

基于 CAM 的闪存无效块管理算法

余辉龙^{1,2}, 何 昕¹, 魏仲慧¹, 王东鹤^{1,2}

(1. 中国科学院长春光学精密机械与物理研究所, 长春 130033; 2. 中国科学院研究生院, 北京 100039)

摘 要: 针对 NAND 型闪存无效块结构, 提出基于 CAM 的闪存无效块分类匹配算法。针对闪存擦除、写入和读取操作过程中无效块管理给出相应策略。在数据写入闪存过程中, 采用片外 SRAM 数据备份的方法防止数据存储错误。通过搭建 FPGA 实验平台, 证明该算法能发现新增长的无效块, 实现连续无效块快速匹配, 并对数据进行冗余备份。

关键词: 无效块管理; NAND 闪存; 内容可寻址存储器; 数据备份

Invalid Block Management Algorithm of Flash Based on CAM

YU Hui-long^{1,2}, HE Xin¹, WEI Zhong-hui¹, WANG Dong-he^{1,2}

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

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

【Abstract】 Based on the structure of the invalid NAND flash, the classified invalid block management algorithm is proposed. The invalid block management strategy is put forward respectively correspond to the erasing, writing and reading operation of the flash. When writing data to the flash, the written data stored into the NAND flash is simultaneously backup by the external SRAM to prevent data storage error. Experiment which is based on FPGA platform indicates the algorithm can quickly identify the new increased invalid blocks and the consecutive invalid blocks. The written data can be backup.

【Key words】 invalid block management; NAND flash; Content Addressable Memory(CAM); data backup

1 概述

闪存具有体积小、功耗与成本低、承受冲击能力强、非易失性等优点, 以闪存为存储介质的存储器得到了广泛的应用。现在主流的闪存分为 NOR 型和 NAND 型, NAND 型闪存存在写入速度和擦除速度上比 NOR 型快, 单元存储容量大, 且成本比较低, 在大容量数据存储上比 NOR 型闪存更具有优势^[1-3]。但 NAND 闪存由于工艺原因存在无效块^[4], 闪存操作过程中的无效块具有随机可增长性, 如何管理闪存无效块是数据可靠存储的关键。目前闪存的无效块管理方案有以下 2 种: (1) 将闪存每块第 1 页的备用区标记, 在遇到写入无效块时将整个写入该块的数据替换到下一个有效块^[4], 但是这样转移整块存储数据耗费大量的时间, 并对数据缓存单元容量提出苛刻的要求, 不利于数据高速存储。(2) 出现写入无效块时不进行整块替换, 而只将出现写入错误的页写入到下一个有效块的相应页, 在闪存每页的备用区设置无效块索引信息^[5], 这样系统在对每页操作前都将花费时间读取索引值, 且在操作结束后写入索引值, 同样不利于数据高速存储。

内容可寻址存储器(Content Addressable Memory, CAM) 能实现数据的快速匹配。当向 CAM 中输入需要匹配的内容时, 若存储器里有该内容, 将及时输出匹配有效信号, 给出该内容所在的存储地址。本文提出一种基于 CAM 的闪存无效块管理方案, 其时间开销小, 能最大限度地利用闪存存储空间。

2 基于 CAM 的匹配机制

2.1 闪存无效块分析

闪存是具有页和块的二维组织结构, 页的上一级单位是

块, 如图 1 所示。

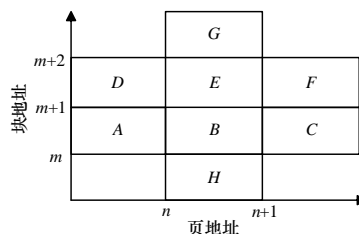


图 1 闪存的二维结构

闪存写入和读取操作在块地址和页地址交叉的 B 区, 闪存擦除操作是块操作, 将 ABC 区同时整块擦除。在闪存写入操作时, 若写入到 m 块 n 页, 即 B 区, 数据未成功写入, 若此时将整个写入的 A 区的数据替换到 m+1 块的 D 区, 将浪费较多时间, 不利于高速数据存储。若仅将该块未成功写入的 B 页替换重新写入到 m+1 块中, 则在闪存读取过程中, 须对无效块信息标记, 才能正确读取数据。

2.2 基于 CAM 的分类匹配机制

在闪存存在操作过程中会出现写入无效块和非写入无效块, 须对无效块信息分类。在闪存擦除操作时, 只须区分无效块与非无效块; 在写入和读取操作时, 还须区分写入无效块和非写入无效块。

作者简介: 余辉龙(1982—), 男, 博士研究生, 主研方向: 数字图像存储, 纠错编码技术; 何 昕, 研究员、博士生导师; 魏仲慧, 研究员; 王东鹤, 博士研究生

收稿日期: 2008-12-19 **E-mail:** lovystory@gmail.com

CAM 通过输入内容检索地址。有该内容, 匹配使能信号有效, 输出匹配地址; 没有该内容, 匹配使能信号无效, 该 CAM 存储器没有存储输入的内容, 整个匹配只须固定的几个时钟周期即可, 因此, CAM 能够实现数据的快速匹配。

基于 CAM 的分类匹配机制如图 2 所示。

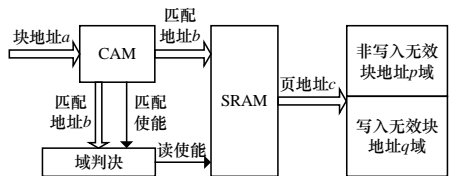


图 2 基于 CAM 的分类匹配机制

在擦除操作时, 只须输入块地址 a , 判断该块是否为无效块即可; 在写入和读取操作时, 将块地址 a 与 CAM 匹配, 若匹配成功, 利用地址判别器判断匹配地址在 p 域还是在 q 域中, 在 q 域中时将 CAM 匹配地址 b 作为 SRAM 的输入, 读出该块的最终写入结束页地址 c 。通过对 CAM 中地址的分类实现了对写入无效块和非写入无效块的匹配。

3 闪存无效块管理算法及其实现

3.1 闪存无效块管理的系统组成

无效块管理实验系统如图 3 所示, 其中, FPGA 中 CAM 实现闪存无效块信息的匹配和更新; 片内 SRAM 存储写入无效块的结束页信息, 控制逻辑部分控制所有模块; 片外 SRAM 在闪存存储数据时进行数据备份; FRAM(铁电存储器)备份 CAM 及 SRAM 中数据; 串口用来发送系统命令; USB 读取验证各单元数据; 16 bit 数据为外部数据源。

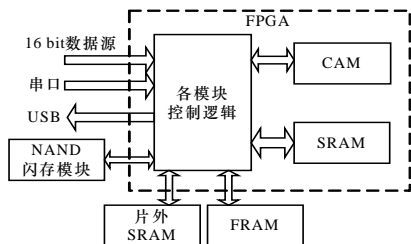


图 3 无效块管理系统组成

3.2 初始无效块的识别

通过读取闪存每块第 1 页和第 2 页备用区的首字节是否为 FF 来识别该块是否为无效块, 若为无效块则将无效块信息写入 CAM, 识别完成后将无效块信息备份到 FRAM 中。

3.3 擦除操作时的无效块管理

在闪存每次擦除前, 通过 CAM 对闪存块地址匹配, 若匹配成功, 该块为无效块, 不用擦除, 块地址递增 1; 若未匹配, 该块为有效块, 进行擦除操作, 等待闪存忙信号无效, 读取状态字节, 若正确擦除, 继续对下一块操作; 若擦除错误, 则该块是新增长无效块, 将无效块数据写入到 CAM 中。继续对下一块操作。

3.4 写入操作时的无效块管理及数据备份

在闪存每块第 1 页写入数据前, 通过 CAM 对闪存块地址匹配, 若匹配成功, 该块为无效块, 不用写入, 块地址递增 1; 若未匹配, 该块为有效块, 将数据按页顺序写入闪存, 同时写入片外 SRAM, 每一页写入完成时, 读取闪存的状态信息, 若正确写入, 继续写入下一页; 若未正确写入, 将该无效块信息写入到特定地址的 CAM 中, 同时将该块的结束页地址写入到片内 SRAM 中, 块地址递增 1, 通过 CAM 匹配找到下一个有效块, 将备份到片外 SRAM 中的数据重新写

入到该有效块第 1 页中; 若正确写入, 则备份完成, 继续进行正常的数据写入操作; 若未正确写入, 则块地址递增 1, 继续上一步操作, 直到备份数据正确写入闪存, 由此实现了数据备份和闪存写入无效块管理。

3.5 读取操作时的无效块管理

在闪存每块第一页读取前, 通过 CAM 对闪存块地址匹配, 若未匹配, 该块为有效块, 进行正常的读取操作; 若匹配成功, 该块为无效块, 读取匹配地址, 如果该地址为非写入无效块域, 块地址递增 1, 继续进行块地址匹配; 如果是写入无效块域, 从片内 SRAM 中读取该块对应写入结束页地址, 将结束页和写入无效块信息组合, 得到该块数据的有效存储结束页信息, 读取数据直到结束页为止, 由此实现数据的正确读取。

4 实验结果与分析

闪存无效块可能的形式有: 单独的非写入无效块或写入无效块; 连续写入无效块; 连续非写入无效块; 连续写入无效块和非写入无效块。在闪存实验过程中并未发现无效块, 采取了一些方法对算法的正确性进行验证。对 CAM 地址单元的内容分配如表 1 所示。与 CAM 写入无效块相对应, 片内 SRAM 地址 32h~3Fh 存储写入无效块的相应页结束信息。

表 1 CAM 地址分配

地址	内容
0~15h	原始无效块
16h~31h	擦除无效块
32h~3Fh	写入无效块

4.1 闪存无效块匹配实验

CAM 的初始值设置为: 00h: 0040h; 32h: 0080h, 其他地址的内容均为 FFFFh, 其中, 片内 SRAM 的内容设为 32h: 25h。0040h 和 0080h 为连续无效块, 其中, 0040h 为原始无效块; 0080h 为写入无效块, 其结束页信息为 25h。通过泰克逻辑分析仪 TLA5201 观察和 USB 读取数据验证得出:

(1) 闪存能被正确擦除, 擦除块地址从 0000 直接跳转到 00C0h。没有擦除无效块 0040h, 0080h 块。证明了擦除操作的正确性, 能够实现对连续无效块的正确匹配。

(2) 闪存能够正确写入数据, 同时将数据写入到 SRAM 中, 闪存写入地址为 003Fh 时跳转到 00C0h, 证明能够实现连续无效块匹配。

(3) 闪存数据能被正确读取, 读取地址为 003Fh 时直接跳转到 0080h, 一直递增到 00A5h, 然后直接跳转到 00C0h, 由于块地址 0080h 加上结束页地址 25h, 即地址为 0080h 的块有效页地址最大到 00A5h, 因此实现了读取无效块的分类匹配, 证明了闪存存在写入无效块时读取算法的有效性。

4.2 闪存新增无效块识别算法的验证

在闪存操作过程中, 未发现真实的无效块信息, 设置无效块信息, 当闪存块地址为 0100h, 将其认为是新增无效块, 写入操作时结束页地址设为 25h。通过泰克逻辑分析仪 TLA5201 观察和 USB 读取数据验证得出:

(1) 通过 USB 读取 CAM 数据, 地址为 16h 时内容为 0100h, 其他均为 FFFFh。片内 SRAM 均为 00h, 验证了擦除无效块识别的正确性, 能够发现新增长擦除无效块。

(2) 通过 USB 读取 CAM 中数据, 地址 32h 时内容为 0100h, 其他均为 FFFFh。片内 SRAM 地址为 32h 时内容为 25h, 其余均为 00h, 闪存地址 0140h 写入操作时, 数据源来

(下转第 255 页)

在新的构架下, VCamFilter 在获取外部数据之后, 数据已经进入了 VCamFilter 内部数据区, 这时如果不加处理直接将数据传递给输出 Pin, 后者只须取得内部数据区的首地址即可, 花费的代价将是最小的。如果用户在将数据传递给输出 Pin 之前需要对其做预处理, 可以直接获取内部数据区, 分别应用预处理算法即可。

3.4 向下游过滤器输出数据的过程

Minidriver 对于数据输出部分功能的支持主要表现在它根据特定数据结构 HW_STREAM_INFORMATION 的描述, 在 KsProxy 中实例化出相应的输出 Pin(包括预览 Pin 和捕捉 Pin), 并同时实现了 IMediaSeeking, IAMStreamControl, IKsPropertySet 等接口。为完全模拟该过程, 须通过继承 CSourceStream 类及 IKsPropertySet 等接口, 实现 VcamCapPin(捕捉 Pin)和 VCamPreviewPin(预览 Pin), 并将其加入到 VCamFilter 之上。其中, 重点实现的方法有 GetMediaType, SetMediaType, DecideBufferSize 和 FillBuffer。GetMediaType 主要用于根据数据流信息声明该输出 Pin 支持的多媒体格式, 下游过滤器在媒体协商的过程中将调用该方法以检查是否可进行连接, SetMediaType 用于在完成多媒体格式协商之后设置实际传输使用的媒体格式, DecideBufferSize 用于设置传输过程中媒体样本的大小, 传输过程中 FillBuffer 从内部数据区取得数据, 同时加上时间戳以保证下游过滤器中音视频的同步播放, 在完成对媒体样本的填充后直接将其传递给下游, 至此虚拟摄像头已经具有了输出实际数据的能力。

4 应用实例

基于新虚拟摄像头开发框架, 本文实现了一个无需驱动程序支持的虚拟摄像头。图 3 为该虚拟摄像头在 GraphEdit^[3]工具中形成的过滤器表。



图 3 虚拟摄像头在 GraphEdit 中的连接

与真实摄像头一样, 该虚拟摄像头有 2 个输出 Pin, 分别是预览 Pin 和捕捉 Pin, 利用 GraphEdit 提供的过滤器表智能构建能力, 该虚拟摄像头下游连接了一个颜色空间转换过滤器(Color Space Converter Filter)和视频渲染过滤器(Video

Render Filter)。

图 4 为该虚拟摄像头在 MSN 应用程序中, 被识别为普通的摄像头(VirtualCam), 其中的外部数据来自用户选择的水流视频片断, 虚拟摄像头在接收到视频数据之后, 通过输出 Pin 中的输出, MSN 应用程序随即取得数据。



图 4 虚拟摄像头被 MSN 识别的界面

5 结束语

针对传统虚拟摄像头开发方案中存在的问题, 本文提出一个基于 DirectShow 技术的开发模型, 具有实现简单、通用性好(支持 MSN, Skype 等众多 IM 软件)、易于维护和扩展等特点, 无须涉及驱动程序的编写, 有利于提高系统的稳定性。由于数据无须经过用户空间和内核空间的 2 次传递过程, 因此效率得到显著提高。

参考文献

- [1] 华 峻, 杨树堂, 李建华. 基于 DirectShow 技术视频流捕捉及压缩的实现方案[J]. 计算机工程, 2004, 30(12): 143-146.
- [2] 余文权, 杜 旭, 熊 琦. 虚拟摄像头开发模式及其应用研究[J]. 微计算机信息, 2007, (25): 288-290.
- [3] 陆其明. DirectShow 开发指南[M]. 北京: 清华大学出版社, 2003.
- [4] Oney W. Programming the Microsoft Windows Driver Model[M]. [S. l.]: Microsoft Press, 2003.

编辑 顾姣健

(上接第 252 页)

自片外 SRAM, 实现了数据备份, 验证了写入过程新增长无效块识别算法的正确性, 并实现了无效块信息的分类。

5 结束语

本文针对无效块的结构, 提出基于 CAM 的闪存无效块分类匹配检测算法, 在闪存写入、擦除和读取操作过程中能够对无效块快速匹配检测, 在操作过程中检测连续无效块。算法能发现新增长无效块, 并对写入无效块和非写入无效块进行分类, 在写入操作时, 数据写入闪存的同时将写入片外 SRAM, 遇到写入无效块时, 算法能将片外 SRAM 中备份的数据回写到闪存中, 实现数据备份。

参考文献

- [1] Kang Jeong-Uk, Kim Jin-Soo, Park C, et al. A Multi-channel Architecture for High-performance NAND Flash-based Storage

- System[J]. Journal of Systems Architecture, 2007, 53(9): 644-658.
- [2] Tsur O, Saba K. Rugged, Reliable, and Secured Data Storage Solutions for Airborne ISR[C]//Proc. of SPIE: Airborne Intelligence, Surveillance, Reconnaissance Systems and Applications. Orlando, FL, USA: [s. n.], 2004.
- [3] Fishell W G. Solid State Memory in Recce System[C]//Proc. of SPIE: Airborne Reconnaissance XIX. San Diego, CA, USA: [s. n.], 1995.
- [4] Samsung Electronics. Application Note for NAND Flash Memory[Z]. (1999-12-28). http://www.samsung.com/global/business/semiconductor/products/flash/downloads/applicationnote/app_nand.pdf
- [5] 朱 岩, 沈卫华, 孙辉先. 基于闪存的固态存储器的数据管理[J]. 计算机工程, 2007, 33(12): 73-75.

编辑 顾姣健