

文章编号: 1001-3806(2002)06-0471-03

光存储中 RS 纠错码两种实现算法的研究*

王 桦 熊剑平 魏轶伟 贾惠波
(清华大学精密仪器与机械学系, 北京, 100084)

摘要: 以光存储应用为背景, 针对盘片记录信息固有的高原生误码率问题, 从目前 CD, DVD 的纠错编码方案出发, 设计并实现一种合适的差错控制方法来保证数据读写的正确性。分析了实现 Reed Solomon(RS) 纠错编码的 Berlekamp Massey(BM) 算法和 Peterson Gorenstein-Zierler(PGZ) 算法的原理和各自的实现途径, 比较了这两种算法的优缺点, 得出了它们的不同应用条件。采用高速硬件描述语言(VHDL) 成功实现了 PGZ 算法, 并得到了理想的仿真结果。

关键词: 光盘存储; RS 编码; BM 算法; VHDL

中图分类号: TP301 文献标识码: A

The research of two error correcting algorithms on Reed Solomon code in optical storage

Wang Hua, Xiong Jianping, Wei Yiwei, Jia Huibo

(Department of Precision Instrument and Mechanics, Tsinghua University, Beijing, 100084)

Abstract: As the error probability is inherently high in optical storage, proper error correction coding methods must be adopted to ensure the validity of recorded data. Reed Solomon(RS) code is one of such codes that meet the requirements. In this paper, two different algorithms which implement RS code, Berlekamp Massey(BM) algorithm and Peterson Gorenstein Zierler(PGZ) algorithm are analyzed and compared under specific application conditions with respect to different error occurrence case. And finally PGZ algorithm is implemented with VHDL and the simulation result shows it is practical to be implemented in FPGA.

Key words: optical storage; RS code; BM algorithm; VHDL

引 言

光存储技术自从问世以来就以其大的数据存储量和较高的存储速度而显示出强大的生命力。近年来, 随着技术的不断发展, 可实现的数据存储量一直在不断地增加。鉴于光存储技术固有的高原生误码率, 需要高效快捷的差错控制方法来保证数据记录和读出的正确性。Reed-Solomon(RS) 码及其纠错算法因其在纠正突发错误和随机错误方面的突出能力而被应用于光存储技术中。在众多的 RS 纠错算法的实现方案中, Berlekamp Massey(BM) 算法和 Peterson Gorenstein Zierler(PGZ) 算法具有一定的代表性, 有其各自的实用场合。针对光存储的特点, 结合

CD, DVD 的纠错编码系统, 设计实现了基于 VHDL 语言的 PGZ 算法, 并进行了功能仿真。另外, 还对 BM 算法的原理和实现结构作了分析, 最后, 对 PGZ 和 BM 两种算法的应用比较进行了研究。

1 RS 纠错算法的原理及比较^{1,2]}

一个定义在 $GF(2^m)$ (伽罗瓦域) 上的可纠正 t 个错误的 RS 纠错编码有以下参数: 码字长为 n , 其中信息序列长为 k , 校验序列长为 $(n-k) = 2t$; n, k, t 的单位为符号, 一个符号由 m 位二进制码元组成。RS 码字可由 (n, k) 表示。光存储中主要应用的是 $GF(2^8)$ 上定义的 RS 码。如 CD 中采用的是 $(28, 24)$ ($n=28, k=24$) 和 $(32, 28)$ 两种 RS 码, DVD 中用的是 $(208, 192)$ 和 $(182, 172)$ 两种 RS 码, 码字中每个符号由 8 位二进制码元组成。

在接收数据的过程中, 设 R 为信宿实际收到的码矢量, V 为原始的正确码矢量, E 为传输过程中

* 国家重点基础研究 973 项目。

作者简介: 王 桦, 男, 1978 年出生。硕士研究生。现从事交通连续数据存储的设计与应用工作。

收稿日期: 2001-11-28; 收到修改稿日期: 2002-01-08

产生的错误码矢量,这3个码矢量之间存在如下的关系:

$$R = V + E \quad (1)$$

由(1)式得错误码矢量为: $E = R - V$ (2)

在模二的运算下, $E = R - V = R + V$ (3)

为了找出并修正错误,定义错误多项式为:

$$e(X) = e_0 + e_1X + \dots + e_{n-1}X^{n-1} \quad (4)$$

如果传输过程中长为 n 的 RS 码的第 i 位出现错误,则多项式中的系数 e_i 不为 0。现设传输过程中出现了 c 个错误 ($0 \leq c \leq t$), 它们发生在未知位置 i_0, i_1, \dots, i_{c-1} , 则错误多项式可以写为:

$$e(X) = e_{i_0}X^{i_0} + e_{i_1}X^{i_1} + \dots + e_{i_{c-1}}X^{i_{c-1}} \quad (5)$$

定义伴随多项式为:

$$S_i = r(\alpha^i) = v(\alpha^i) + e(\alpha^i) = e(\alpha^i) = e_{i_0}\alpha^{i_0 \times i} + e_{i_1}\alpha^{i_1 \times i} + \dots + e_{i_{c-1}}\alpha^{i_{c-1} \times i}, i = 0, \dots, 2t-1 \quad (6)$$

在错误个数小于 t 的情况下,通过求解 S_i 组成的方程组,可以得到错误的位置和值。直接求解这一方程组十分困难,为此采取特别的手段,引入错误位置多项式:

$$\Lambda(x) = (1 - xX_1)(1 - xX_2) \dots (1 - xX_c) = \Lambda_c x^c + \Lambda_{c-1} x^{c-1} + \dots + \Lambda_1 x + 1 \quad (7)$$

其根 X_1, X_2, \dots, X_c 为错误位置数的逆,可以通过 Chien 法求得错误位置多项式的根,得到错误的位置数,然后通过 Forney 法即可求得错值。整个纠错解码的理论流程见图 1。图中错误位置多项式 $\Lambda(x)$ 的计算是关键的一步,可以通过求解其系数 $\Lambda_j (j = 1 \dots c)$ 实现,这一计算过程比较复杂,受到参数 t 的影响,需要采用合适的算法。

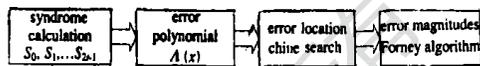


Fig. 1 RS decode flow

PGZ 算法将对系数 Λ_j 的求解转化为矩阵的运算,逻辑简单,运算量较小,易于实现。对于规模较小的 RS 码,在可纠正的错误个数 t 比较小,校验序列长度较低时,适合采用 PGZ 算法来进行系数的求解。PGZ 算法的原理如下。

由于系数 Λ_j 和伴随式因子 S_i 存在有(8)式描述的关系^[2]:

$$\Lambda_1 S_{j+c-1} + \Lambda_2 S_{j+c-2} + \dots + \Lambda_c S_j = -S_{j+c}, j = 1 \dots c \quad (8)$$

因此,可以通过以上关系建立系数和伴随式因子间的矩阵形式关系,如下式:

$$\begin{bmatrix} S_1 & S_2 & \dots & S_c \\ S_2 & S_3 & \dots & S_{c+1} \\ \vdots & \vdots & \vdots & \vdots \\ S_c & S_{c+1} & \dots & S_{2c-1} \end{bmatrix} \begin{bmatrix} \Lambda_c \\ \Lambda_{c-1} \\ \vdots \\ \Lambda_1 \end{bmatrix} = \begin{bmatrix} -S_{c+1} \\ -S_{c+2} \\ \vdots \\ -S_{2c} \end{bmatrix} \quad (9)$$

由此,(8)式的系数可以通过求解方程组的根获得,令(9)式系数矩阵 M 为:

$$M = \begin{bmatrix} S_1 & S_2 & \dots & S_c \\ S_2 & S_3 & \dots & S_{c+1} \\ \vdots & \vdots & \vdots & \vdots \\ S_c & S_{c+1} & \dots & S_{2c-1} \end{bmatrix}.$$

为了找出错误的实际个数,先假设 $c = t$, 计算 M 的行列式值,若非零则所选值正确,错误个数为 t ; 若 $\det(M)$ 为 0, 则将 c 的值减 1, 重复上面的计算和判断,如此进行下去直至找到错误个数,若 c 为 0 时 $\det(M)$ 仍然为 0, 则可判断错误个数多余 t , 无法纠正。找到实际的错误个数 c 后,解(9)式,即可得到系数 Λ_j 的值,然后对错误位置多项式进行 Chien 搜索和 Forney 求值即完成了纠错解码。这一算法逻辑分支少,复杂性较低,易于实现。但是由于算法中应用了矩阵求逆的运算,而对于 t 行 t 列的矩阵求逆运算的计算量正比于 t^3 , 在可纠正错误个数 t 增大时,矩阵运算的开销将会急剧增加;此外,当信道中的错误 $v < t$ 时,在求解 M^{-1} 的过程中,必须将 $t \times t$ 阶的 M 阵一直降到 $v \times v$ 阶,每次降阶都要检验 M 是否满秩,共需检验 $t - v + 1$ 次,实际错误越少,译码时间反而越长。因此,这一算法只能用于 $t < 5$ 的纠错运算^[1]。

对于中大规模的 RS 码,由于 t 较大,PGZ 算法已经不适用,而作为一种采用迭代方式求解错位多项式的方法,BM 算法在这种条件下具有优越性。根据(8)式,对一定的 Λ_j , 它是一个自回归滤波器的方程^[2]。在求得伴随式后,对错误位置多项式系数的计算可以通过迭代求解线性反馈移位寄存器的抽头来实现^[2]。算法的关键在于找出一个方法,以计算出能产生序列 $S_1 \dots S_c$ 的最短移位寄存器 ($L_r, \Lambda^{(r)}(x)$)。可以采用线性反馈移位寄存器来达到这一目的,在必要时修改它的长度和抽头加权值。首先初始化错值多项式为 $\Lambda(x) = 1$, 计数器 $r = 0$, 临时多项式 $B(x) = 1$, 寄存器长度为 $L = 0$, 根据(1)式,在第 r 次计算后,考察下一个校正子 $\sum_{j=1}^L \Lambda_j S_{r-j}$ 的差 $\Delta r = S_r + \sum_{j=1}^L \Lambda_j S_{r-j}$, 如果 $\Delta r = 0$, 则抽头 Λ_L 无误,当前的移位寄存器可以产生下一个校正子,如果 $\Delta r \neq 0$ 则需要更改抽头,计算新的连

接多项式 $T(x)$ 使 $\Delta r = 0$; 判断是否要加长移位寄存器, 如果需要加长则将 $T(x)$ 的值赋给 $\Lambda(x)$, 否则存储当前结果进行下一个校正子的计算, 如此循环。在 $r = 2t$ 时, 如果 $\Lambda(x)$ 的秩为 L , 则可开始 Chien 搜索, 否则错误个数多于 t 个, 无法纠错。BM 算法逻辑分支和参数较多, 比较复杂, 同时, 由于伽罗瓦域上的运算比较多, 实现起来比较困难。但这一算法不必求解矩阵的逆, 而是采用迭代的方式计算错值多项式, 因而, 特别适合于计算机运算和硬件的实现。采用迭代方式求解错位多项式所需时间随实际错误个数的变化而变化, 但是由于流程固定可以采用数学方法估计此时间。硬件构成的译码器的复杂性和译码所需时间大致正比于 nm 和 $n \log_2 n^{[2]}$ 。

2 算法的应用

在 CD 类 (CD, CD-ROM, CD-R 等) 光盘存储设备中采用的是交错交织里德-所罗门 (CIRC) 纠错方案, 原理见图 2。



Fig. 2 CIRC decode sequence

其中两级纠错环节采用 $GF(2^8)$ 上的 (28, 24) 和 (32, 28) RS 码, 每级可纠错误个数 $t = 2$, 属于小规模差错控制, 适合采用 PGZ 算法来进行纠错。而高速硬件描述语言 VHDL 是一种应用于电路设计的高层次描述语言, 具有行为级, 寄存器传输级和门级等多种描述层次, 并且具有简单、易读、易修改、与工艺无关等优点, 可在可编程逻辑器件或 ASIC 上实现。故笔者采用 VHDL 语言对该算法进行行为级的设计。由于 RS 码的译码运算属于有限域上的运算, 与 VHDL 标准算术运算包中定义的加法和乘法不同, 因此, 为有限域上的加法、乘法和符号求逆编写了相应的 VHDL 包^[3]。对于加法, 将符号以二进制的形式表示, 两个符号之间的加法通过异或 (XOR) 运算实现, 运算结果以二进制的形式存储至 RAM 中; 对于乘法, 将符号以伽罗瓦域中的幂的形式表示, 符号间的乘法通过幂之间的加法来完成, 运算结果在模 255 后通过查 (ROM) 表转换成二进制的形式存储; 而有限域的除法则可转化成乘法来做。为此, 笔者编写了用于存储运算临时结果的 RAM 和存储伽罗瓦域中幂与符号对应关系的 ROM 的 VHDL 模型以及相应的控制逻辑。此外, 由于伴随式的计算的输出是错位多项式计算的输入; 而错位

多项式输出恰为 Chien 搜索算法求错位多项式的根; 而求出的错位多项式的根恰可用于求错值, 这 4 个过程之间没有反馈, 因此, 可以采用流水线的结构加以实现。这样, 可以大大提高译码器的译码速度和减少译码的延迟。流水线方式通过控制状态机对各个译码流程的控制得以实现。最终, 以 Aldec 公司的 Active HDL 为设计输入和仿真工具, 将 PGZ 算法在计算机上仿真并得到了正确结果, 其中, (28, 24) RS 码通过译码器后的结果见图 3。

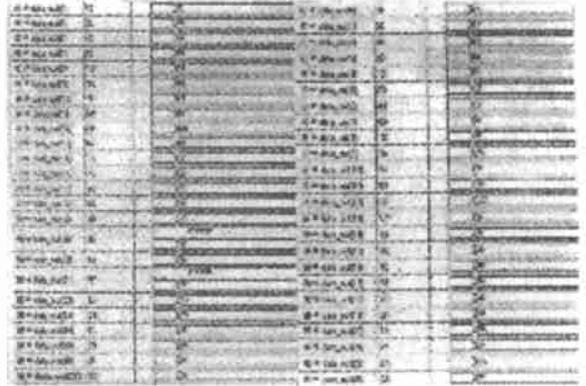


Fig. 3 Behavioral simulation result

左图为原始输入, 其中第 19 和 21 符号为错误符号, 右图为输出结果, 可以看到输出恢复为 24 个符号, 而且错误被纠正, 冗余校验位也被去除。此外, 在 CIRC 纠错方案中还采用了交错环节, 可以将连续两帧的错误分散到多个帧中, 提高了纠错能力。

和 CD 类的编译码方案相比, DVD 中应用了乘积码, 其编码也是两级结构, 对一帧数据进行行和列的 RS 编/解码。其中行和列的 RS 码分别为 (208, 192) 和 (182, 172)。两级可纠错误个数分别为 8 个和 5 个, 这种情况下 t 较大, 如果仍然采用 PGZ 算法, 会使 (2) 式的系数矩阵 M 的阶数增加, 又因高阶矩阵的运算极为复杂, 因此, 这一算法在软件实现角度和硬件实现角度上都是不合适的。而 BM 算法采用迭代的方法来寻找错误的个数和错误多项式, 比较合适。这一算法可以采用移位寄存器循环迭代实现, 具体实现的结构见图 4^[4, 5]。

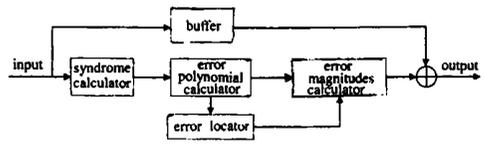


Fig. 4 Decoder structural diagram

由于 BM 算法采用的是有反馈的移位寄存器结构, 对于电路的时钟的上限有要求。在硬件实现的过程中需要注意。
(下转第 477 页)

溶解,端部钝化,由残余奥氏体包围。这种组织在磨损初期石墨脱落,起贮油润滑作用。形成的孔洞虽会产生应力集中,但程度减轻,周围奥氏体有较好塑性变形以缓解应力集中,因而延长了疲劳裂纹的产生和扩展,提高了疲劳寿命。微熔时表面粗糙度增加不大,其熔化层深度小于 0.1mm,是值得推荐的一种激光强化方法。

4 结 语

(1)HT300 孕育铸铁的激光表面强化方法主要有相变硬化处理及熔凝处理,各有其特点,可视具体情况选取,以满足不同的凸轮轴工作表面的要求。采用激光相变硬化处理,所得金相组织为马氏体,可以保证在其尺寸精度不变的条件下达到 600HV_{0.1}显微硬度。采用激光熔凝处理,所得金相组织为细枝晶莱氏体,硬度达到 959HV_{0.1}显微硬度。

(上接第 473 页)

在 DVD 纠错编码系统中通过对一维的纠错进行乘积操作将之扩展为多维纠错码^[2],而且无论是逐行发送还是逐列发送对原始码都是一种交错,大大提高了系统的纠错能力。经计算,这种 DVD 纠错编码约可以纠正 80 个码字的随机错误和 80 个码字的突发错误^[1]。

3 结 论

通过对 PGZ 算法和 BM 算法原理的分析、比较,得到了这两种算法各自的应用特点,说明了针对不同光存储方案的这两种算法各自适用的场合。PGZ 算法适合用于纠错个数小于 5 的系统中,BM 算法适合用于纠错个数较多的系统中。本文中设计

(2)经激光处理后的凸轮轴的硬度和耐磨性都较激冷处理有大幅度的提高,完全可以用于实际工作,文中所述的激光工艺参数为工厂改进凸轮轴处理方法提供依据。

参 考 文 献

- [1] 王中宝. 摩托车技术, 2000(8): 10~ 14.
- [2] 杜 军, 刘耀辉. 汽车工艺与材料, 1999(3): 1~ 4.
- [3] 关振中. 激光加工工艺手册. 北京: 中国计量出版社, 1998.
- [4] Mazumder J. Metals, 1983, 35(5): 18~ 26.
- [5] 赵 新, 金 杰, 丁 欣 *et al.* 天津大学学报, 1999, 32(6): 754~ 757.
- [6] 茆诗松. 回归分析及试验设计. 上海: 华东师范大学出版社, 1981.
- [7] 王晓江. 铸造合金及其熔炼. 北京: 机械工业出版社, 1999.
- [8] 刘江龙, 邹至荣, 苏宝蓉. 高能束加热处理. 北京: 机械工业出版社, 1997.
- [9] 王大承. 金属热处理, 2001(11): 23~ 25.

的 PGZ 算法的 VHDL 仿真验证了其设计的正确性。可看出, 交错技术在一定条件下可将错误分散至多个帧, 从而降低了每一帧中的错误个数, 对纠错系统的设计具有重要的意义。

参 考 文 献

- [1] 林 舒, 科斯特洛. 差错控制编码基础和应用. 北京: 人民邮电出版社, 1986.
- [2] Blahut R E. 差错控制码的理论. 广州: 华南理工大学出版社, 1990.
- [3] Perry D L. 电子设计硬件描述语言 VHDL. 北京: 北京学苑出版社, 1994.
- [4] Smith S, Taylor D, Benaissa M. Microelectronics Journal, 1998, 29: 977~ 982.
- [5] Chang H C, Shuang C. A Reed Solomon Product Code(RS PC) Decoder for DVD Applications. ISSCC. Disk Drive Signal Processing. USA: ISSCC, 1998.