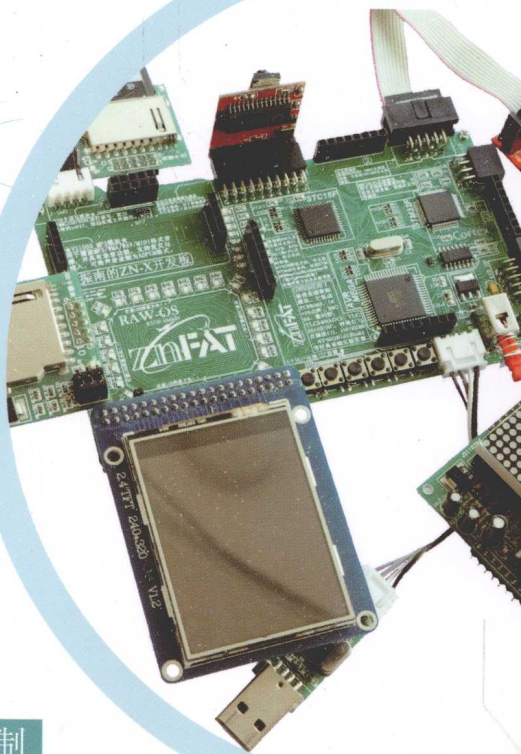


嵌入式 FAT32 文件系统设计与实现

——基于振南znFAT 下

于振南 编著



- 主攻FAT32的重点难点
- 详述znFAT的独特算法
- 独具匠心的创新思想与数据加速机制
- 深入讲解SD卡等存储设备底层驱动
- 呈现更加精彩的配套实验与工程实例



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

014037862

TP333.1
03
V2

全 部 容 内

嵌入式 FAT32 文件系统 设计与实现

——基于振南 znFAT(下)

于振南 编著



北京航空航天大学出版社

TP333.1

03

V2



北航

C1725952

内 容 简 介

本书是此套书的下册,是上册的延续与拓展。本书详细讲解了 FAT32 中长文件名的相关技术及其在 znFAT 中的具体实现。最后,着重介绍了 SD 卡等存储设备的驱动原理与调试方法。应该说,下册比上册更有技术含量、更有思想,会引发读者更多的思考和灵感。

如果说上册是专注于基础和常规内容的话,那么下册则更加侧重于提高与创新。振南将把一些绝对独特的思想和算法,以及它们在 znFAT 中表现出来的优异性能呈现在读者面前。

此书谨献给对 FAT32、SD 卡等嵌入式存储技术感兴趣,以及在这方面有项目应用需求的广大人群。希望此书能够成为此领域内的参考书,对大家的研究与开发工作产生积极意义。

图书在版编目(CIP)数据

嵌入式 FAT32 文件系统设计与实现:基于振南 znFAT.
下 / 于振南编著. — 北京:北京航空航天大学出版社,
2014.4

ISBN 978-7-5124-1510-2

I. ①嵌… II. ①于… III. ①微型计算机—文件系统—系统设计 IV. ①TP311.13

中国版本图书馆 CIP 数据核字(2014)第 042838 号

版权所有,侵权必究。

嵌入式 FAT32 文件系统设计与实现——基于振南 znFAT(下)

于振南 编著

责任编辑 董立娟

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316524

涿州市新华印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:16.25 字数:346 千字

2014 年 4 月第 1 版 2014 年 4 月第 1 次印刷 印数:3 000 册

ISBN 978-7-5124-1510-2 定价:36.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

套书前言

振南这几年一直在研究 FAT32 文件系统与 SD 卡、Flash 等嵌入式存储的相关技术,初衷一方面在于振南对它的浓厚兴趣,因为其中蕴涵着很多非常巧妙的设计思想和理念,通过更加系统化、工程化的开发实践,自身的水平得到了很大的提升;另一方面随着嵌入式存储技术的迅猛发展,基于通用嵌入式 FAT32 文件系统的应用与产品层出不穷,这催生了对这方面技术和功能的极大需求。振南希望通过自己的研究,让广大的爱好者与工程师轻松地进入这一领域,对大家产生一定的参考意义。

基于振南长期而深入的研究,一个精简、优秀而功能完备的嵌入式 FAT32 文件系统方案很早便应运而生,并一直维护至今,这就是振南的 znFAT。经过几年的推广与无数的验证,它已广为流传,在各个硬件平台、各个应用系统中发挥着其不可替代的作用。

为了能让 FAT32 文件系统的嵌入式应用更加深入人心、让更多人受益于它的方便与强大,振南花费三年时间特著此书,全面讲述其各种技术细节、实现技巧、创新策略与算法、应用实例、移植方法等。

振南知道无数人都在急切地盼望着此书问世,但因为振南精益求精的性格特点、做事原则以及写作、出版过程中一些琐碎的事情,使得此书的进度稍显滞后,请读者谅解。

套书内容特点

本套书共分为上、下两册,内容上循序渐进,步步引导,从基础到提高,从常规到创新,从保守到发散,从理论到实践,在把原理与方法讲清楚之后,又基于配套的 ZN-X 开发板做了大量的实验,努力让读者开卷有益,真正有所感悟与收获。

上册侧重于入门与基础,首先通过几个实例让读者初步认识 FAT32 与 SD 卡,随后便全面展开了系统化的讲解,实现了几个基本的文件操作功能,并对 znFAT 的一些简单应用进行了介绍;下册侧重于创新、拓展与提高,振南将继续带领读者实现更多、更具特色的文件操作功能,更重要的是提出了 znFAT 中一些独创的核心算法,并展示了它们在提高系统性能与加速数据读/写方面所起到的重要作用。同时,配套了一些更为精彩的实验,它们绝大部分都是振南的原创,有助于读者掌握、提高。书中包含的实验大部分是基于振南的 ZN-X 开发板的,而且由于 ZN-X 对 51、AVR、STM32 全系列 CPU 芯片的支持,这些实验很多都是跨平台的,这使其更加精彩,也



更加突显了 znFAT 极强的可移植性与实用价值。

关于振南的 znFAT

振南的 znFAT 是一套高效、完备、精简且具有高可移植性的嵌入式 FAT32 文件系统解决方案,主要特性如下:

- 与 FAT32 文件系统高度兼容,提供丰富的文件操作函数,可实现文件与目录的创建、打开、删除,数据的读取与写入等功能。
- 可方便地移植到多种嵌入式 CPU 平台上,如 51、STM32、AVR、PIC、Cold-File、MSP430 等。
- 占用 RAM 与 ROM 资源极少,并可由读者灵活配置。
- 最小配置情况下, RAM 的使用量为 800~900 字节,最大配置下为 1 300 字节。
- 内建独特的数据读/写加速算法以及多种工作模式,均可由用户自行配置,以满足不同的速度与功能需求。
- 创新性地提出数据写入的实时工作模式,可保证写入到磁盘文件中的数据安全,防止因恶劣工作环境、干扰或其他原因引起的目标平台不可预见的死机或故障,而造成数据丢失(实时模式数据写入速度不高,所有数据直接写入物理扇区,而不在 RAM 中暂存,并对文件数据进行实时维护)。
- 底层提供简单的单扇区读/写驱动接口以及可选的硬件多扇区读、写、擦除驱动接口。(在提供硬件多扇区读、写、擦除驱动的情况下,磁盘格式化与数据读/写速度有近 2~4 倍的提升,甚至更高。)
- 提供清晰而强大的功能模块裁减功能,可极大程度地减小最终生成的可执行文件的体积,节省 ROM 资源。
- 提供数据读取重定向功能,使读到的数据无需缓存,直接流向应用。
- 支持长文件名,长文件名最大长度可配置。默认配备并使用 GB2312 中文字符,并可选择是否使用 OEM 字符集,以减少程序体积。方便扩展更多的 OEM 字符集,如日文、韩文等。
- 支持与 Windows、Linux 等操作系统兼容的路径表示,路径分隔可使用/或\;支持无限深目录,支持长名目录。
- 支持对存储设备的格式化,文件系统为 FAT 32,格式化策略为 SFD(即无 MBR)。
- 支持 * 与 ? 文件名通配,长名也支持通配。
- 支持文件与目录的删除,支持非空目录删除。
- 支持无限深级目录创建。
- 支持“多文件”,即同时可对多个文件进行操作。
- 支持“多设备”,即同时可挂载多种不同的存储设备,可在多种存储设备间任意切换。

配套资料

本书配套资料包括以下内容:视频方面,《振南的 FAT32 文件系统入门教程》,共 5 集,时长约 138 分钟;《单片机基础外设九日通》,共 10 集,时长约 553 分钟;《振南带你从零学单片机之 C51 编程》,共 3 集,时长约 278 分钟;《单片机高级外设系列之 VS1003 MP3 解码器》,共 2 集,时长约 103 分钟;《单片机高级外设系列之 TFT 液晶》,共 2 集,时长约 73 分钟;《单片机高级外设系列之 SD 卡》,共 2 集,时长约 73 分钟;《单片机高级外设系列之 HZK 汉字库》,共 2 集,时长约 66 分钟;还包含了与振南的 znFAT 相关的很多实验和实例;以及 ZN-X 开发板配套资料等。详细请通过振南的个人网站和相关的网络平台免费获取,也可以通过这些平台与作者实时互动:

振南的个人主页: www.znmcu.cn (备用网址: bk.znmcu.cn)

论坛: bbs.znmcu.cn

21IC 论坛中振南的 znFAT 个人专区: 21icbbs.znmcu.cn

博客: 21icblog.znmcu.cn

导 读

这套书分上、下两册,共有二十余章,各章在知识上前后关联、相辅相成完整严密,而且适当地进行了实验的穿插,从而使书在整体上显得浑然一体、生动耐读。为了方便读者快速转到自己的兴趣点,振南特设置了导读部分:

- ① 想了解 znFAT 的移植与使用方法,请参见上册的附录《znFAT 的移植与应用》。
- ② 想了解 SD 卡物理扇区读/写驱动的细节、具体实现与调试方法,请参见下册第 12 章的《高性能 SD 卡物理驱动的实现》。
- ③ 想欣赏或借鉴基于 znFAT 的精彩实验与工程应用,请参见上册的第 13 章《真知实践,精彩展现》、下册的第 9 章《青涩果实,缤纷再现》。
- ④ 想了解研究 FAT32 文件系统的意义、用途与基本的入门,请参见上册的第 1 章《端倪初现,实验切入》、第 3 章《逐渐深入,转入正题》与第 4 章《自建模型,会意由衷》。
- ⑤ 想了解振南的 ZN-X 开发板及其使用、测试方法,请参见上册的第 1 章《欲善其事,必利其器》。
- ⑥ 想了解常用的主流存储设备、NOR/NAND FlashROM 上的文件系统应用,请参见下册的第 10 章《物理设备,闪存解惑》。
- ⑦ 想了解 znFAT 的具体性能表现,如数据读/写速度等,以及与几种优秀方案的较量,请参见下册的第 5 章《模式变换,百花争艳》。
- ⑧ 想了解 znFAT 各功能、创新点的具体实现细节与开发方法,希望对 FAT32 进行全面深入的研究,请参见书中各章主要内容,更加细致的介绍与引导请详见目录与各章节内容。

振南
2014.2

前言

读上册后继续启程

读了本书上册之后,你已可称得上是半个专家了,对 FAT32 文件系统、振南 znFAT 的设计思想与实现技巧,以及嵌入式存储的相关技术都已经小有领悟。但振南只能说现在还只是“万里长征”刚走出了不到一半。上册只是实现了“读”操作相关的功能,实际上真正的重点、难点和亮点在于“写”操作,还有在整个系统层面对性能与效率的优化与提升。这其中所产生的一系列新颖的编程方法、独特的算法与策略才是能够发人深省的核心内容。

下册之精妙之处

本书介绍知识的同时注重穿插大量精彩、更具创意的实验,方便读者理解、动手提高。首先介绍了文件与目录创建、数据写入等文件操作功能的实现。随后的几章一直在努力提升数据读/写的速度,提出了诸如预建簇链、连续扇区优化、压缩簇链缓冲与扇区交换缓冲等多种实用而优秀的创新思想与机制。这些正是 znFAT 中的精妙所在,它们让 znFAT 可以满足更高更快的数据存储应用需求,从而跻身于优秀的嵌入式 FAT32 文件系统方案之列,受到人们的广泛承认与好评。为了证明这一切,振南将 znFAT 与多款现有的国际优秀方案进行了全面、深入的“较量”,一决高下。

振南还详细讲解了 SD 卡等存储设备的驱动与调试方法,基于 ZN-X 开发板采用 3 种 CPU 分别对扇区读/写速度进行了实际的测试评估。尤其在 STM32 上为驱动引入了 DMA 后,它的速度表现更加让人满意,甚至令人惊喜。

后面几章中振南还详细介绍了广受读者关注的 FAT32 长文件名和 NOR/NAND Flash ROM 上的文件系统应用等问题。最后,集中展示了几个吸引眼球的 znFAT 的综合应用实验,可谓是本书的点睛之笔。

本书特点

在风格上,下册与上册保持了一致,并保证了知识脉络的连贯性。正如上册对后续内容的不断引导一样,下册在很多关键点上均与上册遥相呼应,共同编织了一张完整而致密的知识网络。应该说,下册比上册更具创意,更具技术含量,也更具工程实用价值。其中的很多实验可能都是一些“可遇而不可求”的高难、精彩创新实验,其中



涉及的很多技术对于实际的工程项目也都具有借鉴与指导意义。

感谢

znFAT 系统的研发、测试与改进工作以及本书的整个写作出版过程,从头到尾振南都不觉得寂寞,因为有无数的热心人和爱好者的协助、支持,这里一并表示感谢。

感谢导师顾国昌教授(哈尔滨工程大学计算机学院院长、博士生导师),正是因为他的谆谆教导和对振南自主研究工作的长期支持,才有了振南的今天。

感谢北京航空航天大学出版社的大力支持,这是本书最终得以出版的主要推动者与执行者;感谢 21IC、EDNChina、Elecfans 等网站与论坛,为振南个人与 znFAT 的推广起到了很大的作用,并且为技术的交流与反馈提供了良好的渠道与平台。

此外,大量的志愿者承担了测试工作,这里一并表示感谢。这些志愿者包括:杜撰、何强、吴俊超、谢明鑫、王志诚、林麟、罗伟东(纬图虚拟仪器)、尚学成、刘磊等。

本书的写作过程中,有很多人参与到了振南的内部书稿评阅中,从读者的角度提出了自己的意见和建议,也对他们表示感谢,包括:王坤、徐茂龙、黄劲松、曾跃飞、张杰、陈宏洲、许江等。

于振南 2014.3.6

点 评 本

本书是一本关于 FAT32 文件系统的书籍,主要介绍了 FAT32 文件系统的原理、设计、实现和应用。本书内容详实,图文并茂,是一本不可多得的好书。本书不仅适合从事嵌入式系统开发的工程师阅读,也适合对文件系统感兴趣的读者阅读。本书的出版,为 FAT32 文件系统的设计与实现提供了有益的参考。



目 录

第 1 章 数据记录,偷梁换柱:使用变通方法实现文件数据存储	1
1.1 把 SD 卡用作一个大容量的 ROM	1
1.1.1 大 ROM 思想的提出	1
1.1.2 思想的验证:数据采集与记录实验	2
1.2 数据“偷梁换柱”——数据替换	5
第 2 章 更及核心,文件创建:修改 FAT 表实现文件创建功能	10
2.1 文件的创建	10
2.1.1 文件目录项的构造	10
2.1.2 文件目录项的“落定”:写入目录簇	12
2.2 为自己开路:簇链的构造	15
2.2.1 目录簇的拓展	15
2.2.2 寻找“路石”:空簇的查找	16
2.2.3 形同虚设的 FSINFO 扇区	19
2.2.4 簇链构造的实现	22
2.3 目录的创建	24
2.3.1 目录项的构造	24
2.3.2 两个特殊的目录项	25
第 3 章 数据写入,细微可见:数据写入的实现	28
3.1 初步实现	28
3.1.1 回顾数据读取	28
3.1.2 从开头写数据	29
3.1.3 从整簇写数据	32
3.2 数据写入的实现	33
3.3 数据写入的典型应用	38
3.3.1 实例 1:数据采集与存储	38
3.3.2 实例 2:简易数码相机	41



第 4 章 巧策良方,数据狂飙:独特算法实现数据高速写入	45
4.1 迫出硬件性能	45
4.1.1 连续多扇区驱动	45
4.1.2 多扇区抽象驱动接口	47
4.2 为数据作“巢”	48
4.2.1 预建簇链思想的提出	48
4.2.2 簇链预建的实现	50
4.2.3 将多扇区用到极致	55
4.3 CCCB(压缩簇链缓冲)	57
4.3.1 CCCB 的提出	57
4.3.2 CCCB 的实现	60
4.3.3 CCCB 的争抢与独立	63
4.4 EXB(扇区交换缓冲)	64
4.4.1 EXB 的提出	64
4.4.2 EXB 的实现	65
第 5 章 模式变换,百花争艳:znFAT 与其他 FAT 的全面 PK	67
5.1 登顶效率之峰	67
5.2 与强者竞速	68
5.2.1 国内外优秀 FAT 方案简介	69
5.2.2 速度的“较量”	71
5.3 znFAT 的工作模式	74
5.3.1 缓冲工作模式	74
5.3.2 自身模式较量	75
5.4 znFAT 的功能裁减	78
5.4.1 功能裁减宏	78
5.4.2 裁减宏的嵌套	79
第 6 章 创新功能,思维拓展:多元化功能特性与数据重定向的实现	81
6.1 多元化文件操作	81
6.1.1 多文件	81
6.1.2 多设备	82
6.2 数据重定向	87
6.2.1 数据重定向的提出	87
6.2.2 数据重定向的实现	88

6.2.3 数据重定向实现 MP3 播放	89
第 7 章 层递删截,通盘格空:文件、目录的删除及磁盘格式化	91
7.1 文件数据的倾倒	91
7.1.1 何为数据倾倒	91
7.1.2 数据倾倒的实现	92
7.2 文件的删除	95
7.2.1 文件删除的实质	95
7.2.2 文件删除的实现	97
7.3 目录的删除	99
7.3.1 目录删除的难处	99
7.3.2 目录删除的实现	100
7.4 格式化	101
7.4.1 格式化的内涵	101
7.4.2 格式化的核心工作	101
7.4.3 格式化的实现	105
第 8 章 突破短名,搞定长名:突破 8·3 短名限制,全面地实现长文件名	110
8.1 FAT32 的长文件名	110
8.1.1 何为长文件名	110
8.1.2 长文件名的存储机理	111
8.2 UNICODE 编码	113
8.2.1 “各自为战”的 DBCS	113
8.2.2 UNICODE 带来的问题	115
8.2.3 编码转换的实现	116
8.2.4 长名的提取与匹配	120
8.3 长名的核心是短名	121
8.3.1 微软长名专利风波	121
8.3.2 长短名的绑定	122
8.3.3 用长名打开文件	123
8.3.4 创建长名文件	125
第 9 章 青涩果实,缤纷再现:套书的第二个实验专题	131
9.1 数据采集导入 EXCEL	131
9.2 串口文件“窃取器”	134
9.3 录音笔	136



9.4 简易数码相机(51)	139
9.5 简易数码录像机(AVR)	141
9.6 简易数码录像机升级版(STM32 直接录制 AVI 视频)	144
9.7 文件无线传输实验	147
9.8 嵌入式脚本程序解释器	150
9.9 AVI 视频播放器	152
9.10 绘图板实验——基于 STM32F4	154
9.11 MEMS 声音传感器录音实验	158
9.12 各种 CPU 上的实例汇总(基于第三方实验平台)	161
第 10 章 存储设备,闪存解惑:主流存储设备及闪存技术详解	173
10.1 当前主流存储设备	173
10.1.1 主流存储设备简介	174
10.1.2 嵌入式存储设备	177
10.2 FlashROM 上的文件系统	179
10.2.1 FTL	180
10.2.2 NFTL	180
第 11 章 物理驱动,深入剖析:SD 卡物理驱动	183
11.1 SD 卡的接口与电路	183
11.1.1 SD 卡的接口	183
11.1.2 SD 卡的电路	184
11.2 振南 SD 卡驱动移植与测试	185
11.2.1 振南 SD 卡驱动简介	185
11.2.2 振南 SD 卡驱动移植	185
11.2.3 SD 卡驱动测试	187
11.3 SD 卡驱动原理	188
11.3.1 通信与命令	188
11.3.2 SD 卡的初始化	191
11.3.3 SD 卡的单扇区读/写	194
11.3.4 SD 卡的多扇区读/写擦除	195
第 12 章 性能提升,底层限制:高性能 SD 卡物理驱动	197
12.1 现有 SD 卡驱动的性能评估	197
12.2 用 DMA 为数据传输提速	202
12.3 高性能 SD 卡驱动的实现	206

附录 A 完整工程实例之 SD 卡 MP3 播放器	211
附录 B 完整工程实例之数码录像机(相机)	218
附录 C 主流 CPU 内核及其典型芯片简介(znFAT 移植平台)	227
附录 D 答网友问	233
附录 E 我的大学系列	238
参考文献	246

第 1 章

数据记录,偷梁换柱:使用变通方法实现文件数据存储

上册一直都是围绕读取操作来进行的,也就是说只用到了扇区读取函数(znFAT_Device_Read_Sector),并没有涉及与扇区写入相关的内容。其实 znFAT 早期的开发工作到这里就基本完成了,并没有实现写入操作相关的功能,仅仅是为了完成上册开篇所说的 MP3 数码相框实验而已,但是后来有很多人看到振南在研究 FAT32,于是经常问有没有实现文件创建和数据写入的功能,想通过 znFAT 来满足他们数据记录的需求,于是决定把 FAT32 继续做下去,最终形成一个完备的方案,实现文件操作的各种常用功能。

虽然当时 znFAT 中还没有成型的数据写入功能,但振南还是使用一些变通,甚至是“偷梁换柱”的方法帮助一些人巧妙地实现了数据记录的功能。振南觉得有必要将这些方法介绍给读者,如果可以满足需求,那也许就可以无须或者尽量少地去牵扯 FAT32 了,因为 FAT32 毕竟还是比较复杂的。当然,这些“变通方法”必然是有一定局限性的,只能针对于个别的、要求不高的应用场合。要想实现真正意义上的、通用的文件创建、数据写入等功能,那么还是要继续对 FAT32 进行研究,一步一步地去实现我们想要的功能。

1.1 把 SD 卡用作一个大容量的 ROM

1.1.1 大 ROM 思想的提出

2010 年,振南完成的一个项目的主要功能就是实现一个塔吊黑匣子,具体的描述请看图 1.1。放置于驾驶室中的一个装置,一方面接收从塔吊总控制器发送过来的吊钩位置与报警信息(通过 CAN 总线进行通信),另一方面把数据存入到 SD 卡中,并向总控制器发送回执。

起初,这个公司因为没有数据存储和文件系统方面的开发经验才找到了振南。项目经理说:“只要能将数据存入 SD 卡,并能方便地导入到计算机中就行了。”于是,振南向他介绍了“大 ROM”思想。其实很简单,就是把 SD 卡直接当成一个大容量的

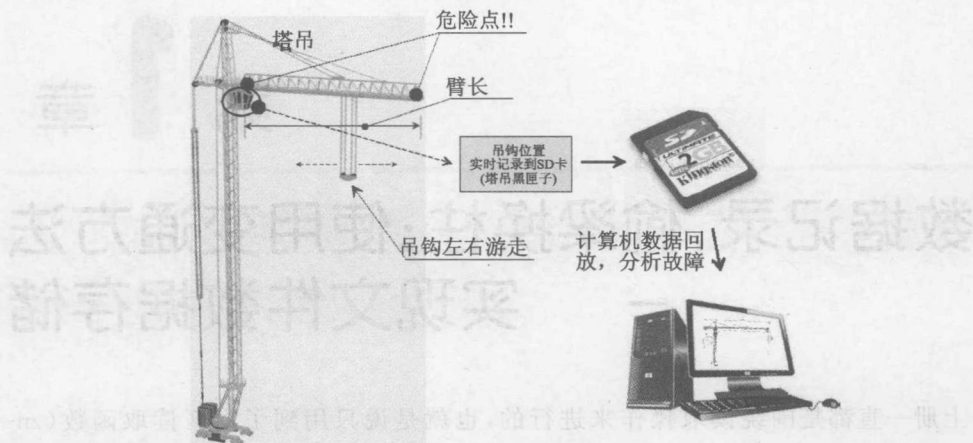


图 1.1 塔吊黑匣子项目功能示意图

ROM 存储器来使用,如图 1.2 所示。

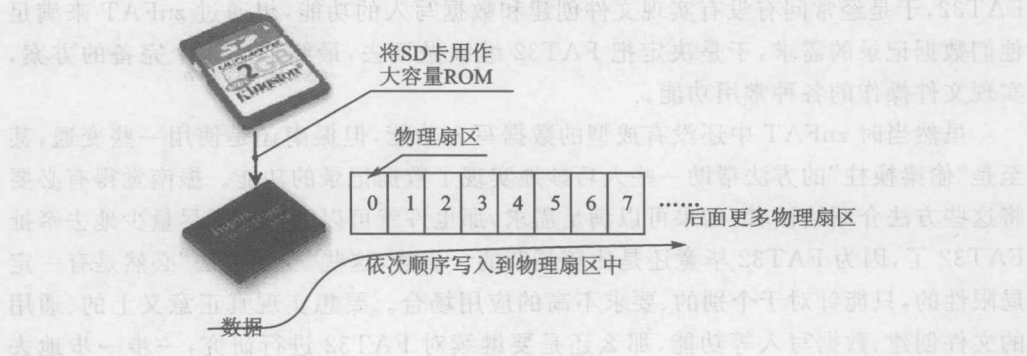


图 1.2 将数据直接顺序写入到物理扇区

此时,项目经理立即反驳:“这样不行,SD 卡放到计算机上识别不出来的。”确实是!根本原因就是因为我们写入数据的方式并没有遵循 FAT32 的标准。数据非但不能识别,而且还会把现有的文件系统毁掉,计算机提示我们“格式化磁盘”(原本用于存储 MBR 或 DBR 的物理扇区被数据覆盖,从而导致整个文件系统的崩溃)。“解铃还须系铃人”,若用直接写物理扇区的方式来进行数据存储,那在计算机上也要以物理扇区方式来获取数据。还记得 WinHex 软件的物理模式吗?

1.1.2 思想的验证:数据采集与记录实验

项目经理迷惑地问道:“似乎可行,你能不能先在开发板上实现一下,做一个演示?”于是就有了数据采集与记录实验,如图 1.3 所示。这个实验基于 ZN-X 开发板(配合基础实验资源模块),如图 1.4 所示。

从 TLC549、DS18B20、PCF8563 分别读取模拟信号、温度数据和时间信息,将它

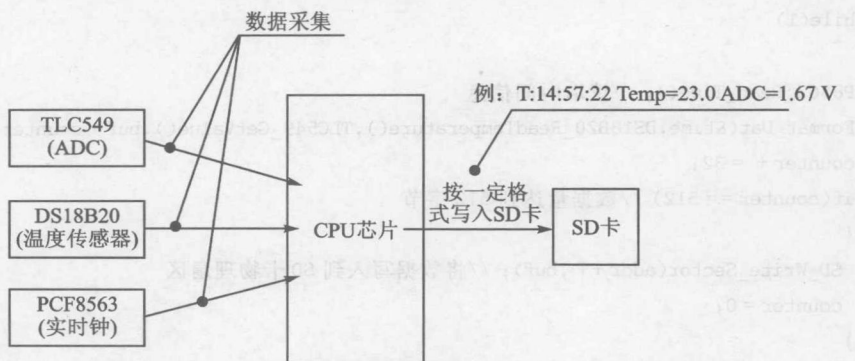


图 1.3 数据采集与记录实验示意图

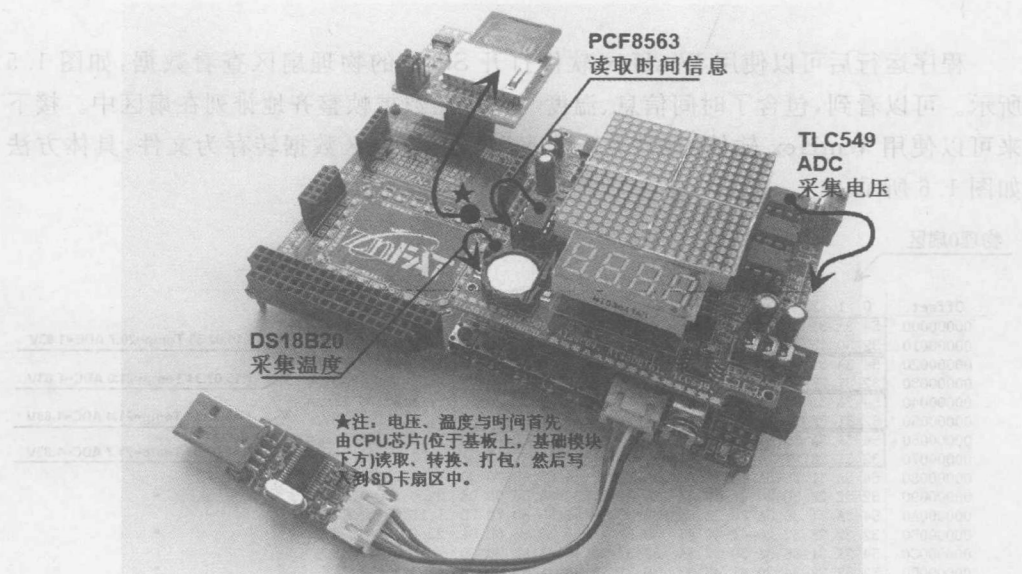


图 1.4 ZN-X 开发板配合基础实验资源模块效果图

们打包为固定 32 字节的数据格式,形如“T:14:57:22 Temp=23.0 ADC=1.67V\r\n”,我们称之为一个数据帧。每采集到 16 个数据帧,即数据量达到 512 字节,就将其写入到物理扇区。具体实现代码(_main.c)如下:

```
unsigned char buf[512];
struct Time time; //用于装载时间数据的结构体变量 time
void main(void)
{
    unsigned int counter = 0;
    unsigned long addr = 0;
    SD_Init(); //SD 卡初始化
```




```
while(1)
{
    P8563_Read_Time(); //读取时间信息
    Format_Dat(&time, DS18B20_ReadTemperature(), TLC549_GetValue(), buf + counter);
    counter + = 32;
    if(counter == 512) //数据量达到 512 字节
    {
        SD_Write_Sector(addr ++ ,buf); //将数据写入到 SD 卡物理扇区
        counter = 0;
    }
}
while(1);
}
```

程序运行后可以使用 WinHex 软件打开 SD 卡的物理扇区查看数据,如图 1.5 所示。可以看到,包含了时间信息、温度与电压的数据帧整齐地排列在扇区中。接下来可以使用 WinHex 软件中的“克隆磁盘”功能,将扇区数据转存为文件,具体方法如图 1.6 所示。

物理 0 扇区

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	54	3A	31	35	3A	30	32	3A	33	33	20	54	65	6D	70	3D	T:15:02:33 Temp=
00000010	32	30	2E	37	20	41	44	43	3D	31	2E	38	33	56	0D	0A	20.7 ADC=1.83V..
00000020	54	3A	31	35	3A	30	32	3A	33	34	20	54	65	6D	70	3D	T:15:02:34 Temp=
00000030	32	31	2E	30	20	41	44	43	3D	31	2E	38	33	56	0D	0A	21.0 ADC=1.83V..
00000040	54	3A	31	35	3A	30	32	3A	33	34	20	54	65	6D	70	3D	T:15:02:34 Temp=
00000050	32	31	2E	34	20	41	44	43	3D	31	2E	38	33	56	0D	0A	21.4 ADC=1.83V..
00000060	54	3A	31	35	3A	30	32	3A	33	35	20	54	65	6D	70	3D	T:15:02:35 Temp=
00000070	32	31	2E	37	20	41	44	43	3D	31	2E	38	33	56	0D	0A	21.7 ADC=1.83V..
00000080	54	3A	31	35	3A	30	32	3A	33	36	20	54	65	6D	70	3D	T:15:02:36 Temp=
00000090	32	32	2E	30	20	41	44	43	3D	31	2E	38	35	56	0D	0A	22.0 ADC=1.85V..
000000A0	54	3A	31	35	3A	30	32	3A	33	36	20	54	65	6D	70	3D	T:15:02:36 Temp=
000000B0	32	32	2E	32	20	41	44	43	3D	31	2E	38	39	56	0D	0A	22.2 ADC=1.89V..
000000C0	54	3A	31	35	3A	30	32	3A	33	37	20	54	65	6D	70	3D	T:15:02:37 Temp=
000000D0	32	32	2E	34	20	41	44	43	3D	31	2E	38	31	56	0D	0A	22.4 ADC=1.81V..
000000E0	54	3A	31	35	3A	30	32	3A	33	37	20	54	65	6D	70	3D	T:15:02:37 Temp=
000000F0	32	32	2E	36	20	41	44	43	3D	31	2E	37	31	56	0D	0A	22.6 ADC=1.71V..
00000100	54	3A	31	35	3A	30	32	3A	33	38	20	54	65	6D	70	3D	T:15:02:38 Temp=
00000110	32	32	2E	36	20	41	44	43	3D	31	2E	36	36	56	0D	0A	22.6 ADC=1.66V..
00000120	54	3A	31	35	3A	30	32	3A	33	39	20	54	65	6D	70	3D	T:15:02:39 Temp=
00000130	32	32	2E	38	20	41	44	43	3D	31	2E	36	34	56	0D	0A	22.8 ADC=1.64V..
00000140	54	3A	31	35	3A	30	32	3A	33	39	20	54	65	6D	70	3D	T:15:02:39 Temp=
00000150	32	32	2E	39	20	41	44	43	3D	31	2E	36	34	56	0D	0A	22.9 ADC=1.64V..

32字节数据帧

图 1.5 物理 0 扇区中的数据帧

其实,克隆磁盘功能可以将从某一扇区开始的任意多个扇区中的数据转存为文件,甚至是整个磁盘。转存得到的文件有一个专门的名字来称呼它——“镜像”。

上面的实验其实是存在一个小问题的:“数据向扇区顺序写入,那怎么知道最后它停在哪个扇区上呢?应该用 WinHex 复制多少个扇区呢?”其实很简单,我们从 1 扇区开始写数据,停止的时候将当前扇区地址写入到 0 扇区中,也就是说,留出 0 扇区专门用于记录结束扇区。