

文章编号 1004-924X(2012)05-1090-12

## 空间图像存储器 NAND Flash 的可靠性

李 进<sup>1,2\*</sup>, 金龙旭<sup>1</sup>, 韩双丽<sup>1</sup>, 李国宁<sup>1</sup>, 王文华<sup>1</sup>

(1. 中国科学院 长春光学精密机械与物理研究所, 吉林 长春 130033)

(2. 中国科学院 研究生院, 北京 100039)

**摘要:**针对空间相机中的图像存储器 NAND Flash 由于坏块和单粒子翻转导致存储数据不可靠的问题,研究了 Flash 坏块的管理策略和纠错算法。分析了 Flash 结构和工作特点,提出了基于并行双遍历机制的坏块管理策略,阐述了双遍历机制的设计思想并分析了它的有效性。在分析 Flash 结构和纠错特点的基础上,提出了在域  $GF(2^8)$  上的缩短码 RS (246, 240)+RS(134, 128) 纠错算法,并说明了编解码算法思想和实现电路。最后,在一空间多光谱相机样机的图像存储设备上进行了试验验证。结果表明,管理策略能快速可靠地处理坏块事件,每次操作仅需 1 个系统时钟周期即可完成坏块判断。纠错算法在 2 KB/page 内可以纠正 27 B 错误,编码速度达到 72.53 MBps,解码器速度达到 54.26 MBps。提出的管理策略和纠错算法有效地解决了 Flash 数据存储的不可靠问题。

**关键词:**空间相机;图像存储器;坏块管理;单粒子翻转;并行双遍历机制;级联 LC;RS 码

**中图分类号:**TP333.5;TP391 **文献标识码:**A **doi:**10.3788/OPE.20122005.1090

## Reliability of space image recorder based on NAND flash memory

LI Jin<sup>1,2\*</sup>, JIN Long-xu<sup>1</sup>, HAN Shuang-li<sup>1</sup>, LI Guo-ning<sup>1</sup>, WANG Wen-hua<sup>1</sup>

(1. Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China)

(2. Graduate University of Chinese Academy of Sciences, Beijing 100039, China)

\* Corresponding author, E-mail: 664910699@qq.com

**Abstract:** For the unreliable data storage problem caused by bad blocks and single event upsets for the NAND flash memory in a space camera, this paper explores a bad block management strategy and an error correction algorithm. Firstly, the bad block management strategy based on parallel double-traverse mechanism was proposed by analyzing the characteristics of structure and operation for the NAND flash memory, the design ideas of the double traverse mechanism were described and its effectiveness was analyzed. Then, the error correction algorithm based on the shortened code RS (246, 240) and RS (134, 128) in the field  $GF(2^8)$  was proposed, and the encode/decode algorithm and corresponding circuits were given. Finally, the verification experiments on an image storage platform in the prototype machine for a space multi-spectral camera were carried out. The experimental results

收稿日期:2011-10-18;修订日期:2011-11-18.

基金项目:国家 863 高技术研究发展计划资助项目(No. 863-2-5-1-13B)

show that the bad block management strategy can fast and reliably dispose the bad block events, and the algorithm can identify the bad blocks in one system clock period. The error correction algorithm can correct 27 B error within the 2 KB/page with a encoder speed of 72.53 MBps and a decoder speed of 54.26 MBps. Proposed stratege effectively solve the problem of unreliable recording data in the NAND flash memory.

**Key words:** space camera; image memory; bad block management; single event upset; parallel double-traverse mechanism; cascade LC; RS code

### 1 引 言

随着基于 CCD 成像技术的空间相机在视场和分辨率指标要求上的不断提高,所采用的 CCD 拼接片数和读出速率也不断增多和提高,从而使数字化后的图像数据量大幅增加,国内现有图像存储器已经无法满足需要,必须开发适应空间相机的高速大容量图像存储器<sup>[1-3]</sup>。作为存储介质 NAND Flash(以下简称 Flash),由于具有高速存取、体积小、功耗低、轻便、抗震、抗冲击、温度适应范围宽、非易失、容量大等特点,而受到了国外航天领域研究人员的广泛关注。20 世纪 90 年代中期,Fairchild 公司为 F-16 侦察装置设计的 SSR(Solid State Recorder),使用的主要存储芯片就是 Flash<sup>[4]</sup>。然而,Flash 也有其自身的缺点:(1)由于工艺原因在生产及使用过程中会产生坏块;(2)由于存储单元问题使用时可能有个别比特翻转以及空间环境的辐射也可能导致存储器翻转,

即单粒子翻转导致差错。因此,如何管理坏块和纠错编码是图像数据可靠存储的关键问题。

本文在参考国内外相关技术的基础上,根据背景项目需求,结合 Flash 结构和工作特点提出了基于并行双遍历机制的坏块管理策略和 RS(246,240)+RS(134,128) Flash 纠错算法。其中,提出的并行坏块管理思想和级联 LC 坏块遍历机制,突破了传统的慢速坏块查询方法;提出的 RS(246,240)+RS(134,128) Flash 纠错算法突破了传统的 Flash 比特层次的纠错,在字节层次上进行纠错,大大提高了纠错能力和效率。本文提出的这些技术已经在某空间多光谱相机样机的图像存储设备上获得了应用。

### 2 Flash 结构特点

Flash 采用块页方式组织,以三星 Flash K9WAG08U1A 为例,其功能和结构图如图 1 所示。

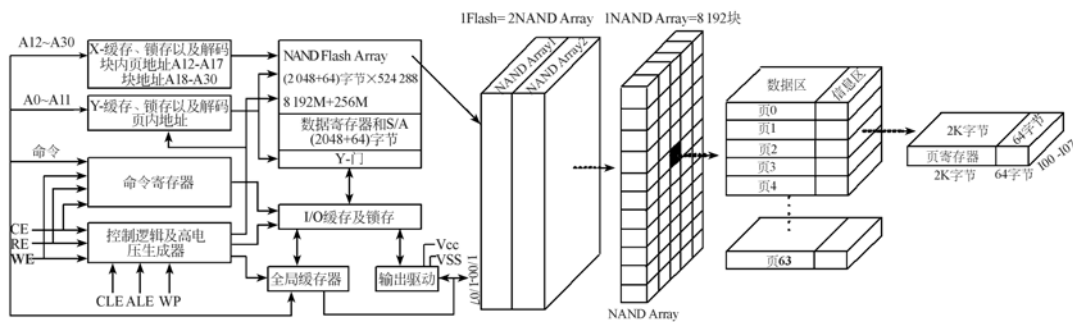


图 1 Flash 结构图

Fig. 1 Flash array organization

每 2 K+64 B 被组织成一个页,其中 2 KB 用于存放数据,64 B 用于存放坏块标记和 ECC(Error Correcting Code)校验等信息。每 64 个页被组织成一个块。命令、地址和数据通过 8 位 I/O

总线时分复用串行读写。读写操作是以页为单位,擦除操作是以块为单位。页内数据不能被反复写入,只有当包含该页的块被擦除后才能重新写入。Flash 的工艺不能保证存储阵列在其生命

周期中保持性能的可靠,使得它存在两类坏块:一类为固有坏块。它是在生产过程中产生的,在出厂时原厂在坏块的信息区进行了标记。二类为使用坏块。使用坏页/块定义<sup>[5-9]</sup>为:编程/擦除某页/块时,不能将其中某些位拉高,造成页编程或块擦除错误,相应的反映到状态寄存器的状态位为 1,该页/块为坏块或坏页。读取某页时,如果取出的数据受单粒子翻转现象导致的错误数超过了 ECC 算法的纠错能力时,该页为坏页。

为了可靠地存储数据,国内外很多学者在不同的应用系统中对坏块管理进行了大量的研究,提出了很多坏块管理策略。文献[7]针对 Flash 随机修改以及随机擦除的应用系统采用虚拟映射、动态调度等策略,写入数据时要进行空间分配,擦除时要进行垃圾回收。但这种方法过于复杂,不适合空间相机多级流水线的高速图像读写操作要求。文献[8]提出一种坏块替换策略,遇到写入坏块时将整个已经写入该块的数据拷贝到下一个好块中,从下一好块开始写数据,但数据拷贝耗费大量的时间且浪费了大量的存储空间。文献[9]提出一种基于 CAM 的块信息快速匹配检测机制。遇到写入坏块时,从下一块首页开始存储数据。但操作前仍花费匹配时间且实现 CAM 存储需占 FPGA 的大量资源,另外,存储空间利用效率也不高。

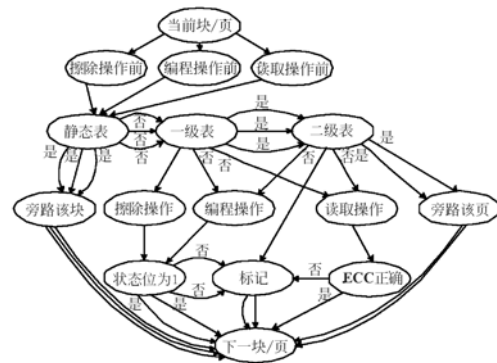
权衡速度、空间和占用资源,本文提出一种基于并行双遍历机制的坏块管理策略。不仅使坏块管理模块不影响 Flash 操作,而且提高存储空间利用效率,节省现场可编程门阵列(FPGA)占用资源。

### 3 坏块管理关键技术

#### 3.1 并行双遍历机制坏块管理

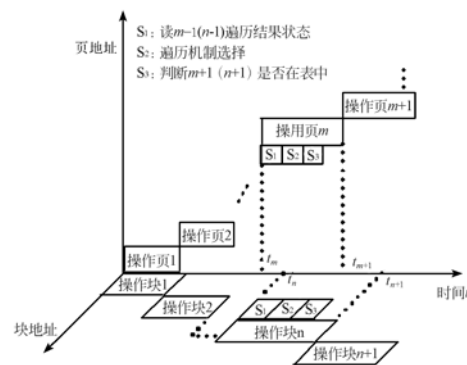
坏块管理过程是围绕擦除、编程和读取操作进行的。整个坏块管理过程分为:系统上电初始化、操作前、操作过程和操作结束 4 个阶段。由于空间相机图像存储器的工作特点是:随时添加记录内容和定期擦除,所以,Flash 的坏块划分为:静态坏块和动态坏块。静态坏块为固有坏块和擦除操作新增坏块,静态表存储静态坏块信息。动态坏块为编程操作新增的坏块,动态表存储动态坏块信息。静态表和动态表详见本文 3.2 和 3.3 节。Flash 操作

和表的遍历工作关系如图 2(a)所示,定期擦除操作时,擦除每块前先遍历静态表,若该块地址在静态表中,则直接旁路该块,擦除下一块;若不在静态表中,则遍历动态表的一级表。若不在一级表中,直接擦除该块;若在一级表中,执行完擦除操作后,把二级表中相应信息导入到坏页信息区中,以便下次系统上电建立动态表。当擦除操作结束后读取 Flash 状态寄存器中状态位的值,若为 0 表示擦除成功,执行下一块擦除操作;若为 1 表示该块为坏块,在该块第一页的信息区第 1 字节标记该块。编程操作时,在块首页写入前,先遍历静态表,若在表中直接旁路该块;若不在该表中,遍历动态表。遍历动态表时,先遍历一级表,若不在一级表中,直接编程每页;若在一级表中,编程每页前遍历二级表。若在二级表中,旁路该页;若不在二级表中,执行编程操作。编程操作结束后读取寄存



(a) 坏块划分

(a) Classification of bad blocks



(b) 并行遍历机制

(b) Parallel traverse mechanism

图 2 坏块划分与并行遍历

Fig. 2 Classification of bad blocks and parallel traverse



级联 LC 遍历机制,通过 17 个时钟周期就可以得到静态表遍历的结果,而 Flash 擦除、编程和读取时间<sup>[5,10]</sup>远超过静态表的遍历时间。可见,级联 LC 遍历机制完全满足并行遍历静态表的要求。

### 3.3 动态表遍历机制

动态表不仅存放读写操作中出现的坏页信息

而且还存放坏页所在的块信息,动态表结构如图 4 所示。擦除操作时需遍历动态表的一级表,操作结束后需把二级表相应坏页信息导入信息区。读写每块首页时遍历一级表,若不在一级表中,以后读写该块的每页均不需遍历二级表;若在一级表中需每页读写时遍历二级表。

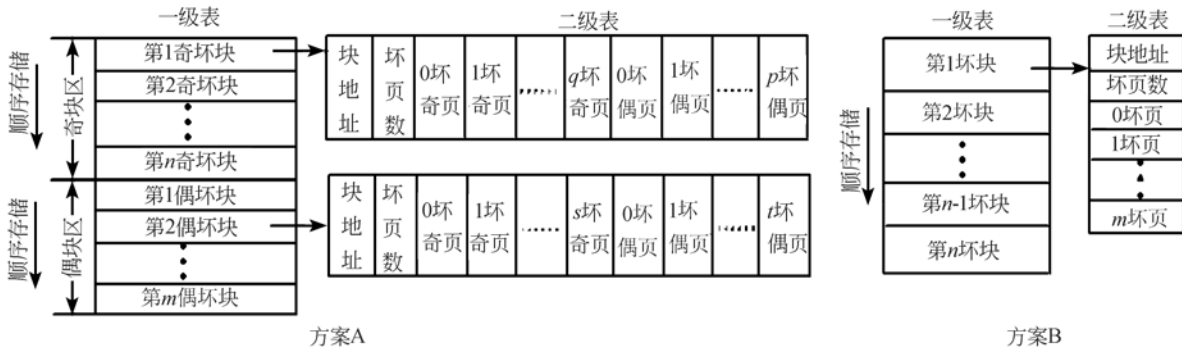


图 4 动态表结构

Fig. 4 Structure of dynamic table

由于二级表的存在,不能采用 LC 级联法,这时遍历动态表有两种方案,如图 4 所示。方案 B 把坏块信息顺序存储到静态表中。方案 A 把坏块分为奇块和偶块,奇块存储到低地址奇数区,偶块存储到高地址偶数区。设待操作的块地址为  $2N$ ,方案 A 偶坏块数为  $L_2$ ,奇坏块数为  $L_2 + 1$ ,查找成功平均比较长度为  $ASL_2$ 。方案 B 坏块数为  $L_1$ ,查找成功平均比较长度为  $ASL_1$ 。其中:

$$ASL_1 = \sum_{i=1}^{L_1} \frac{c_i}{L_1} = \sum_{i=1}^{L_2+1} \frac{c_i}{L_2+1} + \sum_{i=1}^{L_2} \frac{c_i}{L_2}, \quad (1)$$

$$ASL_2 = \sum_{i=1}^{L_2} \frac{c_i}{L_2}, \quad (2)$$

可见,方案 A 比方案 B 平均查找长度要小,故选择方案 A。

对于动态表的奇数区和偶数区目前搜索算法有很多种,其中二分搜索算法的平均工作量最小<sup>[11]</sup>,且二分搜索算法非常适合 FPGA 硬件语言实现。因此,在遍历动态表时先对待遍历的块或页信息进行模 2 运算,以确定在奇数区还是偶数区遍历。在确定的区间内采用二分搜索法。二分搜索法可参考文献[11],这里不再赘述。本文把动态表分区二分搜索法记做二\_二分遍历。

二\_二分遍历机制损耗时间取决于坏块数或坏页数。设坏块数为  $M$ ,系统时钟周期为  $T$ 。以

闪存 K9WAG08U1A 进行擦除操作为例,遍历动态一级表的最坏比较次数为  $\log_2 M$ ,遍历表所花费时间为  $T_{find}$ ,擦除操作需时间为  $T_{erase}$ ,其中:

$$T_{find} = T \times \log_2 M, \quad (3)$$

$$T_{erase} = 7 T_{rw} \times T_{BERS} \cong 1\ 764\ 687 T, \quad (4)$$

为了在擦除操作同时遍历完表,要求

$$(T_{find})_{max} = 1\ 764\ 687 T, \quad (5)$$

求得  $M_{max} = 2^{1\ 764\ 687}$ ,实际的总块数<sup>[5-10]</sup>目要远远小于  $M_{max}$ ,所以该遍历机制满足并行工作要求。

## 4 Flash 纠错算法

### 4.1 Flash 纠错编码算法

Flash 读取数据出现单粒子翻转现象时,如果没有纠错编码,将导致数据差错和存储空间利用率降低。为了提高数据读取的可靠性,本文提出一种基于 RS(246, 240) + RS(134, 128) 码的 Flash 纠错算法。

通常 RS 码<sup>[12]</sup>定义为:在有限域  $GF(q)$  ( $q \neq 2$ ) 上,码长  $n = q - 1$  的本原 BCH 码,其码元符号与生成多项式的根都是  $GF(q)$  上的元素,具有以下特性:

$$n = 2^m - 1, n - k = 2t, d_{min} = 2t + 1, \quad (6)$$

式中,  $m$  为符号比特数;  $n$  为 RS 码长;  $k$  为信息长度;  $t$  为可纠错码元数;  $d_{min}$  最小码距离。此 RS 码

是 1 个可在  $n$  码字上纠正  $t$  码字或更少错误的 RS 码,如  $m=10, n=2^{10}-1=1\ 023$ 。设需纠错码数  $t=4$ ,根据  $1\ 023-k=8$ ,可得  $k=1\ 015, d_{\min}=9$ ,即可得到一个可以纠 4 码元错的 RS(1 023, 1 015)码。

根据 Flash(以 K9WAG08U1A 为例)特点及纠错要求,将 Flash 每页的数据区分为 9 块,前 1 920 B 分为 8 块,每块为 240 B,剩余 128 B 作为 1 块。信息区 10 B 用于存放坏块标记和其他信息,剩余 54 B 存 9 块的校验码。所以设计  $t_1=3$ ,待编码元  $k_1=240$ ,总长  $n_1=246$  的 RS(246,240) 码,以及  $t_2=3$ ,待编码元  $k_2=128$ ,总长  $n_2=134$  的 RS(134,128) 码。由于 RS(246,240) 和 RS(134,128) 码长  $2^7-1 < n_1, n_2 < 2^8-1$ ,所以,RS(246,240) 和 RS(134,128) 其实是有限域  $GF(2^8)$  上标准码 RS(255,249) 的缩短码。本文采用添 0 和去 0 的办法使其变为标准形式的 RS 码。这时,在设计编码器时,当输入 240 B 和 128 B 数据时,编码器会自动分别补 9 B 和 121 B 的 0,将长度转为 249,然后生成 6 B 的校验码元和 255 B 的 RS(255,249) 码。当编码结果输出时,在将所加入的 0 去掉,变成 RS(246,240) 和 RS(134,128) 码输出,然后存入 Flash 中。译码时,把原始数据和校验码一并读出组合输入译码器,译码器也会自动填补 0,译码输出结果后,去掉填补的 0 即可。

RS(255,249) 编码算法思想如下:

(1) 由有限域性质<sup>[13]</sup>求得  $GF(2^8)$  上本原多项式

$$F(x) = x^8 + x^4 + x^2 + 1, \quad (7)$$

对于有限域  $GF(2^8)$  中的每一个元素都可用它的一组自然基底  $\{1, \alpha, \alpha^2, \dots, \alpha^6, \alpha^7\}$  表示,其中,  $\alpha$  是本原域元素。有  $F(\alpha) = 0$ , 得到

$$\alpha^8 = \alpha^4 + \alpha^2 + 1, \quad (8)$$

由式(7)求得  $GF(2^8)$  域的全部元素编码表。

(2) 生成多项式为

$$g(x) = \prod_{i=1}^6 (x - \alpha^i) = \sum_{i=1}^6 g_i x^i, \quad (9)$$

由式(8)和编码表得出生成多项式

$$g(x) = x^6 + 126x^5 + \dots + 49x + 117. \quad (10)$$

(3) 设待输入编码器的码字多项式为  $D(x)$  (共  $k$  位),产生的校验元为  $R(x)$  (共  $n-k=6$  位),则编码后码多项式  $C(x)$  可表示为  $C(x) =$

$x^{n-k}D(x) + R(x)$ , 求出  $R(x)$  便完成了编码。由于码多项式  $C(x)$  是  $g(x)$  的倍式,易得  $R(x) = x^{n-k}D(x) \bmod g(x)$ 。

### 4.2 RS(255,249) 编码算法实现

RS(255,249) 编码中多项式运算是有限域  $GF(2^8)$  上的算术运算,这些多项式可以分别使用有限域加法器和乘法器实现。设计的编码电路如图 5 所示,初始时, Gate 选通,送入信息组  $D(x)$  的系数,并进行移位,高次位系数首先进入电路,它一方面直接经或门输出,另一方面乘以  $x^{n-k}$  次进入除法电路并进行移位,从而完成了  $x^{n-k}D(x)$  的作用。K 次移位后  $D(x)$  全部送入电路,完成了除法作用。此时移位寄存器中保留的是校验码元。Gate 断开,经过  $n-k$  次移位后,校验码元全部输出,与最初的  $k$  位信息组成一个长为  $n$  的码字。

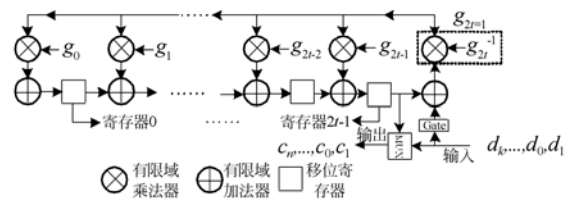


图 5 编码电路

Fig. 5 Encode circuit

### 4.3 Flash 纠错译码算法

RS 译码算法主要包括伴随式计算、计算错误位置和错误值多项式、钱氏搜索计算错误位置、Forney 算法计算错误值、模二和计算解码输出 5 个主要部分,其整体结构图参考文献[14],这里不再赘述。RS(255,249) 译码算法思想如下:

(1) 由输入译码器的  $R(x)$  求伴随式  $S(x)$ 。

设  $R(x) = \sum_{i=1}^{255} c_{255-i} x^{255-i}$ , 伴随矩阵  $S = RH^T$ , 其中,  $H$  为校验矩阵。得到伴随式系数

$$s_k = \sum_{i=0}^{254} c_i \alpha^{ik}, k = 1, 2, \dots, 6, \quad (11)$$

若采用式(11)计算任意  $s_k$  要进行 254 次加法和  $255 \times (255-1)/2 = 32\ 385$  次乘法,计算量较大。本文采用一种改进办法即将式(11)转换为

$$s_k = \{ \dots [(c_{n-1} \alpha^k + c_{n-2}) \alpha^k + c_{n-3}] \alpha^k + \dots \} \alpha^k + c_0, \quad (12)$$

则乘法运算减为 254 次。设计的运算电路如图 6 所示,在第一个时钟周期,输入码元  $c_{n-1}$ ,乘以  $\alpha$ ,

存入寄存器 dff1 中,在下一个时钟周期加上码元  $c_{m-2}$ ,然后再乘以  $\alpha$ ,再存入 dff1 中。依次类推,当 255 个时钟周期过后,得到  $S_1$ 。依次类推,求出所有伴随式。

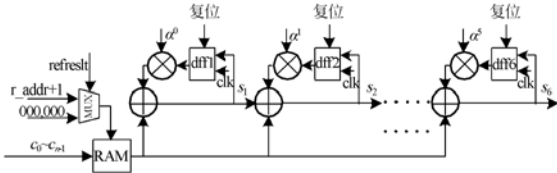


图 6 伴随式计算电路

Fig. 6 Circuit for calculating polynomial coefficients

(2)由伴随式求错误位置多项式  $\sigma(x)$  和错误值多项式  $\omega(x)$ 。目前比较成熟的方法有:BM 算法<sup>[15]</sup>、Euclid 算法<sup>[16]</sup>和 PGZ 算法<sup>[17]</sup>。PGZ 算法无快速递归法,不易于硬件实现。Euclid 算法需要保存大量的中间量,占用大量存储空间。BM 算法是迭代算法,极大地加快了求  $\sigma(x)$  的速度,且易于用 FPGA 实现。权衡考虑速度和资源问题,本文采用 BM 算法并对其进行改进。

在 BM 算法中要用到有限域元素求逆运算,会大大增加逻辑资源用量,并使系统速度受限。所以对该算法进行改进。具体改进如下:

初始化:  $\delta^{(0)}(x) = 1, \omega^{(0)} = 1, l^{(0)} = 0, \gamma^{(j)} = 1$

且  $j > 0$  开始迭代;

循环迭代:定义

$$\delta^{(j+1)}(x) = \gamma^{(j)} \delta^{(j)}(x) - d^{(j+1)} \omega^{(j)} x, \quad (13)$$

其中,  $d^{(j+1)}(x) = \sum_{i=1}^l \delta_i^j S_{m+j-1}$

$$\omega^{(j+1)}(x) = \begin{cases} x\omega^{(j)}(x) & d^{(j+1)} \neq 0, 2l^{(j)} > j \\ \delta^{(j)}(x) & d^{(j+1)} \neq 0, 2l^{(j)} \leq j \end{cases}, \quad (14)$$

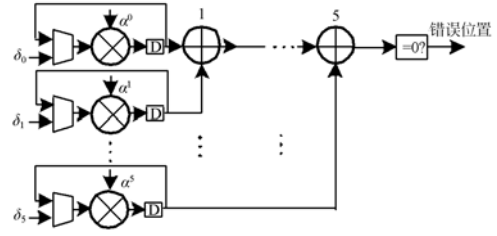
$$l^{(j+1)}(x) = \begin{cases} l^{(j)}(x) & d^{(j+1)} \neq 0, 2l^{(j)} > j \\ x+1-l^{(j)} & d^{(j+1)} \neq 0, 2l^{(j)} \leq j \end{cases}, \quad (15)$$

$$\gamma^{(j+1)}(x) = \begin{cases} \gamma^{(j)} & d^{(j+1)} \neq 0, 2l^{(j)} > j \\ d^{(j+1)} & d^{(j+1)} \neq 0, 2l^{(j)} \leq j \end{cases}, \quad (16)$$

以上循环迭代经过  $2t(t=3)$  次迭代后得到  $\delta^{(2t)}(x)$  和  $\omega^{(2t)}(x)$ ,即为所求的  $\delta(x)$  和  $\omega(x)$ 。改进的 BM 算法中没有用到有限域求逆运算,这样就能简洁快速地完成求解运算,能减少大量的时钟周期。

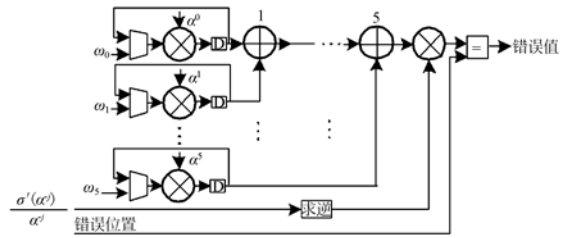
(3)确定错误位置数和错误位置。通过计算

$\sigma(x)$  在  $GF(2^8)$  中所有元素的值从而找出错误位置多项式的根,即依次将  $\alpha^i(0 \leq i \leq 254)$  代入  $\sigma(x)$ 。若  $\sigma(\alpha^i) = 0$ ,则第  $i$  位出现误码,  $\alpha^{255-i}$  为错误位置。设计的电路如图 7(a) 所示。



(a) 错误位置

(a) Error location



(b) 错误值

(b) Error value

图 7 求解错误位置和错误值电路

Fig. 7 Circuit for error location and value

(4)将错误位置  $\alpha^{255-i}$  和错误估值多项式  $\omega(x)$  代入 Forney 公式求出错码值  $e_i$ 。Forney 公式<sup>[14]</sup>为

$$\omega^{(j+1)}(x) = \frac{\omega(X_j^{-1})}{\delta'(X_j^{-1})}, \quad (17)$$

式中:  $\delta'(X_j^{-1})$  是由  $\delta(x)$  当  $x = X_j^{-1}$  且偶数次幂置 0 所得。设计的错误值计算电路如图 7(b) 所示。计算出错误位置和错误值之后,错误值与对应错误位置的码元异或便可得到纠错之后的码元。

## 5 实验验证与结果分析

### 5.1 实验方案

本实验系统来自笔者设计的某空间多光谱相机样机的图像存储设备。闪存选用三星公司的 K9WAG08U1A,单片容量 2 GB。系统如图 8 所示,系统工作过程是:差分接收模块收到的图像数据经重组后为 16 b。重组后的 16 b 数据通过乒乓操作的 1# SDRAM 和 2# SDRAM 进行 RS 编

码。然后在坏块管理下 Flash 以 4 片组成 8 级流水线并行写入数据。当 PC 机通过 RS232 发来读取指令时,在坏块管理机制下读取 Flash 数据,经

过 RS 译码后获得图像数据,经 Camera Link 传到采集卡并显示图像。

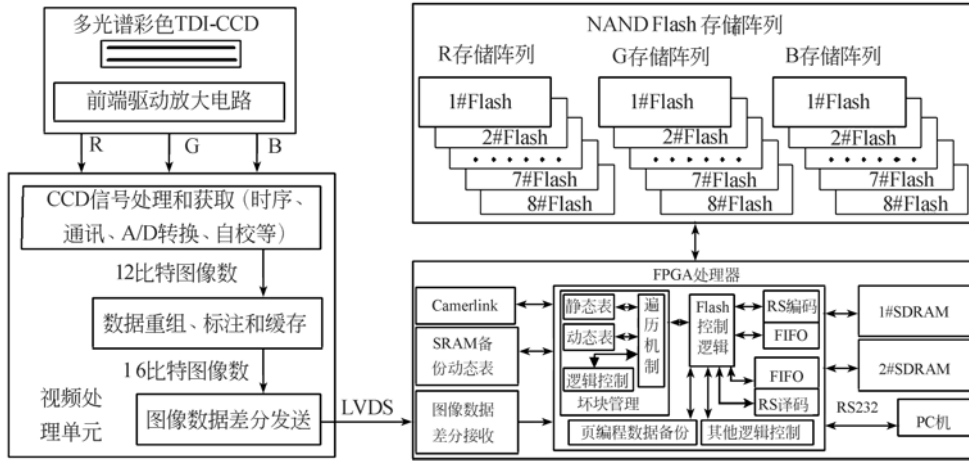


图 8 实验系统结构

Fig. 8 Structure of experiment system

### 5.2 坏块管理实验与分析

为了测试坏块管理策略性能以 R 通道第 1 级流水线 1# Flash 为例,由于实验过程中并未发现真实坏块和坏页,设静态表和动态表的坏块信息如表 1 和表 2 所示,其中,固有坏块是系统读取坏块标记得到的真实坏块。

表 1 静态表信息

Tab. 1 Static table contents

类型	内容
固有	0x007, 0x00a, 0x058, 0x068, 0x21b, 0x3ed, 0x4bb, 0x8af, 0x189b
擦除	0x14, 0x3a, 0x91, 0x12c, 0x1b6, 0x33d, 0x9c4, 0x1408

表 2 动态表信息

Tab. 2 Dynamic table contents

地址	内容
0x00 0x01 0x02 0x03 0x04 0x05 0x06 0x07 0x08	0x35 0x309 0x40 0x07 0x0b 0x21 0x06 0x16 0x11

当擦除到块地址为 0x21a 和 0x21b 时,时序

仿真波形如图 9 所示。擦除整个 Flash 后,开始编程和读取操作,以编程为例,当编程到地址为 0x035 块中的页地址为 0x20 页时,时序仿真波形如图 10 所示。

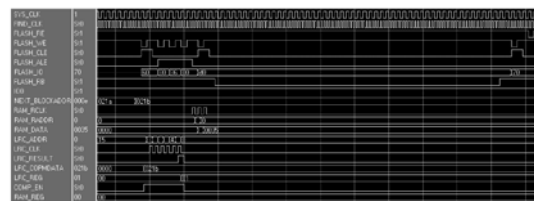


图 9 擦除操作仿真波形

Fig. 9 Time sequence simulation waveform of erasing

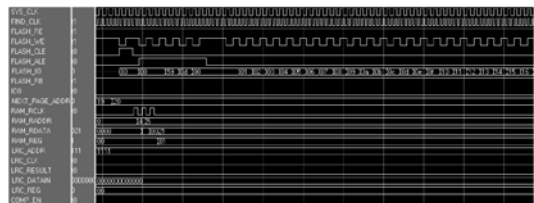


图 10 编程操作仿真波形

Fig. 10 Time sequence simulation waveform of programming

由图可知,在当前坏块和坏页执行擦除和编程操作的同时均可以遍历静态表和动态表,把遍历结果存储到寄存器中。执行下一页或块时只需

一个时钟判断结果寄存器值。

实际上,实验中单通道 CCD 有效像元为 3 072,行频为 1.2 kHz,AD 转换位数为 12 位,经重组后为 16 b 的数据,系统要求每次上电存储 38.7 min,需 Flash 容量为 17.12 GB。Flash 单块容量为 131 KB,因此,单通道需 130 612 块。而 8 片 Flash 总块数为 131 072,即剩余 460 块未用。系统上电检测单通道(以 R 通道为例)8 片 Flash 总坏块数为 69。所以,Flash 操作允许新增坏块数为 391。由式(3)得到二\_二分法遍历机制最大搜索时间  $t_{\max} \approx 9t_{\text{clk}}$ ,擦除操作遍历坏块信息最多需  $26t_{\text{clk}}$ ,读写操作遍历坏块信息最多需  $9t_{\text{clk}}$  完成坏块信息查找,完全满足并行坏块管理要求。这时,Flash 每次操作仅需 1 个时钟读取结果寄存器的值就可判断待操作块是否为坏块。

### 5.3 存储空间利用和资源占用分析

设每块页数为  $b$ ,每页容量为  $c$ ,使用闪存  $k$  次后,写入过程出现的坏块数为  $d$ ,每块第  $e(0 < e \leq b)$  为坏页。采用文献[8]坏块管理策略闪存容量减少  $cdb$ 。采用文献[9]法存储容量减少

$$\sum_{i=1}^d c(b-e+1), \quad (18)$$

采用本文方法存储容量减少  $cd$ ,而  $1 < b-e+1 < b$ ,所以本文坏块管理策略存储要比传统方法节省

$$\sum_{i=1}^d c(b-e). \quad (19)$$

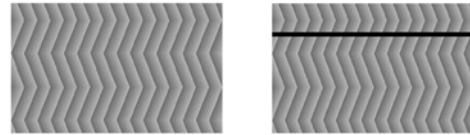
例如, $b=64, c=2\ 048\ \text{B}, d=17, e=30$ ,则第  $k$  次使用后比传统方法节省 1 183 744 B。

实验系统单通道(以 R 通道为例)允许坏块总数为 460,每块 64 页,所以最大需 172.5 Kb 内部块 RAM 存储坏块信息。级联 LC 单元最多约需 1191 个 LUT,而系统选用 Xilinx 生产的 XC2VP40 FPGA,内部块 RAM 最大为 3 456 Kb,LUT 总数为 38 784。所以,坏块管理模块 RAM 占用率小于 5%,LUT 占用率小于 3%。

### 5.4 坏块管理策略验证

为了验证坏块管理策略能否对坏块快速处理,分别采用内部自校图形发生器和实际 CCD 视频信号源对闪存读写实验验证。为了能够直观地观测到效果,从闪存中读取的数据通过 Camera Link 传输到 PC 机上,PC 机通过 IO 公司的 CL160 图像采集卡及配套软件在计算机上直接获取图像信息。

实验表明,坏块管理模块工作稳定可靠,能很好地处理坏块事件,图像清晰。实际有坏块管理和无坏块管理测试得到的图像如图 11 所示。



(a)有坏块管理

(b)无坏块管理

(a) Bad block management (b) No bad block management

图 11 自校图像测试坏块

Fig. 11 Bad block management test by verification image

### 5.5 纠错算法编译码器实验结果与分析

为了测试闪存纠错算法性能,图像自校发生器定时向闪存 ECC 模块发送 2048 B 数据,编码模块时序仿真波形如图 12 所示。

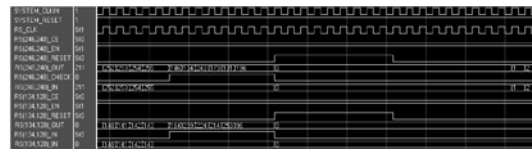


图 12 编码仿真波形

Fig. 12 Time sequence simulation waveform of encoding

译码模块工作时序如图 13 所示,图中码 SOUT 输出 7 个伴随式的值,ERRLOC 输出错误位置多项式  $\sigma(x)$  的系数,ERRDATA 输出错误值多项式  $\omega(x)$  系数。由纠正结果寄存器 ERRSULT 知,错误位置为 0,30,7,分别对应错误值为 5,7,3,把数据序列的第 0 个数据 151 与错误值 5 相异或得到纠正后的值为 146,其他同理。可见,译码器能准确快速的找到错误位置和错误值并纠正错误,输出正确的数据。

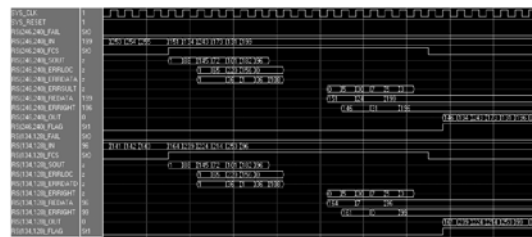


图 13 解码仿真波形

Fig. 13 Time sequence simulation waveform of decoding

RS(246,240)和 RS(134,128)码纠错码元数为  $t=3$ ,则 2 KB/page 可以纠正 27 B 错误,与目前现有 ECC 法 2 KB/page 纠错能力比较如表 3 所示。

表 3 纠错能力比较

Tab.3 Comparison of error correction technologies

范围	传统法[18]	BCH 法[19]	本文法
2KB/page	8 b	16 b	27 B
字节内	1 b	4 b	8 b

在 Flash 单页 2 KB/page,写入速度为 8 MB/s,读取速度为 20 MB/s 下,9 组并行工作的子 RS(255,249)编译码器组成 Flash 编程和读取编译码器,其工作参数如表 4 所示。

表 4 Flash 编解码器工作参数

Tab.4 Parameters of encoder and decoder

指标	编码器	译码器
工作时钟/ MHz	85	85
数据吞吐率/MBps	72.53	54.26
延迟/时钟周期	45	146
占用 LUT	5 122/38 784(13%)	8 116/38 784(20%)
占用 BRAM	27/192(14%)	46/192(23%)

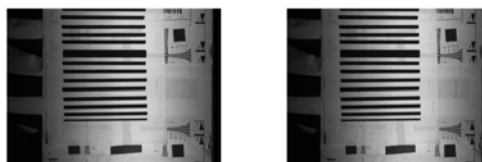
### 5.6 Flash 存储系统实验结果与分析

系统上电后,CCD 采集到图像通过编码模块后,将编码数据写入到 Flash 中,然后读取 Flash 中的图像数据并通过 Camera Link 传到 PC 机上,得到如图 14(a)所示,CCD 采集到的图像不经过 Flash 直接传到 PC 机上如图 14(b)所示。图像大小为  $3072 \times 1000$ (CCD 每行输出 3072 像素,每幅图像推扫 1000 行)。取 20 组如图 14 所示的图像,并对各组图像的数码值进行比较。计算方法借鉴数学上衡量保真度的均方根误差公式:

$$e_{rms} = \sqrt{\frac{\sum_{x=1}^M \sum_{y=1}^N (p(x,y) - q(x,y))^2}{MN}}, \quad (20)$$

其中  $p(x,y)$  为图 14(a)数码值,  $q(x,y)$  为图 14

(b)数码值,  $M$  为 CCD 像元数,  $N$  为扫描行数。由式(20)算得 20 组图像的均方根误差  $e_{rms}$  均等于 0,这是由于 Flash 在没有发生单粒子翻转时,坏块管理策略和纠错算法使 Flash 无损耗地读写数据。另外,笔者为了测试坏块管理策略和纠错算法性能加入了不同错误数的单粒子翻转事件,并对测得的图像数码值采用式(20)计算得到:错误数小于等于 27 B 时,  $e_{rms}$  为 0;错误数等于 28 B 时,平均  $e_{rms}$  为  $1.1 \times 10^{-3}$ ;错误数大于 28 B 时,随着错误数增多  $e_{rms}$  逐渐增大。可见。本文提出的坏块管理策略和纠错算法可以大幅度地提高 Flash 存储数据的可靠性。



(a)直接传输 (b)存储后传输

(a) Direct transmission (b) Transmission after storing

图 14 系统联调结果

Fig.14 Results of experimentation in the system

## 6 结 论

本文提出的基于并行双遍历机制的坏块管理策略,包括坏块划分、并行遍历坏块、级联 LC 以及二二分遍历机制等提高了坏块管理效率。提出的 RS(246,240)+RS(134,128)Flash 纠错算法,包括 Flash 页分块纠错、缩短码的添 0 和去 0 法以及编译码器的设计和优化等提高了 Flash 读取数据的可靠性。最后,在笔者研制的某空间多光谱相机样机的图像存储设备平台进行了试验验证。结果表明,坏块管理策略能快速可靠地处理坏块事件,每次操作仅需 1 个系统时钟周期完成坏块判断。纠错算法在 Flash 单页 2 KB/page,单页写入速度为 8 MB/s,单页读取速度为 20 MB/s 下,可以纠正 27 B 错误,编码速度达到 72.53 MB/s,解码器速度达到 54.26 MB/s,有效地解决了 Flash 数据存储的不可靠问题。为空间相机的图像大容量存储器 Flash 可靠存储数据提供了一种很好的解决方案。

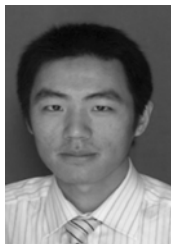
## 参考文献:

- [1] 贾福娟,吴雁林,黄颖,等. 环境减灾-1A、1B 卫星宽覆盖多光谱 CCD 相机技术[J]. 航天器工程, 2009, 18(6): 37-42.  
JIA F J, WU Y L, HUANG Y, *et al.*. Design and on-orbit application of CCD camera on HJ-1A/1B satellites[J]. *Spacecraft Engineering*, 2009, 18(6): 37-42. (in Chinese)
- [2] 田宝凤,徐抒岩,孙荣春,等. 一种适合星上应用的遥感图像无损压缩算法[J]. 光学精密工程, 2006, 14(4): 725-730.  
TIAN B F, XU SH Y, SUN R CH, *et al.*. A lossy compression algorithm of remote sensing image suited to space borne application [J]. *Opt. Precision Eng.*, 2006, 14(4): 725-730. (in Chinese)
- [3] 隋玉萍,何昕,魏仲慧. ROI 的海洋监视卫星遥感图像压缩算法[J]. 光学精密工程, 2008, 16(7): 1323-1329.  
SUI Y P, H E X, WEI ZH H. A compression algorithm of remote sensing image based on ROI for ocean surveillance satellite [J]. *Opt. Precision Eng.*, 2008, 16(7):1323-1329. (in Chinese)
- [4] 田宝凤,徐抒岩. 大容量固态记录器技术[J]. 光学精密工程, 2001, 9(4): 396-400.  
WANG X D, HAO ZH H. Mass solid state recorder technology [J]. *Opt. Precision Eng.*, 2001, 9(4): 396-400. (in Chinese)
- [5] Samsung Electronics. 1G x 8 Bit / 2G x 8 Bit / 4Gx 8 Bit NAND Flash Memory (Revision1. 1) [R]. 2006.
- [6] CHEN J C, KUO T H, CLEVELAND L E, *et al.*. A 2. 7 V only 8MB×16 NAN Flash memory [J]. *VLSI Circuits*, 1996:172-173.
- [7] 柳振中. 基于 NAND 闪存的自适应闪存映射层设计[J]. 现代电子技术, 2009, 311(24):106-109.  
LIU ZH ZH. Design of adaptive flash translation layer for NAND Flash [J]. *Modern Electronics technique*, 2009, 311(24): 106-109. (in Chinese)
- [8] 彭兵,步凯,徐欣. NAND Flash 坏块管理研究[J]. 微处理器, 2009, 3(2): 113-115.  
PENG B, BU K, XU X. NAND Flash bad blocks management research [J]. *Microprocessors*, 2009, 3(2): 113-115. (in Chinese)
- [9] 余辉龙,何昕,魏仲慧,等. 应用 NAND 型闪存的高速大容量图像存储器[J]. 光学精密工程, 2009, 17(10): 2548-2554.  
YU H L, HE X, WEI Z H, *et al.*. High speed and high capacity image recorder based on NAND flash [J]. *Opt. Precision Eng.*, 2009, 17(10): 2548-2554. (in Chinese)
- [10] Samsung Electronics. K9F5608X0D NAND Flash Memory datasheet(Revision1. 1) [R]. 2010.
- [11] KNUTH D E. *The Art of Computer Programming, Vol. 3, Sorting and Searching* [M]. 1937.
- [12] REED I S, SOLOMON G. Polynomial codes over certain finite fields [J]. *J. Soc. Ind. Appl. Math.*, 8: 200-204, June 1960.
- [13] 王新梅,肖国镇. 纠错码—原理与方法[M]. 西安: 西安电子科技大学出版社, 2001.  
WANG X M, XIAO G ZH. *Error-correcting Code-principles and Methods* [M]. Xian: Xidian University Press, 2001. (in Chinese)
- [14] LEE H. High speed VLSI Architecture for reed solomon decoder [J]. *IEEE*, 2003, 11(2): 288-294.
- [15] BERLEKAMP E R. *Algebraic Coding Theory* [M]. New York: McGraw Hill, 1968.
- [16] BAEK J H, SUNWOO M H. New degree computation-less modified Euclid algorithm and architecture for Reed Solomon decoder [J]. *IEEE*, 2006, 14(8): 915-920.
- [17] 刘悦,刘明业,尚振宏. RS(255,223)码的编译码软件实现[J]. 计算机应用与软件, 2006, 23(11): 47-49.  
LIU Y, LIU M Y, SHANG ZH H. A software implementation of encoding and decoding for RS (255,223) [J]. *Computer Applications and Software*, 2006, 23(11):47-49. (in Chinese)
- [18] Samsung Electronics Co., Ltd. ECC Algorithm (512Byte) [EB /OL].
- [19] 李璐,周海燕. 一种含 BCH 编解码器的 SLC/MLC NAND FLASH 控制器的 VLSI 设计[J]. 现代电子技术, 2009, 7(1):167-170.  
LI L, ZHOU H Y. SLC/MLC NAND controller VLSI design with BCH encoder and decoder [J]. *Modern Electronics technique*, 2009, 7(1): 167-170. (in Chinese)

## 作者简介:



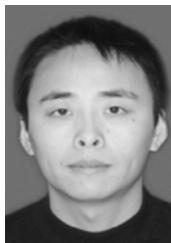
李进(1984—),男,内蒙包头人,博士研究生。2008年于中国地质大学(武汉)获得工学学士学位,主要研究方向为光电成像和图像压缩存储技术。E-mail: 664910699@qq.com



李国宁(1979—),男,辽宁盘锦人,副研究员,2008年于中科院光学精密机械与物理研究所获得博士学位,主要研究方向为光电成像与图像处理。E-mail:lgning@netease.com



金龙旭(1965—),男,吉林延吉人,研究员,博士生导师,2003年于中科院长春光学精密机械与物理研究所获博士学位,研究方向为空间相机智能控制技术。E-mail:jinx@ciomp.ac.cn



王文华(1982—),男,山东泰安人,助理研究员,2005年于东北师范大学获得工学学士学位,2010年于中科院长春光学精密机械与物理研究所获得博士学位,主要研究方向为光电成像与图像处理。E-mail:wangwh900@yahoo.com.cn



韩双丽(1968—),女,吉林长春人,副研究员,硕士生导师,1990年于大连理工大学获得工学学士学位,研究方向为空间光学遥感器热控制技术。E-mail:shuanglihan@yahoo.com.cn

## ●下期预告

## 五自由度并联机械腿静力学性能评价与优化设计

荣 誉<sup>1,2</sup>,金振林<sup>1</sup>

(1. 燕山大学 机械工程学院,河北 秦皇岛 066004;2. 河北科技师范学院,河北 秦皇岛 066004)

为了对一种六足机器人的五自由度并联机械腿进行静力学性能评价和优化设计,提出了一种同时考虑约束力映射关系和驱动力映射关系的腿部机构静力学分析方法,基于此方法对机械腿进行了优化设计。首先,通过分析腿部机构的驱动、约束映射关系,分别建立了腿部机构的驱动雅可比矩阵和约束雅可比矩阵;接着,应用虚功原理建立了腿部机构的驱动静力传递平衡方程,在定义驱动静力学评价指标的基础上绘制了评价指标分布图,得出了结构参数与驱动静力评价指标之间的关系曲线;然后,采用同样的方法建立了腿部机构的约束静力传递平衡方程,定义了约束静力学性能评价指标,得出了结构参数与约束静力评价指标之间的关系曲线;最后,基于驱动、约束静力学性能评价指标,采用蒙特卡罗法对结构参数进行了优化设计,给出了腿部机构静力学综合性能最好时的结构参数。此研究为六足步行机器人的进一步研究奠定了基础。