

44B0X 数据处理装置的存储系统设计

李双全, 吕宁, 武俊峰

(哈尔滨理工大学自动化学院, 哈尔滨 150080)

摘要: 基于嵌入式系统, 采用大容量 Nand 闪存芯片, 设计了数据处理装置的存储系统。介绍数据处理装置的体系结构和 Nand 闪存芯片的使用方法, 给出 Nand 闪存芯片与 S3C44B0X 的接口电路, 编写了嵌入式 uClinux 操作系统下 Nand 闪存芯片的驱动程序, 实现了具有掉电保护功能的存储系统。

关键词: 数据处理装置; 嵌入式系统; Nand 闪存; 存储系统

Design of Storage System for 44B0X Data Processing Device

LI Shuang-quan, LV Ning, WU Jun-feng

(School of Automation, Harbin University of Science and Technology, Harbin 150080)

【Abstract】 Based on embedded system and Nand flash chip technology, the storage system for data processing device is designed. The structure of data processing device and usage of Nand flash chip are introduced. The interface between Nand flash chip and S3C44B0X is diagramed. The driver program of Nand flash in the embedded uClinux system is provided. The storage system with automatic protection while shutting down is realized.

【Key words】 data processing device; embedded system; Nand flash; storage system

随着科技的飞速发展, 生产过程变得越来越复杂, 要求数据处理装置具备更高的处理速度、更多的功能集成和多样化的外部接口。随着计算机技术、信息技术和集成电路制造技术的进步, 人们已能设计复杂的数据处理装置。本文基于嵌入式 CPU 和嵌入式操作系统, 设计了具有多路万能信号采集、大容量存储系统、USB 接口、网络接口和 RS485 接口的数据处理装置, 实现了具有抗掉电功能的文件存储系统。

1 数据处理装置的硬件结构

数据处理装置的硬件结构如图 1 所示。

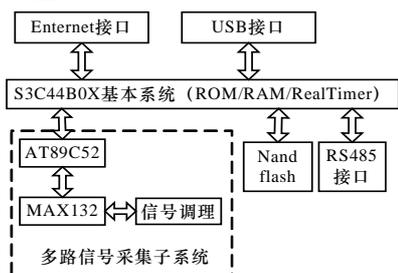


图 1 数据处理装置的硬件结构

1.1 S3C44B0X 主系统

数据处理装置的核心部分是嵌入式 CPU 及其基本的外围电路。三星公司 ARM7 系列的嵌入式芯片 S3C44B0X 是 32 位的 RISC 处理器, 其内部集成了动态 RAM 控制器、LCD 控制器、实时时钟、UART 控制器、定时器、中断控制器、DMA 控制器和大量的 I/O 端口^[1]。这些功能可以使电路设计得到极大简化, 同时嵌入式操作系统 uClinux 提供了对该处理器的支持, 使得应用程序能基于操作系统进行开发, 缩短了开发周期。在设计中选用 S3C44B0X 作为 CPU, 内部实时时钟作为系统的时间基准, 程序存储器选择了 2 MB 的 Flash 芯片 29LV160, 系统 RAM 采用了 4 MB 的 16 位动态 RAM 芯片 HY57V641620。

1.2 多路信号采集子系统

采用模块化设计思想, 单独设计了多路信号采集子系统。用 AT89C52 单片机作为数据采集子系统的 CPU, 采用 18 bit 串行接口的多斜率积分式的 AD 芯片 MAX132 作为数据采集器件, 辅以 PGA 信号放大电路和信号调理电路, 共同完成对工业现场常见模拟信号的采集和标度变换。数据采集子系统通过 RS422 接口和嵌入式主机系统进行信息通信。

1.3 网络系统

数据处理装置实现了网络化, 选择了 10M/100M 自适应网卡接口芯片 91C113, 它与 CPU 的接口简单, 在 uClinux 操作系统中加载 TCP/IP 协议驱动后, 即可实现局域网通信和 Internet 通信。

1.4 USB 接口

配置了 USB 接口, 通过 USB 接口可以实现采样数据导出, 还可以导入用户对数据处理装置的配置信息, 便于灵活配置数据通道和参数组态信息。

ISP1161A 是 PHILIPS 公司推出的一种遵循 USB2.0 规格的通用串行总线控制器。U 盘是 USB 设备接入数据处理装置, 因此, 设计中使用了 ISP1161A 的主机控制器, 将数据处理装置作为 HOST 主机使用, 并采用中断工作模式。

2 NAND 闪存芯片接口电路和驱动程序

2.1 闪存芯片的选择

数据处理装置的另外一个主要功能是数据存储, 选择合适的非易失性存储介质是至关重要的。NOR 和 NAND 是目

基金项目: 哈尔滨市科技攻关计划基金资助项目(2005AA1CG037, 2004AA1CG087)

作者简介: 李双全(1971—), 男, 副教授、硕士, 主研方向: 计算机控制, 过程控制和自动化仪表; 吕宁, 教授、硕士; 武俊峰, 教授、博士

收稿日期: 2008-03-24 **E-mail:** lshquan71@163.com

前 2 种主要的非易失闪存技术。

NOR 闪存的接口方式和 SRAM 相同，可以和 CPU 总线直接相连，主要作为代码存储介质。NAND 闪存的写入速度、单位体积的容量、使用寿命优于 NOR 闪存，并且采用 I/O 串行接口方式，适合于作数据存储介质。数据处理装置的实时数据基本上要 24 小时连续记录，要频繁地擦除和写入，因此，采用了寿命长、容量大、接口简单、写入速度快的 NAND 闪存^[2]。

2.2 NAND 闪存接口电路设计

K9F2808 是 8 bit 的 16 MB NAND 闪存，由 1 024 个块组成，每个块有 32 页，每页 528 Byte^[3]。其与 S3C44B0X 的接口电路如图 2 所示，nGCS4 是 CPU 的片选信号，其与 CPU 的读引脚 nOE 信号、写引脚 nWE 信号做逻辑运算后接入 K9F2808 的读写控制端，用 CPU 的 C 端口 GPC0~GPC4 引脚连接到 K9F2808 的串行时序引脚 RnB, nCE, CLE 和 ALE，用软件产生串行读写时序，K9F2808 的数据线 IO0~IO7 连接到 CPU 的数据总线 D0~D7。

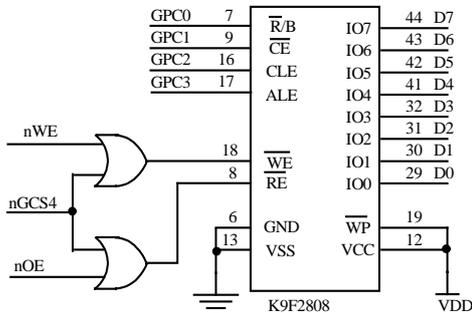


图 2 K9F2808 与 S3C44B0X 的接口电路

2.3 NAND 闪存的 uClinux 驱动程序设计

在 uClinux 系统中使用 K9F2808 芯片需要编写其驱动程序，设置其必要的初始化信息。uClinux 操作系统把 NAND flash 作为块设备，通过 MTD(Memory Technology Devices) 系统实现对 NAND 闪存的管理和使用。NAND 闪存芯片与 MTD 块设备驱动之间的关系如图 3 所示。

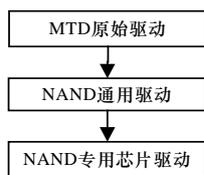


图 3 Nand 与 MTD 之间的关系

由图 3 可知，只需按照一定规则重新编写 NAND 专用芯片驱动层即可实现 K9F2808 芯片的驱动，具体方法如下：

(1) 在 drivers/ mtd/ nand/ 目录下新建一个文件，文件名为 K9F2808.c。

(2) 在 K9F2808.c 中加入 flash 分区定义代码，将 K9F2808 的存储空间分为一个 16 MB 分区，代码如下：

```
const static struct mtd_partition partition_info[] =
{ name: "Nand flash partition 1",    offset: 0,
  size: 16*1024*1024 };
```

(3) 根据实际电路连接，见图 2，定义 K9F2808 的串行时序引脚和地址，具体代码如下：

```
#define DATA_CLE (1<<2)    // CLE 引脚
#define DATA_ALE (1<<3)    // ALE 引脚
#define DATA_NCE (1<<1)    // NCE 引脚
```

```
#define NAND_IO_BASE 0x01d20000 //IO 地址
```

(4) 根据 K9F2808 时序图，编写操作和初始化函数。

(5) 将文件加入 uClinux 内核系统。在 drivers/mtd/nand/ makefile.common 文件中加入代码行：obj-\$(CONFIG_MTD_NAND) += K9F2808.o；在 drivers/mtd/nand/config.in 文件中找到包含文字 if ["\$CONFIG_ARM" = "y"]; then 的行，在其下加入：

```
dep_tristate "K9F2808 Device support"
CONFIG_MTD_NAND_ARM
$CONFIG_MTD_NAND
```

3 具有抗掉电功能的文件系统

3.1 文件系统的选择

数据处理装置的数据文件系统主要用来存储工业现场的实时采样数据和系统的重要配置信息，其基本要求是稳定可靠，并具有抗掉电功能。

个人计算机的文件系统以 FAT32 为主。FAT32 组织文件的格式是在存储介质的特定位置划分出一片空间用来记录文件数据的具体存放位置，这片空间称为文件分配表(FAT)。操作系统访问文件是通过 FAT 来定位具体的文件数据，因此，只要 FAT 被破坏，即使文件中的数据仍然保存在存储介质上，系统也无法找到其位置，根本无法组织起有效文件。FAT 文件系统不具备抗掉电功能，如果操作系统正在写入某文件的 FAT 时突然掉电，会使 FAT 被破坏，从而导致该文件全部数据丢失。

日志型文件系统具有抗掉电功能。uClinux 操作系统支持的日志型文件系统主要有 JFFS2 文件系统和 YAFFS 文件系统，其中，YAFFS 文件系统是专门针对 NAND 闪存优化设计的嵌入式文件系统，是最优的选择方案^[4]。

以 K9F2808 闪存为例，YAFFS 文件系统是按页写入文件数据的，每次写入一页 528 Byte 信息，其中，512 Byte 是文件的数据，另外 16 Byte 是文件的标签，表示该页数据属于哪个文件以及该页数据在文件中的位置。在任何情况下突然掉电，即使操作系统正在写入文件，最严重的情况也只是损失了一页数据，而不会导致整个文件数据的丢失，因此，YAFFS 文件系统具有抗掉电功能。

3.2 YAFFS 文件系统移植

uClinux 系统的文件系统结构如图 4 所示。YAFFS 文件系统处于虚拟文件系统 VFS 下层，移植工作需要将 YAFFS 源代码配置在 uClinux 内核的 VFS 系统模块下，编译出 YAFFS 文件系统模块。

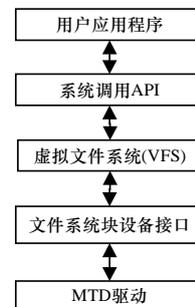


图 4 uClinux 文件系统的结构

将 YAFFS 移植到 uClinux 2.4 内核中的步骤如下：

(1) 在内核中建立 YAFFS 目录 fs/yaffs，并把下载的 YAFFS 代码复制到该目录里面。

(2) 参考 YAFFS 代码中的 Kconfig 文件，按照 uClinux 2.4 内

核风格修改 Config.in 文件, 加入以下代码:

```
if [ "$CONFIG_MTD_NAND" = "y" ]; then
tristate "Yaffs on NAND" CONFIG_YAFFS_FS
fi
dep_bool "yaffs" CONFIG_YAFFS_YAFFS
```

(3)按照 uCLinux 2.4 内核的风格修改 YAFFS 目录中的 Makefile 文件, 加入如下代码:

```
subdir-$(CONFIG_YAFFS_FS) += yaff
```

3.3 YAFFS 文件系统的配置

运行make xconfig命令启动uCLinux内核配置工具^[5], 在内核配置中做如下选择:

(1)在 Memory Technology Devices 配置中选中 MTD Partition Support;

(2)在 Nand Flash Devices Driver 配置中选中 Nand Devices Support, Verify Nand page writes 和 K9F2808 Device support;

(3)在 File System 配置中选中 Yaffs file system on NAND, Yaffs Enabled MTD, Lets Yaffs do its own ECC 和 Yaffs use old mtd。

完成以上配置和修改后, 保存内核配置, 编译内核, 烧写内核文件到程序存储器, 这样就形成了一个基于 NAND 闪存的 YAFFS 文件系统。

3.4 其他抗掉电措施

YAFFS 日志文件系统具有抗掉电功能, 但是掉电情况下还是可能丢失一页 512 Byte 数据。本文采用了其他的软硬件辅助措施来解决这个问题, 确保采样数据不会因为掉电而丢失。

用电源监视芯片监视系统的供电电源, 系统掉电时向 S3C44b0X 的 EINT0 引脚发出中断请求, 让 CPU 捕获掉电事件。电源监视电路如图 5 所示。系统的核心电源是 +3.3 V, 是由 +5 V 电源经过 LM1117 芯片稳压得到的。电源监视芯片监视 +5 V 电源, 当其低于 +4.7 V 时向 CPU 发出中断。如果系统能在电源从 +4.7 V 降低到 +3.5 V 之前将数据写入闪存, 就可以避免因掉电而丢失数据。接着编写掉电中断服务程序。CPU 捕获掉电中断后, 马上进入中断服务程序, 在中断服务程序内将文件缓冲区内的数据写入闪存。



图 5 系统的电源监视电路

4 测试与使用

使用本文设计的存储系统前, 先将 YAFFS 文件系统挂载在 VFS 中, 在 uCLinux 的 rc 文件中加入 mount -t yaffs /dev/mtdblock0 /disk, 这样每次启动操作系统后, 会自动挂载 YAFFS 文件系统。文件系统挂载后, 进行了以下测试工作:

- (1)使用命令 makeyaffs /dev/mtd0 格式化 K9F2808 为 YAFFS 文件系统;
- (2)使用 cp 命令进行文件拷贝测试;
- (3)应用程序通过调用标准 C 函数(fopen, fread, fwrite, fclose 等)实时记录采样数据;
- (4)运行中的掉电测试。

经过长时间连续运行测试, 并在运行中切断供电, 未发现存储系统工作异常, 掉电后也没有发现丢失数据的情况。

5 结束语

本文设计了基于嵌入式芯片和嵌入式操作系统的具有抗掉电能力的存储系统, 采用 NAND 闪存芯片和 YAFFS 文件系统, 添加了掉电监视的软硬件方法, 完善了存储系统的抗掉电能力, 为智能仪表存储系统的设计提供了一种可靠的解决方案。

参考文献

- [1] Samsung Inc.. S3C44B0X Risc Microprocessor Data Manual[Z]. (2002-08-21). <http://www.samsung.com>.
- [2] 罗 钧, 张 宇. 大容量 NAND FLASH 在 ARM 嵌入式系统中的设计与实现[J]. 现代电子技术, 2007, 30(11): 172-174.
- [3] Samsung Inc.. K9F2808U0C Flash Memory Datasheet(Rev2.5)[Z]. (2003-10-10). <http://www.samsung.com>.
- [4] 毛勇强, 黄光明. YAFFS 文件系统在嵌入式 Linux 上的实现[J]. 电子技术应用, 2006, 32(1): 96-98.
- [5] Yagbmour K. 构建嵌入式 Linux 系统[M]. 北京: 中国电力出版社, 2004.

(上接第 229 页)

表 2 DSP 基本信息

制程	SMIC 0.18 μm 1P6M CMOS
芯片面积	5 mm×5 mm
平均功耗	0.98 mW/MHz
峰值速度	150 MHz
片上内存	128 KB

为了验证本方法的可行性, 分别采用了传统方法和本方法进行实验。使用本文介绍的方法使芯片的后端工作时间从预定的 12 天缩短为 8 天, 功耗也有大约 20% 的减少, 如表 3 所示。

表 3 前后端协同设计和传统后端设计方法比较

方法	后端时间/天	动态功耗/(mW·MHz ⁻¹)
传统方法	12	1.20
前后端协同	8	0.98

芯片功耗和速度指标达到了设计要求, 这证明了前后端协同时钟树设计方法的可行性。

5 结束语

本文提出了一种前后端协同的时钟树设计和实现方法。通过在数字设计的前端考虑后端因素及其对后端结果的影响, 在后端实现中考虑前端的问题达到前后端结合, 以达到消除毛刺、提高性能和稳定度、降低功耗的目的。芯片的流片结果证明了本方法的有效性。

参考文献

- [1] Rabaey J M. Digital Integrated Circuits[M]. [S. l.]: Prentice Hall, 2002.
- [2] Cummings C E. Synthesis and Scripting Techniques for Designing Multi-asynchronous Clock Designs[EB/OL]. (2006-06-20). http://www.sunburst-design.com/papers/CummingsSNUG2001SJ_Async_Clk.pdf.
- [3] Crews M, Yong Y. Practical Design for Transferring Signals Between Clock Domains[M]. [S. l.]: Cluiuer Academic Publisher, 2002.
- [4] Chinnery D, Keutzer K. Closing the Gap Between ASIC and Custom: Tools and Techniques for High Performance ASIC Design[M]. [S. l.]: Kluwer Academic Publisher, 2002.