

嵌入式系统固件文件格式分析研究

陈培新, 赵 炯

(同济大学 机械工程学院, 上海 201800)

摘 要: 嵌入式系统固件文件格式是远程固件更新和升级的基础, 固件映像文件格式不尽相同, 这些格式对远程固件进行更新和升级的灵活性和简便性有重要影响。为了使固件更新程序能够正确完成 Flash 写入操作, 固件文件就需要克服现有格式中未包含关于映像文件详细信息的缺点。通过分析几种可更新固件文件的代码、数据和其他辅助信息, 得出现有的几种固件文件的信息格式, 并在现有格式基础上添加映像文件的详细信息。文中研究出的固件文件格式更加具有灵活性和高效性。

关键词: 固件文件格式; 远程固件更新; 映像文件

中图分类号: TP311

文献标识码: A

文章编号: 1673-629X(2009)02-0045-03

Analysis and Research to Format of Embedded System's Firmware Files

CHEN Pei-xin, ZHAO Jiong

(College of Mechanical Engineering of Tongji University, Shanghai 201800, China)

Abstract: The format of the embedded system's firmware files is the basis of remote firmware updating and upgrading, the image formats of firmware may be different, these formats have an important impact on the flexibility and simplicity of remote firmware updating and upgrading. In order to make firmware updating procedure to finish Flash-writing operation correctly, firmware files need to overcome the shortcoming of the existing format of the image files that did not contain detail information. Based on the analysis to the code, data and other supporting information of several updated firmware files, the format of information of several existing firmware files can be found in this article, and the detail information of image files will be added to the existing formats. The format of firmware files researched on this paper will be flexibility and efficiency.

Key words: format of firmware files; remote firmware updating; image file

0 引言

目前嵌入式系统已广泛应用于各个领域, 其中固件的维护日益重要。选用 FTP 传输或 HTTP 上载等手段将包含映像文件和参数的固件下载到远端目标机中就可实现固件的更新和升级。通常各种嵌入式系统中所采用的固件映像文件格式不尽相同, 这些格式对远程固件进行更新和升级的灵活性和简便性有重要影响, 它们是远程固件更新和升级的基础。因此设计一种更有效的固件文件格式, 这种格式具有更高的灵活性和通用性。

首先通过分析几种固件文件(如 trx、usr、bin)的格式, 然后通过分析固件文件所要包含的功能设计出一

种固件文件格式开发方案。这种格式能对传送到目标机中的文件进行校验, 并为 Flash 写入程序提供详细的映像文件信息来完成写 FlashROM 操作。它为远程固件更新和升级提供了有效的技术支持。

1 固件文件格式

固件更新首先应将固件传送到目标机中, 通过校验的方法检验文件数据的完整性, 然后从固件文件中提取有效映像文件, 根据固件文件中的参数通过 Flash 写入程序^[1]将相应映像文件写入到嵌入式设备的 FlashROM 中。因此固件文件中应包含与更新特性有关的映像文件和相应参数, 并能够检验传送到目标机中的数据是否正确。

固件文件^[2]主要有头部和数据部分组成。固件文件头部可分成两个部分: 其一是关于整个固件的相关信息, 可以称之为公共信息; 其二是关于每个包含进固件文件中的映像文件各自的信息。固件文件数据部分应为与更新特性有关的映像文件如 kernel、rootfs 等。

收稿日期: 2008-06-10

基金项目: 国家十一五科技支撑项目(2006BAJ12B01)

作者简介: 陈培新(1985-), 男, 硕士研究生, 研究方向为嵌入式系统开发; 赵 炯, 副教授, 博士, 研究方向为嵌入式系统开发、通信协议分析。

因此一般固件文件格式如图 1 所示。

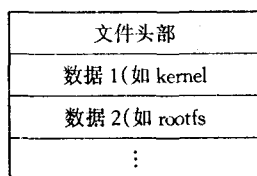


图 1 一般固件文件格式

2 固件格式分析

固件文件格式分析主要是研究文件头部信息,分析头部信息的数据就能够知道整个固件文件的结构。分析固件时应注意文件数据的存储方式。文件存储会因为 CPU 的不同,数据存储的方式分为 Big-endian 和 Little-endian, Big-endian 是大数在前、小数在后, Little-endian 是小数在前、大数在后。所以文件头部会因为 CPU 的不用,数据存储的顺序不一样,分析头部应注意数据存储格式。下面对常见的 trx、usr、bin 固件文件格式进行分析。

2.1 trx 格式固件分析

trx 是 CISCO 公司无线路由器 wrt54g 系列固件文件所采用的格式。文件格式开头是 28 字节的头部,下面是数据部分。文件头部数据结构:

```
struct trx_header {
    uint32_t magic;
    uint32_t len;
    uint32_t crc32;
    uint32_t flag_version;
    uint32_t offsets[3];
};
```

magic 字段用于存放魔数,FLASH 写入程序在使用固件文件时,通过判断这个魔数是否正确就可以知道固件是否是正确的固件文件类型。结构中的 len 字段用于存放包含本头部结构信息的固件文件的长度。

crc_32 字段于存放 CRC32 校验码。生成校验码所使用的数据是从 flag_version 字段开始到文件末尾所有数据,可以通过此校验码来验证传送到目标机中的固件文件数据是否正确。flag_version 字段存放固件标志和版本(0~15 代表标志,16~31 代表版本)。

offsets^[3] 数组存放各个部分在固件文件中的 32 位偏移值。offsets[i] 代表从文件开始到第 i 数据部分的 32 位偏移值,以一个固件文件 openwrt-brcm-2.4-squashfs.trx^[4] 格式头部为例,头部数据如下:

```
48 44 52 30 00 A0 17 00
32 11 18 1F 00 00 10 00
1C 00 00 00 D8 08 00 00 00 E4 07 00
```

文件头部数据分析如下:magic = 0x48445230 是

“HDR0”代表魔数。len = 0x17A000 是代表 32 位文件长度 1548288 个字节。crc32 = 0x1F181132 是 32 位 CRC 校验码。flag_version = 0x00100000 表示标志和版本。offsets[0] = 0x001C 代表第一数据部分从 28 字节开始,offsets[1] = 0x08D8 代表第二数据部分从 2264 字节开始,offsets[3] = 0x07E400 代表第三数据部分从 517120 字节开始。

2.2 usr 格式固件分析

usr 是 U. S. Robotics 固件文件所采用的格式。文件开头是 28 字节的头部,下面就是数据部分。文件头部数据结构:

```
struct usr_header {
    uint32 magic;
    uint32 len;
    uint32 crc32;
    uint32 version;
    uint16 compatibility_id;
    uint16 hardware_revision;
    uint32 reserved[5];
};
```

magic 字段用于存放魔数。len 字段用于存放不包括文件头部结构信息的固件的文件的长度。crc_32 字段用于存放不包含头部的文件 32 位 CRC32 校验码。version 字段用于存放 32 位 EPI 版本号。compatibility_id 字段用于存放 16 位的可兼容 ID 号。hardware_revision 字段用于存放 6 位硬件修改号。reserved^[5] 数组代表预留的 8 个字节文件头部。

以一个固件文件 USR5461-v3.93.35.0.8.Usr^[5] 格式头部为例,头部数据如下:

```
55 53 52 30 00 60 1C 00
12 2C 3B C1 03 5D 23 08
01 00 01 00 00 00 00 00
00 00 00 00
```

文件头部数据分析如下:magic = 0x55535230 是“USR0”代表魔数。len = 0x1C6000 是代表 32 位不包括文件头部的文件的长度 1859584 字节,文件大小是 1859612 字节。crc32 = 0Xc13b2c12 是代表 CRC 校验码。version = 0x06235d03。compatibility_id = 0x01。hardware_revision = 0x01。reserved[i] = 0x00。

2.3 bin 格式固件分析

bin 是 CISCO 子公司 Linksys 公司无线路由器固件文件所采用的格式。文件有 60 字节头部,前 32 字节是 Linksys 公司无线路由器固件文件 bin 格式特有的头部,后 28 字节是 trx 格式文件头部,前 32 字节头部格式如下:

```

struct bin_header {
    uint32 product_abbreviation;
    uint32 reserved1[1];
    uint8 release_date[3];
    uint8 firmware_version[3];
    uint32 magic;
    uint32 reserved2[3];
};

```

product_abbreviation 字段存放固件的产品型号缩写。reserved1[1]代表预留的4个字节。release_date[3] = “年_月_日”代表固件发布日期。firmware_version[3]代表固件版本号。magic 字段用于存放魔数。reserved2[3]是代表预留的12个字节,包含一些辅助信息。

以一个固件文件 dd-wrt.v23-wrt54gs.b-in^[3]为例,前32字节头部数据如下:

```

57 35 34 53 00 00 00 00
05 0C 19 04 46 06 55 32
4E 44 01 00 1F 00 00 00
00 00 00 00 00 00 00 00

```

文件头部数据分析如下:

product_abbreviation = 0x57353453 代表“W54S”; reserved1[1] = 0x00; release_date[3] = 代表固件发布日期 05 年 12 月 19 日; firmware_version[3] 代表固件版本号 04.46.04; magic = 0x55324E44 是“U2rd”代表魔数; reserved2[3] = 0x00 是代表预留的12个字节,包含一些辅助信息。

3 固件文件格式设计

通过上文分析几种常见的固件文件格式可知固件文件头部中会存放一个魔数(Magic Number)字段来确定文件的类型,在头部中存放如文件长度、CRC 校验码等来验证数据的正确性。另外还包括一些辅助信息。

为了使固件文件格式更加具有灵活性和通用性,固件文件头部中应该添加更加详细的信息使固件更新程序能够从固件中提取足够的信息来完成写操作。另外,头部中还应该包含各个部分在固件文件中的偏移值。这种固件文件格式就能正确定位各个映像文件在固件文件的位置,并能将固件文件中的各映像文件准确地写入到嵌入式设备的 FlashROM 中。因此设计固件头部占用256个字节,一共可以保存5个映像文件的相关信息。即目前设计程序中指定一个固件文件中最多可以存放5个映像文件。下面给出固件文件头部中公共信息的具体数据结构信息:

```

struct pak_header {
    uint32_t magic;
    uint32_t len;
    uint32_t crc32;
    uint32_t icount;
    struct img_header
    img[MAX_PAK_IMAGES];
};

```

该结构最后一个字段定义了一个具有5个结构项的数组。其中每个结构项中保存着一个映像文件的具体信息。数据项数据结构定义如下:

```

struct img_header {
    uint32_t offset;
    uint32_t len;
    uint32_t mtdofs;
    uint32_t mtdno;
    char mtdname[MAX_NAME_LEN+1];
    char filename[MAX_NAME_LEN+1];
};

```

定义魔数 magic 字段为“XSCA”,即对应十六进制 0x041435358(Little endian 表示法)。结构中的 len 字段用于存放包含本头部结构信息的固件文件的长度。

crc32 字段用于存放 CRC32 校验码。生成校验码所使用的数据是从 icont 字段开始到文件末尾所有数据。固件文件中计算 CRC32 校验码的程序使用了公共域 C 代码程序。

icont 字段固件文件中所包含的映像文件数目。MAX_PAK_IMAGES = 5 是预定义常数代表固件文件中所包含的最大映像文件数目为5。

字段 offset 指明本映像文件存放在固件文件中的偏移位置。若该项值为-1,则表示本结构项未使用,固件中已没有其他映像文件。len 字段指明本映像文件的长度。

mtdofs 字段指明本映像文件内容将被写入到目标机指定 mtd 分区^[6]中开始的偏移位置。使用这个字段可以让我们将多个映像文件都写入到同一个 MTD 分区的不同偏移开始处。这使得本程序和对应的 FLASH 写入程序具有很大的灵活性。

mtdno 字段指明本映像文件欲被写入目标机 mtd 的分区号。注意,该分区号是内核在 /proc/mtd 文件第1列中给出的分区号,并非 mtd 设备文件中的号码。

mtdname 字段中保存着 MTD 分区的名称。filename 字段中保存着本映像文件的名称,用于 FLASH 写入程序可以从固件文件中抽取和恢复各个独立的映像文件。MAX_NAME_LEN 为预定义常数15。

(下转第51页)

变化;图中直线是该控制曲线的对数,其更直观地表明了控制电流与增益实现了预期的 dB-线性。

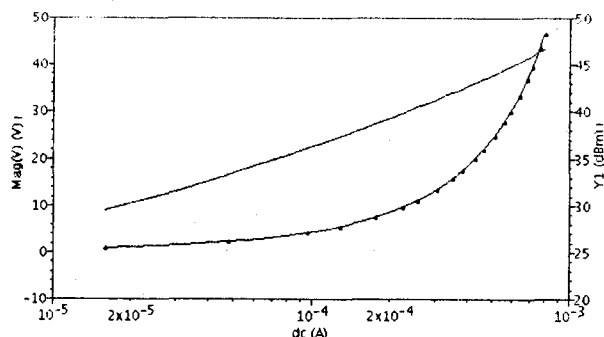


图4 VGA增益与控制信号的关系图

图5是本设计VGA所能实现的最小增益,从图中可以看出它达到了-11.3dB,即对信号衰减。

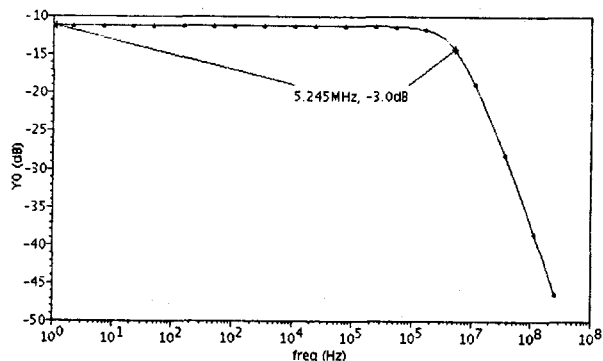


图5 VGA最小增益图

图6是本设计VGA所能实现的最大增益,从图中可看出它达到了33.4dB。

总的仿真结果:控制电流范围为0~800 μ A,电压增益为-11.3dB~33.4dB,带宽为5.2MHz(-3dB),相位裕度为60°,实现了增益随控制信号与的指数变化,即dB-线性。

3 结束语

文中所设计的可变增益放大器以 Gilbert 单元为

基础,利用改进的伪指数电路实现了控制信号对增益的指数控制,即 dB-线性。并采用差分转换、共模反馈和频率补偿等技术完善了该电路的功能。HSPICE 仿真结果表明该可变增益放大器电路实现了信号增益在-11.3dB~33.4dB的连续变化范围,相位裕度60°。

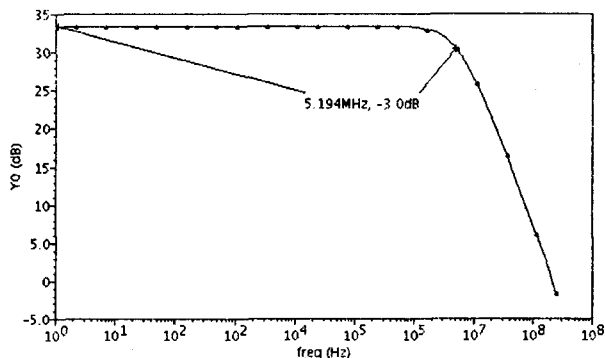


图6 VGA最大增益图

参考文献:

- [1] 侯俊钦. 微伏信号放大系统设计[J]. 计算机技术与发展, 2007,17(10):198-201.
- [2] Abdelfattah K M, Soliman A M. Variable Gain Amplifier Based on a New Approximation Method to Realize the Exponential Function[J]. IEEE Trans on Circuits and Systems I: Fundamental Theory and Applications, 2002,49(9):1348-1354.
- [3] Motamed A, Hwang C, Changku, et al. A Low-voltage Low-power Wide-range CMOS Variable Gain Amplifier[J]. IEEE Trans on Circuits and Systems II: Analog and Digital Signal Processing, 1998,45(7):800-811.
- [4] Harjani R. A Low2power CMOS VGA for 50 Mb/s Disk Drive Read Channels[J]. IEEE Trans on Circuits and Systems II: Analog and Digital Signal Processing, 1995,42(6):370-376.
- [5] Razavi B. Design of Analog CMOS Integrated Circuits[D]. [s.l.]:McGraw-Hill Higher Education, 2000.

(上接第47页)

4 结束语

文中所设计的格式克服现有格式中未包含关于映像文件详细信息的缺点,为固件更新程序提供足够的信息来完成Flash写入操作,因此这种格式具有更高的灵活性和通用性。

参考文献:

- [1] Kylinfs. Grub 源代码分析[EB/OL]. 2006. <http://kylinfs.bokee.com/viewdiary.12791369.html>.

- [2] 江永忠. 深入Linux的LILO[EB/OL]. 2005. http://www.ccw.com.cn/htm/app/linux/admin/01_7_25_6.asp.
- [3] Scheele C. dd-wrt 网站[EB/OL]. 2008. <http://www.dd-wrt.com>.
- [4] Fainelli F. The OpenWrt embedded development framework[EB/OL]. 2008. <http://www.openwrt.org>.
- [5] Ivens K, Scheffy C. USRobo-tics Home Networking For Dummies[EB/OL]. 2006. <http://www.usr.com>.
- [6] Zeus J. linuxMTD 源代码分析[EB/OL]. 2002. <http://www.cndzz.com/download/soft/46808.htm>.