

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第3923143号

(P3923143)

(45) 発行日 平成19年5月30日(2007.5.30)

(24) 登録日 平成19年3月2日(2007.3.2)

(51) Int. Cl.		F I		
<b>G 1 1 B</b>	<b>20/14</b>	<b>(2006.01)</b>	G 1 1 B	20/14 3 4 1 A
<b>H O 3 M</b>	<b>7/14</b>	<b>(2006.01)</b>	H O 3 M	7/14 B
<b>H O 4 L</b>	<b>25/49</b>	<b>(2006.01)</b>	H O 4 L	25/49 A

請求項の数 14 (全 57 頁)

<p>(21) 出願番号 特願平9-203499                  (22) 出願日 平成9年7月29日(1997.7.29)                  (65) 公開番号 特開平10-134520                  (43) 公開日 平成10年5月22日(1998.5.22)                  審査請求日 平成16年7月28日(2004.7.28)                  (31) 優先権主張番号 08/681,692                  (32) 優先日 平成8年7月29日(1996.7.29)                  (33) 優先権主張国 米国(US)</p>	<p>(73) 特許権者 502389641                  ブロードコム・コーポレーション                  アメリカ合衆国、カリフォルニア・926                  19-7013、ピー・オー・ボックス・                  57013・アービン、アルトン・パーク                  ウェイ・16215                  (74) 代理人 110000338                  特許業務法人原謙三国際特許事務所                  (72) 発明者 クリストファー ピー. ズーク                  アメリカ合衆国 コロラド 80503,                  ロングモント, セージ バリー ロー                  ド 8901                  審査官 深沢 正志</p>
--	---

最終頁に続く

(54) 【発明の名称】 17ビットのコードワードを16ビットのデータワードに復号化する16/17 ENDECおよび17ビットのコードワードを16ビットのデータワードに復号化する方法

(57) 【特許請求の範囲】

【請求項1】

nビットのコードワードをmビットのデータワードに復号化するm/n ENDECにおいて、

該nビットのコードワードのうちの所定の第1の数のビットを該データワードの第1のシンボルに復号化し、かつ該nビットのコードワードのうちの所定の第2の数のビットを該データワードの第2のシンボルに復号化するデコーダを備えている、m/n ENDECであって、

(a) 該デコーダが、該データワードの該第1のシンボルを、該データワードの該第2のシンボルの復号化とは独立して復号化し、かつ、該デコーダが、該データワードの該第2のシンボルを、該データワードの該第1のシンボルの復号化とは独立して復号化し、

(b) 該データワードの該第1のシンボルを復号化する時の誤りが、該データワードの該第2のシンボルを復号化する時の誤りをもたらさず、かつ、該データワードの該第2のシンボルを復号化する時の誤りが、該データワードの該第1のシンボルを復号化する時の誤りをもたらさず、

前記デコーダが、前記データワードの前記第1のシンボルおよび前記第2のシンボルを復号化する時に、同一の復号化ロジックに従って動作する、m/n ENDEC。

【請求項2】

nビットのコードワードをmビットのデータワードに復号化するm/n ENDECにおいて、

10

20

該  $n$  ビットのコードワードのうちの所定の第 1 の数のビットを該データワードの第 1 のシンボルに復号化し、かつ該  $n$  ビットのコードワードのうちの所定の第 2 の数のビットを該データワードの第 2 のシンボルに復号化するデコーダを備えている、 $m/n$  ENDEC であって、

(a) 該デコーダが、該データワードの該第 1 のシンボルを、該データワードの該第 2 のシンボルの復号化とは独立して復号化し、かつ、該デコーダが、該データワードの該第 2 のシンボルを、該データワードの該第 1 のシンボルの復号化とは独立して復号化し、

(b) 該データワードの該第 1 のシンボルを復号化する時の誤りが、該データワードの該第 2 のシンボルを復号化する時の誤りをもたらさず、かつ、該データワードの該第 2 のシンボルを復号化する時の誤りが、該データワードの該第 1 のシンボルを復号化する時の誤りをもたらさず、

10

前記デコーダが、以下の表 1 A ~ 表 1 G に従って動作する、 $m/n$  ENDEC。

【表 1 A】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000000	01001001	10000000	01010001
00000001	10101001	10000001	10010010

20

【表 1 B】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000010	10101110	10000010	10001101
00000011	10100101	10000011	01110101
00000100	00110101	10000100	01100001
00000101	00101011	10000101	01011110
00000110	01010110	10000110	10101000
00000111	01010010	10000111	01101001
00001000	00100100	10001000	01011000
00001001	00111010	10001001	00101111
00001010	10111110	10001010	00011100
00001011	00100010	10001011	01001111
00001100	01010011	10001100	10100110
00001101	10100111	10001101	10001011
00001110	11011001	10001110	01000011
00001111	01001010	10001111	01011010
00010000	11101010	10010000	00111011
00010001	00111001	10010001	00001101
00010010	10111100	10010010	11111100
00010011	00101110	10010011	00001011
00010100	11101000	10010100	00111100
00010101	11100111	10010101	00110000
00010110	10111000	10010110	00010000
00010111	11100101	10010111	00110111
00011000	00101101	10011000	10011010
00011001	10100001	10011001	01111101
00011010	01111010	10011010	11101100

10

20

30

40

【表 1 C】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00011011	10011101	10011011	01110110
00011100	11011010	10011100	00110100
00011101	00100111	10011101	11111010
00011110	11101001	10011110	10011001
00011111	10101010	10011111	01011001
00100000	00010010	10100000	01101110
00100001	00100001	10100001	00110001
00100010	10110000	10100010	00011110
00100011	00010001	10100011	00101100
00100100	10111111	10100100	11111000
00100101	10001111	10100101	11110000
00100110	11100001	10100110	00001110
00100111	00001001	10100111	00100011
00101000	11011111	10101000	00001000
00101001	10011110	10101001	00011000
00101010	00111111	10101010	10000000
00101011	11010000	10101011	00000100
00101100	10011100	10101100	11100110
00101101	11000111	10101101	00000010
00101110	10011000	10101110	00001100
00101111	11010111	10101111	01100100
00110000	11010001	10110000	10010110
00110001	10010111	10110001	01100010
00110010	10010000	10110010	11111011
00110011	10010100	10110011	01011111

10

20

30

40

【表 1 D】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00110100	10001110	10110100	11110011
00110101	10001000	10110101	11101111
00110110	01111001	10110110	11011110
00110111	10001010	10110111	01010000
00111000	11010100	10111000	00011010
00111001	11110101	10111001	11111001
00111010	11000001	10111010	00000110
00111011	11101101	10111011	11110111
00111100	11011101	10111100	10010011
00111101	01110010	10111101	01001100
00111110	11000101	10111110	00011001
00111111	10010101	10111111	01010111
01000000	10111011	11000000	01000101
01000001	11010110	11000001	01010100
01000010	10001100	11000010	01000110
01000011	11010011	11000011	01001101
01000100	10000011	11000100	00010110
01000101	00011111	11000101	00010011
01000110	10100000	11000110	10000101
01000111	11001101	11000111	01001011
01001000	00000111	11001000	00000101
01001001	00001111	11001001	11110100
01001010	11111111	11001010	11100011
01001011	00000001	11001011	11110010
01001100	10011111	11001100	01110111

10

20

30

40

【表 1 E】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01001101	10000110	11001101	00110010
01001110	01111110	11001110	11101110
01001111	11001011	11001111	00101010
01010000	10000111	11010000	00101000
01010001	01111100	11010001	11100100
01010010	00000000	11010010	01111111
01010011	01110000	11010011	11100010
01010100	11100000	11010100	11001111
01010101	11000000	11010101	11000011
01010110	00000011	11010110	11011000
01010111	10000001	11010111	00100110
01011000	10000100	11011000	01110100
01011001	01111011	11011001	00011101
01011010	11111110	11011010	11001100
01011011	01101111	11011011	11101011
01011100	01000000	11011100	11011100
01011101	01111000	11011101	11110001
01011110	10000010	11011110	01110001
01011111	10110111	11011111	11010010
01100000	00010101	11100000	00101001
01100001	01101101	11100001	11001001
01100010	01101100	11100010	00010100
01100011	01011011	11100011	10111010
01100100	01100011	11100100	11001110
01100101	00110011	11100101	11001000

10

20

30

40

【表 1 F】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01100110	01110011	11100110	01101011
01100111	01001110	11100111	10111001
01101000	01100110	11101000	00001010
01101001	00011011	11101001	11000100
01101010	01100000	11101010	11000110
01101011	00111101	11101011	00010111
01101100	01101000	11101100	01011100
01101101	00110110	11101101	11011011
01101110	11111101	11101110	11000010
01101111	01001000	11101111	10101011
01110000	10110100	11110000	00100101
01110001	10111101	11110001	10100100
01110010	00111110	11110010	10110011
01110011	10110110	11110011	10100010
01110100	00111000	11110100	10101111
01110101	00100000	11110101	10011011
01110110	11110110	11110110	01000111
01110111	10101100	11110111	10010001
01111000	11001010	11111000	11010101
01111001	01000100	11111001	10001001
01111010	01100111	11111010	01000001
01111011	01000010	11111011	01101010
01111100	10110001	11111100	10110101
01111101	10100011	11111101	01100101
01111110	10110010	11111110	10101101

10

20

30

40

【表 1 G】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01111111	01011101	11111111	01010101

【請求項 3】

50

$n$ ビットのコードワードを $m$ ビットのデータワードに復号化する $m/n$  ENDECにおいて、

該 $n$ ビットのコードワードのうちの所定の第1の数のビットを該データワードの第1のシンボルに復号化し、かつ該 $n$ ビットのコードワードのうちの所定の第2の数のビットを該データワードの第2のシンボルに復号化するデコーダを備えている、 $m/n$  ENDECであって、

(a) 該デコーダが、該データワードの該第1のシンボルを、該データワードの該第2のシンボルの復号化とは独立して復号化し、かつ、該デコーダが、該データワードの該第2のシンボルを、該データワードの該第1のシンボルの復号化とは独立して復号化し、

(b) 該データワードの該第1のシンボルを復号化する時の誤りが、該データワードの該第2のシンボルを復号化する時の誤りをもたさず、かつ、該データワードの該第2のシンボルを復号化する時の誤りが、該データワードの該第1のシンボルを復号化する時の誤りをもたさず、

復号化する前に、前記コードワードのうちの所定の数のビットをビット反転するビット反転回路をさらに備えている、 $m/n$  ENDEC。

【請求項4】

(a) 前記 $n$ ビットのコードワードのうちのマッピングビットを問い合わせる手段と、

(b) 該マッピングビットに応答して、前記データワードの前記第1のシンボルおよび該データワードの前記第2のシンボルを選択的に復号化する手段と、

をさらに備えている、請求項1~3のいずれか1項に記載の $m/n$  ENDEC。

【請求項5】

前記デコーダが、前記コードワードのうちの9ビットを前記データワードの8ビットに復号化する、請求項1~3のいずれか1項に記載の $m/n$  ENDEC。

【請求項6】

$n = 17$ および前記マッピングビットが、前記コードワードの中央ビットである、請求項4に記載の $m/n$  ENDEC。

【請求項7】

複数のコードワードから構成される、受け取られたデータストリームの両インタリーブにおいて、 $k$ ビットの中に非ゼロのサンプル値が発生するように、前記ユーザデータを符号化する手段をさらに備えている、請求項6に記載の $m/n$  ENDEC。

【請求項8】

$n$ ビットのコードワードを $m$ ビットのデータワードに復号化する方法であって、

(a) 前記 $n$ ビットのコードワードのうちの所定の第1の数のビットを前記 $m$ ビットのデータワードの第1のシンボルに復号化する、第1の復号化ステップと、

(b) 該 $n$ ビットのコードワードのうちの所定の第2の数のビットを該データワードの第2のシンボルに復号化する、第2の復号化ステップであって、該データワードの該第1のシンボルを復号化する時の誤りが、該データワードの該第2のシンボルを復号化する時の誤りをもたさないように、該第1の復号化ステップからは独立して動作する、第2の復号化ステップとを含み、

前記第1および第2の復号化ステップが、同一の復号化ロジックに従って動作する、方法。

【請求項9】

$n$ ビットのコードワードを $m$ ビットのデータワードに復号化する方法であって、

(a) 前記 $n$ ビットのコードワードのうちの所定の第1の数のビットを前記 $m$ ビットのデータワードの第1のシンボルに復号化する、第1の復号化ステップと、

(b) 該 $n$ ビットのコードワードのうちの所定の第2の数のビットを該データワードの第2のシンボルに復号化する、第2の復号化ステップであって、該データワードの該第1のシンボルを復号化する時の誤りが、該データワードの該第2のシンボルを復号化する時の誤りをもたさないように、該第1の復号化ステップからは独立して動作する、第2の復号化ステップとを含み、

10

20

30

40

50

前記第 1 および第 2 の復号化ステップが、以下の表 2 A ~ 表 2 G に従って動作する、方法。

【表 2 A】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000000	01001001	10000000	01010001
00000001	10101001	10000001	10010010

【表 2 B】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000010	10101110	10000010	10001101
00000011	10100101	10000011	01110101
00000100	00110101	10000100	01100001
00000101	00101011	10000101	01011110
00000110	01010110	10000110	10101000
00000111	01010010	10000111	01101001
00001000	00100100	10001000	01011000
00001001	00111010	10001001	00101111
00001010	10111110	10001010	00011100
00001011	00100010	10001011	01001111
00001100	01010011	10001100	10100110
00001101	10100111	10001101	10001011
00001110	11011001	10001110	01000011
00001111	01001010	10001111	01011010
00010000	11101010	10010000	00111011
00010001	00111001	10010001	00001101
00010010	10111100	10010010	11111100
00010011	00101110	10010011	00001011
00010100	11101000	10010100	00111100
00010101	11100111	10010101	00110000
00010110	10111000	10010110	00010000
00010111	11100101	10010111	00110111
00011000	00101101	10011000	10011010
00011001	10100001	10011001	01111101
00011010	01111010	10011010	11101100

10

20

30

40

【表 2 C】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00011011	10011101	10011011	01110110
00011100	11011010	10011100	00110100
00011101	00100111	10011101	11111010
00011110	11101001	10011110	10011001
00011111	10101010	10011111	01011001
00100000	00010010	10100000	01101110
00100001	00100001	10100001	00110001
00100010	10110000	10100010	00011110
00100011	00010001	10100011	00101100
00100100	10111111	10100100	11111000
00100101	10001111	10100101	11110000
00100110	11100001	10100110	00001110
00100111	00001001	10100111	00100011
00101000	11011111	10101000	00001000
00101001	10011110	10101001	00011000
00101010	00111111	10101010	10000000
00101011	11010000	10101011	00000100
00101100	10011100	10101100	11100110
00101101	11000111	10101101	00000010
00101110	10011000	10101110	00001100
00101111	11010111	10101111	01100100
00110000	11010001	10110000	10010110
00110001	10010111	10110001	01100010
00110010	10010000	10110010	11111011
00110011	10010100	10110011	01011111

10

20

30

40

【表 2 D】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00110100	10001110	10110100	11110011
00110101	10001000	10110101	11101111
00110110	01111001	10110110	11011110
00110111	10001010	10110111	01010000
00111000	11010100	10111000	00011010
00111001	11110101	10111001	11111001
00111010	11000001	10111010	00000110
00111011	11101101	10111011	11110111
00111100	11011101	10111100	10010011
00111101	01110010	10111101	01001100
00111110	11000101	10111110	00011001
00111111	10010101	10111111	01010111
01000000	10111011	11000000	01000101
01000001	11010110	11000001	01010100
01000010	10001100	11000010	01000110
01000011	11010011	11000011	01001101
01000100	10000011	11000100	00010110
01000101	00011111	11000101	00010011
01000110	10100000	11000110	10000101
01000111	11001101	11000111	01001011
01001000	00000111	11001000	00000101
01001001	00001111	11001001	11110100
01001010	11111111	11001010	11100011
01001011	00000001	11001011	11110010
01001100	10011111	11001100	01110111

10

20

30

40

【表 2 E】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01001101	10000110	11001101	00110010
01001110	01111110	11001110	11101110
01001111	11001011	11001111	00101010
01010000	10000111	11010000	00101000
01010001	01111100	11010001	11100100
01010010	00000000	11010010	01111111
01010011	01110000	11010011	11100010
01010100	11100000	11010100	11001111
01010101	11000000	11010101	11000011
01010110	00000011	11010110	11011000
01010111	10000001	11010111	00100110
01011000	10000100	11011000	01110100
01011001	01111011	11011001	00011101
01011010	11111110	11011010	11001100
01011011	01101111	11011011	11101011
01011100	01000000	11011100	11011100
01011101	01111000	11011101	11110001
01011110	10000010	11011110	01110001
01011111	10110111	11011111	11010010
01100000	00010101	11100000	00101001
01100001	01101101	11100001	11001001
01100010	01101100	11100010	00010100
01100011	01011011	11100011	10111010
01100100	01100011	11100100	11001110
01100101	00110011	11100101	11001000

10

20

30

40

【表 2 F】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01100110	01110011	11100110	01101011
01100111	01001110	11100111	10111001
01101000	01100110	11101000	00001010
01101001	00011011	11101001	11000100
01101010	01100000	11101010	11000110
01101011	00111101	11101011	00010111
01101100	01101000	11101100	01011100
01101101	00110110	11101101	11011011
01101110	11111101	11101110	11000010
01101111	01001000	11101111	10101011
01110000	10110100	11110000	00100101
01110001	10111101	11110001	10100100
01110010	00111110	11110010	10110011
01110011	10110110	11110011	10100010
01110100	00111000	11110100	10101111
01110101	00100000	11110101	10011011
01110110	11110110	11110110	01000111
01110111	10101100	11110111	10010001
01111000	11001010	11111000	11010101
01111001	01000100	11111001	10001001
01111010	01100111	11111010	01000001
01111011	01000010	11111011	01101010
01111100	10110001	11111100	10110101
01111101	10100011	11111101	01100101
01111110	10110010	11111110	10101101

10

20

30

40

【表 2 G】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01111111	01011101	11111111	01010101

【請求項 10】

50

nビットのコードワードをmビットのデータワードに復号化する方法であって、

( a ) 前記 n ビットのコードワードのうちの所定の第 1 の数のビットを前記 m ビットのデータワードの第 1 のシンボルに復号化する、第 1 の復号化ステップと、

( b ) 該 n ビットのコードワードのうちの所定の第 2 の数のビットを該データワードの第 2 のシンボルに復号化する、第 2 の復号化ステップであって、該データワードの該第 1 のシンボルを復号化する時の誤りが、該データワードの該第 2 のシンボルを復号化する時の誤りをもたらさないように、該第 1 の復号化ステップからは独立して動作する、第 2 の復号化ステップとを含み、

前記コードワードのうちの所定数のビットをビット反転させるステップをさらに含んでいる、方法。

10

【請求項 1 1】

前記第 1 および第 2 の復号化ステップが、前記コードワードの 9 ビットを前記データワードの 8 ビットに復号化する、請求項 8 ~ 1 0 のいずれか 1 項に記載の方法。

【請求項 1 2】

( a ) 前記 n ビットコードワードのマッピングビットを問い合わせるステップと、

( b ) 該マッピングビットにตอบสนองして、前記データワードの前記第 1 のシンボルおよび該データワードの前記第 2 のシンボルを選択的に復号化するステップと、  
をさらに含んでいる、請求項 8 ~ 1 0 のいずれか 1 項に記載の方法。

【請求項 1 3】

n = 1 7 および前記マッピングビットが、前記 1 7 ビットコードワードの中央ビットである、請求項 1 2 に記載の方法。

20

【請求項 1 4】

複数のコードワードから構成される、受け取られたデータストリームの両インタリーブにおいて、k ビット中に非ゼロのサンプル値が発生するように、前記ユーザデータを符号化するステップをさらに含んでいる、請求項 1 3 に記載の方法。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、磁気ディスクドライブや光ディスクドライブを有するコンピュータ記憶システムに関する。具体的には、1 コードワードの上位バイトおよび下位バイトを互いに独立して復号化することによって誤りの伝搬を防止する、効率のよい 1 6 / 1 7 エンコーダ/デコーダ ( E N D E C ) に関する。

30

【0 0 0 2】

【従来の技術】

デジタルコンピュータ用の磁気ディスク記憶システムにおいて、(典型的には、誘導性コイル型の) ライトヘッドは、複数の 2 進「1」値および「0」値から構成されるシーケンスを記録するために、磁束の遷移を磁気ディスクの表面上にライトする。記録されたこの 2 進データをリードするために、回転する磁気ディスクの近傍に位置づけられたリードヘッドは、磁束の遷移を検出し、対応するパルスをアナログリード信号に発生する。これらのパルスは、それからリードチャネル回路により、推定された 2 進シーケンスに検出・復号化される。誤差がない場合、推定された 2 進シーケンスは、記録されている 2 進シーケンスに等しいが、推定された 2 進シーケンス内に誤差をもたらす原因になる、アナログリード信号の信号対雑音比の低下を招く数多くの要因がある。

40

【0 0 0 3】

パルス検出処理時の誤差は、符号間干渉 ( I S I ) および/またはチャネルノイズによりもたらされることが最も多い。従来、リードチャネルは、アナログリード信号におけるピークを検出する簡単なアナログピーク検出器のかたちで実現されてきた。しかし、I S I およびチャネルノイズが増大すると、アナログピーク検出器の精度も急速に低下する。この悪影響を緩和するために、最近の記憶システムは、I S I およびチャネルノイズが検出アルゴリズムに与える影響を考慮に入れた、さらに高度なデジタル信号処理回路を有す

50

るサンプリングされた振幅リードチャネルを用いている。

【0004】

この目的のために、サンプリングされた振幅リードチャネルは、アナログリード信号をサンプリングすることによって、離散時間サンプル値のシーケンスを生成するサンプリングデバイスと、サンプル値をポーレート（符号ビットレート）に同期させるタイミングリカバリとを備えている。その後、シーケンス検出器が、それらの同期サンプル値を文脈に従って推定することによって、推定された2進データについて最尤（most likely）のシーケンスを選択する。公知の離散時間シーケンス検出方法としては、最尤シーケンス検出（MLSD）、ピタピ検出（MLSDの近似）を用いたパーシャルレスポンス（PR）、判定帰還等化（DFE）、改善された判定帰還等化（EDFE）、および判定帰還を用いた固定遅延ツリー検索（FDT S / DF）などが挙げられる。

10

【0005】

従来のピーク検出においても、最近のサンプリングされた振幅リードチャネルにおいても、通常は、タイミングリカバリおよび利得制御が適正に動作できるように、連続する「0」ビットの個数を制限する制約に従って、ユーザデータを符号化する。また、ユーザデータは、連続する「1」ビットと「1」ビットとの間の間隔を制約するように（すなわち、「1」ビットと「1」ビットとの間の「0」ビットの個数を確実に最小化できるように）も符号化されうる。後者の制約は、ISIがアナログピーク検出リードチャネルに与える影響を緩和し、離散時間リードチャネルにおける最小距離誤り事象を増大させるとともに、シーケンス検出器のコストを低下させ、その複雑さを緩和する。よって、ユーザデータは、ランレングス制限（RL L）された符号（通常、RL L（d, k）と表される）に従って符号化される。ここで、dは、連続する「0」ビットの最大数を表しており、kは、連続する「1」ビットと「1」ビットとの間の「0」ビットの最小数を表している。

20

【0006】

RL L符号化は、符号化処理に伴うオーバーヘッド（「レートの低下」として知られている）が原因で、記憶システムの効率を低下させる。RL Lを導入することによって、必然的に、mビットのユーザデータをnビットの（ただし、 $m < n$ ）コードワードに符号化することが必要になる。従来、設計者は、レート8/9 ENDECを用いていた。つまり、8ビットのユーザデータを、所望のRL L制約を満たす9ビットのコードワードに符号化する、ENDECを用いていた。サンプリングされた振幅リードチャネルに用いられた8/9 ENDECは、例えば、「ゾーン分けされたデータ記録およびデジタル適応等化によるPRMLサンプリングデータ検出を用いたディスク駆動方法」と題された米国特許第5,422,760号に開示されている。

30

【0007】

高レートENDECが用いられない理由としては、少なくとも以下の2つが考えられる。

【0008】

1. 例えば、16/17 ENDECのような高レートENDECは、コスト面で有利に実施するには、あまりにも複雑である。

【0009】

2. リード時に検出されたシーケンスに誤りがあると、その誤りは、復号化された出力の多数のバイトに伝搬する。

40

【0010】

第1の欠点については、この問題に対処した高レートENDECの従来技術例が少なくとも1つある。IBMのTechnical Disclosure Bulletin、第31巻、第8号、1989年1月には、コードワード内のビット位置に関する符号制約で均等性を活用することによって、符号化/復号化ロジックを大幅に縮小する、レート16/17 ENDECが開示されている。しかし、この文献に開示されているレート16/17 ENDECは、上述した第2の問題点には対処していない。すなわち、復号化処理時に誤りが伝搬することを防止できない。もしあるコードワードの下位バイトに誤りが発生すれば、デコードは、その誤りを復号化された出力の上位バイトにまで伝搬してしまう。同様に、コードワードの上

50

位バイト内の誤りも、復号化処理中に、復号化されたデータワードの下位バイトに伝搬してしまう。

【0011】

【発明が解決しようとする課題】

よって、実施するのがあまり複雑ではなく、復号化された出力における誤りの伝搬を防止できる、レート16/17 ENDECがいま必要とされている。

【0012】

本発明は、上記課題を解決するためになされたものであり、その目的は、実施するのがあまり複雑ではなく、復号化された出力における誤りの伝搬を防止できる、レート16/17 ENDECを提供することである。

10

【0013】

【課題を解決するための手段】

本発明による17ビットのコードワードを16ビットのデータワードに復号化する16/17 ENDECは、該17ビットのコードワードのうちの所定の第1の数のビットを該データワードの8ビットの上位バイトに復号化し、かつ該17ビットのコードワードのうちの所定の第2の数のビットを該データワードの8ビットの下位バイトに復号化するデコーダを備えている、16/17 ENDECであって、(a)該デコーダが、該データワードの該上位バイトを、該データワードの該下位バイトの復号化とは独立して復号化し、かつ、該デコーダが、該データワードの該下位バイトを、該データワードの該上位バイトの復号化とは独立して復号化し、(b)該データワードの該上位バイトを復号化する時の誤りが、該データワードの該下位バイトを復号化する時の誤りをもたらさず、かつ、該データワードの該下位バイトを復号化する時の誤りが、該データワードの該上位バイトを復号化する時の誤りをもたらさず、そのことにより上記目的が達成される。

20

【0014】

ある実施形態では、前記デコーダが、前記データワードの前記上位バイトおよび前記下位バイトを復号化する時に、同一の復号化ロジックに従って動作する。

【0015】

ある実施形態では、前記デコーダが、以下の表3A~表3Gに従って動作する。

【0016】

【表3A】

30

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000000	01001001	10000000	01010001
00000001	10101001	10000001	10010010

【0017】

【表3B】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000010	10101110	10000010	10001101
00000011	10100101	10000011	01110101
00000100	00110101	10000100	01100001
00000101	00101011	10000101	01011110
00000110	01010110	10000110	10101000
00000111	01010010	10000111	01101001
00001000	00100100	10001000	01011000
00001001	00111010	10001001	00101111
00001010	10111110	10001010	00011100
00001011	00100010	10001011	01001111
00001100	01010011	10001100	10100110
00001101	10100111	10001101	10001011
00001110	11011001	10001110	01000011
00001111	01001010	10001111	01011010
00010000	11101010	10010000	00111011
00010001	00111001	10010001	00001101
00010010	10111100	10010010	11111100
00010011	00101110	10010011	00001011
00010100	11101000	10010100	00111100
00010101	11100111	10010101	00110000
00010110	10111000	10010110	00010000
00010111	11100101	10010111	00110111
00011000	00101101	10011000	10011010
00011001	10100001	10011001	01111101
00011010	01111010	10011010	11101100

10

20

30

40

【 0 0 1 8 】

【 表 3 C 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00011011	10011101	10011011	01110110
00011100	11011010	10011100	00110100
00011101	00100111	10011101	11111010
00011110	11101001	10011110	10011001
00011111	10101010	10011111	01011001
00100000	00010010	10100000	01101110
00100001	00100001	10100001	00110001
00100010	10110000	10100010	00011110
00100011	00010001	10100011	00101100
00100100	10111111	10100100	11111000
00100101	10001111	10100101	11110000
00100110	11100001	10100110	00001110
00100111	00001001	10100111	00100011
00101000	11011111	10101000	00001000
00101001	10011110	10101001	00011000
00101010	00111111	10101010	10000000
00101011	11010000	10101011	00000100
00101100	10011100	10101100	11100110
00101101	11000111	10101101	00000010
00101110	10011000	10101110	00001100
00101111	11010111	10101111	01100100
00110000	11010001	10110000	10010110
00110001	10010111	10110001	01100010
00110010	10010000	10110010	11111011
00110011	10010100	10110011	01011111

10

20

30

40

【 0 0 1 9 】

【 表 3 D 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00110100	10001110	10110100	11110011
00110101	10001000	10110101	11101111
00110110	01111001	10110110	11011110
00110111	10001010	10110111	01010000
00111000	11010100	10111000	00011010
00111001	11110101	10111001	11111001
00111010	11000001	10111010	00000110
00111011	11101101	10111011	11110111
00111100	11011101	10111100	10010011
00111101	01110010	10111101	01001100
00111110	11000101	10111110	00011001
00111111	10010101	10111111	01010111
01000000	10111011	11000000	01000101
01000001	11010110	11000001	01010100
01000010	10001100	11000010	01000110
01000011	11010011	11000011	01001101
01000100	10000011	11000100	00010110
01000101	00011111	11000101	00010011
01000110	10100000	11000110	10000101
01000111	11001101	11000111	01001011
01001000	00000111	11001000	00000101
01001001	00001111	11001001	11110100
01001010	11111111	11001010	11100011
01001011	00000001	11001011	11110010
01001100	10011111	11001100	01110111

10

20

30

40

【 0 0 2 0 】

【 表 3 E 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01001101	10000110	11001101	00110010
01001110	01111110	11001110	11101110
01001111	11001011	11001111	00101010
01010000	10000111	11010000	00101000
01010001	01111100	11010001	11100100
01010010	00000000	11010010	01111111
01010011	01110000	11010011	11100010
01010100	11100000	11010100	11001111
01010101	11000000	11010101	11000011
01010110	00000011	11010110	11011000
01010111	10000001	11010111	00100110
01011000	10000100	11011000	01110100
01011001	01111011	11011001	00011101
01011010	11111110	11011010	11001100
01011011	01101111	11011011	11101011
01011100	01000000	11011100	11011100
01011101	01111000	11011101	11110001
01011110	10000010	11011110	01110001
01011111	10110111	11011111	11010010
01100000	00010101	11100000	00101001
01100001	01101101	11100001	11001001
01100010	01101100	11100010	00010100
01100011	01011011	11100011	10111010
01100100	01100011	11100100	11001110
01100101	00110011	11100101	11001000

10

20

30

40

【 0 0 2 1 】

【 表 3 F 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01100110	01110011	11100110	01101011
01100111	01001110	11100111	10111001
01101000	01100110	11101000	00001010
01101001	00011011	11101001	11000100
01101010	01100000	11101010	11000110
01101011	00111101	11101011	00010111
01101100	01101000	11101100	01011100
01101101	00110110	11101101	11011011
01101110	11111101	11101110	11000010
01101111	01001000	11101111	10101011
01110000	10110100	11110000	00100101
01110001	10111101	11110001	10100100
01110010	00111110	11110010	10110011
01110011	10110110	11110011	10100010
01110100	00111000	11110100	10101111
01110101	00100000	11110101	10011011
01110110	11110110	11110110	01000111
01110111	10101100	11110111	10010001
01111000	11001010	11111000	11010101
01111001	01000100	11111001	10001001
01111010	01100111	11111010	01000001
01111011	01000010	11111011	01101010
01111100	10110001	11111100	10110101
01111101	10100011	11111101	01100101
01111110	10110010	11111110	10101101

10

20

30

40

【 0 0 2 2 】

【 表 3 G 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01111111	01011101	11111111	01010101

## 【 0 0 2 3 】

ある実施形態では、( a ) 前記 17 ビットのコードワードのうちのマッピングビットを問い合わせる手段と、( b ) 該マッピングビットに応答して、前記データワードの前記上位バイトおよび該データワードの前記下位バイトを選択的に復号化する手段と、をさらに備えている。

10

## 【 0 0 2 4 】

ある実施形態では、復号化する前に、前記コードワードのうちの所定の数のビットをビット反転するビット反転回路をさらに備えている。

## 【 0 0 2 5 】

ある実施形態では、前記デコーダが、前記コードワードのうちの 9 ビットを前記データワードの 8 ビットに復号化する。

## 【 0 0 2 6 】

ある実施形態では、前記マッピングビットが、前記 17 ビットコードワードの中央ビットである。

20

## 【 0 0 2 7 】

ある実施形態では、複数のコードワードから構成される、受け取られたデータストリームの両インタリーブにおいて、現在のコードワードの中央ビットに続くビットから始まり、後続するコードワードの中央ビットで終わる 17 ビットの中に非ゼロのサンプル値が発生するように、前記ユーザデータを符号化する手段をさらに備えている。

## 【 0 0 2 8 】

本発明による 17 ビットのコードワードを 16 ビットのデータワードに復号化する方法は、( a ) 該 17 ビットのコードワードのうちの所定の第 1 の数のビットを該データワードの 8 ビットの第 1 のバイトに復号化する、第 1 の復号化ステップと、( b ) 該 17 ビットのコードワードのうちの所定の第 2 の数のビットを該データワードの 8 ビットの第 2 のバイトに復号化する、第 2 の復号化ステップであって、該データワードの該第 1 のバイトを復号化する時の誤りが、該データワードの該第 2 のバイトを復号化する時の誤りをもたらさないように、該第 1 の復号化ステップからは独立して動作する、第 2 の復号化ステップと、を含んでおり、そのことにより上記目的が達成される。

30

## 【 0 0 2 9 】

ある実施形態では、前記第 1 および第 2 の復号化ステップが、同一の復号化ロジックに従って動作する。

## 【 0 0 3 0 】

ある実施形態では、前記第 1 および第 2 の復号化ステップが、前記コードワードの 9 ビットを前記データワードの 8 ビットに復号化する。

40

## 【 0 0 3 1 】

ある実施形態では、前記第 1 および第 2 の復号化ステップが、以下の表 4 A ~ 表 4 G に従って動作する。

## 【 0 0 3 2 】

## 【 表 4 A 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000000	01001001	10000000	01010001
00000001	10101001	10000001	10010010

【 0 0 3 3 】

【 表 4 B 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000010	10101110	10000010	10001101
00000011	10100101	10000011	01110101
00000100	00110101	10000100	01100001
00000101	00101011	10000101	01011110
00000110	01010110	10000110	10101000
00000111	01010010	10000111	01101001
00001000	00100100	10001000	01011000
00001001	00111010	10001001	00101111
00001010	10111110	10001010	00011100
00001011	00100010	10001011	01001111
00001100	01010011	10001100	10100110
00001101	10100111	10001101	10001011
00001110	11011001	10001110	01000011
00001111	01001010	10001111	01011010
00010000	11101010	10010000	00111011
00010001	00111001	10010001	00001101
00010010	10111100	10010010	11111100
00010011	00101110	10010011	00001011
00010100	11101000	10010100	00111100
00010101	11100111	10010101	00110000
00010110	10111000	10010110	00010000
00010111	11100101	10010111	00110111
00011000	00101101	10011000	10011010
00011001	10100001	10011001	01111101
00011010	01111010	10011010	11101100

10

20

30

40

【 0 0 3 4 】

【 表 4 C 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00011011	10011101	10011011	01110110
00011100	11011010	10011100	00110100
00011101	00100111	10011101	11111010
00011110	11101001	10011110	10011001
00011111	10101010	10011111	01011001
00100000	00010010	10100000	01101110
00100001	00100001	10100001	00110001
00100010	10110000	10100010	00011110
00100011	00010001	10100011	00101100
00100100	10111111	10100100	11111000
00100101	10001111	10100101	11110000
00100110	11100001	10100110	00001110
00100111	00001001	10100111	00100011
00101000	11011111	10101000	00001000
00101001	10011110	10101001	00011000
00101010	00111111	10101010	10000000
00101011	11010000	10101011	00000100
00101100	10011100	10101100	11100110
00101101	11000111	10101101	00000010
00101110	10011000	10101110	00001100
00101111	11010111	10101111	01100100
00110000	11010001	10110000	10010110
00110001	10010111	10110001	01100010
00110010	10010000	10110010	11111011
00110011	10010100	10110011	01011111

10

20

30

40

【 0 0 3 5 】

【 表 4 D 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00110100	10001110	10110100	11110011
00110101	10001000	10110101	11101111
00110110	01111001	10110110	11011110
00110111	10001010	10110111	01010000
00111000	11010100	10111000	00011010
00111001	11110101	10111001	11111001
00111010	11000001	10111010	00000110
00111011	11101101	10111011	11110111
00111100	11011101	10111100	10010011
00111101	01110010	10111101	01001100
00111110	11000101	10111110	00011001
00111111	10010101	10111111	01010111
01000000	10111011	11000000	01000101
01000001	11010110	11000001	01010100
01000010	10001100	11000010	01000110
01000011	11010011	11000011	01001101
01000100	10000011	11000100	00010110
01000101	00011111	11000101	00010011
01000110	10100000	11000110	10000101
01000111	11001101	11000111	01001011
01001000	00000111	11001000	00000101
01001001	00001111	11001001	11110100
01001010	11111111	11001010	11100011
01001011	00000001	11001011	11110010
01001100	10011111	11001100	01110111

10

20

30

40

【 0 0 3 6 】

【 表 4 E 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01001101	10000110	11001101	00110010
01001110	01111110	11001110	11101110
01001111	11001011	11001111	00101010
01010000	10000111	11010000	00101000
01010001	01111100	11010001	11100100
01010010	00000000	11010010	01111111
01010011	01110000	11010011	11100010
01010100	11100000	11010100	11001111
01010101	11000000	11010101	11000011
01010110	00000011	11010110	11011000
01010111	10000001	11010111	00100110
01011000	10000100	11011000	01110100
01011001	01111011	11011001	00011101
01011010	11111110	11011010	11001100
01011011	01101111	11011011	11101011
01011100	01000000	11011100	11011100
01011101	01111000	11011101	11110001
01011110	10000010	11011110	01110001
01011111	10110111	11011111	11010010
01100000	00010101	11100000	00101001
01100001	01101101	11100001	11001001
01100010	01101100	11100010	00010100
01100011	01011011	11100011	10111010
01100100	01100011	11100100	11001110
01100101	00110011	11100101	11001000

10

20

30

40

【 0 0 3 7 】

【 表 4 F 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01100110	01110011	11100110	01101011
01100111	01001110	11100111	10111001
01101000	01100110	11101000	00001010
01101001	00011011	11101001	11000100
01101010	01100000	11101010	11000110
01101011	00111101	11101011	00010111
01101100	01101000	11101100	01011100
01101101	00110110	11101101	11011011
01101110	11111101	11101110	11000010
01101111	01001000	11101111	10101011
01110000	10110100	11110000	00100101
01110001	10111101	11110001	10100100
01110010	00111110	11110010	10110011
01110011	10110110	11110011	10100010
01110100	00111000	11110100	10101111
01110101	00100000	11110101	10011011
01110110	11110110	11110110	01000111
01110111	10101100	11110111	10010001
01111000	11001010	11111000	11010101
01111001	01000100	11111001	10001001
01111010	01100111	11111010	01000001
01111011	01000010	11111011	01101010
01111100	10110001	11111100	10110101
01111101	10100011	11111101	01100101
01111110	10110010	11111110	10101101

10

20

30

40

【 0 0 3 8 】

【 表 4 G 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01111111	01011101	11111111	01010101

50

## 【 0 0 3 9 】

ある実施形態では、( a ) 前記 1 7 ビットコードワードのマッピングビットを問い合わせるステップと、( b ) 該マッピングビットに応答して、前記データワードの前記第 1 のバイトおよび該データワードの前記第 2 のバイトを選択的に復号化するステップと、をさらに含んでいる。

## 【 0 0 4 0 】

ある実施形態では、前記コードワードのうちの所定数のビットをビット反転させるステップをさらに含んでいる。

## 【 0 0 4 1 】

ある実施形態では、前記マッピングビットが、前記 1 7 ビットコードワードの中央ビットである。

## 【 0 0 4 2 】

ある実施形態では、複数のコードワードから構成される、受け取られたデータストリームの両インタリーブにおいて、現在のコードワードの中央ビットに続くビットから始まり、後続するコードワードの中央ビットで終わる 1 7 ビット中に非ゼロのサンプル値が発生するように、前記ユーザデータを符号化するステップをさらに含んでいる。

## 【 0 0 4 3 】

以下に作用を説明する。

## 【 0 0 4 4 】

少なくとも 1 つの所定の符号制約 (例えば、RLL ( d , k ) 制約) に従って 1 6 ビットの入力データワードを 1 7 ビットのコードワードに符号化する、レート 1 6 / 1 7 E N D E C が開示される。エンコーダは、同一のマッピング回路を用いて入力データワードの上位バイトおよび下位バイトを符号化することによって、コストを下げ、複雑さを緩和する。コードワードは、例えば、コンピュータディスクの記憶装置のような通信路を通して転送される。受信 (またはリード) されると、1 7 ビットのコードワードのうちの最初の 8 ビットが、最後の 8 ビットからは独立して復号化される。これにより、誤りの伝搬を防止する。すなわち、コードワードの最初の 8 ビットは、復号化されたデータワードの上位バイトに復号化され、コードワードの最後の 8 ビットは、復号化されたデータワードの下位バイトに復号化されるが、これらの復号化は、互いに独立しておこなわれる。よって、受け取られたデータシーケンスに発生しており、コードワードの最初の 8 ビットのみに影響を与える誤りは、復号化された下位バイトには伝搬せず、コードワードの最後の 8 ビットのみに影響を与えている誤りは、復号化された上位バイトに伝搬しない。また、デコーダは、同一のマッピング回路を用いて出力データワードの下位バイトおよび上位バイトを復号化することによって、その複雑さを緩和し、コストを低下させる。

## 【 0 0 4 5 】

## 【 発明の実施の形態 】

本願は、以下の同時係属中の米国特許出願、すなわち、「サンプル推定等化、欠陥スキミング、チャネル品質、デジタルサーボ復調、タイミングリカバリ用の P I D フィルタおよび D C オフセット制御を含むサンプリングされた振幅リードチャネル」と題された米国特許出願第 08/341,251 号、「同期パルシャルレスポンス記録のための改善されたタイミングリカバリ」と題された米国特許出願第 08/313,491 号、および「サンプリングされた振幅磁気記録のための改善された欠陥許容シンクマーク検出器」と題された米国特許出願第 08/533,797 号に関連している。本願は、また、以下に掲げるいくつかの米国特許、すなわち、「同期波形サンプリング用のタイミングリカバリ回路」と題された米国特許第 5,359,631 号、「複雑さの低減されたビタビ型シーケンス検出器のための方法および装置」と題された米国特許第 5,291,499 号、「同期波形サンプリングのための利得制御回路」と題された米国特許第 5,297,184 号、「デジタルパルス検出器」と題された米国特許第 5,329,554 号、および「同期リードチャネル」と題された米国特許第 5,424,881 号にも関連している。上に列挙した特許出願および特許のすべては、同一の譲受人に譲渡されており、本願でも、それらのすべてが参考として援用される。

10

20

30

40

50

【 0 0 4 6 】

本発明の上記局面および利点、ならびにその他の局面および利点は、添付の図面を参照しながら以下に述べる本発明の詳細な説明を読めば、よりよく理解できるであろう。

【 0 0 4 7 】

(従来のサンプリングされた振幅リードチャンネル)

図 1 をここで参照すれば、従来のサンプリングされた振幅リードチャンネルの詳細なブロック図が示されている。ライト動作のあいだ、ユーザデータ 2 またはデータ発生器 4 からのプリアンプルデータ (例えば 2 T のプリアンプルデータ) が、媒体上にライトされる。RLL エンコーダ 6 は、RLL 制約条件にしたがって、ユーザデータ 2 を 2 進シーケンス  $b(n)$  に符号化する。  $1 / (1 + D^2)$  プリコーダ 10 は、記録チャンネル 18 および等化器フィルタの伝達関数を補償し、ビット誤りの伝搬を防止するために、2 進シーケンス  $b(n)$  をプリコードする。プリコーダ 10 の出力 (プリコードされたシーケンス  $\sim b(n)$ ) は、  $\sim b(N) = 0$  を  $a(N) = -1$  に、かつ  $\sim b(N) = 1$  を  $a(N) = +1$  に変調する NRZ 変調器 14 によって、シンボル  $a(n)$  に変換される。ライト回路 9 は、シンボル  $a(n)$  に応答して、ポーレート  $1/T$  で記録ヘッドコイル中の電流を変調することによって、2 進シーケンスを媒体上に記録する。周波数シンセサイザ 52 は、ポーレートライトクロック 54 をライト回路 9 に供給する。また、周波数シンセサイザ 52 は、記録ヘッドがその上に位置しているゾーンに従ってチャンネルデータレート信号 (CDR) 30 により調整される。

10

【 0 0 4 8 】

記録された 2 進シーケンスを媒体からリードするとき、タイミングリカバリ 28 は、まず、リードチャンネルへの入力としてマルチプレクサ 60 を通したライトクロック 54 を選択することによって、ライト周波数にロックする。いったんライト周波数にロックされると、マルチプレクサ 60 は、記録されたユーザデータよりも前にディスク上に記録されたアキュジションプリアンプルを得るために、リードヘッドからの信号 19 をリードチャンネルへの入力として選択する。可変利得増幅器 22 は、アナログリード信号 58 の振幅を調整し、アナログフィルタ 20 は、エイリアシングノイズを減衰させるとともに、所望の応答を得るために初期等化をおこなう。サンプリングデバイス 24 は、アナログフィルタ 20 からのアナログリード信号 62 をサンプリングし、離散時間等化器フィルタ 26 は、所望の応答を得るためにサンプル値 25 をさらに等化する。パーシャルレスポンス記録においては、所望の応答は、例えば、しばしば以下の表 5 から選択される。

20

30

【 0 0 4 9 】

【表 5】

チャンネル	伝達関数	ダイパルス応答
PR4	$(1-D)(1+D)$	0, 1, 0, -1, 0, 0, 0, ...
EPR4	$(1-D)(1+D)^2$	0, 1, 1, -1, -1, 0, 0, ...
EEPR4	$(1-D)(1+D)^3$	0, 1, 2, 0, -2, -1, 0, ...

40

【 0 0 5 0 】

等化の後、リード信号 58 の振幅と、サンプリングデバイス 24 の周波数および位相とをそれぞれ調整するために、等化されたサンプル値 32 は、判定により向きの決められた利得制御 50 と、タイミングリカバリ回路 28 とに与えられる。タイミングリカバリは、等化されたサンプル 32 をポーレートに同期させるために、ライン 23 を介してサンプリングデバイス 24 の周波数を調整する。周波数シンセサイザ 52 は、温度、電圧およびプロセスのばらつきに対してタイミングリカバリ周波数を所定の値に揃えるために、センタ周波数の粗い設定値をライン 64 を介してタイミングリカバリ回路 28 に与える。チャンネルデータレート (CDR) 信号 30 は、シンセサイザ 52 の周波数範囲を、現在のゾーンに

50

ついでにデータレートにしたがって調整する。利得制御50は、チャンネルの周波数応答の大きさを所望のパーシャルレスポンスに一致させるために、ライン21を介して可変利得増幅器22の利得を調整する。

#### 【0051】

等化されたサンプル32は、PR4シーケンス検出器34（典型的には、一對のインタリーブされたスライディング閾値検出器として実施される）にも送られる。PR4シーケンス検出器34は、サンプル値から推定された2進シーケンス $b(n)$ 33を検出する。RLLDデコーダ36は、PR4シーケンス検出器34からの推定された2進シーケンス $b(n)$ 33を、推定されたユーザデータ37に復号化する。データシンク検出器66は、RLLDデコーダ36の動作をフレームに同期させるために、データセクタ15の中のシンクマーク70（図2Bに示す）を検出する。誤差がなければ、推定された2進シーケンス $b(n)$ 33は、記録された2進シーケンス $b(n)$ 8と一致し、復号化されたユーザデータ37は、記録されたユーザデータ2と一致する。

#### 【0052】

（データフォーマット）

図2Aは、1連の同心円状データトラック13を備えた磁気媒体のデータフォーマットの1例を示す。ここで、それぞれのデータトラック13は、サーボウェッジ17の埋め込まれた複数のセクタ15を有している。サーボコントローラ（不図示）が、サーボウェッジ17内のサーボデータを処理し、そのサーボデータに応答して、リード/ライトヘッドを所望のトラック上に位置づける。また、サーボコントローラは、サーボウェッジ17内のサーボバーストを処理することによって、データをライトしたり、リードしたりしながら、所望のトラックのセンタライン上にヘッドを位置合わせし続ける。サーボウェッジ17は、簡単な離散時間パルス検出器によっても、あるいは離散時間シーケンス検出器34によっても検出されうる。もしシーケンス検出器34がサーボデータを検出すれば、サーボウェッジ17のフォーマットは、ユーザデータセクタ15と同様に、プリアンブルと、シンクマークとを含むことになる。

#### 【0053】

図2Bは、アクイジションプリアンブル68、シンクマーク70およびデータ72を備えたユーザデータセクタ15のフォーマットを示す。タイミングリカバリは、アクイジションプリアンブル68を処理することによって、データ72をリードする前に正しいサンプリング周波数および位相を得ており、シンクマーク70は、データ72の始まりを区切る。

#### 【0054】

記憶密度の全体を増加させるために、ディスクは、1トラック当たり14個のデータセクタを有する外側のゾーン11と、1トラック当たり7個のデータセクタを有する内側のゾーン27とに区分される。実用時には、ディスクは、実際には、ゾーン当たりのセクタ数が互いに異なる複数のゾーンに区分される。その場合、データは、それぞれ異なるデータレートでそれぞれのゾーンに記録されたり、そこから検出されたりする。

#### 【0055】

（従来のタイミングリカバリ）

図3Aは、結合された記録チャンネル $q(t)$ 42、アナログフィルタ $f(t)$ 44、および離散時間等化器 $c(n)$ 46のPR4ダイパルス応答 $h(t)$ 40を示している。もしサンプリング周波数および位相が正しい（ $\epsilon = 0$ ）なら、等化されたサンプルシーケンスは、 $\{0, 0, +1, 0, -1, 0, 0, \dots\}$ となる。これに対して、位相誤差（ $\epsilon \neq 0$ ）は、等化されたサンプルシーケンスに誤差をもたらす。例えば位相誤差をもつサンプルシーケンスは、 $\{+0.1, -0.2, +0.4, +0.8, -0.4, -0.8, +0.2, -0.1, \dots\}$ でありうる。これらの誤差は、符号間干渉（ISI）が存在する時には大きくなる。図3Bに示されているように、最適なサンプリングの事例は、ISIが最小になる、アイダイヤグラムの開口48において起こる。理想的なPR4記録チャンネルでは、最適なサンプリング位相は、 $\epsilon = 0$ の時（すなわち、リード信号のサンプ

10

20

30

40

50

ルが、ポーレートと同期がとられている時)である。

【0056】

図1に示す従来のPR4リードチャンネルでは、タイミングリカバリ回路28は、サンプリングデバイスを同期させることによって、リード信号のサンプル値をポーレートに同期させる。図4は、従来のサンプリングタイミングリカバリ回路28の概観を示している。可変周波数発振器(VFO)164は、典型的にはデジタルリードチャンネルにおけるアナログ・デジタル変換器(A/D)であるサンプリングデバイス24のサンプリングクロック23を制御する。マルチプレクサ159は、アキュイジションのあいだは等化されていないサンプル値25を選択し、トラッキングのあいだは等化されたサンプル値32を選択することによって、付随するレイテンシを避けるために、アキュイジションのあいだ、離散等化器フィルタ26をタイミンググループから除く。位相誤差推定器155は、ライン149を介して受け取られたサンプル値と、ライン143上の、スライサのようなサンプル値推定器141からの推定されたサンプル値 $X_k$ とに反応して、位相誤差推定値を発生する。ループフィルタ160は、位相誤差をフィルタリングすることによって、サンプリングクロック23と、ポーレートとの間の周波数差に比例した値に落ちつく周波数オフセット $f_{167}$ を発生する。周波数オフセット $f_{167}$ は、周波数シンセサイザ52からのセンタ周波数制御信号64とともに、サンプリングクロック23をVFO164の出力において調整することによって、サンプリングをポーレートに同期させる。

10

【0057】

ゼロ位相スタート回路162は、VFO164の動作をアキュイジションの始めに停止させることによって、サンプリングクロック23と、リード信号62との間の初期位相誤差を最小にする。これは、VFO164をディセーブルし、アナログリード信号62におけるゼロクロスを検出し、検出されたゼロクロスと第1のポーレートサンプルとの間の所定の遅延の後にVFO164を再びイネーブルすることによっておこなわれる。

20

【0058】

図4の位相誤差推定器155は、以下の確率勾配方程式(1)に従って動作する。

【0059】

$$= Y_k \cdot (X_{k-1}) + Y_{k-1} \cdot (X_k) \quad (1)$$

ここで、 $Y_k$ は、実際のサンプル値149であり、 $X_k$ は、スライサ141の出力における推定されたサンプル値143である。

30

【0060】

位相誤差を推定するための上記方程式は、それぞれのサンプル事例におけるアナログリード信号に含まれるパルスの傾きを近似することに基づいている。もしこの傾きが十分に規定されないのなら、位相誤差推定値の精度が低下する。 $1/(1+D^2)$ プリコーダ10を用いる、図1に示す従来のPR4リードチャンネルでは、データシンボルを表すアナログパルスの傾きは、そのシンボルがディスク上にライトされた時のプリコーダ10の状態によってはうまく規定できない。このことは、図5に示されている。図5は、従来のPR4リードチャンネルにおいてディスク上にライトされた同一のデータシンボルに対する2つの可能なアナログリード信号(すなわち、これら2つの可能なアナログリード信号は、 $1/(1+D^2)$ プリコーダの初期状態に依存している)を示している。サンプル事例3とサンプル事例4との間で、上側の波形におけるアナログリード信号の傾きは、下側の波形におけるアナログリード信号の傾きよりも十分によく規定されうる。容易に理解できるように、これは、符号間干渉(ISI)の与える影響が、(破線で示されているように)第2のダイパルス応答の極性次第で変わってくるからである。よって、下側の波形に比べて、上側の波形のほうが、方程式(1)の位相誤差推定値にはるかに近似した傾きを与えることができる。

40

【0061】

(改善されたサンプリングされた振幅リードチャンネル)

図6は、本発明による改善されたサンプリングされた振幅リードチャンネルを示している。符号化スキームを簡単にするために、図1に示されている従来のRLLEncoder6の代

50

わりに、後述するようにタイミングリカバリの位相誤差推定値を改善する本発明による RLL / タイミングエンコーダ 170 が用いられている。さらに、図 1 の  $1 / (1 + D^2)$  プリコーダ 10 および NRZ 変調器 14 は用いられない。その代わり、符号化されたデータシンボル  $b(n)$  8 は、NRZI 変調器 172 を用いて磁気ディスクに直接、ライトされる。すなわち、データシンボル  $b(n)$  8 内の「1」ビットについては、NRZI 変調器 172 は、ライト電流を（正から負に、またはその逆に）トグルし、「0」ビットについては、ライト電流を変更しないままにする。

【0062】

記録された 2 進データをリードする時には、図 6 のリードチャンネルは、上述した図 1 に示す従来技術によるリードチャンネルと同様に動作する。好ましくは 2 つのインタリーブされたスライディング閾値検出器として実施される、ピタビ型の PR4 シーケンス検出器 174 は、図 1 の検出器 34 と同様に、等化されたサンプル値 32 の偶数インタリーブおよび奇数インタリーブを処理する。しかし、従来の  $1 / (1 + D^2)$  プリコーダ 10 が用いられないので、PR4 シーケンス検出器 174 は、データシーケンス 176 内で検出された「1」に対応するそれぞれの磁気遷移について符号ビット 178 を出力する（すなわち、シーケンス検出器 174 は、符号付きの PR4 データを出力する）。符号ビット 178 には、それぞれのインタリーブにおいてシーケンス検出器 174 により出力される「1」ビットおよび「0」ビットのそれぞれが関連づけられる。例えば、もし偶数のインタリーブにおいて正の遷移が検出されれば、シーケンス検出器 174 は、偶数のインタリーブにおいて負の遷移が検出されるまで、複数の「+0」値が後続する「+1」を出力する。表 6 による組み合わせロジックを伴って実施される NRZI 変換器 180 は、符号付きの PR4 データを推定された符号付きの NRZI シーケンス (SNRZI)  $\wedge b(n)$  33 に変換する。そしてこのようにして、ディスクにライトされたデータシンボル  $b(n)$  8 の表現に変換しなおす。

【0063】

【表 6】

PR4 出力 $D_{n-1} \quad D_n$	SNRZI $S_{n-1} \quad S_n$	PR4 出力 $D_{n-1} \quad D_n$	SNRZI $S_{n-1} \quad S_n$
+0 +0	+0 +0	+1 +0	+0 +0
+0 -0	+1 -1	+1 +1	+1 +0
-0 -0	-0 -0	+1 -1	+0 -1
-0 +0	-1 +1	-1 -0	-0 -0
+0 -1	+0 -1	-1 +1	-0 +1
-0 +1	-0 +1	-1 -1	-1 -0
+0 +1	+1 +0	+1 -0	+1 -1
-0 -1	-1 -0	-1 +0	-1 +1

【0064】

PR4 シーケンス検出器 174 および NRZI 変換器 180 について代わりに実施可能な形態が、図 7A および図 7B に示されている。この実施の形態では、スライディング閾値ピタビ検出器 171A および 171B は、等化されたサンプル値 32 の偶数インタリーブおよび奇数インタリーブをそれぞれ処理するが、上述したような符号ビット 178 は出力しない。その代わり、それぞれの検出器は、（従来の NRZI 状態遷移図ではなく）図 7A に示されている NRZ 状態遷移図に従って動作するように改変されている。2 つの状態 173A および 173B は、NRZ シーケンスの状態、すなわち、磁束の極性を示している。遷移の分岐には、参照符号  $\pm S / X$  というラベルがつけられている。ここで、 $\pm S$  は

10

20

30

40

50

信号サンプルを表し、Xは2進NRZシーケンスを表す。検出器171Aおよび171Bにより出力されるNRZシーケンス175Aおよび175Bが、マルチプレクサ177で組み合わせられることによって、シフトレジスタ181へとシフトされる完全なNRZシーケンス179を形成する。シフトレジスタの出力176は、その後、NRZシーケンス179を1-Dフィルタ180に通すことによって、符号付きのNRZIシーケンス $\hat{b}(n)$ 33に変換される。ここで、フィルタ内の加算器183は、10を底とする(base ten)(XORゲートではない)。

#### 【0065】

推定されたNRZIシーケンス $\hat{b}(n)$ 33は、推定されたユーザデータシーケンス37を発生するために、RLL/タイミングエンコーダ170とは逆のマッピングに従って、RLL/タイミングデコーダ182により復号化される。誤り検出および訂正(EDAC)回路184は、推定されたユーザデータシーケンス内の誤りを検出、訂正することによって、ホストコンピュータに転送される訂正されたユーザデータシーケンス186を発生する。

10

#### 【0066】

上述したように、図1に示す従来の $1/(1+D^2)$ プリコーダ10は、ビット誤りの伝搬を防止する。このことは、ビット指向の誤り検出を用い、例えばファイヤコードのような訂正符号を用いる記憶システムには必要である。本発明は、後述する独特のエンコーダ170/デコーダ184実現方式により、このビット誤り伝搬の問題に対処する。EDAC回路184は、例えば、リードソロモン符号のような公知のバイト指向の誤り訂正符号のいずれかに従って動作する。このような符号の詳細については、当業者にはよく知られている。

20

#### 【0067】

(エンコーダ)

本発明の一面は、ディスク上にライトされる磁気遷移を制御し、それによりリードバック時に結果として得られるアナログパルスの各サンプル事例における傾きを制御することによって、図6のタイミングリカバリ回路28における位相誤差推定値を改善することができるように、データを符号化すること(170)にある。本発明によれば、磁束遷移を制御することが可能である。なぜなら、図1に示す従来のPR4リードチャネルに見られる $1/(1+D^2)$ プリコーダ10は、図示されているように、図6のPR4リードチャネルでは用いられないからである。換言すれば、データシンボルのある与えられた入力シーケンスについて、結果として得られるアナログリード信号は、あいまいさなしに決めることができる。よって、このリード信号は、結果として得られる位相誤差推定値の質に従って評価されうる。

30

#### 【0068】

よって、本発明では、ユーザデータシンボルはそれぞれ、そのシンボルが生成することになる最終的な位相誤差推定値の質に従って、そのグレードが決められる。もし、あるデータシンボルのグレードが所定の閾値を超えており、結果として得られる位相誤差推定値が十分に正確であることが示されるのなら、そのデータシンボルは、修正なしでディスクにライトされる。しかし、もしそのグレードが閾値を下回っているのなら、そのデータシンボルは、位相誤差推定値の正確さを改善する新しいシンボルに符号化される。

40

#### 【0069】

本発明によるRLL/タイミングエンコーダ170の詳細なブロック図は、図8Aに示されている。ユーザデータ2は、17ビットシフトレジスタ188に下位バイト190と上位バイト192とをロードすることによって、一度に16ビット処理される。シフトレジスタの中央ビット194は「0」にセットされ、コードワードが当初はマッピングされないことを示す。以下の説明から理解できるように、現在のコードワードのグレードを決めることができるように、以前のコードワードの下位バイト196が、レジスタ198にロードされる。

#### 【0070】

50

グレーダ200は、それぞれのビットがシフトレジスタ188からライン202を介してシフトアウトするとき、現在のコードワードにおけるそれぞれのビットのグレードを決める。コードワードのこれらのビットは、グレードを決められた後、ライン202を介して再びレジスタ188へとシフトしなおされる。ビットのグレードは、レジスタ206に蓄積され(204)、全コードワードのグレード208は、比較器210において以前の閾値と比較される。閾値を超えたかどうかによって、レジスタ188にシフトしなおされていた現在のコードワードが、この時点で符号化される。

【0071】

好ましい実施の形態では、比較器210における閾値は、18である。もしコードワードのグレード208が18を超えているのなら、そのコードワードはマッピングされない。なぜなら、このコードワードは、すでに、正確な位相誤差推定値を発生できるからである。逆に、もしコードワードのグレード208が18を下回っているのなら、上位バイト192および下位バイト190は、以下の表7A~表7Fに従って動作する8/8エンコーダ212により、より正確な位相誤差推定値を与える新しいコードワード214にそれぞれ独立してマッピングされる。この選択的マッピングは、グレード比較器210の出力218と、FORCE MAP信号220およびINHIBIT信号222とによって制御されるマルチプレクサ216を通して実施される。これら2つの信号については、以下に詳しく説明する。

【0072】

【表7A】

マッピングされていない	マッピングされた	マッピングされていない	マッピングされた
00000000	01010010	10000000	10101010
00000001	01001011	10000001	01010111
00000010	10101101	10000010	01011110
00000011	01010110	10000011	01000100
00000100	10101011	10000100	01011000
00000101	11001000	10000101	11000110
00000110	10111010	10000110	01001101
00000111	01001000	10000111	01010000
00001000	10101000	10001000	00110101

【0073】

【表7B】

10

20

30

マッピングされていない	マッピングされた	マッピングされていない	マッピングされた
00001001	00100111	10001001	11111001
00001010	11101000	10001010	00110111
00001011	10010011	10001011	10001101
00001100	10101110	10001100	01000010
00001101	10010001	10001101	10000010
00001110	10100110	10001110	00110100
00001111	01001001	10001111	00100101
00010000	10010110	10010000	00110010
00010001	00100011	10010001	11110111
00010010	00100000	10010010	10000001
00010011	11000101	10010011	10111100
00010100	11100010	10010100	00110011
00010101	01100000	10010101	00111111
00010110	11000100	10010110	10110000
00010111	11101011	10010111	00110001
00011000	10101001	10011000	00101110
00011001	10111110	10011001	10011110
00011010	10111000	10011010	10011000
00011011	01101001	10011011	11110101
00011100	10001010	10011100	00101100
00011101	11011001	10011101	00011011
00011110	10100010	10011110	00101001
00011111	01000101	10011111	01001100
00100000	01110101	10100000	01000110
00100001	00100001	10100001	00011001

10

20

30

40

【 0 0 7 4 】

【 表 7 C 】

マッピングされていない	マッピングされた	マッピングされていない	マッピングされた
00100010	00001011	10100010	11110011
00100011	10100111	10100011	01111101
00100100	00001000	10100100	11110001
00100101	11110000	10100101	00000011
00100110	11010111	10100110	10001100
00100111	00011101	10100111	00001101
00101000	11010000	10101000	10000110
00101001	11100000	10101001	00000001
00101010	11001111	10101010	00011111
00101011	00000101	10101011	11101111
00101100	10100011	10101100	01110111
00101101	00011000	10101101	11111110
00101110	00010011	10101110	00000010
00101111	10001001	10101111	11110100
00110000	10010101	10110000	00100010
00110001	10100001	10110001	01111100
00110010	11001101	10110010	01111110
00110011	01100101	10110011	11110010
00110100	10011100	10110100	01110000
00110101	00000100	10110101	11111100
00110110	01101101	10110110	01110011
00110111	10010111	10110111	01011111
00111000	01110100	10111000	00010110
00111001	00010001	10111001	11100111
00111010	00001001	10111010	11100011

10

20

30

40

【 0 0 7 5 】

【 表 7 D 】

マッピングされていない	マッピングされた	マッピングされていない	マッピングされた
00111011	10010000	10111011	01000000
00111100	10010100	10111100	00010010
00111101	01101011	10111101	01110001
00111110	01110010	10111110	00001010
00111111	00101010	10111111	00100100
01000000	01011100	11000000	01010101
01000001	11111010	11000001	00111010
01000010	01111011	11000010	11101110
01000011	10001110	11000011	11010101
01000100	01111001	11000100	11101001
01000101	11000000	11000101	00111110
01000110	11000010	11000110	11101010
01000111	11110110	11000111	00101101
01001000	01101111	11001000	11100101
01001001	00000000	11001001	11100001
01001010	00001111	11001010	01111000
01001011	11000111	11001011	01001111
01001100	10111101	11001100	11011010
01001101	11000011	11001101	01000111
01001110	01100111	11001110	11100100
01001111	10001011	11001111	11010100
01010000	10110111	11010000	00101011
01010001	10000000	11010001	00110000
01010010	00000111	11010010	11011111
01010011	00001100	11010011	01000011

10

20

30

40

【 0 0 7 6 】

【 表 7 E 】

マッピングされていない	マッピングされた	マッピングされていない	マッピングされた
01010100	11000001	11010100	00111000
01010101	11111111	11010101	11111000
01010110	00000110	11010110	01000001
01010111	10111111	11010111	00101111
01011000	10001000	11011000	11010110
01011001	10011111	11011001	00001110
01011010	10001111	11011010	00011100
01011011	01100011	11011011	11101101
01011100	11101100	11011100	11011100
01011101	01111111	11011101	00111100
01011110	10000101	11011110	10110110
01011111	10110011	11011111	00101000
01100000	01101010	11100000	01010100
01100001	10000100	11100001	00100110
01100010	10110001	11100010	11010011
01100011	01100100	11100011	11001010
01100100	10101111	11100100	11010001
01100101	11111101	11100101	00010111
01100110	01101000	11100110	10101100
01100111	01111010	11100111	00010101
01101000	01101100	11101000	00010100
01101001	10000111	11101001	00011110
01101010	11111011	11101010	00010000
01101011	11100110	11101011	11011011
01101100	01100010	11101100	10011010

10

20

30

40

【 0 0 7 7 】

【 表 7 F 】

マッピングされていない	マッピングされた	マッピングされていない	マッピングされた
01101101	01100001	11101101	00111011
01101110	10100000	11101110	11001110
01101111	01011011	11101111	10110101
01110000	01010011	11110000	10100101
01110001	11011110	11110001	11011101
01110010	00111101	11110010	11001011
01110011	01100110	11110011	10110100
01110100	11011000	11110100	11001001
01110101	10000011	11110101	00111001
01110110	10011011	11110110	01110110
01110111	11001100	11110111	10111011
01111000	01011101	11111000	10100100
01111001	00110110	11111001	10111001
01111010	00011010	11111010	10011101
01111011	01011001	11111011	10110010
01111100	01010001	11111100	10010010
01111101	10011001	11111101	01101110
01111110	01001110	11111110	01011010
01111111	11010010	11111111	01001010

10

20

30

40

## 【 0 0 7 8 】

コードワードに含まれるそれぞれのビットのビットグレードは、以下の表 8 に従って生成される。最も正確な位相誤差推定値を与えるビットシーケンスについては、表 8 は、現在、グレードが決められつつある対応するビットに関連づけられたグレード値を示している。すなわち、図 8 A のグレード 2 0 0 は、現在、グレードが決められつつある ( 2 0 2 ) ビットを、周辺のビットの文脈に則って評価する。もし入力シーケンスと、表 8 に掲げられているシーケンスのうちのいずれか一つとの間に一致が見られれば、グレード 2 0 0 は、グレードの決められたビットに対応するグレード値を出力する。そうでなければ、グレード 2 0 0 は、ゼロのグレードを出力する。

## 【 0 0 7 9 】

## 【 表 8 】

↓	グレード	↓
1001	2	1001
0001	1	1000
010	2	010
1011	1	1101
0110	2	0110
01110	2	01110
011110	1	011110
0111110	2	0111110
1111110	1	0111111
10X	1	ビット16に関して
011	1	ビット13~16に関して

10

20

## 【0080】

例えば、以前のコードワードと現在のコードワードとが以下のような場合を考える。

## 【0081】

以前のコードワード            現在のコードワード  
 ... X X X 0 1 1 0 1            1 1 0 0 1 0 1 1 0 0 1 1 1 0 1 1

図8Aのグレーダ200は、現在のコードワードの最初のビットに対してビットグレード2を出力する。なぜなら、このビットは、表8の第1欄、第6行のシーケンス「01110」の第3のビットであるからである。第2のビットについては、グレード2が出力される。なぜなら、このビットは、第1欄、第6行のシーケンス「01110」の第4のビットであるからである。第3のビットについては、グレード0が出力される。なぜなら、このビットは、表8の行のどれとも一致しないからである。第4のビットには、グレード2が割り当てられる。なぜなら、このビットは、第1行のシーケンスと一致するからである。以下も同様である。

30

## 【0082】

一致が見られるのは、表8のシーケンスが現在のコードワードを超えない場合のみである。例えば、現在のコードワードのビット16のグレードを決める時、表8では、行3の最後の欄、ならびに、行10および11の第1欄のみが用いられる。なぜなら、その他のシーケンスは、すべて現在のコードワードの端を超えているからである。さらに、表8の最後の行は、一致がほかに見られない時には、現在のコードワードのビット13~16のグレードを決めるのに用いられる。

40

## 【0083】

タイミングリカバリのための位相誤差推定値の正確さを最適化するためにユーザデータを符号化することに加えて、RLLE/タイミングエンコーダ170は、コードワードが確実に以下の制約を満たすようにしなければならない。すなわち、

1. リードバック時には、リード信号のサンプル値に含まれる、連続する非ゼロのサンプルの間に最大k個の(kの好ましい値は14である)サンプルが確実に発生するようにする制約と、
2. リードバック時には、リード信号サンプル値に含まれる偶数および奇数のインタリーブの両方で、現在のコードワードの中央ビットに続くビットから始まり、後続するコード

50

ワードの中央ビットで終わる17ビットから構成されるそれぞれのシーケンスに、非ゼロのサンプル値が確実に発生するようにする制約と、の2つである。

【0084】

上記第1の制約は、RLL(d, k)符号の公知のk制約に類似している。すなわち、この制約は、PR4リードチャネル用のタイミングリカバリ28を確実に正しく動作させる。なぜなら、位相誤差推定値は、非ゼロのサンプル値が等化器26の出力において処理される時のみ計算されるからである。上記第2の制約は、PR4シーケンス検出器34において確実にパスを除外(path closure)することができるようにし(すなわち、疑似破局的な誤り事象を符号化の対象から外し(code out))、PR4シーケンス検出器34によりどのような最小距離の誤り事象が発生されても、そのような誤り事象が3つよりも多くのコードワードバイトに影響を及ぼす事態を確実に防止できる。これにより得られる効果については、以下に述べるEDAC回路184の説明を読めば、もっとよく理解できるであろう。

10

【0085】

図8Aにおいて、ランレングス検出器224は、現在のコードワード188の上位バイト228、以前のコードワード226の予め処理された(prepend)下位バイト198およびマッピングされたコードワード214を評価することによって、上記2つの制約が満たされているかどうかを判定する。上記2つの制約についてチェックをおこなうためには、入力シーケンスを1+Dフィルタ(不図示)に通すことによって、その入力シーケンスをPR4フォーマットに変換しなければならない。もし現在のコードワードがマッピングされておらず、上記ランレングス制約に違反しているのなら、ランレングス検出器224は、マルチプレクサ216においてコードワードが強制的にマッピングされるようにするFORCE MAP信号220をアサートする。表7A~表7Fに示すようにマッピング用にセットされたコードワードは、もしFORCE MAP信号220がアクティベートされれば、マッピング後のコードワードが上記制約を必ず満たすことを保証する。

20

【0086】

もし、より正確な位相誤差推定値を与えるために、現在のコードワードが新しいコードワード214にマッピングされる(212)のなら、ランレングス検出器224は、マッピングされたコードワード214と、以前のコードワードの予め処理された下位バイト198とをチェックすることによって、上記2つの制約が満たされているかどうかをチェックする。もしマッピングされたコードワード214が上記制約に違反しているのなら、ランレングス検出器224は、マルチプレクサ216におけるマッピングを禁止するINHIBIT信号222をアサートする。表7A~表7Fのマッピング用にセットされたコードワードは、もしINHIBIT信号222がアクティベートされれば、マッピングされていないコードワードが上記制約を必ず満たすことを保証する。

30

【0087】

8/8エンコーダ212は、表7A~表7Fに従ってマッピングされたコードワード214を以下のように生成する。まず、エンコーダ212は、レジスタ228に格納されている現在のコードワードの上位バイト192を表7A~表7Fに従ってマッピングする。次に、レジスタ230に格納されている現在のコードワードの下位バイト190がビット反転(bit-wise reversed)され(232)、上位バイト228と同じマッピングを用いて(すなわち、表7A~表7Fを用いて)8/8エンコーダ212により符号化される。マルチプレクサ216の出力におけるマッピングされた/マッピングされていないコードワード234は、17ビットのシフトレジスタ226にロードされる。上位バイトは、レジスタ236にロードされ、下位バイトは、ビット反転された(238)後、レジスタ240にロードされる。シフトレジスタ226の中央ビット242は、もしコードワードがマッピングされているのなら、マルチプレクサ244を介して「1」ビットとしてロードされ、もしコードワードがマッピングされていないのなら、「0」ビットとしてロードされる。中央ビット242は、図6のデコーダ182により、リード動作の間にコードワードを復号化するのに用いられる。このことについては、EDAC回路184を説明する次の

40

50

セクションで図 8 B を参照して詳しく説明する。

【 0 0 8 8 】

( 誤り検出および訂正 )

図 6 に示されている本発明によるリードチャネルは、R L L / タイミングデコーダ 1 8 2 により出力された推定されたデータシーケンス 3 7 における誤りを訂正する、誤り検出および訂正回路 ( E D A C ) 1 8 4 においてバイト指向の誤り訂正符号 ( E C C ) を実施する。例えば、リードソロモン符号のようなバイト指向の E D A C システムでは、ディスクにライトされるコードワードを形成するためにデータに付加される冗長シンボルを生成するために、データは数学的に処理される。これらの冗長シンボルは、その後、リードバック時に、推定され復号化されたデータシーケンス 3 7 における誤りを検出し、訂正するために用いられる。

10

【 0 0 8 9 】

ここで、再び図 2 A および図 2 B を参照すると、トラック 1 3 における各セクタ 1 5 は、単一の E C C コードワードを表すことができる。すなわち、1 個のセクタのデータフィールド 7 2 は、ユーザデータと、そのユーザデータの終わりに付加される冗長シンボルとを含むことができる。しかし、磁気記録システムにおいて最もよく発生する誤りは、典型的には、2 つ以上の連続するバイトに及ぶバースト誤りである。多数のコードワードをつくるために 1 個のセクタにおけるデータをインタリーブすれば、E D A C 回路 1 8 4 のパフォーマンスが改善されることはよく知られている。なぜなら、バースト誤りは、単一のコードワードに影響を及ぼすのではなく、多数のコードワードにわたって広がるからである。このことは、図 9 A に示されている。図 9 A は、3 つのインタリーブされたコードワードに分割された、典型的な 5 1 2 バイトセクタを示している。データバイト 0 は、コードワード 1 の第 1 のバイトであり、データバイト 3 は、その第 2 のバイトである。データバイト 1 は、コードワード 2 の第 1 のバイトであり、データバイト 4 は、その第 2 のバイトである。以下も同様である。もしバースト誤りがデータバイト 3 に始まり、データバイト 5 で終わるのなら、インタリーブされていないシステムのように 1 個のコードワード中の 3 つのバイトに誤りがあるのではなく、1 コードワードにつき 1 バイトのみに誤りがあることになる。本発明では、E D A C 回路 1 8 4 は、好ましくは、3 インタリーブ・リードソロモン誤り訂正符号として実施される。このような符号は、例えば、米国特許第 5,446,743 号に開示されている。この特許は、本発明と同一の譲受人に譲渡されている。

20

30

【 0 0 9 0 】

インタリーブされたバイト指向の符号を用いた誤り訂正に関する、従来の技術によるリードチャネルの別の局面は、図 1 に示されている R L L エンコーダ 6 および R L L デコーダ 3 6 ( E N D E C ) である。典型的には、検出されたデータシーケンスにおけるそれぞれのビット誤りが、復号化された出力シーケンスにおける 1 バイトのみに影響を及ぼすだけですむように、従来の E N D E C の符号レートは選択される。例えば、先に引用した米国特許第 5,422,760 号は、9 ビットの各コードワードを 8 ビットのデータバイトに復号化することによって、復号化された出力シーケンスにおいて 9 ビットの誤りからのビット誤り伝搬を 1 バイトまたは 2 バイトに制限する、8 / 9 E N D E C を開示している。しかし、このような従来の 8 / 9 E N D E C に関する欠点としては、レートが 8 / 9 と低いことに伴うオーバーヘッドのために記憶容量が低下することが挙げられる。例えば、レート 1 6 / 1 7 のようなもっと高いレートの E N D E C を従来の技術により実施すれば、2 つのコードワードに広がるバースト誤りが、復号化されたシーケンス 3 7 における 4 つのバイトの悪影響を及ぼす結果、図 9 A のコードワードのうちの 1 つが、誤りを含むバイトを 2 つ含むことになってしまう。本発明は、検出された 2 進シーケンス 3 3 において 9 ビット以下の広がりをもつどのようなバースト誤りでも、復号化された出力シーケンス 3 7 では最大でも 2 バイトに影響を及ぼすようにし、1 7 ビット以下の誤りなら、最大でも 3 バイトに影響を及ぼすようにする、レート 1 6 / 1 7 E N D E C を設けることによってこの問題を克服する。よって、図 9 A におけるインタリーブされたそれぞれのコードワードでは、1 7 ビット以下の広がりをもつどのような誤りに対しても、検出された 2 進シーク

40

50

ンス 33 における誤りを含むバイトは最大でも 1 バイトとなる。

【 0091 】

このことは、本発明によれば、コードワードの最初の 8 ビットを、最後の 8 ビットからは独立して復号化することにより達成される。本発明によるデコーダ 182 は、図 8 B に示されている。PR4/NRZI 変換器 180 により出力された推定された NRZI シーケンス  $b(n)$  33 は、8 ビット上位バイトレジスタ 248、中央ビットレジスタ 250 および下位バイトレジスタ 252 を有するレジスタ 246 へとシリアルにシフトされる。いったん 17 ビットのコードワードの全体がレジスタ 246 内にシフトされれば、上位バイト 248 が、8/8 デコーダ 256 への入力としてマルチプレクサ 254 を通して選択される。8/8 デコーダ 256 は、以下の表 9 A ~ 表 9 G に従って復号化されたバイト 258 を出力する。中央ビット 250 は、受け取られたコードワードが図 8 A の 8/8 RLL/タイミングエンコーダ 170 により符号化されたかどうかによって、復号化されたバイト 258 か、または、上位バイト 262 をマルチプレクサ 260 を介して選択する。マルチプレクサ 260 の出力は、デマルチプレクサ 263 を通してルーティングされ、16 ビットレジスタ 266 の上位バイト 264 に格納される。次に、コードワードの下位バイト 252 がビット反転され (268)、マルチプレクサ 254 を通してルーティングされた後、表 9 A ~ 表 9 G に従って 8/8 デコーダ 256 によりマッピングされる (すなわち、コードワードの上位バイト 248 を復号化するのにも、下位バイト 252 を復号化するのにも同じデコーダが用いられるので、回路のコストを下げ、その複雑さを緩和することができる)。中央ビット 250 は、受け取られたコードワードが図 8 A の 8/8 RLL/タイミングエンコーダ 170 により符号化されたかどうかによって、復号化されたバイト 258 か、または、下位バイト 262 をマルチプレクサ 260 を介して選択する。マルチプレクサ 260 の出力は、デマルチプレクサ 263 を通してルーティングされ、ビット反転された (270) 後、16 ビットレジスタ 266 の下位バイト 272 に格納される。

10

20

【 0092 】

【表 9 A】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000000	01001001	10000000	01010001
00000001	10101001	10000001	10010010

30

【 0093 】

【表 9 B】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00000010	10101110	10000010	10001101
00000011	10100101	10000011	01110101
00000100	00110101	10000100	01100001
00000101	00101011	10000101	01011110
00000110	01010110	10000110	10101000
00000111	01010010	10000111	01101001
00001000	00100100	10001000	01011000
00001001	00111010	10001001	00101111
00001010	10111110	10001010	00011100
00001011	00100010	10001011	01001111
00001100	01010011	10001100	10100110
00001101	10100111	10001101	10001011
00001110	11011001	10001110	01000011
00001111	01001010	10001111	01011010
00010000	11101010	10010000	00111011
00010001	00111001	10010001	00001101
00010010	10111100	10010010	11111100
00010011	00101110	10010011	00001011
00010100	11101000	10010100	00111100
00010101	11100111	10010101	00110000
00010110	10111000	10010110	00010000
00010111	11100101	10010111	00110111
00011000	00101101	10011000	10011010
00011001	10100001	10011001	01111101
00011010	01111010	10011010	11101100

10

20

30

40

【 0 0 9 4 】

【 表 9 C 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00011011	10011101	10011011	01110110
00011100	11011010	10011100	00110100
00011101	00100111	10011101	11111010
00011110	11101001	10011110	10011001
00011111	10101010	10011111	01011001
00100000	00010010	10100000	01101110
00100001	00100001	10100001	00110001
00100010	10110000	10100010	00011110
00100011	00010001	10100011	00101100
00100100	10111111	10100100	11111000
00100101	10001111	10100101	11110000
00100110	11100001	10100110	00001110
00100111	00001001	10100111	00100011
00101000	11011111	10101000	00001000
00101001	10011110	10101001	00011000
00101010	00111111	10101010	10000000
00101011	11010000	10101011	00000100
00101100	10011100	10101100	11100110
00101101	11000111	10101101	00000010
00101110	10011000	10101110	00001100
00101111	11010111	10101111	01100100
00110000	11010001	10110000	10010110
00110001	10010111	10110001	01100010
00110010	10010000	10110010	11111011
00110011	10010100	10110011	01011111

10

20

30

40

【 0 0 9 5 】

【 表 9 D 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
00110100	10001110	10110100	11110011
00110101	10001000	10110101	11101111
00110110	01111001	10110110	11011110
00110111	10001010	10110111	01010000
00111000	11010100	10111000	00011010
00111001	11110101	10111001	11111001
00111010	11000001	10111010	00000110
00111011	11101101	10111011	11110111
00111100	11011101	10111100	10010011
00111101	01110010	10111101	01001100
00111110	11000101	10111110	00011001
00111111	10010101	10111111	01010111
01000000	10111011	11000000	01000101
01000001	11010110	11000001	01010100
01000010	10001100	11000010	01000110
01000011	11010011	11000011	01001101
01000100	10000011	11000100	00010110
01000101	00011111	11000101	00010011
01000110	10100000	11000110	10000101
01000111	11001101	11000111	01001011
01001000	00000111	11001000	00000101
01001001	00001111	11001001	11110100
01001010	11111111	11001010	11100011
01001011	00000001	11001011	11110010
01001100	10011111	11001100	01110111

10

20

30

40

【 0 0 9 6 】

【 表 9 E 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01001101	10000110	11001101	00110010
01001110	01111110	11001110	11101110
01001111	11001011	11001111	00101010
01010000	10000111	11010000	00101000
01010001	01111100	11010001	11100100
01010010	00000000	11010010	01111111
01010011	01110000	11010011	11100010
01010100	11100000	11010100	11001111
01010101	11000000	11010101	11000011
01010110	00000011	11010110	11011000
01010111	10000001	11010111	00100110
01011000	10000100	11011000	01110100
01011001	01111011	11011001	00011101
01011010	11111110	11011010	11001100
01011011	01101111	11011011	11101011
01011100	01000000	11011100	11011100
01011101	01111000	11011101	11110001
01011110	10000010	11011110	01110001
01011111	10110111	11011111	11010010
01100000	00010101	11100000	00101001
01100001	01101101	11100001	11001001
01100010	01101100	11100010	00010100
01100011	01011011	11100011	10111010
01100100	01100011	11100100	11001110
01100101	00110011	11100101	11001000

10

20

30

40

【 0 0 9 7 】

【 表 9 F 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01100110	01110011	11100110	01101011
01100111	01001110	11100111	10111001
01101000	01100110	11101000	00001010
01101001	00011011	11101001	11000100
01101010	01100000	11101010	11000110
01101011	00111101	11101011	00010111
01101100	01101000	11101100	01011100
01101101	00110110	11101101	11011011
01101110	11111101	11101110	11000010
01101111	01001000	11101111	10101011
01110000	10110100	11110000	00100101
01110001	10111101	11110001	10100100
01110010	00111110	11110010	10110011
01110011	10110110	11110011	10100010
01110100	00111000	11110100	10101111
01110101	00100000	11110101	10011011
01110110	11110110	11110110	01000111
01110111	10101100	11110111	10010001
01111000	11001010	11111000	11010101
01111001	01000100	11111001	10001001
01111010	01100111	11111010	01000001
01111011	01000010	11111011	01101010
01111100	10110001	11111100	10110101
01111101	10100011	11111101	01100101
01111110	10110010	11111110	10101101

10

20

30

40

【 0 0 9 8 】

【 表 9 G 】

マッピングされた	マッピングされていない	マッピングされた	マッピングされていない
01111111	01011101	11111111	01010101

【 0 0 9 9 】

以上の説明により、本発明によるデコーダ182が、17ビットのコードワードを上位バイト、下位バイトの順で互いに独立して復号化することは明らかであろう。すなわち、従来の技術とは異なり、本発明では、もしあるコードワードの下位バイトに誤りが含まれてい  
 10  
 れば、その上位バイトを正しく復号化する。逆の場合も同様である。その結果、あるコードワードの第1のビットに始まり、第2のコードワードの第1のバイトを超えて広がっている（つまり、少なくとも25ビット長である）誤りは、復号化された出力の3バイトに影響を及ぼすのみとなる。また、17ビット以下の広がりをもつ誤りは、3つの連続する復号化されたバイトに影響を及ぼすのみとなる。なぜなら、17ビットの誤りは、最大でも1個の中央ビットに影響を及ぼすのみであるからである。よって、17ビット以下の広がりをもつどのような誤りでも、図9Aに示されている3つのインタリーブでは、1コードワードにつき1バイトにしか悪影響を及ぼさない。

【 0 1 0 0 】

図6のPR4シーケンス検出器34における最小距離誤り事象が、17ビットを超える広  
 20  
 がりをもつことにより、連続するコードワードの2つの中央ビットに悪影響を及ぼし、復号化された出力のうち4バイトが誤りを含むようになることもありうる。このことは、図9Bに示されている。図9Bは、あるインタリーブにおいて検出されたシーケンスのトレリス図274およびディスク上に記録された対応するコードワード275を示す図である。このシーケンス例では、チャンネルノイズが原因で、シーケンス検出器は、第1のコードワード280の第5のビット278で偽の正遷移276を検出しており、後続するコードワード286の第10のビット284で負遷移282を検出している。その結果、PR4シーケンス検出器34により出力される誤りを含むシーケンス出力は、正しいシーケンスである... 0 -、0 -、0 -、0 -、0 -、0 -、0 -、0 -、0 -、0 -、0 -、0 -、  
 30  
 ...ではなく、... 1 +、0 +、0 +、0 +、0 +、0 +、0 +、0 +、0 +、0 +、0 +、0 +、1 -、...となる。第1のコードワード280の中央ビット288の絶対値は正しく「0」と検出されるものの、符号ビットは、誤って「+」として検出されるので、PR4/NRZI変換器180は、正しくない値を出力することになる。よって、第1のコードワード280の両バイトともに、誤って復号化されることになる。同様に、第2のコードワード286の第1のバイトの符号ビットも誤っているため、それらのビットもまた、PR4/NRZI変換器180により誤って復号化されることになる。最後に、第2のコードワード286の第2のバイトも誤って復号化される。なぜなら、第1のビット284における偽の負遷移282は、正しい値である「0 -」ではなく、「1 -」として検出されるからである。よって、本発明の一面は、PR4シーケンス検出器における最小距離誤り事象が、2つの連続するコードワードの中央ビットを超えて広がるのを防止することによ  
 40  
 って、その誤り事象の長さを3つの連続するコードワードバイトに制約することである。

【 0 1 0 1 】

本発明では、偶数インタリーブおよび奇数インタリーブの両方において、17ビットのシーケンスごとに非ゼロのサンプル値が検出されるように入力シーケンスを符号化することによって、PR4検出器における最小距離誤り事象の長さを3コードワードバイトに制約する。ここで、それぞれの17ビットシーケンスは、ある17ビットコードワードの中央ビット289の直後のビットから始まり、後続するコードワードの中央ビット290で終わる。このことは、図9Cに示されている。図9Cは、第1のコードワード294の第2のバイトに「1」ビット300が符号化されていることを除いて、図9Bに示すシーケンスと同じコードワードシーケンスに対するトレリス図292を示している。このようにし  
 50

て、第1のコードワード294の中央ビット289に続く最初のビットに始まり、第2のコードワード298の中央ビット290で終わる広がりの中で、非ゼロのサンプル値296が、リード信号のサンプル値において検出される。図からわかるように、これにより、(ノイズサンプル276の振幅が、第1のコードワード294における「1」ビット300により生成される信号サンプル296の振幅よりも小さい(その可能性は高い)とすれば)正のノイズサンプル276によりもたらされた最小距離誤り事象が、2つの連続する中央ビットに広がることを防止することができる。

#### 【0102】

よって、上述したように、図8Aのエンコーダ170は、現在のコードワード188の上位バイト228、以前のコードワード226の予め処理された下位バイト198およびマッピングされたコードワード214を評価するRL検出器224を備えている。もし上記制約が満たされないのなら、RL検出器224は、上述したように(FORCE MAP信号220またはINHIBIT信号222を介して)コードワードが強制的にマッピングされるようにするか、または、マッピングを禁止する。

10

#### 【0103】

よって、各コードワードの上位バイトおよび下位バイトを互いに独立して復号化し、かつ、PR4検出器における最小距離誤り事象が2つの連続するコードワードの中央ビットに広がることを防止することによって、17ビット以下の長さをもつ誤り事象により悪影響を及ぼされる連続するコードワードバイトの数は、最大でも3となる。すなわち、図9Aの3インタリーブシステムの場合、17ビット以下のバースト誤り事象は、それぞれのコードワードでわずか1バイトに悪影響を及ぼすにすぎない。このようにして、本発明によれば、従来の技術による16/17 ENDECで見られるような4コードワードバイトにわたる誤りの伝搬を防止しつつ、効率のよいレート16/17 ENDECを実施することができる。

20

#### 【0104】

本発明の各局面は、本願明細書で以上に開示した実施の形態によって完全に実現される。しかし、本発明のさまざまな局面が、他の異なる実施の形態によっても、本発明の本質的作用からはずれることなく実現可能であることは、当業者には理解できよう。例えば、図8Aに示すエンコーダ170および図8Bに示すデコーダ182は、組み合わせのロジックを用いても、あるいはルックアップテーブルを用いても実施することができる。このような改変およびその他の改変は、以上に開示した教示を考慮すれば、自明であり、単なる設計上の選択事項にすぎない。以上に開示した特定の実施の形態は、あくまでも例示を目的とするものであり、本発明の範囲を限定するためのものではない。本発明の範囲は、添付の請求の範囲により適切に解釈されるべきである。

30

#### 【0105】

##### 【発明の効果】

本発明によれば、実施するのがあまり複雑ではなく、復号化された出力における誤りの伝搬を防止できる、レート16/17 ENDECを提供することができる。

##### 【図面の簡単な説明】

【図1】従来のサンプリングされた振幅リードチャネルのブロック図である。

40

【図2A】複数のユーザデータセクタおよび複数の埋め込みサーボデータセクタから構成される複数の同心円状データトラックを有する磁気ディスク記憶媒体のデータフォーマットの一例を示す図である。

【図2B】ユーザデータセクタのフォーマットの一例を示す図である。

【図3A】タイミング位相誤差をもつPR4リードチャネルのサンプリングされたダイパルス応答を示す図である。

【図3B】PR4リードチャネルの3レベル「アイダイアグラム」を示す図である。

【図4】位相同期ループに用いられる位相誤差推定器を有するPR4リードチャネルのための従来のタイミングリカバリ回路を示す詳細ブロック図である。

【図5】従来のPR4リードチャネルに $1/(1+D^2)$ プリコーダを用いる時の、同一

50

のユーザデータ入力シーケンスから得られる2つのアナログリード波形を示す図である。

【図6】本発明による改善されたサンプリングされた振幅リードチャンネルのブロック図である。

【図7A】ダイコードシーケンスの場合のNRZ状態遷移図である。

【図7B】図7Aの状態遷移図に従って動作する、スライディング閾値ビタビシーケンス検出器の実現形態を示す図である。

【図8A】図4の位相誤差推定器の動作を最適化するために、ユーザデータを符号化する16/17エンコーダを詳細に示す図である。

【図8B】検出されたNRZIデータシーケンスを推定されたユーザデータシーケンスに復号化する際に、図8Aのエンコーダとは逆の演算をおこなう16/17デコーダを詳細に示す図である。

10

【図9A】3つのインタリーブされたECCコードワードのかたちで磁気ディスク上に記録される1セクタのデータを示す図である。

【図9B】最小距離PR4誤り事象が2つの連続するコードワードの中央のビット間に広がることによって、復号化された出力シーケンスの4つのバイトに悪影響を及ぼすことを説明する図である。

【図9C】図9Bの誤り伝搬を防止する、本発明による符号化制約を示す図である。

#### 【符号の説明】

2 ユーザデータ

4 データ発生器

9 ライト回路

18 磁気記録チャンネル

20 アナログ受信フィルタ

22 可変利得増幅器

24 A/D変換器

26 離散等化器フィルタ

28 タイミングリカバリ

50 利得制御

52 周波数シンセサイザ

60 マルチプレクサ

66 データシンク検出器

172 NRZI変調器

174 PR4シーケンス検出器

180 PR4/NRZI変換器

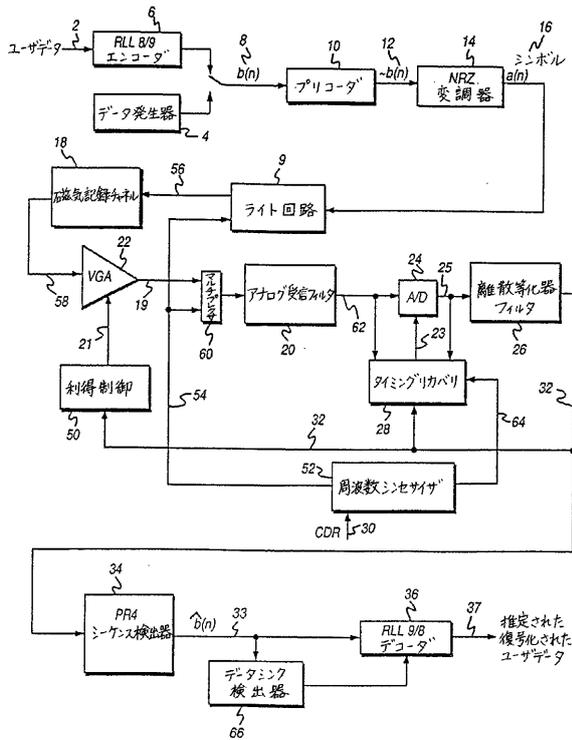
182 RLL/タイミングデコーダ

186 誤り訂正されたユーザデータ

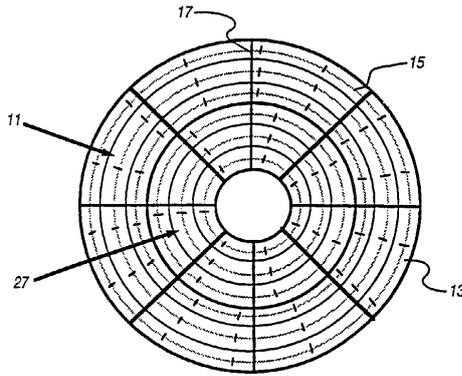
20

30

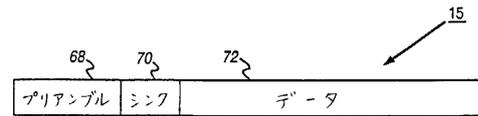
【図1】



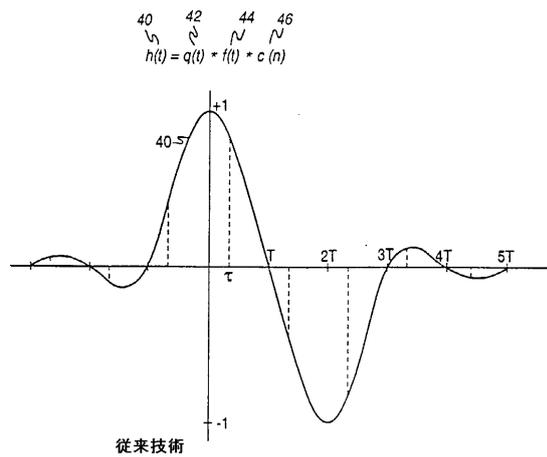
【図2A】



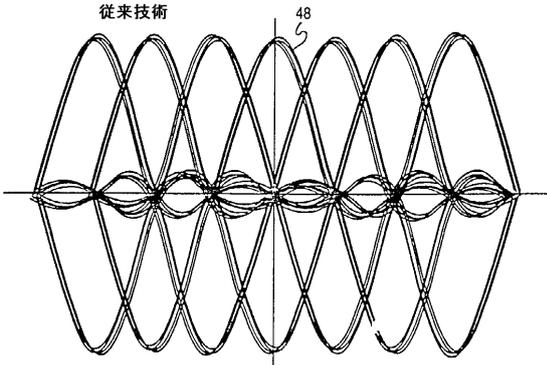
【図2B】



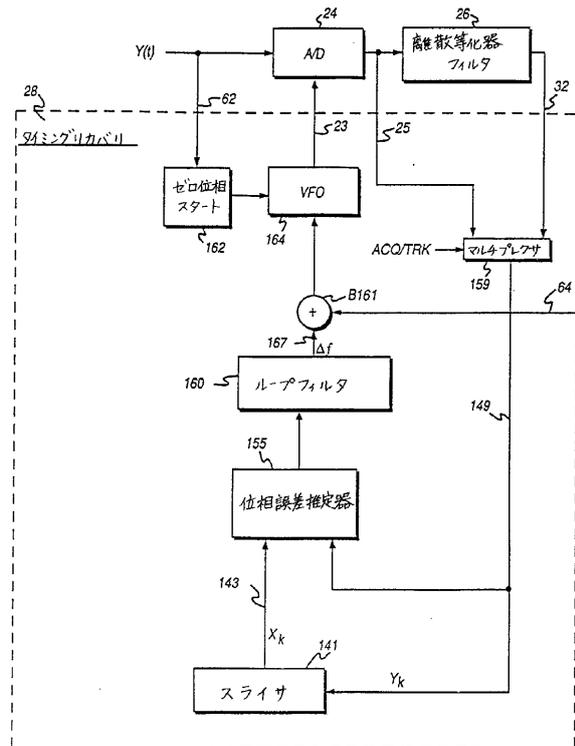
【図3A】



【図3B】

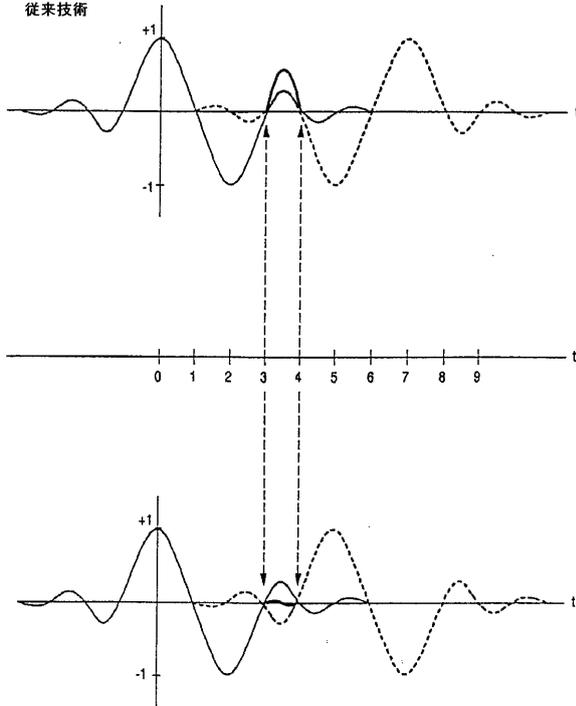


【図4】

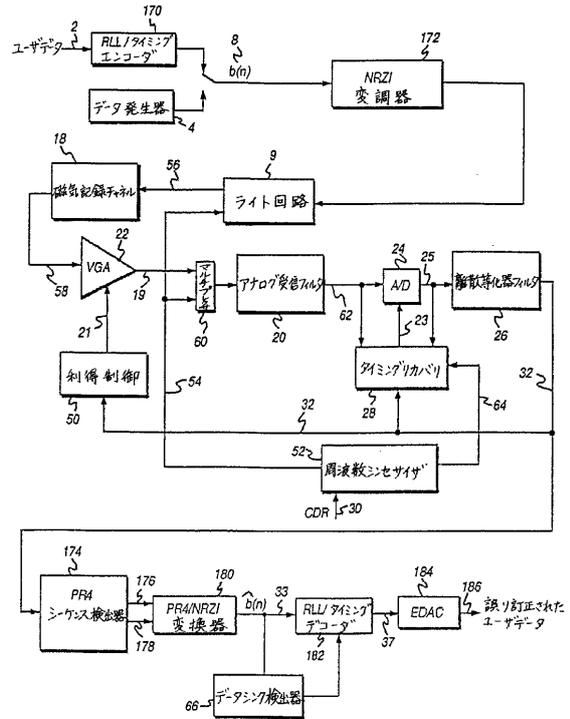


【図5】

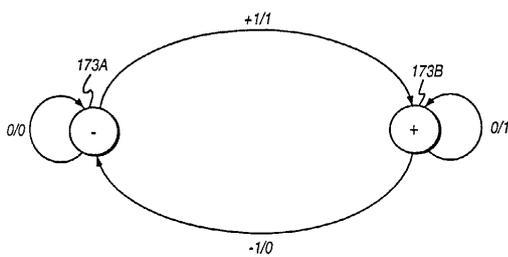
従来技術



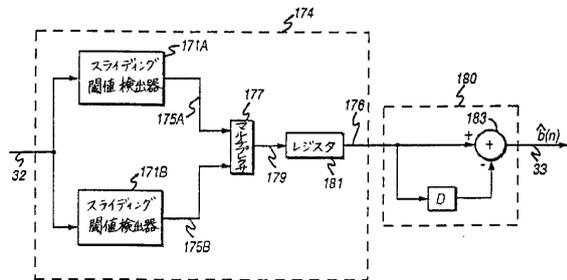
【図6】



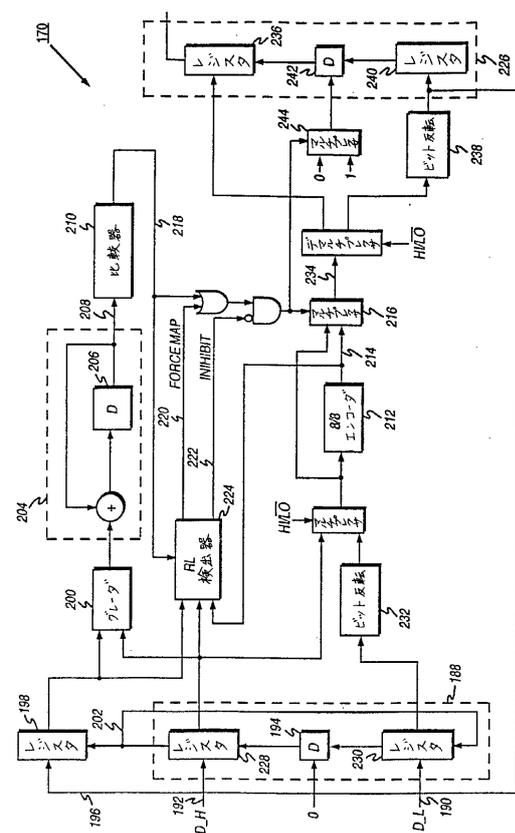
【図7A】



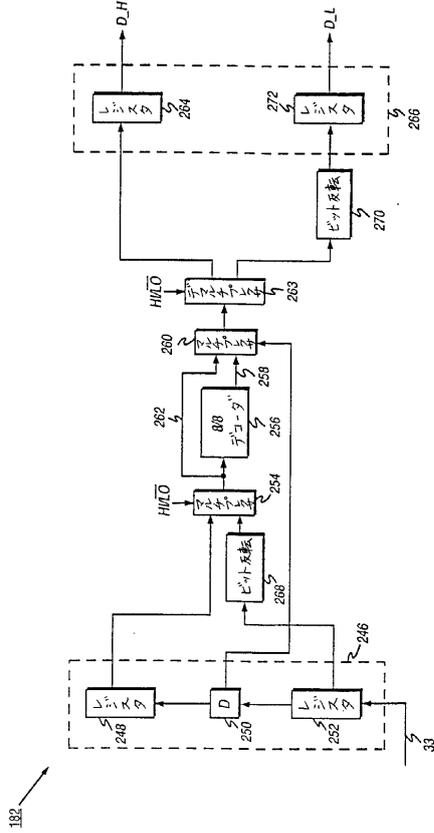
【図7B】



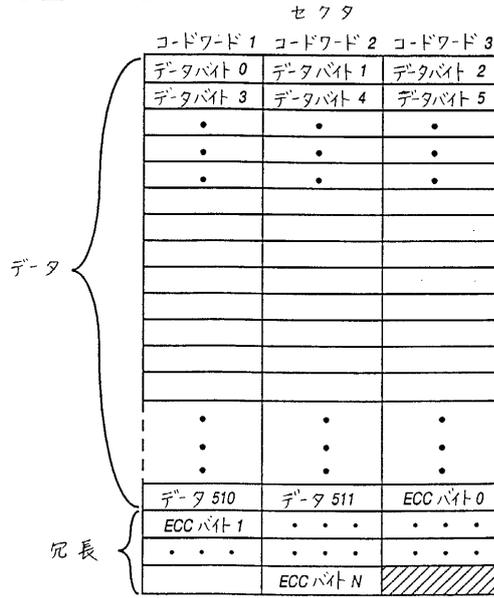
【図8A】



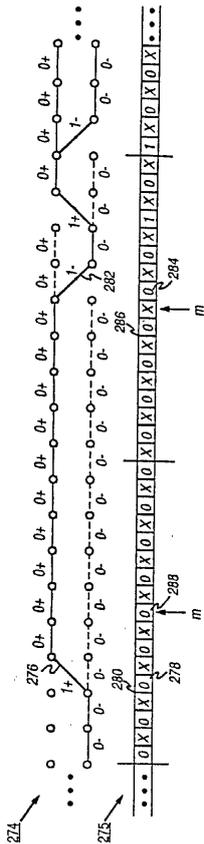
【 図 8 B 】



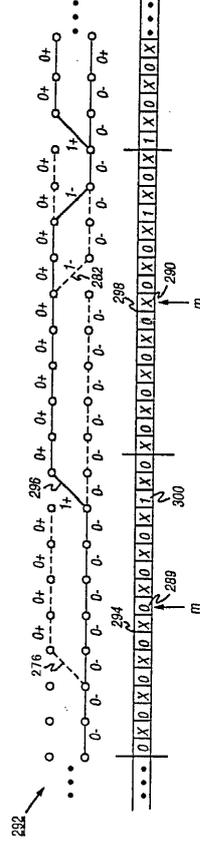
【 図 9 A 】



【 図 9 B 】



【 図 9 C 】



---

フロントページの続き

(56)参考文献 特開平04 - 252472 (JP, A)  
特開平09 - 027171 (JP, A)

(58)調査した分野(Int.Cl., DB名)

G11B 20/14

H03M 7/14

H04L 25/49