

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4888335号
(P4888335)

(45) 発行日 平成24年2月29日 (2012. 2. 29)

(24) 登録日 平成23年12月22日 (2011. 12. 22)

(51) Int. Cl.		F I	
H03M	7/40	(2006.01)	H03M 7/40
H04N	7/26	(2006.01)	H04N 7/13 Z
G10L	19/00	(2006.01)	G10L 19/00 250

請求項の数 20 (全 28 頁)

(21) 出願番号	特願2007-277930 (P2007-277930)	(73) 特許権者	000002185
(22) 出願日	平成19年10月25日 (2007. 10. 25)		ソニー株式会社
(65) 公開番号	特開2009-105838 (P2009-105838A)		東京都港区港南1丁目7番1号
(43) 公開日	平成21年5月14日 (2009. 5. 14)	(74) 代理人	100067736
審査請求日	平成22年3月10日 (2010. 3. 10)		弁理士 小池 晃
		(74) 代理人	100096677
			弁理士 伊賀 誠司
		(72) 発明者	前田 祐児
			東京都港区港南1丁目7番1号 ソニー株式会社内
		審査官	北村 智彦

最終頁に続く

(54) 【発明の名称】 符号化方法及び装置、並びにプログラム

(57) 【特許請求の範囲】

【請求項 1】

入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化方法において、

前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は 0 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出工程と、

前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得工程と、

m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、

m が奇数の場合、連続する 2 つのビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有する符号化方法。

【請求項 2】

前記取得工程では、前記数値に対して前記基準整数 I の逆数に対応する整数を乗算する請求項 1 記載の符号化方法。

【請求項 3】

前記取得工程では、前記商及び前記剰余に対して前記基準整数 I を用いて補正を行う特徴とする請求項 2 記載の符号化方法。

【請求項 4】

前記ビット数 x が小数かつ所定の閾値 X ($X = M / 2$ 、 M は正の奇数) よりも大きい場合、前記ビット数を $x = X$ として上位ビット列を符号化し、下位ビット列を $(m - M) / 2$ ビット長のデータとして固定長符号化する請求項 3 記載の符号化方法。

【請求項 5】

前記符号化工程では、下記 (1) により前記剰余を結合する請求項 1 記載の符号化方法

【数 1】

$$Q_i = q_{2i} + I \cdot q_{2i+1} \quad (1)$$

10

ここで、 Q_i は結合剰余を示し、 q_{2i} 、 q_{2i+1} は連続する 2 つのビット列の剰余を示す。

【請求項 6】

入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化方法において、

前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / n$ 、 m は 0 以上の整数、 n は 3 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出工程と、

前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得工程と、

20

m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、 m が奇数の場合、連続する n 個のビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有する符号化方法。

【請求項 7】

入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化装置において、

前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は 0 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出手段と、

30

前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得手段と、

m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、 m が奇数の場合、連続する 2 つのビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化手段とを有する符号化装置。

【請求項 8】

入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化装置において、

40

前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / n$ 、 m は 0 以上の整数、 n は 3 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出手段と、

前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得手段と、

m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、 m が奇数の場合、連続する n 個のビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化手段とを有する符号化装置。

【請求項 9】

50

入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する処理をコンピュータに実行させるためのプログラムであって、

前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)を算出し、 2^x に近接する基準整数 I を導出する導出工程と、

前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得工程と、

m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、

m が奇数の場合、連続する2つのビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有するプログラム。

10

【請求項10】

入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する処理をコンピュータに実行させるためのプログラムであって、

前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / n$ 、 m は0以上の整数、 n は3以上の整数)を算出し、 2^x に近接する基準整数 I を導出する導出工程と、

前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得工程と、

m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、

m が奇数の場合、連続する n 個のビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有するプログラム。

20

【請求項11】

入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号方法において、

30

前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、

前記符号語を復号し、 m が奇数の場合、前記符号語から連続する2つのビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、

m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出工程と、

m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を2つの剰余に分割する剰余分割工程と、

m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有する復号方法。

40

【請求項12】

前記剰余分割工程では、前記結合剰余に前記基準整数 I の逆数に対応する整数を乗算する請求項11記載の復号方法。

【請求項13】

前記剰余分割工程では、各剰余に対して前記基準整数 I を用いて補正を行う請求項12記載の復号方法。

【請求項14】

前記ビット数 x が小数かつ所定の閾値 X ($X = M / 2$ 、 M は正の奇数)よりも大きい場合、前記ビット数を $x = X$ として上位ビット列を生成し、 $(m - M) / 2$ ビット長のデータを固定長復号して下位ビット列を生成する請求項13記載の復号方法。

50

【請求項 15】

前記剰余分割工程では、下記(2)により前記結合剰余を分割する請求項11記載の復号方法。

【数 2】

$$\begin{cases} \hat{q}_{2i+1} = \hat{Q}_i / \hat{I} \\ \hat{q}_{2i} = \hat{Q}_i - \hat{q}_{2i+1} \cdot \hat{I} \end{cases} \quad (2)$$

ここで、 Q_i は結合剰余を示し、 q_{2i} 、 q_{2i+1} は連続する2つのビット列の剰余を示す。

10

【請求項 16】

入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m/n$ 、 m は0以上の整数、 n は3以上の整数) にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号方法において、

前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、

前記符号語を復号し、 m が奇数の場合、前記符号語から連続する n 個 (n は3以上の整数) のビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、

20

m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出工程と、

m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を n 個の剰余に分割する剰余分割工程と、

m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有する復号方法。

【請求項 17】

入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m/2$ 、 m は0以上の整数) にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号装置において、

30

前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、

前記符号語を復号し、 m が奇数の場合、前記符号語から連続する2つのビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号手段と、

m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出手段と、

m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を2つの剰余に分割する剰余分割手段と、

m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成手段とを有する復号装置。

40

【請求項 18】

入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m/2$ 、 m は0以上の整数) にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号装置において、

前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、

前記符号語を復号し、 m が奇数の場合、前記符号語から連続する n 個 (n は3以上の整数) のビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対

50

応する剰余が結合された結合剰余とを復号する復号手段と、

mが奇数の場合、 2^x に近接する前記基準整数Iを導出する導出手段と、

mが奇数の場合、前記基準整数Iに基づいて前記結合剰余をn個の剰余に分割する剰余分割手段と、

mが奇数の場合、前記商に前記基準整数Iを乗算し、前記商に対応する剰余を加算してビット列を生成する生成手段とを有する復号装置。

【請求項19】

入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する処理をコンピュータに実行させるためのプログラムであって、

前記符号語を復号し、mが偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、

前記符号語を復号し、mが奇数の場合、前記符号語から連続する2つのビット列がそれぞれ表す各数値に対して基準整数Iで除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、

mが奇数の場合、 2^x に近接する前記基準整数Iを導出する導出工程と、

mが奇数の場合、前記基準整数Iに基づいて前記結合剰余を2つの剰余に分割する剰余分割工程と、

mが奇数の場合、前記商に前記基準整数Iを乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有するプログラム。

【請求項20】

入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する処理をコンピュータに実行させるためのプログラムであって、

前記符号語を復号し、mが偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、

前記符号語を復号し、mが奇数の場合、前記符号語から連続するn個 (n は3以上の整数)のビット列がそれぞれ表す各数値に対して基準整数Iで除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、

mが奇数の場合、 2^x に近接する前記基準整数Iを導出する導出工程と、

mが奇数の場合、前記基準整数Iに基づいて前記結合剰余をn個の剰余に分割する剰余分割工程と、

mが奇数の場合、前記商に前記基準整数Iを乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有するプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、音声、画像等のロスレス符号化で広く用いられているエントロピー符号化に関し、高い符号化効率を実現する符号化方法及び装置、並びにプログラムに関するものである。

【背景技術】

【0002】

音声、画像等のロスレス符号化で広く用いられているエントロピー符号化、とりわけGolomb-Rice符号化は、シフト演算やビットマスク演算などの簡易な演算のみで実行できる点で、高速・簡便な手法である。この方法は、信号を個別にあるビット位置で分割し、上位ビットを可変長符号化し、下位ビットを固定長符号化するものである。また、可変長符号化には、ちょうど上位ビットの表す数値が符号化語長に比例するものが多く用いられている。

【 0 0 0 3 】

【特許文献 1】特開 2 0 0 6 - 1 4 0 7 7 2 号公報

【発明の開示】

【発明が解決しようとする課題】

【 0 0 0 4 】

ところで、上述したゴロム・ライス符号化では、分割するビット位置が一般に信号の平均振幅等に応じて決定され、信号の分布に適合されるが、ビット位置が整数値であるため、実際の分布に対し上位ビット又は下位ビットに過剰なビット量が割り当てられ、結果として符号化効率の低下を招く恐れがあった。

【 0 0 0 5 】

本発明は、これらの問題点を鑑みてなされたものであり、符号化効率を向上させることのできる符号化方法及び装置、並びにプログラムを提供することを目的とする。

【課題を解決するための手段】

【 0 0 0 6 】

また、本発明に係る符号化方法は、入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化方法において、前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は 0 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出工程と、前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得工程と、 m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、 m が奇数の場合、連続する 2 つのビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有する。

【 0 0 0 7 】

また、本発明に係る符号化方法は、入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化方法において、前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / n$ 、 m は 0 以上の整数、 n は 3 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出工程と、前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得工程と、 m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、 m が奇数の場合、連続する n 個のビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有する。

【 0 0 0 8 】

また、本発明に係る符号化装置は、入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化装置において、前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は 0 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出手段と、前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得手段と、 m が偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、 m が奇数の場合、連続する 2 つのビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化手段とを有する。

【 0 0 0 9 】

また、本発明に係る符号化装置は、入力信号のビット列を最下位ビットから x ビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する符号化装置において、前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置 x ($x = m / n$ 、 m は 0 以上の整数、 n は 3 以上の整数) を算出し、 2^x に近接する基準整数 I を導出する導出手段と、前記ビット列が表す数値に対して前記基準整数 I で除算して求めた商と剰余とを取得する取得手段と、 m が偶数の場合、商を可変長符

10

20

30

40

50

号化し、剰余を固定長符号化し、符号語を生成し、mが奇数の場合、連続するn個のビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化手段とを有する。

【0010】

また、本発明に係るプログラムは、入力信号のビット列を最下位ビットからxビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する処理をコンピュータに実行させるためのプログラムであって、前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置x ($x = m / 2$ 、mは0以上の整数)を算出し、 2^x に近接する基準整数Iを導出する導出工程と、前記ビット列が表す数値に対して前記基準整数Iで除算して求めた商と剰余とを取得する取得工程と、mが偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、mが奇数の場合、連続する2つのビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有する。

10

【0011】

また、本発明に係るプログラムは、入力信号のビット列を最下位ビットからxビットの位置にて分割し、上位ビット列を可変長符号化し、下位ビット列を固定長符号化する処理をコンピュータに実行させるためのプログラムであって、前記入力信号の振幅値に基づいて符号化の基準となる最下位ビットからのビット位置x ($x = m / n$ 、mは0以上の整数、nは3以上の整数)を算出し、 2^x に近接する基準整数Iを導出する導出工程と、前記ビット列が表す数値に対して前記基準整数Iで除算して求めた商と剰余とを取得する取得工程と、mが偶数の場合、商を可変長符号化し、剰余を固定長符号化し、符号語を生成し、mが奇数の場合、連続するn個のビット列の商をそれぞれ可変長符号化し、当該商に対応する剰余を結合して固定長符号化し、符号語を生成する符号化工程とを有する。

20

【0012】

また、本発明に係る復号方法は、入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置x ($x = m / 2$ 、mは0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号方法において、前記符号語を復号し、mが偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、前記符号語を復号し、mが奇数の場合、前記符号語から連続する2つのビット列がそれぞれ表す各数値に対して基準整数Iで除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、mが奇数の場合、 2^x に近接する前記基準整数Iを導出する導出工程と、mが奇数の場合、前記基準整数Iに基づいて前記結合剰余を2つの剰余に分割する剰余分割工程と、mが奇数の場合、前記商に前記基準整数Iを乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有する。

30

【0013】

また、本発明に係る復号方法は、入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置x ($x = m / n$ 、mは0以上の整数、nは3以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号方法において、前記符号語を復号し、mが偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、前記符号語を復号し、mが奇数の場合、前記符号語から連続するn個 (nは3以上の整数)のビット列がそれぞれ表す各数値に対して基準整数Iで除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、mが奇数の場合、 2^x に近接する前記基準整数Iを導出する導出工程と、mが奇数の場合、前記基準整数Iに基づいて前記結合剰余をn個の剰余に分割する剰余分割工程と、mが奇数の場合、前記商に前記基準整数Iを乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有する。

40

【0014】

また、本発明に係る復号装置は、入力信号のビット列が該入力信号の振幅値に基づいて

50

算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号装置において、前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、前記符号語を復号し、 m が奇数の場合、前記符号語から連続する2つのビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号手段と、 m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出手段と、 m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を2つの剰余に分割する剰余分割手段と、 m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成手段とを有する。

10

【0015】

また、本発明に係る復号装置は、入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する復号装置において、前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、前記符号語を復号し、 m が奇数の場合、前記符号語から連続する n 個 (n は3以上の整数)のビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号手段と、 m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出手段と、 m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を n 個の剰余に分割する剰余分割手段と、 m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成手段とを有する。

20

【0016】

また、本発明に係るプログラムは、入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する処理をコンピュータに実行させるためのプログラムであって、前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、前記符号語を復号し、 m が奇数の場合、前記符号語から連続する2つのビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、 m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出工程と、 m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を2つの剰余に分割する剰余分割工程と、 m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有する。

30

【0017】

また、本発明に係るプログラムは、入力信号のビット列が該入力信号の振幅値に基づいて算出された符号化の基準となる最下位ビットからのビット位置 x ($x = m / 2$ 、 m は0以上の整数)にて分割され、上位ビット列が可変長符号化され、下位ビット列が固定長符号化された符号語を復号する処理をコンピュータに実行させるためのプログラムであって、前記符号語を復号し、 m が偶数の場合、前記上位ビット列と前記下位ビット列とを結合し、ビット列を生成し、前記符号語を復号し、 m が奇数の場合、前記符号語から連続する n 個 (n は3以上の整数)のビット列がそれぞれ表す各数値に対して基準整数 I で除算された商と、当該商に対応する剰余が結合された結合剰余とを復号する復号工程と、 m が奇数の場合、 2^x に近接する前記基準整数 I を導出する導出工程と、 m が奇数の場合、前記基準整数 I に基づいて前記結合剰余を n 個の剰余に分割する剰余分割工程と、 m が奇数の場合、前記商に前記基準整数 I を乗算し、前記商に対応する剰余を加算してビット列を生成する生成工程とを有する。

40

【発明の効果】

【0018】

50

本発明によれば、ビット列の最下位ビットからのビット位置 x を仮想的に m/n として細分化することにより、信号の分布に対し上位ビット又は下位ビットに適切なビット量を割り当てることができ、符号化効率を向上させることができる。

【発明を実施するための最良の形態】

【0019】

以下、本発明の具体的な実施の形態について、図面を参照しながら詳細に説明する。本発明は、主に音声、画像などのデジタル信号をロスレス符号化する際に使用することを意図しており、最下位ビットからのビット位置 x を仮想的に m/n (m は 0 以上の整数、 n は 2 以上の整数) として細分化することにより、信号の分布に対し上位ビット又は下位ビットに適切なビット量を割り当てるものである。ここでは、まず、 $n = 2$ の場合における音声信号の符号化及び復号について説明し、次いで、 n を 2 以上の整数とした場合について説明する。

10

【0020】

〔符号化処理〕

図1は、エンコーダ 10 (符号化器) の構成を示すブロック図である。このエンコーダ 10 は、音声信号 $x[i]$ が入力される入力端子 11 と、短期予測係数 α_p に基づいて近接サンプルから現在の信号を予測する短期予測部 12 と、短期予測係数 α_p を導出する短期予測係数導出部 13 と、入力端子 11 からの音声信号 $x[i]$ と短期予測部 12 からの出力信号とから予測残差信号 $r[i]$ を算出する加算器 14 と、ビット位置 $m/2$ に基づいて予測残差信号 $r[i]$ を符号語 I_γ に変換するエントロピー符号化部 15 と、予測残差信号 $r[i]$ を符号化する最下位ビットからのビット位置 $m/2$ を導出する基準値導出部 16 と、短期予測係数 α_p を符号語 I に変換する短期予測係数符号化部 17 と、符号語 I_γ 、基準ビット位置 m 及び符号語 I を多重化するマルチプレクサ 18 とを備えている。

20

【0021】

入力端子 11 には、所定のフレーム長 N 毎に分割された音声信号が入力される。音声信号は、一般に近接サンプル間の相関が高い信号であり、この相関を除去する短期予測部 12 及び加算器 14 で短期予測処理が実行されることにより、予測残差信号 $r[i]$ が得られる。

【0022】

【数 1】

$$r[i] = x[i] - \sum_{p=1}^P \alpha_p x[i-p] \quad (i = 0 \dots N-1) \quad (1)$$

30

【0023】

ここで、 $x[i]$ は音声信号、 $r[i]$ は予測残差信号、 α_p は短期予測係数、及び P は短期予測係数の次数を示す。

【0024】

短期予測係数 α_p は、短期予測係数導出部 13 にて音声信号の短時間相関を除去する様に求められ、その計算には、レビンソン・ダービンアルゴリズムなどが使用される。短期予測係数 α_p は、短期予測係数符号化部 17 にて符号語 I に変換されマルチプレクサ 18 に送られる。

40

【0025】

基準値導出部 16 は、予測残差信号 $r[i]$ を符号化の基準となる最下位ビットからのビット位置 $m/2$ を算出する。この計算には、例えば (2) 式を用いることができる。

【0026】

【数 2】

$$\frac{\sum_{i=0}^{N-1} |r[i]|}{N} \leq \left\lfloor 2^{m/2} \right\rfloor \quad (2)$$

【0027】

(2)式において、左辺は予測残差信号 $r[i]$ の平均振幅を表す。基準値導出部 16 は、この不等式を成立させる最小の整数 m を決定し、エントロピー符号化部 15 に送る。また、ビット位置 $m/2$ は、マルチプレクサ 18 に送られる。なお、 m の導出は別の方法を用いてもよい。

【0028】

エントロピー符号化部 15 は、後述のようにビット位置 $m/2$ に基づいて予測残差信号 $r[i]$ を符号語 I_γ に変換する。符号語 I_γ はマルチプレクサ 18 に転送される。

【0029】

マルチプレクサ 18 は、各符号語の多重化処理を行い、多重化データを出力する。

【0030】

次に、エントロピー符号化部 15 について、図 2 を用いて詳細に説明する。図 2 はエントロピー符号化部 15 の構成を示すブロック図である。

【0031】

エントロピー符号化部 15 は、予測残差信号 $r[i]$ を正負号なし信号 $y[i]$ に変換する正負号除去部 21 と、 m の L S B (Least Significant Bit) を抽出する L S B 抽出部 22 と、 m の L S B に応じて正負号なし信号 $y[i]$ の出力先を切り替えるスイッチ 23 と、正負号なし信号 $y[i]$ をビット位置 $m/2$ に基づいてシフト演算し、上位ビットと下位ビットに分割する信号分割部 24 と、正負号なし信号 $y[i]$ を $2^{m/2}$ に近接した整数 I で除算する信号除算部 25 と、 $2^{m/2}$ に近接した整数 I を導出する基準整数導出部 26 と、信号分割部 24 における商 p_i 、剰余 q_i 、信号除算部 25 における商 p_{2i} 、 p_{2i+1} 、及び剰余結合部 28 における剰余 q_{2i} 、 q_{2i+1} の結合剰余 Q_i を符号語 I_γ に変換する可変固定長符号化部 27 と、信号除算部 25 で剰余 q_{2i} 、 q_{2i+1} を結合剰余 Q_i に変換する剰余結合部 28 とを備えている。

【0032】

正負号除去部 21 は、予測残差信号 $r[i]$ を (3) 式により正負号なし信号 $y[i]$ に変換する。

【0033】

【数 3】

$$\begin{cases} y[i] = 2 \cdot r[i] & (r[i] \geq 0) \\ y[i] = -2 \cdot r[i] - 1 & (r[i] < 0) \end{cases} \quad (3)$$

【0034】

L S B 抽出部 22 は、 m の値が偶数であるか奇数であるかを、最下位ビット (L S B) を検査して抽出する。なお、 m が 0 なる時偶数として取り扱う (以下、同様)。 m が偶数の場合、正負号なし信号 $y[i]$ の出力先が信号分割部 24 となるようにスイッチ 23 を切り替える。また、 m が奇数の場合、正負号なし信号 $y[i]$ の出力先が信号除算部 25 となるようにスイッチ 23 を切り替える。

【0035】

(1) m が偶数の場合

10

20

30

40

50

mが偶数の場合、すなわちビット位置 $m/2$ が整数の場合、正負号なし信号 $y[i]$ は信号分割部24に入力される。ここで、(2)式の右辺は2の整数乗となるため、 $m/2 = m'$ とし、正負号なし信号 $y[i]$ の値を $2^{m'}$ で除算し、商 p_i と剰余 q_i を求める。実際には、正負号なし信号 $y[i]$ を m' ビット右シフトして商 p_i を求め、下位 m' ビットを取り出して剰余 q_i を得る、信号の分割が実行される。信号分割部24で得られた商 p_i と剰余 q_i は、可変固定長符号化部27に出力される。

【0036】

【数4】

$$\begin{cases} p_i = y[i] \gg m' \\ q_i = y[i] \& (2^{m'} - 1) \end{cases} \quad (4)$$

10

【0037】

可変固定長符号化部27は、商 p_i を可変長符号化し、剰余 q_i を m' ビットで固定長符号化し、符号語 I_Y に変換する。

【0038】

このようにmが偶数の場合には、各正負号なし信号 $y[0 \cdots N-1]$ について商 p_i と剰余 q_i を求め、符号語 I_Y に変換する。

20

【0039】

(2)mが奇数の場合

mが奇数の場合、すなわちビット位置 $m/2$ が小数の場合、正負号なし信号 $y[i]$ は信号除算部25に入力される。基準整数導出部26は、(2)式の右辺において、2のm乗を超えない整数Iを導出する。

【0040】

【数5】

$$I = \lfloor 2^{m/2} \rfloor \quad (5)$$

30

【0041】

ただし、整数Iは2のm乗に近接した値としてもよい。

【0042】

信号除算部25は、連続する正負号なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \cdots N/2 - 1$)を整数Iで除算して、商 p_{2i} 、 p_{2i+1} と剰余 q_{2i} 、 q_{2i+1} を(6)式より得る。なお、(6)式の上段と下段で同一の整数Iを使用しているが、必ずしも同一である必要はない。

【0043】

【数6】

$$\begin{cases} p_{2i} = y[2 \cdot i] / I, & q_{2i} = y[2 \cdot i] \% I \\ p_{2i+1} = y[2 \cdot i + 1] / I, & q_{2i+1} = y[2 \cdot i + 1] \% I \end{cases} \quad (6)$$

40

【0044】

(6)式で求めた商 p_{2i} 、 p_{2i+1} については、可変固定長符号化部27に出力され、剰余 q_{2i} 、 q_{2i+1} は、剰余結合部28に出力される。可変固定長符号化部27に出力された商 p_{2i} 、 p_{2i+1} は、mが偶数の場合と同様に可変長符号化される。

【0045】

50

剰余結合部 28 は、剰余 q_{2i} 、 q_{2i+1} の 2 個の値を組み合わせ、結合剰余 Q_i に変換する。剰余の性質上、剰余 q_{2i} 、 q_{2i+1} は整数 I を超えない。

【0046】

【数7】

$$0 \leq q_{2i}, q_{2i+1} < I \quad (7)$$

【0047】

(7) 式から、(8) 式のように剰余 q_{2i} 、 q_{2i+1} を結合剰余 Q_i に変換する。

10

【0048】

【数8】

$$0 \leq Q_i = q_{2i} + I \cdot q_{2i+1} < I^2 < 2^m \quad (8)$$

【0049】

これは、剰余 q_{2i} 、 q_{2i+1} を組み合わせ、結合剰余 Q_i が m ビットに格納できることを意味する。この結合剰余 Q_i は可変固定長符号化部 27 に出力される。

【0050】

20

可変固定長符号化部 27 は、商 p_{2i} 、 p_{2i+1} を可変長符号化し、結合剰余 Q_i を m ビットで固定長符号化し、符号語 I_Y を生成して出力する。

【0051】

このようにビット位置 $m/2$ が小数の場合には、連続する正負号なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \dots N/2 - 1$) を整数 I で除算して、商 p_{2i} 、 p_{2i+1} と剰余 q_{2i} 、 q_{2i+1} を (6) 式より得、剰余 q_{2i} 、 q_{2i+1} に関しては、2 個の値を組み合わせ、結合剰余 Q_i に変換して、商 p_{2i} 、 p_{2i+1} と結合剰余 Q_i を符号語 I_Y に変換する。

【0052】

続いて、エントロピー符号化部 15 の処理について、図 3 に示すフローチャートを用いて説明する。ステップ S31 において、正負号除去部 21 は、上記 (3) 式により予測残差信号 $r[i]$ を正負号なし信号 $y[i]$ に変換する。

30

【0053】

ステップ S32 において、LSB 抽出部 22 は、 m の値が偶数であるか、奇数であるかを m の値を示す最下位ビット (LSB) より判別する。 m が偶数の場合、すなわち 0 の場合、ステップ S33 に進み、LSB 抽出部 22 は、正負号なし信号 $y[i]$ の出力先が信号分割部 24 となるようにスイッチ 23 を切り替える。また、 m が奇数の場合、すなわち 1 の場合、ステップ S35 に進み、LSB 抽出部 22 は、正負号なし信号 $y[i]$ の出力先が信号除算部 25 となるようにスイッチ 23 を切り替える。

【0054】

40

m が偶数の場合、ステップ S33 において、信号分割部 24 は、ビット位置を $m/2 = m'$ とし、正負号なし信号 $y[i]$ の値を $2^{m'}$ で除算し、商 p_i 、剰余 q_i を求め (ステップ S34)、上位・下位ビットに分割する。商 p_i 、剰余 q_i は、可変固定長符号化部 27 に出力される。

【0055】

ステップ S38 において、可変固定長符号化部 27 は、商 p_i を可変長符号化し、剰余 q_i を m' ビットで固定長符号化し、符号語 I_Y に変換する。

【0056】

また、 m が奇数の場合、ステップ S35 において、基準整数導出部 26 は、上記 (2) 式の右辺において、 2 の m 乗を超えない整数 I を導出する。

50

【 0 0 5 7 】

ステップ S 3 6 において、信号除算部 2 5 は、連続する正負号なし信号 $y [2 i]$ 、 $y [2 i + 1]$ ($i = 0 \cdots N / 2 - 1$) を整数 I で除算して、商 $p_{2 i}$ 、 $p_{2 i + 1}$ と剰余 $q_{2 i}$ 、 $q_{2 i + 1}$ を上記 (6) 式より得る。

【 0 0 5 8 】

ステップ S 3 7 において、剰余結合部 2 8 は、剰余 $q_{2 i}$ 、 $q_{2 i + 1}$ の 2 個の値を組み合わせて結合剰余 Q_i に変換する。

【 0 0 5 9 】

ステップ S 3 8 において、可変固定長符号化部 2 7 は、商 $p_{2 i}$ 、 $p_{2 i + 1}$ を可変長符号化し、結合剰余 Q_i を m ビットで固定長符号化し、符号語 I_Y を生成して出力する。

10

【 0 0 6 0 】

このように m が偶数の場合、ビット位置を $m / 2 = m'$ とし、正負号なし信号 $y [i]$ の値を $2^{m'}$ で除算し、商 p_i 、剰余 q_i を求め、 m が奇数の場合、連続する正負号なし信号 $y [2 i]$ 、 $y [2 i + 1]$ ($i = 0 \cdots N / 2 - 1$) を $2^{m' / 2}$ に近接する整数 I で除算し、剰余 $q_{2 i}$ 、 $q_{2 i + 1}$ の 2 個の値を組み合わせて結合剰余 Q_i に変換することにより、ビット列を分割するビット位置を $1 / 2$ 単位とすることができ、符号化効率を向上させることができる。

【 0 0 6 1 】

なお、信号除算部 2 5 において、上記 (6) 式を用いて除算することとしたが、上記 (5) 式に示す整数 I の逆数を予めテーブル化しておいてもよい。これにより、演算量を減少させることができる。

20

【 0 0 6 2 】

具体的に説明すると、上記 (5) 式の逆数は、下記 (9) 式で表され、この (9) 式に対応する値 I' 、 E をテーブルとして整数化すればよい。

【 0 0 6 3 】

【 数 9 】

$$I' = \lfloor 2^E / I \rfloor \quad (9)$$

30

【 0 0 6 4 】

また、(9) 式を基に (6) 式を変形すると、(1 0) 式が得られる。

【 0 0 6 5 】

【 数 1 0 】

$$\begin{cases} p_{2i} = (y[2 \cdot i] \cdot I') \gg E, & q_{2i} = y[2 \cdot i] - p_{2i} \cdot I \\ p_{2i+1} = (y[2 \cdot i + 1] \cdot I') \gg E, & q_{2i+1} = y[2 \cdot i + 1] - p_{2i+1} \cdot I \end{cases} \quad (10)$$

【 0 0 6 6 】

このように (9) 式に対応する値 I' 、 E をテーブルとして整数化すれば、除算及び剰余の計算をせずに値を求めることができる。一般に除算が必要になると演算量が大きくなるが、上述のように整数 I の逆数に対応する整数を予め記憶しておくことで、乗算を使用することができ、演算量を抑制することができる。

40

【 0 0 6 7 】

また、上記 (1 0) 式を補正することにより、(9) 式の整数化による影響を防ぐことができる。(7) 式のように剰余 $q_{2 i}$ 、 $q_{2 i + 1}$ は 0 以上整数 I 未満であるが、(1 0) 式の剰余 $q_{2 i}$ 、 $q_{2 i + 1}$ の演算では、0 未満になるか、整数 I 以上になる可能性がある。その際は下記 (1 1) 式の補正演算を行う。

【 0 0 6 8 】

50

【数 1 1】

$$\begin{cases} q_i < 0 & \rightarrow p_i - = 1, & q_i + = I \\ q_i \geq I & \rightarrow p_i + = 1, & q_i - = I \end{cases} \quad (11)$$

【0069】

ここで、 $2i$ 、 $2i+1$ とも同一の処理が行われるため、 $2i$ 、 $2i+1$ を i とした。補正演算は、剰余 q_i が0未満の場合、商 p_i からの1の減算に対して剰余 q_i に整数 I を加算し、剰余 q_i が整数 I 以上の場合、商 p_i への1の加算に対して剰余 q_i から整数 I を減算する。これにより、(7)式のように剰余 q_{2i} 、 q_{2i+1} は0以上整数 I 未満とすることができ、符号化効率を向上させることができる。

10

【0070】

さらに、 m に閾値 M を設定し、 m が閾値 M より大なる場合、すなわち、最下位ビットからのビット数 x ($x = m/2$)が少数かつ所定の閾値 X ($X = M/2$)よりも大きい場合について、予め正負号なし信号 $y[i]$ を下位 $(m-M)/2$ ビット t_i とそれより上位ビット $y'[i]$ に分割するようにしてもよい。

【0071】

【数 1 2】

$$t_i = y[i] \& (2^{(m-M)/2} - 1), \quad y'[i] = y[i] \gg ((m-M)/2) \quad (12)$$

20

【0072】

こうして得られた下位ビット t_i は $(m-M)/2$ ビット長のデータとしてマルチプレクサ18に送られ、上位ビット $y'[i]$ は、(6)式又は(10)式で $m=M$ として変換処理がなされるため、必要以上に整数 I の逆数に対応する値のテーブルを装備しなくても済む。

【0073】

〔復号処理〕

以下、上述した符号化データの復号処理について説明する。図4は、デコーダ(復号化器)の構成を示すブロック図である。このデコーダ40は、多重化データを逆多重化し、符号語 I_γ 、基準ビット位置 $m/2$ 及び符号語 I を復号するデマルチプレクサ41と、基準ビット位置 $m/2$ に基づいて符号語 I_γ を予測残差信号 $r[i]$ に変換するエントピー復号化部42と、符号語 I を短期予測係数 ρ に変換する短期予測係数復号化部43と、予測残差信号 $r[i]$ とサンプル間の相関が合成された過去の音声信号とから音声信号 $x[i]$ を算出する加算器44と、短期予測係数 ρ に基づいて近接サンプル間の相関が合成された音声信号を生成する短期合成部45とを備えている。

30

【0074】

デマルチプレクサ41に入力された多重化データは逆多重化が施され、個々の符号語に復号される。デマルチプレクサ41より復号された符号語 I_γ 及び復号されたビット位置 $m/2$ は、復号残差信号 $r[i]$ を復号するエントピー復号化部42に入力される。

40

【0075】

短期予測係数の符号語 I は、短期予測係数復号化部43で復号されて短期予測係数 ρ が生成され短期合成部45に送られる。加算器44及び短期合成部45において短期予測係数 ρ 及び過去の復号信号から音声信号 $x[i]$ が(13)式より生成される。

【0076】

【数 1 3】

$$\hat{x}[i] = \hat{r}[i] + \sum_{p=1}^P \hat{\alpha}_p \cdot \hat{x}[i-p] \quad (13)$$

【0077】

以上のようにして、ロスレスに信号を復号することが可能になる。

【0078】

次に、エントロピー復号化部 4 2 について、図 5 を用いて詳細に説明する。図 5 はエントロピー復号化部 4 2 の構成を示すブロック図である。 10

【0079】

エントロピー復号化部 4 2 は、 m が偶数か奇数かを判別し、符号語 I_γ を商 p_i 、剰余 q_i 、又は商 p_{2i} 、 p_{2i+1} 、結合剰余 Q_i に変換する可変固定長復号化部 5 1 と、商 p_i 、剰余 q_i を m に基づいて上位ビットと下位ビットを結合し、正負号なし信号 $y[i]$ を生成する信号結合部 5 2 と、 2 の m 乗に近接した整数 I を導出する基準整数導出部 5 3 と、結合剰余 Q_i から剰余 q_{2i} 、 q_{2i+1} を復号する剰余分割部 5 4 と、商 p_{2i} 、 p_{2i+1} 、剰余 q_{2i} 、 q_{2i+1} 及び m から正負なし信号 $y[2i]$ 、 $y[2i+1]$ を復号する信号乗算部 5 5 と、 m の最下位ビット(LSB)を抽出する LSB 抽出部 5 6 と、 m の LSB に応じて正負号なし信号 $y[i]$ の出力元を切り替えるスイッチ 5 7 と、正負号なし信号 $y[i]$ を復号残差信号 $r[i]$ に変換する正負号復元部 5 8 とを備えている。 20

【0080】

(1) m が偶数の場合

m が偶数の場合、すなわち、ビット位置 $m/2$ が整数の場合、 $m/2 = m'$ とし、可変固定長復号化部 5 1 は、入力された符号語 I_γ を商 p_i 、剰余 q_i に変換する。 m' は整数であるため、信号結合部 5 2 は、1 サンプルごとにシフト演算し、正負号なし信号 $y[i]$ を復号する。

【0081】

【数 1 4】

$$\hat{y}[i] = (\hat{p}_i \ll \hat{m}') \mid \hat{q}_i \quad (14)$$

【0082】

(2) m が奇数の場合

ビット位置 m が奇数の場合、すなわち、ビット位置 $m/2$ が小数の場合、可変固定長復号化部 5 1 は、入力された符号語 I_γ を商 p_{2i} 、 p_{2i+1} 、結合剰余 Q_i に変換する。 $m/2$ は小数であるため、2 サンプルまとめて正負なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \cdots N/2 - 1$) を復号する。 40

【0083】

基準整数導出部 5 3 は、 2 の m 乗に近接した整数 I を (15) 式より求める。なお、整数 I は、エンコーダ 1 0 と同一の方法で求めれば、他の方法を用いても構わない。

【0084】

【数 1 5】

$$\hat{I} = \left\lfloor 2^{\hat{m}'/2} \right\rfloor \quad (15)$$

【0085】

剰余分割部 54 は、下記 (16) 式を用いて、結合剰余 Q_i から剰余 q_{2i} 、 q_{2i+1} を復号する。

【0086】

【数16】

$$\begin{cases} \hat{q}_{2i+1} = \hat{Q}_i / \hat{I} \\ \hat{q}_{2i} = \hat{Q}_i - \hat{q}_{2i+1} \cdot \hat{I} \end{cases} \quad (16)$$

10

【0087】

信号乗算部 55 は、商 p_{2i} 、 p_{2i+1} 、剰余 q_{2i} 、 q_{2i+1} 及び整数 I から (17) 式より正負なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \cdots N/2 - 1$) を復号する。

【0088】

【数17】

$$\begin{cases} \hat{y}[2 \cdot i] = \hat{p}_{2i} \cdot \hat{I} + \hat{q}_{2i} \\ \hat{y}[2 \cdot i + 1] = \hat{p}_{2i+1} \cdot \hat{I} + \hat{q}_{2i+1} \end{cases} \quad (17)$$

20

【0089】

LSB抽出部 56 は、 m の最下位ビット (LSB) を抽出することにより、偶数か奇数かを判別する。そして、LSB抽出部 56 は、 m が偶数の場合、正負号なし信号 $y[i]$ の出力元が信号結合部 52 となるようにスイッチ 57 を上方に接続し、 m が奇数の場合、正負号なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \cdots N/2 - 1$) の出力元が信号乗算部 55 となるように下方に接続する。

【0090】

正負号復元部 58 は、(18) 式より正負号なし信号 $y[i]$ を復号残差信号 $r[i]$ に変換する。

30

【0091】

【数18】

$$\begin{cases} \hat{r}[i] = \hat{y}[i] \gg 1, & ((\hat{y}[i] \& 0x1) == 0) \\ \hat{r}[i] = \sim (\hat{y}[i] \gg 1), & ((\hat{y}[i] \& 0x1) == 1) \end{cases} \quad (18)$$

【0092】

すなわち、正負号なし信号 $y[i]$ の最下位ビットが 1 の場合に右 1 ビットシフト後ビット反転がなされ、最下位ビットが 0 の場合に右 1 ビットシフトのみ行われ正負号が復元される。

40

【0093】

続いて、エントロピー復号化部 42 の処理について、図 6 に示すフローチャートを用いて説明する。ステップ S61 において、可変固定長復号化部 51 は、符号語 I_y を復号し、 m が偶数か奇数かを判別する (ステップ S62)。

【0094】

m が偶数の場合、ステップ S63 に進み、ビット位置を $m/2 = m'$ とし、信号結合部 52 は、上位ビットと下位ビットとを結合し、正負号なし信号 $y[i]$ を復号する (ステップ S64)。

50

【 0 0 9 5 】

mが奇数の場合、ステップS 6 5に進み、基準整数導出部5 3は、2のm乗に近接した整数Iを導出する。ステップS 6 6において、剰余分割部5 4は、整数Iを用いて、(1 6)式より結合剰余 Q_i から剰余 q_{2i} 、 q_{2i+1} を復号する。ステップS 6 7において、商 p_{2i} 、 p_{2i+1} 、剰余 q_{2i} 、 q_{2i+1} 及び整数Iから(1 7)式より正負なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \cdots N/2 - 1$)を復号する。

【 0 0 9 6 】

ステップS 6 8において、正負号復元部5 8は、(1 8)式より正負号なし信号 $y[i]$ を復号残差信号 $r[i]$ に変換する。

【 0 0 9 7 】

このようにmが偶数の場合、ビット位置を $m/2 = m'$ とし、上位ビットと下位ビットとを結合し、正負号なし信号 $y[i]$ を復号し、mが奇数の場合、整数Iを用いて、結合剰余 Q_i から剰余 q_{2i} 、 q_{2i+1} を復号し、商 p_{2i} 、 p_{2i+1} 、剰余 q_{2i} 、 q_{2i+1} 及び整数Iから正負なし信号 $y[2i]$ 、 $y[2i+1]$ ($i = 0 \cdots N/2 - 1$)を復号することにより、ビット位置を1/2単位とした復号を行うことができる。

【 0 0 9 8 】

なお、剰余分割部5 4において除算があるが、(1 9)式のように整数Iの逆数に相当する値 I' を予めテーブル化しておくことにより乗算を使用することも可能である。

【 0 0 9 9 】

【 数 1 9 】

$$\hat{I}' = \lfloor 2^{\hat{E}} / \hat{I} \rfloor \quad (19)$$

【 0 1 0 0 】

また、(1 9)式より(1 6)式の1番目の式は(2 0)式のように変換可能である。

【 0 1 0 1 】

【 数 2 0 】

$$\hat{q}_{2i+1} = (\hat{Q}_i \cdot \hat{I}') \gg \hat{E} \quad (20)$$

【 0 1 0 2 】

また、(1 9)式で示す値 I' は、整数化されているため、誤差が生じた場合、下記(2 1)式に示すような補正を行うことが好ましい。

【 0 1 0 3 】

【 数 2 1 】

$$\begin{cases} \hat{q}_{2i} < 0 & \rightarrow \hat{q}_{2i+1}^- = 1, \quad \hat{q}_{2i}^+ = \hat{I} \\ \hat{q}_{2i} \geq \hat{I} & \rightarrow \hat{q}_{2i+1}^+ = 1, \quad \hat{q}_{2i}^- = \hat{I} \end{cases} \quad (21)$$

【 0 1 0 4 】

また、エンコーダにてmに閾値Mを設定し、正負号なし信号 $y[i]$ を下位 $(m - M)/2$ ビット t_i とそれより上位ビット $y'[i]$ に分割した場合、mが所定の閾値Mより大であるなら、デマルチプレクサ4 1より取得した符号語より t_i を復号する。つまり、最下位ビットからのビット数xが小数かつ所定の閾値X($X = M/2$ 、Mは正の奇数)よりも大きい場合、ビット数を $x = X$ として上位ビット列 $y'[i]$ を生成し、 $(m - M)/2$ ビット長のデータを固定長復号して下位ビット列 t_i を生成する。具体的には、上記(1 5)式～(1 7)式におけるmをMに置き換え求めた結果を $y'[i]$ とした上で(

10

20

30

40

50

22)式を用いて $y[i]$ を復号する。

【0105】

【数22】

$$\hat{y}[i] = (\hat{y}'[i] \ll ((\hat{m} - M) / 2)) | t_i \quad (22)$$

【0106】

〔一般化した実施の形態〕

上述の符号化方法では、最下位ビットからのビット位置を仮想的に $m/2$ とした $n=2$ の場合について説明したが、ここでは、 n が 2 を越える一般化した符号化方法について説明する。この場合、 m を次式の通り決定する。

【0107】

【数23】

$$\frac{\sum_{i=0}^{N-1} |r[i]|}{N} \leq \lfloor 2^{m/n} \rfloor \quad (23)$$

10

20

【0108】

そして、 m/n が整数である場合は従来通りのゴロム・ライス符号化を行うが、 m/n が小数である場合は整数 I を以下の通り導入する。

【0109】

【数24】

$$I = \lfloor 2^{m/n} \rfloor \quad (24)$$

30

【0110】

ここで、正負号なし信号 $y[n \cdot i]$, $y[n \cdot i + 1]$, ..., $y[n \cdot i + n - 1]$ を以下の通り分解する。

【0111】

【数25】

$$p_{n \cdot i + k} = y[n \cdot i + k] / I, \quad q_{n \cdot i + k} = y[n \cdot i + k] \% I \quad (0 \leq k < n) \quad (25)$$

【0112】

また、結合剰余 Q_i を剰余 $q_{n \cdot i + k}$ より決定する。

【0113】

【数26】

$$\begin{cases} Q_{i,0} = q_{n \cdot i + 0} \\ Q_{i,k} = Q_{i,k-1} \cdot I + q_{n \cdot i + k} \quad (k = 1, \dots, n-1) \\ Q_i = Q_{i,n-1} \end{cases} \quad (26)$$

40

50

【0114】

これは、k = 0 の時は剰余 $q_{n \cdot i + 0}$ を結合剰余 $Q_{i, 0}$ に代入し、k = 1 より順に結合剰余 $Q_{i, k-1}$ を I 倍しながら剰余 $q_{n \cdot i + k}$ を加算している。

【0115】

一方、復号化においては、逆に結合剰余 Q_i から剰余 $q_{n \cdot i + k}$ を次式より求める。

【0116】

【数27】

$$\left\{ \begin{array}{l} \hat{Q}_{i,n-1} = \hat{Q}_i \\ \hat{Q}_{i,k-1} = \hat{Q}_{i,k} / \hat{I}, \quad \hat{q}_{n \cdot i + k} = \hat{Q}_{i,k} - \hat{Q}_{i,k-1} \cdot \hat{I} \quad (k = n-1, \dots, 1) \\ \hat{q}_{n \cdot i + 0} = \hat{Q}_{i,0} \end{array} \right. \quad (27) \quad 10$$

【0117】

このように最下位ビットからのビット位置を仮想的に m/n として細分化することにより、信号の分布に対し上位ビット又は下位ビットに適切なビット量が割り当てることができ、符号化効率を向上させることができる。

【0118】

なお、本発明は上述した実施の形態のみに限定されるものではなく、本発明の要旨を逸脱しない範囲において種々の変更が可能であることは勿論である。例えば、上述の実施の形態では、ハードウェアの構成として説明したが、これに限定されるものではなく、任意の処理を、CPU (Central Processing Unit) にコンピュータプログラムを実行させることにより実現することも可能である。この場合、コンピュータプログラムは、記録媒体に記録して提供することも可能であり、また、インターネットその他の伝送媒体を介して伝送することにより提供することも可能である。

【0119】

〔符号化の具体例〕

次に、上述したエントロピー符号化法について具体例を挙げて説明する。ここで、基準値導出部 16 において入力信号 $r[0, \dots, N-1]$ より求めた $m = 8$ の場合と、 $m = 9$ の場合について、 $r[2i] = -19$ 、 $r[2i+1] = 7$ を符号化することとする。

【0120】

また、(4) 式や (6) 式における商 p_i は、可変固定長符号化部 27 において、表 1 に示す符号化表のように可変長符号化されるものとする。

【0121】

【表1】

表1. 符号化表

p_i	符号語	符号長 [bit]
0	1	1
1	01	2
2	001	3
3	0001	4

【0122】

10

20

30

40

50

予測残差信号 $r[2i]$, $r[2i+1]$ は、(3)式より正負号なし信号 $y[i]$ に変換される。

【0123】

【数28】

$$\begin{cases} y[2 \cdot i] = -2 \cdot (-19) - 1 = 37 \\ y[2 \cdot i + 1] = 2 \cdot 7 = 14 \end{cases}$$

【0124】

10

ここで、 $m = 8$ であるとき、 $m' = m / 2 = 4$ であるため、 $y[i]$ は(4)式より以下の通り分割される。

【0125】

【数29】

$$\begin{cases} p_{2i} = 37 \gg 4 = 2, & q_{2i} = 37 \& 15 = 5 = 0b0101 \\ p_{2i+1} = 14 \gg 4 = 0, & q_{2i+1} = 14 \& 15 = 14 = 0b1110 \end{cases}$$

【0126】

20

したがって、これら2サンプルに対する符号語 I_y は、表1より例えば以下のようになり、 $r[2i]$ を7ビット、 $r[2i+1]$ を5ビットの合計12ビットで表現することになる。

【0127】

【数30】

$$I_y = (0b001, 0b0101, 0b1, 0b1110)$$

【0128】

また、 $m = 9$ であるとき、(5)式より整数 $I = 22$ となるため、 $y[2i]$ 、 $y[2i+1]$ を(6)式より除算する。

30

【0129】

【数31】

$$\begin{cases} p_{2i} = 37 / 22 = 1, & q_{2i} = 37 \% 22 = 15 \\ p_{2i+1} = 14 / 22 = 0, & q_{2i+1} = 14 \% 22 = 14 \end{cases}$$

【0130】

また、(8)式より結合剰余 Q_i を求める。

40

【0131】

【数32】

$$Q_i = 15 + 22 \cdot 14 = 323 = 0b101000011$$

【0132】

したがって、これら2サンプルに対する符号語 I_y は、表1より例えば以下のようになり、 $r[2i]$ 及び $r[2i+1]$ を合計12ビットで表現することになる。

【0133】

50

【数 3 3】

$$I_Y = (0b01, 0b1, 0b101000011)$$

【0 1 3 4】

〔復号化の具体例〕

次に、上述した 2 サンプルの符号語を復号化する手順を説明する。表 1 の商 p_i の項と符号語 I_Y の項を見ると符号語の 1 が現れるまでの 0 の数が商 p_i の値と一致しており、これを使用して符号語より商 p_i を求める。

10

【0 1 3 5】

$m = 8$ であるとき、商 p_{2i} の符号語 I_Y は、最初に 1 が現れるまで 2 ビット 0 があるため、商 $p_{2i} = 2$ が求まり、次の 4 ビットにより剰余 $q_{2i} = 5$ が求まる。同様に、商 p_{2i+1} の符号語 I_Y は、直後に 1 が現れるため、商 $p_{2i+1} = 0$ が求まり、次の 4 ビットにより剰余 $q_{2i+1} = 14$ が求まる。そして、上記 (17) 式より $y[2i]$ 、 $y[2i+1]$ が復号される。

【0 1 3 6】

【数 3 4】

$$\begin{cases} \hat{y}[2 \cdot i] = (2 \ll 4) | 5 = 37 \\ \hat{y}[2 \cdot i + 1] = (0 \ll 4) | 14 = 14 \end{cases}$$

20

【0 1 3 7】

また、 $m = 9$ であるとき、符号語 I_Y の、最初に 1 が現れるまで 1 ビット 0 があるため、商 $p_{2i} = 1$ が求まり、直後に 1 が現れるため、商 $p_{2i+1} = 0$ が求まり、次の 4 ビットにより $Q_i = 323$ が求まる。(15) 式より $I = 22$ であるため、(16) 式より剰余 q_{2i} 、 q_{2i+1} を復号する。

【0 1 3 8】

【数 3 5】

$$\begin{cases} \hat{q}_{2i+1} = 323 / 22 = 14 \\ \hat{q}_{2i} = 323 - 14 \cdot 22 = 15 \end{cases}$$

30

【0 1 3 9】

そして、(17) 式より $y[i]$ を復号する。

【0 1 4 0】

【数 3 6】

$$\begin{cases} \hat{y}[2 \cdot i] = 1 \cdot 22 + 15 = 37 \\ \hat{y}[2 \cdot i + 1] = 0 \cdot 22 + 14 = 14 \end{cases}$$

40

【0 1 4 1】

また、(18) 式より復号残差信号 $r[2i] = -19$ 、 $r[2i+1] = 7$ が復号される。

【0 1 4 2】

〔 m に閾値 M を設定した場合の処理の具体例〕

50

閾値 $M = 27$ である場合において、 $m = 29$ である場合の符号化について説明する。ここで、予測残差信号 $r[2 \cdot i] = 79680$ 、 $r[2 \cdot i + 1] = -137735$ について、正負号なし信号を (3) 式より求めると、 $y[2 \cdot i] = 159360$ 、 $y[2 \cdot i + 1] = 275469$ となる。

【0143】

整数 I は、(5) 式より求められる。

【0144】

【数37】

$$I = \lfloor 2^{M/2} \rfloor = \lfloor 2^{27/2} \rfloor = 11585$$

10

【0145】

また、上記 (12) 式により、上位ビットと下位ビットに分割する。ここで、下位ビットは $(m - M) / 2 = 1$ ビットである。

【0146】

【数38】

$$\begin{cases} t_{2 \cdot i} = y[2 \cdot i] \& 0x1 = 0, & y'[2 \cdot i] = y[2 \cdot i] \gg 1 = 79680 \\ t_{2 \cdot i + 1} = y[2 \cdot i + 1] \& 0x1 = 1, & y'[2 \cdot i + 1] = y[2 \cdot i + 1] \gg 1 = 137734 \end{cases}$$

20

【0147】

(6) 式により、商 $p_{2 \cdot i}$ 、 $p_{2 \cdot i + 1}$ と剰余 $q_{2 \cdot i}$ 、 $q_{2 \cdot i + 1}$ を計算する。

【0148】

【数39】

$$\begin{cases} p_{2 \cdot i} = y'[2 \cdot i] / I = 6, & q_{2 \cdot i} = y'[2 \cdot i] \% I = 10170 \\ p_{2 \cdot i + 1} = y'[2 \cdot i + 1] / I = 11, & q_{2 \cdot i + 1} = y'[2 \cdot i + 1] \% I = 10299 \end{cases}$$

30

【0149】

結合剰余 Q_i は、(8) 式より 27 ビットのデータとして得られる。

【0150】

【数40】

$$Q_i = q_{2 \cdot i} + I \cdot q_{2 \cdot i + 1} = 10170 + 11585 * 10299 = 119324085$$

【0151】

このようにして、以下の 5 変数が求められる。

40

【0152】

【数41】

$$(p_{2 \cdot i}, p_{2 \cdot i + 1}, Q_i, t_{2 \cdot i}, t_{2 \cdot i + 1}) = (6, 11, 119324085, 0, 1)$$

【0153】

続いて、この 5 変数の復号化処理について説明する。まず、ビットストリームより取得した $m (= 29 > M)$ より整数 I を求める。

【0154】

50

【数 4 2】

$$\hat{I} = \lfloor 2^{M/2} \rfloor = \lfloor 2^{27/2} \rfloor = 11585$$

【0 1 5 5】

(16) 式より結合剰余 Q_i から剰余 q_{2i} , q_{2i+1} を求める。

【0 1 5 6】

【数 4 3】

10

$$\begin{cases} \hat{q}_{2\cdot i+1} = \hat{Q}_i / \hat{I} = 119324085 / 11585 = 10299 \\ \hat{q}_{2\cdot i} = \hat{Q}_i - \hat{q}_{2\cdot i+1} \cdot \hat{I} = 119324085 - 10299 \cdot 11585 = 10170 \end{cases}$$

【0 1 5 7】

(17) 式より正負号なし信号 $y'[2i]$, $y'[2i+1]$ を求める。

【0 1 5 8】

【数 4 4】

20

$$\begin{cases} \hat{y}'[2\cdot i] = \hat{p}_{2\cdot i} \cdot \hat{I} + \hat{q}_{2\cdot i} = 6 \cdot 11585 + 10170 = 79680 \\ \hat{y}'[2\cdot i+1] = \hat{p}_{2\cdot i+1} \cdot \hat{I} + \hat{q}_{2\cdot i+1} = 11 \cdot 11585 + 10299 = 137734 \end{cases}$$

【0 1 5 9】

(22) 式より下位ビット t_{2i} , t_{2i+1} (ここでは1ビット) を用いて正負号なし信号 $y[2i]$, $y[2i+1]$ を復号する。

【0 1 6 0】

30

【数 4 5】

$$\begin{cases} \hat{y}[2\cdot i] = (\hat{y}'[2\cdot i] \ll 1) | \hat{t}_{2\cdot i} = (79680 \ll 1) | 0 = 159360 \\ \hat{y}[2\cdot i+1] = (\hat{y}'[2\cdot i+1] \ll 1) | \hat{t}_{2\cdot i+1} = (137734 \ll 1) | 1 = 275469 \end{cases}$$

【0 1 6 1】

(18) 式より正負号を復号し、 $r[2\cdot i] = 79680$, $r[2\cdot i+1] = -137735$ を得る。

【0 1 6 2】

40

〔本発明の適用例1〕

本発明は、44 kHz、16 bit 又は 96 kHz、24 bit のステレオ音源のロスレス圧縮技術へ適用可能である。ここでは、従来のゴロム・ライス符号化を適用した場合と本発明を適用した場合の圧縮効率について検証した。

【0 1 6 3】

表2は、10～30秒程度の44 kHz、16 bit のステレオ音源を使用して圧縮効率を測定した結果を示す。圧縮効率(ratio)は次式により定義した。

【0 1 6 4】

【数46】

$$ratio = \frac{original_file_size - compressed_file_size}{original_file_size} \times 100[\%]$$

【0165】

ここで、original_file_sizeは原音のファイルサイズ、compressed_file_sizeは圧縮したファイルサイズであり、ratioが大きいほど圧縮効率が高いことを示す。

【0166】

10

【表2】

表2 提案方式と従来方式の圧縮効率の比較

音源ファイル	演奏時間[秒]	ratio		
		提案方式[%]	従来方式[%]	差[%]
音源1	10	61.893	61.821	0.072
音源2	20	42.594	42.472	0.122
音源3	20	47.202	47.080	0.122
音源4	20	43.887	43.771	0.116
音源5	20	37.455	37.310	0.145
音源6	20	62.047	62.008	0.039
音源7	30	40.847	40.723	0.124
総計	140	47.523	47.419	0.104

20

30

【0167】

表2に示すように、本発明を適用することにより、圧縮効率を0.1%以上改善することができる。なお、本発明の適用によるデコーダの演算量の増加については、 $f_s = 44.1 \text{ kHz}$ で $0.3 \sim 0.5 \text{ MIPS}$ 、 $f_s = 96 \text{ kHz}$ で $0.7 \sim 1.0 \text{ MIPS}$ 程度で、微小である。

【0168】

〔本発明の適用例2〕

次に、本発明を適用した際の効果の一例について検証する。ここでは、 44.1 kHz 、16ビットの $N = 2048$ サンプルからなる対象信号のあるフレームについて、(2)式から求められた m が、従来のゴロム・ライス符号化の値では24 ($m/2 = 12$)、提案方式では23 ($m/2 = 11.5$)である場合の可変長符号化及び固定長符号化について検証する。

40

【0169】

表3は、可変長符号化を行う2048サンプル分の商 p_i について求めた度数分布を示す。ここで、ゴロム・ライス符号化と同様に商 p_i の符号語のビット長は $p_i + 1$ ビットになるものとする。

【0170】

【表 3】

表3 あるフレームの可変長符号化のビット数の比較

p _i	ビット長	提案方式 (m=23)		従来方式 (m=24)	
		度数	各ビット数	度数	各ビット数
0	1	1228	1228	1388	1388
1	2	327	654	326	652
2	3	186	558	176	528
3	4	130	520	92	368
4	5	75	375	39	195
5	6	48	288	17	102
6	7	24	168	7	49
7	8	17	136	1	8
8	9	7	63	0	0
9	10	3	30	1	10
10	11	1	11	1	11
11	12	0	0	0	0
12	13	0	0	0	0
13	14	0	0	0	0
14	15	2	30	0	0
15	16	0	0	0	0
総計		2048	4061	2048	3311

【0171】

これに固定長符号化のビット数を加算すると、従来方式では $m/2 = 12$ なので、 $3311 + 2048 \cdot 12 = 27887$ ビットとなる。一方、提案方式では $m/2 = 11.5$ なので、 $4061 + 2048 \cdot 11.5 = 27613$ ビットとなり、このフレームでは 274 ビット (0.84%) 節約することができる。

【図面の簡単な説明】

【0172】

【図1】エンコーダ（符号化器）の構成を示すブロック図である。

【図2】エントロピー符号化部の構成を示すブロック図である。

【図3】エントロピー符号化部の処理を示すフローチャートである。

【図4】デコーダ（復号化器）の構成を示すブロック図である。

【図5】エントロピー復号化部の構成を示すブロック図である。

10

20

30

40

50

【図6】エントロピー復号化部の処理を示すフローチャートである。

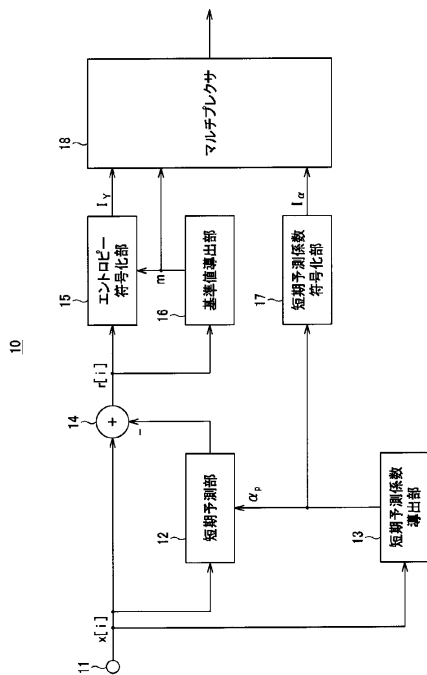
【符号の説明】

【0173】

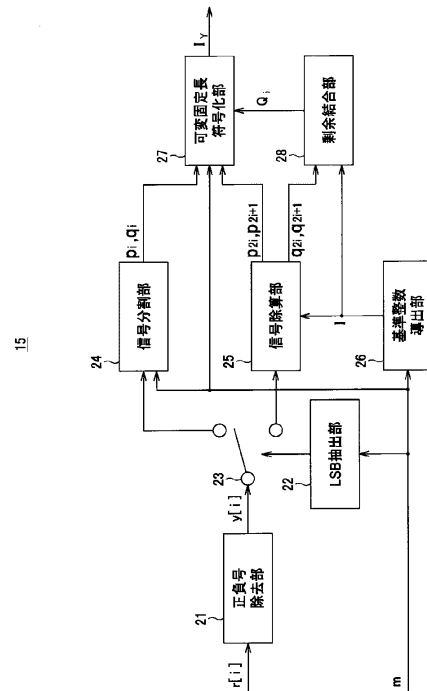
10 エンコーダ、11 入力端子、12 短期予測部、13 短期予測係数導出部、14 加算器、15 エントロピー符号化部、16 基準値導出部、17 短期予測係数符号化部、18 マルチプレクサ、21 正負号除去部、22 LSB抽出部、23 スイッチ、24 信号分割部、25 信号除算部、26 基準整数導出部、27 可変固定長符号化部、28 剰余結合部、41 デマルチプレクサ、42 エントロピー復号化部、43 短期予測係数復号化部、44 加算器、45 短期合成部、51 可変固定長復号化部、52 信号結合部、53 基準整数導出部、54 剰余分割部、55 信号乗算部、56 LSB抽出部、57 スイッチ、58 正負号復元部

10

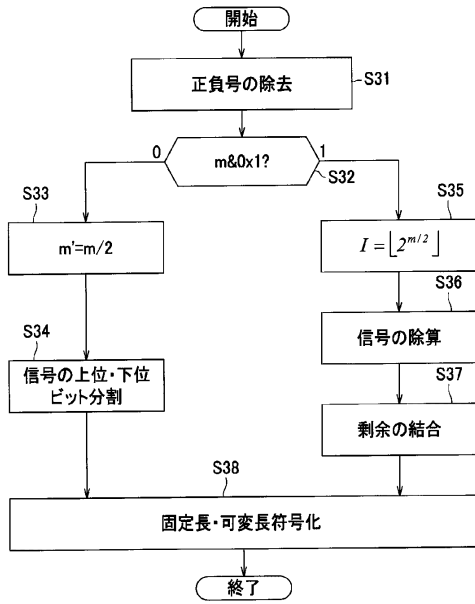
【図1】



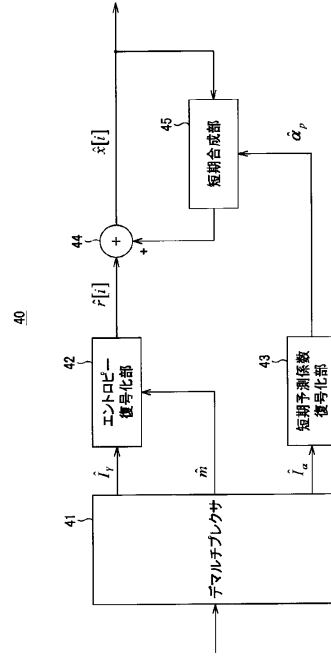
【図2】



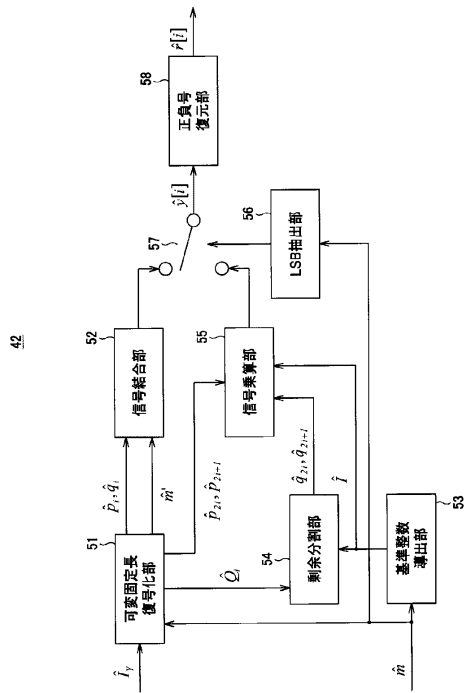
【図3】



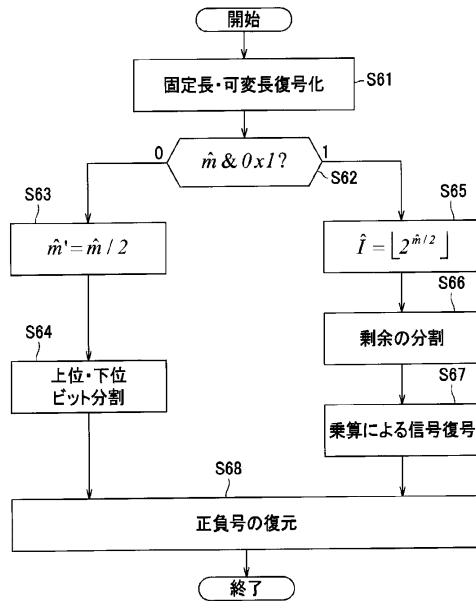
【図4】



【図5】



【図6】



フロントページの続き

(56)参考文献 特開2004-258059(JP,A)
特開平07-111633(JP,A)
特開平11-338853(JP,A)

(58)調査した分野(Int.Cl., DB名)

H03M3/00-11/00
G10L 19/00
H04N 7/26