



(12) 发明专利

(10) 授权公告号 CN 109787707 B

(45) 授权公告日 2021.05.18

(21) 申请号 201711105339.8

(22) 申请日 2017.11.10

(65) 同一申请的已公布的文献号
申请公布号 CN 109787707 A

(43) 申请公布日 2019.05.21

(73) 专利权人 华为技术有限公司
地址 518129 广东省深圳市龙岗区坂田华为总部办公楼

(72) 发明人 王桂杰 张公正 童佳杰

(74) 专利代理机构 北京龙双利达知识产权代理有限公司 11329

代理人 时林 毛威

(51) Int. Cl.

H04L 1/00 (2006.01)

H03M 13/27 (2006.01)

(56) 对比文件

EP 2048789 A1, 2009.04.15

EP 2790328 A1, 2014.10.15

CN 105490776 A, 2016.04.13

CN 103401571 A, 2013.11.20

审查员 杨柳依

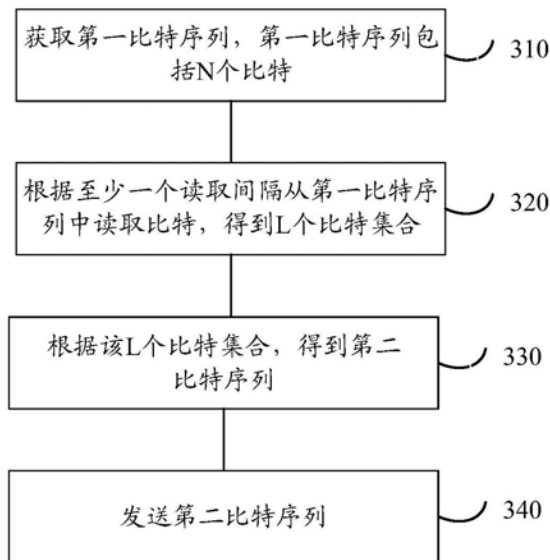
权利要求书1页 说明书12页 附图5页

(54) 发明名称

交织方法和交织装置

(57) 摘要

本申请提供了一种交织方法,能够在不增加交织复杂度的情况下,提升纠错性能。该方法包括:获取第一比特序列,所述第一比特序列包括N个比特;根据至少一个读取间隔从所述第一比特序列中读取比特,得到L个比特集合,每个比特集合中包括所述第一比特序列中的至少一个比特,任意两个比特集合中包括的比特在所述第一比特序列中的索引不同,所述L个比特集合中比特数目之和等于所述N,其中, $N \geq 2$, $L \geq 2$,且N和L为整数;根据所述L个比特集合,得到第二比特序列;发送所述第二比特序列。



1. 一种交织方法,其特征在于,包括:

获取第一比特序列,所述第一比特序列包括N个比特, $N \geq 2$ 且为整数;

根据至少一个读取间隔从所述第一比特序列中读取比特,得到L个比特集合,每个比特集合中包括所述第一比特序列中的至少一个比特,任意两个比特集合中包括的比特在所述第一比特序列中的索引不同,所述L个比特集合中比特数目之和等于所述N,所述至少一个读取间隔中的任意一个读取间隔是根据调制阶数或预设函数确定的,其中, $L \geq 2$ 且为整数;

根据所述L个比特集合,得到第二比特序列;

发送所述第二比特序列;

所述根据至少一个读取间隔从所述第一比特序列中读取比特,得到L个比特集合,包括:根据第i个读取间隔,从所述第一比特序列中除去第1个比特集合至第(i-1)个比特集合后的剩余比特中读取至少一个比特,得到第i个比特集合,其中, $1 \leq i \leq L$,且i为整数。

2. 根据权利要求1所述的方法,其特征在于,所述L个比特集合按照读取顺序依序排列。

3. 根据权利要求1或2所述的方法,其特征在于,所述第i个读取间隔和第(i-1)个读取间隔相同或不同。

4. 一种交织装置,其特征在于,包括:

接收单元,用于获取第一比特序列,所述第一比特序列包括N个比特, $N \geq 2$ 且为整数;

处理单元,用于根据至少一个读取间隔从所述第一比特序列中读取比特,得到L个比特集合,每个比特集合中包括所述第一比特序列中的至少一个比特,任意两个比特集合中包括的比特在所述第一比特序列中的索引不同,所述L个比特集合中比特数目之和等于所述N,所述至少一个读取间隔中的任意一个读取间隔是根据调制阶数或预设函数确定的,其中, $L \geq 2$ 且为整数;

所述处理单元,还用于根据所述L个比特集合,得到第二比特序列;

发送单元,用于发送所述第二比特序列;

所述处理单元具体用于:根据第i个读取间隔,从所述第一比特序列中除去第1个比特集合至第(i-1)个比特集合后的剩余比特中读取至少一个比特,得到第i个比特集合,其中, $1 \leq i \leq L$,且i为整数。

5. 根据权利要求4所述的装置,其特征在于,所述L个比特集合按照读取顺序依序排列。

6. 根据权利要求4或5所述的装置,其特征在于,所述第i个读取间隔和第(i-1)个读取间隔相同或不同。

7. 一种计算机可读存储介质,其特征在于,存储计算机指令,当所述计算机指令在计算机上运行时,使得所述计算机执行权利要求1至3中任一项所述的方法。

8. 一种芯片,其特征在于,包括:

存储器,用于存储计算机程序;

处理器,用于从所述存储器中调用并运行所述计算机程序,使得安装有所述芯片的通信设备执行权利要求1至3中任一项所述的方法。

交织方法和交织装置

技术领域

[0001] 本申请涉及信道编码领域,尤其涉及一种交织方法和交织装置。

背景技术

[0002] 数字通信系统通常采用信道编码提高数据传输的可靠性,其中,一些信道编码采用了交织技术,以进一步提升数据传输过程中的抗干扰性能。在许多同时出现随机错误和突发错误的复合信道上,如果发生一个错误,往往会波及一串数据,导致突发错误超过信道纠错能力,纠错能力下降。而如果首先把突发错误离散成随机错误,再进行随机错误纠错,则系统的抗干扰性能将进一步得到提高。

[0003] 现阶段,根据交织方法的不同,交织方法主要分为随机交织和行列交织。随机交织在离线计算交织序列时,需要存储置换序列供交织和解交织使用,在码长较长的情况下,随机交织所需的存储资源非常大,甚至不可接受。而行列交织对于所及错误的纠错能力较弱,纠错性能较差。

发明内容

[0004] 本申请提供一种交织方法和交织装置,可以在不增加交织复杂度的情况下提升纠错性能。

[0005] 第一方面,本申请提供一种交织方法,该方法包括:获取第一比特序列,第一比特序列包括 N 个比特, $N \geq 2$ 且为整数;根据至少一个读取间隔从第一比特序列中读取比特,得到 L 个比特集合,每个比特集合中包括第一比特序列中的至少一个比特,任意两个比特集合中包括的比特在第一比特序列中的索引不同,该 L 个比特集合中比特数目之和等于 N ,其中, $L \geq 2$ 且为整数;根据该 L 个比特集合,得到第二比特序列;发送第二比特序列。

[0006] 本申请实施例的交织方法,与随机交织相比,交织复杂度低,但是交织性能却与随机交织性能相当甚至更优。因此,在不增加交织复杂度的情况下可以提升纠错性能。

[0007] 结合第一方面,在第一方面的某些实现方式中,该 L 个比特集合按照读取顺序依序排列。

[0008] 结合第一方面,在第一方面的某些实现方式中,该根据至少一个读取间隔从第一比特序列中读取比特,得到 L 个比特集合,包括:根据第 i 个读取间隔,从第一比特序列中除去第1个比特集合至第 $(i-1)$ 个比特集合后的剩余比特中读取至少一个比特,得到第 i 个比特集合,其中, $1 \leq i \leq L$,且 i 为整数。

[0009] 这里需要说明的是,第1个比特集合可以根据确定的读取间隔,直接从第一比特序列中读取。

[0010] 结合第一方面,在第一方面的某些实现方式中,第 i 个读取间隔和第 $(i-1)$ 个读取间隔相同或不同。

[0011] 结合第一方面,在第一方面的某些实现方式中,该至少一个读取间隔中的任意一个读取间隔是根据调制阶数或预设函数确定的。

[0012] 第二方面,提供了一种交织装置,用于执行第一方面或第一方面的任意可能的实现方式中的方法。具体地,该装置包括执行第一方面或第一方面的任意可能的实现方式中的方法的单元。

[0013] 第三方面,本申请提供一种交织设备,该交织设备包括一个或多个处理器,一个或多个存储器,一个或多个收发器(每个收发器包括发射机和接收机)。收发器用于通过天线收发信号。存储器用于存储计算机程序指令(或者说,代码)。处理器用于执行存储器中存储的指令,当指令被执行时,处理器执行第一方面或第一方面的任意可能的实现方式中的方法。

[0014] 第四方面,本申请提供一种计算机可读存储介质,该计算机可读存储介质中存储有指令,当其在计算机上运行时,使得计算机执行上述第一方面或第一方面的任意可能的实现方式中的方法。

[0015] 第五方面,本申请提供一种芯片(或者说,芯片系统),包括存储器和处理器,存储器用于存储计算机程序,处理器用于从存储器中调用并运行该计算机程序,使得安装有该芯片的通信设备执行上述第一方面及其任意一种可能的实现方式中的方法。

[0016] 第六方面,本申请提供一种计算机程序产品,所述计算机程序产品包括:计算机程序代码,当所述计算机程序代码在计算机上运行时,使得计算机执行上述第一方面及其任意一种可能的实现方式中的方法。

[0017] 第七方面,本申请提供一种编码装置,该编码装置具有实现上述第一方面及其第一方面任意一种可能的实现方式中的方法的功能。这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的模块。此外,该编码装置还应具有和编码相关的性能,例如,编码、速率匹配等。

[0018] 在一个可能的设计中,当这些功能的部分或全部通过硬件实现时,编码装置包括:输入接口电路,用于获取第一比特序列;逻辑电路,用于执行上述第一方面及其第一方面的任意一种可能的设计中的交织方法;输出接口电路,用于输出第二比特序列。

[0019] 可选的,编码装置可以是芯片或者集成电路。

[0020] 在一个可能的设计中,当这些功能的部分或全部通过软件实现时,编码装置包括:存储器,用于存储计算机程序;处理器,用于执行所述存储器存储的计算机程序,当所述计算机程序被执行时,编码装置可以实现上述第一方面及其第一方面的任意一种可能的设计中所述的交织方法。

[0021] 在一个可能的设计中,当这些功能的部分或全部通过软件实现时,编码装置包括处理器,用于存储计算机程序的存储器位于编码装置之外,处理器通过电路/电线与存储器连接,用于读取并执行所述存储器中存储的计算机程序。

[0022] 可选的,上述存储器可以是物理上独立的单元,也可以与处理器集成在一起。

[0023] 需要说明的是,本申请实施了中描述的交织方法是由数据和/或信息的发送端来执行的。在数据和/或信息的接收端,需要对接收到的比特序列进行解交织。本领域技术人员公知,解交织是交织的逆过程。在上述第一方面及其任意一种可能的实现方式中描述的交织方法的基础上,本领域技术人员容易得到解交织的方法,本文中不作详述。

[0024] 此外,本申请提供一种解交织的装置,具体地,解交织的装置包括执行解交织的方法的单元。

[0025] 此外,本申请还提供一种解交织的设备,该设备包括一个或多个处理器,一个或多个存储器,一个或多个收发器(收发器包括发射机和接收机)。发射机或接收机通过天线收发信号。存储器用于存储计算机程序指令(或者,代码)。处理器用于执行存储器中存储的指令,当指令被执行时,处理器执行解交织的方法。

[0026] 此外,本申请提供一种计算机可读存储介质,该计算机可读存储介质中存储有计算机指令,当其在计算机上运行时,使得计算机执行解交织的方法。

[0027] 本申请还提供一种计算机程序产品,该计算机程序产品包括:计算机程序代码,当该计算机程序代码在计算机上运行时,使得计算机执行解交织的方法。

[0028] 本申请还提供一种芯片(或者说,芯片系统),包括存储器和处理器,存储器用于存储计算机程序,处理器用于从存储器中调用并运行该计算机程序,使得安装有该芯片的通信设备执行本申请各方法实施例中的交织方法。

[0029] 本申请还提供一种译码装置,该译码装置具有实现本申请实施例中所说的解交织的方法的功能。这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。除此之外,译码装置还具有实现译码的相关功能,例如,解速率匹配、译码等。

[0030] 在本申请实施例中,提出了一种简单易操作的交织方法,能够在不增加交织复杂度的情况下提升纠错性能。

附图说明

[0031] 图1为适用于本申请实施例的无线通信系统100。

[0032] 图2是采用无线技术进行通信的基本流程图。

[0033] 图3是本申请实施例的交织方法的流程图。

[0034] 图4是本申请实施例的从第一比特序列中读取L个比特集合的过程示意图。

[0035] 图5是本申请实施例的交织方法的一个示例。

[0036] 图6是本申请实施例的交织方法的另一个示例。

[0037] 图7是本申请实施例的交织方法的又一个示例。

[0038] 图8为本申请实施例的交织装置500的示意图。

[0039] 图9为本申请实施例的交织设备700的示意性结构图。

[0040] 图10为本申请实施例的终端设备800的示意性结构图。

具体实施方式

[0041] 下面将结合附图,对本申请中的技术方案进行描述。

[0042] 图1为适用于本申请实施例的无线通信系统100。该无线通信系统中可以包括至少一个网络设备101,该网络设备与一个或多个终端设备(例如,图1中所示的终端设备102和终端设备102)进行通信。网络设备101可以是基站,也可以是基站与基站控制器集成后的设备,还可以是具有类似通信功能的其它设备。

[0043] 本申请实施例提及的无线通信系统包括但不限于:窄带物联网系统(Narrow Band-Internet of Things,NB-IoT)、全球移动通信系统(Global System for Mobile Communications,GSM)、增强型数据速率GSM演进系统(Enhanced Data rate for GSM Evolution,EDGE)、宽带码分多址系统(Wideband Code Division Multiple Access,

WCDMA)、码分多址2000系统(Code Division Multiple Access,CDMA2000)、时分同步码分多址系统(Time Division-Synchronization Code Division Multiple Access,TD-SCDMA),长期演进系统(Long Term Evolution,LTE)、下一代5G移动通信系统的三大应用场景eMBB,URLLC和eMTC或者将来出现的新的通信系统。

[0044] 本申请实施例中所涉及到的终端设备可以包括各种具有无线通信功能的手持设备、车载设备、可穿戴设备、计算设备或连接到无线调制解调器的其它处理设备。终端设备可以是移动台(Mobile Station,MS)、用户单元(subscriber unit)、蜂窝电话(cellular phone)、智能电话(smart phone)、无线数据卡、个人数字助理(Personal Digital Assistant,PDA)电脑、平板型电脑、无线调制解调器(modem)、手持设备(handset)、膝上型电脑(laptop computer)、机器类型通信(Machine Type Communication,MTC)终端等。

[0045] 图1中的网络设备101与终端设备之间采用无线技术进行通信。当网络设备发送信号时,其为发送端,终端设备为接收端。当网络设备接收信号时,其为接收端,终端设备为发送端。

[0046] 图2是采用无线技术进行通信的基本流程图。发送端的信源依次经过信源编码、信道编码、速率匹配和调制后在信道上发出。接收端接收到信号后依次经过解调、解速率匹配、信道解码和信源解码后获得信宿。

[0047] 信道编码是无线通信系统的核心技术之一,其性能的改进将直接提升网络覆盖及用户传输速率。为了提高信号的抗干扰性,可以进一步地可以引入交织技术。交织技术的思想是在时间上分离码元,将一个有记忆信道转变为无记忆信道,从而使得纠随机错误的编码也能适用于噪声突发信道。

[0048] 常用的交织方法包括随机交织和行列交织。随机交织在平均性能上较优,但是由于交织的随机性,无法保证每次交织都具有较优的性能。并且在离线交织的情况下,需要存储大量的置换序列供交织和解交织使用。当码长较长时,随机交织所需的存储资源较大,给编码器造成很大的硬件负荷,甚至不可接受。此外,随机交织的复杂度较高。而行列交织的方案比较简单,但是对于数据的随机化处理较弱,交织性能不太理想。

[0049] 为此,本申请提出一种交织方法,可以在不增加交织复杂度的情况下,提升纠错性能。下面对本申请实施例的交织方法进行详细说明。

[0050] 参见图3,图3是本申请实施例的交织方法的流程图。

[0051] 310、获取第一比特序列。

[0052] 其中,第一比特序列包括N个比特,其中, $N \geq 2$ 且为整数。

[0053] 320、根据至少一个读取间隔,从第一比特序列中读取比特,得到L个比特集合。

[0054] 在本申请实施例中,L个比特集合是从第一比特序列中读取比特得到的。L个比特集合中比特的数目之和等于N。并且,任意两个比特集合中包括的比特在第一比特序列中的索引不同。换句话说,将第一比特序列中包括的N个比特分别放置在L个比特集合中。

[0055] 330、根据该L个比特集合,得到第二比特序列。

[0056] 得到L个比特集合之后,根据该L个比特集合可以得到第二比特序列,从而完成交织。

[0057] 在本申请实施例中,第一比特序列是待交织的比特序列,交织后的比特序列为第二比特序列。

[0058] 具体地,得到L个比特集合之后,可以有多种方式生成第二比特序列。

[0059] 例如,将L个比特集合进行组合排序,得到第二比特序列。

[0060] 又例如,按照从第一比特序列中读取得到该L个比特集合的顺序,依序将该L个比特集合排列,得到第二比特序列。

[0061] 340、发送第二比特序列。

[0062] 对第一比特序列完成交织,得到第二比特序列后,对第二比特序列进行映射、调制、发送。这些过程可以与现有技术相同,这里不作详述。

[0063] 下面结合图4,对本申请实施例中从第一比特序列中读取得到L个比特集合的过程作详细说明。

[0064] 参见图4,图4是本申请实施例的从第一比特序列中读取L个比特集合的过程示意图。交织过程可以描述为:首先获取第一比特序列。按照第1个读取间隔从第一比特序列中读取比特,得到第1个比特集合(以下记作比特集合1)。按照第2个读取间隔,从第一比特序列中除去第一比特集合中的比特之外的剩余比特中读取比特,得到第2个比特集合(以下记作比特集合2)。按照第3个读取间隔,从第一比特序列中除去第一比特集合和第二比特集合中的比特之外的剩余比特中读取比特,得到第3个比特集合(以下记作比特集合3)。以此类推,按照第i个读取间隔,从第一比特序列中除去第1个比特集合至第(i-1)个比特集合中的比特之外的剩余比特中读取比特,得到第i个比特集合。以此类推,直至将第一比特序列中的比特全部读出。

[0065] 再根据比特集合1,比特集合2,⋯,比特集合L,可以得到第二比特序列。

[0066] 应理解,i=1时可以看作一个特殊情况,即第1个比特集合可以按照第1个读取间隔,直接从第一比特序列中读出。i≥2时,第i个比特集合是从第一比特序列中除去第1个比特集合至第(i-1)个比特集合中的比特之外的剩余比特中读出的。

[0067] 在本申请实施例中,根据L个比特集合得到第二比特序列,可以有多种方式。

[0068] 为了便于说明,按照从第一比特序列中读出这L个比特集合的先后顺序,将这L个比特集合分别记作比特集合1,比特集合2,⋯,比特集合L。

[0069] 例如,按照先后读出L个比特集合的顺序,依序将这L个比特集合顺序排列,得到第二比特序列。

[0070] 又例如,将先后读出的L个比特集合的顺序打乱,重新排序组成第二比特序列。

[0071] 在本申请实施例中,第i个读取间隔与第i个比特集合对应,其中,第i个读取间隔是从第一比特序列中读取比特组成第i个比特集合时采用的读取间隔。

[0072] 需要注意的是,第i个读取间隔是指从第一比特序列中除去第1个比特集合至第(i-1)个比特集合的剩余比特组成的比特序列中的间隔。

[0073] 例如,第一比特序列={1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,31}。

[0074] 若按照第1个读取间隔等于5,则从第一比特序列中读取得到的比特集合1={1,6,11,16}。

[0075] 除去比特集合1之后,按照第2个读取间隔等于5,从剩余比特组成的比特序列中读取比特,得到比特集合2={2,8,14}。

[0076] 在本申请实施例中,第i个读取间隔和第(i-1)读取间隔可以相同或者不同,i≥1

且为整数。

[0077] 应理解,由第一比特序列得到第二比特序列,即是交织的过程。交织前后,比特序列中包括的比特数量是不会发生变化的。也就是说,如果第一比特序列包括N个比特,那么交织后得到的第二比特序列也包括N个比特, $N \geq 1$ 且为整数。

[0078] 此外,第二比特序列中的N个比特在第二比特序列中的排序与该N个比特在第一比特序列中的排序不同。

[0079] 下面对确定读取间隔的方法进行说明。具体地,读取间隔的可以有多种确定方式。

[0080] 方式1

[0081] 读取间隔可以根据调制阶数确定。

[0082] 针对系统采用的调制阶数(以下将调制阶数的取值记作M),读取间隔可以设置不同的取值。换句话说,读取间隔是调制阶数M的函数。例如,读取间隔等于调制阶数+1。

[0083] 以第一比特序列的长度等于31,调制阶数为4作为示例进行举例说明。

[0084] 参见图5,图5是本申请实施例的交织方法的一个示例。假定系统采用的调制阶数为4,第一比特序列= $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31\}$,即第一比特序列的长度等于31。

[0085] 假定,读取间隔为调制阶数+1,则读取间隔= $4+1=5$ 。

[0086] 从第一比特序列中的第1个比特开始,从左至右,每间隔5个比特读取一个,读出的比特组成比特集合1。如图5,比特集合1= $\{1, 6, 11, 16, 21, 26, 31\}$ 。

[0087] 从第一比特序列中除去比特集合1之外的剩余比特中,从左至右,每间隔5个比特读取一个,读出的比特组成比特集合2。如图5,比特集合2= $\{2, 8, 14, 20, 27\}$ 。

[0088] 依次类推,按照读取间隔等于5,继续读取第一比特序列中的剩余比特,直至第一比特序列中的剩余比特不够通过读取间隔读出,此时,将剩余比特全部读出作为一个比特集合(记作比特集合L)。

[0089] 比特集合1、比特集合2, ..., 比特集合L中的比特顺序组成的序列则为交织后的比特序列,即第二比特序列。

[0090] 因此,交织后的比特序列= $\{1, 6, 11, 16, 21, 26, 31, 2, 8, 14, 20, 27, 3, 10, 18, 25, 4, 13, 23, 5, 17, 29, 7, 22, 9, 28, 12, 15, 19, 24, 30\}$ 。

[0091] 在图5所示的示例中,所有的读取间隔都相等。

[0092] 再以第一比特序列的长度等于31,调制阶数为6作为示例作举例说明。

[0093] 参见图6,图6本申请实施例的交织方法的另一个示例。如图6所示,系统采用的调制阶数为6,第一比特序列= $\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31\}$,即第一比特序列的长度等于31。

[0094] 继续以读取间隔等于调制阶数+1为例,则读取间隔= $6+1=7$ 。

[0095] 从第一比特序列中的第一个比特开始,每间隔7个比特读取一个,从左至右,读出的比特组成比特集合1。类似地,从第一比特序列中剩余的比特中,每间隔7个比特读取一个,从左至右,读出的比特组成比特集合2。

[0096] 依次类推,直至第一比特序列中的比特的个数不够通过读取间隔来读取。此时,将第一比特序列中的剩余比特一次全部读出,组成比特集合L。

[0097] 将比特集合1、比特集合2, ..., 和比特集合L中的比特顺序排列,得到交织后的比特

序列(即第二比特序列)。

[0098] 因此,交织后的比特序列 = {1, 8, 15, 22, 29, 2, 10, 18, 26, 3, 12, 21, 31, 4, 18, 25, 4, 14, 25, 5, 17, 30, 6, 20, 7, 24, 9, 28, 11, 13, 16, 19, 23, 27}。

[0099] 在方式1中,通过调制阶数确定一个读取间隔之后,在每一轮读取时,都采用这个读取间隔。或者说,第*i*个读取间隔和第(*i*-1)个读取间隔是相等的, $i \geq 2$ 。

[0100] 方式2

[0101] 读取间隔可以根据预设函数确定。

[0102] 在方式2中,假定待交织的比特序列的长度等于*N*。

[0103] 将预设函数记作*f*(*M*),*f*(*M*)产生1至*M*之间的 M_1 个数值,将这 M_1 个数值对应 M_1 个位置索引。从待交织的比特序列中,读出这 M_1 个位置索引所指示的位置的比特后,剩余比特再按照*f*(*N*-*M*)产生 M_2 个位置索引,再读取这 M_2 个位置索引所指示的位置的比特。剩余比特以此类推,直至将待交织的比特序列中的全部比特读出。

[0104] 预设函数可以有多种形式。

[0105] 例如,预设函数 $f = \text{mod}(i, 5)$, $i \in \{1, 2, \dots, M\}$ 。

[0106] 在方式2中,与方式1不同的是,读取间隔是根据预设函数确定的。并且,随着第一比特序列中剩余比特的数量的不断减少,每次计算得到的读取间隔可能是不相等的。当然,也有可能某几次计算得到的读取间隔是相等的。

[0107] 又例如,预设函数是 $\text{Ceil}(N/4)$ 。其中,*N*是待读取序列的长度。函数 $\text{Ceil}()$ 的功能是返回大于或者等于指定表达式的最小整数。

[0108] 交织的过程可以使用比特逆序函数。以一个长度*N*=16的比特序列为例,若对该比特序列执行比特逆序操作后,得到的比特序列为{1, 9, 5, 13, 3, 11, 7, 15, 2, 10, 6, 14, 4, 12, 8, 16},读取位置索引为[1, 9, 5, 13]的比特。剩余的12个比特继续执行比特逆序操作,得到序列={1 9 5 13 3 11 7 15 2 10 6 14},读取位置索引为[1, 9, 5]的比特。剩余9个比特继续执行比特逆序操作,得到序列={1, 9, 5, 13, 3, 11, 7, 15, 2},读取位置索引为[1 9 5]的比特。剩余6个比特继续执行比特逆序操作,得到的序列为={1, 5, 3, 7, 2, 6},读取位置索引为[15]的比特。最后剩余的4个比特一次读出。

[0109] 在读取的过程中,若读取的位置索引大于当前待读取序列的长度,则跳过当前的位置索引,并读取后面一个位置索引对应的比特。

[0110] 或者,将比特逆序操作获得的索引序列按照从大到小或从小到大重新排序,然后再读取索引序列中对应位置的比特。

[0111] 例如,若当前待读取序列执行比特逆序操作后的索引构成的序列为{1, 5, 3, 7, 2, 6}(为了便于区分,记作索引序列#1)。可以看到,索引序列#1中的索引“7”大于当前待读取序列的长度6。此时,将索引序列#1中的索引值按照从小到大重新进行排序,得到新的索引序列{1, 2, 3, 5, 6, 7}(记作索引序列#2)。再从当前待读取序列中读取在索引序列#2中的位置为[1, 5, 3, 2, 6, 4]的比特。

[0112] 方式3

[0113] 相邻比特集合的读取间隔可以相同或不同,每个比特集合中的比特是按照相同的读取间隔读取得到的。

[0114] 参见图7,图7是本申请实施例的交织方法的又一个示例。

[0115] 如图7所示,第一比特序列长度等于18,第一个读取间隔等于5,第二个读取间隔等于5。第三个读取间隔和第四个读取间隔等于3,第五个读取间隔等于2。直至第一比特序列中剩余比特的数目不够通过读取间隔来读取。最后,得到的交织后的比特序列 = {1,6,11,16,2,8,14,3,7,12,17,4,10,18,5,13,9,15}。

[0116] 方式4

[0117] 读取间隔可以根据当前待读取序列的长度 N 确定。

[0118] 在方式4中,给定待交织的第一比特序列之后,随着从第一比特序列中不断读出比特,第一比特序列中的剩余比特越来越少。这里,当前待读取序列的长度 N 是指每次读取时第一比特序列中剩余比特的个数。

[0119] 例如,若第一比特序列中包括32个比特,那么第一次读取时,待读取序列的长度等于32。若第一次读出8个比特,则第一比特序列中剩余(32-8)个比特,第二次读取时,待读取序列的长度等于24。

[0120] 作为一个实施例,当 $N < N_1$ 时,读取间隔等于 M_1 。当 $N < N_2$ 时,读取间隔等于 M_2 。当 $N < N_3$ 时,读取间隔等于 M_3 ,以此类推。 N_i 的取值可以等于母码长度或是预先设定的某个长度, $i \geq 1$ 且为整数。例如, N_i 的取值可以为1024,2048,1000,2000,500等。

[0121] 例如,若预先设定 $N_1 = 1024$, $N_2 = 512$,如果待交织的比特序列的长度 $N = 800 < 1024$,则读取间隔等于 M_1 , M_1 可以为一个设定值。如果待交织的比特序列的长度 N 等于 $500 < 512$,则读取间隔等于 M_2 。其中, M_1 和 M_2 可以为预设值, M_1 和 M_2 不相等。

[0122] 方式5

[0123] 读取间隔可以根据当前待读取序列的长度 N 和调制阶数确定。

[0124] 在方式5中,首次读取时采用的读取间隔(以下称作初始读取间隔)可以根据自变量为调制阶数的预设函数确定。后续读取时采用的读取间隔可以在初始读取间隔的基础上递增或者递减。递增或者递减的速度与幅度可以和待读取的序列长度 N 相关。

[0125] 例如,第一比特序列长度 $N = 1000$,调制阶数等于4,首次读取时的读取间隔等于5。后续读取时,若当前待读取的序列长度大于800,读取间隔取6。若当前待读取的序列长度大于600且小于800,读取间隔取7,以此类推。

[0126] 方式6

[0127] 读取间隔可以根据当前待读取序列的长度 K_i 对应的母码长度 Q 确定, $i \geq 1$ 且为整数。

[0128] 在方式6中,当 $Q < N_1$ 时,读取间隔等于 M_1 。当 $Q < N_2$ 时,读取间隔等于 M_2 。当 $Q < N_3$ 时,读取间隔等于 M_3 。以此类推, N_i 的取值可以是母码长度或预先设定的长度,例如, N_i 的取值可以为1024,2048,1000,2000,500等。

[0129] 例如,若预先设定 $N_1 = 128$, $N_2 = 64$,如果待交织的比特序列的长度等于50,对于第一次读取来说,待读取序列的长度 $K_1 = 50$,而长度等于50的比特序列对应的母码长度 $Q = 64 < 128$,则第一次读取时的读取间隔可以取 M_1 。若第一次读出20个比特,对于第二次读取来说,当前待读取序列的长度变为(50-20),即 $K_2 = 30$,而长度等于30的比特序列对应的母码长度 $Q = 32 < 64$,则第二次读取时的读取间隔可以取 M_2 。其中, M_1 和 M_2 可以为预设值, M_1 和 M_2 不相等。

[0130] 本申请实施例的交织方法,能够在不增加交织复杂度的情况下提升纠错性能。

[0131] 以上结合图1至图7,对本申请实施例的交织方法的过程作了详细说明,以下对本申请实施例的交织装置作介绍。

[0132] 图8为本申请实施例的交织装置500的示意图。如图8所示,装置500包括接收单元510、处理单元520和发送单元530。其中,

[0133] 接收单元510,用于获取第一比特序列,第一比特序列包括N个比特, $N \geq 2$ 且为整数;

[0134] 处理单元520,用于根据至少一个读取间隔从第一比特序列中读取比特,得到L个比特集合,每个比特集合中包括第一比特序列中的至少一个比特,任意两个比特集合中包括的比特在第一比特序列中的索引不同,该L个比特集合中比特数目之和等于N,其中, $N \geq 2$, $L \geq 2$,且N和L为整数;

[0135] 发送单元530,用于发送第二比特序列。

[0136] 本申请实施例的装置500中的各单元和上述其它操作或功能分别为了实现本申请各实施例中的交织方法。为了简洁,此处不再赘述。

[0137] 本申请实施例的交织装置,能够在不增加交织复杂度的情况下提升纠错性能。

[0138] 图9为本申请实施例的交织设备700的示意性结构图。如图9所示,设备700包括:一个或多个处理器701,一个或多个存储器702,一个或多个收发器703。处理器701用于控制收发器703收发信号,存储器702用于存储计算机程序,处理器701用于从存储器702中调用并运行该计算机程序,使得交织设备700执行交织方法各实施例的相应流程和/或操作。为了简洁,此处不再赘述。

[0139] 需要说明的是,图8中所示的交织装置500可以通过图9中所示的交织设备700实现。例如,接收单元510、发送单元530可以由图9中的收发器703实现。处理单元520可以由处理器701实现等。

[0140] 交织设备可以为图1中所示的网络设备或终端设备。在上行传输时,交织设备具体为终端设备,终端设备具有实现上述各实施例中描述的交织方法的功能。这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的单元。在下行传输时,交织设备具体为网络设备(例如,基站),网络设备具有实现上述各实施例中描述的交织方法的功能。同样地,这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的单元。

[0141] 当交织设备700具体为终端设备时,终端设备的结构可以如图10所示。图10为本申请实施例的终端设备800的示意性结构图。

[0142] 如图10所示,终端设备800包括:收发器808和处理装置804。终端设备800还可以包括存储器819,存储器819用于存储计算机指令。

[0143] 收发器808,用于获取第一比特序列,第一比特序列包括N个比特, $N \geq 2$ 且为整数;

[0144] 处理器804,用于根据至少一个读取间隔从第一比特序列中读取比特,得到L个比特集合,每个比特集合中包括第一比特序列中的至少一个比特,任意两个比特集合中包括的比特在第一比特序列中的索引不同,该L个比特集合中比特数目之和等于N,其中, $N \geq 2$, $L \geq 2$,且N和L为整数;

[0145] 收发器808,用于根据处理装置804的指示,发送第二比特序列。

[0146] 进一步地,上述处理装置804可以用于执行前面方法实施例中描述的由交织设备内部实现的动作,而收发器808可以用于执行前面方法实施例中描述的交织设备的接收或发送动作。具体请见前面方法实施例中的描述,此处不再赘述。

[0147] 上述处理装置804和存储器819可以集成为一个处理器,处理器用于执行存储器819中存储的程序代码来实现上述功能。具体实现时,该存储器819也可以集成在处理器中。

[0148] 上述终端设备800还可以包括电源812,用于给终端设备800中的各种器件或电路提供电源。上述终端设备800可以包括天线810,用于将收发器808输出的数据或信息通过无线信号发送出去。

[0149] 除此之外,为了使终端设备800的功能更加完善,终端设备800还可以包括输入单元814,显示单元816,音频电路818,摄像头820和传感器822等中的一个或多个。音频电路还可以包括扬声器8182,麦克风8184等。

[0150] 需要说明的是,本申请实施例中提供的交织方法可以适用于各种信道编码,例如,LDPC码、Turbo码、极化(Polar)码等。本申请实施例对此不作限定。

[0151] 此外,本申请提供的交织方法可以作为一个单独的交织模块,用于实现交织处理。也可以作为速率匹配时读取比特的方式,这样就可以将交织和速率匹配集成在一起实现,不需要单独设计交织模块,同样也可以达到与随机交织相同的纠错性能。

[0152] 另外,本申请实施例的交织方法,对于符号(symbol)序列的交织也是适用的,本领域技术人员根据上面描述的对比特序列进行交织的方法,也可以将其应用于符号序列的交织,本文中不再详述。

[0153] 此外,本申请提供一种计算机可读存储介质,该计算机可读存储介质中存储有指令,当其在计算机上运行时,使得计算机执行上述各实施例中的交织方法。

[0154] 本申请还提供一种计算机程序产品,该计算机程序产品包括:计算机程序代码,当该计算机程序代码在计算机上运行时,使得计算机执行上述实施例中描述的交织方法。

[0155] 本申请还提供一种芯片,包括存储器和处理器,存储器用于存储计算机程序,处理器用于从存储器中调用并运行该计算机程序,使得安装有该芯片的通信设备执行上述实施例中描述的交织方法。

[0156] 其中,这里所说的通信设备可以为网络设备或终端设备。

[0157] 本申请还提供一种编码装置,该编码装置具有实现上述实施例中描述的交织方法的功能。这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的模块。除此之外,编码装置还具有实现编码的相关功能。编码装置对待编码序列进行编码后,采用本申请实施例的交织方法,对编码后的序列进行交织。或者,该编码装置也可以将本申请实施例的交织方法应用在速率匹配,这样可以省掉交织模块,但是同样会起到提高纠错性能的作用。

[0158] 在一个可能的设计中,当这些功能的部分或全部通过硬件实现时,编码装置包括:

[0159] 输入接口电路,用于获取第一比特序列;

[0160] 逻辑电路,用于执行上述实施例中描述的交织方法,对第一比特序列进行交织,得到第二比特序列;

[0161] 输出接口电路,用于输出第二比特序列。

[0162] 可选的,编码装置可以是芯片或者集成电路。

[0163] 在一个可能的设计中,当这些功能的部分或全部通过软件实现时,编码装置包括:存储器,用于存储计算机程序;处理器,用于执行存储器存储的计算机程序,当所述计算机程序被执行时,编码装置可以实现上述实施例中任意一种可能的设计中所述的交织方法。

[0164] 在一个可能的设计中,当这些功能的部分或全部通过软件实现时,编码装置包括处理器。用于存储计算机程序的存储器位于编码装置之外,处理器通过电路/电线与存储器连接,用于读取并执行存储器中存储的计算机程序。

[0165] 需要说明的是,本申请实施了中描述的交织方法是由数据和/或信息的发送端来执行的。在数据和/或信息的接收端,需要对接收到的比特序列进行解交织。本领域技术人员公知,解交织是交织的逆过程。在上述第一方面及其任意一种可能的实现方式中描述的交织方法的基础上,本领域技术人员容易得到解交织的方法,本文中不作详述。

[0166] 相对应地,本申请提供一种解交织的装置,用于实现解交织的方法中的相应功能。这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。

[0167] 此外,本申请提供一种计算机可读存储介质,该计算机可读存储介质中存储有计算机指令,当其在计算机上运行时,使得计算机执行解交织的方法。

[0168] 本申请还提供一种计算机程序产品,该计算机程序产品包括:计算机程序代码,当该计算机程序代码在计算机上运行时,使得计算机执行解交织的方法。

[0169] 本申请还提供一种芯片(或者说,芯片系统),包括存储器和处理器,存储器用于存储计算机程序,处理器用于从存储器中调用并运行该计算机程序,使得安装有该芯片的通信设备执行本申请各方法实施例中的交织方法。

[0170] 本申请提供一种解交织的设备,该设备包括一个或多个处理器,一个或多个存储器,一个或多个收发器(每个收发器包括发射机和接收机)。发射机或接收机通过天线收发信号。存储器用于存储计算机程序指令(或者,代码)。处理器用于执行存储器中存储的指令,当指令被执行时,处理器执行解交织的方法。

[0171] 本申请还提供一种译码装置,该译码装置具有实现本申请实施例中所说的解交织的方法的功能。这些功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。除此之外,译码装置还具有实现译码的相关功能,例如,解速率匹配、译码等。

[0172] 可选的,以上实施例中所述的存储器与存储器可以是物理上相互独立的单元,或者,存储器也可以和处理器集成在一起。

[0173] 以上实施例中,处理器可以为中央处理器(Central Processing Unit,CPU)、微处理器、特定应用集成电路(Application-Specific Integrated Circuit,ASIC),或一个或多个用于控制本申请方案程序执行的集成电路等。例如,处理器可以包括数字信号处理设备、微处理器设备、模数转换器、数模转换器等。处理器可以根据这些设备各自的功能而在这些设备之间分配移动设备的控制和信号处理的功能。此外,处理器可以包括操作一个或多个软件程序的功能,软件程序可以存储在存储器中。

[0174] 处理器的所述功能可以通过硬件实现,也可以通过硬件执行相应的软件实现。所述硬件或软件包括一个或多个与上述功能相对应的模块。

[0175] 存储器可以是只读存储器(Read-Only Memory,ROM)或可存储静态信息和指令的其他类型的静态存储设备,随机存取存储器(Random Access Memory,RAM)或者可存储信息和指令的其他类型的动态存储设备。也可以是电可擦可编程只读存储器(Electrically

Erasable Programmable Read-Only Memory,EEPROM)、只读光盘(Compact Disc Read-Only Memory,CD-ROM)或其他光盘存储、光碟存储(包括压缩光碟、激光碟、光碟、数字通用光碟、蓝光光碟等)、磁盘存储介质或者其他磁存储设备、或者能够用于携带或存储具有指令或数据结构形式的期望的程序代码并能够由计算机存取的任何其他介质,但不限于此。

[0176] 结合前面的描述,本领域的技术人员可以意识到,本文实施例的方法,可以通过硬件(例如,逻辑电路),或者软件,或者硬件与软件的结合来实现。这些方法究竟以硬件还是软件方式来执行,取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能,但是这种实现不应认为超出本申请的范围。

[0177] 当上述功能通过软件的形式实现并作为独立的产品销售或使用,可以存储在一个计算机可读取存储介质中。在这种情况下,本申请的技术方案本质上或者说对现有技术做出贡献的部分或者该技术方案的部分可以以软件产品的形式体现出来,该计算机软件产品存储在一个存储介质中,包括若干指令用以使得一台计算机设备(可以是个人计算机,服务器,或者网络设备)执行本申请各个实施例所述方法的全部或部分步骤。而前述的存储介质包括:U盘、移动硬盘、只读存储器(Read-Only Memory,ROM)、随机存取存储器(Random Access Memory,RAM)、磁碟或者光盘等各种可以存储程序代码的介质。

[0178] 以上所述,仅为本申请的具体实施方式,但本申请的保护范围并不局限于此,任何熟悉本技术领域的技术人员在本申请揭露的技术范围内,可轻易想到变化或替换,都应涵盖在本申请的保护范围之内。因此,本申请的保护范围应以所述权利要求的保护范围为准。

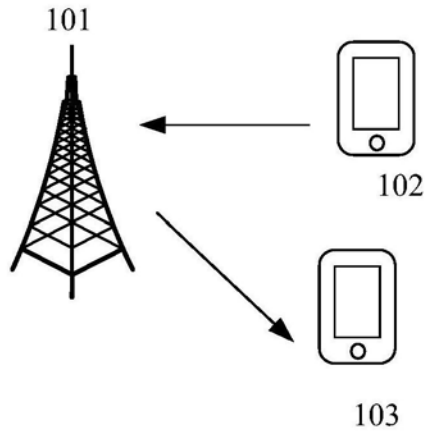


图1

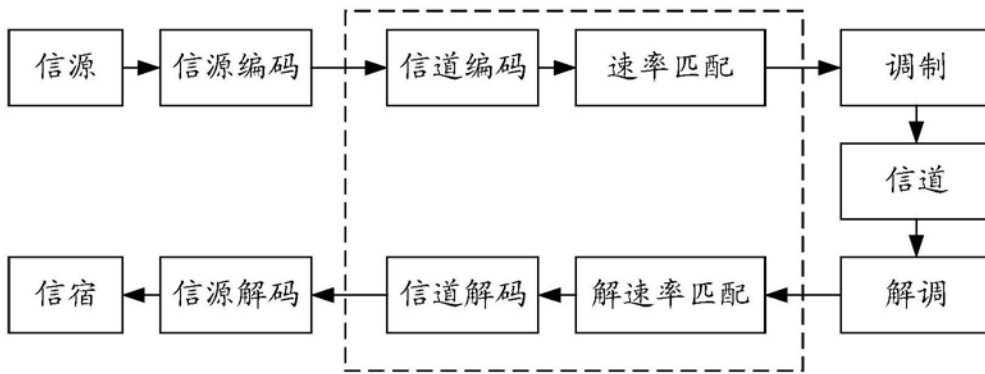


图2

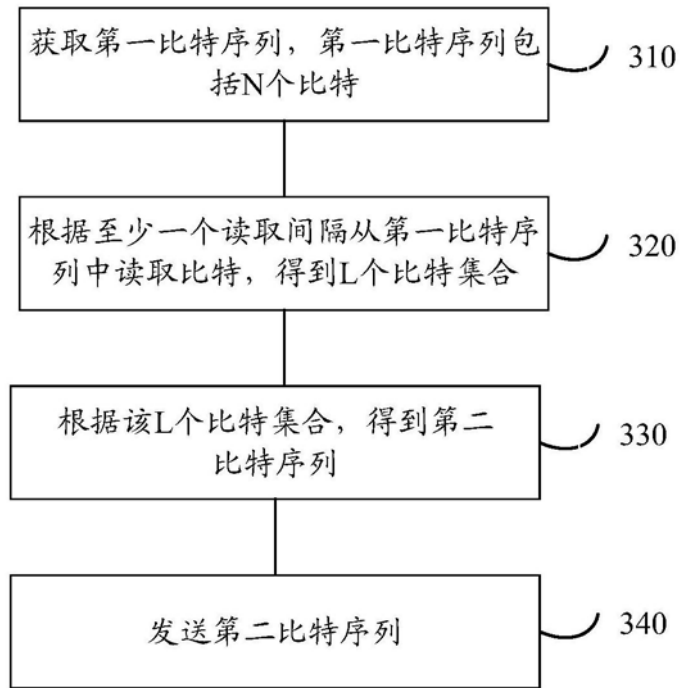


图3

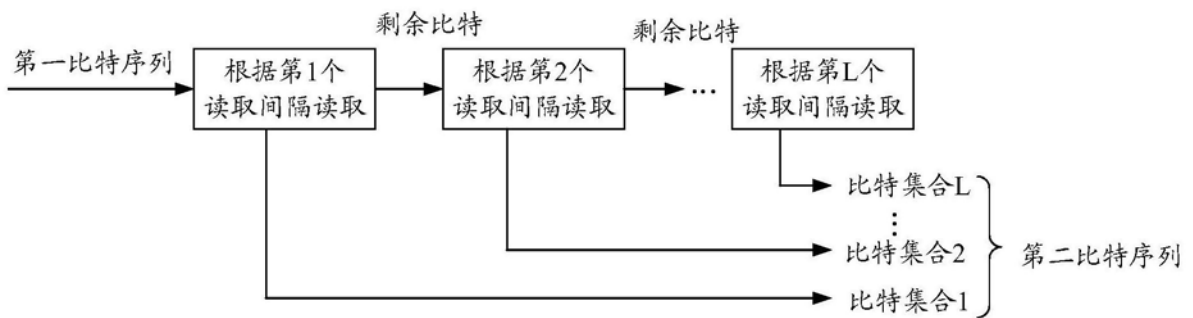


图4

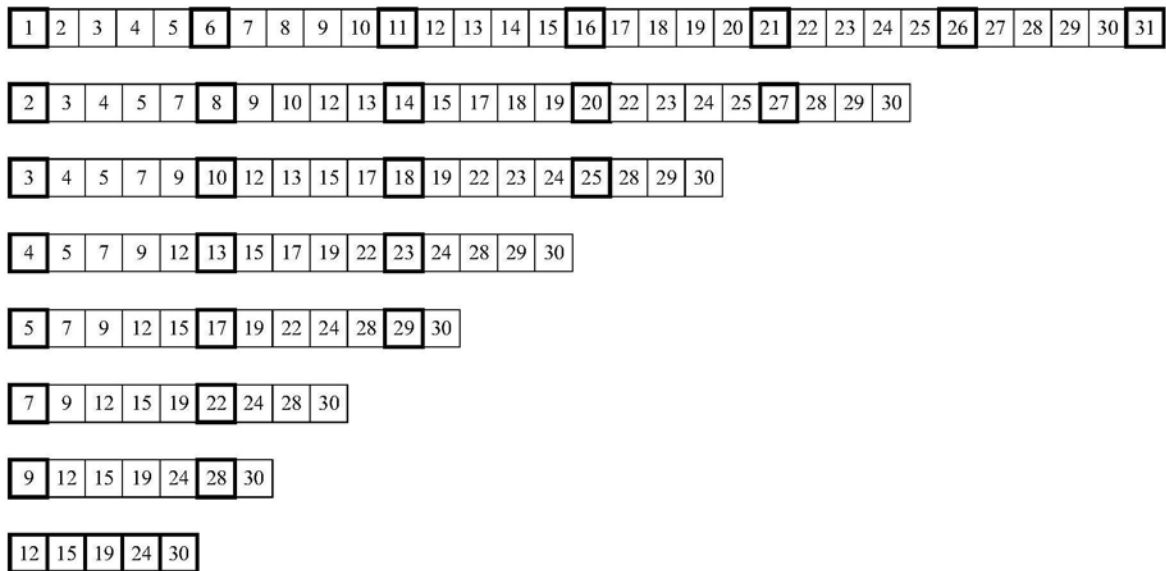


图5

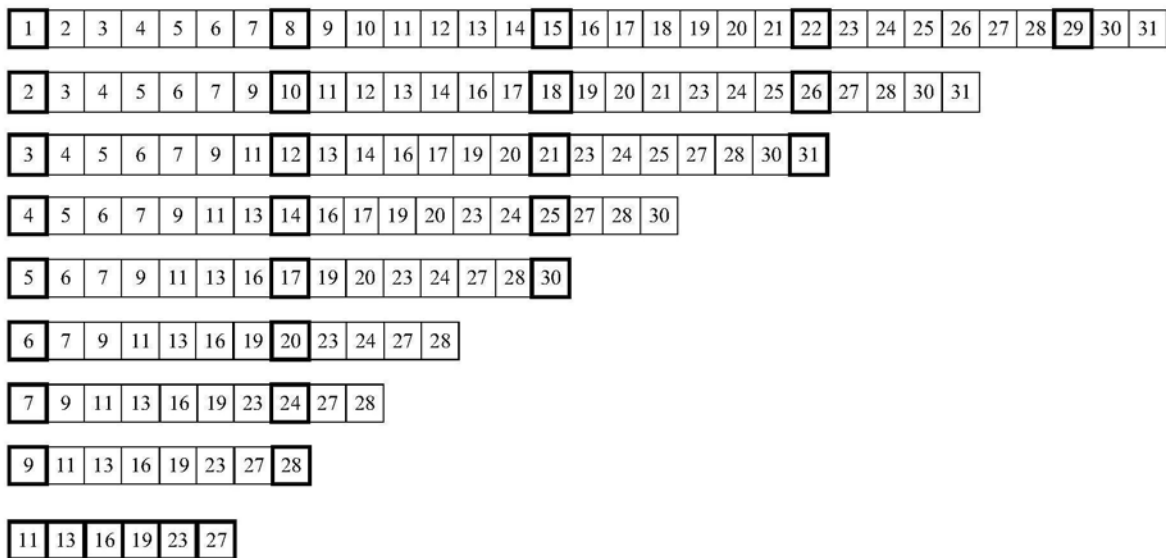


图6

读取间隔

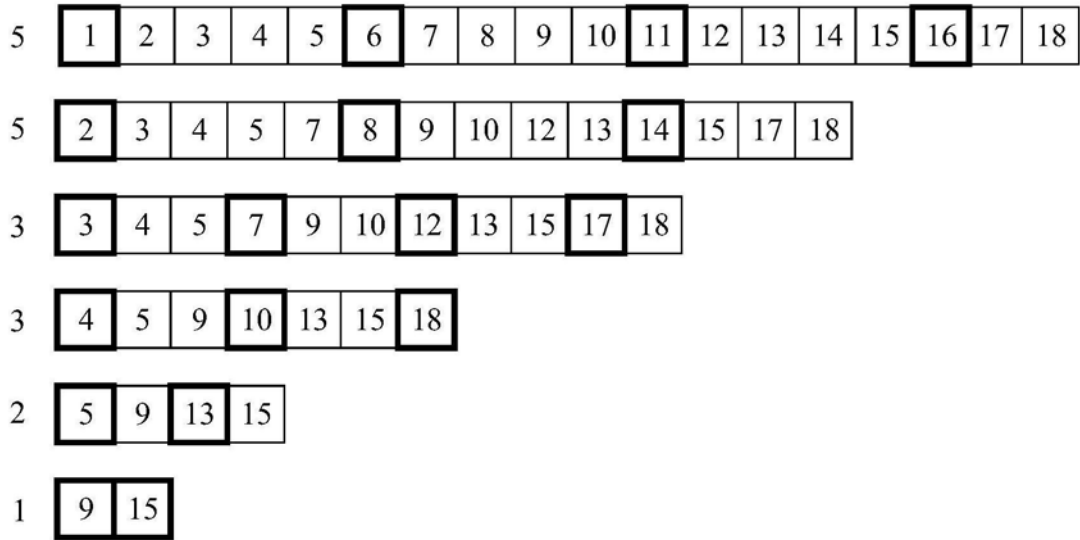


图7

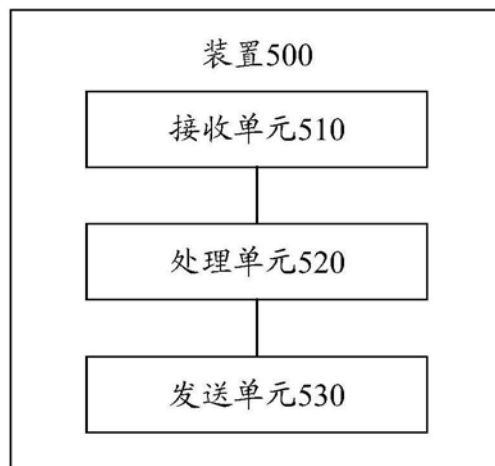


图8

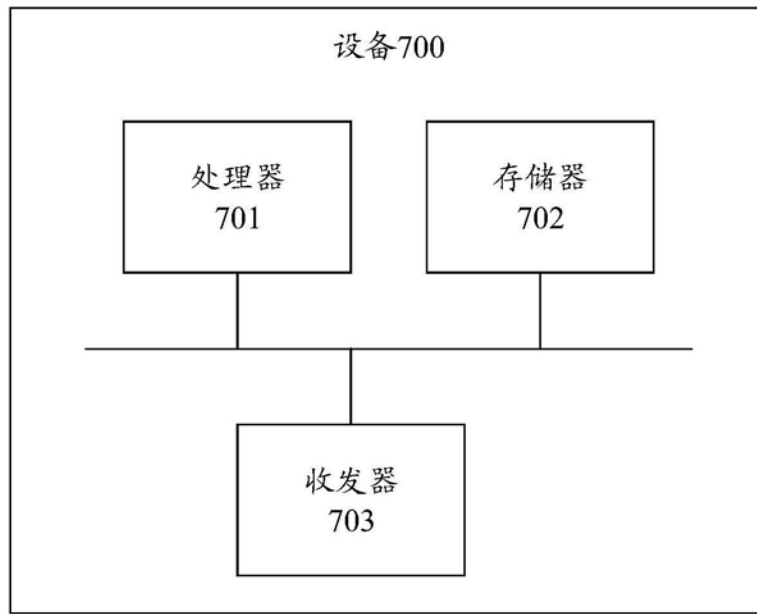


图9

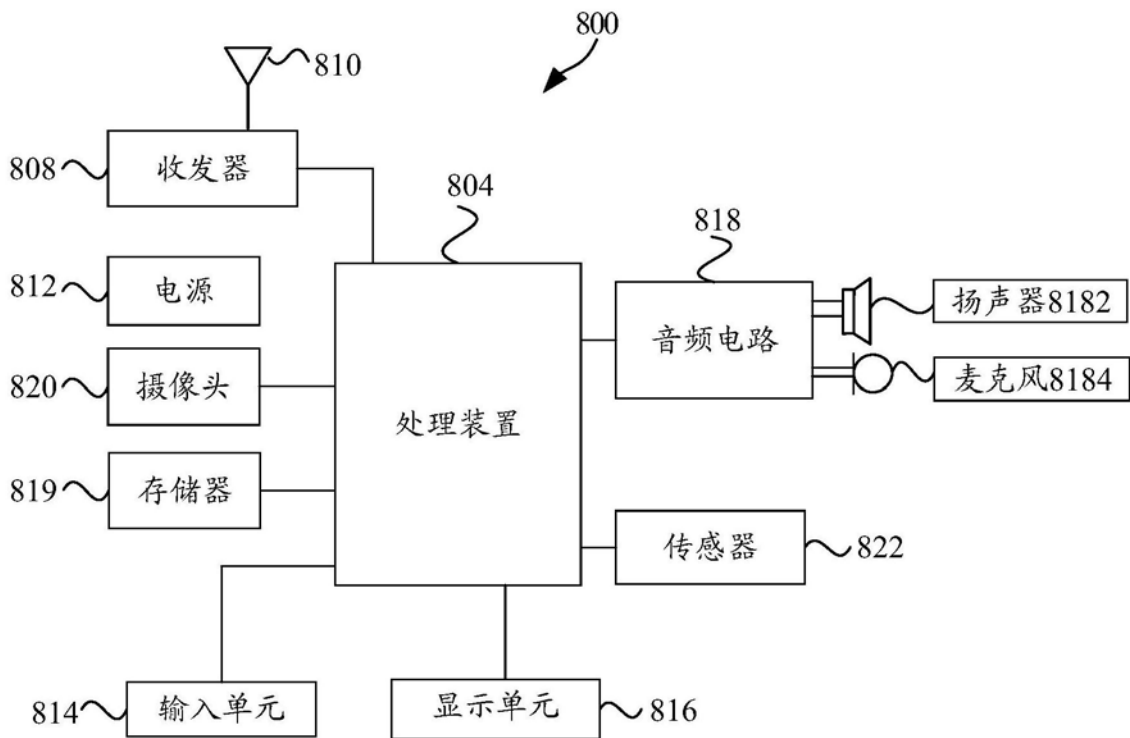


图10