

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5555707号  
(P5555707)

(45) 発行日 平成26年7月23日 (2014. 7. 23)

(24) 登録日 平成26年6月6日 (2014. 6. 6)

(51) Int. Cl.		F I	
<b>G 1 0 L 19/025 (2013. 01)</b>		G 1 0 L 19/025	
<b>G 1 0 L 19/06 (2013. 01)</b>		G 1 0 L 19/06	B
<b>G 1 0 L 19/20 (2013. 01)</b>		G 1 0 L 19/20	
<b>G 1 0 L 19/22 (2013. 01)</b>		G 1 0 L 19/22	

請求項の数 21 (全 75 頁)

(21) 出願番号	特願2011-530415 (P2011-530415)	(73) 特許権者	500341779
(86) (22) 出願日	平成21年10月7日 (2009. 10. 7)		フラウンホーファー・ゲゼルシャフト・ツール・フェルデルング・デル・アンゲヴァンテン・フォルシュング・アインゲトラーゲネル・フェライン
(65) 公表番号	特表2012-505423 (P2012-505423A)		ドイツ連邦共和国, 80686 ミュンヘン, ハンザシュトラッセ 27ツェー
(43) 公表日	平成24年3月1日 (2012. 3. 1)	(73) 特許権者	504226412
(86) 国際出願番号	PCT/EP2009/007205		ヴォイスエイジ・コーポレーション
(87) 国際公開番号	W02010/040522		カナダ・ケベック・H3R・2H6・モン
(87) 国際公開日	平成22年4月15日 (2010. 4. 15)		トリオール・ルサーヌ・ロード・750・
審査請求日	平成23年5月23日 (2011. 5. 23)		スイート・250
(31) 優先権主張番号	61/103, 825	(74) 代理人	100085497
(32) 優先日	平成20年10月8日 (2008. 10. 8)		弁理士 筒井 秀隆
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	08017663. 9		
(32) 優先日	平成20年10月8日 (2008. 10. 8)		
(33) 優先権主張国	欧州特許庁 (EP)		

最終頁に続く

(54) 【発明の名称】 マルチ分解能切替型のオーディオ符号化及び復号化スキーム

(57) 【特許請求の範囲】

【請求項 1】

時間ドメインのオーディオ信号を符号化するオーディオ符号器において、  
 第1符号化アルゴリズムを使用して前記オーディオ信号を符号化し、第1符号化済信号を得る第1符号化分枝であって、前記オーディオ信号を時間ドメインからスペクトルドメインへと変換する第1変換器(410)を含む第1符号化分枝(400)と、  
 前記第1符号化アルゴリズムとは異なる第2符号化アルゴリズムを使用して前記オーディオ信号を符号化し、第2符号化済信号を得る第2符号化分枝であって、前記オーディオ信号を時間ドメインからLPC(線形予測符号化)ドメインへと変換するドメイン変換器(510)と、前記ドメイン変換器(510)により変換された信号をLPCドメインからLPCスペクトルドメインへと変換する第2変換器(523)と、を含む第2符号化分枝(500)と、  
 前記第1符号化分枝と前記第2符号化分枝との間で切替を実行し、入力された前記オーディオ信号のある部分について、前記第1符号化済信号又は前記第2符号化済信号のいずれかが符号器出力信号の中に存在するよう切り替えるスイッチ(200)と、  
 前記オーディオ信号の前記部分を分析し、前記オーディオ信号の前記部分が前記符号器出力信号において前記第1符号化済信号として表現されるか又は前記第2符号化済信号として表現されるかを決定する信号分析器であって、前記オーディオ信号の前記部分を表現する前記第1符号化済信号又は前記第2符号化済信号が生成されるとき、前記第1変換器と前記第2変換器のそれぞれの時間/周波数分解能を可变的に決定する信号分析器(300)

、525)と、  
前記第1符号化済信号及び前記第2符号化済信号と、前記第1符号化済信号及び前記第2符号化済信号を示す情報と、前記第1符号化済信号を符号化するため及び前記第2符号化済信号を符号化するために適用された時間/周波数分解能を示す情報と、を含む符号器出力信号を生成するための出力インターフェイス(800)と、  
を備えるオーディオ符号器。

【請求項2】

請求項1に記載のオーディオ符号器であって、  
前記信号分析器(300, 525)は、前記オーディオ信号の前記部分をスピーチ状のオーディオ信号又は音楽状のオーディオ信号として分類し、且つ、音楽信号の場合には前記第1変換器(410)の時間/周波数分解能を決定するために過渡の検出を実行し、又は前記第2変換器(523)の時間/周波数分解能を決定するために合成による分析を実行することを特徴とする、オーディオ符号器。

10

【請求項3】

請求項1又は2に記載のオーディオ符号器であって、  
前記第1変換器(410)と前記第2変換器(523)とは、可変的なウィンドウサイズを持つウィンドウ関数と可変的な変換長を持つ変換関数とを含む可変的なウィンドウ化変換処理器を備え、  
前記信号分析器(300, 525)は、その信号分析に基づいて、前記ウィンドウサイズ及び/又は前記変換長を制御することを特徴とする、オーディオ符号器。

20

【請求項4】

請求項1乃至3のいずれか1項に記載のオーディオ符号器であって、  
前記第2符号化分枝は、前記ドメイン変換器(510)により決定されたLPCドメインでオーディオ信号を処理する第1処理分枝(522)と、前記第2変換器(523)を有する第2処理分枝(523, 524)と、を含み、  
前記信号分析器は、前記オーディオ信号の前記部分をサブ部分のシーケンスへとサブ分割し、且つ前記第2処理分枝によって処理されたサブ部分の位置に対する前記第1処理分枝によって処理されたサブ部分の位置に依存して、前記第2変換器(523)の時間/周波数分解能を決定することを特徴とする、オーディオ符号器。

30

【請求項5】

請求項4に記載のオーディオ符号器であって、  
前記第1処理分枝はACELP符号器(526)を含み、  
前記第2処理分枝はMDCT-TCX処理デバイス(527)を含み、  
前記信号分析器(300, 525)は、前記第2変換器の時間分解能を、1つのサブ部分の長さにより決定された高い値へと設定するか、又はそのサブ部分の長さにより大きい整数値を乗算した長さにより決定されたより低い値へと設定することを特徴とする、オーディオ符号器。

【請求項6】

請求項1乃至5のいずれか1項に記載のオーディオ符号器であって、  
前記信号分析器(300, 525)は、オーディオサンプルの複数の同一サイズのブロックをカバーする一定のラスタの中で信号分類を決定し、前記オーディオ信号に依存して1つのブロックを可変的な個数のサブブロックへと分割し、そのサブブロックの長さが前記第1変換器の時間/周波数分解能又は前記第2変換器の時間/周波数分解能を決定することを特徴とする、オーディオ符号器。

40

【請求項7】

請求項1乃至6のいずれか1項に記載のオーディオ符号器であって、  
前記信号分析器(300, 525)は前記第1変換器の時間/周波数分解能を、  
2304, 2048, 256, 1920, 2160, 240のサンプルの内の少なくとも2つである複数の異なるウィンドウ長から選択することで決定するか、若しくは各変換ブロックについて1152, 1024, 1080, 960, 128, 120の係数

50

からなるグループの内の少なくとも2つを含む複数の異なる変換長を使用し、又は、前記信号分析器(300, 525)は前記第2変換器の時間/周波数分解能を、640, 1152, 2304, 512, 1024, 2048のサンプルの内の少なくとも2つである複数の異なるウインドウ長の中の1つとして決定するか、若しくは各変換ブロックについて320, 576, 1152, 256, 512, 1024のスペクトル係数からなるグループの内の少なくとも2つを含む複数の異なる変換長を使用することを特徴とする、オーディオ符号器。

【請求項8】

請求項1乃至7のいずれか1項に記載のオーディオ符号器であって、前記第2符号化分枝は、  
オーディオ信号を処理する第1処理分枝(522)と、  
前記第2変換器を含む第2処理分枝(523, 524)と、  
前記第1処理分枝(522)と第2処理分枝(523, 524)との間で切替を実行し、前記第2符号化分枝に入力された前記オーディオ信号の部分について、第1処理済信号又は第2処理済信号のいずれかが前記第2符号化済信号の中に存在するよう切り替える追加的スイッチ(521)と、  
を備えることを特徴とするオーディオ符号器。

10

【請求項9】

時間ドメインのオーディオ信号を符号化する方法において、  
第1符号化分枝(400)内で第1符号化アルゴリズムを使用してオーディオ信号を符号化し、第1符号化済信号を得るステップであって、前記第1符号化分枝は前記オーディオ信号を時間ドメインからスペクトルドメインへと変換する第1変換器(410)を含むステップと、  
第2符号化分枝(500)内で前記第1符号化アルゴリズムとは異なる第2符号化アルゴリズムを使用してオーディオ信号を符号化し、第2符号化済信号を得るステップであって、前記第2符号化分枝は、前記オーディオ信号を時間ドメインからLPC(線形予測符号化)ドメインへと変換するドメイン変換器(510)と、前記ドメイン変換器(510)により変換された信号をLPCドメインからLPCスペクトルドメインへと変換する第2変換器(523)とを含むステップと、  
前記オーディオ信号のある部分について、前記第1符号化済信号又は前記第2符号化済信号のいずれかが符号器出力信号の中に存在するように、前記第1符号化分枝と前記第2符号化分枝との間で切替を実行するステップ(200)と、  
前記オーディオ信号の前記部分が前記符号器出力信号において前記第1符号化済信号として表現されるか又は前記第2符号化済信号として表現されるかを決定するために、前記オーディオ信号の前記部分を分析するステップ(300, 525)と、  
前記オーディオ信号の前記部分を表現する前記第1符号化済信号又は前記第2符号化済信号が生成されるとき、前記第1変換器と前記第2変換器のそれぞれの時間/周波数分解能を可变的に決定するステップと、  
前記第1符号化済信号及び前記第2符号化済信号と、前記第1符号化済信号及び前記第2符号化済信号を示す情報と、前記第1符号化済信号を符号化するため及び前記第2符号化済信号を符号化するために適用された時間/周波数分解能を示す情報と、を含む符号器出力信号を生成するステップ(800)と、  
を備える方法。

20

30

40

【請求項10】

符号化された信号を復号化するオーディオ復号器であって、前記符号化された信号は、スペクトルドメインの第1符号化済信号と、LPC(線形予測符号化)スペクトルドメインの第2符号化済信号と、前記第1符号化済信号及び前記第2符号化済信号を示す情報と、前記第1符号化済信号及び前記第2符号化済信号を復号化するために使用される時間/周波数分解能を示す情報とを含む、オーディオ復号器において、  
制御可能な第1周波数/時間変換器(440)を使用して前記第1符号化済信号を復号化

50

し、第1復号化済信号を取得する第1復号化分枝であって、前記第1周波数/時間変換器(440)は前記第1符号化済信号のための前記時間/周波数分解能を示す情報を用いて前記第1符号化済信号をスペクトルドメインから時間ドメインへ変換する、第1復号化分枝(431, 440)と、

制御可能な第2周波数/時間変換器(534)を使用して前記第2符号化済信号を復号化し、第2復号化済信号を取得する第2復号化分枝であって、前記第2周波数/時間変換器(534)は前記第2符号化済信号のための前記時間/周波数分解能を示す情報を用いて前記第2符号化済信号をLPCスペクトルドメインからLPCドメインへ変換する、第2復号化分枝と、

前記時間/周波数分解能を示す情報を使用して、前記第1周波数/時間変換器(440)と前記第2周波数/時間変換器(534)とを制御する制御器(990)と、

前記第2復号化済信号をLPCドメインから時間ドメインへ変換して合成信号を生成する、ドメイン変換器(540)と、

前記第1復号化済信号と前記合成信号とを結合し、復号化されたオーディオ信号を取得する結合器(600)と、

を備えることを特徴とするオーディオ復号器。

【請求項11】

請求項10に記載のオーディオ復号器であって、前記制御器(990)は、前記第1周波数/時間変換器(440)については、前記時間/周波数分解能が、2304, 2048, 256, 1920, 2160, 240のサンプルの内の少なくとも2つである複数の異なるウィンドウ長から選択されるか、若しくは各変換ブロックについて1152, 1024, 1080, 960, 128, 120の係数からなるグループの内の少なくとも2つを含む複数の異なる変換長から選択されるように制御し、又は、

前記第2周波数/時間変換器(534)については、前記時間/周波数分解能が、640, 1152, 2304, 512, 1024, 2048のサンプルの内の少なくとも2つである複数の異なるウィンドウ長の中の1つとして選択されるか、若しくは各変換ブロックについて320, 576, 1152, 256, 512, 1024のスペクトル係数からなるグループの内の少なくとも2つを含む複数の異なる変換長から選択されるように制御することを特徴とする、オーディオ復号器。

【請求項12】

請求項10又は11に記載のオーディオ復号器であって、前記第2復号化分枝は、前記符号化された信号の中に追加的に含まれた第1処理済信号を逆処理して第1逆処理済信号を取得する第1逆処理分枝(531)を備え、

前記制御可能な第2周波数/時間変換器(534)は、前記第1逆処理済信号のドメインと同じドメインで前記第2符号化済信号を逆処理して第2逆処理済信号を取得する第2逆処理分枝の中に配置され、

前記第2復号化分枝は、前記第1逆処理済信号と前記第2逆処理済信号とを結合して結合済信号を取得する追加的な結合器(532)をさらに備え、

前記結合済信号は前記結合器(600)へと入力されることを特徴とする、オーディオ復号器。

【請求項13】

請求項10乃至12のいずれか1項に記載のオーディオ復号器であって、前記第1周波数/時間変換器(440)と前記第2周波数/時間変換器とは、前記第1符号化済信号と前記第2符号化済信号との中に含まれた時間ドメイン・エイリアシングをキャンセルするためのオーバーラップ/加算ユニット(440c)を有する時間ドメイン・エイリアシング・キャンセル変換器であることを特徴とする、オーディオ復号器。

【請求項14】

請求項10乃至13のいずれか1項に記載のオーディオ復号器であって、前記符号化された信号は、符号化された信号が前記第1符号化済信号であるか前記第2符

10

20

30

40

50

号化済信号であるかを示す符号化モード情報を含み、  
前記符号化モード情報を読み取って、前記符号化された信号を前記第1復号化分枝へ送るか又は前記第2復号化分枝へ送るかを決定する、入力インターフェイス(900)をさらに含むことを特徴とする、オーディオ復号器。

【請求項15】

請求項10乃至14のいずれか1項に記載のオーディオ復号器であって、  
前記第1符号化済信号は算術的に符号化されたものであり、前記第1復号化分枝は算術的復号器を含むことを特徴とする、オーディオ復号器。

【請求項16】

請求項10乃至15のいずれか1項に記載のオーディオ復号器であって、  
前記第1復号化分枝は、前記第1符号化済信号を生成する時に適用された不均一な量子化の結果をキャンセルするための不均一な逆量子化特性を有する逆量子化器を備え、  
前記第2復号化分枝は、前記第2符号化済信号を生成する時に適用された均一な量子化の結果をキャンセルするための逆量子化器を含まない、ことを特徴とする、オーディオ復号器。

10

【請求項17】

請求項10乃至16のいずれか1項に記載のオーディオ復号器であって、  
前記制御器(990)は、複数の使用可能な異なる離散的周波数/時間分解能の内の1つの離散的周波数/時間分解能を適用することにより、前記第1周波数/時間変換器と前記第2周波数/時間変換器とをそれぞれ制御し、前記第2周波数/時間変換器のための前記使用可能な異なる周波数/時間分解能の数は、前記第1周波数/時間変換器のための前記使用可能な異なる周波数/時間分解能の数より大きいことを特徴とする、オーディオ復号器。

20

【請求項18】

請求項10乃至17のいずれか1項に記載のオーディオ復号器であって、  
前記ドメイン変換器は、LPCフィルタ情報を使用して合成信号を生成するLPC合成処理器(540)であり、前記LPCフィルタ情報は前記符号化された信号に含まれていることを特徴とする、オーディオ復号器。

【請求項19】

符号化された信号をオーディオ復号化する方法であって、前記符号化された信号は、スペクトルドメインの第1符号化済信号と、LPC(線形予測符号化)スペクトルドメインの第2符号化済信号と、前記第1符号化済信号及び前記第2符号化済信号を示す情報と、前記第1符号化済信号及び前記第2符号化済信号を復号化するために使用される時間/周波数分解能情報とを含む、方法において、  
制御可能な第1周波数/時間変換器(440)を有する第1復号化分枝(431, 440)により、前記第1符号化済信号を復号化し、第1復号化済信号を取得するステップであって、前記第1周波数/時間変換器(440)は前記第1符号化済信号のための前記時間/周波数分解能情報を用いて前記第1符号化済信号をスペクトルドメインから時間ドメインへ変換する、ステップと、

30

制御可能な第2周波数/時間変換器(534)を有する第2復号化分枝により、前記第2符号化済信号を復号化し、第2復号化済信号を取得するステップであって、前記第2周波数/時間変換器(534)は前記第2符号化済信号のための時間/周波数分解能情報を用いて前記第2符号化済信号をLPCスペクトルドメインからLPCドメインへ変換する、ステップと、

40

前記時間/周波数分解能情報を使用して、前記第1周波数/時間変換器(440)と前記第2周波数/時間変換器(534)とを制御するステップ(990)と、  
ドメイン変換器により、前記第2復号化済信号をLPCドメインから時間ドメインへ変換して合成信号を生成するステップ(540)と、  
前記第1復号化済信号と前記合成信号とを結合(600)し、復号化されたオーディオ信号を取得するステップと、

前記第1復号化済信号と前記合成信号とを結合(600)し、復号化されたオーディオ信号を取得するステップと、

50

を備えることを特徴とする方法。

【請求項 20】

オーディオ信号を記録したコンピュータ読み取り可能な記憶媒体であって、前記オーディオ信号が、  
スペクトルドメインの第 1 符号化済信号と、  
LPC (線形予測符号化) スペクトルドメインの第 2 符号化済信号と、  
前記第 1 符号化済信号及び前記第 2 符号化済信号を示す情報と、  
前記第 1 符号化済信号を復号化するために使用される第 1 の時間 / 周波数分解能情報と、  
前記第 2 符号化済信号を復号化するために使用される第 2 の時間 / 周波数分解能情報と、  
を含む記憶媒体。

10

【請求項 21】

コンピュータ上で作動するときに、請求項 9 又は 19 に記載された方法を実行するためのコンピュータプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、オーディオ符号化に関し、特に低ビットレートでのオーディオ符号化スキームに関する。

【背景技術】

【0002】

当該分野では、MP3 や AAC (高効率オーディオ圧縮符号化) のような周波数ドメインの符号化スキームが公知である。これら周波数ドメインの符号器は、時間ドメイン / 周波数ドメインの変換を基礎とし、その次に、聴覚心理モジュールからの情報を用いて量子化誤差を制御する量子化ステージと、量子化されたスペクトル係数及び対応するサイド情報を符号テーブルを用いてエントロピー符号化する符号化ステージとを有する。

20

【0003】

一方、非特許文献 1 の中で記述されている AMR - WB + (適応型マルチレート・広帯域) のように、スピーチ処理に非常に適合する符号器も存在する。このようなスピーチ符号化スキームは、時間ドメイン信号の LP (線形予測) 濾波を実行する。この LP 濾波は、入力された時間ドメイン信号の線形予測分析から導出される。結果として得られる LP フィルタ係数は、次に量子化 / 符号化され、サイド情報として伝送される。このプロセスは、線形予測符号化 (LPC) として知られる。フィルタの出力では、励振信号として知られる予測残余信号又は予測誤差信号は、ACELP (代数符号励振線形予測) の ABS (合成による分析) ステージを用いて符号化されるか、その代わりに、オーバーラップを伴うフーリエ変換を使用する変換符号器を用いて符号化される。ACELP 符号化と、TCX (変換符号化済励振) 符号化とも呼ばれる変換符号化された励振符号化との間の切替は、閉ループ又は開ループのアルゴリズムを用いて実行される。

30

【0004】

AAC 符号化スキームとスペクトル帯域複製 (SBR) の技術とを組み合わせた高効率 AAC (HE - AAC) 符号化スキーム等の周波数ドメイン・オーディオ符号化スキームは、「MPEG サラウンド」として知られるジョイントステレオや多チャンネル符号化ツールとも組み合わせることができる。

40

【0005】

他方、AMR - WB + 等のようなスピーチ符号器もまた、高周波拡張ステージ及びステレオ機能を備えている。

【0006】

周波数ドメイン符号化スキームは、音楽信号を低ビットレートで高品質に符号化できるという点で有利である。しかし他方、低ビットレートにおけるスピーチの品質が問題となる。

【0007】

50

スピーチ符号化スキームは、スピーチ信号に対しては低ビットレートでも高品質を達成できるが、他方、音楽信号に対しては低ビットレートでは低品質しか達成できない。

【先行技術文献】

【非特許文献】

【0008】

【非特許文献1】3GPP TS 26.290

【非特許文献2】3GPP TS 26.190

【非特許文献3】"Speech Coding: A Tutorial Review", Andreas Spanias, Proceedings of the IEEE, Vol. 82, No. 10, October 1994, pages 1541-1582

【非特許文献4】ISO/IEC 14496-3, subpart 4.1.1.2.

【非特許文献5】ISO/IEC 14496-3, subpart 4, subclause 4.5.2.3.4

【非特許文献6】Table 4.110 to Table 4.128, in ISO/IEC 14496-3, subpart 4, section 4.5.4

【非特許文献7】3GPP TS 26.290 V6.3.0, Extended Adaptive Multi-Rate \_ Wideband (AMR-WB+) codec; Transcoding functions

【非特許文献8】ISO/IEC 14496-3, subpart 4, section 4.5.4 Table 4.129 to Table 4.147

【非特許文献9】ISO/IEC 14496-3:2005, Information technology \_ Coding of audio-visual objects \_ Part 1: Systems

【非特許文献10】ISO/IEC 11172-3:1993, Information technology - Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s, Part 3: Audio.

【非特許文献11】ITU-T Rec.H.222.0(1995) | ISO/IEC 13818-1:2000, Information technology - Generic coding of moving pictures and associated audio information: \_ Part 1: Systems.

【非特許文献12】ISO/IEC 13818-3:1998, Information technology - Generic coding of moving pictures and associated audio information: - Part 3: Audio.

【非特許文献13】ISO/IEC 13818-7:2004, Information technology - Generic coding of moving pictures and associated audio information: - Part 7: Advanced Audio Coding (AAC).

【非特許文献14】ISO/IEC 14496-3:2005, Information technology \_ Coding of audio-visual objects \_ Part 3: Audio

【非特許文献15】ISO/IEC 23003-1:2007, Information technology - MPEG audio technologies - Part 1: MPEG Surround

【非特許文献16】3GPP TS 26.290 V6.3.0, Extended Adaptive Multi-Rate \_ Wideband (AMR-WB+) codec; Transcoding functions

【非特許文献17】3GPP TS 26.190, Adaptive Multi-Rate \_ Wideband (AMR-WB) speech codec; Transcoding functions

【非特許文献18】3GPP TS 26.090, Adaptive Multi-Rate (AMR) speech codec; Transcoding functions

【非特許文献19】ISO/IEC 14496-3, subpart 1, subclause 1.3 (Terms and definitions)

【非特許文献20】3GPP TS 26.290, section 3 (Definitions and abbreviations)

【発明の概要】

【発明が解決しようとする課題】

【0009】

本発明の目的は、符号化及び復号化の改良された概念を提供することである。

【課題を解決するための手段】

【0010】

本発明の目的は、請求項1に記載のオーディオ符号器、請求項9に記載のオーディオ符号

10

20

30

40

50

化方法、請求項 10 に記載のオーディオ復号器、請求項 19 に記載のオーディオ復号化方法、請求項 20 に記載の符号化された信号、又は請求項 21 に記載のコンピュータプログラムにより達成される。

【0011】

本発明は、ハイブリッドモード又はデュアルモードで切り替えられる符号化/復号化スキームが、ある信号特性に対して常に最適な符号化アルゴリズムを選択できるという点で有利である、という知見に基づいている。換言すれば、本発明は、全ての信号の特性に対して完全に適合する 1 つの信号符号化アルゴリズムを探しているわけではない。そのような全般型のスキームは、音楽オーディオ符号器を一方とし、スピーチ符号器を他方とする状態の相互間に大きな差異があることから分かるように、常にある妥協の産物となるであろう。代わりに、本発明は、スピーチ符号化アルゴリズムを一方とし、オーディオ符号化アルゴリズムを他方として、ある切替型のスキーム内で様々な符号化アルゴリズムを組み合わせるものであり、その結果、各オーディオ信号部分のために最も適合する符号化アルゴリズムが選択される。さらに、本発明のもう 1 つの特徴は、両方の符号化分枝が時間/周波数変換器を含み、片方の分枝が L P C 処理器などの追加のドメイン変換器をも備えていることである。このドメイン変換器により、ある信号特性にとっては第 1 分枝よりも第 2 分枝の方がより適合することが確実となる。本発明の他の特徴として、このドメイン処理器により出力された信号がスペクトル表示へと変換されることも挙げられる。

10

【0012】

両方の変換器、即ち第 1 符号化分枝の第 1 変換器と第 2 符号化分枝の第 2 変換器とは、マルチ分解能の変換符号化を適用する。ここでは、対応する変換器の分解能はオーディオ信号に依存するよう設定されており、特に、対応する符号化分枝内で実際に符号化されたオーディオ信号に依存する。その結果、品質を一方としビットレートを他方とする場合の良好な妥協点、又は、ある一定の品質を考慮した場合における最低のビットレート若しくは一定のビットレートを考慮した場合における最高の品質を取得することができる。

20

【0013】

本発明に従えば、2 つの変換器の時間/周波数分解能は、好適には互いに独立するよう設定されており、その結果、各時間/周波数変換器は対応する信号の時間/周波数分解能の要件に対して最適に適合することができる。ビット効率、即ち有用ビットとサイド情報ビットとの間の関係は、より長いブロックサイズ/ウインドウ長の方が高くなる。そのため、両方の変換器がより長いウインドウを偏重する方が好ましい。なぜなら、基本的に、同量のサイド情報は、短いブロックサイズ/ウインドウ長/変換長を適用する場合に比べ、オーディオ信号のより長い時間部分を反映するからである。好適には、符号化分枝内の時間/周波数分解能は、これらの分枝内に位置する他の符号化/復号化ツールからの影響を受けても良い。好適には、L P C 処理器などのドメイン変換器を含む第 2 符号化分枝は、A C E L P 分枝を一方とし T C X スキームを他方とする、もう一つのハイブリッドスキームを含み、ここでは第 2 変換器は T C X スキーム内に含まれる。好適には、T C X 分枝に位置する時間/周波数分解能変換器は符号化の判定からも影響を受け、その結果、第 2 符号化分枝内の信号の 1 つの部分は、第 2 変換器を持つ T C X 分枝内で処理されるか、又は時間/周波数変換器を持たない A C E L P 分枝内で処理される。

30

40

【0014】

基本的に、ドメイン変換器も第 2 符号化分枝も、特に、第 2 符号化分枝内の第 1 処理分枝及び第 2 符号化分枝内の第 2 処理分枝も、スピーチに関連した要素である必要はない。このスピーチに関連した要素とは、ドメイン変換器としての L P C 分析器や、第 2 処理分枝としての T C X 符号器や、第 1 処理分枝としての A C E L P 符号器などである。スピーチを一方とし音楽を他方とするようなオーディオ信号特性とは異なる、他の信号特性が評価されるときには、他のアプリケーションもまた有用である。任意のドメイン変換器や符号化分枝の構成も使用でき、最適なアルゴリズムは、「合成による分析」のスキームにより発見できる。その結果、符号器側では、オーディオ信号の各部分について符号化の選択肢全てが実行され、最適な結果が選択される。このとき、最適な結果とは、符号化の結果に

50

対してある目標関数を適用することで発見されても良い。次に、符号化されたオーディオ信号の所定の部分の根底にある符号化アルゴリズムを復号器に対して指示するサイド情報が、符号器出力インターフェイスによって、その符号化されたオーディオ信号に添付される。その結果、復号器は、符号器側のどの判定又はどの信号特性に対しても注意を払う必要がなく、伝送されたサイド情報に基づいてその復号化分枝を単に選択する。さらに、復号器は、単に正しい復号化分枝を選択するだけでなく、符号化された信号内にある符号化されたサイド情報に基づき、対応する第1復号化分枝と対応する第2復号化分枝とにどの時間/周波数分解能を適用すべきか、を選択する。

**【0015】**

従って、本発明は1つの符号化/復号化スキームを提供し、このスキームは全ての異なる符号化アルゴリズムの利点を結合し、これらの符号化アルゴリズムの欠点、即ち、所定の符号化アルゴリズムには適さないアルゴリズムによって信号部分を符号化せざるを得なかったときに発生するような欠点を防止する。さらに本発明は、異なる符号化分枝内の異なるオーディオ信号部分によって挙げられた異なる時間/周波数分解能の要件が考慮されなかった場合に、招来するであろうあらゆる欠点を回避できる。代わりに、両方の分枝における時間/周波数変換器の可変的な時間/周波数分解能のために、同じ時間/周波数分解能が両方の符号化分枝に適用された場合、又は1つの固定的な時間/周波数分解能だけがどちらの符号化分枝でも使用可能である場合において発生するであろう、あらゆるアーチファクトが少なくとも低減され、あるいは完全に防止される。

**【0016】**

第2のスイッチも2つの処理分枝の間で切り替えを実行するが、しかし、「外側の」第1分枝のドメインとは異なるドメインにおいて実行する。「内側」の分枝の一方では、主としてソースモデルやSNR計算が使用され、「内側」の分枝の他方では、シンクモデル及び/又は聴覚心理モデル、即ちマスキングが使用されても良く、或いは、少なくとも周波数/スペクトルドメイン符号化の態様を含む。一例として、一方の「内側」分枝は周波数ドメイン符号器/スペクトル変換器を備え、他方の分枝は、LPCドメイン等の他のドメインで符号化を実行する符号器、例えばスペクトル変換なしで入力信号を処理するCELP(符号励振線形予測)やACELPの量子化器/スケーラを備える。

**【0017】**

本発明の他の好適な実施例は、スペクトルドメイン符号化分枝のような情報シンク指向の第1符号化分枝と、LPCドメイン符号化分枝のような情報ソース又はSNR指向の第2符号化分枝と、これら第1符号化分枝と第2符号化分枝との間で切り替えを実行するスイッチと、を備えたオーディオ符号器である。この第2符号化分枝は、励振信号を生成するLPC分析ステージのような、時間ドメインとは異なる特定ドメインへの変換器を有し、さらに、この第2符号化分枝は、LPCドメイン処理分枝のような特定のドメインと、LPCスペクトルドメイン処理分枝のような特定のスペクトルドメインと、これら特定ドメイン処理分枝と特定スペクトルドメイン処理分枝との間で切り替えを実行する追加的なスイッチと、を備えている。

**【0018】**

本発明の他の好適な実施例は、スペクトルドメインのような第1ドメインの復号化分枝と、励振信号等の信号をLPCドメインのような第2ドメインで復号化する復号化分枝と、励振信号等の信号をLPCスペクトルドメインのような第3ドメインで復号化する復号化分枝と、を備えたオーディオ復号器であり、この第3ドメインは第2ドメインからの周波数変換により取得される。このオーディオ復号器は、第2ドメイン信号と第3ドメイン信号との間を切り替える第1スイッチと、第1ドメインの復号器と第2ドメイン又は第3ドメインの復号器との間を切り替える第2スイッチと、を含む。

**【0019】**

次に、本発明の好適な実施の形態を、添付の図面を参照しながら説明する。

**【図面の簡単な説明】****【0020】**

- 【図 1 a】本発明の第 1 実施形態に係る符号化スキームのブロック図である。
- 【図 1 b】本発明の第 1 実施形態に係る復号化スキームのブロック図である。
- 【図 1 c】本発明の他の実施形態に係る符号化スキームのブロック図である。
- 【図 2 a】本発明の第 2 実施形態に係る符号化スキームのブロック図である。
- 【図 2 b】本発明の第 2 実施形態に係る復号化スキームの概要ブロック図である。
- 【図 2 c】本発明の他の実施形態に係る符号化スキームのブロック図である。
- 【図 3 a】本発明の他の実施形態に係る符号化スキームのブロック図である。
- 【図 3 b】本発明の他の実施形態に係る復号化スキームのブロック図である。
- 【図 3 c】縦列型スイッチを有する符号化装置 / 方法の概要図である。
- 【図 3 d】縦列型結合器を使用する復号化装置又は方法の概要図である。 10
- 【図 3 e】時間ドメイン信号と、符号化済信号の対応する表示とを示し、2 つの符号化済信号に含まれた短いクロスフェード領域を示す図である。
- 【図 4 a】符号化分枝の前にスイッチを配置したブロック図である。
- 【図 4 b】符号化分枝の次にスイッチを配置した符号化スキームのブロック図である。
- 【図 5 a b】図 5 a は時間ドメインのスピーチ部分の波形を、準周期的又はインパルス状の信号部分として示す図であり、図 5 b は図 5 a の信号分節のスペクトルを示す図である。
- 【図 5 c d】図 5 c は無声音スピーチの時間ドメインスピーチ部分をノイズ状部分の一例として示す図であり、図 5 d は図 5 c の時間ドメイン波形のスペクトルを示す図である。
- 【図 6】合成による分析 C E L P 符号化手段のブロック図である。 20
- 【図 7 a】スピーチ生成システムの線形モデルを示す図である。
- 【図 7 b】スピーチ生成システムの全極フィルタモデルを示す図である。
- 【図 7 c】有声音の励振信号をインパルス状の信号の一例として示す。
- 【図 7 d】無声音の励振信号をインパルス状の信号の一例として示す。
- 【図 7 e】短期間予測情報及び予測誤差 (励振) 信号を提供する符号器側の L P C ステージを示す。
- 【図 7 f】重み付き信号を生成する L P C 装置のさらなる実施例を示す。
- 【図 7 g】図 2 b の変換器 5 3 7 において必要な、逆重み付け操作と後続の励振分析との適用による、重み付き信号から励振信号への変換の構成を示す。
- 【図 8】本発明のある実施形態に係るジョイント多チャネルアルゴリズムのブロック図である。 30
- 【図 9】帯域拡張アルゴリズムの好適なある実施形態を示す。
- 【図 1 0 a】開ループの切替判定を実行する場合のスイッチを詳細に示す。
- 【図 1 0 b】閉ループの切替判定モードで操作する場合のスイッチを示す。
- 【図 1 1 a】本発明の他の態様に従うオーディオ符号器のブロック図を示す。
- 【図 1 1 b】本発明のオーディオ復号器の他の実施例のブロック図を示す。
- 【図 1 2 a】本発明の符号器の他の実施例を示す。
- 【図 1 2 b】本発明の復号器の他の実施例を示す。
- 【図 1 3 a】分解能とウィンドウ / 変換長との間の相互関係を示す。
- 【図 1 3 b】第 1 符号化分枝のための変換ウィンドウのセットと、第 1 符号化分枝から第 2 符号化分枝への遷移を概略的に示す。 40
- 【図 1 3 c】第 1 符号化分枝のためのウィンドウシーケンスと第 2 符号化分枝への遷移のためのシーケンスとを含む、複数の異なるウィンドウシーケンスを示す。
- 【図 1 4 a】第 2 符号化分枝の好適な実施例のフレーミングを示す。
- 【図 1 4 b】第 2 符号化分枝に適用された短いウィンドウを示す。
- 【図 1 4 c】第 2 符号化分枝に適用された中程度のサイズのウィンドウを示す。
- 【図 1 4 d】第 2 符号化分枝に適用された長いウィンドウを示す。
- 【図 1 4 e】1 つのスーパーフレーム分割における A C E L P フレームと T C X フレームの例示的なシーケンスを示す。
- 【図 1 4 f】第 2 符号化分枝のための様々な時間 / 周波数分解能に対応する様々な変換長 50

を示す。

【図 1 4 g】図 1 4 f の定義を使用したウインドウの構築を示す。

【発明を実施するための形態】

【0021】

図 1 1 a はオーディオ信号を符号化するオーディオ符号器の一実施例を示す。この符号器は、第 1 符号化アルゴリズムを用いてオーディオ信号を符号化して第 1 符号化済信号を取得するための、第 1 符号化分枝 4 0 0 を含む。

【0022】

オーディオ符号器はさらに、第 2 符号化アルゴリズムを用いてオーディオ信号を符号化して第 2 符号化済信号を取得するための、第 2 符号化分枝 5 0 0 を含む。第 1 符号化アルゴリズムは第 2 符号化アルゴリズムとは異なる。さらに、オーディオ信号の 1 つの部分について第 1 符号化済信号又は第 2 符号化済信号のいずれかが符号器出力信号 8 0 1 の中に存在するように、第 1 符号化分枝と第 2 符号化分枝との間を切り替えるためのスイッチ 2 0 0 が設けられている。

【0023】

図 1 1 a に示すオーディオ符号器は、信号分析器 3 0 0 / 5 2 5 をさらに備え、この分析器はオーディオ信号の一部を分析し、オーディオ信号のその部分が符号器出力信号 8 0 1 において第 1 符号化済信号として表現されるか、又は第 2 符号化済信号として表現されるかを決定する。

【0024】

信号分析器 3 0 0 / 5 2 5 はさらに、第 1 符号化分枝 4 0 0 内の第 1 変換器 4 1 0 の時間 / 周波数分解能、又は第 2 符号化分枝 5 0 0 内の第 2 変換器 5 2 3 の時間 / 周波数分解能をそれぞれ可変的に決定する。オーディオ信号のある部分を表現する第 1 符号化済信号又は第 2 符号化済信号が生成されたとき、この時間 / 周波数分解能が適用される。

【0025】

オーディオ符号器は、オーディオ信号の前記部分の符号化済表現と、そのオーディオ信号の表現が第 1 符号化済信号であるか第 2 符号化済信号であるかを示し且つその第 1 符号化済信号及び第 2 符号化済信号を復号化するために使用する時間 / 周波数分解能を示す情報と、を含む符号器出力信号 8 0 1 を生成するための、出力インターフェイス 8 0 0 をさらに備えている。

【0026】

第 2 符号化分枝は、好適には以下の点において第 1 符号化分枝とは異なる。即ち、第 2 符号化分枝は、オーディオ信号が第 1 符号化分枝内で処理されるドメインからある別のドメインへとそのオーディオ信号を変換するためのドメイン変換器をさらに備えている。好適には、このドメイン変換器は L P C 処理器 5 1 0 であるが、しかし、このドメイン変換器は、第 1 変換器 4 1 0 及び第 2 変換器 5 2 3 とから異なる限り、任意の方法で構成されても良い。

【0027】

第 1 変換器 4 1 0 は時間 / 周波数変換器であり、好適にはウインドウ化手段 4 1 0 a と変換手段 4 1 0 b とを備える。ウインドウ化手段 4 1 0 a は、入力されたオーディオ信号に対して分析ウインドウを適用し、変換手段 4 1 0 b はそのウインドウ化された信号をスペクトル表示へと変換する。

【0028】

同様に、第 2 変換器 5 2 3 は、好適にはウインドウ化手段 5 2 3 a とその次に接続された変換手段 5 2 3 b とを備える。ウインドウ化手段 5 2 3 a はドメイン変換器 5 1 0 から出力された信号を受け取り、そのウインドウ化された表現を出力する。ウインドウ化手段 5 2 3 a により適用された 1 つの分析ウインドウの結果は、変換手段 5 2 3 b へと入力され、あるスペクトル表現を形成する。この変換手段は F F T でも良く、又は好適には M D C T 処理器でも良く、ソフトウェア内、ハードウェア内又はハードウェア / ソフトウェアが混合した構成内において、対応するアルゴリズムを実行する。代替的に、この変換手段は

10

20

30

40

50

、プロトタイプフィルタの実数値又は複素の変調に基づく、例えばQMFフィルタバンクのようなフィルタバンク構成であっても良い。特定のフィルタバンク構成に対し、あるウィンドウが適用される。しかし、他のフィルタバンク構成に対しては、FFT又はMDCTに基づく変換アルゴリズムのために必要とされるようなウィンドウ処理は使用されない。もし、可変分解能フィルタバンクが使用された場合、そのフィルタバンクは周波数分解能を制御し且つ時間分解能をも制御するか、又は、周波数分解能だけを制御して時間分解能は制御しない。しかし、この変換器がFFT、MDCT又は他の対応する任意の変換器として構成された場合には、時間的に大きなブロック長により得られた周波数分解能の増大分が、低い時間分解能に自動的に対応し、その逆もまた真となるという方法で、周波数分解能と時間分解能とが関連付けられている。

10

## 【0029】

さらに、第1符号化分枝は量子化器/符号器ステージ421を含んでも良く、第2符号化分枝も、1つ又は複数の追加的符号化ツール524を含んでも良い。

## 【0030】

重要な点は、前記信号分析器が第1変換器410及び第2変換器523のための分解能制御信号を生成するという点である。それにより、一方では低いビットレートを提供し、他方ではその低いビットレートから見た最高品質を提供するような符号化のスキームを達成できるように、両方の符号化分枝において独立した分解能制御が実行される。低いビットレートという目的を達成するためには、長いウィンドウ長又は長い変換長が好ましい。しかし、これら長いウィンドウ長や長い変換長が、低い時間分解能に起因するアーチファクトを招く場合には、短いウィンドウ長や短い変換長が適用され、それらは低い周波数分解能をもたらす。好適には、前記信号分析器は、符号化分枝内の対応するアルゴリズムに適したある統計的分析又は他の任意の分析を適用する。第1符号化分枝がAACベースの符号器などのような周波数ドメインの符号化分枝であり、第2符号化分枝がLPC処理器510をドメイン変換器として含む、ある実施モードにおいては、信号分析器がスピーチ/音楽の判別を実行し、その結果、オーディオ信号のスピーチ部分が第2符号化分枝へと送られるようにスイッチ200が制御される。オーディオ信号の音楽部分は、図中にスイッチ制御ラインで示すように、第1符号化分枝400へと伝送されるようにスイッチ200が制御される。代替的に、図1C又は図4Bに関して後述するように、このスイッチは出力インターフェイス800の前に配置されても良い。

20

30

## 【0031】

さらに、前記信号分析器は、スイッチ200へと入力されたオーディオ信号を受け取ることができ、或いはこのスイッチ200により出力されたオーディオ信号を受け取ることができる。さらに、信号分析器は、前記オーディオ信号を対応する符号化分枝へと供給するためだけでなく、信号分析器と変換器とを接続している分解能制御ラインにより示されるように、対応する符号化分枝内の各変換器(第1変換器410と第2変換器523)の適切な時間/周波数分解能を決定するためにも、分析を実行する。

## 【0032】

図11bは、図11aのオーディオ符号器に適合するオーディオ復号器の好適な実施例を示す。

40

## 【0033】

図11bの復号器は、図11aの出力インターフェイス800により出力された符号器出力信号801のような、符号化されたオーディオ信号を復号化する。符号化された信号は、第1符号化アルゴリズムに従って符号化された第1符号化済オーディオ信号と、第1符号化アルゴリズムとは異なる第2符号化アルゴリズムに従って符号化された第2符号化済オーディオ信号と、第1符号化済信号及び第2符号化済信号を復号化するために、第1符号化アルゴリズム又は第2符号化アルゴリズムを使用するか否かを指示する情報と、第1符号化済オーディオ信号及び第2符号化済オーディオ信号のための時間/周波数分解能情報と、を含む。

## 【0034】

50

オーディオ復号器は、第1符号化アルゴリズムに基づいて第1符号化済信号を復号化するための第1復号化分枝431, 440を含む。さらに、オーディオ復号器は、第2符号化アルゴリズムを用いて第2符号化済信号を復号化するための第2復号化分枝を含む。

【0035】

第1復号化分枝は、スペクトルドメインから時間ドメインへと変換するための制御可能な第1変換器440を備える。この制御可能な変換器は、第1符号化済信号からの時間/周波数分解能情報を使用して制御され、第1復号化済信号を取得する。

【0036】

第2復号化分枝は、スペクトル表示から時間表示へと変換するための制御可能な第2変換器を備え、この制御可能な第2変換器534は第2符号化済信号のための時間/周波数分解能情報991を使用して制御される。

10

【0037】

この復号器は、時間/周波数分解能情報991に従って第1変換器440と第2変換器534とを制御するための、制御手段990をさらに備える。

【0038】

さらにこの復号器は、図11aの符号器内のドメイン変換器510によって適用されたドメイン変換をキャンセルするように、第2復号化済信号を使用して合成信号を生成するための、ドメイン変換器を備える。

【0039】

好適には、このドメイン変換器540はLPC合成処理器であり、符号化済信号の中に含まれたLPCフィルタ情報を使用して制御される。このLPCフィルタ情報は図11a内のLPC処理器510によって生成されたものであり、符号器の出力信号の中にサイド情報として挿入されたものである。最後に、オーディオ復号器は結合器600を備え、第1ドメイン変換器440により出力された第1復号化済信号と、前記合成信号とを結合し、復号化されたオーディオ信号609を取得する。

20

【0040】

この好適な実施例においては、第1復号化分枝は逆量子化/復号化ステージ431をさらに備え、このステージは、対応する符号化ステージ421によって実行された操作を逆戻しするか、又は少なくとも部分的に逆戻しする。しかし、量子化は損失が多い操作であるため、量子化を逆戻しできないことは明らかである。とはいえ、逆量子化器は、対数(logarithmic)又は圧伸(companding)量子化のような量子化における所定の不均一性を逆戻しする。

30

【0041】

第2復号化分枝においては、ステージ524により適用された所定の符号化操作をもとに戻すための対応するステージ533が適用される。好適には、ステージ524は均一な量子化を含む。そして、対応するステージ533は、所定の均一な量子化をもとに戻すための特定の逆量子化ステージを持たないであろう。

【0042】

第1変換器440及び第2変換器534は、それぞれ逆変換手段ステージ440a, 534aと、合成ウインドウステージ440b, 534bと、次に接続されたオーバーラップ/加算ステージ440c, 534cとを備える。変換器、詳しくは変換手段ステージ440a, 534aが、変形離散コサイン変換などのエイリアシングを導入する変換を適用した場合には、オーバーラップ/加算ステージが必要となる。次に、オーバーラップ/加算操作は、時間ドメインのエイリアシング・キャンセル操作(TDAC)を実行する。しかし、変換手段が、逆FFTなどのエイリアシングを導入しない変換を適用した場合には、オーバーラップ/加算ステージ440cは不要となる。そのような構成においては、プロッキングアーチファクトを防止するためのクロスフェーディング操作が適用されても良い。

40

【0043】

同様に、結合器600は、切替型の結合器若しくはクロスフェード型の結合器であっても

50

良く、又は、ブロッキングアーチファクトを防止するためにエイリアシングが使用された場合には、その分枝そのものの中にあるオーバーラップ/加算ステージと同様に、遷移(transition)ウィンドウ処理操作が結合器によって実行される。

【0044】

図1aは、2つの縦列型スイッチを有する本発明の一実施例を示す。モノラル信号、ステレオ信号若しくは多チャンネル信号が、スイッチ200へと入力される。スイッチ200は判定ステージ300により制御される。ブロック200へと入力される信号は、判定ステージにも入力として受け取られる。代替的に、判定ステージ300は、上記モノラル信号、ステレオ信号若しくは多チャンネル信号の中に含まれたサイド情報を受け取っても良く、又は、例えば上記モノラル信号、ステレオ信号若しくは多チャンネル信号を元々製造した時に生成された信号であって情報が存在する信号に対して、少なくとも関連付けられたサイド情報を受け取っても良い。

10

【0045】

判定ステージ300はスイッチ200を活性化して、図1aの上側の分枝で示す周波数符号化部分400か、又は図1aの下側の分枝で示すLPCドメイン符号化部分500か、いずれかに対して信号を送る。周波数ドメイン符号化分枝の鍵となる要素は、(後述するような)共通前処理ステージの出力信号をスペクトルドメインへと変換する、スペクトル変換ブロック410である。このスペクトル変換ブロックは、MDCTアルゴリズム、QMF、FFTアルゴリズム、ウェーブレット分析などを含んでも良く、又は、所定個数のフィルタバンクチャンネルを備え、このフィルタバンク内のサブバンド信号が実数値信号或いは複素値信号でも良い、臨界的にサンプリングされたフィルタバンクを含んでも良い。このスペクトル変換ブロック410の出力は、スペクトルオーディオ符号器421を用いて符号化され、この符号器はAAC符号化のスキームから知られる処理ブロックを含んでも良い。

20

【0046】

一般的に、分枝400内の処理は知覚ベースモデル又は情報シンクモデルにおける処理である。従って、この分枝は音を受け取る人間の聴覚システムをモデル化したものである。対照的に、分枝500内の処理は、励振、残余又はLPCドメインにおいて信号を生成するためのものである。一般的に、この分枝500の処理はスピーチモデル又は情報生成モデルにおける処理である。スピーチ信号に対しては、このモデルは音を生成する人間のスピーチ/音生成システムのモデルである。しかし、もし異なる音生成モデルを必要とする異なるソースからの音を符号化すべき場合には、分枝500における処理も異なっても良い。

30

【0047】

下方の符号化分枝500において、鍵となる要素はLPC装置510であり、この装置510はLPCフィルタの特性を制御するために使用されるLPC情報を出力する。このLPC情報は復号器に伝達される。LPCステージ510の出力信号は、励振信号及び/又は重み付き信号からなるLPCドメイン信号である。

【0048】

LPC装置は一般的にLPCドメイン信号を出力し、この信号は、図7eの励振信号や図7fの重み付き信号、又は他の信号等、LPCドメイン内のいかなる信号であっても良く、これらの信号はLPCフィルタ係数をオーディオ信号へと適用することで生成されたものである。LPC装置は、これらの係数を決定することができ、これらの係数を量子化/符号化することもできる。

40

【0049】

判定ステージ300における判定は信号適応型であっても良く、判定ステージは音楽/スピーチ判別を実行し、音楽信号は上方の分枝400へと入力され、スピーチ信号は下方の分枝500へと入力されるようにスイッチ200を制御する。一実施例においては、判定ステージはその判定情報を出力ビットストリームへと提供し、その結果、復号器はこの判定情報を使用して正確な復号操作が実行可能になる。

50

## 【 0 0 5 0 】

図 1 b はこのような復号器を示す。スペクトルオーディオ符号化手段 4 2 1 から出力された信号は、伝送後、スペクトルオーディオ復号器 4 3 1 へと入力される。スペクトルオーディオ復号器 4 3 1 の出力は、時間ドメインの変換器 4 4 0 へと入力される。同様に、図 1 a の L P C ドメイン符号化分枝 5 0 0 の出力は、復号器側で受信され、要素 5 3 1 , 5 3 3 , 5 3 4 , 5 3 2 によって処理され、L P C 励振信号が取得される。L P C 励振信号は L P C 合成ステージ 5 4 0 に入力され、この合成ステージは、さらなる入力として、対応する L P C 分析ステージ 5 1 0 により生成された L P C 情報を受け取る。時間ドメインの変換器 4 4 0 の出力及び / 又は L P C 合成ステージ 5 4 0 の出力は、スイッチ 6 0 0 へと入力される。このスイッチ 6 0 0 はスイッチ制御信号を介して制御されるが、スイッチ制御信号は、例えば判定ステージ 3 0 0 により生成されたか、又は、オリジナルのモノラル信号、ステレオ信号若しくは多チャンネル信号の作成者等により外部的に提供されたものである。スイッチ 6 0 0 の出力は、完全なモノラル信号、ステレオ信号又は多チャンネル信号である。

10

## 【 0 0 5 1 】

スイッチ 2 0 0 と判定ステージ 3 0 0 への入力は、モノラル信号、ステレオ信号、多チャンネル信号などの一般的なオーディオ信号で良い。スイッチ 2 0 0 の入力信号、又はスイッチ 2 0 0 の入力信号の根底にあるオリジナルオーディオ信号の作成者等の外部的なソース、から導出される判定に従い、スイッチは周波数符号化分枝 4 0 0 と L P C 符号化分枝 5 0 0 との間で切替を実行する。周波数符号化分枝 4 0 0 は、スペクトル変換ステージ 4 1 0 と、後続の量子化 / 符号化ステージ 4 2 1 とを有する。量子化 / 符号化ステージは、A A C 符号化手段のような現代の周波数ドメイン符号化手段では公知であるいかなる機能を含んでも良い。さらに、量子化 / 符号化ステージ 4 2 1 内の量子化操作は、周波数に係る聴覚心理マスキング閾値等の聴覚心理情報を生成する聴覚心理モジュールを介して制御されても良く、この場合、その情報はステージ 4 2 1 へと入力される。

20

## 【 0 0 5 2 】

L P C 符号化分枝 5 0 0 では、スイッチからの出力信号は、L P C サイド情報及び L P C ドメイン信号を生成する L P C 分析ステージ 5 1 0 を介して処理される。本発明の励振符号器は、次に、L P C ドメイン信号を量子化 / 符号化ステージ 5 2 2 において L P C ドメインで処理するか、或いは量子化 / 符号化ステージ 5 2 4 において L P C スペクトルドメインで処理するか、の間で切替を実行する追加的なスイッチ 5 2 1 を有する。この L P C スペクトルドメイン処理のために、スペクトル変換器 5 2 3 が量子化 / 符号化ステージ 5 2 4 の入力側に設けられる。スイッチ 5 2 1 は、例えば A M R - W B + の技術仕様書に記載されたような特定の設定に依存して、開ループ状態又は閉ループ状態で制御される。

30

## 【 0 0 5 3 】

閉ループの制御モードのために、本発明の符号器は、L P C ドメイン信号のための逆の量子化 / 符号化器 5 3 1 と、L P C スペクトルドメイン信号のための逆の量子化 / 符号化器 5 3 3 と、この逆の量子化 / 符号化器 5 3 3 の出力を受け入れる逆スペクトル変換器 5 3 4 と、をさらに備える。第 2 符号化分枝の 2 つの処理分枝内にある符号化され再度逆符号化された両方の信号は、スイッチ制御装置 5 2 5 へと入力される。スイッチ制御装置 5 2 5 内では、これら 2 つの出力信号は互いに比較され、及び / 又は目標関数と比較される。目標関数は両方の信号内の歪の比較に基づいて計算されても良く、その結果、スイッチ 5 2 1 の切替位置は、より低い歪を有する信号を使用して決定される。代替的に、両方の分枝が一定でないビットレートを提供する場合には、より低いビットレートを提供する分枝の方を、たとえその分枝の信号対雑音比が他の分枝の信号対雑音比よりも低い場合であっても、選択しても良い。代替的に、目標関数は、各信号の信号対雑音比と各信号のビットレート及び / 又は追加的な基準を入力として使用し、特定の目的のための最良の判定を見つけるようにしても良い。もし、例えばこの目的が「ビットレートをできる限り低くすること」であれば、目標関数は、要素 5 3 1 と 5 3 4 から出力される 2 つの信号のビットレートに大きく依存することになるであろう。しかし、主目的が「所定のビットレートで最

40

50

高品質を有すること」であれば、スイッチ制御装置 5 2 5 は、例えば許容ビットレートを超える各信号を捨て去り、もし両方の信号が許容ビットレート以下であれば、スイッチ制御装置は、より良好な信号対雑音比を有する信号、即ちより小さい量子化 / 符号化歪を有する信号を選択するであろう。

【 0 0 5 4 】

上述したように、本発明に従う復号化のスキームを図 1 b に示す。上述の発生可能な 3 種類の各出力信号のために、特別な復号化 / 逆量子化ステージ 4 3 1 , 5 3 1 , 5 3 3 が存在する。ステージ 4 3 1 は時間スペクトル(周波数ドメイン信号)を出力し、周波数 / 時間変換器 4 4 0 を使用して時間ドメインへと変換される一方で、ステージ 5 3 1 は L P C ドメイン信号を出力し、装置 5 3 3 は L P C スペクトルを出力する。スイッチ 5 3 2 への入力信号の両方が L P C ドメインであることを確保するため、L P C スペクトル / L P C 変換器 5 3 4 が設けられている。スイッチ 5 3 2 の出力データは、L P C 合成ステージ 5 4 0 を用いて時間ドメインへと逆変換され、この合成ステージは、符号器側で生成され伝送された L P C 情報を介して制御されている。ブロック 5 4 0 の後では両方の分枝が時間ドメイン情報を備えており、スイッチ制御信号に従って切替が実行され、最終的には、図 1 a に記載の符号化スキームに入力された信号に依存してモノラル信号、ステレオ信号又は多チャンネル信号等のオーディオ信号が取得される。

【 0 0 5 5 】

図 1 c は、図 4 b の原理に近いスイッチ 5 2 1 の異なる配置を備えた他の実施例を示す。

【 0 0 5 6 】

図 2 a は本発明の第 2 の態様に従う好適な符号化スキームを示す。スイッチ 2 0 0 の入力側に接続された共通前処理のスキームは、サラウンド / ジョイントステレオブロック 1 0 1 を含み、このブロック 1 0 1 はジョイントステレオパラメータとモノラル出力信号とを出力として生成し、このモノラル出力信号は、2 つ以上のチャンネルを有する入力信号をダウンミックスすることで生成したものである。一般的に、ブロック 1 0 1 の出力における信号は 2 つ以上のチャンネルを有する信号であっても良いが、ブロック 1 0 1 のダウンミックス機能により、ブロック 1 0 1 の出力におけるチャンネル数はブロック 1 0 1 へ入力されたチャンネル数よりも小さくなるであろう。

【 0 0 5 7 】

共通前処理のスキームは、ブロック 1 0 1 に対して代替的に、或いはブロック 1 0 1 に対して追加的に、帯域拡張ステージ 1 0 2 を含む。図 2 a に示す実施例では、ブロック 1 0 1 の出力は帯域拡張ブロック 1 0 2 へと入力され、このブロック 1 0 2 は、図 2 a の符号器においては、低帯域信号又は低域通過信号等の帯域制限された信号を出力する。好適には、この信号はさらに(例えば係数 2 で)ダウンサンプルされる。加えて、ブロック 1 0 2 に入力された信号の高帯域のために、スペクトル包絡パラメータ、逆濾波パラメータ、ノイズフロアパラメータ等、M P E G - 4 の H E - A A C プロファイルから公知である帯域拡張(B W E)パラメータが生成され、ビットストリームマルチプレクサ 8 0 0 へと入力される。

【 0 0 5 8 】

好適には、判別ステージ 3 0 0 はブロック 1 0 1 に入力された信号やブロック 1 0 2 に入力された信号を受け取り、例えば音楽モードかスピーチモードかの間で判定を実行する。音楽モードの場合には上方の符号化分枝 4 0 0 が選択され、スピーチモードの場合には下方の符号化分枝 5 0 0 が選択される。好適には、判定ステージはジョイントブロック 1 0 1 及び / 又は帯域拡張ブロック 1 0 2 をさらに制御し、これらブロックの機能を特定の信号に対して適合させる。つまり、判定ステージが、入力信号の所定の時間部分は音楽モードのような第 1 モードであると判定した場合には、ブロック 1 0 1 及び / 又はブロック 1 0 2 の所定の特性を判別ステージ 3 0 0 により制御することができる。一方、判定ステージ 3 0 0 が、入力信号はスピーチモード又は一般的には第 2 の L P C ドメインモードであると判定した場合には、ブロック 1 0 1 及び 1 0 2 の所定の特性を判別ステージの出力に従って制御することができる。

## 【 0 0 5 9 】

好適には、符号化分枝 4 0 0 のスペクトル変換は、M D C T 操作、さらに好適には時間ワーピングされた M D C T 操作を使用して実行されてもよく、このとき、強度又は一般的にはワーピング強度は、ゼロから一定の高さのワーピング強度までの間に制御されても良い。ワーピング強度がゼロの場合、ブロック 4 1 1 における M D C T 操作は当業者の中では公知の単純な M D C T 操作である。時間ワーピング強度と時間ワーピングサイド情報とが一緒に、サイド情報としてビットストリームマルチプレクサ 8 0 0 へと伝送 / 入力されても良い。

## 【 0 0 6 0 】

L P C 符号化分枝においては、L P C ドメインの符号化手段は、ピッチゲイン、ピッチラ  
グ及び / 又はコードブック指標やゲインのようなコードブック情報を計算する、A C E L  
P コア 5 2 6 を含んでも良い。非特許文献 1 から知られるような T C X モードでは、聴覚  
心理的に重み付けられた信号が変換ドメインで処理される。フーリエ変換された重み付き  
信号は、ノイズファクタ量子化を伴う分離マルチレート格子 (split multi-rate lattice)  
量子化 (代数 V Q ) を使用して量子化される。変換は、1 0 2 4 , 5 1 2 又は 2 0 5 6  
サンプルのウィンドウの中で計算される。励振信号は、量子化された重み付き信号を逆の  
重み付けフィルタを通して逆濾波することで回復される。

10

## 【 0 0 6 1 】

第 1 符号化分枝 4 0 0 においては、スペクトル変換器は、好適には、所定の窓関数を有す  
る特別に適合化された M D C T 操作と、それに続く量子化 / エントロピー符号化ステー  
ジとを含んでいる。この量子化 / エントロピー符号化ステージは単一のベクトル量子化ステー  
ジで構成されても良いが、好適には、周波数ドメイン符号化分枝内、即ち図 2 a のステー  
ジ 4 2 1 内の量子化 / 符号化器に類似した結合型のスカラー量子化 / エントロピー符号  
化器が望ましい。

20

## 【 0 0 6 2 】

第 2 符号化分枝には、L P C ブロック 5 1 0 とそれに続くスイッチ 5 2 1 があり、さらに  
A C E L P ブロック 5 2 6 又は T C X ブロック 5 2 7 が続いている。A C E L P は非特許  
文献 2 に記載され、T C X は非特許文献 1 に記載されている。一般的に、A C E L P ブロ  
ック 5 2 6 は図 7 e に記載の処理によって計算された L P C 励振信号を受け取り、T C X  
ブロック 5 2 7 は図 7 f によって生成された重み付き信号を受け取る。

30

## 【 0 0 6 3 】

T C X においては、L P C ベースの重み付けフィルタを通じて入力信号を濾波することで  
計算された重み付き信号に対し、変換が適用される。本発明の好適な実施例において使用  
される重み付けフィルタは、 $(1 - A(z/\gamma)) / (1 - \mu z^{-1})$  である。従って、重  
み付き信号は L P C ドメイン信号であり、その変換型は L P C スペクトルドメインである  
。A C E L P ブロック 5 2 6 により処理された信号は励振信号であり、ブロック 5 2 7 に  
より処理された信号とは異なるが、両方の信号とも L P C ドメインである。

## 【 0 0 6 4 】

図 2 b に示す復号器側では、ブロック 5 3 7 における逆スペクトル変換の後で、重み付け  
フィルタの逆、即ち  $(1 - \mu z^{-1}) / (1 - A(z/\gamma))$  が適用される。その後、この  
信号は  $(1 - A(z/\gamma))$  を通して濾波され、L P C 励振ドメインになる。このように L P  
C ドメインブロック 5 3 4 と T C X<sup>-1</sup> ブロック 5 3 7 への変換は、逆変換と、その後の  
【数 6 4】

40

$$\frac{(1 - \mu z^{-1})}{(1 - A(z/\gamma))} (1 - A(z))$$

を通した濾波とを含み、重み付きドメインから励振ドメインへと変換される。

## 【 0 0 6 5 】

50

図 1 a , 図 1 c , 図 2 a , 図 2 c における装置 5 1 0 は単一のブロックを示しているが、ブロック 5 1 0 は L P C ドメインである限り様々な信号を出力できる。励振信号モードや重み付き信号モード等、ブロック 5 1 0 の実際のモードは実際のスイッチの状態に依存する。代替的に、ブロック 5 1 0 は 2 つの平行処理装置を持つことができ、この場合、1 つの装置は図 7 e に類似する構成を持ち、他の装置は図 7 f のような構成を持つ。そのため、装置 5 1 0 の出力における L P C ドメインは、L P C 励振信号、L P C 重み付き信号、又は他のいずれかの L P C ドメイン信号を表現することができる。

【 0 0 6 6 】

図 2 a 又は図 2 c に示す第 2 符号化分枝 ( A C E L P / T C X ) において、信号は、符号化の前にフィルタ (  $1 - 0.68z^{-1}$  ) により事前強調されてもよい。図 2 b の A C E L P / T C X 復号器においては、合成された信号はフィルタ  $1 / ( 1 - 0.68z^{-1} )$  により非強調される。この事前強調は、信号を L P C 分析及び量子化の前に事前強調する L P C ブロック 5 1 0 の一部で実施することができる。同様に、非強調は L P C 合成ブロック L P C  $^{-1}$  5 4 0 の一部で実施することができる。

10

【 0 0 6 7 】

図 2 c は図 2 a に示す構成の他の実施例を示すが、スイッチ 5 2 1 の配置は異なり、図 4 b の原理に近い。

【 0 0 6 8 】

好適な実施例では、第 1 スイッチ 2 0 0 ( 図 1 a 又は図 2 a を参照 ) は ( 図 4 a のように ) 開ループ判定を通じて制御され、第 2 スイッチは ( 図 4 b のように ) 閉ループ判定を通じて制御される。

20

【 0 0 6 9 】

例えば、図 2 c では、第 2 スイッチ 5 2 1 は図 4 b のように A C E L P 及び T C X 分枝の後に配置されている。第 1 処理分枝では、第 1 L P C ドメインは L P C 励振信号を表し、第 2 処理分枝では、第 2 L P C ドメインは L P C 重み付き信号を表す。つまり、第 1 L P C ドメイン信号は、フィルタ (  $1 - A ( z )$  ) を介した濾波により L P C 残余ドメインへと変換することで取得され、他方、第 2 L P C ドメイン信号は、フィルタ (  $1 - A ( z / \mu ) / ( 1 - \mu z^{-1} )$  ) を介した濾波により L P C 重み付きドメインへと変換することで取得される。

【 0 0 7 0 】

30

図 2 b は、図 2 a の符号化スキームに対応する復号化スキームを示す。図 2 a のビットストリームマルチプレクサ 8 0 0 により生成されるビットストリームは、ビットストリーム・デマルチプレクサ 9 0 0 へと入力される。例えばモード検出ブロック 6 0 1 を介してビットストリームから導出される情報に依存して、復号器側スイッチ 6 0 0 は、上方の分枝からの信号又は下方の分枝からの信号のどちらかを帯域拡張ブロック 7 0 1 へと送り出す。帯域拡張ブロック 7 0 1 は、ビットストリーム・デマルチプレクサ 9 0 0 からサイド情報を受け取り、このサイド情報とモード判定 6 0 1 の出力とに基づき、スイッチ 6 0 0 から出力された低帯域に基づいた高帯域を再構築する。

【 0 0 7 1 】

ブロック 7 0 1 により生成されたフル帯域信号は、ジョイントステレオ/サラウンド処理ステージ 7 0 2 へと入力され、ここで、2 チャンネル即ちステレオチャンネル、又は複数チャンネル即ち多チャンネルが再構築される。一般的に、ブロック 7 0 2 は、このブロックに入力されたチャンネル数よりも多い数のチャンネルを出力する。アプリケーションに依るが、このブロックからの出力が入力よりも多数のチャンネルを有する限り、ブロック 7 0 2 への入力はステレオモード等のように 2 チャンネルを含んでいても良く、さらに 2 以上のチャンネルを含んでいても良い。

40

【 0 0 7 2 】

これまで、スイッチ 2 0 0 は 2 つの分枝の間で切替を実行し、その結果、1 つの分枝だけが処理すべき信号を受け取り、他の分枝は処理すべき信号を受け取らないと説明してきた。しかし、代替的な実施例においては、このスイッチは例えばオーディオ符号化手段 4 2

50

1 及び励振符号化手段 5 2 2 , 5 2 3 , 5 2 4 の後に配置されても良い。つまり、両方の分枝 4 0 0 と 5 0 0 とが同じ信号を並行処理しても良い。しかし、ビットレートが 2 倍にならないように、これら符号化分枝 4 0 0 , 5 0 0 のうちの 1 つによって出力された信号だけが選択され、出力ビットストリームの中へと書き込まれる。この場合、判定ステージは、ビットストリームへと書き込まれた信号が所定のコスト関数を最小化するように作動してもよい。このコスト関数とは、生成されたビットレート、生成された知覚的歪、又はレートと歪との組合せのいずれかのコスト関数であっても良い。このように、このモード又は図示するモードにおいて、判定ステージは閉ループモードでも作動することができ、その結果、与えられた知覚的歪に対しては最小のビットレートで、又は、与えられたビットレートに対しては最小の知覚的歪を持つような符号化分枝出力だけが、最終的にビットストリームへと書き込まれる。閉ループモードでは、フィードバック入力は、図 1 a の 3 つの量子化器 / スケラブロック 4 2 1 , 5 2 2 , 5 2 4 の出力から導出されても良い。

10

#### 【 0 0 7 3 】

2 つのスイッチを有する実施形態においては、即ち、第 1 スイッチ 2 0 0 と第 2 スイッチ 5 2 1 とを有する実施形態においては、第 1 スイッチのための時間分解能が第 2 スイッチのための時間分解能よりも低いことが好ましい。換言すれば、スイッチ操作を介して切替可能な第 1 スイッチへの入力信号のブロックは、L P C ドメインで作動する第 2 のスイッチによって切り替えられるブロックよりも大きい。例として、周波数ドメイン / L P C ドメイン・スイッチ 2 0 0 は 1 0 2 4 個のサンプルから成る長さのブロックを切り替え、第 2 のスイッチ 5 2 1 は各 2 5 6 個のサンプルを持つブロックを切り替えても良い。

20

#### 【 0 0 7 4 】

図 1 a から図 1 0 b までのいくつかは装置のブロック図として表されているが、これらの図は方法を示す図でもあり、各ブロック機能は各方法ステップに対応する。

#### 【 0 0 7 5 】

図 3 a は第 1 符号化分枝 4 0 0 と第 2 符号化分枝 5 0 0 の出力として符号化されたオーディオ信号を生成するオーディオ符号器を示す。さらに、符号化されたオーディオ信号は、共通前処理ステージからの前処理パラメータや、上述した図に関連して説明したようにスイッチ制御情報等のサイド情報を含むのが好ましい。

#### 【 0 0 7 6 】

第 1 符号化分枝はオーディオ中間信号 1 9 5 を第 1 符号化アルゴリズムに従って符号化し、このとき第 1 符号化アルゴリズムは情報シンクモデルを有するのが好ましい。第 1 符号化分枝 4 0 0 は第 1 符号化手段出力信号を生成し、この信号はオーディオ中間信号 1 9 5 の符号化されたスペクトル情報表示である。

30

#### 【 0 0 7 7 】

さらに、第 2 符号化分枝 5 0 0 はオーディオ中間信号 1 9 5 を第 2 符号化アルゴリズムに従って符号化し、このとき第 2 符号化アルゴリズムは情報ソースモデルを有し、第 2 符号化手段出力信号の中にオーディオ中間信号を表示する情報ソースモデルのための符号化されたパラメータを生成するのが好ましい。

#### 【 0 0 7 8 】

オーディオ符号器は、オーディオ入力信号 9 9 を前処理してオーディオ中間信号 1 9 5 を取得するための共通前処理ステージ 1 0 0 をさらに備える。具体的には、この共通前処理ステージは、オーディオ入力信号 9 9 を処理してオーディオ中間信号 1 9 5 即ちこの共通前処理アルゴリズムの出力が、オーディオ入力信号の圧縮済バージョンとなるように処理する。

40

#### 【 0 0 7 9 】

符号化されたオーディオ信号を生成するための好適なオーディオ符号化方法は、次のステップを備える。オーディオ中間信号 1 9 5 を第 1 符号化アルゴリズムに従って符号化するステップ 4 0 0 であって、第 1 符号化アルゴリズムは情報シンクモデルを有し、第 1 出力信号の中に、オーディオ信号の符号化されたスペクトル情報表示を生成するステップと、オーディオ中間信号 1 9 5 を第 2 符号化アルゴリズムに従って符号化するステップ 5 0 0

50

であって、第2符号化アルゴリズムは情報ソースモデルを有し、第2出力信号の中に、オーディオ中間信号195を表示する情報ソースモデルのための符号化されたパラメータを生成するステップと、オーディオ入力信号99に共通前処理を施してオーディオ中間信号195を取得するステップ100であって、この共通前処理ステップは、オーディオ入力信号99に対し、オーディオ中間信号195がオーディオ入力信号99の圧縮済バージョンとなるように処理するステップと、を備える。符号化されたオーディオ信号は、そのオーディオ信号の所定の部分に第1出力信号又は第2出力信号のいずれかを含む。この方法は、オーディオ中間信号の所定の部分を第1符号化アルゴリズム若しくは第2符号化アルゴリズムのいずれかを使用して符号化するか、又はその信号を両方のアルゴリズムを使用して符号化するステップと、第1符号化アルゴリズムの結果若しくは第2符号化アルゴリズムの結果のいずれかを符号化済信号として出力するステップと、をさらに備えるのが好ましい。

10

#### 【0080】

一般的には、第1符号化分枝400で使用されるオーディオ符号化アルゴリズムは、オーディオシンク内の状態を反映し且つモデル化する。オーディオ情報のシンクとは、通常、人間の耳である。人間の耳は周波数分析器としてモデル化されることが可能である。従って、第1符号化分枝は符号化されたスペクトル情報を出力する。好適には、第1符号化分枝は、聴覚心理的マスキング閾値を追加的に適用するための聴覚心理モデルをさらに備える。聴覚心理的マスキング閾値は、オーディオスペクトル値を量子化する際に使用される。スペクトルオーディオ値を量子化することで量子化ノイズが導入されるが、このノイズが聴覚心理マスキング閾値の下で隠されるように、量子化を実行するのが望ましい。

20

#### 【0081】

第2符号化分枝は情報ソースモデルを表し、オーディオ音響の生成を反映する。情報ソースモデルはスピーチモデルを含んでも良く、このスピーチモデルはLPC分析ステージによって反映される。即ち、時間ドメイン信号をLPCドメインへと変換し、次にLPC残余信号、即ち励振信号を処理する中で反映される。代替的な音響ソースモデルは、所定の楽器又は現実の世界に存在する特定の音響ソース等、他のいかなる音響生成物であっても良い。複数の音響ソースモデルが有効である場合には、異なる音響ソースモデル間の選択が例えばSNR計算に基づいて実行されても良い。即ち、オーディオ信号の所定の時間部分及び/又は周波数部分を符号化するために、どのソースモデルが最も適切であるかという計算に基づいて実行されても良い。しかし、好適には符号化分枝間の切替は時間ドメインで実行される。即ち、中間信号の所定の時間部分は1つのモデルを使用して符号化され、異なる所定の時間部分は他の符号化分枝を使用して符号化される。

30

#### 【0082】

情報ソースモデルは所定のパラメータにより表示される。AMR-WB+のような現代的なスピーチ符号器を考慮した場合、スピーチモデルに関しては、パラメータはLPCパラメータであり、符号化された励振パラメータである。AMR-WB+はACELP符号器とTCX符号器を含む。この場合、符号化された励振パラメータは、グローバルゲイン、ノイズフロア、及び可変長符号であっても良い。

#### 【0083】

図3bは図3aの符号器に対応する復号器を示す。全体として図3bは、符号化されたオーディオ信号を復号化し、復号化されたオーディオ信号799を取得する復号器を示している。この復号器は、情報シンクモデルを有する第1符号化アルゴリズムに従って符号化された符号化済信号を復号化するための第1復号化分枝450を含む。また、この復号器は、情報ソースモデルを有する第2符号化アルゴリズムに従って符号化された符号化済情報信号を復号化するための第2復号化分枝550をさらに含む。また、このオーディオ復号器は、第1復号化分枝450及び第2復号化分枝550からの出力信号を結合し、結合済信号を得るための結合器600をさらに含む。この結合済信号は、図3bでは復号化済オーディオ中間信号699として示され、共通後処理ステージ700へと入力される。この共通後処理ステージにおいて、復号化済オーディオ中間信号699であり結合器600

40

50

から出力された結合済信号である信号は、この後処理ステージからの出力信号が結合済信号の拡張済バージョンになるように後処理される。その結果、復号化済オーディオ信号 799 は、復号化済オーディオ中間信号 699 と比較してより強化された情報内容を持つ。このような情報拡張は、共通後処理ステージにより前/後処理パラメータを用いて実行されるが、これらのパラメータは、符号器から復号器へと伝送されるか、又は復号化済オーディオ中間信号そのものから導出されても良い。しかし、前/後処理パラメータは符号器から復号器へと伝送されることが好ましい。なぜなら、その方が復号化されたオーディオ信号の品質を向上させることができるからである。

【0084】

図3cはオーディオ入力信号195を符号化するオーディオ符号器を示し、この入力信号は本発明の好適な実施例に従う図3aのオーディオ中間信号195と等しくても良い。オーディオ入力信号195は、例えば時間ドメインでも良い第1ドメインにおいて存在するが、この第1ドメインはまた、周波数ドメイン、LPCドメイン、LPCスペクトルドメイン又は他のいかなるドメインでも良い。概して、1つのドメインから他のドメインへの変換は、時間/周波数変換アルゴリズム又は周波数/時間変換アルゴリズムのいずれかのような公知の変換アルゴリズムによって実行される。

【0085】

時間ドメインからの変換の代替的な変換として、例えばLPCドメインへの変換があり、このLPCドメインへの変換は、時間ドメイン信号をLPC濾波した結果であり、LPC残余信号又は励振信号として得られる。場合によっては、濾波された信号を生成する操作であり、変換前の信号サンプル数に実質的な影響を与える操作であれば、他のいかなる濾波操作でも変換アルゴリズムとして使用することができる。従って、LPCベースの重み付けフィルタを使用してオーディオ信号を重み付けする操作も、LPCドメインでの信号を生成する他の一変換方法である。時間/周波数変換では、1つのスペクトル値を修正すると、変換前の全ての時間ドメイン値に対して影響を与えることになる。同様に、いかなる時間ドメインサンプルを1つ修正しても、各周波数ドメインサンプルに対して影響を与えるであろう。同様に、LPCドメイン状態の励振信号のサンプルを修正しても、LPCフィルタの長さのために、LPC濾波前の実質的なサンプル数に対して影響を与えることになる。また同様に、LPC変換の前にサンプルを修正しても、LPCフィルタの生来の記憶効果のために、このLPC変換により取得される多くのサンプルに対して影響を与えるであろう。

【0086】

図3cのオーディオ符号器は、第1符号化済信号を生成する第1符号化分枝400を含む。この第1符号化済信号は、第4ドメイン、即ちこの好適な実施例においては時間スペクトルドメイン(周波数ドメイン)であっても良い。つまり、時間ドメイン信号を時間/周波数変換を介して処理することで得られるドメインであっても良い。

【0087】

このように、オーディオ信号を符号化するための第1符号化分枝400は、第1符号化アルゴリズムを使用して第1符号化済信号を取得する。この第1符号化アルゴリズムは、時間/周波数変換アルゴリズムを含んでも含まなくても良い。

【0088】

オーディオ符号器は、オーディオ信号を符号化するための第2符号化分枝500をさらに含む。この第2符号化分枝500は、第1符号化アルゴリズムとは異なる第2符号化アルゴリズムを使用して、第2符号化済信号を取得する。

【0089】

オーディオ符号器は、第1符号化分枝400と第2符号化分枝500との間で切替を実行する第1スイッチ200をさらに含み、その結果、オーディオ入力信号の一部については、ブロック400の出力である第1符号化済信号、又は第2符号化分枝の出力である第2符号化済信号のいずれかが、符号器の出力信号の中に含まれるようになる。つまり、オーディオ入力信号195の所定の部分について、第4ドメインの第1符号化済信号が符号

10

20

30

40

50

器出力信号の中に含まれている場合には、第2符号化済信号、即ち第2ドメインの第1処理済信号又は第3ドメインの第2処理済信号は符号器出力信号の中に含まれない。これにより、当該符号器の高いビットレート効率が保証される。本発明の実施例においては、図3eに関して後述するように、2つの異なる符号化済信号に含まれるオーディオ信号のいかなる時間部分も、1つのフレームのフレーム長に比べれば小さい。これらの小さな部分は、切替イベントの中で、1つの符号化済信号から他の符号化済信号へのクロスフェードのために有用であり、クロスフェードが無い場合に発生する可能性のあるアーチファクトを低減させるのに役立つ。従って、クロスフェード領域を除けば、各時間ドメインブロックは単一ドメインの符号化済信号によって表示される。

【0090】

図3cに示すように、第2符号化分枝500は、第1ドメインのオーディオ信号即ち信号195を第2ドメインへと変換するための変換器510を備える。第2符号化分枝500は、第2ドメインのオーディオ信号を処理して第1処理済信号を得るための第1処理分枝522をさらに備え、好適には、この第1処理済信号も第2ドメインであって、第1処理分枝522はドメイン変換を実行しない。

【0091】

第2符号化分枝500は、第2ドメインのオーディオ信号を第3ドメインへと変換するための第2処理分枝523、524をさらに備え、この第3ドメインは第1ドメインとも第2ドメインとも異なり、オーディオ信号を第3ドメインで処理し、第2処理分枝523、524の出力において第2処理済信号を取得する。

【0092】

第2符号化分枝は、第1処理分枝522と第2処理分枝523、524との間で切替を実行する第2のスイッチ521をさらに備え、その結果、第2符号化分枝へと入力されたオーディオ信号の一部については、第2ドメインの第1処理済信号或いは第3ドメインの第2処理済信号のいずれかが第2符号化済信号の中に存在することになる。

【0093】

図3dは、図3cの符号器により生成された符号化されたオーディオ信号を復号化するための対応する復号器を示す。一般的に、第1ドメインオーディオ信号の各ブロックは、任意のクロスフェード領域は別として、第2ドメイン信号、第3ドメイン信号、又は第4ドメイン符号化済信号のうちのいずれかにより表示される。このクロスフェード領域は、臨界サンプリングの制限において最高のシステムを得るために、1つのフレームよりも短い方が望ましい。符号化されたオーディオ信号は、第1符号化済信号と、第2ドメインの第2符号化済信号と、第3ドメインの第3符号化済信号とを含み、これら第1符号化済信号、第2符号化済信号、及び第3符号化済信号は全て、復号化されたオーディオ信号の異なる時間部分に関連し、また、復号化されたオーディオ信号に係る第2ドメイン、第3ドメイン及び第1ドメインは互いに異なる。

【0094】

この復号器は、第1符号化アルゴリズムに基づいて復号化する第1復号化分枝を備える。この第1復号化分枝は、図3dでは431、440により示され、好適には周波数/時間変換器を含む。第1符号化済信号は好適には第4ドメインであり、復号化された出力信号のドメインである第1ドメインへと変換される。

【0095】

図3dの復号器は、複数の要素を含む第2復号化分枝をさらに備える。これらの要素は、第2符号化済信号を逆処理する第1逆処理分枝531を含み、このブロック531の出力において第1逆処理済信号を第2ドメインで取得する。この第2復号化分枝は、第3符号化済信号を逆処理する第2逆処理分枝533、534をさらに備え、第2逆処理済信号を第2ドメインで取得する。この第2逆処理分枝は第3ドメインから第2ドメインへの変換を実行する変換器を含む。

【0096】

この第2復号化分枝は第1結合器532をさらに備え、第1逆処理済信号と第2逆処理済

10

20

30

40

50

信号とを結合し、第2ドメインにおいて1つの信号を取得する。この結合済信号は、第1の時間的瞬間には第1逆処理済信号によってのみ影響を受け、その後のある時間的瞬間には第2逆処理済信号によってのみ影響を受ける。

【0097】

第2復号化分枝は、結合済信号を第1ドメインへと変換する変換器540をさらに備える。

【0098】

最後に、図3dに示す復号器は、ブロック431, 440から出力された第1復号化済信号と、変換器540の出力信号とを結合し、復号化された出力信号を第1ドメインで取得する第2結合器600をさらに備える。この復号化された第1ドメインの出力信号は、第1の時間的瞬間には変換器540により出力された信号のみの影響を受け、その後のある時間的瞬間には、ブロック431, 440により出力された第1復号化済信号のみの影響を受ける。

【0099】

図3eはこの状態を符号器側の視点から説明する。図3eの上側部分は、概略図的な表現で、時間ドメインオーディオ信号等の第1ドメインオーディオ信号を示し、時間指標は左から右へと増大し、項目3は図3cの信号195を表示するオーディオサンプルのストリームとして捉えられても良い。図3eは、第1符号化済信号と第1処理済信号と第2処理済信号との間で項目4に示されるように切替を実行することで生成されても良い、フレーム3a, 3b, 3c, 3dを示す。第1符号化済信号と第1処理済信号と第2処理済信号とは全て異なるドメインにあり、異なるドメイン間での切替が復号器側でのアーチファクトをもたらすことがないようにするために、時間ドメイン信号のフレーム3aと3bとはクロスフェード領域として示されるオーバーラップ領域を有し、このようなクロスフェード領域はフレーム3bと3cとの間にも存在する。しかし、このようなクロスフェード領域はフレーム3cと3dとの間には存在しない。つまり、フレーム3dもまた第2処理済信号、即ち第3ドメインにおける信号により表示されており、フレーム3cと3dの間にはドメイン変化は存在しないという意味である。従って、一般的には、ドメイン変化がない場合にはクロスフェード領域を設けず、ドメイン変換がある場合つまり2つのスイッチのいずれかの切替動作がある場合には、クロスフェード領域、即ちオーディオ信号の部分であって2つの連続する符号化済/処理済信号により符号化される部分を設けることが望ましい。好適には、クロスフェードは他のドメインへの変化のために実行される。

【0100】

第1符号化済信号又は第2処理済信号が、例えば50%のオーバーラップを有するMDC T処理により生成された場合の本発明の実施例においては、各時間ドメインサンプルは2つの連続するフレームの中に含まれる。しかし、この点は、MDC Tの特徴により、つまりMDC Tが臨界的にサンプリングされたシステムであるから、オーバーヘッドをもたらすことはない。ここで、臨界的にサンプリングされたとは、スペクトル値の数が時間ドメイン値の数と同じであることを意味する。MDC Tは次のような利点を持つ。即ち、特別なクロスオーバー領域が無い状態でクロスオーバー効果が提供されるので、あるMDC Tブロックから次のMDC Tブロックへのクロスオーバーを、臨界サンプリングの条件に違反するいかなるオーバーヘッドも無い状態で提供できる。

【0101】

好適には、第1符号化分枝の中の第1符号化アルゴリズムは情報シンクモデルに基づき、第2符号化分枝の中の第2符号化アルゴリズムは情報ソース又はSNRモデルに基づく。SNRモデルとは、特定の音響生成メカニズムに関連するものではなく、複数の符号化モードの中から例えば閉ループ判定に基づいて選択可能な1つの符号化モードである。このようにSNRモデルは有効な符号化モデルの1つであるが、音響生成器の物理的な構成とは必ずしも関係がなく、情報シンクモデルとは異なるパラメータ化された符号化モデルの1つであり、閉ループ判定に基づいて、特に様々なモデルからの様々なSNRの結果を比較することで選択可能である。

10

20

30

40

50

## 【 0 1 0 2 】

図 3 c に示すように、制御器 3 0 0 , 5 2 5 が設けられる。この制御器は、図 1 a の判定ステージ 3 0 0 の機能を含んでも良く、さらには図 1 a のスイッチ制御装置 5 2 5 の機能を含んでも良い。一般的には、この制御器は、第 1 スイッチ及び第 2 スイッチを信号適応型の方法で制御する。この制御器は、第 1 スイッチへと入力された信号や、第 1 又は第 2 符号化分枝により出力された信号や、第 1 及び第 2 符号化分枝から符号化及び復号化によって得られた信号を、ある目標関数に対して分析する。代替的に又は追加的に、この制御器は、第 2 スイッチへと入力された信号や、第 1 処理分枝又は第 2 処理分枝から出力された信号や、第 1 処理分枝及び第 2 処理分枝から処理と逆処理とによって得られた信号を、ある目標関数に対して分析する。

10

## 【 0 1 0 3 】

1 つの実施例においては、第 1 符号化分枝又は第 2 符号化分枝は、M D C T 又は M D S T (変形離散サイン変換) アルゴリズム等のエイリアシングを導入する時間 / 周波数変換アルゴリズムを含み、このアルゴリズムはエイリアシング効果をもたらさない単純な F F T とは異なるものである。さらに、1 つ又は両方の分枝は量子化 / エントロピー符号化ブロックを備える。具体的には、第 2 符号化分枝の第 2 処理分枝だけがエイリアシング操作を導入する時間 / 周波数変換器を含み、第 2 符号化分枝の第 1 処理分枝は量子化及び / 又はエントロピー符号化器を含むが、いかなるエイリアシング効果も導入しない。エイリアシングを導入する時間 / 周波数変換器は、好適には、分析ウィンドウ及び M D C T 変換アルゴリズムを適用するためのウィンドウ化手段を備える。さらに具体的には、このウィンドウ化手段は連続するフレームに対してオーバーラップする方法で窓関数を適用し、その結果、ウィンドウ化された信号の 1 つのサンプルが少なくとも 2 つの連続するウィンドウ化済フレームの中に生成される。

20

## 【 0 1 0 4 】

ある実施例においては、第 1 処理分枝は A C E L P の符号化手段を備え、第 2 処理分枝は、スペクトル成分を量子化して量子化済スペクトル成分を得る M D C T スペクトル変換器及び量子化器を備える。この時、各量子化済スペクトル成分はゼロであるか、又は複数の異なる発生可能な量子化指標の中の 1 つの量子化指標により定義される。

## 【 0 1 0 5 】

さらに、第 1 スイッチ 2 0 0 は開ループ方式で、第 2 スイッチは閉ループ方式で作動することが望ましい。

30

## 【 0 1 0 6 】

上述したように、両方の符号化分枝はオーディオ信号をブロック単位の方法で符号化する。この場合、第 1 スイッチ又は第 2 スイッチはブロック単位で切替を実行し、その結果、切替動作は最低でも信号の所定数のサンプルから成る 1 つのブロックの後で実行されることになり、この所定数とは対応するスイッチのための 1 つのフレーム長を形成する数である。このように、第 1 スイッチによる切替の 1 単位は例えば 2 0 4 8 個か 1 0 2 8 個のサンプルから成るブロックであっても良く、第 1 スイッチ 2 0 0 の切替の基になるフレーム長は可変的でも良いが、好適にはこのように長い期間に固定される。

## 【 0 1 0 7 】

これとは対照的に、第 2 スイッチ 5 2 1 のためのブロック長、即ち第 2 スイッチ 5 2 1 が 1 つのモードから他のモードへと切替を実行する時間は、第 1 スイッチのブロック長よりも実質的に短い。好適には、両方のスイッチの各ブロック長は、長いブロック長が短いブロック長の整数倍になるように選択される。好適な実施例においては、第 1 スイッチのブロック長は 2 0 4 8 又は 1 0 2 4 であり、第 2 スイッチのブロック長は 1 0 2 4 であるか、より好適には 5 1 2 であり、さらに好適には 1 2 8 サンプルである。その結果、第 1 スイッチがたった 1 回の切替を実行する間に、第 2 スイッチは最大では 1 6 回の切替を実行できる。しかし、好ましい最大ブロック長比率は、4 : 1 である。

40

## 【 0 1 0 8 】

他の実施例においては、制御器 3 0 0 , 5 2 5 は、第 1 スイッチのためのスピーチ・音楽

50

の識別を、スピーチへの判定が音楽への判定に対して優先されるような方法で実行する。この実施例では、第1スイッチへ入力される1つのフレームの50%未満がスピーチであり、フレームの50%を超える比率が音楽である場合でも、スピーチへの判定が実行される。

**【0109】**

さらに、第1フレームのかなり小さな部分がスピーチである場合でも、具体的には、第1フレームのある部分がスピーチであって、この部分がより小さな第2フレームの長さの50%である場合でも、この制御器はスピーチへと切り替える。従って、好適なスピーチ優先切替判定は、例えば第1スイッチのフレーム長に対応するブロックのたった6%又は12%がスピーチである場合でも、スピーチへの切替を実行する。

10

**【0110】**

このような処理は、ある実施例においては有声音のスピーチコアを有しているような第1処理分枝のビットレート節約能力を十分に活用するために好ましく、また、大きな第1フレームの非スピーチである残りの部分についても全く品質低下を招かないためにも好ましい。第2処理分枝は変換器を含み、従って非スピーチ信号をも有するオーディオ信号にとって有益である。好ましくは、この第2処理分枝は臨界サンプリングされたオーバーラップMDCTを含み、このMDCTは、オーバーラップ及び加算等のような時間ドメインのエイリアシング除去処理によって、小さいウィンドウサイズであっても高効率でエイリアシングのない操作を復号器側において提供する。さらに、好適にはAACのようなMDCT符号化分枝である第1符号化分枝に対しては、大きなブロック長が有益である。なぜなら、非スピーチ信号は通常かなり静的であり、長い変換ウィンドウが高い周波数分解能、つまり高品質を提供し、加えて、聴覚心理的に制御された量子化モジュールによってビットレート効率を提供するからである。このモジュールはまた、第2符号化分枝の第2処理分枝における変換ベースの符号化モードに対しても適用することができる。

20

**【0111】**

図3dの復号器に関しては、伝送された信号が図3eに示すサイド情報4aのような明示的な指標を含むことが好ましい。このサイド情報4aは、図3dには示されていないビットストリームパーサによって抽出され、対応する第1符号化済信号、第1処理済信号、又は第2処理済信号を、図3dに記載の第1復号化分枝、第1逆処理分枝又は第2逆処理分枝のような処理器へと正しく出力する。従って、符号化済信号は、符号化済/処理済信号だけでなく、これらの信号に関連するサイド情報も含む。しかし、他の実施例においては、復号化側のビットストリームパーサが所定の信号の間に読み取ることができるような暗示的な信号化があっても良い。図3eに関して説明すれば、第1処理済信号又は第2処理済信号が第2符号化分枝の出力であり、従って、第2符号化済信号である。

30

**【0112】**

好適には、第1復号化分枝及び/又は第2逆処理分枝は、スペクトルドメインから時間ドメインへ変換するためのMDCT変換を含む。このためオーバーラップ・加算器が設けられ、時間ドメイン・エイリアシング除去機能を実行し、同時にクロスフェード効果を提供し、ブロッキングアーチファクトを防止する。一般的に、第1復号化分枝は第4ドメインで符号化された信号を第1ドメインへと変換し、第2逆処理分枝は第3ドメインから第2ドメインへの変換を実行し、第1結合器の次に接続された変換器は第2ドメインから第1ドメインへの変換を実行する。その結果、結合器600の入力においては第1ドメインの信号だけが存在し、これらの信号が図3dの実施例においては復号化された出力信号を表示する。

40

**【0113】**

図4aと図4bは2つの異なる実施例を示し、その相違点はスイッチ200の位置である。図4aでは、スイッチ200は共通前処理ステージ100の出力と2つの符号化分枝400, 500の入力との間に配置される。図4aの実施例では、オーディオ信号が単一の符号化分枝にだけ入力されることが保証され、共通前処理ステージの出力に接続されない他の符号化分枝は作動せず、よってスイッチオフか又はスリープ状態となる。この実施例

50

は次のような利点がある。即ち、非作動の符号化分枝が電力及び演算資源を消費せず、この点がモバイルアプリケーション、特に電池式電源であることから消費電力には全体的な制限があるアプリケーションに対して有益である。

【0114】

しかし、他方、図4bの実施例は消費電力が問題ではない場合には好適と言える。この実施例では、両方の分枝400と500とが常に作動状態であり、所定の時間部分及び/又は所定の周波数部分のために選択された符号化分枝の出力だけがビットストリームフォーマットへと出力される。このフォーマットはビットストリームマルチプレクサ800として構成されても良い。つまり、図4bの実施例では、両方の符号化分枝が常に作動しており、判定ステージ300により選択された符号化分枝の出力だけが出力ビットストリームの中へ組み込まれ、選択されない他の符号化分枝の出力は放棄される。即ち、出力ビットストリームであり符号化されたオーディオ信号の中へは組み込まれない。

10

【0115】

好適には、第2符号化規則/復号化規則はLPCベースの符号化アルゴリズムである。LPCベースのスピーチ符号化においては、準周期的(quasi-periodic)インパルス状の励振信号分節又は信号部分と、ノイズ状の励振信号分節又は信号部分との間の区別が行われる。これは、図7bのような非常に低いビットレートのLPCボコーダ(2.4kbps)に対して実行される。しかし、中位のレートのCELP符号化手段においては、適応型コードブック及び固定型コードブックからのスケールされたベクトルの足し算によって励振信号が得られる。

20

【0116】

準周期的インパルス状の励振信号分節、即ち特定のピッチを持つ信号分節は、ノイズ状の励振信号分節とは異なるメカニズムを用いて符号化される。準周期的インパルス状の励振信号が有声音のスピーチに関連する一方で、ノイズ状信号は無声音のスピーチに関連する。

【0117】

例示的に図5a~図5dを参照する。ここでは、準周期的インパルス状の信号分節又は信号部分と、ノイズ状の信号分節又は信号部分について例示的に説明する。具体的には、図5aでは時間ドメインで示され、図5bでは周波数ドメインで示される有声音スピーチは、準周期的インパルス状の信号部分の一例として説明し、ノイズ状信号の一例としての無声音スピーチは、図5cと図5dに関連して説明する。スピーチは、一般的に、有声音、無声音又はそれらの混合として分類される。サンプル化された有声音及び無声音の分節についての時間及び周波数ドメインの図表を図5a~図5dに示す。有声音のスピーチは、時間ドメインにおいては準周期的であり、周波数ドメインにおいてはハーモニックに構成されている一方で、無声音スピーチは、ランダム状で広帯域である。有声音スピーチの短時間スペクトルは、その繊細でハーモニックなフォルマント構成により特徴付けられる。この繊細でハーモニックな構成は、スピーチの準周期的な性格の結果であり、振動する声帯に起因すると言っても良い。フォルマント構成(スペクトル包絡)は、ソース及び発声経路の相互作用によるものである。発声経路は、咽頭と口腔から成る。有声音スピーチの短時間スペクトルに「適合」するスペクトル包絡の形は、発声経路の変換特性と音門パルスによるスペクトル傾斜(6dB/オクターブ)とに関連する。スペクトル包絡は、フォルマントと呼ばれるピークの集合体により特徴付けられる。フォルマントは発声経路の反響モードである。平均的な発声経路に対し、5kHz未満の3~5個のフォルマントが存在する。通常は3kHz未満で発生する初めの3つのフォルマントの振幅と位置は、スピーチ合成と知覚との両方において非常に重要である。これより高いフォルマントも、広帯域で無声音のスピーチ表現にとって重要である。スピーチの特性は、以下のような物理的なスピーチ生成システムに関連している。つまり、有声音のスピーチは、声帯を振動させることで生成される準周期的な声門空気パルスにより、発声経路を励振することで生成される。周期的パルスの周波数は、基本周波数又はピッチと呼ばれている。無声音スピーチは、発声経路内の狭窄に対して呼気を強制通過させることで生成される。鼻音は鼻孔経

30

40

50

路を発声経路に音響的に結合させることで生成され、破裂音は発声経路の中にある閉鎖部分の後部に形成された呼気圧力を急激に解放することで生成される。

【0118】

図5cや図5dに示すようなオーディオ信号のノイズ状部分は、いかなるインパルス状の時間ドメイン構造もハーモニックな周波数ドメイン構造も示さず、例えば図5aや図5bに示すような準周期的インパルス状部分とも異なる。しかし、後段で説明するように、ノイズ状部分と準周期的インパルス状部分との間の区別は、励振信号のためのLPCの後で実行されても良い。LPCとは、発声経路をモデル化し、その信号から発声経路の励振を抽出する方法である。

【0119】

さらに、準周期的インパルス状部分とノイズ状部分とは時間的な挙動で発生可能である。即ち、オーディオ信号の時間的なある部分はノイズ状であり、オーディオ信号の時間的な他の部分は準周期的、即ち調性があるという意味である。代替的に又は追加的に、異なる周波数帯域では信号の特性も異なる可能性がある。従って、オーディオ信号がノイズ状か又は調性を有するかという判別もまた、ある所定の周波数帯域又は複数の所定の周波数帯域がノイズ状と判別され、他の周波数帯域が調性を有すると判別されるように、周波数選択的に実行されても良い。この場合には、オーディオ信号のある時間部分が調性要素とノイズ要素とを含んでいても良い。

【0120】

図7aはスピーチ生成システムの線形モデルを示す。このシステムは2段階の励振を想定している。即ち、図7cに示すような有声音スピーチのためのインパルス列と、図7dに示すような無声音スピーチのためのランダムノイズである。発声経路は全極フィルタ70としてモデル化され、このフィルタは声門モデル72により生成された図7c又は図7dに示すようなパルス进行处理する。よって、図7aに示すシステムは図7bの全極フィルタモデルに削減されることができ、このモデルは、ゲインステージ77、前方向経路78、フィードバック経路79、及び加算ステージ80を備える。フィードバック経路79には予測フィルタ81が設けられ、図7bに示す全体のソースモデル合成システムはZドメイン関数を用いて次のように表現できる。

$$S(z) = g / (1 - A(z)) \cdot X(z)$$

ここで、 $g$ はゲインを示し、 $A(z)$ はLP分析により決定された予測フィルタであり、 $X(z)$ は励振信号であり、 $S(z)$ は合成スピーチ出力である。

【0121】

図7cと図7dは、線形ソースシステムモデルを使用した有声音と無声音のスピーチ合成を時間ドメインで図示したものである。このシステムと上記等式における励振パラメータとは未知であり、スピーチサンプルの有限セットから決定される必要がある。 $A(z)$ の係数は入力信号の線形予測とフィルタ係数の量子化とを使用して取得される。 $p$ 次の前方線形予測においては、スピーチシーケンスの現在のサンプルは、 $p$ 個の通過サンプルの線形結合から予測される。予測器の係数は、Levinson-Durbinアルゴリズム等の公知のアルゴリズムか、又は一般的には自動相関法(autocorrelation method)や反射法(reflection method)により決定することができる。

【0122】

図7eは、LPC分析ブロック510のさらに詳細な構成を示す。オーディオ信号は、フィルタ情報 $A(z)$ を決定するフィルタ決定ブロック83へと入力される。この情報は、復号器のために必要な短期間予測情報として出力される。この短期間予測情報は現実の予測フィルタ85にとって必要である。減算器86においては、オーディオ信号の現時点のサンプルが入力され、その現時点のサンプルに対する予測値が減算され、その結果、このサンプルのための予測誤差信号がライン84において生成される。図7cや図7dはこのような予測誤差信号サンプルのシーケンスを非常に概略的に示している。従って、図7a、図7bは一種の調整されたインパルス状信号として見ることもできる。

【0123】

10

20

30

40

50

図 7 e は励振信号を計算する好適な方法を示す。他方、図 7 f は重み付き信号を計算する好適な方法を示し、図 7 e とは対照的に、 $\mu$  が 1 でないときにはフィルタ 8 5 は異なる。

$\mu$  は 1 よりも小さな値が好ましい。さらにブロック 8 7 が存在し、 $\mu$  は 1 よりも小さい数が好ましい。一般的に、図 7 e 及び図 7 f の構成要素は、非特許文献 2 又は非特許文献 1 に従って構成されても良い。

#### 【 0 1 2 4 】

図 7 g は逆処理を示し、この逆処理は、復号器側において図 2 b に示す要素 5 3 7 等に対して適用されても良い。特に、ブロック 8 8 は重み付き信号から非重み付き信号を生成し、ブロック 8 9 はこの非重み付き信号から励振信号を計算する。一般的に、図 7 g の非重み付き信号を除く全ての信号は L P C ドメインであるが、励振信号と重み付き信号とは同じドメイン内の異なる信号である。ブロック 8 9 は励振信号を出力し、その後、この出力信号とブロック 5 3 6 の出力信号とを一緒に使用することができる。その後、共通逆 L P C 変換が図 2 b のブロック 5 4 0 において実行される。

#### 【 0 1 2 5 】

次に、合成による分析 C E L P 符号器を図 6 に関連して説明し、このアルゴリズムに適用される修正について説明する。この C E L P 符号器は非特許文献 3 の中で詳細に記載されている。図 6 に示す C E L P 符号器は、長期間予測要素 6 0 と短期間予測要素 6 2 とを含む。さらに、コードブック 6 4 も使用される。知覚的重み付けフィルタ  $W(z)$  6 6 と、誤差最小化制御器 6 8 とが設けられている。 $s(n)$  は時間ドメイン入力信号である。知覚的に重み付けされた後に、重み付き信号は減算器 6 9 へと入力され、この減算器は、ブロック 6 6 の出力における重み付きの合成信号と、元の重み付き信号  $s_w(n)$  との間の誤差を計算する。一般的に、短期間予測フィルタ係数  $A(z)$  は L P 分析ステージで計算され、その係数は図 7 e に示すように  $A(z)$  で量子化される。長期間予測ゲイン  $g$  とベクトル量子化指標、即ちコードブック参照表とを含む長期間予測情報  $A_L(z)$  は、図 7 e の中で番号 1 0 a として示される L P C 分析ステージの出力において、その予測誤差信号について計算される。L T P (長期間予測) パラメータはピッチ遅延とゲインである。C E L P においては、これは通常、過去の励振信号 (残余ではない) を含む適応型コードブックとして構成されている。適応型コードブックの遅延及びゲインは、平均二乗重み付き誤差を (閉ループピッチ探索により) 最小化することで求められる。

#### 【 0 1 2 6 】

次に、C E L P アルゴリズムは、例えばガウス列のコードブックを使用して、短期間予測及び長期間予測の後で得られた残余信号を符号化する。「代数」を表す「A」を持つ A C E L P アルゴリズムは、特定の代数的に設計されたコードブックを有する。

#### 【 0 1 2 7 】

コードブックはいくつかのベクトルを持ち、各ベクトルの長さはいくつかのサンプルの長さを表す。ゲインファクタ  $g$  はコードベクトルをスケールし、その利得コードは長期間予測合成フィルタと短期間予測合成フィルタとにより濾波される。減算器 6 9 の出力における知覚的な重み付き平均二乗誤差が最小となるように、「最適な」コードベクトルが選択される。C E L P における探索プロセスは、図 6 に示すような合成による分析の最適化によって実行される。

#### 【 0 1 2 8 】

1 つのフレームが無声音スピーチと有声音スピーチの混合である場合、又は音楽の上にスピーチが載っている場合のような特異な場合には、T C X 符号化が L P C ドメインにおける励振信号を符号化するためにより適切であり得る。T C X 符号化は、周波数ドメインにおける重み付き信号を、励振生成を全く想定することなく処理する。ゆえに、T C X 符号化は、C E L P 符号化に比べてより一般的であり、励振の有声音又は無声音ソースモデルに限定されることはない。T C X は、スピーチ状信号のフォルマントをモデル化するための、線形予測的フィルタを使用するソース指向のモデル符号化である。

#### 【 0 1 2 9 】

A M R - W B + のような符号化においては、A M R - W B + 解説書から知られるように、

10

20

30

40

50

様々なTCXモード及びACELPの間の選択が行われる。TCXモードには、異なるモードに対してブロック毎の離散フーリエ変換の長さが異なるという点から様々なものがあり、合成による分析の手法や直接的な「フィードフォワード」モードによって最適なモードが選択可能である。

#### 【0130】

図2aと図2bに関連して説明したように、共通前処理ステージ100は、好適にはジョイント多チャンネル(サラウンド/ジョイントステレオ装置)101を備え、帯域拡張ステージ102をさらに備える。これに対応して、復号器は、帯域拡張ステージ701と、次に接続されたジョイント多チャンネルステージ702を含む。好適には、信号処理方向において、符号器では、ジョイント多チャンネルステージ101が帯域拡張ステージ102の  
10  
前に接続され、復号器側では、帯域拡張ステージ701がジョイント多チャンネルステージ702の前に接続される。しかし、代替的に、共通前処理ステージは、後続の帯域拡張を持たないジョイント多チャンネルステージを含んでも良いし、又は、後続のジョイント多チャンネルステージを持たない帯域拡張ステージを含んでも良い。

#### 【0131】

符号器側のジョイント多チャンネルステージ101aと101bと、復号器側のジョイント多チャンネルステージ702aと702bのための好適な例を、図8に関連して説明する。E個のオリジナル入力チャンネルがダウンミキサ101aへと入力され、その結果、ダウンミキサはK個の伝送チャンネルを生成する。ここで、Kは1以上であり且つE以下の数である。  
20

#### 【0132】

好適には、E個の入力チャンネルは、パラメトリック情報を生成するジョイント多チャンネルパラメータ分析器101bへと入力される。このパラメトリック情報は、好適には、差分符号化の後にハフマン符号化を続けるか、又は代わりに算術符号化を続けるか等の操作により、エントロピー符号化されている。ブロック101bにより出力された符号化済パラメトリック情報は、パラメータ復号器702bへと伝送され、この復号器は図2bに示す装置702の一部であっても良い。パラメータ復号器702bは伝送されたパラメトリック情報を復号化し、復号化されたパラメトリック情報をアップミキサ702aへと出力する。アップミキサ702aは、K個の伝送されたチャンネルを受け取り、L個の出力チャンネルを生成する。ここで、Lの数はK以上であり、且つE以下である。  
30

#### 【0133】

パラメトリック情報は、BCC技術から知られ、或いはMP EGサラウンド標準において公知であり且つ詳細に記述されている、チャンネル間レベル差、チャンネル間時間差、チャンネル間位相差、及び/又はチャンネル間コヒーレンスを含んでも良い。伝送されるチャンネルの数は、超低ビットレートアプリケーションのための単一モノラルチャンネルであっても良く、又は、互換性のあるステレオアプリケーション或いは互換性のあるステレオ信号、即ち2つのチャンネルを含んでも良い。典型的には、E個の入力チャンネルの個数とは5又はそれ以上であっても良い。代替的に、E個の入力チャンネルの個数は、空間オーディオオブジェクト符号化(SAOC)において知られるE個のオーディオオブジェクトの個数であっても良い。  
40

#### 【0134】

ある実施形態においては、ダウンミキサは、オリジナルのE個の入力チャンネルの重み付き又は非重み付きの加算、又はE個の入力オーディオオブジェクトの加算を実行する。入力チャンネルがオーディオオブジェクトの場合には、ジョイント多チャンネルパラメータ分析器101bは、オーディオオブジェクトの間の相関行列等のオーディオオブジェクトパラメータを、好適には各時間部分に対し、さらに好適には各周波数帯域に対して計算する。この目的のために、全体の周波数領域は、少なくとも10個、好適には32個或いは64個の周波数帯域に分割されても良い。

#### 【0135】

図9は、図2aの帯域拡張ステージ102と、これに対応する図2bの帯域拡張ステージ  
50

701との構成の好適な実施例を示す。符号器側では、帯域拡張ブロック102は、低域通過フィルタブロック102bとダウンサンプルブロックと高帯域分析器102aとを備えることが好ましく、ダウンサンプルブロックは、低域通過フィルタの後に続くか、又はQMF帯域の半分だけについて作動する逆QMFの一部である。帯域拡張ブロック102へと入力されたオリジナルオーディオ信号は低域通過濾波され、低帯域信号を生成し、次に符号化分枝及びノ又はスイッチへと入力される。この低域通過フィルタは、3kHz～10kHzの領域でも良いカットオフ周波数を持つ。さらに、帯域拡張ブロック102は、スペクトル包絡パラメータ情報、ノイズフロアパラメータ情報、逆濾波パラメータ情報、高帯域内の所定のハーモニクラインに関連する他のパラメトリック情報、及びMP EG4標準のスペクトル帯域複製に関連する章に詳細に記載されている他の追加的なパラメータ等の帯域拡張パラメータを計算するための、高域分析器をさらに含む。

10

## 【0136】

復号器側においては、帯域拡張ブロック701は、パッチャ701aと調整器701bと結合器701cとを備える。結合器701cは、復号化された低帯域信号と、調整器701bにより出力された再構築済及び調整済の高帯域信号とを結合する。調整器701bへの入力はパッチャにより供給されるが、このパッチャは、スペクトル帯域複製や一般的には帯域拡張により低帯域信号から高帯域信号を導出する。パッチャ701aにより実行されるパッチングは、ハーモニクな方法で実行されるパッチングでも、又は非ハーモニクな方法で実行されるパッチングでも良い。このパッチャ701aにより生成された信号は、次に、伝送されたパラメトリック帯域拡張情報を使用して調整器701bにより調整される。

20

## 【0137】

好適な実施例においては、図8及び図9に示すように、上述のブロックがモード制御入力を備えても良い。このモード制御入力は、判定ステージ300の出力信号から導出される。このような好適な実施例においては、判定ステージの出力、即ちオーディオ信号の所定の時間部分がスピーチか音楽かという判定に対し、対応するブロックの特性を順応させても良い。好適には、このモード制御はこれらブロックの1つ以上の機能に関連しても良いが、しかし、ブロックの全ての機能に対して関連したものではない。例えば、上記判定はパッチャ701aにだけ影響を与えるが、図9の他のブロックに対しては影響を与えなくても良いし、例えば図8のジョイント多チャンネルパラメータ分析器101bにだけ影響を与えるが、図8の他のブロックに対しては影響を与えなくても良い。このような構成は、好適には、共通前処理ステージに柔軟性を持たせることで、高い柔軟性と高い品質と低ビットレートの出力信号とを得ることを目的としている。しかし、他方、共通前処理ステージにおいて両方の種類の信号に対してアルゴリズムを使用することで、効率的な符号化/復号化スキームの実行を可能になる。

30

## 【0138】

図10aと図10bは、判定ステージ300の2つの異なる構成を示す。図10aにおいては、開ループの判定が示される。ここでは、判定ステージ内の信号分析器300aが所定の規則を持ち、入力信号の所定の時間部分又は所定の周波数部分の特性を分析し、この信号部分は第1符号化分枝400で符号化すべきか、又は第2符号化分枝500で符号化すべきかについての判定を行う。この目的のため、信号分析器300aは、共通前処理ステージへのオーディオ入力信号を分析しても良く、共通前処理ステージによる出力信号即ちオーディオ中間信号を分析しても良く、又は図8に示すようにモノラル信号やk個のチャンネルを持つ信号でも良いダウンミキサからの出力信号等の共通前処理ステージ内の中間信号を分析しても良い。出力側では、信号分析器300aは、符号器側のスイッチ200と、復号器側の対応するスイッチ又は結合器600とを制御するための切替判定を生成する。

40

## 【0139】

第2スイッチ521については詳細に説明していないが、図4a、図4bに関連して説明したように、第2スイッチ521も第1スイッチ200と同様の方法で配置されても良い

50

。従って、図3cにおけるスイッチ521は、両方の処理分枝522と523, 524との出力側に代替的に配置してもよく、その場合、両方の分枝が並行して作動し、1つの処理分枝の出力だけが図3cには示されていないビットストリーム形成器を介してビットストリームへと書き込まれる。

【0140】

さらに、第2結合器600は図4cで説明したような特定のクロスフェード機能を持っていても良い。代替的又は追加的に、第1結合器532も同じクロスフェード機能を持っていても良い。さらにまた、両方の結合器が同じクロスフェード機能を持っていても良く、異なるクロスフェード機能を持っていても良く、或いは、クロスフェード機能が全くない、つまり両方の結合器がクロスフェード機能を全く持たないスイッチであっても良い。

10

【0141】

既に上述したが、両方のスイッチは、図10a及び図10bに関連して説明したように、開ループの判定を介しても、又は閉ループの判定を介しても制御することができる。このとき、図3cの制御器300, 525は、両方のスイッチに対し異なる機能を持っていても良いし、同じ機能を持っていても良い。

【0142】

さらに、信号適応型の時間ワーピング機能は、第1符号化分枝又は第1復号化分枝に存在できるだけでなく、符号器側の第2符号化分枝の第2処理分枝にも、さらには復号器側にも存在しても良い。処理済信号に依存して、両方の時間ワーピング機能が同じ時間ワーピング情報を持ち、第1ドメインにおいても第2ドメインにおいても同じ時間ワーピングが適用されても良い。この場合には処理の負担が軽減されるので、いくつかの場合、即ち連続するブロックが類似する時間ワーピングの時間特性を持つ場合に役立つであろう。しかし、他の実施例においては、第1符号化分枝と第2符号化分枝の第2処理分枝とのために、それぞれ独立した時間ワーピング評価器を持つことが望ましい。

20

【0143】

本発明の符号化されたオーディオ信号は、デジタル記憶媒体に記憶されることが可能であり、インターネットのような無線伝送媒体又は有線伝送媒体等、伝送媒体を介して伝送されることも可能である。

【0144】

他の実施例では、図1a又は図2aのスイッチ200が、2つの符号化分枝400, 500の間で切替を行う。さらに他の実施例では、第3符号化分枝, 第4符号化分枝, 又はそれ以上の符号化分枝等、追加的な符号化分枝を持つことができる。復号器側においては、図1b又は図2bのスイッチ600が2つの復号化分枝の間、即ち分枝431, 440と、分枝531, 532, 533, 534, 540との間で切替を実行する。さらに他の実施例では、第3復号化分枝, 第4復号化分枝又はそれ以上の復号化分枝等、追加的な復号化分枝を持つことができる。同様に、そのように追加的な符号化/復号化分枝が有る場合には、他のスイッチ521又は532は、2つ以上の異なる符号化アルゴリズムの間で切替を行っても良い。

30

【0145】

図12aは符号器の構成の好適な一実施例を示し、図12bは対応する復号器の構成の好適な一実施例を示す。対応する参照番号について上述した要素に加え、図12aの実施例は、個別の聴覚心理モジュール1200を備え、さらに、図11aではブロック421で示された、追加的な符号化ツールの好適な構成を備えている。これらの追加的なツールとは、時間ノイズシェーピング(TNS)ツール1201と、中央/サイド符号化ツール(M/S)1202である。さらに、上記要素421/524の追加的機能は、ブロック421/542において、スケーリング, ノイズ充填分析, 量子化, 及びスペクトル値の算術符号化が組合せられた構成として示される。

40

【0146】

図12bに示す対応する復号器の構成も追加的な要素を備えている。それらは、M/S復号化ツール1203とTNS-復号器ツール1204である。さらに、これまでの図には

50

示されていない、低音ポストフィルタ(bass postfilter) 1 2 0 5も備えている。遷移ウインドウ化(transition windowing)ブロック 5 3 2は、図 2 Bでスイッチとして示される要素 5 3 2に対応し、オーバーサンプリングされたクロスフェード又は臨界サンプリングされたクロスフェードのいずれかで良い、ある種のクロスフェードを実行する。後者は、あるMDC T操作として実行され、ここでは二つの時間エイリアシングされた部分がオーバーラップされ且つ加算される。この臨界サンプリングされた遷移処理は適切な場所で使用されることが望ましい。なぜなら、品質において全く損失がない状態で、全体のビットレートを低減できるからである。追加的な遷移ウインドウ化ブロック 6 0 0は図 2 Bの結合器 6 0 0に対応し、ここでもスイッチとして示されていたが、しかし、この要素は明らかに、臨界サンプリング又は非臨界サンプリングされたある種のクロスフェードを実行し、その結果、1つのブロックが第1分枝で処理され他のブロックが第2分枝で処理された場合に、ブロッキングアーチファクト、具体的にはスイッチングアーチファクトの発生を防止する。しかし、両方の分枝における処理が互いに完璧に一致する場合には、このクロスフェード操作はあるハードスイッチへと「降格」できる一方で、クロスフェード操作は両分枝間の「ソフト」スイッチとして理解される。

#### 【 0 1 4 7 】

図 1 2 aと図 1 2 bに示す概念により、スピーチとオーディオコンテンツの任意の混合を有する信号を符号化することが可能になり、この概念は、スピーチ又は一般的なオーディオコンテンツのいずれかの符号化のために特別仕様された最良の符号化技術に匹敵するか又はそれ以上の優れた機能を発揮する。符号器及び復号器の構造を一般的に説明すれば、ステレオ又は多チャンネル処理を実行するMPEGサラウンド(MPEGS)機能ユニットと、入力信号内の高いオーディオ周波数のパラメトリック表示を処理する強化されたスペクトル帯域複製(eSBR)ユニットと、からなる共通の前 後処理が存在する構造と言うことができる。さらに、2つの分枝が存在し、1つの分枝は修正された高効率オーディオ符号化(AAC)ツールの経路からなり、他の分枝は線形予測符号化(LP又はLPCドメイン)ベースの経路から成る。後者の経路は、次に、LPC残余の周波数ドメイン表示又は時間ドメイン表示を有する。AACとLPCとの両方に伝送された全てのスペクトルは、量子化と算術符号化に続いてMDC Tドメインで表示される。時間ドメイン表示は、ACELP励振符号化スキームを使用する。これらの基本的構造は符号器については図 1 2 a、復号器については図 1 2 bで示す。これらの図においてデータは、左から右、上から下への方向で流れる。この復号器の機能は、ビットストリームペイロード内の量子化されたオーディオスペクトル又は時間ドメイン表示の記述を発見し、その量子化された値や他の再構築情報を復号化することである。

#### 【 0 1 4 8 】

スペクトル情報が伝送された場合には、復号器は量子化されたスペクトルを再構築し、その再構築されたスペクトルをビットストリームペイロード内の何らかの活性化したツールを用いて処理することで、入力されたビットストリームペイロードにより記述された通りの、実際の信号スペクトルに到達し、最終的には、周波数ドメインのスペクトルを時間ドメインへと変換する。最初の再構築とそのスペクトル再構築のスケールリングに続き、より効率的な符号化を可能にするために1つ又は複数のスペクトルを修正する、任意のツールが存在する。

#### 【 0 1 4 9 】

時間ドメイン信号表示が伝送された場合には、復号器は量子化された時間信号を再構築し、その再構築された時間信号をビットストリームペイロード内の活性化しているあらゆるツールを用いて処理することで、入力されたビットストリームペイロードにより記述された通りの、実際の時間ドメイン信号に到達する。

#### 【 0 1 5 0 】

信号データを操作する前記任意の各ツールについて、「通過する」か否かの選択が維持され、かつ処理が省略された全ての場合においては、入力におけるスペクトル又は時間サンプルが、修正されずにそのままそのツールを通過する。

10

20

30

40

50

## 【 0 1 5 1 】

前述のビットストリームの信号表示が、時間ドメインから周波数ドメイン表示へと変化し、或いはLPドメインから非LPドメインへと変化し、又はそれらの逆に変化する場合、復号器は、ある適切な遷移オーバーラップ・加算ウィンドウ処理を用いて、1つのドメインから他のドメインへ容易に遷移するであろう。

## 【 0 1 5 2 】

eSBRとMPEG Sの処理は、遷移処理の後に、両方の符号化経路に対して同様に適応される。

## 【 0 1 5 3 】

ビットストリームペイロード・デマルチプレクサツールへの入力は、ビットストリームペイロードである。このデマルチプレクサは、ビットストリームペイロードを各ツールのための部分に分割し、各ツールに対してそのツールに関連したビットストリームペイロード情報を提供する。

## 【 0 1 5 4 】

ビットストリームペイロード・デマルチプレクサツールからの出力は、以下の通りである。

( 1 ) 現時点のフレームにおけるコア符号化方式に依存して、以下のいずれかの情報：

- ・量子化されノイズレスに符号化された次で表現されたスペクトル

- ・スケールファクタ情報

- ・算術的に符号化されたスペクトルライン

- ・又は、次のいずれかにより表現された励振信号と共に、線形予測 ( LP ) パラメータ

：

- ・量子化され算術的に符号化されたスペクトルライン ( 変換符号化済励振、TCX ) 又は、

- ・ACELP符号化された時間ドメイン励振

( 2 ) スペクトルノイズ充填情報 ( 任意 )

( 3 ) M / S 決定情報 ( 任意 )

( 4 ) 時間ノイズ整形 ( TNS ) 情報 ( 任意 )

( 5 ) フィルタバンク制御情報

( 6 ) 時間アンワーピング ( time unwarping ) ( TW ) 制御情報 ( 任意 )

( 7 ) 強化されたスペクトル帯域複製 ( eSBR ) 制御情報

( 8 ) MPEG サラウンド ( MPEG S ) 制御情報

## 【 0 1 5 5 】

スケールファクタ・ノイズレス復号化ツールは、ビットストリームペイロード・デマルチプレクサからの情報を引き出し、その情報を解析し、ハフマン符号化及びDPCM符号化されたスケールファクタを復号化する。

## 【 0 1 5 6 】

スケールファクタ・ノイズレス復号化ツールへの入力は、ノイズレスに符号化されたスペクトルのためのスケールファクタ情報である。

## 【 0 1 5 7 】

スケールファクタ・ノイズレス復号化ツールからの出力は、スケールファクタの復号化された整数表現である。

## 【 0 1 5 8 】

スペクトル・ノイズレス復号化ツールは、ビットストリームペイロード・デマルチプレクサから情報を取り出し、その情報を解析し、算術的に符号化されたデータを復号化し、その量子化されたスペクトルを再構築する。このノイズレス復号化ツールへの入力は、ノイズレスに符号化されたスペクトルである。

## 【 0 1 5 9 】

このノイズレス復号化ツールからの出力は、スペクトルの量子化された値である。

## 【 0 1 6 0 】

10

20

30

40

50

逆量子化器ツールは、スペクトルの量子化された値を受け取り、その整数値をスケールされていない再構築されたスペクトルへと変換する。この量子化器は圧伸量子化器であり、その圧伸ファクタは、選択されたコア符号化モードに依存する。

【 0 1 6 1 】

逆量子化器ツールへの入力は、スペクトルの量子化された値である。

【 0 1 6 2 】

逆量子化器ツールからの出力は、スケールされていない逆量子化済スペクトルである。

【 0 1 6 3 】

ノイズ充填ツールは、復号化されたスペクトル内のスペクトルギャップを充填するために使用される。このギャップは、例えば符号器内においてビット要求に対して強い制限があるなどの理由により、スペクトル値がゼロに量子化されたときに発生する。ノイズ充填ツールの使用は任意である。

10

【 0 1 6 4 】

ノイズ充填ツールへの入力は、以下の通りである：

- ・スケールされていない逆量子化済スペクトル
- ・ノイズ充填パラメータ
- ・スケールファクタの復号化された整数表現。

【 0 1 6 5 】

ノイズ充填ツールへの出力は、以下の通りである：

- ・事前にゼロへと量子化されたスペクトルラインのための、スケールされていない逆量子化済スペクトル値
- ・スケールファクタの修正された整数表示。

20

【 0 1 6 6 】

再スケーリングツールは、スケールファクタの整数表示を実際の値に変換し、かつそのスケールされていない逆量子化済スペクトルを、関連するスケールファクタで乗算する。

【 0 1 6 7 】

上記再スケーリングツールへの入力は、以下の通りである：

- ・スケールファクタの復号化された整数表示
- ・スケールされていない逆量子化済スペクトル。

【 0 1 6 8 】

上記再スケーリングツールからの出力は、スケーリングされた、逆量子化済スペクトルである。

30

【 0 1 6 9 】

M / S ツールに関する説明は、非特許文献 4 を参照されたい。

【 0 1 7 0 】

時間ノイズ整形 ( T N S ) ツールに関する説明も、非特許文献 4 を参照されたい。

【 0 1 7 1 】

フィルタバンク/ブロック切替ツールは、符号化器内で実行された周波数マッピングの逆を適応する。フィルタバンクツールのために逆の変形離散コサイン変換 ( I M D C T ) が使用される。この I M D C T は 1 2 0 , 1 2 8 , 2 4 0 , 2 5 6 , 3 2 0 , 4 8 0 , 5 1 2 , 5 7 6 , 9 6 0 , 1 0 2 4 , 1 1 5 2 のスペクトル係数をサポートするように構成されても良い。

40

【 0 1 7 2 】

フィルタバンクツールへの入力は、

- ・ ( 逆量子化された ) スペクトルと、
- ・フィルタバンク制御情報と、である。

【 0 1 7 3 】

フィルタバンクツールからの出力は、時間ドメインの再構築済オーディオ信号である。

【 0 1 7 4 】

時間ワーブされたフィルタバンク/ブロック切替ツールは、時間ワーピングモードが活性

50

化された時に、通常のフィルタバンク/ブロック切替ツールと置き換わる。この時間ワーブされたフィルタバンクはIMDC Tに関して通常のフィルタバンクと同じであるが、追加的に、ウィンドウ処理された時間ドメインサンプルが、時間変化するリサンプリングにより、ワーブされた時間ドメインから線形時間ドメインへとマッピングされる。

**【0175】**

時間ワーブされたフィルタバンクツールへの入力は、

- ・逆量子化されたスペクトルと、
- ・フィルタバンク制御情報と、
- ・時間ワーピングの制御情報と、である。

**【0176】**

このフィルタバンクツールからの出力は、線形時間ドメインの再構築済オーディオ信号である。

**【0177】**

強化されたSBR (eSBR) ツールは、オーディオ信号の高帯域を再構築する。この方法は、符号化の際に省略されたハーモニックのシーケンスの複製に基づいている。この方法は、生成された高帯域のスペクトル包絡を調整し、逆の濾波を適用し、さらにノイズと正弦波成分を加算することで、オリジナル信号のスペクトル特性を再構築する。

**【0178】**

eSBRツールへの入力は、

- ・量子化された包絡データと、
- ・種々の(Misc.)制御データと、
- ・AACコア復号器からの時間ドメイン信号と、である。

**【0179】**

eSBRツールからの出力は、

- ・時間ドメイン信号、又は
- ・例えばMP EGサラウンドツールが使用された場合には、信号のQMF - ドメイン表示、のどちらかである。

**【0180】**

MP EGサラウンド (MP EG S) ツールは、1つ又は複数の入力信号から多数の信号を生成する。この生成は、適切な空間パラメータの制御の下で、ある洗練されたアップミックスの処理を入力信号に適用することで達成される。USAC (統一的なスピーチ及びオーディオ符号化) の文脈において、MP EGは、伝送されたダウンミックス済信号と一緒にパラメトリックサイド情報をも伝送する手法で、多チャンネル信号の符号化のために使用される。

**【0181】**

上記MP EGツールへの入力は、

- ・ダウンミックス済の時間ドメイン信号、又は
- ・eSBRツールからのダウンミックス済信号のQMF - ドメイン表示、である。

**【0182】**

上記MP EGツールからの出力は、

- ・多チャンネル時間ドメイン信号である。

**【0183】**

信号分類ツールは、オリジナル入力信号を分析し、様々の符号化モードを選択する決め手となる制御情報を、その信号から生成する。入力信号の分析は構成に依存するものであり、与えられた入力信号フレームのために最適なコア符号化モードを選択しようと試みる。この信号分類からの出力は、(任意ではあるが) その他のツールの挙動に影響を与えるためにも使用できる。その他のツールとは、例えばMP EGサラウンド、強化されたSBR、時間ワーブされたフィルタバンクなどである。

**【0184】**

信号分類ツールへの入力は、

10

20

30

40

50

- ・オリジナルの未修正入力信号と、
- ・構成に依存した追加のパラメータと、である。

## 【0185】

信号分類ツールの出力は、

- ・コアコーデック（LP濾波されていない周波数ドメイン符号化、LP濾波済周波数ドメイン又はLP濾波済時間ドメイン符号化）の選択を制御するための制御情報である。

## 【0186】

本発明では、図12aのブロック410の中及び図12aの変換器523の中における時間/周波数分解能は、オーディオ信号に依存して制御されている。図13aは、ウィンドウ長と変換長と時間分解能と周波数分解能との間の相関関係を示す。この図から、長いウィンドウ長の場合には、時間分解能は低くなるが周波数分解能は高くなることが明らかであり、また、短いウィンドウ長の場合には、時間分解能は高くなるが周波数分解能は低くなることが明らかである。

10

## 【0187】

第一符号化分枝、即ち好適には図12Aにおいて要素410, 1201, 1202, 421によって示すAAC符号化分枝においては、種々のウィンドウを使用することができる。ここで、ウィンドウ形状はある信号分析器によって決定され、その信号分析器は、好適には信号分類ブロック300内に含まれても良いが、分離したモジュールであってもよい。符号器は、図13Bに示す種々の時間/周波数分解能を持つウィンドウのうちの1つを選択する。1番目の長いウィンドウ、3番目のウィンドウ、4番目のウィンドウ、5番目のウィンドウ及び6番目のウィンドウの時間/周波数分解能は、変換長1024に対応する2048個のサンプリング値に等しい。図13Bの2番目のラインに示す短いウィンドウは、そのウィンドウサイズに応じた256個のサンプリング値の時間分解能を有する。この分解能は変換長128に対応する。

20

## 【0188】

同様に、最後の二つのウィンドウは、2304に等しいウィンドウ長を持ち、これは上述した1番目のラインのウィンドウよりも高い周波数分解能であるが、より低い時間分解能である。これら最後の2つのウィンドウの変換長は、1152に等しい。

## 【0189】

第一符号化分枝においては、図13b内の変換ウィンドウから生成される様々なウィンドウシーケンスが構築可能である。図13cでは短いシーケンスは1つだけ示されており、他の複数の「シーケンス」は単一のウィンドウだけから構成されているが、複数のウィンドウからなる大きなシーケンスも構築可能である。図13bによれば、小さな数の係数、即ち1024の代わりとなる960のような係数については、時間分解能もまた、1024のような対応する高い係数における時間分解能よりも低くなるという点に留意すべきである。

30

## 【0190】

図14a - 14gは、第2符号化分枝における様々な分解能/ウィンドウサイズを示す。本発明のある好適な実施例では、第2符号化分枝は、あるACELP時間ドメイン符号器526である第一の処理分枝と、フィルタバンク523を含む第2処理分枝とを含む。この分枝では、例えば2048個のサンプルからなる1つのスーパーフレームが、256個のサンプルからなるフレームへとサブ分割される。256個のサンプルからなる個々のフレームは別々に使用可能であり、その結果、50%のオーバーラップを持つMDC Tが適用された場合に、各ウィンドウが2つのフレームをカバーする4つのウィンドウのシーケンスが適用できる。次に、図14dに示すように、ある高い時間分解能が使用される。代替的に、信号が長いウィンドウを許可する場合には、図14cに示すようなシーケンスが適用できる。この場合、各ウィンドウ（中間長のウィンドウ）について1024個のサンプルを持つ、あるダブルウィンドウサイズが適用され、その結果、1つのウィンドウが4つのフレームをカバーし、50%のオーバーラップとなる。

40

## 【0191】

50

最後に、長いウィンドウを使用できる信号である場合には、その長いウィンドウは4096個以上のサンプルに渡って延び、オーバーラップはやはり50%となる。

【0192】

上述のように、2つの分枝が存在し、1つの分枝はあるACELP符号器を持つような好適な実施例においては、図14eに示すような1つのスーパーフレーム内で、「A」で示すACELPフレームの位置により、「T」で示す2つの隣接するTCXフレームに対して適用されるウィンドウサイズが決定されても良い。基本的に、可能な場合には長いウィンドウが使用されるものである。しかし、2つのAフレームの間に1つのTフレームがある場合には、短いウィンドウを適用しなければならない。中間長のウィンドウは、2つのTフレームが隣接する場合に適用できる。しかし、3つのTフレームが隣接する場合には、対応する大きなウィンドウを適用することは、複雑性が増すという理由で効果的ではない。従って、3番目のTフレームは、Aフレームが前に無いにも拘わらず、短いウィンドウにより処理されることができる。スーパーフレーム全体がTフレームだけを持つ場合には、長いウィンドウを適用できる。

10

【0193】

図14fはウィンドウのための幾つかの代替案を示す。ここでは、好適な50%のオーバーラップに起因して、ウィンドウサイズは常に、 $2 \times 1g$  ( $1g$ : スペクトル係数の数) となる。しかし、全ての符号化分枝について他のオーバーラップパーセンテージも適用可能であり、その結果、ウィンドウサイズと変換長との間の関係が2とは異なっても良く、時間ドメイン・エイリアシングが適用されない場合のように、1に近づいても良い。

20

【0194】

図14gは、図14fに記載の規則に基づいてウィンドウを構築するための規則を示す。値Zはウィンドウの開始点におけるゼロの個数を示す。値Lはエイリアシングゾーンにおけるウィンドウ係数の個数を示す。部分Mの値は、Mに対応する部分にゼロを持つような隣接するウィンドウとのオーバーラップにより、エイリアシングを全く導入しない「1」の値である。この部分Mの次には、右側のオーバーラップゾーンRが続き、このRの後には、後続のウィンドウの部分Mに対応しても良い、ゼロのZRゾーンが続く。

【0195】

以下に添付する付録を参照されたい。この付録部分は、本発明のオーディオ符号化/復号化スキームの好適かつ詳細な構成を、特に復号化側について記載するものである。

30

【0196】

[付録]

1. ウィンドウ及びウィンドウシーケンス

量子化及び符号化は周波数ドメインで実行される。この目的のため、時間信号は符号器内で周波数ドメインへとマッピングされる。復号器はサブクローズ(subclause) 2に記載のような逆マッピングを実行する。信号に依るが、符号器は3つの異なるウィンドウサイズ、即ち2304, 2048, 256を使用して時間/周波数分解能を変化させても良い。ウィンドウ間を切り替えるために、遷移ウィンドウ(transition window) LONG\_START\_WINDOW, LONG\_STOP\_WINDOW, START\_WINDOW\_LPD, STOP\_WINDOW\_1152, STOP\_START\_WINDOW 及び STOP\_START\_WINDOW\_1152 が使用される。図13Bはウィンドウを示すものであり、対応する変換長を特定し、各ウィンドウの形状を概略的に示す。3つの変換長、即ち、1152と、(長い変換長と呼ぶ)1024(又は960)と、(短い変換長と呼ぶ)128(120)個の係数を持つ変換長が使用される。

40

【0197】

ウィンドウシーケンスは、raw\_data\_block が常に1024(又は960)個の出力サンプルを表現するデータを含むという方法で、ウィンドウから構成されている。データ要素である window\_sequenceは、実際に使用されているウィンドウシーケンスを示す。図13cは各ウィンドウシーケンスがどのように個別のウィンドウから構成されているかを示す。変換とウィンドウについての詳細な情報は、サブクローズ(subclause) 2に記載する。

50

## 【 0 1 9 8 】

## 1.2 スケールファクタ帯域とグルーピング

非特許文献5を参照されたい。

## 【 0 1 9 9 】

非特許文献5に記載のように、スケールファクタ帯域の幅は人間の聴覚システムの臨界帯域を真似て構築されている。そのため、1つのスペクトルにおけるスケールファクタ帯域の個数とそれらの幅とは、変換長とサンプリング周波数とに依存する。非特許文献6は、変換長1024(960)及び128(120)と、サンプリング周波数と、における各スケールファクタ帯域の開始点へのオフセットを示す。元々はLONG\_WINDOW, LONG\_START\_WINDOW 及びLONG\_STOP\_WINDOW のために設計されたそれらの表は、START\_WINDOW\_LPD 及びSTOP\_START\_WINDOWのために使用される。表4～表10はSTOP\_WINDOW\_1152 及び STOP\_START\_WINDOW\_1152 のためのオフセット表である。

10

## 【 0 2 0 0 】

## 1.3 lpd\_channel\_stream() の復号化

lpd\_channel\_stream()ビットストリーム要素は、「線形予測ドメイン」の符号化済信号からなる1つのフレームを復号化するために必要な全ての情報を含む。この要素は、LPCドメインで符号化された、即ちLPC濾波ステップを含む方法で符号化された符号化済信号からなる1つのフレームのためのペイロードを含む。このフィルタの残余(所謂「励振」)は、次にACELPモジュールを用いて表現されるか、又はMDC T変換ドメイン(「変換符号化済励振」, TCX)で表現される。信号特性への近しい調整を可能にするため、1つのフレームは同一サイズを持つ4つの小さなユニットへと分割され、各ユニットはACELP又はTCXの符号化スキームを用いて符号化される。

20

## 【 0 2 0 1 】

上述のプロセスは、非特許文献1に記載の符号化スキームと類似している。この文献から受け継ぐものであるが、僅かに異なる用語使いとして、次のような用語が挙げられる。即ち、1つの「スーパーフレーム」とは1024個のサンプルからなる1つの信号文節を表し、1つの「フレーム」とはその正確な4分の1、即ち256個のサンプルを持つ。これらの各フレームは、同一の長さを持つ4つの「サブフレーム」へとさらに分割される。本件の説明ではこのような用語使いを採用していることに注意されたい。

## 【 0 2 0 2 】

## 1.4 定義、データ要素

acelp\_core\_mode ... このビットフィールドは、ACELPがlpd符号化モードとして使用された場合の正確なビット割当のスキームを示す。

lpd\_mode ... このビットフィールドモードは(1つのAACフレームに対応する)lpd\_channel\_stream()の1つのスーパーフレーム内の4つの各フレームのための符号化モードを定義する。これらの符号化モードはアレイmod[]内に記憶され、0～3の値をとることができる。lpd\_mode からmod[]へのマッピングは以下の表1から決定できる。

30

表1 - lpd\_channel\_stream()のための符号化モードのマッピング

【表 1】

Table 1 – Mapping of coding modes for lpd\_channel\_stream()

lpd_mode	meaning of bits in bit-field mode					remaining mod[] entries
	bit 4	bit 3	bit 2	bit 1	bit 0	
0..15	0	mod[3]	mod[2]	mod[1]	mod[0]	
16..19	1	0	0	mod[3]	mod[2]	mod[1]=2 mod[0]=2
20..23	1	0	1	mod[1]	mod[0]	mod[3]=2 mod[2]=2
24	1	1	0	0	0	mod[3]=2 mod[2]=2 mod[1]=2 mod[0]=2
25	1	1	0	0	1	mod[3]=3 mod[2]=3 mod[1]=3 mod[0]=3
26..31						reserved

mod[0 . . 3] ... アレイmod[]内の値は各フレームにおける各符号化モードを示す。

表 2 - mod[]により示される符号化モード

【表 2】

Table 2 – Coding modes indicated by mod[]

<i>value of mod[x]</i>	<i>coding mode in frame</i>	<i>bitstream element</i>
<b>0</b>	<b>ACELP</b>	<b>acelp_coding()</b>
<b>1</b>	<b>one frame of TCX</b>	<b>tcx_coding()</b>
<b>2</b>	<b>TCX covering half a superframe</b>	<b>tcx_coding()</b>
<b>3</b>	<b>TCX covering entire superframe</b>	<b>tcx_coding()</b>

acelp\_coding() ... ACELP 励振の1フレームを復号化する全てのデータを含むシンタクス要素である。

tcx\_coding() ... MDCTベースの変換符号化済励振(TCX)の1フレームを復号化する全てのデータを含むシンタクス要素である。

first\_tcx\_flag ... 現時点に処理されるTCXフレームが1つのスーパーフレーム内の1番目が否かを示すフラグである。

lpc\_data() ... 現時点のスーパーフレームを復号化するために必要な全てのLPCフィルタパラメータセットを復号化する全てのデータを含むシンタクス要素である。

first\_lpd\_flag ... 現時点のスーパーフレームが、LPCドメインで符号化されたスーパーフレームの列内の1番目が否かを示すフラグであり、このフラグは次に示す表3に従ってビットストリーム要素 core\_mode (channel\_pair\_element の場合にはcore\_mode0 及び core\_mode1)の履歴からも決定できる。

表 3 - first\_lpd\_flagの定義

【表 3】

Table 3 – Definition of first\_lpd\_flag

core_mode of previous frame (superframe)	core_mode of current frame (superframe)	first_lpd_flag
0	1	1
1	1	0

10

last\_lpd\_mode ... 先行して復号化されたフレームの lpd\_mode を示す。

【 0 2 0 3 】

1.5 復号化のプロセス

lpd\_channel\_streamにおける復号化の手順は以下の通りである。

acelp\_core\_modeを取得する。

lpd\_modeを取得してそこからhelper variable mod[]のコンテンツを決定する。

helper variable mod[]のコンテンツに依存して、acelp\_coding 又は tcx\_coding のデータを取得する。

20

lpc\_dataを取得する。

【 0 2 0 4 】

1.6 ACELP / TCX 符号化モードの組合せ

非特許文献 7 の 5 . 2 . 2 章の記載と同様に、lpd\_channel\_streamペイロードの 1 つのスーパーフレーム内には ACELP 又は TCX の許可された組合せが 26 通り存在する。これら 26 通りのモード組合せの 1 つが、ビットストリーム要素 lpd\_mode の中で信号化されている。1 つのサブフレーム内の各フレームにおける lpd\_mode から実際の符号化モードへのマッピングは、表 1 及び表 2 に示される。

表 4 - 44 . 1 k H z 及び 4 8 k H z における STOP\_START\_1152\_WINDOW 及び STOP\_1152\_WINDOW のためのウインドウ長 2 3 0 4 のスケールファクタ帯域

30

【表 4】

**Table 4 – scalefactor bands for a window length of 2304 for STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 44.1 and 48 kHz**

fs [kHz]	44.1,48
num_swb_long_window	49
swb	swb_offset_long_window
0	0
1	4
2	8
3	12
4	16
5	20
6	24
7	28
8	32
9	36
10	40
11	48
12	56
13	64
14	72
15	80
16	88
17	96
18	108
19	120
20	132
21	144
22	160
23	176
24	196

swb	swb_offset_long_window
25	216
26	240
27	264
28	292
29	320
30	352
31	384
32	416
33	448
34	480
35	512
36	544
37	576
38	608
39	640
40	672
41	704
42	736
43	768
44	800
45	832
46	864
47	896
48	928
	1152

10

20

30

表 5 - 3 2 k H z における STOP\_START\_1152\_WINDOW と STOP\_1152\_WINDOW のためのウィンドウ長 2 3 0 4 のスケールファクタ帯域

【表 5】

**Table 5 – scalefactor bands for a window length of 2304 for  
STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 32 kHz**

fs [kHz]	32
num_swb_long_window	51
swb	swb_offset_long_window
0	0
1	4
2	8
3	12
4	16
5	20
6	24
7	28
8	32
9	36
10	40
11	48
12	56
13	64
14	72
15	80
16	88
17	96
18	108
19	120
20	132
21	144
22	160
23	176
24	196
25	216

swb	swb_offset_long_window
26	240
27	264
28	292
29	320
30	352
31	384
32	416
33	448
34	480
35	512
36	544
37	576
38	608
39	640
40	672
41	704
42	736
43	768
44	800
45	832
46	864
47	896
48	928
49	960
50	992
	1152

10

20

30

表 6 - 8 kHz における STOP\_START\_1152\_WINDOW と STOP\_1152\_WINDOW のためのウィンドウ長 2304 のスケールファクタ帯域

【表 6】

**Table 6 – scalefactor bands for a window length of of 2304 for STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 8 kHz**

fs [kHz]	8	swb	Swb_offset_ long_windo w
num_swb_lo ng_window	40	0	0
swb	swb_offset_ long_windo w	21	288
0	0	22	308
1	12	23	328
2	24	24	348
3	36	25	372
4	48	26	396
5	60	27	420
6	72	28	448
7	84	29	476
8	96	30	508
9	108	31	544
10	120	32	580
11	132	33	620
12	144	34	664
13	156	35	712
14	172	36	764
15	188	37	820
16	204	38	880
17	220	39	944
18	236		1152
19	252		
20	268		

10

20

30

表 7 - 1 1 . 0 2 5 k H z , 1 2 k H z 及び 1 6 k H z における STOP\_START\_1152\_WINDOW と STOP\_1152\_WINDOW のためのウインドウ長 2 3 0 4 のスケールファクタ帯域

【表 7】

**Table 7 – scalefactor bands for a window length of 2304 for  
STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 11.025, 12 and 16  
kHz**

fs [kHz]	11.025, 12, 16		
num_swb_lo ng_window	43		
swb	swb_offset _long_win dow	swb	swb_offset l ong_window
0	0	22	228
1	8	23	244
2	16	24	260
3	24	25	280
4	32	26	300
5	40	27	320
6	48	28	344
7	56	29	368
8	64	30	396
9	72	31	424
10	80	32	456
11	88	33	492
12	100	34	532
13	112	35	572
14	124	36	616
15	136	37	664
16	148	38	716
17	160	39	772
18	172	40	832
19	184	41	896
20	196	42	960
21	212		1152

10

20

30

表 8 - 22.05 kHz 及び 24 kHz における STOP\_START\_1152\_WINDOW と STOP\_1152\_WINDOW のための ウィンドウ長 2304 のスケールファクタ帯域

【表 8】

**Table 8 – scalefactor bands for a window length of 2304 for  
STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 22.05 and 24 kHz**

fs [kHz]	22.05 and 24		
num_swb_lo ng_window	47		
swb	swb_offset _long_win dow	swb	swb_offset_l ong_window
0	0	24	160
1	4	25	172
2	8	26	188
3	12	27	204
4	16	28	220
5	20	29	240
6	24	30	260
7	28	31	284
8	32	32	308
9	36	33	336
10	40	34	364
11	44	35	396
12	52	36	432
13	60	37	468
14	68	38	508
15	76	39	552
16	84	40	600
17	92	41	652
18	100	42	704
19	108	43	768
20	116	44	832
21	124	45	896
22	136	46	960
23	148		1152

10

20

30

表 9 - 6 4 k H z における STOP\_START\_1152\_WINDOW と STOP\_1152\_WINDOW のためのウインドウ長 2 3 0 4 のスケールファクタ帯域

【表 9】

**Table 9 – scalefactor bands for a window length of 2304 for  
STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 64 kHz**

fs [kHz]	64
num_swb_lo ng_ window	47 (46)
swb	swb_offset _long_win dow
0	0
1	4
2	8
3	12
4	16
5	20
6	24
7	28
8	32
9	36
10	40
11	44
12	48
13	52
14	56
15	64
16	72
17	80
18	88
19	100
20	112
21	124
22	140
23	156

swb	swb_offset_l ong_ window
24	172
25	192
26	216
27	240
28	268
29	304
30	344
31	384
32	424
33	464
34	504
35	544
36	584
37	624
38	664
39	704
40	744
41	784
42	824
43	864
44	904
45	944
46	984
	1152

10

20

30

表 10 - 88.2 kHz 及び 96 kHz における STOP\_START\_1152\_WINDOW と STOP\_1152\_WINDOW のためのウインドウ長 2304 のスケールファクタ帯域

【表 10】

**Table 10 – scalefactor bands for a window length of 2304 for STOP\_START\_1152\_WINDOW and STOP\_1152\_WINDOW at 88.2 and 96 kHz**

fs [kHz]	88.2 and 96		
num_swb_long_window	41		
swb	swb_offset_long_window	swb	swb_offset_long_window
0	0	21	120
1	4	22	132
2	8	23	144
3	12	24	156
4	16	25	172
5	20	26	188
6	24	27	212
7	28	28	240
8	32	29	276
9	36	30	320
10	40	31	384
11	44	32	448
12	48	33	512
13	52	34	576
14	56	35	640
15	64	36	704
16	72	37	768
17	80	38	832
18	88	39	896
19	96	40	960
20	108		1152

【0205】

1.7 スケールファクタ帯域表の参照文献

他の全てのスケールファクタ帯域表については、非特許文献 8 を参照されたい。

【0206】

1.8 量子化

符号器における AAC スペクトル係数の量子化のために、不均一の量子化器が使用される。そのため、復号器は、スケールファクタのハフマン復号化（サブクローズ 6.3 を参照）とスペクトルデータのノイズレス復号化（サブクローズ 6.1 を参照）の後で、逆の不均一量子化を実行しなければならない。

【0207】

TCX スペクトル係数の量子化のためには、均一の量子化器が使用される。そのため、復号器において、スペクトルデータのノイズレス復号化の後で、逆の量子化は不要となる。

【0208】

2. フィルタバンクとブロック切替

2.1 ツールの説明

信号の時間/周波数表現は、フィルタバンクモジュールへと伝送されることで、時間ドメインへとマッピングされる。このモジュールは、逆の変形離散コサイン変換（IMDCT）に加え、あるウィンドウとあるオーバーラップ・加算の関数を含む。フィルタバンクの時間/周波数分解能を入力信号の特性へと適合させるために、ブロック切替ツールも採用されている。N はウィンドウ長を表し、ここで、N は window\_sequence（サブクローズ 1.1 を参照）の関数である。各チャネルのために、N/2 個の時間 - 周波数値  $X_{i,k}$  が N 個の時間ドメイン値  $x_{i,n}$  へと IMDCT を介して変換される。ウィンドウ関数を適用した

10

20

30

40

50

後に、各チャネルについて、 $z_{i,n}$  シーケンスの前半が、前のブロックのウインドウ化済シーケンス $z_{(i-1),n}$ の後半に対して加算され、各チャネルのための出力サンプル $out_{i,n}$ が再構築される。

【 0 2 0 9 】

## 2.2 定義

window\_sequence ... 2ビットであり、どのウインドウシーケンス（即ちブロックサイズ）が使用されるかを示す。

window\_shape ... 1ビットであり、どのウインドウ関数が選択されたかを示す。

【 0 2 1 0 】

図 1 3 c は 8 個の window\_sequence（即ち、ONLY\_LONG\_SEQUENCE, LONG\_START\_SEQUENCE, EIGHT\_SHORT\_SEQUENCE, LONG\_STOP\_SEQUENCE, STOP\_START\_SEQUENCE, STOP\_1152\_SEQUENCE, LPD\_START\_SEQUENCE, STOP\_START\_1152\_SEQUENCE）を示す。

10

【 0 2 1 1 】

以下においては、LPD\_SEQUENCEは、所謂線形予測ドメイン符号化（セクション 1.3 参照）の中の、許可された全てのウインドウ/符号化モードの組合せを指している。周波数ドメインで符号化済のフレームを復号化する上で、後続のフレームがLPドメインの符号化モードで符号化されたか否かを知ることだけでも重要であり、この点はあるLPD\_SEQUENCEによって表現される。しかし、そのLPD\_SEQUENCE内の正確な構造については、そのLPドメイン符号化済フレームを復号化するとき考慮される。

【 0 2 1 2 】

20

## 2.3 復号化プロセス

### 2.3.1 IMDCT（逆の変形離散コサイン変換）

IMDCTの分析的な表現を次の式で示す。

【 数 2 1 2 】

$$x_{i,n} = \frac{2}{N} \sum_{k=0}^{\frac{N}{2}-1} \text{spec}[i][k] \cos\left(\frac{2\pi}{N}(n+n_0)\left(k+\frac{1}{2}\right)\right) \quad \text{for } 0 \leq n < N$$

30

ここで、n=サンプル指数

i=ウインドウ指数

k=スペクトル係数の指数

N=window\_sequence値に基づくウインドウ長

$n_0=(N/2+1)/2$ である。

【 0 2 1 3 】

逆変換のための合成ウインドウ長Nは、シンタクス要素window\_sequenceとそのアルゴリズムのコンテキストとの関数である。それは以下のように定義される。

ウインドウ長 2 3 0 4 の場合：

【 数 2 1 3 - 1 】

40

$$N = \begin{cases} 2304, & \text{if STOP\_1152\_SEQUENCE} \\ 2304, & \text{if STOP\_START\_1152\_SEQUENCE} \end{cases}$$

ウインドウ長 2 0 4 8 の場合：

【数 2 1 3 - 2】

$$N = \begin{cases} 2048, & \text{if ONLY\_LONG\_SEQUENCE} \\ 2048, & \text{if LONG\_START\_SEQUENCE} \\ 256, & \text{if EIGHT\_SHORT\_SEQUENCE} \\ 2048, & \text{if LONG\_STOP\_SEQUENCE} \\ 2048, & \text{if STOP\_START\_SEQUENCE} \\ 2048, & \text{if LPD\_START\_SEQUENCE} \end{cases}$$

10

【 0 2 1 4 】

有意なブロック遷移は以下の通りである。

【数 2 1 4】

From ONLY\_LONG\_SEQUENCE to { ONLY\_LONG\_SEQUENCE  
LONG\_START\_SEQUENCE  
LPD\_START\_SEQUENCE

from LONG\_START\_SEQUENCE to { EIGHT\_SHORT\_SEQUENCE  
LONG\_STOP\_SEQUENCE

20

from LONG\_STOP\_SEQUENCE to { ONLY\_LONG\_SEQUENCE  
LONG\_START\_SEQUENCE  
LPD\_START\_SEQUENCE

from EIGHT\_SHORT\_SEQUENCE to { EIGHT\_SHORT\_SEQUENCE  
LONG\_STOP\_SEQUENCE  
STOP\_START\_SEQUENCE

30

from LPD\_SEQUENCE to { LPD\_SEQUENCE  
STOP\_1152\_SEQUENCE  
STOP\_START\_1152\_SEQUENCE

from STOP\_START\_SEQUENCE to { EIGHT\_SHORT\_SEQUENCE  
LONG\_STOP\_SEQUENCE

from LPD\_START\_SEQUENCE to { LPD\_SEQUENCE

40

from STOP\_1152\_SEQUENCE to { ONLY\_LONG\_SEQUENCE  
LONG\_START\_SEQUENCE

from STOP\_START\_1152\_SEQUENCE to { EIGHT\_SHORT\_SEQUENCE  
LONG\_STOP\_SEQUENCE

【 0 2 1 5 】

50

### 2.3.2 ウィンドウ化とブロック切替

window\_sequence と window\_shape の要素に依存して、様々な変換ウィンドウが使用される。以下に示すような半分のウィンドウの組合せにより、可能な全てのwindow\_sequence が提供される。

【0216】

window\_shape == 1 に対しては、ウィンドウ係数は次のようなKaiser - Bessel derived (KBD) ウィンドウにより与えられる。

【数216-1】

$$W_{KBD\_LEFT,N}(n) = \frac{\sum_{p=0}^n [W'(p,\alpha)]}{\sum_{p=0}^{N/2} [W'(p,\alpha)]} \quad \text{for } 0 \leq n < \frac{N}{2} \quad 10$$

$$W_{KBD\_RIGHT,N}(n) = \frac{\sum_{p=0}^{N-n-1} [W'(p,\alpha)]}{\sum_{p=0}^{N/2} [W'(p,\alpha)]} \quad \text{for } \frac{N}{2} \leq n < N$$

20

ここで、W' はKaiser - Bessel のカーネル関数であり、非特許文献9も参考となるが、以下のように定義される。

【数216-2】

$$W'(n,\alpha) = \frac{I_0 \left[ \pi\alpha \sqrt{1.0 - \left( \frac{n - N/4}{N/4} \right)^2} \right]}{I_0[\pi\alpha]} \quad \text{for } 0 \leq n \leq \frac{N}{2}$$

$$I_0[x] = \sum_{k=0}^{\infty} \left[ \frac{\left( \frac{x}{2} \right)^k}{k!} \right]^2 \quad 30$$

$$\alpha = \text{kernel window alpha factor, } \alpha = \begin{cases} 4 & \text{for } N = 2048 \text{ (1920)} \\ 6 & \text{for } N = 256 \text{ (240)} \end{cases}$$

【0217】

他の方法として、window\_shape == 0のとき、サインウィンドウは以下のように使用される。

40

【数217】

$$W_{SIN\_LEFT,N}(n) = \sin\left(\frac{\pi}{N}\left(n + \frac{1}{2}\right)\right) \quad \text{for } 0 \leq n < \frac{N}{2}$$

$$W_{SIN\_RIGHT,N}(n) = \sin\left(\frac{\pi}{N}\left(n + \frac{1}{2}\right)\right) \quad \text{for } \frac{N}{2} \leq n < N$$

【0218】

ウィンドウ長Nは、上記KBD及び上記サインウィンドウについて、2048(1920)

50

)又は256(240)になりうる。STOP\_1152\_SEQUENCE及びSTOP\_START\_1152\_SEQUENCEの場合には、Nはやはり2048又は256になり得るが、ウインドウ勾配は類似するものの、フラットトップ領域はより長くなる。

【0219】

LPD\_START\_SEQUENCEの場合にのみ、ウインドウの右側の部分は64個のサンプルのサインウインドウである。

【0220】

取得可能なウインドウのシーケンスをどのように取得するかについては、このサブクローズの以下に記載するa)~h)の項目にて説明する。

【0221】

あらゆる種類のwindow\_sequencesについて、第一変換ウインドウの左半分のwindow\_shapeは、前のブロックのウインドウ形状によって決定される。以下の式が、この事実を示す：

【数221】

$$W_{LEFT,N}(n) = \begin{cases} W_{KBD\_LEFT,N}(n), & \text{if } window\_shape\_previous\_block == 1 \\ W_{SIN\_LEFT,N}(n), & \text{if } window\_shape\_previous\_block == 0 \end{cases}$$

ここで、window\_shape\_previous\_blockは、前のブロック(i-1)のwindow\_shapeを示す。復号化されるべき第一のraw\_data\_block()について、当該ウインドウの左半分及び右半分のwindow\_shapeは同一である。

【0222】

a) ONLY\_LONG\_SEQUENCE:

window\_sequence == ONLY\_LONG\_SEQUENCEは、2048(1920)の全体ウインドウ長N\_1を持つ、1つのLONG\_WINDOWに等しい。

【0223】

Window\_shape == 1については、ONLY\_LONG\_SEQUENCEのためのウインドウは以下のように与えられる：

【数223】

$$W(n) = \begin{cases} W_{LEFT,N_1}(n), & \text{for } 0 \leq n < N_1/2 \\ W_{KBD\_RIGHT,N_1}(n), & \text{for } N_1/2 \leq n < N_1 \end{cases}$$

【0224】

もしwindow\_shape == 0であれば、ONLY\_LONG\_SEQUENCEのためのウインドウは以下のように記述できる：

【数224】

$$W(n) = \begin{cases} W_{LEFT,N_1}(n), & \text{for } 0 \leq n < N_1/2 \\ W_{SIN\_RIGHT,N_1}(n), & \text{for } N_1/2 \leq n < N_1 \end{cases}$$

【0225】

ウインドウ化の後、時間ドメイン値( $z_{i,n}$ ) は以下のように表現できる。

$$Z_{i,n} = W(n) \cdot X_{i,n}$$

【0226】

b) LONG\_START\_SEQUENCE:

LONG\_START\_SEQUENCEは、ONLY\_LONG\_SEQUENCEからEIGHT\_SHORT\_SEQUENCEへのブロック変換のための、正しいオーバーラップ及び加算を得るために必要である。

10

20

30

40

50

## 【 0 2 2 7 】

ウィンドウ長  $N\_l$  と  $N\_s$  とは 2 0 4 8 ( 1 9 2 0 ) と 2 5 6 ( 2 4 0 ) とにそれぞれ設定されている。

$window\_shape == 1$  の場合には、LONG\_START\_SEQUENCEのためのウィンドウは以下の式で与えられる：

## 【 数 2 2 7 】

$$W(n) = \begin{cases} W_{LEFT,N_l}(n), & \text{for } 0 \leq n < N_l/2 \\ 1.0, & \text{for } N_l/2 \leq n < \frac{3N_l - N_s}{4} \\ W_{KBD\_RIGHT,N_s}(n + \frac{N_s}{2} - \frac{3N_l - N_s}{4}), & \text{for } \frac{3N_l - N_s}{4} \leq n < \frac{3N_l + N_s}{4} \\ 0.0, & \text{for } \frac{3N_l + N_s}{4} \leq n < N_l \end{cases} \quad 10$$

## 【 0 2 2 8 】

$window\_shape == 0$  の場合には、LONG\_START\_SEQUENCEのためのウィンドウは以下のようになる：

## 【 数 2 2 8 】

$$W(n) = \begin{cases} W_{LEFT,N_l}(n), & \text{for } 0 \leq n < N_l/2 \\ 1.0, & \text{for } N_l/2 \leq n < \frac{3N_l - N_s}{4} \\ W_{SIN\_RIGHT,N_s}(n + \frac{N_s}{2} - \frac{3N_l - N_s}{4}), & \text{for } \frac{3N_l - N_s}{4} \leq n < \frac{3N_l + N_s}{4} \\ 0.0, & \text{for } \frac{3N_l + N_s}{4} \leq n < N_l \end{cases} \quad 20$$

## 【 0 2 2 9 】

ウィンドウ化された時間ドメイン値は、上記項目 a ) で説明した式を使って計算しても良い。

## 【 0 2 3 0 】

c ) EIGHT\_SHORT

$window\_sequence == EIGHT\_SHORT$  は、それぞれが 2 5 6 ( 2 4 0 ) の長さ  $N\_s$  を持つ、8 個のオーバーラップ及び加算されたSHORT\_WINDOWSを含む。先頭の 0 と後続の 0 とを含む  $window\_sequence$  の全体の長さは 2 0 4 8 ( 1 9 2 0 ) である。8 個の短いブロックそれぞれは、最初は別々にウィンドウ化される。短いブロックの番号は、変数  $j = 0, \dots, M - 1$  ( $M = N\_l / N\_s$ ) により指標化される。

## 【 0 2 3 1 】

前のブロックの  $window\_shape$  は、8 個の短いブロック ( $W_0(n)$ ) の 1 番目にのみ影響する。 $window\_shape == 1$  の場合には、ウィンドウ関数は以下のように与えられても良い：

## 【 数 2 3 1 】

$$W_0(n) = \begin{cases} W_{LEFT,N_s}(n), & \text{for } 0 \leq n < N_s/2 \\ W_{KBD\_RIGHT,N_s}(n), & \text{for } N_s/2 \leq n < N_s \end{cases}$$

$$W_{1-(M-1)}(n) = \begin{cases} W_{KBD\_LEFT,N_s}(n) & \text{for } 0 \leq n < N_s/2 \\ W_{KBD\_RIGHT,N_s}(n), & \text{for } N_s/2 \leq n < N_s \end{cases}$$

## 【 0 2 3 2 】

他に、 $window\_shape == 0$  の場合には、ウィンドウ関数は以下のように記述できる：

30

40

50

【数 2 3 2】

$$w_0(n) = \begin{cases} W_{LEFT,N_s}(n), & \text{for } 0 \leq n < N_s/2 \\ W_{SIN\_RIGHT,N_s}(n), & \text{for } N_s/2 \leq n < N_s \end{cases}$$

$$w_{1-(M-1)}(n) = \begin{cases} W_{SIN\_LEFT,N_s}(n), & \text{for } 0 \leq n < N_s/2 \\ W_{SIN\_RIGHT,N_s}(n), & \text{for } N_s/2 \leq n < N_s \end{cases}$$

10

【0 2 3 3】

EIGHT\_SHORT window\_sequenceの間のオーバーラップ及び加算は、ウインドウ化された時間ドメイン値 $z_{i,n}$ という結果となるが、以下のように記述できる：

【数 2 3 3】

$$z_{i,n} = \begin{cases} 0, & \text{for } 0 \leq n < \frac{N_l - N_s}{4} \\ x_{0,n-\frac{N_l-N_s}{4}} \cdot W_0\left(n - \frac{N_l-N_s}{4}\right), & \text{for } \frac{N_l-N_s}{4} \leq n < \frac{N_l+N_s}{4} \\ x_{j-1,n-\frac{N_l+(2j-3)N_s}{4}} \cdot W_{j-1}\left(n - \frac{N_l+(2j-3)N_s}{4}\right) + x_{j,n-\frac{N_l+(2j-1)N_s}{4}} \cdot W_j\left(n - \frac{N_l+(2j-1)N_s}{4}\right), & \text{for } 1 \leq j < M, \frac{N_l+(2j-1)N_s}{4} \leq n < \frac{N_l+(2j+1)N_s}{4} \\ x_{M-1,n-\frac{N_l+(2M-3)N_s}{4}} \cdot W_{M-1}\left(n - \frac{N_l+(2M-3)N_s}{4}\right), & \text{for } \frac{N_l+(2M-1)N_s}{4} \leq n < \frac{N_l+(2M+1)N_s}{4} \\ 0, & \text{for } \frac{N_l+(2M+1)N_s}{4} \leq n < N_l \end{cases}$$

20

【0 2 3 4】

d) LONG\_STOP\_SEQUENCE

このwindow\_sequenceはEIGHT\_SHORT\_SEQUENCEからONLY\_LONG\_SEQUENCEへの戻り切替を実行するために必要である。

【0 2 3 5】

window\_shape == 1の場合には、LONG\_STOP\_SEQUENCEのためのウインドウは以下のように与えられる：

【数 2 3 5】

$$w(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N_l - N_s}{4} \\ W_{LEFT,N_s}\left(n - \frac{N_l - N_s}{4}\right), & \text{for } \frac{N_l - N_s}{4} \leq n < \frac{N_l + N_s}{4} \\ 1.0, & \text{for } \frac{N_l + N_s}{4} \leq n < N_l/2 \\ W_{KBD\_RIGHT,N_l}(n), & \text{for } N_l/2 \leq n < N_l \end{cases}$$

40

【0 2 3 6】

window\_shape == 0の場合には、LONG\_START\_SEQUENCEのためのウインドウは以下により決定される：

【数 2 3 6】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l - N\_s}{4} \\ W_{LEFT, N\_s}(n - \frac{N\_l - N\_s}{4}), & \text{for } \frac{N\_l - N\_s}{4} \leq n < \frac{N\_l + N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + N\_s}{4} \leq n < N\_l / 2 \\ W_{SIN\_RIGHT, N\_l}(n), & \text{for } N\_l / 2 \leq n < N\_l \end{cases}$$

【0 2 3 7】

ウィンドウ化された時間ドメイン値は、項目 a ) で説明した式によって計算しても良い。

【0 2 3 8】

e ) STOP\_START\_SEQUENCE :

STOP\_START\_SEQUENCEは、ONLY\_LONG\_SEQUENCEだけが必要な場合に、EIGHT\_SHORT\_SEQUENCEからEIGHT\_SHORT\_SEQUENCEへのブロック変換のための正しいオーバーラップ及び加算を得るために必要となる。

【0 2 3 9】

ウィンドウ長  $N\_l$  及び  $N\_s$  は、それぞれ 2 0 4 8 ( 1 9 2 0 ) 及び 2 5 6 ( 2 4 0 ) に設定される。

window\_shape == 1 の場合には、STOP\_START\_SEQUENCEのためのウィンドウは以下のように与えられる：

【数 2 3 9】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l - N\_s}{4} \\ W_{LEFT, N\_s}(n - \frac{N\_l - N\_s}{4}), & \text{for } \frac{N\_l - N\_s}{4} \leq n < \frac{N\_l + N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + N\_s}{4} \leq n < \frac{3N\_l - N\_s}{4} \\ W_{KBD\_RIGHT, N\_s}(n + \frac{N\_s}{2} - \frac{3N\_l - N\_s}{4}), & \text{for } \frac{3N\_l - N\_s}{4} \leq n < \frac{3N\_l + N\_s}{4} \\ 0.0, & \text{for } \frac{3N\_l + N\_s}{4} \leq n < N\_l \end{cases}$$

【0 2 4 0】

window\_shape == 0 の場合には、STOP\_START\_SEQUENCEのためのウィンドウは以下のようにになる：

10

20

30

【数 2 4 0】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l - N\_s}{4} \\ W_{LEFT,N\_s}(n - \frac{N\_l - N\_s}{4}), & \text{for } \frac{N\_l - N\_s}{4} \leq n < \frac{N\_l + N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + N\_s}{4} \leq n < \frac{3N\_l - N\_s}{4} \\ W_{SIN\_RIGHT,N\_s}(n + \frac{N\_s}{2} - \frac{3N\_l - N\_s}{4}), & \text{for } \frac{3N\_l - N\_s}{4} \leq n < \frac{3N\_l + N\_s}{4} \\ 0.0, & \text{for } \frac{3N\_l + N\_s}{4} \leq n < N\_l \end{cases} \quad 10$$

【0 2 4 1】

ウィンドウ化された時間ドメイン値は、a) で説明した式によって計算しても良い。

【0 2 4 2】

f) LPD\_START\_SEQUENCE:

LPD\_START\_SEQUENCEは、ONLY\_LONG\_SEQUENCEからLPD\_SEQUENCEへのブロック変換のための正しいオーバーラップ及び加算を得るために必要となる。

【0 2 4 3】

ウィンドウ長  $N\_l$  及び  $N\_s$  は、それぞれ 2048 (1920) 及び 256 (240) に設定される。

【0 2 4 4】

window\_shape == 1 の場合には、LPD\_START\_SEQUENCEのためのウィンドウは、以下のように与えられる：

【数 2 4 4】

$$W(n) = \begin{cases} W_{LEFT,N\_l}(n), & \text{for } 0 \leq n < \frac{N\_l}{2} \\ 1.0, & \text{for } \frac{N\_l}{2} \leq n < \frac{3N\_l - N\_s}{4} \\ W_{KBD\_RIGHT,N\_s/2}(n + \frac{N\_s}{4} - \frac{3N\_l - N\_s}{4}), & \text{for } \frac{3N\_l - N\_s}{4} \leq n < \frac{3N\_l}{4} \\ 0.0, & \text{for } \frac{3N\_l}{4} \leq n < N\_l \end{cases} \quad 30$$

【0 2 4 5】

window\_shape == 0 の場合には、LPD\_START\_SEQUENCEのためのウィンドウは次のようになる：

【数 2 4 5】

$$W(n) = \begin{cases} W_{LEFT,N\_l}(n), & \text{for } 0 \leq n < \frac{N\_l}{2} \\ 1.0, & \text{for } \frac{N\_l}{2} \leq n < \frac{3N\_l - N\_s}{4} \\ W_{SIN\_RIGHT,N\_s/2}(n + \frac{N\_s}{4} - \frac{3N\_l - N\_s}{4}), & \text{for } \frac{3N\_l - N\_s}{4} \leq n < \frac{3N\_l}{4} \\ 0.0, & \text{for } \frac{3N\_l}{4} \leq n < N\_l \end{cases} \quad 40$$

## 【 0 2 4 6 】

ウィンドウ化された時間ドメイン値は、項目 a ) で上述した式を用いて計算されても良い。

## 【 0 2 4 7 】

g ) STOP\_1152\_SEQUENCE :

STOP\_1152\_SEQUENCEは、LPD\_SEQUENCEからONLY\_LONG\_SEQUENCEへのブロック変換のための、正しいオーバーラップ及び加算を得るために必要となる。

## 【 0 2 4 8 】

ウィンドウ長  $N\_l$  及び  $N\_s$  は、それぞれ 2 0 4 8 ( 1 9 2 0 ) 及び 2 5 6 ( 2 4 0 ) に設定される。

window\_shape == 1 の場合には、STOP\_1152\_SEQUENCEのためのウィンドウは以下のように与えられる：

## 【 数 2 4 8 】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l}{4} \\ W_{LEFT,N\_s}(n - \frac{N\_l}{4}), & \text{for } \frac{N\_l}{4} \leq n < \frac{N\_l + 2N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + 2N\_s}{4} \leq n < \frac{2N\_l + 3N\_s}{4} \\ W_{KBD\_RIGHT,N\_l}(n + \frac{N\_l}{2} - \frac{2N\_l + 3N\_s}{4}), & \text{for } \frac{2N\_l + 3N\_s}{4} \leq n < N\_l + \frac{3N\_s}{4} \\ 0.0, & \text{for } N\_l + \frac{3N\_s}{4} \leq n < N\_l + N\_s \end{cases}$$

## 【 0 2 4 9 】

window\_shape == 0 の場合には、STOP\_1152\_SEQUENCEのためのウィンドウは以下のように見える：

## 【 数 2 4 9 】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l}{4} \\ W_{LEFT,N\_s}(n - \frac{N\_l}{4}), & \text{for } \frac{N\_l}{4} \leq n < \frac{N\_l + 2N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + 2N\_s}{4} \leq n < \frac{2N\_l + 3N\_s}{4} \\ W_{SIN\_RIGHT,N\_l}(n + \frac{N\_l}{2} - \frac{2N\_l + 3N\_s}{4}), & \text{for } \frac{2N\_l + 3N\_s}{4} \leq n < N\_l + \frac{3N\_s}{4} \\ 0.0, & \text{for } N\_l + \frac{3N\_s}{4} \leq n < N\_l + N\_s \end{cases}$$

## 【 0 2 5 0 】

ウィンドウ化された時間ドメイン値は、項目 a ) に上述した式を用いて計算されても良い。

## 【 0 2 5 1 】

h ) STOP\_START\_1152\_SEQUENCE :

STOP\_START\_1152\_SEQUENCEは、ONLY\_LONG\_SEQUENCEだけがが必要な場合に、LPD\_SEQUENCEからEIGHT\_SHORT\_SEQUENCEへのブロック変換のために、正しいオーバーラップと加算を得るために必要となる。

## 【 0 2 5 2 】

10

20

30

40

50

ウィンドウ長  $N\_l$  及び  $N\_s$  は、それぞれ 2048 (1920) 及び 256 (240) に設定される。

【0253】

window\_shape == 1 の場合には、STOP\_START\_SEQUENCE のためのウィンドウは以下のように与えられる：

【数253】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l}{4} \\ W_{LEFT,N\_s}(n - \frac{N\_l}{4}), & \text{for } \frac{N\_l}{4} \leq n < \frac{N\_l + 2N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + 2N\_s}{4} \leq n < \frac{3N\_l}{4} + \frac{N\_s}{2} \\ W_{KBD\_RIGHT,N\_s}(n + \frac{N\_s}{2} - \frac{3N\_l}{4} + \frac{N\_s}{2}), & \text{for } \frac{3N\_l}{4} + \frac{N\_s}{2} \leq n < \frac{3N\_l}{4} + N\_s \\ 0.0, & \text{for } \frac{3N\_l}{4} + N\_s \leq n < N\_l + N\_s \end{cases} \quad 10$$

20

【0254】

window\_shape == 0 の場合には、STOP\_START\_SEQUENCE のためのウィンドウは以下のようになる：

【数254】

$$W(n) = \begin{cases} 0.0, & \text{for } 0 \leq n < \frac{N\_l}{4} \\ W_{LEFT,N\_s}(n - \frac{N\_l}{4}), & \text{for } \frac{N\_l}{4} \leq n < \frac{N\_l + 2N\_s}{4} \\ 1.0, & \text{for } \frac{N\_l + 2N\_s}{4} \leq n < \frac{3N\_l}{4} + \frac{N\_s}{2} \\ W_{SIN\_RIGHT,N\_s}(n + \frac{N\_s}{2} - \frac{3N\_l}{4} + \frac{N\_s}{2}), & \text{for } \frac{3N\_l}{4} + \frac{N\_s}{2} \leq n < \frac{3N\_l}{4} + N\_s \\ 0.0, & \text{for } \frac{3N\_l}{4} + N\_s \leq n < N\_l + N\_s \end{cases} \quad 30$$

【0255】

ウィンドウ化された時間ドメイン値は、項目 a) に上述した式を用いて計算されても良い。

40

【0256】

2.3.3. 前のウィンドウシーケンスとのオーバーラップ及び加算

EIGHT\_SHORT window\_sequence 中のオーバーラップ及び加算とは別に、各 window\_sequence の第1 (左側) 部分は、前の window\_sequence の第2 (右側) 部分にオーバーラップ及び加算され、その結果として最終時間ドメイン値  $out_{i,n}$  をもたらす。この操作を数学的に表現すると、以下ようになる。

ONLY\_LONG\_SEQUENCE, LONG\_START\_SEQUENCE, EIGHT\_SHORT\_SEQUENCE, LONG\_STOP\_SEQUENCE, STOP\_START\_SEQUENCE, LPD\_START\_SEQUENCE の場合には、以下ようになる：

【数 2 5 6】

$$out_{i,n} = z_{i,n} + z_{i-1, n + \frac{N}{2}}; \quad \text{for } 0 \leq n < \frac{N}{2}, \quad N = 2048 (1920)$$

【 0 2 5 7】

STOP\_1152\_SEQUENCE、STOP\_START\_1152\_SEQUENCEの場合には以下ようになる：

【数 2 5 7】

$$out_{i,n} = z_{i,n} + z_{i-1, n + \frac{N_l}{2} + \frac{3N_s}{4}}; \quad \text{for } 0 \leq n < \frac{N_l}{2}, \quad N_l = 2048, N_s = 256$$

10

【 0 2 5 8】

LPD\_START\_SEQUENCEの場合には、次のシーケンスはLPD\_SEQUENCEである。良好なオーバーラップ及び加算を得るために、あるS I N又はK B DウィンドウがLPD\_SEQUENCEの左側部分に適用される。

【数 2 5 8】

$$W_{SIN\_LEFT, N}(n) = \sin\left(\frac{\pi}{N}\left(n + \frac{1}{2}\right)\right) \quad \text{for } 0 \leq n < \frac{N}{2}$$

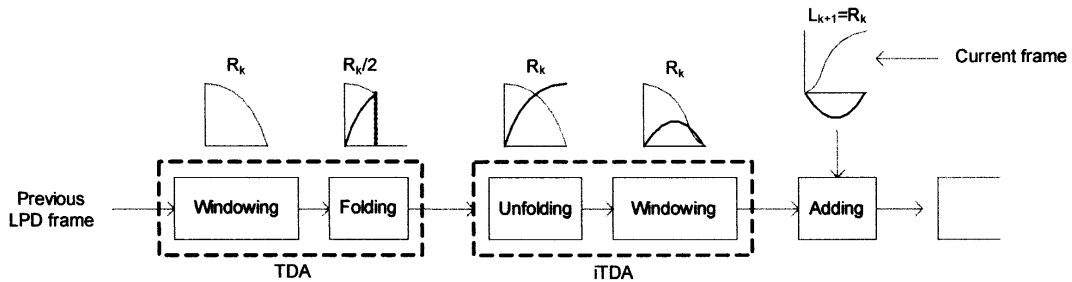
20

ここで、N=128である。

【 0 2 5 9】

STOP\_1152\_SEQUENCEとSTOP\_START\_1152\_SEQUENCEの場合には、前のシーケンスはLPD\_SEQUENCEである。良好なオーバーラップ及び加算を得るために、あるT D A CウィンドウがLPD\_SEQUENCEの左側部分に適用される。

【数 2 5 9】



30

【 0 2 6 0】

3 . I M D T  
サブクローズ 2 . 3 . 1 を参照。

【 0 2 6 1】

3 . 1 ウィンドウ化及びブロック切り替え  
window\_shapeの要素に依存して、オーバーサンプルリングされた様々な変換ウィンドウプロトタイプが使用され、オーバーサンプルリングされたウィンドウの長さは以下のようになる。

40

【数 2 6 1】

$$N_{OS} = 2 \cdot n\_long \cdot os\_factor\_win$$

【 0 2 6 2】

window\_shape == 1の場合には、ウィンドウ係数は、Kaiser - Bessel derived ( K B D )

50

ウィンドウにより以下のように与えられる：

【数 2 6 2 - 1】

$$W_{KBD}\left(n - \frac{N_{OS}}{2}\right) = \sqrt{\frac{\sum_{p=0}^{N_{OS}-n-1} [W(p, \alpha)]}{\sum_{p=0}^{N_{OS}/2} [W(p, \alpha)]}} \quad \text{for } \frac{N_{OS}}{2} \leq n < N_{OS}$$

10

ここで、Kaiser - Besselカーネルウィンドウ関数である  $W'$  は以下のように定義されるが、この点に関しては非特許文献 9 も併せて参照されたい。

【数 2 6 2 - 2】

$$W'(n, \alpha) = \frac{I_0\left[\pi\alpha\sqrt{1.0 - \left(\frac{n - N_{OS}/4}{N_{OS}/4}\right)^2}\right]}{I_0[\pi\alpha]} \quad \text{for } 0 \leq n \leq \frac{N_{OS}}{2}$$

$$I_0[x] = \sum_{k=0}^{\infty} \left[ \frac{\left(\frac{x}{2}\right)^k}{k!} \right]^2$$

20

$\alpha$  = kernel window alpha factor,  $\alpha = 4$

【 0 2 6 3 】

他の例として、`window_shape == 0` の場合には、あるサインウィンドウが以下のように使用される：

30

【数 2 6 3】

$$W_{SIN}\left(n - \frac{N_{OS}}{2}\right) = \sin\left(\frac{\pi}{N_{OS}}\left(n + \frac{1}{2}\right)\right) \quad \text{for } \frac{N_{OS}}{2} \leq n < N_{OS}$$

【 0 2 6 4 】

`Window_sequence` のすべての種類において、左側のウィンドウ部分のために使用されるプロトタイプは、前のブロックのウィンドウ形状によって決定される。以下の式はこの事実を表す：

【数 2 6 4】

40

$$\text{left\_window\_shape}[n] = \begin{cases} W_{KBD}[n] & \text{if } \text{window\_shape\_previous\_block} == 1 \\ W_{SIN}[n] & \text{if } \text{window\_shape\_previous\_block} == 0 \end{cases}$$

【 0 2 6 5 】

同様に、右側のウィンドウ形状のためのプロトタイプは、以下の式によって決定される：

【数 2 6 5】

$$right\_window\_shape[n] = \begin{cases} W_{KBD}[n], & \text{if } window\_shape == 1 \\ W_{SIN}[n], & \text{if } window\_shape == 0 \end{cases}$$

【0 2 6 6】

変換長は既に決定済であるため、EIGHT\_SHORT\_SEQUENCEとその他すべてとを区別するだけで良い。

【0 2 6 7】

10

a) EIGHT SHORT SEQUENCE:

以下に示すc-code状の部分は、あるEIGHT\_SHORT\_SEQUENCEのウィンドウ化及び内部のオーバーラップ - 加算を記述する:

```
tw_windowing_short(X[[]], z, first_pos, last_pos, warped_trans_len_left, warped_trans_len_right, left_window_shape[], right_window_shape[]){
```

```
    offset = n_long - 4*n_short - n_short/2;
```

```
    tr_scale_l = 0.5*n_long/warped_trans_len_left*os_factor_win;
```

```
    tr_pos_l = warped_trans_len_left+(first_pos-n_long/2)+0.5)*tr_scale_l;
```

20

```
    tr_scale_r = 8*os_factor_win;
```

```
    tr_pos_r = tr_scale_r/2;
```

```
    for ( i = 0 ; i < n_short ; i++ ) {
```

```
        z[i] = X[0][i];
```

```
    }
```

```
    for(i=0;i<first_pos;i++)
```

```
        z[i] = 0.;
```

30

```
    for(i=n_long-1-first_pos;i>=first_pos;i--) {
```

```
        z[i] *= left_window_shape[floor(tr_pos_l)];
```

```
        tr_pos_l += tr_scale_l;
```

```
    }
```

```
    for(i=0;i<n_short;i++) {
```

```
        z[offset+i*n_short]=
```

```
            X[0][i+n_short]*right_window_shape[floor(tr_pos_r)];
```

```
        tr_pos_r += tr_scale_r;
```

```
    }
```

40

```
    offset += n_short;
```

```
    for ( k = 1 ; k < 7 ; k++ ) {
```

```
        tr_scale_l = n_short*os_factor_win;
```

```
        tr_pos_l = tr_scale_l/2;
```

```
        tr_pos_r = os_factor_win*n_long-tr_pos_l;
```

```
        for ( i = 0 ; i < n_short ; i++ ) {
```

```
            z[i + offset] += X[k][i]*right_window_shape[floor(tr_pos_r)];
```

```
            z[offset + n_short + i] =
```

50

```

        X[k][n_short + i]*right_window_shape[floor(tr_pos_l)];
        tr_pos_l += tr_scale_l;
        tr_pos_r -= tr_scale_l;
    }
    offset += n_short;
}

tr_scale_l = n_short*os_factor_win;
tr_pos_l = tr_scale_l/2;
10

for ( i = n_short - 1 ; i >= 0 ; i-- ) {
    z[i + offset] += X[7][i]*right_window_shape[(int) floor(tr_pos_l)];
    tr_pos_l += tr_scale_l;
}

for ( i = 0 ; i < n_short ; i++ ) {
    z[offset + n_short + i] = X[7][n_short + i];
}

tr_scale_r = 0.5*n_long/warpedTransLenRight*os_factor_win;
tr_pos_r = 0.5*tr_scale_r+.5;
20

tr_pos_r = (1.5*n_long-(float)wEnd-0.5+warpedTransLenRight)*tr_scale_r;
for(i=3*n_long-1-last_pos ; i<=wEnd; i++) {
    z[i] *= right_window_shape[floor(tr_pos_r)];
    tr_pos_r += tr_scale_r;
}

for(i=lsat_pos+1; i<2*n_long; i++)
    z[i] = 0.;
30
【 0 2 6 8 】
b ) その他すべて :

tw_windowing_long(X[[]], z[], first_pos, last_pos, warpe_trans_len_left, warped_trans
_len_right, left_window_shape[], right_window_shape[]){

for(i=0; i<first_pos; i++)
    z[i] = 0.;
for(i=last_pos+1; i<N; i++)
    z[i] = 0.;
40

tr_scale = 0.5*n_long/warped_trans_len_left*os_factor_win;
tr_pos = (warped_trans_len_left+first_pos-N/4)+0.5)*tr_scale;

for(i=N/2-1-first_pos; i>=first_pos; i-- ) {
    z[i] = X[0][i]*left_window_shape[floor(tr_pos)];
    tr_pos += tr_scale;
}

tr_scale = 0.5*n_long/warped_trans_len_right*os_factor_win;
50

```

```

tr_pos = (3*N/4-last_pos-0.5+warped_trans_len_right)*tr_scale;

for(i=3*N/2-1-last_pos;i<=last_pos;i++) {
  z[i] = X[0][i]*right_window_shape[floor(tr_pos)];
  tr_pos += tr_scale;
}
}

```

【 0 2 6 9 】

4 . M D C T ベースの T C X

4 . 1 ツールの説明

上記core\_modeが1に等しく、3つのTCXモードのうち1つ以上が“線形予測ドメイン”符号化として選択されたとき、即ちmode[]の4つのアレイエントリのうち1つが0より大きいとき、MDC TベースのTCXツールが使用される。MDC TベースのTCXは、量子化されたスペクトル係数を算術復号器から受信する。量子化された係数は、まずコンフォートノイズによって完成された後、逆MDC T変換が適用され、時間ドメインの重み付き合成信号を得るために、次に重み付け合成LPCフィルタへと送られる。

【 0 2 7 0 】

4 . 2 定義

- lg 算術符号器から出力された、量子化済スペクトル係数の数
- noise\_factor ノイズレベルの量子化指標
- noise\_level 再構築されたスペクトル中に注入されたノイズのレベル
- noise[] 生成されたノイズのベクトル
- global\_gain 再スケーリングゲインの量子化指標
- g 再スケーリングゲイン
- rms 合成された時間ドメイン信号x[]の平均二乗根
- x[] 合成された時間ドメイン信号

【 0 2 7 1 】

4 . 3 復号化のプロセス

MDC TベースのTCXは、算術復号器からいくつかの量子化されたスペクトル係数lgを要求し、このlgはmod[]とlast\_lpd\_mode値とによって決定される。これら2つの値は、逆MDC T内に適用されるウィンドウ長及び形状をも定義する。ウィンドウは3つの部分、即ちL個のサンプルからなる左側のオーバーラップ部分と、M個のサンプルの1からなる中央部分と、R個のサンプルからなる右側のオーバーラップ部分とから構成される。次の表11又は図14fに対応して図14gで示したように、2<sup>lg</sup>の長さを持つ1つのMDC Tウィンドウを得るために、ZL個のゼロが左側部分に、ZR個のゼロが右側部分に加算される。

【表11】

表11 -last\_lpd\_mode and mod[]の関数としてのスペクトル係数の数

last_lpd_mode の値	mod[x] の値	スペクトル 係数の数 lg	ZL	L	M	R	ZR
0	1	320	160	0	256	128	96
0	2	576	288	0	512	128	224
0	3	1152	512	128	1024	128	512
1..3	1	256	64	128	128	128	64
1..3	2	512	192	128	384	128	192
1..3	3	1024	448	128	896	128	448

10

20

30

40

50

【 0 2 7 2 】

M D C T ウィンドウは以下のように与えられる。

【 数 2 7 2 】

$$W(n) = \begin{cases} 0 & \text{for } 0 \leq n < ZL \\ W_{SIN\_LEFT,L}(n-ZL) & \text{for } ZL \leq n < ZL+L \\ 1 & \text{for } ZL+L \leq n < ZL+L+M \\ W_{SIN\_RIGHT,R}(n-ZL-L-M) & \text{for } ZL+L+M \leq n < ZL+L+M+R \\ 0 & \text{for } ZL+L+M+R \leq n < 2lg \end{cases} \quad 10$$

【 0 2 7 3 】

量子化されたスペクトル係数quant[]は、算術復号器によってもたらされたものであるが、コンフォートノイズによって完成されている。注入されたノイズのレベルは、復号化されたnoise\_factorによって以下のように決定される：

noise\_level = 0.0625\*(8-noise\_factor)

【 0 2 7 4 】

ノイズベクトルnoise[]は、次に、- 1 又は + 1 の値を無作為にもたらすあるランダム関数random\_sign()を使用して計算される。

noise[i] = random\_sign()\*noise\_level;

20

【 0 2 7 5 】

quant[]とnoise[]のベクトルは、再構築されたスペクトル係数ベクトルr[]を形成するために、quant[]内における8個の連続したゼロのランがnoise[]の成分によって置き換えられるという方法により、結合される。8個の非ゼロのランは、以下の式に従って検出される：

【 数 2 7 5 】

$$\begin{cases} rl[i] = 1 & \text{for } i \in [0, lg/6[ \\ rl[lg/6+i] = \sum_{k=0}^7 |quant[lg/6+8 \cdot \lfloor i/8 \rfloor + k]| & \text{for } i \in [0, 7 \cdot lg/6[ \end{cases} \quad 30$$

【 0 2 7 6 】

再構築されたスペクトルは、以下のようにして取得する。

【 数 2 7 6 】

$$r[i] = \begin{cases} quant[i] & \text{if } rl[i] = 1 \\ noise[i] & \text{otherwise} \end{cases}$$

【 0 2 7 7 】

逆M D C Tを適用する前に、あるスペクトル逆シェーピングが以下のステップに従って適用される。 40

1. スペクトルの第1四半の各8次元ブロックのために、指標mにおける8次元ブロックE<sub>m</sub>のエネルギーを計算する。
2. R<sub>m</sub>=sqrt(E<sub>m</sub>/E<sub>l</sub>)の割合を計算する。ここで、lは全てのE<sub>m</sub>の最大値を持つブロックの指標である。
3. R<sub>m</sub><0.1のときは、R<sub>m</sub>=0.1に設定する。
4. R<sub>m</sub><R<sub>m-1</sub>のときは、R<sub>m</sub>=R<sub>m-1</sub>に設定する。

【 0 2 7 8 】

スペクトルの第1四半に属する各8次元ブロックは、次にファクターR<sub>m</sub>によって乗算される。 50

## 【 0 2 7 9 】

再構築されたスペクトルは、ある逆MDC Tへと送られる。ウィンドウ化されていない出力信号x[]は、ゲインgによって再スケールされる。このゲインgは、次式のように、復号化されたglobal\_gain指標の逆量子化によって得られる。

## 【 数 2 7 9 】

$$g = 10^{\text{global\_gain}/28/(2.\text{rms})}$$

## 【 0 2 8 0 】

ここで、rmsは以下のように計算される：

## 【 数 2 8 0 】

$$\text{rms} = \sqrt{\frac{\sum_{i=1g/2}^{3*lg/2-1} x^2[i]}{L+M+R}}$$

10

## 【 0 2 8 1 】

上記再スケールされた時間ドメイン合成信号は、以下と等しくなる：

## 【 数 2 8 1 】

$$x_w[i] = x[i] \cdot g$$

20

## 【 0 2 8 2 】

再スケールした後で、ウィンドウ化及びオーバーラップ加算が適用される。

## 【 0 2 8 3 】

再構築されたTCX目標x(n)は、次に、ゼロ状態の逆重み付き合成フィルタ

$$\hat{A}(z)(1 - \alpha z^{-1}) / (\hat{A}(z/\lambda))$$

30

を介して濾波され、合成フィルタに適用されるべき励振信号が発見される。この濾波にはサブフレーム毎の補間済LPフィルタが使用されることに留意すべきである。励振が一旦決定すると、上述したように、励振を合成フィルタ

$$1/\hat{A}(z)$$

を介して濾波し、次にフィルタ  $1 / (1 - 0.68z^{-1})$  を介した濾波により逆強化することで、信号が再構築される。

## 【 0 2 8 4 】

ACELP適応型のコードブックを更新し、後続のフレーム内でTCXからACELPへの切替を可能にするためにも、励振が必要であることに留意すべきである。また、TCX合成の長さが、TCXフレーム長（オーバーラップは除く）によって付与されることにも留意すべきである。このTCXフレーム長は、それぞれ1, 2又は3を持つmod[]のための256、512又は1024個のサンプルである。

40

## 【 0 2 8 5 】

標準とする参考文献

[ 1 ] 非特許文献 1 0

[ 2 ] 非特許文献 1 1

[ 3 ] 非特許文献 1 2

[ 4 ] 非特許文献 1 3

50

- [ 5 ] 非特許文献 9
- [ 6 ] 非特許文献 1 4
- [ 7 ] 非特許文献 1 5
- [ 8 ] 非特許文献 1 6
- [ 9 ] 非特許文献 1 7
- [ 1 0 ] 非特許文献 1 8

【 0 2 8 6 】

定義

【 0 2 8 7 】

定義は、非特許文献 1 9 及び非特許文献 2 0 において見いだすことができる。

10

【 0 2 8 8 】

これまでいくつかの態様を装置の文脈において説明して来たが、これらの態様は対応する方法をも表すことは明白である。そこでは、1つのブロック又は装置は、1つの方法ステップ又は方法ステップの特徴に対応する。同様に、方法ステップの文脈において説明された態様もまた、対応するブロック、又は対応する装置の特徴若しくは項目の説明をも表している。

【 0 2 8 9 】

本発明の符号化された信号は、デジタル記憶媒体に記憶可能であり、又は、インターネットのような無線若しくは有線の伝送媒体を通じて伝送可能である。

【 0 2 9 0 】

20

本発明の方法の所定の実施条件に依るが、本発明の方法は、ハードウェア又はソフトウェアにおいて構成可能である。この構成は、その中に格納される電子的に読出し可能な制御信号を有し、本発明の各方法が実行されるようにプログラム可能なコンピュータシステムと協働する（或いは協働できる）、デジタル記憶媒体、例えばディスク、DVD、CD、ROM、PROM、EPROM、EEPROM或いはFLASHメモリなどを使用して実行することができる。

【 0 2 9 1 】

本発明に従う幾つかの実施例は、電子的に読出し可能な制御信号を持つデータキャリアを含み、その制御信号は、この明細書に記載した方法の1つが実行されるように、プログラム可能なコンピュータシステムと協働できるものである。

30

【 0 2 9 2 】

一般的に、本発明は、当該コンピュータプログラムがコンピュータ上で作動するときに、本発明の方法の1つを実行するためのプログラムコードを有する、コンピュータプログラム製品として実現されても良い。そのプログラムコードは、例えば機械読出し可能なキャリアに記憶されても良い。

【 0 2 9 3 】

本発明の他の実施例は、機械読出し可能なキャリアに記憶され、本発明の方法の1つを実行するためのコンピュータプログラムを含む。

【 0 2 9 4 】

換言すれば、本発明の方法のある実施例は、当該コンピュータプログラムがコンピュータ上で作動するときに、本発明の方法の1つを実行するためのプログラムコードを有する、コンピュータプログラムである。

40

【 0 2 9 5 】

本発明の方法のさらに他の実施例は、データキャリア（又はデジタル記憶媒体又はコンピュータ読出し可能な媒体）であり、その中に、本発明の方法の1つを実行するためのコンピュータプログラムを含む本発明の方法の1つを実行するためのプログラムコードを有する。

【 0 2 9 6 】

本発明の方法の別の実施例は、データストリーム又は信号のシーケンスであり、本発明の方法の1つを実行するためのコンピュータプログラムを表すものである。

50

【0297】

本発明の他の実施例は、例えばコンピュータやプログラム可能な論理回路などの処理手段であって、本発明の方法の1つを実行するように構成されるかあるいは適合する、処理手段を含む。

【0298】

本発明のさらに他の実施例は、本発明の方法の1つを実行するコンピュータプログラムがその中にインストールされた、コンピュータを含む。

【0299】

幾つかの実施例においては、本発明の方法の一部或いは全ての機能を実行するために、プログラム可能な論理回路（例えば書換え可能ゲートアレイ）を使用しても良い。幾つかの実施例においては、本発明の方法の1つを実行するために、書換え可能ゲートアレイがマイクロプロセッサと協働しても良い。一般的に、本発明の方法は、任意のハードウェア装置によって実行できることが好ましい。

【0300】

上述した実施例は、本発明の原理を単に例示的に示したにすぎない。本明細書に記載した構成及び詳細について、修正及び変更が可能であることは、当業者にとって明らかである。従って、本発明は、以下に添付する特許請求の範囲の技術的範囲によってのみ限定されるものであり、本明細書に実施例の説明及び解説の目的で提示した具体的詳細によって限定されるものではない。

10

【図1a】

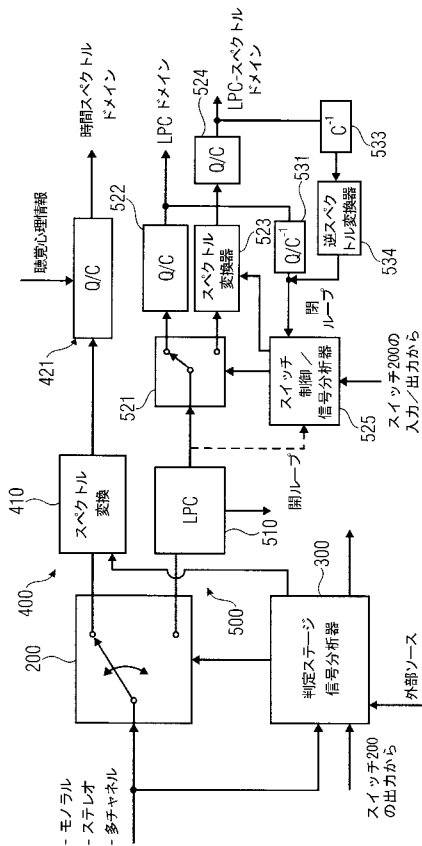


FIGURE 1A (符号器)

【図1b】

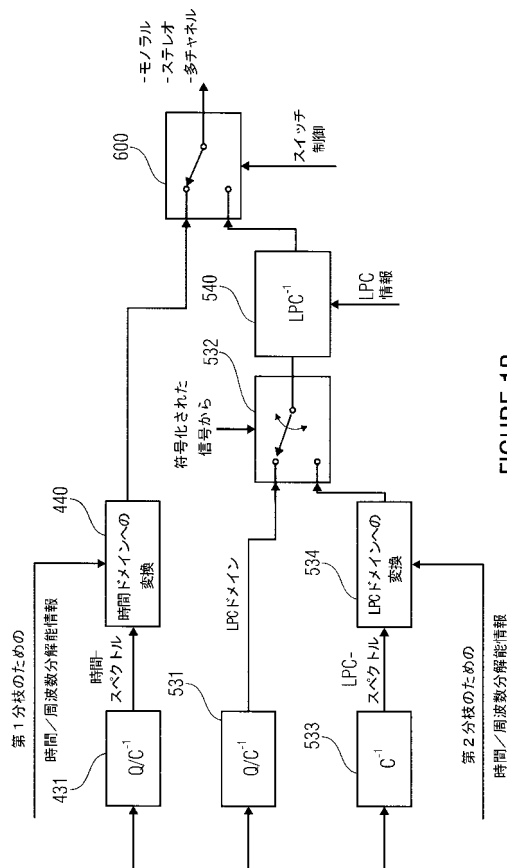


FIGURE 1B (復号器)

【図1c】

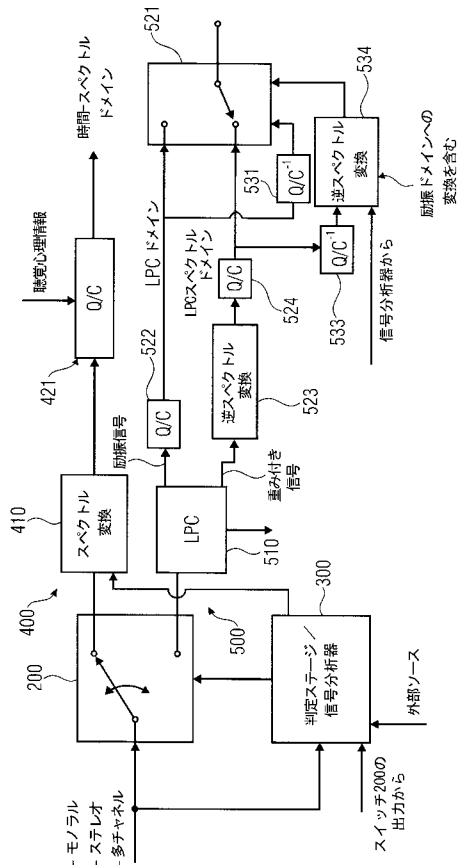


FIGURE 1C  
(符号器)

【図2a】

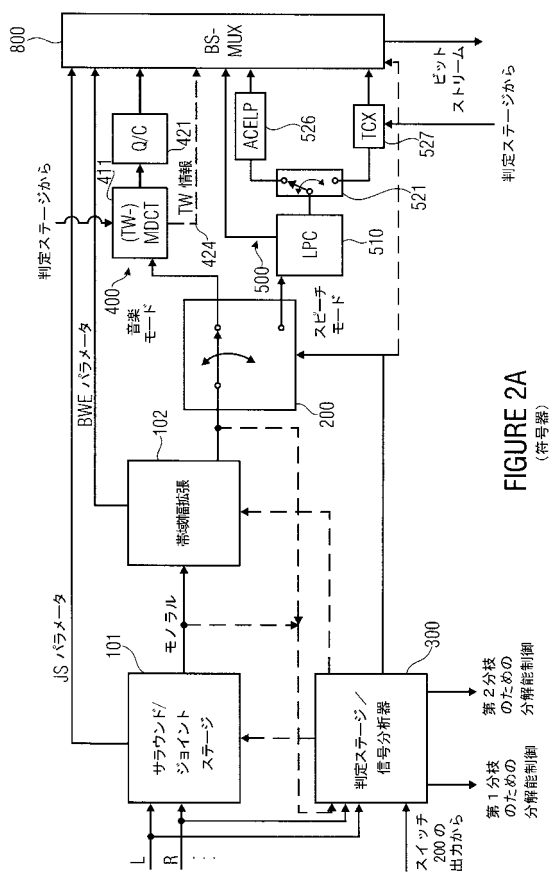


FIGURE 2A  
(符号器)

【図2b】

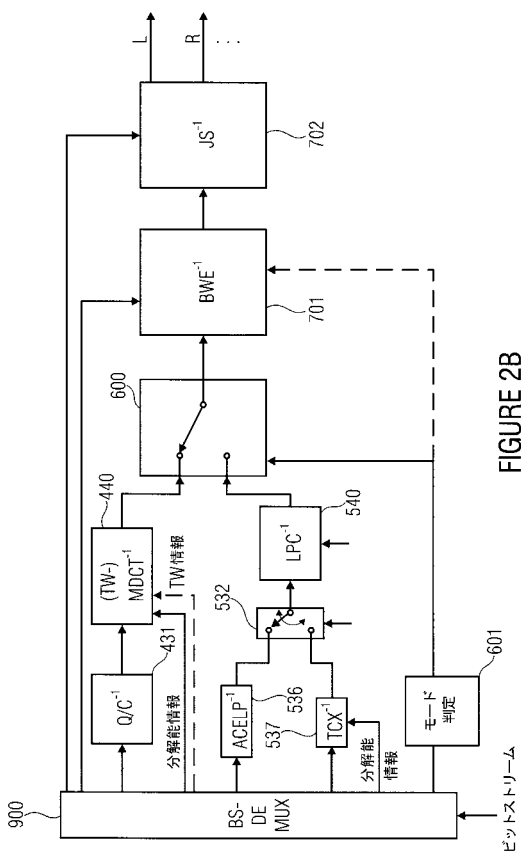


FIGURE 2B  
(復号器)

【図2c】

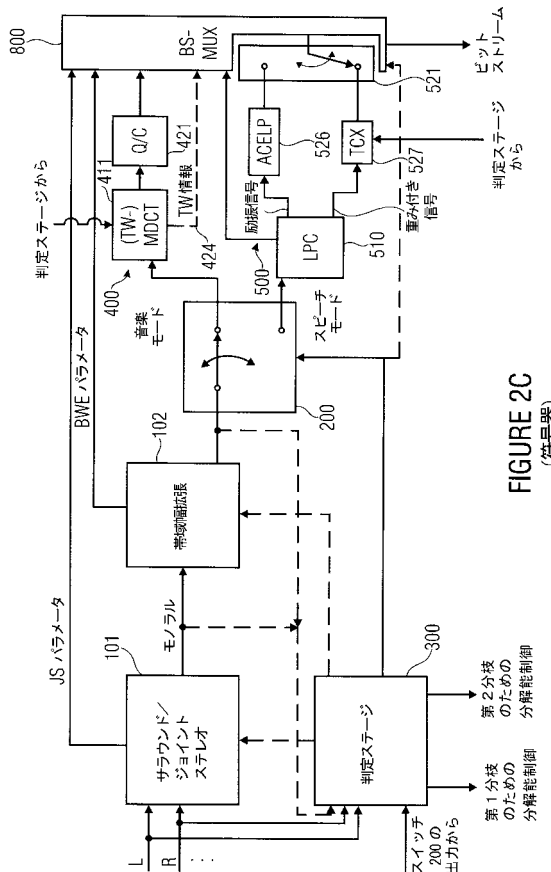


FIGURE 2C  
(復号器)

【図3a】

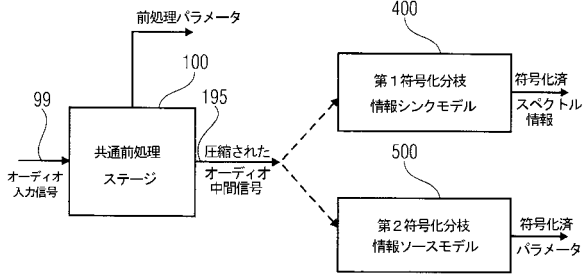


FIGURE 3A

【図3b】

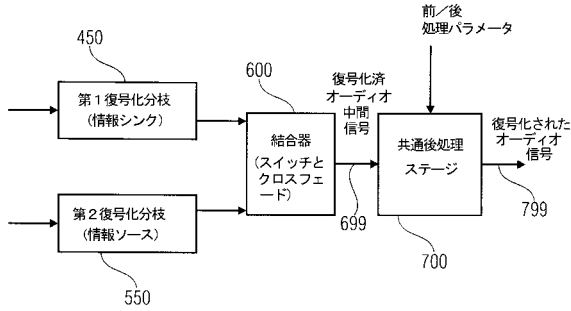


FIGURE 3B

【図3d】

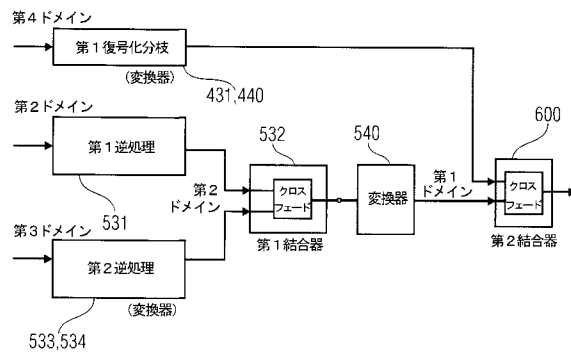


FIGURE 3D

【図3e】

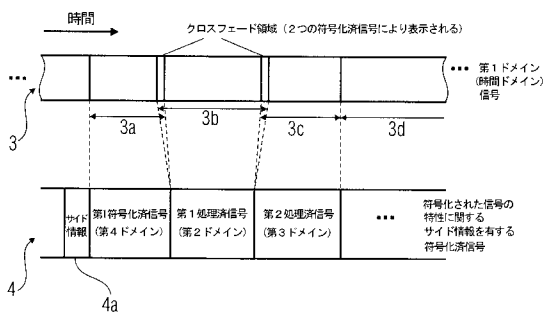


FIGURE 3E

【図3c】

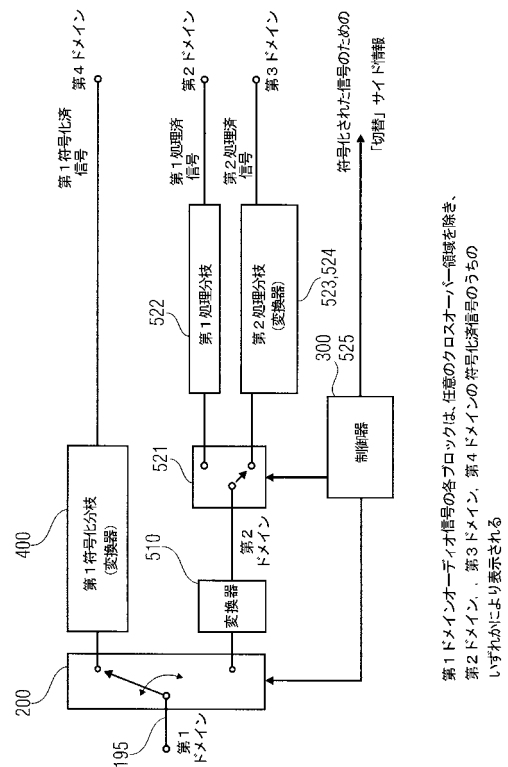


FIGURE 3C

第1ドメインオーディオ信号の各ブロックは、任意のクロスオーバー領域を除き、第2ドメイン、第3ドメイン、第4ドメインの符号化済信号のうちのいずれかにより表示される

【図4a】

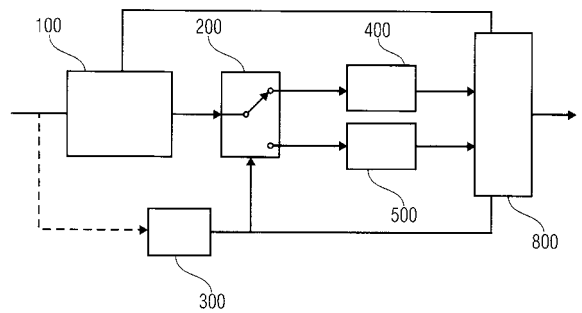


FIGURE 4A

【図4b】

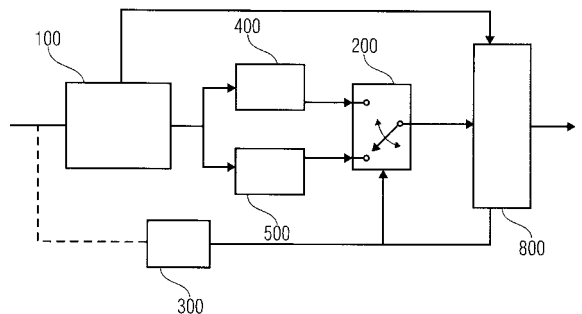


FIGURE 4B

【図 5 a b】

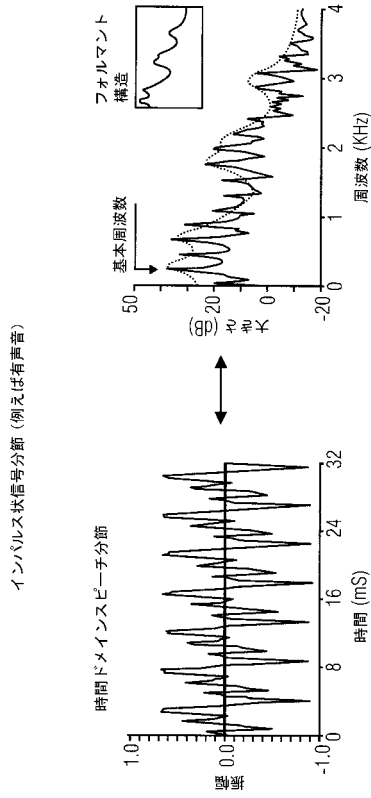


FIGURE 5A

【図 5 c d】

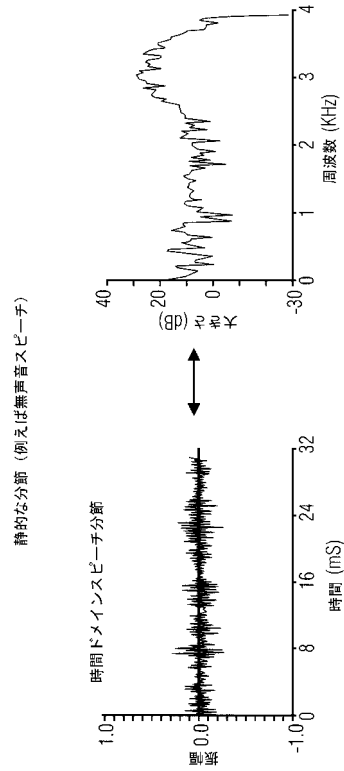


FIGURE 5C

FIGURE 5D

【図 6】

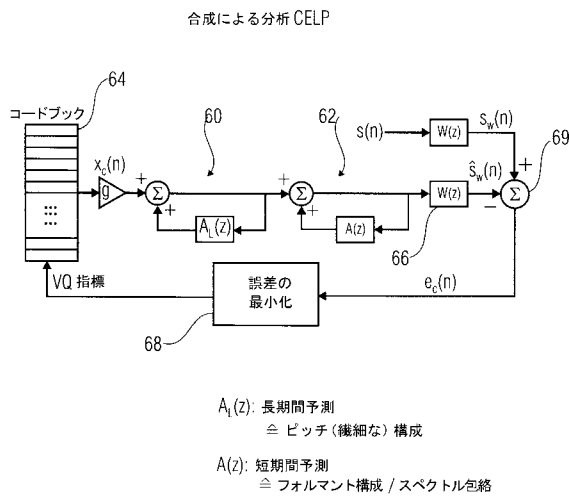


FIGURE 6

【図 7 a】

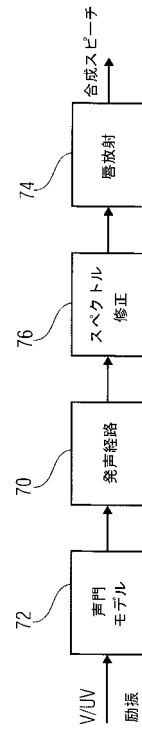


FIGURE 7A

【図7b】

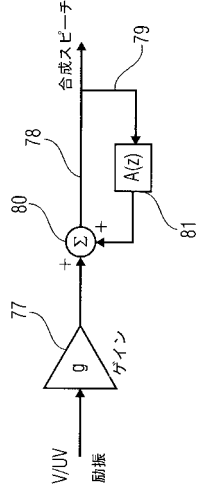


FIGURE 7B

【図7c】

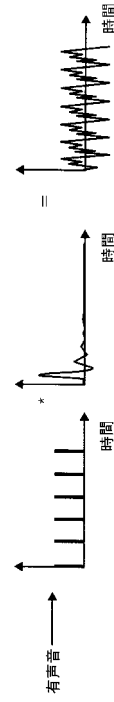


FIGURE 7C

【図7d】

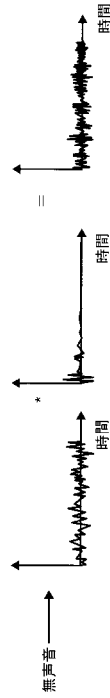


FIGURE 7D

【図7e】

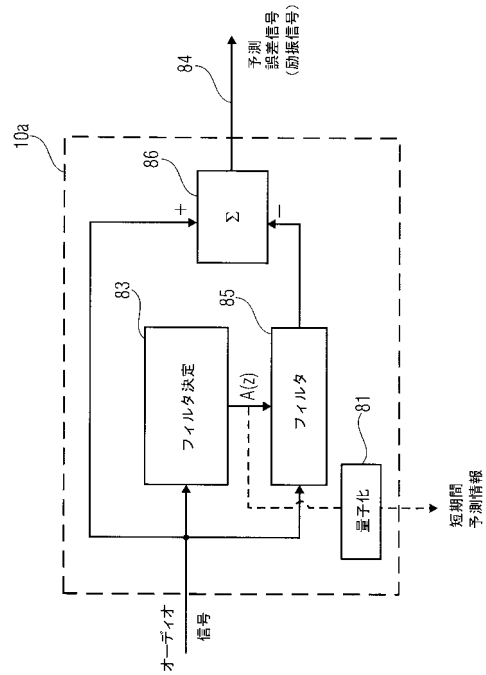


FIGURE 7E

【図7f】

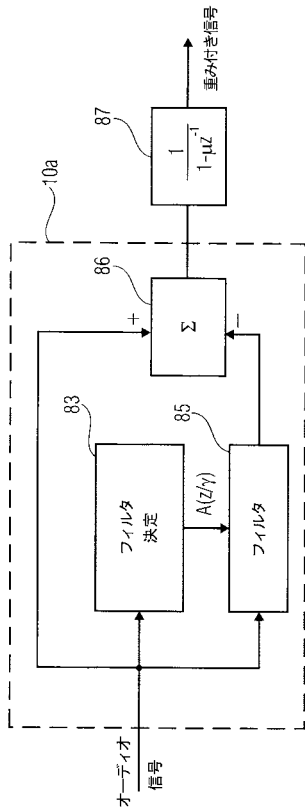


FIGURE 7F  
(符号器側)

【図7g】

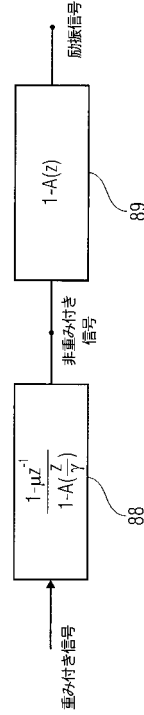


FIGURE 7G  
(復号器側)

【図8】

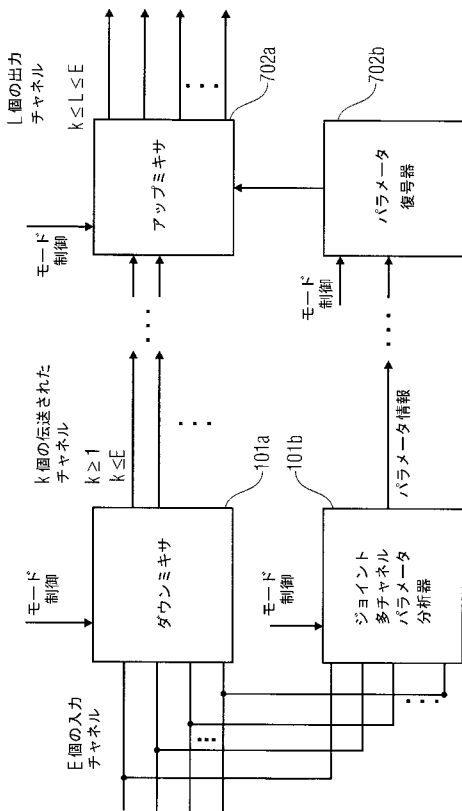


FIGURE 8

【図9】

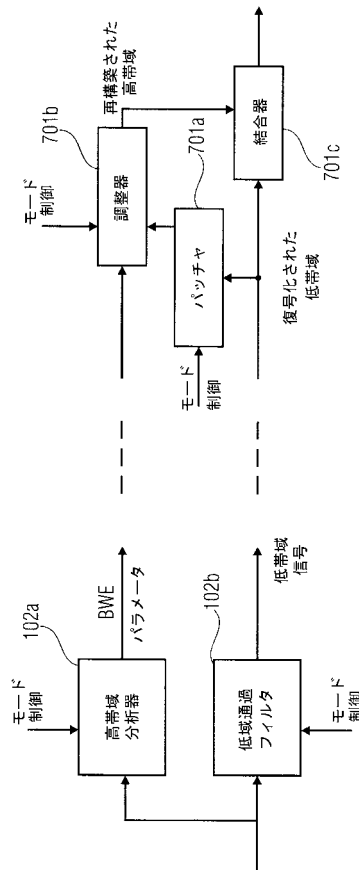


FIGURE 9

【図10a】

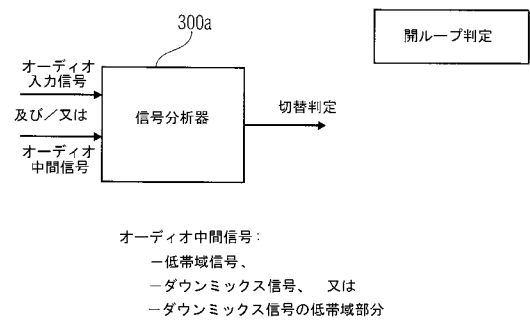


FIGURE 10A

【図10b】

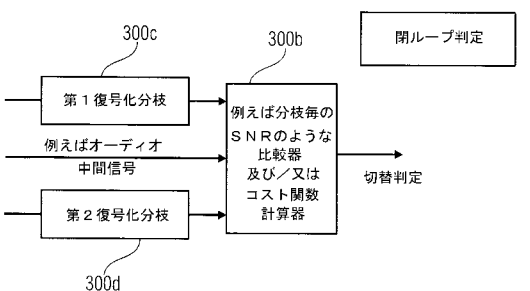


FIGURE 10B

【図11b】

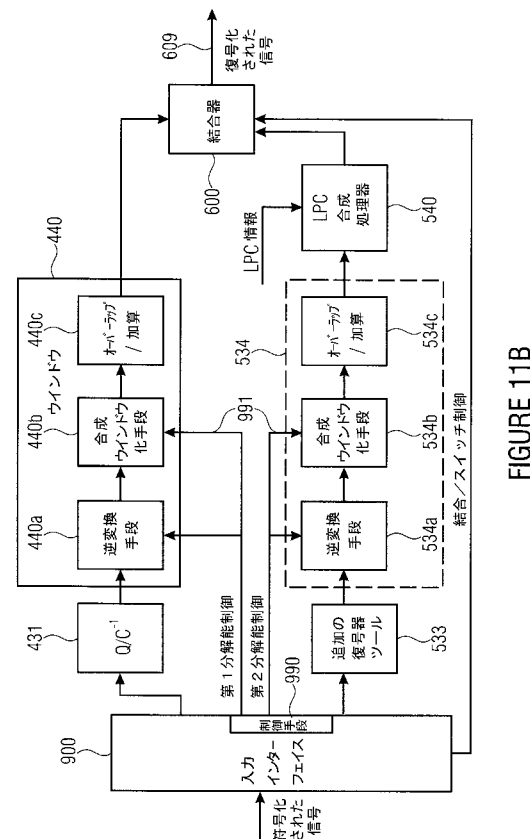


FIGURE 11B

【図11a】

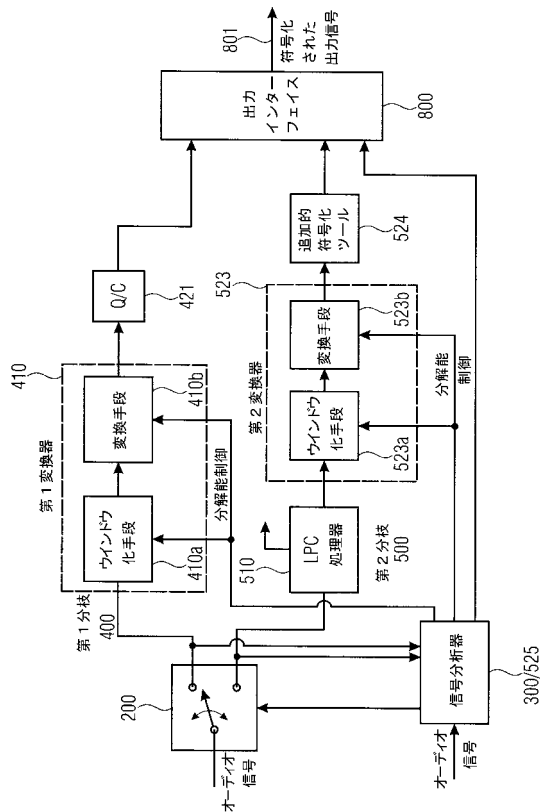


FIGURE 11A

【図12a】

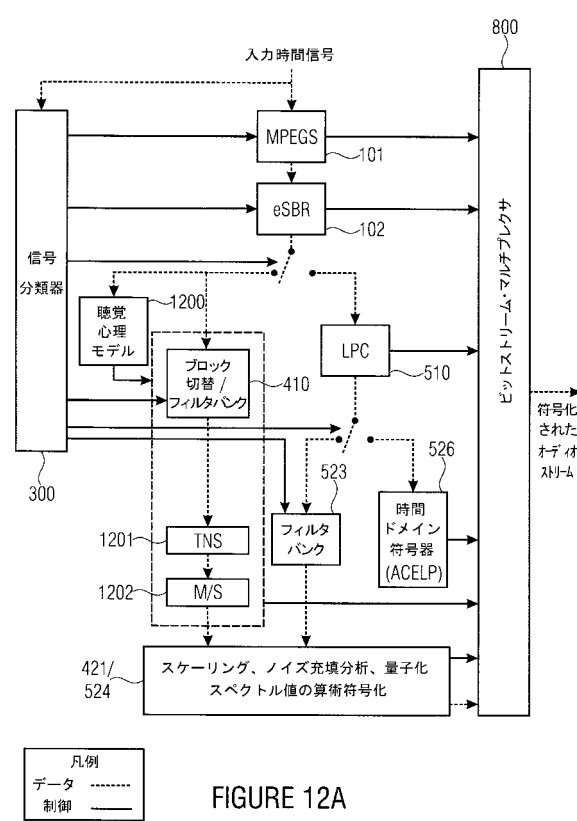


FIGURE 12A

【図12b】

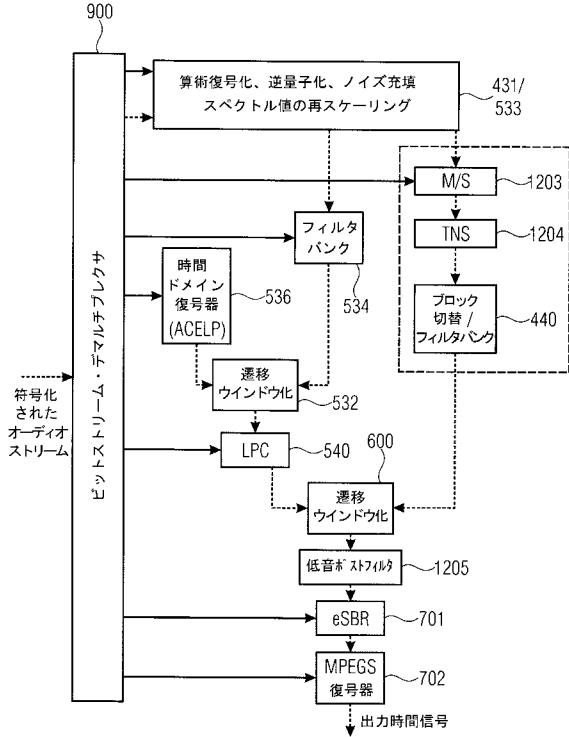


FIGURE 12B

【図13a】

ウィンドウ長	変換長	時間分解能	周波数分解能
短い	小さい	高い	低い
長い	大きい	低い	高い

FIGURE 13A

【図13b】

ウィンドウ	係数の数	形状
LONG_WINDOW	1024/ 960	
SHORT_WINDOW	128/ 120	
LONG_START_WINDOW	1024/ 960	
LONG_STOP_WINDOW	1024/ 960	
STOP_START_WINDOW	1024/ 960	
START_WINDOW_LPD	1024/ 960	
STOP_WINDOW_1152	1152/ 1080	
STOP_START_WINDOW_1152	1152/ 1080	

変換ウィンドウ

FIGURE 13B  
(AAC分枝と遷移)

【図13c】

値	window_sequence	num_windows	形状
0	ONLY_LONG_SEQUENCE = LONG_WINDOW	1	
1	LONG_START_SEQUENCE = LONG_START_WINDOW	1	
2	EIGHT_SHORT_SEQUENCE = 8 * SHORT_WINDOW	8	
3	LONG_STOP_SEQUENCE = LONG_STOP_WINDOW	1	
1	STOP_START_SEQUENCE = STOP_START_WINDOW	1	
3	LPD_START_SEQUENCE = START_WINDOW_LPD	1	
3	STOP_1152_SEQUENCE = STOP_WINDOW_1152	1	
1	STOP_START_1152_SEQUENCE = STOP_START_WINDOW_1152	1	

ウィンドウシーケンス

FIGURE 13C  
(AAC分枝と遷移)

【図14a】

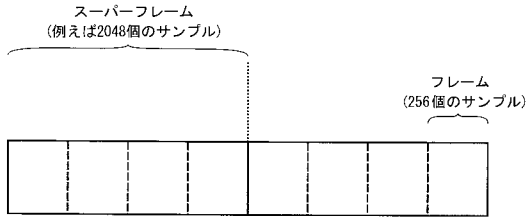


FIGURE 14A

【図14b】

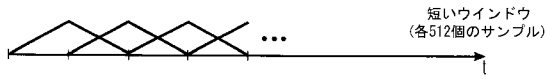


FIGURE 14B

【図14c】

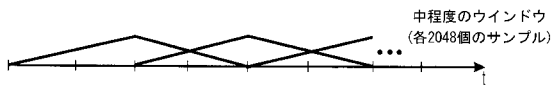


FIGURE 14C

【図14g】

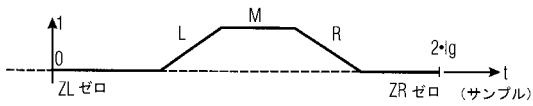


図14Fのウィンドウ定義

FIGURE 14G

【図14d】

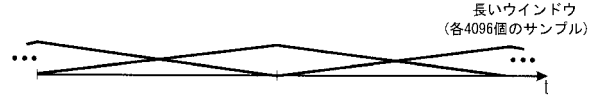


FIGURE 14D

【図14e】

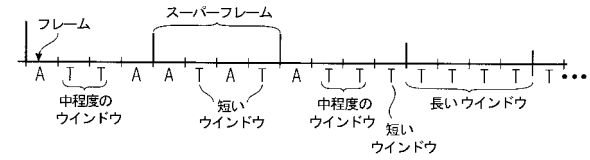


FIGURE 14E

【図14f】

last_lpd_mode の値	mod[x] の値	スペクトル 係数の数 lg	ZL	L	M	R	ZR
0	1	320	160	0	256	128	96
0	2	576	288	0	512	128	224
0	3	1152	512	128	1024	128	512
1..3	1	256	64	128	128	128	64
1..3	2	512	192	128	384	128	192
1..3	3	1024	448	128	896	128	448

FIGURE 14F

## フロントページの続き

(31)優先権主張番号 09002271.6

(32)優先日 平成21年2月18日(2009.2.18)

(33)優先権主張国 欧州特許庁(EP)

(72)発明者 ノイエンドルフ マックス

ドイツ連邦共和国 9 0 4 0 8 ニュルンベルク クライリンクストラーセ 1 5

(72)発明者 バイエル ステファン

ドイツ連邦共和国 9 0 4 1 9 ニュルンベルク ヨハニスストラーセ 1 4 8

(72)発明者 レコンテ ジェレミー

ドイツ連邦共和国 9 0 4 8 9 ニュルンベルク ズルツパッヒャー ストラーセ 3 9

(72)発明者 フッハス ギローム

ドイツ連邦共和国 9 1 0 5 2 エルランゲン エウスゼーレ ブルッカー ストラーセ アプト  
ヴェスト

(72)発明者 ロピラード ジュリアン

ドイツ連邦共和国 9 0 4 0 8 ニュルンベルク インネーラー クラインロイター ヴェーク  
2 5 アー

(72)発明者 レッテルバッハ ニコラス

ドイツ連邦共和国 9 0 4 2 7 ニュルンベルク アモルバヒャー ストラーセ 2 アー

(72)発明者 ナーゲル フレデリック

ドイツ連邦共和国 9 0 4 2 5 ニュルンベルク ヴィルヘルムシャーヴェナー ストラーセ 7  
2

(72)発明者 ガイガー ラルフ

ドイツ連邦共和国 9 8 6 9 3 イルメノイ ミュンツストラーセ 8 ツエー

(72)発明者 ムルトラス マルクス

ドイツ連邦共和国 9 0 4 6 9 ニュルンベルク エツラウプヴェーク 7

(72)発明者 グリル ベルンハルト

ドイツ連邦共和国 9 1 2 0 7 ラウフ ペーター - ヘンライン - ストラーセ 7

(72)発明者 ゲールネイ フィリップ

カナダ ケベック J 1 L 0 A 2 シャーブルック, リュ デ スーピノン 3 0 1 2

(72)発明者 サラミ レドワン

カナダ ケベック H 4 R 2 Y 3 サン - ローラン, アルベール - ドルー プラス 4 0 4 5

審査官 山下 剛史

(56)参考文献 特開平8 - 263098 (JP, A)

特表2007 - 538283 (JP, A)

特表2007 - 525707 (JP, A)

国際公開第2008 / 022566 (WO, A1)

(58)調査した分野(Int.Cl., DB名)

G 1 0 L 1 9 / 0 0 - 1 9 / 2 6