

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2010-141910

(P2010-141910A)

(43) 公開日 平成22年6月24日(2010.6.24)

(51) Int.Cl.	F I	テーマコード (参考)
H04L 1/00 (2006.01)	H04L 1/00 A	5K014
H04W 88/02 (2009.01)	H04Q 7/00 643	5K067

審査請求 有 請求項の数 4 O L (全 23 頁)

(21) 出願番号	特願2010-14916 (P2010-14916)	(71) 出願人	000005223
(22) 出願日	平成22年1月27日 (2010.1.27)		富士通株式会社
(62) 分割の表示	特願2003-526110 (P2003-526110) の分割		神奈川県川崎市中原区上小田中4丁目1番1号
原出願日	平成13年8月31日 (2001.8.31)	(74) 代理人	100084711
			弁理士 斉藤 千幹
		(72) 発明者	村田 秀一
			日本国神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	中谷内 夏彦
			日本国神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		Fターム(参考)	5K014 BA01 BA06
			5K067 AA41 BB04 EE02 EE10 GG11
			HH22 HH25

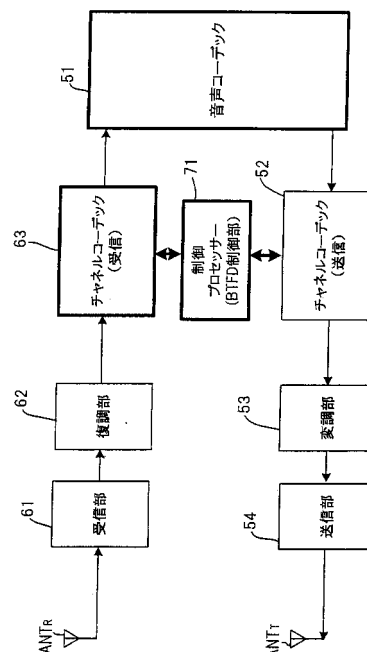
(54) 【発明の名称】 受信装置

(57) 【要約】

【目的】正しく各クラスの音声符号のビット数を識別できるようにする。

【構成】誤り検出用データを含む可変長のデータで変調された信号を受信し、復調する受信手段と、複数のデータ長の候補を順に選択し、復調した信号について該選択したデータ長分だけ抽出して、前記誤り検出用データを用いて誤り検出を行う誤り検出手段と、該誤り検出手段により誤りを検出しなかった場合のデータ長の候補を前記可変長のデータ長として判定する判定手段を備え、前記判定手段は、前記誤り検出手段により誤りを検出しなかった最初のデータ長の候補を前記可変長のデータ長として判定する。

【選択図】図1



【特許請求の範囲】**【請求項 1】**

誤り検出用データを含む可変長のデータで変調された信号を受信し、復調する受信手段と、複数のデータ長の候補を順に選択し、復調した信号について該選択したデータ長分だけ抽出して、前記誤り検出用データを用いて誤り検出を行う誤り検出手段と、該誤り検出手段により誤りを検出しなかった場合のデータ長の候補を前記可変長のデータ長として判定する判定手段を備え、

前記判定手段は、前記誤り検出手段により誤りを検出しなかった最初のデータ長の候補を前記可変長のデータ長として判定する、

ことを特徴とする受信装置。

10

【請求項 2】

前記可変長のデータは、通話音声に基いて順次生成された、異なるデータ長を有する有音データ、無音データ、背景雑音データのいずれかと、誤り検出用データとを含むデータからなり、

前記誤り検出手段は、前記判定手段により判定した可変長のデータ長の候補を後続して受信した次の可変長のデータについて最初に選択するデータ長の候補とする、

ことを特徴とする請求項 1 記載の受信装置。

【請求項 3】

前記順次生成はミリ秒オーダで生成されることを特徴とする請求項 2 記載の受信装置。

【請求項 4】

前記可変長のデータは、通話音声に基いて所定の規則で順次生成された、異なるデータ長を有する有音データ、無音データ、背景雑音データのいずれかと、誤り検出用データとを含むデータからなり、

前記誤り検出手段は、前記判定手段により判定した可変長のデータ長の候補の履歴と前記所定の規則とを参照して、次の可変長のデータについて選択するデータ長の候補の順番を決定する、

ことを特徴とする請求項 1 記載の受信装置。

【発明の詳細な説明】**【技術分野】****【0001】**

30

本発明は、受信装置に係わり、特に、AMR (Adaptive Multi - Rate ; 適応マルチレート) 方式などの音声符号化方式で符号化された所定送信時間間隔の音声符号を複数のクラスに分けると共に各クラスの音声符号をそれぞれ所定ビットレートに応じたビット数で表現し、所定クラスの音声符号に一定長のチェックコードを付加し、かつ、各クラスの音声符号に誤り訂正符号化処理を施し、各クラスの誤り訂正符号化処理が施された音声符号を、前記チェックコード付加クラスを先頭にして多重して送信する通信システムにおける受信装置に関する。

【背景技術】**【0002】**

3 G P P 規格に準ずる W - CDMA システムにおいて、端末装置は基地局装置から複数のトランスポートチャネル TrCH の多重データを受信する際、10 ms のフレーム毎に各フレームにマッピングされている TFCI ビット (Transport Format Combination Indicator) を復号する。そして、この TFCI ビットに基いて、各トランスポートチャネル TrCH のビットレート、すなわち単位時間あたりの情報ビット長 (トランスポートフォーマット) を識別する。しかる後、端末装置は、識別したトランスポートフォーマットに基いて受信した多重データより各トランスポートチャネルの送信データを分離する。

ところで、低ユーザレートのチャネルに関しては、この TFCI ビットが存在しない場合がある。その際、CRC チェックを利用した BTDF (Blind Transport Format Detection) 処理により各トランスポートチャネル TrCH におけるトランスポートフォーマットを判別して受信多重データより各トランスポートチャネルの送信データを再生する。

50

3GPP規格によれば、BTDFD処理は音声符号の受信に際して適用される。すなわち、送信側の音声コーデック部は音声信号を例えばAMR方式により、(1)人間の声道を表現するLSPパラメータ、(2)音声の周期性を表現するピッチ周期成分、(3)音声に含まれる雑音成分、(4)ピッチ周期成分のゲイン、(5)雑音成分のゲインで表現し、入力音声よりこれら各要素を抽出して量子化し、量子化データを音声符号として出力する。LSPパラメータ、ピッチ周期成分、ピッチゲインは重要であり、第1のトランスポートチャネル(クラスAのTrCH)に分配し、雑音成分や雑音ゲインは少しぐらいエラーが乗っても致命的でないため、第2、第3のトランスポートチャネル(クラスB、CのTrCH)に分配する。

送信装置は、音声コーデック部より得られた各クラスの音声符号をそれぞれ所定ビットレートに応じたビット数で表現し、クラスAの音声符号に一定長のチェックコードを付加し、かつ、各クラスの音声符号に誤り訂正符号化処理を施し、各クラスの誤り訂正符号化処理が施された音声符号を、前記チェックコード付加クラスAを先頭にして多重して送信する。

受信装置は、CRCチェックを利用したBTDFD処理により各クラスA～CのトランスポートチャネルTrCHにおけるトランスポートフォーマット(ビット長)を判別し、該ビット長に基いて受信した各クラスの音声符号を抽出し、音声コーデックに入力し、音声コーデックは音声符号より音声信号を復元して出力する。なお、受信側において、受信物理チャネルは高位レイヤにてロジカルチャネルに遷移するが、物理チャネル～ロジカルチャネルに遷移する間に、トランスポートチャネル(TrCH)という状態に遷移する。音声はTrCHが3チャネルで1つのロジカルチャネルとなり、各TrCHがそれぞれ、Class A、Class B、Class Cのトランスポートチャネルとなる。

図9は従来の移動局の構成図である。送信に際して、音声コーデック部1はマイク2から入力する音声信号を送信時間間隔TTI=20ms毎にAMR符号化方式により音声符号に変換し、クラスA～Cの音声符号としてデータ分配部4に入力する。データ分配部4は音声コーデックからの指示によりクラスA～Cの音声符号を選択的に符号化時間長20msの送信バッファ部5₁～5₃に入力する。

各送信バッファ部5₁～5₃はクラスA～Cの音声符号(送信データ)を20ms毎に図示しないバッファメモリに書き込むと共に、次段のエンコード処理部6₁～6₃に入力する。

エンコード処理部6₁～6₃はそれぞれ時間20msの送信データを畳み込み符号あるいはターボ符号に従って符号化し(クラスAについてはCRCチェックビット付加後)、フレーム単位(10ms単位)に分割して多重部7に入力する。多重部7は各エンコード処理部6₁～6₃から入力する誤り訂正符号化データを10ms毎に多重して1フレーム分の多重データを作成し、多重された符号化データを同相成分(IN-Phase component)データとして送出する。

制御信号発生部8はパイロットPILOT、TFCI、TPCなどの制御データを直交成分(Quadrature component)データとして一定シンボル速度で出力する。QPSK拡散&変調器9のQPSK拡散器9aは入力される同相成分(Ich成分)、直交成分(Qch成分)に所定の拡散コードを用いて拡散変調を施し、DA変換してQPSK直交変調器9bに入力する。直交変調器9bはIch信号、Qch信号にQPSK直交変調を施し、無線送信部10は直交変調器から出力するベースバンド信号を高周波数に周波数変換(IF→RF)すると共に、高周波増幅等を行ってアンテナANT_Tより送信する。

受信に際して、無線受信部11はアンテナATN_Rにより受信した高周波信号をベースバンド信号に周波数変換(RF→IF変換)し、しかる後、ベースバンド信号を直交検波して同相成分(I成分)データと直交成分(Q成分)データを発生し、AD変換して逆拡散復調部12に入力する。逆拡散復調部12はI成分信号、Q成分信号に拡散符号と同じ符号を用いて逆拡散処理を施し、送信されてきた符号化データを復調(同期検波)し、分離部13に入力する。

分離部13は入力する多重データよりフレーム毎(10ms毎)にクラスA～Cのデータを分離し、それぞれデコード処理部14₁～14₃に入力する。各デコード処理部14₁

10

20

30

40

50

～ 1 4₃ は 1 0 ms のデータを 2 つ結合して送信時間間隔 TTI = 2 0 ms のデータにし、しかる後、クラス A ～ C のデータに誤り訂正復号処理を施して元のクラス A ～ C の音声符号データを復号し、受信バッファ部 1 5₁ ～ 1 5₃ のバッファメモリに書き込む。受信バッファ部 1 5₁ ～ 1 5₃ はバッファメモリから同期してクラス A ～ C の音声符号データを読み出してデータ分配部 4 に入力し、データ分配部 4 は各クラスの音声符号データを音声コーデック 1 に入力する。音声コーデック部 1 は音声符号より音声信号を復元してスピーカ 3 より出力する。

以上要約すれば、W - CDMA システムにおける送信側チャネルコーデック部 2 1 は、上位レイヤよりクラス A ～ C の音声符号データを受け取り、クラス A ～ C のトランスポートチャネル (TrCH) 毎に符号化処理を行ない、符号化したデータを多重して物理チャネルにマッピングして送信する。逆に、受信側チャネルコーデック部 2 2 は、物理チャネル上の多重データよりクラス A ～ C のトランスポートチャネル (TrCH) 毎にデータを分離して復号処理を行ない、その結果 (クラス A ～ C の音声符号) を上位レイヤに受け渡す。

送信側の音声コーデック部 1 は前述のようにマイク 2 から入力する音声信号を送信時間間隔 TTI = 2 0 ms 毎に AMR 符号化方式により音声符号に変換し、クラス A ～ C の音声符号としてデータ分配部 4 に入力するが、各クラスの音声符号のビットレート (ビット長) は呼接続時に基地局から指示される。すなわち、基地局は呼接続時に各クラス A ～ C のビットレートの組み合わせ候補を複数、発信端末及び着信端末にそれぞれ通知すると共に、どの組み合わせのビットレートで音声符号を送信するか指示する。図 1 0 は 3GPP 規格における音声フォーマット (ビットレートの組み合わせ候補) の一例であり、TTI = 2 0 ms 毎に符号化した各クラス A ～ C の音声符号を表現するためのビット長を示している。図 1 0 には 1 0 種類のビットレートの組み合わせが示され、それぞれを特定するためにフレームタイプ番号が付されている。ビットレートの組み合わせ候補は分類すると、(1) 無音のビットレート組み合わせ (1 1 1 1)、(2) 背景雑音のビットレート組み合わせ (0 0 0 1)、(3) 有音のビットレート組み合わせ (0 0 0 0 ～ 1 1 1 0) がある。有音の場合、どのビットレート組み合わせを使用するかは、呼接続時の通信トラフィックによって決定される。具体的には、トラフィックが空いているときは、高ビットレート 12.2 kbps で高品質の音声データのやりとりを行うが、逆にトラフィックが混んでいるときは、混雑度に応じて低いレートに変えて送受信データのビット長を小さくする。なお、有音のビットレートの組み合わせは呼接続時に一度、決定されると終話時まで、そのビットレートは維持され、途中でビットレートの切り替えはしない。現状では、有音のビットレート組み合わせは 12.2 kbps のフレームタイプ 1 1 1 0 だけである。

背景雑音は聴感上の自然さを与えるために必要なものである。人間の会話には、音声のある区間 (有音区間) と、話の区切りや相手の話を黙って聞いている音声の無い区間 (無音区間) が存在する。また、一般には、音声にオフィス、自動車または街路などで生じる背景雑音が重畳される。従って、実際の音声通信では、音声に背景雑音が重畳されている区間 (有音区間) と背景雑音のみの区間 (無音区間) が存在することになる。そのため、無音区間を検出し、無音区間の情報伝送を停止することにより大幅な伝送量の削減が可能となる。しかし、無音区間で何もしないか、あるいは一定レベルの雑音を出力せざるを得ないため不自然で、聴取者に違和感が生じる。そこで、図 1 1 に示すように無音状態が継続し、7 回無音フレームが連続したとき、背景雑音を生成するために必要な 1 つの背景雑音フレームを挿入し、これにより背景雑音の伝送量を少なくしつつ、受信側で自然な違和感の無い再生を可能にする。

基地局は呼接続時に図 1 0 に示す各クラス A ～ C のビットレートの組み合わせ候補を発信端末及び着信端末にそれぞれ通知すると共に、フレームタイプ番号でどの組み合わせのビットレートで音声符号を送信するかを指示する。発信端末の音声コーデック部 1 (図 9) は、基地局より指示されたビット長で各クラスの音声符号を表現し、送信側チャネルコーデック部 2 1 はクラス A の音声符号に一定長の CRC チェックコードを付加し、各クラスの音声符号に誤り訂正符号化処理を施し、フレーム単位 (1 0 ms 単位) に分割し、各クラスの誤り訂正符号化データを 1 0 ms 毎に多重して 1 フレーム分の多重データを作成して送

10

20

30

40

50

出する。着信端末の受信側チャネルコーデック部 22 は多重データより各クラスのデータを分離し、それぞれに復号処理を施す。

図 12 は分離された後の TTI = 20 ms の各クラスのデータ構成であり、(A) はクラス A のデータ構成図、(B), (C) はクラス B, C のデータ構成図である。クラス A のデータは、(1) 基地局より指示されたフレームタイプに応じたビット長を有するクラス A の音声符号部分 A1、(2) 一定長の CRC チェック符号部分 A2、(3) エンブティ部分 A3 で構成されている。クラス B, C のデータは、(1) 基地局より指示されたフレームタイプに応じたビット長の音声符号部分 B1, C1、(2) エンブティ部分 B2, C2 で構成されている。尚、エンブティ部分には信号の送信がされていないが、雑音による信号が送信されている。

10

受信側チャネルコーデック部 22 は図 12 に示す構成のデータより音声符号部分 A1, B1, C1 (雑音による信号を除去して) のみを正確に切り取って音声コーデック部に入力しなければならない。そこで、従来は CRC チェックを利用した BTDF (Blind Transport Format Detection) 処理により各クラスの音声符号のビット長を判別して分離する。すなわち、受信側チャネルコーデック部 22 は、クラス A の音声符号が、図 10 に示す各フレームタイプにおけるクラス A の単位時間当たりのビット数 (0, 39, 42, 49, . . . 81) で表現されているものと仮定して受信データに誤り訂正復号処理を施す。尚、ビット数の小さい順に番号 $n_{end} = 1, 2, . . .$ が付される。

しかる後、復号データに対して CRC チェックにより全てのパターンについて一通り復号結果が正しいかの検査を行ってから、該検査により復号結果が正しいと判定したものを検索し、図 10 を参照してその時のビットレートの組み合わせにおける各クラスの単位時間当たりのビット数 (ビットレート) を求め、該ビット数を各クラスの音声符号を表現するビット数であると判別する。各クラスのビット長が判別すれば、受信側チャネルコーデック部 22 は図 12 (A) ~ (C) に示す構成のデータより音声符号部分のみを正確に切り取って音声コーデック部に入力する。

20

以下では BTDF 処理について説明するが、その前に BTDF 処理を理解する上で必要な畳み込み符号及びビタビ復号を説明する。

図 13 は畳み込み符号器の例であり、2 ビットのシフトレジスタ SFR と 2 つの排他的論理和回路 EXOR1, EXOR2 を備え、EXOR1 は入力と R_1 の排他的論理和 g_0 を出力し、EXOR2 は入力と R_0 と R_1 の排他的論理和 g_1 ("1" が奇数のとき出力は "1"、それ以外は "0") を出力する。したがって、入力データが 01101 の場合における畳み込み符号器の入出力関係及びシフトレジスタ SFR の状態は図 14 に示すようになる。

30

畳み込み符号器のシフトレジスタ SFR の内容は state (状態) と定義され、図 15 に示すように 00, 01, 10, 11 の 4 つの状態があり、それぞれを状態 a, 状態 b, 状態 c, 状態 d と表現する。図 13 の畳み込み符号器ではシフトレジスタ SFR の状態が a ~ d のいずれの状態であるか、及び、次に入力するデータが "0" であるか "1" であるかにより、出力 (g_0, g_1) 及び次の状態が一意に決まる。図 16 はかかる畳み込み符号器の状態と入出力の関係図であり、点線は "0" 入力、実線は "1" 入力を示している。例えば、

(1) 状態 a において、"0" が入力すると出力は 00 で状態は a になり、"1" が入力すると出力は 11 で状態は c になる。

40

(2) 状態 b において、"0" が入力すると出力は 11 で状態は a になり、"1" が入力すると出力は 00 で状態は c になる。

(3) 状態 c において、"0" が入力すると出力は 01 で状態は b になり、"1" が入力すると出力は 10 で状態は d になる。

(2) 状態 d において、"0" が入力すると出力は 10 で状態は b になり、"1" が入力すると出力は 01 で状態は d になる。

この入出力関係を用いて図 13 の畳み込み符号器の畳み込み符号を格子状表現すると図 17 (a) に示すようになる。ただし、 k は第 k ビット入力時点を意味し、符号器における初期 ($k = 0$) の状態は a (00) である。又、点線 "0" 入力、実線は "1" 入力を示し、線上の 2 つの数値は出力 (g_0, g_1) を示している。したがって、初期状態 a (

50

0 0)において、“0”が入力すると出力は0 0で状態はaになり、“1”が入力すると出力は1 1で状態はcになることがわかる。

この格子状表現図を参照すると、原データが1 1 0 0 1であれば、図17(b)の2点鎖線で示すパスを介して状態cに至り、符号器出力は

1 1 → 1 0 → 1 0 → 1 1 → 1 1

となることがわかる。

復号器の受信データ(g_0, g_1)が1 1 → 1 0 → 1 0 → 1 1 → 1 1で誤りのない理想状態を想定すると、図18(a)の2点鎖線で示すパスが得られ、点線を“0”、実線を“1”とすることにより、図18(b)に示すように1 1 0 0 1の復号結果を得ることができる。しかし、実際には、受信データに誤りが含まれる場合が多い。図18(c)に示すように、第5ビット目に誤りが発生し、硬判定受信データ(g_0, g_1)が1 1 → 1 0 → 0 0 → 1 1 → 1 1であるとする、 $k = 2$ のデータ入力時点で1 0, 0 1のどちらへ分岐すべきか迷う(エラー回数ERR = 1)。1 0とみなして上側のパスを選択すると、 $k = 3$, $k = 4$ で迷うことなく状態cに至る。したがって、2点鎖線パスでのエラー回数ERR = 1となり、その時の復号結果は1 1 0 0 1となる。一方、 $k = 2$ のデータ入力時点で0 1とみなして下側のパスを選択すると、 $k = 3$ の時点でも、どちらへ分岐すべきか迷い、総エラー回数ERR = 2となる。以後、同様にパスを選択し、分岐に迷った時にERRをカウントアップすると、最終的に以下の結果が得られる。すなわち、

復号結果を1 1 0 0 1とした時の総エラー回数ERR : 1

復号結果を1 1 1 0 0とした時の総エラー回数ERR : 2

復号結果を1 1 1 0 1とした時の総エラー回数ERR : 4

復号結果を1 1 1 1 0とした時の総エラー回数ERR : 3

復号結果を1 1 1 1 1とした時の総エラー回数ERR : 3

となる。そこで、エラー回数ERRが最小の復号結果1 1 0 0 1を選択して出力する。このようにすれば、受信データに誤りがあっても原データ1 1 0 0 1を正しく復元できる。

ところで、以上のように受信データに基づいて可能な全パスのエラー回数ERRを求め、そのうち最小のパスより原データを復号する処理は煩雑である。そこで、ビタビ復号は以下のように行う。尚、受信データは硬判定の結果1 1 1 0 0 0であったとする。図18の $k = 3$ の状態aでは、2つの入力パスがある。その該当するパスのみを取り出して記述したものを図19(a)に示す。2つのパスとは図中に示したパス(1)とパス(2)である。受信データと各パスで得られる復号データとの間のハミング距離(以後、パスメトリック値という)を計算すると図19(b)、(c)に示すように、それぞれ3、4となる。

この計算結果から、「パス(1)をたどって状態aに到達した」と仮定する方のパスメトリック値が、「パス(2)をたどって状態aに到達した」と仮定するより小さい。そのため、パス(1)の方が送信されたデータに応じたパスであるという信頼性が高いため生き残りとして残し、他のパスを捨てる。このパスの取捨選択処理を各状態a ~ dについて、時間 $k = 1$ から連続して実行すると任意の時間 k における各状態a, b, c, dにそれぞれ至るパスメトリック値が最小のパス(エラー最小のパス)を求めることができ、以後、同様の取捨選択処理を継続することができる。

以上より、N個の受信データが入力したとき、 $k = N$ における各状態a, b, c, dにそれぞれ至るパスメトリック値最小(エラー最小)の4本のパスのうち、パスメトリック値が最小のパスを決定し、該パスに基づいて復号データを出力する。図20は受信データ1 1 1 0 0 0 1 1 1 1であるときの各時間 k ($= 1 \sim 5$)の各状態a ~ dに至る最短パスを示しており、線上の数値はパスメトリック値である。 $k = 5$ のデータ入力時点では、状態cに至るパスのパスメトリック値が最小となる。従って、 $k = 5$ の時点の状態cから該パスに沿ってトレースバック処理を行うとデータ1 1 0 0 1が得られ、これが復号データとなる。以上の復号アルゴリズムがViterbiアルゴリズムである。

なお、 $k = n_{\text{end}}$ のデータ入力時点における状態a(=状態0)のパスメトリック値を $a_0(n_{\text{end}})$ 、各状態a ~ dのうち最大のパスメトリック値を $a_{\text{max}}(n_{\text{end}})$ 、各状態の

10

20

30

40

50

うち最少のパスメトリック値を $a_{\min}(n_{\text{end}})$ とおくと、符号化データに対して誤りが少ないほど $a_0(n_{\text{end}}) > a_{\min}(n_{\text{end}})$ の関係が顕著になる特性がある。すなわち、符号化データに対して誤りが少ないほど $a_0(n_{\text{end}})$ はより大きくなり、 $a_{\min}(n_{\text{end}})$ はより小さくなる特性がある。このため、 $[a_{\max}(n_{\text{end}}) - a_{\min}(n_{\text{end}})]$ に対する $[a_0(n_{\text{end}}) - a_{\min}(n_{\text{end}})]$ の割合は大きくなる。この特性より、次式

$$S(n_{\text{end}}) = -10 \log\{[a_0(n_{\text{end}}) - a_{\min}(n_{\text{end}})] / [a_{\max}(n_{\text{end}}) - a_{\min}(n_{\text{end}})]\} \text{ [dB]} \quad (1)$$

で与えられる $S(n_{\text{end}})$ 値は、誤りが少ないほど小さくなる。BTFD処理においてはこの $S(n_{\text{end}})$ 値が用いられる。BTFD処理は図 2 1 にしたがって詳細に説明するが、概略、以下の処理を段階的に行う。すなわち、

(a) ビットレートの候補が複数指定される。

(b) 各ビットレート候補におけるクラス A のビットレートに対して、ビットレートの小さい順 (n_{end} の小さな順) にビタビ復号を行い、Add - Compare - Select (以後、ACS) 処理を行いパスメトリック値を求め、そのパスメトリック値を用いて (1) 式により $S(n_{\text{end}})$ を計算する。

(c) $S(n_{\text{end}})$ と閾値 D との大小判断を行う。

(d) $S(n_{\text{end}})$ が閾値 D より小さければ、最終ビット位置においてパスメトリック値が最小の状態から、トレースバック処理を行う。

(e) トレースバック処理によりえられた復号データに対して CRC チェックを行う。

(f) CRC チェックが OK であれば、今回の $S(n_{\text{end}})$ とそれまでの最小値である $S_{\min}$ との比較をおこなう。

(b) ~ (f) をビットレートの候補数分行い、CRC チェック結果が OK で、かつ、最終的に最も確からしいもの、すなわち、 $S(n_{\text{end}})$ が最小のビットレート候補を選択する。このビットレート候補における各クラスのビット数が各クラスの音声符号を表現するビット数であると決定する。ただし、(c) の結果、 $S(n_{\text{end}})$ が閾値 D より大きい場合、つまり、確からしさが低い場合は (d), (e), (f) の処理は行わない。

図 2 1 は BTFD 処理フローである。

上位アプリケーションよりビットレートの候補 (図 1 0) が指定されるから、クラス A のビットレート (単位時間当たりのビット数) の小さい順に番号付けし、 $n_{\text{end}} = 1, 2, 3, \dots$ とする (ステップ 1 0 1)。ついで、 $n_{\text{end}} = 1$ 、 $S_{\min} = D$ 、 $n_{\text{end}}' = 0$ と初期化する (ステップ 1 0 2)。

しかる後、 n_{end} 位置まで ACS 処理を行い (ステップ 1 0 3)、(1) 式により $S(n_{\text{end}})$ を計算する (ステップ 1 0 4)。 $S(n_{\text{end}})$ が求まれば、この $S(n_{\text{end}})$ と閾値 D との大小判断を行う (ステップ 1 0 5)。

$S(n_{\text{end}}) \leq D$ であれば、 n_{end} 位置のパスメトリック値が最小の状態から、トレースバック処理を行う (ステップ 1 0 6)。ついで、トレースバック処理により得られた復号データに対して CRC チェックを行う (ステップ 1 0 7)。CRC チェックが OK であれば、今回の $S(n_{\text{end}})$ とそれまでの最小値である $S_{\min}$ との比較を行う (ステップ 1 0 9)。

$S_{\min} > S(n_{\text{end}})$ であれば $S_{\min} = S(n_{\text{end}})$ として最小値を更新すると共に、 $n_{\text{end}}' = n_{\text{end}}$ としてその時の n_{end} 、すなわち、クラス A における音声符号のビット数を保存する (ステップ 1 1 0)。しかる後、 n_{end} が最後の候補であるかチェックし (ステップ 1 1 1)、最後でなければ $n_{\text{end}} = n_{\text{end}} + 1$ により n_{end} を歩進し (ステップ 1 1 2)、ステップ 1 0 3 以降の処理を繰り返す。

一方、ステップ 1 0 5 において、 $S(n_{\text{end}}) > D$ であれば、あるいは、ステップ 1 0 8 において、CRC チェックが NO であれば、あるいは、ステップ 1 0 9 において、 $S_{\min} \leq S(n_{\text{end}})$ であれば、ステップ 1 1 1 の処理を行う。

全候補の n_{end} について上記処理を繰り返せば、ステップ 1 1 1 において「YES」となるから、 $n_{\text{end}}' = 0$ であるかチェックし (ステップ 1 1 3)、「YES」であればエラー出力する (ステップ 1 1 4)。しかし、 $n_{\text{end}}' = 0$ でなければ、該 n_{end}' が最も確からしいビットレートの組み合わせにおけるクラス A のビットレート (ビット数) と判定し、出力す

10

20

30

40

50

る（ステップ 115）。以後、図 10 のビットレートの組み合わせ候補を参照して他のクラスのビットレート（ビット数）を求める。

図 22 は BTFD 処理を実行する受信側チャネルコーデック部の構成図である。図示しない分離・結合部は多重データよりクラス毎に分離したデータをクラス毎に結合して $TTI = 20 \text{ ms}$ の各クラスのデータを作成する。受信データメモリ部 22a は、この $TTI = 20 \text{ ms}$ の各クラスのデータを受信して保持する。ビタビ復号部 22b は ACS 演算 / パスメトリックメモリ / パスメモリ部 31、トレースバック部 32、トレースバック後メモリ 33 を備えている。パスメモリは各時点 k において、状態 a, b, c, d にそれぞれ至るパスメトリック値が最小の 4 本のパスをそれぞれ記憶するもの、パスメトリックメモリは各パスのパスメトリック値を記憶するものである。図 20 の $k = 5$ の例では、

状態 a に到るパスは 1 1 1 0 0 でパスメトリック値は 2、

状態 b に到るパスは 1 1 1 1 0 でパスメトリック値は 3、

状態 c に到るパスは 1 1 0 0 1 でパスメトリック値は 1、

状態 d に到るパスは 1 1 1 1 1 でパスメトリック値は 3、

である。トレースバック部 32 は状態 a, b, c, d に到る 4 本のパスのうちパスメトリック値が最小となるパスを決定し、該パスに沿ってトレースバック処理を行って復号データを求めてトレースバック後メモリ 33 に保存する。

ビタビ復号部は、A クラスについては n_{end} 位置までのデータに対してビタビ復号処理を施す。又、B, C クラスについてはデータ長が判らないため末尾位置までのビタビ復号処理を行い、各データ長候補に対応する時点 k において、状態 a, b, c, d にそれぞれ至るパスメトリック値が最小の 4 本のパスをパスメモリに記憶し、それぞれのパスメトリック値をパスメトリックメモリに記憶する。

CRC 演算部 22c はクラス A の復号結果に基づいて CRC チェック演算を行う。CRC チェック演算後メモリ 22d は、図 9 における受信バッファ 15 に相当するもので、CRC チェック OK となり、且つ、 S_{min} となったときのクラス A の復号データ（音声符号）を記憶する。又、トレースバック部は BTFD 処理により決定されたクラス B, C のビット数に対応する時点 k における 4 本のパスのうちパスメトリック値が最小のパスを求め、該パスに沿ってトレースバック処理を行って該クラス B, C の音声符号を取得し、CRC チェック演算後メモリ 22d に記憶する。

BTFD 制御部 22e は、図 21 のフローにしたがって BTFD 処理を行って各クラスの音声符号のビット数を決定するものであり、候補レート設定 / 保持部 34 は上位レイヤ（上位アプリケーション）41 より通知される複数のビットレート候補を保持すると共に、受信データメモリ部 22a に所定のビットレートを設定する。

上位レイヤ（上位アプリケーション）41 から、あらかじめ、音声の候補レート情報（図 10 に示すクラス A, B, C の bit 長）が BTFD 制御部 22e に通知され、候補レート設定 / 保持部 34 は通知された候補レート情報（ビットレート組み合わせ情報）を保持する。

一方、図示しない分離部で分離された各クラスの受信データは受信データメモリ部 22a に保持され、BTFD 制御部 22e の候補レート設定 / 保持部 34 は複数のビットレート組み合わせ候補におけるクラス A のビットレートを小さい順に受信データメモリ部 22a に設定する。

ビットレートを受け取った受信データメモリ部 22a は、該ビットレートに応じたビット数のクラス A の受信データをビタビ復号部の ACS 演算 / パスメトリックメモリ / パスメモリ部 31 に入力する。ACS 演算 / パスメトリックメモリ / パスメモリ部 31 は ACS 演算を行い、その演算結果であるパスメトリック値を内蔵のパスメトリックメモリに保持するとともに、最大パスメトリック値 $a_{\text{max}}(n_{\text{end}})$ 、最少パスメトリック値 $a_{\text{min}}(n_{\text{end}})$ 、状態 a （状態 0）のパスメトリック値 $a_0(n_{\text{end}})$ をそれぞれ BTFD 制御部 22e に通知する。

BTFD 制御部 22e は、 $S(n_{\text{end}})$ の計算を行い、さらに、 $S(n_{\text{end}})$ と閾値 D との比較を行う。その結果、トレースバックを行うと判断すれば、トレースバック部 32 に対して

10

20

30

40

50

、トレースバック起動情報を入力する。これによりトレースバック部 3 2 はトレースバックを行い、 n_{end} 位置までの復号結果をトレースバック後メモリ部 3 3 に保存する。

しかる後、受信データメモリ部 2 2 a は、最大レートの組み合わせ（図 1 0 の例ではフレームタイプ 1 1 1 0）におけるクラス B, C のビットレートに応じたビット数のクラス B、クラス C 受信データをビタビ復号部 2 2 b に入力し、ビタビ復号部 2 2 b はビタビ復号処理を行い、得られたパスをパスメモリ部に保存する。すなわち、前述のように、各データ長候補に対応する時点 k において、状態 a, b, c, d にそれぞれ至るパスメトリック値が最小の 4 本のパスをパスメモリに記憶し、それぞれのパスメトリック値をパスメトリックメモリに記憶する。

クラス A の復号が終了すれば、トレースバック後メモリ部 3 3 は CRC 演算部 2 2 c に BTFD 制御部 2 2 e から通知されたビットレート組み合わせ情報が指示するクラス A のビット数 + CRC チェック符号数分の復号結果を入力する。CRC 演算部 2 2 c は CRC チェック演算を行い、その CRC チェック結果を BTFD 制御部 2 2 e に通知する。BTFD 制御部 2 2 e は、その CRC チェック結果をもとに、図 2 1 の処理フローにしたがって S_{min} と $S(n_{end})$ の大小を比較し、 $S_{min} > S(n_{end})$ であれば、 S_{min} 、 n_{end}' 値の更新を行う。又、CRC チェックが OK で、かつ、 $S_{min} > S(n_{end})$ の条件が成立すれば CRC チェック符号を削除した復号結果をそれまでの復号結果に替えて CRC 演算後メモリ 2 2 d に保存する。以後、BTFD 制御部 2 2 e は、クラス A について上記処理をビットレート組み合わせ候補分を行えば、その時の n_{end}' 値に応じたビット数をクラス A の音声符号のビット数であると認識する。なお、 n_{end}' 値に応じたビット数のクラス A の音声符号データは既に CRC 演算後メモリ 2 2 d に格納されている。

ついで、BTFD 制御部 2 2 e は、ビットレート組み合わせ候補テーブル（図 1 0）とクラス A のビット数とからクラス B, C の音声符号のビット数を識別する。以後、BTFD 制御部 2 2 e は、トレースバック部 3 2 を起動し、該トレースバック部はパスメモリ部から、クラス B, C のビット長に相当する時点 k における 4 本のパスのうちパスメトリック値が最小のパスを求め、該パスに沿ってトレースバック処理を行って該クラス B, C の音声符号を取得し CRC 演算部 2 2 c 経由で CRC 演算後メモリ 2 2 d に格納する。CRC 演算後メモリ 2 2 d はクラス A, B, C の音声符号がそろったところで、これら音声符号を音声コーデック部 1 に 1 論理チャネルとして送り、音声コーデック部 1 は受信した音声符号より音声信号を復元する。

3 G P P 規格にのっとった従来の BTFD 処理では、図 2 1 の処理フローより明らかなように、ビットレートすなわちデータ長の短い順にビタビ復号処理、CRC チェック演算を行い、最終的に全てのビットレート組み合わせ候補について処理した後に、最も確からしい各クラスの音声符号のデータ長を決定し、受信データより各クラスの音声符号データを切り取って音声コーデック部に入力する。しかし、かかる従来方法では、毎回全てのレート候補について演算するので、処理量が非常に多くなり、消費電流も増加する問題がある。

従来の BTFD 処理シーケンスでは、真の音声符号と異なるデータ長においても CRC チェック OK が生じることを想定しているため、全ての候補について検出処理を行って特性上良い方に基いて各クラスの音声符号のビット長を決定している。しかし、本来送信している真のデータ長以外で、CRC チェック OK が生じる確率は、CRC - SIZE = 12bit として、 2^{-12} であり、この処理による特性の改善は、微小であると考えざるを得ない。

また、ビットレート組み合わせ候補が無音用、背景雑音用、12.2k 音声データ用の 3 つの場合を考えると、通話中は 12.2k 音声データがある程度続く。それにも拘わらず、従来のシーケンスでは、毎回、無音、背景雑音、12.2k 音声データの順（レートが小さい順）にビタビ復号処理、CRC チェック演算を行い、結局 12.2k 音声データが正しいことを検出して受信するため、処理量が多く消費電流も増加する問題がある。

また、従来の BTFD 処理シーケンスでは、ビタビ復号、トレースバック処理、CRC 演算を終えた各クラスの復号結果を一旦 CRC 演算後メモリ 2 2 d に格納する。このように従来の BTFD 処理シーケンスでは、トレースバック後メモリ部と CRC 演算後メモリの両方が必要になるため使用するメモリ数が非常に多くなる問題がある。

10

20

30

40

50

【発明の概要】**【発明が解決しようとする課題】****【0003】**

以上より本発明の目的は、BTFD処理の処理量及び消費電流を減少でき、しかも、これらを減少しても正しく各クラスの音声符号のビット数を識別し、各クラスの音声符号を正しく分離して音声コーデック部に入力できるようにすることである。

本発明の別の目的は、BTFD処理に使用するメモリ数を減少でき、しかも、使用メモリ数を減少しても正しく各クラスの音声符号のビット数を識別できるようにすることである。

【課題を解決するための手段】**【0004】**

10

上記課題は、本発明によれば、誤り検出用データを含む可変長のデータで変調された信号を受信し、復調する受信手段と、複数のデータ長の候補を順に選択し、復調した信号について該選択したデータ長分だけ抽出して、前記誤り検出用データを用いて誤り検出を行う誤り検出手段と、該誤り検出手段により誤りを検出しなかった場合のデータ長の候補を前記可変長のデータ長として判定する判定手段とを備え、前記判定手段は、前記誤り検出手段により誤りを検出しなかった最初のデータ長の候補を前記可変長のデータ長として判定する受信装置ことにより達成される。

前記可変長のデータは、通話音声に基いて順次生成された、異なるデータ長を有する有音データ、無音データ、背景雑音データのいずれかと、誤り検出用データとを含むデータからなり、前記誤り検出手段は、前記判定手段により判定した可変長のデータ長の候補を後続して受信した次の可変長のデータについて最初に選択するデータ長の候補とする。

20

前記可変長のデータは、通話音声に基いて所定の規則で順次生成された、異なるデータ長を有する有音データ、無音データ、背景雑音データのいずれかと、誤り検出用データとを含むデータからなり、前記誤り検出手段は、前記判定手段により判定した可変長のデータ長の候補の履歴と前記所定の規則とを参照して、次の可変長のデータについて選択するデータ長の候補の順番を決定する。

【発明の効果】**【0005】**

本発明によれば、BTFD処理の処理量及び消費電流を減少でき、しかも、これらを減少しても正しく各クラスの音声符号のビット数を識別し、各クラスの音声符号を正しく分離して音声コーデック部に入力できる。

30

又、本発明によれば、BTFD処理に使用するメモリ数を減少でき、しかも、使用メモリ数を減少しても正しく各クラスの音声符号のビット数を識別できる。

又、本発明によれば、各クラスの音声符号のビット数が送信時間間隔TTI毎に変化しても各クラスのデータを記憶するメモリの先頭アドレスを一定にして先頭アドレスの制御を容易にすることができる。

【図面の簡単な説明】**【0006】**

【図1】本発明のW-CDMAシステムにおける移動端末の概略構成図である。

【図2】送信側チャンネルコーデック部及び受信側チャンネルコーデック部の処理説明図である。

40

【図3】送信側チャンネルコーデック部における各クラスのデータフォーマット及び多重データ構成説明図である。

【図4】受信側チャンネルコーデック部における各クラスの音声符号分離動作説明図である。

【図5】本発明のBTFD処理を実行する受信側チャンネルコーデック部及びBTFD制御部の構成図である。

【図6】本発明の第1実施例のBTFD処理フローである。

【図7】本発明の第2実施例のBTFD処理フローである。

【図8】本発明の第3実施例のBTFD処理フローである。

50

【図 9】従来の移動局の構成図である。

【図 10】3GPP規格における音声フォーマット（クラスA～Cのビットレート）説明図表である。

【図 11】背景雑音フレームの挿入説明図である。

【図 12】復号処理された後におけるTTI = 20msの各クラスの水タ構成である。

【図 13】畳み込み符号器の例である。

【図 14】畳み込み符号器の入出力関係説明図である。

【図 15】畳み込み符号器の状態説明図である。

【図 16】畳み込み符号器の状態と入出力関係図である。

【図 17】格子状表現説明図である。

10

【図 18】畳み込み符号の復号化の説明図である。

【図 19】ビタビ復号説明図である。

【図 20】任意の時点kにおける各状態のエラ一最小パスの説明図である。

【図 21】従来のBTDF処理フローである。

【図 22】BTDF処理を実行する従来の受信側チャンネルコーデック部の構成図である。

【発明を実施するための形態】

【0007】

・発明の開示

本発明の第1は、BTDF処理においてCRCチェックOKを検出した時、それ以降のBTDF処理を停止し、CRCチェックOK時のビットレート組み合わせにおける各クラスのビットレートに基づいて各クラスの音声符号のビット数を決定し、該ビット数に基づいて受信データより各クラスの音声符号を分離して音声コーデック部に入力する。このようにすれば、BTDF処理の処理量及び消費電流を減少でき、しかも、これらを減少しても各クラスの音声符号のビット数を識別し、各クラスの音声符号を正しく分離して音声コーデック部に入力できる。

20

3GPP規格ではクラスAに付加するCRCチェック符号のサイズは12ビットと規定されており、このサイズのCRCチェックにより 2^{12} ビット中1ビット以上の誤りを検出することが可能である。図10の全てのビットレート組み合わせにおいて、クラスAのビットレートは0～81bitであり 2^{12} ビットと比較すると非常に小さい。このため、CRCチェックOKであったビットレートが駄目で、別のビットレートが正解となる確立は非常に小さい。

30

本発明の第2は、BTDF処理においてCRCチェックOKとなるまでビットレートの小さい順にビタビ復号、CRCチェック演算を行う。しかし、CRCチェックOKとなってクラスAのビットレートが確定した時は、そのときのビットレート組み合わせを送信時間間隔TTIの間保持し、次のTTI経過後のBTDF処理をこの記憶してあるビットレートの組み合わせにおけるクラスAのビットレートから開始する。このようにすれば、短時間で各クラスのビット長が判別する確率が大きく、BTDF処理の処理量及び消費電流を減少でき、しかも、これらを減少しても各クラスの音声符号のビット数を識別し、各クラスの音声符号を正しく分離して音声コーデック部に入力できる。

各クラスの音声符号のビット数は、送信側の音声コーデックにより決定される。ところで、音声通話であれば有音部分（12.2kbpsレート）と無音部分はそれぞれ数十秒単位で連続するのが普通であり、音声符号のTTI単位は20msである。このため、音声通話の有音部分／無音部分の変化の間隔は、TTI単位に対して長く、現TTIと次TTIでレートが変化する確率は、変化しない確率に比べて低いことになる。そのため、第2の発明は、ビットレートが確定したら次TTIでも同じビットレートでBTDF処理を行えば、その時点でCRCチェックOKとなり、ビットレートが確定する確率が高くなることを利用するものである。

40

本発明の第3は、無音状態をカウントし、過去に連続して7回の無音状態を検出すれば、次のTTIでは背景雑音に応じたビットレートからBTDF処理を開始する。これは3GPP音声コーデックの仕様によれば、 $8 \times \text{TTI}$ の無音状態において1つの背景雑音を挿入するように定義されているからである。このようにすれば、BTDF処理の処理量を減少できる。

本発明の第4では、BTDF処理により確定したフレームタイプ情報（各クラスのビットレ

50

ートの組み合わせ)を、BTFD制御部から音声コーデック部に通知し、音声コーデック部がこのフレームタイプ情報をもとに、クラスAの音声符号をCRC演算後メモリから読み出し、クラスB、Cの音声符号はトレースバック後メモリから読み出す。このようにすれば、CRC演算後メモリにクラスB、Cの音声符号を保持する必要がなくなり、使用メモリ数を減少できる。なお、CRC演算後メモリを削除し、音声コーデック部が通知されるフレームタイプ情報をもとに、クラスA、B、Cの音声符号をトレースバック後メモリから読み出すように構成することもできる。このようにすれば、CRC演算後メモリを削除できるためますます使用メモリ数を減少できる。

(A) W-CDMAシステムにおける移動端末の構成

図1は本発明のW-CDMAシステムにおける移動端末の概略構成図であり、入力音声信号を符号化すると共に、音声符号を音声信号に復元して出力する音声コーデック部51、音声符号に誤り訂正符号化処理(チャネルエンコード処理)を施して出力する送信側チャンネルコーデック部52、送信データを拡散変調する変調部53、拡散変調されたベースバンド信号を高周波信号に変換して送信アンテナANT_Tより送信する送信部54、受信アンテナANT_Rにより受信した受信信号をベースバンド信号に復調する受信部61、ベースバンド信号を逆拡散する復調部62、逆拡散された受信データに誤り訂正復号処理(チャネルデコード処理)を施し、復号された音声符号を音声コーデック部51に入力する受信側チャンネルコーデック部63、BTFD制御を行う制御プロセッサ(BTFD制御部)71を有している。

音声コーデック部51はAMR音声符号化方式により音声信号をTTI=20ms毎に符号化し、得られた音声符号をクラスA~Cの3つに分け、それぞれを所定のビットレートに応じたビット長で表現し、各クラスの音声符号を3つのトランスポートチャンネルTrCHで送信するように送信側チャンネルコーデック部52に inputs。呼接続時に基地局は発信端末及び着信端末にそれぞれ各クラスA~Cのビットレートの組み合わせ候補を複数(図10参照)通知すると共に、どの組み合わせのビットレートで音声符号を送信するのかを指示する。音声コーデック部51は該指示された組み合わせの各クラスのビットレートに応じたビット数で音声符号を表現してチャンネルコーデック部52に inputs。

送信側チャンネルコーデック部52は、クラスAの音声符号に一定長のCRCチェックコードを付加し、ついで、各クラスA~CのTTI=20msの音声符号にビタビ符号化処理を施して符号化し、レートマッチングしてからフレーム単位(10ms単位)に分割し、10ms毎に各クラスの誤り訂正符号化処理が施された音声符号データをクラスAを先頭にして多重し、変調部53、送信部54を介して送信する。

受信チャンネルコーデック部63は受信部61、復調部62を介して入力する多重データより、フレーム毎(10ms毎)にクラスA~Cのデータを所定間隔で分離すると共に、各クラスの10msの受信データを2つ結合して送信時間間隔TTI=20msのデータにし、しかる後、クラスA~Cのデータに誤り訂正復号処理を施して元のクラスA~Cの音声符号データを復号し、音声コーデック部51に inputs。この際、受信チャンネルコーデック部63は各クラスの所定ビット長の音声符号データを正しく分離して音声コーデック部に inputsする必要がある。このため、BTFD制御部71はBTFD処理により各クラスの音声符号データのビット長を識別し、受信チャンネルコーデック部63は識別されたビット長の各クラスの音声符号データを復号結果から切り取って音声コーデック部51に inputs。なお、BTFD処理において受信チャンネルコーデック部63はクラスAの復号データに対してCRCチェック演算を行い、BTFD制御部71はCRCチェック結果に基づいて各クラスの音声符号データのビット長を識別する。

(B) 送信側/受信側チャンネルコーデック部の処理

図2は送信側チャンネルコーデック部52及び受信側チャンネルコーデック部63の処理説明図である。音声コーデック部51からの各クラスA~Cの音声符号データは、送信側チャンネルコーデック部52に inputs。送信側チャンネルコーデック部52はクラスAの音声符号データに、CRCチェックビットを付加し(ステップ201)、ついで、各クラスの音声符号データにビタビ符号化処理を施す(ステップ202)。しかる後、送信側チャンネルコーデック部52は、符号化した各クラスのデータに1stインタリーブ処理を行ない(ステッ

ブ 2 0 3)、その後レートマッチング操作 (各 TrCH の許容レートに収まるようにデータの伸縮操作)を行なう (ステップ 2 0 4)。送信側チャネルコーデック 5 2 はレートマッチング処理を行なった各クラスの符号化データを 10ms 毎に分割し、10ms 毎に各クラスの符号化データをクラス A が先頭となる順番で多重化 (ステップ 2 0 5)、この多重データに 2 n d インタリーブ処理を施して (ステップ 2 0 6)、変調部 (MO 部) 5 3 へ入力する。変調部 5 3 は QPSK 拡散および QPSK 変調を行い、送信部 5 4 は RF 信号に変換してアンテナより送信する。尚、レートマッチングは伸縮率に制限があり、許容レートに満たない場合は、その部分に所定のデータが挿入され、そのデータの挿入区間では送信を行わないこととする。図 1 2 の A 3 のエンブティ期間に対応する。

受信に際して、受信部 6 1 は RF 信号からベースバンド信号に変換し、復調部 (DEM 部) 6 2 は QPSK 復調および QPSK 逆拡散 / レーク合成して受信データを復調し、復調データを受信側チャネルコーデック部 6 3 に入力する。受信側チャネルコーデック部 6 3 は、受信した多重データに 2 n d デインタリーブ処理を施し (ステップ 2 0 7)、ついで、10ms データ毎に受信した多重データをクラス A ~ C 毎に分離した 10ms データをクラス毎に結合して TTI = 20ms のデータを生成する (ステップ 2 0 8)。以後、受信側チャネルコーデック部 6 3 は、クラス毎に受信データに 1 s t デインタリーブ処理を行ない (ステップ 2 0 9)、ビタビ復号処理を施して復号し (ステップ 2 1 1)、復号後、クラス A の復号データに CRC チェック処理を施し、BTFD 制御にしたがって該 CRC チェック結果に基づいて各クラスの音声符号データのビット長を識別し (ステップ 2 1 2)、該ビット長に基いて各クラスの音声符号を音声コーデック部 5 1 に入力する。

(C) 各クラスのデータフォーマット及び多重 / 分離動作

図 3 は送信側チャネルコーデック部における各クラスのデータフォーマット及び多重データ構成説明図である。音声コーデック部 5 1 は AMR 音声符号化方式により音声信号を TTI = 20ms 毎に符号化し、得られた音声符号をクラス A ~ C の 3 つに分け、それぞれを所定のビットレート、例えば、図 1 0 に示す最大レートのフレームタイプ 1 1 1 0 に応じたビット長 (クラス A : 8 1 ビット、クラス B : 1 0 3 ビット、クラス C : 6 0 ビット) で表現し、各クラスの音声符号を送信側チャネルコーデック部 5 2 に入力する ((a) 参照)。

送信側チャネルコーデック部 5 2 は、クラス A の音声符号に一定長、例えば 12 ビットの CRC チェックコードを付加し ((a))、ついで、TTI = 20ms の各クラスの音声符号にビタビ符号化処理、インタリーブ処理、レートマッチング処理を施して符号化データを発生する (b)、(c) 参照)。畳み込み符号化処理、レートマッチング処理により各クラスのビット長は 8 1'、1 0 3'、6 0' に増大する。畳み込み符号の符号化率を n とすれば各クラスの符号化データの長さは元の長さの n 倍 + α (α はレートマッチングにより増減する量である) となる。

TTI = 20ms の各クラスの符号化データは、実際には 10ms 毎に分割され、10ms 毎にクラス A を先頭にして多重するが、分割しないものとすれば、(d) に示すように各クラスの符号化データを隙間なく多重する。すなわち、最大レートの組み合わせ候補に応じたビット数で各クラスの音声符号を表現する場合には、隙間なく各クラスの符号化データを多重する。尚、最大レートの組み合わせ候補における各クラスの符号化データ長 8 1'、1 0 3'、6 0' は既知の値である。最大レート以外のレートの組み合わせ候補、例えばフレームタイプ 0 1 1 0 の組み合わせに応じたビット数 (クラス A : 6 5 ビット、クラス B : 9 9 ビット、クラス C : 4 0 0 ビット) で各クラスの音声符号を表現する場合、各クラスの符号化データ長 6 5'、9 9'、4 0' は最大符号化データ長 8 1'、1 0 3'、6 0' に満たない。かかる場合は、(e) の斜線で示すように満たない分送信しない区間を示すデータを挿入する。このようにすることにより、受信側において各クラスのデータを容易に分離することが可能になる。ところで、実際には、前述のように、TTI = 20ms の各クラスの符号化データは、10ms 毎に前半と後半に分割され、10ms 毎にクラス A を先頭にして多重する。従って、(e) に示すフレームタイプ 0 1 1 0 の組み合わせの場合、(f) に示すように各クラス A ~ C の符号化データは 10ms の前半と後半に分割され、(g) に示すように 10ms 毎にクラス A を先頭にして多重されて送信される。

10

20

30

40

50

図4は受信側チャンネルコーデック部における各クラスの音声符号分離動作説明図である。受信チャンネルコーデック部63は、受信部61、復調部62を介して入力する多重データ((a)参照)より10ms毎にクラスA~Cのデータを既知の所定長で分離し、(b)に示すように各クラスの10msの受信データを2つ結合して送信時間間隔TTI=20msのデータを生成する。しかる後、受信側チャンネルコーデック部63はクラスA~Cのデータにビタビ復号処理を施して元のクラスA~Cの音声符号データを復号し、音声コーデック51に入力する((c)、(d)参照)。この際、BTFD処理により各クラスの音声符号データのビット長を識別し、識別されたビット長に基いて雑音等によるデータ部分(斜線部分)を除去した各クラスの音声符号データを復号結果から取り出して音声コーデック部51に入力する。

10

(D) BTFD処理を実行する受信側チャンネルコーデック部の構成

図5は本発明のBTFD処理を実行する受信側チャンネルコーデック部及びBTFD制御部の構成図である。

受信側チャンネルコーデック部63の復号前処理部63aは、図2で説明した2ndインターリーブ処理207、クラス毎の分割&結合処理208、1stデインターリーブ処理209を実行する。受信データメモリ部63bは、復号前処理部63aから出力するTTI=20msの各クラスのデータを保存する。この場合、受信データメモリ部63bは、図4(b)に示すように既知の最大レートに応じた符号化データ長に満たない部分(斜線部)にはパディングデータが埋め込まれた状態で各クラスのTTI=20msのデータを記憶する。

ビタビ復号部63cは、ACS演算/パスメトリックメモリ/パスメモリ部63c-1、トレースバック部63c-2、トレースバック後メモリ63c-3を備え、Aクラスについては n_{end} 位置までのデータ部分に対してビタビ復号処理を施し、B、Cクラスについてはデータ全体にビタビ復号処理を施し、各時点におけるパスをパスメモリに保存する。すなわち、B、Cクラスについてはデータ長が判らないため末尾位置までのビタビ復号処理を行い、各データ長候補に対応する時点kにおいて、状態a、b、c、dにそれぞれ至るパスメトリック値が最小の4本のパスをパスメモリに記憶し、それぞれのパスメトリック値をパスメトリックメモリに記憶する。

20

CRC演算部63dはAクラスの復号結果に基づいてCRCチェック演算を行う。CRCチェック演算後メモリ63eは、CRCチェックOKとなり、且つ、BTFD処理においてSminとなったときのクラスAの復号データ(音声符号)を記憶する。

30

BTFD制御部71は後述するフローにしたがってBTFD処理を行って各クラスの音声符号のビット数を決定するもので、候補レート設定/保持部72、CRCチェックOK検出&レート決定制御部73、確定レート保持&レート決定制御部74、無音フレームカウント&レート決定制御部75、フレームタイプ通知手段76を有している。

候補レート設定/保持部72は上位レイヤ(上位アプリケーション)81より通知される複数のビットレート候補(図10参照)を保持すると共に、受信データメモリ部63bにクラスAのビットレートを小さい順に順次設定する。

CRCチェックOK検出&レート決定制御部73は、ビットレート組み合わせ候補を順次変更してクラスAのビット長を小さい順に切り替え、CRCチェックOKを検出した時のビットレート組み合わせ候補に基いて各クラスのビット長を決定する。

40

確定レート保持&レート決定制御部74は、CRCチェックOKが検出されて各クラスのビットレートが確定したとき、その時のビットレート組み合わせ候補を1送信時間間隔TTIの間保持し、TTI時間経過後の次のビットレート決定処理において、その記憶したビットレート組み合わせ候補におけるクラスAのビットレートでBTFD処理を開始して各クラスのビット長を決定する。

無音フレームカウント&レート決定制御部75は、無音フレームを検出した場合に無音フレーム連続数をカウントし、7回に達したら背景雑音に応じたビットレート組み合わせ候補のクラスAのビットレートでBTFD処理を開始して各クラスのビット長を決定する。

フレームタイプ通知手段76はBTFD処理により各クラスのビットレートを決定したとき、その時のビットレート組み合わせ候補を特定するフレームタイプ情報を音声コーデック

50

部 5 1 に通知する。これにより、音声コーデック部 5 1 は、該フレームタイプ情報、すなわち、各クラスのビット長をトレースバック部 6 3 c - 2 に入力して該トレースバック部を起動する。トレースバック部 6 3 c - 2 はパスメモリ部から、クラス B, C のビット長に相当する時点 k における 4 本のパスのうちパスメトリック値が最小のパスを求め、該パスに沿ってトレースバック処理を行って該クラス B, C の音声符号を取得しCRC演算部 2 2 c 経由でCRC演算後メモリ 2 2 d に格納する。音声コーデック部 5 1 は、CRC演算後メモリ 2 2 d にクラス A, B, C の音声符号がそろったところで、これら音声符号を取り込んで音声符号より音声信号を復元する。

(E) 本発明のBTFD処理の第 1 実施例

図 6 は本発明の第 1 実施例のBTFD処理フローである。第 1 実施例のBTFD処理ではCRCチェックOKを検出した時、それ以降のBTFD処理を停止し、CRCチェックOK時のビットレート組み合わせにおける各クラスのビットレートに基づいて各クラスの音声符号のビット数を決定し、該ビット数に基づいて受信データより各クラスの音声符号を抽出して音声コーデック部に入力する。

BTFD制御部 7 1 は、上位アプリケーション 8 1 よりビットレートの候補 (図 10) が指定されるから、クラス A のビットレート (単位時間当たりのビット数) の小さい順に番号付けし、 $n_{end} = 1, 2, 3, \dots$ とする (ステップ 301)。ついで、 $n_{end} = 1$ 、 $S_{min} = D$ 、 $n_{end}' = 0$ と初期化する (ステップ 302)。

しかる後、 n_{end} 位置までACS処理を行い (ステップ 303)、(1) 式により $S(n_{end})$ を計算する (ステップ 304)。 $S(n_{end})$ が求めれば、この $S(n_{end})$ と閾値 D との大小判断を行う (ステップ 305)。

$S(n_{end}) \leq D$ であれば、 n_{end} 位置のパスメトリック値が最小の状態から、トレースバック処理を行う (ステップ 306)。ついで、トレースバック処理により得られた復号データに対してCRCチェックを行う (ステップ 307)。CRCチェックがOKであれば (ステップ 308)、そのときの n_{end} が最も確からしいビットレートの組み合わせにおけるクラス A のビットレート (ビット数) と判定し、出力する (ステップ 309)。以後、図 10 のビットレートの組み合わせ候補を参照して他のクラスのビットレート (ビット数) を求める。

一方、ステップ 305 において、 $S(n_{end}) > D$ であれば、あるいは、ステップ 308 において、CRCチェックがNOであれば、ステップ 310 の処理を行う。ステップ 310 では全候補の n_{end} について上記処理を行ったかチェックする (ステップ 310)。

CRCチェックOKとなることなく全候補の n_{end} について上記処理を繰り返せば、ステップ 310 において「YES」となるからエラー出力する (ステップ 311)。しかし、全候補の n_{end} について上記処理が終了してなければ n_{end} を歩進し (ステップ 312)、ステップ 303 の処理を繰り返す。

以上のBTFD処理を考慮して受信動作を説明する。

上位レイヤ (上位アプリケーション) 8 1 から、あらかじめ、各クラスのビットレートの組み合わせ候補が複数 (図 10 参照)、BTFD制御部 7 1 に通知され、候補レート設定 / 保持部 7 2 はこの通知された候補レート情報を保持する。かかる状態において、受信データメモリ部 6 3 b は、復号前処理部 6 3 a で分離結合された各クラスのTTI = 20msのデータを記憶する。BTFD制御部 7 1 の候補レート設定 / 保持部 7 2 は、複数の候補レートの中からレートの小さい順にクラス A のビットレートを受信データメモリ部 6 3 b に設定する。

受信データメモリ部 6 3 b は、受信したクラス A について設定レートに応じたビット数分順にビタビ復号部のACS演算 / パスメトリックメモリ / パスメモリ部 6 3 c - 1 に入力する。ACS演算 / パスメトリックメモリ / パスメモリ部 6 3 c - 1 はACS演算を行い、その演算結果であるパスメトリック値を内蔵のパスメモリに保持するとともに、最大パスメトリック値 $a_{max}(n_{end})$ 、最少パスメトリック値 $a_{min}(n_{end})$ 、状態 a (= 状態 0) のパスメトリック値 $a_0(n_{end})$ をそれぞれBTFD制御部 7 1 に通知する。

BTFD制御部 7 1 は、 $S(n_{end})$ の計算を行い、さらに、 $S(n_{end})$ と閾値 D との比較を

行う。その結果、トレースバックを行うと判断すれば、すなわち、 $S(n_{end})$ Dであれば、トレースバック部 63c-2 に対して、トレースバック起動情報を入力する。これによりトレースバック部 63c-2 に n_{end} 位置までの復号結果を保存する。

しかる後、あるいは、クラス A のビタビ復号処理前に、受信データメモリ部 63b は、クラス B, C の受信データをビタビ復号部 63c に入力し、ビタビ復号部 63c は復号結果をパスメモリ部に保存する。すなわち、ビタビ復号部 63c は B, C クラスについてはデータ長が判らないため末尾位置までのビタビ復号処理を行い、各データ長候補に対応する時点 k において、状態 a, b, c, d にそれぞれ至るパスメトリック値が最小の 4 本のパスをパスメモリに記憶し、それぞれのパスメトリック値をパスメトリックメモリに記憶する。

クラス A の復号処理が終了すれば、トレースバック部 63c-2 は CRC 演算部 63d に n_{end} 値に応じたビット数 (クラス A のビット数 N_A + CRC チェック符号数) 分の復号結果を入力する。CRC 演算部 63d は CRC チェック演算を行い、CRC チェックが OK であれば、CRC チェック符号を削除した復号結果を CRC 演算後メモリ 63e に保存し、その時の n_{end} 値に応じたクラス A のビット数 (N_A) がクラス A の真の音声符号のビット数であると認識する。

ついで、BTFD 制御部 71 は、ビットレート組み合わせ候補テーブル (図 10) とクラス A のビット数とからビットレート組み合わせ候補を識別し、音声コーデック部 51 に該ビットレート組み合わせ候補を特定するフレームタイプ情報を通知する。音声コーデック部 51 は通知されたビットレート組み合わせ候補より各クラスの音声符号ビット数を識別し、該音声符号ビット数をトレースバック部 63c-2 に入力して該トレースバック部を起動する。トレースバック部 63c-2 はパスメモリ部から、クラス B, C のビット長に相当する時点 k における 4 本のパスのうちパスメトリック値が最小のパスを求め、該パスに沿ってトレースバック処理を行って該クラス B, C の音声符号を取得し CRC 演算部 22c 経由で CRC 演算後メモリ 22d に格納する。音声コーデック部 51 は、CRC 演算後メモリ 22d にクラス A, B, C の音声符号がそろったところで、これら音声符号を取り込んで音声符号より音声信号を復元する。

以上、第 1 実施例によれば、CRC チェック OK の時点で BTFD 処理を終了できるため、BTFD 処理の処理量を減少でき、結果的に消費電流を減少でき、しかも、これらを減少しても各クラスの音声符号のビット数を識別し、各クラスの音声符号を正しく分離して音声コーデック部に入力できる。

(E) 本発明の BTFD 処理の第 2 実施例

図 7 は本発明の第 2 実施例の BTFD 処理フローであり、第 1 実施例の処理フローと同一処理には同一ステップ番号を付している。

第 2 実施例では、CRC チェック OK となってクラス A のビットレートが確定した時のビットレート組み合わせ候補を送信時間間隔 TTI の間保持しておき、次の TTI 経過後の BTFD 処理を該記憶してあるビットレートの組み合わせ候補におけるクラス A のビットレートから開始する。

通常、音声通話であれば有音部分 (12.2kbps レート) と無音部分はそれぞれ数十秒単位で連続するのが普通であり、音声符号の TTI 単位は 20ms である。このため、音声通話の有音部分 / 無音部分の変化の間隔は、TTI 単位に対して長く、現 TTI と次 TTI でのレートが変化する確率は、変化しない確率に比べて低いことになる。そのため、第 2 実施例は、ビットレートが確定したら次 TTI でも同じビットレートで BTFD 処理を行えば、その時点で CRC チェック OK となり、ビットレートが確定する確率が高くなることを利用するものである。

第 1 実施例の処理フローと異なる点は、各クラスのビットレートが確定したときにセットされるレート確定フラグを導入し、このレート確定フラグを用いて BTFD 処理を行う点である。以下では、第 1 実施例と異なる処理を重点的に説明する。

まず、音声受信を開始する前にレート確定フラグをオフする。ついで、音声受信を開始し、レート確定フラグがオンかオフかを判別する (ステップ 401)。初回はオフなので、第 1 実施例と同様にレートが小さい順から BTFD 処理を行う (ステップ 301 ~ 309)

10

20

30

40

50

。ステップ309において、クラスAのビットレートが確定すれば、レート確定フラグをオンし(ステップ412)、その時のクラスAのビットレート(TTI単位のビット長)、すなわち、 n_{end} を次TTIまで保持する(ステップ413)。この結果、次TTIでは、ステップ411においてレート確定フラグがオンとなっているため、前TTI時に保持した n_{end} からACS演算を開始する。

一方、ステップ305において $S(n_{end}) > D$ となり確からしくないデータと判断した場合、レート確定フラグがオンであるかチェックし(ステップ414)、レート確定フラグがオンであればレート確定フラグをオフし(ステップ415)、ステップ411に戻り、以降の処理を繰り返す。又、ステップ411において、レート確定フラグがオフであれば、ステップ310の処理を行う。すなわち、ステップ310において全候補の n_{end} について上記処理を行ったかチェックする(ステップ310)。CRCチェックOKとなることなく全候補の n_{end} について上記処理を繰り返せば、ステップ310において「YES」となるからエラー出力する(ステップ311)。しかし、全候補の n_{end} について上記処理が終了してなければ n_{end} を歩進し(ステップ312)、ステップ411以降の処理を繰り返す。

また、ステップ308においてCRCチェックの結果OKでないと判明した場合、レート確定フラグがオンであるかチェックし(ステップ416)、レート確定フラグがオンであればレート確定フラグをオフし(ステップ415)、ステップ411に戻り、以降の処理を繰り返す。又、ステップ416において、レート確定フラグがオフであれば、ステップ310以降の処理を行う。

以上、第2実施例によれば、ビットレートが確定したら次TTIでも同じビットレートでBTDFD処理を行うようにしたから、少ないBTDFD処理で各クラスのビットレートを決定することができる。

(F) 本発明のBTDFD処理の第3実施例

図8は本発明の第3実施例のBTDFD処理フローであり、第1、第2実施例の処理フローと同一処理部分には同一ステップ番号を付している。第3実施例では、無音状態をカウントし、過去に連続して7回の無音状態を検出すれば、次のTTIでは背景雑音のビットレートからBTDFD処理を開始する。これは3GPP音声コーデックの仕様によれば、 $8 \times \text{TTI}$ の無音状態において1つの背景雑音を挿入するように定義されているからである。このようにすれば、BTDFD処理の処理量を減少できる。

音声受信を開始する前に無音フレーム数 U を0に初期化する。以後、音声受信を開始し、レート確定フラグがオフであれば(ステップ411)、第1実施例と同様にレートが小さい順からBTDFD処理を行う(ステップ301~309)。ステップ309において、クラスAのビットレートが確定すれば、レート確定フラグをオンし(ステップ412)、その時のクラスAのビットレート、すなわち、 n_{end} を次TTIまで保持する(ステップ413)。ついで、確定した n_{end} が無音のビットレートであるかどうかの判断し(ステップ501)、無音でなければ、次のTTIになるのを待つ。一方、無音であれば、無音状態数 U をカウントアップし($U = U + 1$, ステップ502)、次のTTIになるのを待つ。

次TTIにおいて、レート確定フラグがオンであれば(ステップ411)、無音フレーム数 U が7回に達したかチェックする(ステップ503)。7回に達してなければ、 n_{end} を前TTIで確定した n_{end} としてステップ303以降の処理をおこない、7回に達していれば n_{end} を背景雑音とする(ステップ504)。その際、無音フレーム数 U を0に初期化し(ステップ505)、レート確定フラグをオフにし(ステップ506)、ステップ303以降の処理を行う。 $U = 0$ にし、レート確定フラグをオフにしたことにより、次TTIからはレートが小さい順に、つまり、無音の n_{end} からBTDFD処理が開始することになる。

以上では、本発明を移動端末に適用した場合について説明したが、固定端末にも適用できることは勿論である。

(G) 発明の効果

以上本発明によれば、BTDFD処理の処理量及び消費電流を減少でき、しかも、これらを減少しても正しく各クラスの音声符号のビット数を識別し、各クラスの音声符号を正しく分離して音声コーデック部に入力できる。

10

20

30

40

50

又、本発明によれば、BTFD処理に使用するメモリ数を減少でき、しかも、使用メモリ数を減少しても正しく各クラスの音声符号のビット数を識別できる。

又、本発明によれば、各クラスの音声符号のビット数が送信時間間隔TTI毎に変化しても各クラスのデータを記憶するメモリの先頭アドレスを一定にして先頭アドレスの制御を容易にすることができる。

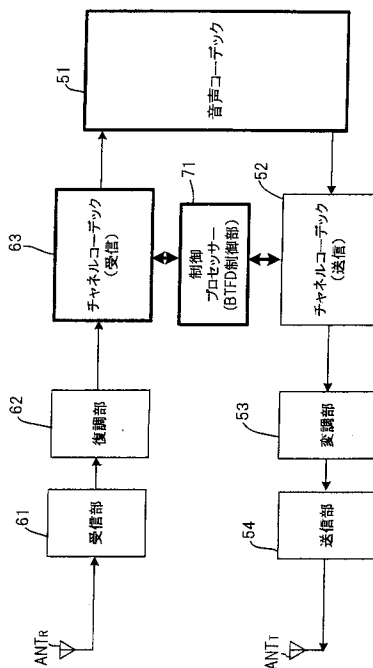
【符号の説明】

【0008】

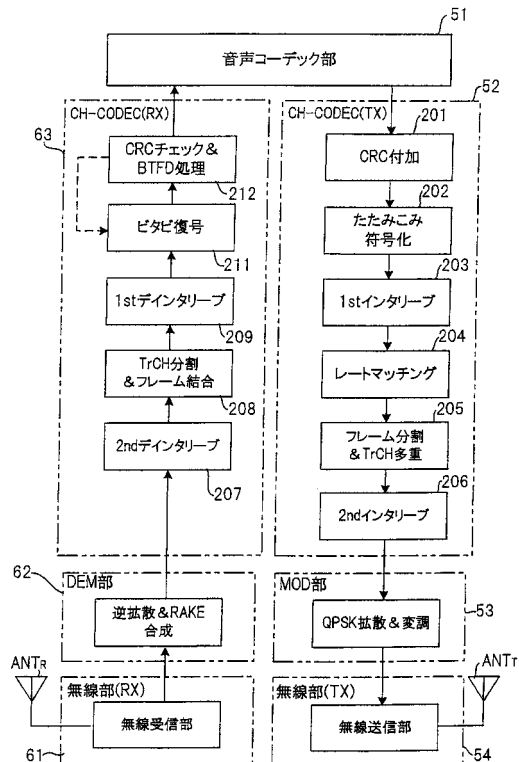
- 51 音声コーデック部
- 52 送信側チャンネルコーデック部
- 53 変調部
- 54 送信部
- 61 受信部
- 62 復調部
- 63 受信側チャンネルコーデック部
- 71 制御プロセッサ（BTFD制御部）

10

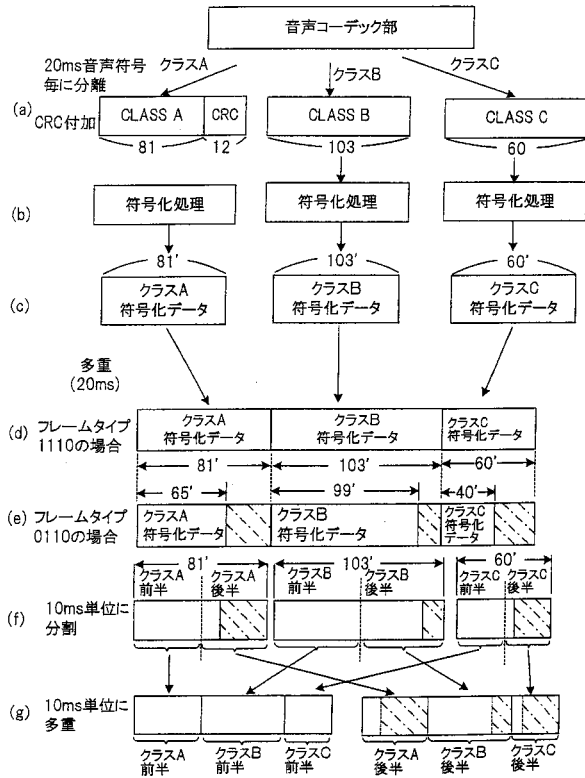
【図1】



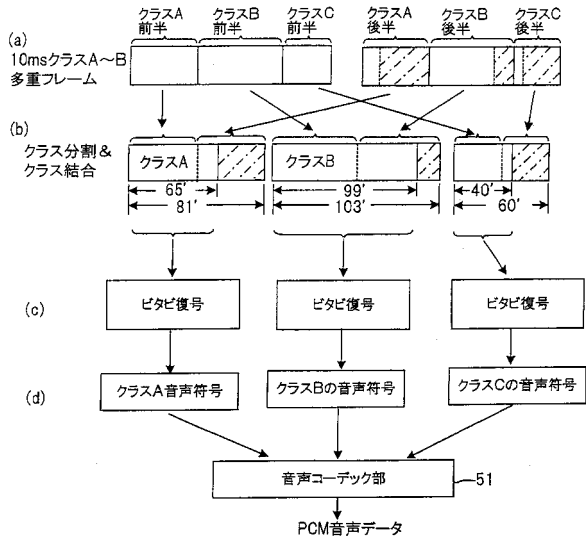
【図2】



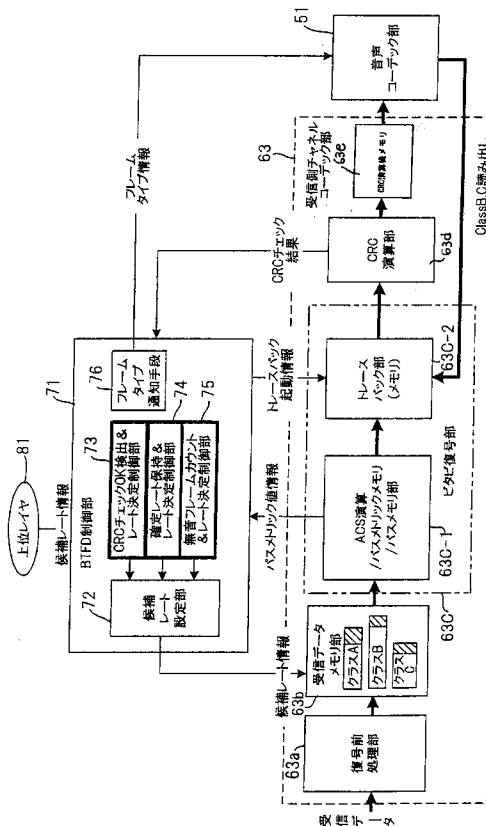
【図3】



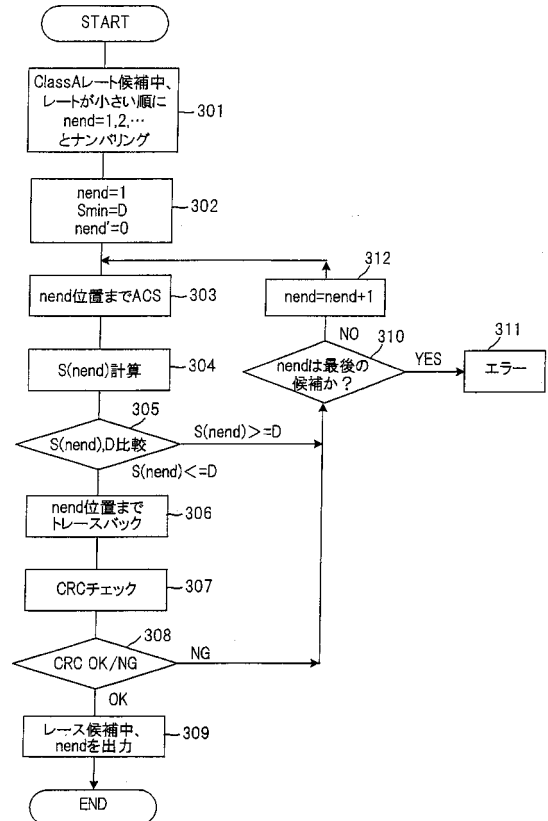
【図4】



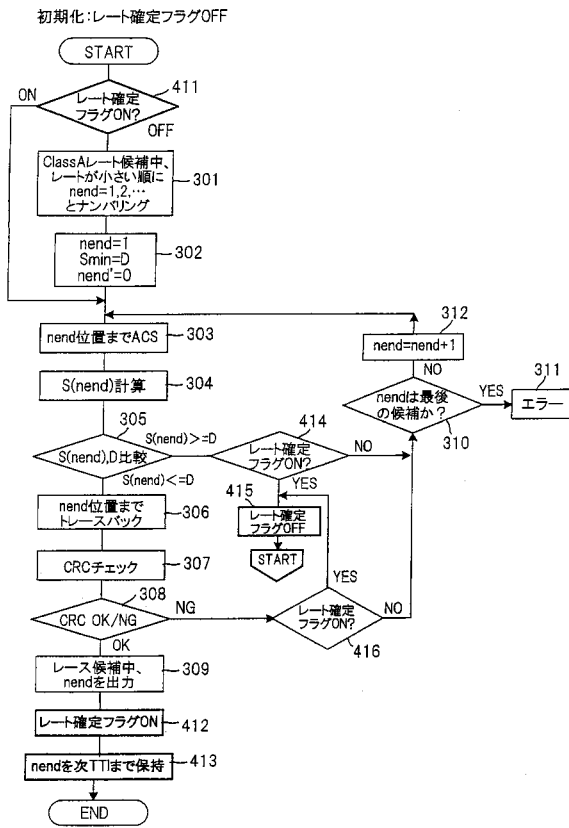
【図5】



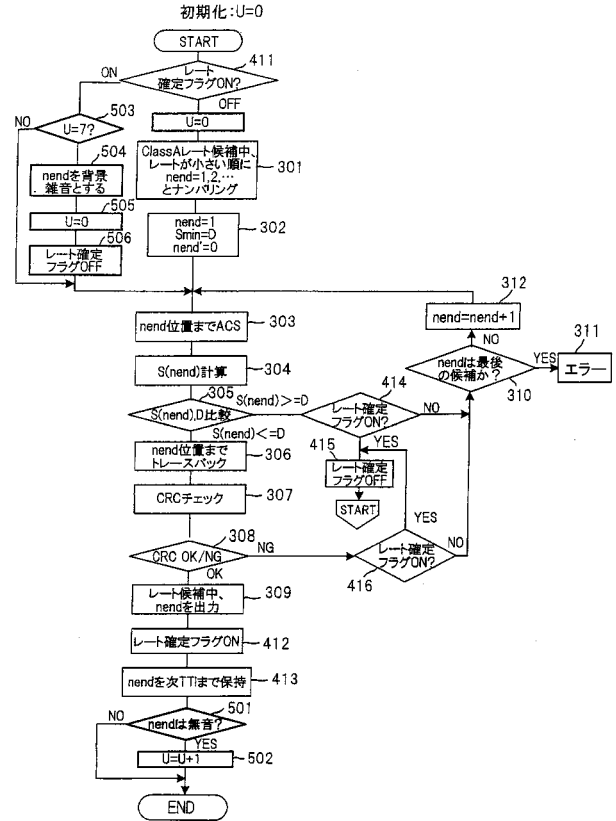
【図6】



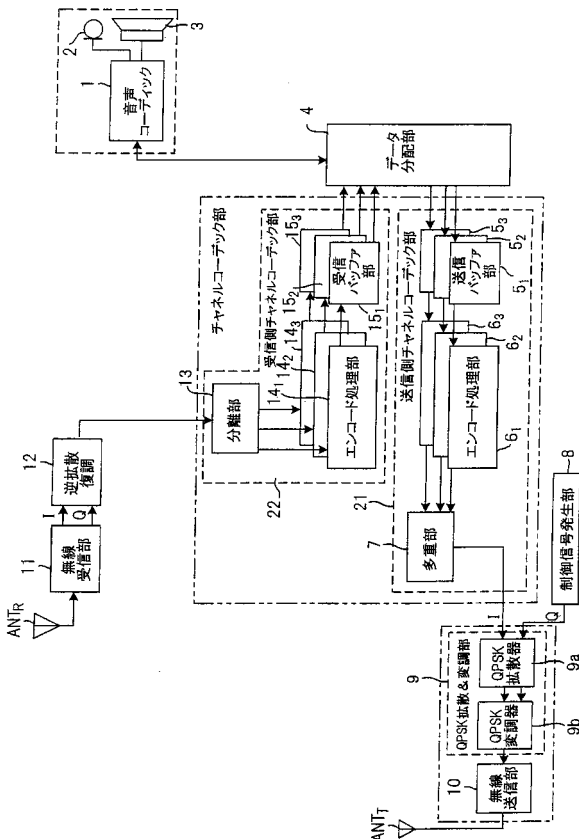
【図 7】



【図 8】



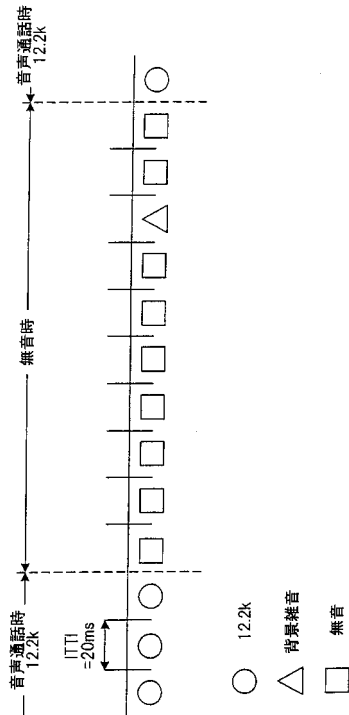
【図 9】



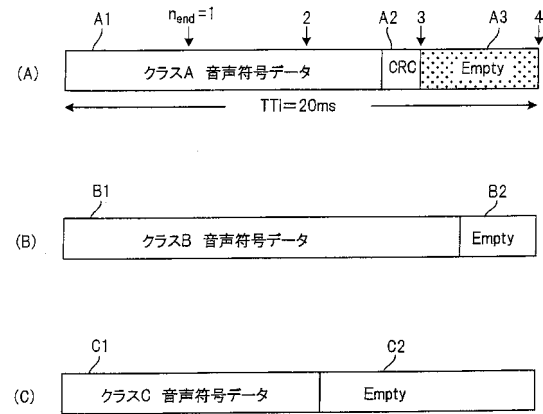
【図 10】

フレームタイプ	クラスA (bit)	クラスB (bit)	クラスC (bit)	備考
0000	42	53	0	4.75Kbps
1000	49	54	0	5.15Kbps
0100	55	63	0	5.90Kbps
1100	58	76	0	6.70Kbps
0010	61	87	0	7.40Kbps
1010	75	84	0	7.95Kbps
0110	65	99	40	10.2Kbps
1110	81	103	60	12.2Kbps
0001	39	0	0	GSM_Amr_Comfort_Noise_frame(背景雑音)
1111	0	0	0	No Transmission/Reception(無音)

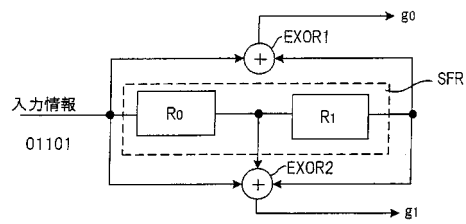
【図 1 1】



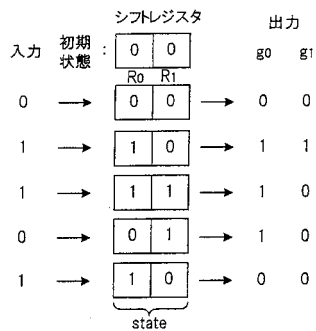
【図 1 2】



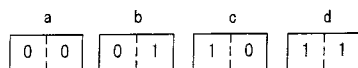
【図 1 3】



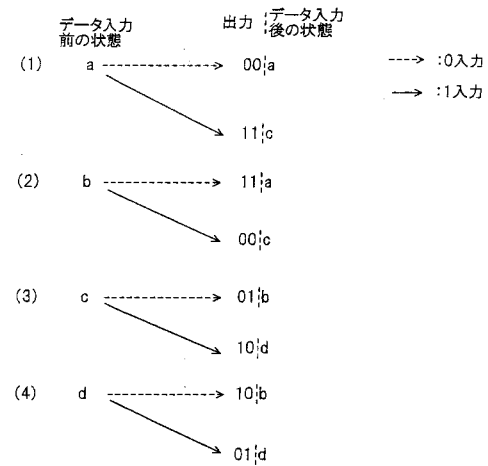
【図 1 4】



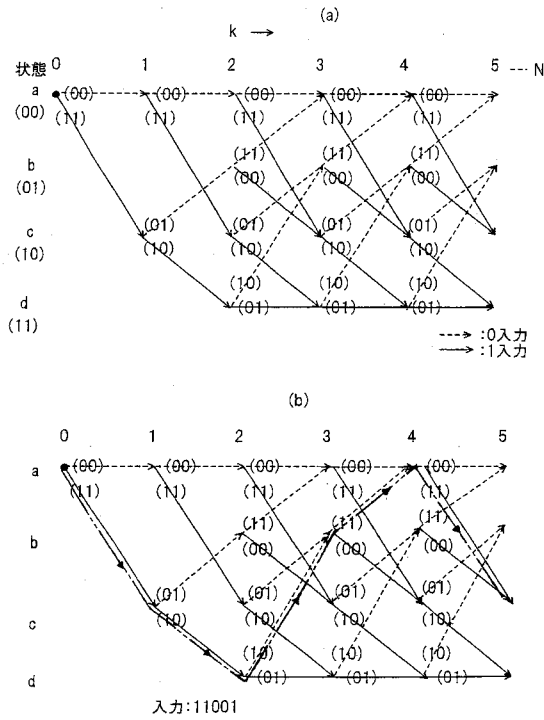
【図 1 5】



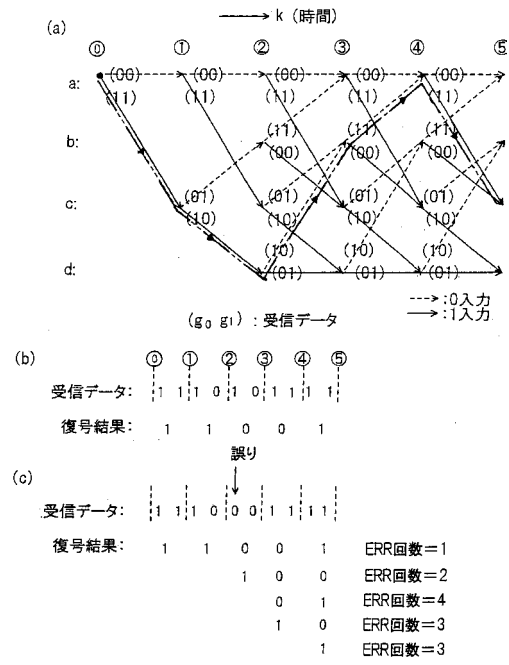
【図 1 6】



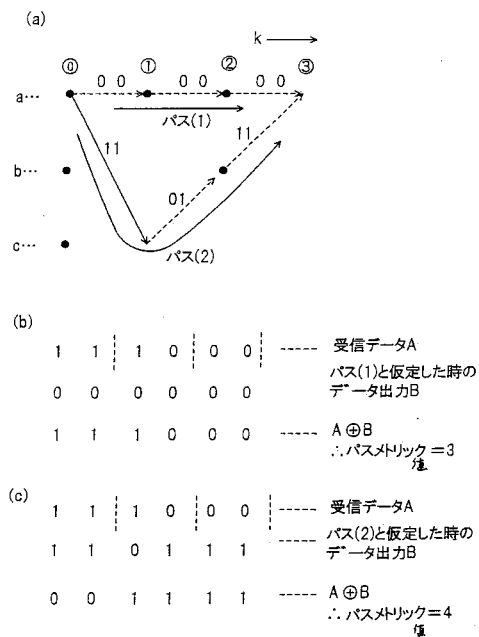
【図 17】



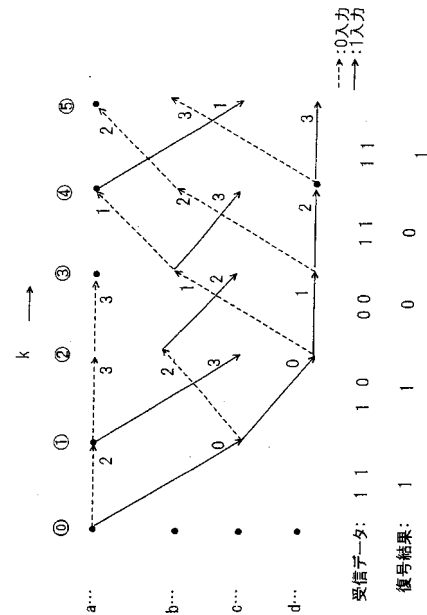
【図 18】



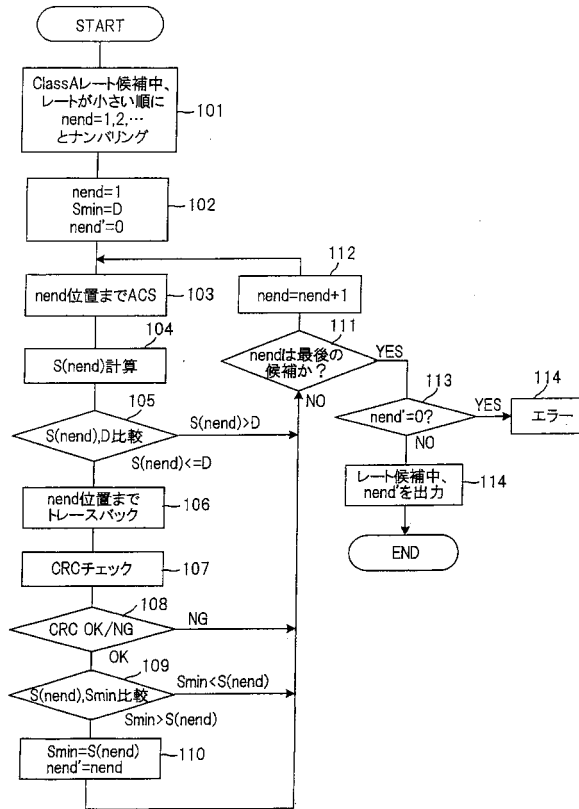
【図 19】



【図 20】



【図 2 1】



【図 2 2】

