

(19)日本国特許庁(JP)

(12)特許公報 ( B 2 )

(11)特許番号

特許第 3 5 2 9 6 4 8 号

( P 3 5 2 9 6 4 8 )

(45)発行日 平成16年5月24日(2004.5.24)

(24)登録日 平成16年3月5日(2004.3.5)

(51)Int. Cl. <sup>7</sup>	識別記号	F I	
H 0 3 M	7/30	H 0 3 M	7/30 Z
G 1 0 L	11/00		1/12 Z
	19/00	G 1 0 L	9/16
H 0 3 M	1/12		9/18 M

請求項の数 2

(全 9 頁)

(21)出願番号	特願平10-368218	(73)特許権者	000004226 日本電信電話株式会社 東京都千代田区大手町二丁目3番1号
(22)出願日	平成10年12月24日(1998.12.24)	(72)発明者	岩上 直樹 東京都新宿区西新宿三丁目19番2号 日本 電信電話株式会社内
(65)公開番号	特開2000-196452(P2000-196452A)	(72)発明者	守谷 健弘 東京都新宿区西新宿三丁目19番2号 日本 電信電話株式会社内
(43)公開日	平成12年7月14日(2000.7.14)	(72)発明者	池田 和永 東京都新宿区西新宿三丁目19番2号 日本 電信電話株式会社内
審査請求日	平成14年4月24日(2002.4.24)	(74)代理人	100064908 弁理士 志賀 正武
		審査官	北村 智彦

最終頁に続く

(54)【発明の名称】オーディオ信号符号化方法

1

(57)【特許請求の範囲】

【請求項 1】 離散サンプル列として入力される入力サンプルであるオーディオ信号をデジタル符号に符号化する方法であって、

複数の入力サンプルからなる分析単位毎の入力サンプル列中の立ち上がりの有無を検出し、立ち上がりが検出された分析単位の情報を表す第 1 のデジタル符号を出力する立ち上がり検出過程と、

前記立ち上がり検出過程において立ち上がりが検出された場合には、入力された入力サンプル中の、前記立ち上がり  
が検出された分析単位の入力サンプル列から開始されるあらかじめ決められたフレーム長のサンプル列を出力し、

前記立ち上がり検出過程において立ち上がりが検出されなかった場合には、入力された入力サンプル中の、前回

2

のフレーム分割過程で出力したサンプル列の直後の入力サンプルから開始されるあらかじめ決められたフレーム長のサンプル列を出力するフレーム分割過程と、  
前記フレーム分割過程から出力されたフレーム単位のサンプル列を符号化して第 2 のデジタル符号を出力する符号化過程とを有することを特徴とするオーディオ信号符号化方法。

【請求項 2】 前記フレーム分割過程は、  
前記フレーム長のサンプル列を出力する際に、フレーム中の最初の分析単位以降に立ち上がりを有する分析単位がある場合に、この分析単位からフレームの最後尾までのフレーム中のサンプルを 0 に設定して出力することを特徴とする請求項 1 記載のオーディオ信号符号化方法。

【発明の詳細な説明】

【 0 0 0 1 】

10

【発明の属する技術分野】本発明は、オーディオ信号の符号化及び復号化に関し、オーディオ信号を、より少ない情報量で表現できるようにデジタル符号に変換する符号化方法及びそのデジタル符号をオーディオ信号に復元する復号化方法に関する。

## 【0002】

【従来の技術】オーディオ信号を高効率に符号化する際、圧縮の度合いによって2つの手法に分けられる。一つは符号を復元すれば原信号と同じ信号が得られる可逆符号化、もう一つは符号を復元しても原信号と同じ信号には戻らない非可逆符号化である。信号を完全に復元するという制約のない非可逆符号化は、高い圧縮率が得られるという利点があるが、復元信号と原信号との差である符号化歪みが生じるという欠点を持つ。

【0003】オーディオ信号の符号化では、符号化歪みが耳で聞いて検知されなければ実用上問題がないため、符号化歪みを検知できないようにするための制御を行った方がより高い圧縮率を得ることができる。オーディオ信号を符号化するには、まず離散信号にサンプリングされたオーディオ信号を一定サンプル数ごとにフレームと呼ばれる単位に束ね、フレームごとに符号化処理をすることが一般的である。

【0004】オーディオ信号は、通常周波数特性に偏りがあり、またフレーム内でほぼ定常であることが多いため、フレームごとに時間-周波数変換をかけ、周波数特性の偏りを情報量の割り当てに利用する変換符号化と呼ばれる方法がある。(図7参照)

変換符号化は定常的な音に対しては、一般的に高い符号化能率が得られる特性がある。また、特に非可逆符号化では、符号化歪みを周波数成分ごとに詳細に制御することが容易であり、検知しにくい歪み制御という観点からも有利である。

【0005】しかし、時間的な処理という観点では、フレームごとに一括の処理をするので、フレーム内で信号が定常でない場合には、高効率を得ることが難しい。また、フレーム内で非定常な入力であっても、符号化歪みは時間的にほぼ一定のレベルで生じるので、特に急激な立ち上がりを含む信号が入力された場合、立ち上がりの直前にプリエコーと呼ばれる検知可能なノイズが生じてしまう。(図8参照)

【0006】この問題を回避するために、変換符号化では、急激な立ち上がりが存在するフレームに限り、フレームを構成するサンプル群をさらに分割して複数のサブフレームを構成し、そのサンプルごとに変換符号化を行う対策をとっている。このようにすると、プリエコーは短くなったフレームの範疇に収まるため、検知されにくくなる。しかし、急激な立ち上がりを含む音でも、周波数成分に大きな偏りを持ったものも存在し、そのような音に関しては、立ち上りを含んでも高効率符号化のためには、依然周波数分解能の高い分析が必要であ

り、サブフレーム分割による周波数分解能の低下は立ち上がり部分での音質劣化につながりやすい。

## 【0007】

【発明が解決しようとする課題】上述したように従来の技術には種々の問題点があり、その解決策が課題となっていた。本発明はこのような背景の下になされたもので、オーディオ信号の変換符号化において、立ち上がり部分を良好な音質で符号化できるオーディオ信号符号化方法及び復号化方法を提供することを目的とする。

## 【0008】

【課題を解決するための手段】請求項1に記載の発明は、離散サンプル列として入力される入力サンプルであるオーディオ信号をデジタル符号に符号化する方法であって、複数の入力サンプルからなる分析単位毎の入力サンプル列中の立ち上がりの有無を検出し、立ち上がりが検出された分析単位の情報を表す第1のデジタル符号を出力する立ち上がり検出過程と、前記立ち上がり検出過程において立ち上がりが検出された場合には、入力された入力サンプル中の、前記立ち上がりが検出された分析単位の入力サンプル列から開始されるあらかじめ決められたフレーム長のサンプル列を出力し、前記立ち上がり検出過程において立ち上がりが検出されなかった場合には、入力された入力サンプル中の、前回のフレーム分割過程で出力したサンプル列の直後の入力サンプルから開始されるあらかじめ決められたフレーム長のサンプル列を出力するフレーム分割過程と、前記フレーム分割過程から出力されたフレーム単位のサンプル列を符号化して第2のデジタル符号を出力する符号化過程とを有することを特徴とするオーディオ信号符号化方法を提供する。

【0009】請求項2に記載の発明は、前記フレーム分割過程は、前記フレーム長のサンプル列を出力する際に、フレーム中の最初の分析単位以降に立ち上りを有する分析単位がある場合に、この分析単位からフレームの最後尾までのフレーム中のサンプルを0に設定して出力することを特徴とする請求項1記載のオーディオ信号符号化方法を提供する。

## 【0010】

## 【0011】

## 【0012】

## 【0013】

## 【0014】

## 【0015】

## 【0016】

## 【0017】

## 【0018】

【0019】請求項12に記載の発明は、前記第3の段階が、前記第1の段階で得られた単位区間ごとのオーディオ信号サンプル列のうち、前記第2の段階で得られた単位区間の区切りの時点までのサンプルのみを切り出

し、これを新たな単位区間とする第 9 の段階と、前記新たな単位区間のうち、先頭の一定数のサンプルと最後尾の同じ一定数のサンプルを除いたサンプル列を切り出す第 10 の段階と、新たな単位区間のうち、最後尾の一定数のサンプルと次の新たな単位区間の先頭の同じ一定数のサンプルとを重ね合わせて加算し、一定数のサンプル列を得る第 11 の段階と、前記第 10 の段階で得られたサンプル列と前記第 11 の段階で得られたサンプル列とを順次つなぎ合わせることを、すべての単位区間について行う第 12 の段階とからなることを特徴とする請求項 7 乃至 11 のいずれかに記載のオーディオ信号復号化方法を提供する。

【0020】請求項 13 に記載の発明は、前記第 11 の段階において、新たな単位区間の最後尾の一定数のサンプルに立ち下がる形状の窓関数をかけ、次の新たな単位区間の先頭の同じ一定数サンプルに前記立ち下がる形状を相補的に立ち上がる形状の窓関数をかけた後、重ね合わせて加算を行うことを特徴とする請求項 7 乃至 12 のいずれかに記載のオーディオ信号復号化方法を提供する。

【0021】上述のように、この発明では入力オーディオ信号の立ち上がりを検出する手段を備える。立ち上がりが検出されたら、その時期に新たなフレームを開始する。その直前のフレームについては立ち上がり時期以降の残りのサンプルについて 0 を割り当てる。または緩やかに減衰するオーバーラップ窓を乗算して符号化する。

【0022】直前のフレームにオーバーラップ窓を乗算した場合には、新たなフレームについても該オーバーラップ窓と相補的に緩やかに増加するオーバーラップ窓を乗算して符号化する。これらはオーディオ信号の記録・再生、放送、通信路を介した伝送を能率よく行う手段として利用される。

【0023】復号化においてはフレーム毎の符号を復号化し、立ち上がり時期から新しいフレームの標本化信号と直前のフレームの標本化信号とを加算したうえで出力する。プリエコーは必ず立ち上がりのあったフレーム内の、立ち上がり時点より前に存在するので、立ち上がりとフレームの先頭を一致させればプリエコーは生じない。また、立ち上がりの前後においても時間 - 周波数変換の単位サンプル数を変えないので、高い周波数分解能を保ったまま変換符号化が行える。

【0024】

【発明の実施の形態】以下、この発明の一実施形態について図を参照しながら説明する。図 1 は本発明の一実施形態のうち、符号化側の構成を示すブロック図である。この図のオーディオ信号入力は、一定の時間間隔でサンプリングされた離散オーディオ信号であり、符号化側は、立ち上がり検出部 1 と、フレーム分割部 2 と、時間 - 周波数変換部と、符号化部 4 と、ビット列生成部とで構成される。

【0025】なお、この実施形態中のサンプルの数の最適値は、入力オーディオ信号のサンプリング周波数に大きく影響されるが、この実施形態では入力オーディオ信号のサンプリング周波数が 48 kHz であるとしてサンプル数を例示している。入力されたオーディオ信号は、立ち上がり検出部 1 とフレーム分割部 2 の両方に入力される。

【0026】立ち上がり検出部 1 は、オーディオ信号を入力とし、1 か 0 の値を持つ立ち上がりフラグを出力とする。立ち上がりフラグは立ち上がりが検出されない限り 0 にセットされる。立ち上がり検出の方法は、例えば入力信号を N 点のサンプル入力ごとにパワーの平均値を計算し、過去 5 回程度のパワーの平均と、今回計算したパワー平均の比が 2.0 程度であった場合には立ち上がりと判断して立ち上がりフラグを 1 にセットすればよい。また別の方法としては、2 次程度の線形予測分析にかけて線形予測誤差を上記パワー平均による判断と同じ検出方法により分析し、立ち上がりの有無を判断してもよい。

20 【0027】さらに別の方法としては、N 点のサンプルごとに 5 1 2 次程度のパワースペクトルを計算し、パワースペクトル同士のユークリッド距離を 3 2 程度に分割されたサブバンドごとに計算して、例えば 3 つ以上のサブバンドで距離とサブバンドの平均パワーとの比が、1.0 を超えた場合に立ち上がりと判断して立ち上がりフラグを 1 にセットしても良い。立ち上がりフラグは、立ち上がり検出の分析単位である入力サンプルの N 点ごとに出力される。N の値は、128 程度に設定すると良好な結果が得られる。(図 2 参照)

30 【0028】図 2 に示すフレーム分割部 2 は、入力オーディオ信号と立ち上がり検出部 1 から出力された立ち上がりフラグを入力とし、一定サンプル数、例えば 1024 程度のフレーム長で構成されるサンプル列と、前のフレームとのオーバーラップ情報とを出力する。フレーム分割部 2 は、フレームバッファ 20、第 1 入力バッファ 21、第 2 入力バッファ 22、窓制御部 23、立ち上がり窓バッファ 25、通常窓バッファ 26、立ち下がり窓バッファ 27、オーバーラップ情報バッファ 28、かけ算器 29 によって構成される。

40 【0029】処理は、第 2 入力バッファ 22 に過去 N 点の入力オーディオ信号を記憶し、N 点の信号入力ごとに 第 1 入力バッファ 21 に内容を複写することから始まる。第 1 入力バッファ 21 は N 点のサンプルを記憶でき、第 2 入力バッファ 22 から信号を受け取り複写した後、掛け算器 29 にデータを送る。掛け算器 29 がデータを受け取った後、直前に受け取った立ち上がりフラグが 1 にセットされていた場合、内容を更新せずにもう一度掛け算器 29 にデータを送り、第 2 入力バッファ 22 からデータが送られてくるのを待つ。直前に受け取った立ち上がりフラグが 0 にセットされていた場合、そのま

ま第2入力バッファ22からデータが送られてくるのを待つ。

【0030】窓制御部23では、立ち上がりフラグをもとに掛け算器29に送る窓バッファを窓バッファ24から次のように選択する。今回の動作時に立ち上がりフラグが1にセットされていて、かつ前回の動作時に立ち上がりフラグが0にセットされていた場合、窓バッファとして立ち下がり窓27を選択する。今回の動作時に立ち上がりフラグが1にセットされていて、かつ前回の動作時に立ち上がりフラグが1にセットされていた場合、窓バッファとして立ち上がり窓25を選択する。今回の動作時に立ち上がりフラグが0にセットされていたならば、通常窓26を選択する。

【0031】掛け算器29では、第1入力バッファ21の各サンプルと、窓制御部23から送られてきた窓バッファの各サンプルを掛け合わせ、その値をポインタで示しているフレームバッファ20の先頭に追加する。フレームバッファ20は、掛け算器29からデータが追加された後、ポインタをN点進める。

【0032】さらに立ち上がりフラグとポインタの位置\*20

表1：オーバーラップ情報の値の例（フレームあたりのサンプル数：1024、立ち上がりフラグの頻度：128の場合）

	オーバーラップなし	オーバーラップあり
OVL_SW（オーバーラップの有無）	0（1 bit）	1（1 bit）
OVL_POS（オーバーラップの位置）	情報なし（0 bit）	0から7（3 bit）

【0035】なお、「オーバーラップあり」の OVL\_POS の値は、

$$OVL\_POS = NF / N$$

であらわされる。ただし、NFはフレーム長である。OVL\_POS の値は整数であり、上記の割り算の端数は切り捨てられる。また、窓バッファの値は、通常窓が、

$$W_{norm}[n] = 1.0$$

立ち上がり窓が1

$$W_{on}[n] = (n + 0.5) / N$$

立ち下がり窓が

$$W_{off}[n] = (N - n - 0.5) / N$$

とする。ただし、nはバッファ中の要素のインデックスで、0 ≤ n < Nとする。

【0036】また、立ち上がり窓と立ち下がり窓の値はそれぞれ以下のように設定しても良い。

$$W_{on}[n] = 0.5 \{ 1 - \cos((n + 0.5) / N) \}$$

$$W_{off}[n] = 0.5 \{ 1 + \cos((n + 0.5) / N) \}$$

【0037】時間 - 周波数変換部3では、フレーム分割部2から受け取ったサンプル列を MDCT を使って周波数領域の係数サンプル列に変換する。時間 - 周波数変換部3は、入力されたNF点のサンプルを格納する入力バッファ

\*により、次の動作を切り替える。立ち上がりフラグが0及びポインタの位置がフレームバッファ20の最後尾であった場合、フレームバッファ20の内容とオーバーラップ情報バッファ28の内容をフレーム分割部2の出力として出力した後、ポインタの位置をフレームバッファ20の先頭に移動させ、オーバーラップ情報バッファ28の内容を表1で示す「オーバーラップ無し」の場合の値に更新する。

【0033】立ち上がりフラグが0で、ポインタの位置がフレームバッファ20の最後尾でなかった場合は何もしない。立ち上がりフラグが1の場合、フレームバッファ20のうちポインタから最後尾までの値を0にした後、フレームバッファ20の内容とオーバーラップ情報バッファ28の内容をフレーム分割部2の出力として出力した後、ポインタの位置をフレームバッファ20の先頭に移動させ、オーバーラップ情報バッファ28の内容を表1で示す「オーバーラップあり」の場合の値に更新する。

【0034】

【表1】

ア30と、該入力バッファ30から渡されたNF点のサンプルを格納する記憶バッファ31と、2×NF点のサンプルを格納する窓バッファ24と、掛け算器33と、MDCT計算器34とから構成される。（図3参照）

入力されたNF点のサンプル列は、入力バッファ30に格納される。記憶バッファ31は、フレーム分割部2からオーバーラップ情報を受け取り、OVL\_SW が1にセットされていた場合、内容を0にクリアする。

【0038】掛け算器33は、NF点の記憶バッファ31の後ろにNF点の入力バッファ30をつなげて作成した2×NF点のサンプル列の各々と、窓バッファ32のサンプル列の各々を掛け合わせて2×NF点のサンプル列を出力する。MDCT 計算器34は、掛け算器33から受け取った2×NF点のサンプルをMDCT変換して、NF点のサンプル列を出力する。（図3参照）

記憶バッファ31は、掛け算器33に内容を渡した後、入力バッファ30の内容を複写することにより更新される。

【0039】また、窓バッファのサンプルの値は、次式であらわされる。

$$W_{MDCT}[n] = \sin((n + 0.5) / NF)$$

ただし、0 ≤ n < (NF - 1)とする。入力バッファ30が入力サンプル列を受け取るまでに、立ち上がり検出

部から入力された立ち上がりフラグが 1 にセットされた  
なら、MDCT 計算器 3 4 が計算を終えた後、入力バッファ  
3 0 を 0 にクリアしてもう一度 MDCT 計算器 3 4 を動作さ  
せるまでの処理を行う。(図 3 参照)

【0040】符号化部 4 では、時間 - 周波数変換部 3 から  
出力された MDCT 係数サンプルを符号化する。MDCT 係数  
を符号化する技術として、Transform-domain Weighted  
Interleave Vector Quantization (TWINVQ) や、Advanc  
ed Audio Coding (AAC) ) などをを用いることができる。

【0041】また符号化部 4 では、I フレーム分割部より  
オーバーラップ情報を受け取り、オーバーラップ情報の  
うち、OVL\_SW が 1 にセットされていた場合には、符号化  
部 4 の内部でフレーム間予測に用いているすべてのメモ  
リをリセットする。(図 1 参照)

ビット列生成部 5 は、フレーム分割部 2 と符号化部 4 から  
受け取った符号インデックス 2 進数に変換し、この 2  
進数をあらかじめ決まった順序で配置し、ビット列とし  
て出力する。

【0042】次に、図 4 にこの発明の一実施形態の復号  
化側の構成のブロック図を示す。復号化側は、ビット列  
復号部 4 0、復号化部 4 1、周波数 - 時間変換部 4 2、  
およびフレーム結合部 4 3 で構成される。ビット列復号  
部 4 0 では、符号化部 2 から受け取ったビット列をあら  
かじめ決まった順序でオーバーラップ情報と符号化部 2 に  
渡す符号インデックスとに分解し、10 進数に変換して  
出力する。

【0043】復号化部 4 1 では、ビット列復号部 4 0 から  
受け取った符号化インデックスを復号化して、MDCT 係  
数サンプル列を出力する。MDCT 係数に復号化する技術と  
して、Transform-domain Weighted Interleave Vector  
Quantization (TWINVQ) や Advanced Audio Coding (AA  
C) ) などをを用いることができる。

【0044】また復号化部 4 1 では、ビット列復号部 4  
0 からオーバーラップ情報を受け取り、オーバーラップ情  
報のうち、OLV\_SW が 1 にセットされていた場合には、復  
号化部 4 1 の内部でフレーム間予測に用いているすべての  
メモリをリセットする。周波数 - 時間変換部 4 2 で  
は、復号化部 4 1 から受け取った MDCT 係数サンプル列  
を時間領域の出力フレームサンプル列に変換する。

【0045】図 5 は周波数 - 時間変換部 4 2 の構成を示  
すブロック図である。周波数 - 時間変換部 4 2 は、IMDC  
T 計算器 5 0、掛け算器 5 1、窓バッファ 5 2、記憶バ  
ッファ 5 3、出力バッファ 5 4、足し算器 5 5 および出  
力制御器 5 6 で構成される。IMDCT 計算器 5 0 は、復号  
化部 4 1 から受け取った NF 点のサンプルを IMDCT 変換し  
て  $2 \times NF$  点の時間領域のサンプル列を出力する。

【0046】掛け算器 5 1 は、IMDCT 計算器 5 0 から受  
け取った  $2 \times NF$  点のサンプル列の各々と窓バッファ 5 2  
に格納されている  $2 \times NF$  点のサンプルの各々を掛け合  
わせて  $2 \times NF$  のサンプル列を出力する。出力されたサン  
50

ル列のうち、前半の NF サンプルは出力バッファに格納さ  
れ、出力バッファに格納されている NF 点のサンプル列の  
各々は、記憶バッファ 5 3 に格納されている NF 点のサン  
プル列の各々と加え合わされ、NF 点のサンプル列を出力  
し出力制御器 5 6 に送られる。

【0047】足し算器 5 5 の結果が出力された後、掛け  
算器 5 1 から出力された  $2 \times NF$  点のサンプルのうち後半  
の NF 点のサンプル列は、記憶バッファ 5 3 に格納され  
る。出力制御器 5 6 は、足し算器 5 5 からサンプル列を  
受け取った後、入力されたオーバーラップ情報のうち OVL\_  
SW が 0 にセットされていた場合に、足し算器から受け取  
ったサンプル列を出力フレームサンプル列として出力す  
る。OVL\_SW が 1 にセットされていた場合、何もしない。

【0048】フレーム結合部 4 3 は、周波数 - 時間変換  
部 4 2 から受け取った出力フレームサンプル列を結合し  
て、オーディオサンプル列を出力する。フレーム結合部  
4 3 は、フレームバッファ 6 0、オーバーラップ制御器 6  
1 および出力バッファ 6 2 により構成される。周波数 -  
時間変換部 4 2 から受け取った出力フレームサンプル列  
は、フレームバッファ 6 0 に格納される。フレームバッ  
ファ 6 0 に格納された信号は、オーバーラップ制御器 6 1  
に渡される。

【0049】オーバーラップ制御器 6 1 は、フレームバッ  
ファ 6 0 からサンプル列を受け取った後、出力バッファ  
6 2 のポインタ位置の制御を行う。入力されたオーバー  
ラップ情報のうち、OLV\_SW が 0 にセットされている場合、  
オーバーラップ制御器 6 1 は出力バッファ 6 2 のポインタ  
に対して何も行わないが、OLV\_SW が 1 にセットされてい  
る場合、出力バッファのポインタを強制的に NS だけ戻す  
NS 値は、オーバーラップ情報のうち OLV\_POS の値をもとに  
次式のように決定する。

$$NS = NF - \{ N ( OVL\_POS - 1 ) \}$$

【0050】オーバーラップ制御器 6 1 は出力バッファ 6  
2 のポインタ位置の制御を行った後、フレームバッファ  
6 0 から受け取ったサンプル列を出力バッファ 6 2 に送  
る。出力バッファ 6 2 は、 $2 \times NF + N$  個のサンプルを  
格納できるバッファである。オーバーラップ制御器 6 1 から  
サンプルを 1 サンプル受け取るたびに、バッファ中の  
ポインタで示されている位置のバッファの要素の値に、  
入力されたサンプルの値を加え合わせ、その新しい値を  
バッファの要素に格納し、ポインタの位置を 1 つ進め  
る。

【0051】進めるべきポインタの位置が、バッファの  
最後尾にあるならば、バッファの要素の 1 番目から NF 番  
目までの値を出力オーディオサンプル列としそ出力した  
後、NF + 1 番目から  $2NF + N$  番目までの要素については  
値を先頭方向に NF 個シフトし、空白になる NF + N + 1 番  
目から  $2NF + N$  番目までの要素については 0 にクリアし  
た後、ポインタの位置を NF だけ戻す。

【0052】以上、本発明の一実施形態の動作を図面を

参照して詳述してきたが、本発明はこの実施形態に限られるものではなく、本発明の要旨を逸脱しない範囲の設計変更等があっても本発明に含まれる。

【0053】

【発明の効果】これまでに説明したように、この発明によるオーディオ信号符号化方法によれば、立ち上がりとともにフレームが開始される。そのため、フレームの途中で立ち上がりが生ずる音声を符号化及び復号化した場合にみられるプリエコーの発生を防げる。よって、ピアノ、ギター、トライアングルのように立ち上がり成分を多く含み、その直後から強いピッチ成分を有するオーディオ信号の高効率な非可逆符号化に際しても品質の劣化を最小限に抑制できる。

【図面の簡単な説明】

【図1】 本発明の符号化側の一実施形態の構成を示すブロック図である。

【図2】 本発明の符号化側の一実施形態の構成のうち、フレーム分割部の詳細な構成を示すブロック図である。

【図3】 本発明の符号化側の一実施形態の構成のうち、時間一周波数変換部の詳細な構成を示すブロック図である。

【図4】 本発明の復号化側の一実施形態の構成を示すブロック図である。

【図5】 本発明の復号化側の一実施形態の構成のうち、周波数 - 時間変換部の詳細な構成を示すブロック図である。

【図6】 本発明の復号化側の一実施形態の構成のうち、フレーム結合部の詳細な構成を示すブロック図である。

【図7】 一般的な変換符号化の基本構成を示すブロック図である。

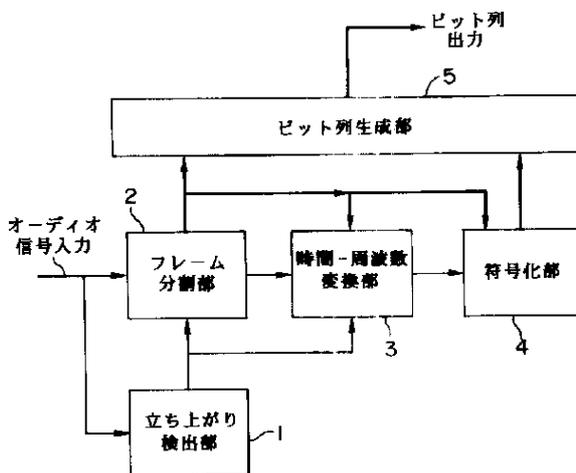
【図8】 プリエコーの生じる原理を説明するための模式図である。

【図9】 フレームの強制開始を説明するための模式図である。

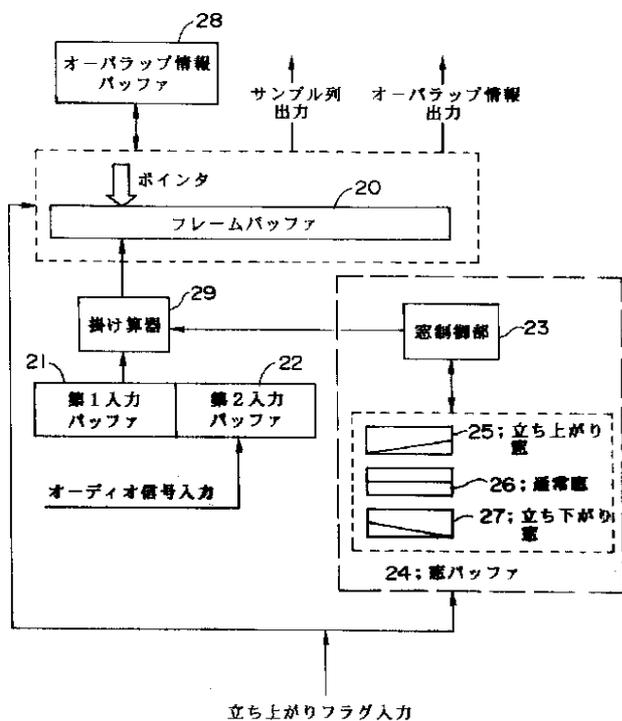
【符号の説明】

- 1...立ち上がり検出部
- 2...フレーム分割部
- 3...時間 - 周波数変換部
- 4...符号化部
- 5...ビット列生成部
- 20...フレームバッファ
- 21...第1入力バッファ
- 22...第2入力バッファ
- 23...窓制御部
- 24...窓バッファ
- 25...立ち上がり窓
- 26...通常窓
- 27...立ち下がり窓
- 28...オーバーラップ情報バッファ
- 30...入力バッファ
- 31...記憶バッファ
- 40...ビット列復号部
- 41...復号化部
- 42...周波数 - 時間変換部
- 43...フレーム結合部
- 50...IMDCT計算器
- 51...掛け算器
- 52...窓バッファ
- 53...記憶バッファ
- 54...出力バッファ
- 55...足し算器
- 56...出力制御器
- 60...フレームバッファ
- 61...オーバーラップ制御器
- 62...出力バッファ

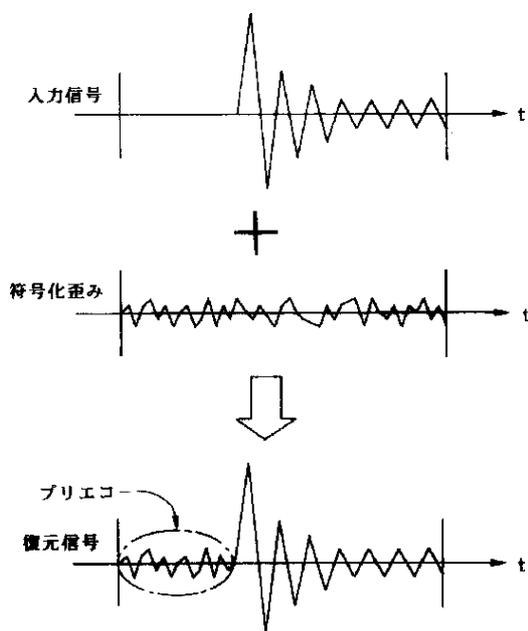
【図1】



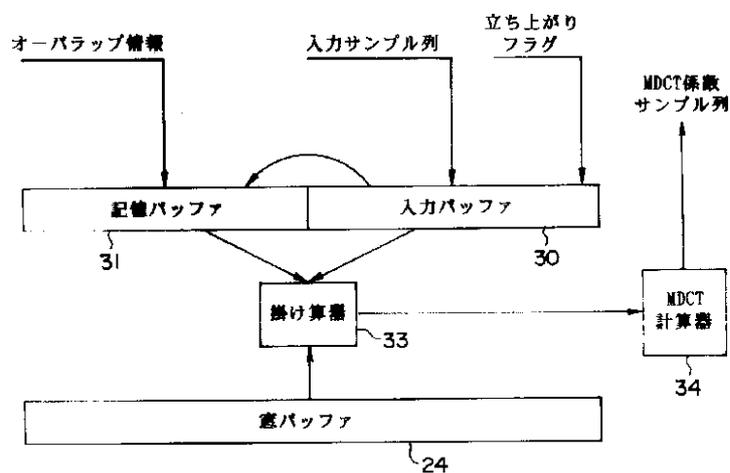
【図 2】



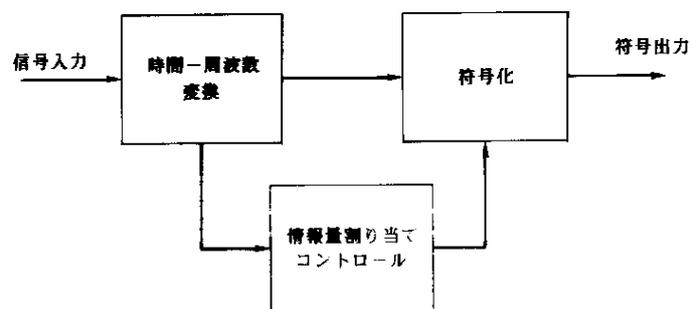
【図 8】



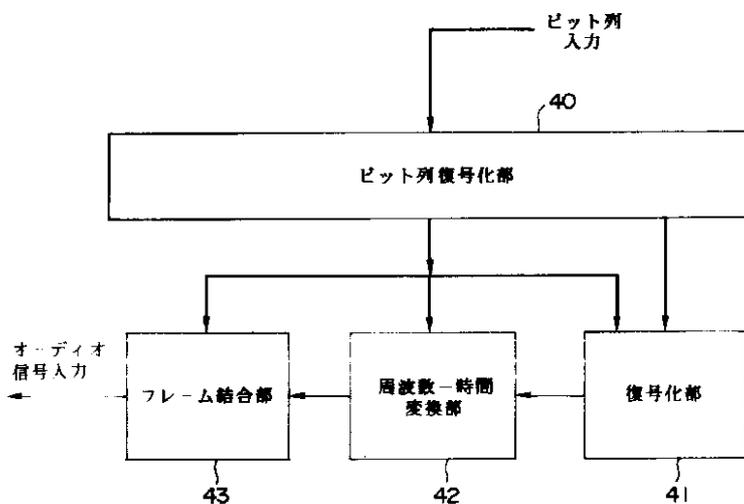
【図 3】



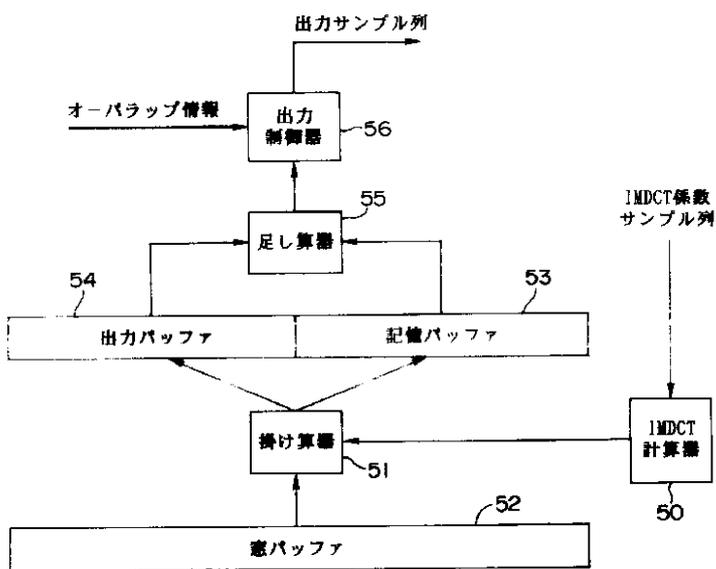
【図 7】



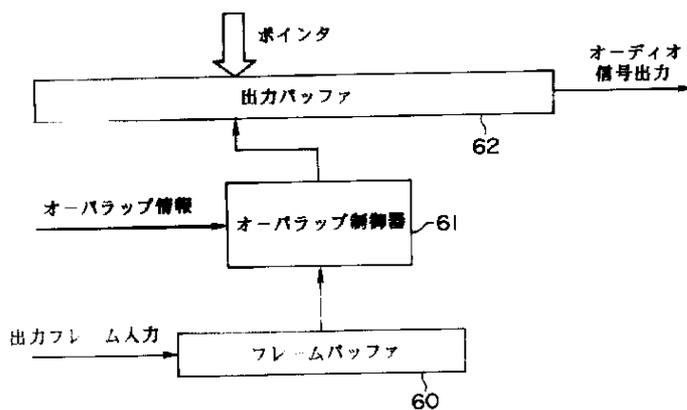
【図 4】



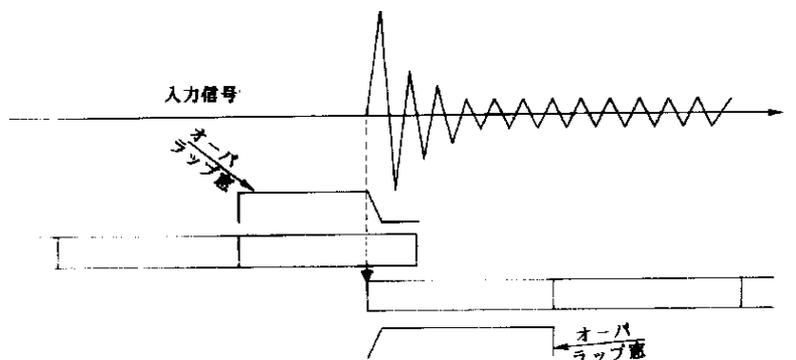
【図 5】



【図 6】



【図 9】



フロントページの続き

(72)発明者 神 明夫  
 東京都新宿区西新宿三丁目19番2号 日  
 本電信電話株式会社内

(56)参考文献 特開 平7 - 44192 ( J P , A )  
 特開 平4 - 304029 ( J P , A )

(72)発明者 森 岳至  
 東京都新宿区西新宿三丁目19番2号 日  
 本電信電話株式会社内

特開 平7 - 66733 ( J P , A )  
 特開 平8 - 160998 ( J P , A )

(72)発明者 千喜良 和明  
 東京都新宿区西新宿三丁目19番2号 日  
 本電信電話株式会社内

(58)調査した分野(Int.Cl.<sup>7</sup>, D B名)

H03M 7/30  
 G10L 11/00  
 G10L 19/00  
 H03M 1/12