックが存在しない場合、対応するリモート動きベクトルは現在の動きベクトルで置き換えられる。更に、現在のブロックがマクロブロックの下側である場合 (図5/J T-H 2 6 3におけるブロック番号3あるいは4)、現在のマクロブロックの下のマクロブロックの、 $8 \times 8$ の輝度成分ブロックのリモート動きベクトルは、現在のブロックの動きベクトルで置き換えられる。

予測値の重みを、付図F. 2/J T-H 2 6 3、付図F. 3/J T-H 2 6 3、付図F. 4/J T-H 2 6 3に示す。

4	5	5	5	5	5	5	4
5	5	5	5	5	5	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	6	6	6	6	5	5
5	5	5	5	5	5	5	5
4	5	5	5	5	5	5	4

付図F. 2 / JT-H263 現在の輝度成分ブロックの動きベクトルを用いた予測値に対する重み、 $H_0$   
(ITU-T H. 263)

2	2	2	2	2	2	2	2
1	1	2	2	2	2	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	2	2	2	2	1	1
2	2	2	2	2	2	2	2

付図F. 3 / JT-H263 現在の輝度成分ブロックの上あるいは下の輝度成分のブロックの  
(ITU-T H. 263) 動きベクトルを用いた予測値に対する重み、 $H_1$

2	1	1	1	1	1	1	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	2	1	1	1	1	2	2
2	1	1	1	1	1	1	2

付図F. 4 / JT-H263 現在の輝度成分ブロックの左あるいは右の輝度成分のブロックの  
(ITU-T H. 263) 動きベクトルを用いた予測値に対する重み、 $H_2$

## 付属資料G

### PBフレームモード

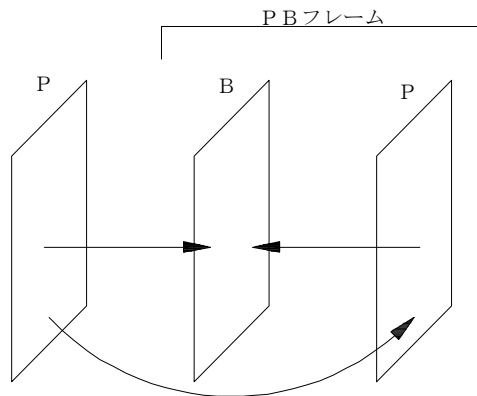
(この付属資料は本標準の必須部分である)

#### G. 1 序論

この付属資料はTTC標準JT-H263のオプションであるPBフレームモードを記述したものである。このモードの能力は外部の手段により通知される(例えばTTC標準JT-H245)。このモードの使用はPTYPEで示される。

PBフレームは1つのユニットとして符号化した2つのピクチャからなる。PBという名称はPピクチャとBピクチャがあるTTC標準JT-H262のピクチャタイプの名前に由来している。PBフレームは、以前に復号されたPピクチャから予測するPピクチャと、以前に復号されたPピクチャ及び現在復号しているPピクチャの両方から予測する一枚のBピクチャからなる。Bピクチャの名称は、Bピクチャの一部分は過去と未来のピクチャから双方向予測されることから来ている。付図G. 1/JT-H263に予測方法を図示している。

PBフレームモードの改良版が、“改良PBフレームモード”として付属資料Mで記述される。この付属資料で記述されるPBフレームモードは改良PBフレームモードの採用より以前に設計されたシステムとの互換性という目的のためだけにここに残される。この理由のため、この付属資料で記述されたPBフレームモードはPLUS PTYPEの使用を必要とするシンタックスの追加機能とともに使うことができない。



付図G. 1/JT-H263 PBフレームモードにおける予測  
(ITU-T H. 263)

#### G. 2 PBフレームとINTRAブロック

PBフレームでは、INTRA符号化モードは次の意味を持つ(5.3.2小節参照)。

- (1) PブロックはINTRA符号化されている。
- (2) BブロックはINTERブロックと同様にINTER予測符号化されている。

PBフレームでは、PTYPEがINTERを示しているピクチャのINTRAマクロブロックにも動きベクトルのデータ(MVD)が存在する。この場合、ベクトルはBブロックにのみ用いられる。符号語MVD<sub>2-4</sub>は、INTRAには使われない(表10/JT-H263参照)。拡張予測モードとPBフレームモードの両方が使われ、かつ周囲のブロックの一つがINTRAで符号化された場合、対応するリモート動きベクトルは現在のブロックの動きベクトルでは置き換えされない。その代わりに、リモート“INTRA”動きベクトルを用いる。

### G. 3 ブロックレイヤ

PBフレームでは、マクロブロックは12のブロックにより構成される。最初に6つのPブロックのデータをTTC標準JT-H263のデフォルトのモードで送り、次に6つのBブロックのデータを送る（5.4節参照）。ブロックレイヤの構造を図11/JT-H263に示す。MCBPCがMBタイプ3または4（表7/JT-H263及び表8/JT-H263参照）の時、マクロブロックのすべてのPブロックにINTRADCが存在する。INTRADCはBブロックにはない。MCBPCあるいはCBPYでPブロックのTCOEFの存在を示す。CBPBでBブロックのTCOEFの存在を示す。

### G. 4 PBフレームでのBピクチャのベクトル計算

Bピクチャのベクトルは以下のように計算される（6.1.1小節参照）。Pピクチャでのベクトル成分MVが1/2画素精度とする（MVは8×8の輝度ブロックのベクトル成分を表しているとする。マクロブロック当たり1本のベクトルのみが送られる場合、MVは4つの8×8輝度ブロックの各々に対し同じ値となる）。Bピクチャを予測するためには、前方及び後方ベクトル、MV<sub>F</sub>とMV<sub>B</sub>が必要である。これらの前方及び後方ベクトルはMV、及びMVD<sub>B</sub>で与えられるデルタベクトルから計算する。

- TR<sub>D</sub>: 最後のピクチャヘッダ（5.1.2小節参照）からの、テンポラルリファレンスTRの増分（あるいはカスタムの画像クロック周波数が使われている時の改良PBフレームモードでの拡張テンポラルリファレンスETRとテンポラルリファレンスTR）。もしTR<sub>D</sub>が負であるならTR<sub>D</sub> = TR<sub>D</sub> + d。ここでCIF画像の周波数ならばd = 256、カスタム画像クロック周波数に対してはd = 1024である。
- TR<sub>B</sub>: 5.1.2.1小節参照。

MV<sub>D</sub>をMVD<sub>B</sub>によって与えられたデルタベクトル成分とし、ベクトル成分MVに対応しているとする。MVD<sub>B</sub>が存在しない場合、MV<sub>D</sub>はゼロにする。MVD<sub>B</sub>が存在する場合、MVD<sub>B</sub>で与えられたのと同じMV<sub>D</sub>をマクロブロック内の4つの輝度成分Bブロックに使う。

このとき、MV<sub>F</sub>とMV<sub>B</sub>は以下の式で1/2画素単位で与えられる。

$$\begin{aligned} MV_F &= (TR_B \times MV) / TR_D + MV_D \\ MV_B &= ((TR_B - TR_D) \times MV) / TR_D && MV_Dがゼロの場合 \\ MV_B &= MV_F - MV && MV_Dがゼロでない場合 \end{aligned}$$

ここで、“/”は、切捨ての除算を意味する。スケーリングはP及びBピクチャの実際の時間的位置を反映している。MV<sub>F</sub>の値の範囲が限定されている利点を利用する。MVD<sub>B</sub>に対する各可変長符号語は、一对の異なった値を表す。一对の値の片方のみが許容範囲内にあるMV<sub>F</sub>の値となる（デフォルトでは[-16, 15.5]、無制限動きベクトルモードでは[-31.5, 31.5]）。MV<sub>F</sub>とMV<sub>B</sub>に対する演算式は、INTRAブロックにおいて、Bブロックを予測するためにベクトルデータを用いる場合にも使われる。

色差ブロックに対しては、MV<sub>F</sub>は4つの対応する輝度成分のMV<sub>F</sub>ベクトルを合計し、その合計を8で割って求める。結果として得られた1/16画素精度のベクトル成分は、付表F.1/JT-H263に示すように最も近い1/2画素の位置へ変換される。MV<sub>B</sub>は4つの対応する輝度成分のMV<sub>B</sub>ベクトルを合計し、その合計を8で割って求める。結果として得られた1/16画素精度のベクトル成分は、付表F.1/JT-H263に示すように最も近い1/2画素の位置へ変換される。

動きベクトルの水平および垂直成分の正の値は、予測される画素の空間的に右あるいは下の参照ピクチャの画素を用いて予測することを表す。

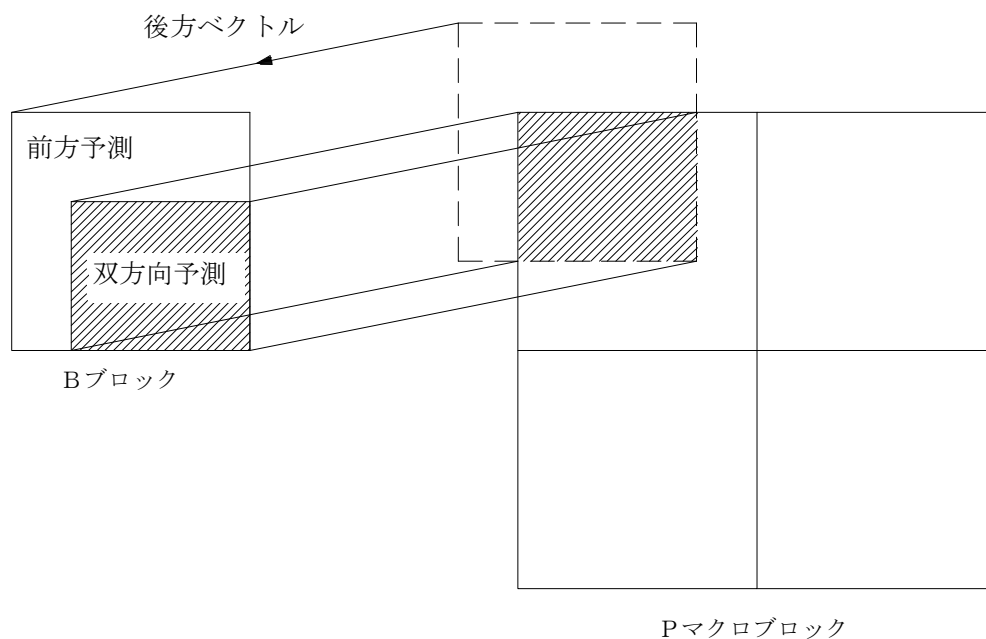
### G. 5 PBフレームでのBブロックの予測

この節では、ブロックとは8×8のブロックを表す。以下の手順は輝度と色差両方のブロックに適用される。最初に、前方と後方ベクトルを計算する。Pマクロブロック（輝度と色差）は、最初に復号、再構築され、クリップされるものとする（6.3.2小節参照）。このマクロブロックはP<sub>REC</sub>と呼ばれる。P<sub>REC</sub>とP<sub>REC</sub>の予測値に基づき、Bブロックに対する予測値を計算する。

Bブロックに対する予測値には2つのモードがあり、それぞれブロックの異なる部分に使われる。

- (1)  $P_{REC}$  の内部を指している後方ベクトル  $MV_B$  を有する画素に対しては双方向予測を行う。これは、前に復号したピクチャに関連する  $MV_F$  を用いた前方予測値と、 $P_{REC}$  に関連する  $MV_B$  を用いた後方予測値を平均して求める。平均は、2つの予測値を合計し2で割って（切捨て）計算する。
- (2) 他の画素は、以前に復号したピクチャに対する  $MV_F$  を用いて前方予測して得る。

付図G. 2/JT-H263は、ブロックのどの部分が双方向予測されるか（Bブロックの影付きの部分）、あるいは前方予測のみか（Bブロックの残りの部分）を示している。



付図G. 2/JT-H263 Bブロックの前方、双方向予測  
(ITU-T H. 263)

双方向予測は、 $P_{REC}$  の内部を指す後方ベクトル  $MV_B$  を有する画素に使われる。これらの画素は、C言語で示される次の手順で定義される。

定義：  $nh$ : マクロブロック内のブロックの水平位置（0または1）  
 $nv$ : マクロブロック内のブロックの垂直位置（0または1）  
 $mh(nh, nv)$ : 1/2画素精度で表したブロック  $(nh, nv)$  の水平ベクトル成分  
 $mv(nh, nv)$ : 1/2画素精度で表したブロック  $(nh, nv)$  の垂直ベクトル成分  
 $mhc$ : 水平色差ベクトル成分  
 $mvc$ : 垂直色差ベクトル成分

#### 輝度に対する手順

```
for (nh = 0; nh <= 1; nh++) {
  for (nv = 0; nv <= 1; nv++) {
    for (i = nh*8 + max(0, (-mh(nh, nv)+1)/2 - nh*8);
         i <= nh*8 + min(7, 15-(mh(nh, nv)+1)/2 - nh*8); i++) {
      for (j = nv*8 + max(0, (-mv(nh, nv)+1)/2 - nv*8);
           j <= nv*8 + min(7, 15-(mv(nh, nv)+1)/2 - nv*8); j++) {
        画素 (i, j) を双方向予測;
      }
    }
  }
}
```

```
    }  
  }  
}
```

#### 色差に対する手順

```
for (i = max(0, (-mhc+1)/2); i <= min(7, 7-(mhc+1)/2); i++) {  
    for (j = max(0, (-mvc+1)/2); j <= min(7, 7-(mvc+1)/2); j++) {  
        画素 (i, j)を双方向予測;  
    }  
}
```

双方向予測しない画素は、前方予測のみを行う。

## 付属資料H

### 符号化されたビデオ信号に対する前方誤り訂正

(この付属資料は本標準の必須部分である)

#### H. 1 序論

この付属資料はTTC標準JT-H263で符号化したビデオデータを伝送するためのオプションの前方誤り訂正方法(符号とフレーム化)を記述している。この前方誤り訂正は、前方誤り訂正が外部の手段、例えば多重化あるいはシステムレベルなど、で提供されない場合に用いられるであろう。TTC標準JT-H324では使わない。フレーム化と前方誤り訂正符号は両方ともTTC標準JT-H261と同じである。

#### H. 2 誤り訂正フレーム化

復号器でビデオデータと誤り訂正パリティの情報を識別するために、誤り訂正フレーム化のパターンが含まれている。このパターンは8フレームのマルチフレームからなり、各フレームは1ビットのフレーム化ビット、1ビットのフィルインジケータ( $F_i$ )、符号化データ(あるいはすべて1で埋めた)492ビット、そして18ビットのパリティで構成されている(付図H.1/JT-H263)。各マルチフレームにおいて、8フレームそれぞれのフレーム化ビットからなるフレームアライメントパターンは以下である。

$$(S_1S_2S_3S_4S_5S_6S_7S_8) = (00011011)$$

フィルインジケータ( $F_i$ )は符号器でゼロにセットしても良い。この場合、492の(すべて1の)連続したフィルビットを、492ビットの符号化されたデータの代わりに用いる。これはスタッフ化データとして使うことができる(3.6節参照)。

#### H. 3 誤り訂正符号

誤り訂正符号は、BCH(511, 493)前方誤り訂正符号である。復号器でのこの使用はオプションである。パリティは493ビットの符号、すなわち1ビットのフィルインジケータ( $F_i$ )と492ビットの符号化されたビデオデータに対して計算される。

生成多項式は次式で与えられる。

$$g(x) = (x^9 + x^4 + 1)(x^9 + x^6 + x^4 + x^3 + 1)$$

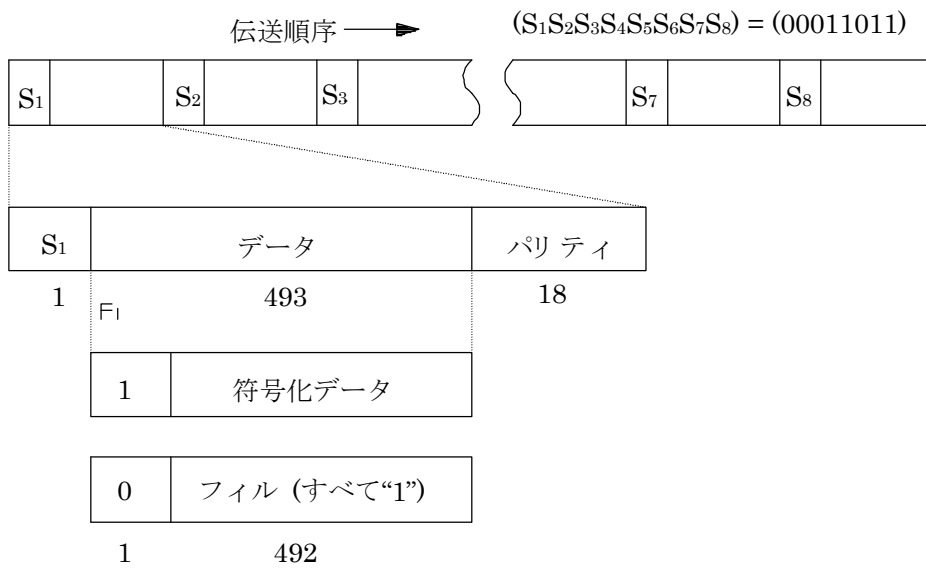
例: 入力データが“01111...11”(493ビット)の場合、訂正パリティビットは“011011010100011011”(18ビット)となる。

#### H. 4 誤り訂正フレーム化の再ロック時間

3つの連続した誤り訂正フレームアライメントパターン(24ビット)を受信することによりフレームロックが実現したとみなす。復号器は、誤り訂正のフレーム化の位相が変化した後、フレームロックが34000ビット内に再確立するように設計されなければならない。

注: これは再ロック期間内に、誤り訂正フレーム化シーケンスの3つの正しい位相のエミュレーションをビデオデータが含んでいないと仮定している。





付図H. 1 / JT-H263 誤り訂正フレーム  
(ITU-T H. 263)

## 付属資料 I

### 拡張 INTRA 符号化モード

(この付属資料は本標準の必須部分である)

この付属資料は、TTC 標準 JT-H 263 のオプションである拡張 INTRA 符号化モードを記述したものである。TTC 標準 JT-H 263 のこのモードの能力は外部の手段により通知される(例えば TTC 標準 JT-H 245)。このモードの使用は、ピクチャヘッダの PLUSPTYPE フィールドで示される。

#### I. 1 序論

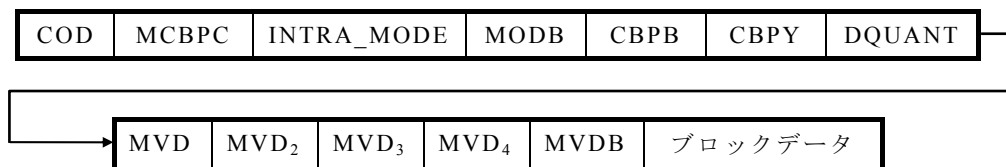
このオプションであるモードは、‘INTRA’タイプのマクロブロックの復号を変える(他のタイプのマクロブロックには影響を与えない)。INTRA マクロブロックの符号化効率は以下の手段を用いることにより改善される。

- (1) 隣接 INTRA ブロックを使用した ( $Y, C_B, C_R$ ) の同じ成分同志での INTRA ブロック予測
- (2) INTRA 係数の修正逆量子化
- (3) INTRA 係数のための個別の可変長符号

ある特定の INTRA 符号化ブロックは、現在復号しているブロックの上部のブロック、現在復号しているブロックの左側のブロック、もしくは両方のブロックから予測してもよい。隣接ブロックが INTRA 符号化されていない状況、または同じビデオピクチャセグメントでないような状況では特別な場合が存在する。ブロック予測は、現在復号しているブロックと同じ輝度または色差成分 ( $Y, C_B, C_R$ ) のデータをいつでも使用する。予測においては DC 係数はある方法でいつも予測される。AC 係数の第 1 行が上部のブロックの第 1 行の AC 係数から予測されるか、または AC 係数の第 1 列が左側のブロックの第 1 列の AC 係数から予測されるか、または DC 係数だけが上部と左側のブロックの平均値として予測され、これらの予測はマクロブロック毎に通知される。残りの AC 係数は決して予測されない。INTRADC 係数には固定ステップサイズ 8 が使われる本標準の本文とは異なり、量子化ステップサイズを変更してもよいように INTRADC 係数の逆量子化が修正される。すべての INTRA 係数の逆量子化は量子化再生空間の不感帯なしで実行される。

#### I. 2 シンタックス

拡張 INTRA 符号化モードを使用するとき、マクロブロックレイヤのシンタックスは付図 I. 1 / JT-H 263 で規定されるように置き換えられる。付図 I. 1 / JT-H 263 に示すシンタックスは INTRA マクロブロックに付加的な INTRA\_MODE を挿入する点を除いて 5. 3 節に定義されたものと同様である。MCBPC が INTRA タイプのマクロブロック (マクロブロックタイプ 3 または 4) を示す場合にだけ INTRA\_MODE は存在する。予測モードは付表 I. 1 / JT-H 263 に示す可変長符号により符号化される。INTRA マクロブロックにつきひとつの予測モードが伝送される。



付図 I. 1 / JT-H 263 マクロブロックレイヤの構成  
(ITU-T H. 263)

付表 I. 1 / JT-H263 INTRA\_MODE の可変長符号  
(ITU-T H. 263)

インデックス	予測モード	可変長符号
0	0 (DC のみ)	0
1	1 (垂直方向の DC と AC)	10
2	2 (水平方向の DC と AC)	11

### 1.3 復号処理

ジグザグ走査に加え 2 種類の走査が採用される。追加された 2 種類の走査を付図 I. 2 / JT-H263 の (a)、(b) に示す。ジグザグ走査は図 14 / JT-H263 に示す。

INTRA 符号化ブロックでは、予測モード 0 の場合、図 14 / JT-H263 に示すジグザグ走査がマクロブロックのすべてのブロックに対して選択され、そうでない場合は、予測方向がマクロブロック走査の選択に使用される。

予測モード 1 は、予測を行なう場合に垂直方向に隣接するブロックを使用する。この予測モードは強い水平周波数成分が支配的である INTRA ブロックのために設計されており、垂直方向に隣接するブロックが現在のブロックの水平周波数成分を予測するために使用され、垂直 AC 成分を表す全ての係数はゼロ予測される。この時、交互水平走査を使用して、強い水平周波数成分を垂直周波数成分より優先して走査する走査パターンが選ばれる。

予測モード 2 は、予測を行う場合に水平方向の隣接ブロックを使用する。この予測モードは強い垂直周波数成分が支配的である INTRA ブロックのために設計されており、水平方向に隣接するブロックが現在のブロックの垂直周波数成分を予測するために使用され、水平 AC 成分を表す全ての係数はゼロ予測される。この時、交互垂直走査を使用して、強い垂直周波数成分を水平周波数成分より優先して走査する走査パターンが選ばれる。

INTRA でないブロックは、 $8 \times 8$  のブロックの変換係数は図 14 / JT-H263 に示される ‘ジグザグ’ 走査により走査される。

1	2	3	4	11	12	13	14
5	6	9	10	18	17	16	15
7	8	20	19	27	28	29	30
21	22	25	26	31	32	33	34
23	24	35	36	43	44	45	46
37	38	41	42	47	48	49	50
39	40	51	52	57	58	59	60
53	54	55	56	61	62	63	64

(a) 交互水平走査

1	5	7	21	23	37	39	53
2	6	8	22	24	38	40	54
3	9	20	25	35	41	51	55
4	10	19	26	36	42	52	56
11	18	27	31	43	47	57	61
12	17	28	32	44	48	58	62
13	16	29	33	45	49	59	63
14	15	30	34	46	50	60	64

(b) 交互垂直走査 (TTC 標準 JT-H262 参照)

付図 I. 2 / JT-H263 拡張 INTRA 符号化における交互 DC T 走査パターン  
(ITU-T H. 263)

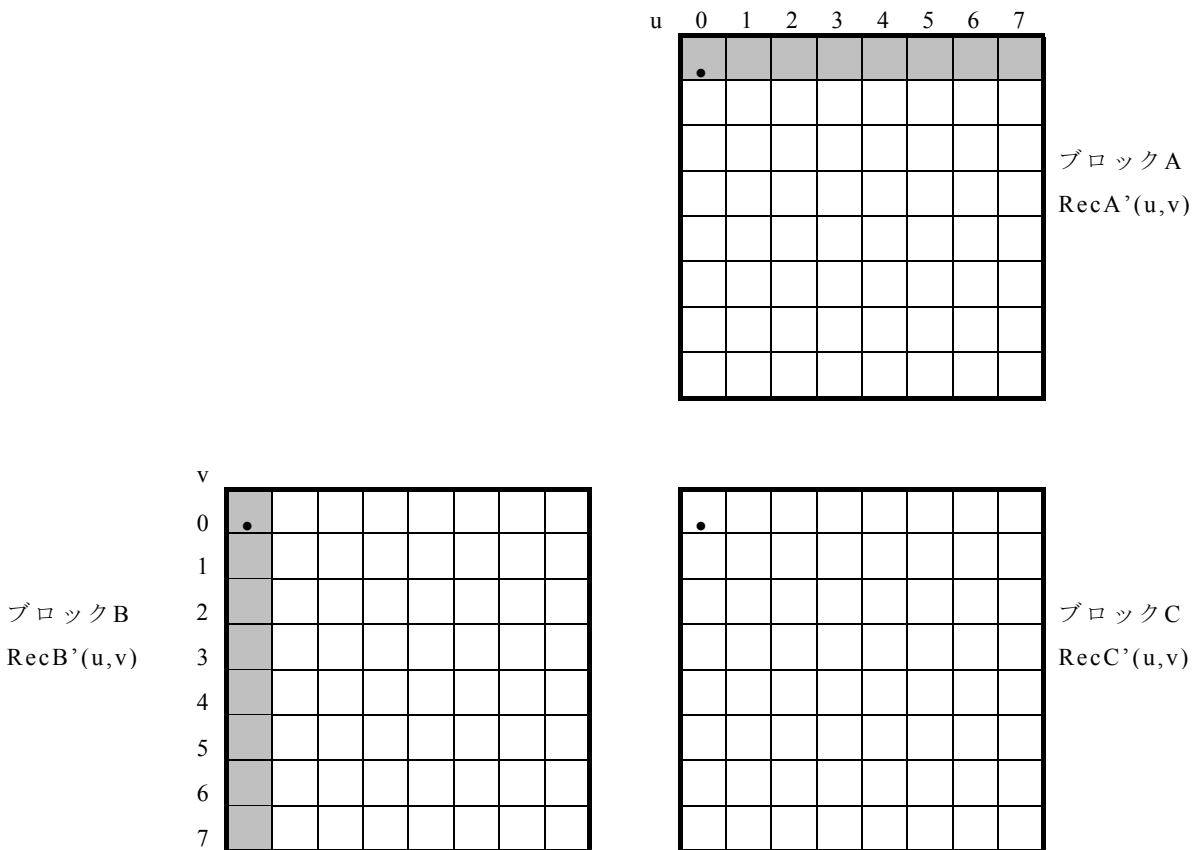
全ての INTRADC 及び INTRAAC 係数のために別々の可変長符号テーブルが使用される。このテーブルは付表 I. 2 / JT-H263 に規定される。付表 I. 2 / JT-H263 に使用されている可変長符号語エントリは、拡張 INTRA 符号化が使用されていない時に使われる通常の TCOEF テーブル (表 16 / JT-H263) で使われているものと同じであるが、LEVEL や RUN の解釈は異なり、また LAST の変更はないことに注意すること。

INTRA\_MODE の値によるが、以下に述べるような予測器に加えなければならない予測誤差である係数が 1 個または 8 個ある。付図 I. 3 / JT-H263 は同じ成分 ( $Y$ ,  $C_B$  または  $C_R$ ) の最終的に再生された DC T レベルの 3 個の  $8 \times 8$  ブロックを示し、それらは  $RecA'(u, v)$ ,  $RecB'(u, v)$ ,  $RecC'(u, v)$  と表示される。ここで  $u, v$  はそれぞれ行 (水平方向) と列 (垂直方向) を示す。再生処理は 6. 2. 1 小節で記述した処理とは異なる。表 15 / JT-H263 を使用方法に対して、INTRADC 誤差は可変ステップサイズを使用する異なった

方法で再生され、最終的な再生係数の値を得るために予測器に誤差が加えられる。6. 2. 1小節での記述とは異なり、INTRADC以外のINTRA係数もまた不感帯のない再生空間を用いることにより再生され、いくつかの場合、最終の再生係数の値を得るために予測器に加算する。ブロックはDCおよびAC予測誤差を含んでもよい。

MCBPCとCBPYの定義は、拡張INTRA符号化が使用されている場合には変更される。拡張INTRA符号化が使用されている場合、INTRADC変換係数は別個の場合としては扱われず、MCBPCやCBPYに関するAC係数と同様な方法で取り扱われる。これは、0のINTRADCはLEVELとして符号化されないで、単に次に続くAC係数のランを増やすことを意味する。

改良PBフレーム（付属資料M参照）のB部分の逆量子化処理は、拡張INTRA符号化モードの使用によって変更されることはない。



付図 I. 3 / JT-H263 DCT領域での3個の近傍ブロック  
(ITU-T H. 263)

RecC(u, v) は、現在のブロックの再生された係数の誤差として定義される。すべてのINTRA係数について再生誤差の値は以下の式で与えられる。

$$\text{RecC}(u, v) = 2 * \text{QUANT} * \text{LEVEL}(u, v) \quad u=0, \dots, 7, v=0, \dots, 7.$$

(上記式中のLEVEL(u, v)は、大きさと符号を持つ量を示していることに注意)

RecC'(u, v) は、現在のブロックの最終的な再生された係数の値として定義される。(これは、予測のための修正、次に示す奇数化処理及びクリッピング処理後の値である) INTRA\_MODEフィールドにおいて通知される適切な予測にRecC(u, v)を加え、奇数化処理が必要な場合にはDC係数の最下位ビットを変更し、クリッピング処理を行うことで最終的な再生係数値RecC'(u, v)が得られる。

RecA'(u, v)は、現在のブロックのすぐ上のブロックの最終再生係数の値を示す。RecB'(u, v)は、現在のブロックのすぐ左側のブロックの最終再生係数の値を示す。

ブロックCの係数値予測においてブロックA、ブロックBの再生係数を使用することができるかどうかは、ブロックCと同じビデオピクチャセグメントにブロックA、ブロックBが含まれているかどうか依存する。次に示さ

れた条件を満たす場合にだけ、あるブロックは他のブロックと同じ‘ビデオピクチャセグメント’に属するブロックと定義される。

- (1) 関連したブロックがピクチャの境界内にある場合。
- (2) スライス構造モード（付属資料K参照）でないとき、関連したブロックが同じGOBにあるか、あるいはGOBヘッダが現在のGOBに存在しない場合。
- (3) スライス構造モードのとき、関連するブロックは同じスライス内にある場合。

復号されるブロックCは、以下に示すとおりブロックCと同じビデオピクチャセグメントにあるINTRAブロックからのみ予測される。

予測モード0（DC予測のみ）で、ブロックAとBがともにブロックCと同じビデオピクチャセグメントにあるINTRAブロックである場合は、ブロックCのDC係数はブロックAとブロックBのDC係数の平均（切り捨てによる）から予測される。2個のブロックAとBのどちらか一方がブロックCと同じビデオピクチャセグメントにあるINTRAブロックである場合は、2個のブロックのうちこの1個のブロックのDC係数が予測モード0に対する予測値として使用される。2個のブロックA、BのどちらもブロックCと同じビデオピクチャセグメントのINTRAブロックでない場合は、予測はDC係数の予測値として値1024を使用する。

予測モードが1または2（垂直DC及びACまたは水平DC及びAC予測）で、参照ブロック（ブロックAまたはブロックB）がブロックCと同じビデオピクチャセグメントのINTRAブロックでない場合は、ブロックCの予測にDC係数の予測値として値1024、AC係数の予測値として値0が使用される。

‘奇数化処理’は、IDCTミスマッチ誤差の影響を最小限にとどめるためにDC係数に対して行われる。ある係数値、特に(0, 0), (0, 4), (4, 0), (4, 4)の係数値は、異なるIDCT実現手段間において丸め誤差ミスマッチを引き起こす可能性がある。例えば、ある整数kに対する $8k+4$ のDC係数は固定値 $k+0.5$ の値をもった逆変換ブロックを発生させ、このわずかな誤差は異なる実現手段の異なる方向への丸め誤差を引き起こす。

-2048から2047の範囲に制限する関数として、clipAC()を定義する。0から2047の範囲に制限する関数として、clipDC()を定義する。oddifyclipDC(x)関数は次のように定義する。

```
If (x が偶数である) {
    result = clipDC(x+1)
} else {
    result = clipDC(x)
}
```

各INTRA予測モードに対する再生を次のように定義する。演算子“/”は、切り捨てを行なう除算である。

モード0：DC予測だけ

```
RecC'(u,v) = clipAC( RecC(u,v) ) (u,v) ≠ (0,0), u = 0, ..., 7, v = 0, ..., 7.
If (ブロック A、B が共に INTRA 符号化され、ブロック C と同じビデオピクチャセグメントにある) {
    tempDC = RecC(0,0) + ( RecA'(0,0) + RecB'(0,0) ) / 2
} else {
    If (ブロック A が INTRA 符号化され、ブロック C と同じビデオピクチャセグメントにある) {
        tempDC = RecC(0,0) + RecA'(0,0)
    } else {
        If (ブロック B が INTRA 符号化され、ブロック C と同じビデオピクチャセグメントにある) {
            tempDC = RecC(0,0) + RecB'(0,0)
        } else {
            tempDC = RecC(0,0) + 1024
        }
    }
}
RecC'(0,0) = oddifyclipDC( tempDC )
```

モード1：上部ブロックからのDC、AC予測

```
If (ブロック A が I N T R A 符号化され、ブロック C と同じビデオピクチャセグメントにある) {  
    tempDC = RecC(0,0) + RecA'(0,0)  
    RecC'(u,0) = clipAC( RecC(u,0) + RecA'(u,0) )           u = 1, ..., 7,  
    RecC'(u,v) = clipAC( RecC(u,v) )                       u = 0, ..., 7, v = 1, ..., 7.  
} else {  
    tempDC = RecC(0,0) + 1024  
    RecC'(u,v) = clipAC( RecC(u,v) )                       (u,v) ≠ (0,0), u = 0, ...,7, v = 0, ..., 7  
}  
RecC'(0,0) = oddifyclipDC( tempDC )
```

モード2：左ブロックからのDC、AC予測

```
If (ブロック B が I N T R A 符号化され、ブロック C と同じビデオピクチャセグメントにある) {  
    tempDC = RecC(0,0) + RecB'(0,0)  
    RecC'(0,v) = clipAC( RecC(0,v) + RecB'(0,v) )           v = 1, ..., 7,  
    RecC'(u,v) = clipAC( RecC(u,v) )                       u = 1, ..., 7, v = 0, ..., 7.  
} else {  
    tempDC = RecC(0,0) + 1024  
    RecC'(u,v) = clipAC( RecC(u,v) )                       (u,v) ≠ (0,0), u = 0, ..., 7, v = 0, ..., 7  
}  
RecC'(0,0) = oddifyclipDC( tempDC )
```

付表 I. 2 / JT-H263 INTRA TCOEF の可変長符号表  
(ITU-T H. 263)

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
0	0	0	1	3	10s
1	0	1	1	5	1111 s
2	0	3	1	7	0101 01s
3	0	5	1	8	0010 111s
4	0	7	1	9	0001 1111 s
5	0	8	1	10	0001 0010 1s
6	0	9	1	10	0001 0010 0s
7	0	10	1	11	0000 1000 01s
8	0	11	1	11	0000 1000 00s
9	0	4	3	12	0000 0000 111s
10	0	9	2	12	0000 0000 110s
11	0	13	1	12	0000 0100 000s
12	0	0	2	4	110s
13	0	1	2	7	0101 00s
14	0	1	4	9	0001 1110 s
15	0	1	5	11	0000 0011 11s
16	0	1	6	12	0000 0100 001s
17	0	1	7	13	0000 0101 0000s
18	0	0	3	5	1110 s
19	0	3	2	9	0001 1101 s
20	0	2	3	11	0000 0011 10s
21	0	3	4	13	0000 0101 0001s
22	0	0	5	6	0110 1s
23	0	4	2	10	0001 0001 1s
24	0	3	3	11	0000 0011 01s
25	0	0	4	6	0110 0s
26	0	5	2	10	0001 0001 0s
27	0	5	3	13	0000 0101 0010s
28	0	2	1	6	0101 1s
29	0	6	2	11	0000 0011 00s
30	0	0	25	13	0000 0101 0011s
31	0	4	1	7	0100 11s
32	0	7	2	11	0000 0010 11s

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
58	1	0	1	5	0111 s
59	1	14	1	10	0000 1100 1s
60	1	20	1	12	0000 0000 101s
61	1	1	1	7	0011 11s
62	1	19	1	12	0000 0000 100s
63	1	2	1	7	0011 10s
64	1	3	1	7	0011 01s
65	1	0	2	7	0011 00s
66	1	5	1	8	0010 011s
67	1	6	1	8	0010 010s
68	1	4	1	8	0010 001s
69	1	0	3	8	0010 000s
70	1	9	1	9	0001 1010 s
71	1	10	1	9	0001 1001 s
72	1	11	1	9	0001 1000 s
73	1	12	1	9	0001 0111 s
74	1	13	1	9	0001 0110 s
75	1	8	1	9	0001 0101 s
76	1	7	1	9	0001 0100 s
77	1	0	4	9	0001 0011 s
78	1	17	1	10	0000 1100 0s
79	1	18	1	10	0000 1011 1s
80	1	16	1	10	0000 1011 0s
81	1	15	1	10	0000 1010 1s
82	1	2	2	10	0000 1010 0s
83	1	1	2	10	0000 1001 1s
84	1	0	6	10	0000 1001 0s
85	1	0	5	10	0000 1000 1s
86	1	4	2	11	0000 0001 11s
87	1	3	2	11	0000 0001 10s
88	1	1	3	11	0000 0001 01s
89	1	0	7	11	0000 0001 00s
90	1	2	3	12	0000 0100 100s

(以下に続く)

付表 I. 2 / JT-H263 INTRA TCOEF の可変長符号表 (続き)  
(ITU-T H. 263)

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
33	0	0	24	13	00001010100s
34	0	0	8	7	010010s
35	0	8	2	11	000001010s
36	0	0	7	7	010001s
37	0	2	4	11	000001001s
38	0	0	6	7	010000s
39	0	12	1	11	000001000s
40	0	0	9	8	0010110s
41	0	0	23	13	00001010101s
42	0	2	2	8	0010101s
43	0	1	3	8	0010100s
44	0	6	1	9	00011100s
45	0	0	10	9	00011011s
46	0	0	12	10	000100001s
47	0	0	11	10	000100000s
48	0	0	18	10	000011111s
49	0	0	17	10	000011110s
50	0	0	16	10	000011101s
51	0	0	15	10	000011100s
52	0	0	14	10	000011011s
53	0	0	13	10	000011010s
54	0	0	20	12	0000100010s
55	0	0	19	12	0000100011s
56	0	0	22	13	00001010110s
57	0	0	21	13	00001010111s

インデックス	LAST	RUN	LEVEL	ビット数	可変長符号
91	1	1	4	12	0000100101s
92	1	0	9	12	0000100110s
93	1	0	8	12	0000100111s
94	1	21	1	13	00001011000s
95	1	22	1	13	00001011001s
96	1	23	1	13	00001011010s
97	1	7	2	13	00001011011s
98	1	6	2	13	00001011100s
99	1	5	2	13	00001011101s
100	1	3	3	13	00001011110s
101	1	0	10	13	00001011111s
102	ESCAPE			7	000011



## 付属資料 J

### デブロッキングフィルタモード

(この付属資料は本標準の必須部分である)

#### J. 1 序論

この付属資料はオプション機能である符号化ループ内のブロック境界フィルタの使用方法を規定する。ブロック境界フィルタの主な目的はブロックによる人工的な影響を軽減する事である。フィルタは $8 \times 8$ の大きさのブロック境界を単位に行われる。動きベクトルは、 $8 \times 8$ または $16 \times 16$ の解像度を持ってよい (J. 2 節参照)。この付属資料で規定される処理は P、I、EP、EI ピクチャ、又は改良 PB フレームの P ピクチャにのみ適用される。(B ピクチャまたは改良 PB フレームの B ピクチャへのフィルタリングの適用も有り得るが、これらへの適用は本標準の範囲外である。しかし、画質改善のためにある種類のフィルタリングの採用が推奨される)。このモードの能力は外部手段 (例えば、TTC 標準 JT-H 2 4 5) により通知される。このモードの使用は、ピクチャヘッダ部の PLUSPTYPE のフィールド内において指示される。

注: IDCT ミスマッチ問題は、小さい量子化ステップサイズを用いて付属資料 J を使い、符号器及び復号器の両方において付属資料 W の IDCT を使わない時、悪化し得る。場合によっては、丸め誤差ミスマッチの量は付属資料 J フィルタ処理の使用によりむしろ増幅されるかもしれない。符号器は、例えば量子化ステップサイズが非常に小さい時は付属資料 J を使わないことにより、この問題を避けるべきである。

#### J. 2 UMV と AP モードとの関係 (付属資料 D、F)

デブロッキングフィルタが単独で使用された場合、付属資料 F で規定されたオーバーラップ動き補償 (OBMC) と同様の画質改善効果がある。両技術が同時に使用された時には、さらなる画質改善効果が達成できる。拡張予測モード (付属資料 F 参照) は 3 つの要素から成り立つ。

- (1) F. 2 節に規定されている、マクロブロックあたり 4 個の動きベクトル
- (2) F. 3 節に規定されている、輝度信号に対するオーバーラップ動き補償
- (3) D. 1 節に規定されているピクチャの領域外を参照する動きベクトル

装置の複雑さに関する制約から、拡張予測モードの一部である OBMC の使用が困難である場合がある。このような場合にも最大限の効果を発揮するために、デブロッキングフィルタモードにおいては、マクロブロックあたり 4 個の動きベクトルを使用する機能やピクチャの領域外を参照する動きベクトルを使用する機能も活用できるようになっている。

要約すると、付属資料 D、F、J に規定されている 3 個のオプションは以下の 5 個の符号化要素を含んでいる。

- (1) ピクチャの領域外を参照する動きベクトル (D. 1 節)
- (2) 動きベクトル範囲の拡大 (D. 2 節)
- (3) マクロブロックあたり 4 個の動きベクトル (F. 2 節)
- (4) 輝度信号のためのオーバーラップ動き補償 (F. 3 節)
- (5) デブロッキングフィルタ (J. 3 節)

付属資料 D、F、J に規定される 3 個のオプションのどれを使用するかにより、上記 5 種類のどの符号化要素が適用されるかを付表 J. 1 / JT-H 2 6 3 は示している。

付表 J. 1 / JT-H263 U MV、AP 及びデブロッキングフィルタに対する符号化要素  
(ITU-T H. 263)

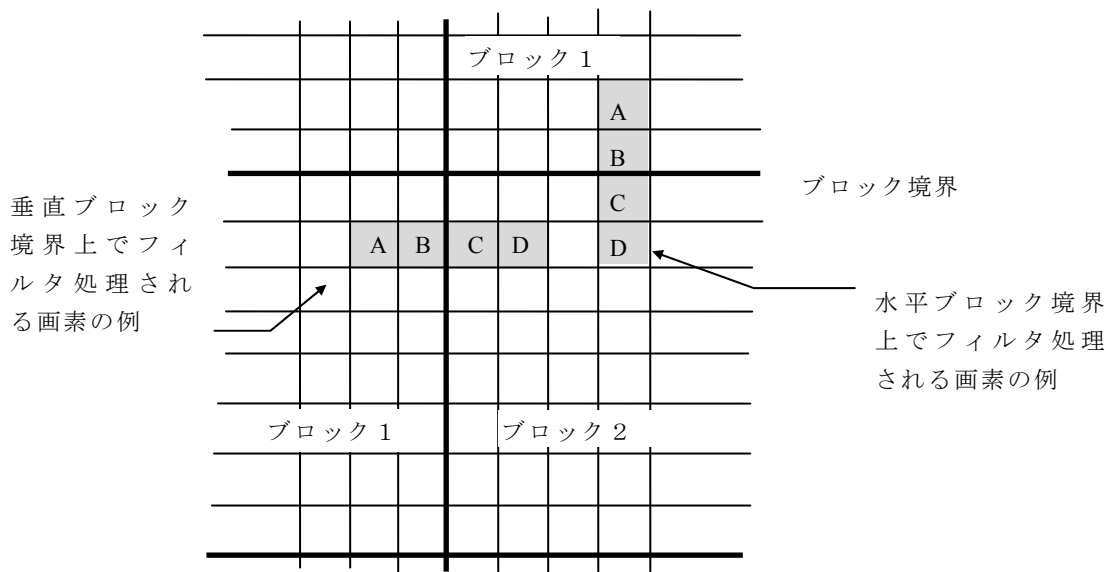
無制限動きベクトルモード	拡張予測モード	デブロッキングフィルタモード	ピクチャの領域外を参照する動きベクトル	動きベクトル範囲の拡大	マクロブロックあたり4個の動きベクトル	輝度信号のためのオーバーラップ動き補償	デブロッキング境界フィルタ
OFF	OFF	OFF	OFF	OFF	OFF	OFF	OFF
OFF	OFF	ON	ON	OFF	ON	OFF	ON
OFF	ON	OFF	ON	OFF	ON	ON	OFF
OFF	ON	ON	ON	OFF	ON	ON	ON
ON	OFF	OFF	ON	ON	OFF	OFF	OFF
ON	OFF	ON	ON	ON	ON	OFF	ON
ON	ON	OFF	ON	ON	ON	ON	OFF
ON	ON	ON	ON	ON	ON	ON	ON

### J. 3 デブロッキングフィルタの定義

フィルタ操作は符号器、復号器ともに  $8 \times 8$  のブロック境界を越えて行われる。再生されたピクチャデータ（予測値と再生された予測誤差との和）は、6.3.2 小節に示される様に、0 から 255 までの範囲にクリップされる。その次にフィルタが適用され、その結果が将来の予測に使うためにピクチャメモリに格納される。フィルタ操作には、最終的な画素値が 0 から 255 の範囲に入る様にさらなるクリッピング処理が含まれる。ピクチャの境界を越えてのフィルタ処理は行われない。そして、独立セグメント復号モードが採用されている場合、スライス構造モードが採用されている時（付属資料 K、R 参照）にはスライスの境界を越えたフィルタ操作は行われず、スライス構造モードが採用されていない時（付属資料 R 参照）には、GOB ヘッダが有る GOB の上側の境界を越えたフィルタ操作は行われない。色差信号は輝度と同様にフィルタ操作される。

この付属資料に記載されているモードが付属資料 M の PB フレームモードと同時に使用される時は、クリッピング処理の後であるがデブロッキング境界フィルタ処理の前に、再生された（G.5 節では  $P_{REC}$  と命名された）P マクロブロックを用いて B マクロブロックの後方予測は行われる。B マクロブロックの前方予測には復号済みの直前のピクチャにフィルタリング操作された信号が用いられる（P マクロブロックの予測に用いられるピクチャ信号と同一）。

デブロッキングフィルタは、水平又は垂直に並ぶ（クリップ済みの）4 個の復号されたピクチャの画素値を用いて行われる。その 4 個の信号を A、B、C、D と呼び、A、B はブロック 1 の内部に有り、C、D はブロック 1 の下側あるいは右側に隣接するブロック 2 の内部に有る。付図 J.1 / JT-H263 にこれらの画素の位置関係を例示する。



付図 J. 1 / JT-H263 フィルタ処理された画素の配置の例  
(ITU-T H. 263)

下記の条件の 1 個、または 2 個が満たされる時のみ、特定の境界を越えてフィルタを適用しなければならない。

条件 1 : ブロック 1 が符号化されたマクロブロックに属する場合  
(COD が 0、又は MB タイプが INTRA の場合)

条件 2 : ブロック 2 が符号化されたマクロブロックに属する場合  
(COD が 0、又は MB タイプが INTRA の場合)

フィルタリングが境界を越えて適用される場合、画素 A、B、C、D は A<sub>1</sub>、B<sub>1</sub>、C<sub>1</sub>、D<sub>1</sub> に置換されなければならない。ここで、

$$B_1 = \text{clip}(B + d_1)$$

$$C_1 = \text{clip}(C - d_1)$$

$$A_1 = A - d_2$$

$$D_1 = D + d_2$$

$$d = (A - 4B + 4C - D) / 8$$

$$d_1 = \text{UpDownRamp}(d, \text{STRENGTH})$$

$$d_2 = \text{clip}_{d_1}((A - D) / 4, d_1 / 2)$$

$$\text{UpDownRamp}(x, \text{STRENGTH}) = \text{SIGN}(x) * (\text{MAX}(0, \text{abs}(x) - \text{MAX}(0, 2 * \text{abs}(x) - \text{STRENGTH}))$$

STRENGTH は QUANT に依存し、フィルタの強さを規定する。

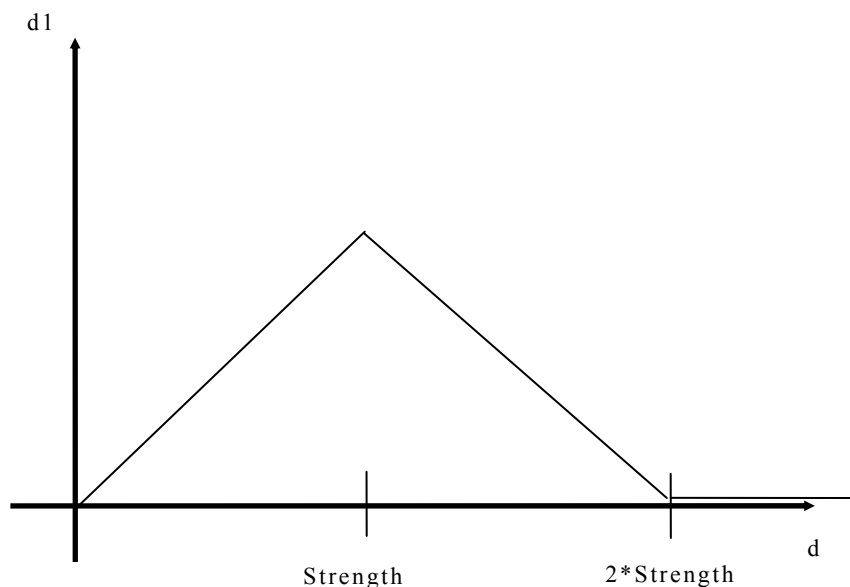
QUANT と STRENGTH との関係は付表 J. 2 / JT-H263 に示す。

QUANT = ブロック 2 の量子化パラメータ (ブロック 2 が符号化マクロブロックに属する場合)、または QUANT = ブロック 1 の量子化パラメータ (ブロック 2 が符号化マクロブロックに属せず、ブロック 1 のみが符号化マクロブロックに含まれる場合)

関数 clip(x) は 6.3.2 小節で定義され、関数 clip<sub>d1</sub>(x, lim) は、x をレンジ ± abs(lim) にクリップする。記号 “/” は、0 方向切り捨ての除算である。

付図 J. 2 / JT-H263 は、変数 d に対する d<sub>1</sub> の値を示す。従って、フィルタは d が 2 \* STRENGTH より小さい時のみ (= 0 を除いて) 効果が有る。これは、ピクチャ信号内の強い真の境界をフィルタで平滑化さ

れる事を防ぐためである。しかし、縮小解像度更新モードが使われている場合は、STRENGTH=∞が適用され、d1は常にdと等しくなる（付属資料QのQ. 7. 2小節参照）。



付図 J. 2 / JT-H263 デブロッキングフィルタモードにおける d に対するパラメータ d1 の値 (ITU-T H. 263)

付表 J. 2 / JT-H263 QUANT とフィルタの STRENGTH の関係 (ITU-T H. 263)

QUANT	STRENGTH	QUANT	STRENGTH
1	1	17	8
2	1	18	8
3	2	19	8
4	2	20	9
5	3	21	9
6	3	22	9
7	4	23	10
8	4	24	10
9	4	25	10
10	5	26	11
11	5	27	11
12	6	28	11
13	6	29	12
14	7	30	12
15	7	31	12
16	7		

d1 の定義は符号器、復号器間の小さな不整合が小さいまま保たれ、複数のピクチャシーケンスに於いて不整合が増大しない事を保証する様に設計されている。例えば、符号器、復号器間で d の ± 1 の大きさの不一致により符

号器側ではフィルタが入り、復号器側では無し（その逆も同様）となるので、単にフィルタの入り切りを切り替える条件が大きな問題となる。

丸め効果のため、フィルタを適用する境界の順番を規定しなければならない。

水平の境界を超えるフィルタリング：

基本的にこの処理が最初に行われる。より正確に言えば、水平の境界を超えるフィルタリングに

用いられる画素  $\begin{pmatrix} A \\ B \\ C \\ D \end{pmatrix}$  は、垂直境界を越える前のフィルタリング処理に影響されてはならない。

垂直境界を越えるフィルタリング：

画素（A， B， C， D）を用いる垂直境界を越えるフィルタリングの前に、画素（A， B， C， D）は水平方向の境界を越えるフィルタリングが行われて画素値の修正がなされていなければならない。

もし、フィルタリング処理に用いられる画素（A， B， C， D）の中の1個又は複数個がピクチャ領域の外側にある時は、フィルタリング処理は行われぬ事に注意すべきである。同様に独立セグメント復号モード（付属資料 R 参照）が採用されており、かつフィルタリング処理に使用される画素の中の1個又は複数個が異なるピクチャセグメントに属する場合（同一ピクチャセグメント中にブロックが有ると考えられる場合に関しては付属資料 I の I. 3 節を参照）にも、フィルタリング処理は行われぬ。

## 付属資料K

### スライス構造モード

(この付属資料は本標準の必須部分である)

#### K. 1 序論

この付属資料はオプションであるTTC標準JT-H263のスライス構造モードを記述している。TTC標準JT-H263のこのモードの能力は外部の手段により通知される(例えばTTC標準JT-H245)。このモードの使用は、ピクチャヘッダのPLUSPTYPEフィールドで示される。いろいろな環境で最適な使用方法を簡単に使えるようにするために、このモードは外部の手段(例えばTTC標準JT-H245)により通知されることが可能な2個のサブモードを含む。2個のサブモードは、矩形スライスが使用されるかどうか、スライスが順番に送信されるかまたは任意の順序で送信されるかどうか、あるいはこれら両方が行われるかどうかを示すために使用される。

スライスは、スライスヘッダとそれに続く走査順に連続したマクロブロックとして定義される。ピクチャのビット列中のピクチャスタートコードの直後に続くスライス(必ずしもマクロブロック0で始まるスライスである必要はない)に対しては例外がある。この場合、K. 2節に記述しているようにスライスヘッダの一部だけが送信される。スライスレイヤはビデオピクチャセグメントを定義し、またこれはオプションであるこのモードにおいてGOBレイヤの代わりに使用される。スライスビデオピクチャセグメントは、ピクチャのマクロブロックの境界から始まり、複数のマクロブロックを含む。同じピクチャの異なるスライスは互いに重なりあってはならず、またすべてのマクロブロックはただひとつのスライスに属さなければならない。

このモードは2個のサブモードをもち、ピクチャヘッダのSSSフィールドで示される。

##### (1) 矩形スライスサブモード (RS)

RSが使用される時、スライスはスライスヘッダの中でマクロブロック単位で表されるSWIパラメータにより規定される幅をもつ矩形領域を占有しなければならない、また矩形領域の中に走査順の複数のマクロブロックを含む。矩形スライスサブモードが使用されない場合は、スライスヘッダにはSWIフィールドはなく、スライスはピクチャ全体で走査順の複数のマクロブロックを含む。

##### (2) 任意スライス順序サブモード (ASO)

ASOが使用される時、スライスはビット列中にどんな順序で現われてもよい。ASOが使用されない場合は、スライスは決まった順序(ただ一つに決まった順序)で送信されなければならない、このときスライスヘッダのMBAフィールドはピクチャにおいて各スライスからそれらの次のスライスまで厳密に増加する。

スライスの境界は、ビット列中のスライスヘッダの位置をビットエラーやパケット損失回復のための再同期点として動作させるために、またピクチャの中で任意の順番でスライスの復号ができるようにするために、単純マクロブロック境界とは異なって扱われる。このようにデータ依存性は現在のピクチャ内のスライス境界を横切ることはできないが、例外として、独立セグメント復号モードのないデブロッキングフィルタモードでは、フィルタはピクチャ内のブロック境界を横切る。しかしオプションの独立セグメント復号モードが使用されなければ、スライス内の動きベクトルにより、予測のために使われる参照ピクチャのスライス境界を横切るデータ依存性が発生し得る。

以下に、スライス境界の位置が再同期点として機能し、付加的な復号遅延を引き起こすことなくスライスを任意の順序で送信することを確実にするために採用された規則を示す。

- (1) 動きベクトルの予測は、あたかもGOBヘッダが存在するという場合(6.1.1小節参照)と同じであり、スライス中での動きベクトルの予測のために現在のスライス外にあるブロックの動きベクトルを使用することを禁止する。
- (2) 拡張INTRA符号化モード(付属資料I参照)ではスライス境界を、あたかもINTRAブロックのDCT係数の予測に関するピクチャ境界のように取り扱う。
- (3) 拡張予測モードでのオーバーラップブロック動き補償で使用するリモート動きベクトルの指定においては、現在のスライス外にあるブロックの動きベクトルをリモート動きベクトルとして使用することを禁止する(F.3節参照)。

## K. 2 スライスレイヤの構造

ピクチャのビット列中のスタートコード直後のスライスを除いて、全てのスライスに適用されるスライスレイヤのシンタックス構造を付図K. 1/JT-H263に示す。ピクチャスタートコードに続くスライスは、エミュレーション防止ビット（以下に規定するようなSEPB1、SEPB3、及び条件によってはSEPB2）及びMBAフィールドのみが含まれ、RSサブモードの場合はさらにSWIフィールドが含まれる。

SSTUF	SSC	SEPB1	SSBI	MBA	SEPB2	SQUANT	SWI	SEPB3	GFID	マクロブロックデータ
-------	-----	-------	------	-----	-------	--------	-----	-------	------	------------

付図K. 1/JT-H263 スライスレイヤの構成  
(ITU-T H. 263)

GFIDについては5. 2. 5小節を、マクロブロックレイヤの記述については5. 3節を参照のこと。

### K. 2. 1 スタッフ化 (SSTUF) (可変長)

8ビット以下からなる可変長の符号語。SSCのバイトアライメントを確保する必要がある場合は常に、符号器は直接SSC符号語の前にこの符号語を挿入しなければならない。SSTUFがある場合には、SSTUFの最後のビットはバイトの最後（最下位）ビットでなければならない。これによりSSC符号語の開始はバイトアライメントされる。復号器はSSTUFを取り去るように設計されなければならない。SSTUFのスタッフ化には0が使用されることに注意すること。

### K. 2. 2 スライススタートコード (SSC) (17ビット)

17ビットの符号語である。値は、“0000 0000 0000 0000 1”である。スライススタートコードはバイトアライメントされなければならない。これはスタートコードの最初のビットがバイトの先頭ビット（最上位ビット）となるようにスタートコードの前にSSTUFを挿入することにより達成される。スライススタートコードはピクチャスタートコードに続くスライスには付かない。

### K. 2. 3 スライスエミュレーション防止ビット1 (SEPB1) (1ビット)

“1”の値を常にもつ1ビットで、スタートコードエミュレーションを防ぐために含まれている。

### K. 2. 4 スライスサブビット列インジケータ (SSBI) (4ビット)

4ビットの符号語で、ピクチャヘッダのCPMが“1”の場合にだけ付けられる。SSBIは付属資料Cに説明されているコンティニューアスペゼンスマルチポイントやビデオ多重化操作のためのスライスに対するサブビット列番号を示す。SSBIの値からサブビット列番号への対応関係を付表K. 1/JT-H263に示す。SSBIはピクチャスタートコードに続くスライスには付かない。

### K. 2. 5 マクロブロックアドレス (MBA) (5/6/7/9/11/12/13/14ビット)

符号語は、現在のピクチャサイズと縮小解像度更新モード（付属資料Qを参照）が有効になっているかによって決まる符号長をもつ。この複数のビットは、左上隅のマクロブロック番号を0として始め、走査順でピクチャの起点から数えた時の、現在のスライスにおける最初のマクロブロックのマクロブロック番号の2進数表現である。MBAは、現在のスライスの先頭のマクロブロックがピクチャ中のどのマクロブロックかを一意に規定する。この符号語の符号長を付表K. 2/JT-H263に示す。カスタムピクチャサイズのMBAフィールド幅は、そのマクロブロックの数以上をもつ、表における第1エントリにより与えられる。MBAの最大値は現在のピクチャの中のマクロブロック数から1を引いた値となる。縮小解像度更新モードの場合、適切なピクチャサイズはピクチャヘッダに示されたピクチャサイズではなく、解像度の低い更新されたピクチャサイズとなる（付属資料Qを参照）。

### K. 2. 6 スライスエミュレーション防止ビット2 (SEPB2) (1ビット)

常に“1”の値をもつ1ビットで、ある条件のときにスタートコードエミュレーションを防ぐために付けられる。ピクチャスタートコードに続くスライス以外のスライスに対して、MBAフィールド幅が11ビットより大きくピ

クチャヘッダのCPMの値が“0”の場合、もしくはMBAフィールド幅が9ビットより大きくピクチャヘッダのCPMの値が“1”の場合にだけ、SEPB2は付けられる。ピクチャスタートコードに続くスライスに対しては、SEPB2は矩形スライスサブモードが使用されている場合にだけ付けられる。

#### K. 2. 7 量子化情報 (SQUANT) (5ビット)

後続のDQUANTによって更新されるまで、そのスライスに使用される量子化器QUANTを示す5ビットの固定長符号語である。符号語は、ステップサイズの半分の値をもつQUANT値の自然2進表現で、範囲が1から31までである。SQUANTは、ピクチャスタートコードに続くスライスには存在しない。

#### K. 2. 8 マクロブロックを単位とするスライス幅インジケーション (SWI) (3/4/5/6/7ビット)

矩形スライスサブモードが有効な場合にだけ存在する符号語である。符号長は、付表K. 3/JT-H263に示すように現在のピクチャサイズと縮小解像度更新モードが有効であるかどうかによって依存する。カスタムピクチャサイズにおけるフィールド幅は、その幅がカスタムピクチャサイズ以上で最も近い標準フォーマットサイズ(QCIF、CIF、...)によって与えられ、最大値はピクチャを横切るマクロブロック総数から1を引いた値である。表の最後の行は16CIFより大きいピクチャサイズに対するフィールド幅を示している。SWIは、MBAにより規定される第1マクロブロック(左上のもの)を持つ現在の矩形スライス幅を参照する。実際のスライス幅の計算は次式で与えられる。

$$\text{実際のスライス幅} = \text{SWI} + 1$$

縮小解像度更新モードの場合、関連するピクチャサイズはピクチャヘッダに示されたピクチャサイズではなく、更新情報におけるより低い解像度のピクチャサイズである。

#### K. 2. 9 スライスエミュレーション防止ビット3 (SEPB3) (1ビット)

常に“1”の値をもつ1ビットで、スタートコードエミュレーションを防ぐために付けられる。

付表K. 1/JT-H263 サブビット列番号とSSBIの値  
(ITU-T H. 263)

サブビット列番号	SSBIフィールド値	エミュレートされたGN値
0	1001	25
1	1010	26
2	1011	27
3	1101	29

付表K. 2/JT-H263 MBAパラメータ規定  
(ITU-T H. 263)

画像 フォーマット	デフォルト		RRUモード	
	最大値	フィールド幅	最大値	フィールド幅
サブQCIF	47	6	11	5
QCIF	98	7	29	6
CIF	395	9	98	7
4CIF	1583	11	395	9
16CIF	6335	13	1583	11
2048×1152	9215	14	2303	12



付表K. 3 / JT-H263 SWIパラメータ規定  
(ITU-T H. 263)

画像 フォーマット	デフォルト		RRUモード	
	最大値	フィールド幅	最大値	フィールド幅
サブQCIF	7	4	3	3
QCIF	10	4	5	3
CIF	21	5	10	4
4CIF	43	6	21	5
16CIF	87	7	43	6
1412...2048 画素の幅	127	7	63	6

## 付属資料 L

### 付加拡張情報仕様

(この付属資料は本標準の必須部分である)

#### L. 1 序論

本付属資料は T T C 標準 J T - H 2 6 3 のピクチャレイヤの P S U P P フィールドで送信される付加拡張情報のフォーマットを記述する。復号器が本付属資料で記述するいくつかまたは全ての拡張能力を提供できるかどうかは外部の手段（例えば T T C 標準 J T - H 2 4 5）により送信してもよい。拡張能力を提供しない復号器はビット列に現れる P S U P P 情報ビットを単純に破棄してもよい。この付加拡張情報の存在は P E I で示され、追加の P E I ビットは P S U P P データの各オクテット間に挿入される（5. 1. 2 4 小節と 5. 1. 2 5 小節で記述）。

本付属資料では“復号ピクチャ”と“表示ピクチャ”を区別する。本付属資料の目的でもあるが、“表示ピクチャ”とは現在のピクチャのためにビデオビット列シンタクスのピクチャレイヤによって定められたものと同様の画像フォーマットを有するピクチャである。本付属資料に記載されている様に、“表示ピクチャ”は復号ピクチャ、前に表示されたピクチャ、ここに記述されている付加拡張情報、そしてある場合には外部で制御される背景ピクチャの一部から構成される。

#### L. 2 P S U P P フォーマット

P S U P P データは 4 ビットの機能タイプ指示 F T Y P E と、それに続く 4 ビットのパラメータデータサイズを決定する D S I Z E、さらにそれに続く機能パラメータデータ D S I Z E オクテットから構成され、オプションで別の機能タイプ指示が続くことが出来る。1 つの機能タイプ指示値は将来の拡張性、すなわち 1 5 以上の別機能を定義できるようにエスケープコードとして定義される。復号器はその復号器がサポートしない機能タイプ指示を受信した場合、その機能に対する機能パラメータデータを破棄し、後続のサポートできそうな機能タイプ指示を調べる。定義された F T Y P E 値を表 L. 1 / J T - H 2 6 3 に示す。

付表 L. 1 / J T - H 2 6 3 F T Y P E 機能タイプ値  
( I T U - T H. 2 6 3 )

0	予約
1	“無動作”
2	全画面凍結要求
3	部分画面凍結要求
4	サイズ変更部分画面凍結要求
5	部分画面凍結解除要求
6	全画面スナップショットタグ
7	部分画面スナップショットタグ
8	ビデオ時間セグメント開始タグ
9	ビデオ時間セグメント終了タグ
10	プログレッシブ高画質化セグメント開始タグ
11	プログレッシブ高画質化セグメント終了タグ
12	クロマキー情報
13	予約
14	予約
15	拡張機能タイプ

#### L. 3 “無動作”

“無動作”機能によって、何も動作しないことが要求される。この機能はスタートコードエミュレーションを防止するために使用される。前の P S U P P オクテットの最終オクテットの終り 5 ビット以上が全てゼロで、他の P S U P P 機能要求を送信しない場合はいつでも、スタートコードエミュレーションの可能性を防止するために“無動作”機能を P S U P P に挿入しなければならない。“無動作”機能は前文で述べた規定により要求されない場合においても送信してよい。無動作機能の D S I Z E はゼロでなければならない。

#### L. 4 全画面凍結要求

全画面凍結要求機能は直前に表示したビデオピクチャの全内容を、現在の復号したピクチャ内容を用いて更新することなく保持しなければならないことを示す。その後、表示ピクチャは現在または後続の P T Y P E 中の画面凍結解除ビットが“1”にセットされるか、タイムアウト発生のいずれかが最初に起こるまで更新されずに保持されなければならない。この要求は、5秒後もしくは5枚後のピクチャのいずれか長い時間のタイムアウトにより失効しなければならない。タイムアウト時間終了時またはそれ以前に別の全画面凍結要求を発行することでタイムアウトを防ぐことができる。（例：発行後5秒以上の時間間隔を示すテンポラルリファレンスを持つ最初のピクチャのヘッダで繰り返し要求する、または発行後5枚目のピクチャのヘッダで繰り返し要求する。）全画面凍結要求機能の D S I Z E はゼロでなければならない。

#### L. 5 部分画面凍結要求

部分画面凍結要求機能は直前に表示したビデオピクチャ内容の特定矩形領域を、現在の復号した同領域のピクチャ内容によって更新することなく保持しなければならないことを示す。その後、表示ピクチャの特定領域は現在または後続の P T Y P E 中の画面凍結解除ビットが“1”にセットされるか、該当領域に影響を与える部分画面凍結解除要求を受信するか、またはピクチャヘッダ中で定められた情報源フォーマットが以前のフォーマットと異なるか、またはタイムアウト発生のいずれかが最初に起こるまで更新されずに保持されなければならない。画像情報源フォーマットに何らかの変更が生じたら全ての有効な部分画面凍結要求は解除されなければならない。この要求は、5秒後もしくは5枚後のピクチャのいずれか長い時間のタイムアウトにより失効しなければならない。タイムアウト時間終了時またはそれ以前に同じ部分画面凍結要求を発行することでタイムアウトを防ぐことができる（例：発行後5秒以上の時間間隔を示すテンポラルリファレンスを持つ最初のピクチャのヘッダで繰り返し要求する、または発行後5枚目のピクチャのヘッダで繰り返し要求する）。部分画面凍結要求の D S I Z E は4でなければならない。それに続く P S U P P の4オクテットには凍結された矩形画面の左上の水平・垂直位置と矩形の幅・高さが含まれ、それぞれに8ビットを用い8画素単位で表現される。例えばビデオ表示の左上端から24画素幅と16画素高さの領域は4つのパラメータで(0, 0, 3, 2)と定義される。

#### L. 6 サイズ変更部分画面凍結要求

サイズ変更部分画面凍結要求機能は直前に表示したビデオピクチャの特定矩形領域の内容をより小さい面積で表示するようにサイズ変更し、現在の復号したピクチャ内容で表示ピクチャの特定領域を更新することなく保持しなければならないことを示す。表示ピクチャの特定領域は現在または後続の P T Y P E 中の画面凍結解除ビットが“1”にセットされるか、該当領域の部分画面凍結解除要求を受信するか、またはピクチャヘッダ中で定められた情報源フォーマットが以前のフォーマットと異なるか、またはタイムアウト発生のいずれかが最初に起こるまで更新されずに保持されなければならない。画像情報源フォーマットに何らかの変更が生じたら全ての有効な部分画面凍結要求は解除されなければならない。この要求は、5秒後もしくは5枚後のピクチャのいずれか長い時間のタイムアウトにより失効しなければならない。タイムアウト時間終了時またはそれ以前に同一の部分画面凍結要求を発行することでタイムアウトを防ぐことができる（例：部分画面凍結要求発行後5秒以上の時間間隔を示すテンポラルリファレンスを持つピクチャのヘッダの中で要求を発行する、または発行後5枚目のピクチャのヘッダで要求を発行する）。サイズ変更部分画面凍結要求の D S I Z E は8でなければならない。それに続く P S U P P データの8オクテットには該当する表示ピクチャの矩形領域を特定するために使用する32ビットが含まれ、次の32ビットはそれに対する復号ピクチャの矩形領域を特定するために用いられる。復号ピクチャ中の矩形領域の幅と高さはいずれも表示ピクチャの矩形領域の幅と高さの $2^i$ 倍に等しくなければならない。ここで  $i$  は1～8の範囲の整数である。これら2つの矩形領域のそれぞれ位置とサイズは、部分画面凍結要求機能で領域が定められたフォーマットと同様のフォーマットを使用して指定される。

#### L. 7 部分画面凍結解除要求

部分画面凍結解除要求機能は表示されたビデオピクチャの特定矩形領域の内容が現在と後続の復号ピクチャによって更新されなければならないことを示す。部分画面凍結解除要求の D S I Z E は4でなければならない。それに続く P S U P P データの4オクテットは部分画面凍結要求機能で領域が定められたフォーマットと同様のフォーマットにて表示ピクチャの矩形領域を指定する。

## L. 8 全画面スナップショットタグ

全画面スナップショットタグ機能は外部で使用するためにビデオ内容の静止画スナップショットとして現在のピクチャにラベル付けすることを示す。全画面スナップショットタグ機能の D S I Z E は 4 でなければならない。それに続く P S U P P データの 4 オクテットは外部用途のためのスナップショット識別番号を指定する。

## L. 9 部分画面スナップショットタグ

部分画面スナップショットタグ機能は外部で使用するためにビデオ内容の静止画スナップショットとして現在のピクチャの特定矩形領域にラベル付けすることを示す。部分画面スナップショットタグ機能の D S I Z E は 8 でなければならない。それに続く P S U P P データの始め 4 オクテットは外部用途のためのスナップショット識別番号を指定し、残り 4 オクテットは部分画面凍結要求機能で領域が定められたフォーマットと同様のフォーマットにて復号ピクチャの矩形領域を指定する。

## L. 10 ビデオ時間セグメント開始タグ

ビデオ時間セグメント開始タグ機能は外部で使用するためにビデオデータの特定のサブシーケンスの始まりがビデオ内容の有用なセクションとしてラベル付けされ、それが現在のピクチャから始まることを示す。タグ付けされたビデオデータのサブシーケンスは、対応するビデオ時間セグメント終了タグ機能の受信により停止されるか、タイムアウトのいずれかが最初に起こるまで継続されなければならない。タグ付けされたサブシーケンスは、5 秒後もしくは 5 枚後のピクチャのいずれか長い時間のタイムアウトにより終了しなければならない。タイムアウト時間終了時またはそれ以前に同一のビデオ時間セグメント開始タグ機能を発行することでタイムアウトを防ぐことができる（例：ビデオ時間セグメント開始タグ機能発行後 5 秒以上の時間間隔を示すテンポラルリファレンスを持つ最初のピクチャのヘッダの中で繰り返し要求する、または要求発行後 5 枚目のピクチャのヘッダで繰り返し要求する）。ビデオ時間セグメント開始タグ機能の D S I Z E は 4 でなければならない。それに続く P S U P P データの 4 オクテットは外部用途のためのビデオ時間セグメント識別番号を指定する。

## L. 11 ビデオ時間セグメント終了タグ

ビデオ時間セグメント終了タグ機能は外部で使用するためにビデオデータの特定のサブシーケンスの終了がビデオ内容の有用なセクションとしてラベル付けされ、直前のピクチャで終了することを示す。ビデオ時間セグメント終了タグ機能の D S I Z E は 4 でなければならない。それに続く P S U P P データの 4 オクテットは外部用途のためのビデオ時間セグメント識別番号を指定する。

## L. 12 プログレッシブ高画質化セグメント開始タグ

プログレッシブ高画質化セグメント開始タグは、連続する動画シーンの表現ではなく、むしろ「現在のピクチャを高画質化するための 0 以上の一連のピクチャが後続する現在のピクチャ」としてラベル付けされるビデオデータの特定のサブシーケンスの開始を意味する。タグ付けされたビデオデータのサブシーケンスは対応するプログレッシブ高画質化セグメント終了タグ機能の受信により停止されるか、タイムアウトのいずれかが最初に起こるまで継続されなければならない。タグ付けされたサブシーケンスは、5 秒後もしくは 5 枚後のピクチャのいずれか長い時間のタイムアウトにより終了しなければならない。タイムアウト時間終了時またはそれ以前に同一のプログレッシブ高画質化セグメント開始タグ機能を発行することでタイムアウトを防ぐことができる（例：プログレッシブ高画質化セグメント開始タグ機能発行後 5 秒以上の時間間隔を示すテンポラルリファレンスを持つ最初のピクチャのヘッダの中で繰り返し要求する、または発行後 5 枚目のピクチャのヘッダで繰り返し要求する）。プログレッシブ高画質化セグメント開始タグ機能の D S I Z E は 4 でなければならない。それに続く P S U P P データの 4 オクテットは外部用途のためのプログレッシブ高画質化識別番号を指定する。

## L. 13 プログレッシブ高画質化セグメント終了タグ

プログレッシブ高画質化セグメント終了タグは、「初期ピクチャを高画質化するための 0 以上の一連のピクチャが後続する初期ピクチャ」としてラベル付けされるビデオデータの特定のサブシーケンスの終了を示し、直前のピクチャで終了する。プログレッシブ高画質化セグメント終了タグ機能の D S I Z E は 4 でなければならない。それに続く P S U P P データの 4 オクテットは外部用途のためのプログレッシブ高画質化セグメント識別番号を指定する。

#### L. 14 クロマキー情報

クロマキー情報機能 (CKIF) は“クロマキー”技術が復号ビデオピクチャにおいて“透過”画素や“半透過”画素を表現するのに使用されることを示す。ディスプレイに表示される際“透過”画素は表示されない。その代わりに前の参照ピクチャまたは外部から制御されたピクチャである背景ピクチャが表示される。半透過画素は現在のピクチャの画素値と背景ピクチャでのそれに対応する画素の値とを混合することによって表示される。1 オクテットはクロマキーに使われる各色成分 ( $Y$ ,  $C_B$  または  $C_R$ ) のキーカラー値の表示に使用される。“半透過”な画素を表現するには二つのしきい値 ( $T_1$  と  $T_2$  とする) を使用する。 $\alpha$  を画素の透過度とすると、 $\alpha = 255$  はその画素が不透明であることを示し、 $\alpha = 0$  はその画素が透過することを示す。その他の値  $\alpha$  に対して、結果として得られる画素値は現在のピクチャの画素値と (外部で規定される) 背景ピクチャの画素値を重み付けし結合したものであるべきである。 $\alpha$  値は“アルファマップ”と呼ばれるイメージを形成するのに用いられてもよい。それゆえそれぞれの成分の結果として得られる値は、次式で与えられる。

$$[\alpha \cdot X + (255 - \alpha) \cdot Z] / 255$$

ここで  $X$  は復号した画素の成分値 ( $Y$ ,  $C_B$  または  $C_R$ ) で、 $Z$  は背景ピクチャの対応する画素の成分値である。 $\alpha$  値は次のように計算される。まずキーカラー値と画素カラーの距離を計算する。

$$d = A_Y (X_Y - K_Y)^2 + A_B (X_B - K_B)^2 + A_R (X_R - K_R)^2$$

ここで  $X_Y$ ,  $X_B$ ,  $X_R$  は  $Y$ ,  $C_B$ ,  $C_R$  の復号画素カラー値であり、 $K_Y$ ,  $K_B$ ,  $K_R$  はそれに対応するキーカラーパラメータ、そして  $A_Y$ ,  $A_B$ ,  $A_R$  はキーフラグビットで、どの色成分がキーとして使われているかを示している。一度距離  $d$  を計算すれば、 $\alpha$  値は次の擬似符号で定められる式で算出してもよい。

```
各々の画素に対して
if (d < T1) then α = 0;
else if (d > T2) then α = 255;
else α = [255 · (d - T1)] / (T2 - T1)
```

しかし、復号器のクロマキー操作を実行する厳密な方法はここでは定義しない。それはその方法の標準的定義が相互接続性にとって必要ないからである。ここに示した処理過程はデータパラメータの意図する解釈法を伝えるために説明の目的で提示した。

得られた  $\alpha$  値は単純に  $X_Y$ ,  $X_B$ ,  $X_R$  の関数であるから、上記の操作を達成するためにルックアップテーブル (LUT) を作るができる。そのような LUT は全ての画素値に対応して、 $2^{8 \times N}$  のエントリーをもつ。ここで  $N$  はキーとして使われる色成分の数である。そのため LUT の各エントリーは対応した  $\alpha$  値を含むだろう。

クロマキーの DSIZE は CKIF と共に送信されるデータ量に従って 1 から 9 (を含む) の範囲でなければならない。1 ピクチャに 2 つ以上の CKIF を送信してはならない。

DSIZE オクテットに続く最初のオクテットは現在のピクチャの表現順序を含まなければならない。つまり、低い表現順序をもつストリームは高い表現順序をもつストリームの背景ピクチャを形成すると仮定される。

もし DSIZE が 1 より大きければ、表現順序オクテット後のオクテットは以下に定める 6 つのフラグビットを送信するのに使用されなければならない。

- 第 1 ビット:  $A_Y$ : 輝度  $Y$  値に対するキーパラメータ  $K_Y$  の存在を示すフラグビット
- 第 2 ビット:  $A_B$ : 色差  $C_B$  値に対するキーパラメータ  $K_B$  の存在を示すフラグビット
- 第 3 ビット:  $A_R$ : 色差  $C_R$  値に対するキーパラメータ  $K_R$  の存在を示すフラグビット
- 第 4 ビット:  $A_1$ : 透過度に対するしきい値パラメータ  $T_1$  の存在を示すフラグビット
- 第 5 ビット:  $A_2$ : 不透過度に対するしきい値パラメータ  $T_2$  の存在を示すフラグビット
- 第 6 ビット: RPB: 背景ピクチャとしての参照ピクチャの使用を示すフラグビット
- 第 7 ビット: 予約
- 第 8 ビット: 予約

DSIZEは1であるか、または1にセットされた $A_Y$ 、 $A_B$ 、 $A_R$ のフラグビット数に2を加え、さらに1にセットされた $A_I$ 、 $A_2$ のビット数の2倍を加えたものでなければならない。もしDSIZEが1より大きい場合、1にセットされたそれぞれのフラグビット $A_Y$ 、 $A_B$ 、 $A_R$ に対して色成分の値を規定するためにそれぞれ追加の1オクテットを送らなければならない。また、おのおの $T_I$ 、 $T_2$ に対してフラグが立てられたしきい値を規定するためにおのおの追加の2オクテットを送らなければならない。これらのオクテットはフラグビットと同じ順序で続く。

もしDSIZEが1であるか3つ全てのキーカラーフラグビット $A_Y$ 、 $A_B$ 、 $A_R$ が0であるなら、前のキーピクチャに使用したキーフラグビット $A_Y$ 、 $A_B$ 、 $A_R$ とキーカラー $K_Y$ 、 $K_B$ 、 $K_R$ を現在のピクチャにも使用すべきである。もしビデオシーケンスに対して前の値が送られていない場合、デフォルトのキーフラグビット $A_Y=1$ 、 $A_B=1$ 、 $A_R=1$ とデフォルトのキーカラー $K_Y=50$ 、 $K_B=220$ 、 $K_R=100$ を前の値として使用すべきである。

もしDSIZEが1であるか、キーしきいフラグビット $A_I$ と $A_2$ の両方が0であるなら、前のキーピクチャに使用したキーしきい値 $T_I$ と $T_2$ を現在のピクチャにも使用すべきである。もしビデオシーケンスに対して前の値が送られていない場合、デフォルトのしきい値 $T_I=48$ と $T_2=75$ を前の値として使用すべきである。

画素が“半透過”（すなわち $T_I < d < T_2$ ）である部分では一般に、復号した画素はクロマキーを使用している成分においてクロマキーカラーを含む。これによりある人工的な色を生じるかもしれない。この問題を解決するために、これらの画素値を背景色と混合する前に調整してもよい。このような修正処理を、フラグビットにより示されたクロマキー操作で使用されている色成分に適応してもよい。その処理を次に示す。

$$X' = K + (T_2 / d) (X - K)$$

ここで $X$ は元の復号画素成分値で、 $X'$ は修正された値である。

調整された画素値 $X'_Y$ 、 $X'_B$ 、 $X'_R$ は $X_Y$ 、 $X_B$ 、 $X_R$ の関数であるから、LUTを用いて色修正をすることができる。このLUTは全ての画素値に対応して $2^{8 \times N}$ のエントリーをもつ。ここで $N$ はキーとして使われる色成分の数である。そのため各エントリーは対応した修正値を含んでいる。

もし参照ピクチャ背景（RPB）フラグビットが1にセットされるならば、これは（現在のピクチャに対して実行される付属資料Pの再サンプリング以前の）時間的に前の参照ピクチャはRPBフラグが“1”である別のピクチャを受信して入れ替えるまで、現在のピクチャ及びその後の全てのクロマキーピクチャの（不透明な）背景として保持すべきであることを示す。もし現在のピクチャが時間的に前にある参照ピクチャを持たない場合（すなわち、現在のピクチャがINTRAまたはEIタイプのピクチャの場合）、RPBフラグビットが参照するピクチャは、もし現在のピクチャがINTERやEPタイプであったなら通常参照ピクチャとなっていたであろうピクチャである。もしRPBフラグビットが“0”であるならば、これは背景は前もって（外部制御により、またはRPBが1の前のピクチャを受信したときに前もって保存した参照ピクチャを用いて）制御されたままであるべきであることを示す。

クロマキー情報機能の発行によって引き起こされるクロマキーの使用は現在のピクチャから開始しなければならない。その後INTRAまたはEIタイプのピクチャが発生するか、タイムアウト期間が完了するかどちらかが先に生じるまで継続しなければならない。クロマキーの使用は、5秒後または5枚のピクチャ更新のいずれか長い時間のタイムアウトにより終了しなければならない。タイムアウト期間内に同一のクロマキー情報機能を発行することによってタイムアウトを防ぐことができる（例えば、発行から5秒以上の時間間隔を示すテンポラルリファレンスを持つ最初のピクチャのヘッダで、または発行後5枚目のピクチャのヘッダでクロマキー情報機能を繰り返す）。それぞれINTRAもしくはEIタイプのピクチャの発生時とそれぞれのタイムアウト間隔以内に行われる完全な再同期のために、符号器はクロマキー情報機能を含む十分な情報を送信しなければならない（符号器は蓄積された、またはデフォルトのキーカラー値やしきい値の使用に頼るべきでない）。

#### L. 15 拡張機能タイプ

拡張機能タイプ指示は引き続きPSUPPオクテットに拡張機能が含まれることを知らせるために使用される。拡張機能の使用は、より多くの後方互換性のあるPSUPPデータ機能を今後定義できるように予約されている。拡張機能タイプ指示のDSIZEは0でなければならない。拡張機能タイプ指示の将来の使用にあたり後方互換性を可能とするため、復号器は拡張機能タイプ指示に続くオクテット中の後半4ビットをDSIZE値として扱わな

なければならない。このときD S I Z E 値は、拡張機能パラメータデータのためにスキップされるべき後続のP S U P P のオクテット数を表し、その拡張機能パラメータデータに別のF T Y P E 指示が続いてもよい。

## 付属資料M

### 改良PBフレームモード

(この付属資料は本標準の必須部分である)

#### M. 1 序論

この付属資料はTTC標準JT-H263のオプションである改良PBフレームを規定する。付属資料Gに定義されている従来のPBフレームに代わり、改良PBフレームを採用する事により、より多くの改善がなされる様に考慮されている。TTC標準JT-H263の中のこのモードの能力は別の手段(例えばTTC標準JT-H245)で通知される。このモードの使用はピクチャヘッダの中のPLUSTYPEフィールドの中で通知される。

このオプションの大部分は、付属資料Gで規定されたPBフレームのオプションと同一である。

付属資料Oで規定されるBピクチャとの混同を避けるために、本付属資料ではBピクチャ、Bマクロブロック、Bブロックの言葉は使用しない。その代わりにB<sub>PB</sub>なる用語を改良PBフレームのB部分として用いる。付属資料Gを参照する場合は、Bピクチャ、Bブロックは、B<sub>PB</sub>ピクチャ、B<sub>PB</sub>ブロックと読み替えなければならない。

従来のPBフレームモードと改良PBフレームモードの主な違いは、改良PBフレームのB<sub>PB</sub>マクロブロックでは双方向予測に加えて、前方予測、後方予測モードが可能な事である。この付属資料では、MVD<sub>B</sub>(存在する場合には)は前方予測を意味する(付属資料Gでは、MVD<sub>B</sub>は単なる前方動きベクトルという意味ではなく、双方向予測のために重み付けされた前方又は後方ベクトルの補正という意味で使われていた)。

付属資料Gとの違いは全てこの付属資料に記述されている。記述されていない項目は全て、付属資料Gで記述されている処理過程が使われる事を意味する。

#### M. 2 B<sub>PB</sub>マクロブロック予測モード

B<sub>PB</sub>マクロブロックの符号化には3通りの異なる方法がある。どの符号化モードを用いるかはパラメータMOD Bにて指定される。B<sub>PB</sub>マクロブロックの符号化方法は以下の通りである。

##### M. 2. 1 双方向予測

双方向予測モードでは、B<sub>PB</sub>ピクチャの前後の参照ピクチャを用いて予測される(改良PBフレームのシーケンスでは、時間的に直前にある改良PBフレームのPピクチャ部と、現在の改良PBフレームのPピクチャ部の事である)。MVD=0の時にはこの予測は付属資料Gに規定されている予測と等価である。このモードでは(かつこのモードのみ)、PマクロブロックがINTRA符号化の場合でも、PBマクロブロック内に、動きベクトルデータ(MVD)が含まなければならないことに注意すべきである(MVD、即ち動きベクトルデータと、MVD<sub>B</sub>、即ち付属資料Gで規定されているデルタベクトルの違いに注意すること)。

##### M. 2. 2 前方予測

前方予測モードでは、MVD<sub>B</sub>に含まれるベクトル値は、直前の参照ピクチャ(INTRA、INTERピクチャ、またはPBや改良PBフレームのPピクチャ部分)からの前方予測に使われる。この事は、この予測モードでは、B<sub>PB</sub>マクロブロックに対して、16×16の動きベクトルが常にただ1個存在する事を意味する。

前方動きベクトルの符号化に対しては、単純な予測器が使われる。この予測器の規則は以下の通りである。現在のマクロブロックがピクチャやスライス左端では無く、かつ左側のマクロブロックが前方動きベクトルを持つ場合には、現在のマクロブロックの前方動きベクトル予測器には左側のブロックの前方動きベクトルが設定され、他の場合は0が設定される。予測器と所望の動きベクトルの差分はPピクチャで使われるベクトルデータ(MVD)と同一の方法で可変長符号化される。

D. 1節に規定されるピクチャ境界を越える動きベクトルに関しては、この特徴が採用されるならば、ここで記述された技術は前方B<sub>PB</sub>ベクトルにも適用される(この考え方は前方予測と双方向予測モードに適用される)。

##### M. 2. 3 後方予測

後方予測モードではB<sub>PB</sub>マクロブロックの予測はG. 5節で規定されたP<sub>REC</sub>と同一である。後方予測では、動きベクトルのデータは使用しない。



### M. 3 Bマクロブロックの双方向予測におけるベクトルの計算

双方向予測が採用される時、 $MV_D=0$  の場合は付属資料Gに記載された様に重み付けられた前方、後方ベクトルが計算される。

### M. 4 MODB表

MODBの新しい定義（表1 1 / JT-H 2 6 3を置換して）を付表M. 1 / JT-H 2 6 3に示す。この表はB<sub>PB</sub>ブロックに対して適用可能な符号化モードを示す。

付表M. 1 / JT-H 2 6 3 改良PBフレームモードにおけるMODB表  
(ITU-T H. 2 6 3)

インデックス	CBPB	MVDB	符号語長	符号語	符号化モード
0			1	0	双方向予測
1	X		2	10	双方向予測
2		X	3	110	前方予測
3	X	X	4	1110	前方予測
4			5	11110	後方予測
5	X		5	11111	後方予測

注：Xは関連するシンタックス要素が存在する事を示す。

## 付属資料N

### 参照ピクチャ選択モード

(この付属資料は本標準の必須部分である)

#### N. 1 序論

この付属資料は、TTC標準JT-H263のオプションである参照ピクチャ選択モードについて記述したもので、これは“NEWPREL”と呼ばれる改良されたフレーム間予測手法を使用して動作する。TTC標準JT-H263のこのモードの能力は、外部の手段により通知される(例えばTTC標準JT-H245)。符号器のメモリ管理を助けるために、復号器で設けられる付加ピクチャメモリの量も外部の手段により通知してよい。このモードは、復号器でどのピクチャのどの部分が正しく復号されているかを符号器に知らせるために、復号器から符号器へ送信される逆方向チャンネルメッセージを使用することができる。このモードの使用は、ピクチャヘッダのPLUSPTYPEフィールドに示される。このモードは2つの逆方向チャンネルモードスイッチを有しており、それらは、逆方向チャンネルが使用されるかどうか、および復号器からどのようなメッセージが逆方向チャンネルに返されるかを定義する。そして、このモードは逆方向チャンネルメッセージのためのチャンネルについて定義したもう一つのサブモードを有する。

このモードの2つの逆方向チャンネルモードスイッチは、逆方向チャンネル上で送られるメッセージのタイプを決定しており、そのメッセージのタイプはACK(確認メッセージ)かNACK(非確認メッセージ)のどちらが送られるのかを指定している。併せて、この2つのスイッチは4つの基本的な動作方法を定義する。

- (1) NEITHER: 復号器から符号器へ返される逆方向チャンネルデータがない
- (2) ACK: 復号器は確認メッセージのみを返す
- (3) NACK: 復号器は非確認メッセージのみを返す
- (4) ACK+NACK: 復号器は確認メッセージと非確認メッセージの両方を返す

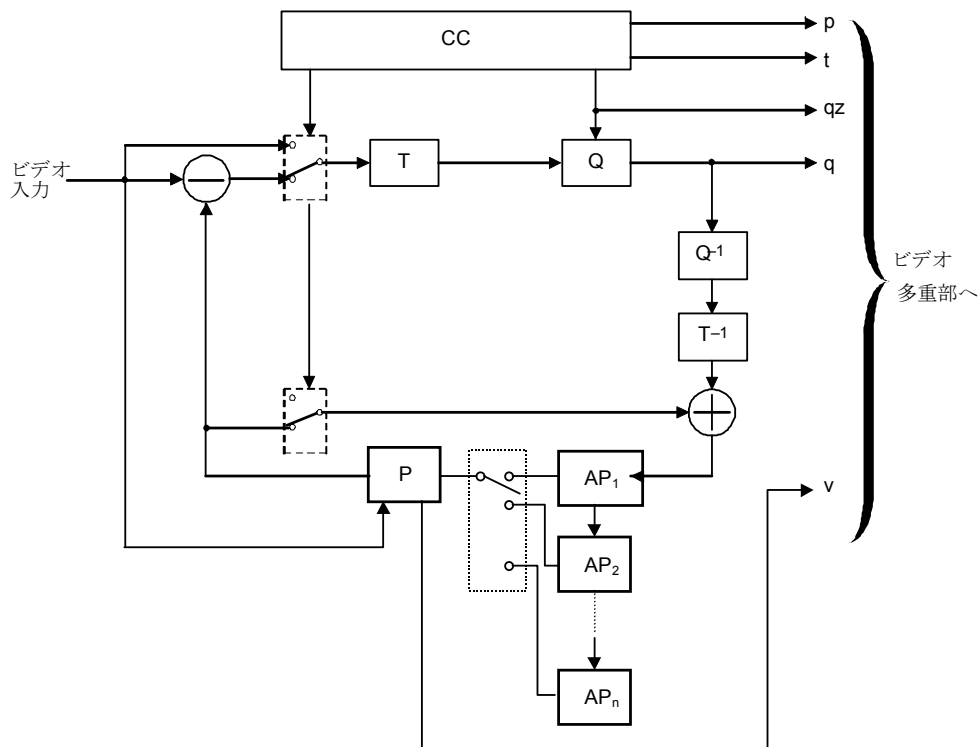
上記概要のように送られるメッセージの特定のタイプは、ピクチャヘッダの中で示される。また、逆方向チャンネルメッセージのためのチャンネルに関しては2つの動作方法がある。

- (1) 独立論理チャンネルモード: この動作方法は、システムの多重化レイヤの独立した論理チャンネルを通して逆方向チャンネルデータを伝達する。
- (2) ビデオ多重モード: この動作方法は、符号化されたビデオストリームの順方向ビデオデータ内で、受信されたビデオに対する逆方向チャンネルデータを伝達する。

この付属資料は、順方向チャンネルメッセージと逆方向チャンネルメッセージのシンタックスを定義する。

#### N. 2 ビデオ情報源符号化アルゴリズム

このモードの情報源符号器の一般的な構成を付図N.1/JT-H263に示す。この図は数個のピクチャメモリを使用する構成を示している。情報源符号器は、フレーム間予測符号化による時間的な誤り伝播を抑制するために、これらのピクチャメモリの一つを選択してもよい。GOBやスライスのようなピクチャよりも小さい単位にこのモードが適用される場合には、GOB境界またはスライスの境界をまたがった動き補償による誤り伝播を防ぐために、空でないヘッダを有するGOB境界またはスライスの境界をピクチャ境界として取り扱う独立セグメント復号モード(付属資料R参照)を使用することができる。予測のためにどのピクチャが選択されているかを知らせる情報は、符号化されたビット列内に含まれている。予測に使用するピクチャを選択するために符号器が用いる方法については、本標準の範囲外である。



- T 変換器
- Q 量子化器
- P 動き補償用可変遅延機能を持つピクチャメモリ
- AP 付加ピクチャメモリ
- CC 符号化制御
- p INTRA/INTER識別フラグ
- t 伝送/非伝送識別フラグ
- qz 量子化器表示
- q 変換係数の量子化出力インデックス
- v 動きベクトル

付図N. 1 / JT-H263 NEWPREDのための情報源符号器  
(ITU-T H. 263)

### N. 3 逆方向チャンネルメッセージ用のチャンネル

このモードは、逆方向チャンネルメッセージのためのチャンネルタイプについて、2つの動作方法を有する。ひとつは独立論理チャンネルモードであり、もうひとつはビデオ多重モードである。独立論理チャンネルモードは推奨されるモードであり、専用の論理チャンネルを通してN. 4. 2節で定義される逆方向チャンネルメッセージを伝える。ビデオ多重モードは、チャンネルの組み合わせ数の制約により逆方向チャンネルメッセージ用に独立した別チャンネルを備えることのできないシステムのために用意されている。ビデオ多重モードは反対方向にある順方向ビデオデータと同じ論理チャンネルを通して、逆方向チャンネルメッセージを伝える。

#### N. 3. 1 独立論理チャンネルモード

独立論理チャンネルモードは、逆方向チャンネルメッセージの目的にのみ開かれる専用の論理チャンネルを通して逆方向メッセージを伝える。ビデオデータを伝える順方向チャンネルと連携した機構は、外部の手段（例えばTTC標準JT-H245）により提供される。この中で定義された逆方向チャンネルシンタックスは同期フラグワードを含まないため、独立論理チャンネル動作は、逆方向チャンネル内のメッセージ同期のための外部フレーミング機構を必要とする。

#### N. 3. 2 ビデオ多重モード

ビデオ多重モードは、反対方向にある順方向ビデオデータと同じ論理チャンネルを通して、逆方向チャンネルメッセージを伝える。多重化されたビット列のシンタックスは、N. 4. 1小節で定義される。逆方向チャンネルメッセー

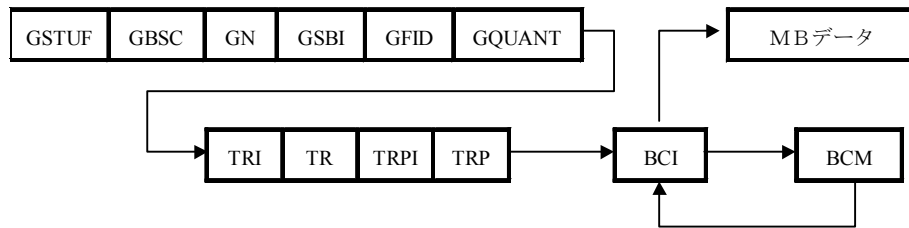
ジは、逆方向チャンネルメッセージインジケーション（BCI）を用いることによってGOBヘッダまたはスライスヘッダ内に挿入されてもよい。

## N. 4 シンタックス

### N. 4. 1 順方向チャンネル

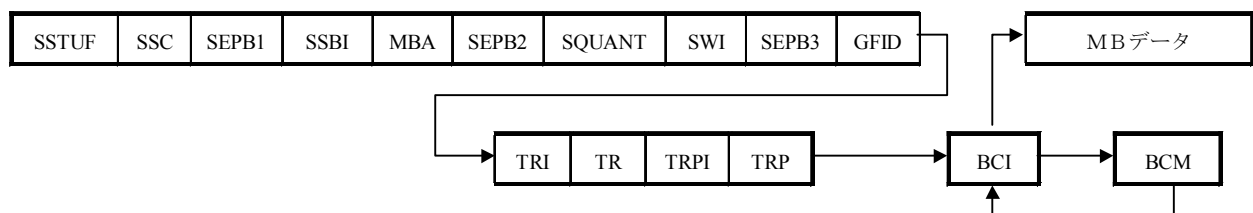
圧縮されたビデオ信号を伝送する順方向チャンネルデータのシンタックスは、グループオブブロック（GOB）レイヤまたはスライスレイヤの中だけで変更される。

GOBレイヤのシンタックスを付図N. 2 / JT-H263に示す。TRI、TR、TRPI、TRP、BCI、BCMのフィールドが図9 / JT-H263に追加される。



付図N. 2 / JT-H263 NEWPREDのためのGOBレイヤの構成  
(ITU-T H. 263)

オプションのスライス構造モード（付属資料K参照）を使っている場合、スライスレイヤのシンタックスはGOBレイヤと同じ方法で変更される。シンタックスを付図N. 3 / JT-H263に示す。



付図N. 3 / JT-H263 NEWPREDのためのスライスレイヤの構成  
(ITU-T H. 263)

#### N. 4. 1. 1 TRI（テンポラルリファレンスインジケータ）（1ビット）

TRIは、次に来るTRフィールドが存在するかどうかを示す。

- 0 : TRフィールドは存在しない
- 1 : TRフィールドは存在する

#### N. 4. 1. 2 テンポラルリファレンス（TR）（8 / 10ビット）

TRが存在するとき、カスタム画像クロック周波数がない場合は、TRは8ビットの数であり、使用されている場合は、ピクチャヘッダのETRとTRの結合からなる10ビットの数である。

#### N. 4. 1. 3 TRPI（予測用テンポラルリファレンスインジケータ）（1ビット）

TRPIは、次に来るTRPフィールドが存在するかどうかを示す。

- 0 : TRPフィールドは存在しない
- 1 : TRPフィールドは存在する

TRPIは、ピクチャがIピクチャまたはEIピクチャであればいつでも“0”でなければならない。

#### N. 4. 1. 4 予測用テンポラルリファレンス (TRP) (10ビット)

TRが存在するとき (TRPIで示されるが)、これはBピクチャおよび改良PBフレームのBピクチャ部の場合を除き、TRPは符号化時の予測に用いられるテンポラルリファレンスを示す。Bピクチャまたは改良PBフレームのBピクチャ部については、テンポラルリファレンスTRPを有するピクチャが、前方予測に用いられる。(逆時間方向の予測には常に、時間的にすぐ後のピクチャを使う。) TRPは10ビットの数である。参照ピクチャにカスタム画像クロック周波数が使用されていない場合は、TRPのMSB2ビットは“0”であり、LSBには参照ピクチャのピクチャヘッダ内にある8ビットのTRを含んでいる。参照ピクチャにカスタム画像クロック周波数が使用されている場合は、TRPは参照ピクチャヘッダからのETRとTRの結合からなる10ビットの数である。

参照ピクチャ選択モードでないときのように、TRPが存在しない場合は、時間的に前の最も最近のアンカーピクチャが予測に用いられなければならない。TRPは、次のPSC、GSCまたはSSCまで有効である。

#### N. 4. 1. 5 逆方向チャンネルメッセージインジケーション (BCI) (可変長)

このフィールドは1または2ビットを含んでおり、“1”が設定された場合は、次に来るビデオ逆方向チャンネルメッセージ (BCM) フィールドが存在することを示している。もしそうでなければ、フィールド値は“01”であり、ビデオ逆方向チャンネルメッセージフィールドが存在しないこと、あるいはその終わりであることを示している。BCMとBCIの組み合わせは存在しなくてもよいし、存在する場合には繰り返されてもよい。もしビデオ多重モードが使用されていないならば、BCIは常に“01”に設定されなければならない。

#### N. 4. 1. 6 逆方向チャンネルメッセージ (BCM) (可変長)

逆方向チャンネルメッセージはN. 4. 2小節で定義するシンタックスを有しており、前にあるBCIフィールドが“1”に設定されている場合のみ存在する。

#### N. 4. 2 逆方向チャンネルメッセージ (BCM) のシンタックス

確認/非確認メッセージを伝送する逆方向チャンネルのシンタックスは、付図N. 4/JT-H263に示される。このメッセージは、順方向チャンネルデータが正しく復号されたかどうかを伝えるために、復号器から符号器へ返される。

BT	URF	TR	ELNUMI	ELNUM	BCPM	BSBI	BEPB1	GN/MBA	BEPB2	RTR	BSTUF
----	-----	----	--------	-------	------	------	-------	--------	-------	-----	-------

付図N. 4/JT-H263 NEWPREDのための逆方向チャンネルメッセージ (ITU-T H. 263) (BCM) シンタックスの構成

#### N. 4. 2. 1 逆方向チャンネルメッセージタイプ (BT) (2ビット)

逆方向チャンネルメッセージタイプは、符号化されたメッセージの対応する部分が正しく復号されたかどうかを示している。符号器がどのメッセージタイプを要求しているのかは、順方向チャンネルのピクチャヘッダ内で示される。

- 00 : 将来のための予約
- 01 : 将来のための予約
- 10 : NACK 順方向チャンネルデータの対応する部分が、誤って復号されたことを示す
- 11 : ACK 順方向チャンネルデータの対応する部分が、正しく復号されたことを示す

#### N. 4. 2. 2 アンリライアブルフラグ (URF) (1ビット)

TRまたはGN/MBAとして信頼できる値を復号器が利用できない場合、アンリライアブルフラグは1に設定される。(BTがNACKの場合、信頼できるTRは復号器で利用できなくてもよい。)

- 0 : 信頼できる
- 1 : 信頼できない

#### N. 4. 2. 3 テンポラルリファレンス (TR) (10ビット)

テンポラルリファレンスは、逆方向チャンネルメッセージ内でACK/NACKが示されるビデオピクチャセグメントのTR情報を含んでいる。(注：ここで使用される“ビデオピクチャセグメント”という言葉の意味は、付属資料Rで定義されている。) 参照ピクチャにカスタム画像クロック周波数が使われていない場合は、TRのMSB 2ビットは“0”であり、LSBには参照ピクチャのピクチャヘッダ内にある8ビットのTRを含んでいる。参照ピクチャにカスタム画像クロック周波数が使われている場合は、TRは参照ピクチャヘッダからのETRとTRの結合からなる10ビットの数である。

#### N. 4. 2. 4 高品質化レイヤ番号インジケーション (ELNUM1) (1ビット)

オプションである時間、SNR、空間スケーラビリティモード(付属資料O参照)が順方向チャンネルデータで使用されており、かつ順方向チャンネルのいくつかの高品質化レイヤがひとつの論理チャンネル上で結合されており、かつ逆方向チャンネルメッセージが(基本レイヤではなく)高品質化レイヤを参照する場合、高品質化レイヤ番号指示は“1”でなければならず、そうでない場合には高品質化レイヤ番号指示は“0”である。

#### N. 4. 2. 5 高品質化レイヤ番号 (ELNUM) (4ビット)

高品質化レイヤ番号は、ELNUM1が“1”の場合にのみ存在し、この場合には逆方向チャンネルメッセージ内で参照される高品質化レイヤのレイヤ番号を含んでいる。

#### N. 4. 2. 6 BCPM (1ビット)

BCPMは、順方向チャンネルデータ内でCPMモードが使用されていない場合は“0”であり、使用されている場合は“1”である。もしBCPMが“1”であれば、BSBIが存在することを示している。

#### N. 4. 2. 7 逆方向チャンネルサブビット列インジケータ (BSBI) (2ビット)

BCPMが“1”の場合にのみ存在する2ビットの固定長符号である。BSBIは5.2.4小節および付属資料Cで記述されているように、逆方向チャンネルメッセージ内でACK/NACKメッセージが示される順方向チャンネルデータにおける適切なサブビット列番号の自然2進数表現である。

#### N. 4. 2. 8 逆方向チャンネルエミュレーション防止ビット1 (BEPB1) (1ビット)

ビデオ多重モードが使用されている場合のみ存在するフィールドである。スタートコードエミュレーションを防ぐため、このフィールドは常に“1”が設定される。

#### N. 4. 2. 9 GOB番号/マクロブロックアドレス (GN/MBA) (5/6/7/9/11/12/13/14ビット)

このフィールドには、GOB番号またはマクロブロックアドレスが存在する。オプションのスライス構造モード(付属資料K参照)が使用されていない場合、このフィールドには、逆方向チャンネルメッセージ内でNACK/ACKメッセージが示されたビデオピクチャセグメントの先頭のGOB番号を含んでいる。オプションのスライス構造モードが使用されている場合、このフィールドには、逆方向チャンネルメッセージ内でNACK/ACKメッセージが示されたスライスの開始のマクロブロックアドレスを含んでいる。このフィールドの長さは、GNまたはMBAに関して本標準内の他の箇所定義した長さとなる。

注：このフィールドがビデオ多重モード内で受信された時、オプションのスライス構造モードの使用とは、BCMが適用されるビデオビット列内でのそのモードの使用をいうのであって、BCMデータを伝送するビデオビット列内のことではない。

#### N. 4. 2. 10 逆方向チャンネルエミュレーション防止ビット2 (BEPB2) (1ビット)

ビデオ多重モードが使用されている場合のみ存在するフィールドである。スタートコードエミュレーションを防ぐため、このフィールドは常に“1”が設定される。

#### N. 4. 2. 11 要求テンポラルリファレンス (RTR) (10ビット)

要求テンポラルリファレンスは、BTがNACKの場合にのみ存在する。RTRは、NACKに対応したGOBまたはスライスの要求テンポラルリファレンスを示している。通常それは、復号器におけるNACKに対応する位置での最後に正しく復号されたビデオピクチャセグメントのTRである。要求された参照ピクチャにカスタム画像クロック周波数が使われていない場合は、RTRのMSB 2ビットは“0”であり、LSBには要求された参照ピクチャのピクチャヘッダ内にある8ビットのTRを含んでいる。要求された参照ピクチャにおいてカスタム画像クロック周波数が使われている場合は、RTRは要求された参照ピクチャヘッダからのETRとTRの結合からなる10ビットの数である。

#### N. 4. 2. 12 スタッフ化 (BSTUF) (可変長)

このフィールドは、独立論理チャンネルモードが使用されており、逆方向チャンネルメッセージが外部フレームの最後である場合にのみ存在する。BSTUFは、ゼロビット以上の値“0”からなる可変長の符号である。このフィールドは、外部フレームの最後にのみ存在する。

### N. 5 復号処理

このモードの復号器は、正しく復号されたビデオ信号とそれらのテンポラルリファレンス (TR) 情報を蓄積するために、追加のピクチャメモリを必要としてもよい。順方向チャンネルデータにTRPフィールドが存在する場合、復号器はフレーム間復号における参照ピクチャとして、最新の復号ピクチャの代わりに、TRがTRPである蓄積されたピクチャを使用する。TRがTRPであるピクチャが復号器で使えない場合、復号器は強制INTRA更新信号を外部の手段 (例えばTTC標準JTH245) によって符号器に送信してもよい。もしそれと異なるピクチャ蓄積方法が外部の手段によって交渉されない場合には、正しく復号されたビデオピクチャセグメントはそれ以後参照ピクチャとして使用するために、付図N. 1 / JTH263に示すようなファーストインファーストアウトの原理でメモリに蓄積されなければならない (ただし参照ピクチャとして使われないBピクチャは除く)。そして、正しく復号されていないと検知されたビデオピクチャセグメントは、このメモリ領域にある正しく復号されたビデオピクチャセグメントと、置き換えられるべきではない。

確認メッセージ (ACK) と非確認メッセージ (NACK) は、逆方向チャンネルメッセージとして定義される。復号器がビデオピクチャセグメントの復号に成功した場合にはACKを返してもよい。復号器がビデオピクチャセグメントの復号に失敗した場合にはNACKを返してもよく、要求したTRPまたはINTRA更新を含む期待する順方向チャンネルデータを復号器が受信するまでNACKを返し続けてもよい。どのタイプのメッセージを送るべきかは、順方向チャンネルデータのピクチャヘッダのRPSMFフィールドに示される。

“ビデオ冗長符号化”として知られている使用シナリオでは、(通常は異なる参照ピクチャを使用して) 時間的に同じ瞬間の画像シーンに対して1つ以上の表現を送るという方法で、ある符号器は参照ピクチャ選択モードを使用してもよい。参照ピクチャ選択モードが使用されており、ビット列で隣接したピクチャが同じテンポラルリファレンスを持つ場合には、復号器はこのことを、ほぼ同じ画像シーンの内容の冗長である複写が送られたことを示していると見なさなければならず、後に続く冗長であるピクチャを捨て、このような最初に受信したピクチャを復号し使用しなければならない。

## 付属資料O

### 時間、SNR、空間スケーラビリティモード

(この付属資料は本標準の必須部分である)

本付属資料は時間、SNR及び空間スケーラビリティをサポートするTTC標準JT-H263のオプションモードについて記述する。本モードは誤り制御の仕組みと関連しても使われてもよい。本モードの能力及びそのモードの特徴がサポートされる範囲は、(例えばTTC標準JT-H245のような)外部の手段で通知される。このモードの使用方法はPLUSPTYPEの中で示される。

#### O. 1 序論

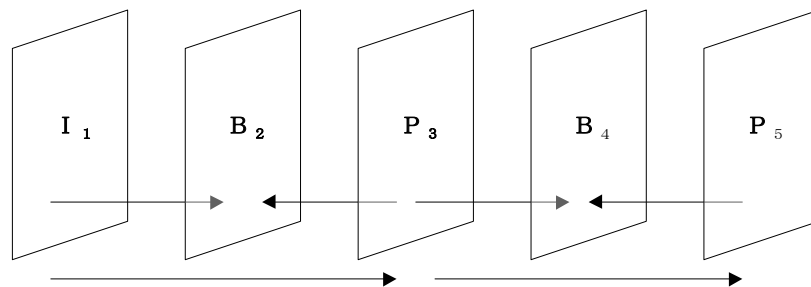
スケーラビリティは一つ以上の品質レベルにおけるシーケンスの復号を許す。これは、ピクチャの階層及び一つ以上のレイヤに分割された高品質化ピクチャを用いることにより行われる。スケーラビリティに使用されるピクチャには3つのタイプがある、すなわち以下で説明するB、EI及びEPピクチャである。これらのおおのほは、どのレイヤに属するかを示す高品質化レイヤ番号ELNUM及び、その予測にどのレイヤが使われるかを示す参照レイヤ番号RLNUMを持つ。最も低いレイヤは基本レイヤと呼ばれ、レイヤ番号1を持つ。

スケーラビリティは3つの基本的な方法により実現される。すなわち、時間、SNR及び空間高画質化である。

##### O. 1. 1 時間スケーラビリティ

時間スケーラビリティは双方向予測ピクチャつまりBピクチャを使用することにより実現される。Bピクチャは、参照レイヤにおける、前と、その次の再生ピクチャのどちらかあるいは両方からの予測が許される。この特性は一般的にPピクチャと比べて圧縮効率が向上する。ビット列において別々のものであるという意味で、これらのBピクチャはPB(あるいは改良PB)フレーム(付属資料G及びM参照)のBピクチャ部分とは異なる。これらはシンタックスにおいては次のP(あるいはEP)ピクチャと混合されない。

Bピクチャ(及びPBまたは改良PBフレームのBの部分)は、他のピクチャを予測するための参照ピクチャとして使用されない。この特性は、必要ならば次のピクチャに悪影響を与えることなくBピクチャが廃棄されることを許し、このように時間スケーラビリティを提供している。付図O. 1/JT-H263はP及びBピクチャの予測構造を説明している。



付図O. 1/JT-H263 Bピクチャ予測依存の説明  
(ITU-T H. 263)

ビット列中のBピクチャの位置は、厳密な時間的順序よりもむしろデータに依存する順序に従う。(この規則は、ビット列中における他のピクチャの並びとは矛盾しないが、しかしBピクチャではないすべてのピクチャタイプにとっては、データに依存する順序と時間的順序の間にはそのような矛盾は起こらない。)例えば、もしビデオシーケンスのピクチャに1, 2, 3, ... という番号がふられると、符号化されたビット列の順序はI<sub>1</sub>, P<sub>3</sub>, B<sub>2</sub>, P<sub>5</sub>, B<sub>4</sub>, ...となる。ここで、添え字は付図O. 1/JT-H263で説明した元のピクチャの番号を示す。

参照レイヤにおいて参照ピクチャの対の間に挿入されるBピクチャの数に制限はない(ピクチャヘッダ内のテンポラルリファレンスフィールドのオーバーフローによる時間的な不明確さが生じることを防ぐ必要性を除き)。しかしそのようなピクチャの最大数を、(例えばTTC標準JT-H245の)外部の手段により通知してもよい。

Bピクチャのピクチャの高さ、幅及び画素アスペクト比は、時間的に次の参照レイヤのピクチャと常に等しくならなければならない。



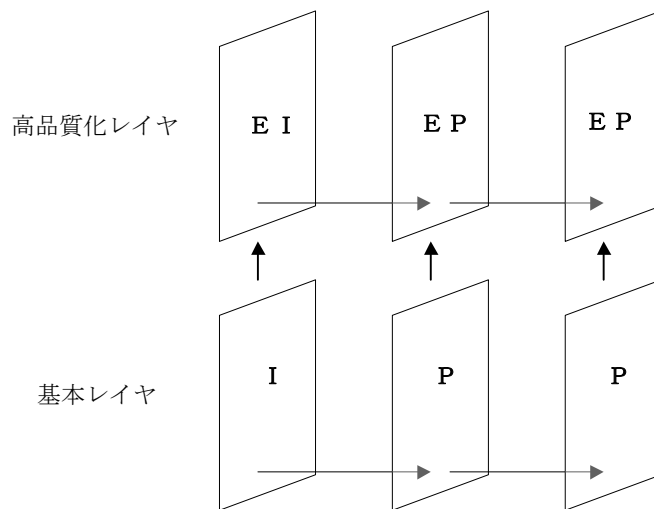
動きベクトルはBピクチャのピクチャの境界を超えることを許している。

### 0. 1. 2 SNRスケーラビリティ

スケーラビリティを実現する他の基本的方法は、空間／SNR高画質化を通して行われる。空間スケーラビリティとSNRスケーラビリティは、簡単に述べるが補間の使い方を除いては等価である。圧縮は人工的雑音と歪みをもたらすため、符号器での再生ピクチャと入力された元のピクチャとの差分は（ほとんど常に）零でない値のピクチャとなり、符号化誤りと呼ばれるものを含んでいる。通常、この符号化誤りは符号器で失われ、決して回復はできない。SNRスケーラビリティを用いれば、これらの符号化誤りのあるピクチャを符号化でき、復号器に送ることができ、復号したピクチャを高画質化することができる。余分なデータはピクチャのSN比を上げるために役立つ。それゆえ、これをSNRスケーラビリティと呼ぶ。付図O. 2/JT-H263はSNRスケーラビリティにおけるデータの流れを説明する。下位レイヤからの垂直の矢印は、高品質化レイヤにおけるピクチャが参照（下位）レイヤにおいて再生された近似ピクチャから予測されることを説明している。

もし予測が下位レイヤのみから行われれば、高品質化レイヤのピクチャはEIピクチャと呼ばれる。一方、先の高品質化レイヤピクチャと時間的に同時の下位レイヤの参照ピクチャの両方を用いる修正双方向予測ピクチャを生成することが可能である。このピクチャのタイプはEPピクチャあるいは拡張Pピクチャと呼ばれる。EI及びEPピクチャの予測の流れを付図O. 2/JT-H263に示す。（付図O. 2/JT-H263には特に示していないが、高品質化レイヤにおけるEIピクチャは下位レイヤ参照ピクチャとしてのPピクチャを持ってもよく、EPピクチャは下位レイヤ参照ピクチャとしてのIピクチャを持ってもよい。）

EI及びEPピクチャ共に、参照レイヤからの予測で動きベクトルを使わない。しかし、普通のPピクチャのように、EPピクチャは同じレイヤにおける時間的に前の参照ピクチャからの予測では動きベクトルを使う。



付図O. 2/JT-H263 SNRスケーラビリティの説明  
(ITU-T H. 263)

### 0. 1. 3 空間スケーラビリティ

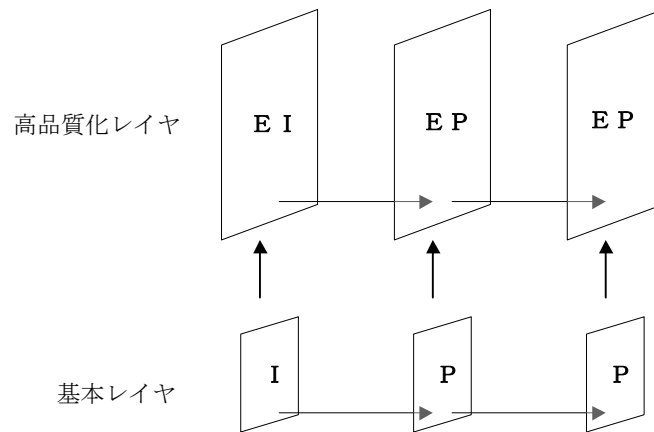
時間、SNR及び空間スケーラビリティモードにおいて、3番目そして最後のスケーラビリティ手法は空間スケーラビリティであり、それはSNRスケーラビリティとかなり関連する。唯一の違いは、空間高品質化レイヤにおけるピクチャを予測するために参照レイヤのピクチャが使われる前に、水平または垂直のどちらか（1次元の空間スケーラビリティ）あるいは水平及び垂直両方（2次元の空間スケーラビリティ）に係数2で内挿補間されることである。この操作のための内挿補間フィルタはO. 6節で定義される。空間スケーラビリティのいくつかの形態を可能とする復号器は、カスタム画像フォーマットを使用できる能力が必要であるかもしれない。例えば、もし基本レイヤがサブQCIF（128×96）の場合、2次元の空間高品質化レイヤピクチャは256×192となり、これは標準画像フォーマットと一致しないであろう。他の例は、基本レイヤが12:11の標準画素アスペクト比を持つQCIF（176×144）の場合であろう。このとき1次元の空間高品質化レイヤは、24:11の画素アスペクト比を持つ352×144の画像フォーマットを持つであろう。このように、これらの場合においては高品質化レイヤでカスタム画像フォーマットが使用されるべきであろう。カスタム画像フォーマットを必要としない

例は、C I Fの二次元空間高品質化レイヤをもつQ C I Fの基本レイヤを使う場合であろう。空間スケーラビリティは付図O. 3/J T-H 2 6 3で説明される。

符号化処理のための参照として使う前に、参照レイヤのピクチャサイズを大きくするためのアップサンプリング処理を必要とする以外は、空間スケーラビリティピクチャの処理とシンタックスは、S N Rスケーラビリティピクチャのそれと機能的には同一である。

S N Rスケーラビリティを使用したピクチャと、空間スケーラビリティを使用したピクチャでは、シンタックスでの相違点は非常に小さいため、両方の目的で使われるピクチャはE I、E Pピクチャと呼ばれる。

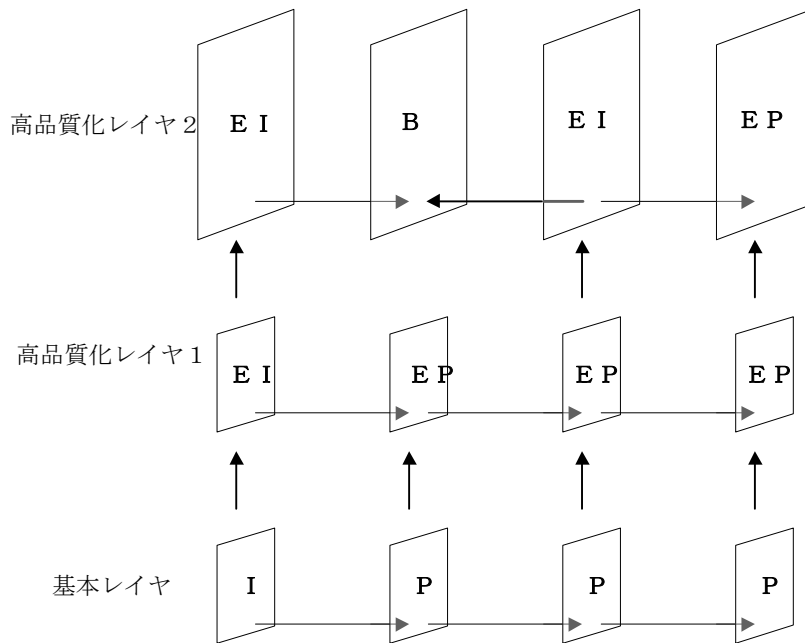
E IまたはE Pピクチャの上方予測のために使われる基本レイヤにおけるピクチャは、Iピクチャ、Pピクチャ、P Bまたは改良P BフレームのP部分であってもよい（しかし、Bピクチャ、P Bまたは改良P BフレームのB部分であってはならない）。



付図O. 3/J T-H 2 6 3 空間スケーラビリティの説明  
( I T U-T H. 2 6 3 )

#### O. 1. 4 多階層スケラビリティ

S N Rまたは空間高品質化ピクチャのいずれかで構成する場合であろうと、BピクチャはI、P、P B及び改良P Bのピクチャタイプの間だけでなく、E I、E Pのピクチャタイプの間にも時間的に挿入することができる。また、基本レイヤと関連する2つ以上のS N Rあるいは空間高品質化レイヤを持つこともできる。このように、多階層スケラブルビット列はS N Rレイヤ、空間レイヤ、及びBピクチャの組み合わせとなり得る。しかし、レイヤ番号が増加してもピクチャサイズを小さくすることはできない。それは、ひとつの次元かあるいは両方の次元における係数2により、ピクチャサイズは同じであるかあるいは増加する。付図O. 4/J T-H 2 6 3は多階層スケラブルビット列を説明している。



付図O. 4 / JT-H 263 多階層スケーラビリティの説明  
(ITU-T H. 263)

多階層スケーラビリティの場合、E IあるいはE Pピクチャの上方予測に使われる参照レイヤにおけるピクチャはI、P、E IまたはE Pピクチャであるか、または基本レイヤにおけるPBまたは改良PBフレームのP部分でもよい（しかし、Bピクチャ、PBまたは改良PBフレームのB部分であってはならない）。

二つのレイヤの場合、Bピクチャはどこのレイヤに現れてもよい。しかし、参照レイヤにおけるBピクチャと同時の高品質化レイヤにおけるピクチャはBピクチャ、PBまたは改良PBフレームのBピクチャ部分でなければならない。これはBピクチャは廃棄可能である性質を保存する。しかし、Bピクチャが下位レイヤにおいて関係するピクチャを持たないレイヤにおいても発生してもよいことに注意する必要がある。これは、符号器に、下位レイヤよりも高いピクチャレートの高品質化ピクチャを送ることを可能とする。

各高画質化ピクチャ（B、E IあるいはE P）の高品質化レイヤ番号と参照レイヤ番号は、（もしあれば）ピクチャヘッダのELNUM及びRLNUMフィールドでおのおの指定される。これらのフィールドがない場合には、5. 1. 4. 4 小節で述べられた推測規則を参照すること。もし、時間的に囲まれたSNR又は空間スケーラビリティピクチャが現れる高品質化レイヤにBピクチャが現れる場合、Bピクチャの参照レイヤ番号（RLNUM）は高品質化レイヤ番号（ELNUM）と同じでなければならない。

Bピクチャのピクチャの高さ、幅及び画素アスペクト比は常に時間的に続く参照レイヤピクチャのものと常に同じでなければならない。

## O. 2 ピクチャの伝送順序

他のピクチャに依存するピクチャは、依存するピクチャの後のビット列の中に位置しなければならない。

ビット列シンタックスの順序は、参照ピクチャ（例えば、I、P、E I、E P、PBまたは改良PBのP部分のタイプを持つピクチャになる）が次の二つの規則に従わなければならないように規定される。

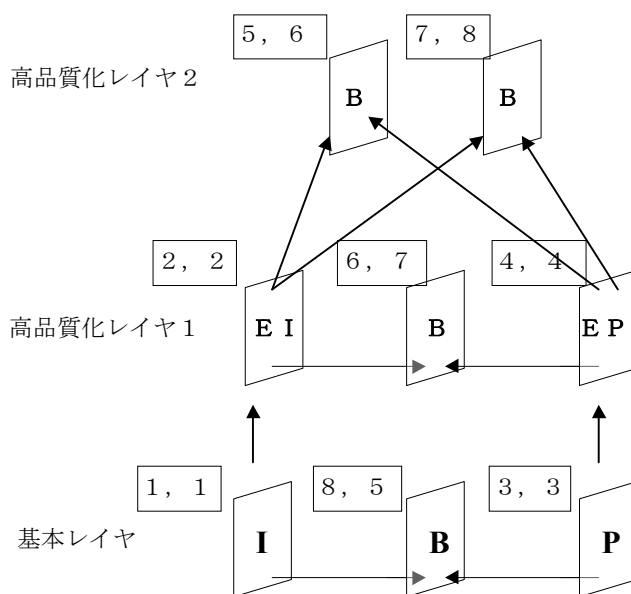
- (1) 同じテンポラルリファレンスを持つすべての参照ピクチャは、高品質化レイヤの下位レイヤから順に（おのおのの下位レイヤ参照ピクチャは、上位レイヤ参照ピクチャを復号するために必要であるため）現れなければならない。
- (2) 前1項で述べたすべての時間的に同時に起こる参照ピクチャは、（Bピクチャの参照として必要とされるすべての参照ピクチャの復号遅延を減らすため）これらのいかなる参照ピクチャが、Bピクチャのある参照レイヤにおける時間的に最初に次に続く参照ピクチャである、いかなるBピクチャにも先行して、ビット列の中に現れなければならない。

それゆえ、より早いテンポラルリファレンスを持つBピクチャが続かなければならない（おのおの高品質化レイヤの中で時間的順序で）。

各Bピクチャのビット列での位置は以下の規則に従わなければならない。

- (1) (Bピクチャの復号は一般的にその参照レイヤでの先行する復号に依存するため) Bピクチャのビット列の位置は、参照レイヤの中で時間的に最初に続く参照ピクチャの後でなければならない。
- (2) (Bピクチャの参照に必要なすべての参照ピクチャの復号遅延を減らすため) Bピクチャのビット列の位置は、参照レイヤの中の時間的に最初に続く参照ピクチャと時間的に同時のすべての参照ピクチャの後でなければならない。
- (3) Bピクチャのビット列の位置は、その参照レイヤの中のBピクチャ以外に付随する時間的に続くいかなるピクチャの位置に先行しなければならない(そうしなければ、参照レイヤのピクチャのために必要なピクチャ蓄積用メモリを増やすため)。
- (4) Bピクチャのビット列の位置は、時間的に最初に続く参照ピクチャと時間的に同時であるすべてのE I及びE Pピクチャの後でなければならない。
- (5) Bピクチャのビット列の位置は、同じ高品質化レイヤ内のすべての時間的に続くピクチャの位置よりも先行しなければならない(そうしなければ、不必要な遅延を招き、高品質化レイヤのために必要なピクチャ蓄積用メモリを増やすため)。

付図O. 5/J T-H 2 6 3は、そこで示された階層化構造のための上記の規則によって与えられる2つの可能なピクチャ伝送順序を示す(点線の箱の番号はビット列の順序を示し、2つの選択がある場合はカンマで区切っている)。



付図O. 5/J T-H 2 6 3 ピクチャ伝送順序の例  
( I T U - T H . 2 6 3 )

### O. 3 ピクチャレイヤのシンタックス

高品質化レイヤ番号 (E L N U M、5. 1. 1 1 小節参照) は、いかなる高品質化ピクチャ (B、E I 又はE P) の中でも常に現われ、I、Pピクチャ、又はP B、改良P Bフレームの中には現れてはならない。5. 1. 1 2 小節に示すように、参照レイヤ番号 (R L N U M、5. 1. 1 2 小節参照) はある高品質化ピクチャに現われ、他の高画質化ピクチャに対しては推測される。

明らかに一つの基本レイヤがあり、そのE L N U M、R L N U Mは1である。R L N U Mは、E I 及びE Pピクチャに対しては上方参照ピクチャの高画質化レイヤ番号を、Bピクチャに対しては前方及び後方参照ピクチャの高

品質化レイヤ番号を与える。基本レイヤの参照ピクチャは、ピクチャヘッダのなかにELNUM又はRLNUM(これらの値は暗黙的に1である)を含まないI、PB、改良PB及びPピクチャから構成されてもよい。

BピクチャのRLNUMは、ELNUM以下でなければならない。一方、EI及びEPピクチャのRLNUMは、ELNUMより小さくならない。

ELNUMは、システムレベルで使われるレイヤ番号とは異なってもよい。Bピクチャは、それら自身に依存する他のピクチャを持たないため、それらはTTC標準JT-H263以外のシステム構成要素(例えばTTC標準JT-H245及びTTC標準JT-H223)による別の高画質化レイヤに委ねられてもよい。さらに、高画質化ピクチャが別の映像チャンネルで送られるか基本レイヤピクチャと一緒に多重化されたままで送られるかは、実装者による。

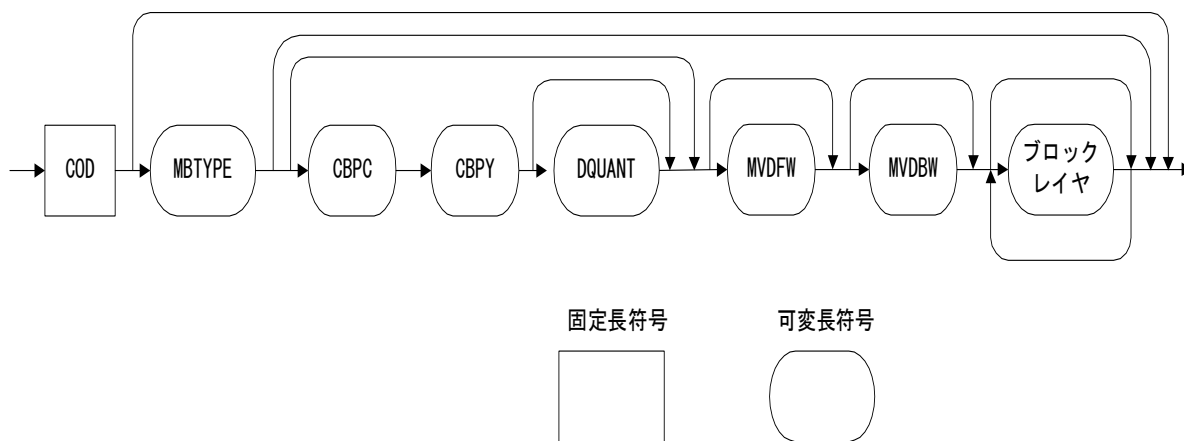
5.1.4.5小節で述べたように、デブロッキングフィルタモード(付属資料J参照)はBピクチャにおいては適用されない。これは、Bピクチャが他のどのピクチャの予測にも使われず、そしてこのように、これらピクチャのデブロッキングフィルタの適用は、本標準の規定範囲外の完全にポストプロセッシング技術に関するものであるためである。しかし、Bピクチャにあるタイプのデブロッキングフィルタを使用することは奨励され、そのような使用目的では付属資料Jで述べられるようなフィルタは、実際良い設計となるかもしれない。

テンポラルリファレンス(TR)(5.1.11小節参照)はI及びPピクチャのために厳密に規定される。

B、EI、EPピクチャのピクチャヘッダの中には、TR<sub>B</sub>(5.1.22小節参照)又はDBQUANT(5.1.23小節参照)フィールドは存在してはならない。

#### ○.4 マクロブロックレイヤのシンタックス

B及びEPピクチャのマクロブロックレイヤのシンタックスは、おのおのが類似する方法によりふたつの参照ピクチャを使うため、同じである。しかし、その解釈はピクチャタイプにより少し異なる。付図○.6/JT-H263はB及びEPのシンタックスを示している。MBTYPEフィールドは、直接モード予測、前方予測、後方/上方予測又は双方向予測のいずれがあるかを示している。MBTYPEは以下に示すように、B及びEPピクチャで異なる定義がされる。



付図○.6/JT-H263 EP及びBピクチャのマクロブロックシンタックス (ITU-T H.263)

直接予測モードはBピクチャについてのみ利用できる。それは、改良PBフレームモード(付属資料M)における双方向予測モードに似た双方向予測モードである。唯一の違いは、完全な後方予測ピクチャは復号器において既知であるため、どの画素が後方予測されるかについて制約がないことである。双方向予測モードは、前方及び後方予測の別々の動きベクトルを使用する。直接及び双方向モードにおいて、予測画素値は前方及び後方予測画素の平均をとることにより計算される。平均は二つの予測の和を2で割ること(端数を切り捨てによる割り算)により算出される。直接モードにおいては、参照マクロブロックにおいて4つの動きベクトルがある場合、4つのすべての動きベクトルは、改良PBフレームモード(付属資料M)においてと同じように使われる。

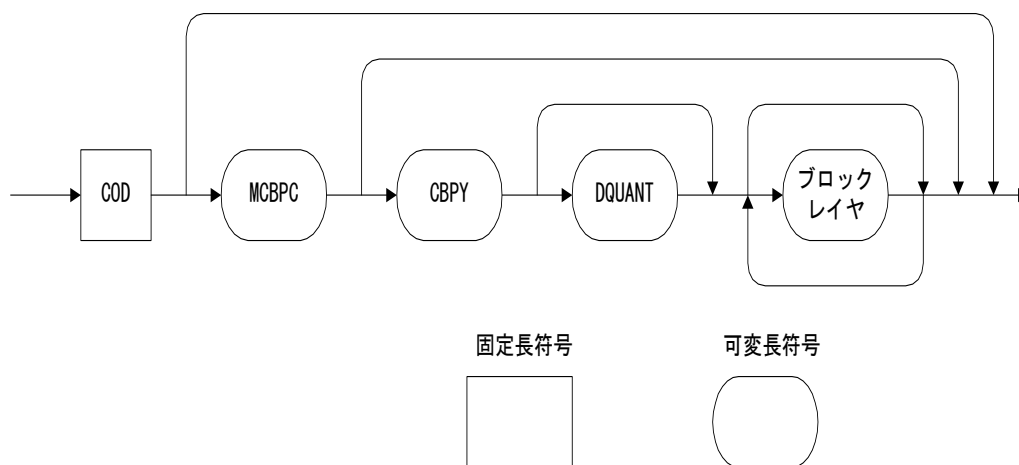
Bピクチャにとって、前方予測は参照レイヤの前の参照ピクチャからの予測であることを意味する。後方予測は参照レイヤの時間的に後の参照ピクチャからの予測であることを意味する。

E Pピクチャにとって、前方予測は同じ参照レイヤの前のE I又はE Pピクチャからの予測であることを意味する。一方、上方予測は、参照レイヤにおいて、時間的に同時である（おそらく補間された）参照ピクチャからの予測を意味する。前方予測のために動きベクトルは使用されるが、上方予測（シンタックスにおいてはBピクチャの後方予測と同じ位置にある）では、動きベクトルは使用されない。

E Iピクチャにおけるマクロブロックのシンタックスは少し異なっている。付図O. 7/J T-H 2 6 3に示すようにM B T Y P E及びC B P CはM C B P Cフィールドに結合される。前方予測は使用されない。参照レイヤにおいて、時間的に同時の参照ピクチャからの上方予測のみ使われる。動きベクトルは使われない。

B及びE Pピクチャでは、（もし無制限動きベクトルモードもまた使われるならば、D. 2節で記述されるように、拡張動きベクトル範囲のみが有効となるが）付属資料DのD. 1節に記述されるように、ピクチャの境界を越える動きベクトルが使われてもよい。

符号器は、直接予測モードを用いて予測されたものを含む全てのマクロブロックに対して、D. 1. 1小節への適合を保証しなければならない。すなわち、直接予測モードは、直接モード予測処理により推測された動きベクトル値が符号化ピクチャ領域外に15画素より大きい水平または垂直距離をもつ16×16（または8×8）予測領域の要素をひとつも生じないのでない限り、符号器により選択されてはならない。



付図O. 7/J T-H 2 6 3 E Iピクチャにおけるマクロブロックシンタックス  
( I T U - T H . 2 6 3 )

#### O. 4. 1 符号化されたマクロブロック指示 (COD) (1ビット)

ビットが“0”ならばマクロブロックは符号化されている。もしそれが“1”ならばマクロブロックのためのその他の情報は伝送されず、マクロブロックは下に示すように、スキップしたものと扱われる。

#### O. 4. 2 M B T Y P E / M C B P C (可変長符号)

B及びE Pピクチャには異なるM B T Y P Eの表がある。E Iピクチャには、代わりにM C B P Cの表がある。付表O. 1/J T-H 2 6 3は、BピクチャのためのM B T Y P Eの表である。付表O. 2/J T-H 2 6 3は、E PピクチャのためのM B T Y P Eの表である。付表O. 3/J T-H 2 6 3は、E IピクチャのM C B P Cの表である。

Bピクチャでは、“直接 (スキップ)” 予測タイプは、そのマクロブロックではM B T Y P Eもどのデータも伝送されず、復号器が前方及び後方動きベクトル及び対応する双方向予測を導出することを示している。これは、C O Dビットにより通知される。Bピクチャの“前方 (テクスチャなし)”、“後方 (テクスチャなし)”及び“双方向 (テクスチャなし)” 予測タイプは、係数がない前方、後方及び双方向予測を示し、また前方、後方予測では一つの動きベクトルが伝送され、双方向予測では二つの動きベクトルが伝送されることを示す。

E Pピクチャでは、“直接 (スキップ)” 予測タイプは、マクロブロックのための付加情報が伝送されないことを示し、よって復号器はゼロ動きベクトルを持ち係数なしの前方予測を使うべきである。E Pピクチャでは、“上方 (テクスチャなし)”及び“双方向 (テクスチャなし)” 予測タイプは、ゼロ動きベクトルを持ち係数なしの上方及び双方向予測を示す。

E Iピクチャでは、“上方 (スキップ)” 予測タイプはマクロブロックのために付加情報を送らないことを示し、したがって復号器はゼロ動きベクトルを持ち係数なしの上方予測を使うべきである。

付表O. 1 / JT-H263 BピクチャのMBTYPE可変長符号  
(ITU-T H. 263)

インデックス	予測タイプ	MVDFW	MVDBW	CBPC+ CBPY	DQUANT	MBTYPE	ビット数
—	直接(スキップ)					(COD=1)	0
0	直接			X		11	2
1	直接 + Q			X	X	0001	4
2	前方(テクスチャなし)	X				100	3
3	前方	X		X		101	3
4	前方 + Q	X		X	X	0011 0	5
5	後方(テクスチャなし)		X			010	3
6	後方		X	X		011	3
7	後方 + Q		X	X	X	0011 1	5
8	双方向(テクスチャなし)	X	X			0010 0	5
9	双方向	X	X	X		0010 1	5
10	双方向 + Q	X	X	X	X	0000 1	5
11	I NTRA			X		0000 01	6
12	I NTRA + Q			X	X	0000 001	7
13	スタッフ化					0000 0000 1	9

付表O. 2 / JT-H263 EPピクチャのMBTYPE可変長符号  
(ITU-T H. 263)

インデックス	予測タイプ	MVDFW	MVDBW	CBPC+ CBPY	DQUANT	MBTYPE	ビット数
—	前方(スキップ)					(COD=1)	0
0	前方	X		X		1	1
1	前方 + Q	X		X	X	001	3
2	上方(テクスチャなし)					010	3
3	上方			X		011	3
4	上方 + Q			X	X	0000 1	5
5	双方向(テクスチャなし)					0001 0	5
6	双方向	X		X		0001 1	5
7	双方向 + Q	X		X	X	0000 01	6
8	I NTRA			X		0000 001	7
9	I NTRA + Q			X	X	0000 0001	8
10	スタッフ化					0000 0000 1	9

付表O. 3 / JT-H263 EIピクチャのMVBPC可変長符号  
(ITU-T H. 263)

インデックス	予測タイプ	符号化ブロックパターン(56)	CBPY	DQUANT	MVBPC	ビット数
—	上方(スキップ)				(COD = 1)	0
0	上方	00	X		1	1
1	上方	01	X		001	3
2	上方	10	X		010	3
3	上方	11	X		011	3
4	上方 + Q	00	X	X	0001	4
5	上方 + Q	01	X	X	0000 001	7
6	上方 + Q	10	X	X	0000 010	7
7	上方 + Q	11	X	X	0000 011	7
8	INTRA	00	X		0000 0001	8
9	INTRA	01	X		0000 1001	8
10	INTRA	10	X		0000 1010	8
11	INTRA	11	X		0000 1011	8
12	INTRA + Q	00	X	X	0000 1100	8
13	INTRA + Q	01	X	X	0000 1101	8
14	INTRA + Q	10	X	X	0000 1110	8
15	INTRA + Q	11	X	X	0000 1111	8
16	スタッフ化				0000 0000 1	9

#### O. 4.3 色差の符号化ブロックパターン (CBPC) (可変長)

CBPCがある時、それは付表O. 4 / JT-H263で記述されるような色差ブロックの符号化ブロックパターンを示す。CBPCは、その存在がMBTYPE (付表O. 1 / JT-H263及びO. 2 / JT-H263参照)により示される場合、EP及びBピクチャでのみ現れる。

付表O. 4 / JT-H263 CBPC可変長符号  
(ITU-T H. 263)

インデックス	符号化ブロックパターン(56)	CBPC	ビット数
0	00	0	1
1	01	10	2
2	10	111	3
3	11	110	3

#### O. 4.4 輝度の符号化ブロックパターン (CBPY) (可変長)

CBPYはもしあれば、それはマクロブロックの輝度部分におけるブロックが存在することを示す。CBPYはMBTYPE (付表O. 1、O. 2及びO. 3 / JT-H263参照)によりその存在が示される場合にのみ存在する。CBPYは5. 3. 5小節及び表12 / JT-H263で記述されるように符号化される。EI、EPピクチャにおける上方予測マクロブロック、EPピクチャにおける双方向予測マクロブロック及びEI、EP、BピクチャにおけるINTRAマクロブロックは、INTRAマクロブロックのためのCBPY定義を使用し、EI、EP、Bピクチャにおける他のマクロブロックタイプはINTERマクロブロックのためのCBPY定義を使用する。



#### ○. 4. 5 量子化情報 (QDUANT) (2ビット/可変長)

DQUANTは他のピクチャのマクロブロックタイプの場合と同様に使用される。5. 3. 6小節及び付属資料T参照。

#### ○. 4. 6 動きベクトルデータ (MVDFW、MVDBW) (可変長)

MVDFWはもしあれば、それは前方ベクトルのための動きベクトルデータである。(Bピクチャでのみ許されるが)MVDBWはもしあれば、それは後方ベクトルのための動きベクトルデータである。可変長符号語は表14/JT-H263で与えられるが、無制限動きベクトルモードが使われるなら付表D. 3/JT-H263で与えられる(付属資料D参照)。

### ○. 5 動きベクトルの復号

#### ○. 5. 1 差分動きベクトル

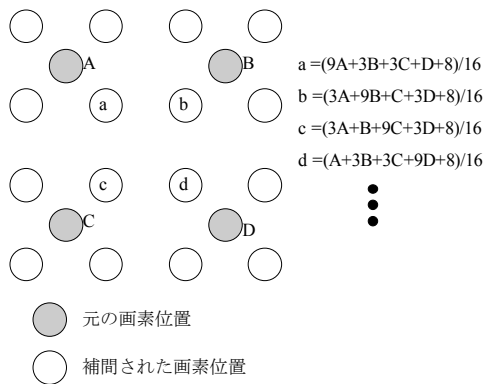
前方、後方又は双方向予測ブロックにおける動きベクトルは差分符号化される。マクロブロック動きベクトルを回復するため、予測が動きベクトルの差分に加えられる。前方動きベクトルは周囲のマクロブロックの前方動きベクトルからのみ予測され、後方動きベクトルは周囲のマクロブロックの後方動きベクトルからのみ予測されることを除けば、予測は、6. 1. 1小節で述べられたのと似た方法で作られる。6. 1. 1小節で述べたように、ピクチャ、GOBあるいはスライスの境界での特別な場合と同じ決定規則が適用される。もし、近傍のマクロブロックが、(前方又は後方の)同じタイプの動きベクトルを持たなければ、そのマクロブロックの候補予測値はその動きベクトルタイプではゼロとなる。

#### ○. 5. 2 直接モードの動きベクトル

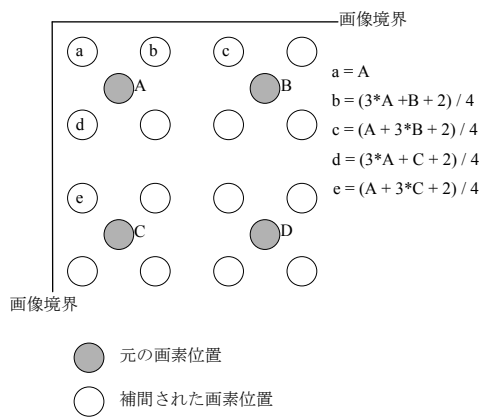
直接モードで符号化されたマクロブロックでは、ベクトルの差分は伝送されない。その代わりに前方及び後方動きベクトルは、 $MV_D$ が常にゼロという制限の下でG. 4節に述べられた時間的に続くPベクトルから直接算出される。これらの得られたベクトルは他の動きベクトルの予測では使われない。もし時間的に後続する参照ピクチャの対応する領域がINTRAモードで符号化されるならば、直接モードに対する予測処理における使用のためにその領域に割り当てられる前方及び後方動きベクトルは、値ゼロを持たなければならない。

### ○. 6 内挿補間フィルタ

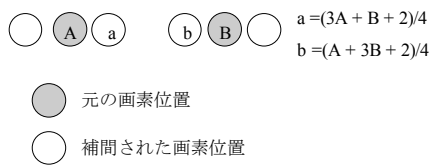
二次元スケーラビリティのためにピクチャを内挿補間する方法を付図○. 8/JT-H263及び付図○. 9/JT-H263に示す。最初の図は内部の画素を補間する方法を示し、2番目の図はピクチャ境界近傍の画素を補間する方法を示す。これは、付属資料Pのいくつかの場合及び付属資料Qで使われるのと同じ技術である。一次元空間スケーラビリティにおいてピクチャを補間する方法を付図○. 10/JT-H263及び付図○. 11/JT-H263に示す。付図○. 10/JT-H263は水平方向における内部画素の補間を示す。垂直方向の補間についても同様である。付図○. 11/JT-H263はピクチャ境界の画素補間方法を示す。また垂直方向の補間についても同様である。これは、付属資料Pのいくつかの場合で使われるのと同じ技術である。



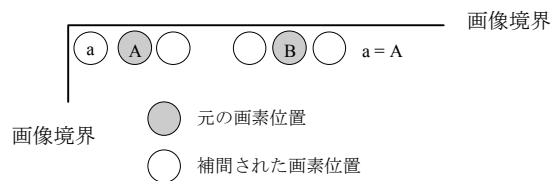
付図〇． 8 / JT-H 2 6 3 二次元スケーラビリティにおける画素補間方法  
(ITU-T H. 263)



付図〇． 9 / JT-H 2 6 3 境界における二次元画素補間方法  
(ITU-T H. 263)



付図〇． 10 / JT-H 2 6 3 一次元スケーラビリティの画素補間方法  
(ITU-T H. 263)



付図O. 11 / JT-H263 ピクチャ境界の一次元画素補間方法  
(ITU-T H. 263)

## 付属資料P 参照ピクチャ再サンプリング

(この付属資料は本標準の必須部分である)

### P. 1 序論

この付属資料は、現在のピクチャの予測に使用する「ワープ」ピクチャを生成するために、前に復号された参照ピクチャに適用される再サンプリングプロセスの使用とシンタックスを記述したものである。この再サンプリングシンタックスは、異なった情報源フォーマットを持つ先行するピクチャに対する現在のピクチャの関係を指定することが出来、また現在のピクチャに対する前のピクチャの形、サイズ、位置の「グローバル動き」ワープ変更を指定することができる。特に、符号化している間にピクチャの解像度を適応的に変更するために参照ピクチャ再サンプリングモードを使用することが出来る。双線形内挿補間係数の生成するために高速アルゴリズムが使用される。このモードを使用する能力とその特徴がサポートされる程度は、外部で交渉される(例えばTTC標準JT-H245)。このモードは、能力交換(例えば、“ファクター4”ピクチャのサイズ変更のみ、1/2画素解像度ピクチャのワープのみ、あるいは任意のイメージサイズのサイズ変更と変位のサポート)する間に定義してもよい限定されたシナリオにおいて使用される。

注：ピクチャ領域のエッジ及び輝度と色差標本の相対的な位置の空間アライメントを維持するために、異なる解像度のピクチャ間のデフォルトのイメージ変換が定義される。符号器で様々な解像度のピクチャを生成するため及び復号後様々な解像度のピクチャの表示(特に再サンプリングにおいて引き起こされる位相シフトにより発生する空間位置におけるシフトについて)のために用いられる再サンプリング操作の設計において、このことが関連してもよい。またこのモードを動的適応ピクチャ解像度変更を使用することが出来るので、ピクチャ表示のサイズ変更をすることなく符号化されたピクチャサイズを切り替えることを許容するために、符号化されたピクチャサイズより高い解像度で復号ピクチャを表示するための外部の交渉をこのモードの操作で利用してもよい。

もし、PLUSPTYPEフィールド中の参照ピクチャ再サンプリングビットがセットされていないがPLUSPTYPEが存在し、ピクチャがINTERかBかEPピクチャ又は改良PBフレームであり、時間的に前に符号化されたピクチャとピクチャサイズが異なる場合には、ワープパラメータ(P. 2. 2節参照)が0に、フィルモード(P. 2. 3節参照)がclipに、変位精度(P. 2. 1節参照)が $\frac{1}{16}$ の精度にセットされた、参照ピクチャ再サンプリングモードが起動される。これにより、再サンプリングプロセスはピクチャ解像度における予想的に符号化された変化として容易に動作する。CIFと4CIF間の遷移のような簡単な“ファクター4”解像度変更の場合、丸め制御の適用を除き空間スケーラビリティ(付属資料O)あるいは縮小解像度更新(付属資料Q)に適用されるのと同じ簡単なフィルタに再サンプリングプロセスは縮小される。

そのピクチャがEPピクチャであり、参照ピクチャ再サンプリングビットが参照レイヤのピクチャヘッダのPLUSPTYPEフィールドにセットされている場合には、このビットは、高品質化レイヤのEPピクチャのピクチャヘッダにもセットされなければならない。

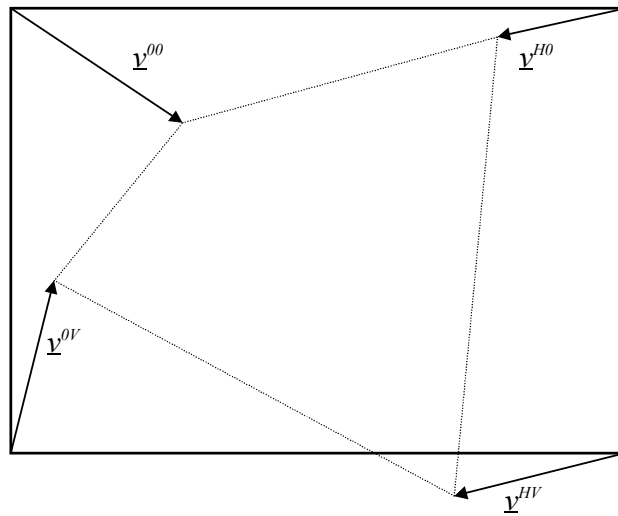
Bピクチャにおいて参照ピクチャ再サンプリングモードが用いられるならば、時間的に後に引き続きピクチャでなく時間的に前のアンカーピクチャに再サンプリングプロセスは適用されなければならない。再サンプリングプロセスが適用される時間的に前のアンカーピクチャは復号ピクチャ(すなわち、このモードが後続の参照ピクチャに対しても実施されるなら、参照ピクチャ再サンプリングモードが適用されるどの再サンプリングよりも前)でなければならない。時間的に後のアンカーピクチャはBピクチャと同じピクチャサイズを持たなければならない。

改良PBフレームに対して参照ピクチャ再サンプリングモードが呼び出されるならば、ワープパラメータ一式が送信され、再サンプリング参照ピクチャは改良PBフレームのB、P部分の両方の参照として用いられる。

参照ピクチャ再サンプリングモードは、現在のピクチャの全てのピクチャ、GOB、スライスヘッダのTRPI及びTRPの値が同じ参照ピクチャの使用を指定するのでない限り(その場合、指示された参照ピクチャは、参照ピクチャ再サンプリング処理が暗黙に呼び出されるかどうかを決定するピクチャであり、かつ、再サンプリング処理が適用されなければならないピクチャである)、参照ピクチャ選択モード(付属資料N参照)が使用されている時に呼び出されてはならない。

参照ピクチャ再サンプリングは、現在のピクチャ領域の4つのコーナーの変位に関して定義される。水平方向サイズHと垂直方向サイズVをもつ現在のピクチャの輝度フィールドにおいて、ピクチャのそれぞれ左上、右上、左下、右下のコーナーに対して4つの概念的な動きベクトル  $\underline{v}^{00}$ ,  $\underline{v}^{H0}$ ,  $\underline{v}^{0V}$ ,  $\underline{v}^{HV}$  が定義される。付図P. 1 / JT-H 2 6 3に示すように、これらのベクトルは現在のピクチャのコーナーを前の復号ピクチャの対応するコーナーにマッピングするために現在のピクチャのコーナーをどのように動かすかを記述するものである。これらのベクトルの単位は、参照ピクチャ格子内での単位と同じである。現在のピクチャ内部の真の値の位置 (x, y) におけるベクトル  $\underline{v}(x, y)$  を生成するために、双線形内挿補間の近似が使われる。すなわち、

$$\underline{v}(x, y) = \left(1 - \frac{y}{V}\right) \left[ \left(1 - \frac{x}{H}\right) \underline{v}^{00} + \left(\frac{x}{H}\right) \underline{v}^{H0} \right] + \left(\frac{y}{V}\right) \left[ \left(1 - \frac{x}{H}\right) \underline{v}^{0V} + \left(\frac{x}{H}\right) \underline{v}^{HV} \right].$$



付図P. 1 / JT-H 2 6 3 ワープに使用される動きベクトルの概念例 (ITU-T H. 2 6 3)

現在のピクチャの水平方向サイズHと垂直方向サイズV、参照ピクチャの水平方向サイズ $H_R$ と垂直方向サイズ $V_R$ は、これらの値が16で割り切れるか否かに関わらず、ピクチャヘッダによって示される。ピクチャ幅または高さが16で割り切れない場合には、再サンプリングプロセスで使用されるのと同じフィルモードを用いて再サンプリングピクチャに画素を付加することにより付加的領域を生成しなければならない。

簡潔な記述のため、再サンプリングベクトル  $\underline{r}^0$ ,  $\underline{r}^x$ ,  $\underline{r}^y$ ,  $\underline{r}^{xy}$  を以下のように定義する。

$$\begin{aligned} \underline{r}^0 &= \underline{v}^{00} \\ \underline{r}^x &= \underline{v}^{H0} - \underline{v}^{00} \\ \underline{r}^y &= \underline{v}^{0V} - \underline{v}^{00} \\ \underline{r}^{xy} &= \underline{v}^{00} - \underline{v}^{H0} - \underline{v}^{0V} + \underline{v}^{HV}. \end{aligned}$$

この定義を用いて、双線形内挿補間の式は以下のように書き直すことができる。

$$\underline{v}(x,y) = \underline{r}^0 + \left(\frac{x}{H}\right)\underline{r}^x + \left(\frac{y}{V}\right)\underline{r}^y + \left(\frac{x}{H}\right)\left(\frac{y}{V}\right)\underline{r}^{xy}$$

ワーブを目的として、ピクチャ領域の左上コーナーの座標は $(x,y)=(0,0)$ であり、それぞれの画素は高さと幅を持ち、その画素の中心は $(x,y)=(i_L + \frac{1}{2}, j_L + \frac{1}{2})$  ( $i_L = 0, \dots, H-1$  及び  $j_L = 0, \dots, V-1$ ) に位置すると仮定する。

ここで下添字Lは、 $i_L$ と $j_L$ が輝度フィールドに属していることを示している。(画素のアスペクト比は通常一定であるかあるいはアスペクト比の変換はこの再サンプリングで実行されるため、ワーブのために実際の画素アスペクト比を考慮する必要はない。) この規定により、参照ピクチャの輝度フィールドの各位置での  $x$  と  $y$  の変位は次のようになる。

$$v_x(i_L, j_L) = \frac{1}{HV} \left[ HVr_x^0 + (i_L + \frac{1}{2})Vr_x^x + (j_L + \frac{1}{2})Hr_x^y + (i_L + \frac{1}{2})(j_L + \frac{1}{2})r_x^{xy} \right]$$

$$v_y(i_L, j_L) = \frac{1}{HV} \left[ HVr_y^0 + (i_L + \frac{1}{2})Vr_y^x + (j_L + \frac{1}{2})Hr_y^y + (i_L + \frac{1}{2})(j_L + \frac{1}{2})r_y^{xy} \right]$$

すべての位置と位相は、座標  $(x,y) = (\frac{1}{2}, \frac{1}{2})$  である左上コーナーの画素の中心との比較で算出しなければならないので、基本的に重要な量は次のようになる。

$$\begin{aligned} x_R(i_L, j_L) - \frac{1}{2} &= \left(i_L + \frac{1}{2}\right) + v_x(i_L, j_L) - \frac{1}{2} \\ &= \frac{1}{HV} \left[ HVr_x^0 + \left(i_L + \frac{1}{2}\right)(HV + Vr_x^x) + \left(j_L + \frac{1}{2}\right)Hr_x^y + \left(i_L + \frac{1}{2}\right)\left(j_L + \frac{1}{2}\right)r_x^{xy} \right] - \frac{1}{2} \end{aligned}$$

$$\begin{aligned} y_R(i_L, j_L) - \frac{1}{2} &= \left(j_L + \frac{1}{2}\right) + v_y(i_L, j_L) - \frac{1}{2} \\ &= \frac{1}{HV} \left[ HVr_y^0 + \left(j_L + \frac{1}{2}\right)Vr_y^x + \left(j_L + \frac{1}{2}\right)(HV + Hr_y^y) + \left(i_L + \frac{1}{2}\right)\left(j_L + \frac{1}{2}\right)r_y^{xy} \right] - \frac{1}{2} \end{aligned}$$

これらの近似式を使用して、前に復号された参照ピクチャにおける位置が決定されると、再サンプリングされた画素の値を生成するために、後でこの付属資料の中で指定される双線形内挿補間を使用しなくてはならない。

それぞれの再サンプリングベクトルは、幾何学的なワーブを記述する第1成分と、予測されるピクチャ（水平方向サイズHと垂直方向サイズV）と参照ピクチャ（水平方向サイズ  $H_R$  と垂直方向サイズ  $V_R$ ）のサイズの相違を説明する第2成分の2つの成分に分解される。

このベクトルの分解値は次の通りである。

$$\begin{aligned} \underline{v}^{00} &= \underline{v}_{warp}^{00} + \underline{v}_{size}^{00} = \underline{v}_{warp}^{00} + (0,0) \\ \underline{v}^{H0} &= \underline{v}_{warp}^{H0} + \underline{v}_{size}^{H0} = \underline{v}_{warp}^{H0} + (H_R - H, 0) \\ \underline{v}^{0V} &= \underline{v}_{warp}^{0V} + \underline{v}_{size}^{0V} = \underline{v}_{warp}^{0V} + (0, V_R - V) \\ \underline{v}^{HV} &= \underline{v}_{warp}^{HV} + \underline{v}_{size}^{HV} = \underline{v}_{warp}^{HV} + (H_R - H, V_R - V) \end{aligned}$$

## P. 2 シンタックス

ピクチャヘッダの PLUSPTYPEフィールドに参照ピクチャ再サンプリングビットがセットされる時はいつでも、ピクチャヘッダの RPRPフィールドは参照ピクチャ再サンプリングプロセスを制御するパラメータを含

む。この節で記述されるように、このパラメータは2ビットのワーブ変位精度（WDA）フィールドを含み、8個のワーブパラメータあるいは1ビットのワーブパラメータ拡張を含んでもよく、またフィルモードを含む。

**P. 2. 1 ワーブ変位精度（WDA）（2ビット）**

2ビットのワーブ変位精度フィールドWDAはビット列の中のRPRPフィールドの最初に現れ、それぞれの画素の変位精度を示す。“10”はそれぞれの画素のxとy変位が1/2画素精度で量子化されることを示す。“11”は、変位が $\frac{1}{16}$ 画素精度で量子化されることを示す。その他の値は将来のために予約されている。

**P. 2. 2 ワーブパラメータ（可変長）**

INTER、Bピクチャまたは将来のPBフレームにおいて参照ピクチャ再サンプリングパラメータが送信される時、付表D. 3/JT-H263に示す可変長符号を用いた8個のワーブパラメータがピクチャヘッダに含まれる。SNRスケラビリティを用いるEPピクチャでは、下位レイヤのワーブパラメータが使われワーブパラメータは送信されない。もし空間スケラビリティを用いるEPピクチャにおけるピクチャヘッダのPLUSPTYPEフィールドに参照ピクチャ再サンプリングビットがセットされているならば、下位レイヤのそれぞれのアップサンプリングされた次元量（下添字xかつあるいはyをもつワーブパラメータ）を2倍し、さらにワーブパラメータのLSBを定義するために関連するワーブパラメータの位置に置かれて送られるもう1ビットの値を加えることにより、下位レイヤのワーブパラメータは現在のレイヤに必要な精度に更新される。

8つの整数値のワーブパラメータ（あるいはそれらの1ビットの詳細）は、次の順序で送られる。

$$w_x^0, w_y^0, w_x^x, w_y^x, w_x^y, w_y^y, w_x^{xy}, w_y^{xy}$$

これらのパラメータが1ビットの詳細ではないとき、PLUSPTYPEがある時の無制限動きベクトルモードでの動きベクトル差分と類似した方法で、すなわちパラメータ範囲の制限（つまり-4095から+4095の範囲）が無い付表D. 3/JT-H263を用いてこれらのパラメータは送られる。動きベクトル差分の対を符号化するときのように、各対のワーブパラメータ送信後必要に応じてエミュレーション防止ビットを付加しなければならない。

つまり、表D. 3/JT-H263のオールゼロの符号語（1/2画素単位での+1の値）が（ $w_x^0$ と $w_y^0$ 、 $w_x^x$ と $w_y^y$ 、あるいは $w_x^{xy}$ と $w_y^{xy}$ ）の対で両方のワーブパラメータに対して用いられる場合、スタートコードエミュレーションを防ぐために値が1である1ビットが1対の符号語の後に続く。

これらの8つのワーブパラメータは、再サンプリングベクトルのサイズ変更成分を取り除くことにより推測される変位と比較してのピクチャコーナー変位として解釈される。現在のピクチャの輝度フィールドで1/2画素オフセットを表現するため、ワーブパラメータの大きさが変えられる。これらのパラメータ値の範囲は、-4095~+4095である。ワーブパラメータは、次の関係式により再サンプリングベクトルを使用して定義される。

$$\begin{aligned} w_x^0 &= 2r_x^0 & w_y^0 &= 2r_y^0 \\ w_x^x &= 2(r_x^x - (H_R - H)) & w_y^x &= 2r_y^x \\ w_x^y &= 2r_x^y & w_y^y &= 2(r_y^y - (V_R - V)) \\ w_x^{xy} &= 2r_x^{xy} & w_y^{xy} &= 2r_y^{xy} \end{aligned}$$

**P. 2. 3 フィルモード（FILL\_MODE）（2ビット）**

INTERあるいはBピクチャあるいは改良PBフレームで、ピクチャヘッダにおいて直後に続く可変長符号化されたワーブパラメータは、参照ピクチャにおいて計算された位置が参照ピクチャ領域外にある画素の値に対して

取られるフィルモード動作を定義する2ビットである。2ビットの意味は付表P. 1/JT-H263において、またそれらの位置は付図P. 2/JT-H263において示される。EPピクチャでは、フィルモード動作は参照レイヤと同じであり、フィルモードビットの2ビットは送信されない。

付表P. 1/JT-H263 フィルモードビット/動作  
(ITU-T H. 263)

フィルモード ビット	フィル動作
0 0	<i>Color</i>
0 1	<i>Black</i>
1 0	<i>Gray</i>
1 1	<i>Clip</i>

フィルモードが*clip*の場合、前の参照ピクチャの領域外の画素値がイメージ境界画素の値から外挿補間により推測されるように、前の参照ピクチャでの位置座標は無制限動きベクトルモードにおける場合のように独立に制限される。フィルモードが*Black*の場合、前の参照ピクチャの領域外の輝度画素は $Y=16$ の値が割り当てられ、色差画素は $C_B=C_R=128$ の値が割り当てられる。フィルモードが*Gray*の場合、輝度と色差値は、 $Y=C_B=C_R=128$ の値が設定される。フィルモードが*Color*の場合、次の節で記述されるようにフィルカラーを指定するために追加のフィールドが送られる。

**P. 2. 4 フィルカラー仕様 (Y\_FILL, C<sub>B</sub>\_EPB, C<sub>B</sub>\_FILL, C<sub>R</sub>\_EPB, C<sub>R</sub>\_FILL) (26ビット)**

フィルモードが*Color*でありピクチャがEPピクチャでないならば、フィルカラーを正確に指定する3つの8ビット整数、 $y\_fill$ 、 $C_{B\_fill}$ 、 $C_{R\_fill}$ がビット列においてフィルモードビットの後に続く。3つの8ビット整数の間には、各々値が1である2つのエミュレーション防止ビット( $C_{B\_EPB}$ と $C_{R\_EPB}$ )が置かれる。フィルモードが*Color*である時だけ存在するこの色仕様のフォーマットは、付図P. 2/JT-H263に示される。それぞれの8ビット整数フィールドは自然表現により送られる。EPピクチャでは、フィルモード動作(とフィルカラー)はリファレンスレイヤのものと同じである、そしてフィルカラー仕様は送信されない。

FILL_MODE	Y_FILL	C <sub>B</sub> _EPB	C <sub>B</sub> _FILL	C <sub>R</sub> _EPB	C <sub>R</sub> _FILL
-----------	--------	---------------------	----------------------	---------------------	----------------------

付図P. 2/JT-H263 フィルモードとフィルカラーの表示データのフォーマット  
(ITU-T H. 263)

**P. 3 再サンプリングアルゴリズム**

この節で記述される方法は、再サンプリングされた参照ピクチャの標本の生成に使用される方法と結果として数学的に同等でなければならない。整数ワープパラメータ  $w_x^0, w_y^0, w_x^x, w_y^x, w_x^y, w_y^y, w_x^{xy}, w_y^{xy}$  を用いて、 $\frac{1}{32}$ 画素精度(実際の変位は32で除算することで得られる)での輝度フィールドのコーナーの  $x, y$  変位を表す整数パラメータ  $u_x^{00}, u_y^{00}, u_x^{H0}, u_y^{H0}, u_x^{0V}, u_y^{0V}, u_x^{HV}, u_y^{HV}$  は、以下のように定義される。

$$\begin{aligned}
 u_x^{00} &= 16w_x^0 & u_y^{00} &= 16w_y^0 \\
 u_x^{H0} &= 16(w_x^0 + w_x^x + 2(H_R - H)) & u_y^{H0} &= 16(w_y^0 + w_y^x) \\
 u_x^{0V} &= 16(w_x^0 + w_x^y) & u_y^{0V} &= 16(w_y^0 + w_y^y + 2(V_R - V)) \\
 u_x^{HV} &= 16(w_x^0 + w_x^x + w_x^y + w_x^{xy} + 2(H_R - H)) & u_y^{HV} &= 16(w_y^0 + w_y^x + w_y^y + w_y^{xy} + 2(V_R - V))
 \end{aligned}$$



次に、仮想フレームの水平方向と垂直方向のサイズを示す $H'$ と $V'$ は、次の条件を満たす最小の整数として定義される。

$$H' \geq H, V' \geq V, H' = 2^m, V' = 2^n, m \text{ と } n \text{ は正の整数}$$

双線形外挿補間を輝度フィールドのコーナーベクトルに適用することにより、 $\frac{1}{32}$ 画素精度（実際の変位は32で除算することで得られる）での仮想ポイント $(x, y) = (0, 0), (H', 0), (0, V'), (H', V')$ における輝度フィールドの $x, y$ 変位を表す整数パラメータ $u_x^{LT}, u_y^{LT}, u_x^{RT}, u_y^{RT}, u_x^{LB}, u_y^{LB}, u_x^{RB}, u_y^{RB}$ は、以下のように定義される。

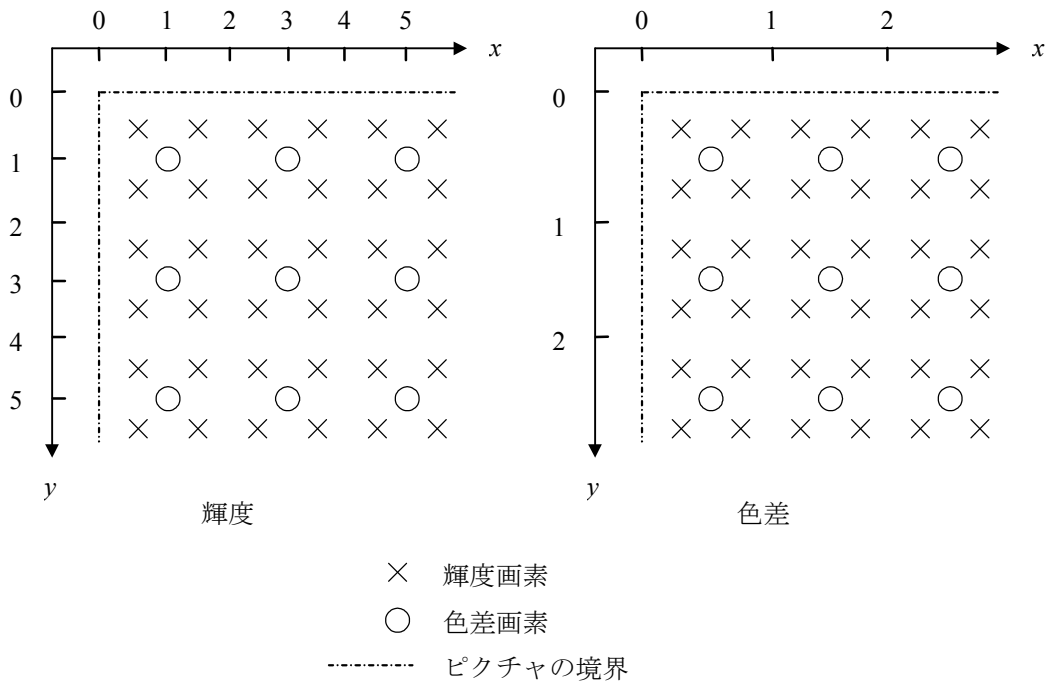
$$\begin{aligned} u_x^{LT} &= u_x^{00} & u_y^{LT} &= u_y^{00} \\ u_x^{RT} &= \left( (H - H')u_x^{00} + H'u_x^{H0} \right) // H & u_y^{RT} &= \left( (H - H')u_y^{00} + H'u_y^{H0} \right) // H \\ u_x^{LB} &= \left( (V - V')u_x^{00} + V'u_x^{0V} \right) // V & u_y^{LB} &= \left( (V - V')u_y^{00} + V'u_y^{0V} \right) // V \\ u_x^{RB} &= \left( (V - V') \left( (H - H')u_x^{00} + H'u_x^{H0} \right) + V' \left( (H - H')u_x^{0V} + H'u_x^{HV} \right) \right) // (HV) \\ u_y^{RB} &= \left( (V - V') \left( (H - H')u_y^{00} + H'u_y^{H0} \right) + V' \left( (H - H')u_y^{0V} + H'u_y^{HV} \right) \right) // (HV) \end{aligned}$$

ここで、「//」は、商に最も近い整数に丸めかつ1/2整数値を0から遠い値に丸める整数による除算を意味する。

この付属資料の残りにおいて、輝度と色差フィールドともに画素のセンターはポイント $(x, y) = (i + \frac{1}{2}, j + \frac{1}{2})$ にあると仮定する。整数パラメータ $i, j$ は、以下のように定義される。

- (1)  $i = 0, \dots, H - 1$  かつ  $j = 0, \dots, V - 1$  : 輝度,
- (2)  $i = 0, \dots, \frac{H}{2} - 1$  かつ  $j = 0, \dots, \frac{V}{2} - 1$  : 色差

これは、付図P. 3 / JT-H263で示すように輝度と色差に対して異なる座標系が使われることを意味する。色差のための座標系を用いることにより、上記に定義される整数パラメータ $u_x^{LT}, u_y^{LT}, u_x^{RT}, u_y^{RT}, u_x^{LB}, u_y^{LB}, u_x^{RB}, u_y^{RB}$ を、 $\frac{1}{64}$ 画素精度（実際の変位は64でこれらの値を除算することで得られる）での仮想ポイント $(x, y) = (0, 0), (H'/2, 0), (0, V'/2)$ と $(H'/2, V'/2)$ における色差フィールドの $x$ と $y$ 変位と見なすこともできる。これらのパラメータ及び輝度では2色差では1に定義されている付加パラメータ $S$ を用いて、輝度と色差フィールドの再サンプリングアルゴリズムは共通の方程式を使用して定義される。



付図P. 3 / JT-H263 輝度と色差フィールドの座標系  
(ITU-T H. 263)

$\frac{S}{64}$  画素精度 (実際の変位は、 $64/S$ でこれらの値を除算することによって得られる) において、 $(x, y) = (0, j + \frac{1}{2})$  と  $(SH'/2, j + \frac{1}{2})$  でのピクチャフィールドのxとy変位を示す整数パラメータ  $u_x^L(j), u_y^L(j), u_x^R(j), u_y^R(j)$  は、1次元線形内挿補間を用いて以下のように定義される。

$$\begin{aligned} u_x^L(j) &= \left( (SV'-2j-1)u_x^{LT} + (2j+1)u_x^{LB} \right) // (SV') & u_y^L(j) &= \left( (SV'-2j-1)u_y^{LT} + (2j+1)u_y^{LB} \right) // (SV') \\ u_x^R(j) &= \left( (SV'-2j-1)u_x^{RT} + (2j+1)u_x^{RB} \right) // (SV') & u_y^R(j) &= \left( (SV'-2j-1)u_y^{RT} + (2j+1)u_y^{RB} \right) // (SV') \end{aligned}$$

ここで、「//」は、商に最も近い整数に丸めかつ1/2整数値を0から遠い値に丸める整数による除算を意味する。最終的に、参照ピクチャにおける変換位置を指定するパラメータは、

$$\begin{aligned} I_R(i, j) &= Pi + \left( (SH'-2i-1)u_x^L(j) + (2i+1)u_x^R(j) + 32H'/P \right) /// (64H'/P) \\ J_R(i, j) &= Pj + \left( (SH'-2i-1)u_y^L(j) + (2i+1)u_y^R(j) + 32H'/P \right) /// (64H'/P) \\ i_R(i, j) &= I_R(i, j) /// P & j_R(i, j) &= J_R(i, j) /// P \\ \varnothing_x &= I_R(i, j) - (I_R(i, j) /// P)P & \varnothing_y &= J_R(i, j) - (J_R(i, j) /// P)P \end{aligned}$$

ここで、「///」： 負の無限方向への切り捨てを行う整数値による除算  
「/」： 整数値による除算 (精度を失うことのない結果の場合)  
P： xとy変位精度 (WDAが“1 0”の場合P=2、WDAが“1 1”または存在しない場合、P=16。ワーブ配置精度 (WDA) の定義についてはP. 2. 1小節を参照)

$$\left( \frac{I_R(i, j)}{P} + \frac{1}{2}, \frac{J_R(i, j)}{P} + \frac{1}{2} \right) : \text{位置}(x, y)\text{の変換位置 } (I_R(i, j)\text{と}J_R(i, j)\text{とも整数である})$$

$(i_R(i,j)+\frac{1}{2}, j_R(i,j)+\frac{1}{2})$  : 位置  $(x, y)$  の変換された位置に近いサンプリング点  $(I_R(i,j)$ と  $J_R(i,j)$ と共に整数である)

$(\varnothing_x, \varnothing_y)$ : 変換された位置の双線形内挿補間係数 ( $\varnothing_x$ と  $\varnothing_y$ は共に整数である。)

$P = 16$  のとき  $64H'/P = 2^{m+2}$ ,  $P = 2$  のとき  $64H'/P = 2^{m+5}$  であるので、除算をシフト演算に置換する事によりこの式の計算を簡単に行うことができる。これらのパラメータを使うと、再サンプリングされたピクチャにおいて、 $(x, y) = (i + \frac{1}{2}, j + \frac{1}{2})$  に位置する画素の標本値  $E_P(I, j)$  は双線形内挿補間を使って次のように得られる。

$$E_P(i, j) = \left( (P - \varnothing_y) \left( (P - \varnothing_x) E_R(i_R, j_R) + \varnothing_x E_R(i_R + 1, j_R) \right) + \varnothing_y \left( (P - \varnothing_x) E_R(i_R, j_R + 1) + \varnothing_x E_R(i_R + 1, j_R + 1) \right) + P^2 / 2 - 1 + RCRPR \right) / P^2$$

ここで、「/」は切り捨てによる除算、 $i_R$ と  $j_R$ は  $i_R(i, j)$ と  $j_R(i, j)$ の簡素化した記述、 $E_R(i_R, j_R)$ は必要に応じて適切なフィルモードを使用しての外挿補間を行った後の参照ピクチャでの  $(x, y) = (i_R + \frac{1}{2}, j_R + \frac{1}{2})$  に位置する画素の標本値を示す。パラメータ  $RCRPR$  の値は、次のように定義される。

- (1) 時間的に引き続きアンカーピクチャとして  $P$  ピクチャを持つ  $B$  ピクチャ (あるいは改良  $PB$  フレームの  $B$  部分) では、 $RCRPR$  は、時間的に後の  $P$  ピクチャの  $MPPTYPE$  (5. 1. 4. 3 小節を参照) での丸めタイプ ( $RTYPE$ ) ビットと等しい。これは改良  $PB$  フレームの  $RCRPR$  が  $P$  部分と  $B$  部分とで同じ値を持っていることを意味する。
- (2) 他のタイプのピクチャでは、 $RCRPR$  は現在のピクチャの  $RTYPE$  ビットと等しい。

#### P. 4 実装方法の例

この節では、前の節で記述されたアルゴリズムの実装例が疑似コードとして提供される。

##### P. 4. 1 仮想のポイントの変位

大きいピクチャが符号化される時、P. 3 節で示されるパラメータ  $u_x^{RB}$  と  $u_y^{RB}$  を得るための方程式の簡単な実装には、2進数表現で32ビット以上を必要とする変数の使用が必要となるかもしれない。容易に64ビットの整数あるいは浮動小数点レジスタを使用できないシステムに対しては、 $u_x^{RB}$  と  $u_y^{RB}$  の算出のために32ビット以上の変数を必要としないアルゴリズムの例を以下に示す。

$H, V, H', V'$  が4で割ることが出来るので、 $u_x^{RB}$  の定義は次のように書き直すことができる。

$$u_x^{RB} = \left( (V_Q - V_Q') \left( (H_Q - H_Q') u_x^{00} + H_Q' u_x^{H0} \right) + V_Q' \left( (H_Q - H_Q') u_x^{0V} + H_Q' u_x^{HV} \right) \right) // A$$

ここで、 $H_Q = H/4$ ,  $V_Q = V/4$ ,  $H_Q' = H'/4$ ,  $V_Q' = V'/4$ ,  $A = H_Q V_Q$  であり、「//」は、商に最も近い整数に丸めかつ1/2整数値を0から遠い値に丸める整数による除算を意味する。次に記述の簡略化のために、パラメータ  $T_T$  と  $T_B$  は以下のように定義される。

$$T_T = (H_Q - H_Q') u_x^{00} + H_Q' u_x^{H0}$$

$$T_B = (H_Q - H_Q') u_x^{0V} + H_Q' u_x^{HV}$$

負の無限方向に切り捨てられた整数での除算を示す演算子「///」と、 $a \% b = a - (a /// b) \cdot b$  で定義される演算子「%」を使うと、 $u_x^{RB}$  の値は次の疑似コードによって得られる。

```

q = (V_Q - V_Q') * (T_T /// A) + V_Q' * (T_B /// A) + ((V_Q - V_Q') * (T_T % A) + V_Q' * (T_B % A)) /// A;
r = ((V_Q - V_Q') * (T_T % A) + V_Q' * (T_B % A)) % A;
if (q < 0)
    u_x^{RB} = q + (r + (A - 1) / 2) / A;
else
    u_x^{RB} = q + (r + A / 2) / A;

```

$u_y^{RB}$  の値も、このアルゴリズムを使って計算される。

#### P. 4. 2 再サンプリングアルゴリズム

アルゴリズムの記述を簡略化するために、関数 `prior_sample` を定義する。その目的は、前の参照ピクチャのサンプリング格子に対しての、任意の整数位置  $(i_p, j_p)$  における画素値を生成することである。

```

clip(x_min, x, x_max) {
    if (x < x_min) {
        return x_min;
    } else if (x > x_max) {
        return x_max;
    } else {
        return x;
    }
}

prior_sample(i_p, j_p) {
    if (FILL_MODE = clip) {
        i_c = clip(0, i_p, S * H_R / 2 - 1);
        j_c = clip(0, j_p, S * V_R / 2 - 1);
        return prior_ref[i_c, j_c];
    } else {
        if ((i_p < 0) OR (i_p > S * H_R / 2 - 1) OR (j_p < 0) OR (j_p > S * V_R / 2 - 1)) {
            return fill_value;
        } else {
            return prior_ref[i_p, j_p];
        }
    }
}

```

疑似コードにおいて、`prior_ref [i, j]` は時間的に前の参照ピクチャでの行  $i$  と列  $j$  の標本を示す。

次に、P. 3 節で記述される双線形内挿補間を実行するフィルタ関数が定義される。次の関数の全引数は整数であり、双線形内挿補間係数、 $\theta_x$  と  $\theta_y$  は、0 から  $P - 1$  (両端を含む) の範囲で量子化されると仮定する。

```

filter(x0, y0, O_x, O_y) {
    return [(P-O_y) * ((P-O_x) * prior_sample(x0, y0) + O_x * prior_sample(x0+1, y0)) +
            O_y * ((P-O_x) * prior_sample(x0, y0+1) + O_x * prior_sample(x0+1, y0+1)) +
            P^2 / 2 - 1 + RCRPRJ / P^2;
}

```

最終的に、現在のピクチャの予測を生成する参照ピクチャのワープの方法は、これらの関数により定義される。予測ピクチャの画素はラスタ走査順序で生成される。 $u_x^L(j), u_y^L(j), u_x^R(j)$ , と  $u_y^R(j)$  の値はすでに算出され、変数  $u_x^L, u_y^L, u_x^R, u_y^R$  にロードされているものと仮定する。パラメータDを  $D=64H'/P$  と定義し、 $H'=2^m$  と記述すると、再サンプルされたフィールド（最上位ラインは第0番目のラインと定義される）のj番目のラインの画素標本値は、次の疑似コードによって得られる

```

a_x^i = D * P + 2 * (u_x^R - u_x^L);
a_y^i = 2 * (u_y^R - u_y^L);
a_x = u_x^L * S * 2^m + (u_x^R - u_x^L) + D / 2;
a_y = j * D * P + u_y^L * S * 2^m + (u_y^R - u_y^L) + D / 2;
for (i = 0; i < S * H / 2; i++) {
    I_R = a_x /// D;
    J_R = a_y /// D;
    i_R = I_R /// P;
    j_R = J_R /// P;
    O_x = I_R - (i_R * P);
    O_y = J_R - (j_R * P);
    new_ref[i, j] = filter(i_R, j_R, O_x, O_y);
    a_x += a_x^i;
    a_y += a_y^i;
}

```

ここで、このコードで使用されるすべての変数は整数変数であり、 $new\_ref[i, j]$  は、再サンプルされた参照ピクチャでの行iと列jのために生成される標本を示す。パラメータの定義により、このコードでのすべての除算は、2進数シフト操作に置き換えることが出来る。例えば、 $P=16$  のとき、

```

I_R = a_x /// D;
J_R = a_y /// D;
i_R = I_R /// P;
j_R = J_R /// P;

```

$$\emptyset_x = I_R - (i_R * P);$$

$$\emptyset_y = J_R - (j_R * P);$$

は、 $a_x, a_y, I_R$ と $J_R$ が2の補数表現で2進符号化される整数変数であると仮定して、次のように書き直すことができる。

$$I_R = a_x \gg (m+2);$$

$$J_R = a_y \gg (m+2);$$

$$i_R = I_R \gg 4;$$

$$j_R = J_R \gg 4;$$

$$\emptyset_x = I_R \& 15;$$

$$\emptyset_y = J_R \& 15;$$

ここで、「 $\gg N_{\text{shift}}$ 」は、 $N_{\text{shift}}$ ビット ( $N_{\text{shift}}$ は正の整数である) の右方向への2進算術シフト演算を示し、「 $\&$ 」は、ビット毎のAND演算を示す。

## P. 5 “ファクター4”再サンプリング

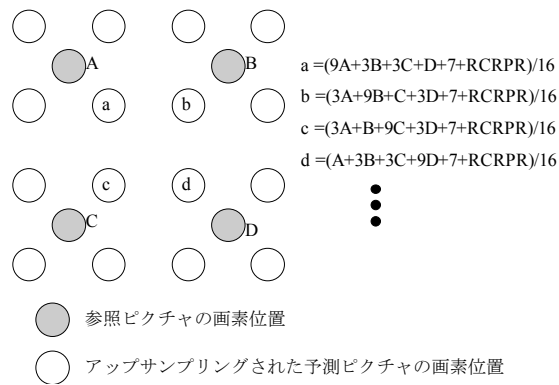
2または $\frac{1}{2}$ の係数によってピクチャの水平、垂直方向のサイズを変換する“ファクター4”再サンプリングは、

P. 3節で記述される再サンプリングアルゴリズムの特別な場合である。この特別な場合の再サンプリングアルゴリズムの簡略化した記述はこの節で規定される。

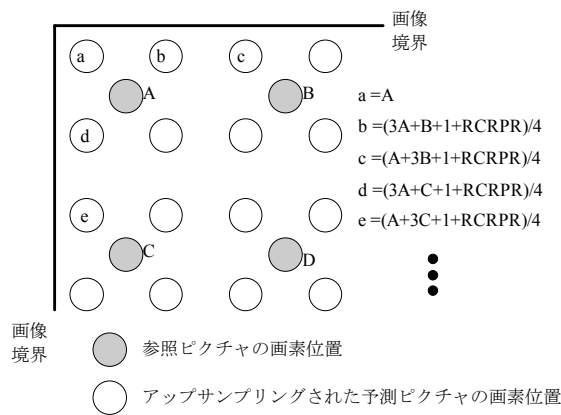
以下の図で使われるパラメータRCRPR値は、P. 3節で記述されるようにMPPTYPE (5. 4. 1. 3 小節参照) の丸めタイプ (RTYPE) ビットにより決定される。さらに、以下の図の「/」は、切捨ての除算を示す。

### P. 5. 1 “ファクター4”アップサンプリング

内部の画素のために“ファクター4”アップサンプリングで使用される画素値内挿補間方法は、付図P. 4 / J T-H 2 6 3で示される。選択されたフィルモード (P. 2. 3小節とP. 2. 4小節参照) に従い、ピクチャの外に画素が存在すると仮定することによって、境界画素に同じ内挿補間方法が応用される。フィルモードとしてclipが選択されるとき境界画素の内挿補間方法は、付図P. 5 / J T-H 2 6 3に示される。正確な“ファクター4”アップサンプリングは、少なくとも $\frac{1}{4}$ 画素精度でのxとy変位を必要とするので、P. 2. 1小節で規定されるワープ変位精度 (WDA) フィールドは、“1 1”にセットされなくてはならない、もしくはこのアップサンプリング方法を使うために再サンプリングが暗黙的に呼び出されなければならない。



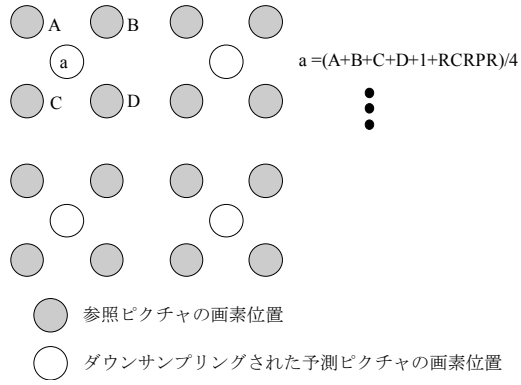
付図 P. 4 / JT-H 2 6 3 ピクチャ内画素 “ファクター 4” アップサンプリング (ITU-T H. 263)



付図 P. 5 / JT-H 2 6 3 ピクチャ境界画素の “ファクター 4” アップサンプリング (ITU-T H. 263) (Fillモード=制限)

P. 5. 2 “ファクター 4” ダウンサンプリング

“ファクター 4” ダウンサンプリングのための画素値内挿補間方法は、付図 P. 6 / JT-H 2 6 3 で示される。  
 $\frac{1}{2}$  画素精度での x と y 変位は正確な “ファクター 4” ダウンサンプリングに十分であるため、P. 2. 1 小節で規定される変位精度 (WDA) フィールド (もし存在するなら) の値として、“1 0” と “1 1” が共に許容される。



付図P. 6 / JT-H263 “ファクター4”ダウンサンプリング  
(ITU-T H. 263)



## 付属資料Q

### 縮小解像度更新モード

(この付属資料は本標準の必須部分である)

#### Q. 1 序論

この付属資料は、TTC標準JT-H263のオプションである縮小解像度更新モードを記述したものである。このモードの使用は、外部の手段により通知される(たとえばTTC標準JT-H245)。このモードの使用はピクチャヘッダのPLUSPTYPEフィールドで示す。

縮小解像度更新モードは、動きの大きなシーンの符号化における使用が期待され、主観的に十分な画質を保ちつつ符号化ピクチャレートを増加できる見込みがある。このモードにおいては、高解像度の最終的なピクチャを生成するために、高解像度の参照ピクチャを保持しつつ、符号器は低解像度で符号化したピクチャに対する更新情報を送信する。

このモードにおけるビット列のシンタックスは、本モード以外の符号化シンタックスと同じであるが、ビット列の意味や解釈が少々異なる。このモードにおいて、マクロブロックにより覆われるピクチャの小領域の高さおよび幅は、それぞれ2倍となる。よって、マクロブロックの数は、このモードでない場合の約4分の1である。動きベクトルデータはまた、高さおよび幅が普通の2倍のブロック、すなわち通常の $16 \times 16$ および $8 \times 8$ の代わりに、それぞれ $32 \times 32$ および $16 \times 16$ のブロックを参照する。一方、DCTあるいはテクスチャデータは、低解像度のピクチャにおける $8 \times 8$ のブロックを表すものとして考えるべきである。最終的なピクチャを生成するために、テクスチャデータは低解像度で復号され、フル解像度のピクチャへとアップサンプリングされる。アップサンプリングの後、表示用および後続の参照用となるピクチャを生成するために、フル解像度のテクスチャピクチャが(すでにフル解像度で)動き補償されたピクチャと加算される。

このモードにおいては、ピクチャヘッダにおいて示される水平サイズHと垂直サイズVを持つピクチャが、表示のための最終的なピクチャとして生成される。

このモードでは、予測のために使用され、後続の復号のために生成される参照ピクチャは、4.1節に記述されるデフォルトモードの場合と同様の水平サイズ $H_R$ と垂直サイズ $V_R$ を持つ。すなわち、 $H_R$ および $V_R$ は、次式で計算される。

$$H_R = ((H + 15) / 16) \times 16$$
$$V_R = ((V + 15) / 16) \times 16$$

ここで、HとVはピクチャヘッダにおいて示される水平サイズと垂直サイズを表し、“/”は切捨ての除算を表す。

またこの付属資料においては、次式で計算される高さ $H_C$ と幅 $V_C$ の縮小解像度でテクスチャは符号化される。

$$H_C = ((H_R + 31) / 32) \times 32$$
$$V_C = ((V_R + 31) / 32) \times 32$$

ここで、“/”は切捨ての除算を表す。

QCIFフォーマットの場合のように、 $H_C$ と $H_R$ あるいは $V_C$ と $V_R$ が同じ値でない場合は、参照されたピクチャは拡張されて、幅と高さが $H_C$ と $V_C$ である場合と同様の方法でピクチャは復号される。そして、復号の結果として得られた $32 \times 32$ のマクロブロックで覆われているピクチャは、幅 $H_R$ と高さ $V_R$ になるように右端と下端が切り取られ、右端と下端が切り取られたピクチャは後続の復号における参照ピクチャとして保存される。

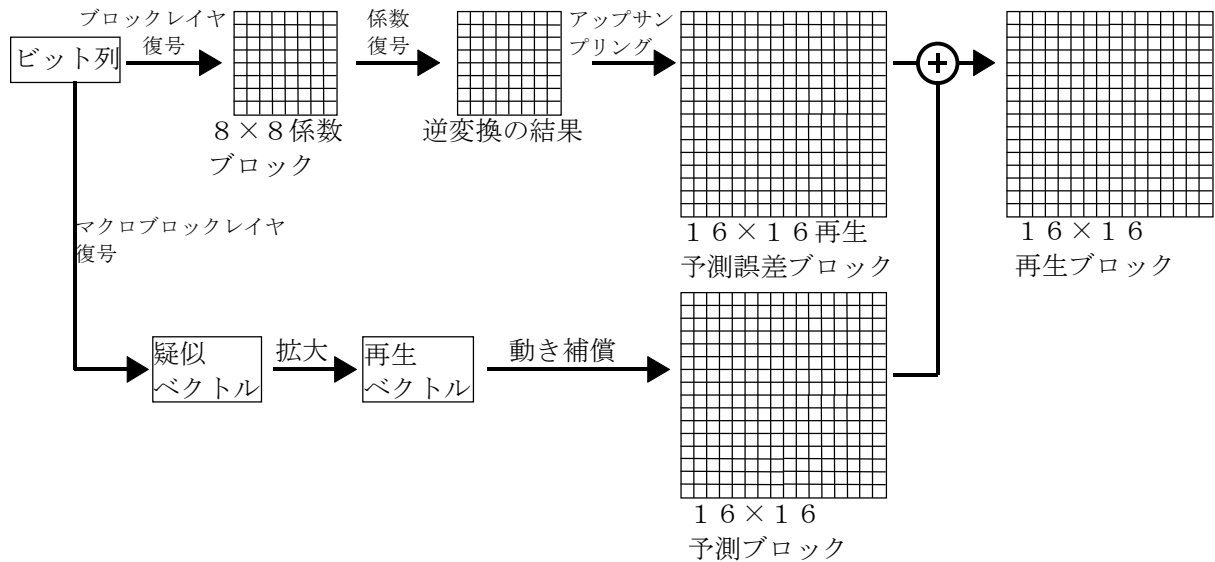
HとVが、結果として得られるピクチャの幅 $H_C$ と高さ $V_C$ に等しい場合、このピクチャは表示に使用される。そうでない場合は、サイズが幅 $H \times$ 高さ $V$ になるようにさらに切り取られ、表示目的のみに使用される。

時間、SNR、空間スケーラビリティモード(付属資料O参照)あるいは参照ピクチャ再サンプリングモード(付属資料P参照)が、このオプションと併用される場合、現在のピクチャの情報源フォーマットは参照ピクチャのフォーマットとは異なるかもしれない。この場合には、参照ピクチャ再サンプリングは、復号の前に行われなければならない。

注：このオプションが使用されるとき、参照ピクチャ（参照ピクチャ再サンプリングモードにより可能な再サンプリング後の参照ピクチャ）のサイズは現在のピクチャヘッダに示されるサイズと同じであるので、このモードは修正なしで参照ピクチャ選択モード（付属資料N参照）との併用が可能である。

**Q. 2 復号処理**

付図Q. 1 / JT-H263は縮小解像度更新モードにおけるブロック復号のブロック図を示している。



付図Q. 1 / JT-H263 縮小解像度更新モードにおけるブロック復号のブロック図 (ITU-T H. 263)

復号手順を以下の小節で述べる。

**Q. 2. 1 参照ピクチャの準備**

利用可能な参照ピクチャのサイズが $H_c$ と $V_c$ と異なる場合がある。その場合には、Q. 2. 1. 1小節またはQ. 2. 1. 2小節による復号手順の前に、参照ピクチャを変換しなければならない。

**Q. 2. 1. 1 参照ピクチャ再サンプリング**

時間、SNR、空間スケーラビリティモード（付属資料O参照）、あるいは参照ピクチャ再サンプリングモード（付属資料P参照）がこのオプションと併用された場合は、現在のピクチャと参照ピクチャの情報源フォーマットとは異なる可能性がある。この場合には、まず参照ピクチャ再サンプリングを、各付属資料に従って行わなければならない。

**Q. 2. 1. 2 参照ピクチャの拡張**

QCIFフォーマットのように、 $H_R$ または $V_R$ が32で割り切れない場合は、参照ピクチャは拡張される。この拡張の詳細な手順は、Q. 3節で定義される。

**Q. 2. 2 マクロブロックレイヤ復号**

復号は、輝度においては $32 \times 32$ のサイズに、色差においては $16 \times 16$ のサイズに“拡大された”ブロック上の操作として考えることができる。Q. 2. 2. 1小節およびQ. 2. 2. 2小節の記述のように、それぞれの拡大ブロックのテクスチャおよび動きデータは復号され、 $32 \times 32$ の動きブロックおよび $32 \times 32$ のテクスチャブロックを生成する。Q. 2. 2. 3小節の記述のように、これらの動きブロックとテクスチャブロックは、その後加算される。

### Q. 2. 2. 1 動き補償

はじめに、マクロブロック動きベクトル（または4本のマクロブロック動きベクトル）のそれぞれの成分は、 $MVD$ （および場合により $MVD_{2-4}$ ）で構成される。改良PBフレームモードにおいては、Bピクチャの $MV_F$ と $MV_B$ は $MV_{DB}$ から構成される。この動きベクトルの構成の詳細な手順は、Q. 4節で定義される。現在のピクチャのモードがBピクチャまたはEPピクチャの場合においても、Q. 4節の手順で前方向および後方向の動きベクトルが得られる。

マクロブロックの2つの色差ブロックの動きベクトルは、6. 1. 1小節の手順でマクロブロックの動きベクトルから得られる。拡張予測モードまたはデブロッキングフィルタモードのいずれかが使用され、そのためにマクロブロックに4本の動きベクトルが定義されている場合、両方の色差ブロックの動きベクトルはF. 2節の手順で4本の動きベクトルから得られる。改良PBフレームモードの場合、色差ベクトルの生成は付属資料Mで記述される。BピクチャまたはEPピクチャが使用されている場合、色差ベクトルの生成は付属資料Oで述べられている。

次に、予測ブロックが、INTERマクロブロックの動きベクトルから生成される。4つの $16 \times 16$ 輝度予測ブロックがマクロブロック動きベクトルから得られ、2つの $16 \times 16$ 色差予測ブロックが色差動きベクトルから得られる。サブ画素予測のための内挿補間については、6. 1. 2小節に示す。拡張予測モードと併用される場合は、拡大オーバーラップ動き補償が実施され、拡大重み付けマトリックスを用いて4つの $16 \times 16$ 輝度予測ブロックが得られる。そのための詳細な手順はQ. 5節で定義される。現在のピクチャが、改良PBフレーム、Bピクチャ、あるいはEPピクチャの場合は、予測ブロックのサイズが $8 \times 8$ ではなく $16 \times 16$ であることを除いて、他の関連付属資料の手順で予測ブロックが得られる。

### Q. 2. 2. 2 テクスチャ復号

はじめに、5. 4節の手順でブロックレイヤのビット列が復号される。そして、係数が復号され、6. 2節の逆変換の結果として、再生された縮小解像度の $8 \times 8$ 予測誤差ブロックが得られる。

次に、再生された縮小解像度の $8 \times 8$ 予測誤差ブロックのアップサンプリングにより、 $16 \times 16$ 再生予測誤差ブロックが得られる。それぞれの $16 \times 16$ 再生予測誤差ブロックの端に位置する画素の生成には、そのブロックに属する画素のみが使用される。詳細な手順はQ. 6節で定義される。

### Q. 2. 2. 3 ブロックの再生

それぞれの輝度、色差ブロックに対し、予測値と予測誤差の加算が実施される。ブロックのサイズが $8 \times 8$ ではなく $16 \times 16$ であること以外は、手順は6. 3. 1小節と同じである。そして、6. 3. 2小節によりクリッピングが行われる。

次に、ブロック境界フィルタが、再生された $16 \times 16$ ブロックの境界画素に適用される。詳細な手順はQ. 7節で定義される。

### Q. 2. 3 ピクチャの保存

CIFフォーマットのように $H_R$ と $V_R$ の両方が32で割り切れる場合、Q. 2. 2小節に記述されるように、再生された結果得られたピクチャは、それ以降の復号のためにそのまま参照ピクチャとして保存される。QCIFフォーマットのように割り切れない場合、 $32 \times 32$ のマクロブロックでちょうど覆われる再生されたピクチャは、幅 $H_R$ 、高さ $V_R$ になるように右端と下端が切り取られ、この切り取られたピクチャがそれ以降の復号のための参照ピクチャとして保存される。

### Q. 2. 4 表示

$H$ と $V$ の両方が $H_C$ と $V_C$ に等しい場合、Q. 2. 2小節において得られたピクチャは、そのまま表示に使用される。そうでなければ、この得られたピクチャは $H \times V$ のサイズまでさらに切り取られ、切り取られたピクチャは表示のみに使用される。

### Q. 3 参照ピクチャの拡張

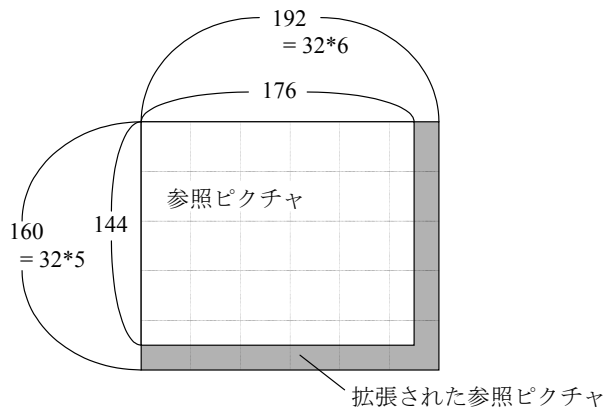
QCIFフォーマットのように $H_R$ と $V_R$ が32で割り切れない場合、マクロブロック／ブロックレイヤの復号の前に参照ピクチャの拡張が行われる。輝度のために拡張された参照ピクチャの幅と高さは、32で割り切れる次に大きなサイズであり、色差のために拡張された参照ピクチャは16で割り切れる次に大きなサイズである。

注：幅と高さが16で割り切れる次に大きなサイズで、ピクチャは復号されなければならないため、デフォルトモードにおける参照ピクチャの幅と高さは、画像フォーマットの幅と高さが16で割り切れない場合においても、常に16で割り切れるサイズに拡張される（4.1節参照）。

無制限動きベクトルモード、拡張予測モード、デブロッキングフィルタモードのいずれも、このオプションと併用されていない場合、拡張された画素は復号されたピクチャの参照画素として再生や表示のために使用されないこと  
がないので、拡張された画素は任意の値を取ることができる。

無制限動きベクトルモード、拡張予測モード、デブロッキングフィルタモードのいずれかが、このオプションと併用されている場合、動きベクトルがピクチャの右端や下端の外を指す場合の復号を確実にするために、参照ピクチャの端の画素を複製することにより参照ピクチャの拡張が行われる。

例えば、縮小解像度更新モードがQCIFに対して使用された場合、参照ピクチャの幅は176で高さは144であり、これらは32で割り切れない。QCIFピクチャを32×32のサイズのマクロブロックで覆うためには、マクロブロックの横列の数は6、マクロブロックの縦列は5でなければならない。よって、拡張された参照ピクチャの幅は192、高さは160となる。



付図Q. 2 / JT-H263 QCIFピクチャサイズの参照ピクチャの拡張  
(ITU-T H. 263)

付図Q. 2 / JT-H263に、QCIFにおける参照ピクチャの拡張を示す。輝度のための拡張された参照ピクチャは、次式により与えられる。

$$R_{RRU}(x, y) = R(x', y')$$

ここで、

$x, y$  = 画素領域における拡張された参照ピクチャの空間座標

$x', y'$  = 画素領域における参照ピクチャの空間座標

$R_{RRU}(x, y)$  = 拡張された参照ピクチャの  $(x, y)$  における画素値

$R(x', y')$  = 参照ピクチャの  $(x', y')$  における画素値

$$\begin{aligned} x' &= 175 && x > 175 \text{ かつ } x < 192 \text{ の場合} \\ &= x && \text{上記以外} \end{aligned}$$

$$\begin{aligned} y' &= 143 && y > 143 \text{ かつ } y < 160 \text{ のとき} \\ &= y && \text{上記以外} \end{aligned}$$

色差のための参照ピクチャも、同様に拡張される。

#### Q. 4 動きベクトルの再生

縮小解像度更新モードにおいては、動きベクトルの範囲が縦横の両方向とも約2倍に拡大される。表14/JT-H263に定義されるMVDの可変長符号を用いて、拡大された範囲を実現するために、各ベクトルの成分は1/2画素値または0の値のみに制限される。よって、デフォルトの縮小解像度更新モードにおける各動きベクトルの成分の範囲は、[-31.5, 30.5]となる。無制限動きベクトルモードが使用されている場合は、D. 2節に定義されるベクトルの範囲[-limit, limit]が疑似動きベクトルに当てはまり、動きベクトルのために[-(2×limit-0.5), 2×limit-1.5]に変換される。CIFでは、疑似動きベクトルの範囲が[-32, 31.5]、動きベクトルの範囲が[-63.5, 62.5]であることを意味する。UIフィールドが“01”に設定される場合、動きベクトルは無制限となる。しかし、動きベクトル(疑似動きベクトルだけでなく)は、D. 1. 1小節に述べられるように符号化領域の15画素より外を指さないように常に制限される。付図Q. 3/JT-H263は、値が(0, 0)のベクトル周辺において、マクロブロックの動きベクトルまたは4本の動きベクトル予測がとり得る位置を図示している。破線は整数の座標を示す。

Bピクチャにおいて、マクロブロックに対し差分動きベクトルを用いると、前方および後方予測のための動きベクトルが独立に得られる。縮小解像度更新モードでは、輝度ブロックの動きベクトル成分MV<sub>C</sub>は、以下のようにしてMVDおよびMVD<sub>2-4</sub>から再生される。

- (1) 疑似予測ベクトル成分“疑似P<sub>C</sub>”を、予測ベクトル成分P<sub>C</sub>から生成する。

$$\begin{aligned} \text{疑似 } P_C &= 0 && P_C = 0 \text{ の場合} \\ \text{疑似 } P_C &= \text{sign}(P_C) \times (|P_C| + 0.5) / 2.0 && P_C \neq 0 \text{ の場合} \end{aligned}$$

“/”は浮動小数点精度の割り算(正確さの損失なし)を示す。予測ベクトル成分P<sub>C</sub>は、6. 1. 1小節およびF. 2節で定義されるように、ベクトル成分MV<sub>1</sub>、MV<sub>2</sub>、MV<sub>3</sub>の中央値として定義される。

- (2) 疑似マクロブロックベクトル成分“疑似MV<sub>C</sub>”を、表14/JT-H263による差分動きベクトルMVD(およびMVD<sub>2-4</sub>)と疑似P<sub>C</sub>の加算により求める。

デフォルトの縮小解像度更新モードにおいては、疑似MV<sub>C</sub>の値は[-16, 15.5]の範囲に制限される。唯一の組み合わせが、許容範囲に収まる疑似MV<sub>C</sub>を生成する。6. 1. 1小節に定義されている方法と似た方法で、手順が実行される。

無制限動きベクトルモードが縮小解像度更新モードと併用される場合、付表D. 3/JT-H263から定まる差分動きベクトルMVD(およびMVD<sub>2-4</sub>)の加算により、疑似MV<sub>C</sub>が得られる。

4本の動きベクトルが存在する場合、F. 2節で定義される方法と似た方法で、この手順が実行される。

- (3) 動きベクトル成分MV<sub>C</sub>を、疑似MV<sub>C</sub>から次式により求める。

$$\begin{aligned} MV_C &= 0 && \text{疑似 } MV_C = 0 \text{ の場合} \\ MV_C &= \text{sign}(\text{疑似 } MV_C) \times (2.0 \times |\text{疑似 } MV_C| - 0.5) && \text{疑似 } MV_C \neq 0 \text{ 場合} \end{aligned}$$

結果として、各ベクトルの成分は1/2画素値または0の値に制限され、各ベクトルの成分の範囲は疑似動きベクトルの範囲の約2倍に拡大される。

- (4) 現在のピクチャが改良PBフレームモードの場合、あるいはBピクチャにおいてMBTYPEがダイレクトモードを示している場合は、前方および後方予測のための動きベクトル成分MV<sub>F</sub>かつ、またはMV<sub>B</sub>を生成する。

はじめに、疑似動きベクトル成分“疑似MV<sub>F</sub>”かつ、または“疑似MV<sub>B</sub>”を、付属資料Oまたは付属資料Mで定義される予測モードの規則に基づき計算する。

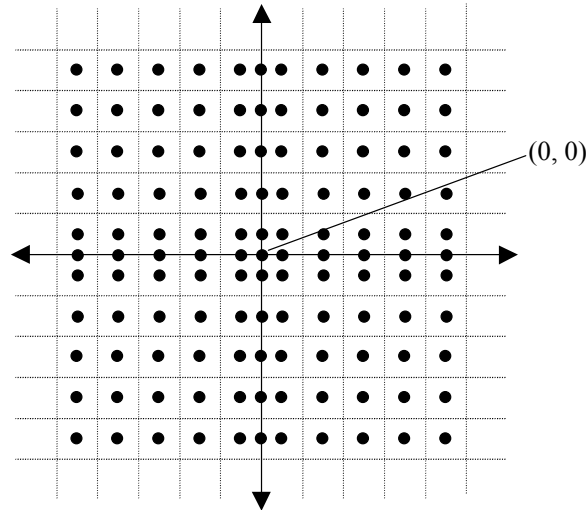
改良PBフレームモード(M. 2. 1小節参照)における双方向予測の場合、あるいはBピクチャ(O. 5. 2小節参照)においてMBTYPEがダイレクトモードを示している場合は、付属資料Gおよび付属資料Mで定義さ

れるように、疑似 $MV_D$ を0、疑似 $MV_C$ を $MV$ と仮定し、疑似 $MV_F$ および疑似 $MV_B$ を疑似 $MV_D$ および疑似 $MV_C$ から計算する。

改良PBフレームモード（M. 2. 2小節参照）における前方予測の場合、表12/JT-H263による可変長符号 $MVDB$ により疑似 $MV_{DB}$ が得られる。そして、疑似 $MV_{DB}$ を疑似予測値に加算することにより、疑似 $MV_F$ が得られる。疑似予測値を生成するために、付属資料MのM. 2. 2小節で定義される手順により得られる予測値を、この節の(1)項に定義される式により疑似予測値ベクトルに変換する。

改良PBフレームモード（M. 2. 3小節参照）における後方予測の場合は、疑似 $MV_B$ は0に設定される。

そして、この節の(3)項で定義される式により疑似 $MV_F$ かつ、または疑似 $MV_B$ から、前方および後方予測のための動きベクトル $MV_F$ かつ、または $MV_B$ が得られる



付図Q. 3/JT-H263 動きベクトルの再生  
(ITU-T H. 263)

### Q. 5 輝度信号用の拡大オーバーラップ動き補償

縮小解像度更新モードと拡張予測モードが併用されている場合、オーバーラップ動き補償を実施するために、拡大された重み付けマトリックスが使用される。各ブロックおよび重み付けマトリックスのサイズが $16 \times 16$ であること以外、予測ブロックの生成手順は付属資料FのF. 3節の記述と同じである。

$16 \times 16$ の輝度予測のための拡大された重み付けマトリックスは、付図Q. 4/JT-H263、付図Q. 5/JT-H263、および付図Q. 6/JT-H263で与えられる。

4	4	5	5	5	5	5	5	5	5	5	5	5	5	4	4
4	4	5	5	5	5	5	5	5	5	5	5	5	5	4	4
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
5	5	5	5	6	6	6	6	6	6	6	6	6	5	5	5
4	4	5	5	5	5	5	5	5	5	5	5	5	5	4	4
4	4	5	5	5	5	5	5	5	5	5	5	5	5	4	4

付図Q. 4/JT-H263 現在の $16 \times 16$ 輝度ブロックの動きベクトルを用いた  
(ITU-T H. 263) 予測値に対する重み付け $H_0$ 。

2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1
1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1
1	1	1	1	2	2	2	2	2	2	2	2	1	1	1	1
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

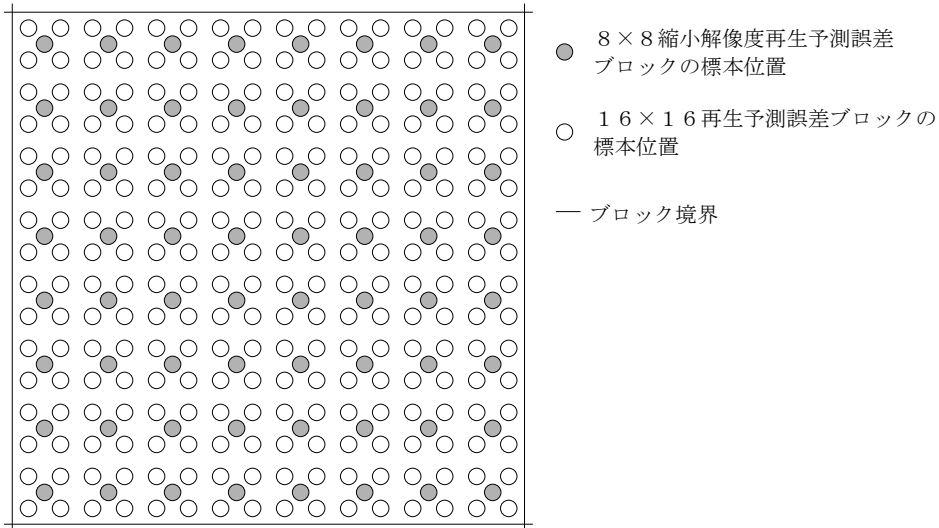
付図Q. 5 / JT-H263 現在の16×16輝度ブロックの上または下に位置する16×16輝度  
(ITU-T H. 263) ブロックの動きベクトルを用いた予測値に対する重み付けH<sub>1</sub>

2	2	1	1	1	1	1	1	1	1	1	1	1	2	2	2
2	2	1	1	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	2	2	1	1	1	1	1	1	1	1	1	2	2	2
2	2	1	1	1	1	1	1	1	1	1	1	1	2	2	2
2	2	1	1	1	1	1	1	1	1	1	1	1	2	2	2

付図Q. 6 / JT-H263 現在の16×16輝度ブロックの左または右に位置する16×16輝度  
(ITU-T H. 263) ブロックの動きベクトルを用いた予測値に対する重み付けH<sub>2</sub>

**Q. 6 縮小解像度再生予測誤差のアップサンプリング**

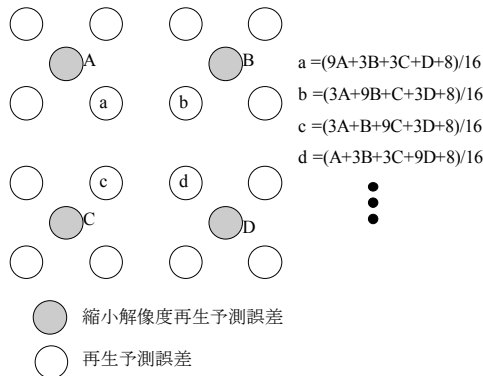
8×8縮小解像度再生予測誤差ブロックをアップサンプリングすることにより、16×16再生予測誤差ブロックが得られる。簡単な実装を実現するために、フィルタ機能はブロック内で閉じており、ブロック単位での個々のアップサンプリングが可能である。付図Q. 7 / JT-H263は標本の位置を示している。16×16再生予測誤差ブロック内部の輝度および色差画素のアップサンプリング手順は、Q. 6. 1小節で定義される。16×16再生予測誤差ブロック境界部の輝度および色差画素を生成するための手順は、Q. 6. 2小節で定義される。色差ブロックも輝度ブロック同様アップサンプリングされる。付図Q. 8 / JT-H263および付図Q. 9 / JT-H263内の“/”は、切捨ての除算を表す。



付図Q. 7 / JT-H263 8x8縮小解像度再生予測誤差ブロックおよび  
(ITU-T H.263) 16x16再生予測誤差ブロックの標本位置

**Q. 6. 1 16x16再生予測誤差ブロック内部画素のアップサンプリング手順**

ブロック内部の画素に対する再生予測誤差の生成方法を、付図Q. 8 / JT-H263に示す。“/”は切捨ての除算を表す。

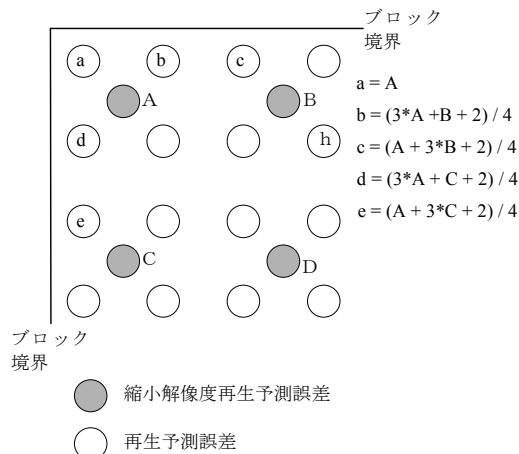


付図Q. 8 / JT-H263 ブロック内部の画素に対する再生予測誤差の生成  
(ITU-T H.263)

**Q. 6. 2 16x16再生予測誤差ブロック境界部画素に対するアップサンプリング手順**

16x16ブロックの画素に対する再生予測誤差の生成方法を、付図Q. 9 / JT-H263に示す。





付図Q. 9 / J T - H 2 6 3 ブロック境界の画素に対する再生予測誤差の生成 ( I T U - T H . 2 6 3 )

### Q. 7 ブロック境界フィルタ

復号器と同様に符号器においても、再生された  $16 \times 16$  ブロックの境界に沿って、このフィルタ処理は実施される。デブロッキングフィルタモードが使用されるか否かによる、2つの選択可能なフィルタがある。

縮小解像度更新モードにおけるデフォルトのフィルタ処理は、Q. 7. 1 小節に従って実施される。

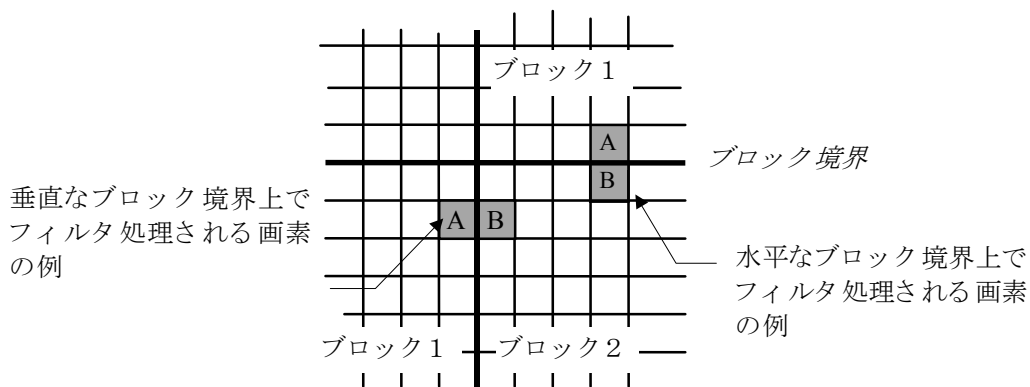
デブロッキングフィルタモードが縮小解像度更新モードと併用された場合には、Q. 7. 2 小節に従ってフィルタ処理される。

どちらの場合においても、後続の予測のためピクチャメモリにデータを保存する前に、完全に再生されたピクチャに対してフィルタ処理される。ピクチャの境界や、スライス構造モード (付属資料K参照) におけるスライスの境界、あるいは独立セグメント復号モード (付属資料R参照) においてGOBヘッダを持つGOB境界に対しては、フィルタ処理されない。色差も輝度データ同様、フィルタ処理される。

#### Q. 7. 1 デフォルトブロック境界フィルタの定義

縮小解像度更新モードにおいては、この小節に従ってデフォルトフィルタ処理される。

AおよびBが、再生ピクチャの水平あるいは垂直な同一ライン上にある2つの画素値であり、Aは  $16 \times 16$  ブロックのブロック1に属し、Bはブロック1の右または下に隣接する  $16 \times 16$  ブロックのブロック2に属すると仮定する。付図Q. 10 / J T - H 2 6 3 は、これらの画素の位置の例を表す。



付図Q. 10 / J T - H 2 6 3 デフォルトブロック境界フィルタ ( I T U - T H . 2 6 3 )

ある境界に対してフィルタ処理するためには、次の条件を満たさなければならない。

- (1) ブロック1は符号化マクロブロック (CODが0、又はMBタイプがINTRA) に属する、または
- (2) ブロック2は符号化マクロブロック (CODが0、又はMBタイプがINTRA) に属する

AはA 1に、BはB 1に置き換えられなければならない。“/”は切捨ての除算を表す。

$$A 1 = (3 \times A + B + 2) / 4$$

$$B 1 = (A + 3 \times B + 2) / 4$$

境界にフィルタ処理する順序は、付属資料JのJ. 3節の記述と同じである。

#### Q. 7. 2 デブロッキングフィルタが使用されている場合のブロック境界フィルタの定義

デブロッキングフィルタ（付属資料J参照）が縮小解像度更新モードと併用されている場合、Q. 7. 1小節で記述されたフィルタ処理の代わりに、付属資料Jで定義されるフィルタ処理に1箇所変更を施したフィルタ処理が、16×16の輝度および色差ブロックの境界画素に対して実施される。付属資料Jのフィルタ処理に対する1箇所の変更点は、パラメータSTRENGTHが正の無限大の値で与えられることである。J. 3節で定義される関数UpDownRamp(x, STRENGTH)がxの線形関数になることを意味している。

結果として、J. 3節で定義されるデブロッキングフィルタの手順は、次のように再定義される。

$$B 1 = \text{clip} (B + d 1)$$

$$C 1 = \text{clip} (C - d 1)$$

$$A 1 = A - d 2$$

$$D 1 = D + d 2$$

$$d 1 = (A - 4 B + 4 C - D) / 8$$

$$d 2 = \text{clip} d 1 ( (A - D) / 4, d 1 / 2 )$$

## 付属資料R

### 独立セグメント復号モード

(この付属資料は本標準の必須部分である)

#### R. 1 序論

この付属資料は、TTC標準JT-H263のオプションである、独立セグメント復号モードについて記述したものである。このモードでは、スライス境界を越えて、あるいは空でないGOBヘッダを持つGOBの境界を越えてデータ従属性が存在しないときに、復号することを許可する。ピクチャヘッダのPLUSPTYPEフィールドにおいて、このモードの使用を宣言する。このオプションモードを使用する能力は、外部の手段（例えば、TTC標準JT-H245）により通知される。

このモードの使用が宣言されると、ビデオピクチャセグメントの境界（スライスの境界で定義されるもの、あるいはGOBヘッダが送られたGOBの上部の境界で定義される、あるいはピクチャの境界で定義される、どれであっても最小の方法で領域に接しているもの）は、それらの境界を横切る動きベクトルの扱いを含む復号時には、ピクチャ境界として扱われる。（その境界は、無制限動きベクトルモード、拡張予測モード、デブロッキングフィルタモードあるいは時間、SNR、空間スケーラビリティモードが使用されるとき、境界での外挿補間となり、それらオプションのモードのどれも使われていないとき、禁止される。）

#### R. 2 動作モード

ビデオピクチャセグメントは以下のように定義される。

もしスライス構造モード（付属資料K参照）が使用されていないならば、1つのGOBあるいは連続する複数のGOBにより、1つのビデオピクチャセグメントは形成される。各々のビデオピクチャセグメントの先端の位置は、空でないGOBヘッダ、あるいはピクチャの先端の位置うち、後ろに位置するもので示される。空でないGOBヘッダでは、ビデオピクチャセグメントの境界は、ヘッダが存在するGOB内のマクロブロックの直前に位置する。各ビデオピクチャセグメントの後端の位置は、次のビデオピクチャセグメントの先端、あるいはピクチャの後端のうち、前に位置するもので示される。

もしスライス構造モード（付属資料K参照）が使用されていれば、各々のスライスにより、1つのビデオピクチャセグメントが形成される。

独立セグメント復号モードでは、個々のビデオピクチャセグメントは、全ての他のビデオピクチャセグメントと、完全に独立なものとして復号される。そして、（1つあるいは複数の）参照ピクチャの中では、同じ位置にあるビデオピクチャセグメントの外側にある全てのデータに対しても独立である。このことは、以下を含む。

- (1) 動きベクトル予測（6.1.1小節のような）において、現在のビデオピクチャセグメントの外側の動きベクトルは使用しない。
- (2) 拡張予測モードが使用されているとき（F.3節参照）、オーバーラップブロック動き補償に対するリモート動きベクトルとして、現在のビデオピクチャセグメントの外側の動きベクトルは使用しない。
- (3) ビデオピクチャセグメントの境界を越えて、デブロッキングフィルタは働かない（J.3節参照）。
- (4) 無制限動きベクトルモード（付属資料D参照）、拡張予測モード（付属資料F参照）、デブロッキングフィルタモード（付属資料J参照）あるいは時間、SNR、空間スケーラビリティモード（付属資料O参照）を使うことがないなら、現在のビデオピクチャセグメントの外側のデータを参照する動きベクトルは使用しない。この場合、境界外側の領域を参照する画素の予測を行うために、前のピクチャにおける現在のビデオピクチャセグメントの境界は、付属資料Dで述べられている方法で外挿補間される。
- (5) 空間スケーラビリティEI、EPピクチャ（付属資料Oで定義されるように）の上方予測のために、現在のビデオピクチャセグメントに対応する1/4サイズあるいは1/2サイズ領域の境界を越えて、双線形補間を行うことはない。
- (6) 縮小解像度更新モード（付属資料Q）が使用されているとき、ビデオピクチャセグメントの境界を越えて、ブロック境界フィルタは働かない。
- (7) 独立セグメント復号モードといっしょに、参照ピクチャ再サンプリングモードは使えない。

### R. 3 使用制約

独立セグメント復号モードを使用するときには、ビデオ符号化シンタックスの他の立場により、いくつかの制限が発生する。これらの制限は、もし制限がなければ独立セグメント復号モードの働きを難しくしてしまう2つの異常な場合を避けるために設けられている。

#### R. 3. 1 セグメント形状の制約

矩形スライスサブモード (K. 1 節) を用いないスライス構造モード (付属資料K) を使用した場合、ビデオピクチャセグメントの形状が凸でない場合 (「内側に角を持つ」あるいは2つの別個の離れた領域を含んでしまうことさえある) が起こりうる。

それゆえ、独立セグメント復号モードは、スライス構造モードの矩形スライスサブモード (付属資料K参照) を同時に使用しないスライス構造モードとは一緒に使用してはならない。この制約は、個々のビデオピクチャセグメントの外挿補間を、どのようにいつ実行すべきかということを決定するために、困難で特別な場合の扱いが必要となることを防ぐよう命じている。

#### R. 3. 2 セグメント形状変更の制約

もしビット列内でピクチャからピクチャへなんらかの方法でビデオピクチャセグメントの形状を変更することが許されるなら、ビット列の復号が困難になってしまうことが起こりうる。これは、そのような場面において、ビット列内の動きベクトルが現れる前に個々のビデオピクチャセグメントの形状を決定するには、ビット列の内容が不十分となるからである。動きベクトルを適切に解釈するためには、ビデオピクチャセグメントの形状が必要である。

それゆえ、独立セグメント復号モードでは、時間方向予測 (すなわち全てのP、B、E Pピクチャとすべての改良PBフレーム) が許可されている全てのピクチャとフレームにおけるビデオピクチャセグメントは、時間方向参照ピクチャで用いられるものと同じでなければならない。また、独立セグメント復号モードでは、すべてのE Iピクチャのビデオピクチャセグメントは、参照ピクチャにおいて用いられるビデオピクチャセグメントと同じであるか、または参照ピクチャでのビデオピクチャセグメントをさらに分割することが異なるだけでなければならない。また、現在のピクチャに対する全ての (1つあるいは複数の) 参照ピクチャで、独立セグメント復号モードが使われなければ、参照ピクチャ (INTRAを除くすべてのピクチャタイプ) を使用しているピクチャあるいはフレームにおいて、独立セグメント復号モードを使ってはならない。この制約の結果、独立セグメント復号モードでのビデオピクチャセグメントの形状は、IとE Iピクチャ (E Iピクチャ内でセグメンテーションが変化できることは、それ自身幾らか、制約を受けている) での変化を除いて、ピクチャからピクチャにおいて変化することはあってはならない。

## 付属資料S

### 代替 INTER 可変長符号モード

(この付属資料は本標準の必須部分である)

#### S. 1 序論

この付属資料は、TTC標準JT-H263のオプションである、代替 INTER 可変長符号モードについて記述したものである。このモードでは、ピクチャ内において重要な変化が明らかな時に、INTERピクチャの符号化の効率を改善する。この効率の改善は、本来 INTRAピクチャ用に設計された可変長符号を、INTERピクチャの係数とCBPYデータとに同様に使用することを許可することで得られている。ピクチャヘッダのPLUSPTYPEフィールドにおいて、このモードを使用することを宣言する。このオプションモードを使用する能力は、外部の手段(例えば、TTC標準JT-H245)により通知される。このモードでは2つのシンタックスの選択を含む。一つはINTER係数の符号化に対してであり、もう一つはINTER CBPY値に対するものである。

#### S. 2 係数用の代替 INTER 可変長符号

付属資料IにおけるINTRA可変長符号の表の設計で背景となる概念は、元のINTER可変長符号と同じ符号語を使用するがLEVELとRUNの解釈が異なる、というものである。多くのかつ／あるいは大きな値の係数があるときには、INTRA可変長符号の方がより適している。

符号語がINTERおよびINTRA両方の表において同じLAST値(0か1)を持つように、INTRA可変長符号は作られている。それゆえ、LASTが同じ値を持つ符号語の意味を「再編成」することによって、INTRAの表は作られる。さらにINTRAの表は、大きな|LEVEL|を持つイベントに対して、INTERの表の中では大きなRUNを持つ符号語を使用している。大きな値の係数をたくさん持つINTERブロックでは時々、INTERの表よりもINTRAの表を使用する方がより効果的である。そのようないくつかの場合には、可変長符号表の選択が復号器にとって明白なものとなることが出来る。なぜなら、INTERの表を用いた復号では、1つのブロックに対して64個以上の係数が存在するように見えるほどRUN値が大きくなるからである。そのような状況では、INTERの符号化の効率を改善するのに、INTRAの表を用いることができる。

##### S. 2.1 符号器の動作

復号器はその使用を検出できる時、言い換えれば、INTER可変長符号の表を用いた復号で、ブロックの64個の係数の外側の係数が指し示されてしまう時はいつでも、符号器はINTERブロックの符号化にINTRA可変長符号の表を使用してもよい。

上記の条件が満たされたときだけでなく、INTRA可変長符号の使用により同じ係数値に対してINTER可変長符号を使うよりもビット数が少なくなるときも、符号器は通常INTERブロックの符号化にINTRA可変長符号の表を用いることを選択するであろう。INTRA可変長符号が作られた方法に起因し、多くの大きな値の係数が存在するときに、しばしばこの場合に相当するだろう(なぜなら、大きな係数の振幅値を含んでいるINTRA可変長符号表における符号語と同じ符号語に対して、通常のINTER可変長符号の表では、より長いランレングスを含んでいるからである)。

##### S. 2.2 復号器の動作

復号は以下のように行われる。

- (1) 復号器はまず、ブロックの全ての係数符号を受信する。
- (2) 符号語は、INTER可変長符号が使われていると仮定して解釈される。もし係数の位置がブロックの64個の係数の内部に存在するならば、可変長符号の復号は終了する。
- (3) もし係数がブロックの外側に位置する場合には、符号語はINTRA可変長符号を用いて解釈されなければならない。

#### S. 3 CBPY用の代替 INTER 可変長符号

INTER CBPY符号語(表12/JT-H263)は、少なくとも1つの非0係数を持ったYブロックよりもすべてが0係数であるYブロックの方が多い、ということ仮定して設計されている。CBとCRとの両方のブ

ロックが少なくとも1個の非0係数を持っている場合、すなわち $CBPC_5=CBPC_6=1$ の場合、この仮定はもはや成り立たない。この理由で、代替INTER可変長符号モードが用いられているとき、 $CBPC_5=CBPC_6=1$ のときはいつでも、INTRAマクロブロックに対して表12/JT-H263で定義されているCBPY符号語が、INTERマクロブロックに対しても使用されなければならない。

**付属資料 T**  
**修正量子化モード**

(この付属資料は本標準の必須部分である)

**T. 1 序論**

この付属資料は、TTC標準 JT-H 2 6 3 のオプションである、修正量子化モードについて記述したものである。このモードでは、量子化器の動作を変更している。ピクチャヘッダの PLUS TYPE フィールドにおいて、このモードを使用することを宣言する。このオプションモードを使用する能力は、外部の手段（例えば、TTC標準 JT-H 2 4 5）により通知される。

このモードでは、4つの鍵となる特徴がある。

- (1) DQUANT フィールドのシンタックスを切り替えることにより、符号化でのビットレート制御能力が改善される。
- (2) 輝度データに対するステップサイズより細かなステップサイズを色差データに定義することにより、色差の精度が改善される。
- (3) 可能性のある係数の真値を、量子化ステップサイズで許される精度で表現できるように、係数値の表現できる範囲が拡張されている。
- (4) エラー検出を改良し復号の複雑さを軽減するために、量子化された係数レベルの範囲を合理的に起り得る範囲に制限する。

**T. 2 修正 DQUANT の更新**

このモードでは、DQUANT フィールドの意味を変更している。このモードを利用して、小さな値を加えるか減ずることにより QUANT を修正するため、またはある特定の新しい QUANT 値を知らせるために、DQUANT を用いることができる。小さな値による修正量は、現在の QUANT 値に依存する。このモードを使用することにより、DQUANT フィールドで量子化ステップサイズのより柔軟な制御を定めることができる。

このモードでの DQUANT の符号語は、もはや 2 ビットの固定長フィールドではない。2 ビットあるいは 6 ビット長の可変長フィールドである。2 ビットであるか、あるいは 6 ビットであるかは、符号の最初のビットに依存する。以下の記述は、最初のビットにより 2 つの部分に分かれる。

**T. 2. 1 小ステップ QUANT 切り換え**

DQUANT フィールドの最初のビットが“1”のとき、付加的な 1 ビットだけが DQUANT により送られる。付加的な 1 ビットは差分値として QUANT の修正に用いられる。付表 T. 1 / JT-H 2 6 3 に示すように、QUANT 値の変化は、DQUANT の 2 番目のビットと QUANT の以前の値に依存する。

例：QUANT の以前の値が 29 で、DQUANT が符号語“11”を伝えた場合、差分値は +2 となり、結果として新しい QUANT の値は 31 になる。

付表 T. 1 / JT-H 2 6 3 小ステップ QUANT 切り替えにおける意味 (ITU-T H. 2 6 3)

前の QUANT	QUANT の変化	
	DQUANT = 1 0	DQUANT = 1 1
1	+ 2	+ 1
2 - 1 0	- 1	+ 1
1 1 - 2 0	- 2	+ 2
2 1 - 2 8	- 3	+ 3
2 9	- 3	+ 2
3 0	- 3	+ 1
3 1	- 3	- 5

## T. 2. 2 任意のQUANT選択

DQUANTフィールドの最初のビットが“0”のとき、付加的な5ビットがDQUANTにより送られる。続く5ビットは、5. 1. 19小節で定義されているような新しいQUANTの値を表現する。

例：QUANTの現在の値に関わらず、もしDQUANTが符号語“001111”を知らせた場合、新しいQUANTの値は15である。

## T. 3 色差係数用の変更量子化ステップサイズ

修正量子化モードが使われているとき、色差係数の量子化パラメータは輝度の量子化パラメータと異なる。輝度の量子化パラメータはビット列の中で知らされる。これはQUANTと呼ばれる。このモードが使用されているときは、QUANT\_Cと呼ばれる異なる量子化パラメータを色差係数の逆量子化に用いる。QUANTとQUANT\_Cとの間の関係は、付表T. 2/JT-H263に示した関係にある。もしデブロッキングフィルタモード(付属資料J参照)が用いられているときは、QUANT\_Cは色差データへのデブロッキングフィルタの適用においても利用されなければならない。他のいかなる状況においてQUANTが論じられるときも、それは輝度の量子化ステップサイズを意味しなければならない。

付表T. 2/JT-H263 QUANTとQUANT\_Cとの関係  
(ITU-T H. 263)

QUANTの範囲	QUANT_Cの値
1-6	QUANT_C = QUANT
7-9	QUANT_C = QUANT - 1
10-11	9
12-13	10
14-15	11
16-18	12
19-21	13
22-26	14
27-31	15

## T. 4 修正された係数の範囲

修正量子化モードが使用されているとき、127より大きな量子化レベルの量子化DCT係数もビット列で表現することができる。これは2つの利点がある。

- (1) 表現される可能性のある係数値の真の全範囲が許されることにより、符号器の性能が改善される。
- (2) 他の方法で表現できない大きな係数値に出くわした時に、量子化ステップサイズを増加する必要性を除去することにより、符号器の複雑さを減少させる。

符号器内で量子化以前のDCT係数の正しい値は、2040の大きさになりうる。したがって、量子化パラメータQUANTあるいはQUANT\_Cが8未満のときはいつでも、存在する係数値の全ての範囲をLEVELの-127から+127でカバーするのは不十分である。真の係数値がより適切に符号化されるように、拡張された係数範囲はLEVELの範囲を広げる。

修正量子化モードが使用されているとき、ESCAPE符号(5. 4. 2小節に従い、“0000011”)に続くLEVELフィールドの意味を変更する。このモードでは、ビットシーケンス“10000000”は禁じられているのではなく、EXTENDED\_ESCAPEを表すのに用いられる。127を超える大きさのAC係数は、11ビット固定長のEXTENDED\_LEVELフィールドを直後に伴うEXTENDED\_ESCAPE符号を送信することで表現される。拡張された係数値は、LEVELを2の補数表現し、そこから下位11ビットを取り出し、5ビット右に循環ローテーションして、EXTENDED\_LEVELフィールドに符号化される。このローテーションは、スタートコードエミュレーションを防ぐのに必要である。循環ローテーションは、付図T. 1/JT-H263に示してある。



LEVELフィールドのビット

b 1 1	b 1 0	b 9	b 8	b 7	b 6	b 5	b 4	b 3	b 2	b 1
-------	-------	-----	-----	-----	-----	-----	-----	-----	-----	-----

EXTENDED-LEVELフィールドのビット

b 5	b 4	b 3	b 2	b 1	b 1 1	b 1 0	b 9	b 8	b 7	b 6
-----	-----	-----	-----	-----	-------	-------	-----	-----	-----	-----

付図T. 1 / JT-H 2 6 3 係数の表現における循環ローテーション  
(ITU-T H. 2 6 3)

## T. 5 使用制約

修正量子化モードを使用しているとき、符号化係数値にいくつかの制限が生じる。  
これにより、いくつかの利点がある。

- (1) 特定の不当な係数値を禁止することにより、ビットエラーの検出が改良される。したがって、そのような値は復号器でビットエラーとして認識することができる。
- (2) クリッピング以前の逆量子化で必要となるワード長が短くなるため、復号器の複雑さが軽減される。
- (3) T. 4 節で述べたEXTENDED-ESCAPEの方法を係数符号化に用いることで、スタートコードエミュレーションを防ぐことができる。

制限事項は、以下の通りである。修正量子化モードを使っている時に、

- (1) 任意の係数に対し、QUANTあるいはQUANT\_\_Cの現在値を適切に用いて6. 2. 1 小節で述べられた逆量子化プロセスにより生成された再生レベル振幅|REC|、及び符号化されたLEVELの値は、4096より小さくならなければならない。EXTENDED-ESCAPEのメカニズムを用いて係数が送られるか否かに関わらず、この追加の制限事項はすべての係数に適用される。
- (2) 適用可能な可変長符号の表である表16 / JT-H 2 6 3 (5. 4. 2 小節) あるいは付表I. 2 / JT-H 2 6 3 (I. 3 節) にある登録された符号語がある、LAST、RUN、LEVELの組の符号化を行うのに、ビット列は通常のESCAPE符号あるいはEXTENDED-ESCAPE符号を使用してはならない。
- (3) 係数の量子化パラメータ (QUANTあるいはQUANT\_\_C) が8未満であるときだけ、EXTENDED-ESCAPEを使用しなければならない。
- (4) 範囲-127から+127の外側にあるLEVELの値を表現するEXTENDED-LEVELフィールドが後ろに続くときにのみ、EXTENDED-ESCAPEを使わなければならない。

## 付属資料U

### 拡張参照ピクチャ選択モード

(この付属資料は本標準の必須部分である)

#### U. 1 序論

この付属資料では、本標準にあるオプションの拡張参照ピクチャ選択 (ERPS) モードについて述べる。このオプションモードを用いることができるかどうかは、外部の手段 (例えば、TTC標準 JT-H 2 4 5) によって交渉される。ERPS 処理のために復号器に実装されるピクチャメモリの大きさも、外部の手段によって通知されるべきである。PLUSPTYPEのオプション部分 (OPPTYPE) の予め予約された第16ビットを“1”にセットすることにより、このモードを使用することが示されなければならない。このモードは、参照ピクチャのメモリバッファを用いることにより、誤り耐性及び符号化効率の双方に対して有効である。

ERPSモードのサブモードの1つにサブピクチャ除去がある。サブピクチャ除去の目的は、複数参照ピクチャを格納するのに必要なメモリ量を削減することである。メモリの削減はそれぞれの参照ピクチャをサブピクチャと呼ばれるより小さな矩形単位に分割することによって実現される。符号器は復号器に対して、特定の参照ピクチャの特定のサブピクチャ領域が後続のピクチャの予測の参照として用いられないことを指示することができる。それゆえ、復号器に実装されたこれらの領域を格納するためのメモリを、他の参照ピクチャからのデータの格納に用いることができる。このサブモードと、ここで定義されたサブピクチャ除去のための、ピクチャメモリの最小ピクチャ単位 (MPUs) への分割のサポートは、外部の手段 (例えば、TTC標準 JT-H 2 4 5) によって交渉される。

ERPSモードのサブモードの1つは、Bピクチャにおいて2つのピクチャの後方予測を可能にする。このサブモードは、前方予測に複数の参照を用いるだけでなく、後方予測に2つ以上の参照ピクチャを用いる能力をBピクチャのために符号器に与えることによって性能を高めることができる。このモードのサポートは、外部の手段 (例えば、TTC標準 JT-H 2 4 5) によって交渉される。

誤り耐性のために、ERPSモードは逆方向チャンネルメッセージを用いることができる。これは、どのピクチャが、あるいはピクチャのどの部分が誤って復号されたかを符号器に知らせるために、復号器から符号器へ送られる外部の手段 (例えば、TTC標準 JT-H 2 4 5) によって通知される。ERPSモードは、付属資料Nで定義されている参照ピクチャ選択 (RPS) モードと比べて、より性能が強化されている。ERPSモードはRPSモードと同時に用いてはならない (ERPSモードは、RPSモードと本質的に同様の機能を提供するように使うことができる。)

符号化効率を上げるために、動き補償を複数のピクチャからの予測に拡張することができる。マルチピクチャ予測への動き補償の拡張は、複数参照ピクチャのいずれかにおける動き補償のためのマクロブロックあるいはブロック予測領域を識別するのに用いられるピクチャ参照パラメータによって、各動きベクトルを拡張することで実現される。ピクチャ参照パラメータは、相対的なバッファインデックスを示す可変長符号である。参照ピクチャは、符号器によって制御されるバッファリング方式により蓄積される。

シンタックス算術符号化モード (付属資料E参照) またはデータパーティションスライスモード (付属資料V参照) において、ERPSモードを用いてはならない。

ERPSモードを一旦アクティブにしたら、IピクチャかE Iピクチャで最初に非アクティブにしてから、IピクチャかE Iピクチャでバッファリセット (RESET = “1”) によって再びアクティブにしない限り、ビット列中の後続のピクチャにおいてERPSモードを非アクティブにしてはならない。非アクティブにした場合、ERPSマルチピクチャバッファの内容全体を“使用されない”状態にしなければならない。

#### U. 2 ビデオ情報源符号化アルゴリズム

このモードの情報源符号器の一般的な形態を付図U. 1 / JT-H 2 6 3に示す。この図では、複数のピクチャメモリを用いた構成を示している。

このビデオ情報源符号化アルゴリズムは、マルチピクチャ動き補償への拡張が可能である。マクロブロックレベルでの参照ピクチャの選択を許容することで、符号化効率の改善が達成されるであろう。相対的インデックスを用いたピクチャバッファリング方式は、このマルチピクチャバッファにおけるピクチャの効率的なアドレス指定に用いられる。このマルチピクチャバッファの制御は、異なる2種類の操作で動作してもよい。

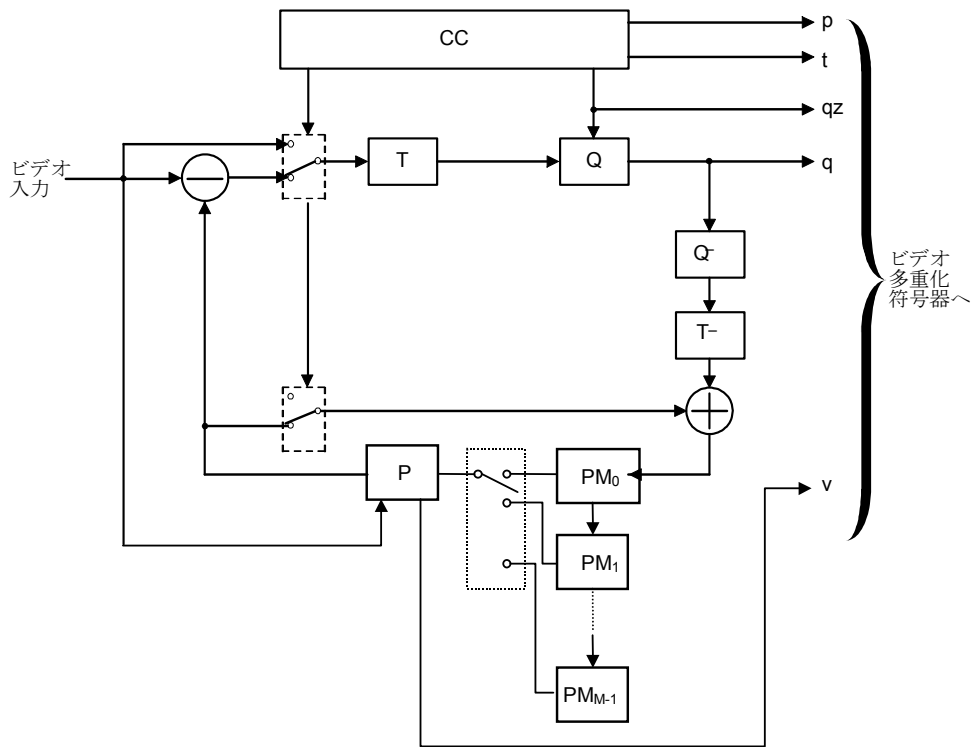
2種類の操作のうちの1番目の、時間に対する“スライディングウィンドウ”は、バッファ制御装置によって実現することができる。M個のピクチャメモリ  $PM_0, \dots, PM_{M-1}$  を用いるバッファリング方式において、最も

最近に復号され再構築された（M個までの）ピクチャが、ピクチャメモリに格納され、復号のための参照として用いることが可能である。マルチピクチャバッファに蓄積できる最大ピクチャ数がMで、 $m$ が $0 \leq m \leq M-1$ であるなら、ピクチャ $m$ を符号化する時の動き推定には $m$ 個のピクチャが利用可能である。また、 $m \geq M$ のとき、ピクチャ $m$ の符号化にはピクチャの最大数Mが利用可能である。一方、2番目の“適応メモリ制御”操作は、単純な“スライディングウィンドウ”方式を用いた場合よりも柔軟で特殊なピクチャメモリの制御のために用いることが可能である。

このERPSモードの操作により、復号器に送信済の複数のピクチャ、あるいは複数のサブピクチャ領域に“使用されない”状態を割り当てる。一旦、あるピクチャ、あるいは、ピクチャのある領域に“使用されない”状態が割り当てられたならば、ビット列は、それ以降のピクチャの予測にその“使用されない”領域を参照するいかなるデータも含んではならない。符号器は、先行するピクチャの“使用されない”状態の割り当てを管理することで、後続のピクチャの表示のために必要な全てのデータを格納できる十分なメモリが復号器に残っていることを保証しなければならない。全バッファサイズとその構成は、ビット列によって復号器へ伝えられる。さらに符号器は、“使用されない”状態が割り当てられていない格納されたピクチャデータによって、指定された総容量を超えないようバッファを制御しなければならない。

この情報源符号器は、INTERピクチャの符号化により引き起こされる時間方向の誤り伝播を抑制するために1つあるいは複数のピクチャメモリを選択してもよい。このモードが、例えばGOBやスライスのようなピクチャより小さい単位に適用されるとき、GOBまたはスライスの境界を越える動き補償による空間方向の誤り伝播を避けるために、空でないヘッダをもつ、GOBやスライスの境界をピクチャ境界として扱う独立セグメント復号モード（付属資料R参照）を用いることが可能である。予測のためにどのピクチャが選択されたかを示す情報は、符号化されたビット列に含まれる。

予測のためにピクチャあるいは複数のピクチャを符号器が選択する方法については、本標準の範囲外である。



- T 変換器
- Q 量子化器
- P 動き補償用可変遅延機能を持つピクチャメモリ
- PM ピクチャメモリ
- CC 符号化制御
- p INTRA/INTER 識別フラグ
- t 伝送/非伝送識別フラグ
- qz 量子化器インジケーション
- q 変換係数の量子化出力インデックス
- v 動きベクトル

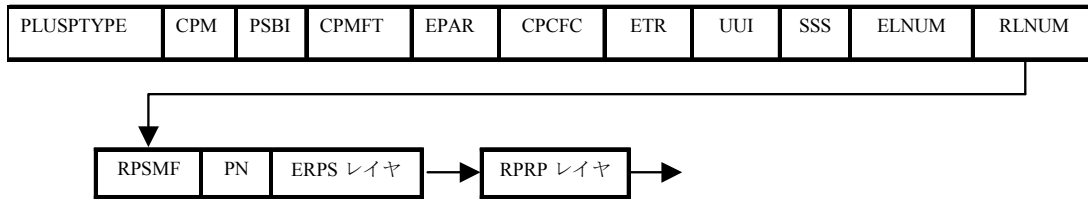
付図U. 1 / JT-H 263 拡張参照ピクチャ選択モードの情報源符号器  
(ITU-T H. 263)

### U. 3 順方向チャンネルのシンタックス

このシンタックスは、ピクチャ、グループオブブロック (GOB)、スライスレイヤで異なる。パラメータMRPAに“1”がセットされている場合、このシンタックスはマクロブロックレイヤでも異なる。ピクチャ、GOB、スライスレイヤにおいては、拡張参照ピクチャ選択レイヤ (ERPSレイヤ) が挿入される。マクロブロックレイヤにおいては、マルチピクチャ動き補償を可能にする条件の下で、ピクチャ参照パラメータが挿入される。

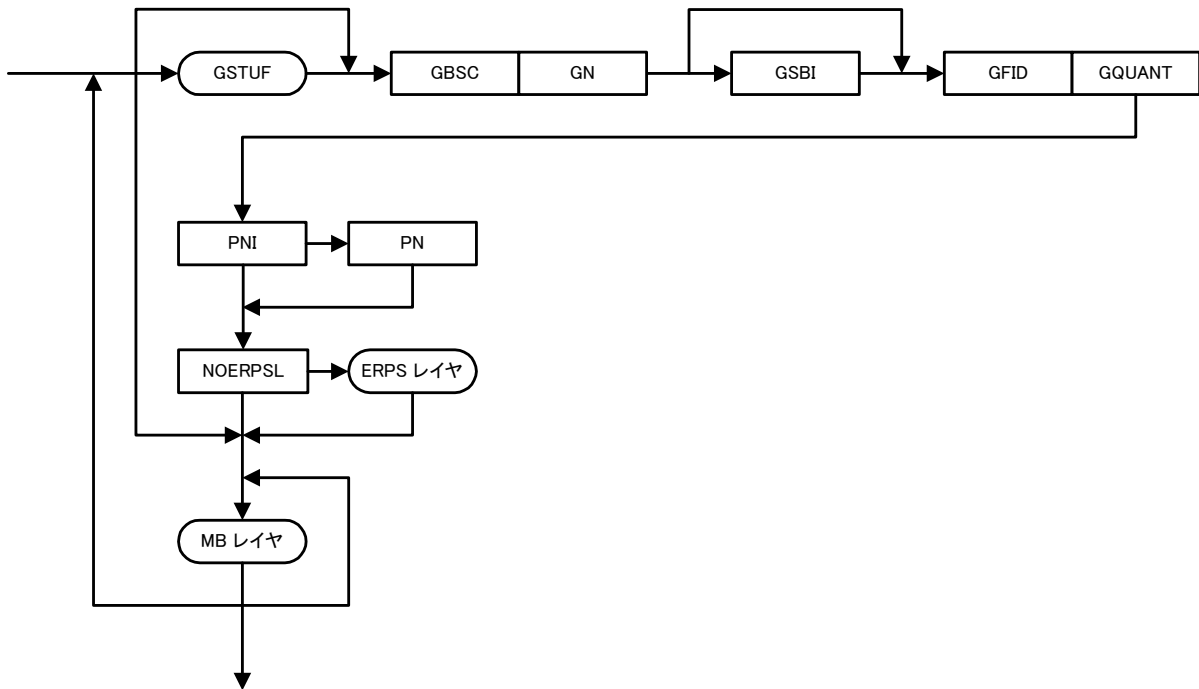
#### U. 3.1 ピクチャ、GOB、スライスレイヤのシンタックス

PLUSヘッダにおける拡張参照ピクチャ選択のシンタックスを付図U. 2 / JT-H 263に示す (その他は図8 / JT-H 263に示される)。RPSMF、PNフィールド、およびERPSレイヤが、PLUSヘッダの中に挿入される。TRPI、TRP、BCI、およびBCMフィールドは存在しない (これらは、ERPSモードがアクティブな場合は許可されない付属資料NのRPSモードでのみ必要であるため)。



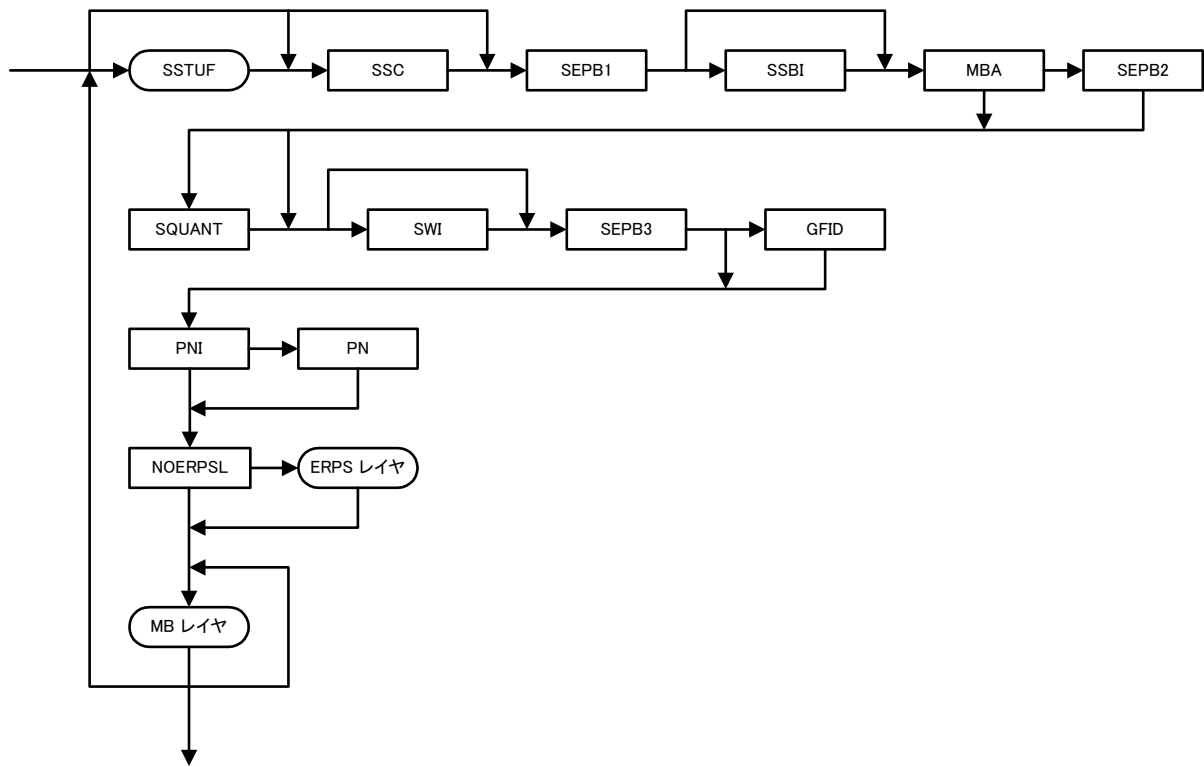
付図U. 2 / JT-H263 ERPSモードのPLUSヘッダ構造  
(ITU-T H. 263)

GOBレイヤのシンタックスを付図U. 3 / JT-H263に示す。PNI、PN、NOERPSLフィールドおよびERPSレイヤがシンタックスに追加される（その他は図9 / JT-H263で定義）。



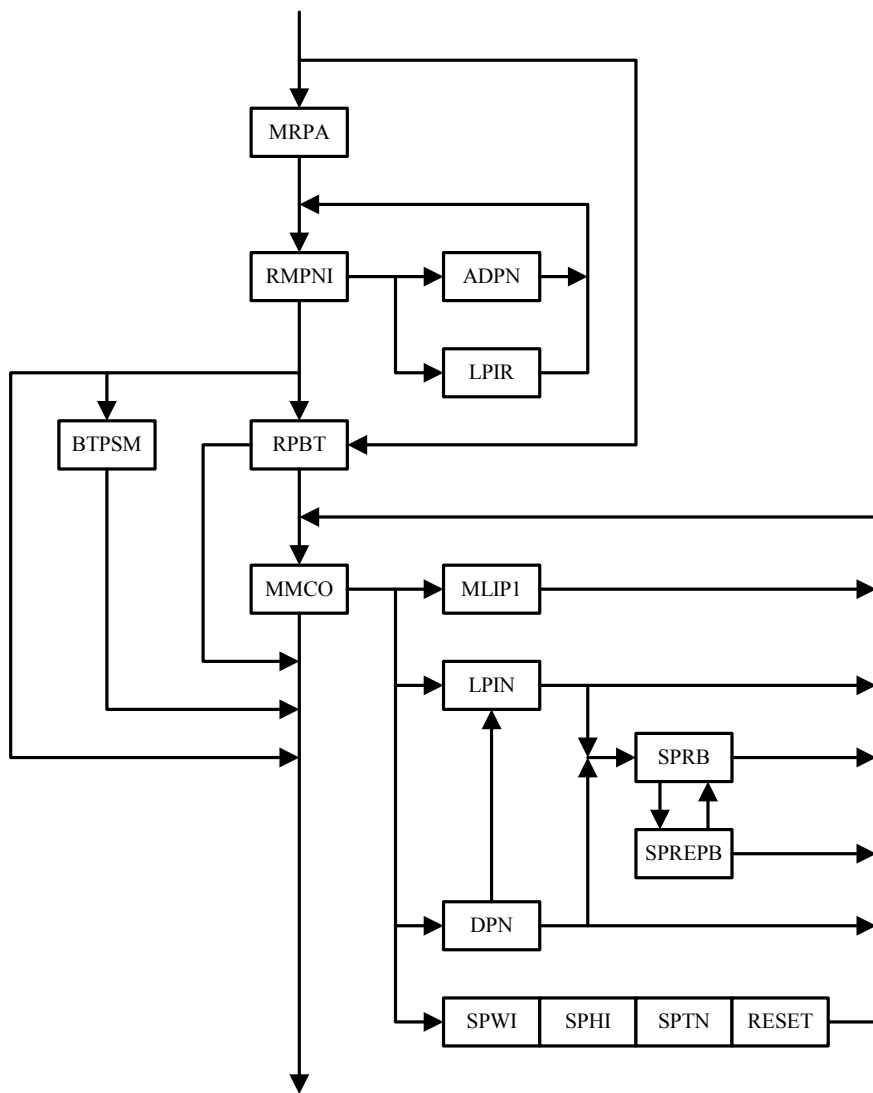
付図U. 3 / JT-H263 ERPSモードのGOBレイヤ構造  
(ITU-T H. 263)

オプションのスライス構造モード（付属資料K参照）を使用する場合、スライスレイヤのシンタックスはGOBレイヤと同じ方法で修正される。スライスレイヤのシンタックスを付図U. 4 / JT-H263に示す。ビット列中のピクチャスタートコードの直後に続くスライスもまた、追加されたPNI、PN、NOERPSLフィールドおよびERPSレイヤの全てを含む。



付図U. 4 / JT-H263 ERPSモードのスライスレイヤ構造  
(ITU-T H. 263)

ERPSレイヤを付図U. 5 / JT-H263に示す。



付図U. 5 / JT-H263 ERPSレイヤ構造  
(ITU-T H. 263)

ADPN、LPIR、MLIP1、DPN、LPIN、SPTN、PR、PR<sub>0</sub>、PR<sub>2</sub>、PR<sub>3</sub>、PR<sub>4</sub>、PR<sub>B</sub>およびPRFWフィールドの可変長符号は付表U. 1 / JT-H263で与えられる。

付表U. 1 / JT-H263 ADPN、LPIR、MLIP1、DPN、LPIN、SPTN、PR、PR<sub>0</sub>、  
(ITU-T H. 263) PR<sub>2</sub>、PR<sub>3</sub>、PR<sub>4</sub>、PR<sub>B</sub>、PRFWの可変長符号

絶対位置	ビット数	符号
0	1	1
"x <sub>0</sub> " + 1 (1:2)	3	0x00
"x <sub>1</sub> x <sub>0</sub> " + 3 (3:6)	5	0x1x <sub>0</sub> 0
"x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 7 (7:14)	7	0x21x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 15 (15:30)	9	0x31x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 31 (31:62)	11	0x41x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 63 (63:126)	13	0x51x <sub>4</sub> 1x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>6</sub> x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 127 (127:254)	15	0x61x <sub>5</sub> 1x <sub>4</sub> 1x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>7</sub> x <sub>6</sub> x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 255 (255:510)	17	0x71x <sub>6</sub> 1x <sub>5</sub> 1x <sub>4</sub> 1x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>8</sub> x <sub>7</sub> x <sub>6</sub> x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 511 (511:1022)	19	0x81x <sub>7</sub> 1x <sub>6</sub> 1x <sub>5</sub> 1x <sub>4</sub> 1x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>9</sub> x <sub>8</sub> x <sub>7</sub> x <sub>6</sub> x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 1023 (1023:2046)	21	0x91x <sub>8</sub> 1x <sub>7</sub> 1x <sub>6</sub> 1x <sub>5</sub> 1x <sub>4</sub> 1x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0
"x <sub>10</sub> x <sub>9</sub> x <sub>8</sub> x <sub>7</sub> x <sub>6</sub> x <sub>5</sub> x <sub>4</sub> x <sub>3</sub> x <sub>2</sub> x <sub>1</sub> x <sub>0</sub> " + 2047 (2047:4094)	23	0x101x <sub>9</sub> 1x <sub>8</sub> 1x <sub>7</sub> 1x <sub>6</sub> 1x <sub>5</sub> 1x <sub>4</sub> 1x <sub>3</sub> 1x <sub>2</sub> 1x <sub>1</sub> 1x <sub>0</sub> 0

#### U. 3. 1. 1 参照ピクチャ選択モードフラグ (RPSMF) (3ビット)

RPSMFは、(UFEFの値に関わらず) ERPSモード使用時に必ずPLUSヘッダ中に現れる3ビットの固定長符号語である。RPSMFはGOBやスライスレイヤに存在してはならない。このフラグが存在するとき、RPSMFは符号器がどのタイプの逆方向チャンネルメッセージを必要としているかを示す。RPSMFの値は5.1.1.3小節の定義に従わなければならない。

#### U. 3. 1. 2 ピクチャ番号インジケータ (PNI) (1ビット)

PNIは1ビットの固定長符号語で、ERPSモードが使用されるときGOBやスライスレイヤには必ず存在し、PLUSヘッダには存在しない。PNIは後続のPNフィールドが存在するか否かを示す。

- “0” : PNフィールドは存在しない
- “1” : PNフィールドが存在する

#### U. 3. 1. 3 ピクチャ番号 (PN) (10ビット)

PNは10ビットの固定長符号語であり、ERPSモードが使われているときはPLUSヘッダに常に存在し、PNIによって示される時のみGOBやスライスレイヤに存在する。

PNは、符号化され、かつ伝送された全てのピクチャについて、直前に蓄積されたピクチャのPNに対し、10ビットのモジュロ演算で値を1だけ増加させなければならない。“蓄積されたピクチャ”は、U. 3. 1. 5. 7小節で定義される。EIおよびEPピクチャについては、同一のスケラビリティ高品質化レイヤにおける直前に蓄積されたEIまたはEPピクチャの値からPNを増加させなければならない。Bピクチャについては、このBピクチャの参照レイヤ内で時間的に最も新しく蓄積された非Bピクチャ(ビット列順序ではこのBピクチャより前にあり、時間的にはこのBピクチャの後のピクチャ)の値からPNを増加させなければならない。Bピクチャは後続のピクチャの参照として用いられることはないため、マルチピクチャバッファには蓄積されない。従って、Bピクチャの参照レイヤにおけるこのBピクチャの直後のピクチャ、または同じ高品質化レイヤにおいてあるBピクチャの直後の別のBピクチャは、このBピクチャと同じPNを持たねばならない。同様に、蓄積されない非Bピクチャがビット列中に存在する場合、(付属資料Oの操作の場合は、同じ高品質化レイヤにおいて)この非Bピクチャに続くピクチャも蓄積されない非Bピクチャと同じPNを持たねばならない。

“ビデオ冗長符号化”として知られる使用シナリオにおいては、ERPSモードは、同一時刻の画像シーンについて2つ以上の表現(たいていの場合、異なる参照ピクチャを用いる)を送出する方法を、符号器で用いてもよい。ERPSモードを用いる場合、および同じテンポラルリファレンスかつ同じピクチャ番号をビット列内の隣接ピクチャが持つような場合、復号器はこれをほぼ同一の画像シーン内容の冗長なコピーが送られたとみなし、最初に受信したピクチャを復号し使用するが、後続する冗長なピクチャは破棄しなければならない。

PNは、マルチピクチャバッファに蓄積された各ピクチャに対して(付属資料Oの操作の場合は特定の高品質化レイヤについて)、1024枚の符号化および蓄積されたピクチャの範囲内で固有のIDとして働く。よって、以下に記載される長期間ピクチャのインデックスが割り当てられない限り、(付属資料Oの操作の場合は同じ高品質化レイヤにおいて)あるピクチャの1024枚以上の後続するピクチャが符号化および蓄積された後に、そのピクチャがバッファに残っていることはない。符号器は、ビット列が1024枚以上の後続の蓄積されたピクチャの後にいかなる短期間ピクチャの保持も示唆しないことを保証しなければならない。復号器は、現在のピクチャのピクチャ番号が、マルチピクチャバッファに短期間蓄積されたある他のピクチャのピクチャ番号と同じであることを発見した場合(付属資料Oの操作での同じ高品質化レイヤの場合、また前の段落で記述したビデオ冗長符号化を除いた場合において)、この状態を誤りとして取り扱うべきである。

#### U. 3. 1. 4 拡張参照ピクチャ選択レイヤ無効 (NOERPSL) (1ビット)

NOERPSLは1ビットの固定長符号語で、ERPSモードが使用されるときには必ずGOBまたはスライスレベルに存在する。PLUSヘッダには存在しない。NOERPSLの値は以下でなければならない。

- “0” : ERPSレイヤが送信される
- “1” : ERPSレイヤは送信されない



NOERPSLが“1”の場合には、GOBまたはスライスレイヤのデータに続く関連するビデオピクチャセグメントに対しても、このピクチャに対して有効な全てのERPS設定および再マッピングを適用しなければならない。GOBまたはスライスレベルで送信されるERPSレイヤ情報は、GOBまたはスライスレベルのデータに続くビデオピクチャセグメントに対する復号処理を支配し、他のいずれのビデオピクチャセグメントの復号処理にも影響を及ぼさない（ビデオピクチャセグメントの定義については付属資料Rを参照）。

#### U. 3. 1. 5 拡張参照ピクチャ選択レイヤ (ERPS) (可変長)

ERPSレイヤは、ERPSモードが使用されるときにはピクチャレベルに常に存在し、また、NOERPSLが“0”の場合にはGOBまたはスライスレベルに存在する。ERPSレイヤは現在のピクチャまたはビデオピクチャセグメントを復号するために使用されるバッファのインデックスを指定し、ピクチャバッファの内容を管理する。

##### U. 3. 1. 5. 1 複数参照ピクチャアクティブ (MRPA) (1ビット)

MRPAは1ビットの固定長符号語で、ピクチャ符号化タイプがPピクチャ、EPピクチャ、改良PBフレームまたはBピクチャの場合にのみ存在する。もし存在する場合は、MRPAはERPSレイヤにおける最初の要素である。MRPAは、現在のピクチャまたはビデオピクチャセグメントの前方予測復号または後方予測復号のためのアクティブな参照ピクチャ数が1より大きいか否かを指定する。MRPAの値は以下でなければならない。

“1”：2つ以上の参照ピクチャが前方または後方動き補償に使用され得る。

“0”：1つの参照ピクチャのみが前方または後方動き補償に使用される。この場合、U. 3. 2小節に指定されるマクロブロックレイヤのシンタックスの拡張は適用されない。

MRPAは、異なるビデオピクチャセグメントが異なる数の参照ピクチャを指示し得るように、ビデオピクチャセグメント毎に変更されてもよい。

MRPAは、参照ピクチャ再サンプリングモード（付属資料Pを参照）を起動するどのピクチャにおいても“0”でなければならない。また、このような現在のピクチャに対してはいずれも、ピクチャおよびGOBまたはスライスレベルにおいて使用される前方参照ピクチャとして、同一のピクチャを指示しなければならない。現在のピクチャがBピクチャの場合、後方参照ピクチャは現在のピクチャと同一のサイズでなければならない。参照ピクチャの再サンプリング処理はいずれも、前方参照ピクチャに対してのみ適用されねばならない。再サンプリングされた前方参照ピクチャを格納するための十分な“使用されない”容量がマルチピクチャバッファにある場合にのみ、参照ピクチャ再サンプリングを起動させなければならないが、再サンプリングされた参照ピクチャが現在のピクチャの復号処理に使用された後には、再サンプリングされた前方参照ピクチャをマルチピクチャバッファに格納してはならない。

##### U. 3. 1. 5. 2 ピクチャ番号インジケータの再マッピング (RMPNI) (可変長)

RMPNIは可変長符号語で、そのピクチャがP、EP、改良PB、またはBピクチャである場合にERPSレイヤに存在する。RMPNIは、任意のデフォルトのピクチャインデックスが現在のピクチャ、またはビデオピクチャセグメントの動き補償に対して再マッピングされているかどうか、およびマルチピクチャバッファ中の相対的インデックスの再マッピングがもし表示されているならどのように定められているのかを示す。RMPNIは付表U. 2/JT-H263を使って伝送される。もしRMPNIがADPNもしくはLPiRフィールドの存在を示すなら、追加のRMPNIフィールドがADPNもしくはLPiRフィールドの直後に続く。

ピクチャ参照パラメータは、整列されたピクチャのセットに対する相対的インデックスである。RMPNI、ADPN、およびLPiRフィールドは、特定のピクチャまたはビデオピクチャセグメントの復号のため、デフォルトのインデックスによる順序から一時的に変更されるマルチピクチャバッファ内の相対的インデックスによる順序を許す。デフォルトのインデックス順序では、短期間ピクチャ（つまり、長期間インデックスを与えられないピクチャ）は、参照インデックス順序において長期間ピクチャに先行する。短期間ピクチャにおいて、デフォルトの順序は、最も新しくバッファリングされた参照ピクチャに始まり、最も古い参照ピクチャまで続くようピクチャに付けられた順序である（すなわち、10ビット長のピクチャ番号フィールドに変換されていないピクチャ番号の降順で）。長期間ピクチャにおいて、デフォルトの順序は、最も小さい長期間インデックスを持つピクチャに始まり、

長期間インデックスがMLIP1の最新の値-1と等しくなるピクチャまで続くようピクチャに付けられた順序である。

例えば、パッファが、(ピクチャ番号の昇順に伝送された) 短期間ピクチャ番号が300、302、303である3つの短期間ピクチャと、長期間ピクチャインデックスが0と3である2つの長期間ピクチャを含むとすると、デフォルトのインデックス順序は、以下の通りとなる。

- ・ デフォルト相対的インデックス0はピクチャ番号303を持つ短期間ピクチャを参照する。
- ・ デフォルト相対的インデックス1はピクチャ番号302を持つ短期間ピクチャを参照する。
- ・ デフォルト相対的インデックス2はピクチャ番号300を持つ短期間ピクチャを参照する。
- ・ デフォルト相対的インデックス3は長期間ピクチャインデックスが0となる長期間ピクチャを参照する。
- ・ デフォルト相対的インデックス4は長期間ピクチャインデックスが3となる長期間ピクチャを参照する。

(もしあれば) 受信した1番目のADPNまたはLPIRフィールドは、特定のピクチャをデフォルトの順序から相対的インデックス0に移動させる。2番目のそのようなフィールドは特定のピクチャを相対的インデックス1に移動させる、以下同等。この方法で相対的インデックス順序の先頭に移動されないままの一連のピクチャは、それら自体の中でデフォルトの順序を保持し、パッファの先頭に相対的インデックス順序で動かされたピクチャに続かなければならない。

もしMRPAが“0”なら、現在のピクチャがBピクチャである場合を除いて、1つより多いADPNやLPIRフィールドが同じERPSレイヤに存在してはならない。もし、現在のピクチャがBピクチャで、MRPAが“0”なら、2つより多いADPNやLPIRフィールドが同じERPSレイヤに存在してはならない。

あるピクチャに指定されるピクチャ番号の再マッピングはいずれも、他のどのピクチャの復号処理にも影響を及ぼしてはならない。あるビデオピクチャセグメントに指定されたピクチャ番号の再マッピングは、他のどのビデオピクチャセグメントの復号処理にも影響を及ぼしてはならない。あるピクチャに指定されるピクチャ番号の再マッピングは、そのピクチャ内の任意のビデオピクチャセグメントの復号処理にのみ、次の2つの方法で影響を及ぼさなければならない。

- ・ GOBもしくはスライスレベルにおいてNOERPSLが“1”ならば、ピクチャレベルで指定される再マッピングは対応するビデオピクチャセグメントにおいても使われる。
- ・ もしピクチャがBピクチャならば、ピクチャレベルで指定される再マッピングは直接双方向予測のためのTR<sub>B</sub>とTR<sub>D</sub>の値の計算を指定しなければならない。

RMPNI “エンドループ” 指示は、もしMRPAが“0”ならBピクチャのERPSレイヤの最終要素である。MRPAが“1”であるBピクチャにおいては、RMPNI “エンドループ” 指示の後にBTPSMが続く。PまたはEPピクチャまたは改良PBフレームにおいては、RMPNI “エンドループ” 指示の後にRPBTが続く。

1つのERPSレイヤ内において、RMPNIは、1つの参照ピクチャを相対的インデックス順序において1つより多い再マッピング位置に配置するような指定をしてはならない。

付表U. 2/JT-H263 参照ピクチャの再マッピングのためのRMPNI操作  
(ITU-T H. 263)

値	指定された再マッピング
‘1’	ADPNフィールドが存在し、ピクチャ番号予測値に加えるための負の差分に相当する
‘010’	ADPNフィールドが存在し、ピクチャ番号予測値に加えるための正の差分に相当する
‘011’	LPIRフィールドが存在し、参照ピクチャのための長期間インデックスを指定する
‘001’	デフォルトの順序により相対的インデックス付けされたピクチャの再マッピングのためのエンドループ

### U. 3. 1. 5. 3 ピクチャ番号の絶対差分 (ADPN) (可変長)

ADPNは可変長符号語で、RMPNIにより指示されている場合にのみ存在する。RMPNIが存在する時、ADPNはRMPNIに続く。ADPNは付表U. 1/JT-H263を使って伝送され、表中のインデックスにおいてADPN-1に相当する。ADPNは最も新しく再マッピングしたピクチャのピクチャ番号とピクチャ番号の予測値との間の絶対差分を表す。もし現在のERPSレイヤ内で前にADPNフィールドが送られていなければ、予測値は現在のピクチャのピクチャ番号でなければならない。もし前にADPNフィールドが送られていたなら、予測値はADPNを使って再マッピングされた最後のピクチャのピクチャ番号でなければならない。

ピクチャ番号の予測をPNPとし、求めようとするピクチャ番号をPNQとすると、復号器は以下と数学的に等価な手法によって、PNPとADPNからPNQを決定しなければならない。

```
if (RMPNI == "1") { //負の差分
  if (PNP - ADPN < 0)
    PNQ = PNP - ADPN + 1024;
  else
    PNQ = PNP - ADPN;
} else { //正の差分
  if (PNP + ADPN > 1023)
    PNQ = PNP + ADPN - 1024;
  else
    PNQ = PNP + ADPN;
}
```

符号器は、ADPNの復号値が1024以上にならないようにRMPNIとADPNを制御しなければならない。

実施例として、符号器は、求めようとする再マッピングされたピクチャ番号PNQを指定するためのADPNとRMPNIの値を決定するために、次の処理を使ってもよい。

```
DELTA = PNQ - PNP;
if (DELTA < 0) {
  if (DELTA < -511)
    MDELTA = DELTA + 1024;
  else
    MDELTA = DELTA;
} else {
  if (DELTA > 512)
    MDELTA = DELTA - 1024;
  else
    MDELTA = DELTA;
}
ADPN = abs (MDELTA);
```

ここで、abs () は絶対値演算を示す。付表U. 1/JT-H263におけるインデックスは、ADPNそのものでなくADPN-1の値に相当することに注意。次に、RMPNIがMDELTAの符号により決定される。

### U. 3. 1. 5. 4 再マッピングのための長期間ピクチャインデックス (LPIR) (可変長)

LPIRは可変長符号語でRMPNIによって指示されている場合にのみ存在する。LPIRが存在する時、LPIRはRMPNIに続く。LPIRは付表U. 1/JT-H263を使って伝送される。それは再マッピングされるための長期間ピクチャインデックスを示す。任意の後続のADPNの再マッピングにより使用される予測値は、LPIRによって影響を受けない。

### U. 3. 1. 5. 5 Bピクチャ2ピクチャ予測サブモード (BT PSM) (1ビット)

BT PSMは1ビット固定長符号語で、Bピクチャ(付属資料O参照)の場合のみ、かつ、MRPAが“1”の場合にのみ存在する。BT PSMが存在する場合、RMPNI“エンドループ”指示に続き、BピクチャのERPSレイヤの最終要素となる。BT PSMは2ピクチャ後方予測サブモードがそのピクチャの中で使われているかどうかを以下のように示す。

“0” : 単一ピクチャの後方予測

“1” : 2ピクチャ後方予測

BT PSMは存在しない場合(MRPAが“0”の時)、“0”の暗黙の値を持つ。

前方予測参照として利用できるピクチャセットは、マルチピクチャバッファ内の後方参照ピクチャセット以外のピクチャセットである。後方参照ピクチャセットはBT PSMの値より決定される。もし、単一ピクチャ後方予測がBT PSMにより指定されるならば、(おそらく再マッピングされた)相対的インデックス順序で最初のピクチャは唯一の後方参照ピクチャである。もし2ピクチャ後方予測がBT PSMにより指定されるなら、(おそらく再マッピングされた)相対的インデックス順序で最初の2つのピクチャは2つの後方参照ピクチャである。また前方予測のための相対的インデックスは、前方参照ピクチャセットへの相対的インデックスとなる。

マルチピクチャバッファの内容は、Bピクチャの存在により影響を受けない。Bピクチャはマルチピクチャバッファに蓄積されず、以後のピクチャの符号化の参照として使用されない。

### U. 3. 1. 5. 6 参照ピクチャバッファリングタイプ (RPBT) (1ビット)

RPBTは1ビットの固定長符号語で、現在復号されたピクチャのバッファタイプを指定する。ピクチャがI、EIおよびBピクチャでない時、RPBTはRMPNI“エンドループ”指示に続く。もしピクチャがIもしくは、EIピクチャであるならば、RPBTはERPSレイヤの最初の要素である。もしピクチャがBピクチャであるならば、RPBTは存在しない。RPBT値は以下のように定義される。

“1” : スライディングウインドウ

“0” : 適応メモリ制御

“スライディングウインドウ”バッファリングタイプにおいて、現在復号されたピクチャはデフォルト相対的インデックス0でバッファに追加されなければならない。バッファでのピクチャの“使用されない”というマーキングは、短期間ピクチャセットの中でファーストイン-ファーストアウト方式で自動的に実行される。この場合において、バッファが現在のピクチャを蓄積するだけの十分な“使用されない”容量を持っているなら、どの追加のピクチャもバッファで“使用されない”としてマーキングされてはならない。もし、バッファが現在のピクチャを蓄積するだけの十分な“使用されない”容量を持っていないければ、バッファ中の短期間ピクチャの中で、最大のデフォルトインデックス(もしくは、サブピクチャ除去の場合、必要に応じて複数のインデックス)を持つピクチャ(もしくは、サブピクチャ除去の場合、必要なメモリ量を解放するため必要に応じて複数のピクチャ)が、“使用されない”としてマーキングされなければならない。スライディングウインドウバッファリングタイプにおいて、バッファ内容を制御するための付加情報は一切伝送されない。

“適応メモリ制御”のバッファリングタイプにおいて、符号器はバッファへの追加やバッファ中の“使用されない”データのマーキングを明示的に指定する。また、長期間インデックスを短期間ピクチャに割り当ててもよい。現在のピクチャと他のピクチャは、符号器により指定されるように、バッファ内に“使用されない”として明示的にマーキングされてもよい。このバッファリングタイプは、メモリ管理制御操作(MMCO)パラメータにより制御される更なる情報を必要とする。

RPBTは、もしGOBもしくはスライスレイヤに存在するなら、ピクチャレイヤでのそれと同じでなければならない。GOBもしくはスライスレイヤに存在するMMCOコマンドは、ピクチャレイヤにおけるMMCOコマンドと同じ操作を伝えなければならない。

もしピクチャがBピクチャなら、RPBTは存在してはならず、復号されたピクチャをマルチピクチャバッファに蓄積してはならない。これは、Bピクチャがマルチピクチャバッファの内容に影響してはいけないことを保証する。

同様に、改良PBフレームのBピクチャ部はバッファに蓄積してはならない。改良PBフレームの蓄積制御に関係する全ての制御フィールドは、改良PBフレームのPピクチャ部のみの蓄積制御に関係すると考えられなければならない。

#### U. 3. 1. 5. 7 メモリ管理制御操作 (MMCO) (可変長)

MMCOは可変長符号語で、RPBTが“適応メモリ制御”を指示する時にのみ存在し、MMCOは存在する場合、複数回発生してもよい。MMCOはマルチピクチャバッファメモリの管理に用いられる制御操作を指定する。MMCOパラメータの後に、MMCOの値により指定された操作に必要なデータが続く。またMMCO値がその様な操作リストの終わりを指示するまで、追加のMMCOパラメータが続く。MMCOコマンドはバッファ内容および現在のピクチャの復号処理に影響せず、むしろそれらはビット列における後続のピクチャを復号するために必要なバッファ状態を指定する。MMCOに関係する値と制御操作は付表U. 3/JT-H263にて定義される。

MMCOを使って指定される全てのメモリ管理制御操作は、ピクチャレイヤ内にて指定されなければならない。ピクチャレイヤで指定されたいくつかのまたは全てと同じ操作が、GOBまたはスライスレイヤにおいても(同じ関連するデータで)指定されるかもしれない。ピクチャレイヤで同じ関連するデータで指定されないメモリ操作を、GOBまたはスライスレイヤでMMCOが指定してはならない。

バッファサイズと構造を指定するMMCOコマンドは存在する場合、最初のMMCOコマンドでなければならない。1つのERPSレイヤ内には、バッファサイズと構造を指定するMMCOコマンドは、1つしか存在してはならない。RESETが“1”のバッファサイズと構造を指定するMMCOコマンドは、最初のピクチャ内に存在すべきであり、その最初のピクチャで、ビット列におけるERPSモードピクチャのシリーズに対してERPSモードがアクティブにされる。RESETが“1”のバッファサイズと構造を指定するMMCOコマンドは、短期間もしくは長期間ピクチャのサブピクチャ領域を“使用されない”とマーキングを指示するなどのMMCOの使用より先行しなければならない。バッファサイズと構造を指定するMMCOコマンドにおいて指定されたサブピクチャの幅と高さは、現在のピクチャがRESETが“1”のIもしくはEIピクチャでない限り、前のバッファサイズと構造を指定するMMCOコマンドでのこれらのパラメータ値と異なってはいけない。ピクチャの高さと幅は、RESETが“1”のバッファサイズと構造を指定するMMCOコマンドを含むピクチャ内(もしくはERPSモードが使われていないピクチャ内)を除いて、ビット列の中で変わってはならない。

もし単一ピクチャ後方予測を使用しているBピクチャがビット列中に存在するなら、O. 2節で指定されるように、Bピクチャの参照レイヤにおいて厳密に時間的に1つ後方の非Bピクチャは、ビット列順序のBピクチャより先行しなければならない。参照レイヤピクチャはBピクチャの復号が終わるまで表示するために必要であるため、時間的に直後に続く非Bピクチャのいずれかの部分を“使用されない”とマーキングする、Bピクチャの参照レイヤ内の、時間的に直後に続く非BピクチャのどのERPSレイヤ内にも、メモリ管理制御操作は存在してはならない。

O. 2節で指定される伝送順序の制限は、2ピクチャ後方予測を使用するBピクチャに対して必要性に応じて調節される。もし2ピクチャ後方予測を使用するBピクチャがビット列中に存在するなら、Bピクチャの参照レイヤにおける厳密に時間的に2つ後の非Bピクチャは、ビット列順序においてBピクチャより先行しなければならない。O. 2節で指定されたビット列におけるBピクチャの伝送順序における他の制限が適用されなければならないが、2つの時間的に直後の参照レイヤピクチャの使用のために調節される。これら参照レイヤピクチャはBピクチャの復号が終わるまで表示するために必要であるため、これら2つの非Bピクチャのいずれかの部分を“使用されない”とマーキングする、Bピクチャの参照レイヤ内の、これら2つの時間的に直後に続く非BピクチャのどのERPSレイヤ内にも、メモリ管理制御操作は存在してはならない。

“蓄積されたピクチャ”は、そのピクチャ(全体)を“使用されない”にマーキングする、ERPSレイヤ内のMMCOコマンドを含まない非Bピクチャとして定義される。もし現在のピクチャが蓄積されたピクチャでないなら、ERPSレイヤは以下のタイプのMMCOコマンドを含んではならない。

- RESETが“1”のバッファサイズと構造を指定するMMCOコマンド
- 以前蓄積されたピクチャのERPSレイヤにおいて“使用されない”とマーキングされたことがない(現在のピクチャ以外の)他のピクチャを“使用されない”としてマーキングするいくつかのMMCOコマンド
- 以前蓄積されたピクチャのERPSレイヤにおいてそれと同じ長期間インデックスが割り当てられたことがないピクチャにその長期間インデックスを割り当てるいくつかのMMCOコマンド

- ・ 以前蓄積されたピクチャのERPSレイヤ内において“使用されない”とマーキングされたことがないピクチャのサブピクチャ領域を“使用されない”とマーキングするいくつかのMMCOコマンド

付表U. 3/JT-H263 メモリ管理制御操作 (MMCO) 値  
(ITU-T H. 263)

値	メモリ管理制御操作	後続する関連するデータ・フィールド
'0'	MMCOループの終了	なし (ERPSレイヤの終了)
'011'	短期間ピクチャを“使用されない”にマーキング	DPN
'0100'	長期間ピクチャを“使用されない”にマーキング	LPIN
'0101'	ピクチャへの長期間インデックスの割り当て	DPNおよびLPIN
'00100'	短期間サブピクチャ領域を“使用されない”にマーキング	DPNおよびSPRB
'00101'	長期間サブピクチャ領域を“使用されない”にマーキング	LPINおよびSPRB
'00110'	最大長の期間ピクチャインデックスを指定	MLIP1
'00111'	バッファサイズと構造を指定	SPWI、SPHI、SPTN、およびRESET

#### U. 3. 1. 5. 8 ピクチャ番号の差分 (DPN) (可変長)

DPNはMMCOによって指示される場合に存在する。DPNはもし存在するならばMMCOに続く。DPNは付表U. 1/JT-H263内の符号語を用いて伝送され、メモリ制御操作のためピクチャのPNを算出するために用いられる。またDPNは、ピクチャに長期間インデックスを割り当てるために、短期間ピクチャを“使用されない”とマーキングするために、あるいは短期間ピクチャのサブピクチャ領域を“使用されない”とマーキングするために用いられる。もし復号された現在のピクチャ番号がPNCであり、付表U. 1/JT-H263から復号された値がDPNであるならば、指定された当該ピクチャ番号であるPNQの算出のため、下記の等式と数学的に等価な演算が用いられなければならない。

```
if (PNC - DPN < 0)
    PNQ = PNC - DPN + 1024;
else
    PNQ = PNC - DPN;
```

同様に、符号器は符号化するDPN値を下記の関係を用いて計算してもよい。

```
if (PNC - PNQ < 0)
    DPN = PNC - PNQ + 1024;
else
    DPN = PNC - PNQ;
```

例えば、もしDPNの復号された値が0であり、MMCOが短期間ピクチャを“使用されない”とマーキングすることを指示しているならば、現在復号されているピクチャは“使用されない”としてマーキングされなければならない（したがって現在のピクチャは蓄積されたピクチャではないことを示している）。

#### U. 3. 1. 5. 9 長期間ピクチャインデックス (LPIN) (可変長)

LPINはMMCOによって指示される場合に存在する。LPINは付表U. 1/JT-H263内の符号語を用いて伝送され、あるピクチャの長期間ピクチャインデックスを指定する。もし操作があるピクチャに対する長期間インデックスの割り当てであるならば、LPINはDPNに続く。もしその操作が長期間ピクチャへの“使用されない”とのマーキング、あるいは長期間ピクチャのサブピクチャ領域への“使用されない”とのマーキングであるならば、LPINはMMCOに続く。

#### U. 3. 1. 5. 10 サブピクチャ除去ビットマップ (SPRB) (固定長)

SPRBはピクチャのそれぞれのサブピクチャ領域に対して1ビットを持つ固定長符号語であり、MMCOによって指示される場合に存在する。SPRBデータのビット数は最新のSPWIおよびSPHIの値によって決定される。SPRBはバッファされたピクチャのどのサブピクチャ領域が“使用されない”とマーキングされなければならないかを指示するために用いられる。もし操作が短期間ピクチャのサブピクチャ領域への“使用されない”とのマーキングであるならば、SPRBはDPNに続き、もし操作が長期間ピクチャのサブピクチャ領域への“使用されない”とのマーキングであるならば、SPRBはLPINに続く。

サブピクチャはピクチャの左上隅から始まるラスタ走査順序で番号付けされる。例えば、DPNによって指定された参照ピクチャが6個のサブピクチャに区画分けされている場合について考える。“ $S_1 S_2 S_3 S_4 S_5 S_6$ ”は6ビットのSPRBデータを表現するとする。もしビット $S_i$ が“1”であるならば、復号器は指示された参照ピクチャ内の $i$ 番目のサブピクチャを“使用されない”とマーキングすべきである。例えば、もしSPRBが‘000110’であるならば、4番目と5番目のサブピクチャ領域が“使用されない”とマーキングされる。

スタートコードエミュレーションを防ぐために、U. 3. 1. 5. 11小小小節において指定されるように、すべての必要なSPREPBエミュレーション防止ビットが、SPRBデータの内部あるいはSPRBデータに続いて挿入されなければならない。

もしSPRBが存在して、かつ特定のピクチャが前のSPRBビットマップによって以前に影響を受けているならば、SPRBによって指定されるビットマップはその以前のSPRBビットマップの中で“1”を含んでいたどのサブピクチャ領域に対しても“1”を含まなければならない。各々のSPRBビットマップは値“0”を持つビットを少なくとも1つ、および値“1”を持つビットを少なくとも1つ含まなければならない。

#### U. 3. 1. 5. 11 サブピクチャ除去エミュレーション防止ビット (SPREPB) (1ビット)

SPREPBは値“1”を持つ1ビットの固定長符号語であり、SPRBデータの8個の連続する0ビットのビット列の直後に挿入されなければならない。

#### U. 3. 1. 5. 12 最大長期間ピクチャインデックスプラス1 (MLIP1) (可変長)

MLIP1はもしMMCOによって指示されるならば存在する可変長符号語である。MLIP1はもし存在するならばMMCOに続く。MLIP1は付表U. 1/JT-H263内の符号語を用いて伝送される。もし存在するならば、MLIP1は(別のMLIP1の値を受け取るまで)長期間参照ピクチャに対して許される最大インデックスを決定するために用いられる。復号器は別の値を受け取るまで、最初はMLIP1が“0”であると仮定しなければならない。MLIP1パラメータを受け取るとすぐ、後続のピクチャの復号処理での参照のために、MLIP1の復号値-1よりも大きなインデックスを持つ全ての長期間ピクチャを“使用されない”と復号器は考えなければならない。マルチピクチャバッファ内の他の全てのピクチャに対して、いかなる状態の変化もMLIP1によって指示されてはならない。

#### U. 3. 1. 5. 13 サブピクチャ幅インジケーション (SPWI) (7ビット)

SPWIはもしMMCOによって指示されるならば存在する7ビットの固定長符号語である。SPWIは指示されている場合にMMCOに続く。SPWIは、指示されたサブピクチャ幅が $16 \times (SPWI + 1)$ 個の輝度標本であるように、サブピクチャの幅を16個の輝度標本を単位として指定する。現在のピクチャは幅方向に $\text{ceil}(pw / 16) / (SPWI + 1)$ 個のサブピクチャを有する。ここで $pw$ はピクチャの幅であり、“/”は浮動小数点除算を示している。正の値に対しては、切り上げ関数である $\text{ceil}(x)$ は、 $x$ が整数の場合は $x$ と等しく、その他の場合は $\text{ceil}(x)$ は $x$ の整数部分に1を加えた値と等しくなる。もしサブピクチャの最小の幅および高さを定義している最小ピクチャ単位(MPU)サイズが外部手段(例えばTTC標準JT-H245)によって交渉されているならば、SPWIによって指定されるサブピクチャ幅はMPUの幅の整数倍でなければならない。もしそうでなければ、SPWIによって指定されるサブピクチャ幅は、SPWIが $\text{ceil}(pw / 16) - 1$ となるようなサブピクチャ幅でなければならない。

#### U. 3. 1. 5. 14 サブピクチャ高さインジケーション (SPHI) (7ビット)

SPHIは、もしSPWIが(MMCOによって指示されるように)存在するならば存在する7ビットの固定長符号語である。SPHIはもし存在するならばSPWIに続く。SPHIは、指示されたサブピクチャの高さが1

6 × SPHI であるように、サブピクチャの高さを 16 個の輝度標本を単位として指定する。SPHI の許される値の範囲は 1 から 72 である。現在のピクチャは高さ方向に  $\text{ceil}(\text{ceil}(ph/16)/SPHI)$  個のサブピクチャを有する。ここで、 $ph$  はピクチャの高さであり、“/” は浮動小数点除算を示している。もしサブピクチャの最小の幅および高さを定義している最小ピクチャ単位 (MPU) サイズが外部手段 (例えば TTC 標準 JT-H 245) によって交渉されているならば、SPHI によって指定されるサブピクチャの高さは MPU の高さの整数倍でなければならない。もしそうでなければ、SPHI によって指定されるサブピクチャの高さは、SPHI が  $\text{ceil}(ph/16)$  となるようなサブピクチャの高さでなければならない。

#### U. 3. 1. 5. 15 サブピクチャ総数 (SPTN) (可変長)

SPTN は、もし SPWI および SPHI が (MMCO によって指示されるように) 存在するならば存在する可変長符号語である。SPTN はもし存在するならば SPHI に続く。SPTN は付表 U. 1 / JT-H 263 を用いて符号化される。ここで付表 U. 1 / JT-H 263 のインデックスは SPTN の復号値 - 1 に対応する。SPTN の復号値は、SPWI および SPHI によって指定されるサブピクチャを単位としたマルチピクチャバッファの使用できる総容量である。現在のピクチャの復号のために必要とされるメモリ容量は、SPTN には含まれない。つまり、SPTN は、他のピクチャの予測のために用いる参照ピクチャを格納するために必要とされるメモリ容量だけである。サブピクチャ除去が使用されない場合 (すなわち SPWI および SPHI が全ピクチャ範囲である場合) は、(例えば、スライディングウインドウ操作に対する) アクティブな短期間参照ピクチャの最大個数は、SPTN から長期間インデックスに割り当てられていてかつ引き続き “使用されない” とマーキングされていないピクチャ数を引いた数によって与えられる。

#### U. 3. 1. 5. 16 バッファリセットインジケータ (RESET) (1ビット)

RESET は 1 ビットの固定長符号語であり、もし SPWI、SPHI、および SPTN が (MMCO によって指示されるように) 存在するならば存在する。もし存在するならば RESET は SPTN に続く。RESET の値は以下のようになければならない。

- “0” : バッファ内容はリセットされない
- “1” : バッファ内容はリセットされる

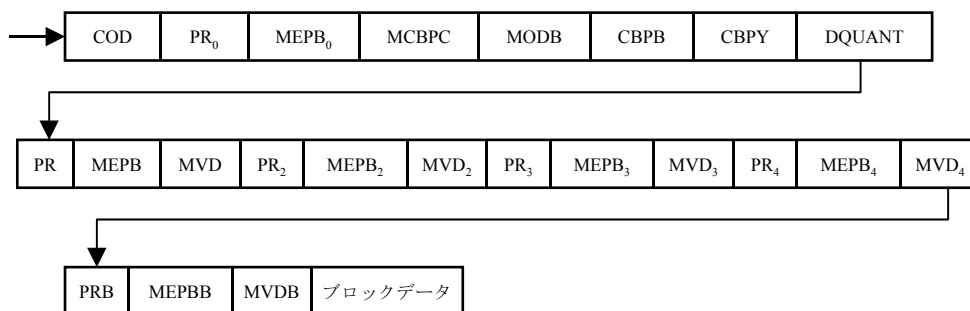
もし RESET が “1” であるならば、(別途指定されない限り現在のピクチャを除き) マルチピクチャバッファ内のすべてのピクチャは (短期間および長期間ピクチャの両方を含み) “使用されない” とマーキングされなければならない。

### U. 3. 2 マクロブロックレイヤのシンタックス

#### U. 3. 2. 1 Pピクチャおよび改良PBフレームにおけるマクロブロックのシンタックス

MRPA によって指示されるように、選択された前方参照ピクチャの数が 1 よりも大きいかもしれないとき、Pピクチャおよび改良PBフレームに対して ERPS レイヤが存在するならば、マクロブロックレイヤのシンタックスは変更される。MRPA フィールドは ERPS レイヤの中で通知される。MRPA が “1” である場合、マクロブロックレイヤのシンタックスは付図 U. 6 / JT-H 263 で示される。そうでなければ、Pピクチャあるいは改良PBフレームにおけるマクロブロックのシンタックスは図 10 / JT-H 263 から変更されない。





付図U. 6 / JT-H263 ERPモードでのPピクチャおよび改良PBフレームの  
(ITU-T H. 263) マクロブロックレイヤの構造

### U. 3. 2. 1. 1 CODの解釈

もしCODビットが“1”であるならば、そのマクロブロックに対して更なる情報は伝送されない。その場合、復号器は、そのマクロブロックを、全部のマクロブロックに対する動きベクトルが0に等しく、ピクチャ参照パラメータが0に等しく、およびいかなる係数データをも持たない、INTERマクロブロックとして扱わなければならない。もしCODビットが“0”であるならば、そのマクロブロックが符号化されていることを指示しており、マクロブロックレイヤのシンタックスは、そのシンタックスに含まれることになるPR<sub>0</sub>、PR、PR<sub>2</sub>、PR<sub>3</sub>、PR<sub>4</sub>、およびPRBフィールドと一緒に付図U. 6 / JT-H263に図示される。PR<sub>0</sub>、PR、PR<sub>2</sub>、PR<sub>3</sub>、PR<sub>4</sub>、およびPRBは、それぞれ付表U. 1 / JT-H263によって与えられる可変長符号語から構成される。

### U. 3. 2. 1. 2 ピクチャ参照パラメータ0 (PR<sub>0</sub>) (可変長)

PR<sub>0</sub>は付表U. 1 / JT-H263において定義されている可変長符号語である。PR<sub>0</sub>はCODが“0”であるときはいつでも存在する。もしPR<sub>0</sub>が復号値0 (符号語‘1’)を持つならば、そのマクロブロックに対して更なる情報がその後続くことを指示している。もし0でない復号されるならば、そのマクロブロックがピクチャ参照パラメータのみを用いて符号化されていることを指示している。

もしPR<sub>0</sub>フィールドが復号値0 (符号語‘1’)を持たないのであれば、そのマクロブロックに対しては更なる情報はその後伝送されない。その場合復号器は、そのマクロブロックを、全部のブロックに対する動きベクトルが0に等しく、ピクチャ参照パラメータがPR<sub>0</sub>に等しく、いかなる係数データも持たない、INTERマクロブロックとして扱わなければならない。

もしPR<sub>0</sub>フィールドが復号値0 (符号語‘1’)を持つのであれば、そのマクロブロックは符号化されている。MCBPC、CBPB、CBPY、およびDQUANTフィールドの意味および取り扱いは変更されないままである。全てのINTERマクロブロックに対して (および改良PBフレームモードにおけるINTRAマクロブロックに対しても)、PRフィールドはMVDフィールドと一緒に含まれる。改良PBフレームにおけるMODBの使用についてはU. 3. 2. 1. 4 小小小節に記述されている。

### U. 3. 2. 1. 3 マクロブロックエミュレーション防止ビット0 (MEPB<sub>0</sub>) (1ビット)

MEPB<sub>0</sub>は値“1”を有する1ビットの固定長符号語であり、もしPR<sub>0</sub>が存在し復号値“1” (符号語‘000’)を持ち、以下の2つの条件のいずれか一方を満たす場合にのみPR<sub>0</sub>に続く。

1. スライス構造モード (付属資料K参照) が用いられている
2. COD = “0” でありかつ PR<sub>0</sub> = “1” (符号語‘000’) である別のマクロブロックの直後に現在のマクロブロックに対するCODが続き、前のマクロブロックのPR<sub>0</sub>にMEPB<sub>0</sub>ビットが後続しない

MEPB<sub>0</sub>の目的はスタートコードエミュレーションを防ぐことであり、同時にまたスライス構造モードにおいて1つのスライスの中のマクロブロック数を決めるのに役立つことである。

#### U. 3. 2. 1. 4 マクロブロックピクチャ参照パラメータ (PR、PR<sub>2-4</sub>、およびPRB) (可変長)

PRは主なピクチャ参照パラメータである。PRはMVDが存在するときはいつでも存在する。もしPTYPEによって指示され、かつMCBPCがINTER4VあるいはINTER4V+Qマクロブロック(表8および表9/JT-H263の中でタイプ2あるいはタイプ5であるマクロブロック)を指示しているならば、3個の符号語PR<sub>2-4</sub>はMVD<sub>2-4</sub>と一緒に含まれる。PR<sub>2-4</sub>およびMVD<sub>2-4</sub>は拡張予測モード(付属資料F参照)あるいはデブロッキングフィルタモード(付属資料J参照)にある場合のみ存在する。MVD<sub>B</sub>が存在していることをMODBが指示している場合のみ、PRBは改良PBフレームの中に存在する。PR、PR<sub>2-4</sub>、およびPRBは、各々マルチピクチャバッファのピクチャ参照相対的インデックスを指定する。

もしマクロブロックがINTER4VでもINTER4V+Qでもないならば、PRはそのすべてのマクロブロックの動き補償に対するピクチャ参照パラメータとして用いられる。もしマクロブロックがINTER4VあるいはINTER4V+Qであるならば、PRはそのマクロブロックにおける4個の8×8輝度ブロックの最初のブロックの動き補償予測に対して用いられ、またそのマクロブロックにおける2個の色差ブロックに対して(6.1小節で定義されている動き補償処理と共に)用いられる。PR<sub>2-4</sub>はそのマクロブロックにおける残り3個の8×8輝度データの動き補償に対して用いられる。もしMVD<sub>B</sub>が存在することをMODBが指示しているならば、PRBは改良PBフレームのB部分の前方予測に対するピクチャ参照パラメータである。

MODBがB<sub>PB</sub>双方向予測を指示しているときの改良PBフレームにおいては、最も新しい前の参照ピクチャが差分相対的インデックス順序に再マッピングされているか否か、あるいは“使用されない”とマーキングされているか否か、あるいは長期間インデックスに割り当てられているか否かにかかわらず、TR<sub>D</sub>とTR<sub>B</sub>の値は、現在のピクチャおよび最も新しい前の参照ピクチャのテンポラルリファレンスデータに基づくテンポラルリファレンスの増分として算出されなければならない。改良PBフレームにおけるB<sub>PB</sub>双方向予測のための前方参照ピクチャとして用いられるピクチャは、PRにより指定されるピクチャでなければならない。

#### U. 3. 2. 1. 5 マクロブロックエミュレーション防止ビット (MEPB、MEPB<sub>2-4</sub>、およびMEPBB) (各1ビット)

MEPB、MEPB<sub>2-4</sub>、およびMEPBBはもし存在するならばそれぞれ値“1”を持つ1ビットである。非制限動きベクトルモード(付属資料D参照)が使用されておらず、関連するPR、PR<sub>2-4</sub>、またはPRBフィールドが存在して復号値“1”(符号語‘000’)を持つ場合のみ、MEPB、MEPB<sub>2-4</sub>、MEPBBのそれぞれは存在しなければならない。これらの目的はスタートコードエミュレーションを防ぐことである。

#### U. 3. 2. 2 BピクチャとEP-Pピクチャのマクロブロックにおけるシンタックス

BおよびEPピクチャ(付属資料O参照)に対するマクロブロックシンタックスはPピクチャと同様な方法で変更される。CODビットが“1”である場合、それは、EPピクチャの前方(スキップされた)予測、及びBピクチャの直接(スキップされた)双方向予測の前方部分に対してはピクチャ参照パラメータ0を用いて、また(B<sub>S</sub>B<sub>B</sub>Wが存在し、その値が“0”である時の2ピクチャ後方予測の場合)Bピクチャの直接(スキップされた)双方向予測の後方部分に対しては最初の後方予測ピクチャを用いて、付属資料Oに指定されるようなスキップされるマクロブロックを表す。CODビットが“0”の場合は、パラメータPR<sub>0</sub>はシンタックスに挿入され、U. 3. 2. 1. 2小節と同様の方法で使われる。PR<sub>0</sub>が存在し復号値が0(符号語が“1”)では無い時、それは、値が0の動きベクトル値とピクチャ参照パラメータPR<sub>0</sub>を用いた前方INTER予測でそのマクロブロックは予測されるべきであることを示す。PR<sub>0</sub>の復号値が0の時は、MBTYPEは後に続きマクロブロックタイプを指定する。CBPC、CBPY、およびDQUANTのフォーマットは変わらない。ERPSモードが使用されない場合と同様の方法で、MVD<sub>DFW</sub>やMVD<sub>BW</sub>フィールドが符号化される。しかし、これらのフィールドは各々ピクチャ参照や場合によってはエミュレーション防止用ビットと一緒に使われる。

Bピクチャに対して、マルチピクチャバッファ内の後方参照ピクチャは以下の様に定義される。

- 単一ピクチャ後方予測の場合は、唯一の後方参照ピクチャが有り、それは(場合により再マッピングされた)相対的インデックス順序での最初のピクチャであり、
- 2ピクチャ後方予測の場合は、2つの後方参照ピクチャが有り、それらは(場合により再マッピングされた)相対的インデックス順序での最初の2つのピクチャである。

マルチピクチャバッファ内の前方参照ピクチャは、後方参照ピクチャ以外のマルチピクチャバッファ内のピクチャであると定義される。前方予測の相対的インデックスは、前方参照ピクチャセットの中の相対的インデックスであり、後方予測の相対的インデックスは、後方参照ピクチャセットの中の相対的インデックスである。

例えばそのバッファが、（ピクチャ番号が増加する順序に伝送された）ピクチャ番号が各々300、301、302を持つ3つの短期間ピクチャを含み、長期間ピクチャインデックスが0と3の2つの長期間ピクチャを含んでいる時、2ピクチャ後方予測におけるデフォルトのインデックス順序は以下の通りとなる。

- ・ デフォルトの後方参照インデックス0はピクチャ番号303の短期間ピクチャであり、
- ・ デフォルトの後方参照インデックス1はピクチャ番号302の短期間ピクチャであり、
- ・ デフォルトの前方参照インデックス0はピクチャ番号300の短期間ピクチャであり、
- ・ デフォルトの前方参照インデックス1は長期間ピクチャインデックス0の長期間ピクチャであり、
- ・ デフォルトの前方参照インデックス2は長期間ピクチャインデックス3の長期間ピクチャである。

また単一ピクチャ後方予測の場合には、以下の通りである。

- ・ 単一のデフォルトの後方参照ピクチャはピクチャ番号303の短期間ピクチャであり、
- ・ デフォルトの前方相対的インデックス0はピクチャ番号302の短期間ピクチャであり、
- ・ デフォルトの前方相対的インデックス1はピクチャ番号300の短期間ピクチャであり、
- ・ デフォルトの前方相対的インデックス2は長期間ピクチャインデックス0の長期間ピクチャであり、
- ・ デフォルトの前方相対的インデックス3は長期間ピクチャインデックス3の長期間ピクチャである。

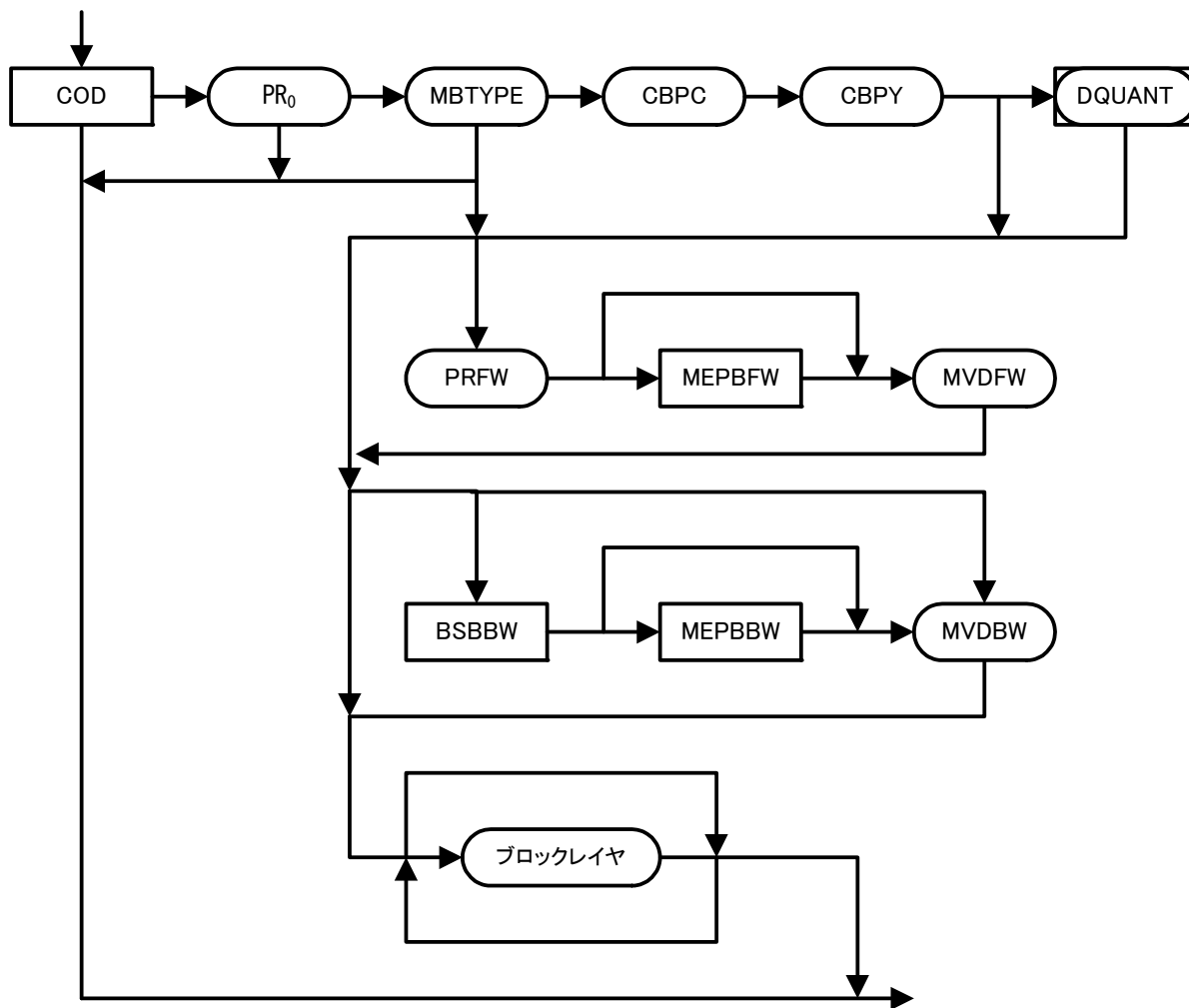
そして仮に、これらのピクチャが、短期間ピクチャ302、短期間ピクチャ303、長期間ピクチャ0、短期間ピクチャ300、長期間ピクチャ3の順序に新たな相対的インデックス順序に再マッピングされた場合は、2ピクチャ後方予測における新しい相対的インデックス順序は以下の通りである。

- ・ 再マッピングされた後方相対的インデックス0はピクチャ番号302の短期間ピクチャであり、
- ・ 再マッピングされた後方相対的インデックス1はピクチャ番号303の短期間ピクチャであり、
- ・ 再マッピングされた前方相対的インデックス0は長期間ピクチャインデックス0の長期間ピクチャであり、
- ・ 再マッピングされた前方相対的インデックス1はピクチャ番号300の短期間ピクチャであり、
- ・ 再マッピングされた前方相対的インデックス2は長期間ピクチャインデックス3の長期間ピクチャである。

また単一ピクチャ後方予測の場合は以下の通りである。

- ・ 単一の再マッピングされた後方参照ピクチャはピクチャ番号302の短期間ピクチャであり、
- ・ 再マッピングされた前方インデックス0はピクチャ番号303の短期間ピクチャであり、
- ・ 再マッピングされた前方インデックス1はインデックス0の長期間ピクチャであり、
- ・ 再マッピングされた前方インデックス2はピクチャ番号300の短期間ピクチャであり、
- ・ 再マッピングされた前方インデックス3はインデックス3の長期間ピクチャである。

Bピクチャの直接双方向予測に使われる $TR_D$ は、（場合により再マッピングされた）相対的インデックス順序の最初の方参照ピクチャと（場合により再マッピングされた）相対的インデックス順序の最初の後方参照ピクチャ（即ち、もし2ピクチャ後方予測が使われるなら、これは、U. 3. 2. 2. 3小節に記述されている様にBSBWが“0”である時に参照されるピクチャ）との間のテンポラルリファレンス増加量として計算されなければならない。Bピクチャにおいて直接双方向予測として使われる $TR_B$ は、Bピクチャと（場合により再マッピングされた）相対的インデックス順序の最初の方参照ピクチャとの間のテンポラルリファレンス増加量として計算されなければならない。 $TR_D$ や $TR_B$ の計算の中で使われる相対的インデックス順序は、BピクチャシンタックスのピクチャレベルのERPSレイヤにより指定されるものでなければならない（即ち、GOBやスライスレベルにて再マッピングされるものは、 $TR_D$ や $TR_B$ の値に影響を与えてはならない）。



付図U. 7 / JT-H263 ERPSモードに対するEPおよびBピクチャのマクロブロックレイヤの構造 (ITU-T H. 263)

**U. 3. 2. 2. 1 前方予測用のピクチャ参照 (PRFW) (可変長)**

PRFWは可変長のピクチャ参照パラメータであり、MVDFW有る場合には必ず存在する。PRFWは付表U. 1 / JT-H263を用いて符号化される。PRFWは前方参照ピクチャセットに対する相対的インデックスである。

**U. 3. 2. 2. 2 前方予測のためのエミュレーション防止ビット (MEPBFW) (1ビット)**

MEPBFWは値“1”である符号語長が1の固定長符号である。PRFWが存在し、復号値が“1” (符号語が“000”)であり、かつ、非制限動きベクトルモード (付属資料D参照) が使われていない時のみ、MEPBFWはPRFWの直後に挿入されなければならない。

**U. 3. 2. 2. 3 後方予測におけるBピクチャ選択ビット (BSBBW) (1ビット)**

BSBBWは1ビットの固定長符号である。MVDBWが存在し、かつ、Bピクチャ処理として2ピクチャ後方予測が指定される時のみ、このBSBBWはBピクチャに対して現れる。このビットの意味は以下の通りに定義されなければならない。

“0” : 相対的インデックス順序の最初の後方参照ピクチャからの予測 (デフォルトの順序では、仮にそのピクチャが長期間インデックスが割り当てられず、かつ“使用されない”とマーキングされない場合は、これは最も新しい短期間参照ピクチャであろう)。

“1”：相対的インデックス順序の2番目の後方参照ピクチャからの予測（デフォルトの順序では、仮に最後の2つのピクチャのどちらも長期間インデックスが割り当てられず、かつ“使用されない”とマーキングされない場合は、これは2番目に最も新しい短期間参照ピクチャであろう）。

#### U. 3. 2. 2. 4 後方予測のためのエミュレーション防止ビット (MEPBW) (1ビット)

MEPBWは値“1”である長さ1ビットの固定長符号語である。以下の条件が成立する場合のみ現れる。

- ・ BSBWが存在し、値が“0”に等しく、かつ、
- ・ 非制限動きベクトル（付属資料D参照）が使われず、かつ、
- ・ BSBWの直前に値“0000”の5ビットが有る時

#### U. 4 復号処理

ERPSモードにおける復号器は、フレーム間ピクチャの復号のために参照ピクチャをマルチピクチャバッファ内に格納する。この復号器は複数の復号されたピクチャを格納するために（ERPSモードに対応しない場合に必要とされるメモリ容量と比較して）追加的なメモリ容量を必要とするかもしれない。この復号器は参照ピクチャのバッファリングモードとビット列で指定されたメモリ管理制御操作に対応して符号器のマルチピクチャバッファの複製を作る。部分的に誤りを含むピクチャを復号した場合にも、バッファリング方式は動作するであろう。

伝送され蓄積された各々のピクチャにはピクチャ番号 (PN) が割り当てられ、PNはマルチピクチャバッファ内にピクチャと一緒に格納される。PNは格納されたピクチャを識別するための連続したピクチャ計数識別子を表す。PNはモジュロ1024の算術操作により制限される。最初に伝送されたピクチャに対して、PNは“0”であるべきである。伝送され蓄積されるその他のそれぞれのピクチャに対して、PNは（付属資料0が用いられている場合、各々のスケーラビリティレイヤの中で）1つづつ増加しなければならない。受信し格納した2個の連続するピクチャのPNの差異が（モジュロ1024で）1では無いなら、その復号器はピクチャの損失かデータ誤りが有ったと見なすべきである。そのような場合、ピクチャの損失を示す逆方向チャンネルメッセージが符号器に伝送されても良い。

PNに加えて、マルチピクチャバッファに格納された各々のピクチャはデフォルト相対的インデックスと呼ばれる、関連付けられたインデックスを持つ。マルチピクチャバッファにある1つのピクチャが最初に加えられた場合、（長期間インデックスが割り当てられていなければ）デフォルト相対的インデックス0が与えられる。ピクチャがマルチピクチャバッファに加えられる、またはマルチピクチャバッファから削除される場合、または短期間ピクチャに長期間インデックスが割り当てられる場合、マルチピクチャバッファ内のピクチャのデフォルト相対的インデックスは変更される。

マルチピクチャバッファに格納されたピクチャは、長期間ピクチャと短期間ピクチャの2種類に分類できる。長期間ピクチャはマルチピクチャバッファ内に長期間（1023を超える、符号化され格納されたピクチャ間隔）滞在できる。現在のピクチャは当初、短期間ピクチャと見なされる。ビット列内の情報に従って長期間インデックスを割り当てる事により、どの短期間ピクチャも長期間ピクチャに変更できる。PNは、マルチピクチャバッファ内の全ての短期間ピクチャに対してユニークなIDである。短期間ピクチャが長期間ピクチャに変更される時、長期間ピクチャインデックス (LPIN) も割り当てられる。PNをLPINに関連付ける事により長期間ピクチャインデックスをピクチャに割り当てる。一度長期間ピクチャインデックスをあるピクチャに割り当てると、ビット列内でその長期間ピクチャのPNをその後で使用することができるのは、長期間インデックスの割り当ての繰り返しの場合のみに限られなければならない。長期間ピクチャのPNは、1024個の伝送され格納されたピクチャの中で固有である。それゆえ、長期間ピクチャインデックスのPNは、それに続いて伝送され蓄積された1023個のピクチャの後で、長期間インデックスの割り当てに使用されてはならない。LPINは長期間ピクチャの生存期間内で固有なIDになる。

効率的な参照ピクチャアドレッシングのために、（短期間ピクチャに対する）PNと（長期間ピクチャに対する）LPINは、再マッピングされたインデックスに従ってピクチャを再マッピングするために使用できる。

#### U. 4. 1 短期間／長期間ピクチャ管理のための復号処理

復号器はマルチピクチャバッファ内に長期間ピクチャと短期間ピクチャを持っても良い。MLIP1フィールドはそのバッファ内で許容出来る最大の長期間ピクチャインデックスを示すために使われる。前にMLIP1値が伝送されていない場合、長期間ピクチャを使用してはならない。即ち、ERPSモードを起動したら、MLIP1は

暗黙的に値“0”を初期値として持たねばならない。MLIP1パラメータを受信したら別の値のMLIP1を受信するまで、新しいMLIP1が使われなければならない。新しいMLIP1をビット列内で受信したら、MLIP1以上の長期間インデックスに対応する全ての長期間ピクチャは、“使用されない”とマーキングされていると見なさなければならない。MLIP1の伝送周期は本標準では指定しない。しかし、INTRA要求メッセージなどのエラーメッセージを受信したら、符号器はMLIP1を伝送するべきである。

MMCOコマンドとそれに関連したDPNやLPINにより、短期間ピクチャは長期間ピクチャに変更出来る。短期間ピクチャ番号はDPNから得られ、長期間インデックスはLPINから得られる。MMCOコマンドを受信したら、復号器はDPNにより指定されたPNを持つ短期間ピクチャを長期間ピクチャに変更しなければならない、LPINにより指定される長期間インデックスをそのピクチャに割り当てなければならない。そのバッファに同一の値の長期間インデックスを持つ長期間ピクチャが存在する場合は、以前に存在していた長期間ピクチャを“使用されない”とマーキングされなければならない。符号器はMLIP1-1より大きい長期間インデックスをいかなるピクチャに対しても割り当ててはならない。LPINがMLIP1-1よりも大きい場合は、復号器はこの条件を誤りと見なすべきである。誤り耐性のために、符号器は同一の長期間インデックス割り当てやMLIP1指定メッセージを繰り返し伝送しても良い。長期間インデックス割り当て操作により指定されたピクチャが、要求されたLPINに既に関連している時、復号器は動作してはならない。符号器は1つのピクチャに対して2つ以上の長期間インデックス値を割り当ててはならない。長期間インデックス割り当て操作において指定されたピクチャに異なった長期間インデックスが既に割り当てられている時は、この条件は誤りとして扱われるべきである。符号器は伝送され連続的に格納された1024個のピクチャの中からのみ短期間ピクチャを長期間ピクチャに変換しなければならない。他の言い方をすれば、1023個を超えて連続して格納されるピクチャを伝送した後は、短期間ピクチャは短期間バッファに留まってはならない。ビット列内の最初のメッセージ割り当てに先立つ復号処理の中で“使用されない”とマーキングされた短期間ピクチャには長期間インデックスを割り当ててはならない。符号器は伝送されていないピクチャ番号に対して長期間インデックスを割り当ててはならない。

#### U. 4. 2 参照ピクチャバッファマッピングのための復号処理

PR<sub>0</sub>、PR、PR<sub>2</sub>、PR<sub>3</sub>、PR<sub>4</sub>、PRB、PRFW、及びBSBBWのフィールドを用いてマクロブロックレイヤでの動き補償のためにピクチャを参照する時に、復号器はインデックスを使用する。Bピクチャ以外のピクチャでは、これらのインデックスは、ADPNとLPIRフィールドが現在のピクチャ、GOB、又はスライスレイヤ（当てはまれば）の中に存在しない時はマルチピクチャバッファ内のピクチャのデフォルト相対的インデックスであり、ADPNとLPIRフィールドが存在する時は再マッピングされた相対的インデックスである。Bピクチャでは、相対的インデックス順序の最初の1つ又は2つのピクチャ（BTFSMに依存する）は、後方予測に使われ、かつ、前方ピクチャ参照パラメータは、前方予測に使用する残りのピクチャに相対的インデックスを指定する。

RMPN1、ADPN、LPIRフィールドの伝送により、マルチピクチャバッファ内のピクチャのインデックスは新しく指定されるインデックスに再マッピング出来る。RMPN1はADPN又はLPIRが存在する事を示す。ADPNが存在するとき、RMPN1はピクチャ番号予測値に加算される差分の正負を指定する。ADPN値は、再マッピングされるピクチャのPNとPNの予測値との絶対差分に相当する。最初に伝送されるADPNは、現ピクチャのPNと再マッピングされるピクチャのPNとの絶対差分として計算される。次に伝送されるADPNフィールドは、ADPNを用いて再マッピングされた前ピクチャのPNと、再マッピングされたもう1つのPNとの差異を表す。この処理は、全ての必要な再マッピングが完了するまで続けられる。LPIRの使用により指定される再マッピングの存在は、ADPNを使用して引き続いて行う再マッピングのための予測値には影響を与えない。RMPN1がLPIRフィールドの存在を示す場合、再マッピングされたピクチャは、LPIRの長期間インデックスを持つ長期間ピクチャに相当する。どのピクチャもRMPN1により特定の順序に再マッピングされない場合、これら残りのピクチャは、インデックス方式によって再マッピングされた順序をもつ全てのピクチャの後に続き、かつ再マッピングされないこれらのピクチャの中ではデフォルトの順序に従わなければならない。

復号器がピクチャの喪失を検知した場合、何らかのコンシールメント処理を行っても良いし、マルチピクチャバッファ内にエラーコンシールメントされたピクチャを挿入しても良い。単一又は複数のピクチャ番号が失われるか、又は、伝送されたADPNかLPIRによりマルチピクチャバッファ内に格納されていないピクチャが指示されれば、ピクチャが失われた事を識別出来る。マルチピクチャバッファで利用出来る時間的に最近のピクチャを喪失したピクチャの位置にコピーする事によりコンシールメントが行われても良い。マルチピクチャバッファ内の短期間ピクチャの時間順序は、デフォルトの相対的インデックス順序とPNフィールドにより推定出来る。それに加

えて、又はそれに代わり、復号器は外部手段（例えばTTC標準JT-H245）により符号器に対して強制INTRA更新信号を伝送しても良いし、復号器は外部手段や逆方向チャンネルメッセージ（例えばTTC標準JT-H245）を用いて符号器に対してピクチャ喪失を伝えてもよい。“スライディングウィンドウ”バッファリングタイプを用いている時は、マルチピクチャバッファにコンシールメントされたピクチャが挿入されても良い。GOBやスライスレイヤの復号時にピクチャ喪失が検出された場合も、あたかもピクチャレイヤでピクチャ喪失が検出されたかのように、そのピクチャに対してコンシールメントが行われても良い。

#### U. 4. 3 サブピクチャ除去のための復号器処理

複数参照ピクチャを保存するために要求されるメモリ量を削減するために、サブピクチャ除去が使用されてもよい。サブピクチャ除去の操作では、各参照ピクチャはより小さく互いに等しい大きさのサブピクチャに分割される。不要なサブピクチャ領域を“使用されない”とマーキングすることによりメモリ削減が行われる。符号器がどのサブピクチャを“使用されない”としてマーキングするかを決定するための方式は本標準の規定範囲外である。符号器は拡張参照ピクチャ選択（ERPS）レイヤのMMCOコマンドを使用することで、復号器にサブピクチャのサイズとどのサブピクチャを“使用されない”としてマーキングするかを通知する。符号器は、“使用されない”とマーキングされた参照ピクチャとサブピクチャ内のいかなる画素に対しても、以降のピクチャの予測にそれを使用する情報をビット列内に伝送してはならない。

サブピクチャ除去の能力は外部の手段（例えば、TTC標準JT-H245）によって交渉される。加えて、復号器は、（16輝度標本を単位として）サブピクチャの最小の幅と高さによって記述される最小分割単位（MPU）と、マルチピクチャバッファのために必要な全メモリ量も、外部の手段によって通知する。以下に記述した分割規則によりメモリ管理が実現される。

各参照ピクチャは同一サイズを持つ複数の矩形のサブピクチャに分割される。符号器は、MPUの整数倍でなければならないサブピクチャのサイズを指定する。サブピクチャの幅と高さは、外部の手段により交渉されたMPUの幅と高さの整数倍でなければならない。最初のサブピクチャの左上隅は参照ピクチャの左上隅と一致する。結果として、サブピクチャの幅と高さを指定することにより全体の分割が一意に記述される。もし参照ピクチャのサイズがサブピクチャのサイズの整数倍でなければ、幾つかのサブピクチャは参照ピクチャの右もしくは下の境界を越えてもよい。参照ピクチャの境界を越えるサブピクチャを保存するときに便利なメモリ管理方式は、サブピクチャの中に存在する参照ピクチャの部分のみを保存するために必要な量のメモリだけでなく、サブピクチャ全体を保存するために十分なメモリを準備しておくことである。これはバッファが満杯になったか判定することを目的とした、いかなるバッファ空き容量の計算の中でも従わなければならない規則である。上記計算は、例えば、“スライディングウィンドウ”操作で蓄積されたピクチャを“使用されない”として自動的にマーキングするかどうか決定するためにおこなわれる。各サブピクチャが同じ量のメモリを占有するように設計された復号器は、メモリフラグメンテーションが発生する可能性を防止することができる。

サブピクチャ除去使用時に参照ピクチャの画素へアクセスするために設計された手法の一例を以下に手短に示す。参照ピクチャへのアクセス技術において重要な要素のひとつに、各サブピクチャに含まれる画素がメモリのどこに格納されているか識別する仕組みがある。もし、R枚の参照ピクチャがあり、各ピクチャがS個のサブピクチャに分割されているとすると、全体で $K = R \cdot S$ 個のサブピクチャが存在する。例えば、最初の参照ピクチャの左上隅のサブピクチャをサブピクチャ番号0とし、その右のサブピクチャをサブピクチャ番号1とみなす等、K個のサブピクチャが全てサブピクチャ番号を持つまで、参照ピクチャ0からR-1までラスタ走査順に番号付けする。全バッファ容量はSPTN個のサブピクチャ用メモリバッファとなり、通常、SPTNはKより小さい値となる。K個の要素を持つ配列をsubPicMem[K]と定義し、 $t = \text{subPicMem}[k]$ をサブピクチャkの画素を含むサブピクチャ用メモリ領域に対応するものとする。例えば、R=5個の参照ピクチャがそれぞれS=12個のサブピクチャを含む場合を考える。このとき、参照ピクチャ3、サブピクチャ6の画素は、サブピクチャ用メモリ領域 $t = \text{subPicMem}[k]$ （ここで、 $k = 3 \cdot S + 6 = 42$ ）で見つけることができる。

例えば、拡張予測モードと縮小解像度更新モードを使用しない場合は、輝度もしくは色差の1ブロック分の動き補償予測のために画素を参照する際には、 $n \times m$ の画素を獲得することが必要になる。ここで、n、mは1/2画素精度の動き補償の利用するため8もしくは9の値をとってもよい。1つのブロックの画素は4つまでの異なるサブピクチャにまたがってもよく、4つの異なるケースを考えねばならない。全てのケースに共通の最初のステップは、参照されるブロックの左上隅の画素（U）を含むメモリ位置を見つけることである。Uを含むサブピクチャの位置はサブピクチャの幅と高さでUの水平位置と垂直位置を除算することで識別できる。もし、Uがサブピクチャkに存在するならば、画素はsubPicMem[k]のサブピクチャ用メモリ領域中に位置する。次にUから右にm-1番

目の画素（すなわち、ブロック右上隅の画素）とUから下に $n-1$ 番目の画素（すなわち、ブロック左下隅の画素）の両方がサブピクチャ $k$ 内に存在するならば、これは1番目のケースと見なされる。もし、Uから下に $n-1$ 番目の画素がサブピクチャ $k$ 内に存在するがUから右に $m-1$ 番目の画素がそうでなければ、2番目のケースとして見なされる。もし、Uから右に $m-1$ 番目の画素がサブピクチャ $k$ 内に存在するがUから下に $n-1$ 番目の画素がそうでなければ、3番目のケースとして見なされる。そうではなく、Uから右に $m-1$ 番目の画素とUから下に $n-1$ 番目の画素が両方ともサブピクチャ $k$ の範囲外に存在するならば、4番目のケースとして見なされる。

1番目のケースの場合、参照ブロックの全ての画素は $k$ 番目のサブピクチャに含まれる。この場合、全ての $n \times m$ 画素は $\text{subPicMem}[k]$ のサブピクチャ用メモリ領域に存在する。2番目のケースの場合、 $k$ 番目のサブピクチャ内に存在する画素は $\text{subPicMem}[k]$ のサブピクチャ用メモリ領域から得られ、残りは $\text{subPicMem}[k_r]$ から得られる。ここで、 $k_r$ は $k$ の右のサブピクチャである。3番目のケースの場合、 $k$ 番目のサブピクチャ内に存在する画素は $\text{subPicMem}[k]$ のサブピクチャ用メモリ領域から得られ、残りは $\text{subPicMem}[k_d]$ から得られる。ここで、 $k_d$ は $k$ の下のサブピクチャである。4番目のケースの場合、 $k$ 番目のサブピクチャ内に存在する画素は $\text{subPicMem}[k]$ のサブピクチャ用メモリ領域から得られ、残りは $\text{subPicMem}[k_r]$ 、 $\text{subPicMem}[k_d]$ 、 $\text{subPicMem}[k_{rd}]$ から得られる。ここで、 $k_r$ と $k_d$ は上で定義された値で $k_{rd}$ は $k$ の右下のサブピクチャである。

#### U. 4. 4 マルチピクチャ動き補償の復号器処理

MRPAフィールドが1を超える数（2以上）の参照ピクチャの使用を示しているならば、マルチピクチャ動き補償が適用される。マルチピクチャ動き補償のために、復号器はマクロブロックレイヤの $PR_0$ 、 $PR$ 、 $PR_2$ 、 $PR_3$ 、 $PR_4$ 、 $PRB$ 、 $PRFW$ 、 $BSBBW$ フィールドを使用することで指定された参照ピクチャを選択する。一旦、参照ピクチャが指定されると、動き補償のためのデコード処理が6.1節で記述されたように進められる。

1ブロック当たり4つの動きベクトルが使用され、MRPAフィールドが1を超える数（2以上）の参照ピクチャの使用を指示している場合、両色差ブロックに対するピクチャの参照インデックスは4つの動きベクトルの最初のものに関連付けられる（動き補償は6.1節に指定される）。

#### U. 4. 5 参照ピクチャバッファリングのための復号器処理

現在の復号されたピクチャのバッファリング方法は、非Bピクチャのための参照ピクチャバッファリングタイプ（RPBT）によって指定される。バッファリングは、ファーストインファーストアウト（“スライディングウインドウ”）モードに従ってもよい。代替りの手段として、順方向チャンネルで符号器によって指定されるカスタマイズ可能な適応型バッファリング（“適応メモリ制御”）操作に従ってもよい。Bピクチャはバッファの中身には影響を及ぼさない。

“スライディングウインドウ”バッファリングタイプは以下のように動作する。まず、復号器は“使用されない”状態のバッファにピクチャを格納できるか決定する。もし、“使用されない”状態のバッファの空き容量が不十分ならば、最大のデフォルト相対的インデックスを持つ短期間ピクチャ（すなわち、バッファ中で最も古い短期間ピクチャ）は“使用されない”にマーキングされなければならない。（サブピクチャ除去の場合に）必要ならば、この処理は、現在の復号ピクチャを保持するために十分なメモリ容量が解放されるまで繰り返される。現在のピクチャはバッファに格納され、デフォルト相対的インデックスとして0が割り当てられる。他の全ての短期間ピクチャのデフォルト相対的インデックスは1増加される。全ての長期間ピクチャのデフォルト相対的インデックスは、（1-削除された短期間ピクチャ数）だけ増加される。

“適応メモリ制御”バッファタイプでは、指定されたピクチャもしくはサブピクチャの領域が明示的にマルチピクチャバッファから取り除かれてもよい。現在の復号ピクチャは最初は短期間ピクチャとみなされるが、デフォルト相対的インデックスが0のバッファに挿入されてもよいし、長期間インデックスに割り当てられてもよいし、符号器によって“使用されない”としてマーキングされてもよい。他の短期間ピクチャが長期間インデックスに割り当てられてもよい。バッファリング処理は以下の方法と機能的に同一な方法により動作しなければならない。つまり、まず現在のピクチャがデフォルト相対的インデックス0でバッファに追加され、他の全てのピクチャのデフォルト相対的インデックスが1増加される。次にMMCOコマンドが以下のように処理される。

もし、MMCOがRESETフィールドを1にすることによりバッファの中身の初期化を指示していれば、現在のピクチャを除いてバッファ内の全ピクチャは“使用されない”としてマーキングされる（U.3.1.5.7小小節で要求されたようにバッファリセットは最初のMMCOコマンドでなければならないため、現在のピクチャのデフォルト相対的インデックスは0になる）。



- ・ もしMMCOがMLIP1により長期間インデックスの最大値を指定していれば、MLIP1以上の長期間インデックスを持つ全ての長期間ピクチャは“使用されない”としてマーキングされる。このとき、残ったピクチャのデフォルト相対的インデックス順序は影響を受けない。
- ・ もしMMCOがマルチピクチャバッファ内のあるピクチャを“使用されない”にマーキングすることを指示し、そのピクチャがまだ“使用されない”とマークされていないならば、指定されたピクチャはマルチピクチャバッファ内で“使用されない”としてマーキングされ、後続する全てのピクチャのデフォルト相対的インデックスは1減少される。
- ・ もしMMCOがマルチピクチャバッファ内のサブピクチャの領域を“使用されない”としてマーキングすることを指示していれば、指定されたサブピクチャの領域は“使用されない”としてマーキングされる。このとき、ピクチャのデフォルト相対的インデックスの順序は影響を受けない。また、U. 3. 1. 5. 10小小節で要求されたように、特定のピクチャの全てのサブピクチャがサブピクチャ除去MMCOコマンドによって“使用されない”とマーキングされることはない（代わりに、符号器はピクチャ全体として“使用されない”となるMMCOコマンドを送るべきである）。
- ・ もしMMCOがある指定された短期間ピクチャへの長期間インデックスの割り当てを指示し、指定された長期間インデックスがまだその指定された短期間ピクチャに割り当てられていなければ、バッファ中の指定された短期間ピクチャは指定された長期間インデックスを持つ長期間ピクチャとしてマーキングされる。もし、指定された長期間インデックスと同じ長期間インデックスを持つ他のピクチャがバッファ内に既にあれば、そのピクチャが“使用されない”としてマーキングされる。デフォルト相対的インデックスの順序で指定された短期間ピクチャに後続する全ての短期間ピクチャと、指定された長期間インデックスより小さい長期間インデックスを持つ長期間ピクチャは、関係するデフォルト相対的インデックスが1減少される。指定されたピクチャは、減少されたデフォルト相対的インデックスの最大値+1の、もしくは、減少されたインデックスがなければ0のデフォルト相対的インデックスに割り当てられる。

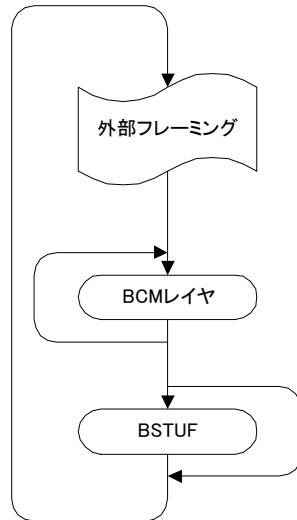
上記操作の結果、“使用されない”としてマーキングされていないピクチャもしくはサブピクチャの領域のバッファ量が最新のSPTN値で指示されたバッファ容量を超過してはならない。もし復号器がこのような条件を検出した場合にはエラーとして扱うべきである。

## U. 5 逆方向チャンネルメッセージ

逆方向チャンネルメッセージの伝送には、アウトバンドチャンネル（必ずしも信頼性のある必要はない）が使われる。（例えば、TTC標準JT-H223やTTC標準JT-H225.0を使用した、独立した論理チャンネルである）このアウトバンドチャンネルのシンタックスはここで定義されたものであるべきである。付属資料Nで定義された逆方向チャンネルメッセージの“ビデオ多重化操作”はERPSモードではサポートされない。

### U. 5. 1 BCM独立論理チャンネルレイヤ

U. 5. 2小節で指定されるように、BCMレイヤは付図U. 8/JT-H263で示されたBCM独立論理チャンネルレイヤによって伝送されるべきである。



付図U. 8 / JT-H263 E-RS-Pモード用BCM独立論理チャンネルレイヤの構造  
(ITU-T H. 263)

#### U. 5. 1. 1 外部フレーミング

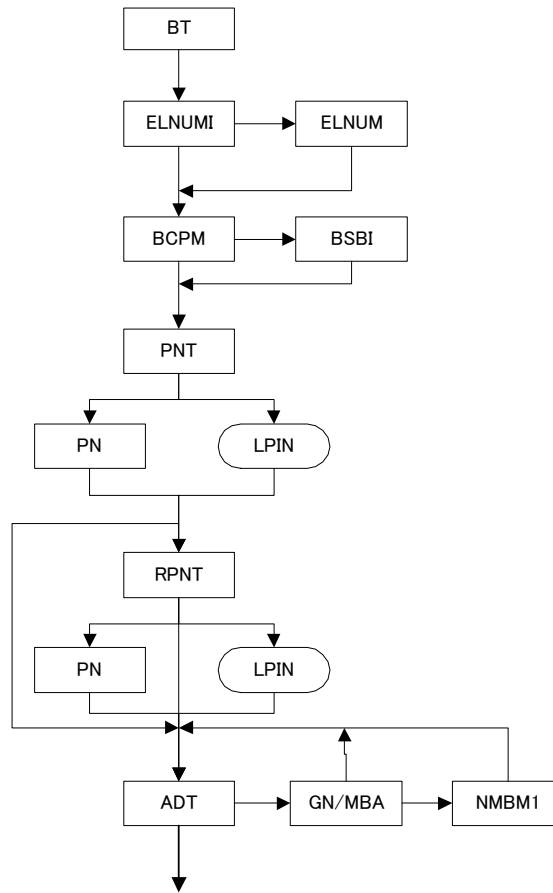
逆方向チャンネルメッセージの外部フレーミングは付図U. 8 / JT-H263に従って提供されるべきである。外部フレーミングは逆方向チャンネルメッセージの開始点とそれに続く逆方向チャンネルメッセージのデータ量を決定するために用いられる。

#### U. 5. 1. 2 逆方向チャンネルスタッフ化 (BSTUF) (可変長)

BSTUFは、外部フレームにおける最後の逆方向チャンネルメッセージの後にのみ現れてよい可変長符号語である。BSTUFは1つ以上の値0のビットからなる可変長符号語から構成される。

#### U. 5. 2 逆方向チャンネルメッセージレイヤシンタックス

本節で定義される逆方向チャンネルメッセージ (BCM) レイヤのシンタックスは、付図U. 9 / JT-H263に示すようでなければならない。



付図U. 9 / JT-H263 ERPSモード用逆方向チャンネルメッセージ (BCM) レイヤの構造 (ITU-T H. 263)

#### U. 5. 2. 1 逆方向チャンネルメッセージタイプ (BT) (2ビット)

BTは逆方向チャンネルメッセージのタイプを示す2ビットの固定長符号語である。BTは各逆方向チャンネルメッセージの最初に存在する。符号器がどのメッセージのタイプを要求したかは、順方向チャンネルのシンタックスのRPSMFフィールドによって示される。BTの値は次のように定義されなければならない。

- “00” : 将来のための予約
- “01” : 将来のための予約
- “10” : NACK : 順方向チャンネルデータの対応する部分の喪失または誤った復号を示す
- “11” : ACK : 順方向チャンネルデータの対応する部分の正常な復号を示す

#### U. 5. 2. 2 高品質化レイヤ番号インジケーション (ELNUMI) (1ビット)

ELNUMIは逆方向チャンネルのBTの後に続く1ビットの固定長符号語である。オプションである時間、SNR、空間スケーラビリティモード (付属資料O参照) が順方向チャンネルデータで使用されており、かつ順方向チャンネルのいくつかの高品質化レイヤが1つの論理チャンネル上で結合されており、かつ、逆方向チャンネルメッセージが (基本レイヤではなく) 高品質化レイヤを参照する場合には、ELNUMIは“1”でなければならない、そうでない場合にはELNUMIは“0”でなければならない。

#### U. 5. 2. 3 高品質化レイヤ番号 (ELNUM) (4ビット)

ELNUMはELNUMIが“1”の場合にのみ存在する4ビットの固定長符号語である。存在するならELNUMはELNUMIの後に続く。存在するとき、ELNUMは逆方向チャンネルメッセージ内で参照される高品質化レイヤのレイヤ番号を含む。

#### U. 5. 2. 4 逆方向チャンネルCPMインジケータ (BCPM) (1ビット)

BCPMは逆方向チャンネルのELNUMIもしくはELNUMの後に続く1ビットの固定長符号語である。順方向チャンネルでCPMモード(5. 2. 4小節と付属資料C参照)が使用されていない場合にはBCPMは“0”でなければならない、使用されている場合には“1”でなければならない。BCPMが“1”ならば、BSBIが存在することを示している。

#### U. 5. 2. 5 逆方向チャンネルサブビット列インジケータ (BSBI) (2ビット)

BSBIは存在すればBCPMに続く2ビットの固定長符号語である。BCPMが“1”のときのみ、BSBIは存在する。BSBIは、逆方向チャンネルメッセージ内で参照する順方向チャンネルデータにおけるサブビット列番号の自然2進数表現である(5. 2. 4小節と付属資料C参照)。

#### U. 5. 2. 6 ピクチャ番号タイプ (PNT) (1ビット)

PNTは常に存在し逆方向チャンネルメッセージのBCPMもしくはBSBIに続く1ビットの固定長符号語である。PNTの値は以下のように定義されなければならない。

“0” : メッセージが短期間ピクチャ番号 (PN) によって指定されたピクチャに関わる

“1” : メッセージが長期間ピクチャインデックス (LPIN) によって指定されたピクチャに関わる

PNTの値に従って、PNもしくはLPINが後に続く。PNとLPINは、順方向チャンネルデータ中での使用に際し、それぞれ、U. 3. 1. 3小節とU. 3. 1. 5. 9小節での指定に従うように表現されなければならない。

#### U. 5. 2. 7 要求ピクチャ番号タイプ (RPNT) (2ビット)

RPNTはBTがNACKメッセージを示す場合のみ存在する2ビットの固定長符号語である。RPNTが存在するときはPNもしくはLPINの後に続き、後続するピクチャの符号化の参照として使用してもよい、マルチピクチャバッファ内のピクチャをどのように識別するかを決定する。RPNTの値は次のように定義されなければならない。

“00” : バッファに有効なピクチャはない。すなわち、バッファはRESETを”1”に設定したIもしくはEIピクチャによって初期化されるべきである。

“01” : 参照として識別される特定のピクチャはない

“10” : 参照に使用してもよいピクチャは短期間ピクチャ番号 (PN) によって識別される

“11” : 参照に使用してもよいピクチャは長期間ピクチャインデックス (LPIN) によって識別される

もし、RPNTが“10”もしくは“11”ならば、RPNTの値に応じて、PNもしくはLPINがRPNTの後に続く。PNとLPINは順方向チャンネルデータ中での使用に際し、それぞれ、U. 3. 1. 3小節とU. 3. 1. 5. 9小節での指定に従うように表現されなければならない。通常は、RPNTを使用して指定されたPNもしくはLPINにより、逆方向チャンネルメッセージ中で識別されたピクチャやサブピクチャ領域と空間的に対応する位置にある最後に正しく復号したピクチャ領域を識別する。

#### U. 5. 2. 8 付加データタイプ (ADT) (2ビット)

ADTはPNもしくはLPINもしくはRPNTの後に存在する2ビットの固定長符号語であり、(ACKメッセージ中に存在する)PNTもしくは(NACKメッセージ中に存在する)RPNTによって決定される。ADTは複数回現れてもよい。ADTは、逆方向チャンネルメッセージを適用する関連したピクチャ領域を識別するために使用される付加データのタイプを指定する。ADTの値は次のように定義されなければならない。

“00” : 付加データの終了

“01” : GN/MBAフィールドによってのみ識別される領域

“10” : GN/MBAとNMBM1によってピクチャの中のラスタ走査領域として識別される領域

“11”：GN/MBAとNMBM1によって矩形スライス中のラスタ走査領域として識別される領域

もし、ADTが“00”ならば、この後に続く逆方向チャンネルメッセージのデータはない。もし、ADTが“01”ならば、ADTの後にGN/MBAが続き、その後に別のADTが続く。もし、ADTが“10”もしくは“11”ならば、ADTの後にGN/MBAとNMBM1が続き、その後に別のADTが続く。

もし、ADTが“10”ならば、領域は、GN/MBAにより指定された特定の空間位置から始まり、ピクチャ内のラスタ走査順で数えて、指定された数のマクロブロックを含む領域として識別される。もし、ADTが“11”ならば、領域は、GN/MBAで指定された特定の空間的位置から始まり、矩形スライス内のラスタ走査順で数えて、指定された数のマクロブロックを含む領域として識別される。もしADTが一回のみ現れ“00”ならば、領域はピクチャ全体として識別される。もし、ADTが1を超える回数（2回以上）現れるならば、“00”は領域を識別するのではなくループの終了のためだけに使用される。

#### U. 5. 2. 9 GOB番号/マクロブロックアドレス (GN/MBA) (5/6/7/9/11/12/13/14ビット)

GN/MBAはGOB番号もしくはマクロブロックアドレスを指定する固定長の符号語である。GN/MBAが存在するときはADTの後に続く。GN/MBAはADTによって指示されたときのみ存在する。オプションのスライス構造モード（付属資料K参照）が使用されていない場合、GN/MBAは逆方向チャンネルメッセージが参照する領域の始まりのGOB番号を含んでいる。オプションのスライス構造モードが使用されている場合、GN/MBA番号は逆方向チャンネルメッセージが参照する領域の始まりのマクロブロックアドレスを含んでいる。このフィールドの長さは、本標準の他の箇所で定義されたGNもしくはMBAの指定に従わなければならない。

#### U. 5. 2. 10 1減じたマクロブロック数 (NMBM1) (5/6/7/9/11/12/13/14ビット)

NMBM1はマクロブロック数を指定する固定長の符号語である。NMBM1はADTによって指示されたときのみ存在し、存在するときはGN/MBAの後に続く。NMBM1は指定されたマクロブロック数から1減じた値の自然表現を含んでいる。このフィールドの長さは、K. 2. 5小節と付表K. 2/JT-H263のマクロブロックアドレスの長さに従わなければならない。

## 付属資料V

### データパーティションスライスモード

(この付属資料は本標準の必須部分である)

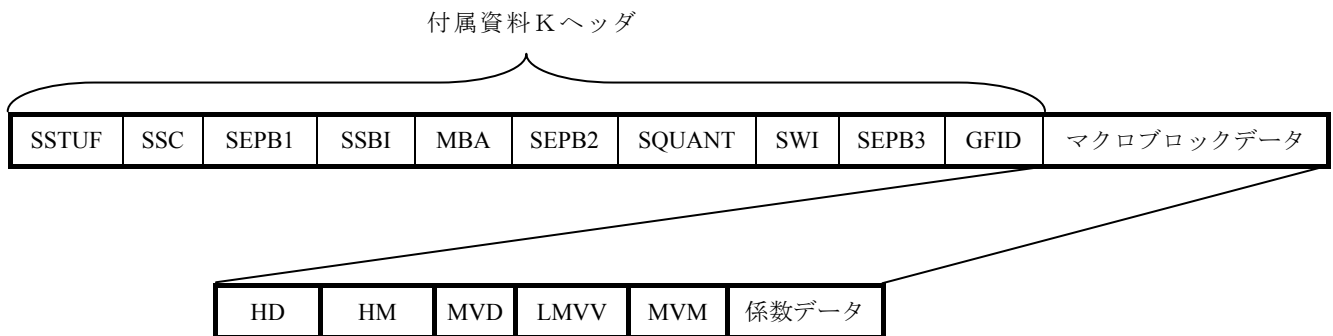
#### V. 1 序論

本付属資料は、TTC標準JT-H263のオプションであるデータパーティションスライス(DPS)モードについて記述している。本モードの能力は、外部手段によって示される(例えば、TTC標準JT-H245)。本モードの使用は、PLUSPTYPE(OPPTYPE)のオプション部分の以前予約されていた第17ビットを“1”にセットすることで、示されなければならない。本モードは付属資料Kに定義されたヘッダ構造を使用する。

データパーティショニングは、誤りの発生しやすい環境での誤り耐性を提供する。これは、伝送中に引き起こされた誤りの早期検出および誤りからの回復を可能にするために、TTC標準JT-H263シンタックスを再配列することにより行われる。

#### V. 2 データパーティショニングの構造

データパーティショニングが使用される場合、R. 2節で定義されるように、データはビデオピクチャセグメントとして配列される。セグメント中の全てのMBのヘッダ情報が一緒に伝送され、その次にそのセグメント中の全てのMBのMVが続き、さらにそのセグメント中のすべてのMBのDCT係数が続くようにセグメントのMBは再配列される。セグメントヘッダは、K. 2節で記述されるのと同じシンタックスを用いる。ヘッダ、MV、及びDCTパーティションは誤りが発生したパーティションの終わりにおいて再同期が可能なようにマーカによって分離される。各セグメントは整数個分のMBのデータを含まなければならない。本モードが使用される時は、付図V. 1/JT-H263で示されるシンタックスが使用されなければならない。



付図V. 1/JT-H263 データパーティショニングシンタックス  
(ITU-T H. 263)

本付属資料を使用しないときは、ビデオピクチャセグメント中のすべてのMBについて、MVとDCTデータがインタリーブされる方法で伝送され、この場合の誤り発生により、通常はパケットの残りのMBの全ての情報を失うことに注意。

##### V. 2. 1 ヘッダデータ(HD)(可変長)

ヘッダデータフィールドは、パケット中の全てのMBに対して、CODとMCBPC情報を含んでいる。更にPBフレーム、または改良PBフレームの場合にはMODBデータも含まれる。パケット中の全てのMBに対して、CODとMCBPCとを結合したリバーシブル可変長符号(RVLC)が使用される。この符号は、付表V. 1/JT-H263から付表V. 5/JT-H263で示される。付属資料Oが使用される場合、CODは、MBTYPEのみと結合され、付表V. 3/JT-H263と付表V. 4/JT-H263を用いてBピクチャとEPピクチャに対するRVLCを形成する。CBPCは付表O. 4/JT-H263の符号語で符号化される。もしCOD=0かつ、付属資料Gまたは付属資料Mが使用される場合は、COD+MCBPCの符号語の直後に、マクロブロックのMODBフィールドに対応するリバーシブル可変長符号化されたデータが続かなければならない。付表

V. 6 / J T - H 2 6 3 は P B フレームに対して、また付表 V. 7 / J T - H 2 6 3 は改良 P B フレームに対して使用されなければならない。

### V. 2. 2 ヘッダマーカ (HM) (9ビット)

ヘッダマーカは 9 ビットの符号語である。値は '1010 0010 1' である。HM はヘッダパーティションを終端する。復号器においてリバース復号が用いられる時、復号器はこのマーカを探索する。この値は通常 HD フィールドには発生しない。

### V. 2. 3 動きベクトルデータレイヤ (可変長)

#### V. 2. 3. 1 動きベクトル差分符号化

動きベクトルに対しては、付表 D. 3 / J T - H 2 6 3 で示される R V L C 符号語が用いられ、動きベクトルと動きベクトル予測値との差分を符号化する。本付属資料は、単に付属資料 D のエントロピー符号化を使用するだけであり、付属資料 D も一緒に使用するのであれば、付属資料 D の他の機能は使用しないことに注意。

#### V. 2. 3. 2 動きベクトル値の予測

パケットにおける最初の動きベクトルは、水平と垂直の両成分とも予測値に 0 を使って符号化される。後続の符号化された MB の MV は、差分動きベクトル (MVD) を使って予測符号化される。これは、スキップされた MB あるいは I N T R A MB に続く MV が水平と垂直の両成分に予測値 0 を使って符号化される場合の MV の符号化方法と異なる。

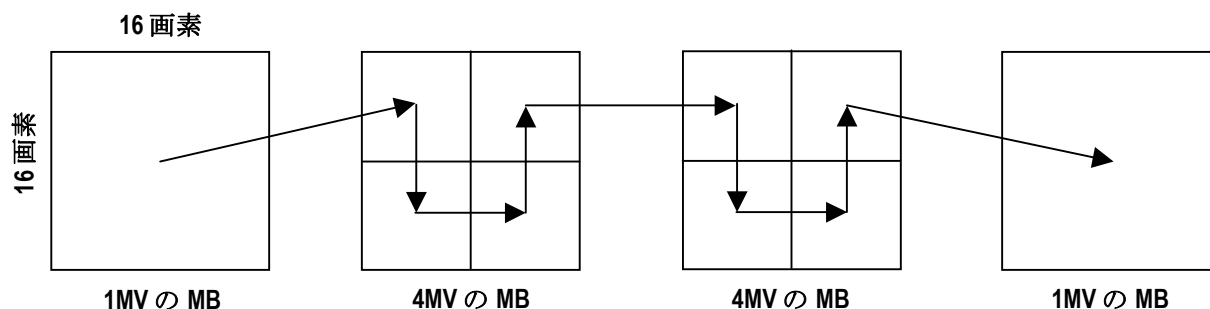
$$\text{前方向: } MV_i = MV_{i-1} + MVD_i = MV_{i-1} + (MV_i - MV_{i-1})$$

$$\text{後方向: } MV_{i-1} = MV_i - MVD_i = MV_i - (MV_i - MV_{i-1})$$

( $MV_i$  と  $MVD_i$  はそれぞれ、パケット中における  $i$  番目の MV と MV 差分を示している)

パケットの最後の動きベクトルに対する動きベクトル情報はこの方法で符号化され、V. 2. 4 小節に記述されるように L M V V フィールドでも再び符号化される。これは復号器が、2 つの異なった予測パスを使用して、一連の MV を独立に復号することを可能にする。つまり、1) 前方向においては パケットの動きデータの先頭から復号を開始し、2) 後方向においてはパケットの中の動きデータの最後から復号を開始する。これは、より良い誤り検出とコンシールメントのための有効手段を提供する。

注: D P S モードが使用されない時は、6. 1. 1 小節で記述されるように隣接する場所の 3 つの動きベクトルの中央値である、現在の動きベクトルの予測値を使って、動きベクトルは予測符号化される。本付属資料のパケットでは、各パケットで符号化された MB の数は可変なので、(フレームの異なる行の動きベクトルを含む) 中間値予測符号化による方法は、スライス中における動きベクトルのリバース復号を妨げる。D P S モードが使用される時は、パケット全体の MV に対し単一の予測スレッドが形成される。これは付図 V. 2 / J T - H 2 6 3 で示される。



付図 V. 2 / J T - H 2 6 3 単一動きベクトル予測  
( I T U - T H. 2 6 3 )

BピクチャあるいはEPピクチャ（付属資料O）の場合、付表V. 3/JT-H263と付表V. 4/JT-H263の中のMBTYPE符号語により示されるように、MVDFWとMVDBWは存在してもよい。MVDFWは上述したのと同じ単一の予測スレッドを用いて予測符号化される。（Bピクチャ中に存在している場合に）MVDBWは、O. 4. 6小節で指定されるように符号化されなければならない。MVDFWとMVDBWは、付表D. 3/JT-H263の符号語で符号化されなければならない。

PBフレーム（付属資料G）と改良PBフレーム（付属資料M）の場合、MVDBデータはそれぞれ対応する付属資料で指定されるように符号化されなければならない、付表D. 3/JT-H263の符号語で符号化されなければならない。

注：後方復号モードがBフレーム（付属資料O）または改良PBフレーム（付属資料M）内で使用される場合は、後方予測に対する動きベクトルデータがパケット境界にまたがって適切に回復されないかも知れないので、MVDBとMVDBWは復号器によって廃棄されるべきである。

### V. 2. 3. 3 動きベクトル差分符号化におけるスタートコードエミュレーションの防止

後方向に対する独立な解析を容易にするため、MVDスタートコードエミュレーションの回避方法が、付属資料DのD. 2節で記述された方法から変更される。MVパーティションは、左から右へと走査されなければならない。また、両方とも1（符号語“000”）に等しい任意の2つのMVDの後には、1つのMVD=0（符号語“1”）を挿入しなければならない。3番目のMVD=1の符号語が（挿入する前の）元のビット列中で2つのMVD=1の符号語に続く場合は、それを、MVパーティションに残っている符号語の中で検出された最初のMVD=1の符号語と見なさなければならない。それを2番目のMVD=1の符号語と見なしてはならない。またその後MVD=0の符号語が挿入されてはならない。これは、2つの連続したMVD=1（符号語“000”）が1つの対を形成する場合（すなわち、最初のMVDが水平成分で2番目が垂直成分である時）のみそのビットが挿入される付属資料Dとは異なる。付属資料Dと付属資料Vが共に使用される場合、付属資料Vのスタートコードエミュレーション回避方法が、D. 2節で記述された方法の代わりに使用されなければならない。

### V. 2. 4 最終動きベクトル値（LMVV）（可変長）

LMVVフィールドはパケット中の最後のMVを含んでいる。これは水平と垂直の両成分とも予測値に0を使って符号化される。パケット中に動きベクトルが無いあるいは1つしかない場合、LMVVは存在してはならない（この固定値ゼロによる予測は、リバーシブル復号を可能とする）。

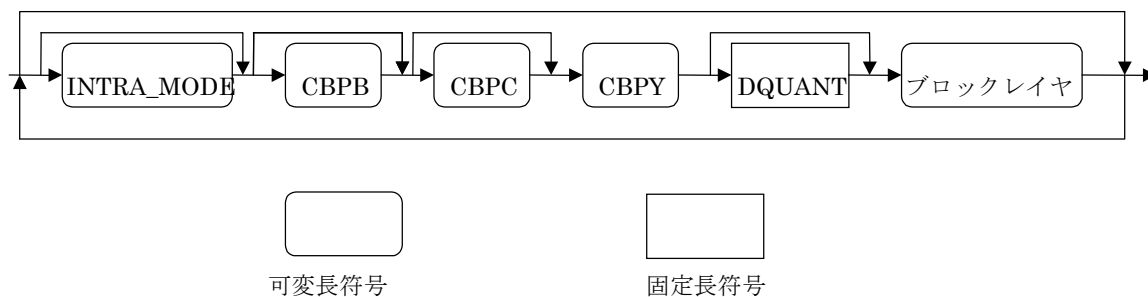
### V. 2. 5 動きベクトルマーカ（MVM）（10ビット）

動きベクトルマーカの10ビットの符号語の値は‘0000 0000 01’である。MVMは動きベクトルパーティションを終端する。復号器においてリバース復号が用いられる時、復号器はこのマーカを探索する。動きベクトルマーカ（MVM）は、パケット中に動きベクトルデータを含んでいないとき（パケット中の全てのマクロブロックがINTRA符号化された時、あるいはCODが1に等しい時）、そのパケットに存在してはならない。

### V. 2. 6 係数データレイヤ（可変長）

DCTデータレイヤは、それぞれI. 2節、5. 3. 4小節、O. 4. 3小節、5. 3. 5小節、5. 3. 6小節、及び5. 4. 2小節で指定される、INTRA\_\_MODE（存在する時）、CBPB（存在する時）、CBPC（存在する時）、CBPY、DQUANT（存在する時）、及びDCT係数を含んでいる。DCTデータのシンタックス図を付図V. 3/JT-H263に示す。CBPCの存在は付表V. 3/JT-H263と付表V. 4/JT-H263で示される。





付図V. 3 / JT-H263 係数データシンタックス  
(ITU-T H. 263)

### V. 3 他のオプションモードとの相互作用

DPSモードは付属資料Kのスライス構造モードのサブモードとして有効に働き、またその外側のピクチャとスライスヘッダ構造を用いている。そのためDPSモードが使用されている時はいつでもSSモードが使用されていると示されなければならない。スライス構造モードの他の両方のサブモード（任意スライス順序と矩形スライスサブモード）は、DPSモードと一緒に使われてもよい。

付属資料Eのシンタックス算術符号化モードは、リバーシブル復号が認められていないので、本付属資料と同時に使われてはならない。

付属資料Hの前方誤り訂正は、ビット列を望ましくない場所で分裂させてしまうので、本付属資料と同時に使われるべきではない。しかし、付属資料Hで定義されたFECは既存の標準システムの中で必要とされているので、DPSモードと一緒に付属資料Hを利用することは禁止されない。

付属資料Oの時間、SNR、及び空間スケーラビリティ（TSSS）モードは、DPSモードと一緒に利用されてもよい。TSSSモードとDPSモードと一緒に使われる場合は、付属資料Oで定義される表に代わって付表V. 3 / JT-H263、付表V. 4 / JT-H263、及び付表V. 5 / JT-H263で指定される符号語を用いなければならない。

付属資料Uは本付属資料と同時に使用されてはならない。

付表V. 1 / JT-H263 INTRA MBに対するCOD+MCBPCのRVLC表  
(ITU-T H. 263)

MB タイプ	CBPC (56)	符号語 (COD+MCBPCの結 合に対する)	ビット数
3 (INTRA)	00	1	1
3	01	010	3
3	10	0110	4
3	11	01110	5
4 (INTRA+Q)	00	00100	5
4	01	011110	6
4	10	001100	6
4	11	0111110	7
スタッフ化		0011100	7

付表V. 2 / JT-H263 INTER MBに対するCOD+MCBPCのRVLC表  
(ITU-T H. 263)

MB タイプ	CBPC (56)	符号語 (COD+MCBPC の 結 合に対する)	ビット数
スキップ		1	1
0 (INTER)	00	010	3
0	10	00100	5
0	01	011110	6
0	11	0011100	7
1 (INTER + Q)	00	01110	5
1	10	00011000	8
1	01	011111110	9
1	11	0111111110	11
2 (INTER4V)	00	0110	4
2	10	01111110	8
2	01	00111100	8
2	11	000010000	9
3 (INTRA)	00	001100	6
3	11	0001000	7
3	10	001111100	9
3	01	000111000	9
4 (INTRA + Q)	00	0111110	7
4	11	0011111100	10
4	10	0001111000	10
4	01	0000110000	10
5 (INTER4V + Q)	00	00111111100	11
5	01	00011111000	11
5	10	00001110000	11
5	11	00000100000	11
スタッフ化		011111110	10

付表V. 3 / JT-H263 B MBに対するMBTYPEのRVLC表  
(ITU-T H. 263)

インデックス	予測タイプ	MVDFW	MVDBW	CBPC + CBPY	DQUANT	MBTYPE	ビ ッ ト
—	直接 (スキップ)					1 (COD=1)	1
0	直接			X		010	3
1	直接 + Q			X	X	001100	6
2	前方 (テクスチャなし)	X				00100	5
3	前方	X		X		011110	6
4	前方 + Q	X		X	X	01111110	8
5	後方 (テクスチャなし)		X			0110	4
6	後方		X	X		01110	5
7	後方 + Q		X	X	X	00111100	8
8	双方向 (テクスチャなし)	X	X			0011100	7
9	双方向	X	X	X		0001000	7
10	双方向 + Q	X	X	X	X	0111110	7
11	INTRA			X		00011000	8
12	INTRA + Q			X	X	011111110	9
13	スタッフ化					001111100	9

付表V. 4 / JT-H263 EP MBに対するMBTYPEのRVLC表  
(ITU-T H. 263)

インデックス	予測タイプ	MVDFW	MVDBW	CBPC + CBPY	DQUANT	MBTYPE	ビット
—	前方 (スキップ)					1 (COD=1)	1
0	前方	X		X		010	3
1	前方 + Q	X		X	X	0110	4
2	上方 (テクスチャなし)					01110	5
3	上方			X		00100	5
4	上方 + Q			X	X	011110	6
5	双方向 (テクスチャなし)					001100	6
6	双方向	X		X		0111110	7
7	双方向 + Q	X		X	X	0011100	7
8	INTRA			X		0001000	7
9	INTRA + Q			X	X	01111110	8
10	スタッフ化					00111100	8

付表V. 5 / JT-H263 EI MBに対するCOD+MCBPCのRVLC表  
(ITU-T H. 263)

予測タイプ	QCBP (56)	符号語 (COD+MCBPC の結 合に対する)	ビット数
上方 (スキップ)		1	1
0 (上方)	00	010	3
0	01	0110	4
0	10	01110	5
0	11	00100	5
1 (上方 + Q)	00	011110	6
1	01	001100	6
1	10	0111110	7
1	11	0011100	7
2 (INTRA)	00	0001000	7
2	01	01111110	8
2	10	00111100	8
2	11	00011000	8
3 (INTRA + Q)	00	011111110	9
3	01	001111100	9
3	10	000111000	9
3	11	000010000	9
スタッフ化		0111111110	10

付表V. 6 / JT-H263 B MODBに対するRVLC表  
(ITU-T H. 263)

インデックス	CBPB	MVDB	ビット数	符号
0			3	010
1		x	4	0110
2	x	x	5	01110

注：“x”はマクロブロック中に存在することを示す。

付表V. 7 / JT-H263 改良PBフレームモードのMODBに対するRVLC表  
(ITU-T H. 263)

インデックス	CBPB	MVDB	ビット数	符号	符号モード
0			3	010	双方向予測
1	x		4	0110	双方向予測
2		X	5	01110	前方予測
3	x	X	5	00100	前方予測
4			6	011110	後方予測
5	x		6	001100	後方予測

注：上の表中“x”は関連するシンタックス要素が存在することを示す。

## 付属資料W

### 追加付加拡張情報仕様

(この付属資料は本標準の必須部分である)

#### W. 1 序論

この付属資料は、TTC標準JT-H263のピクチャレイヤのPSUPPで送られる追加付加拡張情報のフォーマットを記述し、その追加付加拡張情報は付属資料Lで定義された機能を拡張する。この付属資料で記述される能力のいくつか、またはすべての能力を持つ復号器の能力は、外部手段（例えば、TTC標準JT-H245）で通知されてもよい。追加能力を持たない復号器は単純に、ビット列中に現れる新しく定義されたどのPSUPP情報ビットを破棄してもよい。この付加拡張情報の存在は、PEIビットとそれに続くPSUPPオクテットの両者の存在により示される。PSUPPオクテットのFTYPEフィールドは、2つの新たに定義された値の内1つの値を持つ。PEI、PSUPP、FTYPEそしてDSIZEの基本的な解釈は、付属資料L、5.1.24小節、5.1.25小節と同一である。

#### W. 2 参照

以下の勧告及びその他の標準は、本標準で参照することにより本標準の規定となる事項を規定している。出版時には以下に示す版が有効であった。すべての勧告／標準及びその他の標準は、改定されることがあり、本標準を使用する場合は、以下に示す勧告／標準及びその他の標準の最新版が適用可能かどうかを調べるのが望ましい。現在の有効なITU-T勧告／TTC標準のリストが定期的な出版されている。本標準内の文書への参照は、それに単体文書としての勧告／標準のステータスを与えるものではない。

- ISO/IEC 10646:2003, Information technology - Universal Multiple Octet Coded Character Set (UCS).
- IETF RFC 2396 (1998), Uniform Resource Identifiers (URI): Generic Syntax.

#### W. 3 追加FTYPE値

付属資料Lの付表L.1で予約されていた2つの値は、以下の様に定義される。

付表W.1 / JT-H263 FTYPE機能タイプ値  
(ITU-T H.263)

13	固定小数点IDCT
14	ピクチャメッセージ

#### W. 4 PSUPPオクテットの推奨される最大値

この付属資料で定義される前述のFTYPE機能の何れかを使用するとき、1ピクチャ当たりのPSUPPオクテットの総数は、符号化ピクチャサイズとの関係の中で適度に少なく保たれるべきであり、さらに、符号化ピクチャサイズに関わりなく256オクテットを超えるべきではない。

注：ビデオビット列の伝達に使用される幾つかのデータ伝送手順は、誤り耐性を目的として、ピクチャヘッダの内容の外部での繰り返しを提供するかも知れず、ピクチャヘッダから繰り返されるこの様なデータの総計に制限を与えるかも知れない（例えば、IETF RFC 2429パケット化フォーマットの504ビット）。大量のPSUPPオクテットを含まうとすると、ピクチャヘッダ内容の完全な繰り返しを提供できる様な外部手順は無いかも知れない。

#### W. 5 固定小数点IDCT

固定小数点IDCT機能は、ビット列の構成において特定のIDCT近似が使用されていることを示している。DSIZEは、固定小数点IDCT機能では1に等しくなければならない。それに続くPSUPPデータのオクテットは特定のIDCTの実装を指定する。値0はW.5.3小節に示される参照IDCT0を示し、値1から255は予約されている。

### W. 5. 1 復号器の動作

特定の固定小数点IDCTを実行する復号器の能力は、外部手段（例えば、TTC標準JT-H245）により符号器に送られるかも知れない。固定小数点IDCTのインジケーションのある符号化ビット列を受信するとき、もし特定の固定小数点IDCT処理能力が有るなら、復号器はそれを使用しなければならない。

### W. 5. 2 強制画面更新の除去

付属資料Aは逆離散コサイン変換（IDCT）の精度に対する要求条件を指定しており、それに準拠する多数の実装を認めている。符号器と復号器のIDCTミスマッチによる誤差の蓄積を制御するために、4.4節の強制画面更新は、係数を伝送するときには少なくとも132回に1回はマクロブロックをINTRAモードで符号化することを要求している。もし、ビット列内で固定小数点IDCT機能タイプが示されているなら、強制画面更新は必要でなく、INTRA符号化の頻度は規制されない。しかし、復号器がここで指定される特定の固定小数点IDCTの能力を持つことが外部手段により確認されていない限り、符号器は強制画面更新を使用し続けるべきである。さもなければ、ミスマッチが生じるかも知れない。

### W. 5. 3 参照IDCT0

参照IDCT0は、すべての入力ブロックに対して、以下に載せられたCソースプログラムと同一の出力値を作り出す任意の実装である。

注：この固定小数点IDCTは付属資料Aに準拠するが、TTC標準JT-H262 | ISO/IEC13818-2の付属資料Aにある、拡張された値の範囲の要求条件には準拠しない。

```
/*
 *
 *          FIXED-POINT IDCT
 *
 * Fixed-point fast, separable idct
 * Storage precision: 16 bits signed
 * Internal calculation precision: 32 bits signed
 * Input range: 12 bits signed, stored in 16 bits
 * Output range: [-256, +255]
 * All operations are signed
 *
 */

/*
 * Includes
 */

#include <stdlib.h>
#include <stdio.h>

/*
 * Typedefs
 */

typedef short int REGISTER; /* 16 bits signed */
typedef long int LONG; /* 32 bits signed */

/*
 * Global constants
 */

const REGISTER cpo8   = 0x539f; /* 32768*cos(pi/8)*1/sqrt(2) */
const REGISTER spo8   = 0x4546; /* 32768*sin(pi/8)*sqrt(2) */
const REGISTER cpo16  = 0x7d8a; /* 32768*cos(pi/16) */
const REGISTER spo16  = 0x18f9; /* 32768*sin(pi/16) */
const REGISTER c3po16 = 0x6a6e; /* 32768*cos(3*pi/16) */
const REGISTER s3po16 = 0x471d; /* 32768*sin(3*pi/16) */
const REGISTER OoR2   = 0x5a82; /* 32768*1/sqrt(2) */

/*
 * Function declarations
 */

void Transpose(REGISTER block[64]);
void HalfSwap(REGISTER block[64]);
void Swap(REGISTER block[64]);
void Scale(REGISTER block[64], signed char sh);
void Round(REGISTER block[64], signed char sh,
           const REGISTER min, const REGISTER max);
REGISTER Multiply(const REGISTER a, REGISTER x, signed char sh);
```

```

void Rotate(REGISTER *x, REGISTER *y,
            signed char sha, signed char shb,
            const REGISTER a, const REGISTER b,
            int inv);
void Butterfly(REGISTER column[8], char pass);
void IDCT(REGISTER block[64]);

/*
 * Transpose():
 *   Transpose a block
 * Input:
 *   REGISTER block[64]
 * Output:
 *   block
 * Return value:
 *   none
 */
void Transpose(REGISTER block[64])
{
    int i, j;
    REGISTER temp;

    for (i=0; i<8; i++) {
        for (j=0; j<i; j++) {
            temp = block[8*i+j];
            block[8*i+j] = block[8*j+i];
            block[8*j+i] = temp;
        }
    }
    return;
}

/*
 * HalfSwap():
 *   One-dimensional swap
 * Input:
 *   REGISTER block[64]
 * Output:
 *   block
 * Return value:
 *   none
 */
void HalfSwap(REGISTER block[64])
{
    int i;
    REGISTER temp;

    for (i=0; i<8; i++) {
        temp = block[8+i];
        block[8+i] = block[32+i];
        block[32+i] = temp;
        temp = block[24+i];
        block[24+i] = block[48+i];
        block[48+i] = temp;
        temp = block[40+i];
        block[40+i] = block[56+i];
        block[56+i] = temp;
    }
    return;
}

/*
 * Swap():
 *   Swap and transpose a block
 * Input:
 *   REGISTER block[64]
 * Output:
 *   block
 * Return value:
 *   none
 */
void Swap(REGISTER block[64])
{
    HalfSwap(block);
    Transpose(block);
    HalfSwap(block);
}

/*
 * Scale():
 *   Scale a block
 * Input:
 *   REGISTER block[64]
 *   signed char sh
 * Output:
 *   block
 * Return value:
 *   none
 */
void Scale(REGISTER block[64], signed char sh)
{

```

```

int i;

if (sh>0) {
    for (i=0; i<64; i++)
        block[i] >>= sh;
}
else {
    for (i=0; i<64; i++)
        block[i] <<= -sh;
}
}

/*
 * Round():
 * Performs the final rounding of an 8x8 block
 * Input:
 * REGISTER block[64]
 * signed char sh
 * const REGISTER min
 * const REGISTER max
 * Output:
 * block
 * Return value:
 * none
 */
void Round(REGISTER block[64], signed char sh,
           const REGISTER min, const REGISTER max)
{
    int i;

    for (i=0; i<64; i++) {
        if (block[i] < 0x00007FFF - (1<<(sh-1)))
            block[i] += (1<<(sh-1));
        else
            block[i] = 0x00007FFF;
        block[i] >>= sh;
        block[i] = (block[i]<min) ? min : ((block[i]>max) ? max : block[i]);
    }
    return;
}

/*
 * Multiply():
 * Multiply by a constant with shift
 * Input:
 * const REGISTER a
 * REGISTER x
 * signed char sh
 * Output:
 * none
 * Return value:
 * REGISTER, the result of the multiply
 */
REGISTER Multiply(const REGISTER a, REGISTER x, signed char sh)
{
    LONG tmp;
    REGISTER reg_out;

    /* multiply */
    tmp = (LONG)a * (LONG)x;

    /* shift */
    if (sh > 0)
        tmp >>= sh;
    else
        tmp <<= -sh;

    /* rounding and saturating */
    if (tmp < 0x7FFFFFFF - 0x00007FFF)
        tmp = tmp + 0x00007FFF;
    else
        tmp = 0x7FFFFFFF;

    reg_out = (REGISTER)(tmp >>16);

    return(reg_out);
}

/*
 * Rotate():
 * Perform rotate operation on two registers
 * Input:
 * REGISTER *x          pointer to the 1st register
 * REGISTER *y          pointer to the 2nd register
 * signed char sha      shift associated with factor a
 * signed char shb      shift associated with factor b
 * const REGISTER a     factor a
 * const REGISTER b     factor b
 * int inv              1 for inverse det, 0 for forward det
 * Output:
 * *x, *y

```



```

* Return value:
*   none
*/
void Rotate(REGISTER *x, REGISTER *y,
            signed char sha, signed char shb,
            const REGISTER a, const REGISTER b,
            int inv)
{
    LONG tmp1xa, tmp1ya, tmp1xb, tmp1yb;
    LONG tmp11, tmp12;

    /*
     * intermediate calculation
     */

    tmp1xa = (LONG)(*x) * (LONG)a;
    if (sha > 0)
        tmp1xa >>= sha;
    else
        tmp1xa <<= -sha;

    tmp1ya = (LONG)(*y) * (LONG)a;
    if (sha > 0)
        tmp1ya >>= sha;
    else
        tmp1ya <<= -sha;

    tmp1xb = (LONG)(*x) * (LONG)b;
    if (shb > 0)
        tmp1xb >>= shb;
    else
        tmp1xb <<= -shb;

    tmp1yb = (LONG)(*y) * (LONG)b;
    if (shb > 0)
        tmp1yb >>= shb;
    else
        tmp1yb <<= -shb;

    /*
     * rounding and rotation
     */

    if (inv) {
        tmp1xa += 0x00007FFF;
        tmp1xb += 0x00007FFF;

        tmp11 = tmp1xb - tmp1ya;
        tmp12 = tmp1xa + tmp1yb;
    }
    else {
        tmp1ya += 0x00007FFF;
        tmp1yb += 0x00007FFF;

        tmp11 = tmp1xb + tmp1ya;
        tmp12 = -tmp1xa + tmp1yb;
    }

    /*
     * final rounding
     */

    *x = (REGISTER) (tmp11 >>16);
    *y = (REGISTER) (tmp12 >>16);

    return;
}

/*
 * Butterfly():
 *   Perform 1D IDCT on a column
 * Input:
 *   REGISTER column[8]
 *   char pass
 * Output:
 *   column
 * Return value:
 *   none
 */
void Butterfly(REGISTER column[8], char pass)
{
    int i;
    REGISTER shadow_column[8];

    /*
     * For readability, we use a shadow column
     * that contains the state of column at the
     * preceding stage of the butterfly.
     */

    /*

```

```

* Initialization
*/
for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
* First Phase
*/

Rotate(column+2, column+6, pass-2, pass-1, cpo8, spo8, 1);
Rotate(column+1, column+7, pass-1, pass-1, cpo16, spo16, 1);
Rotate(column+3, column+5, pass-1, pass-1, c3po16, s3po16, 1);

if (pass) {
    int a, tmp=column[4], b=column[0];
    a = b+tmp;
    b = b-tmp;
    column[0] = (a - ((tmp<0) ? 1 : 0)) >> 1;
    column[4] = (b - ((tmp<0) ? 1 : 0)) >> 1;
}
else {
    column[0] = shadow_column[0] + shadow_column[4];
    column[4] = shadow_column[0] - shadow_column[4];
}

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
* Second Phase
*/

column[1] = shadow_column[1] - shadow_column[3];
column[3] = shadow_column[1] + shadow_column[3];

column[7] = shadow_column[7] - shadow_column[5];
column[5] = shadow_column[7] + shadow_column[5];

column[0] = shadow_column[0] + shadow_column[6];
column[6] = shadow_column[0] - shadow_column[6];

column[4] = shadow_column[4] + shadow_column[2];
column[2] = shadow_column[4] - shadow_column[2];

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
* Third Phase
*/

column[7] = shadow_column[7] - shadow_column[3];
column[3] = shadow_column[7] + shadow_column[3];

column[1] = Multiply(0oR2, shadow_column[1], -2);
column[5] = Multiply(0oR2, shadow_column[5], -2);

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
* Fourth Phase
*/

column[4] = shadow_column[4] + shadow_column[3];
column[3] = shadow_column[4] - shadow_column[3];

column[2] = shadow_column[2] + shadow_column[7];
column[7] = shadow_column[2] - shadow_column[7];

column[0] = shadow_column[0] + shadow_column[5];
column[5] = shadow_column[0] - shadow_column[5];

column[6] = shadow_column[6] + shadow_column[1];
column[1] = shadow_column[6] - shadow_column[1];

return;
}

/*
* IDCT():
* Perform 2D IDCT on a block
* Input:
* REGISTER block[64]
* Output:
* block
* Return value:
* none
*/
void IDCT(REGISTER block[64])

```

```

{
  int i;
  Scale(block, -4);
  for (i=0; i<8; i++)
    Butterfly(block+8*i, 0);
  Transpose(block);
  for (i=0; i<8; i++)
    Butterfly(block+8*i, 1);
  Round(block, 6, -256, 255);
  Swap(block);
}

```

情報を与える目的で、関連する順方向離散コサイン変換 (FDCT) の実装を以下に示す。この固定小数点FDCTは、この標準の必須部分ではない。

```

/*****
 *
 *                               FIXED-POINT FDCT
 *
 * Fixed-point fast, separable fdct
 * Storage precision: 16 bits signed
 * Internal calculation precision: 32 bits signed
 * Input range: 9 bits signed, stored in 16 bits
 * Output range: [-2048, +2047]
 * All operations are signed
 *
 *****/

/*
 * Function declarations
 */

void FButterfly(REGISTER column[8]);
void FDCT(REGISTER block[64]);

/*
 * FButterfly():
 *   Perform 1D FDCT on a column
 * Input:
 *   REGISTER column[8]
 * Output:
 *   column
 * Return value:
 *   none
 */
void FButterfly(REGISTER column[8])
{
  int i;
  REGISTER shadow_column[8];

  /*
   * For readability, we use a shadow column
   * that contains the state of column at the
   * preceding stage of the butterfly.
   */

  /*
   * Initialization
   */
  for (i=0; i<8; i++)
    shadow_column[i] = column[i];

  /*
   * First Phase
   */
  for (i=0; i<4; i++) {
    column[i] = shadow_column[i] + shadow_column[7-i];
    column[7-i] = shadow_column[i] - shadow_column[7-i];
  }

  for (i=0; i<8; i++)
    shadow_column[i] = column[i];

  /*
   * Second Phase
   */
  column[0] = shadow_column[0] + shadow_column[3];
  column[3] = shadow_column[0] - shadow_column[3];
}

```

```

column[1] = shadow_column[1] + shadow_column[2];
column[2] = shadow_column[1] - shadow_column[2];

column[4] = Multiply(0oR2, shadow_column[4], -2);
column[7] = Multiply(0oR2, shadow_column[7], -2);

column[6] = shadow_column[6] - shadow_column[5];
column[5] = shadow_column[6] + shadow_column[5];

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Third Phase
 */

column[0] = shadow_column[0] + shadow_column[1];
column[1] = shadow_column[0] - shadow_column[1];

column[6] = shadow_column[6] - shadow_column[4];
column[4] = shadow_column[6] + shadow_column[4];

column[7] = shadow_column[7] - shadow_column[5];
column[5] = shadow_column[7] + shadow_column[5];

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Fourth Phase
 */

Rotate(column+2, column+3, -2, -1, cpo8, spo8, 0);
Rotate(column+4, column+5, -1, -1, cpo16, spo16, 0);
Rotate(column+6, column+7, -1, -1, c3po16, s3po16, 0);

return;
}

/*
 * FDCT():
 * Perform 2D FDCT on a block
 * Input:
 * REGISTER block[64]
 * Output:
 * block
 * Return value:
 * none
 */
void FDCT(REGISTER block[64])
{
    int i;

    for (i=0; i<8; i++)
        FButterfly(block+8*i);

    Transpose(block);

    for (i=0; i<8; i++)
        FButterfly(block+8*i);

    Round(block, 3, -2048, 2047);

    Swap(block);
}

```

## W. 6 ピクチャメッセージ

ピクチャメッセージ機能は、メッセージデータを表現する1またはそれ以上のオクテットの存在を示す。メッセージデータの最初のオクテットは、以下の構造をもつメッセージヘッダである。

CONT	EBIT	MTYPE
------	------	-------

付図W. 1 / J T - H 2 6 3 最初のメッセージオクテット構造  
( I T U - T H . 2 6 3 )

D S I Z Eは、付図W. 1 / J T - H 2 6 3に示された最初のオクテットを含んだ、ピクチャメッセージ機能に対応するメッセージデータのオクテット数に等しくなければならない。

復号器は、基本P S U P Pシンタックスにより要求されるように、ピクチャメッセージデータを文法解析しなければならないが、復号器のピクチャメッセージへの応答は、定義されていない。

### W. 6. 1 継続 (CONT) (1ビット)

もし“1”に等しいなら、CONTはこのピクチャメッセージ機能に関連するメッセージデータが、次のピクチャメッセージデータ機能に関連するメッセージデータと同じ論理メッセージの一部であることを示す。もし“0”に等しいなら、CONTは、このピクチャメッセージ機能に関連するメッセージデータが、現在の論理メッセージの終わりであることを示す。CONTは、例えば、14オクテットを越える論理メッセージを表現するために使用されても良い。

### W. 6. 2 終了ビット位置またはトラック番号 (EBIT) (3ビット)

非テキストピクチャメッセージに対して、EBITは、最後のメッセージオクテットにおいて無視されなければならない最下位ビットの数を示す。非テキストピクチャメッセージでは、もしCONTが“1”なら、または、もし1メッセージオクテットのみ（すなわち、付図W. 1/JT-H263のオクテット）なら、EBITは“0”に等しくなければならない。CONT/EBIT/MTYPEを除いた非テキストメッセージ機能に対する有効なメッセージビットの数は、 $(DSIZE-1) \cdot 8 - EBITS$ に等しい。論理メッセージに対する有効メッセージビット数は継続によりさらに大きいかも知れない。

テキスト情報を含むピクチャメッセージタイプに対して、EBITはテキストトラック番号を含まなければならない。テキストトラック番号の正確な意味はここでは指定されないが、テキストに対する特定のタイプ（例えば、言語）を示すべきである。トラック番号ゼロはデフォルトトラックと考えるべきである。

### W. 6. 3 メッセージタイプ (MTYPE) (4ビット)

MTYPEはメッセージのタイプを示す。定義されているタイプは付表W. 2に示される。

付表W. 2/JT-H263 MTYPEメッセージタイプ値  
(ITU-T H. 263)

0	任意バイナリデータ
1	任意テキスト
2	著作権テキスト
3	字幕テキスト
4	ビデオ記述テキスト
5	同型リソース識別子テキスト
6	現ピクチャヘッダ繰り返し
7	前ピクチャヘッダ繰り返し
8	次ピクチャヘッダ繰り返し (リライアブルTR)
9	次ピクチャヘッダ繰り返し (アンリライアブルTR)
10	トップインタレースフィールドインジケーション
11	ボトムインタレースフィールドインジケーション
12	ピクチャ番号
13	予備参照ピクチャ
14..15	予約

#### W. 6. 3. 1 任意バイナリデータ

任意バイナリデータは、ISO/IEC 10646 UTF-8で符号化されていないバイナリメッセージを伝えるために使用される。任意バイナリデータの内容の解釈はこの標準の範囲外であるが、このようなデータの1つのタイプを他から区別するのを助けるためのある識別パターン（例えば、4オクテット識別符号）で開始すべきである。

#### W. 6. 3. 2 任意テキスト

任意テキストは、一般的なISO/IEC 10646 UTF-8で符号化されたテキストメッセージを伝えるために使用される。著作権情報のようなより特定のテキストメッセージは、最適な他のメッセージタイプ（例えば、著作権テキスト）で表現されるべきである。

### W. 6. 3. 3 著作権テキスト

著作権テキストは、ビット列の中の情報源または符号化された表現物に関する知的財産情報を伝えるためだけに使用されなければならない。著作権メッセージは ISO/IEC 10646 UTF-8 により符号化されなければならない。

### W. 6. 3. 4 字幕テキスト

字幕テキストは、ビット列の現在及びその後のピクチャに関する字幕情報を伝えるためのみに使用されなければならない。字幕メッセージは ISO/IEC 10646 UTF-8 により符号化されなければならない。字幕テキストは、別個のテキスト領域において、新しいテキストが前のテキストの終わりに追加され、古いテキストが挿入位置からスクロールされるよう表示されるかのようにビット列に挿入されなければならない。Form Feed 制御コード（16進の“0x000C”）は可視テキスト領域のクリアを示すことに使用されなければならない。End of Medium 制御コード（16進の“0x0019”）は“字幕オフ”状態を示すことに使用されなければならない。しかし、この標準はどのような字幕テキストが実際に表示され、そして蓄積されるかを制限しない。

### W. 6. 3. 5 ビデオ記述テキスト

ビデオ記述テキストは、現在のビット列の内容情報に関係する記述情報を伝えるためのみに使用されなければならない。ビデオ記述は ISO/IEC 10646 UTF-8 により符号化されなければならない。ビデオ記述テキストは、別個のテキスト領域において、新しいテキストが前のテキストの終わりに追加され、古いテキストが挿入位置からスクロールされるよう表示されるかのようにビット列に挿入されなければならない。Form Feed 制御コード（16進の“0x000C”）は可視テキスト領域のクリアを示すことに使用されなければならない。End of Medium 制御コード（16進の“0x0019”）は“記述オフ”状態を示すことに使用されなければならない。しかし、この標準はどのようなビデオ記述が実際に表示され、そして蓄積されるかを制限しない。

### W. 6. 3. 6 同型リソース識別子テキスト (URI)

メッセージは、IETF RFC 2394 で定義されているような同型リソース識別子テキストから成る。URI は ISO/IEC 10646 UTF-8 により符号化されなければならない。

### W. 6. 3. 7 現ピクチャヘッダ繰り返し

現在のピクチャのピクチャヘッダがこのメッセージ内で繰り返される。繰り返されるビットは、付加拡張情報 (PEI/PSUPP) を除外する。W. 4 の限定を条件として、GOB またはスライスレイヤまでのすべての他のビットは含まれるべきである。

### W. 6. 3. 8 前ピクチャヘッダ繰り返し

以前に伝送されたピクチャのピクチャヘッダがこのメッセージ内で繰り返される。繰り返されるビットは、ピクチャスタートコード (PSC) の最初の 2 バイトと付加拡張情報 (PEI/PSUPP) を含まない。W. 4 の限定を条件として、GOB またはスライスレイヤまでのすべての他のビットは含まれるべきである。

### W. 6. 3. 9 次ピクチャヘッダ繰り返し (リライアブル TR)

これから送られる次のピクチャのピクチャヘッダがこのメッセージ内で繰り返される。繰り返されるビットは、ピクチャスタートコード (PSC) の最初の 2 バイトと付加拡張情報 (PEI/PSUPP) を含まない。W. 4 の限定を条件として、GOB またはスライスレイヤまでのすべての他のビットは含まれるべきである。

### W. 6. 3. 10 次ピクチャヘッダ繰り返し (アンリライアブル TR)

これから送られる次のピクチャのピクチャヘッダがこのメッセージ内で繰り返される。繰り返されるビットは、ピクチャヘッダの最初の 3 バイトと付加拡張情報 (PEI/PSUPP) を含まない。W. 4 の限定を条件として、GOB またはスライスレイヤまでのすべての他のビットは含まれるべきである。繰り返されるピクチャヘッダの任意の TR または ETR ビットは、次のピクチャヘッダの対応するビットと同一である必要はない。

### W. 6. 3. 11 インタレースフィールドインジケーション

インタレースフィールドインジケーションの場合、メッセージはインタレースフィールド符号化のインジケーションからなる。このインジケーションは復号処理には影響しない。しかし、それは現在のピクチャが、実際にはプログレッシブ画像のように走査されていないことを示す。言い換えれば、それは、現在符号化されているピクチャがフル解像度の情報源画像の半分のラインしか含まないことを示す。インタレースフィールドインジケーションのために、DSIZEは1、CONTは0、そしてEBITは0でなければならない。インタレースフィールド符号化の場合、テンポラルリファレンスの各々の増加は、完全な2つの画像間の時間でなく、1つの画像の交互の半画像フィールド標本の間の時間を意味する。トップインタレースフィールドインジケーションの場合、現在のピクチャが完全な画像の1番目（すなわちトップ）、3番目、5番目等のラインを含む。ボトムインタレースフィールドインジケーションの場合、現在のピクチャが完全な画像の2番目、4番目、6番目等のラインを含む。インタレースフィールドインジケーションを送るとき、符号器は以下の規則に従わなければならない。

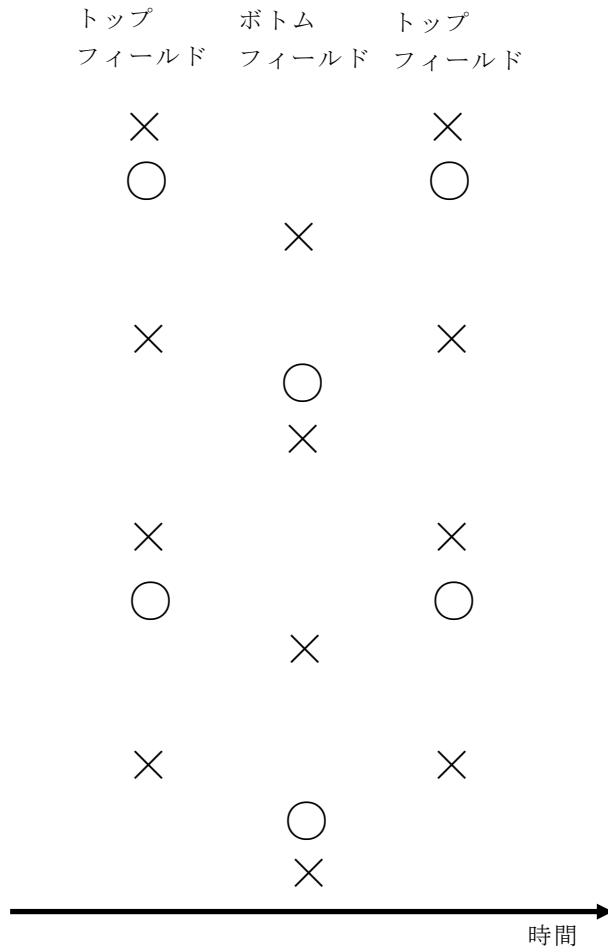
- 1) 符号器は、オリジナルの情報源ビデオの各々の新しいフィールドがテンポラルリファレンスの増加1に対応するような画像クロック周波数（もし必要なら、カスタム画像クロック周波数）を使用しなければならない。
- 2) 符号器は、ピクチャの寸法が1フィールドの寸法に対応するようなピクチャサイズ（もし必要なら、カスタムピクチャサイズ）を使用しなければならない。
- 3) 符号器は、完全な高さの画像アスペクト比が、現在符号化されたピクチャにより表現される1フィールドの画素アスペクト比から導かれる画像アスペクト比に対応するような画素アスペクト比（もし必要なら、カスタム画素アスペクト比）を使用しなければならない。

インタレースフィールド走査は、もともとはアナログビデオ圧縮技術として取り入れられた。プログレッシブ画像走査は一般的にデジタル圧縮、そして表示において優位であると考えられているが、多くのカメラとディスプレイ設計においてインタレースフィールド走査が引き続き使用されている。それゆえ、（インタレースフィールドレートの半分でのインタレースフルピクチャ符号化及びプログレッシブ走査画像符号化のどちらよりも低い遅延で実装可能である）インタレースフィールド符号化は、このインジケーションによりサポートされる。

復号器がこの様なフィールドベースのピクチャの受信と適切な処理する能力が外部手段（例えば、TTC標準JTH245）により確立されるまで、符号器はインタレースフィールドインジケーションを送ってはいけない。このような復号器能力の確立失敗は、復号器により受信そして表示される復号画像が、視覚的に目立つ小幅の垂直方向の振れを引き起こすかも知れない。

例えば、符号器は、2つ以上前のフィールドのアドレッシングを許す、（付属資料Nに指定される）参照ピクチャ選択モードまたは（付属資料Uに指定される）拡張参照ピクチャ選択モードを適用したインタレースフィールド符号化を使用してもよい。1ラインあたり704の符号化輝度標本と1フィールドあたり240の符号化ラインをもつ4:3画像アスペクト比の“525/60”インタレースフィールド符号化のために、符号器は、704のピクチャ幅と240のピクチャ高さをもつカスタムピクチャサイズ、5:11のカスタム画素アスペクト、及びクロック変換符号“1”とクロック除数30で規定されるカスタム画像クロック周波数を使用しなければならない。1ラインあたり704の符号化輝度標本と1フィールドあたり288の符号化ラインをもつ4:3画像アスペクト比の“625/50”インタレースフィールド符号化のために、符号器は、704のピクチャ幅と288のピクチャ高さをもつカスタムピクチャサイズ、6:11のカスタム画素アスペクト、及びクロック変換符号“0”とクロック除数36をもつカスタム画像クロック周波数を使用しなければならない。

トップフィールドピクチャのインタレース符号化における色差標本の垂直標本化位置は、これらの標本がフルピクチャ標本グリッドに対して垂直方向に通常的位置に配置するように、フィールド標本グリッドに対して1/4輝度標本高さだけ上方向にずれるように規定される。ボトムフィールドピクチャのインタレース符号化における色差標本の垂直標本化位置は、これらの標本がフルピクチャ標本グリッドに対して垂直方向に通常的位置に配置するように、フィールド標本グリッドに対して1/4輝度標本高さだけ下方向にずれるように規定される。色差標本の水平標本位置はインタレースフィールド符号化の適用により影響を受けないように規定される。垂直標本化位置は、それらに対応する時間的な標本位置とともに付図W. 2/JTH263に示される。



× = 輝度標本    ○ = 色差標本

付図W. 2 / JT-H263 インタレースフィールド符号化における色差標本の垂直、時間配置  
(ITU-T H. 263)

### W. 6. 3. 12 ピクチャ番号

もし付属資料Uを使用中なら、このメッセージは使用してはいけない。メッセージは、10ビットのピクチャ番号を伝える2データバイトを含む。結果として、DSIZEは3、CONTは0、そしてEBITは6でなければならない。ピクチャ番号は、10ビットモジュロ演算で、各々符号化され伝送されるI、Pピクチャ、PB、改良PBフレーム毎に1増加されなければならない。EIとEPピクチャにおいて、ピクチャ番号は、同じスケラビリティの高品質化レイヤ内の各々のEIまたはEPピクチャに対して増加されなければならない。Bピクチャにおいて、ピクチャ番号は、そのBピクチャの参照レイヤ内の、ビット列内の順序でBピクチャに先行する最も新しい非Bピクチャ（時間的にBピクチャに続くピクチャ）の値に対して増加されなければならない。同一の高品質化レイヤ内で隣接したピクチャが同じテンポラルリファレンスを持ち、なおかつ、参照ピクチャ選択モード（付属資料N）を使用している場合、復号器はこのことを、ほぼ同じ画像シーンの内容の冗長である複製が送られたことを示していると見なさなければならない。そして、これらのピクチャのすべては同じピクチャ番号を共有しなければならない。もし、同一の高品質化レイヤにおける2つの連続して受信した非Bピクチャのピクチャ番号の差が（モジュロ1024で）1でないなら、そして、もしそれらのピクチャが上述されたようなほぼ同じ画像シーン内容を表示していないのなら、復号器はピクチャの紛失、またはデータの誤りと推測すべきである。

### W. 6. 3. 13 予備参照ピクチャ

符号器は、もし実際の参照ピクチャが伝送中に失われたとき、どのピクチャが予備参照ピクチャとして使用可能なほど現在の動き補償参照ピクチャに似ているかを復号器に教えるためにこのメッセージを使用することができる。もし復号器が本来の参照ピクチャを欠き、予備参照ピクチャを利用できるなら、その復号器はINTRAピク



チャ更新を要求するべきではない。もし予備参照ピクチャがあるなら、その選択は符号器に任されている。メッセージデータバイトは、(もっとも好ましいものが最初に現れる) 好みの順で予備参照ピクチャのピクチャ番号を含む。ピクチャ番号は、付属資料UまたはW. 6. 3. 1 2 小節に従い、伝送される値を参照する。このメッセージはP、B、PB、改良PB、そしてEPピクチャタイプで使用できる。しかし、もし付属資料Nまたは付属資料Uが使用されており、もしそのピクチャが複数参照ピクチャに関係するなら、このメッセージを使用してはならない。EPピクチャに対しては、上方予測は常に時間的に対応する参照レイヤピクチャから行われるが、前方予測に対してのみメッセージを使用しなければならない。B、PB、そして改良PBピクチャタイプに対しては、前方動き予測参照としての使用のためのピクチャをメッセージは指定する。もしそのピクチャがIまたはEIピクチャであるなら、このメッセージを使用してはならない。

## 付属資料X

### プロファイルとレベルの定義

(この付属資料は本標準の必須部分である)

#### X. 1 本標準の規定範囲

本標準では様々なオプションのモードが利用できることから、オプション拡張された端末が“ベースライン”より高度なシンタックスを用いて高い確率で相互接続出来るように、いくつかの優先モードの組み合わせを定義することが重要である。本付属資料は、サポートする“プロファイル”を構成する、優先機能の組み合わせのリストを含む。これらのプロファイルがサポートする“レベル”として、最大の性能パラメータのグループ化も定義している。本付属資料の主な目的は、以下の通りである。

- 1) (プロファイルとレベルのパラメータを指定することによって) 復号器の能力を記述、または交渉する簡単な方法を提供すること
- 2) 最大限の相互接続性を実現するために復号器がサポートすべき共通の拡張機能を推奨すること
- 3) 特定のキーアプリケーションにおいて最適であるように選択された機能セットを示すこと

以下の節および付表X. 1 / JT-H 2 6 3、付表X. 2 / JT-H 2 6 3にプロファイルとレベルを定義する。付表X. 2 / JT-H 2 6 3に規定される最小ピクチャ間隔は、ビット列中で連続しているピクチャを復号する際の最小時間差である。レベル4 5以外のあるレベルをサポートすることは、それ以下の全てのレベルをサポートすることを意味する。

レベル4 5をサポートすることは、レベル1 0をサポートすることを意味する。

#### X. 2 優先モードサポートプロファイル

サポートプロファイルは、各々のプロファイルの復号器がサポートする機能セットによって定義される。与えられたあるプロファイルが復号器がサポートするという事は、そのプロファイルを構成するモードの有効な部分集合の組み合わせの全てをサポートすることを意味する。モードの組み合わせに関する符号器の選択の制限が最小になるように、この要求が存在する。これは本付属資料の主な目的に沿っており、特定の小さなモード組み合わせのセットを符号器に強いるのではなく、キーアプリケーションに向けて復号器においてどのオプションモードがサポートされるべきかについて述べている。

##### X. 2. 1 ベースラインプロファイル (プロファイル0)

本標準における最低限の“ベースライン”能力に対するプロファイル名称を規定するため、プロファイル0としてベースラインプロファイルをここで定義する。“ベースライン”は、どのオプション動作モードも使用しない本標準のシンタックスのことである。このサポートプロファイルは、ベースライン設計のみで構成される。

##### X. 2. 2 TTC標準JT-H 3 2 0符号化能力第2版後方互換プロファイル (プロファイル1)

TTC標準JT-H 3 2 0の回線交換端末システムが使うTTC標準JT-H 2 4 2の能力交換方式に採用された機能セットとの互換性を提供するために、プロファイル1としてTTC標準JT-H 3 2 0符号化能力第2版後方互換プロファイルをここで定義する。それは、(付属資料U、V、Wを含まない)本標準の第2版で有効な機能セットの中で、基本的な拡張符号化効率と簡単な拡張機能を規定する。このサポートプロファイルは、ベースライン設計に以下のモードを追加して構成される。

- 1) 拡張INTRA符号化 (付属資料I) – このモードを利用することにより (INTRAピクチャ内または予測符号化ピクチャ内に関わらず) INTRAマクロブロックに対する符号化効率を改善する。このモードでの付加的な計算量の要求は符号器と復号器の両方において最小 (符号化効率の顕著な向上を得るために、復号処理における8×8ブロックあたり最大8つの加算/減算に加え、可変長符号表と異なるがそれに非常に類似する可変長符号を使用する程度の小さな計算量) である。これらの理由から拡張INTRA符号化はこの基本的なサポートパッケージに含まれる。

- 2) デブロッキングフィルタ（付属資料J）－デブロッキングフィルタの利用により実現されるかも知れない顕著な主観的品質の改善のため、これらのフィルタはビデオ通信端末における後処理の方法として広く使われている。付属資料Jは、符号化ループ中にフィルタを置くので、デブロッキングフィルタ動作の優先モードを示す。フィルタをこの位置に置くことは、（要求されるメモリの低減により）フィルタの実装を容易にし、そして後処理の実装において符号化の性能をいくらか改善させる。このデブロッキングフィルタモードは拡張予測モードと共に用いられる場合、共に更なる符号化効率の改善が可能であるマクロブロックあたり4本の動きベクトル機能および動き補償でのピクチャ境界外挿補間も含む。デブロッキングフィルタの計算量の要求は符号化マクロブロックあたり数百演算だが、メモリアクセス、および計算の依存性は複雑でない。この最後の点が、ある実装においてはデブロッキングフィルタを拡張予測に対して優位にしている。また、デブロッキングフィルタが拡張予測と一緒に使われる時は、拡張予測の恩恵はそれほど大きくない。従って、デブロッキングフィルタはこの基本的なサポートパッケージに含まれる。
- 3) 全画面凍結 付加拡張情報（付属資料L L. 4節）－全画面凍結は実装するのが非常に簡単であり、復号器が出力バッファからビデオディスプレイへの新しいピクチャの転送を止めることが出来ることが必要であるだけである。符号器が忠実度のより高いピクチャを生成する間、低い忠実度のピクチャを表示することを防止するため、この能力は有効である。
- 4) 修正量子化（付属資料T）－このモードは、拡張DCT係数範囲、修正DQUANTシンタックス、そして色差に対する修正ステップサイズを含む。はじめの2つの機能は符号器に更なる自由度を与え、そして（係数レベルの飽和が発生したときに必要なマクロブロックの再符号化の削除により）実質的に符号器の計算負荷を軽減するかもしれない。3番目の機能は、特に殆どビットレートコストを増加させることなく、また実質的に計算量を増やすことなく、色差の忠実度を著しく改善する。復号器における唯一の顕著な追加の計算負荷は、いくつかの新しいビット列シンボルを解釈する能力である。

### X. 2. 3 第1版後方互換プロファイル（プロファイル2）

（付加拡張情報あるいはPLUSPTYPEを利用するどのオプション機能も含まない）TTC標準JT-H263の第1版で有効な機能セットの範囲の中で、拡張された符号化効率性能を提供するために、プロファイル2として第1版後方互換プロファイルをここで定義する。このサポートプロファイルは、ベースライン設計に以下の1つのモードを追加して構成される。

- 1) 拡張予測（付属資料F）－符号化効率の観点から、このモードは本標準の最初の版（第1版）において有効な最も重要なモードである。このモードはオーバーラップブロック動き補償、マクロブロックあたり4本の動きベクトル機能を含み、動きベクトルがピクチャ境界の外側を指すことを許容する。拡張予測を使うことで主観的および客観的の両面において著しい改善を得ることができる。しかし、これはかなりの計算量の増加を伴い、そして復号器での処理の順序において複雑なデータ依存性を招く。しかし、このリスト中の他モードの採用に先だって設計された本標準の実装においては、単独で拡張予測を実装しているかもしれないので、第1版の復号器との最大限の後方互換のために、拡張予測のみの処理が推奨される。

### X. 2. 4 第2版インタラクティブ・ストリーミング無線プロファイル（プロファイル3）

（付属資料U、V、Wは含まない）本標準の第2版で有効な機能セットの中で、拡張された符号化効率性能と無線デバイスへの配信のために拡張された誤り耐性を提供するために、プロファイル3として第2版インタラクティブ・ストリーミング無線プロファイルをここで定義する。このプロファイルは、ベースライン設計に以下のモードを追加して構成される。

- 1) 拡張INTRA符号化（付属資料I）－X. 2. 2小節の1)項を参照。
- 2) デブロッキングフィルタ（付属資料J）－X. 2. 2小節の2)項を参照。
- 3) スライス構造モード（付属資料K）－データ誤りやデータ損失を回復させるビデオビット列内の再同期点を提供する能力を拡張するため、スライス構造モードがここに含まれる。復号器に対する複雑さに関する要求条件を制限するため、スライス構造モードの任意スライス順序サブモード（ASO）と矩形スライスサブモード（RS）のサポートをこのプロファイルは含まない。スライス構造モードによる追加の計算の負荷は最小で、主にビット列の生成と解釈に限られる。
- 4) 修正量子化（付属資料T）－X. 2. 2小節の4)項を参照。

### X. 2. 5 第3版インタラクティブ・ストリーミング無線プロファイル（プロファイル4）

本標準の第3版で拡張された機能を利用すると同時に、拡張された符号化効率性能と無線デバイスへの配信のために拡張された誤り耐性を提供するために、プロファイル4として第3版インタラクティブ・ストリーミング無線プロファイルをここで定義する。このサポートプロファイルはベースライン設計に以下の追加機能を追加して構成される。

- 1) プロファイル3－この機能セットは無線ビデオ伝送のサポートに有効ないくつかの拡張を提供する。
- 2) データパーティションスライスモード（付属資料V）－この機能は、スライス内のDCT係数データから動きベクトルデータを分離することにより誤り耐性能力を拡張し、そしてリバーシブル可変長符号化を使って（マクロブロックでの詳細なデータの中で最も重要な部分である）動きベクトル情報を保護する。復号器に対する複雑さに関する要求条件を制限するため、任意スライス順序サブモード（ASO）と矩形スライスサブモード（RS）のサポートをこのプロファイルは含まない。
- 3) 前ピクチャヘッダ繰り返し付加拡張情報（付属資料W W. 6. 3. 8小節）－この機能は、データ損失やデータ誤りが発生した場合に、復号器が前のピクチャからヘッダ情報を受け取り、修復することを許容する。

### X. 2. 6 会話型高圧縮プロファイル（プロファイル5）

Bピクチャの使用に関連する遅延を追加することなく、また誤り耐性機能を追加せず、拡張された符号化効率性能を提供するために、プロファイル5として会話型高圧縮プロファイルをここで定義する。このサポートプロファイルはベースライン設計に以下の追加機能を追加して構成される。

- 1) プロファイル1－この機能セットは拡張符号化効率のために有効ないくつかの拡張を提供する。
- 2) プロファイル2－このプロファイルは拡張予測モード（付属資料F）を追加する。これは符号化効率性能の更なる拡張と、本標準の第1版の実装との後方互換を提供する。
- 3) UUI = “1” の場合の無制限動きベクトル（付属資料D）－付属資料Dは2つの主要な機能を有している。それらは、
  - a) ピクチャ境界外挿補間法
  - b) 拡張動きベクトルのサポート

である。これらの機能の最初の機能に関しては、プロファイル1において付属資料Jが含まれることにより既にサポートされている。拡張動きベクトルのサポートは、大きなピクチャサイズ、速い動き、カメラの動き、そして低ピクチャレートに対して特に符号化効率の著しい改善をもたらす。PLUSPTYPEと共に使われる時、符号化演算を著しく簡単にすることが出来る拡張動きベクトル差分を使用することも、このモードは許容する。拡張動きベクトルは復号器にとってメモリアクセスの意味で問題を与える可能性があるが、最大動きベクトルサイズをピクチャサイズに依存する大きさに制限することにより、これが実装での明らかな障害になることを防止する。

- 4) 拡張参照ピクチャ選択（付属資料U）－後続するピクチャのマクロブロックレベルの予測において複数の先行するピクチャを参照データとして使うことができるため、このモードは圧縮効率性能を著しく向上させる。拡張参照ピクチャ選択モードにおけるサブピクチャ除去サブモード（付属資料U U. 4. 3小節）は、プロファイル5には含まれない。

### X. 2. 7 会話型インターネットプロファイル（プロファイル6）

Bピクチャの使用に関連する遅延を追加することなく、しかし（比較的大きなパケットを用いるパケットベースのデータプロトコルを使用し、データ間違いよりもむしろデータ損失が発生する）インターネットプロトコル（IP）ネットワークでの使用に適した誤り耐性を追加して、拡張された符号化効率性能を提供するために、プロファイル6として会話型インターネットプロファイルをここで定義する。このプロファイルはベースライン設計に以下の追加機能を追加して構成される。

- 1) プロファイル5－この機能セットは拡張符号化効率に有用ないくつかの拡張を提供する。

- 2) 任意スライス順序 (ASO) サブモードを使用するスライス構造モード (付属資料K) - データパケット損失を回復させるためのビデオビット列内の再同期ポイントを提供する能力を拡張するために、スライス構造モードがここに含まれる。動き補償されたエラーコンシールメントのためのインターリーブパケット化及び順序が狂ったデータの受信を許容するために、スライス構造モードの任意スライス順序 (ASO) サブモードも含まれる。復号器に対する複雑さに関する要求条件を制限するために、このプロファイルにはスライス構造モードの矩形スライス (RS) サブモードのサポートは含まれない。スライス構造モードによる追加の計算の負荷は最小で、主にビット列の生成と解釈に限られる。

#### X. 2. 8 会話型インタレースプロファイル (プロファイル7)

低遅延アプリケーションに対する拡張された符号化効率性能に加えて、インタレースビデオ情報源のサポートを提供するために、プロファイル7として会話型インタレースプロファイルをここで定義する。このプロファイルはベースライン設計に以下の追加機能を追加して構成される。

- 1) プロファイル5 - この機能セットは、遅延を追加することなく符号化効率の向上に有用ないくつかの拡張を提供する。
- 2) 240ラインおよび288ラインのピクチャのためのインタレースフィールドインジェクション (付属資料W W. 6. 3. 11 小節) - この機能は、存在するカメラ設計との互換性のためにインタレース情報源画像フォーマットでビデオが送られることを許容する。

#### X. 2. 9 高遅延プロファイル (プロファイル8)

クリティカルな遅延制約のないアプリケーションに対する拡張された符号化効率性能を提供するために、プロファイル8として高遅延プロファイルをここで定義する。このプロファイルはベースライン設計に以下の追加機能を追加して構成される。

- 1) プロファイル6 - この機能セットは拡張された符号化効率とデータ損失に対する耐性に有用ないくつかの拡張を提供する。
- 2) 参照ピクチャ再サンプリング (暗黙の“ファクター4”モードのみ) (付属資料P P. 5 節) - 参照ピクチャ再サンプリングの暗黙の“ファクター4”モードは、ピクチャヘッダで示されるような、新しいフレームのサイズが変更される場合のみ、自動的な参照ピクチャ再サンプリングを許容する。このモードの処理に必要なビット列のオーバーヘッドはない。予測的な動的解像度変更を使うことにより、符号器は時間的解像度と空間的解像度との間で高度なトレードオフが可能である。さらに、付属資料Pのこの最も簡単なモードの処理 (“ファクター4”アップサンプリングまたはダウンサンプリングのみ) は、符号器と復号器の両方にさほど多くない計算の複雑さを加えるだけである。なぜなら、“ファクター4”は、(せいぜい画素あたりおよそ4処理程度が必要な) 簡単な固定FIRフィルタを使うからである。
- 3) Bピクチャ (時間的スケラビリティ、付属資料O O. 1. 1 小節) - この機能は双方向の時間的予測が許されているBピクチャから構成される。Bピクチャを加えることは符号化効率性能を向上させるが、その代わりに処理負荷及び符号化と復号の遅延を増大させる。拡張参照ピクチャ選択モード (付属資料U U. 3. 1. 5. 5 小節) のBピクチャに対する2つのピクチャの後方予測サブモードは、プロファイル8ではサポートされない。

#### X. 3 画像フォーマットと画像クロック周波数

高い品質レベルの相互接続性を確保するために、ある大きな標準画像フォーマット (QCIF、CIF、4CIF、16CIF) をサポートする符号器と復号器はより小さい全ての標準画像フォーマットをサポートするべきである。これは、本付属資料に定義されたプロファイルとレベルに従う全ての復号器に対する要求条件である。(本標準の他の箇所規定されているように、復号器はサブQCIFかつQCIFをサポートしなければならない。符号器はサブQCIFまたはQCIFをサポートしなければならない。) 例えば、この付属資料に定義されているプロファイルとレベルに従う4CIF画像の復号が可能な復号器は、CIF画像の復号もサポートしなければならない。

復号器は、より大きな標準画像フォーマットにおいて処理できる最大ピクチャレート以上の最大ピクチャレートでより小さな標準画像フォーマットを処理することが可能であるべきである。これは、本付属資料に定義されたプ

ロファイルとレベルに従う全ての復号器に対する要求条件である。例えば、この付属資料に定義されているプロファイルとレベルに従う1秒あたり25ピクチャで4CIF画像を復号できる復号器は、CIF、QCIF、サブQCIF画像も少なくとも1秒あたり25ピクチャで復号出来なければならない。

カスタム画像フォーマットおよびカスタム画像クロック周波数の両方もしくは一方をサポートする符号器と復号器は、本節に定義される以下の規則に従うことが推奨される。この規則は本付属資料に定義されたプロファイルとレベルに従う全ての復号器に対する要求条件である。

- 1) ある最大画像フォーマットをサポートするここに定義されたどのプロファイル及びレベルに対応する復号器も、サポートする最大画像フォーマットと高さおよび幅の両方において同じかまたはより小さい全ての標準画像フォーマットをサポートしなければならない。例えば、720×288のカスタム画像フォーマットをサポートする復号器は、CIF、QCIF、サブQCIF画像の復号もサポートしなければならない。
- 2) カスタム画像フォーマットをサポートするここに定義されたどのプロファイル及びレベルに対応する復号器も、サポートする最大画像フォーマットと高さおよび幅の両方において同じかまたはより小さい全ての標準もしくはカスタム画像フォーマットをサポートしなければならない。
- 3) 1秒あたり(30000)/1001の標準画像クロック周波数で表現された最小ピクチャ間隔をサポートするここに定義されたどのプロファイル及びレベルに対応する復号器も、最小ピクチャ間隔が規定された最大画像フォーマットと高さおよび幅の両方において同じかまたはより小さいサポートする全ての画像フォーマットに対して、同じかまたは小さい最小ピクチャ間隔をサポートしなければならない。
- 4) 最小ピクチャ間隔とカスタム画像クロック周波数をサポートするここに定義されたどのプロファイル及びレベルに対応する復号器も、最小ピクチャ間隔が規定された最大画像フォーマットと高さおよび幅の両方において同じかまたはより小さいサポートする全ての画像フォーマットに対して、同じかまたはより大きいピクチャ間隔となる画像クロック周波数の使用をサポートしなければならない。

#### X. 4 動作能力のレベル

8レベルの動作能力を復号器の実装に対して定義する。仮想標準復号器は、プロファイル0から4の全てのレベルに対して付表X. 1/JT-H263に規定された最小サイズをもつ。プロファイル5から8では、仮想標準復号器はサイズが増加し、複数の参照ピクチャをもつ拡張参照ピクチャ選択をサポートしている。付表X. 2/JT-H263は、以下のレベルの動作パラメータの詳細を定義する。

- 1) レベル10-QCIF、サブQCIFの解像度の復号をサポートし、1秒あたり(15000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり64000bitまでのビットレートでの動作が可能である。
- 2) レベル20-CIF、QCIF、サブQCIFの解像度の復号をサポートし、CIF画像に対しては1秒あたり(15000)/1001ピクチャまでのピクチャ復号レート、QCIF画像とサブQCIF画像に対しては1秒あたり(30000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $2 \cdot (64000) = 128000$ bitまでのビットレートでの動作が可能である。
- 3) レベル30-CIF、QCIF、サブQCIFの解像度の復号をサポートし、1秒あたり(30000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $6 \cdot (64000) = 384000$ bitまでのビットレートでの動作が可能である。
- 4) レベル40-CIF、QCIF、サブQCIFの解像度の復号をサポートし、1秒あたり(30000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $32 \cdot (64000) = 2048000$ bitまでのビットレートでの動作が可能である。
4. 5) レベル45-QCIF、サブQCIFの解像度の復号をサポートし、1秒あたり(15000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $2 \cdot (64000) = 128000$ bitまでのビットレートでの動作が可能である。さらに、プロファイル0と2以外は、QCIF以下のカスタム画像フォーマットをサポートする。
- 5) レベル50-CIF以下のカスタムおよび標準画像フォーマットをサポートし、CIF以下の画像フォーマットに対しては1秒あたり50ピクチャまでのピクチャ復号レート、352×240以下の画像フォーマットに対しては1秒あたり(60000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $64 \cdot (64000) = 4096000$ bitまでのビットレートでの動作が可能である。
- 6) レベル60-720×288以下のカスタムおよび標準画像フォーマットをサポートし、720×288以下の画像フォーマットに対しては1秒あたり50ピクチャまでのピクチャ復号レート、720×240

以下の画像フォーマットに対しては1秒あたり(60000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $128 \cdot (64000) = 8192000$ bitまでのビットレートでの動作が可能である。

- 7) レベル70 – 720 × 576以下のカスタムおよび標準画像フォーマットをサポートし、720 × 576以下の画像フォーマットに対しては1秒あたり50ピクチャまでのピクチャ復号レート、720 × 480以下の画像フォーマットに対しては1秒あたり(60000)/1001ピクチャまでのピクチャ復号レートで、1秒あたり $256 \cdot (64000) = 16384000$ bitまでのビットレートでの動作が可能である。

システムで使用される特定のプロファイルおよびレベルにおけるビットレートは、本付属資料で規定した値を決して超えてはならない。しかし、特定のシステムは、ビットレートに関してさらなる制限を送るための他の手段を含んでいてもよい。特定のシステムで使用される場合、プロファイルとレベルの能力に関する他の側面は、付加的な能力の制限に従わなければならないかもしれないが、ここに定義された特定のプロファイルとレベルに対するビット列の復号に要求される能力は、本付属資料に規定された値を決して超えてはならない。

付表 X. 1 / JT-H263 プロファイルのまとめ  
(ITU-T H. 263)

右に示すプロファイルに対する付属資料もしくは節	0	1	2	3	4	5	6	7	8
5.1.5: カスタム画像フォーマット (CPFMT)	L	L	L	L	L	L	L	L	L
5.1.7: カスタム画像クロック周波数符号 (CPCFC)	L	L	L	L	L	L	L	L	L
C: コンティニューアスプレゼンスマルチポイントとビデオ多重化									
D.1: ピクチャの境界を越えた動きベクトル		X	X	X	X	X	X	X	X
D.2 (UUI='1' または UUIなし): 動きベクトルの範囲の拡大						X	X	X	X
D.2 (UUI='01'): 動きベクトルの範囲の無制限拡大									
E: シンタックス算術符号化									
F.2: マクロブロックあたり 4 本の動きベクトル		X	X	X	X	X	X	X	X
F.3: オーバーラップ動き補償			X			X	X	X	X
G: PBフレーム									
H: 前方誤り訂正 (TTC標準JT-H.320のようにシステムレベルに組み込んでよい)									
I: 拡張INTRA符号化		X		X	X	X	X	X	X
J: デブロッキングフィルタ		X		X	X	X	X	X	X
K (サブモードなし): スライス構造符号化 (サブモードなし)				X	X		X		X
K (ASOあり): スライス構造符号化 (任意スライス順序サブモードあり)							X		X
K (RSあり): スライス構造符号化 (矩形スライスサブモードあり)									
L.4: 付加拡張 (全画面凍結)		X				X	X	X	X
L: 付加拡張 (他の付加拡張情報の機能)									
M: 改良PBフレーム									
N: 参照ピクチャ選択 (サブモードを含む)									
O.1.1 (時間 (Bピクチャ)): 時間、SNR、空間スケーラビリティ (時間スケーラビリティに対するBピクチャ)									X
O (SNRと空間): 時間、SNR、空間スケーラビリティ (SNRと空間スケーラビリティに対するEIとEPピクチャ)									
P.5: 参照ピクチャ再サンプリング (暗黙の“ファクター4”)									X
P: 参照ピクチャ再サンプリング (より一般的な再サンプリング)									
Q: 縮小解像度更新									
R: 独立セグメント復号									
S: 代替INTER可変長符号									
T: 修正量子化		X		X	X	X	X	X	X
U (サブモードなし): 拡張参照ピクチャ選択 (サブモードなし)						X	X	X	X
U (サブピクチャ除去サブモードあり): 拡張参照ピクチャ選択 (サブピクチャ除去サブモードあり)									
U (BTPSMあり): 拡張参照ピクチャ選択 (Bピクチャ2ピクチャサブモードあり)									
V: データパーティションスライス					X				
W.6.3.8: 追加付加拡張情報仕様 (前ピクチャヘッダ繰り返し)					X				
W.6.3.11: 追加付加拡張情報仕様 (インタレースフィールドインジケーション)								X	
W: 追加付加拡張情報仕様 (他の追加付加拡張情報の機能)									

(注)

- 1) “X”は、機能のサポートがプロファイルの一部であることを示す。
- 2) “L”は、含まれる機能がプロファイル内のレベルに依存することを示す。



付表X. 2 / J T - H 2 6 3 動作レベル  
( I T U - T H . 2 6 3 )

右に示すレベルに対するパラメータ	10	20	30	40	45	50	60	70
最大画像フォーマット	QCIF (176×144)	CIF (352×288)	CIF (352×288)	CIF (352×288)	QCIF (176×144) プロファイル0と2以外は CPFMTサポート	CIF (352×288) CPFMTサポート	CPFMT: 720×288 CPFMTサポート	CPFMT: 720×576 CPFMTサポート
最小ピクチャ間隔	2002/ (30000) 秒	CIF 2002/ (30000) 秒  QCIF, sub-QCIF 1001/ (30000) 秒	1001/ (30000) 秒	1001/ (30000) 秒	2002/ (30000) 秒  プロファイル0と2以外はCPCFCサポート	CIF 以下で 1/50 秒  352×240 以下で 1001/ (60000) 秒 CPCFCサポート	720×288 以下で 1/50 秒  720×240 以下で 1001/ (60000) 秒 CPCFCサポート	720×576 以下で 1/50 秒  720×480 以下で 1001/ (60000) 秒 CPCFCサポート
最大ビットレート (64 000 bit/s単位)	1	2	6	32	2	64	128	256
最大 HRD B (16 384 bit単位)	1: プロファイル5-8	2: プロファイル5-8	6: プロファイル5-8	32: プロファイル5-8	2: プロファイル5-8	64: プロファイル5-8	64: プロファイル5-8	256: プロファイル5-8
最大 BPPmaxKb (1 024 bit 単位)	128: プロファイル5-8	512: プロファイル5-8	512: プロファイル5-8	512: プロファイル5-8	128: プロファイル5-8	512: プロファイル5-8	1024: プロファイル5-8	1024: プロファイル5-8
最大ERPS参照ピクチャ (付属資料U)	5: プロファイル5-7 10: プロファイル8	5: プロファイル5-7 10: プロファイル8  プロファイル5-8に対してQCIFまたはサブQCIFは2倍する。	5: プロファイル5-7 10: プロファイル8  プロファイル5-8に対してQCIFまたはサブQCIFは2倍する。	5: プロファイル5-7 10: プロファイル8  プロファイル5-8に対してQCIFまたはサブQCIFは2倍する。	5: プロファイル5-7 10: プロファイル8	5: プロファイル5-7 10: プロファイル8  プロファイル5-8に対してQCIFまたはそれ以下は2倍する。	5: プロファイル5-7 10: プロファイル8  プロファイル5-8に対してCIFまたはそれ以下は2倍し、QCIFまたはそれ以下は4倍する。	5: プロファイル5-7 10: プロファイル8  プロファイル5-8に対してCIFまたはそれ以下は2倍し、QCIFまたはそれ以下は4倍する。

(注)

- 1) 付表X. 2 / J T - H 2 6 3 で参照ピクチャバッファの最大数が規定されていないプロファイルに対しては、複数の参照ピクチャバッファのサポートは要求されない。
- 2) 付表X. 2 / J T - H 2 6 3 で最大B P P m a x K bおよびHRD Bが規定されていないプロファイルに対しては、付表X. 1 / J T - H 2 6 3 に規定された最小値が、最大ビットレートおよび解像度に適用される。

X. 5 TTC標準JT-H245との組み合わせに関するGeneric capability定義

付表X. 3/JT-H263は、能力決定にTTC標準JT-H245を用いるシステムでの使用のためにTTC標準JT-H263能力の確立に関する能力識別子を規定する。TTC標準JT-H245のVideoCapability構造におけるgenericVideoCapabilityとして、またVideoMode構造におけるgenericVideoModeとしてのみ、これらのパラメータが含まれなければならない。付表X. 4/JT-H263から付表X. 14/JT-H263は、関連する能力パラメータを規定する。

論理チャンネルシグナリングやモード要求に含まれる場合は、0から8の範囲のパラメータ識別子の値を持つただ1つのパラメータだけが存在しなければならない。すなわち、ただ1つのプロファイルだけが規定されなければならない。

付表X. 3/JT-H263 JT-H263能力の能力識別子  
(ITU-T H. 263)

能力名:	H.263
能力クラス:	Video codec
能力識別子タイプ:	標準
能力識別子値:	ITU-T (0) 標準 (0) h (8) 263 generic-capabilities (1) 0
MaxBitRate:	maxBitRateフィールドは常に含まれなければならない。
NonCollapsingRaw:	このフィールドは含まれてはいけない。
Transport:	このフィールドは含まれてはいけない。

付表X. 4/JT-H263 ベースラインプロファイル (プロファイル0) 能力  
(ITU-T H. 263)

パラメータ名:	baselineProfile
パラメータ記述:	これはcollapsing GenericParameterである。 baselineProfileは、能力交換において存在する場合は、ベースラインプロファイルに対するサポートの最大レベルを示し、論理チャンネルシグナリングに存在する場合は 伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	0
パラメータ状態:	必須
パラメータタイプ:	unsignedMin.
Supersedes:	-

付表X. 5/JT-H263 JT-H320符号化能力第2版後方互換プロファイル  
(ITU-T H. 263) (プロファイル1) 能力

パラメータ名:	h320Profile
パラメータ記述:	これはcollapsing GenericParameterである。 h320Profileは、能力交換において存在する場合はTTC標準JT-H320符号化能力第2版後方互換プロファイルに対するサポートの最大レベルを示し、論理チャンネルシグナリングに存在する場合は 伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	1
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin.
Supersedes:	-

付表 X. 6 / JT-H 263 第1版後方互換プロファイル (プロファイル2) 能力  
(ITU-T H. 263)

パラメータ名:	backwardCompatibleProfile
パラメータ記述:	これはcollapsing GenericParameterである。 backwardCompatibleProfile は、能力交換において存在する場合は第1版後方互換プロファイルに対するサポートの最大レベルを示し、論理チャンネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	2
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表 X. 7 / JT-H 263 第2版インタラクティブ・ストリーミング無線プロファイル  
(ITU-T H. 263) (プロファイル3) 能力

パラメータ名:	v2WirelessProfile
パラメータ記述:	これはcollapsing GenericParameterである。 v2WirelessProfile は、能力交換において存在する場合は第2版インタラクティブ・ストリーミング無線プロファイルに対するサポートの最大レベルを示し、論理チャンネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	3
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表 X. 8 / JT-H 263 第3版インタラクティブ・ストリーミング無線プロファイル  
(ITU-T H. 263) (プロファイル4) 能力

パラメータ名:	v3WirelessProfile
パラメータ記述:	これはcollapsing GenericParameterである。 v3WirelessProfile は、能力交換において存在する場合は第3版インタラクティブ・ストリーミング無線プロファイルに対するサポートの最大レベルを示し、論理チャンネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	4
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表 X. 9 / JT-H 263 会話型高圧縮プロファイル (プロファイル5) 能力  
(ITU-T H. 263)

パラメータ名:	conversationalProfile
パラメータ記述:	これはcollapsing GenericParameterである。 conversationalProfile は、能力交換において存在する場合は会話型高圧縮プロファイルに対するサポートの最大レベルを示し、論理チャンネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	5
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表X. 10 / JT-H263 会話型インターネットプロファイル (プロファイル6) 能力  
(ITU-T H. 263)

パラメータ名:	conversationalInternetProfile
パラメータ記述:	これはcollapsing GenericParameterである。 conversationalInternetProfile は、能力交換において存在する場合は会話型インターネットプロファイルに対するサポートの最大レベルを示し、論理チャネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	6
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表X. 11 / JT-H263 会話型インタレースプロファイル (プロファイル7) 能力  
(ITU-T H. 263)

パラメータ名:	conversationalInterlaceProfile
パラメータ記述:	これはcollapsing GenericParameterである。 conversationalInterlaceProfile は、能力交換において存在する場合は会話型インタレースプロファイルに対するサポートの最大レベルを示し、論理チャネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	7
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表X. 12 / JT-H263 高遅延プロファイル (プロファイル8) 能力  
(ITU-T H. 263)

パラメータ名:	highLatencyProfile
パラメータ記述:	これはcollapsing GenericParameterである。 highLatencyProfile は、能力交換において存在する場合は高遅延プロファイルに対するサポートの最大レベルを示し、論理チャネルシグナリングに存在する場合は伝送されるべき最大レベルを示し、モード要求に存在する場合は要求レベルを示す。
パラメータ識別子値:	8
パラメータ状態:	オプション
パラメータタイプ:	unsignedMin
Supersedes:	-

付表X. 13 / JT-H263 時間的および空間的トレードオフ能力  
(ITU-T H. 263)

パラメータ名:	temporalSpatialTradeOffCapability
パラメータ記述:	これはcollapsing GenericParameterである。 このパラメータの存在は、相手端末からの指示により時間的および空間的な解像度に対するトレードオフを符号器側が変更することができることを示す。受信能力の一部になる場合は意味を持たない。
パラメータ識別子値:	9
パラメータ状態:	オプション
パラメータタイプ:	logical
Supersedes:	-

付表X. 14 / JT-H263 Video Bad Macroblocks 能力  
(ITU-T H. 263)

パラメータ名:	videoBadMBsCap
パラメータ記述:	これはcollapsing GenericParameterである。 このパラメータの存在は、videoBadMBsコマンドを符号器が受信できること、または復号器が送信できることを示す。送信能力の一部となる場合は、符号器がvideoBadMBsコマンドを処理することが可能で、画像品質の回復に対する適切な調整処置が行えることを示す。受信能力の一部となる場合は、適切なvideoBadMBsインジケーションを送信することができる復号器の能力を示す。
パラメータ識別子値:	10
パラメータ状態:	オプション
パラメータタイプ:	logical
Supersedes:	-

## 付録 I

### 誤り追跡

(この付録は本標準の必須部分ではない)

#### I. 1 序論

本付録は、誤りの起ったマクロブロックがフィードバックチャネルにより符号器に報告される場合、伝送誤りの後に効果的に回復する方法について記述する。フィードバック情報を送り処理する能力は外部の手段（例えば、TTC標準 JT-H 2 4 5）により通知される。さらに、フィードバックメッセージのフォーマットと内容は外部で（例えば、TTC標準 JT-H 2 4 5で）定義される。

#### I. 2 誤り追跡

INTRA符号化は時間的な誤り伝播を止めるので、伝送誤りにより重大な影響を受けたマクロブロックに対してこれを用いるべきである。この利用のためには、画像の歪みの位置と程度の情報が符号器に利用できるようにする必要がある。以下のアルゴリズムにより、符号器により受信されたフィードバック情報に基づいた推定誤り分布が得られる。これは、動き補償による予測により引き起こされた空間的誤り伝播と、フィードバックメッセージを受信するまでの遅延を考慮している。このアルゴリズムは、空間・時間的誤り追跡のためのフィードバックメッセージを評価する一つの可能なアプローチを説明している。よって、他のアルゴリズムも可能である。

各フレームに、左上から右下へと  $mb = 1$  から  $N$  と番号付けされた  $N$  個のマクロブロックがあると仮定する。 $\{n_{err}, mb_{first}, mb_{last}\}$  を符号器へのフィードバックメッセージとする。ここで、 $mb_{first} \leq mb \leq mb_{last}$  はフレーム  $n_{err}$  における誤りのあるマクロブロックのセットを示す。

フィードバックメッセージを評価するために、符号器は各フレームを符号化する間、継続的に情報を記録しなければならない。まず、フレーム  $n$  でのマクロブロック  $mb$  の喪失による初期誤り  $E_0(mb, n)$  を蓄積する必要がある。誤りのあるマクロブロックを符号化されないマクロブロックとして取り扱う単純な誤りコンシールメントを仮定すると、 $E_0(mb, n)$  はフレーム  $n$  と  $n-1$  でのマクロブロック  $mb$  の絶対誤差の和 (SAD) として計算される。次に、フレーム  $n-1$  のマクロブロック  $mb_{source}$  からフレーム  $n$  のマクロブロック  $mb_{dest}$  に伝送された画素数を、依存度  $d(mb_{source}, mb_{dest}, n)$  に蓄える。これらの依存度は動きベクトルから導出される。

フレーム  $n_{next}$  ( $n_{next} > n_{err}$ ) が符号化される前にフィードバックメッセージが到着するものと仮定すると、フレーム  $n_{err}$  のマクロブロック  $mb$  での推定誤り  $E(mb, n_{err})$  は次のように初期化される。

$$E(mb, n_{err}) = \begin{cases} E_0(mb, n_{err}) & mb_{first} \leq mb \leq mb_{last} \text{ の場合} \\ 0 & \text{その他の場合} \end{cases}$$

$n_{err} < n < n_{next}$  である後続フレーム  $n$  に対して、誤りは次のように推定してもよい。ここで、各反復計算後、各マクロブロックにおける誤りは均一に分布すると仮定している。

$$E(mb, n) = \sum_{i=1}^N E(i, n-1) d(i, mb, n) / 256$$

推定誤り  $E(mb, n_{next}-1)$  は次のフレームのモード決定に利用される。例えば、もし  $E(mb, n_{next}-1)$  があるしきい値を超える場合、マクロブロック  $mb$  は INTRA モードで符号化される。

実際には、最新の  $M$  フレーム分の誤り追跡情報しか蓄えられないであろう。よって、もし  $n_{err} < n_{next} - M$  ならば、誤り追跡情報は利用できず、符号器は特別な動作を取らなければならない。例えば、次のフレームは INTRA モードで符号化されるかもしれない。しかし、他の処理が可能であり、それがより効果的であるかもしれない。

## 付録Ⅱ

### 推奨されるオプション拡張

(この付録は本標準の必須部分ではない)

本標準第2版に存在した付録Ⅱ（対応するITU-T勧告において1998年2月に承認されたITU-T H.263 Appendix II Recommended optional enhancement）の内容は、本標準付属資料Xの承認後、廃止となった。本付録は多くのTTC標準/ITU-T文書利用者から参照に用いられているため、本付録は主に付属資料Xの内容に気づいていない利用者のための参照として提供される。