

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G11B 20/18 (2006.01)

H03M 13/00 (2006.01)



[12] 发明专利说明书

专利号 ZL 200410102239.6

[45] 授权公告日 2008 年 4 月 23 日

[11] 授权公告号 CN 100383886C

[22] 申请日 1999.2.24

[21] 申请号 200410102239.6

分案原申请号 200410000027.9

[30] 优先权

[32] 1998.2.25 [33] JP [31] 43219/98

[73] 专利权人 松下电器产业株式会社

地址 日本大阪府

[72] 发明人 中迁文男 桥本祐一

[56] 参考文献

CN1040698A 1990.3.21

CN1122025A 1996.5.8

US4845713A 1989.7.4

JP10-55630A 1998.2.24

US5631914A 1997.5.20

审查员 赵梅芳

[74] 专利代理机构 中国专利代理(香港)有限公司
代理人 叶恺东

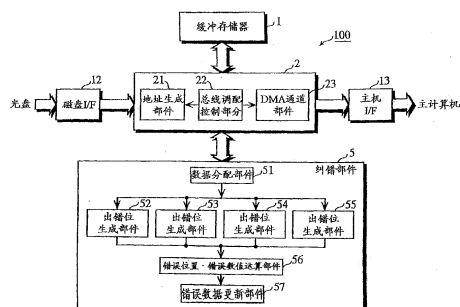
权利要求书 3 页 说明书 41 页 附图 39 页

[54] 发明名称

纠错装置和纠错方法

[57] 摘要

一种纠错装置，用于对接收的数据进行纠错，并且输出纠正后的数据，其特征在于包括：存储单元，用于存储所述接收的数据；纠错单元，用于检测并纠正所述接收的数据中行方向上的错误码，并且检测并纠正来自所述存储单元的所存储的数据中列方向上的错误码。



1.一种纠错装置，用于对输入的块码数据进行纠错，并且输出纠正后的数据，其特征在于包括：

存储器，用于存储从输入单元接收的数据；

纠错单元，用于纠正从所述存储器读出的块码数据中列方向上的错误码，

其中，所述纠错单元通过将从所述存储器读出的从第1列到第 n 列的连续的 n 列数据中的一列作为一个单位以在列方向上对每一行进行纠错，然后，不重复地读取该 n 列数据，而对每一行读取从第 $n+1$ 列到第 $2n$ 列的另外的连续的 n 列数据，这里， n 为2或以上的自然数。

2. 如权利要求1所述的纠错装置，其特征在于，

所述自然数 n 以2的 m 次幂表示，这里， m 为自然数。

3. 如权利要求1所述的纠错装置，其特征在于，

所述自然数 n 是2、4、8的其中之一。

4. 如权利要求1~3中的任一项所述的纠错装置，其特征在于，

所述块码被存储在作为记录介质的光盘上。

5. 如权利要求1所述的纠错装置，其特征在于，

从所述存储器读出的 n 列数据以DMA码被传送。

6. 如权利要求1所述的纠错装置，其特征在于，

所述纠错单元包括进行校正子计算的校正子产生单元，所述纠错单元包括 n 个校正子产生单元，每一个校正子产生单元从 n 列数据中接收一列不同的数据，

所述纠错单元包括进行校正子计算的校正子产生单元，所述校正子产生单元采用时分方式对 n 列进行校正子计算，

所述纠错单元包括进行校正子计算的校正子产生单元，由所述校正子产生单元所进行的校正子计算包括对每一单位数据进行的积和运算。

7. 如权利要求6所述的纠错装置，其特征在于，

所述校正子产生单元包括用以暂时存储数据的FIFO存储器。

8. 如权利要求6所述的纠错装置，其特征在于，

还包括一个错误码更新单元，用以当所述纠错单元检测出错误码时，使用

一个校正值得对应于检测的错误码的一个代码写入存储器。

9. 一种光盘控制装置, 其特征在于包括:

如权利要求4所述的纠错装置;

一个盘接口单元, 用以从一个光盘接收信号, 将所接收的信号解调, 和将所解调的信号输出到所述存储器;

一个主接口单元, 用以从所述存储器接收数据, 并将所接收的数据输出到一个主计算机; 以及

一个总线控制单元, 用于在所述存储器、所述盘接口单元、所述纠错装置以及所述主接口单元中的至少2个之间执行数据传输。

10. 一种光盘读取装置, 其特征在于包括:

如权利要求9所述的光盘控制装置;

一个主轴电动机, 用以使光盘旋转;

一个读取装置, 用以将所述光盘暴露于激光, 根据反射光以读取在光盘上的数据, 并将所读取的数据转换为电信号;

一个放大器, 用以将所述读取装置输出的信号放大;

一个前端处理器, 用以根据由所述放大器输出的信号产生反馈控制信号, 并将该反馈控制信号输出到所述光盘控制装置; 和

一个伺服控制器, 用以根据由所述前端处理器所输出的反馈控制信号控制所述主轴电动机的旋转和所述读取装置的透镜位置。

11. 一种纠错方法, 用以对块码的输入数据进行纠错, 并输出经纠错的数据, 其特征在于包括以下步骤:

将从外部源接收的输入数据存储到一个存储器中;

通过从所述存储器读取数据, 将一列作为一个单位在列方向上对块数据进行纠错;

其中, 在列方向的数据读取中, 对每一个行读取从第1列到第 n 列的 n 列连续的数据, 然后不重复读取该 n 列的数据, 而对每一行读取从第 $n+1$ 列到第 $2n$ 列的另外的 n 列连续的数据, 这里, n 为2或以上的自然数。

12. 如权利要求11所述的纠错装置, 其特征在于,

所述自然数 n 是2、4、8的其中之一。

13. 如权利要求11或12所述的纠错方法, 其特征在于,

所述块码被存储在作为记录介质的光盘上。

14. 如权利要求 11 所述的纠错方法，其特征在于，
纠错是根据校正子计算的结果而进行的，所述校正子计算采用时分方式对 n 列进行。

15. 如权利要求 11 所述的纠错方法，其特征在于，
纠错是根据校正子计算的结果而进行的，所述校正子计算包括对每一单位数据进行的积和运算。

16. 如权利要求 11 所述的纠错方法，其特征在于，
当在所述纠错中检测到一个错误码时，采用校正值写入在所述存储器中的对应于该检出的错误码。

纠错装置和纠错方法

本申请是申请号为 200410002795.6、申请日为 1999 年 2 月 24 日的原案申请的分案申请，该原案的在先申请号为 JP98-43219、在先申请日为 1998 年 2 月 25 日。

技术领域

本发明涉及进行伴随于纠正在编码数据中发生的错误的处理的装置，特别涉及对由 2 维构成的块码进行高速纠错的装置。

背景技术

在磁盘和光盘等作为记录媒体的数据存储装置中，会由于记录媒体上的划伤和污渍而引起数据错误。为了使恢复这样的错误数据成为可能，在将数据记录到记录媒体上时，把用于纠错的代码（以下称之为“纠错码”或“奇偶校验数据”）附加到数据中，在进行再现时利用纠错码检测出错误的的数据，并进行将其纠正为正确数据的处理。在这种附加纠错码及纠错处理（以下将这两种处理合称为“伴随纠错的处理”）中，为了提高其纠错能力，在大多数系统中采用 Reed-Solomon 码作为纠错码，并采用乘积码作为数据的构成方法。

图 1 示出了乘积码。对于由 $k1 \times k2$ 个字节构成的信息数据，在行方向上（以下称为“C1 序列”）上附加 $m1$ 字节的奇偶校验数据，在列方向上（以下称为“C2 序列”）上附加 $m2$ 字节的奇偶校验数据。在由这些信息数据、C1 个奇偶校验数据及 C2 个奇偶校验数据构成了一个作为纠错最大单位的数据块时，其字节数为 $n1 \times n2$ 字节。通常，作为将信息数据和奇偶校验数据存储到 DRAM 等存储器中的方法，使 C1 序列中的存储器地址加 1 以便于进行存储。由此，构成 C1 序列的代码串的数据被存储到地址连续的存储区域中，但构成 C2 序列的代码串的数据则被存储到地址不连续的存储区域中。构成乘积码的各个代码被简称为“数据”。

图 2 示出了与 C1 代码串对应的纠错流。首先对第 1 行的代码串进行纠错。接着对第 2 行的代码串进行纠错。对全部 $n2$ 行重复这样的操作。即，在列方向上逐行进行扫描该行中的全部数据并进行纠错的操作。所谓“扫描”就是在

从存储区域中读出作为错误检测和纠正的对象的数据时的读出顺序。并且，在“错误检测”中不包含所谓的纠错处理，但在“纠错”中包含作为其前置处理的错误检测。

图3示出了与C2代码串对应的纠错流。首先对第1列的代码串进行纠错。接着对第2列的代码串进行纠错。对全部n1列重复这样的操作。即，在行方向上逐列进行扫描该列中的全部数据并进行纠错的操作。

图4是沿着上述流向进行纠错的现有纠错装置的构成示意图。其中，d1到d100是100字节的信息数据，p1到p10是10字节的奇偶校验数据，并且这些数据构成了1个代码串。

从存储器中读出这些信息数据及奇偶校验数据并顺序输入到出错位生成装置900中。出错位生成装置900在输入一个数据时进行预定的运算，到完成了全部输入构成1个代码串的信息数据d1~d100和奇偶校验数据p1~p10时为止，生成10个出错位。在这10个出错位中存在非零数的情况下，由于在该代码串中发生了错误，所以错误位置·错误数值运算装置901利用这些出错位计算出错误位置和错误数值。其中，错误位置是表示代码串中的第几个数据是错误数据的信息，错误数值表示错误的大小。最后，错误数据更新装置902通过这些错误位置和错误数值，从存储器中读出该代码串中的错误数据，并将纠正后的数据回写到原来的位置上。在每个代码串中重复上述操作。

如上所述，现有的纠错装置通过在对C1序列进行逐行扫描和纠错时，对C2序列进行逐列扫描和纠错，进行对一个数据块的全部代码串的纠错。

但是，随着最近以光盘驱动器装置为代表的数据存储装置对处理速度高速化的要求越来越严格，现有的纠错装置适应这样的要求是非常困难的。

为了满足这样的严格要求，虽然考虑到在1台数据存储装置内并列设计多个纠错装置，却过度增大了电路规模而显著降低了性能价格比。

发明内容

本发明鉴于上述问题，其目的是提供以较小的电路规模高速执行伴随于纠错的处理的纠错装置。

为了达到上述目的，根据本发明的第一项发明，一种反复执行伴随于对构成R行L列的块码的行方向及列方向的代码串纠错的运算的纠错装置，其特征在于，包括：

存储装置，用于存储所述块码；

运算装置，用于以所述块码的 1 行或 1 列代码串为单位进行误码纠正所需要的运算；

传输装置，具有行方向传输部件，用于在 $R1$ 为大于 2 小于 R 的整数时，对 R 行分别按顺序反复进行如下操作：读出存储在所述存储装置中的块码的 $R1$ 行代码串并将其传输到所述运算装置中，

所述行方向传输部件在 $L1$ 为大于 2 小于 L 的整数时，在行方向上只移位 $L1$ 个代码，反复进行 $R1$ 行 $L1$ 列传输：所述 $R1$ 行 $L1$ 列传输是指按顺序对所述 $R1$ 行中的各行进行连续读出并列的 $L1$ 个代码并将其传输到所述运算装置中；

所述运算装置在代码是从所述行方向传输部件传送来的情况下，对传送来的每 $L1$ 个代码并列进行对所述 $R1$ 行代码串的所述运算，来构成与所述 $R1$ 行不同行的代码串。

由此，行方向上的纠错是对多个代码串并列进行并在行方向上分割多行代码串，并使得到的多个小数据块分别按照 Z 字形扫描顺序进行纠错。即，按照一边换行一边只对各行的一部分代码反复扫描的顺序并列地进行纠错，因此与单纯的换行对各行全部代码反复扫描的并列处理相比，虽然需要以相同的速度进行扫描和纠错，但降低了输入到各行的纠错电路中的代码的平均输入速度，并缩小了每行的纠错电路中所必需的队列缓冲器等电路的规模。而且，由于各行的扫描在同一行中是连续的并且以编码为对象，所以在块码被存储在具有 2 维地址的 DRAM 等存储区域中的情况下，可以对同一行地址中的连续列地址进行访问，从而使高速扫描成为可能。

并且，根据本发明的第 2 项发明，在本发明的第 1 项发明的纠错装置中，所述块码为乘积码，在构成所述块码的行方向及列方向的代码串中分别包含信息代码和纠错码，

所述运算装置包括用于以 1 行或 1 列代码串为单位，检测是否存在误码的错误检测装置，和用于在通过所述错误检测装置检测出存在误码的情况下，将与该误码相当的所述存储装置中的代码重写成纠正后的值的误码更新装置，

所述运算装置在代码是从所述行方向传输部件传送来的情况下，对传送来的每 $L1$ 个代码并列进行对所述 $R1$ 行代码串的所述运算，来构成与所述 $R1$ 行不同行的代码串。

由此，由于依据本发明的纠错装置能够适用于对乘积码纠错中的行方向上的错误检测，从而提高了光盘等的读出装置及数据通信中的接收装置等执行对接收的乘积码进行纠错的装置的错误检测速度。

并且，根据本发明的第3项发明，在本发明的第2项发明的纠错装置中，在所述存储装置中，属于所述块码同一行的代码被存入具有连续地址的存储区域中，

所述行方向传输部件从所述存储装置的具有连续地址的存储区域中连续读出所述L1个代码。

由此，为了从存储装置中读出属于同一行的代码，增加输出到存储装置中的地址，从而缩短了从存储装置中读出的时间。

并且，根据本发明的第4项发明，在本发明的第3项发明的纠错装置中，所述存储装置为动态随机存取存储器，

所述行方向传输部件以页面模式从所述存储装置中读出所述L1个代码。

由此，为了从存储装置中读出属于同一行的代码，在将1个RAS地址输出到存储装置中之后连续输出CAS地址，从而谋求DRAM的命中页的读出速度高速化。

并且，根据本发明的第5项发明，在本发明的第4项发明的纠错装置中，由所述行方向传输部件执行的传输是通过对所述存储装置的直接存储器访问而产生的传输。

由此，由于作为能够在错误检测装置中并列处理的对象的全部代码被连续从存储装置中读出和输入，从而避免了在由错误检测装置对1个较小的数据块进行错误检测时中断代码串的输入，有效地实现基于流水线处理的纠错。

并且，根据本发明的第6项发明，在本发明的第2项发明的纠错装置中，所述错误检测装置包括R1个用于进行对1个代码串的错误检测的单位错误检测部件和用于在从所述行方向传输部件传送代码的情况下，反复进行将每次传送的L1个代码分别按顺序分配到所述R1个单位错误检测部件中的分配部件，

所述R1个单位错误检测部件分别相互独立并且与所述行方向传输部件所执行的传输并行地进行所述错误检测。

由此，在行方向上的纠错中，各个单位错误检测装置以一次L1个代码的频率从分配装置输入R1次代码并在从分配装置向其它单位错误检测装置输入

代码期间进行错误检测，从而减轻了对各个单位错误检测装置要求的处理能力并削减了电路规模。

并且，根据本发明的第7项发明，在本发明的第6项发明的纠错装置中，所述R1个单位错误检测部件分别在将来自所述分配部件的L1个代码反复进行分配的情况下，在比该重复周期短的周期内，完成对L1个代码的错误检测。

由此，在列方向上的纠错中，对于各个单位错误检测装置来说，代码的输入率（在单位时间内输入的代码个数）在处理能力（单位时间内接受错误检测处理的代码个数）的范围之内，从而缩小了各个单位错误检测装置的输入段中所必需的队列缓冲器的大小。

并且，根据本发明的第8项发明，在本发明的第7项发明的纠错装置中，所述R1个单位错误检测部件分别具有用于对1个代码串计算出多个出错位的积和运算电路。

由此，依据本发明的纠错装置是适用于Reed-Solomon码的装置。

并且，根据本发明的第9项发明，在本发明的第2项发明的纠错装置中，在所述L除以所述L1所得的商为D、余数为E时，所述行方向传输部件在一边在行方向上只移位L1个代码一边重复D次所述的R1行L1列传输之后，对所述R1行分别按顺序对各行连续读出并列的E个代码并将其传输到所述错误检测装置中，

所述错误检测装置在进行了预定次数的对每L1个代码的所述错误检测之后，进行对每E个代码的所述错误检测。

由此，即使构成乘积码的行方向上的码长是一次连续读出的代码数L1的倍数，直到各代码串的结尾，都可以无问题地并列执行纠错，从而实现了与乘积码的大小无关的柔性纠错装置。

并且，根据本发明的第10项发明，在本发明的第2项发明的纠错装置中，所述传输装置包括列方向传输部件，用于在L2为大于2小于L的整数时，对L列按顺序反复进行如下操作：读出存储在所述存储装置中的块码的L2列代码串并将其传输到错误检测装置中，

所述列方向传输部件在行方向上只移位L2个代码，反复进行R行L2列传输：所述R行L2列传输是指按顺序对各行进行连续读出并列的L2个代码并将其传输到所述错误检测装置中，

所述错误检测装置在从所述列方向传输部件传送代码的情况下，将每次传送来的 $L2$ 个代码并列进行对所述 $L2$ 列代码串的错误检测，作为属于相对应的所述 $L2$ 列代码串的代码。

由此，对多个代码串并列执行行方向上的纠错和列方向上的纠错，而且，按照换行只对各行的一部分代码反复扫描的顺序执行对列方向的纠错。这样，与单纯的换行对各行的全部代码反复扫描的并列处理相比，虽然需要以相同的速度进行扫描和纠错，但降低了输入到各行的纠错电路中的代码的平均输入速度，并缩小了每行的纠错电路中所必需的队列缓冲器等的电路规模。

并且，根据本发明的第 11 项发明，在本发明的第 10 项发明的纠错装置中，所述 $R1$ 和所述 $L2$ 为 $R1=L2=j$ ，

所述错误检测装置包括 j 个用于进行对 1 个代码串的错误检测的单位错误检测部件和用于在从所述行方向传输部件传送代码的情况下，反复进行将每次传送的 $L1$ 个代码分别按顺序分配到所述 j 个单位错误检测部件中，并在从所述列方向传输部件传送代码的情况下，反复进行将传送的 $L2$ 个代码分别分配到相对应的所述 j 个单位错误检测部件中的分配部件，

所述 j 个单位错误检测部件分别相互独立并且与所述行方向传输部件和所述列方向传输部件所执行的传输并行地进行所述错误检测。

由此，作为行方向纠错中的并列处理对象的行数与作为列方向纠错中的并列处理对象的列数相等，从而各方向的纠错中所需要的单位错误检测装置的个数相同，无论进行哪个方向上的纠错，都能够充分利用错误检测装置的能力。

并且，根据本发明的第 12 项发明，在本发明的第 11 项发明的纠错装置中，所述 j 个单位错误检测部件分别在将来自所述分配部件的 $L1$ 个代码反复进行分配的情况下，在比该重复周期短的周期内，完成对 $L1$ 个代码的错误检测，在将来自所述分配部件的 $L2$ 个代码中的一个反复进行分配的情况下，在比该重复周期更短的周期内，完成对所分配的 1 个代码的错误检测。

由此，在列方向纠错中，对于各个单位错误检测装置来说，代码的输入率（在单位时间内输入的代码个数）在处理能力（单位时间内接受错误检测处理的代码个数）的范围之内，从而缩小了各个单位错误检测装置的输入段中所必需的队列缓冲器的大小。

并且，根据本发明的第 13 项发明，在本发明的第 12 项发明的纠错装置中，

所述R1、所述L1和所述L2为 $R1=L1=L2=j$ 。

由此，在行方向及列方向中的任一方向上的纠错中，乘积码的读出顺序相同，从而使与对存储装置的访问相关的传输装置的控制电路简单化。

并且，根据本发明的第14项发明，一种反复执行伴随于对构成R行L列的块码的行方向及列方向的代码串纠错的运算的纠错装置，其特征在于，包括：

存储装置，用于存储所述块码；

运算装置，用于以所述块码的1行或1列代码串为单位进行误码纠正所需要的运算；

传输装置，具有列方向传输部件，用于在L2为大于2小于L的整数时，对L列按顺序反复进行如下操作：读出存储在所述存储装置中的块码的L2列代码串并将其传输到运算装置中，

所述列方向传输部件在行方向上只移位L2个代码，反复进行：对所述R行分别按顺序进行的R行L2列传输，所述R行L2列传输是指对各行连续读出并列的L2个代码并将其传输到所述运算装置中；

其中

所述运算装置在从所述列方向传输部件传送代码的情况下，将每次传送来的L2个代码并列进行对所述L2列代码串的所述运算，作为属于相对应的所述L2列代码串的代码。

由此，列方向上的纠错是对多个代码串并列进行并在行方向上分割多行代码串，并使得到的多个小数据块分别按照Z字形扫描顺序进行纠错。即，按照换行只对各行的一部分代码反复扫描的顺序并列地进行纠错，因此与单纯的换列对各列全部代码反复扫描的并列处理相比，虽然需要以相同的速度进行扫描和纠错，但降低了输入到各列的纠错电路中的代码的平均输入速度，并缩小了每行的纠错电路中所必需的队列缓冲器等电路的规模。而且，由于各行的扫描在同一行中是连续的并且以编码为对象，所以在块码被存储在具有2维地址的DRAM等存储区域中的情况下，可以对同一行地址中的连续列地址进行访问，从而使高速扫描成为可能。

而且，由于进行行方向纠错的纠错装置所具备的上述特征构成要素也适用于列方向纠错，所以能够得到与行方向纠错相关的纠错装置相同的效果。

而且，为了达到上述目的，根据本发明的第23项发明的一种用于在对从

外部的第一装置接收的代码进行纠错后输出到外部的第二装置的纠错装置，其特征在于，

存储装置，具有用于存储所述代码的存储区域；

错误检测装置，用于以预定数目的代码为单位，检测是否存在错误的代码；

纠错装置，用于以预定数目的代码为单位，纠正所述存储装置中的错误代码；

第一传输装置，用于将从所述第一装置输出的代码并行传输到所述存储装置和所述错误检测装置中，在将该代码存入所述存储装置的同时，在所述错误检测装置中进行所述检测；

第二传输装置，用于将通过所述错误检测装置检测出存在错误代码的所述预定数目的代码从所述存储装置传输到所述纠错装置，并在所述纠错装置中对该代码进行所述纠正；

第三传输装置，用于将通过所述错误检测装置或所述纠错装置检测出不存在错误代码或错误代码经过纠正的预定数目的代码从所述存储装置传输到所述第二装置；

传输控制装置，用于控制所述第一至第三传输装置，以使其分别排他地执行所述代码传输。

由此，在从第一装置输出的代码被存储在存储装置中的同时接受错误检测处理，从而与现有的在被暂时存入存储装置之后读出并输入到纠错装置中的处理顺序相比，在缩短了纠错所要求的整个处理时间的同时减少了对存储装置的访问次数，并实现了高速的低功耗纠错装置。

并且，根据本发明的第24项发明，在本发明的第23项的发明的纠错装置中，所述错误检测装置具有检测结果记录部件，用于记录是否存在错误代码的检测结果，

所述纠错装置通过参照记录在所述检测结果记录部件中的检测结果，控制第二传输装置以便仅仅将检测出存在错误代码的所述预定数目的代码从所述存储装置中传输到所述纠错装置中。

由此，在存入存储装置的同时进行的错误检测中，对于没有检测出错误的代码，跳过了所谓从存储装置向纠错装置传输及通过纠错装置纠错的两个处理，从而在使纠错高速化的同时，避免了对存储装置的无效访问。

并且,根据本发明的第25项发明,在本发明的第24项的发明的纠错装置中,所述第一装置按顺序反复输出构成 R 行 L 列块码的行方向代码串,

所述块码为乘积码,在构成所述块码的行方向及列方向的代码串中分别包含信息代码和纠错码,

所述纠错装置对行方向及列方向的代码串进行纠错,在对行方向的代码串进行纠错的情况下,通过参照记录在所述检测结果记录部件中的检测结果,控制第二传输装置以便仅仅将检测出存在错误代码的行方向的代码串从所述存储装置中传输到所述纠错装置中。

由此,依据本发明的纠错装置能够用于要求高速执行对乘积码的纠错的光盘等读出装置和数据通信中的接收装置中。

并且,根据本发明的第26项发明,在本发明的第25项的发明的纠错装置中,所述错误检测装置将对构成1个块码的全部行方向的代码串的检测结果存入并记录到所述检测结果记录部件中,

所述纠错装置在对列方向的代码串进行纠错的情况下,通过参照记录在所述检测结果记录部件中的检测结果,控制第二传输装置以便仅仅将检测出存在错误代码的构成块码的列方向的代码串从所述存储装置中传输到所述纠错装置中。

由此,在构成1个块码的行方向上的全部代码串中,在没有检测出错误的情况下,跳过对该块码的列方向上的代码串的错误检测及纠错处理,从而在从光盘等记录媒体中连续读出的块码中的错误发生率较小的情况下,增大了相应的平均读出速度。

并且,根据本发明的第27项发明,在本发明的第23项的发明的纠错装置中,所述第二传输装置具有行方向传输部件,用于在 $R1$ 为大于2小于 R 的整数时,对 R 行按顺序反复进行如下操作:从所述存储装置中读出检测出存在错误代码的 $R1$ 行代码串并将其传输到所述纠错装置中,

所述行方向传输部件在 $L1$ 为大于2小于 L 的整数时,在行方向上只移位 $L1$ 个代码,反复进行 $R1$ 行 $L1$ 列传输:所述 $R1$ 行 $L1$ 列传输是指按顺序对所述 $R1$ 行中的各行进行连续读出并列的 $L1$ 个代码并将其传输到所述纠错装置中;

所述纠错装置在代码是从所述行方向传输部件传送来的情况下,对传送来的每 $L1$ 个代码并列进行对所述 $R1$ 行代码串的纠错,作为构成与所述 $R1$ 行不同行的代码串。

由此,对多个行方向的纠错不仅被并列执行,而且是通过反复对较小的数

据块纠错而进行的,从而与单纯的并列执行纠错的情况相比,虽然以同一速度进行扫描和纠错,但降低了纠错装置中被要求的处理能力并缩小了电路规模。

并且,根据本发明的第28项发明,在本发明的第27项的发明的纠错装置中,所述第二传输装置还具有列方向传输部件,用于在 $L2$ 为大于2小于 L 的整数时,对 L 列按顺序反复进行如下操作:从所述存储装置中读出检测出存在错误代码的块码的 $L2$ 列代码串并将其传输到所述纠错装置中,

所述列方向传输部件在行方向上只移位 $L2$ 个代码,反复进行 R 行 $L2$ 列传输,所述 R 行 $L2$ 列传输是指按顺序对所述 R 行中的各行进行连续读出并列的 $L2$ 个代码并将其传输到所述纠错装置中,

所述纠错装置在从所述列方向传输部件传送代码的情况下,将每次传送来的 $L2$ 个代码并列进行对所述 $L2$ 列代码串的纠错,作为属于相对应的所述 $L2$ 列代码串的代码。

由此,并列执行行方向上的纠错和列方向上的纠错,从而以较小的电路规模实现了高速的纠错装置。

并且,根据本发明的第29项发明,在本发明的第23项的发明的纠错装置中,由所述第一传输装置执行的从所述第一装置向所述存储装置的代码传输、由所述第二传输装置执行的从所述存储装置向所述纠错装置的代码传输及由所述第三传输装置执行的从所述存储装置向所述第二装置的代码传输为由对所述存储装置的直接存储器访问所执行的传输,

所述传输控制装置,在需要执行由所述第一传输装置执行的DMA传输的情况下,在正在执行所述第一至第三传输装置的DMA传输中的任一个时,在完成这个执行中的DMA传输后执行由所述第一传输装置执行的DMA传输,在没有执行所述第一至第三传输装置的DMA传输中的任一个时,立刻执行由所述第一传输装置执行的DMA传输。

由此,最优先执行对存储装置的缓冲,从而在将依据本发明的纠错装置用于光盘驱动器等读出装置中的情况下,光拾取器可以连续读出光盘上的代码。

并且,根据本发明的第30项发明,在本发明的第29项的发明中的纠错装置,所述传输控制装置,在需要执行由所述第二传输装置执行的DMA传输的情况下,只有在没有执行所述第一至第三传输装置的DMA传输中的任一个时,才执行由所述第二传输装置执行的DMA传输。

由此,通过考虑到由与对存储装置的缓冲并列的错误检测而导致的纠错高速化和由对多个代码串进行并列处理而导致的纠错高速化的高效调度,使对存储装置的排他性访问控制成为可能。

并且,根据本发明的第31项发明,在本发明的第30项的发明的纠错装置中,所述纠错装置在根据通过所述第二传输装置传输的代码串确定错误代码和纠正后的值之后,在所述传输控制装置的控制下,将所述存储装置中所对应的错误代码重写成纠正后的值,

所述传输控制装置只有在不执行所述第一至第三传输装置的DMA传输中的任一个时,才执行由所述纠错装置进行的错误代码的重写。

由此,能够更平衡地对与1个存储装置资源相对应的4种访问进行排他性控制,实现较高性能价格比的纠错装置。

而且,为了达到上述目的,根据本发明的第32项发明的一种用于在对从外部的第一装置接收的代码进行必要的纠错后输出到外部的第二装置的纠错装置,其特征在于,

存储装置,具有用于存储所述代码的存储区域;

纠错装置,用于以预定数目的代码为单位,检测错误代码并纠正对应的所述存储装置中的错误代码;

代码选择装置,用于选择从所述第一装置传送的代码和从所述存储装置传送的代码中的任一个,送至所述纠错装置中,并在所述纠错装置中进行所述检测和纠正;

第一传输装置,用于将从所述第一装置输出的代码并行传输到所述存储装置和所述代码选择装置中,在将该代码存入所述存储装置的同时,在所述代码选择装置中进行选择并将其送至纠错装置,在所述错误检测装置中对该代码进行所述检测和纠正;

第二传输装置,用于将预定数目的代码从所述存储装置传输到所述代码选择装置,在所述代码选择装置中选择该代码并将其送至所述纠错装置中,在所述纠错装置中对该代码进行所述检测和纠正;

第三传输装置,用于将通过所述纠错装置检测出不存在错误代码或错误代码经过纠正的预定数目的代码从所述存储装置传输到所述第二装置;

传输控制装置,用于控制所述第一至第三传输装置,以使其分别排他地执行所述代码传输。

由此,从第一装置输出的代码在被存入存储装置的同时,通过选择装置被输入到纠错装置中,从而与现有的在被暂时存入存储装置之后读出并输入到纠错装置中的处理顺序相比,在缩短了纠错所要求的整个处理时间的同时减少了对存储装置的访问次数,并实现了高速的低功耗纠错装置。

并且,根据本发明的第33项发明,在本发明的第32项的发明的纠错装置

中,所述第一装置按顺序反复输出构成R行L列块码的行方向代码串,

所述块码为乘积码,在构成所述块码的行方向及列方向的代码串中分别包含信息代码和纠错码,

所述第一传输装置,将从所述第一装置输出的代码并行传输到所述存储装置和所述代码选择装置中,在将该代码存入所述存储装置的同时,在所述代码选择装置中进行选择并将其送至纠错装置,在所述错误检测装置中对该代码进行所述检测和纠正;

所述第二传输装置,将所述块码的列方向代码串从所述存储装置传输到所述代码选择装置,在所述代码选择装置中选择该代码并将其送至所述纠错装置中,在所述纠错装置中对该代码进行所述检测和纠正。

由此,实现了能够适用于要求高速执行对乘积码的纠错的光盘等读出装置和数据通信中的接收装置等中的纠错装置。

并且,根据本发明的第34项发明,在本发明的第33项的发明的纠错装置中,所述纠错装置包括用于以行方向或列方向的代码串为单位,检测是否存在错误代码的错误检测部件,和

用于在通过所述错误检测部件检测出存在错误代码的情况下,将与该错误代码相对应的所述存储装置中的代码重写成纠正后的值的代码更新部件,

所述错误检测部件包括用于记录对通过所述第一传输装置从所述第一装置传输来的行方向代码串的错误检测结果的行方向检测结果记录部件,和

用于记录对通过所述第二传输装置从所述存储装置传输来的列方向代码串的错误检测结果的列方向检测结果记录部件,

所述错误代码更新部件利用记录在所述行方向检测结果记录部件及所述列方向检测结果记录部件中的检测结果,重写所述存储装置中的代码。

由此,将行方向的纠错结果和列方向的纠错结果进行单独保存,从而利用这些结果进行后续处理的代码更新装置等可以以更慢的速度进行处理。

并且,根据本发明的第35项发明,在本发明的第34项的发明的纠错装置中,所述错误检测部件在进行行方向的错误检测的情况下,将对构成1个块码的全部行方向代码串的检测结果显示存入和记录到所述行方向检测结果记录部件中,在进行列方向纠错的情况下,通过参照记录在所述检测结果记录部件中的检测结果,控制第二传输装置以便仅仅将检测出存在错误代码的构成块码的列方向的代码串从所述存储装置中传输到所述纠错装置中。

由此,在构成1个块码的行方向上的全部代码串中,在没有检测出错误的情况下,跳过对该块码的列方向上的代码串的错误检测及纠错处理,增大了相

应的平均读出速度。

并且,根据本发明的第36项发明,在本发明的第35项的发明的纠错装置中,所述传输控制装置以构成这些代码串的部分代码为单位切换由所述第一传输装置执行的从所述第一装置向所述代码选择装置的1行代码串的传输和由所述第二传输装置执行的从所述存储装置向所述代码选择装置的1列代码串的传输,交互地执行,

所述错误检测部件通过以所述部分代码为单位,交互地切换对由所述第一传输装置从所述第一装置传输来的行方向代码串的错误检测和对由所述第二传输装置从所述存储装置传输来的列方向代码串的错误检测,并列进行对各代码串的错误检测,并将各自的结果记录到行方向检测结果记录部件及列方向检测结果记录部件中,

所述错误代码更新部件通过顺序参照记录在所述行方向检测结果记录部件和所述列方向检测结果记录部件中的检测结果,顺序地进行行方向上的所述存储装置中的代码重写和列方向上的所述存储装置中的代码重写。

由此,代码更新装置可以按照并列进行行方向纠错和列方向纠错的顺序进行操作,从而削减了电路规模。

并且,根据本发明的第37项发明,在本发明的第32项的发明的纠错装置中,所述第二传输装置还具有列方向传输部件,用于在 $L2$ 为大于2小于 L 的整数时,对 L 列按顺序反复进行如下操作:从所述存储装置中读出检测出存在错误代码的块码的 $L2$ 列代码串并经过所述代码选择装置将其传输到所述纠错装置中,

所述列方向传输部件在行方向上只移位 $L2$ 个代码,反复进行 R 行 $L2$ 列传输,所述 R 行 $L2$ 列传输是指按顺序对所述 R 行中的各行进行连续读出并列的 $L2$ 个代码并将其传输到所述纠错装置中,

所述纠错装置在从所述列方向传输部件传送代码的情况下,将每次传送来的 $L2$ 个代码并列进行对所述 $L2$ 列代码串的纠错,作为属于相对应的所述 $L2$ 列代码串的代码。

由此,不仅并列执行对多个行方向的纠错,而且还反复执行对小数据块的纠错,从而与单纯进行并列处理的情形相比,虽然需要以相同的速度进行扫描和纠错,但是降低了纠错装置中所要求的处理能力,并缩小了电路规模。

而且,把向存储器装置的存取作为DMA,其中设置一定的优先顺序来进行调度,由此能够更平衡地对与1个存储装置资源相对应的4种访问进行排他性控制,实现性能价格比较高的纠错装置。

附图说明

图1示出了乘积码的数据结构。

图2示出了按照每行的顺序进行C1序列的纠错时的处理顺序。

图3示出了按照每列的顺序进行C2序列的纠错时的处理顺序。

图4示出了现有的纠错装置的纠错处理顺序。

图5是本发明第一实施例中的纠错装置100的构成示意图。

图6是装置100的总线控制器2的详细结构示意图。

图7是装置100的数据分配器51的详细结构示意图。

图8是装置100的1个出错位生成器52(53、54、55)的详细结构示意图。

图9是装置100中的主要处理(数据输入、纠错、数据输出)的时序图。

图10示出了在进行C1序列的纠错时读出缓冲存储器1内的乘积码的顺序。

图11示出了在按照图10所示的顺序从缓冲存储器1传输数据时,数据分配器51的操作。

图12示出了C1序列的纠错中通过从缓冲存储器1向纠错器5的数据传输与4个出错位生成器52~55的积和运算时序。

图13示出了在进行C2序列的纠错时读出缓冲存储器1内的乘积码的顺序。

图14示出了在按照图13所示的顺序从缓冲存储器1传输数据时,数据分配器51的操作。

图15示出了C2序列的纠错中通过从缓冲存储器1向纠错器5的数据传输与4个出错位生成器52~55的积和运算时序。

图16示出了对C1序列的端数部分进行的处理时序。

图17示出了对C2序列的端数部分进行的处理时序。

图18示出了以两列在C1方向上进行纠错时,从缓冲存储器读出数据的顺序。

图19示出了以两列在C1方向上进行纠错时数据分配器的操作。

图20示出了以两列在C1方向上进行纠错时数据传输与出错位计算的时序。

图21示出了以两列在C2方向上进行纠错时,从缓冲存储器读出数据的顺序。

图22示出了以两列在C2方向上进行纠错时数据分配器的操作。

图23示出了以两列在C2方向上进行纠错时数据传输与出错位计算的时

序。

图 24 是本发明第二实施例中的光盘读取装置 2000 的结构示意图。

图 25 是装置 2000 的光盘控制器 2100 的详细结构示意图。

图 26 是装置 2000 的总线控制器 2180 的详细结构示意图。

图 27 示出了通过装置 2000 的总线调配控制器 2182 的 3 个排他性 DMA 传输的控制例。

图 28 是装置 2000 的纠错器 2130 的详细结构示意图。

图 29 是示出了光盘控制器 2100 内的处理过程的时序图。

图 30 是图 29 所示的时序图的更详细的示图。

图 31 是示出了在进行 C1 序列的纠错时纠错器 2130 内的行单位处理过程的时序图。

图 32 示出了在纠正图 31 所示的代码串时，访问存储在缓冲存储器 2110 中的乘积码的顺序。

图 33 示出了在以 4 个代码串并行对图 31 所示的 C1 序列纠错时，从磁盘接口 2120 读出数据的顺序。

图 34 示出了对图 31 所示的序列 C1 的端数部分的读出顺序。

图 35 是本发明第三实施例中的光盘读取装置 3000 的结构示意图。

图 36 是装置 3000 的光盘控制器 3100 的详细结构示意图。

图 37 是装置 3000 的纠错器 3130 的详细结构示意图。

图 38 是示出了在连续处理多个数据块的数据时光盘控制器 3100 内的处理过程的时序图。

图 39 是图 38 所示的时序图的更详细的示图。

图 40 是示出了在纠错器 2130 内的行方向及列方向上的单位处理过程的时序图。

具体实施方式

以下，参照附图详细说明本发明的实施例。

（第一实施例）

第一实施例涉及对行方向及列方向中的任一方向上的 4 个代码串并列进行纠错的高速纠错装置。

图 5 是本发明第一实施例中的纠错装置 100 的构成示意图。装置 100 是包

含在光盘驱动器装置中并用于对从光盘中读出的数据进行纠错的装置，由缓冲存储器 1、包含在该缓冲存储器 1 中的进行访问的 3 个处理块（磁盘接口部件 12、纠错部件 5 及主机接口部件 13）以及进行访问的排他性控制的总线控制器 2 构成。图中的箭头表示作为纠错对象的数据的流向。装置 100 采用 Reed-Solomon 码作为纠错码类型，采用乘积码作为代码结构。

缓冲存储器 1 是利用二维地址（行地址和列地址）具有特定存储区域的 DRAM 等半导体存储器，具有暂时存储多个作为纠错对象的数据块的存储容量。

磁盘接口部件 12 是将从光盘等记录媒体中读出的数据接着写入缓冲存储器 1 中的接口电路。

纠错部件 5 是用于顺序读出从磁盘接口部件 12 写入缓冲存储器 1 中的数据并检测是否发生了错误，如果发现错误，就在缓冲存储器 1 上进行纠错的电路。具体地，纠错部件 5 是对 4 个代码串同时并列执行纠错的电路，由数据分配部件 51、4 个出错位生成部件 52~55、错误位置·错误数值运算部件 56 及错误数据更新部件 57 构成。

数据分配部件 51 将从缓冲存储器 1 传输来的数据分配到 4 个出错位生成部件 52~55 中的一个中。此时，这样进行分配以便于属于同一代码串中的数据被输入到同一出错位生成部件中。4 个出错位生成部件 52~55 都是相同的电路，分别在输入了所分配的数据时就进行积和运算，当包含 10 个奇偶校验数据的 1 个代码串被全部输入之后，生成 10 个出错位。

错误位置·错误数值运算部件 56 分别对 4 个出错位生成部件 52~55 依次判断所生成的 10 个出错位中是否存在非零数，如果存在，就利用这些出错位计算出错误位置和错误数值。具体地，以 10 个出错位为基础，利用欧基里德算法等方法算出用于确定错误位置多项式和错误数值多项式的各项系数，然后利用链查寻和调用方法等方法算出该错误位置多项式的根（错误位置）并利用错误数值多项式算出错误数值。

错误数据更新部件 57 根据由错误位置·错误数值运算部件 56 算出的错误位置，读出缓冲存储器 1 中的错误数据，计算读出的错误数据与由错误位置·错误数值运算部件 56 算出的错误数值的异或逻辑和，将得出的数值回写到缓冲存储器 2110 中的原位置上。就该纠错部件 5 而言，只有出错位计算电路（出错位生成部件 52~55）被并列化，而对于与其相连的电路（错误位置·错误数

值运算部件 56 及错误数据更新部件 57) 并没有被并列化, 关于出错位计算, 无论发生错误与否, 对 1 个数据都要进行 10 次积和运算, 但对于以后的计算处理, 只要根据在 1 个代码串中生成的 10 个出错位进行计算就可以了, 从而建设了运算次数。

主机接口部件 13 是用于在纠错部件 5 的处理结束之后, 从缓冲存储器 1 读出数据并将该数据输出到计算机等主机中的接口电路。

总线控制器 2 是在缓冲存储器 1 与 3 个处理块 12、5、13 中的任一个之间, 通过 8 位数据总线进行数据传输的控制电路, 包括用于排他地控制对缓冲存储器 1 进行访问的总线调配控制部件 22, 生成并输出用于对缓冲存储器 1 进行访问的地址的地址生成部件 21, 以及在磁盘接口部件 12、纠错部件 5 及主机接口部件 13 中的任一个与缓冲存储器 1 之间进行总线连接并通过 DMA (Direct Memory Access) 进行数据传输的 DMA 通道部件 23。

图 6 是总线控制器 2 的详细结构示意图。在本图中, 3 个 DMA 通道 23a~23b 是构成图 5 所示的 DMA 通道部件的 DMA 控制器。这个总线控制器 2 具有排他性地许可与缓冲存储器 1 相对应的 3 个 DMA 的功能。即, 总线调配控制部件 22 在接收到来自在磁盘接口部件 12、纠错部件 5 及主机接口部件 13 的访问缓冲存储器 1 的请求时, 根据预定的优先顺序和调度顺序, 只允许其中之一的请求。

具体地, 总线调配控制部件 22 通过启动第一 DMA 通道 23a, 对磁盘接口部件 12 输出允许进行数据传输的控制信号, 从地址生成部件 21 生成缓冲存储器 1 的写入地址, 进行将从磁盘接口部件 12 向缓冲存储器 1 的“数据输入”, 即从光盘等记录媒体读出的数据通过磁盘接口部件 12 存入缓冲存储器 1 中的数据传输。同样地, 总线调配控制部件 22 通过启动第二 DMA 通道 23b, 对纠错部件 5 输出允许进行数据传输的控制信号, 从地址生成部件 21 生成缓冲存储器 1 中的读出或写入地址, 进行“用于纠错的数据传输”, 即将存在缓冲存储器 1 中数据读出到纠错部件 5, 将纠正后的数据从纠错部件 5 回写到缓冲存储器 1 中的数据传输。而且, 总线调配控制部件 22 通过启动第三 DMA 通道 23c, 对主机接口部件 13 输出允许进行数据传输的控制信号, 从地址生成部件 21 生成缓冲存储器 1 中的读出地址, 进行将从缓冲存储器 1 向主机接口部件 13 的“数据输出”, 即存在纠错后的缓冲存储器 1 中的数据通过主机接口部件 13 输出到主机等中的数据传输。

在这3种DMA传输中,地址生成部件21对缓冲存储器1进行页面模式下的高速访问。具体地,在命中页的范围内,确定并原样输出对缓冲存储器1的行地址(RAS),使列地址(CAS)增加所需要的数据个数。

图7是数据分配部件51的详细结构示意图。数据分配部件51由用于将从总线控制器2传送来的数据穿过4个出错位生成部件52~55之一的选择器51a和用于根据来自总线控制器2的指示,控制选择器51a的分配的切换控制部件51b构成。切换控制部件51b通过内部的计数逻辑电路等,在数据被输入到选择器51a的同时将控制信号输出到选择器51a中,但是在进行行方向的纠错和列方向的纠错的情况下,如后所述,以不同的顺序控制选择器51a。

图8是1个出错位生成器52(53、54、55)的详细结构示意图。出错位生成部件52由一次最多存储8个输入的数据的FIFO(First-In First-Out)形式的队列缓冲器62和与该队列缓冲器62并联的10种出错位运算部件63~65构成。各出错位运算部件63~65是反复进行积和运算的电路,由用于存储1个积和值的寄存器68、用于读出存在寄存器68中的积和值,与固有的系数进行乘法运算的伽罗瓦域乘法器66以及对该乘法运算的结果和下次输入的1个代码进行加法运算的伽罗瓦域加法器67构成。

在从队列缓冲器62中读出最早输入的1个数据d时,该数据d被并列输入到10个出错位运算部件63中,根本进行1次积和运算 $d+S_n \times \alpha_n$,将该结果作为新的 S_n 存入寄存器68中。例如,由100字节的信息数据 $d_1 \sim d_{100}$ 和10个奇偶校验数据 $p_1 \sim p_{10}$ 构成的代码串被顺序输入到出错位生成部件52中,并经过队列缓冲器62被并列地提供给10个出错位运算部件63~65。此时,在第一出错位运算部件63中,首先在输入第一信息数据 d_1 时将该 d_1 原样保存在寄存器68中,接着在输入信息数据 d_2 时进行积和运算 $d_2+d_1 \times \alpha_0$,该结果被保存到寄存器68中。

这样每输入1个代码,就进行1次积和运算,到输入了最后的奇偶校验数据 p_{10} 时,保存在寄存器68中的数据就变成了 S_0 出错位。与此同时,在其它出错位运算部件64~65中也进行着同样的运算,因此在最后的奇偶校验数据被输入到这10个出错位运算部件63~65中之后,最后同时生成了从 S_0 到 S_9 这10种出错位。

下面,说明具有以上结构的纠错装置100的操作。

图9是装置100中的主要处理（数据输入、纠错、数据输出）中作为处理对象的数据的时序图。即，示出了在对利用图1所示的乘积码的代码串进行连续的多个数据块处理时的处理流向。图中的第n数据块与图1所示的1个乘积码的全部数据相对应。图中的“数据输入”、“纠错”及“数据输出”与上述总线控制器2的“数据输入”、“用于纠错的数据传输”及“数据输出”相对应。

首先，在第1周期进行第1数据块的数据输入，在第2周期并列进行第2数据块的数据输入和第1数据块的纠错，在第3周期并列进行第3数据块的数据输入、第2数据块的纠错和第数据块的数据输出。在第4周期以后，与第3周期一样，并列进行新数据块的数据输入，前一输入数据块的纠错和纠错结束后的数据块的数据输出。

这样，总线控制器2控制3种数据传输（从磁盘接口部件12向缓冲存储器1的数据传输、缓冲存储器1与纠错部件5之间的数据传输以及从缓冲存储器1向主机接口部件13的数据传输）以便于对各数据块的数据实施流水线处理。即，3种数据传输都以缓冲存储器1为传输目的地或传输源，但缓冲存储器1只有1个输入输出端口，因此总线控制器2通过在这3种数据传输中设计优先顺序，而只瞬时许可数据传输中的一种，通过将这3种数据传输按时间分割进行切换，从表面上控制了3种数据传输的并行。

图10示出了在进行C1序列的纠错时读出缓冲存储器1内的乘积码的顺序。总线调配控制部件22在启动第二DMA通道23b，并通知纠错部件5传送用于C1序列纠错的数据之后，按照预定的顺序将读出地址从地址生成部件21输出到缓冲存储器1，如下所述，将存在缓冲存储器1中的数据传输到纠错部件5中。

即，总线控制器2首先从缓冲存储器1中连续读出第1行4个字节的数据d1~d4并传输到纠错部件5中。接着连续读出并传输第2行4个字节的数据d5~d8。接着连续读出并传输第3行4个字节的数据d9~d12。接着连续读出并传输第4行4个字节的数据d13~d16。至此再次返回第1行，同样连续读取和传输4个字节的数据d17~d20。下面通过重复同样的读出和传输，在完成从第1行到第4行的全部数据的读出和传输时，对从第5行到第8行的4行数据重复同样的处理。如此对1个数据块的全部行重复并列读出和传输4行数据的处理。

总线控制器2以DMA模式进行对行方向上连续4个字节的数据读出和传输，以作为时间上不可分的处理连续进行。即，此期间禁止用于其它处理（数据输

入、数据输出)的对缓冲存储器 1 的访问。这是由于:在行方向上连续的数据通常被存储在缓冲存储器 1 的连续地址中,因而能够利用由 DRAM 页命中进行的高速访问。

图 11 示出了在按照图 10 所示的顺序从缓冲存储器 1 传输数据时,纠错部件 5 的数据分配器 51 的操作。切换控制部件 51b 在从总线控制器 2 接收开始对 C1 序列纠错的指示时,通过控制选择器 51a,将从总线控制器 2 传送来的数据如下进行分配。

即,数据分配部件 51 对最先传输的属于第 1 行的 4 个字节数据 d1~d4 进行分配以将其输入到出错位生成部件 52 中,对接着传输的属于第 2 行的 4 个字节数据 d5~d8 进行分配以将其输入到出错位生成部件 53 中,对接着传输的属于第 3 行的 4 个字节数据 d9~d12 进行分配以将其输入到出错位生成部件 54 中,对接着传输的属于第 4 行的 4 个字节数据 d13~d16 进行分配以将其输入到出错位生成部件 55 中。同样地,再将接着传输的属于第 1 行的数据 d17~d20 输入到出错位生成部件 52 中。这样,数据分配部件 51 在 C1 序列的纠错中,一边以 4 个字节为单位按照 4 个出错位生成部件 52~55 的顺序切换出错位生成部件,一边从缓冲存储器 1 经过总线控制器 2 输入反复以 4 个字节为单位传送的数据。

图 12 示出了 C1 序列的纠错中通过从缓冲存储器 1 向纠错器 5 的数据传输与 4 个出错位生成器 52~55 的积和运算时序。

从缓冲存储器 1 读出具有连续地址的 4 个字节的数据 d1~d4、d5~d8 等并将其传输到纠错部件 5 中分别需要时间 t1,以 4 个字节为单位的数据传输以时间 t2 为间隔进行重复。从完成以 4 个字节为单位的数据传输之后到开始以接着的 4 个字节为单位的数据传输,需要时间,这是因为要考虑到由于这 4 个字节分别属于不同的行而在读出新的 4 个字节数据时可能产生每次 DRAM 的页面错误。即,假设在访问 4 个字节中的第 1 字节时总线控制器 2 必须将新的行地址输出到缓冲存储器 1 中。

4 个出错位生成部件 52~55 分别进行第 1 行~第 4 行的出错位计算,但在 $4 \times t_2$ 的时间内对输入的 4 个字节的数据进行必要的 4 个积和运算。例如,执行第 1 行出错位计算的出错位生成部件 52 在 $4 \times t_2$ 的时间内对输入的 4 个字节数据 d1~d4 进行积和运算。从出错位生成部件 52 的角度出发,这是由于最先输

入的 4 个字节数据为 d1~d4, 接着输入的 4 个字节数据为 d17~d20, 它们的时间间隔为 $4 \times t_2$, 因而只要在这个时间内完成 4 个字节的积和运算就可以。然后, 为了维持这样的运算速度, 在各出错位生成部件 52~55 的前面所设计的队列缓冲器 62 中所存储的数据量最大为 4 个字节, 即使在瞬间也不能超过 4 个字节。

这样, 各出错位生成部件 52~55 只要以每 1 个字节 t_2 时间的速度就可以完成积和运算, 与数据传输速度 (4 字节/ t_2) 相比, 只需已 $1/4$ 的速度进行出错位计算即可。而且, 由于对不同的多个代码串, 一边以较小的节距 (4 个字节为单位) 切换代码串, 一边进行并列处理, 所以与处理每 1 代码串的方法相比, 在降低了每 1 代码串所要求的出错位计算速度的同时, 还缩小了队列缓冲器的大小。换言之, 这意味着能够使从缓冲存储器 1 到纠错部件 5 的实际数据传输速度 (整个纠错速度) 比原来更高速, 并且与简单地设置多个纠错电路相比, 能够以更小的电路规模进行 4 列纠错。

通过增加同一代码串中连续读出的数据数, 可以谋求利用 DRAM 页命中的数据传输的高速化。此时, 希望增加纠错的并列度。通过随增加纠错并列处理的并列度而增加存储器地址的连续传输字节数, 与每 1 代码串的处理负荷相对应的电路规模减轻了负担并实现了高效的纠错并列化。

现在参照图 13~图 15 说明在进行 C2 序列的纠错时本装置 100 的操作。

图 13 示出了在进行 C2 序列的纠错时读出缓冲存储器 1 内的乘积码的顺序。总线调配控制部件 22 在启动第二 DMA 通道 23b, 并通知纠错部件 5 传送用于 C2 序列纠错的数据之后, 按照预定的顺序将读出地址从地址生成部件 21 输出到缓冲存储器 1, 如下所述, 将存在缓冲存储器 1 中的数据传输到纠错部件 5 中。

即, 总线控制器 2 首先从缓冲存储器 1 中连续读出第 1 行 4 个字节的数据 d1~d4 并传输到纠错部件 5 中。接着连续读出并传输第 2 行 4 个字节的数据 d5~d8。以下, 同样地, 对第 1 列~第 4 列的全部数据反复从各行读出并传输 4 个字节的数据。由此, 完成 C2 序列的 4 个代码串的数据传输。同样地, 再对第 5 列~第 8 列的全部数据反复读出并传输 4 个字节的数据。由此, 对 1 个数据块的全部列重复进行并列读出并传输 4 列数据的处理。

与 C1 序列的数据传输相同, 总线控制器 2 以 DMA 模式进行对行方向上连续 4 个字节的数据读出和传输, 就象连续进行时间不可分的处理一样。即, 此

期间通过禁止用于其它处理（数据输入、数据输出）的对缓冲存储器 1 的访问而利用了根据 DRAM 页命中的高速访问。

图 14 示出了在按照图 13 所示的顺序从缓冲存储器 1 传输数据时，纠错部件 5 的数据分配器 51 的操作。切换控制部件 51b 在从总线控制器 2 接收开始对 C2 序列纠错的指示时，通过控制选择器 51a，将从总线控制器 2 传送来的数据如下进行分配。

即，数据分配部件 51 对最先传输的属于第 1 行的 4 个字节数据 d1~d4 进行分配以将各数据按照顺序输入到出错位生成部件 52~55 中。具体地，将数据 d1 送到出错位生成部件 52 中，将数据 d2 送到出错位生成部件 53 中，将数据 d3 送到出错位生成部件 54 中，将数据 d4 送到出错位生成部件 55 中。对接着传输的属于第 2 行的数据 d5~d8 同样分配以将各数据按照顺序输入到出错位生成部件 52~55 中。这样，数据分配部件 51 在 C2 序列的纠错中，重复将以 4 个字节为单位反复传送的数据的各个字节分配到 4 个出错位生成部件 52~55 中的处理。

由此，属于第 1 列的数据 d1、d5、...被输入到出错位生成部件 52 中，属于第 2 列的数据 d2、d6、...被输入到出错位生成部件 53 中，属于第 3 列的数据 d3、d7、...被输入到出错位生成部件 54 中，属于第 4 列的数据 d4、d8、...被输入到出错位生成部件 55 中。同样，属于第 5 列的数据被输入到出错位生成部件 52 中，属于第 6 列的数据被输入到出错位生成部件 53 中，属于第 7 列的数据被输入到出错位生成部件 54 中，属于第 8 列的数据被输入到出错位生成部件 55 中。

图 15 示出了 C2 序列的纠错中通过从缓冲存储器 1 向纠错器 5 的数据传输与 4 个出错位生成器 52~55 的积和运算时序。

从缓冲存储器 1 读出具有连续地址的 4 个字节的数据 d1~d4、d5~d8 等并将其传输到纠错部件 5 中分别需要时间 t1，在时间 t2 的间隔内重复以 4 个字节为单位的数据传输，这与图 12 中所示的 C1 情况相同。

4 个出错位生成部件 52~55 分别进行第 1 列~第 4 列的出错位计算，但在 t2 的时间内进行关于 1 个字节的数据的 1 次积和运算。例如，执行第 1 行的出错位计算的出错位生成部件 52 在 t2 时间内对最先输入的 1 个字节数据 d1 进行积和运算，在 t2 时间内对接着输入的 1 个字节数据 d5 进行积和运算。从出

错位生成部件 52 的角度出发,这是由于在时间 t_2 中,1 个字节的数据被输入。然后,为了维持这样的运算速度,在各出错位生成部件 52~55 的前面所设计的队列缓冲器 62 中所存储的数据量最大为 1 个字节,即使在瞬间也不能超过 1 个字节。

这样,各出错位生成部件 52~55 只要以每 1 个字节 t_2 时间的速度就可以完成积和运算,可以理解,这与进行 C1 序列的纠错时的处理速度相同。而且,在此 C2 序列的纠错中,向各个出错位生成部件 52~55 的数据输入与丛发进行的 C1 序列不同,是均匀进行的(通常以 1 字节/ t_2 的速度输入数据),所以没有必要在各出错位生成部件 52~55 的前面设置队列缓冲器。

如上所述,在第一实施例中,由于纠错部件 5 中的并列处理的并列度与在从缓冲存储器 1 读出具有连续地址的数据(乘积码的同一行中的连续数据)并将其传输到纠错部件 5 时的数据量相等,所以在 C1 序列和 C2 序列两个方向上的纠错中,从缓冲存储器 1 读出数据的速度与通过在各出错位生成部件进行积和运算的速度之间的关系同等,从而缩小了在各出错位生成部件前面所需要的队列缓冲器的大小。由此,对于 C1 序列和 C2 序列两个方向上的纠错,用于充分发挥出错位生成部件能力的总线控制器 2 的控制顺序也被共享,实现了能够以较小的电路规模进行高速纠错的纠错装置。

即,能够更平衡地实施 C1 序列的纠错和 C2 序列的纠错,削减了纠错装置中所需要的电路的冗余度。

虽然图 10 及图 13 中所示的读出顺序是以行方向上的代码串的字节数是 4 的倍数为前提的,但是即使在与之不同的情况下,本装置 100 也能够进行正确的纠错。例如,在行方向上的代码串的长度为其字节数除 4 还余 2 的情况下,只要将缓冲存储器 1 内的乘积码的读出顺序作如下变动即可。

图 16 示出了在行方向上的代码串的长度为其字节数除 4 还余 2 的情况下,进行 C1 序列的纠错时缓冲存储器 1 内的乘积码的读出顺序。总线控制器 2 针对 4 行代码串,对除去各代码串最后的 2 个字节以外的部分按照前面所述的顺序重复连续 4 个字节的读出和传输操作,而对最后的 2 个字节进行读出和传输各代码串的 2 个字节的操作。与之相伴,数据分配部件 51 也进行同样的操作,即,将反复以 4 个字节为单位从缓冲存储器 1 中传送来数据以 4 个字节为单位分别分配到 4 个出错位生成部件 52~55 中,将以 2 个字节为单位被传送来的

数据以 2 个字节为单位分别分配到 4 个出错位生成部件 52~55 中即可。

同样, 图 17 示出了在列方向上的代码串的长度为其字节数除 4 还余 2 的情况下, 进行 C2 序列的纠错时缓冲存储器 1 内的乘积码的读出顺序。总线控制器 2 对除去最后 2 列的代码串, 从最先输入的 4 列开始, 重复 4 个字节的数据读出和传输操作, 而对最后 2 列代码串, 反复读出和传输 2 个字节。与之相伴, 数据分配部件 51 也进行同样的操作, 即, 将反复以 4 个字节为单位从缓冲存储器 1 中传送来数据以 4 个字节为单位分别分配到 4 个出错位生成部件 52~55 中, 将以 2 个字节为单位被传送来的数据以 2 个字节为单位分别分配到 4 个出错位生成部件 52~55 中即可。

象这样变更总线控制器 2 用的读出顺序以及数据分配部件 51 用的分配顺序可以通过具体地变更总线控制器 2 中总线调配控制部件 22 用的控制顺序和数据分配部件 51 中切换控制部件 b 用的控制顺序容易地实现。

第一实施例是用于并列执行对应于 4 个代码串的出错位计算的纠错装置, 但是本发明并不仅限于这样的并列度, 例如, 2 列和 8 列的纠错装置也是可以的。即, 在第一实施例中, 纠错部件 5 具有 4 个出错位生成部件 52, 但是也可以根据需要设置为 2 个和 8 个。

图 18、图 19 和图 20 用于说明在以 2 列进行纠错的纠错装置中, C1 序列的纠错操作, 分别示出了数据从缓冲存储器中读出的顺序、以该顺序输入数据时的纠错部件中的数据分配部件的操作以及该情况下进行数据传输和出错位计算的时序。

同样, 图 21、图 22 和图 23 用于说明在以 2 列进行纠错的纠错装置中, C2 序列的纠错操作, 分别示出了数据从缓冲存储器中读出的顺序、以该顺序输入数据时的纠错部件中的数据分配部件的操作以及该情况下进行数据传输和出错位计算的时序。

在第一实施例中, 对读取记录在光盘等记录媒体中的数据、从读出的数据串中查找错误、进行纠正以及将纠正后的数据传输到主计算机中的数据流向进行了说明, 但另一方面, 本发明也能够用于从主计算机输出数据、在该数据中附加用于纠错的奇偶校验码以及写入光盘等记录媒体中的数据流向。本发明是这样一种技术: 对于暂时存储在设置于记录媒体和主计算机之间的数据总线中途的缓冲存储器中的乘积码, 通过有效地读出构成该乘积码的全部代码串并将

其传输到第三装置（除记录媒体和主计算机之外的处理装置），可以在第三装置中通过高速且小规模的对 C1 序列和 C2 序列进行代码处理。

具体地，可以进行如下变更：将纠错部件 5 中的 4 个出错位生成部件 52~55 和错误位置·错误数值运算部件 56 置换为奇偶校验数据生成电路，将错误数据更新部件 57 置换为奇偶校验数据写入电路。从而实现了能够高速执行在向记录媒体写入数据时附加纠错码和在从记录媒体中读出数据时纠正错误数据的所谓“纠错伴随处理”的纠错装置。

在第一实施例中，总线控制器 2 通过 1 次 DMA 传输进行从缓冲存储器 1 向纠错部件 5 的 4 字节数据的传输，在这次 DMA 传输与下次 DMA 传输期间允许除对传输缓冲存储器 1 进行访问之外的数据传输，但是也可以通过 1 次 DMA 传输进行 16 字节数据的传输。由此，在 C1 序列的纠错中，通过 1 次 DMA 传输可以将图 10 中所示的 16 字节数据 d1~d16 从缓冲存储器 1 传输到纠错部件 5 中，在此期间禁止插入缓冲存储器 1 中用于访问的其它数据的传输，因而 16 字节数据的传输时间，即分别向 4 个出错位生成部件 52~55 中的每一个提供 4 个字节的数据所需要的时间被大致固定，在显著降低了出错位计算的处理速度的情况下，避免了向纠错部件 5 进行的数据传输成为瓶颈。

在第一实施例中，纠错装置 100 内部的数据总线宽度为 8 位，但也可以是 32 位数据总线。此时，通过 1 次存储器访问从缓冲存储器 1 中读出 4 字节数据并传输到纠错部件 5 中。然后，数据分配部件 51 在进行 C1 序列的纠错时，将 32 位数据按照每 8 位进行分割，分别分配到 4 个出错位生成部件 52~55 中即可。

在第一实施例中，在纠错部件 5 中只有出错位计算被并列执行，但也可以对与其连接的错误位置·错误数值运算部件 56 和错误数据更新部件 57 进行并列化。由此，即使在出错位计算中发生大量错误的情况下，也能够确保一定的纠正处理。

（第二实施例）

第二实施例涉及这样的高速光盘读取装置：分离错误检测和纠错，在将数据存入缓冲存储器的同时并行执行错误检测，通过将该结果反映到纠错中，避免了从缓冲存储器向纠错部件传输没有发生错误的数据。

图 24 是本发明第二实施例中的光盘读取装置 2000 的结构示意图。光盘读

取装置 2000 是用于读取记录在光盘 2010 中的数据,并在对读出的数据进行纠错后将该数据传输到主计算机 2070 中的装置,由光拾取器 2020、放大器 2030、前端处理机 2040、光盘控制部件 2100、旋转电动机 2050、伺服控制器 2060 和系统控制部件 2080 构成。

系统控制部件 2080 是用于通过光盘控制部件 2100 接收来自主计算机 2070 的数据请求的命令,解释该命令,控制伺服控制器 2060 及光盘控制部件 2100 并实现与数据请求相应的功能的微处理器。

旋转电动机 2050 用于旋转光盘 2010,伺服控制器 2060 用于以来自前端处理机 2040 的信息为基础接收系统控制部件 2080 的指示,并控制旋转电动机 2050 的旋转及光拾取器 2020 的镜头位置。

光盘 2010 为 DVD-ROM,光拾取器 2020 用于通过镜头光线的反射光读取记录在光盘 2010 中的数据并将其转换为电信号,放大器 2030 用于放大光拾取器 2020 的输出信号并将其输出到前端处理机 2040 中。

前端处理机 2040 用于根据输入的信号,在伺服控制器 2060、旋转电动机 2050 及光拾取器 2020 中进行反馈控制,将稳定的信号输出到光盘控制部件 2100 中,包括均衡器、AGC (自动增益控制电路)、PLL (Phase Locked Loop) 等。而且前端处理机 2040 输出到光盘控制部件 2100 中的信号为调制数据。

光盘控制部件 2100 将来自主计算机 2070 的数据请求命令通知给系统控制部件 2080,根据系统控制部件 2080 的控制,对从前端处理机 2040 输入的信号进行解调,取出所需要的数据,进行纠错并将数据输出到主计算机 2070 中。

在图 24 中,粗箭头表示记录在光盘 2010 中的数据到达主计算机 2070 中的流程。即,记录在光盘 2010 中的数据经过光拾取器 2020、放大器 2030、前端处理机 2040 以及光盘控制部件 2100 而到达主计算机 2070。而且,记录在光盘 2010 中的数据结构与第一实施例中的数据结构相同,采用图 1 所示的乘积码并采用 Reed-Solomon 码为纠错码。

图 25 是光盘控制器 2100 的详细结构示意图。光盘控制部件 2100 是用于对从前端处理机 2040 传送来的信号进行解调并纠错,然后将其送到主计算机 2070 中的电路,包括缓冲存储器 2110、磁盘接口部件 2120、纠错部件 2130、主机接口部件 2140、第一传输部件 2150、第二传输部件 2160、第三传输部件 2170、总线控制器 2180、错误检测部件 2190 和错误代码串存储部件 2200。图

25 中的粗箭头表示在向缓冲存储器 2110 的由 DMA 传输执行的数据写入以及从缓冲存储器 2110 的由 DMA 传输执行的数据读出中的数据通路。

缓冲存储器 2110 是利用二维地址（行地址和列地址）具有特定存储区域的 DRAM 等半导体存储器，具有暂时存储多个图 1 所示的乘积码的存储容量。

磁盘接口部件 2120 在对从前端处理机 2040 输入的信号进行解调后，将得到的数据输出到第一传输部件 2150 中以将数据存入缓冲存储器 2110 中。在将从磁盘接口部件 2120 输出的数据通过第一传输部件 2150 传输到缓冲存储器 2110 中的同时，将该数据并行输入到错误检测部件 2190 中。

对于经过第一传输部件 2150 从磁盘接口部件 2120 传送来的数据，错误检测部件 2190 判断 C1 序列中的每个代码串中是否存在错误，在检测出错误的情况下，将该代码串的编号存入由 RAM 构成的错误代码串存储部件 2200 中。该错误检测部件 2190 还计算出每 1 列代码串中一定个数的出错位，判断这些出错位中是否存在非零数，如果存在非零数，则断定有错误发生。具体地，错误检测部件 2190 由第一实施例中的纠错部件 5 所具有的 4 个出错位生成部件 52～55 之一（图 8 所示的电路）和用于判断所生成的出错位是否全部为零的与门电路构成。

纠错部件 2130 以从磁盘接口部件 2120 传输并存入缓冲存储器 2110 的乘积码中产生了错误的数据为对象，进行 C1 序列及 C2 序列的纠错。具体地，纠错部件 2130 参照存在错误代码串存储部件 2200 中的编号，只以发生了错误的 C1 序列的代码串或发生了错误的代码块的 C2 序列代码串为对象，纠正缓冲存储器 2110 中的错误数据。

主机接口部件 2140 是用于通过第三传输部件 2170，从缓冲存储器 2110 中读出完成了由纠错部件 2130 所执行的处理的数据并将其输出到主计算机 2070 中的接口电路。

第一传输部件 2150 是 DMA 控制器，用于接收总线控制器 2180 对来自磁盘接口部件 2120 的传输请求的许可通知，同时并列进行从磁盘接口部件 2120 向缓冲存储器 2110 的 DMA 数据传输和向错误检测部件 2190 的同一数据的传输。

第二传输部件 2160 是 DMA 控制器，用于接收总线控制器 2180 对来自纠错部件 2130 的传输请求的许可通知，进行从缓冲存储器 2110 向纠错部件 2130 的 DMA 数据传输。

第三传输部件 2170 是 DMA 控制器，用于接收总线控制器 2180 对来自主机接口部件 2140 的传输请求的许可通知，进行从缓冲存储器 2110 向主机接口部件 2140 的 DMA 数据传输。

总线控制器 2180 按照预定优先顺序，控制这 3 个传输部件 2150、2160 和 2170，以便于排他性地执行分别由第一传输部件 2150、第二传输部件 2160 和第三传输部件 2170 进行的 DMA 传输并控制在重写缓冲存储器 2110 中的错误数据时由纠错部件 2130 对缓冲存储器 2110 的普通（非 DMA）访问。

图 26 是总线控制器 2180 的详细结构示意图。总线控制器 2180 由生成并输出用于对缓冲存储器 1 进行访问的地址的地址生成部件 2181、将第一传输部件 2150、第二传输部件 2160、第三传输部件 2170 以及纠错部件 2130 中的一个与缓冲存储器 2110 进行总线连接的总线切换部件 2183 以及按照预定优先顺序，对这 3 个传输部件 2150、2160、2170 及总线切换部件 2183 进行控制以便于排他地执行这 3 个传输部件 2150、2160、2170 的 DMA 传输及纠错部件 2130 对缓冲存储器 2110 的访问的总线调配控制部件 2182 构成。

地址生成部件 21 在这 3 个 DMA 传输中，对缓冲存储器 1 进行页面模式下的高速访问。具体地，在命中页的范围内，确定并原样输出对缓冲存储器 1 的行地址（RAS），仅使列地址（CAS）增加所需要的数据个数并输出。

图 27 是表示由总线调配控制部件 2182 进行的 3 个排他的 DMA 传输的调度示例，即由第一传输部件 2150 进行的从磁盘接口部件 2120 向缓冲存储器 2110 的 DMA 传输（DMA#1）、由第二传输部件 2160 进行的从缓冲存储器 2110 向纠错部件 2130 的 DMA 传输（DMA#2）以及由第三传输部件 2170 进行的从缓冲存储器 2110 向主机接口部件 2140 的 DMA 传输（DMA#3）的时序图。在本图中，横轴表示经过的时间，三角形印记表示对个 DMA 传输的请求的发生，划有剖面线的四边形表示 DMA 传输在执行中。

总线调配控制部件 2182 在最高的优先级下允许执行 DMA#1，在最低的优先级下允许执行 DMA#2。即，总线调配控制部件 2182 在没有任何一个 DMA 传输处于执行当中的情况下，不管与各 DMA 相对应的请求的产生顺序如何，只要产生了与 DMA#1 相对应的请求，就最优先地执行 DMA#1，如果没有产生与 DMA#1 相对应的请求，就执行 DMA#3，只有在与 DMA#1 相对应的请求和与 DMA#3 相对应的请求都没有产生时从执行 DMA#2。但是，在连续产生与 DMA#1 相对应的请求

并且还产生了其他的与 DMA#2 或 DMA#3 相对应的请求时,不连续执行 DMA#1,而是执行其他的 DMA#2 或 DMA#3。

对于纠错部件 2130 在重写缓冲存储器 2110 中的错误数据时所进行的对缓冲存储器 2110 的普通访问,总线调配控制部件 2182 执行与 DMA#3 相同的处理。即,在与 DMA#1 相对应的请求和与 DMA#3 相对应的请求都没有产生时,允许纠错部件 2130 对缓冲存储器 2110 所进行的普通访问。

图 28 是示出了与纠错部件 2130 的详细结构相关联的其他构成部件的电路图。纠错部件 2130 是用于针对构成图 1 所示的乘积码的 C1 序列及 C2 序列中预先检测出错误的代码串,执行每 1 行或 1 列代码串纠错的电路,由 C1 代码串传输请求部件 2135、出错位计算部件 2131、欧基里德计算部件 2132、链计算部件 2133 以及错误数据更新部件 2134 构成。

C1 代码串传输请求部件 2135 向第二传输部件 2160 提出请求以便从缓冲存储器 2110 中读出包含了存在错误代码串存储部件 2200 中的错误的 C1 序列的代码串。

出错位计算部件 2131 是用于在通过总线控制器 2180、第二传输部件 2160 而从缓冲存储器中取得了包含来自 C1 代码串传输请求部件 2135 的请求中所具有的错误的 C1 代码串之后,计算该代码串的出错位的电路,其计算结果被提供给欧基里德计算部件 2132。

另外,此出错位计算部件 2131 是用于计算出每 1 行或 1 列代码串中的出错位的电路,与第一实施例中的纠错部件 5 所具有的 4 个出错位生成部件 52~55 之一(图 8 所示的电路)相同。即,在乘积码的大小为 $k_1=k_2=100$ 字节、 $m_1=m_2=10$ 字节的情况下,出错位计算部件 2131 由用于对输入的数据进行暂时缓冲存储的队列缓冲器和 10 个用于分别计算出 10 个出错位的积和运算电路组成,各积和运算电路由用于存储 1 个积和值的寄存器、用于读出存储在该寄存器中的积和值并将其与固有的系数相乘的伽罗瓦域乘法器以及用于将相乘的结果与下次输入的 1 个数据相加的伽罗瓦域加法器组成。

欧基里德计算部件 2132 是用于根据从出错位计算部件 2131 中传送来的出错位,利用欧基里德算法,计算出用于确定错误位置多项式和错误数值多项式各项系数的电路,计算出的系数被提供给链计算部件 2133。

链计算部件 2133 是利用来自欧基里德计算部件 2132 的系数,通过链查寻

和调用法等方法算出所确定的错误位置多项式的根的电路，将其运算结果与来自欧基里德计算部件 2132 的用于确定错误数值多项式的系数一起提供给错误数据更新部件 2134。

错误数据更新部件 2134 在利用来自链计算部件 2133 的错误位置和错误数值多项式，求出位于该错误位置上的错误数值之后，通过总线控制器 2180 读取缓冲存储器 2110 内该错误位置上的信息数据或奇偶校验数据，计算它们的异或逻辑和，将得出的值回写到缓冲存储器 2110 内的原位置上。

另外，由第二传输部件 2160 进行的从缓冲存储器 2110 向出错位计算部件 2131 的数据传输是通过由 DMA 传输进行的存储器访问而进行的，但是缓冲存储器 2110 与错误数据更新部件 2134 之间的数据传输是通过由总线控制器 2180 进行的普通（非 DMA）存储器访问而进行的。这是由于通过纠错而成为重写对象的数据不受存储在缓冲存储器 2110 内的连续地址上的数据限制。而且，构成纠错部件 2130 的 4 个电路 2131~2134 在分别进行同步并列处理，即进行流水线方式下的顺序处理。

图 29 是示出了在连续处理多个数据块的数据的情况下，光盘控制器 2100 内的处理过程的时序图。在本图中，第 1 数据块到第 5 数据块是分别存储在缓冲存储器 2110 中的 1 个数据块的数据。而且，“缓冲”表示由第一传输部件 2150 进行的从磁盘接口部件 2120 向缓冲存储器 2110 的数据传输，“错误检测”表示由第一传输部件 2150 进行的从磁盘接口部件 2120 向错误检测部件 2190 的数据传输和在错误检测部件 2190 中进行的错误检测，“纠错”表示由第二传输部件 2160 进行的从缓冲存储器 2110 向纠错部件 2130 的数据传输和在纠错部件 2130 中进行的纠错，“主机传输”表示由第三传输部件 2170 进行的从缓冲存储器 2110 向主机接口部件 2140 的数据传输。

如本图所示，将与同一数据块相对应的缓冲及错误检测、对完成了缓冲及错误检测的数据块的纠错以及对完成了纠错的数据块的主机传输作为流水线顺序且并列地执行。例如，在周期 t_5 ，在从磁盘接口部件向缓冲存储器 2110 传输和存储第 5 数据块的同时，在错误检测部件 2190 中进行错误检测，从缓冲存储器 2110 向纠错部件 2130 传输第 4 数据块并进行纠错，从缓冲存储器 2110 向主机接口部件 2140 传输第 3 数据块。

另外，在本图中，并列进行 3 种用于访问缓冲存储器 2110 的数据传输，

但这是从数据块这个大方面的数据传输观点上出发而表现出来的，从构成数据块的各数据的角度考虑（瞬时地），只能执行其中一种数据传输。这样的排他性数据传输及上述流水线控制，如图 27 中的时序图所示，是通过按照预定的优先顺序，将由 3 个传输部件 2150、2160、2170 进行的数据传输以较小的时间单位进行切换并许可而实现的。

而且，在本图中，不对第 3 数据块进行纠错就是在错误检测过程中，这个数据块中没有发现任何错误。即，由于在错误检测部件 2190 中没有对第 3 数据块检测出任何错误，所以接收该通知的纠错部件 2130 不进行与对该数据块的纠错相关的一切处理。即，通过纠错部件 2130 将上述意思通知总线控制器 2180，对属于第 3 数据块的全部代码串，不进行由第二传输部件 2160 执行的从缓冲存储器 2110 向纠错部件 2130 的数据传输。

这样，在并列进行对同一数据块的缓冲和错误检测，并在该错误检测中没有检测到错误的情况下，由于通知了纠错部件 2130，所以能够避免纠错部件 2130 对不需要纠错的数据进行从缓冲存储器 2110 读出的所谓无效数据传输。即，若在现有技术中，对于从光盘读出的全部数据，进行存入缓冲存储器、从缓冲存储器中读出以进行错误检测・纠正、从缓冲存储器中读出以将该数据输出到主计算机的所谓 3 种数据传输，但在本第二实施例，对于没有发生错误的数数据，只进行存入缓冲存储器和从缓冲存储器中读出以将数据输出到主计算机的所谓 2 种数据传输。

图 30 是图 29 所示的时序图的更详细的示图，在与错误检测和纠错这 2 个处理相关的过程中分别示出了 C1 序列及 C2 序列的处理。其中，在错误检测中，仅对 C1 序列进行，而在纠错中，先对 C1 序列进行 1 次纠错，接着对 C2 序列进行 1 次纠错。

具体地，在周期 T1，错误检测部件 2190 对于通过第一传输部件 2150 从磁盘接口部件 2120 传送来的第 1 数据块，检查 C1 序列的代码串中是否发生错误。如果检测出错误的发生，错误检测部件 2190 就将发生了错误的代码串的编号记录到代码串存储部件 2200 中。

其中，错误检测部件 2190 之所以对 C1 序列进行错误检测，是因为从磁盘接口部件 2120 向缓冲存储器 2110 的数据传输是按照从第 1 行到第 n2 行，反复在行方向上扫描乘积码的顺序而进行的，因而使 C1 序列的错误检测变得很

容易。而之所以对 C2 序列不进行错误检测，是因为对以前的 C1 序列纠错的结果如果存在 1 次纠错都不能进行的代码串，就需要执行 C2 序列的纠错，因此根本没有必要执行 C2 序列的错误检测。

接着在周期 T2，纠错部件 2130 首先从缓冲存储器 2110 中读出 C1 序列的代码串并对其进行纠错。但是，纠错部件 2130 通过参照存储在错误代码串存储部件 2200 中的编号，从缓冲存储器 2110 中只读出检测出错误的 C1 序列的代码串并对其进行纠错。然后，在完成了对全部检测出错误的 C1 序列的代码串的纠错之后，在该 C1 序列中存在没有进行 1 次纠错的代码串的情况下，接着从缓冲存储器 2110 中顺序读出构成该数据块的全部 C2 序列的代码串并对其进行纠错。关于 C2 序列，对构成数据块的全部代码串进行纠错是由于错误检测部件 2190 只对 C1 序列进行了错误检测，而不确定 C2 序列的代码串中是否有错误所致。

图 31 是示出了在进行 C1 序列的纠错时纠错部件 2130 内的行单位处理过程的时序图。其中，示出了在通过错误检测部件 2190 在第 1 行、第 4 行、第 6 行及第 11 行中检测错误时的由出错位计算部件 2131、欧基里德计算部件 2132、链计算部件 2133 及错误数据更新部件 2134 执行的并列处理。

由于错误检测部件 2190 在第 2 行、第 3 行、第 5 行、第 8~10 行的代码串中没有检测出错误，因此不将其从缓冲存储器 2110 读到出错位计算部件 2131 中，从而，不作为出错位计算、欧基里德计算、链计算及错误数据更新的对象。

图 32 示出了在纠正图 31 所示的代码串时，对存储在缓冲存储器 2110 中的乘积码的各数据的访问顺序。其中，示出了首先执行对第 1 行的代码串的纠错，然后跳过第 2 行、第 3 行，对第 4 行的代码串执行纠错，然后跳过第 5 行对第 6 行的代码串执行纠错，接着对第 7 行的代码串执行纠错的流程。与图 2 所示的逐行扫描不同的是只对检测出错误的行进行扫描。

如上所述，在此第二实施例中，纠错部件 2130 对于 C1 序列的纠错，通过参照错误代码串存储部件 2200，从缓冲存储器 2110 中只读出检测出错误的代码串，即需要进行纠错的代码串并对其进行纠错，避免了无效地从缓冲存储器 2110 中读出不需要纠错的代码串。

由此，对于没有检测出错误的 C1 序列的代码串，不进行所谓的 3 种数据传输：将从光盘读出的全部数据存入缓冲存储器、从缓冲存储器读出数据以用

于错误检测•纠错以及从缓冲存储器中读出数据以将数据传送到主计算机中，而是只进行 2 种数据传输：存入缓冲存储器和从缓冲存储器读出数据以将其传送到主计算机。即，减少了对缓冲存储器 2110 的访问次数，以此在提高整个光盘控制部件 2100 的处理速度的同时谋求降低功耗，实现更高速的低功耗光盘读取装置 2000。

此外，在本第二实施例中，如图 32 所示，纠错部件 2130 反复进行逐行或逐列纠错，但也可以象第一实施例一样，进行每隔 4 行和 4 列等多代码串的并列纠错。

图 33 示出了在以 4 个代码串并行对图 31 所示的 C1 序列纠错时，从磁盘接口 2120 读出数据的顺序。这种读出顺序与第一实施例中图 10 所示的顺序基本相同，但在图 33 中，与图 10 所示的不同之处在于只以检测出错误的行（第 1 行、第 4 行、第 6 行、第 7 行...）为对象，每隔 4 行代码串并列读出。在代码串的长度不能被 4 整除的情况下，如图 34 所示，先分别对检测出错误的 4 列代码串（第 1 行、第 4 行、第 6 行、第 7 行）反复进行 4 个字节的吃素，最后分别对该 4 列代码串进行 2 个字节的传输即可。

第二实施例中象这样的由纠错部件 2130 对 4 个代码串并列进行纠错的变形例只要象第一实施例中的纠错部件 5 一样，在纠错部件 2130 中设置 4 个出错位计算部件和用于将来自缓冲存储器 2110 的数据分配到 4 个出错位计算部件中的数据分配部件即可。由此，就能够得到由第一实施例中的并列化而产生的纠错高速化和由第二实施例中的预先进行错误检测而产生的纠错高速化这两方面的效果。

而且，在第二实施例中，在纠错部件 2130 内设置了 C1 代码串传输请求部件 2135，但是也可以将 C1 代码串传输请求部件 2135 或者设置在错误检测部件 2190 内，或者与错误检测部件 2190 和纠错部件 2130 一起构成独立的电路。

而且，错误检测部件 2190 只通知检测出错误的代码串的编号，但是也可以同时通知由该检测得到的出错位。然后，纠错部件 2130 也可以针对检测出错误的代码串，从错误检测部件 2190 接收出错位，进行欧基里德计算和链计算。由此，能够避免在错误检测部件 2190 和纠错部件 2130 中进行重复的出错位计算。

同样地，在错误检测部件 2190 中，也可以不仅进行错误检测，还接着进

行欧基里德计算和链计算。在这种情况下,纠错部件 2130 对于检测出错误的代码串,从错误检测部件 2190 中接收错误位置和错误数值并对缓冲存储器 2110 内的错误数据进行纠正即可。这样,通过在能够确保处理时间的范围内,在错误检测部件 2190 中执行用于进行 C1 序列的纠错的出错位计算、欧基里德计算及链计算,可以使执行计算与从磁盘接口部件 2120 向缓冲存储器 2110 的数据传输同时进行,并缩短纠错部件 2130 进行纠错所需要的时间。

(第三实施例)

第三实施例涉及这样的高速光盘读取装置:通过在将数据存入缓冲存储器中时并行执行 C1 序列的纠错,同时在该 C1 序列的纠错中,从缓冲存储器中只读出发生了错误的数据块并执行 C2 序列的纠错,从而削减了对缓冲存储器的访问次数。

图 35 是本发明第三实施例中的光盘读取装置 3000 的结构示意图。光盘读取装置 3000 是用于读取记录在光盘 2010 中的数据,并在对读出的数据进行纠错后将该数据传输到主计算机 2070 中的装置,由光拾取器 2020、放大器 2030、前端处理机 2040、光盘控制部件 3100、旋转电动机 2050、伺服控制器 2060 和系统控制部件 2080 构成。

此外,光盘读取装置 3000 与第二实施例中的光盘读取装置 2000 相比,只在光盘控制部件 3100 方面有所不同,因此对于其它的构成部件采用与图 24 所示的符号相同的符号,在此省略了对这些相同部件的说明。

图 36 是光盘控制器 3100 的详细结构示意图。光盘控制部件 3100 是用于将来自前端处理机 2040 的信号解调并纠错后传送到主计算机 2070 的电路,包括缓冲存储器 3110、磁盘接口部件 3120、纠错部件 3130、主机接口部件 3140、第一传输部件 3150、第二传输部件 3160、第三传输部件 3170、总线控制器 3180 及数据选择部件 3190。图 36 中的粗箭头表示在向缓冲存储器 3110 的由 DMA 传输执行的数据写入以及从缓冲存储器 3110 的由 DMA 传输执行的数据读出中的数据通路。

缓冲存储器 3110 是利用二维地址(行地址和列地址)具有特定存储区域的 DRAM 等半导体存储器,具有暂时存储多个图 1 所示的乘积码的存储容量。

磁盘接口部件 3120 在对从前端处理机 2040 输入的信号进行解调后,将得到的数据输出到第一传输部件 3150 中以将数据存入缓冲存储器 3110 中。在将

从磁盘接口部件 3120 输出的数据通过第一传输部件 3150 传输到缓冲存储器 3110 中的同时, 将该数据并行输入到数据选择部件 3190 中。

数据选择部件 3190 是用于选择作为纠错部件 3130 的纠错对象的代码串的传输源的选择器。在通过第一传输部件 3150 从磁盘接口部件 3120 向缓冲存储器 3110 进行数据传输的情况下, 该数据也被同时并行传送到数据选择部件 3190 中, 因此对该数据进行选择并通过数据选择部件而到达纠错部件 3130。另一方面, 在通过第二传输部件 3160 从缓冲存储器 3110 传送 C2 序列的代码串的情况下, 对该数据进行选择并通过数据选择部件而到达纠错部件 3130。另外, 数据选择部件 3190 还根据来自第一传输部件 3150 和第二传输部件 3160 的指示, 选择这些数据传输源中的一个。

纠错部件 3130 一边区分来自数据选择部件 3190 的 2 种 (C1 序列和 C2 序列) 代码串, 一边进行纠错。具体地, 纠错部件 3130 在数据选择部件 3190 选择并通过来自第一传输部件 3150 的数据 (C1 序列的代码串) 的情况下, 执行 C1 序列的纠错, 另一方面, 在数据选择部件 3190 选择并通过来自第二传输部件 3160 的数据 (C2 序列的代码串) 的情况下, 执行 C2 序列的纠错。

主机接口部件 3140 是用于通过第三传输部件 3170, 从缓冲存储器 3110 中读出完成了由纠错部件 3130 所执行的处理的数据并将其输出到主计算机 2070 中的接口电路。

第一传输部件 3150 同时并列执行从磁盘接口部件 3120 向缓冲存储器 3110 的 DMA 数据传输和对同一数据的向数据选择部件 3190 的 DMA 数据传输, 第二传输部件 3160 执行从缓冲存储器 3110 向纠错部件 3130 的 DMA 数据传输, 并且第三传输部件 3170 执行从缓冲存储器 3110 向主机接口部件 3140 的 DMA 数据传输。

总线控制器 3180 按照预定的优先顺序, 控制 3 个传输部件 3150、3160、3170, 以便于排他地执行分别由第一传输部件 3150、第二传输部件 3160 和第三传输部件 3170 执行的 DMA 传输, 并控制在重写缓冲存储器 3110 中的成为时间时由纠错部件 3130 执行的对缓冲存储器 3110 的普通 (非 DMA) 访问。另外, 总线控制器 3180 的详细结构和由总线控制器 3180 执行的对缓冲存储器 3110 的访问的排他控制分别与第二实施例中的图 26 及图 27 所示的内容相同。

图 37 是示出了与纠错部件 3130 的详细结构相关联的其他构成部件的电路

图。纠错部件 3130 是用于一边时分地切换 1 行代码串和 1 列代码串，一边并行纠错的电路，由出错位计算部件 3131、欧基里德计算部件 3132、链计算部件 3133 以及错误数据更新部件 3134 构成。

出错位计算部件 3131 是用于一边区分来自数据选择部件 3190 的 2 种代码串，一边计算各自的出错位的电路，包括 2 个暂时存储器（输入数据出错位存储部件 3135 和缓冲数据出错位存储部件 3136）。输入数据出错位存储部件 3135 是具有暂时存储与构成 1 个数据块的全部 C1 序列的代码串相对应的出错位的存储容量的存储器，缓冲数据出错位存储部件 3136 是具有暂时存储与构成 1 个数据块的全部 C2 序列的代码串相对应的出错位的存储容量的存储器。

出错位计算部件 3131 在数据选择部件 3190 选择并通过了来自第一传输部件 3150 的数据（C1 代码串）时，对该 C1 代码串进行出错位计算，将得到的出错位储存到各代码串中并存入输入数据出错位存储部件 3135 中，另一方面，在数据选择部件 3190 选择并通过了来自第二传输部件 3160 的数据（C2 代码串）时，对该 C2 代码串进行出错位计算，将得到的出错位储存到各代码串中并存入缓冲数据出错位存储部件 3136 中。

另外，此出错位计算部件 3131 是用于瞬时计算出每 1 行或 1 列代码串中的出错位的电路，与第一实施例中的纠错部件 5 所具有的 4 个出错位生成部件 52~55 之一（图 8 所示的电路）相同。即，在乘积码的大小为 $k_1=k_2=100$ 字节、 $m_1=m_2=10$ 字节的情况下，出错位计算部件 3131 由用于对输入的数据进行暂时缓冲存储的队列缓冲器和 10 个用于分别计算出 10 个出错位的积和运算电路组成，各积和运算电路由用于存储 1 个积和值的寄存器、用于读出存储在该寄存器中的积和值并将其与固有的系数相乘的伽罗瓦域乘法器以及用于将相乘的结果与下次输入的 1 个数据相加的伽罗瓦域加法器组成。

欧基里德计算部件 3132 是用于顺序读出分别存在出错位计算部件 3131 的输入数据出错位存储部件 3135 和缓冲数据出错位存储部件 3136 中的出错位，计算出用于确定错误位置多项式和错误数值多项式各项系数的电路，计算出的系数被提供给链计算部件 3133。

链计算部件 3133 是利用来自欧基里德计算部件 3132 的系数，通过链查寻和调用法等方法算出所确定的错误位置多项式的根的电路，将其运算结果与来自欧基里德计算部件 3132 的用于确定错误数值多项式的系数一起提供给错误

数据更新部件 3134。

错误数据更新部件 3134 在利用来自链计算部件 3133 的错误位置和错误数值多项式, 求出位于该错误位置上的错误数值之后, 通过总线控制器 3180 读取缓冲存储器 3110 内该错误位置上的信息数据或奇偶校验数据, 计算它们的异或逻辑和, 将得出的值回写到缓冲存储器 3110 内的原位置上。

另外, 由第二传输部件 3160 进行的从缓冲存储器 3110 向出错位计算部件 3131 的数据传输是通过由 DMA 传输进行的存储器访问而进行的, 但是缓冲存储器 3110 与错误数据更新部件 3134 之间的数据传输是通过由总线控制器 3180 进行的普通 (非 DMA) 存储器访问而进行的。而且, 构成纠错部件 3130 的 4 个电路 3131~3134 在分别进行同步并列处理, 即进行流水线方式下的顺序处理。

图 38 是示出了在连续处理多个数据块的数据的情况下, 光盘控制器 3100 内的处理过程的时序图。在本图中, 第 1 数据块到第 5 数据块是分别存储在缓冲存储器 3110 中的 1 个数据块的数据。而且, “缓冲”表示由第一传输部件 3150 进行的从磁盘接口部件 3120 向缓冲存储器 3110 的数据传输, “纠错 1”表示纠错部件 3130 对由第一传输部件 3150 进行的经过数据选择部件 3190 从磁盘接口部件 3120 输入到纠错部件 3130 中的 C1 序列的代码串的纠错, “纠错 2”表示纠错部件 3130 对由第二传输部件 3160 进行的经过数据选择部件 3190 从缓冲存储器 3110 输入到纠错部件 3130 中的 C2 序列的代码串的纠错, “主机传输”表示由第三传输部件 3170 进行的从缓冲存储器 3110 向主机接口部件 3140 的数据传输。

如本图所示, 将与同一数据块相对应的并列的“缓冲”及“纠错 1”、对完成了“缓冲”及“纠错 1”的数据块的“纠错 2”以及对完成了“纠错 2”的数据块的“主机传输”作为流水线顺序且并列地执行。例如, 在周期 T5, 在从磁盘接口部件向缓冲存储器 3110 传输和存储第 5 数据块的同时, 在纠错部件 3130 中进行“纠错 1”(C1 序列的纠错), 从缓冲存储器 3110 向纠错部件 3130 传输第 4 数据块并进行“纠错 2”(C2 序列的纠错), 从缓冲存储器 3110 向主机接口部件 3140 传输第 3 数据块。

另外, 在本图中, 并列进行 3 种用于访问缓冲存储器 3110 的数据传输, 但这是从数据块这个大方面的数据传输观点上出发而表现出来的, 从构成数据块的各数据的角度考虑 (瞬时地), 只能执行其中一种数据传输。这样的排他

性数据传输及上述流水线控制,是通过总线控制器 3180 按照预定的优先顺序,将由 3 个传输部件 3150、3160、3170 进行的数据传输以较小的时间单位进行切换并许可而实现的。

而且,在本图中,不对第 3 数据块进行“纠错 2”(C2 序列的纠错)就是在“纠错 1”(C1 序列的纠错)中,这个数据块中没有发现任何错误。即,在纠错部件 3130 的出错位计算部件 3131 中为第 3 数据块的 C1 序列的代码串计算出全部为零的出错位,并将其存入输入数据出错位存储部件 3135 的情况下,或在周期 T3 进行的纠错 1 中对 C1 序列的全部代码串执行纠错并且不存在错误的情况下,纠错部件 3130 通过参照出错位存储部件 3135 的内容,明确了不必对第 3 数据块进行 C2 序列的纠错,并且不进行与对该数据块的 C2 序列纠错相关的一切处理。即,通过纠错部件 3130 将上述意思通知给总线控制器 3180,对第 3 数据块的 C2 序列的全部代码串,不进行由第二传输部件 3160 执行的从缓冲存储器 3110 向纠错部件 3130 的数据传输。

这样,由于并列进行对同一数据块的缓冲和“纠错 1”,并将与错误检测相关的信息存储到输入数据出错位存储部件 3135 中,因此在 C1 序列的代码串中都没有发生错误的情况下,纠错部件 3130 通过参照输入数据出错位存储部件 3135 而知道了此情况,因而能够避免从缓冲存储器 2110 读出不需要进行 C2 序列纠错的数据的所谓无效数据传输。即,若在现有技术中,对于从光盘读出的全部数据,要进行存入缓冲存储器、从缓冲存储器中读出以进行错误检测·纠正、从缓冲存储器中读出以将该数据输出到主计算机的所谓 3 种数据传输,但在本第三实施例中,对于没有发生错误的数据,只进行存入缓冲存储器和从缓冲存储器中读出以将数据输出到主计算机的所谓 2 种数据传输。

图 39 是图 38 所示的时序图的更详细的示图。在与“纠错 1”和“纠错 2”这 2 个处理相关的过程中分别示出了对 C1 序列及 C2 序列的处理。其中,在“纠错 1”中,只进行对 C1 序列的纠错,而在“纠错 2”中,只进行对 C2 序列的纠错。

具体地,在周期 T1,纠错部件 3130 对于通过第一传输部件 3150 从磁盘接口部件 3120 传送来的第 1 数据块,进行对 C1 序列的代码串的纠错。此时,出错位计算部件 3131 将关于 C1 序列的全部代码串而得出的出错位存储到输入数据出错位存储部件 3135 中。

其中, 纠错部件 3130 与缓冲并行进行 C1 序列的纠错 (“纠错 1”), 是因为从磁盘接口部件 3120 向缓冲存储器 3110 的数据传输是按照从第 1 行到第 n2 行的顺序, 反复在行方向上扫描乘积码而进行的, 因而使 C1 序列的纠错变得很容易。

接着在周期 T2, 纠错部件 3130 并列进行对第 2 数据块的 C1 序列纠错和对第 1 数据块的 C2 序列纠错。具体地, 这些第 2 数据块和第 1 数据块的数据是时分地从数据选择部件 3190 传送来的, 因此纠错部件 3130 在经过数据选择部件 3190 从磁盘接口部件 3120 输入第 2 数据块时, 对其代码串进行 C1 序列的纠错, 另一方面, 在经过数据选择部件 3190 从缓冲存储器 3110 输入第 1 数据块时, 对其代码串进行 C2 序列的纠错。

另外, 如果在对第 1 数据块进行的 C1 序列纠错中, 没有检测出错误, 或全部错误被纠正以后的情况下, 与第 3 数据块和第 4 数据块一样, 不进行对第 1 数据块的 C2 序列纠错。即, 不进行通过第二传输部件 3160 将第 1 数据块的 C2 序列的代码串从缓冲存储器 3110 传输到纠错部件 3130 的操作。

图 40 是示出了在纠错器 2130 内的行方向及列方向上的单位处理过程的时序图。本图相当于图 39 中所示的时序图中的周期 T2 开头部分中的 “纠错 1” 和 “纠错 2”, 示出了在纠错部件 3130 中并列进行第 2 数据块的 C1 序列纠错 (“纠错 1”) 和第 1 数据块的 C2 序列纠错 (“纠错 2”) 的样子。

在时间 t1, 出错位计算部件 3131 并列进行对第 2 数据块第 1 行代码串的出错位计算和对第 1 数据块第 1 列代码串的出错位计算, 并将各结果存入输入数据出错位存储部件 3135 和缓冲数据出错位存储部件 3136 中。实际上, 由于第 2 数据块第 1 行代码串和第 1 数据块第 1 列代码串是按照时间切换而从数据选择部件 3190 中传送的, 因而出错位计算部件 3131 在瞬时只是对其中一个代码串进行出错位计算。

在时间 t2, 出错位计算部件 3131 并列进行对第 2 数据块第 2 行代码串的出错位计算和对第 1 数据块第 2 列代码串的出错位计算, 并将各结果存入输入数据出错位存储部件 3135 和缓冲数据出错位存储部件 3136 中。

与此并行, 欧基里德计算部件 3132 首先读出存在输入数据出错位存储部件 3135 中的第 2 数据块第 1 行代码串的出错位, 利用它进行欧基里德计算, 在将结果通知给链计算部件 3133 之后, 接着进行对第 1 数据块第 1 列代码串

的欧基里德计算，并将结果通知给链计算部件 3133。

而且，在时间 t_2 的后一半时间中，链计算部件 3133 根据来自欧基里德计算部件 3132 的通知，对第 2 数据块第 1 行的代码串进行链计算，并将结果通知给错误数据更新部件 3134。

在时间 t_3 ，同样地，出错位计算部件 3131 并列进行对第 2 数据块第 3 行代码串的出错位计算和对第 1 数据块第 3 列代码串的出错位计算，与此并行，欧基里德计算部件 3132 在进行对第 2 数据块第 2 行代码串的欧基里德计算之后，接着进行对第 1 数据块第 2 列代码串的欧基里德计算，与此并行，链计算部件 3133 在对第 1 数据块第 1 行的代码串进行链计算之后，接着对第 2 数据块第 2 列的代码串进行链计算，与此并行，错误数据更新部件 3134 在对第 2 数据块第 1 行代码串进行错误数据的更新（缓冲存储器 3110 内错误数据的重写）之后，接着对第 1 数据块第 1 列代码串进行错误数据的更新。

在以后的个时间 t_n 中，同样地并列执行对 1 行代码串和 1 列代码串的出错位计算、欧基里德计算、链计算及错误数据纠正。另外，在本图中，虽然对全部行和列进行了欧基里德计算、链计算及错误数据纠正，但这是基于在对这些代码串进行出错位计算时检测出错误的原因，如果不是这样，则不进行对该代码串的欧基里德计算、链计算及错误数据的纠正。

如上所述，在第三实施例中，在进行“缓冲”（从磁盘接口部件 3120 向缓冲存储器 3110 的数据传输）的同时，并列进行对同一数据的 C1 序列纠错（“纠错 1”）。由此，在将数据存入缓冲存储器 3110 之后，就不需要所谓的读出存在缓冲存储器 3110 中的数据以用于 C1 序列纠错的过程了。

而且，在与缓冲同时进行的纠错中，由于一个数据块的 C1 序列的错误检测结果（出错位）被记录到输入数据出错位存储部件 3135 中，因此在构成 1 个数据块的全部 C1 序列代码串中没有发生错误的情况下，通过参照该记录，能够避免不必要的从缓冲存储器 3110 读出数据以进行 C2 序列纠错的所谓无效数据传输。

这样，依据第三实施例，不需要从缓冲存储器 3110 读出数据以进行 C1 序列纠错和从缓冲存储器 3110 中读出数据以对没有发生错误的的数据块进行 C2 序列纠错，从而削减了对缓冲存储器 3110 的访问次数，并实现了更高速的低功耗光盘读取装置 2000。

另外,在第三实施例中,纠错部件 3130 反复进行逐行或逐列纠错,但也可以象第一实施例一样,进行每隔 4 行和 4 列等多代码串的并列纠错。

例如,第三实施例中象这样的由纠错部件 3130 对 4 个代码串并列进行纠错的变形例只要象第一实施例中的纠错部件 5 一样,在纠错部件 3130 中设置 4 个出错位计算部件和用于将来自缓冲存储器 3110 的数据分配到 4 个出错位计算部件中的数据分配部件即可。由此,就能够得到由第一实施例中的并列化而产生的纠错高速化和由第三实施例中的预先进行 C1 序列的纠错而产生的纠错高速化这两方面的效果。

而且,在第三实施例中,通过数据选择部件 3190 和出错位计算电路 3131,按时间对分别从第一传输部件 3150 和第二传输部件 3160 传送来的代码串进行错误检测,但是也可以代替这种结构,通过将只对来自第一传输部件 3150 的代码串进行错误检测的出错位计算电路和只对来自第二传输部件 3160 的代码串进行错误检测的出错位计算电路设计成 2 个独立的出错位计算电路,来同时并列进行对这 2 个代码串的错误检测。

以上根据第一至第三实施例对本发明进行了说明,但这并不意味着本发明只限于这些实施例。

例如,在第一至第三实施例中,对于 1 个数据块的纠错,只进行 1 次 C1 序列的纠错和接在其后的 C2 序列纠错,但是也可以在进行了这些纠错之后,继续进行 C1 序列的纠错和接在其后的 C2 序列纠错。由于通过增加纠错的次数可以提高纠错能力,因此只要考虑发现错误的数量和在其它相关处理中所容许的时间,决定对各方向反复进行纠错的次数即可。

而且,通过组合 3 个实施例中各自不同的结构部件,可以实现各种不同的变形。例如,图 27 中所示的第二及第三实施例中的对 3 种 DMA 传输的调度也可用于第一实施例中。

而且,第一至第三实施例是作为 DVD-ROM 等光盘的记录媒体的磁盘驱动装置,但是本发明涉及的纠错装置并不仅限于在这样的数据存储装置中使用,也可以用于要求能够对应转播的数据高速地进行纠错的机器,例如网络通信中的中继器等。

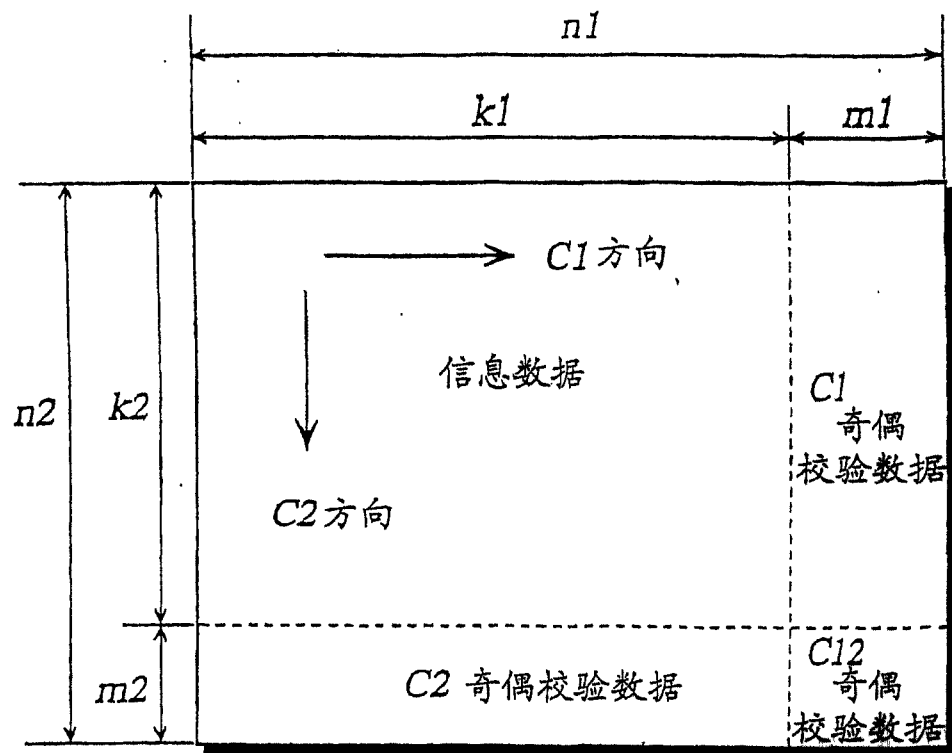


图 1

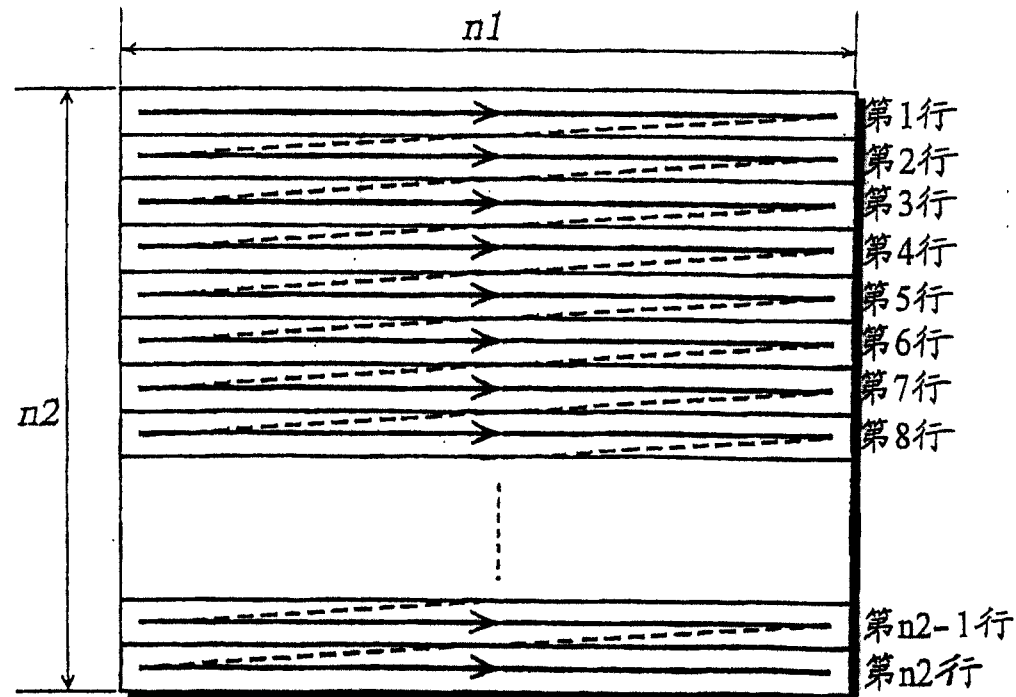


图 2

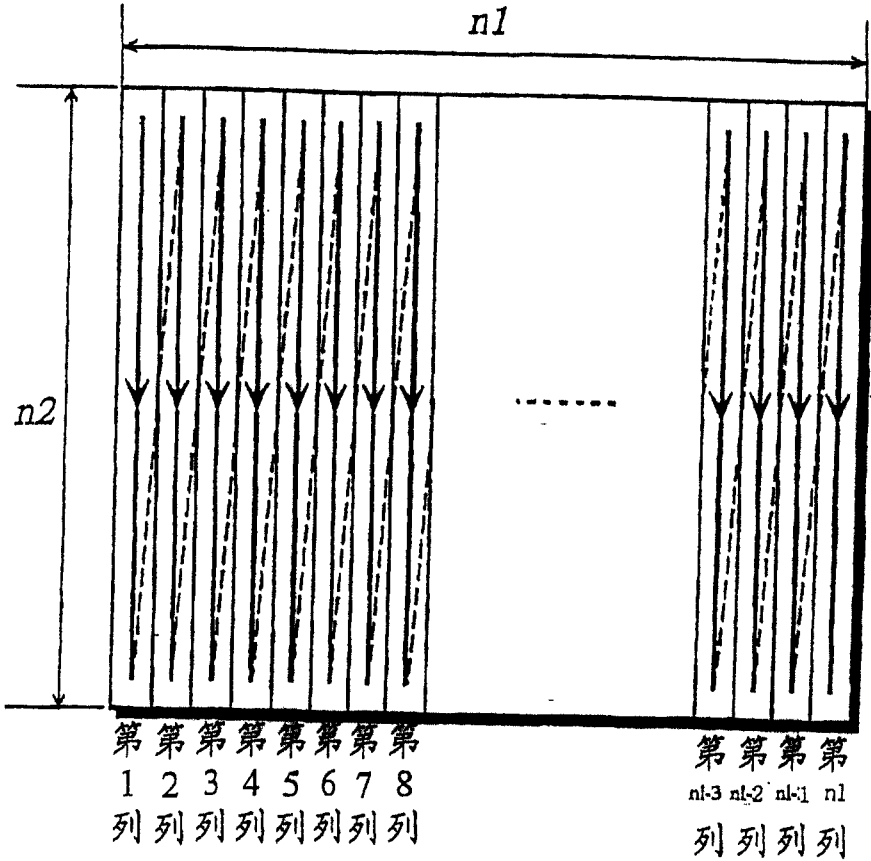


图 3

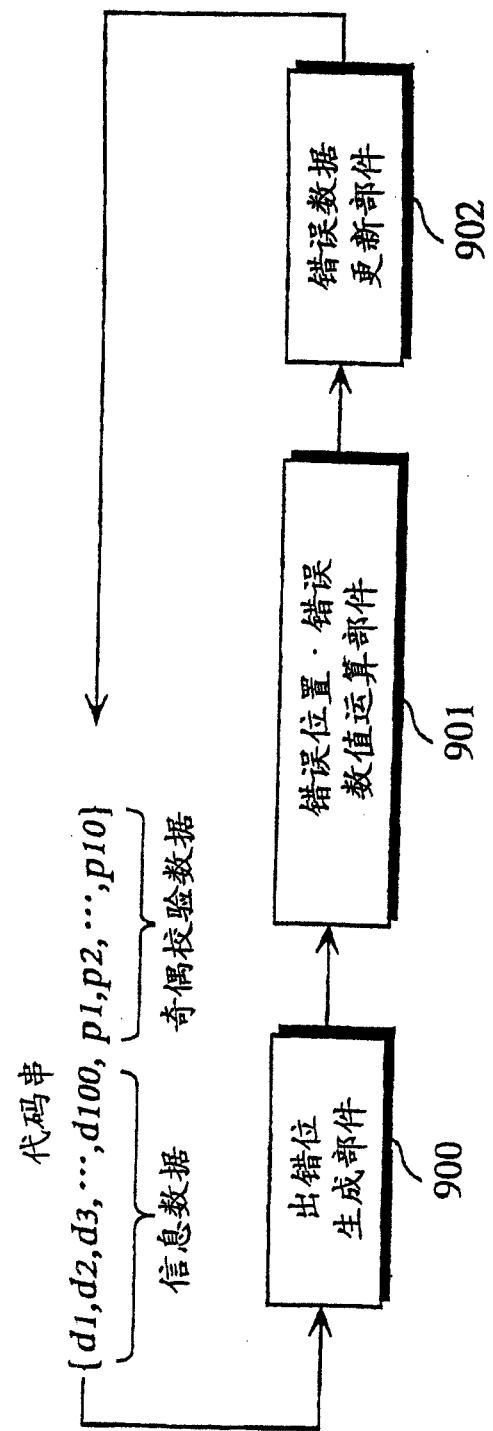


图 4

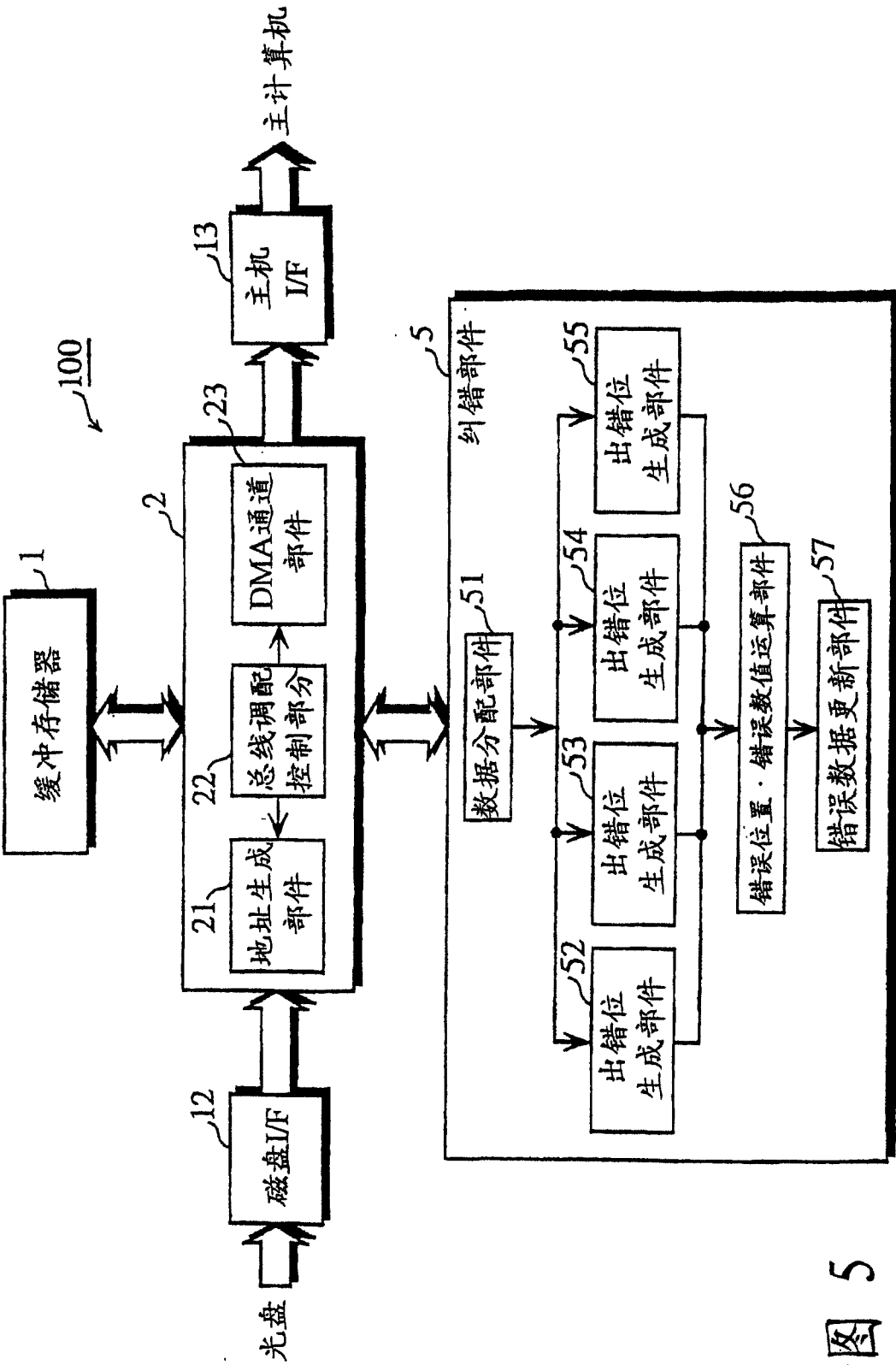


图 5

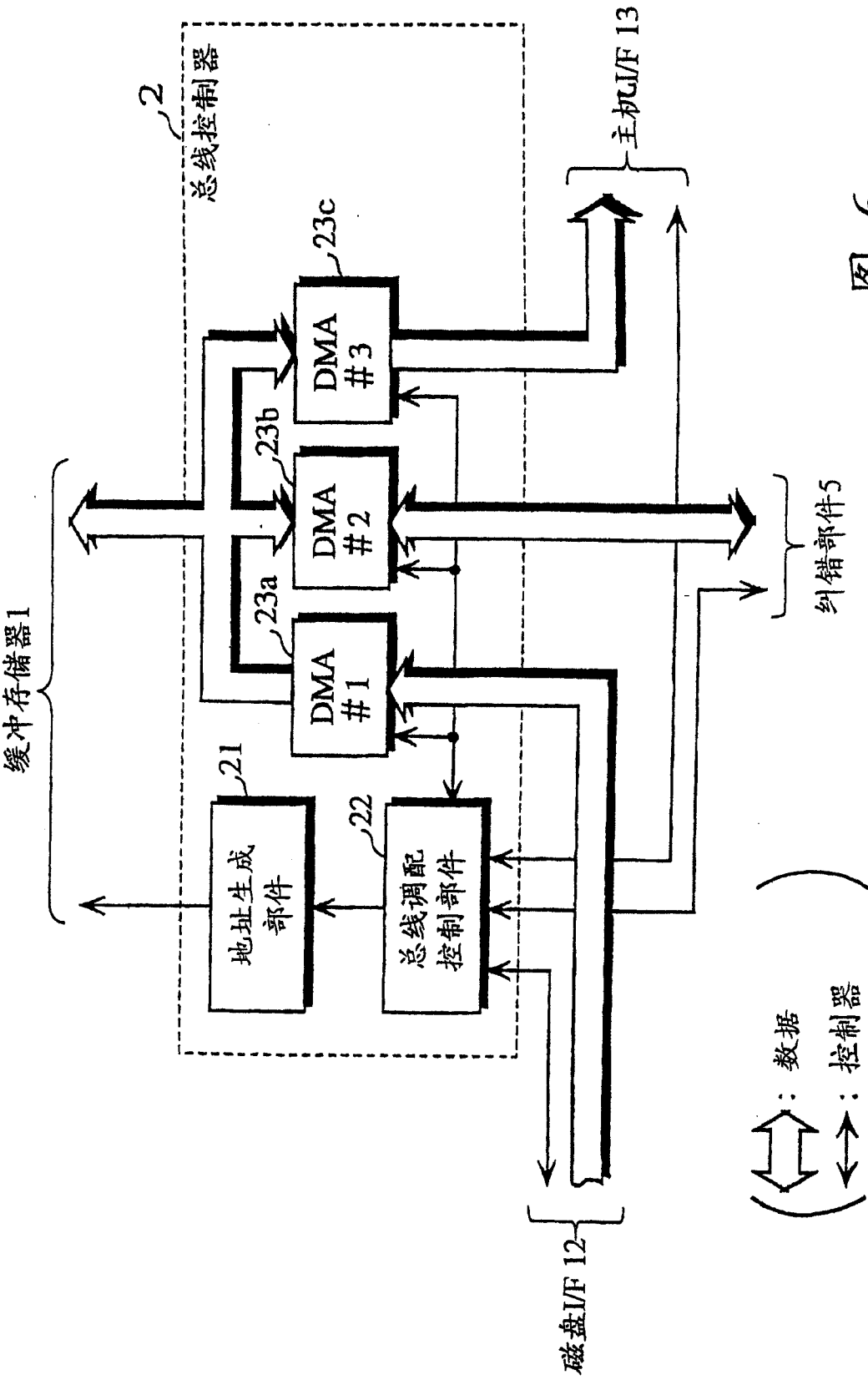


图 6

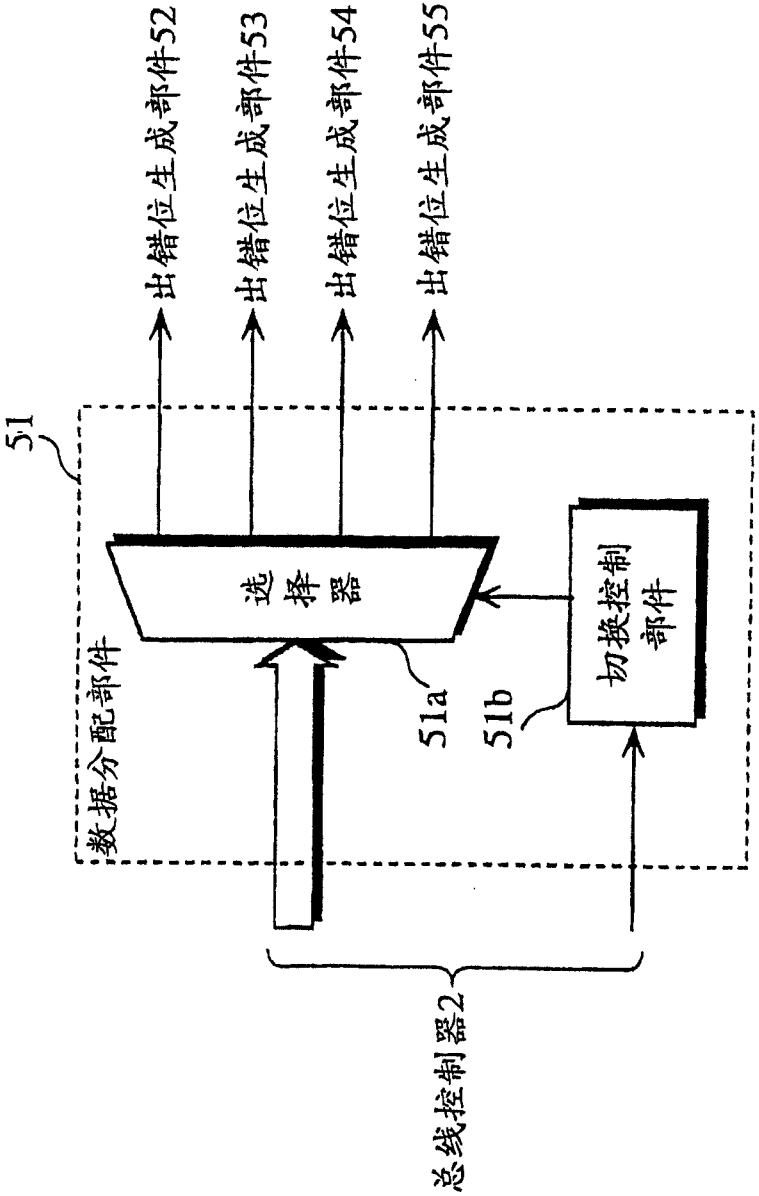


图 7

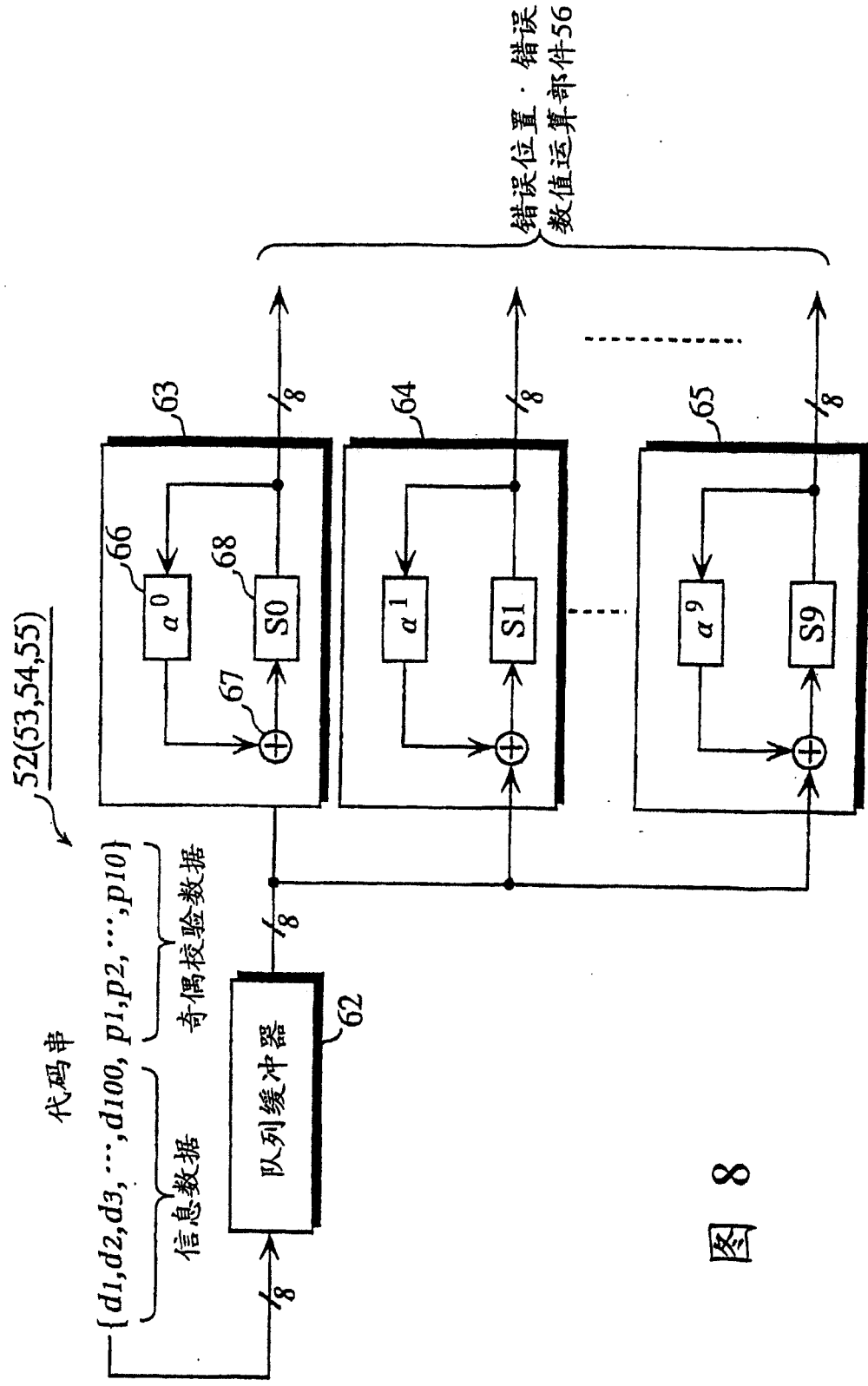


图 8

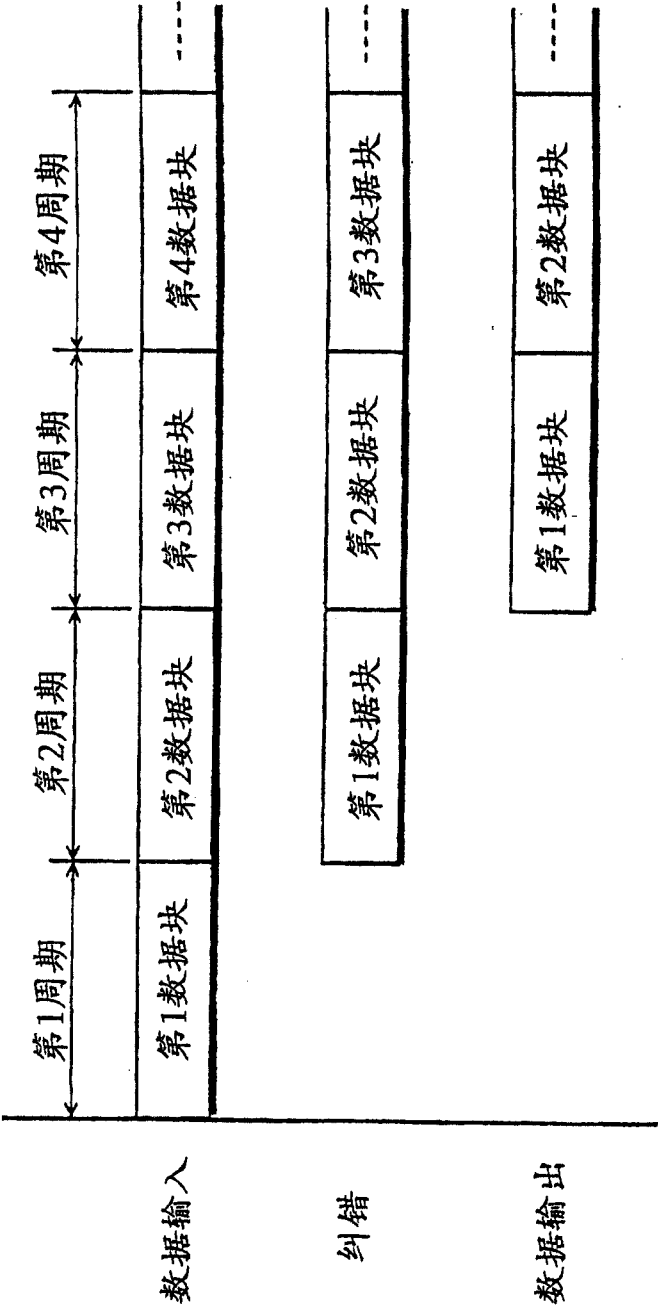


图 9

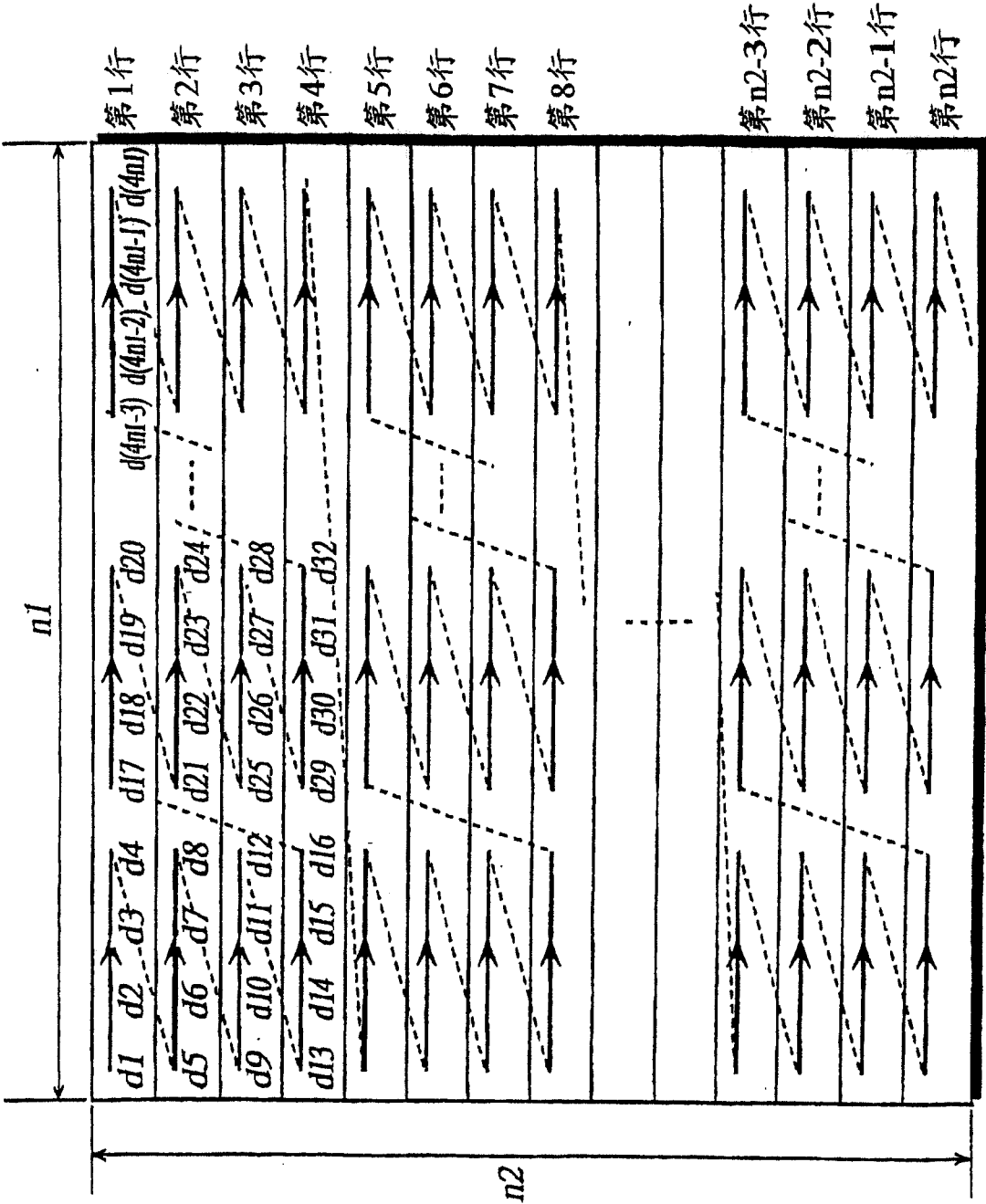


图 10

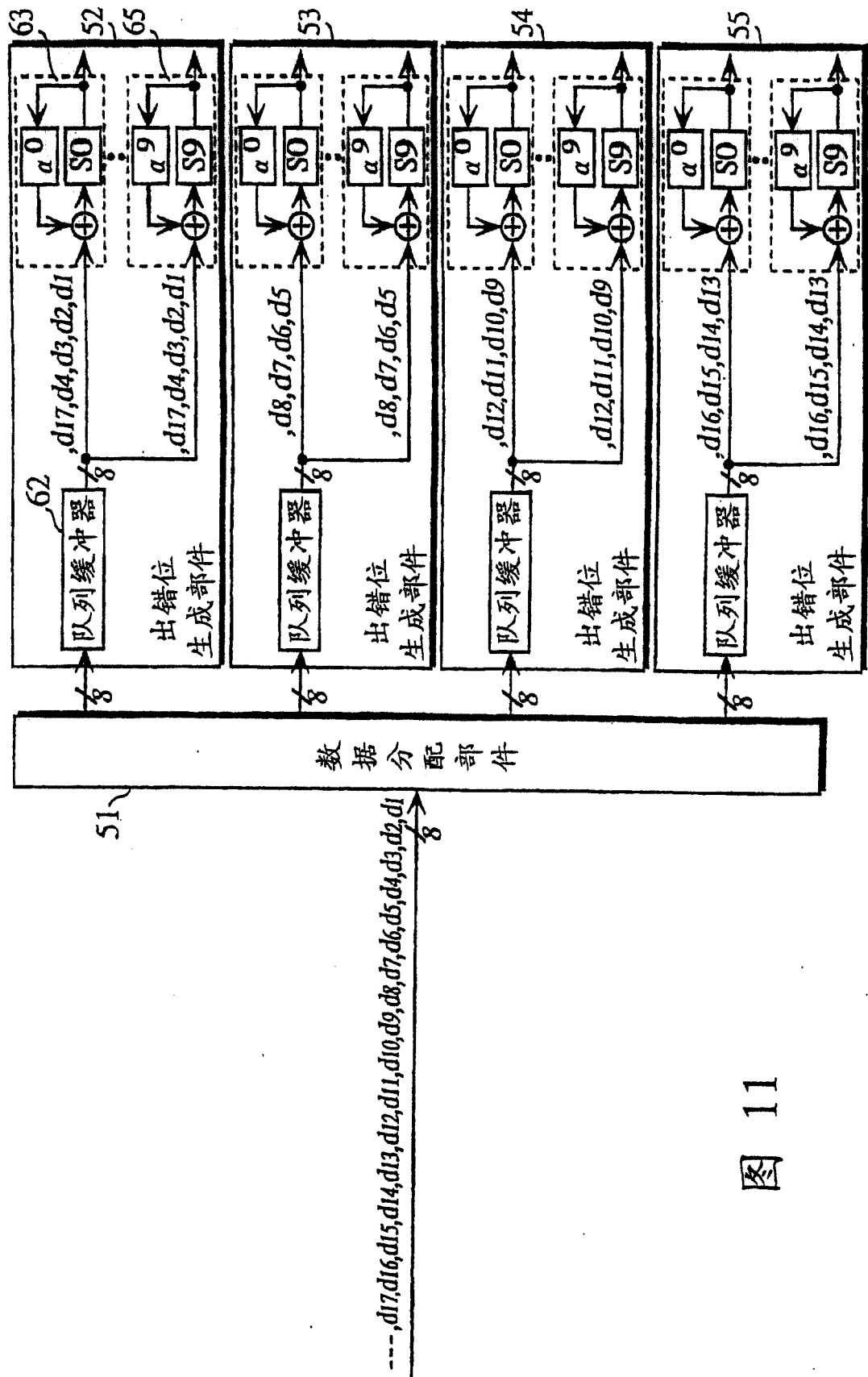


图 11

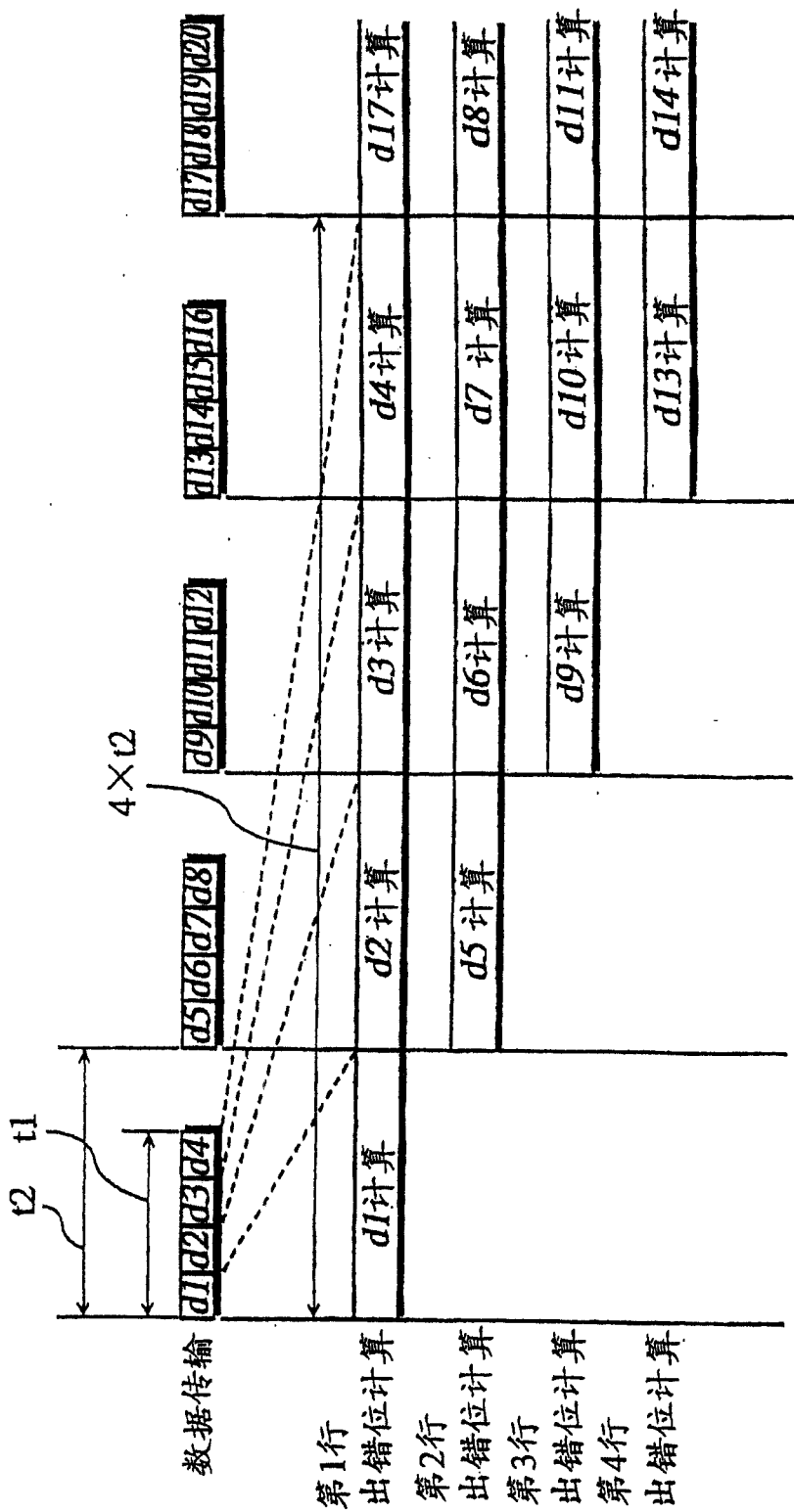


图 12

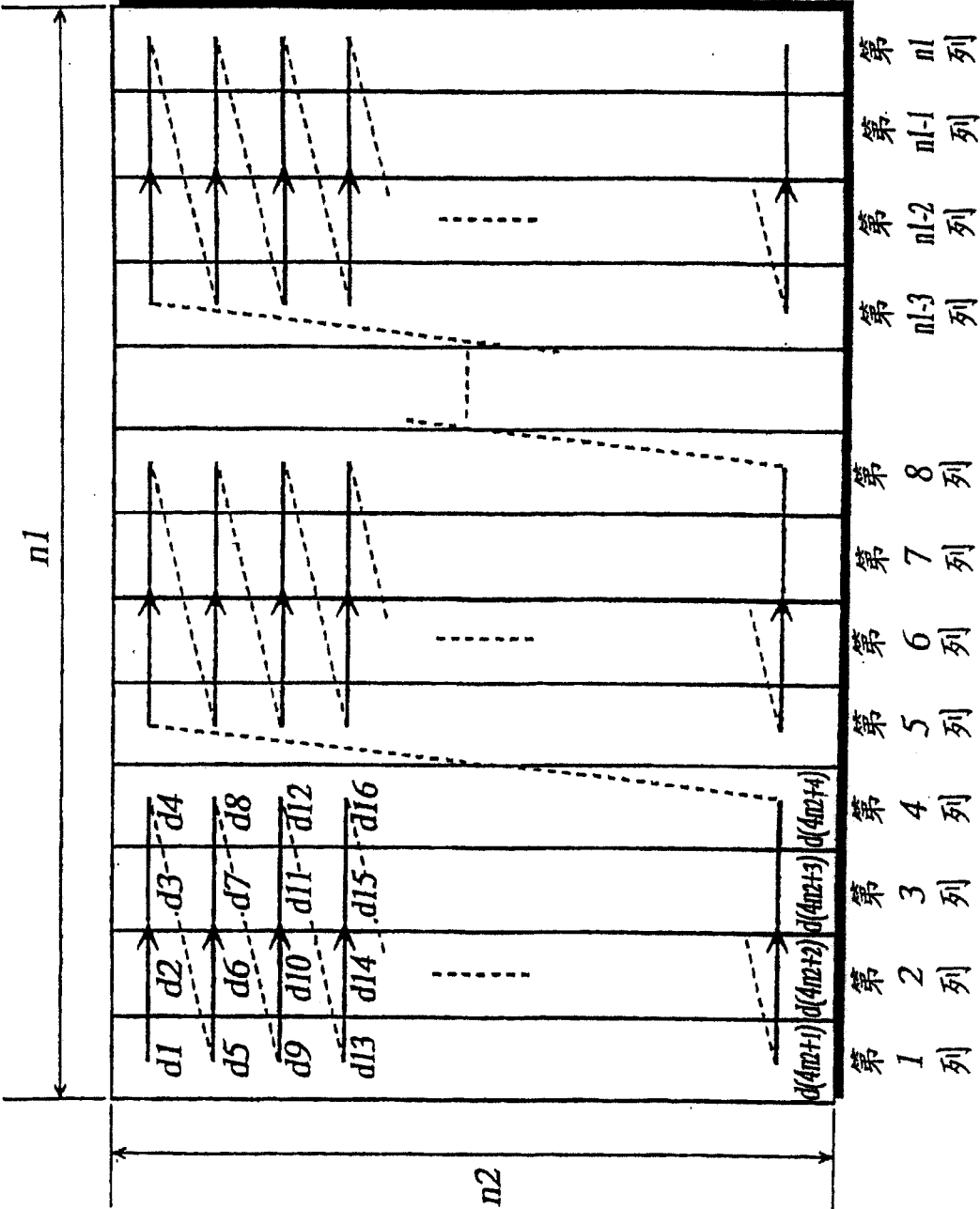


图 13

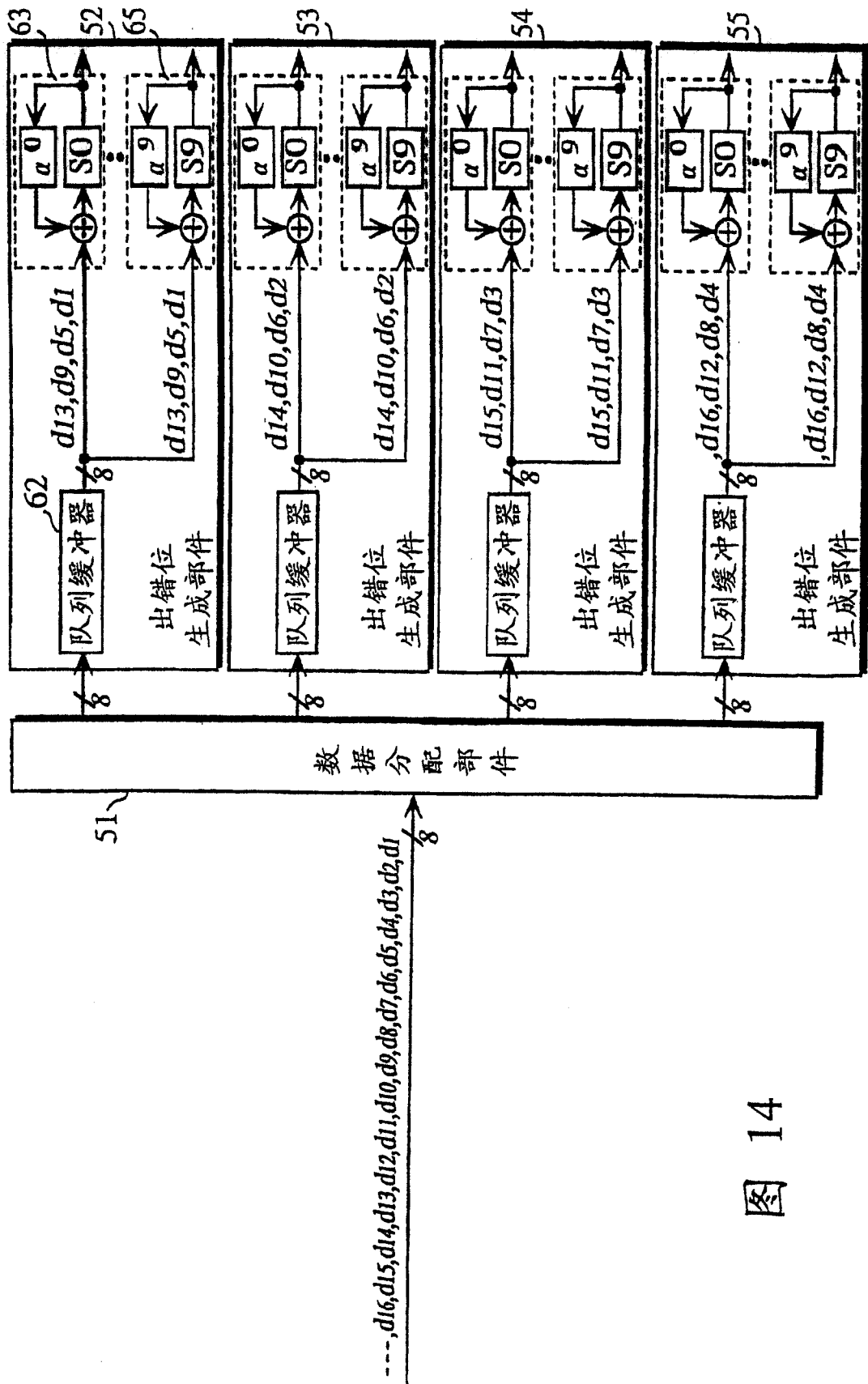


图 14

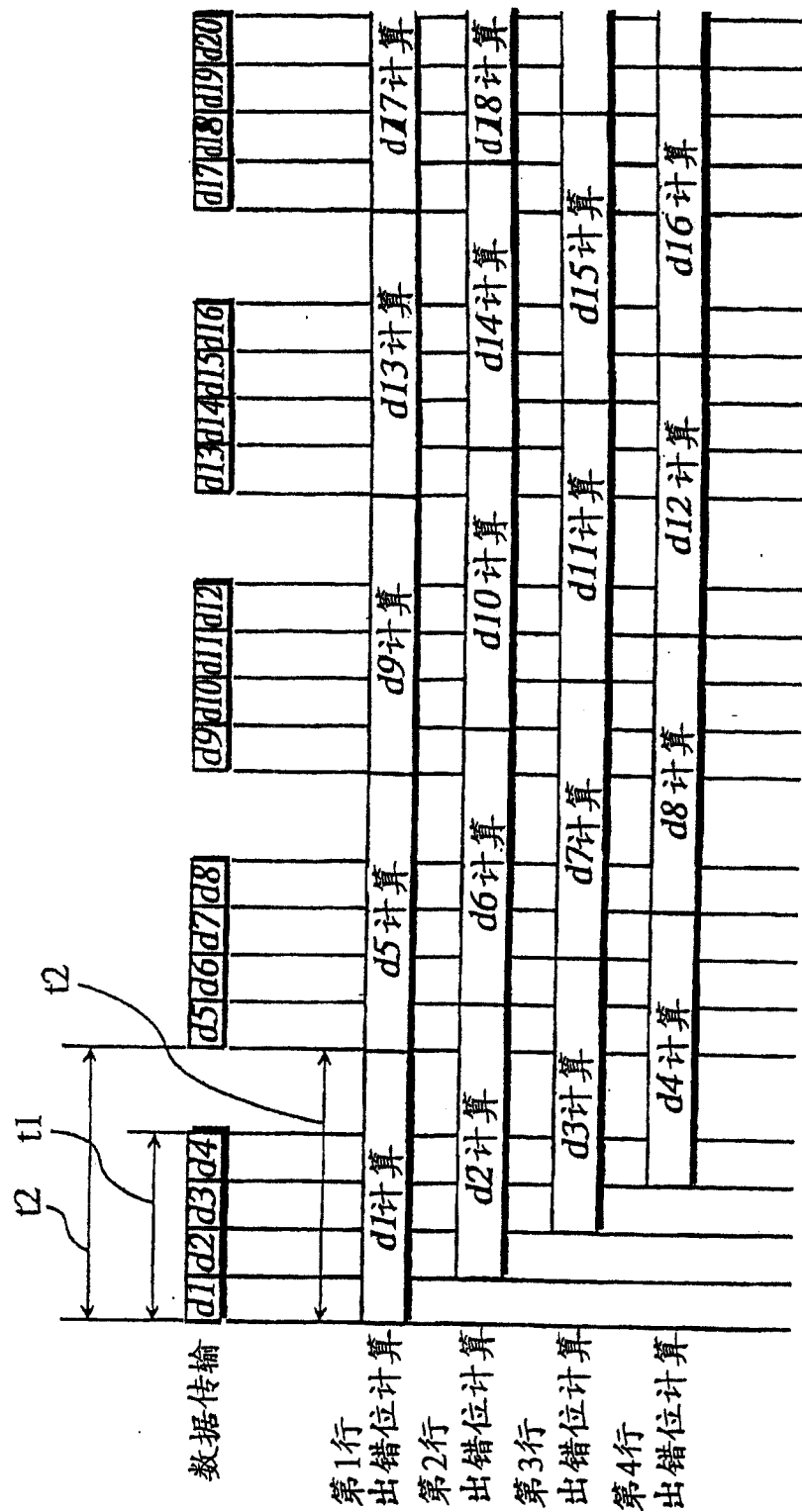


图 15

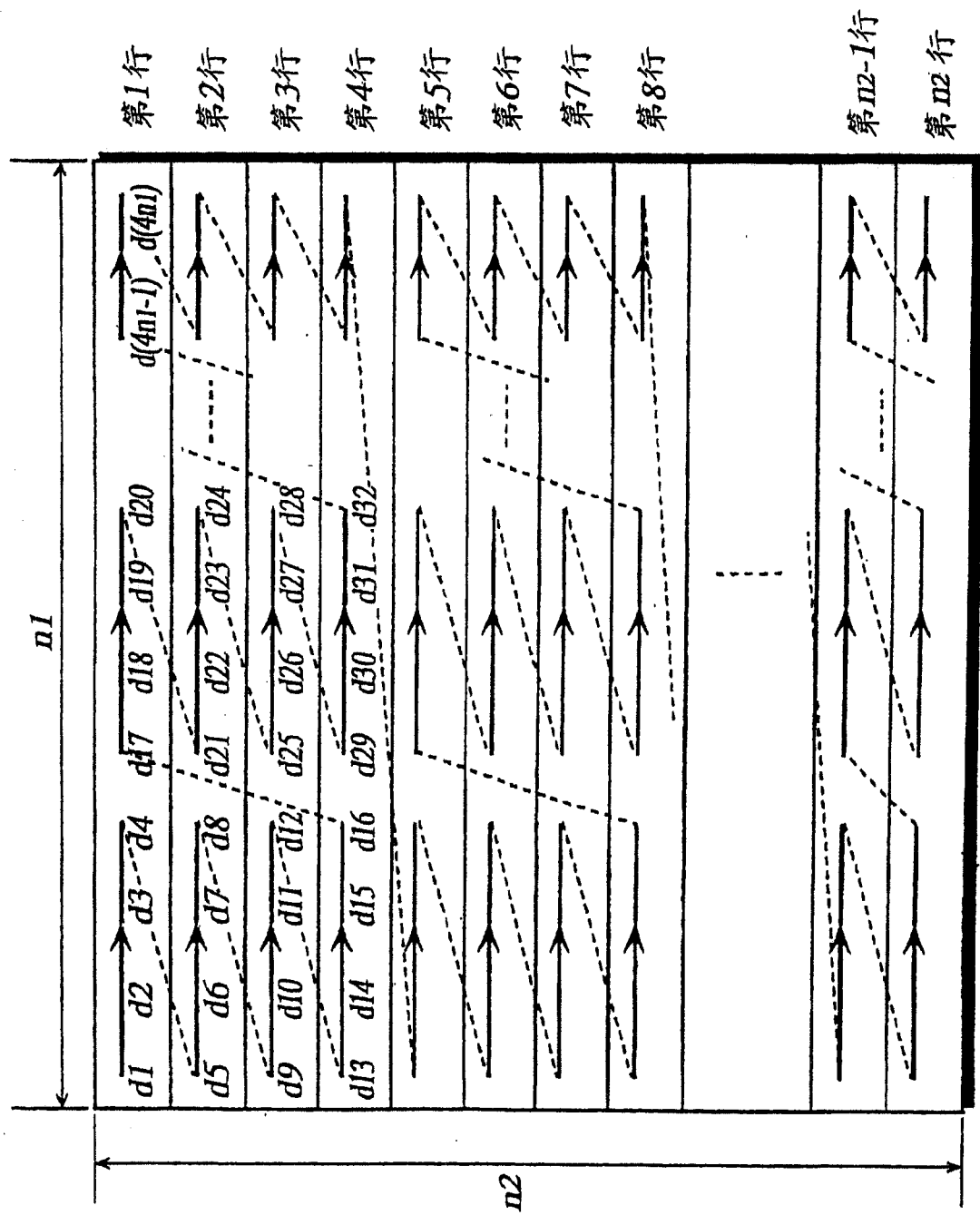


图 16

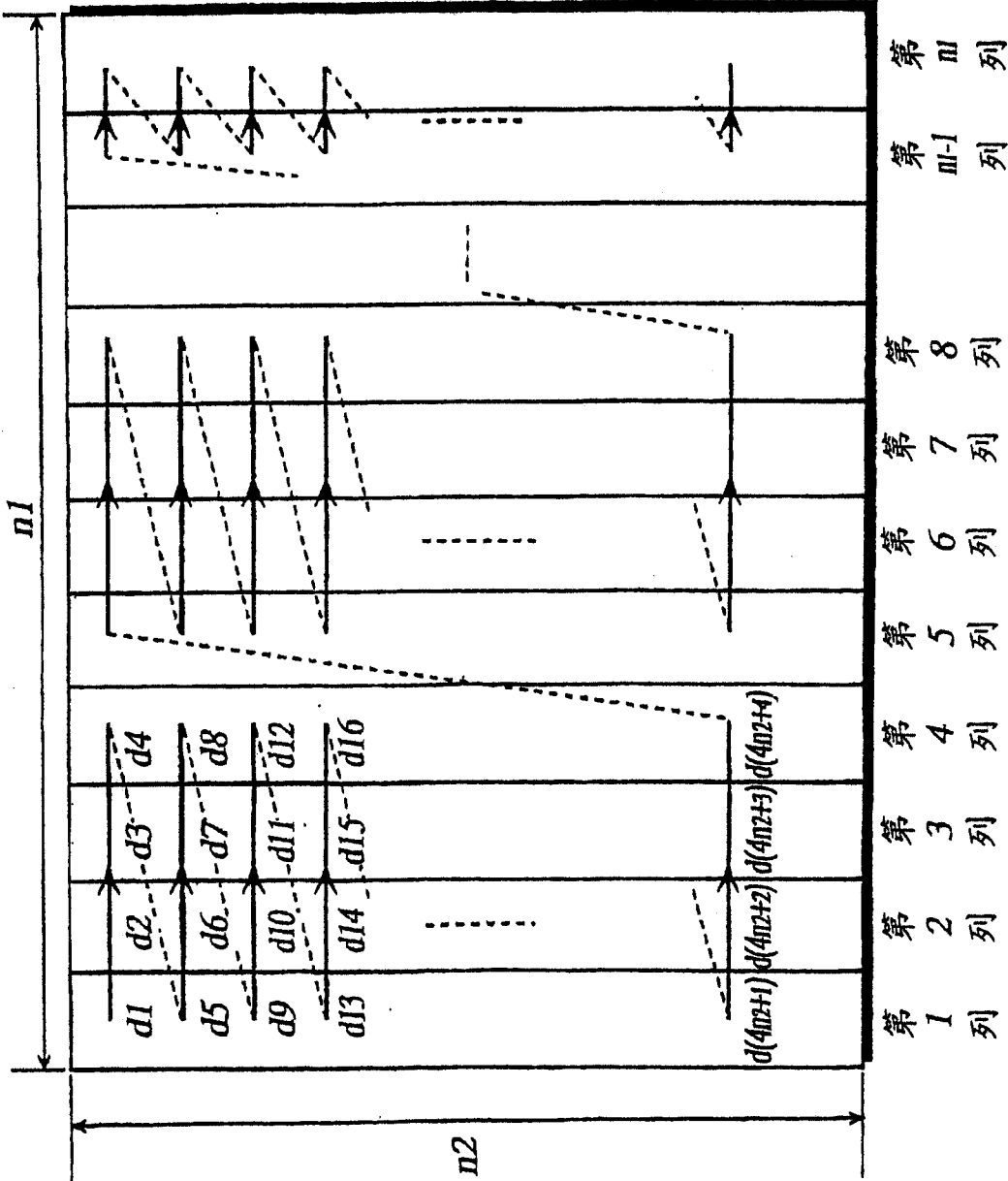


图 17

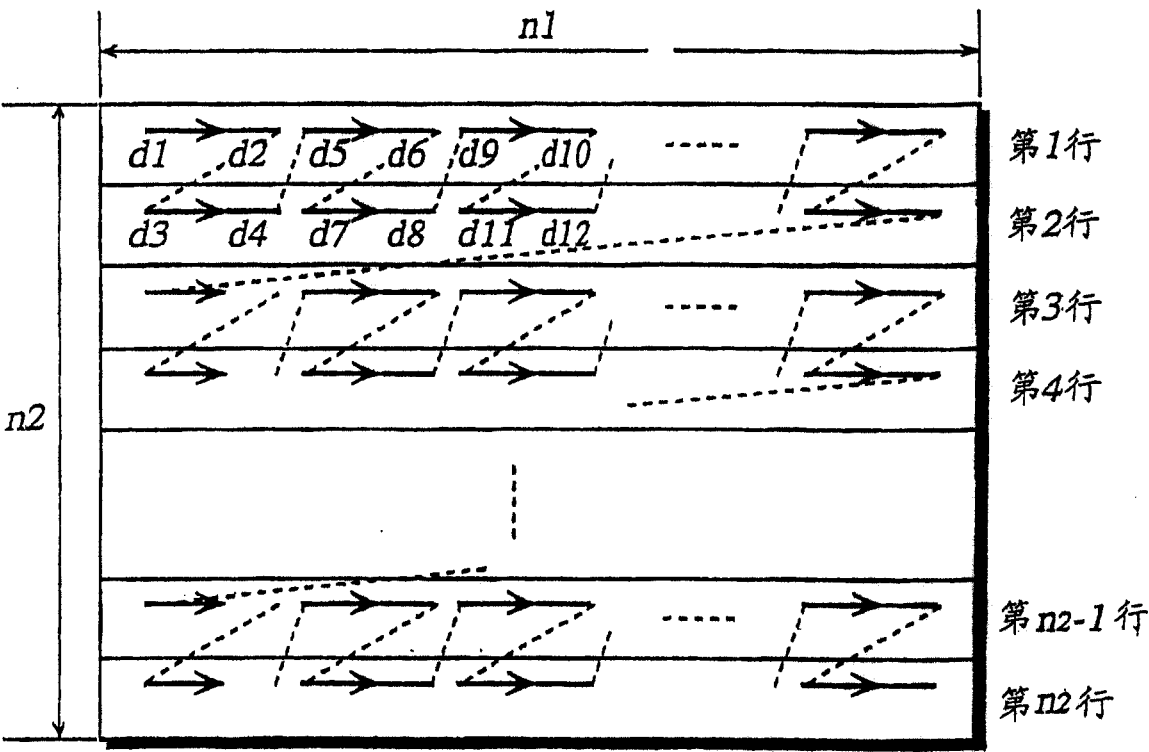


图 18

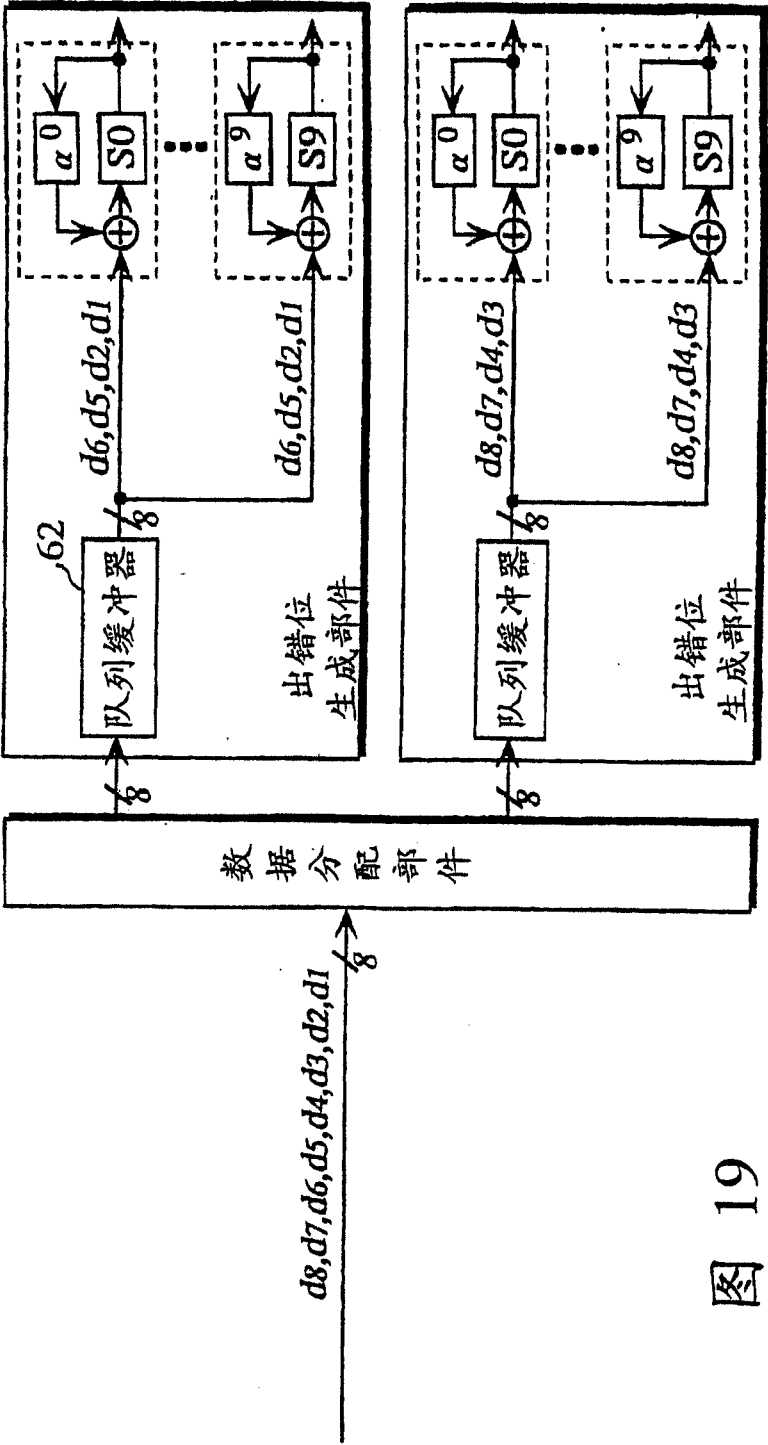


图 19

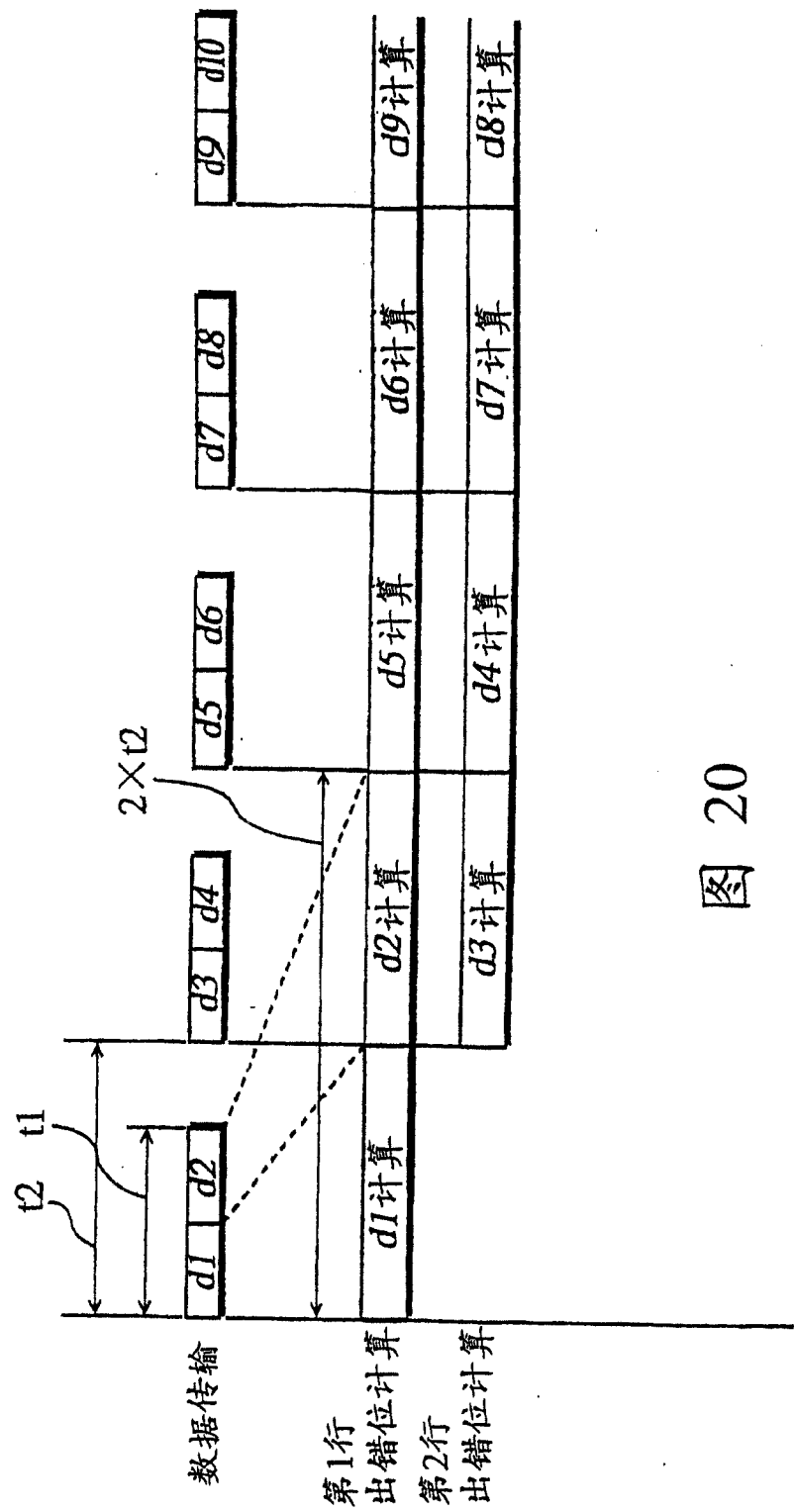


图 20

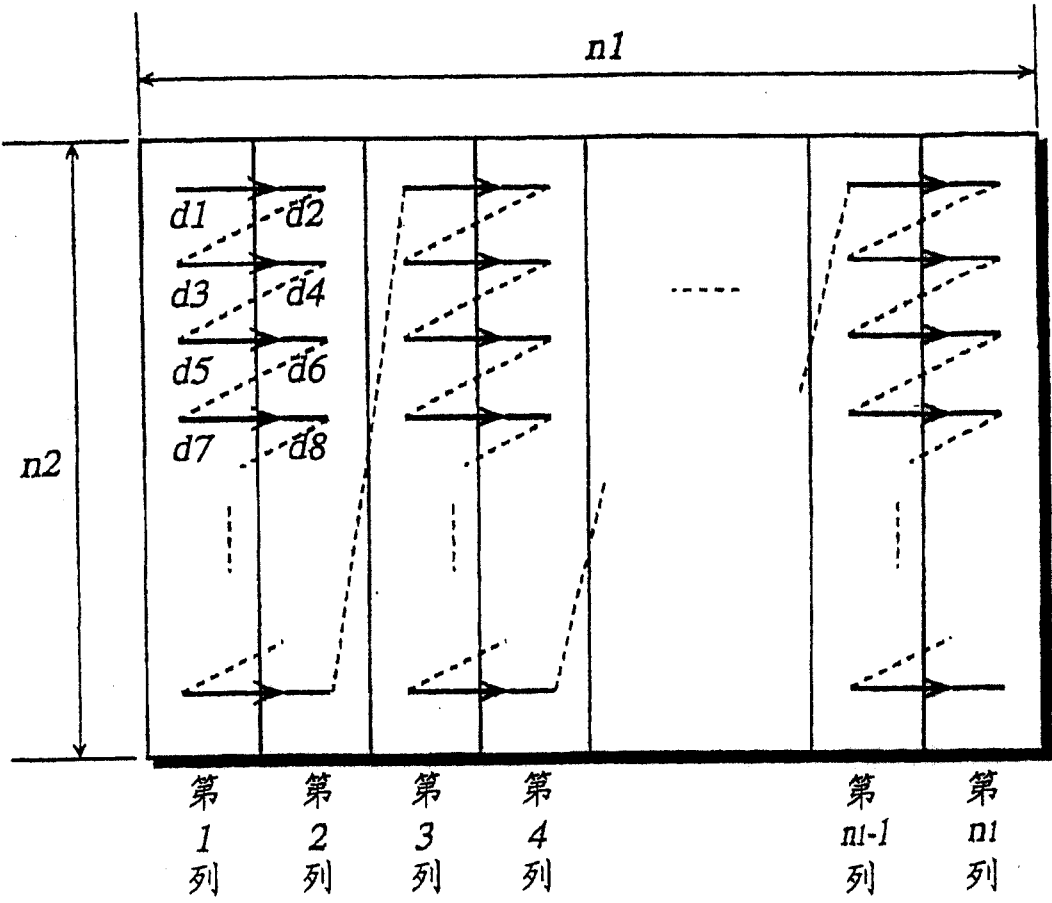


图 21

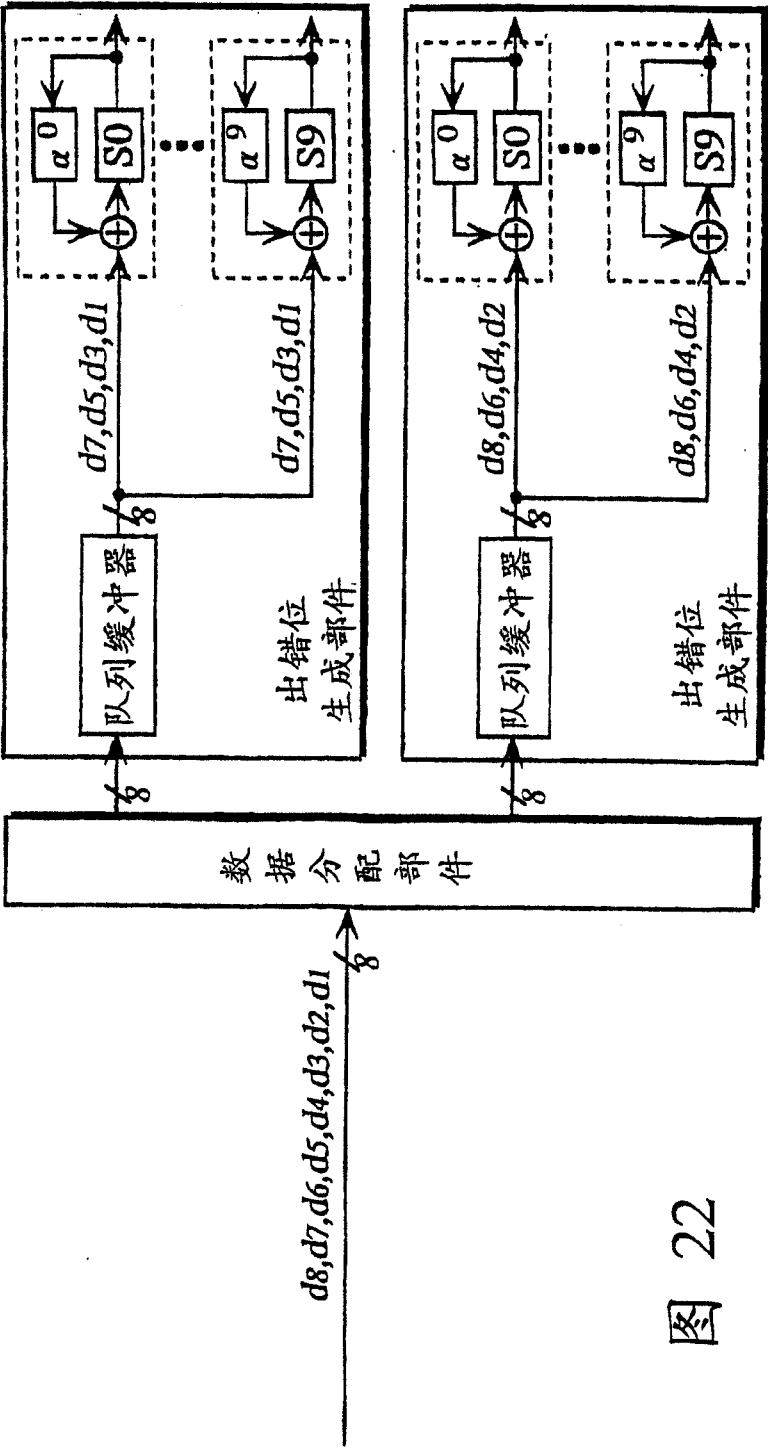


图 22

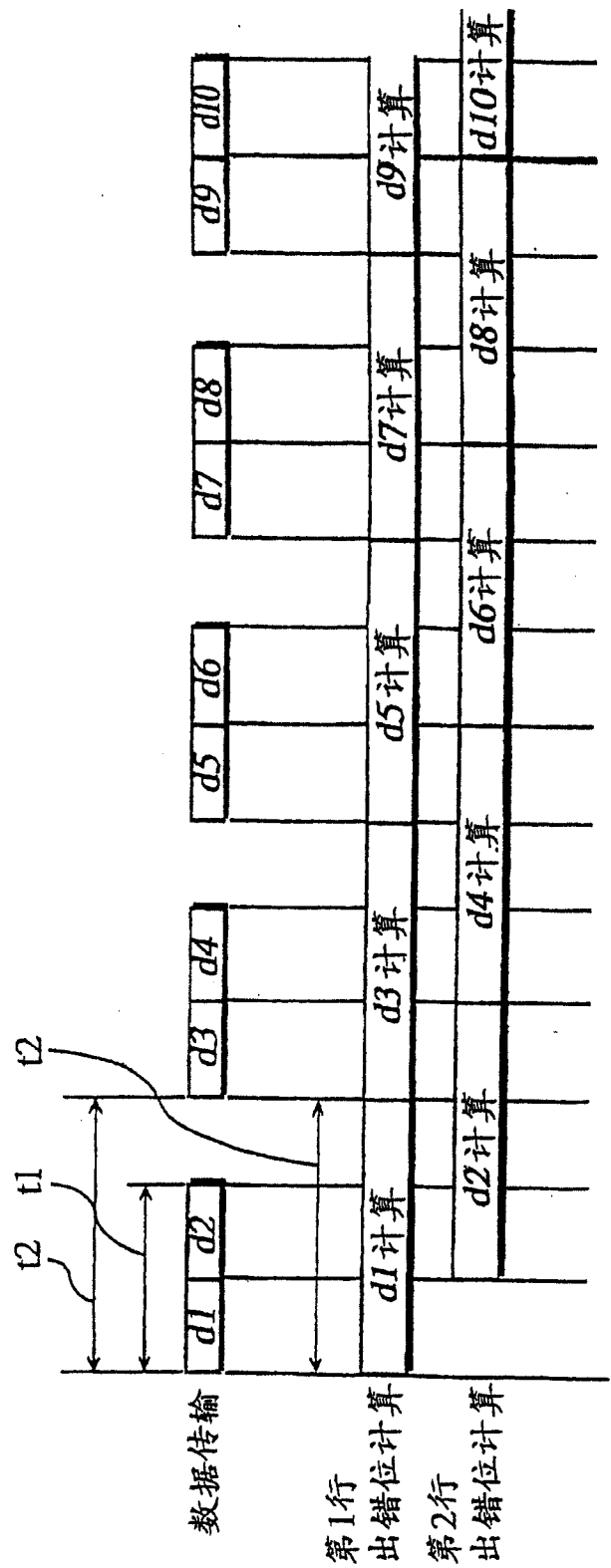


图 23

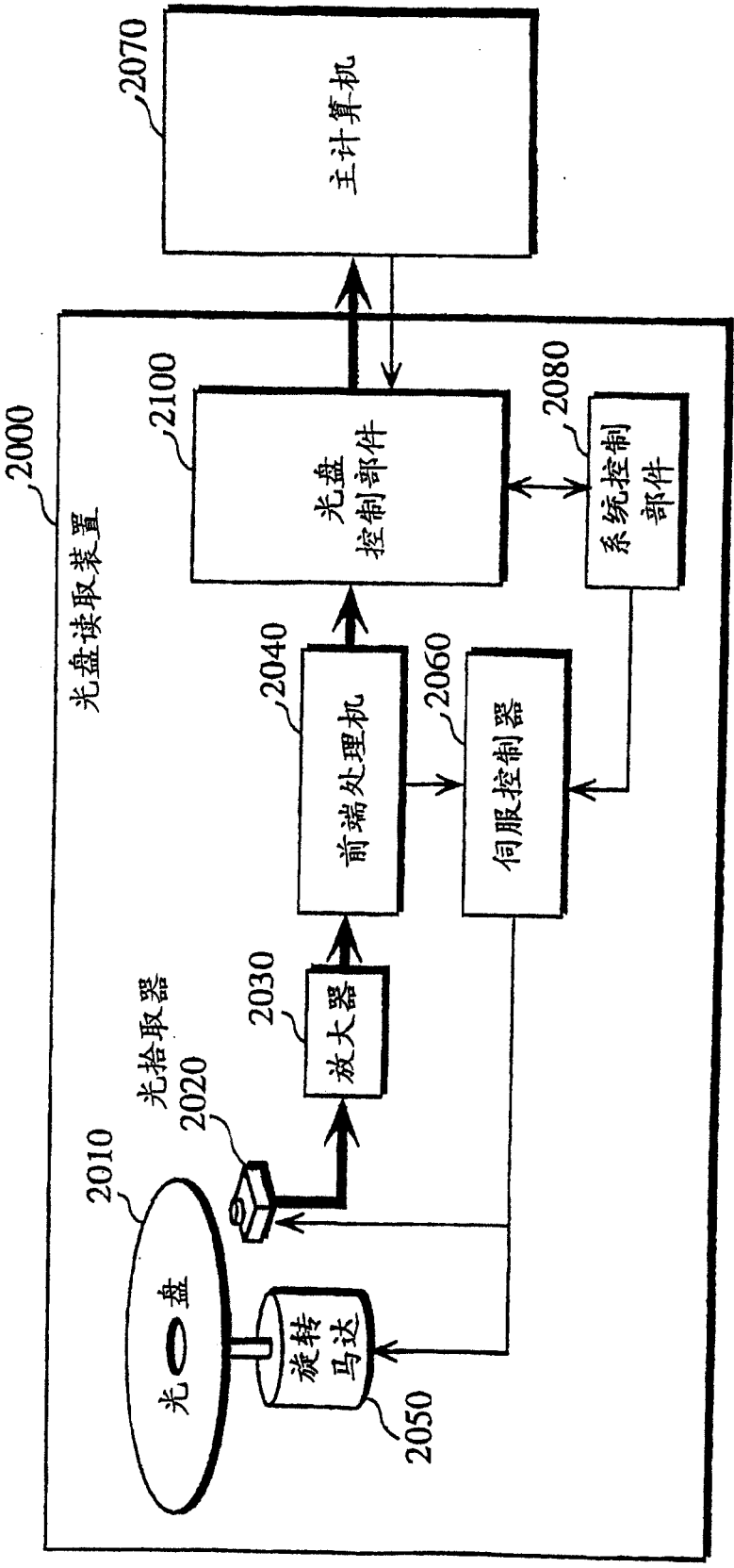


图 24

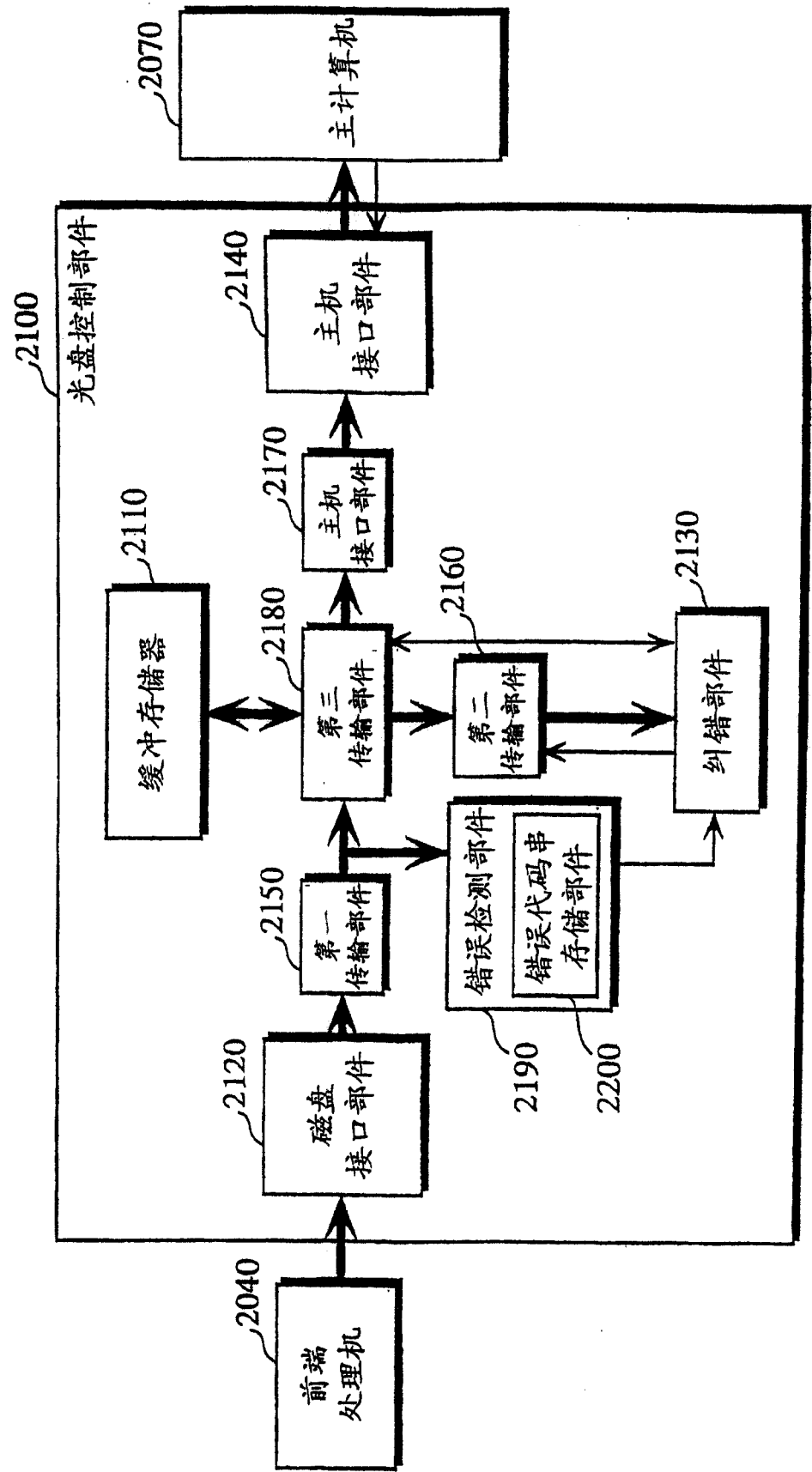


图 25

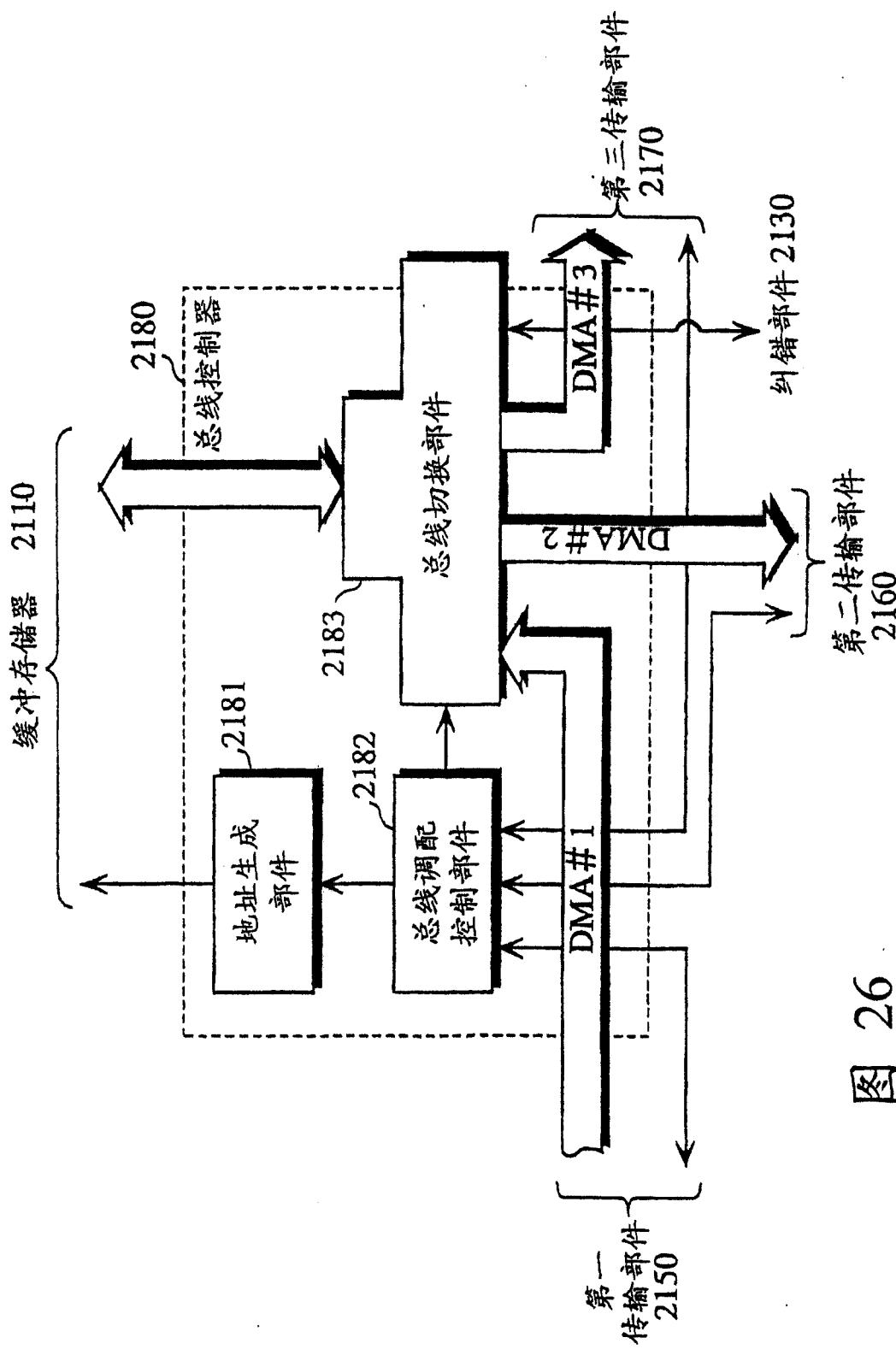


图 26

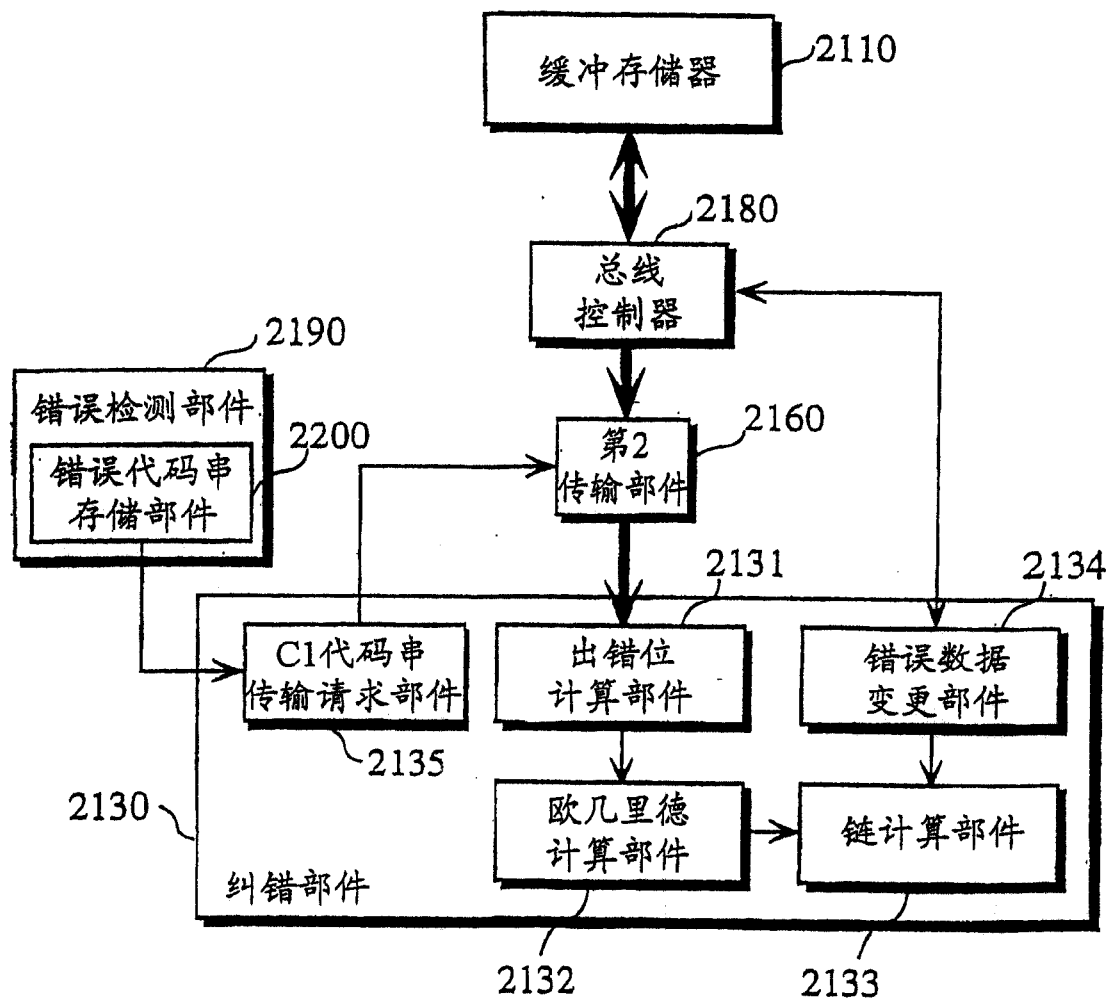


图 28

时间	T1	T2	T3	T4	T5	
缓冲						
	第1数据块	第2数据块	第3数据块	第4数据块	第5数据块	
错误检测						
	第1数据块	第2数据块	第3数据块	第4数据块	第5数据块	
纠错						
			第2数据块	——	第4数据块	
主机传输						
			第1数据块	第2数据块	第3数据块	

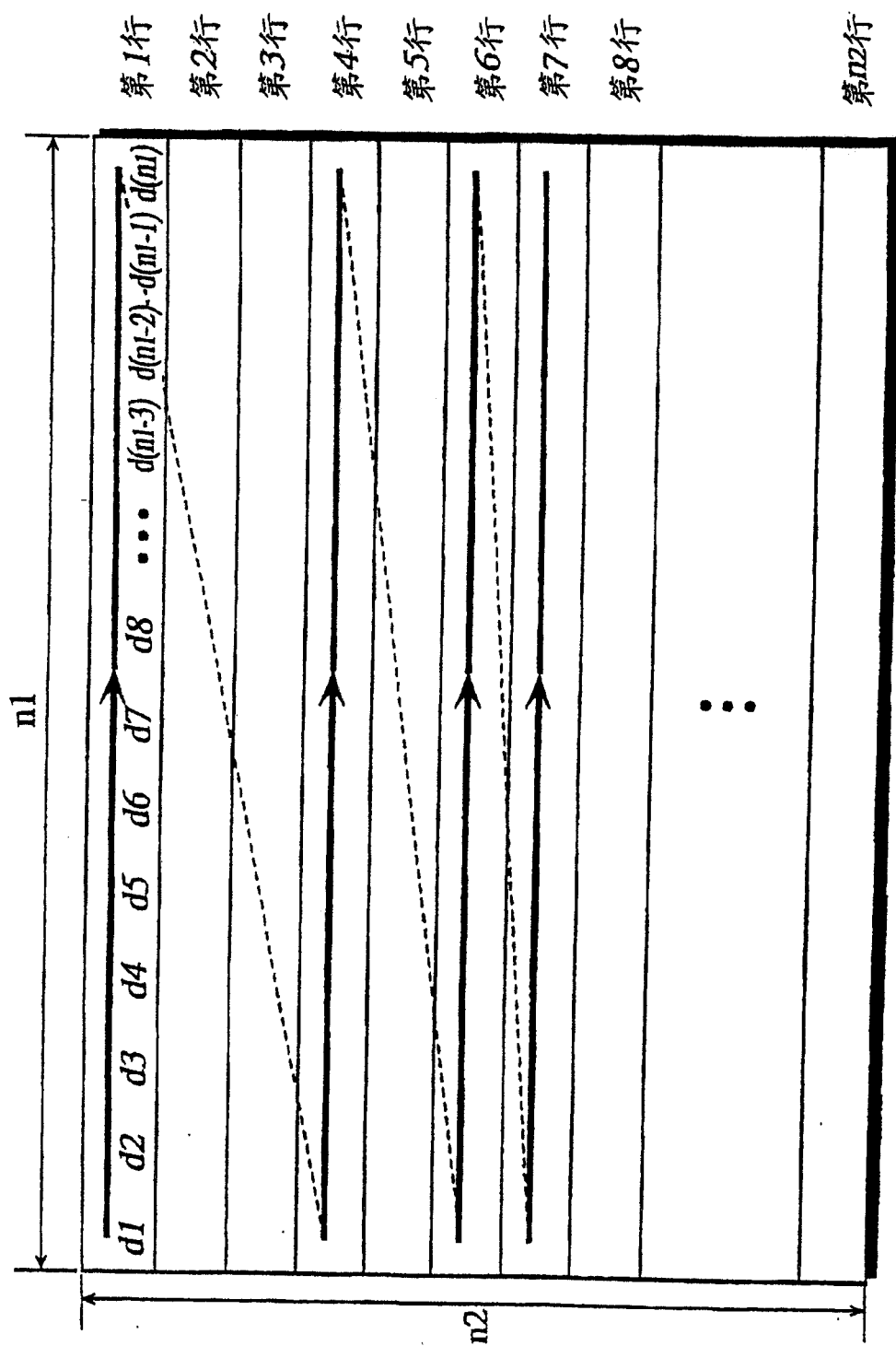
图 29

时间	T1	T2	T3
缓冲			
	第1数据块	第2数据块	第3数据块
错误检测	第1数据块	第2数据块	第3数据块
	C1 错误检测	C1 错误检测	C1 错误检测
纠错			
		第1数据块	第2数据块
		C1ECC C2ECC	C1ECC C2ECC
主机传输			
			第1数据块

图 30

时间	11	12	13	14	15	
出错位计算						
	第1行	第4行	第6行	第7行	第11行	
欧几里德计算						
		第1行	第4行	第6行	第7行	
链计算						
			第1行	第4行	第6行	
错误数据更新						
				第1行	第4行	

图 31



32 圖

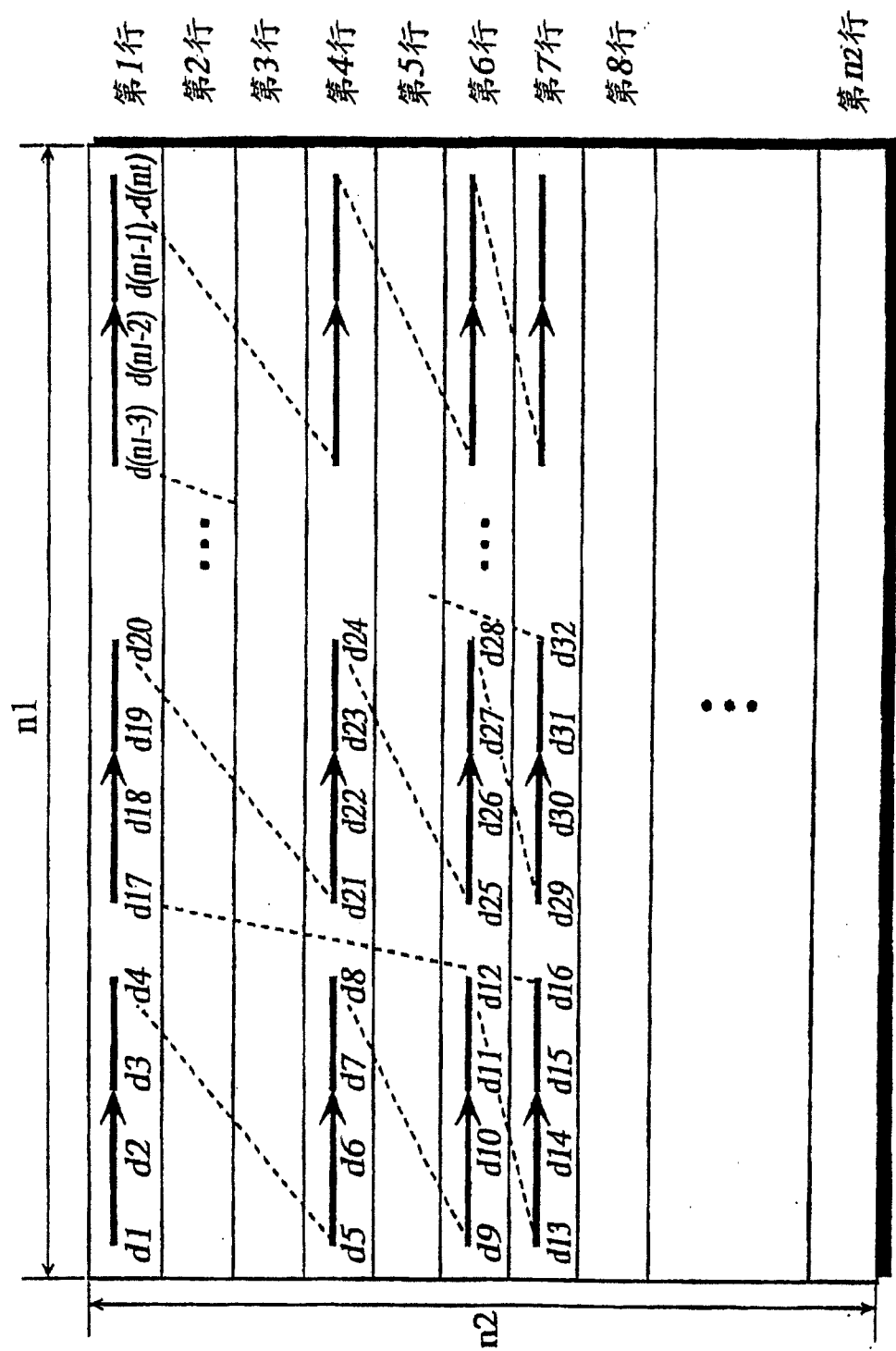


图 33

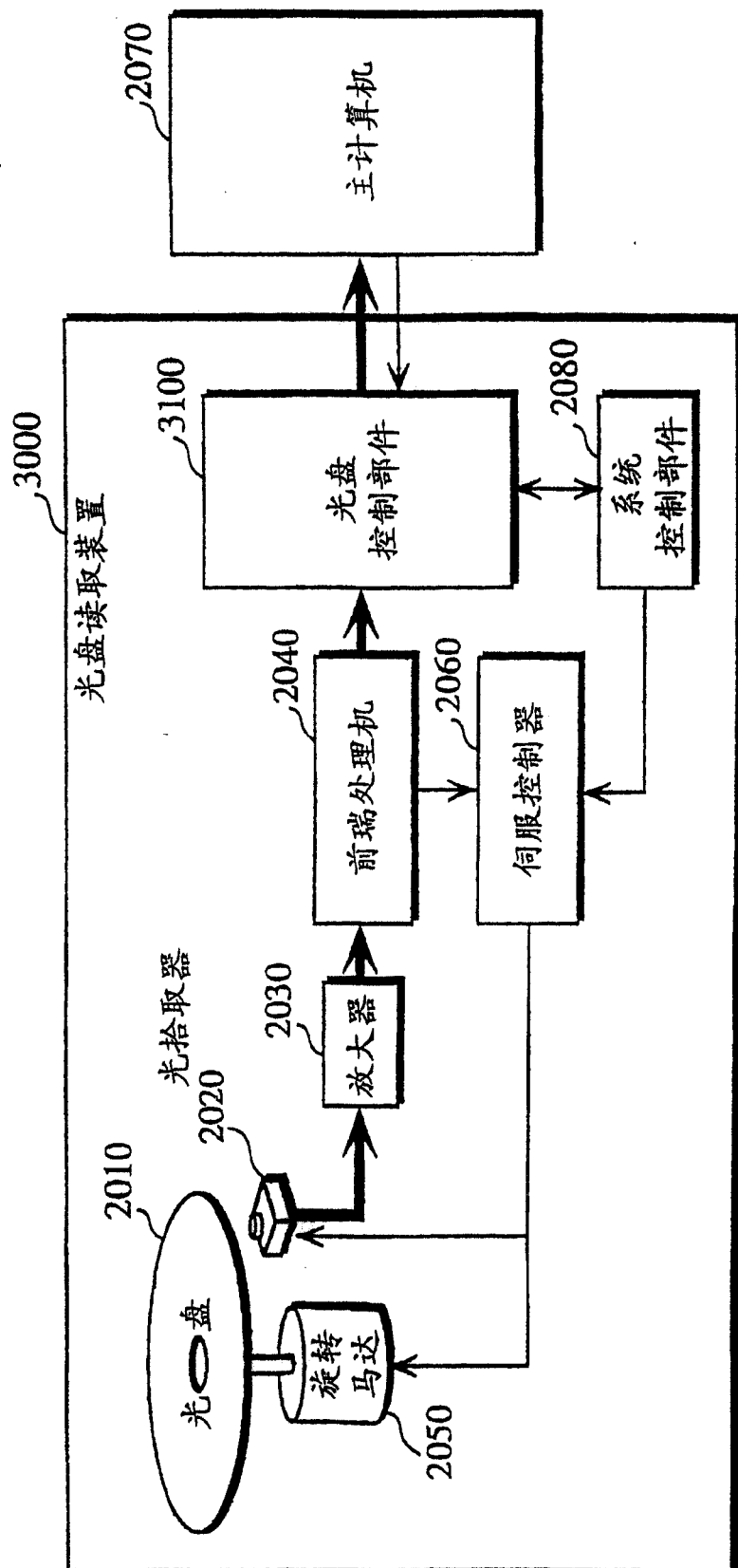


图 35

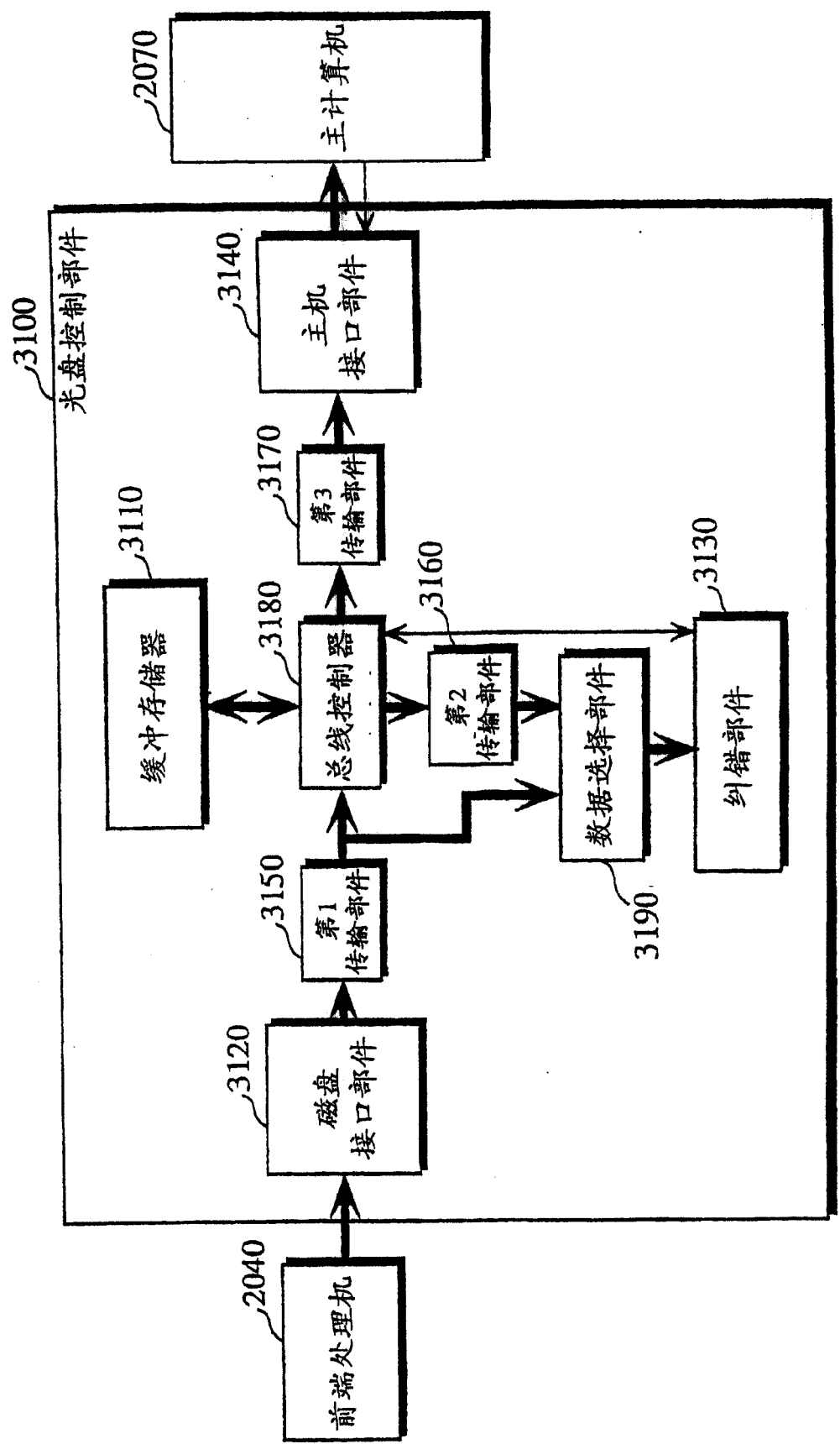


图 36

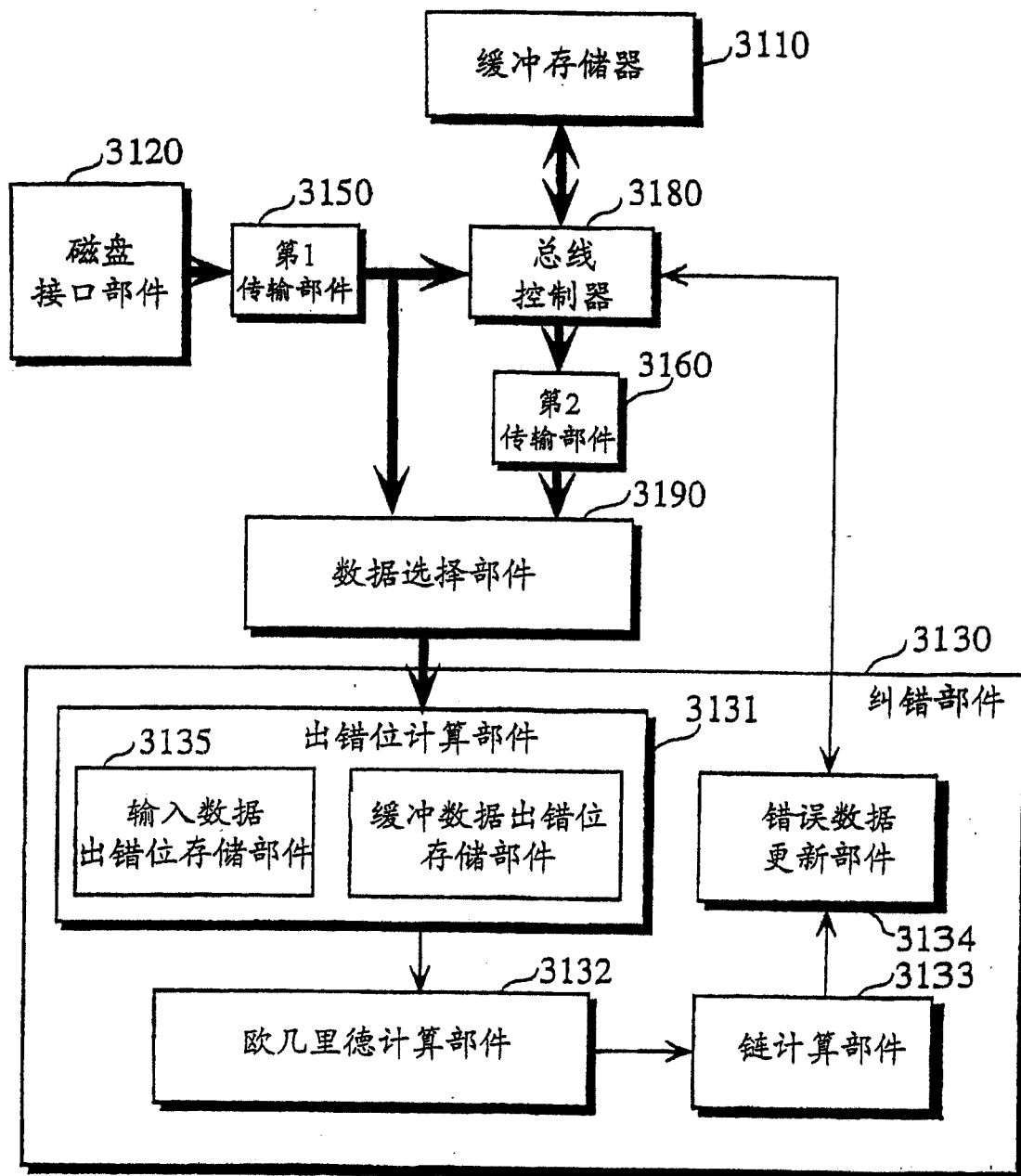


图 37

时间	T1	T2	T3	T4	T5	
缓冲						
	第1数据块	第2数据块	第3数据块	第4数据块	第5数据块	
纠错1						
	第1数据块	第2数据块	第3数据块	第4数据块	第5数据块	
纠错2						
			第1数据块	——	第4数据块	
主机传输						
			第1数据块	第2数据块		
					第3数据块	

图 38

时间	T1	T2	T3
缓冲			
	第1数据块	第2数据块	第3数据块
纠错1	第1数据块	第2数据块	第3数据块
	C1ECC	C1ECC	C1ECC
纠错2			
		第1数据块	第2数据块
		C2ECC	C2ECC
主机传输			
			第1数据块

图 39

时间	t1	t1	t1	t1	t5
C1 出错位计算	第2数据块 第1行	第2数据块 第2行	第2数据块 第3行	第2数据块 第4行	第2数据块 第5行
	第1数据块 第1列	第1数据块 第2列	第1数据块 第3列	第1数据块 第4列	第1数据块 第5列
C2 出错位计算					
欧几里德计算		第2数据块 第1行	第2数据块 第2行	第2数据块 第3行	第2数据块 第4行
		第1数据块 第1列	第1数据块 第2列	第1数据块 第3列	第1数据块 第4列
链计算		第2数据块 第1行	第1数据块 第1列	第1数据块 第2列	第1数据块 第3列
错误数据纠正		第2数据块 第1行	第2数据块 第1列	第2数据块 第2列	第2数据块 第3列

图 40