

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H03M 13/27 (2006.01)

H03M 13/29 (2006.01)

H03M 13/41 (2006.01)



# [12] 发明专利申请公开说明书

[21] 申请号 200510108431.0

[43] 公开日 2006年4月19日

[11] 公开号 CN 1761160A

[22] 申请日 2005.10.9

[21] 申请号 200510108431.0

[30] 优先权

[32] 2004.10.14 [33] JP [31] 2004-299670

[71] 申请人 恩益禧电子股份有限公司

地址 日本神奈川县

[72] 发明人 织尾正雄

[74] 专利代理机构 中原信达知识产权代理有限责任公司

代理人 穆德骏 陆锦华

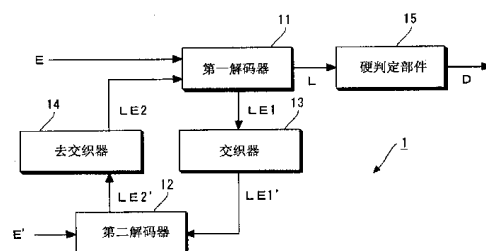
权利要求书 3 页 说明书 16 页 附图 9 页

## [54] 发明名称

解码方法和设备

## [57] 摘要

根据本发明一个实施例的解码设备包括：第一解码器，用于根据通过对未编码数据进行编码而获得的第一编码数据，以及第二软输出数据，来执行第一解码，以生成第一软输出数据；第二解码器，用于根据通过对未编码数据进行交织和对交织的数据进行编码而获得的第二编码数据，以及第一软输出数据，来执行第二解码，以生成第二软输出数据；以及硬判定部件，用于通过对第一软输出数据进行硬判定，来输出解码数据。



1. 一种解码设备，包括：

5 第一解码器，用于根据通过对未编码数据进行编码而获得的第一编码数据以及第二软输出数据，来执行第一解码，以生成第一软输出数据；

第二解码器，用于根据通过对未编码数据进行交织和编码而获得的第二编码数据以及第一软输出数据，来执行第二解码，以生成第二软输出数据；以及

10 硬判定部件，用于通过对第一软输出数据进行硬判定，来输出解码数据。

2. 如权利要求 1 所述的解码设备，其中

15 对由第二解码器生成的第二软输出数据进行去交织，以应用于第一解码器；并且

对由第一解码器生成的第一软输出数据进行交织，以应用于第二解码器。

3. 如权利要求 1 所述的解码设备，其中第一解码器和第二解码器将第二解码和接在第二解码之后的第一解码重复预定次数。

20

4. 如权利要求 1 所述的解码设备，其中第一解码器包括：

向后处理部件，从第一位置向上对数据进行解码；以及

25 向前处理部件，在向后处理部件的解码之后，从第二位置向下对数据进行解码。

5. 如权利要求 1 所述的解码设备，其中第二解码器包括：

向前处理部件，从第二位置向下对数据进行解码；以及

30 向后处理部件，在向前处理部件的解码之后，从第一位置向上对数据进行解码。

6. 如权利要求 1 所述的解码设备, 进一步包括错误检测部件, 用于对从硬判定部件输出的解码数据执行错误检测,

5 其中如果作为错误检测部件的错误检测的结果, 在解码数据中检测到错误, 则第一解码器和第二解码器重复第二解码和第一解码, 并且

如果作为错误检测部件的错误检测的结果, 在解码数据中没有检测到错误, 则第一解码器和第二解码器不重复第二解码和第一解码。

10 7. 一种解码设备, 包括:

第一解码器, 用于根据通过对未编码数据进行编码而获得的第一编码数据来执行解码, 以生成第一软输出数据;

第二解码器, 用于根据通过对未编码数据进行交织和编码而获得的第二编码数据来执行解码, 以生成第二软输出数据; 以及

15 硬判定部件, 用于通过对第一软输出数据进行硬判定, 来输出解码数据。

8. 如权利要求 7 所述的解码设备, 其中第一解码器和第二解码器将第二解码器的解码和接在第二解码器之后的第一解码器的解码重  
20 复预定次数。

9. 如权利要求 7 所述的解码设备, 其中第一解码器包括:  
向后处理部件, 从第一位置向上对数据进行解码; 以及  
25 向前处理部件, 在向后处理部件的解码之后, 从第二位置向下对数据进行解码。

10. 如权利要求 7 所述的解码设备, 进一步包括错误检测部件, 用于对从硬判定部件输出的解码数据执行错误检测,

30 其中如果作为错误检测部件的错误检测的结果, 在解码数据中检测到错误, 则第一解码器和第二解码器重复解码; 并且

如果作为错误检测部件的错误检测的结果，在解码数据中没有检测到错误，则第一解码器和第二解码器不重复解码。

11. 一种解码方法，包括：

5 根据通过对未编码数据进行编码而获得的第一编码数据以及第二软输出数据，来执行第一解码，以生成第一软输出数据；

根据通过对未编码数据进行交织和编码而获得的第二编码数据以及第一软输出数据，来执行第二解码，以生成第二软输出数据；以及通过对第一软输出数据进行硬判定，来输出解码数据。

10

12. 如权利要求 11 所述的解码方法，其中

第一解码使用对通过第二解码生成的第二软输出数据进行去交织而得到的数据；并且

15 第二解码使用对通过第一解码生成的第一软输出数据进行交织而得到的数据。

13. 如权利要求 11 所述的解码方法，其中第二解码和接在第二解码之后的第一解码被重复预定次数。

20 14. 如权利要求 11 所述的解码方法，其中第一解码包括：

向后处理，从第一位置向上对数据进行解码；以及

向前处理，在向后处理之后，从第二位置向下对数据进行解码。

25 15. 如权利要求 11 所述的解码方法，进一步包括对输出的解码数据执行错误检测，

其中如果作为错误检测的结果，在解码数据中检测到错误，则重复第二解码和第一解码，并且

如果作为错误检测的结果，在解码数据中没有检测到错误，则不重复第二解码和第一解码。

30

## 解码方法和设备

## 5 发明领域

本发明涉及一种解码方法和设备。本发明特别涉及用于对第一编码数据和第二编码数据进行重复解码的解码方法和设备。

## 背景技术

10 数字通信系统采用了错误校验代码，用于校验在发送信道中所发生的代码错误。特别是在移动通信系统中，由于因相位调整而导致电波强度大幅度波动，因此很容易发生代码错误。因此，错误校验代码需要较高的校验能力。Turbo 代码作为一种错误校验能力接近 Shannon 极限的错误校验代码，已经引起了人们的注意，并且已经在作为第三代移动通信系统的例如 W-CDMA（宽带码分多址）或 CDMA-2000 中  
15 得到使用。

图 8 为框图，示出了用于生成 turbo 代码的典型编码设备的结构。编码设备 80 位于例如通信系统的发送侧上，并且将作为未编码数据的信息序列 U 编码成 turbo 代码的 PCCC（并行链接卷积代码），以将  
20 结果输出到外部，例如发送信道。

如图 8 所示，编码设备 80 包括作为系统卷积编码器的第一编码器 81 和第二编码器 82，以及用于对数据进行交织（重新排列）的交织器 83。  
25

第一编码器 81 对输入的信息序列（系统部分）U 进行编码，以产生冗余位 P，并且将冗余位 P 输出到外部。交织器 83 将所接收的系统部分 U 的位重新排列成预定交织样式，以产生系统部分 U'。系统  
30 部分 U' 被送到第二编码器 82。第二编码器 82 对系统部分 U' 进行编码，

以产生冗余位 P'，并且将冗余位 P'输出到外部。

因此，编码设备 80 输出系统部分 U、冗余位 P、系统部分 U'和冗余位 P'。系统部分 U 和冗余位 P 对 (U, P) 被称为第一基本码 (编码数据)，而系统部分 U'和冗余位 P' 对 (U', P') 被称为第二基本码 E'。在本申请中，没有加撇号的数据指的是第一基本码或未经交织的数据；加撇号的数据指的是第二基本码或交织数据。

对这样编码的 turbo 代码进行解码被称为 turbo 解码。在对第一基本码 E 进行解码的解码器和对第二基本码 E'进行解码的解码器之间交换外部信息的同时，turbo 解码执行解码重复。例如，在日本未审专利公开 No.2004-80508 中所公开的解码器就是已知的用于 turbo 解码的现有解码器。

图 9 为框图，示出了用于 turbo 解码的现有解码设备的结构。现有解码设备 90 对从图 8 的编码设备 80 提供的第一基本码 E 和第二基本码 E'进行解码。如图 9 所示，现有解码设备 90 包括第一解码器 91，用于对第一基本码 E 进行解码；第二解码器 92，用于对第二基本码 E'进行解码；交织器 93，用于对数据进行交织；去交织器 94 和 95，用于对数据进行去交织（将交织样式重新排列成原始样式）；硬判定部件 96，用于对软输出数据执行硬判定。

第一解码器 91 和第二解码器 92 根据软输入/软输出解码方法对数据进行解码。作为软输入/软输出解码方法，还存在已知的 SOVA（软输出 Viterbi 算法）和 MAP（最大后验法）。例如，第一解码器 91 和第二解码器 92 根据作为 MAP 的例子的 Max-Log-MAP 算法对数据进行解码。

根据 Max-Log-MAP 算法，Viterbi 解码是根据被称为 Trellis 图的状态转换图来进行的。连接在各个时间点上的状态之间的路径与被解

码的代码序列相对应。对所有可能路径都进行似然（路径度量）计算，并且将最大似然路径设定为生存路径，并且使用该路径上的数据序列作为解码数据。

5           图 10 示出了在 Max-Log-MAP 算法中使用的 Trellis 图的例子。根据 Max-Log-MAP 算法，在点 A 上对数据进行解码时，计算自点 A 之前的第二位置（图中的起点）的路径度量和自点 A 之后的第一位置（图中的终点）的路径度量。用于在从 Trellis 图上的起点到终点的方向上（向下）进行 Viterbi 解码以计算路径度量的处理被称为“向前处理”。通过向前处理计算的路径度量被称为 $\alpha$ 路径度量。相反，用于在与向前处理相反的方向上（从终点到起点的方向：向上方向）进行 Viterbi 解码以计算路径度量的处理被称为“向后处理”。通过向后处理计算的路径度量被称为 $\beta$ 路径度量。

10           此外，根据 Max-Log-MAP 算法，在向前处理或向后处理之后，计算用于连接 $\alpha$  路径度量和 $\beta$  路径度量的分支的似然（分支度量），以从 $\alpha$  路径度量、 $\beta$  路径度量和分支度量得到对数似然比率（LLR）和外部信息。对数似然比率或外部信息是作为软输入/软输出解码结果的软输出数据（可信度）。在日本未审专利公开 No.2004-80508 中公开了其计算表达式。

15           图 11 为示意图，示出了基于 turbo 解码的现有解码方法。现有解码方法是通过使用图 9 的现有解码设备来实施的。根据现有解码方法，在对第二基本码 E'进行解码（1102）之前对第一基本码 E 进行解码（1101）。例如，第一解码器 91 使用所接收的第一基本码 E 和由去交织器 94 进行去交织的外部信息 LE2，来对数据进行解码，以将外部信息 LE1 发送到交织器 93。接下来，第二解码器 92 使用所接收的第二基本码 E'和由交织器 93 进行交织的外部信息 LE1'，来对数据进行解码，以将外部信息 LE2'输出到去交织器 94。

20

此外，在对第一基本码 E 进行解码时，在进行向后处理（1104）之前进行向前处理（1103）。同样地，在对第二基本码 E' 进行解码时，在执行向后处理（1106）之前执行向前处理（1105）。然后，将对第一基本码 E 和第二基本码 E' 进行的解码处理定义为一个循环，并且该解码循环被重复 N 次（1107）。

在用于第一基本码 E 和第二基本码 E' 的解码循环被重复 N 次之后，执行去交织处理（1108），接着是硬判定，以输出解码序列 D（1109）。例如，去交织器 95 对由第二解码器 92 计算的对数似然比率 L' 进行去交织，并且硬判定部件 96 对去交织的对数似然比率 L 执行硬判定，以输出解码序列 D。

现已发现，使用现有的解码设备 90，在重复解码 N 次之后，第二解码器 92 输出交织的对数似然比率 L'，因此有必要对该比率进行去交织，以对其执行硬判定。另外，对于第二解码器 92，向后处理接在向前处理之后，因此从数据的尾端到顶端输出对数似然比率 L'。因此，该比率需要以逆序进行重新排列（从顶端到尾端）。

同时，对于 turbo 解码，已知控制第一基本码 E 和第二基本码 E' 的解码重复次数（重复控制）的方法。图 12 为框图，示出了执行重复控制的现有解码设备的结构。如图 12 所示，除了包括图 9 的现有解码设备 90 的组件之外，现有解码设备 120 还包括用于重复控制的 CRC（循环冗余检验）部件 97。

图 13 为示意图，示出了用于重复控制的现有解码方法。现有解码方法是通过使用图 12 的现有解码设备 120 来实施的。除了包括图 11 的解码方法的步骤之外，现有解码方法还包括 CRC 处理。

如图 13 所示，在对第一基本码 E 进行解码处理（1110）期间，CRC 检验与向前处理（1103）同时进行。例如，硬判定部件 96 对由去交



织器 94 进行了去交织的外部信息 LE2 执行硬判定，并且 CRC 部件 97 对作为硬判定结果的解码序列 D 执行 CRC。作为 CRC 的结果，如果发现了错误，则重复进行解码。在解码重复了 N 次之后，如图 11 所示，对对数似然比率 L' 进行去交织，接着对对数似然比率 L 进行硬判定，然后再次执行 CRC 检验（1111）。

5

如果解码重复中作为 CRC 检验的结果，没有检测出错误，则不再重复解码。然后，对进行了 CRC 的外部信息 LE2 进行去交织，然后对其执行硬判定，以输出解码序列 D。

10

现在已经发现，根据现有解码设备 120，对关于由去交织器 94 进行了去交织的外部信息 LE2 的硬判定结果进行 CRC 检验。通常，外部信息不是用于硬判定以生成解码序列的数据。因此，存在重复控制准确性低的问题。另一方面，为了使用对数似然比率 L' 而非外部信息 LE2，需要一个存储容量较大的存储器，用于暂时存储对数似然比率 L。

15

此外，用于 CRC 的外部信息 LE2 是在前一个解码重复（循环）中解码的第二基本码 E'。为了对外部信息 LE2 执行 CRC，外部信息 LE2 应该被去交织，因此又消耗了用于该去交织的处理时间（一个解码循环（解码重复）所需时间段的一半）。

20

如上所述，根据现有解码设备 90，在解码被重复 N 次之后，需要对交织的对数似然比率 L' 进行去交织或者重新排列逆序的用于硬判定的数据。因此，去交织处理或重新排列占多了大量时间，延长了解码处理。

25

### 发明内容

根据本发明的解码设备包括：第一解码器，用于根据通过对未编码数据进行编码而获得的第一编码数据，以及第二软输出数据，来执

30

行第一解码，以生成第一软输出数据；第二解码器，用于根据通过对未编码数据进行交织和编码而获得的第二编码数据，以及第一软输出数据，来执行第二解码，以生成第二软输出数据；以及硬判定部件，用于通过对第一软输出数据进行硬判定来输出解码数据。因此，可能  
5 够略去交织或重新排列处理，并且缩短解码处理所需的时间。

根据本发明的解码设备包括：第一解码器，用于根据通过对未编码数据进行编码而获得的第一编码数据，来执行解码，以生成第一软输出数据；第二解码器，用于根据通过对未编码数据进行交织和编码  
10 而获得的第二编码数据，来执行解码，以生成第二软输出数据；以及硬判定部件，用于通过对第一软输出数据进行硬判定来输出解码数据。因此，能够省略去交织或重新排列处理，并且缩短解码处理所需的时间。

根据本发明的解码方法，包括：通过对未编码数据进行编码而获得的第一编码数据以及第二软输出数据，来执行第一解码，以生成第一软输出数据；根据通过对未编码数据进行交织和编码而获得的第二编码数据以及第一软输出数据，来执行第二解码，以生成第二软输出数据；以及通过对第一软输出数据进行硬判定，来输出解码数据。因此，能够省略去交织或重新排列处理，并且缩短解码处理所需的时间。  
15  
20

根据本发明，能够提供一种能够缩短解码处理所需时间的解码方法和设备。

## 25 附图说明

从结合附图的如下讲述中，本发明的上述和其它目标、优点和特征将更加清楚，其中：

图 1 为框图，示出了根据本发明的解码设备的结构；

图 2A 和 2B 为框图，每一个示出了根据本发明的解码设备的结构；  
30

- 图 3 示出了根据本发明的解码方法；  
图 4 为框图，示出了根据本发明的解码设备的结构；  
图 5 示出了根据本发明的解码方法；  
图 6 是示出了根据本发明的解码设备的模拟结果的图；  
5 图 7 是示出了根据本发明的解码设备的模拟结果的图；  
图 8 为框图，示出了典型解码设备的结构；  
图 9 为框图，示出了现有解码设备的结构；  
图 10 示出了典型解码器中所采用的 Trellis 图的例子；  
图 11 示出了现有的解码方法；  
10 图 12 为框图，示出了现有解码设备的结构；以及  
图 13 示出了现有的解码方法；

### 具体实施方式

现在将参考解释性实施例在此说明本发明。本领域技术人将认识到，使用本发明的讲述还可以实现许多可选的实施例，并且本发明并不限于用于讲解目的的实施例。

#### 本发明的第一实施例

首先，参照图 1~图 3 来讲述根据本发明第一实施例的解码方法和设备。根据本实施例的解码方法和设备的特征是，依次对第二基本码  $E'$  和第一基本码  $E$  进行重复解码，并且解码重复在完成用于第一基本码  $E$  的解码处理中的向前处理时结束。

现在参照图 1 来讲述根据本实施例的解码设备的结构。解码设备 1 用于 turbo 解码，并且例如对来自图 8 的编码设备 80 的第一基本码  $E$  和第二基本码  $E'$  进行解码。也就是说，解码设备 1 接收通过对信息序列  $U$ （未编码数据）进行编码而获得的第一基本码  $E$ （第一编码数据），以及通过对被编码的信息序列  $U$  进行交织而得到的第二基本码  $E'$ （第二编码数据）。第一基本码  $E$  和第二基本码  $E'$  通过发送信道被输入到解码设备 1，并且在发送信道上噪音被叠加于其上，因此会发

生代码错误。解码设备 1 对第一基本码 E 和第二基本码 E'进行 turbo 解码，并且以其高的校验能力来校验代码错误。

5 如图 1 所示，解码设备 1 包括第一解码器 11、第二解码器 12、交织器 13、去交织器 14 和硬判定部件 15。

10 第一解码器 11 与编码设备 80 的第一编码器 81 相对应，并且对第一基本码 E 进行解码。第一解码器 11 为能够根据 SOVA 和 MAP 进行解码的软输入/软输出解码器。第一解码器 11 根据例如 Max-Log-MAP 算法来解码数据。通过解码，产生了软输出数据。软输出数据为表示可信度的数据，也就是说，概率为“0”或“1”（似然）。例如，如果软输出数据为较大的正值，则为“0”的概率较高。对之对照，如果软输出数据为较大的负值，则为“1”的概率较高。根据 Max-Log-MAP 算法，外部信息或对数似然比率是作为软输出数据来生成的。外部信息是解码重复中在第一解码器 11 和第二解码器 12 之间交换的数据。  
15 对数似然比率为要经过硬判定以生成解码序列的数据。

20 第一解码器 11 使用接收的第一基本码 E 和由去交织器 14 进行去交织的外部信息 LE2 来执行解码，以输出通过解码得到的外部信息 LE1。另外，第一解码器 11 与第二解码器 12 一起重复预订次数的解码，然后将通过解码得到的对数似然比率 L 输出到硬判定部件 15。在解码循环（解码重复）中，第二解码器 12 在第一解码器 11 执行解码之前执行解码，并且该循环被重复 N 次。

25 第二解码器 12 与编码设备 80 的第二编码器 82 相对应，并且对第二基本码 E'进行解码。第二解码器 12 是与第一解码器 11 相类似的软输入/软输出解码器。此外，第二解码器 12 使用接收的第二基本码 E'和由交织器 13 进行交织的外部信息 LE1'来执行解码，以输出通过解码得到的外部信息 LE2'。

30

交织器 13 将数据交织成预定交织样式。例如，交织器 13 具有存储器，因此能够通过将数据写入存储器/从存储器读取数据，来对数据进行交织。在将数据写入存储器时，数据可以根据交织样式来进行重新排列，在从存储器读取数据时，数据可以从顶端依次被读取。可选地，在将数据写入存储器时，数据可以从顶端依次被写入，在从存储器读取数据时，数据可以根据交织样式来进行读取。另外，交织器 13 对从第一解码器 11 输出的外部信息 LE1 进行交织，以将外部信息 LE1' 输出到第二解码器 12。

去交织器 14 将交织的数据去交织成原始的数据排列。与交织器 13 类似，去交织器 14 能够通过将数据写入存储器/从存储器读取数据，来对数据进行去交织。在将数据写入存储器时，数据可以根据与交织样式相反的去交织样式来进行重新排列，并且在从存储器读取数据时，数据可以从顶端依次被读取。此外，在将数据写入存储器时，数据可以从顶端依次被写入，并且在从存储器读取数据时，数据可以根据去交织样式来进行读取。另外，去交织器 14 对从第二解码器 12 输出的外部信息 LE2' 进行去交织，以将外部信息 LE2 发送到第一解码器 11。

硬判定部件 15 对软输出数据执行硬判定，以生成解码序列。在硬判定中，根据诸如软输出数据等呈现给定值的数据来选择诸如“0”或“1”的预定数据。例如，如果软输出数据近似为“0”，则数据被确定为“0”；如果数据近似为“1”，则数据被确定为“1”。

此外，硬判定部件 15 重复解码之后对从第一解码器 11 提供的对数似然比率 L 执行硬判定，以在输出硬判定结果作为解码序列 D。在本实施例中，由于从第一解码器 11 而非第二解码器 12 提供的对数似然比率 L 直接进行硬判定，因此没有必要通过去交织来重新排列对数似然比率 L。

30

接下来，参照图 2A 和 2B 来讲述根据本实施例的解码设备的结构。图 2A 示出了第一解码器 11 的结构，图 2B 示出了第二解码器 12 的结构。

5           如图 2A 所示，第一解码器 11 包括向前处理部件 21、向后处理部件 22、存储器 23 和外部信息计算部件 24。存储器 23 存储向前处理部件 21、向后处理部件 22 和外部信息计算部件 24 中的处理所需要的信息。

10           向后处理部件 22 使用第一基本码 E 和由去交织器 14 进行去交织的外部信息 LE2，来执行向后处理。向后处理部件 22 从实际解码位置后面的位置（终点）起向上参考外部信息 LE2 和第一基本码 E，以从 Trellis 图的终点起计算 $\beta$ 路径度量。然后，向后处理部件 22 将第一基本码 E 和用于向后处理的外部信息 LE2，以及通过向后处理计算的  
15            $\beta$ 路径度量存储在存储器 23 中。

            向前处理部件 21 使用通过向后处理部件 22 存储在存储器 23 中的第一基本码 E 和外部信息 LE2，来执行向前处理。向前处理部件 21 从实际解码点前面的位置（起点）起向下参考外部信息 LE2 和第一基本码 E，以从 Trellis 图的起点起计算 $\alpha$ 路径度量。然后，向前处理部件 21 将通过向前处理计算的 $\alpha$ 路径度量存储在存储器 23 中。  
20

            外部信息计算部件 24 使用通过向后处理部件 22 和向前处理部件 21 存储在存储器 23 中的 $\alpha$ 路径度量和 $\beta$ 路径度量，来计算对数似然比率 L，并且得到外部信息 LE1。例如，外部信息计算部件 24 计算连接  
25            $\alpha$ 路径度量和 $\beta$ 路径度量的分支度量，以从 $\alpha$ 路径度量、 $\beta$ 路径度量和分支度量得到对数似然比率 L，并且基于对数似然比率 L 获取外部信息 LE1。然后，外部信息计算部件 24 输出计算出的对数似然比率 L 或外部信息 LE1。

30

例如，外部信息计算部件 24 按照其中 $\alpha$ 路径度量和 $\beta$ 路径度量分别计算的次序来获取对数似然比率 L 和外部信息 LE1，并且按照所述的次序来输出对数似然比率 L 或外部信息 LE1。在第一解码器 11 中，在向后处理部件 22 执行向后处理之后，向前处理部件 21 执行向前处理。因此，在计算出 $\beta$ 路径度量之后，从 Trellis 图的起点起向下计算 $\alpha$ 路径度量。因此，对数似然比率 L 和外部信息 LE1 是以该次序（也就是从起点向下）来计算的，以按该次序来输出对数似然比率 L 或外部信息 LE1。因此，对数似然比率 L 是从数据的顶端输出的，这使得能够在不对数据进行重新排列的情况下直接让数据经过硬判定。

如图 2B 所示，第二解码器 12 包括向前处理部件 21'、向后处理部件 22'、存储器 23'和外部信息计算部件 24'。第二解码器 12 的各个组件与第一解码器 11 的各组件是相同的。这里，第二解码器 12 与第一解码器的不同之处在于向前处理部件 21 和向后处理部件 22 之间的数据输入/输出关系，也就是执行向前处理和向后处理的顺序。

向前处理部件 21'从起点起向下参考所接收的第二基本码 E'和外部信息 LE1'来计算 $\alpha$ 路径度量，以将第二基本码 E'、外部信息 LE1'和 $\alpha$ 路径度量存储在存储器 23'中。向后处理部件 22'从终点起向上参考通过向前处理部件 21' 存储在存储器 23'中的外部信息 LE1'和第二基本码 E'，来计算 $\beta$ 路径度量，以将所计算出的 $\beta$ 路径度量存储在存储器 23'中。与外部信息计算部件 24 类似，外部信息计算部件 24'得到外部信息 LE2'。

在第二解码器 12 中，在向后处理部件 22'执行向后处理之前，向前处理部件 21'执行向前处理。因此，在计算出 $\alpha$ 路径度量之后，从 Trellis 图的终点起向上计算 $\beta$ 路径度量。因此，外部信息 LE2'是以该次序（也就是从终点向上）得到的，以按该次序输出外部信息 LE2'。

接下来参照图 3 来讲述根据本实施例的解码方法。本解码方法是

由图 1 的解码设备 1 来实施的。也就是说，该解码方法对 turbo 编码的第一基本码 E 和第二基本码 E' 进行重复解码。

5 根据本解码方法，首先，对第二基本码 E' 进行解码 (301)。在该解码中，向后处理 (304) 接在向前处理 (303) 之后。例如，如果第二解码器 12 接收第二基本码 E' 和外部信息 LE1'，则向前处理部件 21' 执行向前处理，其后向后处理部件 22' 执行向后处理。然后，外部信息计算部件 24' 根据通过向前处理和向后处理计算的路径度量，来获取对数似然比率 L 和外部信息 LE2'。用于第二基本码 E' 的解码次序并不限于此，并且向前处理可以接在向后处理之后。

10 此外，如果第二基本码 E' 是第一次进行解码，则还没有获取外部信息 LE1'，因此使用初始值。如果是第二次和后续次解码，则使用通过第一解码器 11 获取的外部信息 LE1'。

15 接下来，对第一基本码 E 进行解码 (302)。在该解码中，向前处理 (306) 接在向后处理 (305) 之后。例如，当第一解码器 11 接收第一基本码 E 和外部信息 LE2，则在向前处理部件 21 执行向前处理之前，向后处理部件 22 执行向后处理。然后，外部信息计算部件 24 根据通过向后处理和向前处理计算的路径度量，来获取对数似然比率 L 和外部信息 LE1。

20 接下来，用于第二基本码 E' 和第一基本码 E 的解码处理被定义为一个循环，并且该循环被重复 N 次 (307)。代码错误校验能力是与解码循环的次数成比例提高的；解码循环的次数为例如 8~10。

25 在 N 次解码循环之后执行硬判定 (308)。例如，第一解码器 11 对第一基本码 E 所进行的解码次数进行计数，并且当计数值达到 N 时，将相应的对数似然比率 L 输出到硬判定部件 15。硬判定部件 15 对从第一解码器 11 提供的对数似然比率 L 进行硬判定，以输出解码序列



D。

5 这样，根据本实施例，在重复解码之后，第一解码器输出对数似然比率，因此对数似然比率可以在不进行重新排列的情况下直接经过硬判定。也就是说，与从第二解码器输出的对数似然比率不同，没必要进行去交织处理。此外，在第一解码器中，向前处理是在向后处理之后进行，以输出对数似然比率，这就消除了对逆序的对数似然比率进行重新排列的必要。因此，可以略过去交织或重新排列处理，以减少解码时间，另外还可节约功耗。这一点对于诸如蜂窝电话等需要低  
10 功耗的电子设备是尤为有效的。

例如，去交织处理需要一个解码循环所需时间的一半。然而，本实施例能够省却该处理。在 8 个解码循环的情况下，现有例子（图 9）需要与 8.5 个解码循环相对应的处理时间。然而，本实施例（图 1）  
15 只需要与 8 个解码循环相对应的处理时间，因此可以节约与 0.5 个解码循环相对应的处理时间（约 0.6%）。

#### 本发明的第二实施例

20 下面参照图 4 至图 7 来讲述根据本发明第二实施例的解码设备和方法。根据本实施例的解码设备和方法所具有的特点是，解码的第一基本码 E 要经过硬判定，并且进一步要进行用于重复控制的 CRC。

现在参照图 4 来讲述根据本实施例的解码设备的结构。在图 4 中，与图 1 中相同的组件用相似的标号来表示，并且在此省略其详细讲述。  
25

如图 4 所示，除了图 1 的组件之外，解码设备 1 还包括 CRC 部件 16。CRC 部件 16 为错误检测部件，用于对用于错误检测的解码序列 D 执行 CRC 判定。

30 接下来参照图 5 来讲述根据本实施例的解码方法。该解码方法是

由图 4 的解码设备 1 来执行的。另外，除了图 3 的解码方法的步骤之外，该解码方法还包括 CRC。除了 CRC 之外，图 5 的解码方法与图 3 的解码方法是相同的，因此只对其进行简要讲述。

5            与图 3 的解码方法类似，本解码方法解码第二基本码 E' (301) 和第一基本码 E (302)，该解码被重复 N 次 (307)。在本实施例中，每一次重复解码，都根据对第一基本码 E 进行解码的结果来进行 CRC (309)。例如，在每一个解码处理中，第一解码器 11 执行向后处理 (305) 和向前处理 (306)，以将外部信息 LE1 输出到交织器 13 并且  
10            将对数似然比率 L 输出到硬判定部件 15。硬判定部件 15 对对数似然比率 L 执行硬判定，以将解码序列 D 输出到 CRC 部件 16。然后，CRC 部件 16 对用于错误检测的解码序列 D 执行 CRC 计算。

             如果作为 CRC 的结果没有发现错误，则在不进行重复解码的情况下输出解码序列 D。相反，如果作为 CRC 的结果发现了错误，则  
15            连续地重复解码。例如，如果 CRC 计算值与解码序列 D 中的 CRC 值一致，则 CRC 部件 16 确定不存在错误并输出解码序列 D。如果 CRC 计算值与 CRC 值不一致，则 CRC 部件通知第一解码器 11 发现了错误。作为对该通知的响应，只要检测到错误，第一解码器 11 和第二解码器 12 就连续地进行重复解码。  
20            在解码被重复 N 次之后，与图 3 相类似，对从第一解码器 11 提供的对数似然比率 L 进行硬判定，以对解码序列 D 执行 CRC 判定，并且输出得到的解码序列 D。

25            这样，根据本实施例，通过用第一解码器对第一基本码 E 进行解码所获得的对数似然比率 L 经过硬判定和随后的 CRC 判定，从而与对外部信息执行硬判定和 CRC 检验的现有情况相比，提高了重复控制的准确度。也就是说，CRC 判定是根据用于实际获取解码序列的对数似然比率 L 进行的，因此基于 CRC 的错误检测可信度得到提高，  
30

并且能够以较高的可信度来减少解码循环的次数。因此，能够缩短解码时间，并且节约功耗。

此外，从第一解码器输出的对数似然比率  $L$  可以直接经过硬判定和 CRC，从而能够省略去交织处理，来更大程度地缩短解码时间。特别地，每一次重复解码时都执行用于重复控制的 CRC，因此缩短时间的效果大并且与解码循环的次数成正比。

图 6 和图 7 示出了根据本实施例的解码设备和方法的模拟结果。

10

图 6 示出了解码设备的代码错误率。图 6 的横轴表示  $E_b/N_0$ 。 $E_b/N_0$  为信噪比 (SNR)，换句话说，即每比特的信号能量  $E_b$  与基于频带的白噪声功率密度  $N_0$  的比率。小  $E_b/N_0$  表示较大的噪声成分。大  $E_b/N_0$  表示较小的噪声成分。图 6 的纵轴代表表示基于位的错误率的位错误率 (BER) 和表示基于包括有多个位的块的错误率的块错误率 (BLER)。

15

在图 6 中，(a-1) 表示现有技术例子 1 (图 9 和 11) 的 BER 特性，也就是在执行固定次数的解码循环的情况下的 BER 特性，(b-1) 表示现有技术例子 2 (图 12 和 13) 的 BER 特性，(c-1) 表示本实施例 (图 4 和 5) 的 BER 特性。此外，(a-2) 表示现有技术例子 1 的 BLER 特性，(b-2) 表示现有技术例子 2 的 BLER 特性，(c-2) 表示本实施例的 BLER 特性。

20

如图 7 所示，现有技术例子 1、现有技术例子 2 和本实施例显示出几乎相同的 BER 和 BLER 特性。这是因为错误率特性的改善是与解码循环的次数成比例的，因此，在重复给定次数的解码之后，在每一种情况下都会得到类似的特性。

25

图 7 示出了图 6 的代码错误特性中的解码循环的次数，换句话说，

30

在每一个  $E_b/N_0$  处的解码循环的次数。在图 7 中，(a) 表示现有技术例子 1 的解码循环的次数，(b) 表示现有技术例子 2 的解码循环的次数，(c-2) 表示本实施例的解码循环的次数。

5            在图 6 中，各个情况都显示出类似的特性，但是如图 7 所示，在解码循环的次数上是各不相同的。如图 7 所示，在现有技术例子 1 中，在每一个  $E_b/N_0$  处解码循环的次数被固定为 8。在现有技术例子 2 中，解码循环的次数小于现有技术例子 1 的次数；解码循环的次数从约为 8 减小到约为 4.5，与  $E_b/N_0$  成反比。本实施例的解码循环的次数在所有情况中是最小的，并且从约为 7.5 减小到约为 3.5，与  $E_b/N_0$  成反比。  
10           例如，以  $BLER=10^{-2}$  为参考，从图 6 可以知道，此时  $E_b/N_0$  为 0.8。从图 7 可以看出，与现有技术例子 2 中的约 5.5 相比，在本实施例中，在  $E_b/N_0$  为 0.8dB 处的解码循环的次数约为 4。也就是说，显示出能够减少与平均 1.5 次解码循环相对应的解码时间（约为 27%）。

15

#### 本发明的其它实施例

上述实施例讲述了其中对第二基本码  $E'$  和第一基本码  $E$  进行重复解码的例子，但是本发明并不限于此，并且可以采用其中对从第一解码器输出的对数似然比率进行硬判定的任何结构。

20

此外，上述实施例讲述了其中使用两个解码器对数据进行重复解码的例子，但是本发明并不限于此，并且可以使用给定个数的解码器来进行重复解码。

25

很明显，本发明并不限于上述实施例，并且在不偏离本发明的范围和精神的情况下可以对其进行修正和改变。

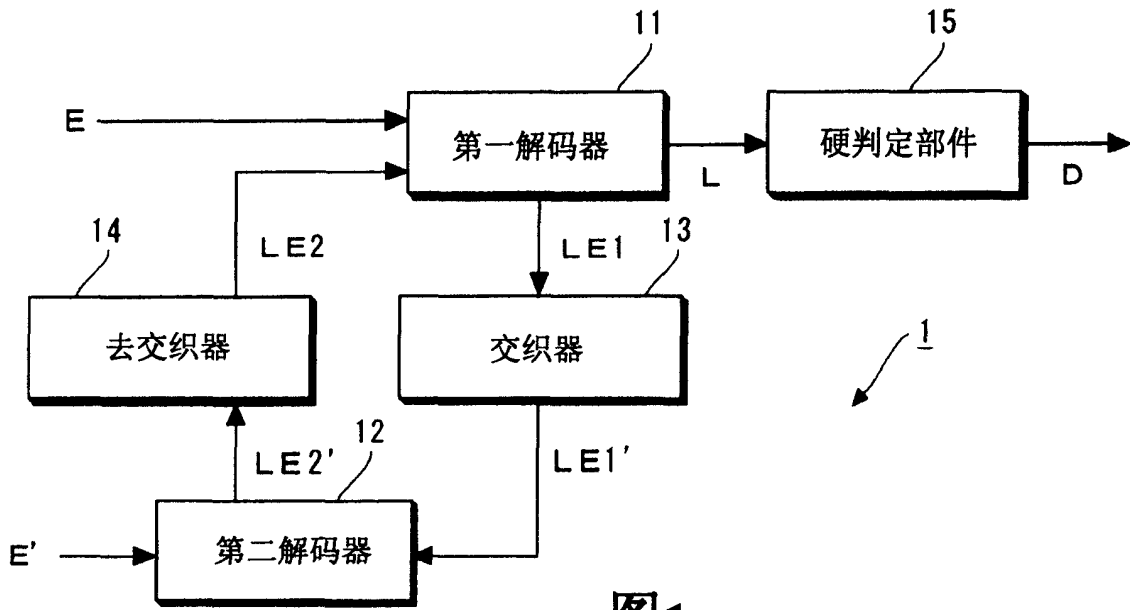


图1

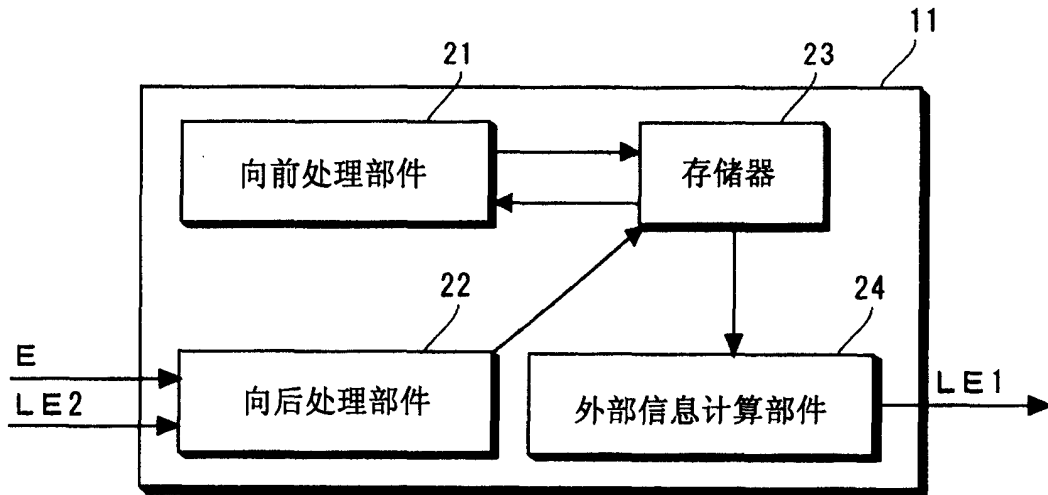


图2A

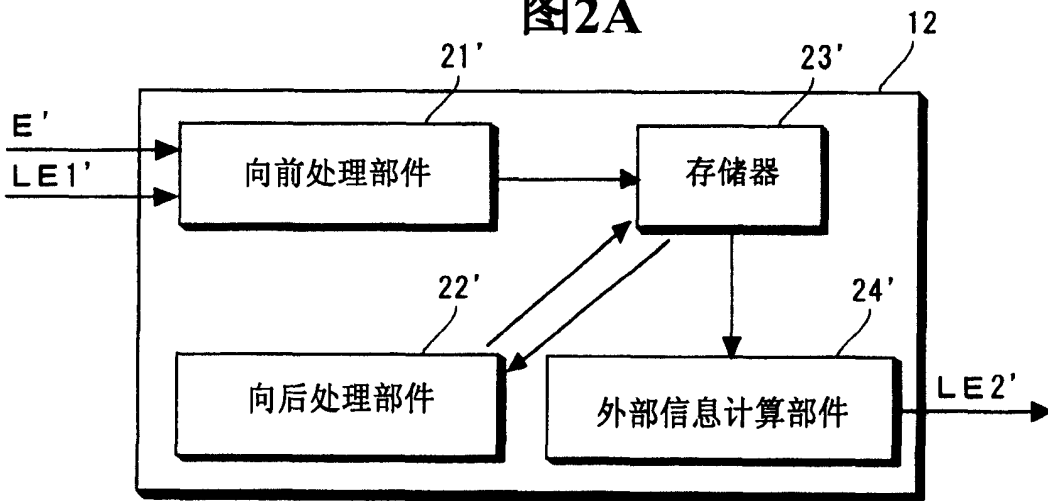


图2B

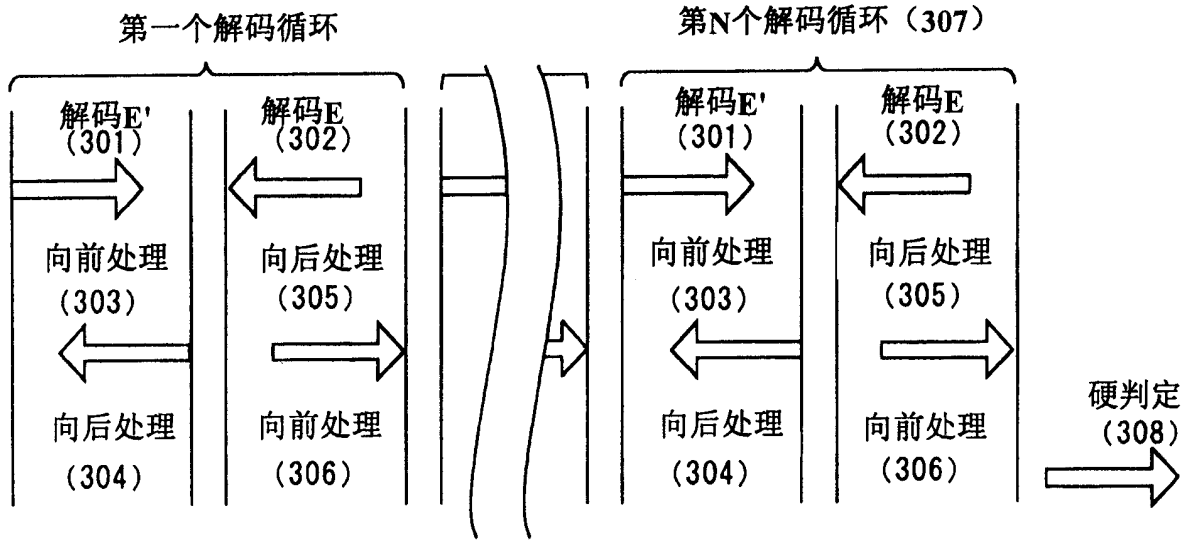


图3

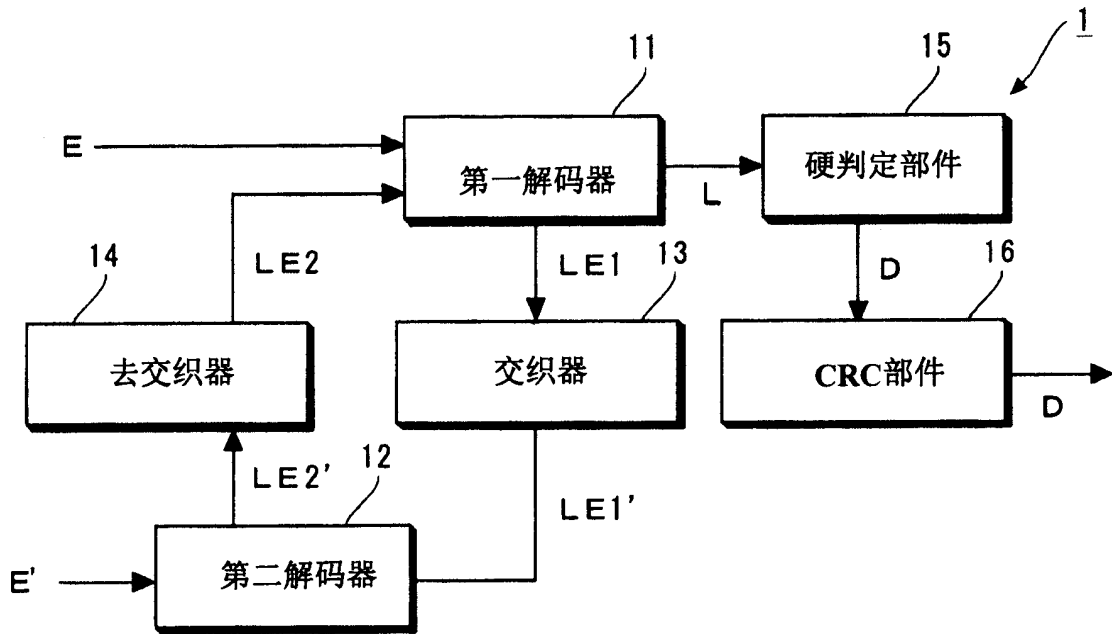


图4

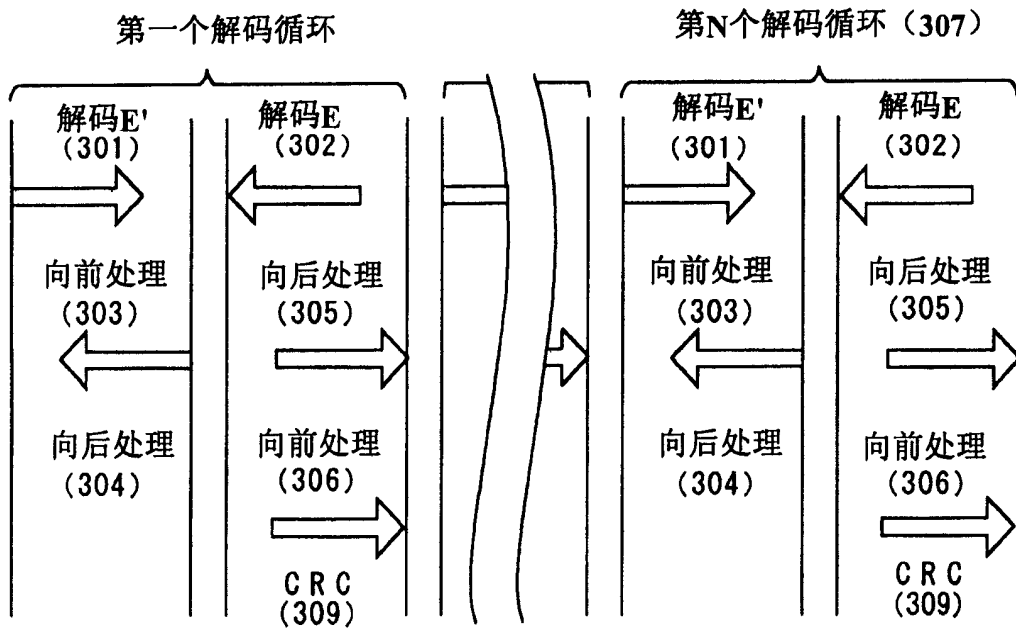


图5

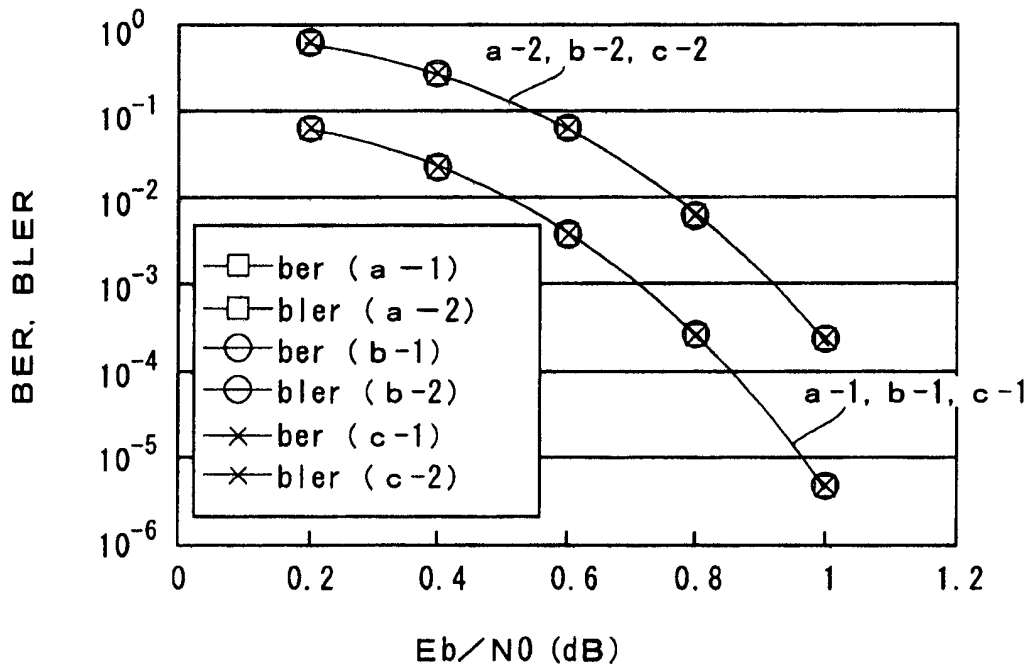


图6

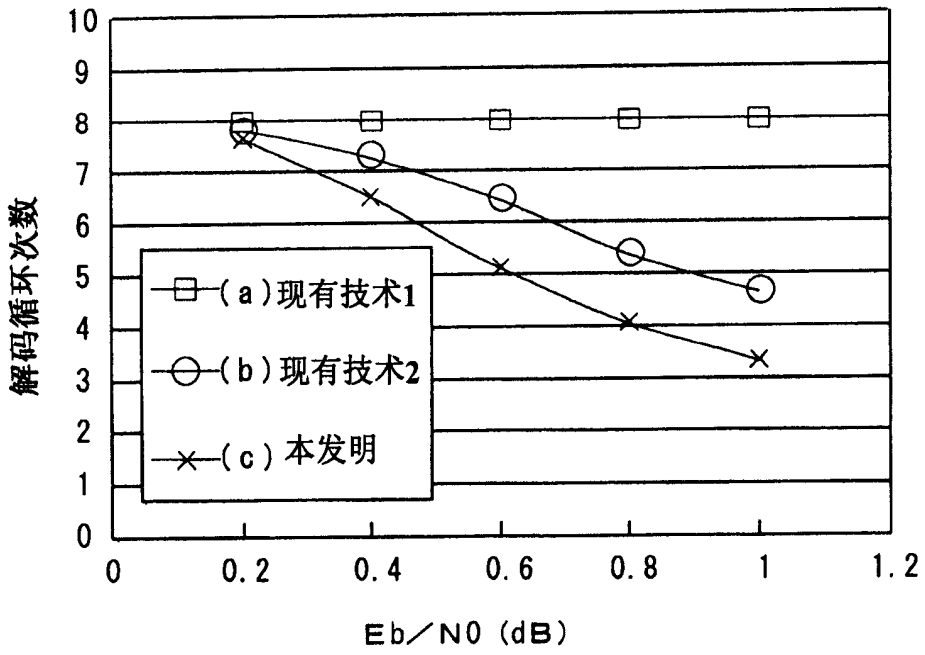


图7

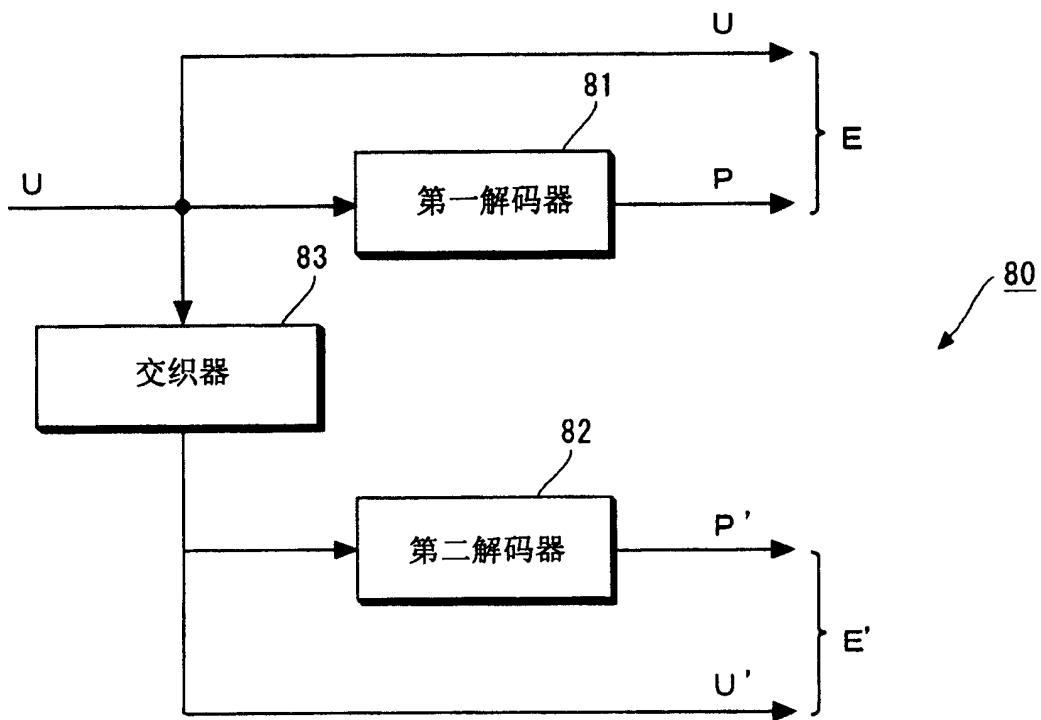


图8 现有技术



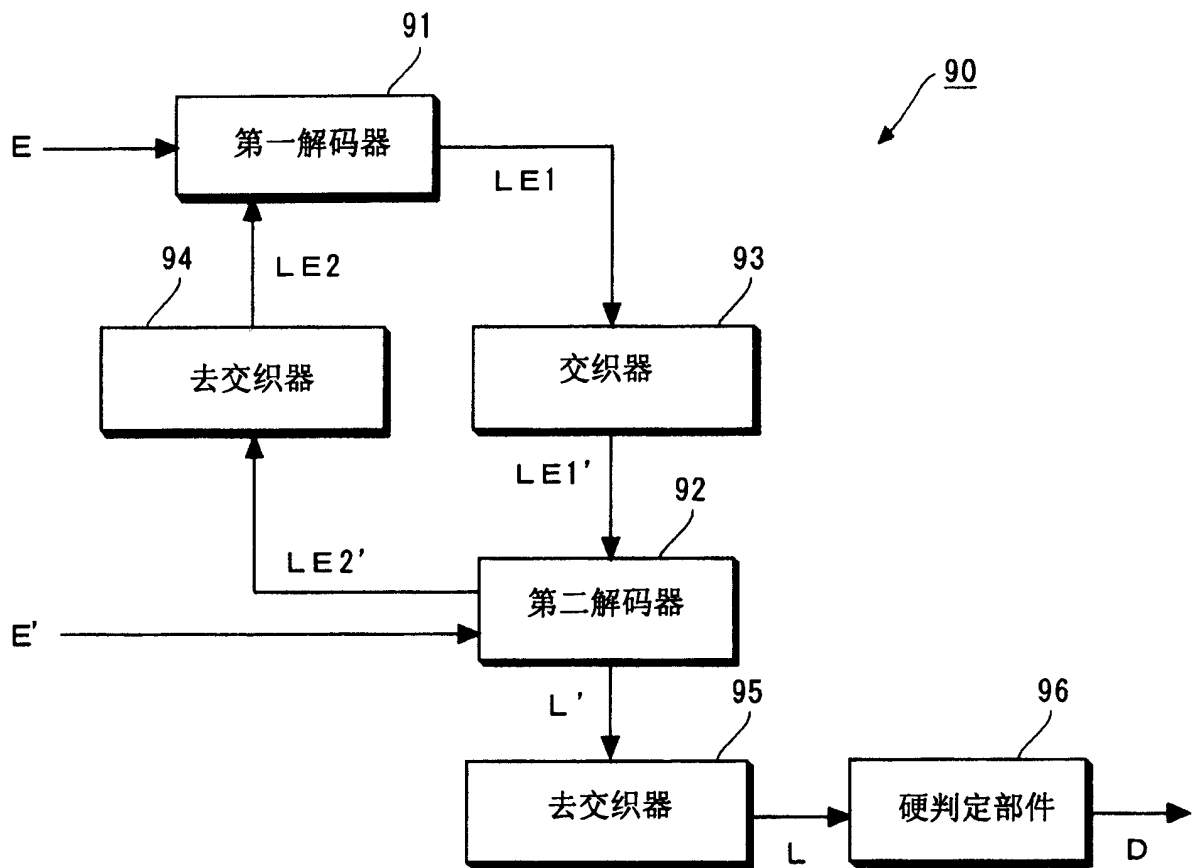


图9 现有技术

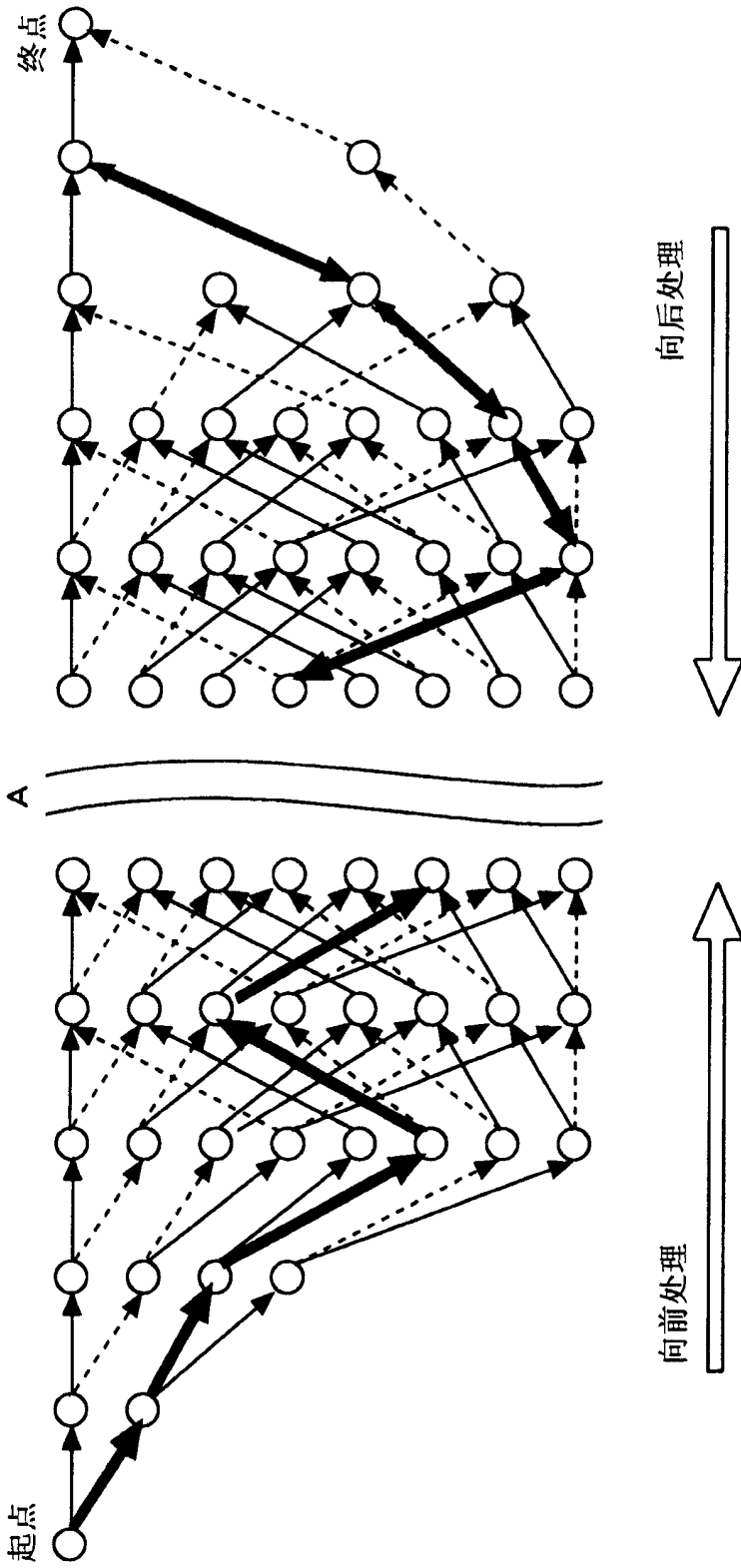


图10 现有技术

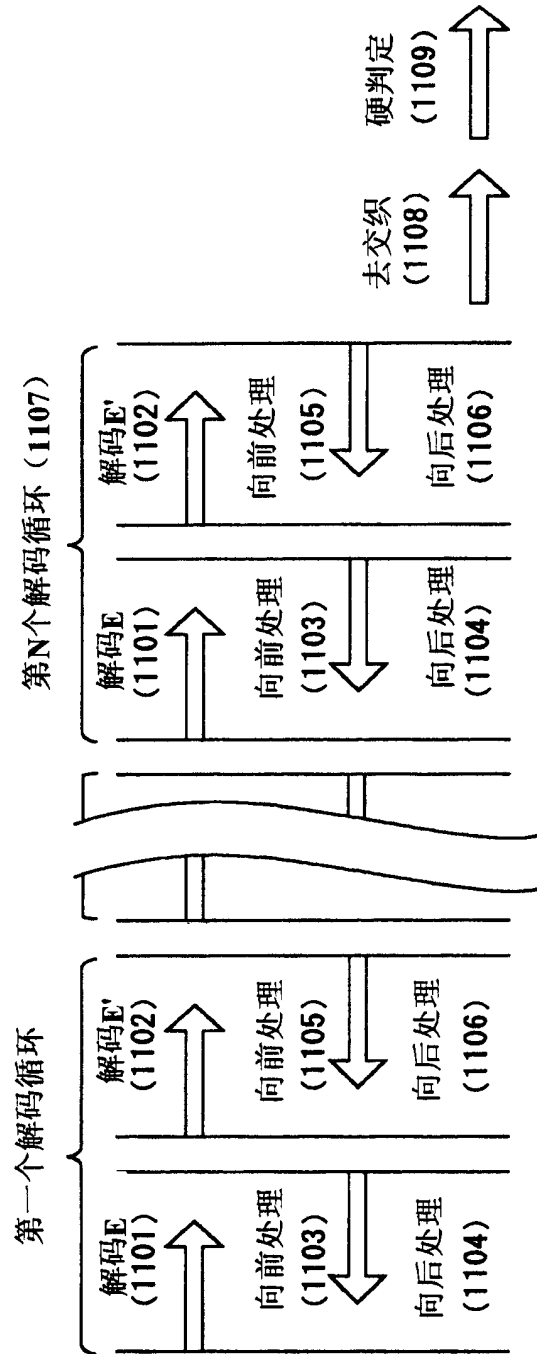


图11 现有技术

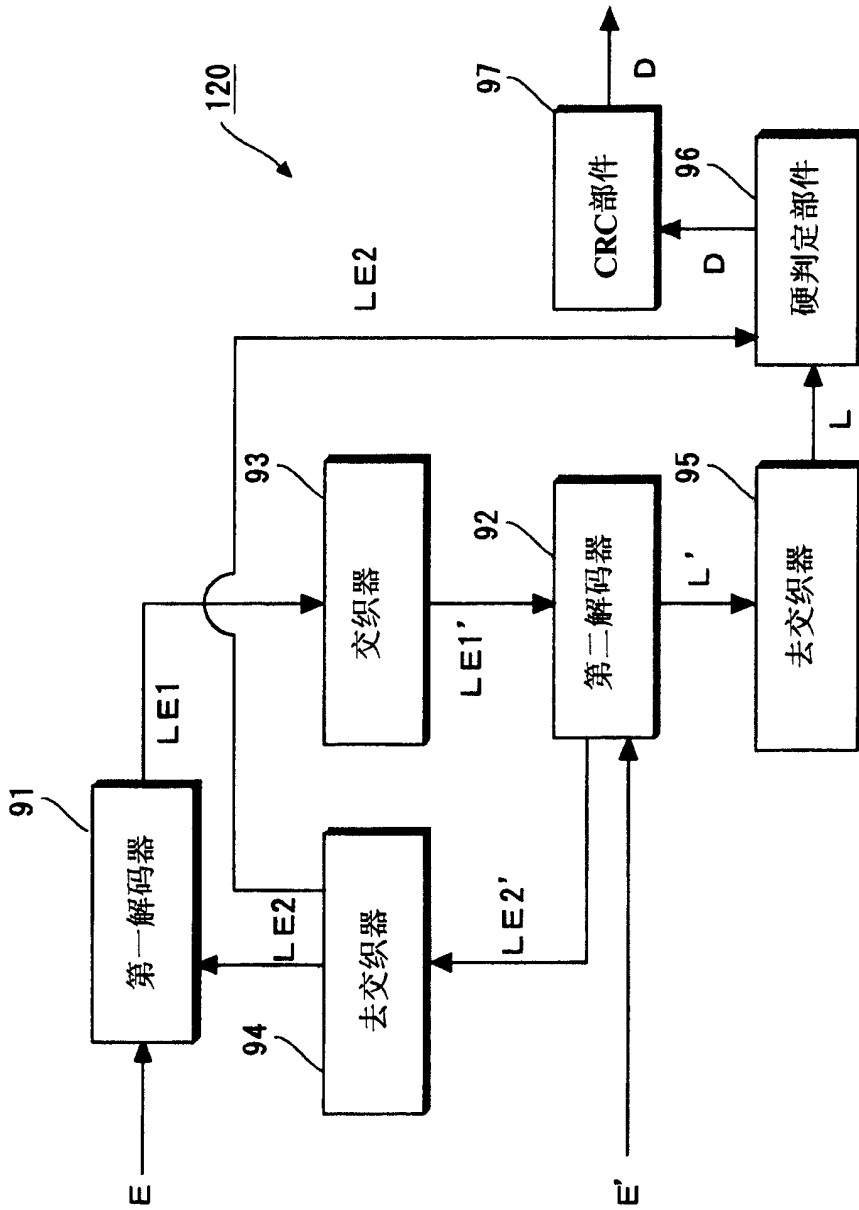


图12 现有技术

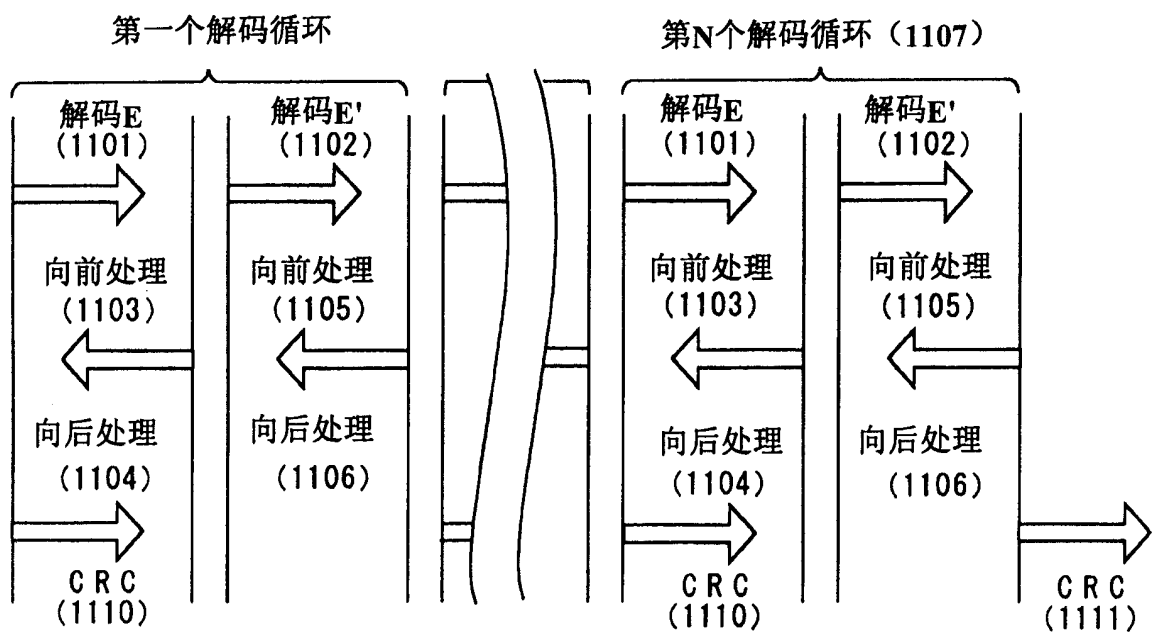


图13 现有技术