

(12) 发明专利申请

(10) 申请公布号 CN 103166735 A

(43) 申请公布日 2013. 06. 19

(21) 申请号 201110424371. 9

(22) 申请日 2011. 12. 15

(71) 申请人 无锡中星微电子有限公司

地址 214135 江苏省无锡市新区太湖国际科技园清源路 530 大厦 A 区 10 层

(72) 发明人 黄晓

(51) Int. Cl.

H04L 1/00 (2006. 01)

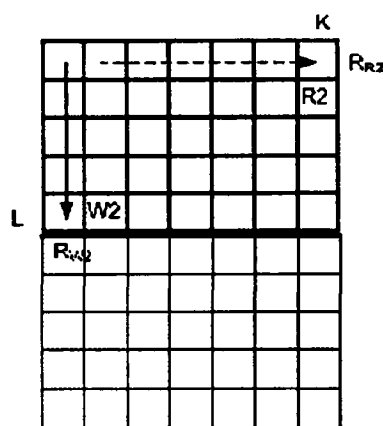
权利要求书1页 说明书5页 附图1页

(54) 发明名称

一种交织器

(57) 摘要

本发明公开了一种交织器,预先为奇数码元和偶数码元分别设定各自的读速率和写速率,根据不同的内存单元,调整读速率和写速率,从而可实现通过设定不同的时钟来驱动码元数据的读写,使码元数据的读写速率更加灵活,可到达减小交织器传输延迟的目的。



1. 一种交织器,包括以下步骤:

预先设定第奇数个码元的数据按写速率 R_{w1} 写入行按读速率 R_{r1} 从列读出,第偶数个码元的数据按写速率 R_{w2} 写入行按读速率 R_{r2} 从列读出;同时为每个码元的数据开辟的内存单元大小为 $\text{Max}(K, L)2 - (K-L)2$,其中 K 为行数, L 为列数,且 K 、 L 均为自然数;

当所采用的内存单元为单端口内存时,每个大小为 $K*L$ 的码元中的数据按下面的方式读写到该内存单元,其中 K 为行数, L 为列数,且 K 、 L 均为自然数:

当第一个奇数码元中的 $K*L$ 个数据以速率 R_{w1} 按逐行写入所述内存单元后,该奇数码元中的数据以速率 R_{r1} 按逐列读出;待第一列数据被读出后,下一个偶数码元的数据以速率 R_{w2} 按逐列写入内存单元,并将前面的数据覆盖;当前一个奇数码元中的数据都读出后,该偶数码元中 $K*L$ 个数据也都写入到内存单元;随后,该偶数码元中的数据以速率 R_{r2} 按逐行读出;待第一行数据被读出之后,下一个奇数码元的数据以速率 R_{w1} 按逐行写入到内存单元中去,并将前面的数据覆盖;如此反复,直到数据被全部从该内存单元读出。

2. 如权利要求 1 所述交织器,其特征在于:所述写速率和读速率相等。

3. 如权利要求 1 所述交织器,其特征在于:第奇数个码元的数据的写速率与第偶数个码元的数据的写速率相等,第奇数个码元的数据的读速率与第偶数个码元的数据的读速率相等。

4. 一种交织器,包括以下步骤:

预先设定第奇数个码元的数据按写速率 R_{w1} 写入行按读速率 R_{r1} 从列读出,第偶数个码元的数据按写速率 R_{w2} 写入行按读速率 R_{r2} 从列读出;同时为每个码元的数据开辟的内存单元大小为 $\text{Max}(K, L)2 - (K-L)2$,其中 K 为行数, L 为列数,且 K 、 L 均为自然数;

当所采用的内存单元为双端口内存时,令 $KR_{r1} \leq LR_{w2}$ 和 $KR_{r2} \leq LR_{w1}$,每个大小为 $K*L$ 的码元中的数据按下面的方式读写到该内存单元,其中 K 为行数, L 为列数,且 K 、 L 均为自然数:

当第一个奇数码元中的 $K*L$ 个数据以速率 R_{w1} 逐行写入内存单元后,该奇数码元中的数据以速率 R_{r1} 按逐列读出;待第一列数据被读出后,下一个偶数码元的数据以速率 R_{w2} 按逐列写入内存单元,并将前面的数据覆盖;此时,每一列写入 L 个数据的时间 LxR_{w2} 不应该小于每一列按速率 R_{r1} 读出前一个奇数码元中 K 个数据的时间 KxR_{r1} ;当前一个奇数码元中的数据都读出后,该偶数码元中 $K*L$ 个数据也都写入到内存单元;随后,该偶数码元中的数据以速率 R_{r2} 按逐行读出;待第一行数据被读出之后,下一个奇数码元的数据以速率 R_{w1} 按逐行写入到内存单元中去,并将前面的数据覆盖;此时,每一行写入 L 个数据的时间 LxR_{w1} 不应该小于每一列按速率 R_{r2} 读出前一个奇数码元中 K 个数据的时间 KxR_{r2} ;如此反复,直到数据被全部从该内存单元读出。

5. 如权利要求 4 所述交织器,其特征在于:所述写速率和读速率相等。

6. 如权利要求 4 所述交织器,其特征在于:第奇数个码元的数据的写速率与第偶数个码元的数据的写速率相等,第奇数个码元的数据的读速率与第偶数个码元的数据的读速率相等。

一种交织器

技术领域

[0001] 本发明涉及移动多媒体通信领域,为一种交织器。

背景技术

[0002] 交织是通信系统中进行数据处理而采用的一种技术,交织器从其本质上来说就是一种实现最大限度的改变信息结构而不改变信息内容的器件。从传统上来讲就是使信道传输过程中所突发产生集中的错误最大限度的分散化。交织器通常是对输入的原始信息序列进行随机置换后从前向后读出。交织器的作用是:一、可以产生长码。二、使两个 RSC 编码器的输入不相关,编码过程趋于独立。交织使编码产生随机度,使码随机化、均匀化,起着对码重量整形的作用,直接影响 Turbo 码的性能。在译码端,对于某一个子译码器来说不可纠正的错误事件,交织后在另一个译码器被打散,成为可纠正差错。

[0003] 交织方式主要有规则交织,不规则交织和随机交织 3 种。通常规则交织即行写列读,效果不好。随机交织指交织格式是随机分配的,是理论上性能最好的交织方式,但是由于要将整个交织信息位置信息传送给译码器,降低了编码效率。实际应用中一般采用不规则交织,这是一种伪随机交织方式,对每一编码块采用固定的交织方式,但块与块之间交织器结构不一样。往往为了获得高的编码增益对交织器的长度提出要求。在无线移动通信系统对时延要求较高,因此采用交织长度为 400 左右的伪随机短交织器。

[0004] 目前,在很多数字通信系统中,大型的交织器正在被广泛的应用。由于这些交织器的规模较大,往往需要占用较大的内存单元,并且完成一次交织或解交织操作所需的运算时间也较长,这都给系统的硬件设计带来了很多问题。

发明内容

[0005] 针对现有技术的不足,本发明的目的在于,提供一种交织器,可减少传输延迟。

[0006] 基于上述发明目的,本发明提供一种交织器,包括以下步骤:

[0007] 预先设定第奇数个码元的数据按写速率 R_{w1} 写入行按读速率 R_{r1} 从列读出,第偶数个码元的数据按写速率 R_{w2} 写入行按读速率 R_{r2} 从列读出;同时为每个码元的数据开辟的内存单元大小为 $\text{Max}(K, L)2 - (K-L)2$,其中 K 为行数, L 为列数,且 K 、 L 均为自然数;

[0008] 当所采用的内存单元为单端口内存时,每个大小为 $K*L$ 的码元中的数据按下面的方式读写到该内存单元,其中 K 为行数, L 为列数,且 K 、 L 均为自然数:

[0009] 当第一个奇数码元中的 $K*L$ 个数据以速率 R_{w1} 按逐行写入所述内存单元后,该奇数码元中的数据以速率 R_{r1} 按逐列读出;待第一列数据被读出后,下一个偶数码元的数据以速率 R_{w2} 按逐列写入内存单元,并将前面的数据覆盖;当前一个奇数码元中的数据都读出后,该偶数码元中 $K*L$ 个数据也都写入到内存单元;随后,该偶数码元中的数据以速率 R_{r2} 按逐行读出;待第一行数据被读出之后,下一个奇数码元的数据以速率 R_{w1} 按逐行写入到内存单元中去,并将前面的数据覆盖;如此反复,直到数据被全部从该内存单元读出。

[0010] 作为本发明的第二目的,本发明还提供一种交织器,包括以下步骤:

[0011] 预先设定第奇数个码元的数据按写速率 R_{w1} 写入行按读速率 R_{r1} 从列读出,第偶数个码元的数据按写速率 R_{w2} 写入行按读速率 R_{r2} 从列读出;同时为每个码元的数据开辟的内存单元大小为 $\text{Max}(K, L)2 - (K-L)2$,其中 K 为行数, L 为列数,且 K, L 均为自然数;

[0012] 当所采用的内存单元为双端口内存时,令 $KR_{r1} \leq LR_{w2}$ 和 $KR_{r2} \leq LR_{w1}$,每个大小为 $K \times L$ 的码元中的数据按下面的方式读写到该内存单元,其中 K 为行数, L 为列数,且 K, L 均为自然数:

[0013] 当第一个奇数码元中的 $K \times L$ 个数据以速率 R_{w1} 逐行写入内存单元后,该奇数码元中的数据以速率 R_{r1} 按逐列读出;待第一列数据被读出后,下一个偶数码元的数据以速率 R_{w2} 按逐列写入内存单元,并将前面的数据覆盖;此时,每一列写入 L 个数据的时间 $L \times R_{w2}$ 不应该小于每一列按速率 R_{r1} 读出前一个奇数码元中 K 个数据的时间 $K \times R_{r1}$;当前一个奇数码元中的数据都读出后,该偶数码元中 $K \times L$ 个数据也都写入到内存单元;随后,该偶数码元中的数据以速率 R_{r2} 按逐行读出;待第一行数据被读出之后,下一个奇数码元的数据以速率 R_{w1} 按逐行写入到内存单元中去,并将前面的数据覆盖;此时,每一行写入 L 个数据的时间 $L \times R_{w1}$ 不应该小于每一列按速率 R_{r2} 读出前一个奇数码元中 K 个数据的时间 $K \times R_{r2}$;如此反复,直到数据被全部从该内存单元读出。

[0014] 本发明采用上述方案,将设定不同的时钟来驱动码元数据的读写,从而使码元数据的读写速率更加灵活,以到达减小交织器传输延迟的目的。

附图说明

[0015] 图 1 为传统交织器实现方法原理图,

[0016] 其中图 1a 表示第奇数个码元的数据按行方向写入内存,按列方向读出;图 1b 表示第偶数个码元的数据按列方向写入内存,按行方向读出;

[0017] 图 2 为本发明的交织器实现方法原理图,

[0018] 其中图 2a 表示第奇数个码元的数据按行方向写入内存,按列方向读出;图 2b 表示第偶数个码元的数据按列方向写入内存,按行方向读出。

具体实施方式

[0019] 传统交织器的实现如图 1 所示。其中,实线表示将数据写到内存空间的方向,虚线表示将数据从内存空间读出的方向。图 1a 表示第奇数个码元的数据按行方向写入内存,按列方向读出;图 1b 表示第偶数个码元的数据按列方向写入内存,按行方向读出。在传统实现方法中,由于码元数据的读写速率相同,所以完成交织器所需的内存单元不能少于:

$$[0020] \quad \text{Max}(K, L)2 - (K-L)2$$

[0021] 其中, K 为行数, L 为列数,且 K, L 均为自然数。则交织器的大小为 $K \times L$ 。如图 1 所示的交织器的大小为 7×5 ,按照传统交织器的硬件实现方法,所需的内存单元至少为:

$$[0022] \quad 7 \times 7 - (7-5) \times (7-5) = (7+2) \times (7-2) = 9 \times 5。$$

[0023] 如果采用单端口内存,即对于同一块内存不能同时进行读写偶做,交织器的读写速率为 R ,即每读写一个数据,需要 R 个时钟单元。那么在传统实现方案中,每读写 $2KL$ 个数据所花的时间为:

$$[0024] \quad T_{2KL} = KLR + KLR + KLR + KLR = 4KLR$$

[0025] 此时,传统方案的平均延时,即平均每交织一个数据,需要消耗的时钟单元为 $2R$ 。

[0026] 同样的条件下,如果采用双端口内存,即可以对同一块内存在不同的地址里同时进行读写操作,那么传统实现方案每读写 $2KL$ 个数据所花的时间为

$$[0027] \quad T_{2KL} = KLR + KLR + |K-L| \min(K, L)R + KLR$$

[0028] 此时,传统方案的平均延时为:

$$[0029] \quad r = \left[\frac{3}{2} + \frac{\min(K, L)}{2KL} |K-L| \right] R$$

[0030] 即:

$$[0031] \quad r = \begin{cases} \frac{4L-K}{2L} R, & K \leq L \\ \frac{4K-L}{2K} R, & K > L \end{cases}$$

[0032] 本发明提出的交织器的实现方案中,将设定不同的时钟来驱动码元数据的读写,从而使码元数据的读写速率更加灵活,以到达减小交织器传输延迟的目的。具体方法如下:

[0033] 1、预先设定:第奇数个码元的数据按写速率 R_{w1} 写入行,即每写一个数据需要 R_{w1} 个时钟单元,按读速率 R_{r1} 从列读出,即每读出一个数据需要 R_{r1} 个时钟单元。

[0034] 2、预先设定:第偶数个码元的数据按写速率 R_{w2} 写入行,即每写一个数据需要 R_{w2} 个时钟单元,按读速率 R_{r2} 从列读出,即每读出一个数据需要 R_{r2} 个时钟单元。

[0035] 3、预先设定:为每个码元的数据开辟的内存单元大小为 $\max(K, L)2 - (K-L)2$,其中 K 为行数, L 为列数,且 K, L 均为自然数;

[0036] 4、如果采用单端口内存,那么该码元中的数据按下面的方式读写到该内存单元:

[0037] 当第一个奇数码元中的 $K*L$ 个数据以速率 R_{w1} 按逐行写入内存单元后,该奇数码元中的数据以速率 R_{r1} 按逐列读出。待第一列数据被读出后,下一个偶数码元的数据以速率 R_{w2} 按逐列写入内存单元,并将前面的数据覆盖。当前一个奇数码元中的数据都读出后,该偶数码元中 $K*L$ 个数据也都写入到内存单元。随后,该偶数码元中的数据以速率 R_{r2} 按逐行读出。待第一行数据被读出之后,下一个奇数码元的数据以速率 R_{w1} 按逐行写入到内存单元中去,并将前面的数据覆盖。这样的过程重复若干次,直到该码元的数据被全部从该内存单元读出。

[0038] 5、如果采用双端口内存,可以令 $KR_{r1} \leq LR_{w2}$ 和 $KR_{r2} \leq LR_{w1}$,那么该码元中的数据按下面方式读写到该内存单元:

[0039] 当第一个奇数码元中的 $K*L$ 各数据以速率 R_{w1} 逐行写入内存单元后,该奇数码元中的数据以速率 R_{r1} 按逐列读出。待第一列数据被读出后,下一个偶数码元的数据以速率 R_{w2} 按逐列写入内存单元,并将前面的数据覆盖。此时,每一列写入 L 个数据的时间 LxR_{w2} 不应该小于每一列按速率 R_{r1} 读出前一个奇数码元中 K 个数据的时间 KxR_{r1} 。当前一个奇数码元中的数据都读出后,该偶数码元中 $K*L$ 个数据也都写入到内存单元。随后,该偶数码元中的数据以速率 R_{r2} 按逐行读出。待第一行数据被读出之后,下一个奇数码元的数据以速率 R_{w1} 按逐行写入到内存单元中去,并将前面的数据覆盖。此事,每一行写入 L 个数据的时间 LxR_{w1} 不应该小于每一列按速率 R_{r2} 读出前一个奇数码元中 K 个数据的时间 KxR_{r2} 。这样的过程重

复若干次,直到该码元的数据被全部从该内存单元读出。

[0040] 这样,如果采用单端口内存,本发明提出的方案每读写 $2KL$ 个数据所花的时间为:

$$[0041] \quad T_{2KL} = KLR_{W1} + KLR_{R1} + KLR_{W2} + KLR_{R2}$$

[0042] 而平时延时为:

$$[0043] \quad r = \frac{R_{W1} + R_{R1} + R_{W2} + R_{R2}}{2}$$

[0044] 如果采用双端口内存,由于 $KR_{R1} \leq LR_{W2}$ 和 $KR_{R2} \leq LR_{W1}$,本发明每读写 $2KL$ 个数据所花的时间为:

$$[0045] \quad T_{2KL} = KLR_{W1} + KLR_{R1} + |K-L| \min(K, L) R_{W2} + KLR_{R2}$$

[0046] 而本发明的平均延时,即平均每交织一个数据,需要消耗的时钟单元为:

$$[0047] \quad r = \frac{R_{W1} + R_{R2} + R_{R1} + \frac{\min(K, L)}{2KL} |K-L| R_{W2}}{2}$$

[0048] 可以看到,如果 $R_{W1} = R_{W2} = R_{R1} = R_{R2} = R_1$,本发明的平均延时和传统方案一致,特别的,如果 $R_{W1} = R_{W2} = R_w$ 且 $R_{R1} = R_{R2} = R_r$,本发明的平均延时为:

$$[0049] \quad r = \frac{[KL + \min(K, L)|K-L]|}{2KL} R_w + R_r$$

[0050] 对于 K 和 L 的不同取值情况有:

$$[0051] \quad r = \begin{cases} \left(1 - \frac{K}{2L}\right) R_w + R_r, & K \leq L \\ \left(1 - \frac{L}{2K}\right) R_w + R_r, & K > L \end{cases}$$

[0052] 另外,特别的如果另 $KR_r = LR_w$,进一步有:

$$[0053] \quad r = \begin{cases} \left(1 + \frac{2L^2 - K^2}{2KL}\right) R_w = \left(1 + \frac{2KL - K^2}{2L^2}\right) R_r, & K \leq L \\ \left(1 + \frac{L}{2K}\right) R_w = \left(\frac{1}{2} + \frac{L}{K}\right) R_r, & K > L \end{cases}$$

[0054] 可以看到,只要该码元的写入速率 R_{W1} 和 R_{W2} 与读出速率 R_{R1} 和 R_{R2} 满足一定关系,该码元的数据就可以正确的从该内存单元中写入和读出,并完成交织运算。特别的,可以令 $KR_{R2} = LR_{W1}$, $KR_{R1} = LR_{W2}$ 。同时,我们也看到,当交织器的行列维数相同时,即 $K = L$ 时,本发明提出的实现方案可以与传统方案在传输延时性能方面取得一致。而对于行列维数不同的交织器,本发明提出的方案通过设置不同的读写速率,可以更灵活的控制交织器的读写延时,并使之与交织器的维数相匹配,改善交织器的整体传输延时性能。

[0055] 本发明提出的交织器的实现方案中,设定不同的时钟来驱动码元数据的读写,从而使码元数据的读写速率更加灵活,更好的与交织器的维数相匹配,已达到减小交织器传输延迟的目的。通过利用可变速率的写操作交织器,有效的改善了交织器在传输延时方面的性能。对于采用大型交织器的通信系统来说有较好的应用前景。

[0056] 本领域内的技术人员应明白,本发明的实施例可提供为方法、系统、或计算机程序产品。因此,本发明可采用完全硬件实施例、完全软件实施例、或结合软件和硬件方面的实施例的形式。而且,本发明可采用在一个或多个其中包含有计算机可用程序代码的计算机

可用存储介质（包括但不限于磁盘存储器和光学存储器等）上实施的计算机程序产品的形式。

[0057] 本发明是参照根据本发明实施例的方法、设备（系统）、和计算机程序产品的流程图和 / 或方框图来描述的。应理解可由计算机程序指令实现流程图和 / 或方框图中的每一流程和 / 或方框、以及流程图和 / 或方框图中的流程和 / 或方框的结合。可提供这些计算机程序指令到通用计算机、专用计算机、嵌入式处理机或其他可编程数据处理设备的处理器以产生一个机器，使得通过计算机或其他可编程数据处理设备的处理器执行的指令产生用于实现在流程图一个流程或多个流程和 / 或方框图一个方框或多个方框中指定的功能的装置。

[0058] 这些计算机程序指令也可存储在能引导计算机或其他可编程数据处理设备以特定方式工作的计算机可读存储器中，使得存储在该计算机可读存储器中的指令产生包括指令装置的制品，该指令装置实现在流程图一个流程或多个流程和 / 或方框图一个方框或多个方框中指定的功能。

[0059] 这些计算机程序指令也可装载到计算机或其他可编程数据处理设备上，使得在计算机或其他可编程设备上执行一系列操作步骤以产生计算机实现的处理，从而在计算机或其他可编程设备上执行的指令提供用于实现在流程图一个流程或多个流程和 / 或方框图一个方框或多个方框中指定的功能的步骤。

[0060] 显然，本领域的技术人员可以对本发明进行各种改动和变型而不脱离本发明的精神和范围。这样，倘若本发明的这些修改和变型属于本发明权利要求及其等同技术的范围之内，则本发明也意图包含这些改动和变型在内。

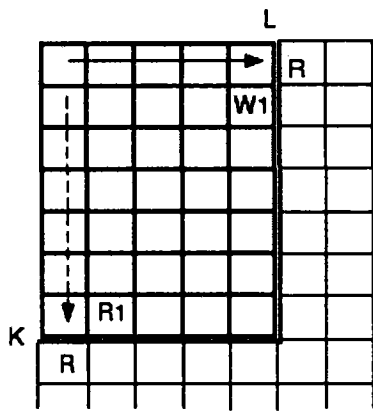


图 1a

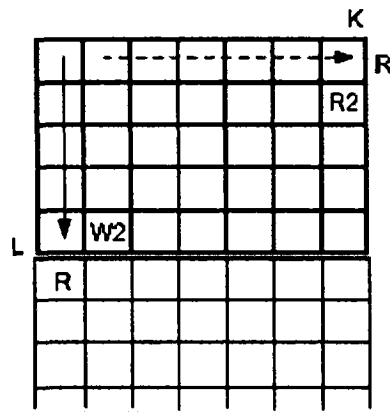


图 1b

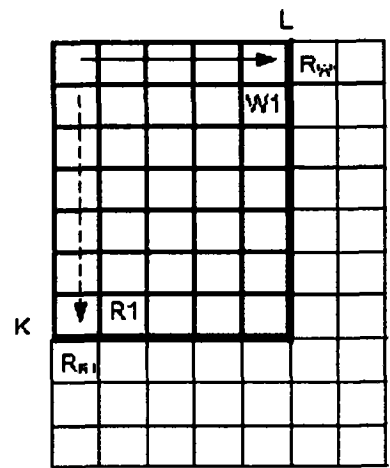


图 2a

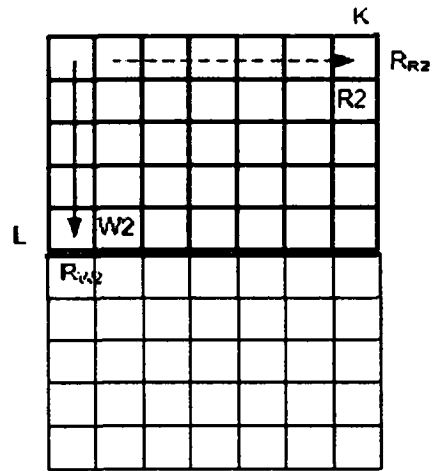


图 2b