

# JT-G711.1

## G. 711パルス符号変調に対する 広帯域エンベデッド拡張

Wideband embedded extension for G.711 pulse code  
modulation

第2版

2011年5月31日制定

一般社団法人  
情報通信技術委員会

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

本書は、一般社団法人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一般社団法人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> .....	5
1. 本標準の規定範囲 .....	7
2. 参照文献 .....	7
2.1 必須文献 .....	7
2.2 参考文献 .....	7
3. 定義 .....	7
4. 略語 .....	7
5. 慣例 .....	8
6. 符復号器の概要記述 .....	13
6.1 符号器 .....	13
6.2 復号器 .....	14
6.3 符号化モード .....	15
6.4 ビット割り当て .....	15
6.5 アルゴリズム遅延 .....	16
6.6 演算量と所要メモリ量 .....	16
6.7 符復号器の記述 .....	17
6.8 異なるモード間や既存のG. 711とのトランスコード .....	17
7. 符号化器の機能記述 .....	18
7.1 前処理の高域通過フィルタ .....	18
7.2 分析QMF .....	18
7.3 低域符号化 .....	20
7.3.1 G. 711に基づくコアPCM符号化器(レイヤ0) .....	21
7.3.2 聴覚フィルタリング .....	23
7.3.3 デッドゾーン量子化器 .....	27
7.3.4 精細信号の適応多重化(レイヤ1) .....	29
7.4 MDCT .....	30
7.5 高域符号化(レイヤ2) .....	32
7.5.1 周波数重み付け .....	33
7.5.2 RMS演算とターゲット正規化 .....	34
7.5.3 インターリーブ共役構造ベクトル量子化 .....	34
7.5.4 ゲインスカラ量子化部 .....	38
8. 復号器の機能記述 .....	39
8.1 エンベデッド低域PCM復号器 .....	39
8.1.1 G. 711(レイヤ0)に基づくコアPCM復号器 .....	40
8.1.2 低域拡張復号器(レイヤ1) .....	41
8.2 高域復号器(レイヤ2) .....	42
8.2.1 インターリーブベクトル逆量子化部 .....	42
8.2.2 利得スカラ逆量子化 .....	42
8.2.3 逆重み付け .....	43
8.3 逆MDCTとオーバーラップ加算 .....	43
8.4 低域FERC .....	44
8.4.1 消失フレームの外挿:グッドフレーム後の消失フレームの場合 .....	45
8.4.2 消失フレームの外挿:消失フレームの次の消失フレームの場合 .....	53
8.4.3 信号の再同期アルゴリズムおよびクロスフェーディング .....	53
8.4.4 動的なエネルギー調整アルゴリズム .....	56
8.4.5 補償信号と復号信号とのクロスフェーディング .....	57
8.5 高域FERC .....	57
8.5.1 正常フレームの場合における高域信号バッファの更新 .....	58
8.5.2 正常フレーム受信直後に消失フレームを受信した場合における正規化関連の計算および高域ピッチラグの推定 .....	58
8.5.3 相関が高い場合 .....	59
8.5.4 相関が低い場合 .....	59
8.6 合成QMF .....	60
8.7 ノイズゲート .....	60
9. 伝送パラメータインデックスの詳細 .....	61
10. JT-G711. 1コーデックのビットイグザクト記述 .....	63
10.1 シミュレーションソフトウェアの使い方 .....	63
10.2 シミュレーションソフトウェアの構成 .....	64
付属資料A 浮動小数点実装 .....	67

A. 1	アルゴリズム記述	67
A. 2	ANSI Cコード	67
付属資料B	JT-H245におけるJT-G711.1 1 使用法	69
B. 1	本付属資料の規定範囲	69
B. 2	参考文献	69
B. 3	JT-G711.1 フレームに対するパケット構造	69
B. 4	JT-H245に対するJT-G711.1 機能定義	69
B. 5	JT-G711.1 に対する互換性	71
付録I	R1モードに対する低域ポストフィルタ	72
I. 1	窓掛けとFFT	73
I. 2	量子化ノイズのPSD	73
I. 3	ポストフィルタ処理	74
I. 4	時間領域ポストフィルタ処理 (補間付き時間OLS処理)	76
I. 5	振幅補正の制限	76
I. 6	ポストフィルタのビットイグザクト記述	77
I.6.1	シミュレーションソフトウェアの使用法	77
I.6.2	シミュレーションソフトウェアの構成	77
	用語対照表	79

## <参考>

### 0. 本標準の概要

本標準は、64, 80 および 96kbit/s で動作する、G. 711 エンベデッド広帯域音声・オーディオ符号化アルゴリズムを記述したものである。

符号化器入力信号および復号器出力信号はデフォルトでは 16kHz サンプリングであるが、8kHz サンプリングにも対応している。16kHz サンプリング時には、J T-G 711. 1 符号器は 50-7000Hz の帯域幅の信号を 80kbit/s および 96kbit/s のビットレートで符号化し、8kHz サンプリング時には、64kbit/s および 80kbit/s のビットレートで動作し、50-4000Hz の帯域幅の信号を出力する（復号器からの狭帯域信号出力の帯域幅は、遮断周波数が 4000Hz の分割帯域フィルタバンクによって特性付けられる）。64kbit/s において J T-G 711. 1 は G. 711 と互換性があり、既存の G. 711 ベースの V o I P システムに対して本標準の効果的な導入が見込める。本符号器は 5ms のフレームで動作し、最大のアルゴリズム遅延は 11.875ms であり、演算量の最悪値は 8.70wMOPS である。

符号化器は、動作可能な 3 種類のビットレート、64, 80 および 96kbit/s に対応する 3 つのレイヤで構成されるエンベデッドなビットストリームを生成する。ビットストリームは、所望のビットレートに調整するために、復号器側または通信システムの任意の構成機器で切捨てができるが、どのレイヤが含まれているかを示す情報はビットストリームに含まれていないため、実装の際には、どのレイヤまでが利用可能かを示すアウトバンドシグナリングを必要とする。

符号化アルゴリズムは、ノイズフィードバックを伴う低域の対数圧伸パルス符号変調（PCM）、低域の基本レイヤの音質向上のための適応ビット割り当て用いるエンベデッド PCM 拡張、および変形離散コサイン変換（MDCT）に基づく高域の重み付きベクトル量子化、の 3 つのレイヤの符号化で構成される。

付属資料 A では、浮動小数点演算に適したハードウェア上での使用を容易にするための浮動小数点算術演算を用いた J T-G 711. 1 アルゴリズムの実装を定義する。日本 I T U 協会から入手可能な、本浮動小数点演算による C コードは、固定小数点演算に基づく C コードと完全に相互接続性があり等価な性能を供する。

付録 I は、復号器で使用する付加的なポストフィルタについて記述したものである。ポストフィルタは既存の G. 711、または J T-G 711. 1 の基本対数圧伸 PCM 部のビットストリームのみが利用可能な場合の復号信号の音質を向上させる。本ポストフィルタは、エンドユーザ端末用に意図されたものであり、（信号ミキサーやビットストリーム変換器におけるような）タンデムでの使用は避けるべきである。

A N S I C ソースコードは、本体の固定小数点演算実装、および付属資料 A の浮動小数点演算実装のものが共に用意されており、日本 I T U 協会から入手可能である。

### 1. 国際勧告等との関連

本標準は、2008年2月に I T U-T S G 16 で A A P に進むことが合意され、2008年3月に承認された I T U-T 勧告 G. 711. 1、および、2008年10月に I T U-T S G 16 で A A P に進むことが合意され、2008年11月に承認された、I T U-T 勧告 G. 711. 1 に対する Amendment 1 に準拠したものである。

また、本標準の付属資料 B は、2009年3月に承認された I T U-T 勧告 G. 711. 1 に対する A m e n d m e n t 2 に準拠して改定されたものである。

### 2. 上記国際勧告等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、削除した項目はない。
- (4) 本標準は、上記ITU-T勧告に対し、変更した項目はない。

### 2.4 原勧告との章立て構成比較

上記国際勧告等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	2009年2月25日	制定
第2版	2011年5月31日	付属資料Bの追加、及び標準本体の誤記修正

## 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、標準等

ITU-T勧告： G. 711

ITU-T勧告： G. 191

- (2) TTC標準JT-G711.1は、ITU-T勧告G. 711.1に準拠しており、本標準中で言及しているCコードおよびテストシーケンスとは、ITU-T勧告G. 711.1のものをさし、日本ITU協会から入手可能である。
- (3) 本標準においては、上記ITU-T勧告G. 711.1に記載されているA則符号化方式に対する事項も便宜上、記載されているが、我が国では $\mu$ 則符号化方式のみが採用されており、A則符号化方式は使用されていないことに留意されたい。またG. 711.1にA則符号化方式に対する記載もあることから、本標準での拡張のベースとする符号化標準として、ITU-T勧告であるG. 711を参照している。なお、G. 711の $\mu$ 則符号化方式に関しては、TTC標準JT-G711で規定され、その記述がある。

## 1. 本標準の規定範囲

本標準は、狭帯域および広帯域の音声およびオーディオ信号に対する 64~96kbit/s スケーラブル符号化のための、ITU-T 勧告 G. 711 に対する拡張アルゴリズムを記述したものである<sup>TTC 注</sup>。

本標準は、以下のように構成される。2章、3章、4章および5章で、本標準で用いられる参考文献、定義、略語および慣例がそれぞれ定義される。6章では JT-G 711. 1 のアルゴリズムの概要を示す。そして7章および8章において、JT-G 711. 1 の符号器および復号器の動作原理をそれぞれ記述する。9章では伝送パラメータについて説明する。10章では、本標準の符号器を 16-32 ビットの固定小数点演算で定義したソフトウェアについて記述する。更に、本標準本体で規定するアルゴリズムに加えて、付録 I では、既存の G. 711、すなわち基本となる対数圧伸 PCM 部のみが得られる場合に、復号信号の音質改善を図る補助的な機能を提供する。

TTC 注) 本標準では、符号化アルゴリズムのみを規定し、システムへの適用に関連する規定（例：IP 網伝送時の RTP ペイロードフォーマット、等）は本標準とは別に規定される。

## 2. 参考文献

### 2.1 必須文献

下記の ITU-T 勧告は、本標準での参照を通して本標準の規定を構成するものである。全ての標準および他の参考文献は、改定に従うものとする。従って、本標準のユーザには、以下の標準やその他の参照すべき文献について、最新版の適用の可能性を調査するよう奨励される。現在有効な ITU-T 勧告のリストは定期的に出版されている。

本標準内での文書の参照は、単独の文書としては、それを標準の扱いとはしない。

- ITU-T 勧告 G. 711 *Pulse Code Modulation (PCM) of Voice Frequencies*
- ITU-T 勧告 G. 191 *Software Tools for Speech and Audio Coding Standardization*

### 2.2 参考文献

[ITU-T G.192] ITU-T Recommendation G.192 (1996), A common digital parallel interface for speech standardization activities

[Blahut, 1985] Blahut, R.E., Fast Algorithms for Digital Signal Processing, Addison-Wesley, 1985

[Levinson, 1947] N. Levinson, "The Wiener RMS (root mean square) error criterion in filter design and prediction," J. Math. Phys., vol.25, no.4, pp.261-278, 1947

## 3. 定義

本章は空白とする。

## 4. 略語

本標準で使用される頭字語を Table 1 / JT-G711.1 に示す。

Table 1 / JT-G711.1 Glossary of acronyms  
(ITU-T G.711.1)

Acronym	Description
DeMUX	Demultiplexer
CSVQ	Conjugate Structured Vector Quantization
FER	Frame erasure
FERC	Frame erasure concealment
FIR	Finite impulse response
FFT	Fast Fourier Transform
HB	Higher Band
iFFT	Inverse FFT
iMDCT	Inverse MDCT
LB	Lower-Band
LP	Linear Prediction
LPC	Linear Prediction Coefficient
LSB	Least Significant Bit
LTP	Long-Term Prediction
MDCT	Modified Discrete Cosine Transform
MSB	Most Significant Bit
MUX	Multiplexer
NB	Narrow Band
NG	Noise Gate
OLA	Overlap and add
OLS	Overlap and save
PCM	Pulse code modulation
PSD	Power spectral density
QMF	Quadrature-mirror filterbank
RMS	Root mean square
RSX	Re-synchronized
SNR	Signal to (quantization) noise ratio
TDAC	Time Domain Aliasing Cancellation
VQ	Vector Quantization
WB	Wideband
WMOPS	Weighted million operations per second

## 5. 慣例

表記に関する慣例を以下に示す。

- 時間領域の信号は、そのシンボルと丸括弧で括られたサンプル番号で記述する(例  $s(n)$ )。変数  $n$  は、サンプル番号である。
- 周波数領域に変換された信号は、対応する時間領域の信号を大文字に変えることにより記述する (例  $S(k)$  は  $s(n)$  の変換)。変数  $k$  は、係数の番号である。
- 丸括弧で括られた上付きの添字は、時間に依存する変数に用いる(例  $g^{(m)}(n)$ )。変数  $m$  は、その前後関係よりフレーム番号、あるいはサブフレーム番号に対応し、変数  $n$  は、サンプル番号に対応している。
- 再帰を示す添字は、角括弧で括られた上付きで記述する(例  $E^{[k]}$ )。
- 下付きの添字は、係数配列の各要素を示す。



- 記号  $\hat{\cdot}$  は量子化されたパラメータを示す(例  $\hat{g}_c$ )。
- パラメータの範囲は、角括弧で括られた値で記述する。この値は境界値を含む(例  $[0.6, 0.9]$ )。
- 極性関数は、値の極性を与え、 $\text{sgn}(x)$  で表される。ここで、

$$\text{sgn}(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ -1 & \text{if } x < 0 \end{cases}$$

- 整数化関数  $\lfloor x \rfloor$  は、負の無限大側への  $x$  の丸めを示す( $\lfloor x \rfloor = \max\{n \in \mathbb{Z} \mid x \geq n\}$ )。
- 個所によっては、ビット特有の演算が使用される。ここで、 $\otimes$  および  $\oplus$  は、AND ビット演算および XOR ビット演算を表す。
- 先頭に“0x”が付いた定数は、16進で表記された値であることを示す。
- 変数  $x$  の  $N$  ビット右シフト演算は、2の $-N$ 乗の floor 化付き乗算で表される。(例  $\lfloor 2^{-N} x \rfloor$ )
- 16ビット固定小数点ANSI Cでの実装においては、浮動小数点の値は、対応する16/32ビット表現で丸めが行われる。

本標準全般にわたる主要なシンボルを、Table 2 / JT-G711.1 に示す。

Table 2 / JT-G711.1 Glossary of most relevant symbols  
(ITU-T G.711.1)

Type	Name	Description
Filters	$F(z)$	Perceptual weighting filter in the lower-band encoder/decoder
	$P(z)$	Lower-band pre-emphasis filter
	$A(z)$	Short term LP analysis filter in lower-band FERC
	$H_{dec}(z)$	Low-pass filter in lower-band FERC
	$B(z / \gamma_{PLC})(z)$	Weighting filter in lower-band FERC
	$H_{pre}(z)$	Pre-processing filter in FERC
	$W_{p2}(k)$	The noise-reduction filter in lower-band postfilter (in Appendix I)
	$Wf_{p1}(k)$	The first noise-reduction filter in lower-band postfilter (Appendix I)
	$Wf_{p2}(k)$	The second noise-reduction filter in lower-band postfilter (Appendix I)
Signals	$s_{WB}(n)$	Wideband input signal
	$s_{NB}(n)$	Narrowband input signal
	$\tilde{s}_{WB}(n)$	Pre-processed wideband input signal
	$\tilde{s}_{NB}(n)$	Pre-processed narrowband input signal
	$s_{LB}(n)$	Lower-band signal after QMF processing (decimated)
	$s'_{LB}(n)$	Perceptually weighted lower-band target signal
	$d_{L0}(n)$	Lower band difference signal of $s_{LB}(n)$ and $s'_{LB}(n)$
	$\hat{s}_{L0}(n)$	Decoded signal of Layer 0 bitstream
	$s'_{L0}(n)$	Decoded signal of Layer 0 bitstream, without offset $c_{Loff}$
	$s''_{L0}(n)$	Pre-emphasized decoded-signal of Layer 0 for LP analysis of lower-band perceptual filter

Type	Name	Description
	$\hat{S}_{L0w}(n)$	Windowed pre-emphasized decoded-signal in LP analysis of perceptual filter
	$r_{L0}(k)$	Autocorrelation function used in the lower-band perceptual filter
	$r'_{L0}(k)$	Lag-windowed autocorrelation function used in the lower-band perceptual filter
	$s'_{LB\text{exp}}(n)$	Lower-band target signal exponent (3-bit precision)
	$s'_{LB\text{ref}}(n)$	Lower-band target signal's refinement signal (3-bit precision)
	$s_{HB}(n)$	Higher-band signal after QMF processing (decimated)
	$s_{HB}^{\eta}(n)$	Pre-scaled higher-band signal of $s_{HB}(n)$
	$S_{HB}^{TDAC}(k)$	Higher-band MDCT coefficients before post-scaling
	$S_{HB}(k)$	Higher-band MDCT coefficients
	$S_{HBw}(k)$	Weighted higher-band MDCT coefficients
	$\bar{S}_{HBw}(k)$	Weighted and normalized higher-band MDCT coefficients
	$S'_{HB}(v)$	Decimated higher-band MDCT coefficients as CSVQ targets
	$\hat{S}'_{LB\text{exp}}(n)$	Lower-band exponent signal in decoder
	$\hat{S}'_{LB\text{sgn}}(n)$	Lower-band signal sign in decoder
	$\hat{e}_{L1}(n)$	Lower-band enhancement signal in decoder
	$\hat{S}_{L1}(n)$	Layer 1 decoded signal in decoder
	$\hat{S}_{LB}(n)$	Lower-band signal after decoding
	$\hat{S}_{HBw}(k)$	Decoded higher-band MDCT coefficients with weighting and normalization
	$\hat{S}_{HBw}(k)$	Decoded higher-band MDCT coefficients with weighting
	$\hat{S}_{HBm}(k)$	Decoded higher-band MDCT coefficients
	$\hat{S}_{HB\_OLA}(n)$	Higher-band signal before OLA in MDCT
	$\hat{S}_{LB}^{pre}(n)$	Pre-processed decoded signal in lower-band FERC
	$r_{LB}(n)$	LP residual signal in lower-band FERC
	$\hat{S}_{LB}^{extr}(n)$	Extrapolated signal in lower-band FERC
	$t(n)$	Decimated signal for pitch estimation in lower-band FERC
	$t_w(n)$	Weighted and decimated signal for pitch estimation in lower-band FERC
	$\hat{S}_{LB}^{extrg}(n)$	Extrapolated signal with gain adjustment in lower-band FERC
	$\hat{S}_{LB}^{RSX}(n)$	Re-synchronized signal in lower-band FERC
	$\hat{S}_{LB}^{SYN}(n)$	Scaled re-synchronized signal in lower-band FERC
	$\hat{S}_{HB}(n)$	Higher-band signal after decoding

Type	Name	Description
	$\hat{S}_{HB,PLCbuff}(n)$	A higher-band signal buffer in higher-band FERC
	$\hat{S}_{HB}(k)$	Higher-band MDCT coefficients after decoding
	$\hat{S}_{WB}(n)$	Decoded wideband signal
	$\hat{S}_{L0}(k)$	Layer 0 decoded signal in frequency domain (Appendix I)
	$\hat{S}'_{L0}(k)$	Output signal of the postfilter (Appendix I)
Parameters	$h_0^{qmf}(i)$	QMF coefficient set 1
	$h_1^{qmf}(i)$	QMF coefficient set 2
	$\beta_e$	Lower-band pre-emphasis filter coefficient
	$c_{zcl}$	Zero-crossing rate calculated in lower-band encoder
	$w_{LP1}(i)$	LP analysis window for lower-band perceptual filter
	$w_{lag}(i)$	LP analysis lag window of lower-band perceptual filter
	$c_{Loff}$	Core encoder offset value
	$\gamma_{P1}$	Perceptual weighting coefficient of lower-band
	$a_i$	LP coefficient of the lower-band perceptual filter
	$k_i$	Reflection coefficient obtained in lower-band perceptual filter
	$\eta_{LB}$	Lower-band signal normalization factor
	$\alpha_s$	Attenuation factor of the perceptual filter in the lower-band
	$M_{exp}$	Exponent map of the lower-band signal used for Layer 1 calculation
	$B_A(n)$	Bit-allocation table of Layer 1
	$w_{TDAC}(i)$	Higher-band MDCT overlap window
	$\eta_{HB}^{TDAC1}$	Higher-band MDCT normalization factor for pre-scaling
	$\eta_{HB}^{TDAC2}$	Higher-band MDCT normalization factor for post-scaling
	$\eta_{HB}$	Higher-band MDCT normalization factor
	$w_{HB}$	Higher-band MDCT coefficient weighting factor
	$g_{HB}$	RMS value of weighted MDCT coefficients $S_{HBw}(k)$
	$\epsilon_g$	A value added to avoid zero-divide in gain index calculation of MDCT coefficients
	$I_{Hs0pre}(v,l),$ $I_{Hs1pre}(v,l)$	Pre-selected Channel-0 and -1 candidate indices of $v$ -th decimated vector in Layer 2
	$p_{Hs0pre}(v,l),$ $p_{Hs1pre}(v,l)$	Pre-selected Channel-0 and -1 candidate signs of $v$ -th decimated vector in Layer 2

Type	Name	Description
	$D_{Hs0pre}(v, l)$ , $D_{Hs1pre}(v, l)$	Pre-selected Channel-0 and -1 candidate distortion measure of $v$ -th decimated vector in Layer 2
	$g_{Hlog}$	RMS value of weighted MDCT coefficients $S_{HBw}(k)$ in u-law domain
	$\hat{g}_{HB}$	Gain value of weighted and normalized MDCT coefficients $\hat{S}_{HBw}(k)$ in decoder
	$\hat{\eta}_{HB}$	Higher-band MDCT normalization factor in decoder
	$w_{LP2}(i)$	LP analysis window for lower-band FERC
	$T_{LB}$	Lower-band pitch lag obtained in lower-band FERC
	$R_{max}$	Maximum of the auto-correlation in lower-band FERC
	$g_{mute}$	Adaptive gain factor in lower-band FERC
	$I_{L0}$	G.711 compatible core bitstream (Layer 0)
	$I_{L1}$	Lower-band enhancement bitstream (Layer 1)
	$I_{L2}$	Higher-band enhancement bitstream (Layer 2)
	$\gamma_{PLC}$	Weighting factor in lower-band FERC
	$R'_{norm}$	Normalized cross-correlation in lower-band FERC
	$T_{ds}$	Preliminary estimation of the pitch delay in lower-band FERC
	$n_0$	First zero crossing position of weighted decimated signal in lower-band FERC
	$i_0$	Temporary index for maximum correlation search in lower-band FERC
	$i_1$	The first delay in $[1, 35]$ for maximum correlation search in lower-band FERC
	$i_2$	The lower bound for the maximum correlation search in lower-band FERC
	$c_{zc2}$	Zero-crossing rate of Pre-processed decoded signal in lower-band FERC
	$c_{peak}$	Number of large peaks detected in last pitch period in lower-band FERC
	$n_{maxr}$	The position of the maximum amplitude sample in the repetition period in lower-band FERC
	$A_{maxr}$	The amplitude of the maximum amplitude sample in the repetition period in lower-band FERC
	$A_{meanr}$	The mean amplitude of the repetition period in lower-band FERC
	$n_{maxr2}$	A second maximum amplitude position at the other bound of the repetition period in lower-band FERC
	$\Delta_e$	A threshold used for Modifications of sample amplitudes in the repetition period in lower-band FERC
	$\alpha_{es}$	Dynamic energy scaling factor in lower-band FERC
	$w_{cf}$	A triangular window for cross-fading in lower-band FERC
	$T_{HB}$	Estimated higher-band pitch lag in higher-band FERC

Type	Name	Description
	$R_{HB}$	Normalized correlation in higher-band FERC
	$R_{HB\_MAX}$	Maximum correlation in higher-band FERC
	$\zeta_{HB}$	High correlation flag in higher-band FERC
	$T_{HBmax}$	Maximum pitch lag in higher-band FERC
	$T_{HBmin}$	Minimal pitch lag in higher-band FERC
	$\alpha_{HBPLC1}^{(n)}$	Sample-by-sample attenuation weighting factor in higher-band FERC
	$\alpha_{HBPLC2}$	An attenuation factor in higher-band FERC
	$ N_{L0}(k) ^2$	A frequency-dependent noise power spectral density (PSD) estimation in lower-band postfiltering
	$w_a(n)$	A Hanning asymmetrical window in lower-band postfiltering
	$\Gamma_{L0}$	A load factor in lower-band postfiltering
	$SNR_{post}(k)$	<i>A posteriori</i> SNR for each frequency bin in lower-band postfiltering
	$SNR_{prior1}^{(m)}(k)$	<i>A priori</i> SNR on the current frame in lower-band postfiltering
	$\gamma_w(n)$	First half of a 16-length Hanning window in lower-band postfiltering
	$d'_{max}$	Allowed maximum distortion in lower-band postfiltering

## 6. 符号器の概要記述

JT-G711.1 広帯域拡張コーデックは、ITU-T 勧告 G.191 ソフトウェアツールライブラリ (2005) で定義された基本演算子 (第 2.2 版) を用いた固定小数点演算により実現される。本標準では、その詳細アルゴリズムを記述する。

### 6.1 符号器

Figure 6-1/JT-G711.1 に符号器の概略ブロック図を示す。0-50Hz の成分を取り除くため、16kHz サンプリングの入力信号  $s_{WB}(n)$  に対して前処理の高域通過フィルタが適用される。前処理後の信号  $\tilde{s}_{WB}(n)$  は、32 タップの直交ミラーフィルタバンク (QMF) を用いて、8kHz サンプリングの低域信号  $s_{LB}(n)$  と高域信号  $s_{HB}(n)$  とに分割される。低域信号は、64kbit/s の G.711 互換のコア (レイヤ 0) のビット列  $I_{L0}(n)$  と 16kbit/s の低域の拡張 (レイヤ 1) のビット列  $I_{L1}$  を生成するエンベデッドな低域符号器で符号化される。高域信号は、変形離散コサイン変換 (MDCT) 領域へ変換され、周波数領域係数  $S_{HB}(k)$  は、その正規化係数  $\eta_{HB}$  と共に、16kbit/s の高域の拡張 (レイヤ 2) のビット列  $I_{L2}$  を生成する高域符号器で符号化される。全てのビット列は、スケーラブルなビット列として多重化される。

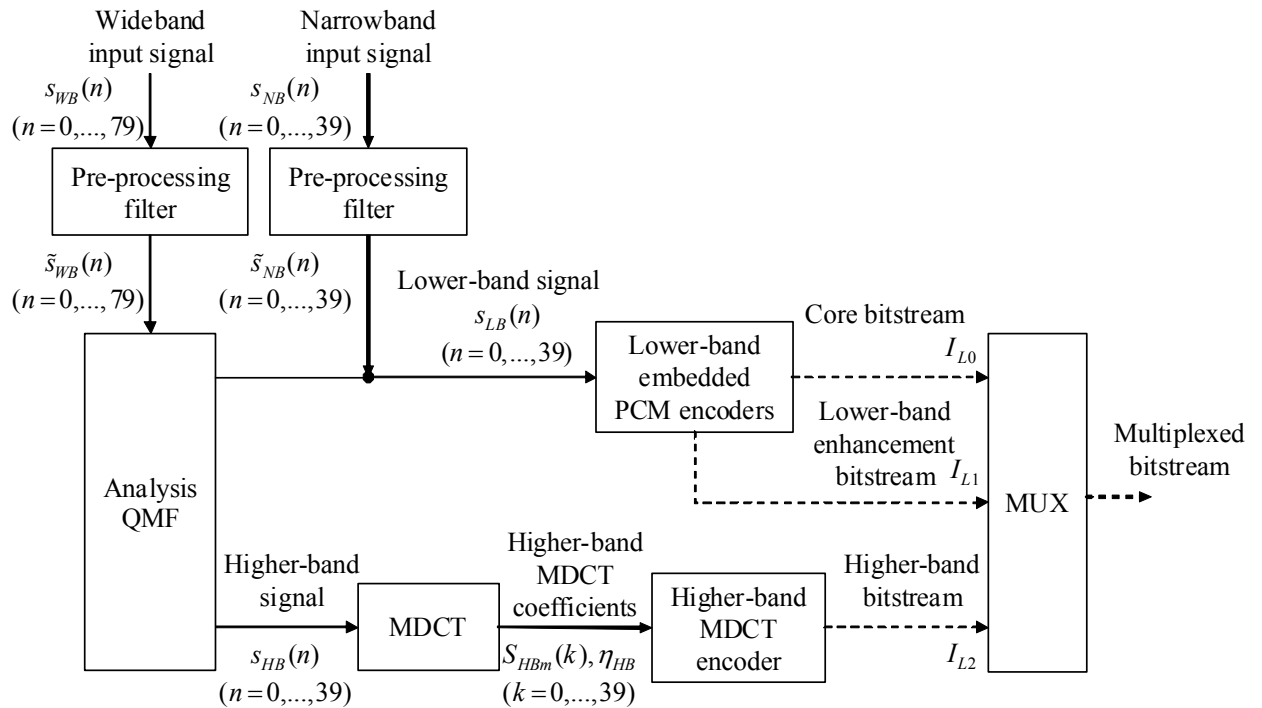


Figure 6 – 1 / JT-G711.1 High-level encoder block diagram  
(ITU-T G.711.1)

## 6.2 復号器

Figure 6 – 2 / JT-G711.1 に復号器の概略ブロック図を示す。全体のビット列は、G. 711 互換のコア（レイヤ0）のビット列  $I_{L0}$ 、低域の拡張（レイヤ1）のビット列  $I_{L1}$  および高域の拡張（レイヤ2）のビット列  $I_{L2}$  に分離される。レイヤ2のビット列は高域復号器へ与えられ、その結果得られる周波数領域の復号信号  $\hat{S}_{HB}(k)$  は、正規化係数  $\eta_{HB}$  と共に逆MDCT（iMDCT）に入力される。この、逆MDCT処理により、時間領域の高域信号  $\hat{S}_{HB}(n)$  が得られる。パケット損失のような通信路誤りによるフレーム消失が生じた際の品質を改善するために、低域信号と高域信号に対して独立に、フレーム消失補償(FERC)アルゴリズムが適用される。補償処理は独立に行われるが、低域のFERCで推定されたピッチラグ  $T_{LB}$  は、高域のFERCに補助情報として与えられる。低域信号  $\hat{S}_{LB}(n)$  および高域信号  $\hat{S}_{HB}(n)$  は、合成QMFフィルタバンクを用いて結合され、広帯域信号  $\hat{S}_{QMF}(n)$  を生成する。低レベルの背景雑音を低減するため、QMF出力に対してノイズゲート処理が適用される。復号器出力として、16kHz サンプリングの音声  $\hat{S}_{WB}(n)$ （または8kHz サンプリングの音声  $\hat{S}_{NB}(n)$ ）が合成される。

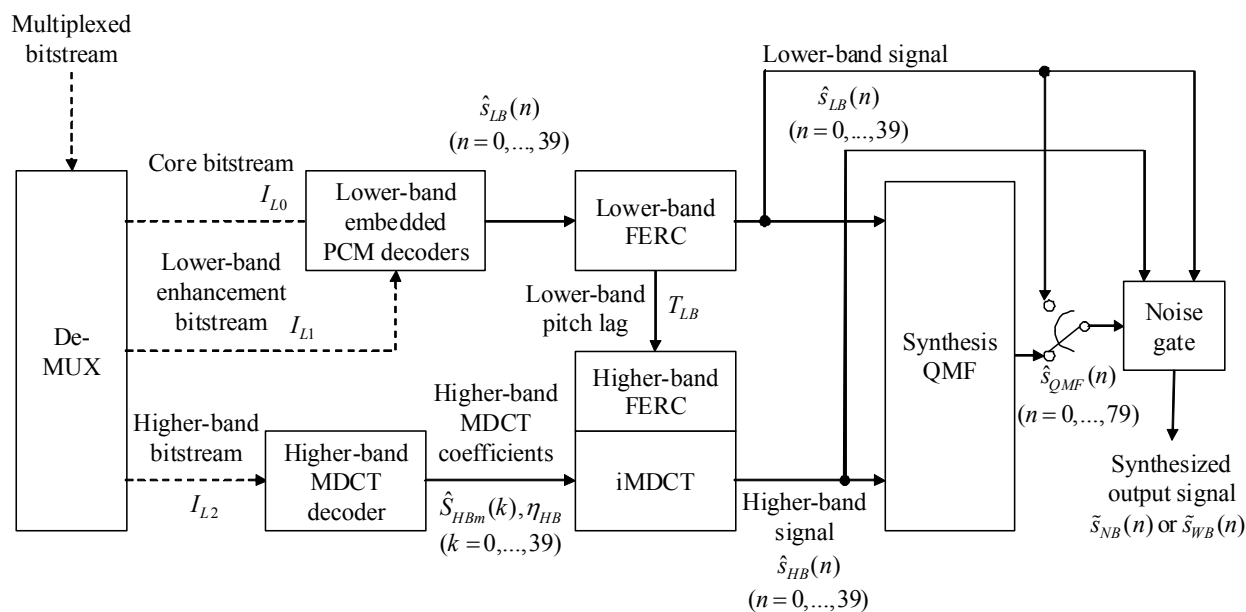


Figure 6 – 2 / JT-G711.1 High-level decoder block diagram  
(ITU-T G.711.1)

### 6.3 符号化モード

とり得るビット列の組み合わせを Table 3 / JT-G711.1 に示す。ビット列の組み合わせに対し、全部で 4 モード、R1, R2a, R2b および R3 が定義される。R2a と R2b とはビット列自身では区別できないため、モードを明示するアウトバンドシグナリングが必要である。

Table 3 / JT-G711.1 Sub-bitstream combination for each mode  
(ITU-T G.711.1)

Mode	Sampling rate (kHz)	Core layer (Layer 0, $I_{L0}$ )	Lower-band enhancement layer (Layer 1, $I_{L1}$ )	Higher-band enhancement layer (Layer 2, $I_{L2}$ )	Overall bit-rate (kbit/s)
		64 kbit/s	16 kbit/s	16 kbit/s	
R1	8	x	-	-	64
R2a	8	x	x	-	80
R2b	16	x	-	x	80
R3	16	x	x	x	96

### 6.4 ビット割り当て

符号器のビット割り当てを Table 4 / JT-G711.1 に示す。本テーブルはビット列レイヤ毎に対応させて構成している。6.3 節に記述するように、与えられたビットレートに対して、それに寄与するレイヤの連結によりビット列が得られる。

Table 4 / JT-G711.1 Bit allocation per 5 ms frame  
(ITU-T G.711.1)

Parameter	Symbol	Bits per frame
<b>Layer 0 (G.711 compliant)</b>		
G.711 code	$I_{L0}(0), \dots, I_{L0}(39)$	8 bits x 40 samples
<b>Layer 0 subtotal</b>		<b>320 bits</b>
<b>Layer 1</b>		
G.711 refinement code	$I_{L1}(0), \dots, I_{L1}(39)$	$\sum_{n=0}^{39} B_A(n), 0 \leq B_A(i) \leq 3$
<b>Layer 1 subtotal</b>		<b>80 bits</b>
<b>Layer 2</b>		
MDCT gain	$I_{Hg}$	8 bits
MDCT VQ index	$I_{Hs0}(0), \dots, I_{Hs0}(5)$	5 bits x 6 sub-vectors
MDCT VQ index	$I_{Hs1}(0), \dots, I_{Hs1}(5)$	5 bits x 6 sub-vectors
MDCT vector sign	$I_{Hp0}(0), \dots, I_{Hp0}(5)$	1 bit x 6 sub-vectors
MDCT vector sign	$I_{Hp1}(0), \dots, I_{Hp1}(5)$	1 bit x 6 sub-vectors
<b>Layer 2 subtotal</b>		<b>80 bits</b>
<b>TOTAL</b>		<b>480 bits</b>

## 6.5 アルゴリズム遅延

J T - G 7 1 1 . 1 符号器は、11.875ms (16kHz で 190 サンプル) のアルゴリズム遅延を有する。遅延の内訳は以下の通りである。

- 入力フレームの 5ms
- M D C T 分析のための 5ms (先読み)
- Q M F 分析-合成フィルタバンクのための 1.875ms

## 6.6 演算量と所要メモリ量

I T U - T 勧告 G . 1 9 1 のソフトウェアツールライブラリ STL2005 (第 2.2 版) の基本演算子に基づく、J T - G 7 1 1 . 1 符号器 (符号器+復号器) の演算量の最悪値は、8.70WMOPS である。最悪値の演算量の詳細を Table 5 / JT-G711.1 に示す。全ての数値は u または A 則のいずれかの最悪値を示す。ポストフィルタありとなしの各々のケースでの J T - G 7 1 1 . 1 の 16-bit ワードでの所要メモリ量を、それぞれ、Table 6 / JT-G711.1 および Table 7 / JT-G711.1 に示す。RAM の数値は、支配的な部分を占める配列を基に求めたものであり、単一変数を含まない。配列に必要なサイズに比べると単一変数の数は重要ではない。



Table 5 / JT-G711.1 Worst computational complexity of the JT-G711.1 coder (WMOPS)  
(ITU-T G.711.1)

Encoder	Decoder		
5.396	No FER	R1	0.700
		R1 with postfilter	2.679
		R2a	1.181
		R2b	1.852
		R3	2.333
	FER	R1	2.476
		R1 with postfilter	3.049
		R2a	2.476
		R2b	3.304
		R3	3.304

Table 6 / JT-G711.1 Storage requirements of the G.711.1 coder  
(ITU-T G.711.1)

	Encoder	Decoder
Static RAM (kWords)	0.18	1.50
Scratch RAM (kWords)	0.66	0.70
Data ROM (kWords)	2.21	
Program ROM (number of basic ops)	1943	

Table 7 / JT-G711.1 Increase in the storage requirements of the G.711.1 decoder with R1 postfilter  
(ITU-T G.711.1)

	Decoder
Static RAM (kWords)	+0.33
Scratch RAM (kWords)	+0.20
Data ROM (kWords)	+0.19
Program ROM (number of basic ops)	+527

## 6.7 符復号器の記述

本標準の符号化アルゴリズムは、ビットイグザクトな固定小数点算術演算で記述されている。10章で示されるANSI Cコードは、本標準の必須部分を構成するものであり、このビットイグザクトな固定小数点での記述を反映している。符号器および復号器の数学的な記述は、他の方法でも実装し得るが、本標準に準拠しないコーデックを実装することになってしまう可能性がある。従って、不一致が生じた場合には、数学的な記述よりも、10章のANSI Cコードによるアルゴリズム記述の方が優先される。ANSI Cコードと共に用いられるテスト信号の非網羅的なセットは、日本ITU協会から入手可能である。

## 6.8 異なるモード間や既存のG. 711とのトランスコード

中継器においてビット列をトランスコードしなければならない場合、レイヤビット列の切り捨て（下方トランスコード）または、レイヤの追加（上方トランスコード）を行うことができる（R1モードは既存G. 711と互換である）。下方トランスコードは自明であるが、上方トランスコードは、追加されたレイヤにおける全てのビットを“0”にセットすることによって実現できる。それぞれのレイヤでの復号器が全てのビットを零に設定したビット列を受信した場合には、そのレイヤ出力はトランスペアレント、すなわち、各々の拡張信号は

0になる。

- レイヤ1に対して、 $I_{L2}$ が全て0の場合は、拡張信号 $e_{L1}(n)$ は0になり、その結果、 $\hat{s}_{L1}(n)$ は零になる(8.1節を参照)。
- 同様に、 $I_{L1}$ が全て0の場合、高域MDCT係数 $\hat{S}_{HBm}(k)$ は0になる。ただし、これは、復号広帯域信号 $\hat{s}_{WB}(n)$ が4000Hz以上でパワーを全く持たないということにはならない。それは、QMFが理想的な帯域分割フィルタバンクではないためである。

## 7. 符号化器の機能記述

### 7.1 前処理の高域通過フィルタ

前処理フィルタは、16kHz サンプリングの入力信号 $s_{WB}(n)$ に対して適用され、下記で定義される。

$$\tilde{s}_{WB}(n) = 0.984375 \cdot \tilde{s}_{WB}(n-1) + s_{WB}(n) - s_{WB}(n-1) \quad n = 0, \dots, 79 \quad (7-1)$$

ここで、 $\tilde{s}_{WB}(n)$ はフィルタ出力である。入力信号が狭帯域の8kHz サンプリングの信号の場合、フィルタ処理は以下により行われる。

$$\tilde{s}_{NB}(n) = 0.96875 \cdot \tilde{s}_{NB}(n-1) + s_{NB}(n) - s_{NB}(n-1) \quad n = 0, \dots, 39 \quad (7-2)$$

ここで、 $\tilde{s}_{NB}(n)$ は同様にフィルタ出力である。いずれの場合の前処理の高域通過フィルタ共に50Hzを遮断周波数として設計されている。

### 7.2 分析QMF

高域通過フィルタ処理後の入信号 $\tilde{s}_{WB}(n)$ に対して分析QMF  $h^{qmFA}$ を適用し、2つの8kHz サンプリングの信号である、低域信号 $s_{LB}(n)$ と高域信号 $s_{HB}(n)$ に分割する。8kHz サンプリングの低域信号 $s_{LB}(n)$ は、以下の処理により得られる。まず、下式で与えられる32個の係数をもつ対称FIR低域通過フィルタにより16kHz サンプリングの入力に対してフィルタ処理を行う。

$$s_{WL}(n) = \sum_{i=0}^{31} h_L^{qmFA}(i) \tilde{s}_{WB}(n-i) \quad n = 0, \dots, 79 \quad (7-3)$$

ここで、 $s_{WL}(n)$ は16kHz サンプリングの低域信号、 $h_L^{qmFA}(i)$ はフィルタ係数である。そして、下式のように $s_{WL}(n)$ を係数2で間引きする。

$$s_{LB}(n) = s_{WL}(2n+1) \quad n = 0, \dots, 39 \quad (7-4)$$

同様に、8kHz サンプリングの高域信号 $s_{HB}(n)$ は以下により得られる。

$$s_{WH}(n) = \sum_{i=0}^{31} h_H^{qmFA}(i) \tilde{s}_{WB}(n-i) \quad n = 0, \dots, 79 \quad (7-5)$$

$$s_{HB}(n) = s_{WH}(2n+1) \quad n = 0, \dots, 39 \quad (7-6)$$

高域通過および低域通過フィルタの係数、 $h_H^{qmFA}(i)$ および $h_L^{qmFA}(i)$ は、下記の関係がある。

$$h_H^{qmFA}(i) = (-1)^i h_L^{qmFA}(i) \quad i = 0, \dots, 31 \quad (7-7)$$

従って、 $s_{LB}(n)$ および $s_{HB}(n)$ は、以下のようにして直接算出できる。

$$s_{LB}(n) = \sum_{i=0}^{15} h_0^{qmF}(i) \tilde{s}_{WB}(2(n-i)) + \sum_{i=0}^{15} h_1^{qmF}(i) \tilde{s}_{WB}(2(n-i)+1) \quad n = 0, \dots, 39 \quad (7-8)$$

$$s_{HB}(n) = -\sum_{i=0}^{15} h_0^{qmF}(i) \tilde{s}_{WB}(2(n-i)) + \sum_{i=0}^{15} h_1^{qmF}(i) \tilde{s}_{WB}(2(n-i)+1) \quad n = 0, \dots, 39 \quad (7-9)$$

ここで、

$$\begin{aligned} h_0^{qmf}(i) &= h_H^{qmfA}(2i) \\ h_1^{qmf}(i) &= h_L^{qmfA}(2i+1) \end{aligned} \quad i = 0, \dots, 15 \quad (7-10)$$

Table 8 / JT-G711.1 に、係数  $h_0^{qmf}$  および  $h_1^{qmf}$  の値を示し、QMF の周波数特性を Figure 7-1 / JT-G711.1 に示す。

Table 8 / JT-G711.1 QMF coefficients  
(ITU-T G.711.1)

<b>i</b>	$h_0^{qmf}(i)$	$h_1^{qmf}(i)$
0	$-6.5064660 \times 10^{-4}$	$-1.3508480 \times 10^{-3}$
1	$-1.2601150 \times 10^{-3}$	$4.1581240 \times 10^{-3}$
2	$1.4272050 \times 10^{-3}$	$-9.3636330 \times 10^{-3}$
3	$-1.7219030 \times 10^{-4}$	$1.7881950 \times 10^{-2}$
4	$-4.1094160 \times 10^{-3}$	$-3.1155320 \times 10^{-2}$
5	$1.4468810 \times 10^{-2}$	$5.2909300 \times 10^{-2}$
6	$-3.9244910 \times 10^{-2}$	$-9.9800110 \times 10^{-2}$
7	$1.2846510 \times 10^{-1}$	$4.6645830 \times 10^{-1}$
8	$4.6645830 \times 10^{-1}$	$1.2846510 \times 10^{-1}$
9	$-9.9800110 \times 10^{-2}$	$-3.9244910 \times 10^{-2}$
10	$5.2909300 \times 10^{-2}$	$1.4468810 \times 10^{-2}$
11	$-3.1155320 \times 10^{-2}$	$-4.1094160 \times 10^{-3}$
12	$1.7881950 \times 10^{-2}$	$-1.7219030 \times 10^{-4}$
13	$-9.3636330 \times 10^{-3}$	$1.4272050 \times 10^{-3}$
14	$4.1581240 \times 10^{-3}$	$-1.2601150 \times 10^{-3}$
15	$-1.3508480 \times 10^{-3}$	$-6.5064660 \times 10^{-4}$

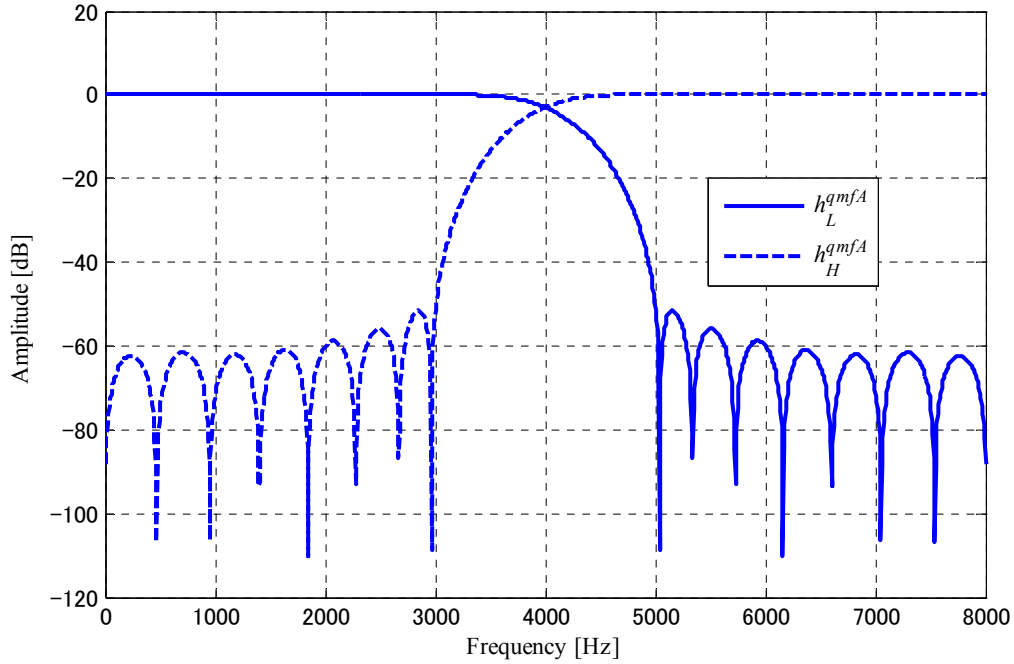


Figure 7-1 / JT-G711.1 QMF Frequency response  
(ITU-T G711.1)

入力が 8kHz サンプリングの場合、この QMF 処理は行わずに、下記に示すように前処理後の入力信号が低域信号にコピーされる。

$$s_{LB}(n) = \tilde{s}_{NB}(n), \quad n = 0, \dots, 39 \quad (7-11)$$

### 7.3 低域符号化

低域信号  $s_{LB}(n)$  は、パルス符号変調 (PMC) 符号化器の符号化ノイズを聴覚的にシェーピングするためのノイズフィードバックを有するエンベデッドな  $\mu$  則または A 則 PCM を使用して符号化される。重み付きノイズフィードバックループによる低域の符号化器を Figure 7-2 / JT-G711.1 に示す。図において、 $Q_{L0}$  は、G. 711 を改良したレイヤ 0 量子化器であり、 $Q_{L1}$  は、レイヤ 1 量子化器である (ここで、レイヤ 1 量子化器は、各サンプルに分解能が可変長のビットを付加する)。また、 $Q_{L0}^{-1}$  は、レイヤ 0 の復号器であり、G. 711 に準拠した復号器である。ここで、低域入力信号  $s_{LB}(n)$  に、ノイズフィードバック信号とオフセット値  $c_{Loff}$  が加算され、その加算後の信号  $s'_{LB}(n)$  がレイヤ 0 の量子化器  $Q_{L0}$  に入力される。得られたレイヤ 0 のビットストリーム  $I_{L0}(n)$  を基に、 $Q_{L0}^{-1}$  が局部復号を行い、信号  $\hat{s}'_{L0}(n)$  を得る。G. 711 の量子化精度を上げるために、精細信号  $s'_{LBref}(n)$  とその指数  $s'_{LBexp}(n)$  をレイヤ 1 の量子化器  $Q_{L1}$  に入力し、適応的に多重化されたビットストリーム  $I_{L1}(n)$  を出力する。一方、聴覚フィルタ  $F(z)$  は、 $\hat{s}'_{L0}(n)$  を用いて計算され、 $F(z)$  によってフィルタリングされた量子化ノイズ  $d_{L0}(n)$  は、フィードバックされて、入力信号  $s_{LB}(n)$  と加算される。なお、レイヤ 0 の符号化において、エネルギーが低い入力信号に対しては通常対数 PCM 符号化は適用されず、7.3.3 節で後述する「デットゾーン量子化器」と呼ばれる異なる符号化方法が用いられる。また、ノイズシェーピングを符号化器と復号器に共に用いる。ノイズシェーピングは、符号化器においては、従来の G. 711 復号器との相互接続性を保つようにレイヤ 0 信号に適用される一方で、復号器においては、レイヤ 0 の信号と加算する前のレイヤ 1 の信号に適用する。このように、適切なノイズシェーピングをレイヤ 0 だけではなくレイヤ 1 においても行なう。

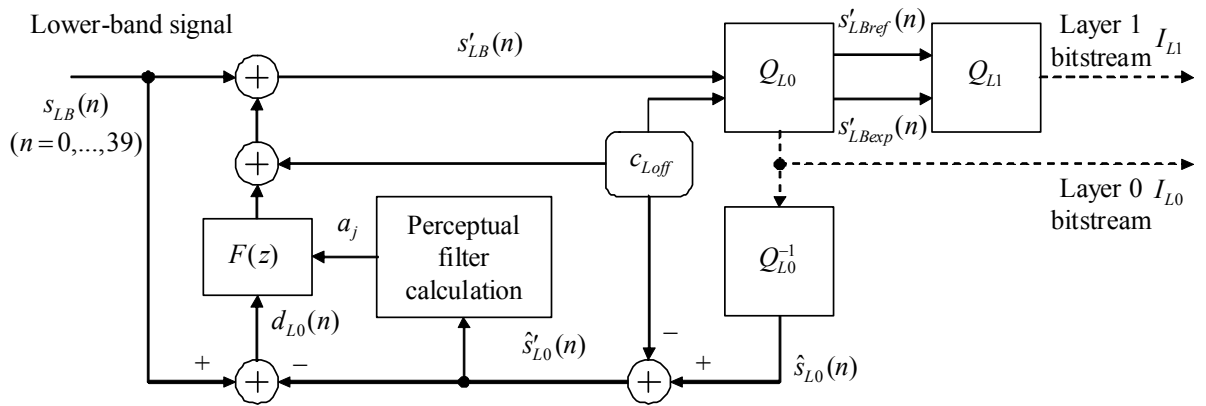


Figure 7-2 / JT-G711.1 Lower band encoder with weighted noise feedback loop  
(IUT-T G.711.1)

### 7.3.1 G. 711に基づくコアPCM符号化器（レイヤ0）

低域の符号化器は、レイヤ0のビットストリーム  $I_{L0}$  を生成するコアPCM符号化器と、それに続く、拡張ビットが付加されてレイヤ1のビットストリーム  $I_{L1}$  を生成する拡張ステージの2段階で入力信号を処理する。

コアのPCM符号化器は、既存のG. 711符号化器の変形であるが、既存のコーデックとの相互接続性は保持されるようになっている。各  $s'_{LB}(n)$  の16ビットのサンプルを8ビットに符号化する。ここで、1ビットは極性(S)、3つの指数ビット( $E_2, E_1, E_0$ )は圧縮セグメントを示し、4つの仮数部ビット( $M_3, M_2, M_1, M_0$ )は圧縮セグメントの位置を示す。しかしながら、G. 711に対して幾つかの変更点があり、詳細を以下の2つの節に記載する。

量子化は、 $\mu$ 則かA則のいずれかで行なわれる。各フレームには、40個のサンプルが含まれ、サンプル単位で量子化され、レイヤ0ビットストリーム  $I_{L0}(n)$  と、レイヤ1ビットストリーム  $I_{L1}(n)$ （いずれも  $n=0,1,\dots,39$ ）が生成される。符号化の過程で、指数  $s'_{LB\text{exp}}(n)$ 、拡張（または精細化）符号  $s'_{LB\text{ref}}(n)$ 、および局部復号された信号  $\hat{s}_{L0}(n)$  を計算して蓄積する。そのうちの最初の2つの値は、レイヤ1の適応多重化器で後に使用される（7.3.4参照）。一方、3番目の値は、ノイズフィードバックに用いられる（7.3.2参照）。

更に、符号化器の動作は、7.3.3節で説明する信号の条件に依存する。

使用される圧伸則に従って  $c_{Loff}$  のオフセット値を以下の式で定義する。

$$c_{Loff} = \begin{cases} 0 & \text{for } \mu\text{-law} \\ 8 & \text{for } A\text{-law} \end{cases} \quad (7-12)$$

$c_{Loff}$  が  $\mu$  則の場合にゼロであるため、この変数は以下の記述で省略する場合がある。

#### 7.3.1.1 $\mu$ 則 符号化過程

$$x = \min\left(|s'_{LB}(n) + c_{Loff}|, 32635\right) + 132$$

とし、極性を以下で与える。

$$s = \begin{cases} 0x80 & \text{if } s'_{LB}(n) \geq 0 \\ 0 & \text{if } s'_{LB}(n) < 0 \end{cases}$$

$$\begin{aligned}
e &= \lfloor \log_2(x) \rfloor - 7, \\
r &= \lfloor 2^{-e} \cdot x \rfloor \otimes 0x07, \\
m &= \lfloor 2^{-(e+3)} \cdot x \rfloor - 16, \\
\hat{s}_{L0}(n) &= \begin{cases} 2^e \cdot (2^3(m+16)+4) - 132 & \text{if } s = 0x80 \\ -(2^e \cdot (2^3(m+16)+4) - 132) & \text{if } s = 0 \end{cases} \\
I_{L0}(n) &= (s + 2^4 e + m) \oplus 0x7F
\end{aligned}$$

$s$ 、 $e$ 、 $r$  および  $m$  は、各サンプルを計算した一時的な変数であり、 $x$  の“極性”、“指数”、“量子化誤差”および“仮数”を意味する。レイヤ0の符号化処理の間、中間の変数 $e$  および  $r$  は以下のように保存され、7.3.4節に記述する適応多重化器に入力され、レイヤ1を生成する。

$$\begin{aligned}
s'_{LB\text{exp}}(n) &= e, \\
s'_{LB\text{ref}}(n) &= r
\end{aligned}$$

### 7.3.1.2 A則 符号化過程

$$x = \min\left(\left|s'_{LB}(n) + c_{L\text{off}}\right|, 32767\right)$$

とし、極性を以下で与える。

$$s = \begin{cases} 0x80 & \text{if } s'_{LB}(n) + c_{L\text{off}} \geq 0 \\ 0 & \text{if } s'_{LB}(n) + c_{L\text{off}} < 0 \end{cases}$$

if  $|x| > 255$

$$\begin{aligned}
e &= \lfloor \log_2(x) \rfloor - 7, \\
r &= \lfloor 2^{-e} \cdot x \rfloor \otimes 0x07, \\
m &= \lfloor 2^{e+3} \cdot x \rfloor - 16, \\
\hat{s}_{L0}(n) &= \begin{cases} 2^{e-1} \cdot ((2^4 m + 8) + 256) & \text{if } s = 0x80 \\ -2^{e-1} \cdot ((2^4 m + 8) + 256) & \text{if } s = 0 \end{cases}
\end{aligned}$$

else

$$\begin{aligned}
e &= 0 \\
r &= \left\lfloor \frac{x}{2} \right\rfloor \otimes 0x07 \\
m &= \lfloor 2^{-4} x \rfloor \\
\hat{s}_{L0}(n) &= \begin{cases} 2^4 m + 8 & \text{if } s = 0x80 \\ -(2^4 m + 8) & \text{if } s = 0 \end{cases}
\end{aligned}$$

いずれの条件においても、レイヤ0のインデックスは、 $I_{L0}(n) = (s + 2^4 e + m) \oplus 0x55$ により計算される。 $s$ 、 $e$ 、 $r$  および  $m$  は、各サンプルを計算した一時的な変数であり、 $x$  の“極性”、“指数”、“量子化誤差”および“仮数”を意味する。レイヤ0の符号化処理の間、以下のように中間の変数 $e$  および  $r$  が保存され、7.3.4節に記述する適応多重化器に入力され、レイヤ1ビットストリームを生成する。

$$\begin{aligned} s'_{LB\text{exp}}(n) &= e \\ s'_{LB\text{ref}}(n) &= r \end{aligned}$$

### 7.3.2 聴覚フィルタリング

#### 7.3.2.1 聴覚フィルタの算出

聴覚ノイズシェーピングフィルタの伝達関数は、以下の式で与えられる。

$$F(z) = A_0(z/\gamma_{p1}) - 1 \quad (7-13)$$

ここで、 $A_0(z)$ は、線形予測（LP）フィルタの伝達関数であり、 $\gamma_{p1}$ は聴覚重み係数である。上記のフィルタを使用して、PCMのスペクトル量子化ノイズを $1/A_0(z/\gamma_{p1})$ のスペクトルを用いてシェーピングする。ノイズシェーピングにおいて傾斜を制御するため、適応的なプリエンファシス係数を用いたプリエンファシス信号に基づくフィルタ $A_0(z)$ を計算する。このノイズシェーピングは、低域符号化の聴覚品質を改善し、符号化側ではレイヤ0に、復号側ではレイヤ1に適用されている（8.1節参照）。符号化器と復号器の不整合を避けるため、フィルタ $A_0(z)$ は、復号器でも利用できる過去の復号信号 $\hat{s}_{L0}(n)$ に基づいて計算される。

LPフィルタを範囲-80から-1の過去に復号化した信号 $\hat{s}_{L0}(n)$ に基づいて算出する。負の値のインデックスは、過去の信号（過去2フレームを示す範囲-80,...,-1）を参照する。7.3.1節で示したように、仮数の値はオフセット $c_{Loff}$ が加算された入力信号 $s'_{LB}(n)$ を用いて算出されていることを思い起こすべきである。まず、このオフセット値は、以下のように局部復号信号から差し引かれる。

$$\hat{s}'_{L0}(n) = \hat{s}_{L0}(n) - c_{Loff} \quad (7-14)$$

復号信号 $\hat{s}'_{L0}(n)$ におけるサンプルは、プリエンファシスされる。プリエンファシスフィルタは、伝達関数が $P(z) = 1 - \beta_e z^{-1}$ である1次のフィルタであり、 $\beta_e$ は、信号に依存して、以下のように算出される。

$$\beta_e = 0.38275 + 0.007813c_{zcl} \quad (7-15)$$

ここで、 $c_{zcl}$ はゼロクロス比率である。ゼロクロス比率は、以下の様に算出する。

$$c_{zcl} = \frac{1}{2} \sum_{n=-79}^{-1} |\text{sgn}[\hat{s}'_{L0}(n-1)] + \text{sgn}[\hat{s}'_{L0}(n)]| \quad (7-16)$$

この結果、 $0.38 < \beta_e < 1.0$ となる。プリエンファシス信号 $\hat{s}''_{L0}(n)$ は、以下のように算出される。

$$\hat{s}''_{L0}(n) = \hat{s}'_{L0}(n) - \beta_e \hat{s}'_{L0}(n-1) \quad n = -80, \dots, -1 \quad (7-17)$$

非対称窓を使用して、1フレームごとに、4次のLP分析をプリエンファシス信号に適用する。窓を2つに分割する。前半部の窓長は60サンプルであり、後半部の窓長は20サンプルである。窓関数は以下のように与えられる。

$$w_{LP1}(n) = \begin{cases} 0 & n = 0 \\ \frac{1}{2} \cos\left(\frac{\pi}{2L_1}\left(n + \frac{1}{2}\right) - \frac{\pi}{2}\right) + \frac{1}{2} \cos^2\left(\frac{\pi}{2L_1}\left(n + \frac{1}{2}\right) - \frac{\pi}{2}\right) & n = 1, \dots, L_1 - 1 \\ \frac{1}{2} \cos\left(\frac{\pi}{2L_2}\left(n - L_1 + \frac{1}{2}\right)\right) + \frac{1}{2} \cos^2\left(\frac{\pi}{2L_2}\left(n - L_1 + \frac{1}{2}\right)\right) & n = L_1, \dots, L_1 + L_2 - 1 \end{cases} \quad (7-18)$$

ここで  $L_1=60$  と  $L_2=20$  を用いる。過去のプリエンファシス信号  $\hat{s}_{L0}''(n)$  をこの窓関数を用いて掛け合わせると、信号  $\hat{s}_{L0w}(n)$  を得ることができる。

$$\hat{s}_{L0w}(n) = w_{LP1}(n) \hat{s}_{L0}''(n - 80) \quad n = 0, \dots, 79 \quad (7-19)$$

窓掛けされた信号  $\hat{s}_{L0w}(n)$  の自己相関関数は、以下のように算出される。

$$r_{L0}(k) = \varepsilon_{L0}(k) + \sum_{n=k}^{79} \hat{s}_{L0w}(n) \hat{s}_{L0w}(n-k) \quad k = 0, \dots, 4 \quad (7-20)$$

ここで、 $\varepsilon_{L0}(k)$  は、低エネルギーの信号に対してノイズシェーピングフィルタの適切な形状を保証するために各相関係数に加算される初期値であり、 $\varepsilon_{L0}(k) = 0.95^k \cdot 100^2$  で与えられる。そして、自己相関関数に対するラグ窓かけにより、120Hz 帯域幅拡張が適用される。ラグ窓かけは、下記のように自己相関関数に  $w_{lag}(k)$  を乗じることにより得られる。

$$r'_{L0}(k) = \begin{cases} r_{L0}(k) & k = 0 \\ w_{lag}(k) r_{L0}(k) & k = 1, \dots, 4 \end{cases} \quad (7-21)$$

窓関数は下記で定義される。

$$w_{lag}(i) = \frac{1}{1.0001} \exp\left[-\frac{1}{2} \left(\frac{2\pi f_0 i}{f_s}\right)^2\right] \quad i = 1, \dots, 4 \quad (7-22)$$

ここで、 $f_0=120$  Hz はバンド幅拡張で、 $f_s=8000$  Hz はサンプリング周波数である。乗算定数の  $1/1.0001$  は、LPフィルタ係数を安定化するための白色雑音補正に対する係数である。これは、窓掛けした信号レベルに対して-40dBのノイズフロアを加算することと等価である。

バンド幅を拡張した自己相関  $r'_{L0}(k)$  を使用して、レビンソン-ダービンアルゴリズム[Levinson, 1947]によりLPフィルタ係数  $a_k$  を得る。アルゴリズムは、以下の再帰的なステップにおいて実行される。

1. 反復番号  $i = 1$ ,  $a_0^{[0]} = 1.0$ , および  $E^{[0]} = r'_{L0}(0)$  とする

2.  $k_i = -\frac{1}{E^{[i-1]}} \left( r'_{L0}(i) + \sum_{j=1}^{i-1} a_k^{[i-1]} r'_{L0}(i-j) \right)$  を算出する

3.  $a_i^{[i]} = k_i$  とする。



4.  $a_j^{[i]} = a_j^{[i-1]} + k_i a_{i-j}^{[i-1]}$  for  $j = 1, \dots, i-1$  を算出する

5.  $E^{[i]} = (1 - k_i^2) E^{[i-1]}$  を算出する

6.  $i$  を1つインクリメントして、 $i$ が4になるまでステップ2に戻る

最終的な結果が以下で得られる。  $a_j = a_j^{[4]}, j=1, \dots, 4$

LP分析の結果は、以下の伝達関数を持つフィルタである。

$$A_0(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3} + a_4 z^{-4} \quad (7-23)$$

ここで、 $a_i, i=1, \dots, 4$  は、4次のLPフィルタの係数である。この関数は、以下のような重み係数 $\gamma_{P1}$ を使用して重み付けされる。

$$A_0(z / \gamma_{P1}) = 1 + \sum_{i=1}^4 \gamma_{P1}^i a_i z^{-i} \quad (7-24)$$

式(7-13)に式(7-23)を代入することで、ノイズフィードバック伝達関数は以下のように算出される。

$$F(z) = A_0(z / \gamma_{P1}) - 1 = \sum_{i=1}^4 \gamma_{P1}^i a_i z^{-i} \quad (7-25)$$

重み係数 $\gamma_{P1}$ は、一般的に0.92が設定されるが、ある特別な場合、すなわち、低いレベルの入力信号、または、高周波数に偏った信号の場合には変更される。聴覚フィルタを変更するには、2つの値に基づいて決定する。正規化係数 $\eta_{LB}$ （以下の式(7-26)）と、第1次の反射係数 $k_1$ である。

If  $\eta_{LB} \geq 16$

微小レベル信号の場合、聴覚フィルタを減衰する（7.3.2.2節に記載の通り）

else if  $k_1 > 0.9844$

特別な傾きをもつ入力信号の場合、聴覚フィルタを減衰する（7.3.2.3節に記載の通り）

else

$\gamma_{P1} = 0.92$  で式(7-25)の聴覚フィルタを使用する（減衰しない）

end

上記の判定を行うために、正規化係数 $\eta_{LB}$ を算出する。

$$\eta_{LB} = 30 - \left\lfloor \log_2 (r_{L0}(0)) \right\rfloor \quad (7-26)$$

ここで、 $r_{L0}(0)$ は、式(7-20)で算出される最初の自己相関係数である。第1次の反射係数 $k_1$ は、上述のレビンソンダービンアルゴリズムによって算出される。

### 7.3.2.2 微小レベル信号に対する聴覚フィルタの減衰

入力エネルギーが微小な場合、ノイズシェーピングは、G. 711の量子化ノイズを適切にマスクできない。これは、ノイズによる復号信号のパワー不整合を引き起こし、ノイズフィードバックループの飽和が生じる。飽和状態を避けるために微小なレベル信号に対しては、聴覚ノイズシェーピングフィルタは以下の手順で減衰される。正規化係数 $\eta_{LB}$ が以下の条件の場合

$$\eta_{LB} \geq 16 \quad (7-27)$$

微小レベル信号の減衰を行なう。この場合、減衰された式(7-25)の聴覚重みフィルタは以下となる。

$$F(z) = \sum_{i=1}^4 2^{-(i+\eta_{LB}-16)} a_i z^{-i} \quad (7-28)$$

微小レベルの入力信号に対して聴覚ノイズシェーピングフィルタを減衰させることにより、ノイズフィードバックループが、聴覚的に低いノイズフロアを持つようにするという恩恵を得ることなく客観的なノイズレベルを増大させてしまうというケースを避けることができる。また、符号化器と復号器のフィルタが不整合になる影響も抑えられる。

### 7.3.2.3 高域周波数に集中するエネルギーをもつ信号に対する聴覚フィルタの減衰

ある限定される場合において、信号のエネルギーは 4000Hz（低域のナイキスト周波数）近傍の単一信号周波数ピークに集中する。この特別な場合においては、フィルタが非常に発振し易くなり、ノイズシェーピングフィードバックは不安定となる、これにより、ノイズシェーピングループが安定状態に収束するまでの間は、いくつかのフレームで持続する可聴異音の原因となる。この問題を避けるため、高域周波数に集中するエネルギーをもつ信号が符号化器で検出される場合にはいつも、ノイズシェーピングフィードバックを減衰させる。

信号のスペクトル傾斜を決めるために、反射係数  $k_1$  を用いて、以下の条件を満足しなければならない。

$$k_1 > 0.9844 \quad (7-29)$$

この場合、以下の定義される重み係数  $\gamma_{p1}$  と共に、式(7-25)の聴覚重みフィルタが使用される。

$$\gamma_{p1} = 0.92\alpha_s \quad (7-30)$$

ここで、 $\alpha_s$  の減衰係数は、以下に与えられる  $k_1$  の関数である。

$$\alpha_s = 16 \cdot (1.047 - k_1) \quad (7-31)$$

### 7.3.2.4 変形入力信号の算出

変形信号  $s'_{LB}(n)$  は、エンベデッドPCM符号化器への入力信号として、入力信号  $s_{LB}(n)$ （QMF の出力である低域信号）から得られる。変形信号  $s'_{LB}(n)$  は、量子化ノイズ  $d_{L0}(n)$  の重み関数を入力信号  $s_{LB}(n)$  に加算することにより算出される。

$$s'_{LB}(n) = s_{LB}(n) + \sum_{i=1}^4 \gamma_{p1}^i a_i \cdot d_{L0}(n-i) + c_{Loff} \quad (7-32)$$

定数  $c_{Loff}$  は、既に定義されているコア符号化器の圧縮則に依存するオフセット値である。量子化誤差  $d_{L0}(n)$  は、レイヤ0のみを用いて計算される。すなわち、レイヤ1からの寄与分は考慮しない。従って、Z変換領域において、Figure 7-2/JT-G711.1 に示すフィードバックループは、入力信号  $s_{LB}(n)$  とレイヤ0合成  $\hat{s}_{L0}(n)$  の関係として以下で与えられる。

$$\hat{S}_{L0}(z) = S_{LB}(z) + \left( \frac{1}{A_0(z/\gamma_{p1})} \right) D_{L0}(z) \quad (7-33)$$

ここで、 $D_{L0}(z)$ は、レイヤ0の量子化器 $Q_{L0}$ からの量子化ノイズの $z$ 変換表現である。フィードバックループフィルタにおいて、 $A_0(z/\gamma_{p1})-1$ を使用することで1サンプル遅延が生じ、誤差サンプル $d_{L0}(n)$ は $s_{LB}(n)$ に加算する前に遅延されてフィルタ処理される。

### 7.3.3 デッドゾーン量子化器

微小のエネルギーの信号の品質をさらに改善するために、エンベデッド低域符号化器と復号器の代わりに、デッドゾーン量子化器を使用する。デッドゾーン量子化器は以下の条件を満足する場合にだけ有効になる。

$$\eta_{LB} \geq 16 \quad \text{and} \quad \begin{cases} s'_{LB}(n) \in [-7, 7] & \text{for } \mu\text{-law} \\ s'_{LB}(n) - c_{Loff} \in [-11, 11] & \text{for A-law } (c_{Loff}=8) \end{cases} \quad (7-34)$$

ここで、 $\eta_{LB}$ は7.3.2.1節の低レベル信号検出において算出された正規化係数である。上記の条件を満足する場合、エンベデッド低域量子化器 $Q_{L0}$ と $Q_{L1}$ 、および逆量子化 $Q_{L0}^{-1}$ は使用せずに、以下に示す量子化の方法を代わりに採用する。

式(7-34)の条件に見られるように、デッドゾーン量子化器は、微小レベル入力信号 $s'_{LB}(n)$ に対してのみ有効にされる。その区間はデッドゾーンと呼ばれ、この区間内においては、レイヤ0の局部復号信号 $\hat{s}_{L0}(n)$ はゼロに抑圧される。このデッドゾーン量子化器において、サンプル $s'_{LB}(n)$ は、以下に記述される式に従って量子化される。

#### 7.3.3.1 デッドゾーンにおける $\mu$ 則 符号化過程

$$\begin{aligned} s'_{LB\text{exp}}(n) &= 0 \\ s'_{LB\text{ref}}(n) &= \begin{cases} 0 & s'_{LB}(n) \in [-7, -2] \\ 2 & s'_{LB}(n) = -1 \\ 4 & s'_{LB}(n) \in [0, 1] \\ 8 & s'_{LB}(n) \in [2, 7] \end{cases} \\ \hat{s}_{L0}(n) &= 0 \\ I_{L0}(n) &= 0\text{xFF} \end{aligned}$$

Figure 7-3/JT-G711.1 に量子化結果を示す。x軸は、デッドゾーン量子化の入力値を表し、y軸は、対応する復号された出力値を表す（図に示す結果は、聴覚重みを考慮していない）

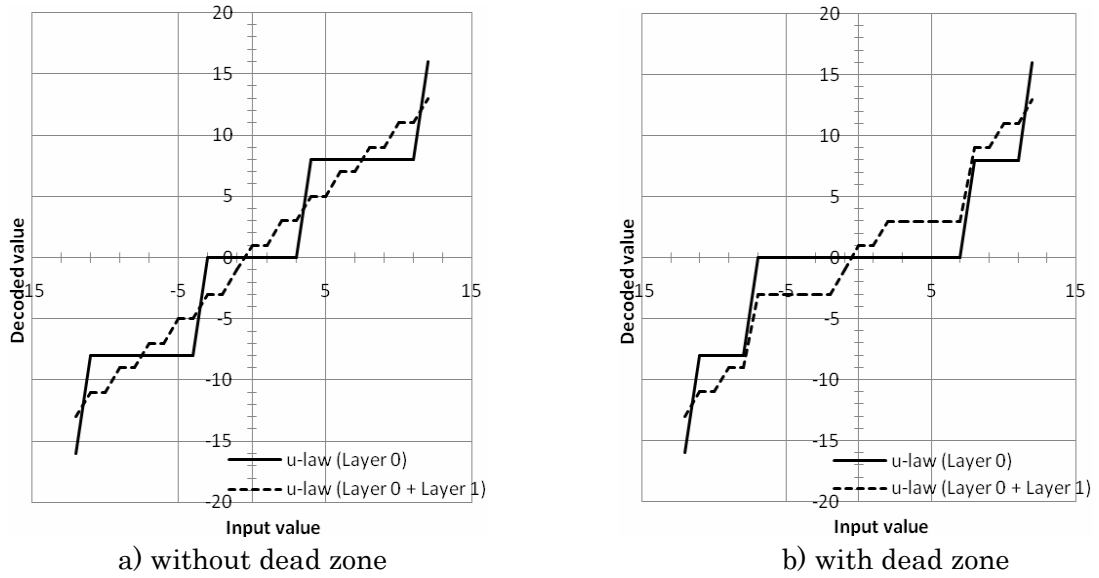


Figure 7 – 3 / JT-G711.1 Dead-zone quantizer characteristics ( $\mu$ -law case)

(ITU-T G.711.1)

### 7.3.3.2 デッドゾーンにおけるA則符号化過程

$$s'_{LBexp}(n) = 0$$

$$s'_{LBref}(n) = \begin{cases} 0 & s'_{LB}(n) - c_{Loff} \in [-11, -7] \\ s'_{LB}(n) / 2 & s'_{LB}(n) - c_{Loff} \in [-6, 6] \\ 7 & s'_{LB}(n) - c_{Loff} \in [7, 11] \end{cases}$$

$$\hat{s}_{L0}(n) = 0$$

$$I_{L0}(n) = 0xD5$$

Figure 7 – 4 / JT-G711.1 に量子化結果を示す。Figure 7 – 3 / JT-G711.1 と同様に、x 軸はデッドゾーン量子化の入力値を示し、y 軸は、対応する復号された出力値を示す。(図に示す結果は、聴覚重みを考慮していない)

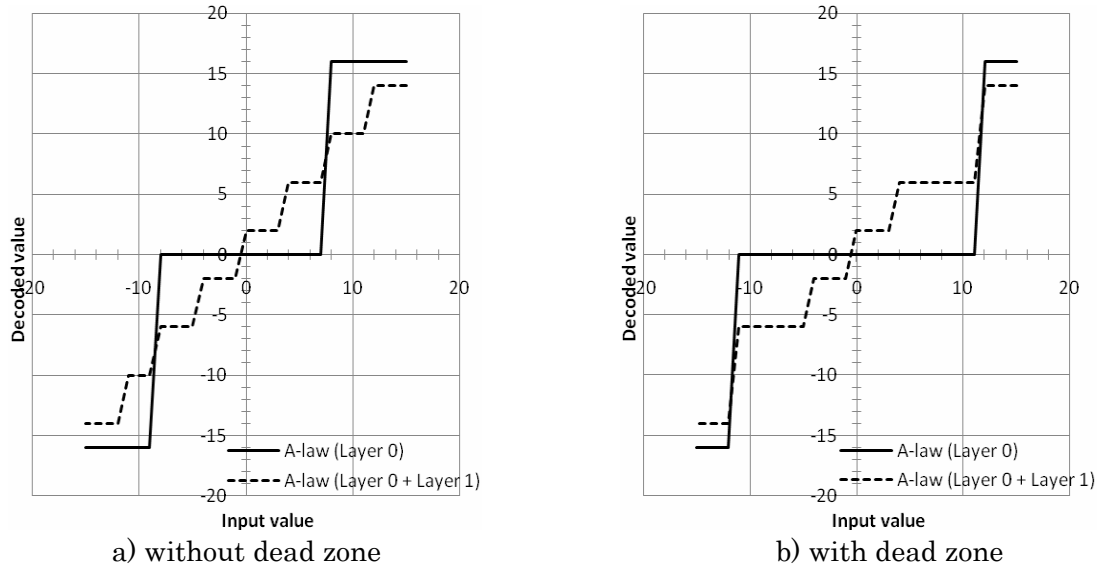


Figure 7-4/JT-G711.1 Dead-zone quantizer characteristics (A-law case)  
(ITU-T G.711.1)

### 7.3.4 精細信号の適応多重化（レイヤ 1）

低域拡張レイヤ（レイヤ 1）の量子化器は、コア PCM 符号化器で算出される精細信号  $s'_{LBref}(n)$ （7.3.1 節に記述の通り）を符号化する。精細信号  $s'_{LBref}(n)$  は、サンプルあたり 3 ビットの分解能をもつ。しかし、レイヤ 1 ビットストリームの利用可能なビット数が 16 kbit/s、すなわち、サンプルあたり 2 ビットであるため、ビットを削減するための適応的多重化を行う。この適応的な多重化は、より大きな指数値  $s'_{LBexp}(n)$  をもつサンプルにより多くのビットを割当て、各サンプルに 0 から 3 ビットの可変ビットを割り当てる。この適応的な多重化での符号化は、ビット割り当てテーブルの生成、および精細信号の多重化の 2 段階で行われる。この手順は両方の圧伸則で共通の手順である。

#### 7.3.4.1 ビット割り当てテーブルの生成

ビット割り当てテーブルは、指数値  $s'_{LBexp}(n)$  を使用して生成される。最初に、指数信号が、指数マップ  $M_{exp}(j, n)$  ( $j=0, \dots, 9$  および  $n=0, \dots, 39$ ) に展開される。指数マップは、 $10 \times 40$  のマトリックスで、精細信号  $s'_{LBref}(n)$  を表すための特定の指数インデックス  $j$  を使用したサンプルのインデックスを保存する。同じ指数インデックスを持ったサンプル数をカウントするために、配列  $N_{exp}(j)$  ( $j=0, \dots, 9$ ) が、0 に初期化される。各サンプル  $s'_{LBexp}(n)$  に対して、次のステップが行われる。

1.  $j = s'_{LBexp}(n) + i$  により 1 つのサンプル指数インデックスを算出する
2.  $M_{exp}(j, N_{exp}(j)) = n$  として指数マップを更新する
3. 指数インデックスに適合するサンプル数をインクリメントする

$$N_{exp}(j) = N_{exp}(j) + 1$$

そして、ビット割り当てテーブル  $B_A(n)$  ( $n=0, \dots, 39$ ) は、以下のステップにより算出される。

1.  $B_A(n)$  ( $n=0, \dots, 39$ ) の全て要素を 0 に初期化する。残りの利用可能なビット数  $b^{[0]} = 80$ 、指数インデックス  $j = 9$  と反復数  $i = 0$  を設定する。
2. 残りのビットバジェット  $b^{[i]}$  と現在の指数インデックス  $j$  におけるサンプル数  $N_{exp}(j)$  を比較して

小さい方を、 $q = \min[b^{[i]}, N_{\text{exp}}(j)]$ により、指数インデックスにおける利用可能なビット数  $q$  として設定する。

3. 指数インデックス  $j$  の全てのサンプルのインデックスに対して、現在の指数インデックスにおけるサンプルの割り当てビットを以下のようにインクリメントする。 $B_A(n) = B_A(n) + 1$ , for  $n = M_{\text{exp}}(j, k), k = 0, \dots, q - 1$
4.  $b^{[i+1]} = b^{[i]} - q$  として残りの利用可能なビット数を更新する。
5. 残りの利用可能なビット数がなくなった、すなわち、 $b^{[i+1]} = 0$  であるかどうかを確認する。無くなっていなければ、 $i$  を 1 だけインクリメントし、指数インデックをデクリメント ( $j = j - 1$ ) してステップ 2 へ進む。

#### 7.3.4.2 多重化

得られたビット割り当てテーブル  $B_A(n)$  ( $n=0, \dots, 39$ ) は、多重化される  $s'_{L\text{Bref}}(n)$  の最上位ビットの数を与える。精細符号  $I_{L1}(n)$  は、精細信号  $s'_{L\text{Bref}}(n)$  の MSB である  $B_A(n)$  から算出される。

$$I_{L1}(n) = \left\lfloor \frac{s'_{L\text{Bref}}(n)}{2^{(3-B_A(n))}} \right\rfloor, \quad n = 0, \dots, 39 \quad (7-35)$$

の 40 個の精細符号  $I_{L1}(n)$  ( $n=0, \dots, 39$ ) は、レイヤ 1 ビットストリーム  $I_{L1}$  において連続して多重化される。

#### 7.4 MDCT

8kHz サンプリングの高域信号  $s_{HB}(n)$  は MDCT 係数  $S_{HB}(k)$  に変換される。変換処理は、フレーム長 5ms、分析窓長 10ms で行われる。MDCT 領域の高域側スペクトル  $S_{HB}(k)$  は次のように定義される。

$$S_{HB}(k) = \frac{1}{80} \sum_{n=0}^{79} w_{TDAC}(n) \cos\left(\frac{\pi}{40}(n+20.5)(k+0.5)\right) s_{HB}(n), \quad k = 0, \dots, 39 \quad (7-36)$$

ここで  $w_{TDAC}(n)$  は分析窓である。

$$w_{TDAC}(n) = \sqrt{2} \sin\left(\frac{\pi}{80}(n+0.5)\right), \quad n = 0, \dots, 79. \quad (7-37)$$

MDCT 係数を算出する過程で、可変のダイナミックレンジを扱うために固定小数点 Q フォーマットを保持する変数が使われる。正規化パラメータは次式によって計算される。

$$\eta_{HB}^{TDAC1} = 14 - \left\lfloor \log_2 \left( \max_{n=0, \dots, 79} |s_{HB}(n)| \right) \right\rfloor \quad (7-38)$$

40 個の MDCT 係数  $S_{HB}(k)$  の実際の計算は次のように行われる。

##### 1. 事前正規化

最初に、正規化された信号  $s_{HB}^{\eta}(n)$  が計算される。

$$s_{HB}^{\eta}(n) = 2^{\eta_{HB}^{TDAC1}} s_{HB}(n), \quad n = 0, \dots, 79 \quad (7-39)$$

## 2. 窓掛けと折り返し

複素数  $z(n) = z_R(n) + jz_I(n)$  (ただし  $j = \sqrt{-1}$ ) の実部と虚部は、次式によって計算される。

$$\begin{aligned} z_R(n) &= w_{TDAC}(2n)s_{HB}^{\eta}(2n) - w_{TDAC}(39-2n)s_{HB}^{\eta}(39-2n) \\ z_I(n) &= w_{TDAC}(79-2n)s_{HB}^{\eta}(79-2n) + w_{TDAC}(40+2n)s_{HB}^{\eta}(40+2n) \end{aligned} \quad (7-40)$$

## 3. 事前複素乗算

複素数  $z(n)$  は  $z'(n)$  に変換される。

$$z'(n) = W_{80}^n \cdot z(n), \quad n = 0, \dots, 19 \quad (7-41)$$

ここで、 $W_N = e^{j\frac{2\pi}{N}}$  は 1 の N 乗根であり、 $W_{80}^n = \cos\left(\frac{2\pi n}{80}\right) + j\sin\left(\frac{2\pi n}{80}\right)$  となる。この事前複素乗算は次のように展開される。

$$\begin{aligned} z'_R(n) &= \cos\left(\frac{2\pi n}{80}\right)z_R(n) - \sin\left(\frac{2\pi n}{80}\right)z_I(n) \\ z'_I(n) &= \sin\left(\frac{2\pi n}{80}\right)z_R(n) + \cos\left(\frac{2\pi n}{80}\right)z_I(n) \end{aligned} \quad (7-42)$$

ここで、 $z'_R(k)$  と  $z'_I(k)$  は  $z'(k) = z'_R(k) + jz'_I(k)$  の実部と虚部である。

## 4. 逆複素 FFT

$z'(n)$  のスケーリングされた逆複素フーリエ変換は、係数  $Z'(k)$  を得るために計算される。

$$Z'(k) = \sum_{n=0}^{19} z'(n)W_{20}^{nk} = \sum_{n=0}^{19} z'(n)e^{j\frac{2\pi nk}{20}}, \quad k = 0, \dots, 19. \quad (7-43)$$

サイズ 20 のこの変換は、Good-Thomas FFT アルゴリズム[Blahut, 1985]によって実装される。

## 5. 事後複素乗算

計算された係数  $Z'(k)$  は、次式のように変換される。

$$Z(k) = \frac{1}{80} \left( (-1)^{k+1} W_8^{-1} W_{320}^{4k+1} \cdot Z'(k) \right), \quad k = 0, \dots, 19 \quad (7-44)$$

ここで、 $W_8^{-1} = \frac{1-j}{\sqrt{2}}$ 、 $W_{320}^{4k+1} = \cos\left(\frac{2\pi(4k+1)}{320}\right) + j\sin\left(\frac{2\pi(4k+1)}{320}\right)$

この事後複素乗算は次式のように展開される。

$$\begin{aligned}
Z_R(k) &= \frac{(-1)^{k+1}}{80\sqrt{2}} [\cos(\theta) + \sin(\theta)] Z'_R(k) - \frac{(-1)^{k+1}}{80\sqrt{2}} [\sin(\theta) - \cos(\theta)] Z'_I(k) \\
Z_I(k) &= \frac{(-1)^{k+1}}{80\sqrt{2}} [\sin(\theta) - \cos(\theta)] Z'_R(k) + \frac{(-1)^{k+1}}{80\sqrt{2}} [\cos(\theta) + \sin(\theta)] Z'_I(k)
\end{aligned} \tag{7-45}$$

ここで  $Z_R(k)$ 、 $Z_I(k)$  は  $Z(k) = Z_R(k) + jZ_I(k)$  の実部と虚部、 $\theta = \frac{2\pi(4k+1)}{320}$  である。

## 6. 並べ替え

MDC T 係数  $S_{HB}(k)$  は最初に次式によって与えられる。

$$\begin{cases} S_{HB}^{TDAC}(2k) = Z_I(k) \\ S_{HB}^{TDAC}(39-2k) = -Z_R(k) \end{cases}, \quad k = 0, \dots, 19 \tag{7-46}$$

## 7. 事後再正規化

連続した高域の量子化精度を向上させるため、もう一つの正規化パラメータが計算される。

$$\eta_{HB}^{TDAC2} = 14 - \left\lfloor \log_2 \left( \max_{k=0, \dots, 39} |S_{HB}^{TDAC}(k)| \right) \right\rfloor \tag{7-47}$$

また、MDC T スペクトルは最終的に次式によって正規化される。

$$S_{HB}(k) = 2^{\eta_{HB}^{TDAC2}} S_{HB}^{TDAC}(k), \quad k = 0, \dots, 39 \tag{7-48}$$

全体の正規化係数  $\eta_{HB} = \eta_{HB}^{TDAC1} + \eta_{HB}^{TDAC2}$  はゲイン量子化器に渡され、後述の 7.5.2 節で使われる。ここで、 $\eta_{HB}$  の値は次式のように展開されることができるとに注意されたい。

$$\eta_{HB} = 28 - \left( \left\lfloor \log_2 \left( \max_{n=0, \dots, 79} |s_{HB}(n)| \right) \right\rfloor + \left\lfloor \log_2 \left( \max_{k=0, \dots, 39} |S_{HB}^{TDAC}(k)| \right) \right\rfloor \right) \tag{7-49}$$

## 7.5 高域符号化(レイヤ 2)

高域のコーデックとして、インターリーブ共役構造ベクトル量子化 (CSVQ) を用いた MDC T による変換符号化が使われる。高域符号化器の詳細は Figure 7-5 / JT-G711.1 に示される。MDC T 係数  $S_{HB}(k)$  と各正規化パラメータ  $\eta_{HB}$  に変換された高域信号  $s_{HB}(n)$  には、周波数域の窓かけと重み付け  $w_{HB}(k)$  が適用される。重み付けされた MDC T 係数  $S_{HBw}(k)$  は、RMS ファクタ  $g_{HB}$  によって正規化される。正規化された係数  $\bar{S}_{HBw}(k)$  は、6 セットの 6 次サブベクトルに間引きされ、それらのサブベクトルは 2 チャンネルのベクトル量子化 (VQ) によって独立に量子化され、ベクトル形状インデックス  $I_{Hs0}(v)$ 、 $I_{Hs1}(v)$  と、それらの各極性インデックス  $I_{Hp0}(v)$ 、 $I_{Hp1}(v)$  が得られる。この RMS ファクタ  $g_{HB}$  は、正規化ファクタ  $\eta_{HB}$  と同様に、 $\mu$  則領域で均一にスカラ量子化され、ゲインインデックス  $I_{Hg}$  を生成する。それから全てのインデックスは多重化され、高域のビット列  $I_{L2}$  として送られる。高域コーデックのビット割り当ては Table 9 / JT-G711.1 に示される。



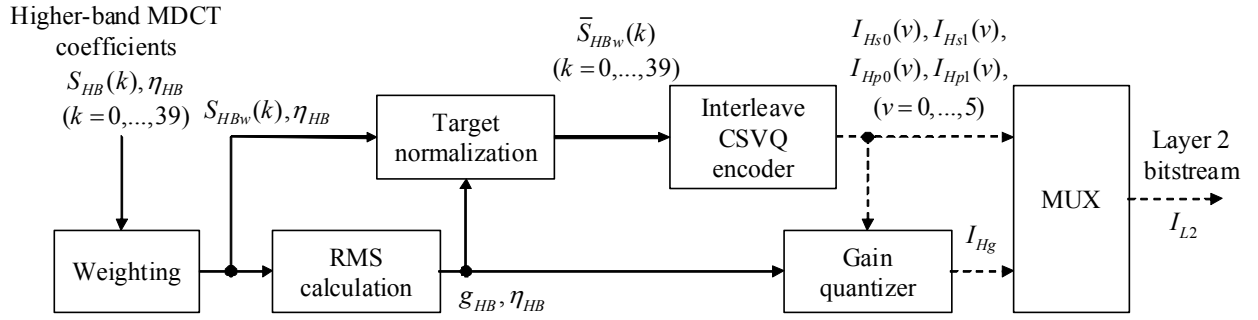


Figure 7 – 5 / JT-G711.1 - Block diagram of higher-band encoder  
(ITU-T G.711.1)

Table 9 / JT-G711.1 - Bit-allocation of Layer 2 sub-codec  
(ITU-T G.711.1)

Parameter	Bits per sub-vectors	Bits per 5-ms frame
MDCT coefficients (VQ)	5 + 5	60
Sign	1 + 1	12
Gain	-	8
Total	12	80
Bit-rate	16 kbit/s	

### 7.5.1 周波数重み付け

MDCT 係数  $S_{HB}(k)$  は周波数域で次のように重み付けされる。

$$S_{HBw}(k) = \begin{cases} 0, & (0 \leq k \leq 3) \\ w_{HB}(5) \sin\left(\frac{\pi}{24}(2(k-4)+1)\right) S_{HB}(k) & (4 \leq k \leq 9) \\ w_{HB}(4) S_{HB}(k) & (10 \leq k \leq 15) \\ w_{HB}(3) S_{HB}(k) & (16 \leq k \leq 21) \\ w_{HB}(2) S_{HB}(k) & (22 \leq k \leq 27) \\ w_{HB}(1) S_{HB}(k) & (28 \leq k \leq 33) \\ w_{HB}(0) S_{HB}(k) & (34 \leq k \leq 39) \end{cases} \quad (7-50)$$

ここで、 $w_{HB}(j)$  は周波数域重み付け係数である。その係数は Table 10 / JT-G711.1 によって与えられる。

Table 10 / JT-G711.1 - Frequency-domain weighting factors  
(ITU-T G.711.1)

$j$	0	1	2	3	4	5
$w_{HB}(j)$	1	1	4/3	5/3	2	1

### 7.5.2 RMS 演算とターゲット正規化

ゲイン算出部では、重み付けされたMDC T係数 $S_{HBw}(k)$ のRMS値 $g_{HB}$ が次式によって算出される。

$$g_{HB} = \sqrt{\frac{1}{36} \sum_{k=0}^{35} (S_{HBw}(k+4))^2} \quad (7-51)$$

MDC T係数 $S_{HBw}(k)$ は、 $\bar{S}_{HBw}(k)$ を得るためにさらに $g_{HB}$ によって次のように正規化される。

$$\bar{S}_{HBw}(k) = \frac{S_{HBw}(k)}{g_{HB} + \varepsilon_g} \quad (7-52)$$

ここで $\varepsilon_g$ は0除算を避けるために加算される値である。

$$\varepsilon_g = 32768 \cdot 2^{-(n_{HB}+15)} \quad (7-53)$$

### 7.5.3 インターリーブ共役構造ベクトル量子化

Figure 7-6/JT-G711.1 に、インターリーブCSVQのブロック図を示す。最初に、正規化されたMDC T係数 $\bar{S}_{HBw}(k)$ は、6セットの6次サブベクトル $S'_{HB}(v)$  ( $v=0, \dots, 5$ )を構成するために間引きされる。それから、各サブベクトルは2チャンネルのCSVQによって量子化される。ここで、コードベクトルは2つの異なるコードブックからの2つのサブベクトルの平均で表される。CSVQによって $v$ 番目のサブベクトル $S'_{HB}(v)$ を量子化するために、コードインデックス $I_{Hs0}(v)$ 、 $I_{Hs1}(v)$ とそれに関連付けられた極性(1 or -1)  $p_{H0}(v)$ 、 $p_{H1}(v)$ が、歪み $d_{HB}(I_{Hs0}(v), I_{Hs1}(v), p_{H0}(v), p_{H1}(v))$ を最小化するように次式で選ばれる。

$$d_{HB}(I_{Hs0}(v), I_{Hs1}(v), p_{H0}(v), p_{H1}(v)) = \sum_{j=0}^5 \left( S'_{HB}(v, j) - \frac{p_{H0}(v)C_{H0w}(I_{Hs0}(v), j) + p_{H1}(v)C_{H1w}(I_{Hs1}(v), j)}{2} \right)^2 \quad \begin{cases} k=0, \dots, 5, \\ j=0, \dots, 5 \end{cases} \quad (7-54)$$

ここで $C_{H0w}(I_{Hs0}(v))$ はサブコードブック0中のインデックス $I_{Hs0}(v)$ を持つコードベクトルであり、 $C_{H1w}(I_{Hs1}(v))$ はサブコードブック1中のインデックス $I_{Hs1}(v)$ を持つコードベクトルである。Figure 7-6/JT-G711.1 に示されるように、各サブコードブックを事前を選択する方法は、上述の歪みに従った最適なコードベクトルの組合せを選択するよりも前に導入される。予備選択と本選択の詳細は、それぞれ7.5.3.2節と7.5.3.3節で述べる。

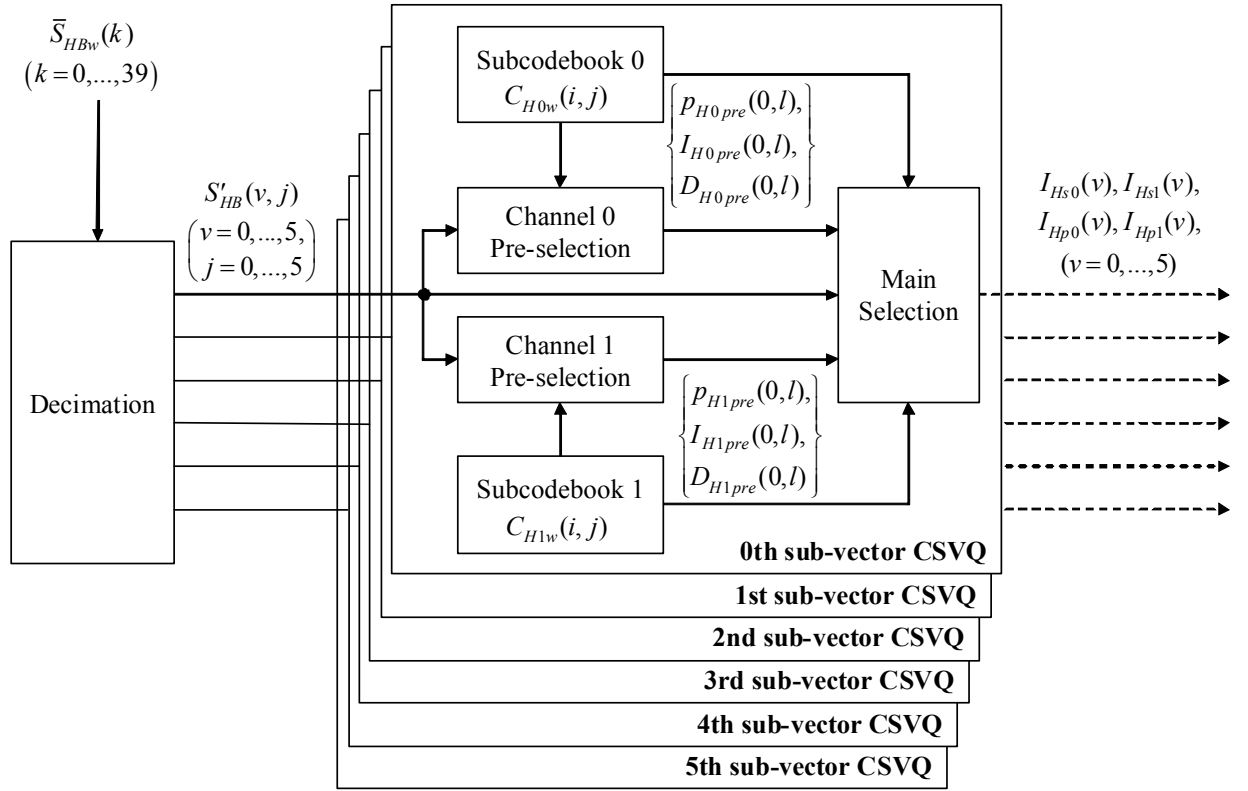


Figure 7 – 6 / JT-G711.1 - Block diagram of interleave CSVQ  
(ITU-T G.711.1)

### 7.5.3.1 間引き

重み付け、正規化されたMDCT係数 $\bar{S}_{HBw}(k)$ は、6セットの6次サブベクトルに間引きされる。

$$S'_{HB}(v, j) = \bar{S}_{HBw}(39 - v - 6j), \quad v = 0, \dots, 5, j = 0, \dots, 5 \quad (7-55)$$

### 7.5.3.2 予備選択

コードブック探索の演算量を削減するため、負荷の大きい主コード選択の前に演算量が軽減された予備選択が実行される。予備選択においては、各 $v$ 番目のサブベクトル $S'_{HB}(v)$ 毎に、8個の候補( $l = 0, \dots, 7$ )が、32のコードベクトルを含む各コードブックの中から選ばれる。ここでは、それぞれ $p_{H0pre}(v, l)$ 、 $p_{H1pre}(v, l)$ という極性のあるインデックス $I_{Hs0pre}(v, l)$ 、 $I_{Hs1pre}(v, l)$ を持つコードベクトル候補が選択される。

この予備選択の段階では、ターゲットサブベクトル $S'_{HB}(v)$ とコードブック0のコードベクトルの距離 $d_{0pre}(v, i_0)$ と、ターゲットサブベクトル $S'_{HB}(v)$ とコードブック1のコードベクトルの距離 $d_{1pre}(v, i_1)$ がそれぞれ32のコードベクトルインデックスとして計算される。

$$d_{0pre}(v, i_0) = \sum_{j=0}^5 \left( S'_{HB}(v, j) - p(v, i_0) \frac{C_{H0w}(i_0, j)}{2} \right)^2 \quad \left. \begin{array}{l} v = 0, \dots, 5 \\ i_0, i_1 = 0, \dots, 31 \end{array} \right\}, \quad (7-56)$$

$$d_{1pre}(v, i_1) = \sum_{j=0}^5 \left( S'_{HB}(v, j) - p(v, i_1) \frac{C_{H1w}(i_1, j)}{2} \right)^2$$

ここで、極性 $p(v, i_0)$ 、 $p(v, i_1)$ は、それぞれ $\sum_{j=0}^5 (S'_{HB}(v, j) \cdot C_{H0w}(i_0, j))$ 、 $\sum_{j=0}^5 (S'_{HB}(v, j) \cdot C_{H1w}(i_1, j))$

の極性(1 or -1)と同じになるように設定される。

ここでは、各コードブックは8つ( $l=0, \dots, 7$ )のセグメントに分割され、セグメント中の4つ( $q=0, \dots, 3$ )のコードベクトルの間で、 $v$  番目のターゲットサブベクトル  $S'_{HB}(v)$  との距離が最小となるものが  $l$  番目の予備選択された候補として選択される。つまり、コードブックインデックス  $i$ , ( $0 \leq i \leq 31$ ) と候補のインデックス  $l$  の関係は次式のようになる。

$$i = 4l + q, \quad \begin{cases} l = 0, \dots, 7, \\ q = 0, \dots, 3 \end{cases} \quad (7-57)$$

演算量削減のため、予備選択は次式のような距離尺度である  $d'_{0pre}$  と  $d'_{1pre}$  を最大化する候補を見つけるように実行される。

$$\begin{aligned} d'_{0pre}(v, l, q) &= 4|\varphi_0(v, l, q)| - \sigma_{H0w}^2(4l + q) \\ d'_{1pre}(v, l, q) &= 4|\varphi_1(v, l, q)| - \sigma_{H1w}^2(4l + q) \end{aligned} \quad \begin{cases} v = 0, \dots, 5, \\ l = 0, \dots, 7, \\ q = 0, \dots, 3 \end{cases} \quad (7-58)$$

ここで、

$$\begin{aligned} \varphi_0(v, l, q) &= \sum_{j=0}^5 S'_{HBw}(v, j) \cdot C_{H0w}(4l + q, j) \\ \varphi_1(v, l, q) &= \sum_{j=0}^5 S'_{HBw}(v, j) \cdot C_{H1w}(4l + q, j) \end{aligned}, \quad \begin{cases} v = 0, \dots, 5, \\ l = 0, \dots, 7, \\ q = 0, \dots, 3 \end{cases} \quad (7-59)$$

かつ、

$$\begin{aligned} \sigma_{H0w}^2(4l + q) &= \sum_{j=0}^5 (C_{H0w}(4l + q, j))^2 \\ \sigma_{H1w}^2(4l + q) &= \sum_{j=0}^5 (C_{H1w}(4l + q, j))^2 \end{aligned}, \quad \begin{cases} l = 0, \dots, 7, \\ q = 0, \dots, 3 \end{cases} \quad (7-60)$$

候補のパラメータセット  $\{p_{H0pre}(v, l), I_{Hs0pre}(v, l), D_{H0pre}(v, l)\}$  と  $\{p_{H1pre}(v, l), I_{Hs1pre}(v, l), D_{H1pre}(v, l)\}$  は次式のように選択される。

$$\begin{aligned} q_{Hs0pre}(v, l) &= \arg \max_q [d'_{0pre}(v, l, q)] \\ I_{Hs0pre}(v, l) &= 4l + q_{Hs0pre}(v, l) \\ p_{H0pre}(v, l) &= \text{sgn}[\varphi_0(v, l, q_{Hs0pre}(v, l))] \\ D_{H0pre}(v, l) &= d'_{0pre}(v, l, q_{Hs0pre}(v, l)) \end{aligned}, \quad \begin{cases} v = 0, \dots, 5, \\ l = 0, \dots, 7 \end{cases} \quad (7-61)$$

$$\begin{aligned}
q_{Hs1pre}(v, l) &= \arg \max_q [d'_{1pre}(v, l, q)] \\
I_{Hs1pre}(v, l) &= 4l + q_{Hs1pre}(v, l) \\
p_{H1pre}(v, l) &= \text{sgn} \left[ \varphi_1 \left( v, l, q_{Hs1pre}(v, l) \right) \right] \\
D_{H1pre}(v, l) &= d'_{1pre}(v, l, q_{Hs1pre}(v, l))
\end{aligned}$$

コードベクトル  $C_{H0w}(i)$  と  $C_{H1w}(i)$  は、実際には、 $C_{H0w}(i, j) = w_{HB}(j) \cdot C_{H0}(i, j)$  と  $C_{H1w}(i, j) = w_{HB}(j) \cdot C_{H1}(i, j)$  ( $j = 0, \dots, 5$ ) のように元のコードベクトルに重み付けされたものであることを注意すべきである。重みは一定であるため、コードベクトルは演算量削減のため重み付けされた形式で保持されている。 $i$  番目の重み付けされたコードベクトルのパワーである  $\sigma_{H0w}^2(i)$ 、 $\sigma_{H1w}^2(i)$  も同様に補助的なコードブックとして保持されている。

### 7.5.3.3 本選択

サブベクトル毎に、式(7-54)で定義された距離関数を最小化する組合せを選択するために、2つの予備選択されたパラメータセット  $\{p_{H0pre}(v, l), I_{Hs0pre}(v, l), D_{H0pre}(v, l)\}$ 、及び  $\{p_{H1pre}(v, l), I_{Hs1pre}(v, l), D_{H1pre}(v, l)\}$  の全ての組合せについて、網羅的な探索が実施される。定数項を無視することによって、主探索手順は次式の歪み尺度を最小化するように簡略化される。

$$\begin{aligned}
&d'_{HB}(v, l_0, l_1) \\
&= 2p_{H0pre}(v, l_0)p_{H1pre}(v, l_1) \cdot R_{H0H1}(I_{Hs0pre}(v, l_0), I_{Hs1pre}(v, l_1)) - D_{Hs0pre}(v, l_0) - D_{Hs1pre}(v, l_1) \quad (7-62) \\
&\qquad\qquad\qquad \left. \begin{array}{l} l_0 = 0, \dots, 7, \\ l_1 = 0, \dots, 7 \end{array} \right\}
\end{aligned}$$

ここで、 $R_{H0H1}(i_0, i_1)$  は次式によって与えられる。

$$R_{H0H1}(i_0, i_1) = \sum_{j=0}^5 C_{H0w}(i_0, j) C_{H1w}(i_1, j) \quad (7-63)$$

前述された  $\sigma_{H0w}^2(i)$ 、 $\sigma_{H1w}^2(i)$  のように、内積項  $R_{H0H1}(i_0, i_1)$  も同様に補助的なコードブックとして与えられる。最小距離  $d'_{HB}$  を与える組合せ  $(I_{Hs0}, I_{Hs1})$  は次のように選ばれる。

$$\begin{aligned}
I_{Hs0}(v) &= \arg \max_{I_{Hs0pre}(v, l_0)} [d'_{HB}(v, l_0, l_1)] \\
I_{Hs1}(v) &= \arg \max_{I_{Hs1pre}(v, l_1)} [d'_{HB}(v, l_0, l_1)] \quad (7-64) \\
&\qquad\qquad\qquad \left. \begin{array}{l} v = 0, \dots, 5, \\ l_0 = 0, \dots, 7, \\ l_1 = 0, \dots, 7 \end{array} \right\}
\end{aligned}$$

また、極性インデックスは次式によって得られる。

$$I_{Hp0}(v) = \begin{cases} 0 & \text{if } p_{H0pre}(v, \arg \max_{l_0} [d'_{HB}(v, l_0, l_1)]) = 1 \\ 1 & \text{otherwise} \end{cases} \quad \left\{ \begin{array}{l} v = 0, \dots, 5, \\ l_0 = 0, \dots, 7, \\ l_1 = 0, \dots, 7 \end{array} \right\} \quad (7-65)$$

$$I_{Hp1}(v) = \begin{cases} 0 & \text{if } p_{H1pre}(v, \arg \max_{l_1} [d'_{HB}(v, l_0, l_1)]) = 1 \\ 1 & \text{otherwise} \end{cases}$$

ゲインスカラ量子化（後述する）の前に、サブベクトルは次式によって局部復号される。

$$\hat{S}'_{HBw}(v, j) = \frac{1}{2} (p_{H0}(v) \cdot C_{H0w}(I_{Hs0}(v), j) + p_{H1}(v) \cdot C_{H1w}(I_{Hs1}(v), j)) \quad \left\{ \begin{array}{l} v = 0, \dots, 5, \\ j = 0, \dots, 5 \end{array} \right\} \quad (7-66)$$

ここで、極性は次式で与えられる。

$$\begin{aligned} p_{H0}(v) &= 1 && \text{if } I_{Hp0}(v) = 0 \\ &= -1 && \text{otherwise} \\ p_{H1}(v) &= 1 && \text{if } I_{Hp1}(v) = 0 \\ &= -1 && \text{otherwise} \end{aligned} \quad v = 0, \dots, 5 \quad (7-67)$$

#### 7.5.4 ゲインスカラ量子化部

ゲインスカラ量子化器においては、調整されたゲインパラメータ  $\tilde{g}_{HB}$  が、局部復号されたサブベクトル  $\hat{S}'_{HBw}(v, j)$  を使って次式のように算出される。

$$\tilde{g}_{HB} = \frac{\sum_{v=0}^5 \sum_{j=0}^5 S'_{HBw}(v, j) \hat{S}'_{HBw}(v, j)}{\sum_{v=0}^5 \sum_{j=0}^5 (\hat{S}'_{HBw}(v, j))^2} g_{HB} \quad (7-68)$$

ここで、 $g_{HB}$  は前に 7.5.3 節で算出された入力 MDC T 係数の RMS 値である。

調整ゲイン  $\tilde{g}_{HB}$  は  $g_{Hscaled} = 4\sqrt{10}\tilde{g}_{HB}$  としてスケーリングされる。係数  $4\sqrt{10}$  は値を 0 から 32767 の間に収めるために使われる。量子化の前に、スケーリングされたゲインは、次式のように小さい範囲は例外として  $\mu$  則領域の値に変換され、スカラ量子化される。

$$g_{Hlog} = \begin{cases} g_{Hscaled}, & \text{if } 0 \leq g_{Hscaled} < 4 \\ \frac{32768 \log_{10} \left( 1 + \frac{255 g_{Hscaled}}{32768} \right)}{\log_{10} 256}, & \text{if } 4 \leq g_{Hscaled} \leq 32767 \end{cases} \quad (7-69)$$

ここで、 $g_{Hlog}$  は  $\mu$  則領域の RMS 値である。この  $\mu$  則量子化は次のように実行される。

$$I_{Hg} = \begin{cases} \lfloor g_{Hscaled} \rfloor & \text{if } 0 \leq g_{Hscaled} < 4 \\ 2^5 \cdot e_g + \left\lfloor \frac{1}{4} \left( \left\lfloor 2^{-e_g} (g_{Hscaled} + 130) \right\rfloor - 128 \right) \right\rfloor + 3 & \text{if } 4 \leq g_{Hscaled} < 30590 \\ 255 & \text{if } 30590 \leq g_{Hscaled} \end{cases} \quad (7-70)$$

ここで、 $e_g = \lfloor \log_2(g_{Hscaled} + 130) \rfloor - 7$  は  $g_{scaled}$  の指数値であり、 $I_{Hg}$  はMDC T信号のゲインインデックスである。

## 8. 復号器の機能記述

### 8.1 エンベデッド低域PCM復号器

エンベデッドPCM復号器は、レイヤ0と1の復号器に実装されており、聴覚フィルタ演算部と聴覚ノイズシェーピングフィルタはFigure 8-1/JT-G711.1に示される通りである。

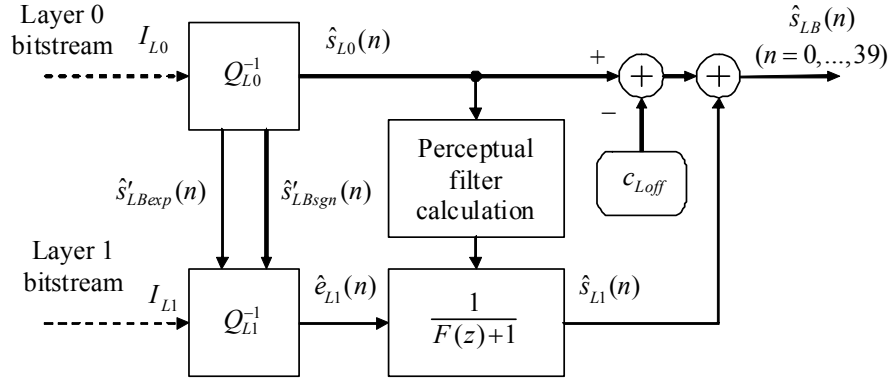


Figure 8-1/JT-G711.1 - Lower-band embedded PCM decoder  
(ITU-T G.711.1)

レイヤ0のビット列  $I_{L0}$  から、レイヤ0復号器( $Q_{L0}^{-1}$ )は8-bit PCM G. 711と同様の方法で  $\hat{s}_{L0}(n)$  を生成する。レイヤ1ビット列が利用できない場合は、この信号が低域復号器の出力となる。つまり次式となる。

$$\hat{s}_{LB}(n) = \hat{s}_{L0}(n) - c_{Loff} \quad (8-1)$$

逆に、レイヤ1のビット列  $I_{L1}$  が利用できる場合は、 $Q_{L1}^{-1}$ 、 $\hat{e}_{L1}(n)$  の出力信号は、レイヤ1の復号信号  $\hat{s}_{L1}(n)$  を得るために、聴覚重み付けフィルタによって整形される。この聴覚重み付けフィルタの伝達関数は、 $1/(F(z)+1) = 1/A_0(z/\gamma_{p1})$  であり、これによってレイヤ0の符号化と同様のノイズシェーピングが実現される。そのシェーピング処理は次式となる。

$$\hat{s}_{L1}(n) = \hat{e}_{L1}(n) - \sum_{i=1}^4 a_i \gamma_{p1}^i \hat{s}_{L1}(n-i) \quad (8-2)$$

ここで、聴覚フィルタ  $1/A_0(z/\gamma_{p1})$  は符号化器で使われるものと同じであり、レイヤ0 (7.3.2節を参照) の復号信号を使って計算されることに注意されたい。また同様に、符号化器と同じ減衰の規則が適用されること

にも注意されたい (7.3.2.2 節、7.3.2.3 節を参照)。レイヤ 1 のビット列が復号される時、出力信号は基本的に二つの復号信号  $\hat{s}_{L0}(n)$ 、 $\hat{s}_{L1}(n)$  の和として計算される。

$$\hat{s}_{LB}(n) = \hat{s}_{L0}(n) - c_{Loff} + \hat{s}_{L1}(n) \quad (8-3)$$

ここで  $c_{Loff}$  は式(7-12)で与えられる圧伸則に依存したオフセット値である。 $I_{L0}$  が失われる場合、これはフレーム消失を意味し、全体の低域復号処理、つまりレイヤ 0、1、は実行されずに、8.4 節で述べられるパケット消失補償が代わりに低域信号を生成する。

### 8.1.1 G. 711 (レイヤ 0) に基づくコア PCM 復号器

コアレイヤ (レイヤ 0) は各フレームのサンプル毎に G. 711 と互換のあるインデックス  $I_{L0}(n)$  から構成される。復号器においてレイヤ 0 のみが利用可能である場合、選択された G. 711 の圧伸則に従って次のように復元処理が実行される。符号化器 (7.3.1 節) と同様、 $s$ 、 $e$ 、及び  $m$  は、それぞれ、極性、指数、仮数を意味する。また、信号の指数部は、極性  $\hat{s}'_{LBsgn}(n) = s$  と一緒に  $\hat{s}'_{LBexp}(n) = e$  として保存される。これらは両方とも、精細信号  $\hat{e}_{L1}(n)$  を得るためのビット割り当てテーブルを算出するために使われる。これは 8.1.2 節で後述される。

#### 8.1.1.1 $\mu$ 則復号過程

Given  $s = I_{L0}(n) \otimes 0x80$  and  $y = (I_{L0}(n) \oplus 0x7F) \otimes 0x7F$ ,

$$e = \left\lfloor \frac{y}{2^4} \right\rfloor,$$

$$m = y \otimes 0x0F$$

$$\hat{s}_{L0}(n) = \begin{cases} 2^e \cdot (2^3 m + 128 + 4) - 132 & \text{if } s = 0x80 \\ -(2^e \cdot (2^3 m + 128 + 4) - 132) & \text{if } s = 0 \end{cases}$$

$$\hat{s}'_{LBexp}(n) = e,$$

$$\hat{s}'_{LBsgn}(n) = s$$

#### 8.1.1.2 A 則復号過程

Given  $s = I_{L0}(n) \otimes 0x80$  and  $y = (I_{L0}(n) \oplus 0x55) \otimes 0x7F$ ,

$$e = \left\lfloor \frac{y}{2^4} \right\rfloor,$$

$$m = y \otimes 0x0F$$

if  $e > 0$ ,

$$\hat{s}_{L0}(n) = \begin{cases} 2^{e-1} \cdot (2^4 m + 8 + 256) & \text{if } s = 0x80 \\ -(2^{e-1} \cdot (2^4 m + 8 + 256)) & \text{if } s = 0 \end{cases}$$

else

$$\hat{s}_{L0}(n) = \begin{cases} 2^4 m + 8 & \text{if } s = 0x80 \\ -(2^4 m + 8) & \text{if } s = 0 \end{cases}$$



$$\begin{aligned}\hat{s}'_{LB\text{exp}}(n) &= e, \\ \hat{s}'_{LB\text{sgn}}(n) &= s\end{aligned}$$

### 8.1.2 低域拡張復号器（レイヤ1）

精細信号を生成するため、指数値  $\hat{s}'_{LB\text{exp}}(n)$  を使って、まず 7.3.4.1 節記載と同様の手順でビット割り当てテーブル  $B_A(n)$  が再構成される。このビット割り当てテーブルは、符号器で計算されたものと同一である。精細信号のための符号  $\hat{s}'_{LB\text{ref}}(n)$  は 0 に初期化され、そして、得られた  $B_A(n)$  で、レイヤ1 ビットストリーム  $I_{L1}$  から MSB 側から分離される。

$$\hat{s}'_{LB\text{ref}}(n) = 2^{4-B_A(n)} I_{L1}(n), \quad n = 0, \dots, 39. \quad (8-4)$$

その後、再構成された  $B_A(n)$ 、 $\hat{s}'_{LB\text{ref}}(n)$  およびレイヤ0 復号器出力  $\hat{s}'_{LB\text{sgn}}(n)$ 、 $\hat{s}'_{LB\text{exp}}(n)$  を用いて、拡張された信号  $\hat{e}_{L1}(n)$  が、選択された G.711 圧縮則に従って以下のように計算される。

#### 8.1.2.1 $\mu$ 則の場合

$s = \hat{s}'_{LB\text{sgn}}(n)$ 、 $e = \hat{s}'_{LB\text{exp}}(n)$ 、 $r = \hat{s}'_{LB\text{ref}}(n)$ 、 $b = B_A(n)$  とし、以下のように計算する。

$$m = r - 8 + \left\lfloor \frac{8}{2^b} \right\rfloor$$

もし、 $e > 0$  ならば、

$$\hat{e}_{L1}(n) = \begin{cases} 2^{e-1} m & \text{if } s = 0x80 \\ -2^{e-1} m & \text{if } s = 0 \end{cases}$$

そうでないときは、

$$\hat{e}_{L1}(n) = \begin{cases} \left\lfloor \frac{m}{2} \right\rfloor & \text{if } s = 0x80 \\ -\left\lfloor \frac{m}{2} \right\rfloor & \text{if } s = 0 \end{cases}$$

#### 8.1.2.2 A則の場合

$s = \hat{s}'_{LB\text{sgn}}(n)$ 、 $e = \hat{s}'_{LB\text{exp}}(n)$ 、 $r = \hat{s}'_{LB\text{ref}}(n)$ 、 $b = B_A(n)$  とし、以下のように計算する。

$$m = r - 8 + \left\lfloor \frac{8}{2^b} \right\rfloor$$

もし、 $e > 0$  ならば、

$$\hat{e}_{L1}(n) = \begin{cases} 2^{e-1} m & \text{if } s = 0x80 \\ -2^{e-1} m & \text{if } s = 0 \end{cases}$$

そうでないときは、

$$\hat{e}_{L1}(n) = \begin{cases} m & \text{if } s = 0x80 \\ -m & \text{if } s = 0 \end{cases}$$

レイヤ1 ビットストリーム  $I_{L1}$  の復号の特別な場合があることに注意が必要である。ビットストリーム  $I_{L1}$  の全てのビットが 0 のとき、上記のアルゴリズムは実行されず、拡張信号  $\hat{e}_{L1}(n)$  はゼロにセットされる。

$$\hat{e}_{L1}(n) = 0, \quad n = 0, \dots, 39. \quad (8-5)$$

## 8.2 高域復号器（レイヤ2）

利得正規化 MDCT 係数は、ベクトル逆量子化器により受信インデックスから逆量子化される。出力された MDCT 係数は、復号されたゲインの乗算と逆重み付けにより再構成される。高域復号器のブロック図を Figure 8-2/JT-G711.1 に示す。

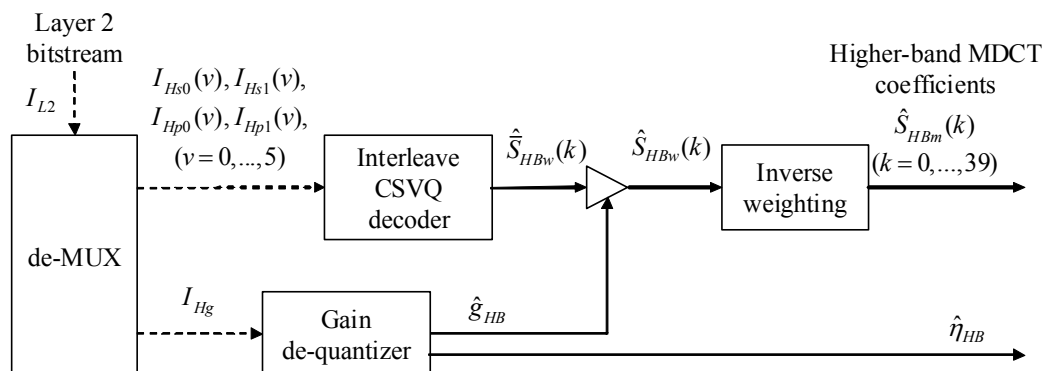


Figure 8-2/JT-G711.1 Higher-band MDCT decoder  
(ITU-T G.711.1)

### 8.2.1 インターリーブベクトル逆量子化部

#### 8.2.1.1 ベクトル逆量子化器

サブベクトル  $\hat{S}'_{HBw}(v, j)$  の逆量子化は、7.5.3.3 節記載のものと同一である。ここで、コードブックベクトルの極性  $p_{H0}(v)$  と  $p_{H1}(v)$  は、式(7-67)を用いて得られる。そして、復号されたサブベクトル  $\hat{S}'_{HBw}(v, j)$  は、式(7-66)で計算される。

#### 8.2.1.2 インターリーブ

復号されたサブベクトル  $\hat{S}'_{HBw}(v, j)$  は、一連の重み付け MDCT 係数  $\hat{S}_{HBw}(k)$  を形成するために以下のようにインターリーブされる。

$$\hat{S}_{HBw}(k) = \begin{cases} \hat{S}'_{HBw}\left(39-k-6\left\lfloor\frac{39-k}{6}\right\rfloor, \left\lfloor\frac{39-k}{6}\right\rfloor\right) & \text{for } 4 \leq k \leq 39 \\ 0 & \text{for } 0 \leq k \leq 3 \end{cases} \quad (8-6)$$

### 8.2.2 利得スカラ逆量子化

利得スカラ逆量子化器において、復号された利得  $\hat{g}_{HB}$  は利得インデックス  $I_{Hg}$  から以下のように逆量子化される。

$$\hat{g}_{HB} = \begin{cases} I_{Hg}, & (0 \leq I_{Hg} < 4) \\ 2^{\lfloor 2^{-5}(I_{Hg}-3) \rfloor} \left( \left\lfloor 2^{-2}(I_{Hg}-3) \otimes 0x1F \right\rfloor - 130 \right) - 130, & (4 \leq I_{Hg} \leq 255) \end{cases} \quad (8-7)$$

これは、8 ビット  $\mu$  則伸長と同等である。そして、逆量子化された重み付け MDCT 係数  $\hat{S}_{HBw}(k)$  は、 $\hat{S}'_{HBw}(k)$  を得るために、以下のように、復元された  $\hat{g}_{HB}$  (7.5.4 節参照) が乗ぜられる。

$$\hat{S}_{HBw}(k) = \frac{\hat{g}_{HB}}{4\sqrt{10}} \hat{S}_{HBw}(k), \quad k = 0, \dots, 39. \quad (8-8)$$

### 8.2.3 逆重み付け

再生された MDCT 係数  $\hat{S}_{HBw}(k)$  は重み付けされているので、重み付け係数  $w_{HB}(k)$  (7.5.1 節参照) の逆数を  $\hat{S}_{HBw}(k)$  に乗じることにより、重み付けされていない MDCT 係数  $\hat{S}_{HBm}(k)$  が以下のように計算される。

$$\hat{S}_{HBm}(k) = \begin{cases} 0, & (0 \leq k \leq 3) \\ \frac{1}{w_{HB}(5)} \sin\left(\frac{\pi}{24}(2(k-4)+1)\right) \hat{S}_{HBw}(k) & (4 \leq k \leq 9) \\ \frac{1}{w_{HB}(4)} \hat{S}_{HBw}(k) & (10 \leq k \leq 15) \\ \frac{1}{w_{HB}(3)} \hat{S}_{HBw}(k) & (16 \leq k \leq 21) \\ \frac{1}{w_{HB}(2)} \hat{S}_{HBw}(k) & (22 \leq k \leq 27) \\ \frac{1}{w_{HB}(1)} \hat{S}_{HBw}(k) & (28 \leq k \leq 33) \\ \frac{1}{w_{HB}(0)} \hat{S}_{HBw}(k) & (34 \leq k \leq 39) \end{cases}. \quad (8-9)$$

サンプル  $4 \leq k \leq 9$  に対しては、エンコーダと同じ  $\sin()$  関数が乗ぜられることに注意が必要であり、これは、入力信号の減衰結果が  $\sin^2()$  になる効果を有する。

### 8.3 逆 MDCT と オーバラップ 加算

再構成された高域 MDCT スペクトル  $\hat{S}_{HBm}(k)$  は、以下の逆 MDCT 変換により時間領域に変換される。

$$\hat{S}_{HB\_OLA}^{(m)}(n) = \sqrt{2} \sum_{k=0}^{39} \cos\left(\frac{\pi}{40}(k+0.5)(n+20.5)\right) \hat{S}_{HBm}(k), \quad (8-10)$$

ここで  $\hat{S}_{HB\_OLA}^{(m)}(n)$  は、オーバラップ加算 (OLA) 前の、現在の  $m$  番目のフレームの中間逆変換信号である。40 個の係数  $\hat{S}_{HB\_OLA}^{(m)}(n)$  の実際の計算は以下のようになされる。

#### 1. 事前複素乗算

実信号  $\hat{S}_{HBm}(k)$  は、以下のように展開される事前複素乗算により、 $V(k)$ ,  $k = 0, \dots, 19$  に修正される。

$$\begin{aligned} V_R(k) &= -(-1)^{k+1} \sqrt{2} [\cos(\theta) + \sin(\theta)] \hat{S}_{HBm}(39-2k) + (-1)^{k+1} \sqrt{2} [\sin(\theta) - \cos(\theta)] \hat{S}_{HBm}(2k) \\ V_I(k) &= (-1)^{k+1} \sqrt{2} [\sin(\theta) - \cos(\theta)] \hat{S}_{HBm}(39-2k) + (-1)^{k+1} \sqrt{2} [\cos(\theta) + \sin(\theta)] \hat{S}_{HBm}(2k) \end{aligned}, \quad (8-11)$$

ここで、 $V_R(k)$  と  $V_I(k)$  は、 $V(k) = V_R(k) + jV_I(k)$  の実部と虚部であり、 $\theta = \frac{2\pi(4k+1)}{320}$  である。

#### 2. 複素 FFT

係数  $v(n)$  を得るために、 $V(k)$  の複素 FFT が計算される。

$$v(n) = \sum_{k=0}^{19} V(k) \cdot e^{-j \frac{2\pi nk}{20}}, \quad k = 0, \dots, 19 \quad (8-12)$$

サイズ 20 の複素 FFT は Good-Thomas アルゴリズムを用いて実行される。

### 3. 複素事後乗算

複素信号  $v(n)$  は以下のように展開される複素事後乗算により、 $v'(n)$  に修正される。

$$\begin{aligned} v'_R(n) &= \cos\left(\frac{2\pi n}{80}\right) v_R(n) + \sin\left(\frac{2\pi n}{80}\right) v_I(n) \\ v'_I(n) &= -\sin\left(\frac{2\pi n}{80}\right) v_R(n) + \cos\left(\frac{2\pi n}{80}\right) v_I(n) \end{aligned}, \quad (8-13)$$

ここで、 $v(n) = v_R(n) + jv_I(n)$ 、 $v'(n) = v'_R(n) + jv'_I(n)$  である。

### 4. 並べ替え

時間領域信号  $\hat{s}_{HB\_OLA}^{(m)}(n)$  は以下のように与えられる。

$$\begin{cases} \hat{s}_{HB\_OLA}^{(m)}(n) = v_R(n) \\ \hat{s}_{HB\_OLA}^{(m)}(39-n) = -v_R(n) \\ \hat{s}_{HB\_OLA}^{(m+1)}(n) = v_I(n) \\ \hat{s}_{HB\_OLA}^{(m+1)}(39-n) = v_I(n) \end{cases} \quad n = 0, \dots, 19. \quad (8-14)$$

そして、OLA が以下のように実行される。

$$\hat{s}_{HB}(n) = \frac{w_{TDAC}(n) \hat{s}_{HB\_OLA}^{(m)}(n)}{2^{\hat{\eta}_{HB}}} + \frac{w_{TDAC}(39-n) \hat{s}_{HB\_OLA}^{(m-1)}(n)}{2^{\hat{\eta}_{HB}^{prev}}} \quad n = 0, \dots, 39 \quad (8-15)$$

ここで、 $\hat{\eta}_{HB}$  と  $\hat{\eta}_{HB}^{prev}$  は、それぞれ現フレームと過去フレームの復号された調整パラメータであり、 $w_{TDAC}(n)$  は分析窓と同一の合成 OLA 窓である。そして、 $\hat{s}_{HB\_OLA}^{(m-1)}(n)$  は OLA 前の過去の (m-1) 番目のフレームの中間逆変換信号の後半部であり、OLA 後に以下のように更新される。

$$\hat{s}_{HB\_OLA}^{(m-1)}(n) = \hat{s}_{HB\_OLA}^{(m+1)}(40+n) \quad n = 0, \dots, 39 \quad (8-16)$$

フレーム消失もしくはパケットロスのようなバッドフレームの場合は、上記 OLA は 8.5 節記載の高域 FERC 処理で置き換えられることに注意が必要である。

## 8.4 低域 FERC

フレーム消失時は、低域信号は低域 FERC により再構成される。Figure 8-3/JT-G711.1 に、FERC アルゴリズムの概略ブロック図を示す。フレーム消失時は、復号器は過去に復号された低域信号の分析を行い、消失フレームの失われた信号を、ピッチ反復に基づく線形予測符号化 (LPC) と適応ミュートを用いて外挿する (8.4.1

節および 8.4.2 節)。正常フレームが受信されたときは、直前の消失フレームで置き換えられた外挿信号は正しく復号された信号で再同期され、復号信号 (8.4.3 節) でクロスフェードされる。クロスフェードの前に、外挿信号に対して動的なエネルギー調整が実行される (8.4.4 節)。

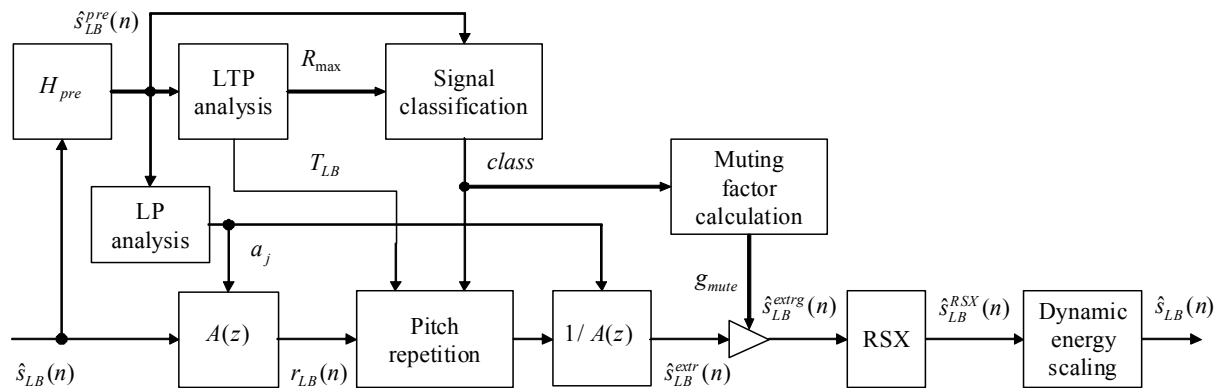


Figure 8 – 3 / JT-G711.1 Block diagram of lower-band extrapolation of erased frame Higher-band MDCT decoder (ITU-T G.711.1)

有効フレームの後の最初の消失フレームにおいて、処理量の最悪ケースが発生することに注意が必要である。処理量の最悪値を低減するために、最初の消失フレームに対する FERC 手順は、その消失フレームとその次のフレームに分けて処理される。これは、高域の先読みにより一フレーム分遅延が発生するため可能である。最初のステップは、最初の消失フレームで実行される前処理ステップである。このステージは分析 (LPC および長期予測分析) の一部を実行するのみであり、失われたサンプルの生成はしない。次に続くフレーム (消失フレームか否かにかかわらず) で実行される 2 番目のステージは、分析 (クラス分け) を完了し、最初の消失フレームに対する失われたサンプルを生成する。

#### 8.4.1 消失フレームの外挿：グッドフレーム後の消失フレームの場合

最初の消失フレームの低域外挿は、過去の有効な復号低域信号  $\hat{s}_{LB}(n)$ 、 $n < 0$  の分析と、それに続く信号  $\hat{s}_{LB}^{PLC}(n)$ 、 $n = 0, \dots, 39$  の合成からなる。低域信号の消失した 5ms フレームは  $\hat{s}_{LB}(n)$  で表記される。過去の信号  $\hat{s}_{LB}(n)$ 、 $n = -295, \dots, -1$  は、以下のように分割される 295 サンプル長のバッファを用いてバッファリングされる。

- FERC アルゴリズムで使われる最大ピッチ遅延の 2 倍 (2x144) に相当する 288 サンプル
- ピッチ揺らぎのための 1 サンプル
- LPC メモリのために使われる 6 サンプル

バッファ長は、低域再構成の最後の 2 ピッチ周期を保存するための余裕を持つ。フレーム消失の場合、前処理信号  $\hat{s}_{LB}^{pre}(n)$  を得るために、過去の信号  $\hat{s}_{LB}(n)$  は最初にハイパスフィルタ  $H_{pre}$  で前処理される (8.4.1.1 節)。そして、短時間分析フィルタ  $A(z)$  を得るために、LP 分析 (8.4.1.2 節) が実行される。ピッチ遅延  $T_{LB}$  とそれに関連する最大正規化相関  $R_{max}$  を得るために、長期予測 (LTP) 分析 (8.4.1.3 節) が実行される。信号クラス ( $class$  と表記) を決定する信号クラス分け (8.4.1.4 節) がそれに続く。過去の励振信号  $r_{LB}(n)$ 、 $n < 0$  を得るために、過去の信号  $\hat{s}_{LB}(n)$  はフィルタ  $A(z)$  で逆フィルタリングされる。LTP 分析と信号クラス分けの結果に基づき、消失フレームに対する励振信号  $r_{LB}(n)$  ( $n = 0, \dots, 39$ ) は、過去の励振信号  $r_{LB}(n)$ 、 $n < 0$  の一部から長期予測 (ピッチ反復、8.4.1.5 節) により外挿される。この外挿された励振信号  $r_{LB}(n)$ 、 $n = 0, \dots, 39$  は、重み付けのない推定された消失復号信号  $\hat{s}_{LB}^{extr}(n)$  を与える、LP 合成 (8.4.1.6 項) の入力である。最終的な推定された失われた復号信号  $\hat{s}_{LB}^{PLC}(n)$  は、信号クラスに依存するミュート関数  $g_{mute}(n)$  である適応ミュート関数 (8.4.1.7 節)

の後に得られる。

#### 8.4.1.1 前処理

過去の復号された信号  $\hat{s}_{LB}(n)$ 、 $n = -295, \dots, -1$  は、 $\hat{s}_{LB}^{pre}(n)$ 、 $n = -295, \dots, -1$  を得るために、カットオフ周波数 50Hz の一次ハイパスフィルタで処理される。

$$H_{pre}(z) = \frac{1 - z^{-1}}{1 - 0.97z^{-1}} \quad (8-17)$$

#### 8.4.1.2 LP分析

短期分析および合成フィルタ、 $A(z)$ 、 $1/A(z)$  は、6 次線形予測 (LP) フィルタに基づく。LP 分析フィルタは以下のように表される。

$$A(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_6 z^{-6} \quad (8-18)$$

LP 分析は、窓掛けと自己相関計算、およびレビンソンダービンアルゴリズムの二つのパートからなる。自己相関の計算は、7.3.2 節と同一である、ただし、LP 窓  $w_{LP1}(i)$  は以下で定義される非対象ハミング窓  $w_{LP2}(i)$  であり、

$$w_{LP2}(i) = \begin{cases} 0.54 - 0.46 \cos\left(\frac{(i+80)\pi}{69}\right), & i = -80, \dots, -11 \\ 0.54 + 0.46 \cos\left(\frac{(i+11)\pi}{10}\right), & i = -10, \dots, -1 \end{cases} \quad (8-19)$$

ラグ窓は以下で与えられる。

$$w_{lag}^{FERC}(i) = \frac{1}{1.0001} \exp\left[-\frac{1}{2} \left(\frac{2\pi f_0 i}{f_s}\right)^2\right], \quad i = 1, \dots, 6 \quad (8-20)$$

ここで、 $f_0 = 60$  Hz はバンド幅拡張であり、 $f_s = 8000$  Hz はサンプリング周波数である。40dB のホワイトノイズ補正は同一であることに注意が必要である。処理量を削減するために 80 サンプル (8kHz サンプリング周波数における 10ms) に制限されたこの窓  $w_{LP2}(i)$  は、 $\hat{s}_{LB}^{pre}(n)$ 、 $n = -80, \dots, -1$  の最後の 10ms に適用される。レビンソンダービンアルゴリズムは、7.3.2 節のものと同じであるが、LP 次数は 4 の代わりに 6 である。LP 分析の後、LP 残差信号  $r_{LB}(n)$  を得るために、過去の信号  $\hat{s}_{LB}(n)$  は  $A(z)$  を用いてフィルタリングされる。

$$r_{LB}(n) = \hat{s}_{LB}(n) + \sum_{i=1}^6 a_i \hat{s}_{LB}(n-i) \quad n = -289, \dots, -1. \quad (8-21)$$

#### 8.4.1.3 LTP分析

FERC アルゴリズムは、LP 残差領域でピッチ周期反復を用いる。このため、ピッチラグ  $T_{LB}$  は、消失の直前の過去の有効前処理信号  $\hat{s}_{LB}^{pre}(n)$ 、 $n = -288, \dots, -1$  上で決定される。 $T_{LB}$  は、LTP 分析により開ループで推定される。

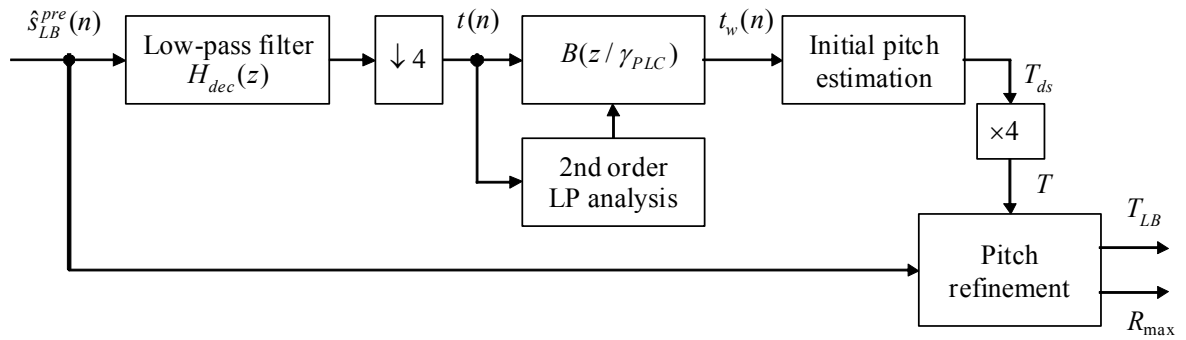


Figure 8 – 4 / JT-G711.1 Block diagram of LTP analysis  
(ITU-T G.711.1)

Figure 8 – 4 / JT-G711.1 で図示されるピッチ推定は、以下のステップで実行される：

- 信号  $\hat{s}_{LB}^{pre}(n)$ 、 $n = -288, \dots, -1$  は  $H_{dec}(z)$  でローパスフィルタリングされる。ここで、

$$H_{dec}(z) = 0.0563(1 + z^{-8}) + 0.0945(z^{-1} + z^{-7}) + 0.1301(z^{-2} + z^{-6}) + 0.1554(z^{-3} + z^{-5}) + 0.1646z^{-4} \quad (8-22)$$

は 8 次 FIR フィルタであり、2kHz でサンプリングされた信号  $t(n)$ 、 $n = -72, \dots, -1$  を得るために 1/4 に間引きされる。フィルタメモリは、毎回 0 に初期化される。

- 信号  $t(n)$ 、 $n = -72, \dots, -1$  は、信号  $t_w(n)$ 、 $n = -72, \dots, -1$  を得るためにフィルタ  $B(z/\gamma_{PLC})$  で重み付けされる。ここで、 $B(z) = 1 - b_1z^{-1} - b_2z^{-2}$ 、 $\gamma_{PLC} = 0.94$  である。 $B(z)$  の係数は、前節で記載された窓掛け、自己相関計算、レビンソンダービンアルゴリズムを用いて  $t(n)$  に対する 2 次 LP 分析で得られる。2kHz サンプリング周波数で 36ms の時間対応を与える、窓  $w_{LP2}(n)$ 、 $n = -72, \dots, -1$  の最後の 72 サンプルのみが使われることに注意が必要である。
- ピッチ遅延  $T_{ds}$  の予備推定は、以下のように、正規化相互相関で、重み付けされ間引きされた信号領域で計算される。
  - a) 初期化：  $T_{ds} = 18$
  - b) 正規化相互相関の計算：

$$R'_{\text{norm}}(i) = \frac{\sum_{n=-35}^{-1} t_w(n) t_w(n-i)}{\max\left(\sum_{n=-35}^{-1} t_w^2(n), \sum_{n=-35}^{-1} t_w^2(n-i)\right)} \quad i = 1, \dots, 35. \quad (8-23)$$

- c)  $n = -1$  から  $n = -35$  までの間の  $t_w(n)$  の最初のゼロクロス位置  $n_0$  の検出：  
一時インデックス  $i_0 = 1 - n_0$  にセット
- d)  $R'_{\text{norm}}(i) < 0$  となる、 $[1, 35]$  の中の最初の遅延  $i_1$  の決定。  $\min_{i=1, \dots, 35} (R'_{\text{norm}}(i)) \geq 0$  の場合は、残りの 2 つのステップは省略される。
- e) 最大相関探索のための下限値  $i_2$  の決定：  $i_2 = \max(i_0, i_1, 4)$
- f) 範囲  $[i_2, 35]$  における最大相関の探索：

$$T_{ds} = \arg \max_{i=i_2, \dots, 35} R'_{\text{norm}}(i). \quad (8-24)$$

倍ピッチの選択を避けるためにより小さいピッチを優先する手順が適用される。

- 以下のように、 $T = 4T_{ds}$  の近隣における最大相互相関の探索により、最初のピッチ遅延推定  $T_{ds}$  は前処理信号を用いて精細化され、 $T_{LB}$  を得る。

$$T_{LB} = \arg \max_{i=T-2, \dots, T+2} R_{\text{norm}}(i) \quad (8-25)$$

ここで、

$$R_{\text{norm}}(i) = \frac{\sum_{j=-T}^{-1} \hat{s}_{LB}^{\text{pre}}(n) \hat{s}_{LB}^{\text{pre}}(n-i)}{\max \left( \sum_{n=-T}^{-1} \hat{s}_{LB}^{\text{pre}}(n)^2, \sum_{n=-T}^{-1} \hat{s}_{LB}^{\text{pre}}(n-i)^2 \right)}. \quad (8-26)$$

相互相関の最大値  $R_{\text{max}}$  は以下のように計算される：

$$R_{\text{max}} = R_{\text{norm}}(T_{LB}) \quad (8-27)$$

そして、これは、次節の信号クラス分けにおいて、後に使用される。8.4.1.5 節にて説明されるとおり、ピッチ遅延  $T_{LB}$  は信号クラス分けに従ってさらに修正されるかもしれないことに注意が必要である。

#### 8.4.1.4 信号クラス分け

品質を最適にするため、FERC の方法は信号の特徴に依存する。例えば、消失に先行するフレームが非定常なセグメント（例えば、破裂音）のとき、信号はすばやくミュートされるべきであり、もし定常的なセグメント（例えば強い有声音）のときは、ピッチ同期して繰返され、ゆっくりと弱められるべきである。このため、クラス分けは、消失に先行する信号  $\hat{s}_{LB}^{\text{pre}}(n)$ 、 $n = -288, \dots, -1$  において実行される。クラス分けの結果は、LP 残差外挿（8.4.1.5 節）およびミュート制御（8.4.1.7 節）で使用される。4つのクラスがある。

- 大きなエネルギー変化の過渡部分（例えば破裂音）である、*TRANSIENT*
- 無声信号部分である、*UNVOICED*
- 弱い有声音部分（例えば、母音の開始部分もしくは終了部分）である、*WEAKLY\_VOICED*
- 有声音部分（例えば、安定した母音）である、*VOICED*

クラス分けに用いられる特性は以下に列挙される。

- LTP 分析の副産物である正規化相関  $R_{\text{max}}$
- 以下で定義される  $\hat{s}_{LB}^{\text{pre}}(n)$ 、 $n = -80, \dots, -1$  のゼロクロスレート  $c_{zc2}$

$$c_{zc2} = \sum_{n=-80}^{-1} \left[ (\hat{s}_{LB}^{\text{pre}}(n) \leq 0) \otimes (\hat{s}_{LB}^{\text{pre}}(n-1) > 0) \right] \quad (8-28)$$

ここで、 $\leq$ 、 $>$  の比較は、2 値の結果を与える（1 は真、0 は偽）。また、 $c_{zc2}$  は正値から負値へのゼロクロスのみをカウントする。

- 最後のピッチ周期中に検知された大きなピークの数  $c_{\text{peak}}$



$$c_{peak} = \sum_{n=-T_{LB}}^{-1} \left[ \frac{|r_{LB}(n)|}{8} > \max_{i=-2, \dots, 2} (|r_{LB}(n - T_{LB} + i)|) \right] \quad (8-29)$$

ここで、>の比較は2値の結果を与え（1は真、0は偽）、 $T_{LB}$ はLTP分析で計算されたピッチ遅延である。

これらの特性に基づき、信号カテゴリ *class* は以下のように得られる。もし、 $c_{zc2} \geq 20$ であれば、*class*はUNVOICEDに設定され、それ以外の場合は、 $R_{max}$ が閾値0.7と比較される。 $R_{max} > 0.7$ であれば、*class*はVOICEDとなり、そうでなければ、決定は、 $c_{peak}$ が0より大きいかどうか依存する。もし $c_{peak} > 0$ であれば*class*はTRANSIENTに設定され、そうでなければ、*class*はWEAKLY\_VOICEDに設定される。クラス分けの決定は、Figure 8-5/JT-G711.1で与えられるフローチャートに図示される。

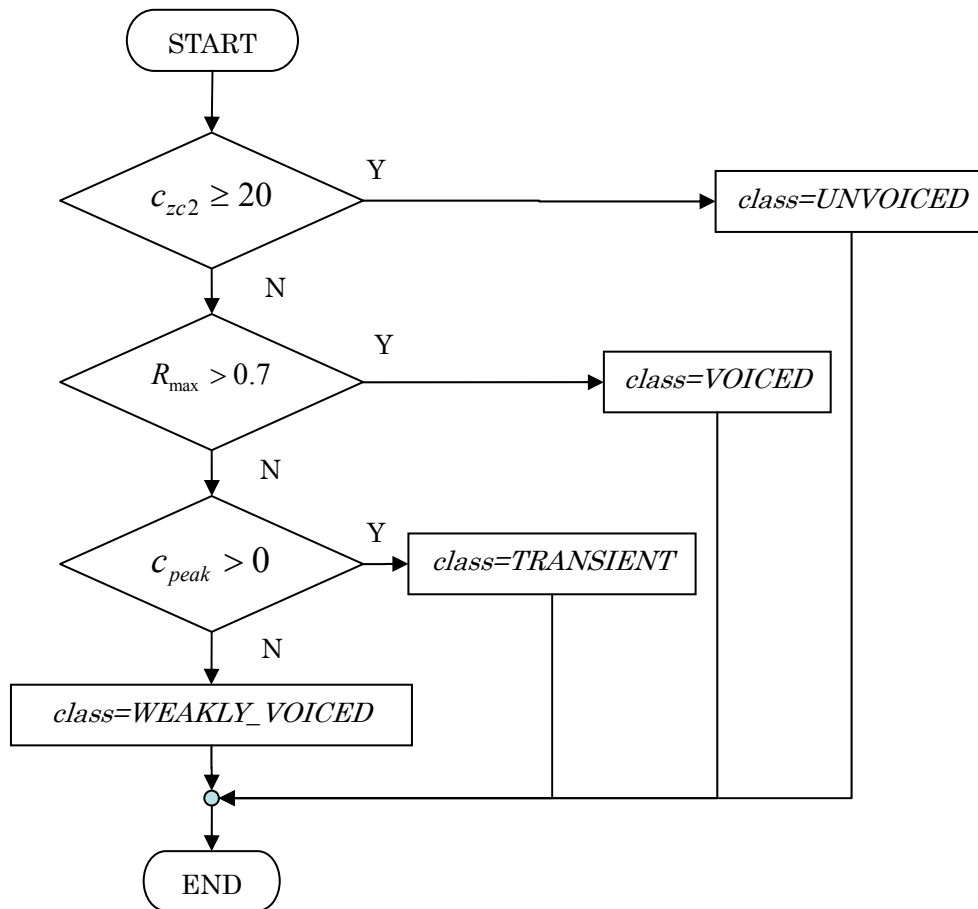


Figure 8-5/JT-G711.1 Class decision in lower-band FEREC (ITU-T G.711.1)

#### 8.4.1.5 LP残差のピッチ反復

ピッチ反復手順は、反復周期の残差信号  $r_{LB}(n)$  ( $n < 0$ ) から、消失フレームにおけるLP残差信号  $r_{LB}(n)$ 、 $n = 0, \dots, 39$ を計算するために用いられる。いわゆるピッチ遅延  $T_{LB}$ は、残差信号生成手順に使われる反復周期を決定する。反復周期は、最後の有効  $T_{LB}$  残差サンプル  $r_{LB}(n)$ 、 $n = -T_{LB}, \dots, -1$ を含む。

ピッチ反復周期は信号クラス分けの結果 *class* に依存して修正される場合がある。その長さ ( $T_{LB}$ ) 及びそのサンプル ( $r_{LB}(n)$ 、 $n = -T_{LB}, \dots, -1$ ) の振幅が共に変更される場合がある。

#### 8.4.1.5.1 ピッチ遅延修正

$T_{LB}$  の可能な修正は以下の通りである。

- a) もし  $class = UNVOICED$  かつ  $T_{LB} < 32$  ならば、 $T_{LB} = 2T_{LB}$   
このピッチ反復周期の倍増は、小さいピッチ遅延値のために生じる異音を避けるために実行される。
- b) もし  $class = TRANSIENT$  かつ  $T_{LB} > 40$  ならば、 $T_{LB} = 40$   
さらに、 $class \neq VOICED$  ( $class = UNVOICED, TRANSIENT, WEAKLY_VOICED$ ) かつ  $T_{LB}$  が偶数ならば、 $T_{LB} = T_{LB} + 1$
- c)  $class = VOICED$  のとき：  
反復周期のなかに二つの声門パルスが存在するかどうかを検証される（一つは最初に、二つ目は最後に）。これは、ピッチが減少するケースに起こり得るもので、以下のように検知される。

最初に、ピッチ反復周期における最大振幅サンプルの位置  $n_{\max r}$ 、振幅  $A_{\max r}$ 、およびその正負が探索され、保存される。

$$\begin{aligned} n_{\max r} &= \arg \max_{n=-T_{LB}, \dots, -1} |r_{LB}(n)| \\ A_{\max r} &= |r_{LB}(n_{\max r})| \end{aligned} \quad (8-30)$$

反復周期における平均振幅もまた計算される。

$$A_{\text{mean}r} = \frac{1}{T_{LB}} \sum_{n=-T_{LB}}^{-1} |r_{LB}(n)| \quad (8-31)$$

もし、 $A_{\max r} > 4A_{\text{mean}r}$  かつ、位置  $n_{\max r}$  が反復周期境界 ( $-T_{LB} \leq n_{\max r} \leq -T_{LB} + 4$  もしくは  $-5 \leq n_{\max r} \leq -1$ ) の一つに近い場合、二つ目の最大振幅位置  $n_{\max r2}$  が反復周期の他の境界において探索される。

$$\begin{aligned} n_{\max r2} &= \arg \max_{n=n_{\max r}+T_{LB}-5, \dots, -1} |r_{LB}(n)| & \text{if } -T_{LB} \leq n_{\max r} \leq -T_{LB} + 4 \\ n_{\max r2} &= \arg \max_{n=-T_{LB}, \dots, n_{\max r}-T_{LB}+5} |r_{LB}(n)| & \text{if } -5 \leq n_{\max r} \leq -1 \end{aligned} \quad (8-32)$$

もしも、 $A_{\max r2} > \frac{1}{2}A_{\max r}$ 、 $A_{\max r2} = |r_{LB}(n_{\max r2})|$ 、かつその極性がメインパルスと同一であれば、そのパルスは反復周期内の二番目の声門パルスであると考えられる。

二つの声門パルスが反復周期内で検知されたとき、ピッチ遅延  $T_{LB}$  は、発見された二つの声門パルス間の距離で書き換えられる。

$$T_{LB} = |n_{\max r} - n_{\max r2}| \quad (8-33)$$

ピッチ反復周期は信号クラス分け結果  $class$  に依存して修正される場合がある。その長さ ( $T_{LB}$ ) 及びそのサンプル ( $r_{LB}(n)$ 、 $n = -T_{LB}, \dots, -1$ ) の振幅が共に変更される場合がある。

#### 8.4.1.5.2 反復周期におけるサンプル振幅の修正

反復周期サンプル  $r_{LB}(n)$ 、 $n = -T_{LB}, \dots, -1$  の振幅修正は以下の通りである。

$class$  が *WEAKLY\_VOICED* の場合、反復周期のサンプルは、以下のようにその振幅を制限することにより修正される。

$$r_{LB}(n) = \text{sgn}(r_{LB}(n)) \cdot \min\left(\max_{i=-2, \dots, +2} (|r_{LB}(n - T_{LB} + i)|), |r_{LB}(n)|\right), \quad n = -T_{LB}, \dots, -1. \quad (8-34)$$

この  $r_{LB}(n)$  の修正は、破裂音のような大きな振幅過渡信号の反復を検知して避けるように意図されている。  
 $class$  が *UNVOICED* の場合、 $\Delta_e$  よりも大きい振幅を持つ反復周期のサンプルは4で除算される。ここで、 $\Delta_e$  は、 $r_{LB}(n)$  の最新の 10ms (80 サンプル) の平均振幅の 2.5 倍である。

$$\Delta_e = 2.5 \left( \frac{1}{80} \sum_{-1}^{-80} |r_{LB}(n)| \right) \quad (8-35)$$

$$r_{LB}(n) = r_{LB}(n) / 4, \quad \text{if } r_{LB}(n) > \Delta_e \quad (8-36)$$

#### 8.4.1.5.3 ピッチ反復手続き

消失フレームにおける LP 残差  $r_{LB}(n)$ 、 $n = 0, \dots, 39$  は、反復周期から外挿される。このピッチ反復手続きもまた信号クラス分けの結果  $class$  に依存する。

- $class$  が *VOICED* の場合、失われた信号  $r_{LB}(n)$  は、反復周期をピッチに同期して反復することにより得られる。

$$r_{LB}(n) = r_{LB}(n - T_{LB}). \quad n = 0, \dots, 39 \quad (8-37)$$

- $class$  が *VOICED* でない場合、ピッチ同期した反復手続きは、以下のような手続きを使用してサンプル毎に小さな揺らぎを導入することにより、過度の有声化を避けるために修正される。反復周期のサンプルは、二つずつのグループとみなすことができる。そして、グループを形成する各 2 サンプルは交換され、交換されたグループは外挿された残差信号を形成するために連結される。この手続きにより、外挿された信号  $r_{LB}(n)$  は、以下のように得られる。

$$r_{LB}(n) = r_{LB}(n - T_{LB} + (-1)^n). \quad n = 0, \dots, 39 \quad (8-38)$$

$T_{LB} < 40$  の場合は、外挿された残差信号は反復周期を延長し、手続きは反復される。

#### 8.4.1.6 LP合成

再生された消失フレームは、以下のように外挿が行われる。

$$\hat{s}_{LB}^{extr}(n) = r_{LB}(n) - \sum_{i=1}^6 a_i \hat{s}_{LB}^{extr}(n-i) \quad n = 0, \dots, 39 \quad (8-39)$$

ここで、 $r_{LB}(n)$  は外挿された残差信号である。

#### 8.4.1.7 適応的ミューティング

外挿された信号  $\hat{s}_{LB}^{extr}(n)$  のエネルギーは、適応的に変化するゲインファクタ  $g_{mute}$  を用いて以下のようにミューティングされる。

$$\hat{s}_{LB}^{extrg}(n) = g_{mute}^{(n)} \cdot \hat{s}_{LB}^{extr}(n) \quad (8-40)$$

以下において、 $g_{mute}$  の計算は  $\Delta\tau$ 、 $\delta_1$ 、 $\delta_{2p}$ 、 $\delta_{3p}$  の 4 つのパラメータに基づいている。これらのパラメ

ータは、Table 1 1 / JT-G711.1 に示すように、*class* の値に依存する。

Table 1 1 / JT-G711.1 Adaptive muting parameters with respect to *class*  
(ITU-T G.711.1)

Parameter	TRANSIENT	VOICED or WEAKLY_VOICED	UNVOICED
$\Delta\tau$	4	1	1
$\delta_1$	$8.3313 \cdot 10^{-3}$	$3.0518 \cdot 10^{-4}$	$3.0518 \cdot 10^{-4}$
$\delta_{2p}$	0	$3.0518 \cdot 10^{-4}$	$3.0518 \cdot 10^{-4}$
$\delta_{3p}$	0	Adaptively calculated (see clause 8.4.1.7.2)	$2.2888 \cdot 10^{-3}$

パラメータ  $\Delta\tau$  は、消失フレームにおけるミュートファクタの継続時間のインデックスを示し、正常フレームで0に初期化される。消失フレームの場合は、このパラメータは以下のようにサンプル毎に更新される。

$$\tau(n) = \tau(n-1) + \Delta\tau \quad (8-41)$$

#### 8.4.1.7.1 最初および二つ目の消失フレームにおけるミュートファクタ

最初の消失フレームの場合、すなわち、現在の  $m$  番目のフレームが消失し、その直前 ( $m-1$  番目) のフレームが消失していない場合、ミュートファクタ  $g_{mute}$  は以下のように算出される。1 ピッチの間隔の連続するセグメントの波形およびエネルギーは母音の立下りでしばしば迅速に変化するので、固定値のミュートファクタとする代わりに動的に更新するために、直前の2ピッチ周期のサンプルのエネルギー比が計算される。このエネルギー比パラメータ  $\rho^{(m)}$  は、最初の消失フレーム (フレーム番号  $m$ ) で以下のように計算される。

$$\rho^{(m)} = \begin{cases} \frac{1 - \sqrt{E_{p1} / E_{p2}}}{T_{LB}} & \text{if } E_{p1} < E_{p2} \text{ and } E_{p2} > 360 \cdot 2^{11} T_{LB} \\ 0 & \text{otherwise} \end{cases} \quad (8-42)$$

ここで  $E_{p1}$  および  $E_{p2}$  は、以下のように、それぞれ直前およびその前のピッチ周期のエネルギーである。

$$E_{p1} = \sum_{n=-1}^{-T_{LB}} (\hat{s}_{LB}(n))^2 \quad (8-43)$$

$$E_{p2} = \sum_{n=-(T_{LB}+1)}^{-2T_{LB}} (\hat{s}_{LB}(n))^2 \quad (8-44)$$

次に、以下のように、ミュートファクタがサンプルごとに再帰的に計算される。

$$g_{mute}^{(n)} = \begin{cases} g_{mute}^{(n-1)} - \delta_1 & \text{if } \delta_1 > \rho^{(m)} \\ g_{mute}^{(n-1)} - \rho^{(m)} & \text{otherwise} \end{cases} \quad n = 0, \dots, 39 \quad (8-45)$$

ここで、 $g_{mute}^{(-1)}$  は1に設定される。二つ目の消失フレームの場合は、 $\rho^{(m)}$  が以下のように更新される。

$$\rho^{(m)} = 0.5\rho^{(m-1)} \quad (8-46)$$

ミュートファクタは、 $g_{mute}^{(0),(m)} = \begin{cases} g_{mute}^{(39),(m-1)} - \delta_1 & \text{if } \delta_1 > \rho^{(m)} \\ g_{mute}^{(39),(m-1)} - \rho^{(m)} & \text{otherwise} \end{cases}$  のように初期化される。ここで

$g_{mute}^{(0),(m)}$  および  $g_{mute}^{(39),(m-1)}$  は、それぞれ現フレームの初期値および直前のフレームの最後の値である。二つ目のフレームでは、このファクタは式 8-45 を用いて同様に更新される。

### 8.4.1.7.2 三つ目以降の消失フレームでのミュートファクタ

二つの連続した消失フレームの後、ミュートファクタ  $g_{mute}$  は、 $\tau(n)$  の値に依存してサンプルごとに更新される。

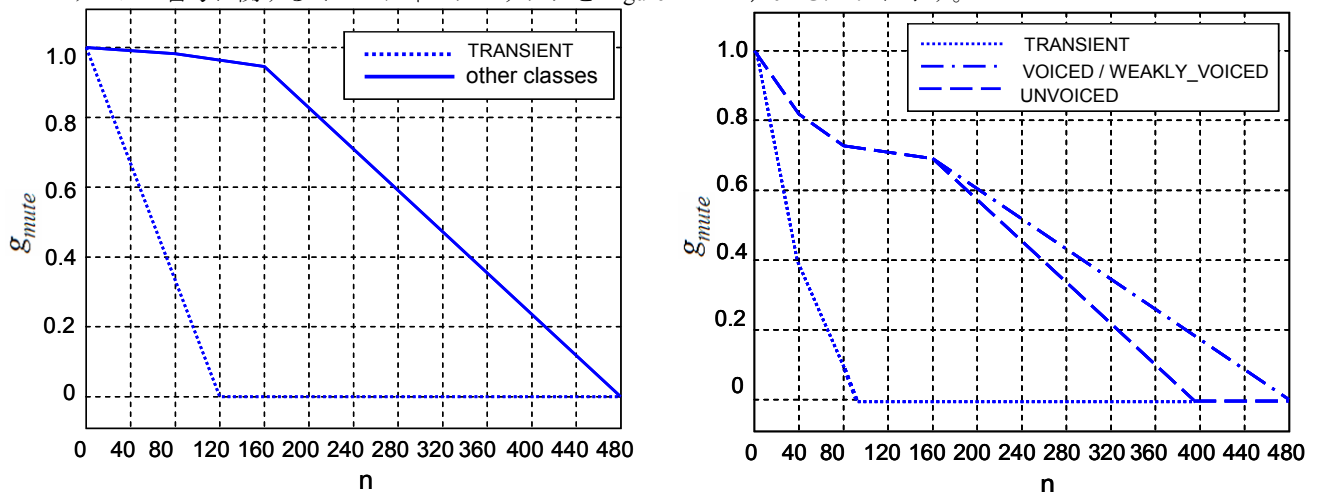
Table 1 2 / JT-G711.1 Adaptive muting factor calculation  
(ITU-T G.711.1)

Time index $\tau(n)$	Muting factor $g_{mute}^{(n)}$
$80 \leq \tau(n) < 160$	$g_{mute}^{(n)} = g_{mute}^{(n-1)} - (\delta_1 + \delta_{2p})$
$160 \leq \tau(n) < 480$	$g_{mute}^{(n)} = g_{mute}^{(n-1)} - (\delta_1 + \delta_{2p} + \delta_{3p})$
$480 \leq \tau(n)$	$g_{mute}^{(n)} = 0$

$class$  が *TRANSIENT* の場合は、三つ目の消失フレームの先頭ではミュートファクタ時間インデックス  $\tau(n)$  がすでに 320 であり、最初の条件すなわち  $80 \leq \tau(n) < 160$  は適用されない。 $class$  が *VOICED* あるいは *WEAKLY\_VOICED* の場合は、 $\delta_{3p}$  は以下のように適応的に計算される。

$$\delta_{3p} = \begin{cases} \max\left(\frac{g_{mute}^{(n)}}{320} - (\delta_1 + \delta_{2p}), 0\right) & \text{if } g_{mute}^{(n)} < 0.9277 \\ 2.2888 \cdot 10^{-3} & \text{otherwise} \end{cases} \quad (8-47)$$

サンプル番号に関するミュートファクタを Figure 8-6 / JT-G711.1 に示す。



(a) For  $\rho^{(m)} = 0$

(b) For  $\rho^{(m)} = 0.015$  for *TRANSIENT* class and 0.0046 for *other class*

Figure 8-6 / JT-G711.1 Muting factor  $g_{mute}$  as a function of the sample number  $n$   
(ITU-T G.711.1)

### 8.4.2 消失フレームの外挿：消失フレームの次の消失フレームの場合

現在および直前のフレームの双方が消失フレームであるならば、分析パラメータ  $(\{a_j\}_{j=0, \dots, 6}, T_{LB}, class)$  は、最初の消失フレームで計算され、再利用される。さらに、LPフィルタ  $A(z)$  が多少減衰される。 $m$  番目の連続する消失フレームについて、新たなフィルタは以下のように得られる。

$$a_j^{(m)} = 0.99^j a_j^{(m-1)} \quad j = 1, \dots, 6 \quad (8-48)$$

### 8.4.3 信号の再同期アルゴリズムおよびクロスフェーディング

消失した有音フレームの補償期間においては、推定されたピッチラグを用いて過去の信号が繰り返される。

一連の消失フレームのあとに正常フレームが受信された場合、ピッチの不連続性が現れ、それによって不快な異音を生じる。この問題を回避するために、一連の消失フレームの後の最初の正常フレームにおける有声音信号に信号の再同期が処理される。この再同期は、信号の遷移を平滑化して異音の発生源を回避するために、最後の補償フレームおよび最初の正常な復号フレームに適用される。信号の再同期の原理を Figure 8-7/JT-G711.1 に示す。

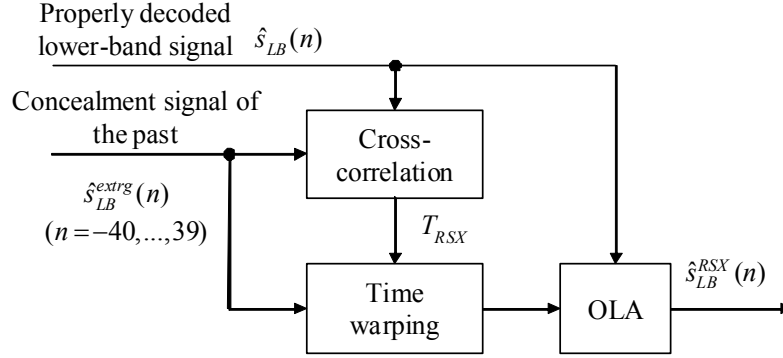


Figure 8-7/JT-G711.1 Details of the signal resynchronization algorithm (ITU-T G.711.1)

ステップは以下ようになる。最初の正常フレームが受信され、 $\hat{s}_{LB}(n)$  が正しく復号される場合は、追加拡張フレームが FERC アルゴリズム（すなわち直前の消失フレームにおける補償信号のもうひとつの外挿）によって生成される。現フレームにおける、正常フレームの復号信号  $\hat{s}_{LB}(n)$ 、 $n = 0, \dots, 39$  および現在の正常フレームに対して追加拡張された補償信号  $\hat{s}_{LB}^{extrg}(n)$ 、 $n = -40, \dots, 39$  の間の相互相関分析が実行される。ピッチ遅延  $T_{RSX}$  がこの相互相関分析に基づいて抽出される。この遅延に基づいて、（直前および現在のフレームを連結したものに对应する）補償信号は、この追加拡張された補償信号を時間方向に変形することにより、復号信号に同期する。これは、補償信号の位相を、正常に復号した信号の位相に一致させるために行う。補償信号を時間方向に変形したあと、直前のフレームに对应する部分は抽出および出力され、現在のフレームに对应する部分は、外挿された補償信号と正しく復号された信号とが円滑に遷移するように、復号された正常フレームとクロスフェーディング処理される。

一連の消失フレームのあとの最初の復号フレームにおいて、この補償アルゴリズムは、（その復号フレームが消失したかのように）補償信号の追加フレームを生成する。そして補償信号  $\hat{s}_{LB}^{extrg}(n)$  と復号された信号  $\hat{s}_{LB}(n)$  との相互相関分析が  $[-5, 5]$  の範囲で実行される。相関関数は以下のように定義される。

$$R_{RSX}(i) = \sum_{n=0}^{34} \hat{s}_{LB}^{extrg}(n+i) \hat{s}_{LB}(n) \quad i = -5, \dots, 5 \quad (8-49)$$

相関関数  $R_{RSX}$  の最大値およびこの最大値に对应する遅延  $T_{RSX}$  が、それぞれ以下のように探索される。

$$\begin{aligned} R_{RSX \max} &= \max_{i=-5, \dots, 5} (R_{RSX}(i)) \\ T_{RSX}^{RSX} &= \arg \max_{i=-5, \dots, 5} (R_{RSX}(i)) \end{aligned} \quad (8-50)$$

最大の相関値を正規化するために、以下の2つのエネルギーが計算される。

$$\begin{aligned} E_0^{RSX} &= \sum_{n=0}^{39} (\hat{s}_{LB}(n))^2 \\ E_1^{RSX} &= \sum_{n=0}^{39} (\hat{s}_{LB}^{extrg}(T_{RSX} + n))^2 \end{aligned} \quad (8-51)$$

そして  $R_{RSX \max}$  がその2つのエネルギーの積の平方根で除算される。

$$C_{RSX} = \frac{R_{RSX \max}}{\sqrt{E_0^{RSX} E_1^{RSX}}} \quad (8-52)$$

この再同期処理は、外挿されたフレームと正常フレームとのエネルギー間に大きな相違がある場合は適用されない。そのため、エネルギー比  $r^{RSX}$  が次の式にしたがって計算される。

$$r_{RSX} = \frac{\max(E_0^{RSX}, E_1^{RSX})}{\min(E_0^{RSX}, E_1^{RSX})} \quad (8-53)$$

この再同期を処理する条件は、以下のように定義される。

$$[(class == VOICED) \text{ AND } (C_{RSX} > 0.7) \text{ AND } (r_{RSX} > 0.5)]$$

もしこの条件が満たされない場合は、再同期処理は実行されない。(すなわち、 $\hat{s}_{LB}^{RSX}(n) = \hat{s}_{LB}^{extrg}(n)$ ,  $n = -40, \dots, 39$ となる。) 一方、この条件が満たされる場合は、補償信号はその前に得られた遅延パラメータによって拡張されたり短縮(圧縮)されたりする。これは、補償信号  $\hat{s}_{LB}^{extrg}(n)$ ,  $n = -40, \dots, 39$  全体に対して行われることに注意すること。

信号の圧縮あるいは拡張は、線形補間を用いた“再サンプリング”によって実行される。元の信号の隣接するサンプル点間の距離を“1”とした場合、再サンプリングされた隣接したサンプル点間の距離は以下のように定義される。

$$\Delta_{RSX} = \frac{79 - T_{RSX}}{79} \quad (8-54)$$

$d^{RSX}$  は[-5, 5]の範囲を取ることにしか許されていないため、 $\Delta_{RSX}$  は[0.9367, 1.0633]の範囲の値しか取れない。再サンプリングされた信号のこれらの値は、 $\Delta_{RSX}$  の倍数によって与えられた位置について、元の信号の値から計算される。すなわち、以下ようになる。

$$p(n) = (n + 40) \cdot \Delta_{RSX} \quad n = -40, \dots, 39 \quad (8-55)$$

上述のように、この再サンプリングは、補償信号  $\hat{s}_{LB}^{extrg}(n)$ ,  $n = -40, \dots, 39$  全体について実行される。再サンプリングされた補償信号  $\hat{s}_{LB}^{RSX}(n)$  は以下によって与えられる。

$$\hat{s}_{LB}^{RSX}(n) = (1 - \alpha^{RSX}(n)) \hat{s}_{LB}^{extrg}(\lfloor p(n) \rfloor) + \alpha^{RSX}(n) \hat{s}_{LB}^{extrg}(\lfloor p(n) \rfloor + 1) \quad (8-56)$$

$$n = -40, \dots, \min(79, 79 + d^{RSX})$$

ここで、 $\alpha_{RSX}(n) = p(n) - \lfloor p(n) \rfloor$  である。

$T_{RSX} > 0$  の場合は、消失したサンプル  $\hat{s}_{LB}^{RSX}(n)$ ,  $n = 39 - T_{RSX}, \dots, 39$  はゼロに設定される。これについては、この再同期処理に続くクロスフェーディング処理 (OLA) が三角窓を用い、通常、再同期信号の最後のサンプル列はゼロに近い係数を乗算されるので、特に問題ない。再同期処理の原理を Figure 8-8/JT-G711.1 に示す。この図では、2つのサンプルの拡張が行われている。

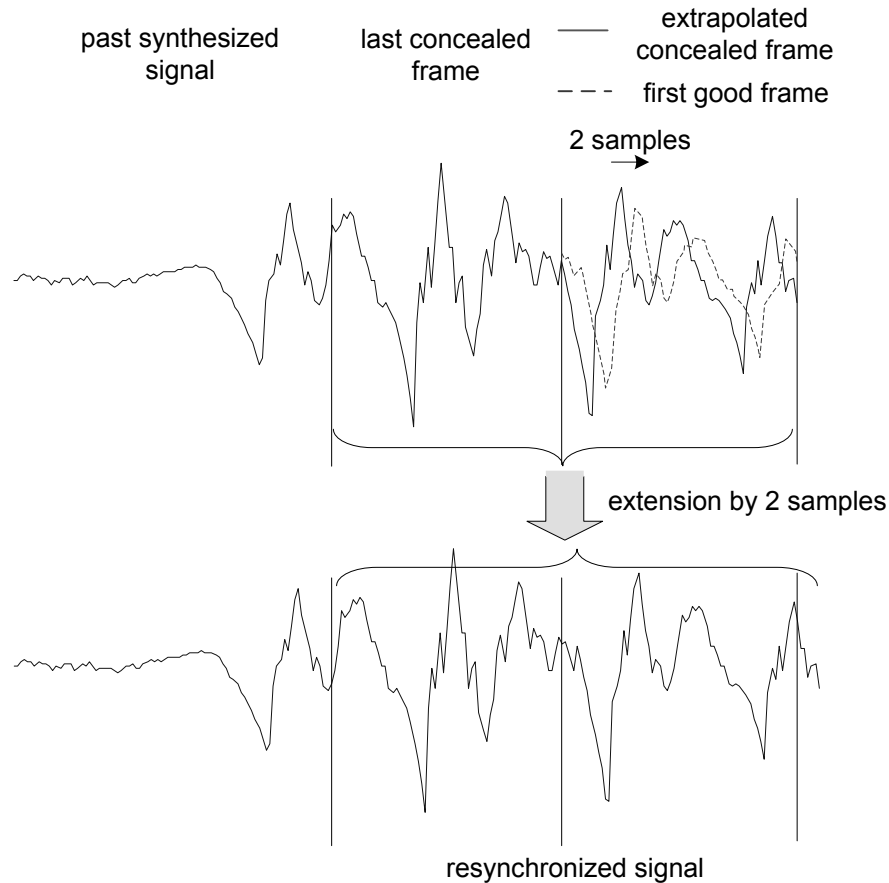


Figure 8-8 / JT-G711.1 Principle of resynchronization  
(ITU-T G.711.1)

信号が再同期処理されたあと、そのエネルギーは動的なエネルギー調整アルゴリズムにしたがってスケージング処理され、再同期かつスケージングされた信号と復号信号とのクロスフェーディング処理 (OLA) が実行される。これは、8.4.4 節および 8.4.5 節で述べる。

#### 8.4.4 動的なエネルギー調整アルゴリズム

再同期された補償信号  $\hat{s}_{LB}^{RSX}(n)$ ,  $n=0, \dots, 39$  のエネルギーが新たに復号された信号  $\hat{s}_{LB}(n)$  のエネルギーに一致しないことがよく起きる。これにより、連続する定常的な音声フレームに対して異音がたびたび生じてしまう。直前のフレームと新たに復号されたフレームとのエネルギーについて、再生信号をより平滑にするために、再同期のあとに、動的にエネルギーを調整する。

動的にエネルギーを調整するために、新たに復号された信号と再生信号とのエネルギー比が用いられる。波形の歪みや予期せぬ振幅の変化による雑音を回避するために、動的なエネルギー調整は  $E_1^{RSX} > E_0^{RSX}$  の場合にのみ実行可能となる。

この条件の場合、動的なエネルギー調整係数は以下のように計算される。

$$\alpha_{es} = \frac{1}{80} \sqrt{1 - 1/r_{RSX}} \quad (8-57)$$

ここで、 $E_0^{RSX}$  および  $E_1^{RSX}$  は式(8-51)で定義され、 $r^{RSX}$  は式(8-53)で定義されている。そして、再生信号は以下のようにスケージングされる。

$$\hat{s}_{LB}^{SYN}(n) = \left(1 - (n+40)\alpha_{es}\right) \hat{s}_{LB}^{RSX}(n) \quad n = -40, \dots, 39 \quad (8-58)$$

動的エネルギースケージングアルゴリズムの原理および効果を Figure 8-9 / JT-G711.1 に示す。最初の正常フレームにおける再同期信号のエネルギーは、元の信号のエネルギーと一致していない。動的なエネルギー調整およ



びクロスフェーディング（クロスフェーディングは 8.4.5 節で説明する次の動作である）の後、再生信号は元の信号に近くなる。

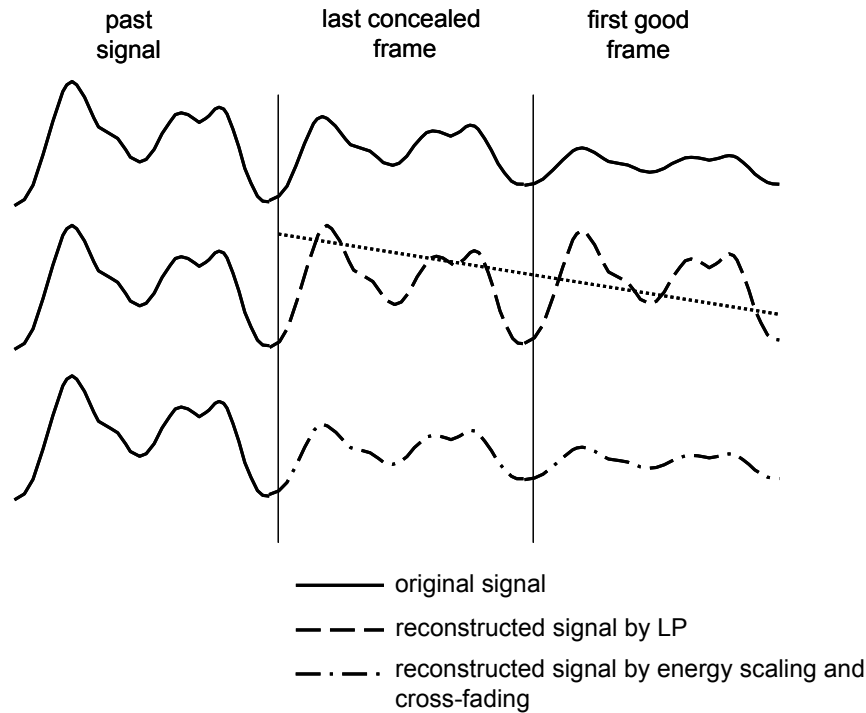


Figure 8-9 / JT-G711.1 Principle of dynamic energy scaling  
(ITU-T G.711.1)

再同期化されかつエネルギーがスケールされた補償信号が得られた後は、次節で述べるように、過去のフレームに相当する部分（すなわち  $\hat{s}_{LB}^{SYN}(n)$ ,  $n = -40, \dots, -1$ ）が抽出され、現在のフレームに相当する部分（すなわち  $\hat{s}_{LB}^{SYN}(n)$ ,  $n = 0, \dots, 39$ ）が復号信号  $\hat{s}_{LB}(n)$ ,  $n = 0, \dots, 39$  とクロスフェーディング処理される。

#### 8.4.5 補償信号と復号信号とのクロスフェーディング

クロスフェーディング処理は、1 フレーム長、すなわち 40 サンプルの、再同期化されかつエネルギーがスケールされた補償信号  $\hat{s}_{LB}^{SYN}(n)$  および新たに復号された信号  $\hat{s}_{LB}(n)$  に対して適用される。クロスフェーディングされた信号は、以下のように、復号信号  $\hat{s}_{LB}(n)$  に置き換わる。

$$\hat{s}_{LB}(n) = (1 - w_{cf}(n))\hat{s}_{LB}^{SYN}(n) + w_{cf}(n)\hat{s}_{LB}(n) \quad n = 0, \dots, 39 \quad (8-59)$$

ここで、 $w_{cf}(n)$  は次のように定義される三角窓である。

$$w_{cf}(n) = \frac{(n+1)}{40} \quad n = 0, \dots, 39 \quad (8-60)$$

### 8.5 高域 FERC

消失フレームの場合、中間信号を回復させるために、高域 FERC アルゴリズムが用いられる。高域 FERC は、補償のために生成された回復信号における潜在的な品質の劣化を最小化するために、ピッチ的な高域信号と雑音的な高域信号とを区別するものである。

直前に復号された高域信号が高い相関を示す場合は、低域の FERC のために計算されたピッチパラメータの近傍で高域のピッチラグが推定される。高域 FERC の履歴バッファにおける直前のピッチ周期のサンプルが、現在の消失フレームの逆 MDCT の基本データとして用いられる。この逆 MDCT の基本データに正弦波窓が適用され、続いて OLA が実行される。

高い相関を示さない場合は、最後の正常フレームから減衰された逆 MDCT 信号がコピーされ、消失フレーム

の逆 MDCT データとして用いられる。回復された高域信号を生成するために、OLA が実行される。

Figure 8-10/JT-G711.1 は、高域 FERC のブロック図を示す。

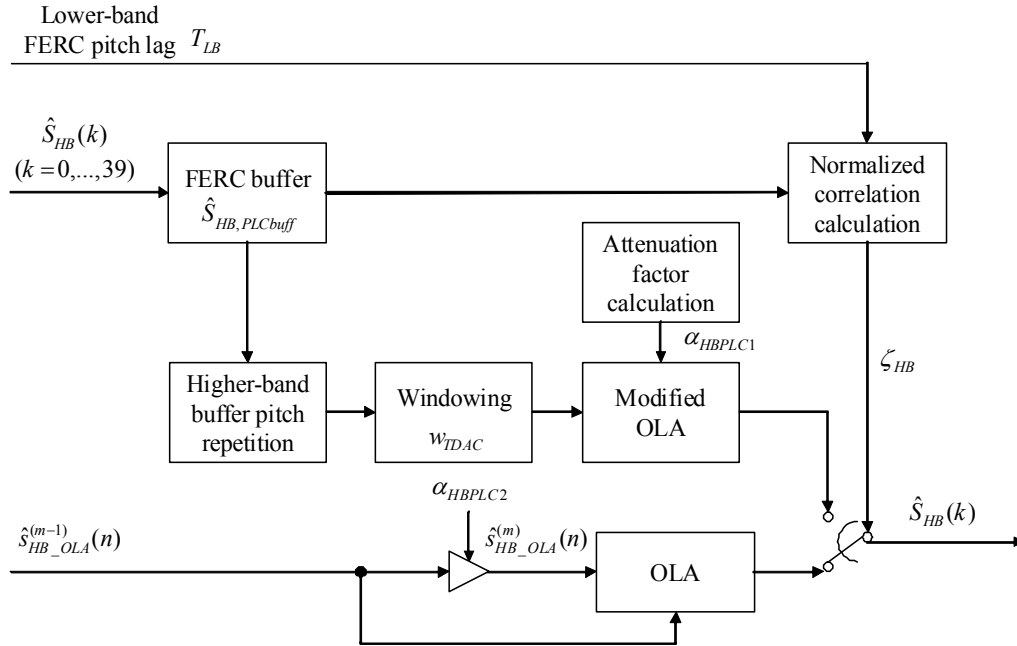


Figure 8-10/JT-G711.1 Block diagram of higher-band FERC  
(ITU-T G.711.1)

### 8.5.1 正常フレームの場合における高域信号バッファの更新

復号された高域信号  $\hat{S}_{HB}(n)$  は、将来のフレームが消失した場合に備え、高域の再生のために常に蓄積される。高域 FERC バッファ  $\hat{S}_{HB,PLCbuff}(n)$  はゼロに初期化される。バッファ長は  $T_{HBmax} + 120$  である。ここで  $T_{HBmax} = 144$  はピッチラグの最大値である。このバッファは二つの部分を持つ。前半部分は直前に復号された高域信号を直積するために用いられ、後半部分はピッチ周期で反復された高域信号を生成するために用いられる。この二つの部分の長さは、それぞれ  $T_{HBmax} + 40$  および 80 である。

正常フレームの場合、高域 FERC バッファ  $\hat{S}_{HB,PLCbuff}$  は以下のように更新される。

$$\hat{S}_{HB,PLCbuff}(n) = \hat{S}_{HB,PLCbuff}(n + 40) \quad n = 0, \dots, T_{HBmax} - 1 \quad (8-61)$$

$$\hat{S}_{HB,PLCbuff}(n + T_{HBmax}) = \hat{S}_{HB}(n) \quad n = 0, \dots, 39 \quad (8-62)$$

### 8.5.2 正常フレーム受信直後に消失フレームを受信した場合における正規化相関の計算および高域ピッチラグの推定

高域ピッチラグ  $T_{HB}$  は、以下のステップで、正規化相関関数から推定される。

#### 1. 低域ピッチラグの近傍での正規化相関関数の計算

$$R_{HB}(i) = \frac{\sum_{n=0}^{39} \hat{S}_{HB,PLCbuff}(T_{HBmax} + n) \cdot \hat{S}_{HB,PLCbuff}(T_{HBmax} + n - i)}{\sqrt{\sum_{n=0}^{39} \hat{S}_{HB,PLCbuff}^2(T_{HBmax} + n) \cdot \sum_{n=0}^{39} \hat{S}_{HB,PLCbuff}^2(T_{HBmax} + n - i)}} \quad (8-63)$$

$$\max(T_{LB} - 3, T_{HBmin}) \leq i \leq \min(T_{LB} + 3, T_{HBmax})$$

ここで、 $T_{LB}$  は低域 FERC モジュールから得られた低域ピッチラグであり、 $T_{HBmin} = 16$  はピッチラグの最小値であり、 $R_{HB}(i)$  は正規化された相関関数である。 $T_{HB}$  は  $T_{LB}$  に初期化される。

2. 相関関数の最大値の探索

$$R_{HB\_MAX} = \max(R_{HB}(i)) \quad i = \max(T_{LB} - 3, T_{HBmin}), \dots, \min(T_{LB} + 3, T_{HBmax}) \quad (8-64)$$

3. 高域ピッチラグの推定

$$T_{HB} = \arg \max_i (R_{HB}(i)) \quad i = \max(T_{LB} - 3, T_{HBmin}), \dots, \min(T_{LB} + 3, T_{HBmax}) \quad (8-65)$$

4.  $|R_{HB\_MAX}| \geq 0.7$  の場合は、高相関フラグ  $\zeta_{HB} = 1$  とし、それ以外の場合は  $\zeta_{HB} = 0$  とする。

これらの処理は、最初の消失フレームにおける高相関フラグ  $\zeta_{HB}$  を得るためにのみ実行されることに注意しなくてはならない。消失フレームが続いた場合、相関強度フラグの値は繰り返し用いられる。

### 8.5.3 相関が高い場合

信号の相関が高い場合 ( $\zeta_{HB} = 1$ )、高域信号は、直前のフレームのピッチ周期での反復、減衰、および OLA によって補償される。

直前のフレームも消失フレームである場合は、以下のように、高域 FERC バッファの前半部分は、バッファの内容をシフトすることによって複製される。

$$\hat{s}_{HB,PLCbuff}(n) = \hat{s}_{HB,PLCbuff}(n + 40) \quad n = 0, \dots, T_{HBmax} + 39 \quad (8-66)$$

次に、 $\hat{s}_{HB,PLCbuff}(n)$  の後半部分は、前半部分のピッチ反復によって生成される。

$$\hat{s}_{HB,PLCbuff}(n + T_{HBmax} + 40) = \hat{s}_{HB,PLCbuff}(n - T_{HB} + T_{HBmax} + 40) \quad n = 0, \dots, 79 \quad (8-67)$$

消失した（そして現在の）フレーム  $\hat{s}_{HB\_OLA}^{(m)}(n)$  の中間逆変換高域信号が、OLA 重み付け窓  $w_{TDAC}(n)$  を適用することによって推定される。

$$\hat{s}_{HB\_OLA}^{(m)}(n) = \hat{s}_{HB,PLCbuff}(n + T_{HBmax} + 40)w_{TDAC}(n) \quad n = 0, \dots, 79 \quad (8-68)$$

次に、高域信号  $\hat{s}_{HB}(n)$  が、減衰を持った変形 OLA 窓を用いて計算される。

$$\hat{s}_{HB}(n) = \begin{cases} \alpha_{HBPLC1}^{(n)} (w_{TDAC}(n + 40)\hat{s}_{HB\_OLA}^{(m-1)}(n) + w_{TDAC}(n)\hat{s}_{HB\_OLA}^{(m)}(n)) & \text{if } \alpha_{HBPLC1}^{(0)} \geq 0.2 \\ 0 & \text{otherwise} \end{cases} \quad n = 0, \dots, 39 \quad (8-69)$$

ここで、 $\alpha_{HBPLC1}^{(n)}$  はサンプルごとの減衰重み付け係数である。この減衰重み付け係数  $\alpha_{HBPLC1}^{(n)}$  は、以下のように再帰的に更新される。

$$\begin{cases} \alpha_{HBPLC1}^{(n)} = \alpha_{HBPLC1}^{(n-1)} - 0.005 & \text{if } \alpha_{HBPLC1}^{(0)} \geq 0.2 \\ \alpha_{HBPLC1}^{(n)} = \alpha_{HBPLC1}^{(0)} & \text{otherwise} \end{cases} \quad n = 0, \dots, 39 \quad (8-70)$$

ここで、 $\alpha_{HBPLC1}^{(0)}$  は、直前のフレームの最後で  $\alpha_{HBPLC1}^{(39)} - 0.005$  に初期化される。もしこのフレームが正常フレームの次の最初の消失フレームであれば、 $\alpha_{HBPLC1}^{(0)}$  はこのフレームの最後に 1 に再設定される。

次に、 $\hat{s}_{HB\_OLA}^{(m-1)}$  が以下のように更新される。

$$\hat{s}_{HB\_OLA}^{(m-1)}(n) = \hat{s}_{HB\_OLA}^{(m)}(n + 40) \quad n = 0, \dots, 39 \quad (8-71)$$

消失フレームの次の正常フレームの場合、中間逆変換高域信号は次のように計算される。

$$\hat{s}_{HB\_OLA}^{(m-1)}(n) = \alpha_{HBPLC1}^{(39),(m-1)} \hat{s}_{HB\_OLA}^{(m-1)}(n) \quad n = 0, \dots, 39 \quad (8-72)$$

ここで  $\alpha_{HBPLC1}^{(39),(m-1)}$  は直前の  $(m-1)$  番目のフレームの最後の係数である。

### 8.5.4 相関が低い場合

信号の相関が低い場合 ( $\zeta_{HB} = 0$ ) は、中間逆変換高域信号  $\hat{s}_{HB\_OLA}^{(m)}(n)$  は、以下のように、直前の信号  $\hat{s}_{HB\_OLA}^{(m-1)}(n)$  の減衰したものをを用いて補償される。

$$\hat{s}_{HB\_OLA}^{(m)}(n) = \alpha_{HBPLC2} \hat{s}_{HB\_OLA}^{(m-1)}(n) \quad n = 0, \dots, 79 \quad (8-73)$$

ここで、 $\alpha_{HBPLC2} = 0.875$  は減衰係数である。次に、出力信号  $\hat{s}_{HB}(n)$  を合成するために、OLA が式 (8-15) を用いて実行される。

## 8.6 合成QMF

合成QMFは、低域復号信号と高域復号信号から 16kHz サンプリングの出力信号を合成するのに用いられる。低域と高域における 8kHz サンプリングの信号は、それぞれ 2 倍にアップサンプリングされる。さらに、アップサンプリングした信号は、それぞれの帯域の合成フィルタによってフィルタリングされる。これら二つの合成フィルタ係数は、以下の通りに与えられる。

$$\begin{cases} h_L^{qmfS}(i) = h_L^{qmfA}(i) \\ h_H^{qmfS}(i) = -h_H^{qmfA}(i) \end{cases} \quad i = 0, \dots, 31 \quad (8-74)$$

ここで、 $h_L^{qmfS}$  と  $h_H^{qmfS}$  は低域と高域合成フィルタ係数であり、 $h_L^{qmfA}$  と  $h_H^{qmfA}$  は 7.2 章で記述した分析QMFのフィルタ係数である。16kHz 標本化出力  $\hat{s}_{QMF}$  は、以下のように二つのフィルタ出力信号を加算することで得られる。

$$\hat{s}_{QMF}(n) = \sum_{i=0}^{31} h_L^{qmfS}(i) \hat{s}_{LW}(n-i) + \sum_{i=0}^{31} h_H^{qmfS}(i) \hat{s}_{HW}(n-i), \quad n = 0, \dots, 79 \quad (8-75)$$

ここで、 $\hat{s}_{LW}(n)$  と  $\hat{s}_{HW}(n)$  は、それぞれ、低域と高域におけるアップサンプル信号である。

演算量削減のため、上記演算は以下のように最適化される。はじめに、二つの中間変数  $\bar{s}_{sum}(n)$  と  $\bar{s}_{diff}(n)$  が、下式から得られる。

$$\begin{aligned} \hat{s}_{sum}(n) &= \sum_{i=0}^{15} h_0^{qmf}(i) (\hat{s}_{LB}(n-i) + \hat{s}_{HB}(n-i)) \\ \hat{s}_{diff}(n) &= \sum_{i=0}^{15} h_1^{qmf}(i) (\hat{s}_{LB}(n-i) - \hat{s}_{HB}(n-i)) \end{aligned} \quad n = 0, \dots, 39 \quad (8-76)$$

ここで、 $\hat{s}_{LB}(n)$  は低域復号信号、 $\hat{s}_{HB}(n)$  は高域復号信号、 $h_0^{qmf}$  と  $h_1^{qmf}$  は Table 8/JT-G711.1 で示したフィルタ係数である。そして、中間信号  $\hat{s}_{sum}(n)$  と  $\hat{s}_{diff}(n)$  は、16kHz 標本化信号  $\hat{s}_{QMF}(n)$  を得るために、以下のようにインターリーブされる。

$$\begin{aligned} \hat{s}_{QMF}(2n) &= 2\hat{s}_{diff}(n), \\ \hat{s}_{QMF}(2n+1) &= 2\hat{s}_{sum}(n) \end{aligned} \quad n = 0, \dots, 39 \quad (8-77)$$

## 8.7 ノイズゲート

準無音区間における合成信号の静粛性をさらに向上させるため、ノイズゲートが復号器に付加される。ノイズゲートの機能は、フレームエネルギーが非常に小さい時に、出力信号を減衰させることである。レベルと時間の両方において、徐々に減衰させる。減衰のレベルは、信号に依存し、サンプル毎に徐々に更新する。

はじめに、低域復号信号  $\hat{s}_{LB}(n)$  は、1 次のハイパス FIR フィルタでフィルタリングされる。

$$z(n) = \hat{s}_{LB}(n) - 0.75 \cdot \hat{s}_{LB}(n-1) \quad n = 0, \dots, 39 \quad (8-78)$$

そして、そのエネルギーは、以下のように計算される。

$$E_0 = \sum_{n=0}^{39} z^2(n) \quad (8-79)$$

ノイズゲートの頻繁な開閉を避けるため、1つ前のフレームのエネルギー  $E_{-1}$  を、現フレームのエネルギーに加算し、トータル（統合）エネルギーとする。

$$E_t = E_0 + E_{-1} \quad (8-80)$$

ここで、 $E_{-1}$  は最初に 0 に初期化され、 $E_{-1}$  は各フレームの終わりに  $E_0$  によって置き換えられる。以下のように、信号エネルギーに基づき、目標ゲインは、総エネルギーの平方根で算出される。但し、下限値は 0.25 に制限される。

$$g_t = \max\left(\frac{\sqrt{E_t}}{2^7}, 0.25\right) \quad (8-81)$$

ノイズゲートは、目標ゲイン  $g_t = 1.0$  とすることで、徐々ににその動作を停止させる（開かれる）。これは、復号器における合成信号エネルギーが、高域に、すなわち 4000-8000Hz に、集中した場合に起きる。この状態を検出するため、パワー尺度が、現フレームの低域および高域合成信号に対して算出される。具体的には、低域信号のパワー  $\hat{s}_{LB}(n)$  が、下式で与えられ、

$$P_{LB} = \sum_{n=0}^{39} |\hat{s}_{LB}(n)| \quad (8-82)$$

高域信号のパワー  $\hat{s}_{HB}(n)$  は、下式で与えられる。

$$P_{HB} = \sum_{n=0}^{39} |\hat{s}_{HB}(n)| \quad (8-83)$$

レイヤ 2 が復号されない場合、すなわち信号が狭帯域の場合、ノイズゲートは常時動作し、その減衰効果は、算出したターゲットゲインの値に依存する。一方、レイヤ 2 が復号され、かつ以下の条件、

$$P_{HB} > \frac{800}{2^{31}-1} \quad \text{and} \quad P_{HB} > 16P_{LB} \quad (8-84)$$

が満たされる場合、ターゲットゲインは  $g_t = 1.0$  に設定し、ノイズゲートを徐々に停止させる（開ける）。最終的に、出力合成信号  $\hat{s}_{QMF}(n)$  のそれぞれのサンプルは、下記ゲインを乗算することで得られる。

$$g_{NG}^{(n)} = 0.9893g_{NG}^{(n-1)} + 0.0107g_t \quad n = \begin{cases} 0, \dots, 39 & \text{for narrowband output} \\ 0, \dots, 79 & \text{for wideband output} \end{cases} \quad (8-85)$$

上式は、サンプル毎に更新される。ゲイン  $g_{NG}^{(-1)}$  は、1.0 に初期化される。このゲインは、ターゲットゲイン  $g_t$  にゆっくりと収束する。このようにして、広帯域出力に対する出力信号は、以下のように更新される。

$$\hat{s}_{WB}(n) = \begin{cases} g_{NG}^{(n)} \hat{s}_{QMF}(n) & \text{if } g_{NG}^{(n)} < 1.0 \\ \hat{s}_{QMF}(n) & \text{otherwise} \end{cases} \quad n = 0, \dots, 79 \quad (8-86)$$

同様に、狭帯域出力は以下のように更新される。

$$\hat{s}_{NB}(n) = \begin{cases} g_{NG}^{(n)} \hat{s}_{LB}(n) & \text{if } g_{NG}^{(n)} < 1.0 \\ \hat{s}_{LB}(n) & \text{otherwise} \end{cases} \quad n = 0, \dots, 39 \quad (8-87)$$

## 9. 伝送パラメータインデックスの詳細

R3 モードのビットストリーム順序は、Table 1 4 / JT-G711.1 に降順で示される。それぞれのパラメータは、

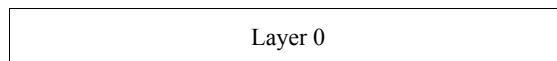
MS Bが最初に送出される。

Table 1 4 / JT-G711.1 Description of transmitted parameter indices in R3 mode  
(ITU-T G.711.1)

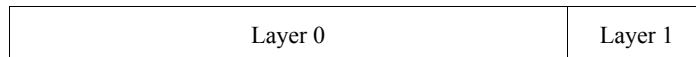
Layer	Symbol	Description	Bits
Layer 0	$I_{L0}(0)$	Lower-band core (G.711 compatible) – 1st sample in frame	8
	$I_{L0}(1)$	Lower-band core (G.711 compatible) – 2nd sample in frame	8
	...	...	...
	$I_{L0}(39)$	Lower-band core (G.711 compatible) – 40th sample in frame	8
Layer 1	$I_{L1}(0)$	Lower-band enhancement bit(s) – 1st sample in frame	$0 \leq B_A(0) \leq 3$
	$I_{L1}(1)$	Lower-band enhancement bit(s) – 2nd sample in frame	$0 \leq B_A(1) \leq 3$
	...	...	...
	$I_{L1}(39)$	Lower-band enhancement bit(s) – 40th sample in frame	$0 \leq B_A(39) \leq 3$
Layer 2	$I_{Hp0}(0)$	Higher-band MDCT sign index – 1st subvector	1
	$I_{Hs0}(0)$	Higher-band MDCT vector code – 1st subvector	5
	$I_{Hp1}(0)$	Higher-band MDCT sign index – 1st subvector	1
	$I_{Hs1}(0)$	Higher-band MDCT vector code – 1st subvector	5
	$I_{Hp0}(1)$	Higher-band MDCT sign index – 2nd subvector	1
	$I_{Hs0}(1)$	Higher-band MDCT vector code – 2nd subvector	5
	$I_{Hp1}(1)$	Higher-band MDCT sign index – 2nd subvector	1
	$I_{Hs1}(1)$	Higher-band MDCT vector code – 2nd subvector	5
	...	...	...
	$I_{Hp0}(5)$	Higher-band MDCT sign index – 6th subvector	1
	$I_{Hs0}(5)$	Higher-band MDCT vector code – 6th subvector	5
	$I_{Hp1}(5)$	Higher-band MDCT sign index – 6th subvector	1
	$I_{Hs1}(5)$	Higher-band MDCT vector code – 6th subvector	5
	$I_{Hg}$	Higher-band MDCT gain	8

各フレーム毎にレイヤは、Figure 9 – 1 / JT-G711.1 に示される順番で配置される。

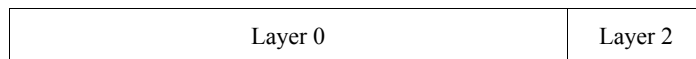
R1 mode



R2a mode



R2b mode



R3 mode



Figure 9 – 1 / JT-G711.1 Order for layer placement for the various modes  
(ITU-T G.711.1)

## 10. JT-G711. 1コーデックのビットイグザクト記述

16 ビット固定小数点 JT-G711. 1 コーデックをシミュレートする ANSI C コードは、日本 ITU 協会から入手可能である。以下の節では、このシミュレーションコードの使い方と、ソフトウェアがどのように構成されているかをまとめている。

### 10.1 シミュレーションソフトウェアの使い方

C コードは、2つのメインプログラム `encoder.c` と `decoder.c` とで構成されており、それぞれ、メイン符号化器と復号器である。

符号化器に対するコマンドラインは、以下の通りである。

**encoder [-options] <law> <infile> <codefile>**

where

`law` is the desired JT-G.711 law (A or u)  
`infile` is the name of the input file to be encoded  
`codefile` is the name of the output bitstream file

Options:

`-mode #` encoder mode (1, 2, 3, 4).  
Mode number 1, 2, 3, 4 correspond to R1, R2a, R2b, R3 respectively.  
When not specified, R3 will be generated by default  
`-nb` narrowband input  
`-hardbit` Output bitstream file is in multiplexed hardbit format.  
`-quiet` quiet processing

復号器に対するコマンドラインは、以下の通りである。

**decoder [-options] <law> -mode <modenum> <codefile> <outfile>**

where

`law` is the desired JT-G.711 law (A or u)  
`modenum` is the mode of the bitstream file (1, 2, 3, 4)  
Mode number 1, 2, 3, 4 correspond to R1, R2a, R2b, R3 respectively.

`codefile` is the name of the input bitstream file  
`outfile` is the name of the decoded output file

Options:

`-decmode #` is to specify the decoder mode (1, 2, 3, 4).  
Mode number 1, 2, 3, 4 correspond to R1, R2a, R2b, R3 respectively.  
When not specified, it will be the same as `modenum` (mode of the bitstream file).  
`-hardbit` Input bitstream file is in multiplexed hardbit format.  
`-quiet` quiet processing

符号化器の入力と復号器の出力ファイルは、16 ビット PCM 信号を含む標準化データファイルである。符号化器の出力と復号器の入力ファイルは、デフォルトでは、ITU-T G.192 ビットストリームフォーマットに準拠する。符号化されたビットストリームのマッピングテーブルは、シミュレーションソフトウェアに含まれている。

## 10.2 シミュレーションソフトウェアの構成

Table 1 5/JT-G711.1 から Table 1 8/JT-G711.1 に、シミュレーションソフトウェアの構成を記述する。

Table 1 5/JT-G711.1 Tables in C-code

(ITU-T G.711.1)

Table name	Symbol	Size	Format	Description
sQmf0	$h_0^{qmf}(i)$	16	Q15	QMF filter coefficients (polyphase representation)
sQmf1	$h_1^{qmf}(i)$	16	Q15	QMF filter coefficients (polyphase representation)
NS_window	$w_{LP1}(i)$	80	Q15	Window for LPC analysis in noise shaping
NS_lag_h,	$w_{lag}(i)$	4	Q15	Lag windowing in noise shaping (MSB of double precision format)
NS_lag_l		4	Q16	Lag windowing in noise shaping (LSB of double precision format)
LBFEC_lag_h	$w_{lag}^{FERC}(i)$	16	Q15	Lag windowing in low-band FERC (MSB and LSB of double precision format)
LBFEC_lag_l		16	Q16	Lag windowing in low-band FERC (LSB of double precision format)
LBFEC_lpc_win_80	$w_{LP2}(i)$	80	Q15	Window for LPC analysis in low-band FEC
LBFEC_fir_lp	$H_{dec}(z)$	9	Q16	FIR decimation filter coefficients in low-band FERC
MDCT_xcos	-	25	Q15	Cosine table for FFT
MDCT_xsin	-	25	Q15	Sine table for FFT
MDCT_tab_map	-	10	Q0	Index mapping table for Good-Thomas FFT
MDCT_tab_map2	-	10	Q0	Index mapping table for Good-Thomas FFT
MDCT_tab_rev_ipp	-	1	Q0	Table for Good-Thomas FFT
MDCT_tab_rev_i	-	1	Q0	Table for Good-Thomas FFT
MDCT_rw1	-	20	Q15	FFT twiddle factors (cosine part)
MDCT_rw2	-	20	Q15	FFT twiddle factors (sine part)
MDCT_wcos	$W_{80}^n$	20	Q15	Cosine table for MDCT and iMDCT
MDCT_wsine		20	Q15	Sine table for MDCT and iMDCT
MDCT_wetr	$\frac{(-1)^{k+1}W_8^{-1}W_{320}^{4k+1}}{80}$	20	Q21	Table for complex post-multiplication in MDCT (real part)
MDCT_weti		20	Q21	Table for complex post-multiplication in MDCT (imaginary part)
MDCT_wetrm1	-	20	Q14	Table for complex pre-multiplication in iMDCT (real part)
MDCT_wetim1	-	20	Q14	Table for complex pre-multiplication in iMDCT (imaginary part)
MDCT_h	$w_{TDAC}(i)$	80	Q14	MDCT window
gsCodebook_0ch	$C_{H0w}(i,j)$	32*6	Q12	Interleave CSVQ sub-codebook 0
gsCodebook_1ch	$C_{H1w}(i,j)$	32*6	Q12	Interleave CSVQ sub-codebook 1
glCodebook_0ch_pow	$\sigma_{H0w}^2(i)$	32 (x2)	Q22	Power-term for CSVQ pre-selection
glCodebook_1ch_pow	$\sigma_{H1w}^2(i)$	32 (x2)	Q22	Power-term for CSVQ pre-selection



Table name	Symbol	Size	Format	Description
gsCodebook_cross	$R_{H0H1}(i_0, i_1)$	32*32	Q9	Cross-terms for CSVQ main selection
max_err_quant	$d'_{\max}(j)$	16	Q0	A-law quantization step size
Hann_sh16	$w_a(n)$	64	Q15	Asymmetric Hanning window (for 64-point FFT processing)
Hann_sh16_p6	$\gamma_w(n)$	7	Q15	Half of 16-point Hanning window for filter interpolation
Hann_sh16_p6m1	$1 - \gamma_w(n)$	7	Q15	Complementary half of Hanning window for filter interpolation
r	-	64	Q0	Index mapping table for FFT
w	-	16	Q15	Twiddle factors for FFT
WinFilt	$W_{t_p}(n)$	17	Q15	Truncating window for impulse response of noise reduction filter
Table_sqrt_w	-	49	Q15	Table for square-root operation
Table_isqrt	-	49	Q15	Table for inverse square-root operation

Table 1 6 / JT-G711.1 Summary of encoder specific routines  
(ITU-T G.711.1)

Filename	Description
encoder.c	G.711.1 encoder interface
G711wbeenc.c	G.711.1 main encoder
prehpf.c	High-pass pre-filter
lowband_enc.c	Lower-band encoder
highband_enc.c	Higher-band encoder
mulaw.c	Mu-law compression of higher-band gain
norm_spectrum.c	MDCT spectrum normalization by RMS
vq_mainselect.c	CSVQ main selection routine
vq_preselect.c	CSVQ pre-selection routine
vqenc_spectrum.c	Interleave CSVQ spectrum coding

Table 1 7 / JT-G711.1 Summary of decoder specific routines  
(ITU-T G.711.1)

Filename	Description
decoder.c	G.711.1 decoder interface
g711wbedec.c	G.711.1 main decoder
fec_lowband.c	Frame erasure concealment (FERC) for lower-band signal
lowband_dec.c	Lower-band decoder
fec_highband.c	Frame erasure concealment (FERC) for higher-band signal
highband_dec.c	Higher-band decoder
mulawinv.c	Inverse mu-law compression of higher-band gain
vqdec_spectrum.c	Interleave CSVQ spectrum decoding

Table 1 8 / JT-G711.1 Summary of common routines  
(ITU-T G.711.1)

Filename	Description
qmfilt.c	QMF filterbank
softbit.c	Conversion between hardbit and softbit
table_qmfilt.c	Tables for QMF filterbank
autocorr_ns.c	Autocorrelation of signal for noise shaping
g711a.c	Embedded PCM coder and decoder (A-law)
g711mu.c	Embedded PCM coder and decoder (mu-law)
lpctools.c	Linear prediction tools
table_lowband.c	Tables for lower-band modules
cfft.c	20-point complex FFT/iFFT
mdct.c	80-point MDCT and iMDCT
mux_bits.c	Multiplexing/demultiplexing of higher-band parameters
table_highband.c	Tables for higher-band coding
table_mdct.c	Tables for MDCT computation
dsputil.c	Fixed-point utility routines
errexit.c	Exit routine
mathtool.c	Square-root routines
oper_32b.c	Basic operators in double precision (32 bits)
table_mathtool.c	Tables for square-root routines

## 付属資料A

(標準 J T - G 7 1 1 . 1 に対する)

### 浮動小数点実装

本付属資料は、J T - G 7 1 1 . 1 符号化アルゴリズムの浮動小数点実装を記述する。

#### A. 1 アルゴリズム記述

標準 J T - G 7 1 1 . 1 の本浮動小数点版は、固定小数点版と同様のアルゴリズムステップを踏んでいる。同様にビットストリームは標準 J T - G 7 1 1 . 1 のビットストリームと同一のものとなっている。アルゴリズムの詳細に関しては、標準 J T - G 7 1 1 . 1 本体を参照すること。

#### A. 2 ANS I Cコード

本付属資料にて定義されている標準 J T - G 7 1 1 . 1 浮動小数点版をシミュレートする ANS I Cコードは、日本 I T U協会から入手可能である。現在の ANS I Cソースコードのバージョンは、2008年9月18日付け、バージョン1.0である。浮動小数点ソースコードの構成は、対応する固定小数点ソースコードと関連付けられている。Table A. 1/JT-G711.1 から Table A. 4/JT-G711.1 に、ソフトウェアのファイル名をその簡潔な記述と共に示す。基本演算子と算術演算に関連するファイルは、浮動小数点演算には用いられていない。浮動小数点関連のルーチンのセットは floatutil.c として追加されている。

Table A. 1/JT-G711.1 Summary of encoder specific routines  
(ITU-T G711.1)

Filename	Description
encoder.c	G.711.1 encoder interface
g711wbeenc.c	G.711.1 main encoder
prehpf.c	High-pass pre-filter
lowband_enc.c	Lower-band encoder
highband_enc.c	Higher-band encoder
mulaw.c	Mu-law compression of higher-band gain
norm_spectrum.c	MDCT spectrum normalization by RMS
vq_mainselect.c	CSVQ main selection routine
vq_preselect.c	CSVQ pre-selection routine
vqenc_spectrum.c	Interleave CSVQ spectrum coding

Table A. 2/JT-G711.1 Summary of decoder specific routines  
(ITU-T G711.1)

Filename	Description
decoder.c	G.711.1 decoder interface
g711wbedec.c	G.711.1 main decoder
fec_lowband.c	Frame erasure concealment (FERC) for lower-band signal
lowband_dec.c	Lower-band decoder
fec_highband.c	Frame erasure concealment (FERC) for higher-band signal
highband_dec.c	Higher-band decoder

Filename	Description
mulawinv.c	Inverse mu-law compression of higher-band gain
vqdec_spectrum.c	Interleave CSVQ spectrum decoding

Table A. 3 / JT-G711.1 Summary of common routines

(ITU-T G.711.1)

Filename	Description
qmfilt.c	QMF filter bank
softbit.c	Conversion between hardbit and softbit
table_qmfilt.c	Tables for QMF filter bank
autocorr_ns.c	Autocorrelation of signal for noise shaping
g711a.c	Embedded PCM coder and decoder (A-law)
g711mu.c	Embedded PCM coder and decoder (mu-law)
lpctools.c	Linear prediction tools
table_lowband.c	Tables for lower-band modules
cfft.c	20-point complex FFT/iFFT
mdct.c	80-point MDCT and iMDCT
mux_bits.c	Multiplexing/demultiplexing of higher-band parameters
table_highband.c	Tables for higher-band coding
table_mdct.c	Tables for MDCT computation
dsputil.c	Fixed-point utility routines
errexit.c	Exit routine
<b>floatutil.c</b>	<b>Floating point specific routines</b>
mathtool.c	Square-root routines
oper_32b.c	Basic operators in double precision (32 bits)
table_mathtool.c	Tables for square-root routines

Table A. 4 / JT-G711.1 Summary of Appendix I specific routines

(ITU-T G.711.1)

Filename	Description
post.c	Main routine that calls all post-processing subroutines
post_anasyn.c	Analysis/synthesis subroutines for post-processing
post_gainfct.c	Subroutines for estimating of the post-processing filter
post_rfft.c	64-point real FFT and iFFT
table_post.c	Tables for postfilter

## 付属資料B

(標準 JT-G 7 1 1. 1 に対する)

JT-H 2 4 5 における JT-G 7 1 1. 1 使用法

### B. 1 本付属資料の規定範囲

本付属資料は、JT-G 7 1 1. 1 オーディオコーデックと JT-H 2 4 5 仕様に対する RTP ペイロードフォーマット、及び機能シグナリングを定義する。シームレスな互換性を許可するために、フォーマット及び機能パラメータはいずれも、対応する JT-G 7 1 1. 1 RTP 定義と完全互換である。

### B. 2 参考文献

- [1] IETF RFC 3550 (2003), RTP: A Transport Protocol for Real-Time Applications.
- [2] IETF RFC 5391 (2008), RTP Payload format for ITU-T Recommendation G.711.1.

### B. 3 JT-G 7 1 1. 1 フレームに対するパケット構造

JT-G 7 1 1. 1 に対する RTP ペイロードフォーマットは IETF RFC 5391 において指定される。このペイロードフォーマットはコーデックのマルチレート機能をサポートし、“*ModeIndex (MI)*” を伝達する追加のペイロードヘッダにより、パケット単位のレート変更を可能にする。このパラメータは後続するオーディオフレームのモードを記述する。Table B. 1/JT-G711.1 に、本標準の Table 3/JT-G711.1 において定義されるビットレートモードに対応する *MI* 値を示す。

Table B. 1/JT-G711.1 G.711.1 Modes in Mode Index payload header  
(ITU-T G.711.1)

Mode Index	G.711.1 bitrate mode
1	R1
2	R2a
3	R2b
4	R3

*MI* において、その他の全ての値は使用されるべきではなく、未定義の *MI* 値を持つペイロードは廃棄されるべきである。もし、シグナリングにより、制限されたモードセットが設定されている場合には、JT-G 7 1 1. 1 機能パラメータにおいて示されるように、このセットに含まれない *MI* 値を持つ受信されたペイロードは廃棄されるべきである。

### B. 4 JT-H 2 4 5 に対する JT-G 7 1 1. 1 機能定義

JT-G 7 1 1. 1 機能は、JT-H 2 4 5 の付録 7 以降の JT-H 2 4 5 一般機能として定義される。Table B. 2/JT-G711.1 及び Table B. 3/JT-G711.1 は、JT-G 7 1 1. 1 コア符号化則に基づいて JT-G 7 1 1. 1 機能識別子を定義している。2 つの JT-G 7 1 1. 1 機能パラメータは、Table B. 4/JT-G711.1 及び Table B. 5/JT-G711.1 において定義される。これら 2 つの機能パラメータは、どちらのコア符号化則 ( $\mu$  則、及び A 則) の機能識別子に対しても適用される。

Table B. 2/JT-G711.1 G.711.1 ( $\mu$ -law core) capability identifier  
(ITU-T G.711.1)

Capability name	G.711.1Ulaw
Capability class	Audio
Capability identifier type	Standard
Capability identifier value	{ itu-t (0) recommendation (0) g (7) g711(711) dot(1) part1(1) generic-capabilities (1) U-law (0) 0 }
maxBitRate	Shall be present. Possible values are 640, 800, and 960 (units 100 bits/s). This value must be consistent with the capability parameter “modeSet”, if indicated. The general value for maxBitRate is 960 (= 96 kbps).
collapsing	The field shall contain the G.711.1 capability parameters from Table[X].4.
nonCollapsing	This field shall contain the G.711.1 capability parameters from Table[X].5.
nonCollapsingRaw	This field shall not be included.
Transport	This field shall not be included.

Table B. 3 / JT-G711.1 G.711.1 (A-law core) capability identifier  
(ITU-T G.711.1)

Capability name	G.711.1Alaw
Capability class	Audio
Capability identifier type	Standard
Capability identifier value	{ itu-t (0) recommendation (0) g (7) g711(711) dot(1) part1(1) generic-capabilities (1) A-law (1) 0 }
maxBitRate	Shall be present. Possible values are 640, 800, and 960 (units 100 bits/s). This value must be consistent with the capability parameter “modeSet”, if indicated. The general value for maxBitRate is 960 (= 96 kbps).
collapsing	The field shall contain the G.711.1 capability parameters from Table[X].4.
nonCollapsing	This field shall contain the G.711.1 capability parameters from Table[X].5.
nonCollapsingRaw	This field shall not be included.
Transport	This field shall not be included.

Table B. 4 / JT-G711.1 G.711.1 capability parameter - maxAL-sduAudioFrames  
(ITU-T G.711.1)

Parameter name	maxAL-sduAudioFrames
Parameter description	This is a collapsing GenericParameter. It indicates the maximum number of audio frames per AL-SDU.
Parameter identifier value	1
Parameter status	Optional
Parameter type	unsignedMin
Supersedes	-

Table B. 5 / JT-G711.1 G.711.1 capability parameter - modeSet  
(ITU-T G.711.1)

Parameter name	modeSet
Parameter description	This is a nonCollapsing GenericParameter. It indicates a supported bitrate mode index from Table B.1. Each parameter can appear several times.
Parameter identifier value	2
Parameter status	Optional. If this parameter is omitted, it means that one is capable of all modes. Multiple instances of this parameter may be included, specifying the supported bitrate modes in preference order. Bitrate modes that exceed maxBitRate should not be signalled.
Parameter type	unsignedMin
Supersedes	-

#### B. 5 JT-G711. 1に対する互換性

JT-G711. 1のコアはJT-G711に対して完全互換であり、またトランスコードの労力は最小化され得る。基本的なJT-G711機能は、JT-H245 *AudioCapability* 構造において定義されている。 $\mu$  則コアにおけるJT-G711. 1は、*g711Ulaw64k* に対して互換であり、A 則コアにおけるJT-G711. 1は、*g711Alaw64k* に対して互換である。唯一の制約は、JT-G711の8kHz サンプリング周波数におけるサンプル毎の符号化の代わりに、JT-G711. 1は5ms フレーム長で符号化することである。

望ましい順番としては、JT-G711. 1はより良い品質を提供出来るため、JT-G711の前に現れるべきである。

RFC 5391において定義されるRTPペイロードフォーマットは、既存のJT-G711エンドポイントによっては理解されないため、2つのJT-G711. 1エンドポイント間の通信にのみ使用されるべきであるということによく注意されたい。

## 付録 I

(標準 J T - G 7 1 1 . 1 に対する)

R 1 モードに対する低域ポストフィルタ

(この付録は、本標準を構成する必須部分ではない)

量子化ノイズポストフィルタ処理は、既存の G . 7 1 1 ビットストリームと互換性のあるレイヤ 0 だけが復号された時に、低域における P C M 量子化ノイズを低減するために、オプションとして処理される。このポストフィルタ処理の主眼は、従来の G . 7 1 1 符号化器で符号化された信号に含まれる可聴量子化ノイズを低減することにある。ここで、タンデム接続されるような場合、例えば会議ブリッジ接続のような場合、このポストフィルタ処理の使用は、避けた方がよいことに注意されたい。このような場合でも、エンドユーザ端末の復号器では、オプションとして使用されても良い。これは、このような端末では、13 ないしは 14 ビットへのビット切り捨てをすることなく、フル 16 ビットのリニア P C M 解像度 A D 変換が使用されていると思われるからである。

このポストフィルタ処理への入力信号は、64kbit/s (4kHz 帯域) のエンベデッド P C M コーデックによって復号された信号である。ここで、このポストフィルタは、R 1 モードに対して適用可能であり、 $\hat{s}_{LB}(n)$  は (8-3) ではなく、(8-1) で算出されていなければならないことに注意されたい。はじめに、G . 7 1 1 の性質についての先験的な情報に基づき、符号化ノイズの性質が決定される。そして、時変フィルタが計算される。最後に、P C M 量子化ノイズを低減するために、復号信号をこのフィルタに適用することで、改善された信号が得られる。

Figure I - 1 / J T - G 7 1 1 . 1 に示したように、連続フレームにおけるノイズ低減方法は、以下のプロセスを適用することで構成されている。

- 入力信号  $\hat{s}_{LB}(n)$  のスペクトル  $\hat{S}_{LB}(k)$  が、周波数領域への変換によって算出される (I.1 章を参照)。
- 周波数依存ノイズパワーのスペクトル密度 ( P S D ) 指標  $|N_{LB}(k)|^2$  が、現フレームのロードファクタ  $\Gamma_{LB}$  を評価した後に得られる (I.2 章を参照)。
- ノイズ低減フィルタの伝達関数  $W_{p2}(k)$  が、これらパワースペクトル密度を用いて得られる。この計算は、2 段階処理に基づいている (I.3 章を参照)。
- 最後に、この伝達関数に基づくブロックフィルタ処理 ( $w_p$ ) とオーバーラップ/セーブ ( O L S ) を、P C M 復号信号  $\hat{s}_{LB}(n)$  へ適用することで、量子化ノイズを低減した信号  $\hat{s}'_{LB}(n)$  を導出する (I.4 章を参照)。

この処理は、G . 7 1 1 の両圧伸則に共通である。

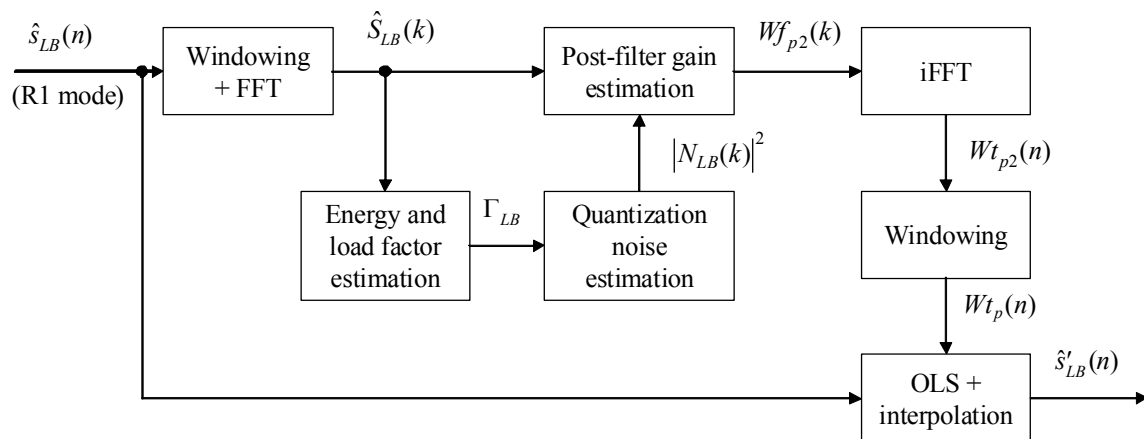




Figure I - 1 / JT-G711.1 Block diagram of R1 Postfilter  
(ITU-T G.711.1)

## I. 1 窓掛けとFFT

ポストフィルタは、高速フーリエ変換（FFT）を用いて、周波数領域で推定される。このFFTは、復号信号 $\hat{s}_{LB}(n)$ 、 $n=-24, \dots, 39$ の64サンプル・ブロックで算出される。負のインデックスは、過去の信号（範囲 $-24, \dots, -1$ は、過去フレームの最も新しいサンプル列を表している）を参照していることに注意されたい。64サンプル・ブロック $\hat{s}_{LB}(n)$ 、 $n=-24, \dots, 39$ は、現フレームの40サンプルと前フレームの24サンプルから成っており、最も新しいサンプルを重視した64点非対称ハニング窓（ $3/4 - 1/4$ ） $w_a(n)$ による窓掛け処理後、64周波数成分のブロック $\hat{S}_{LB}(k)$ 、 $k=0, \dots, 63$ へ変換される。

$$\begin{cases} w_a(n) = 0.5 - 0.5 \cos\left(\frac{\pi n}{48}\right) & n = 0, \dots, 47 \\ w_a(n) = 0.5 + 0.5 \cos\left(\frac{\pi(n-47)}{16}\right) & n = 48, \dots, 63 \end{cases} \quad (\text{I-1})$$

$$\hat{S}_{LB}(k) = \text{FFT}\{\hat{s}_{LB}(n-24) \cdot w_a(n)\} \quad (\text{I-2})$$

フーリエ変換の対称性（入力データが実数）から、 $k=0, \dots, 32$ の前半の周波数成分のみ考慮される。

入力信号が実数であることを考慮し、低演算量FFT処理が用いられる。その出力スペクトラムは、 $[\text{Re}(0) \text{ Re}(1) \dots \text{Re}(32) \text{ Im}(31) \dots \text{Im}(1)]$ の順に保存される。

## I. 2 量子化ノイズのPSD

G. 711の符号化特性から、信号対量子化ノイズ比（SNR）の推定、そしてその量子化ノイズPSDの推定は、ロードファクタ $\Gamma_{LB}$ から行うことができる。そのロードファクタ $\Gamma_{LB}$ は、符号化前の入力信号 $s_{LB}(n)$ の分散の平方根の逆数として定義され、量子化ノイズPSD $|N_{LB}(k)|^2$ の推定値は、この信号の分散から直接導出される。しかしながら、多くの場合、量子化ノイズの分散は、符号化前の信号の分散に比べて非常に小さいため、このポストフィルタでは、入力信号 $s_{LB}(n)$ の分散は、復号器で利用可能なPCM復号信号 $\hat{s}_{LB}(n)$ の分散と等しい、と仮定する。

上記仮定に基づき、ノイズPSD $|N_{LB}(k)|^2$ は、以下のように、PCM復号信号 $\hat{s}_{LB}(n)$ の分散から推定される。

1. PCM復号信号の現フレームエネルギー $E_{LB}$ は、下式で算出される。

$$E_{LB} = \sum_{n=0}^{39} \hat{s}_{LB}^2(n) \quad (\text{I-3})$$

2. このフレームエネルギーからロードファクタ $\Gamma_{LB}$ が、以下のように算出される。

$$\Gamma_{LB}^2 = \frac{40}{E_{LB}} \quad (\text{I-4})$$

3. このロードファクタ $\Gamma_{LB}$ は、量子化器の線形領域、ないしは対数領域において符号化された信号かを判定するための閾値と比較される。このため、量子化ノイズのPSD $|N_{LB}(k)|^2$ は、以下のように算出される。

- $\Gamma_{LB}^2 < 4.8018 \cdot 10^2$ （すなわち  $E_{LB} > 8.3303 \cdot 10^2$ ）の場合、信号は、量子化器の対数領域で符号化されていて、 $|N_{LB}(k)|^2$ は下式で算出される。

$$|N_{LB}(k)|^2 = 1.5276 \cdot 10^{-4} \cdot E_{LB} \quad (I-5)$$

- $10^5 \geq \Gamma_{LB}^2 > 4.8018 \cdot 10^2$  (すなわち  $4 \cdot 10^{-4} \leq E_{LB} < 8.3303 \cdot 10^{-2}$ ) の場合、信号は、量子化器の線形領域で符号化されていて、 $|N_{LB}(k)|^2$  は下式で算出される。

$$|N_{LB}(k)|^2 = 3.1789 \cdot 10^{-7} \quad (I-6)$$

- $\hat{\Gamma}_{LB}^2 > 10^5$  (すなわち  $E_{LB} < 4 \cdot 10^{-4}$ ) の場合、信号は、量子化器の線形領域で符号化されているが、(I-6)式ではよい推定にならない。この事象は、ロードファクタが最も大きい(すなわち、フレームエネルギーが最も小さい)時に発生する。この場合、量子化ノイズの分散は、信号分散から 15dB 低い値に設定される。

$$|N_{LB}(k)|^2 = 3.1623 \cdot 10^{-2} \cdot E_{LB} \quad (I-7)$$

注記：ANSI C コードにおいて時間のかかる除算を回避するため、ロードファクタは計算せず、フレームエネルギーとの比較によって(上述した対応する比較による表現として)処理される。

### I. 3 ポストフィルタ処理

#### I.3.1 周波数領域におけるポストフィルタ処理

推定されたノイズの分散に従い、その量子化ノイズを低減するフィルタが構成される。このフィルタ処理は、ウィナーフィルタを2段階で推定する手順を用いた短期間スペクトル減衰手法によって得られる。

量子化ノイズの推定 PSD  $|N_{LB}(k)|^2$  は、それぞれの周波数成分において同一ノイズレベルである。このノイズスペクトル形状における他の情報が無い場合、ウィナーフィルタは、そのノイズが白色であると仮定して動作する。しかしながら、この処理は、ノイズが白色であるという仮定から多少はずれても動作する。さらに、帰納的制御によって、過剰な減衰を避け、推定エラーによる歪みを制限している。

最初の段階では、それぞれの周波数成分に対する帰納的 SNR  $\text{SNR}_{post}(k)$  は、以下のように推定される。

$$\text{SNR}_{post}(k) = \frac{|\hat{S}_{LB}(k)|^2}{|N_{LB}(k)|^2} \quad k = 0, \dots, 32 \quad (I-8)$$

ここで、除算回数を制限することで演算量を削減するために、ANSI C コードにおいては、この SNR は  $|N_{LB}(k)|^2$  で仮想的に乗算する。その結果、上式は下式となる。

$$\text{SNR}_{post}(k) = |\hat{S}_{LB}(k)|^2 \quad k = 0, \dots, 32 \quad (I-9)$$

したがって、以下で用いる SNR の値は、正確な信号対ノイズ比には対応していない。このようにして、インデックス  $m$  の現フレームにおける最初の段階の演繹的 SNR  $\text{SNR}_{prior}^{(m)}(k)$  は、前フレームで計算した最初の所望信号推定  $\tilde{S}_1^{(m-1)}(k)$  を用い、下式 ( $\beta = 0.98$  として) により推定される (式 I-15 を参照)。

$$\text{SNR}_{prior}^{(m)}(k) = \beta \cdot |\tilde{S}_1^{(m-1)}(k)|^2 + (1 - \beta) \max \left[ \text{SNR}_{post}^{(m)}(k) - |N_{LB}^{(m)}(k)|^2, 0 \right] \quad k = 0, \dots, 32 \quad (I-10)$$

ここで、前フレームで推定された変数  $\tilde{S}_1^{(m-1)}(k)$  の時間依存性から、ここに記載したフレームインデックス  $m$  と  $(m-1)$  は、本式のみで使用される。よって、より明確にするために、他式において表記する現フレーム  $m$  の変数、例えば式 I-10 における  $\text{SNR}_{post}(k) = \text{SNR}_{post}^{(m)}(k)$  のように、フレームインデックスは以降省略する。このようにして、最初のノイズ低減フィルタの伝達関数  $Wf_{p1}(k)$  は、最初の段階の演繹的 SNR  $\text{SNR}_{prior}(k)$  を

用い、周波数領域で計算される。

$$Wf_{p1}(k) = \frac{\text{SNR}_{prio1}(k)}{|N_{LB}(k)|^2 + \text{SNR}_{prio1}(k)} \quad k = 0, \dots, 32 \quad (\text{I-11})$$

第二段階目の所望信号  $\tilde{S}_2(k)$  は、入力信号  $\hat{S}_{LB}(k)$  のスペクトルと、最初のノイズ低減フィルタ  $Wf_{p1}(k)$  の伝達関数との合成により、周波数領域で計算される。

$$|\tilde{S}_2(k)| = |\hat{S}_{LB}(k)| \cdot |Wf_{p1}(k)| \quad k = 0, \dots, 32 \quad (\text{I-12})$$

第二段階における、演繹的 SNR 推定は、所望信号としての  $\tilde{S}_2(k)$  を用いて改善される。

$$\text{SNR}_{prio2}(k) = |\tilde{S}_2(k)|^2 \quad k = 0, \dots, 32 \quad (\text{I-13})$$

そして、第二段階のノイズ低減フィルタ  $Wf_{p2}(k)$  の伝達関数は、第二段階の演繹的 SNR  $\text{SNR}_{prio2}(k)$  を用い、周波数領域で計算される。しかしながら、このフィルタは、特定の周波数成分を過剰に減衰させるかもしれない。そこで、 $Wf_{p2}(k)$  が 6dB 以下となるのを避けるよう、閾値が適用される。

$$Wf_{p2}(k) = \max \left( \frac{\text{SNR}_{prio2}(k)}{|N_{LB}(k)|^2 + \text{SNR}_{prio2}(k)}, 10^{-\frac{6}{20}} \right) \quad k = 0, \dots, 32 \quad (\text{I-14})$$

この処理のこの段階において、現フレームインデックス  $m$  に対する周波数領域のポストフィルタ  $Wf_{p2}(k)$  の計算が完了する。

しかしながら、次フレーム（フレーム  $m+1$ ）において使用する最初の段階の演繹的 SNR を推定するために、第一の所望信号  $\tilde{S}_1^{(m)}(k)$ （これはすなわち式(I-10)における  $\tilde{S}_1^{(m-1)}(k)$ ）が計算され、これは  $\hat{S}_{LB}(k)$  と  $Wf_{p2}(k)$  を乗算することで得られる。

$$|\tilde{S}_1^{(m)}(k)| = |\hat{S}_{LB}(k)| \cdot |Wf_{p2}(k)| \quad k = 0, \dots, 32 \quad (\text{I-15})$$

### 1.3.2 時間領域におけるポストフィルタ処理

フィルタ  $Wf_{p2}(k)$  は実数である（位相情報は処理されないものとする）。時間領域のフィルタ係数を算出するために、以下のアルゴリズムは、虚部をゼロとした  $Wf_{p2}(k)$  の逆 FFT で処理する。

$$Wt_{p2}(n) = \text{iFFT} \{ Wf_{p2}(k) \} \quad k = 0, \dots, 32 \quad n = 0, \dots, 63 \quad (\text{I-16})$$

$Wf_{p2}(k)$  が実数なので、下記の通り係数を交換することにより、時間領域フィルタの因果性が、確保される。

$$Wt_c(n) = \begin{cases} Wt_{p2}(n+32) & n = 0, \dots, 31 \\ Wt_{p2}(n-32) & n = 32, \dots, 63 \end{cases} \quad (\text{I-17})$$

そして、時間インパルス応答  $Wt_p(n)$  は、33 タップに切り詰められ、対称ハニング窓により重み付けられ、結果として 16 サンプル遅延の 33 タップ FIR フィルタが得られる。

$$Wt_p(n) = \left( 0.5 - 0.5 \cos \left( \frac{\pi n}{16} \right) \right) \cdot Wt_c(n+15) \quad n = 0, \dots, 32 \quad (\text{I-18})$$

入力信号スペクトルが実数かつ対称であり、それ故その出力も実数であることを考慮し、非常に低演算量の逆

F F T 処理が適用される。入力スペクトルが実数かつ対称であることから、33 実数入力値のみが必要とされ、残りの 31 値は、虚部をゼロとして、ミラーリングによって生成される。その出力は、33 値の実時間領域の信号である。

#### I. 4 時間領域ポストフィルタ処理（補間付き時間 O L S 処理）

最後に、P C M 復号信号  $\hat{s}_{LB}(n)$  は、量子化ノイズを低減した信号  $\hat{s}'_{LB}(n)$  を生成するために、ノイズ低減フィルタ  $\tilde{W}_{t_p}$  によって、フィルタ処理される。

時間領域における O L S 手法が、このフィルタ処理を実行するために利用される。加えて、このフィルタ  $\tilde{W}_{t_p}$  は、フレーム毎に変わるので、聴感上知覚される非連続音が生成される可能性がある。このため、フィルタ毎の変化が、平滑化される。現フレームの最初の 8 出力サンプル  $\hat{s}'_{LB}(n)$  は、フィルタ  $\tilde{W}_{t_{p,n}}$  が適用され、これは、前フレームのフィルタ  $\tilde{W}_{t_p}^{(m-1)}$  と現フレームのフィルタ  $\tilde{W}_{t_p}^{(m)}$  の合成である。フィルタ  $\tilde{W}_{t_{p,n}}$  は、 $n$  番目のサンプル  $\hat{s}'_{LB}(n)$  を生成するために、入力サンプル  $\hat{s}_{LB}(n)$  に下式のように適用される。

$$\tilde{W}_{t_{p,n}}(i) = \gamma_w(n) \cdot \tilde{W}_{t_p}^{(m)}(i) + (1 - \gamma_w(n)) \cdot \tilde{W}_{t_p}^{(m-1)}(i) \quad i = 0, \dots, 32, n = 0, \dots, 7 \quad (I-19)$$

ここで、重み付け係数  $\gamma_w(n)$  は、長さが 16 のハニング窓の前半分である。

$$\gamma_w(n) = 0.5 - 0.5 \cos\left(\frac{2\pi n}{15}\right) \quad n = 0, \dots, 7 \quad (I-20)$$

そして、最初のフィルタ処理された 8 サンプル  $\hat{s}'_{LB}(n-16)$  は、下記畳み込み処理により与えられる。

$$\hat{s}'_{LB}(n-16) = \sum_{i=0}^{32} \tilde{W}_{t_{p,n}}(i) \cdot \hat{s}_{LB}(n-i) \quad n = 0, \dots, 7 \quad (I-21)$$

$n = 8, \dots, 39$  に対するフィルタ処理されたサンプル  $\hat{s}'_{LB}(n-16)$  は、 $W_{t_p}$  で記述される現フレームフィルタ  $\tilde{W}_{t_p}^{(m)}$  で入力データをフィルタ処理することで得られる。

$$\hat{s}'_{LB}(n-16) = \sum_{i=0}^{32} W_{t_p}(i) \cdot \hat{s}_{LB}(n-i) \quad n = 8, \dots, 39 \quad (I-22)$$

ここで、フレーム  $m$  で得られる最初の 16 サンプル  $\hat{s}'_{LB}(n-16)$  は、前フレーム  $(m-1)$  ( $\hat{s}_{LB}^{(m-1)}(n)$ ,  $n = 24, \dots, 39$ ) の最後の 16 サンプルに対応し、さらに、フレーム  $m$  で得られる最後の 24 サンプルは、現フレーム  $m$  の最初の 24 サンプル  $\hat{s}_{LB}^{(m)}(n)$ ,  $n = 0, \dots, 23$  に対応する。

#### I. 5 振幅補正の制限

ポストフィルタ処理の最終段階では、振幅補正の制限を行う。これは、復号信号  $\hat{s}_{LB}(n)$  に対するポストフィルタ処理によって発生する補正を制限する。この制限は、 $|\hat{s}'_{LB}(n)| > 24$  の時にのみ適用される。この制限されたポストフィルタ処理後の信号  $\hat{s}''_{LB}(n)$  は、ポストフィルタ処理後信号  $\hat{s}'_{LB}(n)$  ( $|\hat{s}'_{LB}(n) - \hat{s}_{LB}(n)| \leq d'_{\max}(\tilde{\eta}_{LB}(n))$  の場合) の現振幅か、あるいは、 $\hat{s}'_{LB}(n)$  と  $\hat{s}_{LB}(n)$  との中間値のどちらか一方となる。

$$\begin{aligned} \text{If } \hat{s}'_{LB}(n) < \hat{s}_{LB}(n) - d'_{\max}(\tilde{\eta}_{LB}(n)), \quad \hat{s}''_{LB}(n) &= \hat{s}_{LB}(n) - d'_{\max}(\tilde{\eta}_{LB}(n)) \\ \text{If } \hat{s}'_{LB}(n) > \hat{s}_{LB}(n) + d'_{\max}(\tilde{\eta}_{LB}(n)), \quad \hat{s}''_{LB}(n) &= \hat{s}_{LB}(n) + d'_{\max}(\tilde{\eta}_{LB}(n)) \end{aligned}$$

ここで、 $\tilde{\eta}_{LB}(n) = 14 - \left\lfloor \log_2(|\hat{s}_{LB}(n)|) \right\rfloor$  で、許容される最大距離  $d'_{\max}$  は Table I. 1 / JT-G711.1 で与えられる。

Table I. 1 / JT-G711.1 Allowed maximum distances for postfiltering  
(ITU-T G.711.1)

$j$	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$d'_{\max}(j)$	512	256	128	64	32	16	8	8	8	8	8	8	8	8	8

線形位相 FIR フィルタと、この制限との組み合わせの結果、この後処理は、いかなる場合でも継ぎ目無くオン/オフすることが可能となる。

## 1. 6 ポストフィルタのビットイグザクト記述

JT-G711.1 復号器に対するポストフィルタをシミュレートする 16 ビット固定小数点の ANSIC コードは日本 ITU 協会から入手可能である。ポストフィルタの動作を可能にするためには、JT-G711.1 復号器のソースコードは、“-DAPPENDIX\_I\_POSTFILTER” というコンパイルスイッチによりコンパイルする必要がある。以下の節において本シミュレーションコードの使用法とソフトウェアの構成を示す。

### 1.6.1 シミュレーションソフトウェアの使用法

ポストフィルタを動作可能にすることで、復号器に対するコマンドラインは下記に示すように新たなオプション “-r1pf” が有効となる。

**decoder [-options] <law> -mode <modenum> <codefile> <outfile>**

where

law	is the desired G.711 law (A or u)
modenum	is the mode of the bitstream file (1, 2, 3, 4) Mode number 1, 2, 3, 4 correspond to R1, R2a, R2b, R3 respectively.
codefile	is the name of the input bitstream file
outfile	is the name of the decoded output file
Options:	
-decmode #	is to specify the decoder mode (1, 2, 3, 4). Mode number 1, 2, 3, 4 correspond to R1, R2a, R2b, R3 respectively. When not specified, it will be the same as modenum (mode of the bitstream file).
-r1pf	R1 postfilter enabled This option is for R1 mode only.
-hardbit	Input bitstream file is in multiplexed hardbit format.
-quiet	quiet processing

### 1.6.2 シミュレーションソフトウェアの構成

Table I. 2 / JT-G711.1 にポストフィルタのシミュレーションソフトウェアルーチンの一覧を示す。“-DAPPENDIX\_I\_POSTFILTER” というスイッチなしでコンパイルする場合には、それらのファイルを JT-G711.1 復号器の他のファイルと一緒にコンパイルする必要はない。

Table I. 2 / JT-G711.1 Summary of Appendix I specific routines  
(ITU-T G.711.1)

Filename	Description
post.c	Main routine that calls all post-processing subroutines
post_anasyn.c	Analysis/synthesis subroutines for post-processing
post_gainfct.c	Subroutines for estimating of the post-processing filter
post_rfft.c	64-point real FFT and iFFT
table_post.c	Tables for postfilter

付録  
(標準 J T - G 7 1 1 . 1 に対する)

用語対照表

英 語	T T C 標準用語
A-law	A 則
artefact	異音
audio	オーディオ
autocorrelation function	自己相関関数
background noise	背景雑音
basic operator	基本演算子
bit allocation	ビット割り当て
bitstream	ビット列
codec	コーデック
coder	符号器
compander segment	圧伸セグメント
companding law	圧伸則
dead-zone quantizer	デッドゾーン量子化器
decimation	間引き
decoder	復号器
embedded	エンベデッド
encoder	符号化器
excitation signal	励振信号
exponent bits	指数ビット
fold	折り返す
frame erasure concealment	パケット消失補償
good frame	正常フレーム
higher band	高域
layer	レイヤ
legacy	既存の
locally decoded	局部復号された
log companded PCM	対数圧伸 PCM
lower band	低域
mantissa bits	仮数部ビット
modified discrete cosine transform	変形離散コサイン変換
multiplexer	多重化器
narrowband	狭帯域
noise gate	ノイズゲート
noise shaping	ノイズシェーピング
open loop	開ループ
packet losses	パケット損失
perceptual filter	聴覚フィルタ
pitch lag	ピッチラグ
pre-selection	予備選択
quadrature mirror filter	直交ミラーフィルタ
refinement signal	精細信号
scalable	スケーラブル
sign	極性
synthesis QMF filterbank	合成 QMF フィルタバンク
u-law	u 則
wideband	広帯域