

NAND FLASH 编程实现研究分析

陆林燕, 王鲁静, 郑正奇
(华东师范大学 电子系, 上海 200062)

摘要:以阐明编程支持 NAND FLASH 的方法为目的。总览了 NAND FLASH 层次结构; 讨论了 NAND FLASH 的写操作过程, 这种写操作的特点也是 FLASH 有别于其它存储介质的地方, 同时也对 NAND FLASH 的擦除操作做了介绍; 在对整体的框架和特点有了了解之后, 进一步对编程支持 NAND FLASH 的过程中会遇到的一些概念和细节给出了具体的说明。对 NAND FLASH 的编程支持有一定的复杂性, 但只要了解了它的工作方式, 也并非难事。

关键词:NAND FLASH; 写操作; 地址转换; ECC 校验

中图分类号:TP311

文献标识码:A

文章编号:1673-629X(2008)03-0118-03

Study and Analysis on Realization Coding of NAND FLASH

LU Lin-yan, WANG Lu-jing, ZHENG Zheng-qi

(Dept. of Electronics, East China Normal University, Shanghai 200062, China)

Abstract: Aim at describing the method of making the NAND FLASH work by coding. Firstly the architecture of NAND FLASH is talked. Then the programming process, which is the difference between NAND FLASH and other storage media, is discussed. In the same time, the erase operation is also introduced. After a clear image of the whole frame of NAND FLASH is built, concepts and details will meet at the coding progress to support NAND FLASH is accounted for specifically. It is complicated to support NAND FLASH by coding, however it is not so difficult as long as know the work style of NAND FLASH.

Key words: NAND FLASH; writing operation; address switching; ECC

0 背景情况

NAND FLASH 是目前应用比较广泛的一种存储解决方案^[1,2]。2006 年, 由于 NAND FLASH 制造工艺的改变, 使许多芯片支持业者措手不及, 大多 MLC NAND FLASH 的支持出了问题。笔者也从事芯片支持的工作, 在开发过程中也曾遇到很多问题, 在问题解决之后总结了一些实际开发的经验。

1 NAND 的存储组织结构

根据 2006 年 12 月 Hynix, Intel, Sony 等公司联合发布的《开放 NAND FLASH 接口说明》^[3], 将 NAND FLASH 的内存结构由粗到细依次规范为器件 (device)、目标 (target)、单元 (unit)、分块 (block)、分页 (page) 以及扇区 (sector)。

器件即指 NAND FLASH 本身, 一个器件由一个或多个目标构成。而各个目标是否工作则由 CE (chip

enable) 信号来控制, 同一时刻只有一个目标处于工作状态。一个目标又可以分成一个或多个逻辑单元。逻辑单元是独立地执行命令和报告状态的最小单位, 原则上 NAND FLASH 支持多逻辑单元操作。每个逻辑单元都包含至少一个分页寄存器和一个存储阵列。分页寄存器用于暂存由外部读入 NAND FLASH 的数据以及由 NAND FLASH 读出的数据。存储阵列由许多分块组成。一个分块是逻辑单元中进行擦除操作的最小单位。一个分块又包含若干个分页, 对于支持约束性连续分页操作的器件来说, 分页是最小的可寻址操作单位。一个分块所包含的分页数规定为 32 的倍数。最后为了与其他存储器件的统一并且满足分页在字节数上的要求, 引进了扇区这个单位。一个扇区仍然是 512 个字节, 由于 NAND FLASH 寻址及工作原理的特殊性, 在每个 512 字节的扇区后会有一个 16 个字节的备注区域。在这个备注区域中不存放用户数据, 而是存放了与地址转换、ECC 校验、坏块标记/判断等有关的信息。

可以看到只含有一个目标的 NAND FLASH 和含有多个目标的 NAND FLASH 的差别只是在 CE 这个

收稿日期: 2007-06-14

作者简介: 陆林燕 (1982-), 女, 上海人, 硕士研究生, 研究方向为通信与信息系统、智能卡驱动; 郑正奇, 博士生导师, 博士, 研究方向为通信与信息系统、信号检测。

控制信号上,除此之外它们的工作原理是完全一样的。因此在讨论 NAND FLASH 的工作部分时,只关心从逻辑单元开始往下的内存结构。内存结构图如图 1 所示。

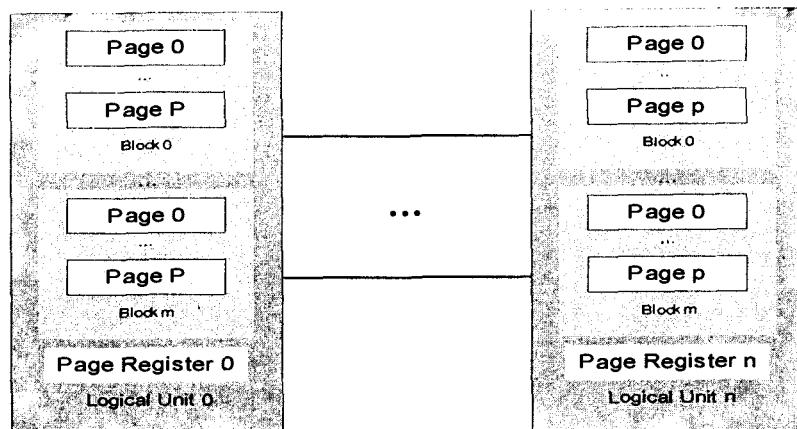


图 1 NAND FLASH 结构示意图

2 NAND FLASH 工作中的特点

2.1 NAND FLASH 的写方式

NAND FLASH 在还没有写入任何东西前,存储阵列都表现为存储 0xFF。进行写操作实际上就是把阵列中某些 1 位变为 0 的过程。事实上,NAND FLASH 在写操作中只能把 1 变为 0,而不能把 0 变为 1,所以要直接对一个已经有内容的存储单元进行写操作是被禁止的。这也是为什么 NAND FLASH 要求采取先擦除后写入的原因。擦除操作实际上就是把一个分块的各个位全部置 1 的过程。NAND FLASH 的这些特性使得它的写操作过程呈现出一定的特殊性。

要对 NAND FLASH 进行写操作,首先要做的就是地址的转换,从逻辑位置转换到单元号、逻辑分块号和逻辑扇区号。为叙述方便,令逻辑分块号为 N 的分块对应的物理分块为 J ,又有一物理分块为 K ,逻辑扇区号为 I 。则如果经过地址转换后得到的逻辑扇区号 I 为 0,那么本次操作的位置即为 N 逻辑分块的开始位置,那么只要为逻辑分块 N 分配物理分块 J (注意物理分块 J 中是不含有任何内容的)后,就可以进行写入操作了。如果逻辑扇区号是 $I(I \neq 0)$,即要从 N 分块的中间位置开始写,那么就需要先找到一个空白的物理分块 K ,将逻辑分块 N 对应的物理分块 J 中位于 I 之前的部分先拷贝到 K 中,然后再在物理分块 K 中的 I 位置处写入本次操作要写的内容,假设进行了 M 个

扇区的写操作。等到下一次写命令来的时候,如果要操作的逻辑扇区号是 $I + M$,即本次操作的扇区与上次操作的扇区是连续的,那么就可以直接继续写下去;如果与上次操作的扇区不是连续的,那么需要将 J 分块

中 $M + I$ 及之后的扇区先拷贝到 K 分块的对应位置,然后建立逻辑分块 N 与物理分块 K 的对应关系并且把逻辑分块 N 与物理分块 J 的对应关系取消,之后才可以进行新一次的写操作。操作过程如图 2 所示。

2.2 NAND FLASH 的擦除操作

NAND FLASH 的擦除操作以分块为单位,将目标分块中的各位(包括数据区和备注区)全部置 1 的过程^[4]。一般擦除操作被用于两种情况:一种即上面提到过的擦除/写入的过程;另一种是在

低层次格式化时,即把整个 NAND FLASH 全部置 1 的时候。后一种情况,在擦除前,应先检查目标擦除块是否已被标记为坏块(检查的方式为读取备注区域中的字节 0 或字节 5,后详)。这是因为擦除操作也会将备注区全部置 1,即坏块的标志信息也会被清除,这样在写入操作时会因为无法识别而造成系统将坏块分配的情况,造成数据的错误。

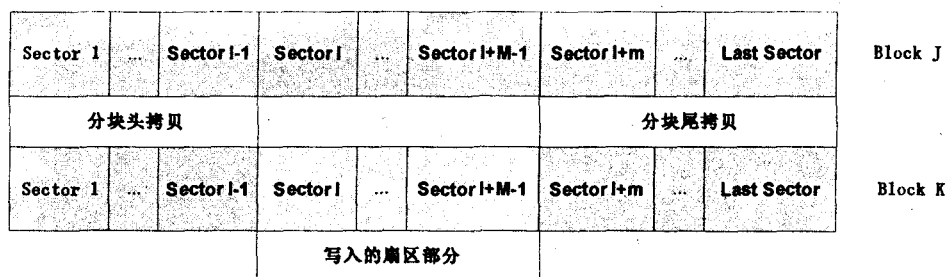


图 2 NAND FLASH 写操作过程

3 NAND FLASH 的一些工作细节

3.1 逻辑地址和物理地址的转换(LP 表)

NAND FLASH 是通过存储阵列来存储信息的。如上面所说,分页是寻址的最小单位,因此只要知道了单元号、分块号、分页号就可以通过以下公式确定存储阵列中的某个特定的物理位置,以达到物理寻址的目的:

阵列中的位置 = 单元号 * 单元中最大分块数 + 物理分块号 * 分块中最大分页数 + 分页号

每个分块按其在存储阵列中的位置都有唯一的物理分块号与之对应,但在上层应用时并不直接以物理地址为对象,而是逻辑地址。并且由于 NAND FLASH

的工作特性,使得物理地址和逻辑地址的对应关系不可能一成不变,因此逻辑地址要通过一定的处理转换成物理地址,转换的逻辑流程图如图 3 所示。

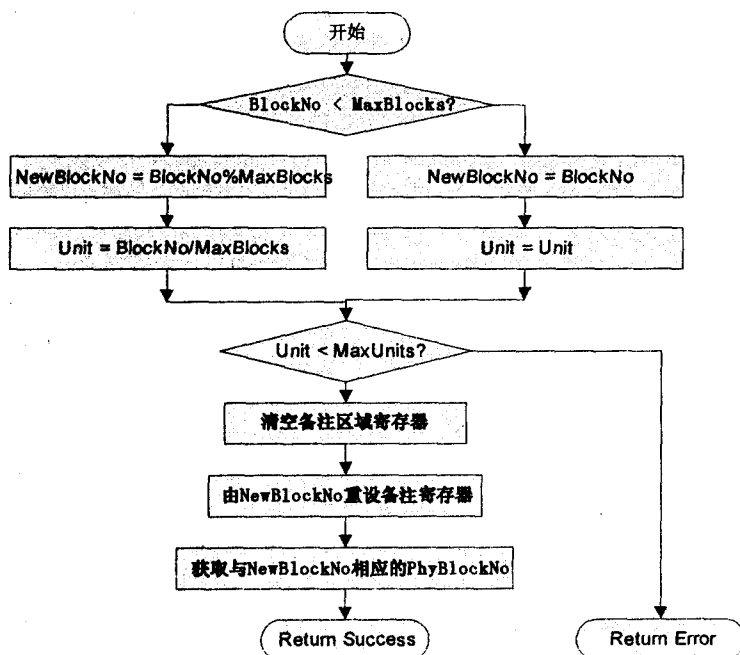


图 3 NAND FLASH 地址转换流程图

图 3 中 BlockNo 和 NewBlockNo 都是指逻辑分块号,最后一步“获取与 NewBlockNo 相应的 PhyBlockNo”是通过 LP 表实现的。LP 表的索引号为逻辑分块号,LP 表的内容为物理分块号,比如 $LP[i] = j$ 表示的就是逻辑分块号为 i 的分块对应的物理分块为 j 。

3.2 每页 1 个扇区和每页多个扇区在操作上的不同

不同类型的 NAND FLASH 每个分页所包含的扇区数是不同的。对于每个分页只包含一个扇区的 NAND FLASH 来说,只需要简单地有一个扇区写一个扇区的操作就可以了。而对于一个分页含有多个扇区的情况,说明书上都会特别指出:在没有擦除操作的情况下对一个分页的擦除操作不可以超过一次^[5],即 NOP(number of page)应为 1。对这一点的忽视也就是造成 2006 年 10 月出现的大量芯片控制业者无法支持 MLC NAND FLASH 的原因。所以对于分页数不为 1 的 NAND FLASH 就需要进行一定的管理。以每页有 4 个扇区为例可以这样实现:

```
char page_buffer[4][528]; //集中存放 4 个扇区的数据及其备注区
```

```
BOOL page_buffer_rd[4]; //标记第 i 个 page_buffer 中是否有数据。若没有,写硬件时该扇区填充 0xff
```

```
BOOL page_buffer_main[4]; //标记第 i 个 page_buffer 中是否有备注区域。若没有,写硬件时该备注区域填充 0xff
```

```
UINT32 page_buffer_index; //记录当前操作的分页号,当送来的扇区的地址不属于该分页时启动硬件写入操作,将 page_
```

buffer 中的四个扇区写入

3.3 ECC 校验

ECC 校验的目的是检查 NAND FLASH 在读操作时是否发生错误,并在有限错误范围内 (2bit/256Byte) 纠正出现的错误。ECC 校验以半个扇区 (256 个字节) 为单位。前面提过 NAND FLASH 在每个数据区 (512 个字节) 的后面,会有 16 字节的备注区域,其中第 8 字节到第 a 字节用于校验本数据区后 256 个字节,第 d 字节到第 f 字节用于校验本数据区前 256 个字节。

由于 ECC 校验是要耗费系统资源以及时间的,并不建议在所有情况下都使用。比如对于本身性能比较稳定的 NAND FLASH^[6] (如 SLC) 来说, ECC 校验并不是必需的,而对于性能并不是很稳定的 MLC 来说,使用 ECC 校验则是保证读写正确所必需的。

3.4 备注区域说明

备注区域是 NAND FLASH 的管理字节存放的区域。备注区各字节的功能如下:

字节 0: 当一个分页由 4 个扇区构成时,标记分块是否为坏块,值为 0x00 时为坏块

字节 1~字节 3: 保留

字节 4: 数据状态标志字节,用于判断扇区是否为非法扇区

字节 5: 当一个分页不是由 4 个扇区构成时,标记分块是否为坏块,值为 0x00 时为坏块

字节 6,7: 地址存放区 1,用于存放与本物理分块相关的逻辑分块号

字节 8~a: ECC 校验区,用于存放对应扇区后 256 字节的奇偶校验值

字节 b,c: 地址存放区 2,用于存放与本物理分块相关的逻辑分块号,值与字节 6,7 相同

字节 d~f: ECC 校验区,用于存放对应扇区前 256 字节的奇偶校验值

4 结束语

在各种存储介质的编程实现中 NAND FLASH 是较为复杂的一种。首先,程序的层次多。在上层要判断是否需要重新分配物理分块,在下层要根据分页包含的扇区数进行数据的集中;其次 NAND FLASH 的种类多。任何制造工艺上的不同都可能要求芯片控制业者重新调整代码构架。因此掌握 NAND FLASH 的

(下转第 124 页)



图 7 R 分量图



图 8 G 分量图



图 9 B 分量图

图 5 为原彩色图像;图 6 是直接利用 Sobel 算子得到的结果,由结果图可以看出有些主要边缘信息没有

被检测出,而且图中房屋、树木等多处边缘不连续;图 7、图 8 和图 9 为 RGB 空间分量图像。分析实验结果,与传统算法相比较,文中提出的彩色图像边缘检测改进算法能较好并更多地保留原有图像的颜色和轮廓特征,算法实现简单,实现效果较好,但是原有图像中的一些细节信息缺失,这有待进一步对算法进行改进。

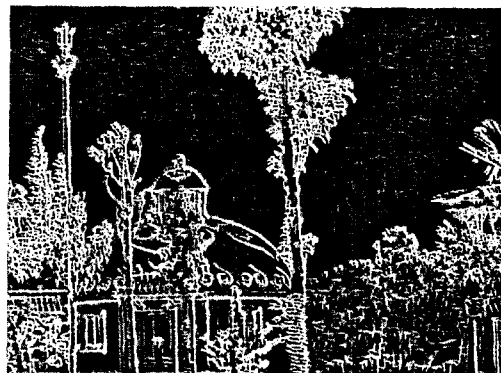


图 10 文中算法检测结果图像

参考文献:

- [1] Castleman K R. Digital Image Processing[M]. 北京:清华大学出版社, 2003.
- [2] Health A, Sarkar S, Sanocki T, et al. Comparison of Edge Detectors: A Methodology and Initial Study[J]. Computer Vision and Image Understanding, 1998, 69(1): 38 - 54.
- [3] Dony R D, Wesolkowski S. Edge Detection on Color Images using RGB Vector angle[C]//IEEE Canadian Conference on Electrical and Computer Engineering. [s. l.]: [s. n.], 1999: 687 - 692.
- [4] Chen H C, Chien W J, Wang S J. Contrast - based Color Image Segmentation[J]. IEEE Signal Processing Letters, 2004, 11(7): 641 - 644.
- [5] Cai C, Miran S K. A Normalized color Difference edge Detector Based on Quaternion Representation[C]//In: Proceedings of IEEE International Conference on Image Processing. Vancouver, Canada: [s. n.], 2000: 816 - 819.
- [6] 章毓晋. 图像分割[M]. 北京: 科学出版社, 2001.

(上接第 120 页)

工作特点及相关细节就显得尤为重要,有了对 NAND FLASH 的深入了解,碰到问题时就可以从容面对了。

参考文献:

- [1] 马丰玺, 杨 斌, 卫洪春. 非易失存储器 NAND Flash 及其在嵌入式系统中的应用[J]. 计算机技术与发展, 2007, 17(1): 203 - 205.
- [2] 周 涛, 刘有源. 基于 MIPS 的 Au 1500NC 板上的启动程序 U-Boot 设计[J]. 计算机技术与发展, 2006, 16(8): 94

- 96.

- [3] Hynix, Intel Corporation, Micron Technology Inc. Open NAND Flash Interface Specification [R]. America: Hynix, 2006.
- [4] 刘洲洲, 张 捷. 基于嵌入式系统的 NANDFlash 坏块处理和 FMM 实现[J]. 计测技术, 2006(6): 42 - 44.
- [5] Hynix. 4Gb NAND FLASH HY27UT084G2M[R]. Korea: Hynix, 2006.
- [6] Hynix. NAND Flash PART NUMBERING[R]. Korea: Hynix, 2006.