

非易失存储器 NAND Flash 及其在嵌入式系统中的应用

马丰玺, 杨 斌, 卫洪春

(西南交通大学 信息科学与技术学院, 四川 成都 610031)

摘 要:文中介绍了 NAND Flash 的芯片内部组织结构、主要的外部引脚及其功能、NAND Flash 的类型、NAND Flash 所支持的文件系统、与微处理器的连接和 NAND Flash 的固有特性, 并对比了 NAND 和 NOR Flash 的异同。最后简述了闪存固件程序架构和如何在 Linux 操作系统中加入对 NAND Flash 的支持。

关键词:NAND Flash; Linux; 闪存固件; 嵌入式系统; 文件系统

中图分类号: TP333

文献标识码: A

文章编号: 1673-629X(2007)01-0203-03

Nonvolatile Memorizer NAND Flash and Applications in Embedded System

MA Feng-xi, YANG Bin, WEI Hong-chun

(School of Information Science & Technology, Southwest Jiaotong University, Chengdu 610031, China)

Abstract: Requirement for storage devices increased day by day as development of products in embedded system. Introduced NAND flash in organization inside chip, leading pins outside, type and file system supporting NAND flash. Then showed NAND flash connects with ARM microprocessor and inhere features. NAND flash was compared with NOR flash. Last, program structure in flash firmware was given and how to add NAND flash in Linux operating system was discussed.

Key words: NAND flash; Linux; flash firmware; embedded system; file system

0 引 言

NOR 和 NAND 是现在市场上两种主要的非易失闪存技术。NOR Flash 较早为业界采用, 但由于其成本较高及写入速度较慢的先天弱势, 使其仅能在注重执行速度或小量数据储存的地方使用。NAND Flash 结构强调降低每比特的成本, 更高的性能, 并且像磁盘一样可以通过接口轻松升级, 是高密度数据存储的理想解决方案。

1 NAND Flash 芯片

1.1 芯片介绍

NAND Flash^[1,2]是采用 NAND 结构技术的非易失存储器, NAND Flash 芯片设计成了页的阵列, 每一页由 256/512 字节的数据区和 8/16 字节的备用区(也称为组外区)组成, 如图 1 所示, 最新的芯片中的每一叶拥有 2048 字节的数据区和 64 字节的备用区的容量。备用区用于存储错误更正编码、坏块信息和与数据相关文件系统。由 32 页构成一个块。对数据的读写操作是以每一个页为基本单位进行的(也能够以字节为单位进行读写), 而擦除操作是以每个块为基本单位的。在命令锁存使能引脚为高电平时, 发给芯片的读/写/擦除命令就可以通过写命令写进芯片, 当地址锁存使能引脚为高电平时, 就可以

通过写命令给出地址。Flash 的 I/O 接口可用于控制命令和地址的输入, 也可用于数据的输入和输出。Flash 页的大小和块的大小不同, 因而类型块结构不同, 块结构有两种: 小块和大块。大块与小块的结构极为相似, 只是大小不同。文中以小块为例来介绍。小块(见图 1)Flash 包含 32 个页, 每页 512+16 字节。

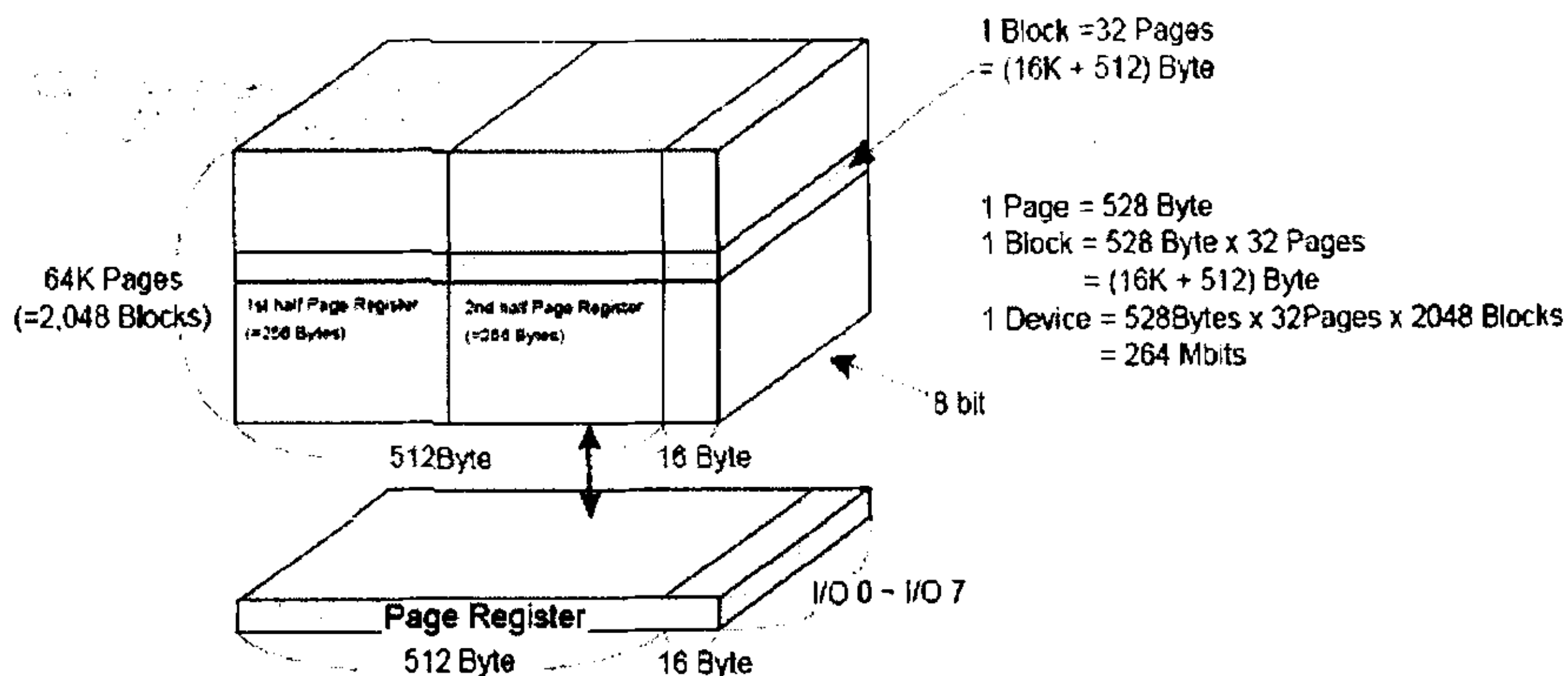
Flash 都有与页大小相同的页寄存器, 用于数据缓存。当读数据时, 先从 Flash 内存单元把数据读到页寄存器, 外部通过访问 Flash I/O 端口获得页寄存器中数据(地址自动累加); 当写数据时, 外部通过 Flash I/O 端口输入的数据首先缓存在页寄存器, 写命令发出后才写入到内存单元中。FLASH 典型的读操作时间为 50ns/字, 写操作时间为 200 μ s/页, 擦除操作时间为 2ms/块, 块擦写次数超过 100k, 数据保存时间超过 10 年。

1.2 外部引脚

访问 NAND Flash 存储器仅有一些引脚是必需的(如表 1 所示), 下面仅介绍这些必需的引脚及其功能: I/O0~I/O7: 8 个 I/O 口; CE: 片选信号, 低电平有效; WE: 写使能信号, 低电平有效; RE: 读使能信号, 低电平有效; CLE: 命令使能信号; ALE: 地址使能信号, 高电平有效; WP: 写保护信号, 低电平有效; RY/BY: 高电平时为 READY 信号, 低电平时为 BUSY 信号。当使用备用区时/SE 引脚是必须的, 该引脚应该接地, /CE, CLE 和 ALE 引脚应当接 GPIO 引脚或是锁存信号, 使 ALE 和 CLE 有效, 使用地址线是有可能的, 但必须掌握好芯片的严格的时序^[2,3]。

收稿日期: 2006-04-07

作者简介: 马丰玺(1980-), 男, 山东鄄城人, 硕士研究生, 研究方向为嵌入式系统; 杨 斌, 副教授, 研究方向为嵌入式系统。



| | I/O 0 | I/O 1 | I/O 2 | I/O 3 | I/O 4 | I/O 5 | I/O 6 | I/O 7 | |
|-----------|-------|-------|-------|-------|-------|-------|-------|-------|----------------------------|
| 1st Cycle | A0 | A1 | A2 | A3 | A4 | A5 | A6 | A7 | Column Address |
| 2nd Cycle | A8 | A9 | A10 | A11 | A12 | A13 | A14 | A15 | Row Address (Page Address) |
| 3rd Cycle | A16 | A17 | A18 | A19 | A20 | A21 | A22 | A23 | A24 |

图 1 (小块)存储结构示意图

表 1 各个引脚的有效电平及功能

| 引脚 | 有效电平 | 功能 |
|-------------|------|---------|
| I/O 0~7(15) | 高/低 | 数据输入/输出 |
| /CE | 低 | 片选使能 |
| CLE | 高 | 命令锁存使能 |
| ALE | 高 | 地址锁存使能 |
| /RE | 低 | 读允许引脚 |
| /WE | 低 | 写允许引脚 |
| /WP | 低 | 写保护引脚 |
| /SE | 低 | 备用区允许引脚 |
| R/B | 高/低 | 准备好/忙输出 |

1.3 与微处理器的连接及驱动程序

当前嵌入式领域的主流处理器当属 ARM。图 2 是 NAND Flash 与 ARM 处理器的一般连接方法。与 NOR Flash 不同, NAND Flash 需要驱动程序才能正常工作^[2,4]。图中 PB4, PB5, PB6 是 ARM 处理器的 GPIO 口, 可用来控制 NAND Flash 的片选信号。CS1 是处理器的片选信号, 低电平有效。IORD、IOWR 分别是处理器的读、写信号, 低电平有效。写保护信号接高电平有效, 接低电平无效。

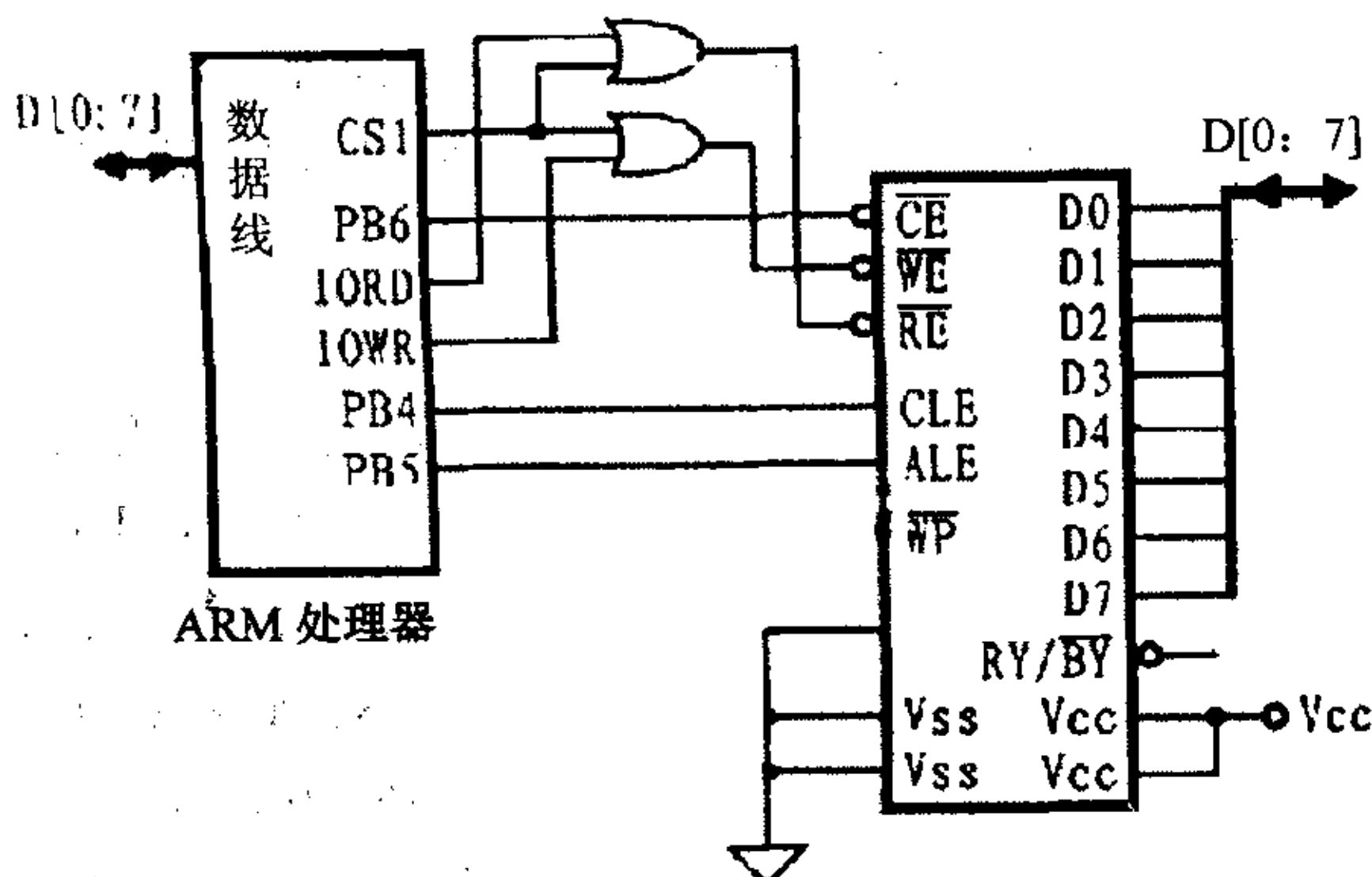


图 2 NAND Flash 与 ARM 处理器的连接图

驱动程序中的读操作的实现函数伪代码如下:

```
Read_func(cmd, addr)
```

```
{ //以下五条语句是为发送命令所做的必要设置
```

```
    读禁止;
    地址锁存禁止;
    命令锁存允许;
    写允许;
    片选使能;
    Send_cmd(cmd); //发送命令, 由参数
    决定, 这里为 00
    写命令禁止;
    片选无效;
    命令锁存无效; //发送命令结束
    地址锁存有效; //开始发送地址
    For(i=0; i<4; i++) //要四个周期发
```

```
送地址
```

```
    写命令允许;
    片选有效;
```

```
    Send_add(addr); //发送地址
```

```
    写命令禁止;
    片选无效;
```

```
    //所有数据发送结束, 等待读取数据
```

```
    片选允许;
```

```
    写命令禁止;
```

```
    地址锁存禁止;
```

```
    延时 2 毫秒;
```

```
    如果忙则等待;
```

```
    Read_data(buf) //开始读取数据
```

1.4 固有特性

NAND Flash 自身有一些特性^[1], 导致其不能像普通磁盘那样进行操作。主要特性如下:

- 1) 出厂时可能存在坏块, 并且使用过程中也会有坏块出现。
- 2) 按页写, 按块擦除。
- 3) 写操作只能在空或已擦除的单元内进行。
- 4) 块的擦写寿命有次数限制。
- 5) 块擦除时间与页读写时间相比十分长。

需要通过软件的支持来消除这些特性导致的不足, 使得能像操作普通磁盘一样进行 Flash 操作。

2 NAND Flash 和 NOR Flash

除了不同的硅单元设计外, NANA Flash 和 NOR Flash^[2,3]的最重要的不同之处在于总线接口, NOR Flash 像其他的存储器如同步随机访问存储器 SRAM 等一样直接接到地址/数据总线上, 而使用复杂的 I/O 接口并配备一些额外的控制引脚。NAND Flash 是适合海量存储应用的顺序存取设备, 而 NOR Flash 是适合代码存储应用的随机存取设备。NOR Flash 常用于代码的存储并在其上执行, 代码也能存储在 NAND Flash 中, 但不能在 NAND Flash 中直接执行, 需将代码加载到随机存储器 RAM 中

才能运行执行。NAND Flash 和 NOR Flash 的比较参考如表 2 所示。由于 NAND Flash 比 NOR Flash 具有更便宜和更灵活的接口,因此是应用于大容量非易失性存储应用的最佳选择,这些应用包括固态文件存储、数字视频/音频录象/音机、数字静态摄影机和一些要求非易失性的便携式应用。

表 2 NAND Flash 与 NOR Flash 的性能比较

| | NOR | NAND |
|-------------|---------|--------------|
| 接口 | 地址/数据总线 | 复杂 I/O |
| 芯片大小 | 大 | 小 |
| 芯片价格 | 高 | 低 |
| 读速度 | 快 | 慢 |
| 单字节编程 | 快 | 慢 |
| 多字节编程 | 慢 | 快 |
| 擦除时间 | 长 | 短 |
| 耗能 | 高 | 低 但需要额外的 RAM |
| 能否直接执行代码 | 能 | 不能 |
| 最大擦写次数 | 十万次 | 百万次 |
| 在运输时是否允许有坏块 | 不允许 | 允许 |
| 单元密度 | 低 | 高 |

3 NAND Flash 的类型和所支持的文件系统

有多种类型的 NAND Flash^[2,5]可见到,比较常见的有:裸体 NAND 芯片、智能媒体卡(SmartMediaCards)、片上磁盘(DiskOnChip)、压缩卡(CompactFlash)、安全数字(Secure Digital)、PC Cards 和 MMC 存储卡等。

智能媒体卡(SmartMediaCards)是在裸体 NAND 芯片的表面覆盖了一层薄塑料。它在数字相机和 MP3 播放器中相当常见,这个卡本身并不具有智能,它是由软件来具备智能的。

片上磁盘(DiskOnChip)是一种附加有胶合逻辑的 NAND Flash。这种胶合逻辑给一个小地址窗口提供了直接访问内存的路径,这个地址窗口包含了从 NAND Flash 中加载真正引导代码的一个引导加载器。该逻辑还包括了控制静态 NAND Flash 芯片控制线和一个硬件错误更正码产生器的控制寄存器。

使用 NAND Flash 的一个主要的问题是你不能像你通常想象的那样对一页进行写。在擦除之前,对一页的连续的写被限制在 1 到 3 次,具体限制几次由厂商的说明决定。这也同样使用于备用区。这就使得它有必要由文件系统来处理写缓冲,写缓冲中包含的数据少于一页的数据。

现在仅有几个类型的文件系统支持 NAND,它们是:

●JFFS2 文件系统和 YAFFS 文件系统:这两个文件系统支持 bare NAND Flash 和 SmartMediaCards。

●NTFL 文件系统:该文件系统支持 r DiskOnChip devices。

●TRUEFFS 文件系统:该文件系统是由 M-Systems

公司为支持 DiskOnChip devices 而开发的。

●SmartMedia DOS - FAT:该文件系统是由 SSFDC Forum 定义的。

4 闪存固件程序架构

闪存固件程序^[1]的需求来源于 Flash 本身的物理机制(用 I/O 接口传送控制命令、地址和数据信息),和其固有特性导致的不足。所以在考虑软件设计的时候,应该区别基本的读/写/擦操作和高一级的用于磁盘仿真和闪存管理的算法。Flash 基本的读/写/擦操作,通常称作为内存技术驱动程序(MTD, Memory Technology Devices);高一级的磁盘仿真和闪存管理,通常称作为闪存转换层(FTL, Flash Translation Layer)。FTL 层的管理程序调用 MTD 接口进行数据的存取,闪存固件程序的架构如图 3 所示。

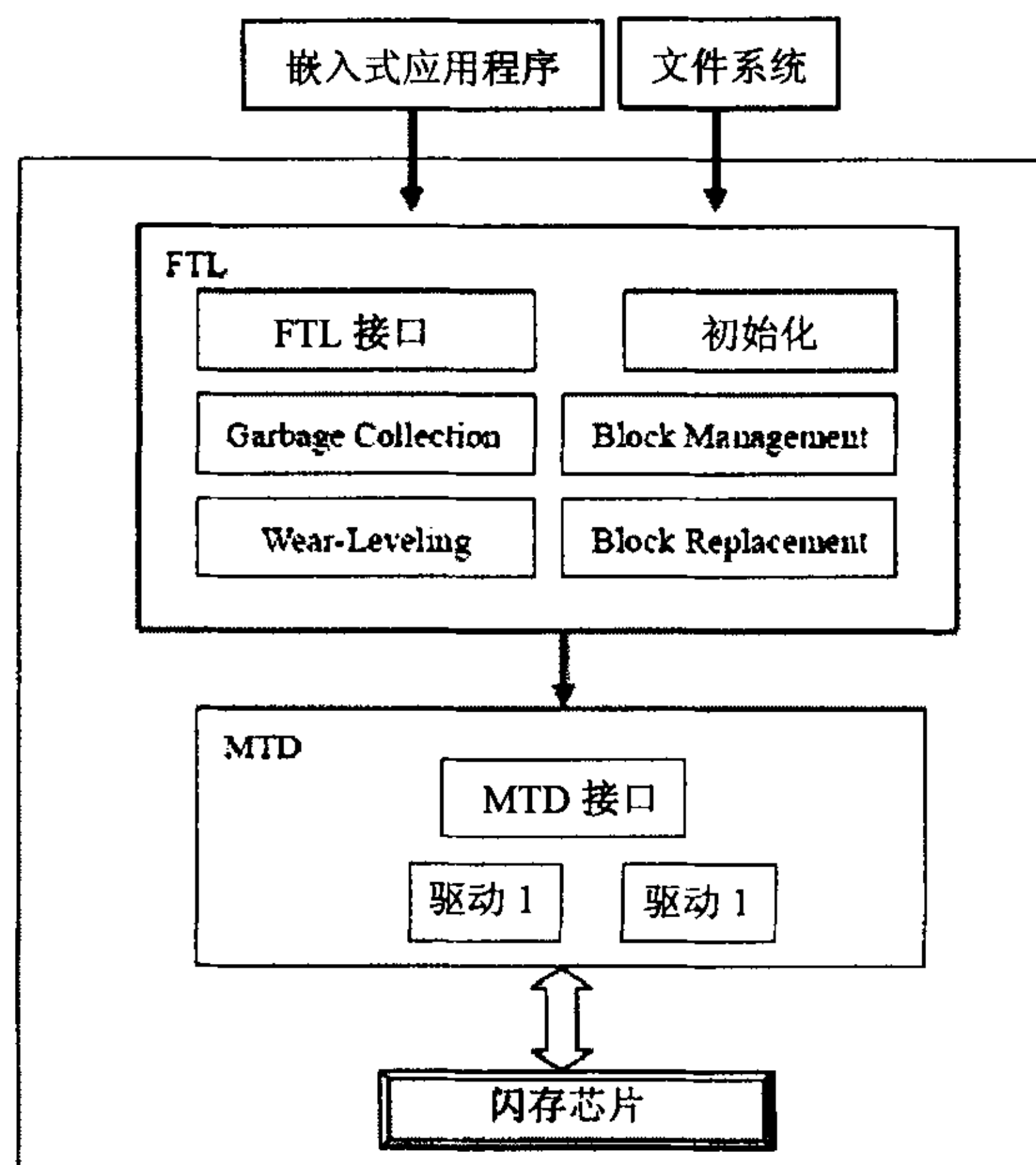


图 3 闪存固件程序架构

5 Linux 系统对 NAND Flash 的支持

Linux 操作系统虽然已经支持 NAND Flash,但要使用 NAND Flash 设备,还必须先对内核进行设置,方法如下^[2,4]:

(1)在 /usr/src/(内核路径名)目录中输入 make menuconfig 命令,再打开主菜单,进入 Memory Technology Devices(MTD)选项,选中 MTD 支持。

(2)进入 NAND Flash Device Drivers 选项,NAND 设备进行配置。不过此时对 NAND 的支持仅限于 Linux 内核自带的驱动程序,还有一些芯片的驱动程序没有包含,比如 Toshiba 芯片,为此需要对 Linux 内核进行修改,以 Toshiba 芯片为例,方法如下:

①修改内核代码的 drivers.in 文件,添加下面一行:

```
dep -- tristate 'Toshiba NAND Device Support' CONFIG-
MTD - TOSHIBA $ CONFIG - MTD
```

(下转第 209 页)

能进行数据交换。

数据处理模块负责测试金属液的各种元素含量,需要进行大量的计算和复杂的判断。根据不同的金属液种类处理模块调用相应的专家系统模块,进行针对性的分析。采集的温度变化与理想状态有很大差别,当温度取样过程中出现噪音时检测仪要对输入温度曲线进行数字化滤波,对多种情况进行智能识别,从而获取真实的特征信息。

其余模块如数据初始化、打印、输出等,分别完成资料信息和用户设定的输入、打印结果信息、USB 以及网络传送结果信息等功能。

3 结束语

文中针对智能检测仪的设计进行了分析,根据功能需求和现场的限制提出完整的设计方案。目前此系统已经在诸多铸造领域发挥着重要的作用,为嵌入式系统在工业仪表仪器方面的应用提供一个可借鉴的模型。随着嵌入式技术的不断发展,嵌入式系统的应用在工业铸造领域的应用必将更为广泛和深入。

(上接第 202 页)

- 现[J]. 计算机应用研究,2002,19(5):82-84.
- [3] Harp S. Towards the Genetic Synthesis of Neural Networks [C]// In Proc 3rd Int Conf Genetic Algorithms. [s. l.]: [s. n.], 1999:360-369.
- [4] 孙宏,杜文. 航空公司飞机排班问题的分阶段指派算法[J]. 系统工程学报,2003,18(2):35-36.
- [5] 赵岩,赵慧娟. 数据挖掘理论与技术[J]. 福建电脑,2006(2):54-55.
- [6] 刘同明. 数据挖掘技术及其应用[M]. 北京:国防工业出版社,2001.

(上接第 205 页)

其中 CONFIG-MTD-TOSHIBA 是该设备的名称,将在 Makefile 文件中用到。\$CONFIG-MTD 的意思是只有选择 \$CONFIG-MTD 时,该菜单才会出现,即依赖于 \$CONFIG-MTD 选项。Toshiba 的 NAND 设备将被加入 Linux 系统内核菜单中。

②修改相应的 Makefile 文件,以便编译内核时能加入该设备的驱动程序。

```
obj-$(CONFIG-MTD-TOSHIBA) += toshiba.o
```

此行语句的意思是如果选择了该设备,编译内核时加入 toshiba.o(假设驱动程序是 toshiba.o),反之不编译进内核。

6 结论

NAND Flash 这种高密度、存取性能高、单片容量不断增加的存储器可满足对大批量数据存储的需求;针对

参考文献:

- [1] 张大波. 嵌入式系统原理、设计与应用[M]. 北京:机械工业出版社,2004.
- [2] Bovet D P, Cesati M. Understanding The Linux Kernel[M]. San Francisco:O'Reilly & Associates,2000.
- [3] Andersen E. How do I build a Busy Box-based system[EB/OL]. 2006. <http://www.busybox.net>.
- [4] 郭晋峰. RTLinux 的原理及其实时程序开发[J]. 微型机与应用,2001(4):8-10.
- [5] Blanchette J, Summerfield M. C++ GUI Programming with Qt 3[M]. Englewood:Prentice Hall PTR,2004.
- [6] Bell D I. Nano-X Programming Tutorial[EB/OL]. 2006. <http://www.microwindows.org/Nano-XTutorial.html>.
- [7] 魏永明. MiniGUI 编程指南[M/CD]. 北京:飞漫软件技术有限公司,2003.
- [8] Advantech Co. Ltd. ADAM 4000 Data Acquisition Modules User's Manual[EB/OL]. 2005. <http://support.elmark.com.pl/advantech/ia/ADAM4000/MANUAL/Adam-4000-ed10.pdf>.

- [7] 陈国良. 遗传算法及其应用[M]. 北京:人民邮电出版社,1999.
- [8] Herrea F, Lozano M. Adaption of genetic algorithm parameters based on fuzzy logic controllers[C]//in Herrea F, Verdegay J. Genetic Algorithms and Soft Computing. [s. l.]: [s. n.], 1996:95-125.
- [9] 杜红亮,张洪伟,王琦. 基于遗传算法的多目标定位技术[J]. 电脑开发与应用,2005,18(1):29-30.
- [10] 黄岚,庞巍. 基于遗传算法求解带时间窗的车辆路由问题[J]. 小型微型计算机系统,2005,26(2):214-216.

Flash 固有特性的程序设计能有效地消除 NAND Flash 自身的缺陷,满足对数据存取高可靠性的要求。

参考文献:

- [1] 李斌,薛质,潘玉华. NAND FLASH 在基于 CCM3118 税控收款机上的应用[J]. 中国集成电路,2005(7):64-67.
- [2] 袁文菊,孙天泽,付晓江. 大容量 NAND Flash TC58DVG 02A1FT00 在嵌入式系统中的应用[J]. 国外电子元器件,2004(12):63-65.
- [3] Samsung Electronics Datasheet Samsung-K9F5608(16)q(u)0b[R]. [s. l.]: [s. n.], 2004.
- [4] 李驹光. ARM 应用系统开发详解[M]. 北京:清华大学出版社,2004.
- [5] Yagbmour K. 构建嵌入式 LINUX 系统[M]. 韩存兵,龚波,译. 北京:中国电力出版社,2004.