



《振南 znFAT--嵌入式 FAT32 文件系统设计与实现》一书 【上下册】已正式出版发行

全国各渠道全面发售

（在当当、京东、亚马逊、淘宝等网络平台上搜索
关键字"**znFAT**"即可购买，各地实体书店也有售）

此书是市面上 唯一 一套详细全面而深入讲解嵌入式存储技术、**FAT32** 文件系统、**SD** 卡驱动与应用方面的专著。全套书一共 **25** 章，近 **70** 万字。从基础、提高、实践、剖析、创新、应用等很多方面进行阐述，力求通俗，振南用十年磨一剑的精神编著此书，希望对广大工程师与爱好者产生参考与积极意义。

此书在各大电子技术论坛均有**长期的「抢楼送书活动」**，如 211C、elecfans 等等。

振南的**【ZN-X 开发板】**是市面上唯一全模块化、多元化的开发板，可支持 **51、AVR、STM32 (M0/M3/M4)**

详情请关注 www.znmcu.cn （振南个人主页!!）

第 7 章

文件在手,剖析寻数: 文件目录项分析与信息提取

经过前两章看似乏味的参数解析,我们已经获取了 FAT32 最为核心的参数,完成了 znFAT 的初始化函数(znFAT_Init)。此时,FAT32 的整体轮廓已清晰可见,但是文件到底在哪里?答:文件在目录中!我们现在已经找到了首目录的所在,所以文件其实已经在我们手中,唾手可得。在本章中文件会在我们面前“显出原形”,我们要做的就是去分析它,获取文件信息并读取它的数据,最终完善我们的 SD 卡 MP3 数码相框,到时候就可以称之为“带有 znFAT 雏形的 SD 卡 MP3 数码相框”了。振南提醒,不要因为中间隔了 MBR 和 DBR 这两章,就忘记了我们在第 3 章中提出的问題:复制到 SD 卡中的 MP3 和 BIN 文件的数据到底在哪?

7.1 从首目录下手

1. 我们所熟知的文件形式

我们平时所看到的文件是以怎样的形式存在的呢? Windows 中的一个图标、DOS 下的一个条目、手机里一个小说、MP3 里的一首歌曲……,如图 7.1 所示。图 7.1 都是我们对文件最直观的认识,但是透过这些图标、条目等形式背后所隐藏的文件本原是什么样的呢?一个文件之所以可以显示在屏幕上、展现我们面前,它必然对应于存储设备某个存储单元上的某条记录,以一定的格式记录了文件的名称、大小等信息,因为文件不可能凭空存在。

2. 文件的实质

前面我们说文件就是存储单元中的一条记录,确实是这样。回想一下第 3 章所讲到的逼近模型中的文件索引机制(当时我们使用它是为了能快速找到文件),它指示了文件数据的开始位置,其实它在 FAT32 中被称为文件目录项(振南称之为 FDI,FAT32 中其实没有这个名词)所以,可以说,在 FAT32 中文件实质上就是一个文件目录项。我们看到的文件是在目录中,所以文件目录项就存储在目录的簇中。上一章我们已经知道了首目录的所在(开始扇区地址为 15191),下面就打开首目录簇来看一下,如图 7.2 所示。

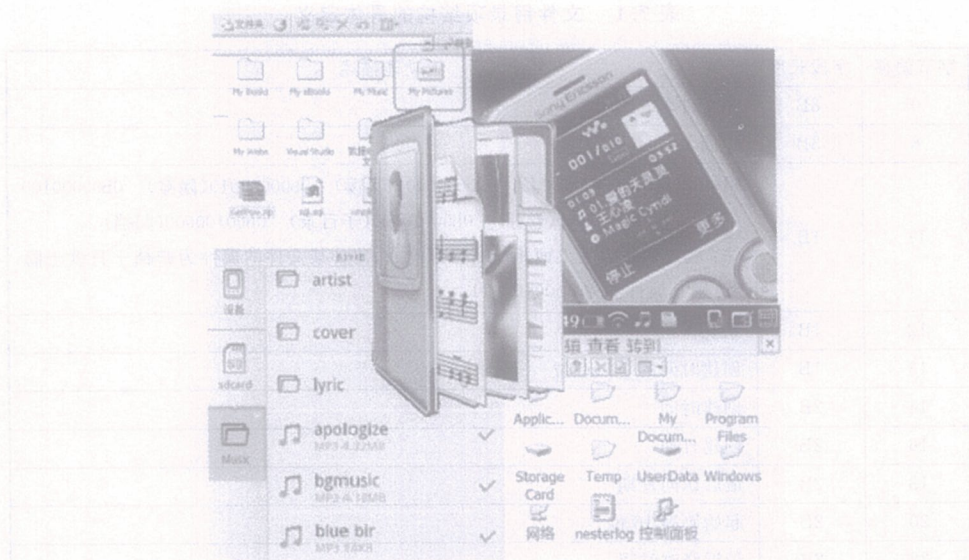


图 7.1 平时我们所见到的文件形式

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
00763000	5A	4E	46	41	54	20	20	20	20	20	20	08	00	AB	5C	B4	ZNFAT	«\
00763010	7C	42	7C	42	00	00	5C	B4	7C	42	00	00	00	00	00	00	B B \` B	
00763020	50	49	43	54	55	52	45	20	42	49	4E	20	18	B0	BD	4E	PICTURE BIN	°½N
00763030	7D	42	7D	42	00	00	00	BF	73	42	03	00	00	58	02	00	}B}B ŁsB X	
00763040	54	45	53	54	20	20	20	20	4D	50	33	20	00	39	C4	4E	TEST MP3	9½N
00763050	7D	42	7D	42	00	00	7B	A0	74	42	29	00	0D	FE	18	00	}B}B { tB} p	
00763060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00763070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00763080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
00763090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		
007630A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		

图 7.2 首目录簇开始扇区中的数据

图 6.2 的标注处似乎与我们复制的文件(picture. bin 与 TEST. MP3)有些关联。它是一个小的数据块,显得很整齐,前面好像是文件名,后面跟着一些数据,这个数据块就是用来记录文件信息的文件目录项 FDI 的。

7.2 文件目录项

7.2.1 文件目录项的定义

文件目录项的存储结构定义如表 7.1 所列。

其实我们平时在 Windows 中右击,然后在弹出的级联菜单中选择“属性”,这时调出的文件属性对话框中所看到的信息,就是按照以上定义对文件目录项解析的结果,如图 7.3 所示。

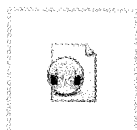


表 7.1 文件目录项结构的具体定义

字节偏移	字段长度	功能描述
0	8B	主文件名
8	3B	扩展名
11	1B	属性: 0b00000000(读写) 0b00000001(只读) 0b00000010(隐藏) 0b00000100(系统) 0b00001000(卷标) 0b00010000(子目录) 0b00100000(归档) 注: 实际属性值为上面相应属性码的或, 比如某文件的属性为归档+只读+隐藏, 则其属性值为 0b00100011
12	1B	系统保留
13	1B	创建时间的 10 ms 位
14	2B	创建时间
16	2B	创建日期
18	2B	最后访问日期
20	2B	起始簇高 16 位
22	2B	最近修改时间
24	2B	最近修改日期
26	2B	起始簇低 16 位
28	4B	文件数据长度



picture.bin



TEST.MP3

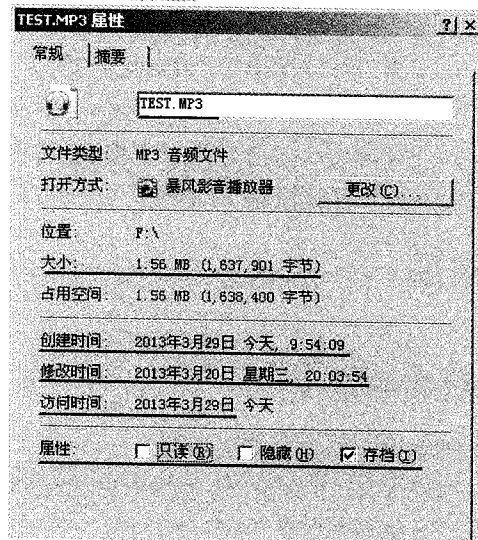


图 7.3 TEST.MP3 文件的属性对话框

细心的读者是不是有过这样的疑问:“为什么文件属性里的访问时间这一项只有日期没有时间呢?”其根源就在于文件目录项的定义,文件项目项中只有“最后访问日期”,却没有“最后访问时间”。

7.2.2 手工解析 FDI

现在就拿 TEST.MP3 的文件目录项为例来进行解析,如图 7.4 所示。依照上面对文件目录项的定义,则可以得到以下参数:

文件名: "TEST MP3"
 文件属性: 0X20
 创建时间 10 ms 位: 0X78
 创建时间: 0X4EC4
 创建日期: 0X427D
 访问日期: 0X427D
 开始簇高字: 0X0000
 修改时间: 0XA07B
 修改日期: 0X4274
 开始簇低字: 0029
 文件大小: 0X0018FE0D = 1637901

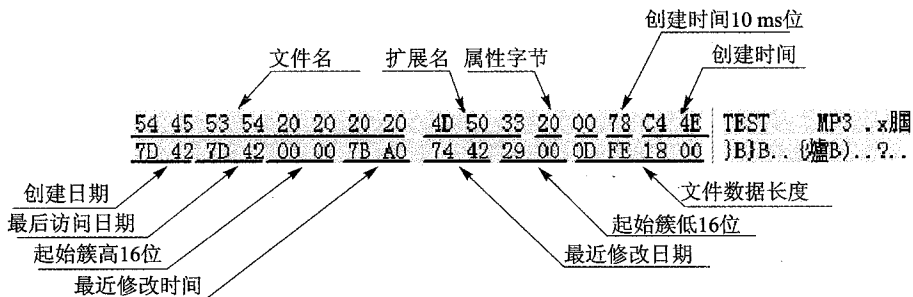


图 7.4 TEST.MP3 的文件目录项

将开始簇的高、低字合并即得到开始簇,为 0X00000029=41,意思就是说这个文件的数据是从第 41 簇开始的。

可以看到,文件目录项中用于记录文件名的字段加起来只有 11 个字节,其中包括 8 个字节的主文件名和 3 个字节的扩展名。所以,我们也就知道了为什么在传统的 DOS 操作系统上文件名要符合 8·3 格式,比如 ABCDEFGH.TXT、TEST.MP3 或 A.C。就算超过了 8·3,系统也会自动将文件名截断,比如 ABCDEFGHI.TXT 会变成 ABCDEF~1.TXT。“可是…这…这文件名……”我知道你要问什么,一会儿再问! 字段中不足 8·3 的字符用空格(ASCII 码为 0X20)来填充,所以 TEST.MP3 这个文件名在文件目录项中为“TEST MP3”。但是我们又发现,picture.bin 文



件目录项中的文件名为“PICTURE BIN”，似乎文件名字段只能记录大写字符，那它如何来表达小写文件名呢？后面会详细进行解答。

上面你一定是要问：“这里所讲的文件名与我平时看到的怎么不太一样啊？”确实是！平时我们所见到的文件名长度比 8·3 更长，比如 abcdefghijk. rmvb 等。那这些文件名岂不是不能在文件目录项中进行存储了吗？非也！这些文件名都属于 FAT32 中的长文件名，它们是 FAT32 为了使文件名更为灵活、能够表达更多信息而专门做的一些改进和增强，可以说，是有别于 8·3 文件名机制的另一套机制。关于长文件名，本套丛书的下册中会有专门章节进行详细介绍，这里只是简单提一下。读者只需要记住文件目录项中只能存储标准 8·3 格式的文件名，而那些 8·3 格式无法表达的合法文件名基本均属于长文件名。

其实 FAT32 中文件名的“名堂”是比较多的，除了上面说的 8·3 格式、长文件名以外，还有文件名的通配、文件名的变换、文件名的生成等，后面会一一讲解。

至于文件的属性，根据上面的定义，这个字节不同位代表不同的属性，0X20 说明文件属性为存档。而且属性是可以迭加的，比如一个文件可以是存档，同时又是隐藏的，那么它的属性字节就是 0X22。

剩下的就是文件的时间和日期，在解析得到的数值中似乎不能直接看出年月日时分秒这些信息。时间与日期参数有自己独特的组织方式，其实非常简单，只是比较繁琐，它的参数划分已经精细到了位的级别。

7.2.3 解析时间信息

FAT32 中文件的时间信息包括文件的创建时间日期、修改时间日期及访问时间。时间与日期参数都占用了两字节来进行存储，这两个字节(16 位)被划分为不同长度的位段从而表达时分秒与年月日信息。这样说可能不太好明白，我们还是实际对 TEST. MP3 的文件目录项中的时间字段解析一下看看吧。

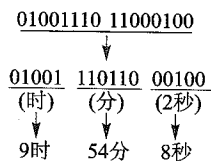


图 7.5 时间字段 16 个位的划分

我们已经知道创建时间所对应的字段值为 0X4EC4，即二进制的 01001110 11000100。这 16 位我们可以进行划分，如图 7.5 所示。这 16 位被划分为长度分别为 5、6、5 的位段，各自的单位为时、分、2 秒。因此，如图 7.5 所示，TEST. MP3 的创建时间就是 9 时 54 分 8 秒。

然后就是对日期的解析，和上面大同小异，仍然是 16 位，只是位段的划分方法不同。它的值为 0X427D，即 01000010 01111101，具体的划分方法和解析过程如图 7.6 所示。

从图 7.6 中可以看到月和日都很直观，唯独年是 33，这是什么意思？从图 7.3 中可以知道这个 TEST. MP3 文件的创建日期明明是 2013 年 3 月 29 日，如果它是只取年份的后两位，那也应该是 13 才对，怎么会是 33 呢？到底是怎么算的？其实这也

很简单,FAT32 中文件的日期中的年份都是要加 1 980 的(称为年份基数),1 980+33 正好是 2 013。从另一个方面,我们也知道了 FAT32 是有一个最大的期限的。用于记录年份的位段长度为 7,即最大能表达的数值为 127,那么 FAT32 到了 1 980+127=2 107 年以后,就不能有效地记录文件的日期信息了。这其实也只是一个很理想化的假设,

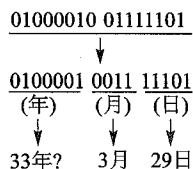


图 7.6 日期字段 16 个位的划分

只能说 FAT32 能够用到的极限日期是 2107 年。但这期间,不乏会有更加优秀、更加革命性的文件系统的出现,从而使得 FAT32 文件系统成为历史(其实现在在很多其他的文件系统,如 EXT2/3、NTFS 等,都已经优于 FAT32 了,只是在短时间内 FAT32 的地位和普及度仍不会动摇,尤其是在嵌入式方面的应用)。这个时间可能只是几年、十几年或几十年,但绝不会是一百多年(其实就算它能一直延用下去,超过了 2107 年,我们也只需要重新定义一下年份基数即可)。在信息技术如此发达的现在,一项技术或一个方案,被延用了几十年,乃至上百年,简直是不可想象的。

上面就是对创建时间与日期的解析了,其实都非常简单。后面编程实现的时候将涉及大量的位拼接、位与或等操作,可能会比较麻烦。细心的读者在我们讲解时间解析的时候应该看出了一个小漏洞:图 7.3 中显示的文件创建时间的秒位为 9,但是解析的结果却是 8,好像有一些小出入。确实,文件目录项中秒位的单位被定义为“2 秒”,似乎让人觉得很奇怪。如果是这样的话,那文件的时间信息岂不是只能表达偶数了吗?这其中一定是我们忽略了什么。

我们看上面对文件目录项的定义中有一个字段(第 14 个字节)叫“创建时间的 10 ms 位”。顾名思义,它是用来表达文件创建时间的 10 毫秒精度的参数。TEST.MP3 的文件目录项中此参数值为 0X78=120,那么毫秒位所表达的时间就是 120×10 毫秒=1200 毫秒=1.2 秒,把它与文件的创建时间加在一起 9 时 54 分 8 秒+1.2 秒=9 时 54 分 9 秒,这就对了!

至于修改时间、日期以及访问日期解析的方法也是完全相同的,就不赘述了。

7.2.4 篡改文件“时间戳”

其实我们一直所说的“文件的时间与日期”,在 FAT32 中有一个专门的概念叫作“时间戳”。平时在计算机上创建一些文件的时候,操作系统自动提取当前的系统时间作为文件的创建时间。人们发现,这个创建时间就如同在邮票上盖邮戳一样,“落地生根”,以后再也无法修改。这个问题有时候让一些人头疼,他们希望文件的创建时间也可以由自己来随意修改。因此,在网上一些文件信息修改软件应运而生,比如 New FileTime、File Property Editor、File Changer 等。现在通过对文件目录项的研究,没有这些软件我们同样可以随意修改文件的信息。

比如,现在把 TEST.MP3 的创建时间由 9 时 54 分 9 秒改为 1 时 32 分 1 秒,请

看图 7.7。图中所表达的其实就是时间解析的逆过程——“时间的合成”，即由时间信息得到对应的时间字段值。将创建时间字段更改为此值，于是便实现了时间戳的篡改。此时，再来看一下文件的属性，如图 7.8 所示。

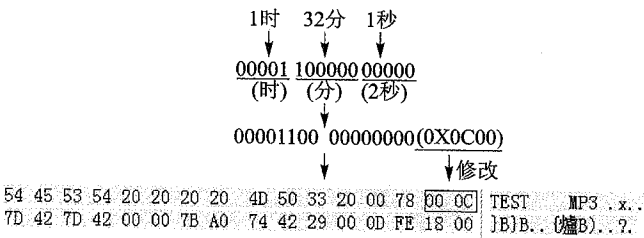


图 7.7 对文件创建时间字段进行修改

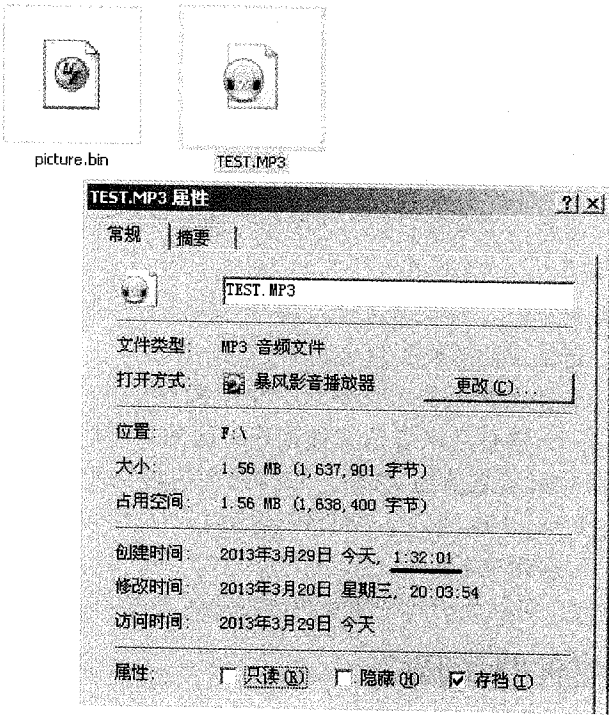


图 7.8 修改完创建时间后的文件属性

可以看到，“创建时间”确实被改为了我们想要的时间。

7.3 FDI 解析的代码实现

按照我们的套路，现在该是用代码来对文件目录项解析进行实现的时候了。

此文因版权仅节选一部分，请各位读者见谅！！

完全内容请购买正版书籍!!

感谢对振南及 znFAT 的关注与支持，希望振南在嵌入式 FAT32 文件系统方面的研究对您有所帮助！

更多内容请关注 振南电子网站

www.znmcu.cn

Lesson

张南亲临
现场培训（北京区）



是否想与张南本人
现场交流？
是否想听张南的亲
自授课？
按南现场培训正在
招收学员，详情点
击进入！
(暂仅限北京区)



ZN-X

中国首个全固态金属离子电池
全球首款可商用微流控芯片
全球首款PUFET、nan-Si/SiO₂/PMOS器件

国内ZN-X开发版介绍
精彩实验，资料速览发布
购买渠道与技术支持
校企合作团队与内部操作演示
发明发表区与意见反馈

产品进入

Audio video

海康长期投入嵌入式音视频编解码器研发，取得了一系列成果，在此与您分享！













JPEG/GIF/PNG/BMP等
常见图片格式编解码器

AVI/MPEG/MP3等
视频编解码器

资料、案例、演示发布

[点击进入](#)

Support

产品创意以用户为中心
工艺与供应链的洞察
更重要的是前瞻的技术支持

拥有经验丰富的技术团队
及时的迭代出及对市场响应的敏捷
我们丰富的研发经验与感得度
力将是您的坚实后盾

[点击进入](#)



最新网络技术
软件资源免费下载
最新电子杂志及图文内容资源平台

最新的技术交流平台
最新原创头像
资源发布与分享
这里的气氛更加活跃，
欢迎加入

云友京城 EFFoundations










GUI

图形用户界面开发库的图形能力
 支持了 6 名加入为 GUI 库成员
 2016 年 6 月 GUI 库在下一季度开发完成



感谢的 uCUI/miWIN 团队
 - 提供图形库支持
 - 在库成员加入 GUI 库支持
 支持 GUI 库、ZIG/SLI 等
 基于 GUI 库、X 库开发的 GUI
 库成员感谢

ucui/miWIN
 X GUI/SLI 成员

[点击查看](#)

Project

- 项目承接技术与前期规划
- 项目合作协议与合同
- 数据库与项目管理软件联系方式

产品进入