

# LPC3250 嵌入式系统的图像采集

## Image Acquisition Based on LPC3250 Embedded System

冯欢 张红梅 宋天枢

(桂林电子科技大学信息与通信学院,广西 桂林 541004)

**摘要:** 图像采集是进行图像处理的前提,而驱动图像采集设备又是实现图像采集的基础。针对嵌入式系统的图像采集,给出了基于 LPC3250 硬件平台和嵌入式 Linux 软件平台的图像采集系统。系统对摄像头设备进行静态加载驱动,并利用 Video4Linux (V4L) 视频采集应用程序接口编程技术编写图像采集程序,实现了基于中星微 ZC301B DSP 控制芯片的 USB 摄像头的图像采集。实验结果表明,该系统具有较好的实时性,采集到的图像对于进行进一步图像处理有重要的实际意义。

**关键词:** 图像采集 嵌入式系统 Linux 影像设备接口 摄像头驱动

**中图分类号:** TP368.2;TP752.1

**文献标志码:** A

**Abstract:** Image acquisition is the premise of image processing, and driving image acquisition equipment is the basis for implementing image acquisition. Aiming at the image acquisition upon embedded system, the image acquisition system based on LPC3250 hardware platform and embedded Linux software platform are given. The camera equipment is loading driven statically by the system, and the image acquisition program is written with the programming interface technology for Video4Linux (V4L) application program, the USB camera image acquisition based on ZC301B DSP control chip is implemented. The result of experiment indicates that this system features better real-time performance; and the images acquired are significant to further image processing.

**Keywords:** Image acquisition Embedded system Linux Imaging equipment interface Camera driver

## 0 引言

随着图像处理技术及嵌入式系统的发展,利用嵌入式系统进行图像处理,已使如视频监控、视频电话和视频会议等应用成为可能。而如何在嵌入式系统上进行图像采集则是实现这些应用的前提。

目前,已有很多方案实现了在嵌入式 Linux 上进行图像采集的功能,这些方案的硬件平台大多是基于三星公司的 S3C2410 或 S3C2440<sup>[1-4]</sup>。但是由 S3C2410 与 S3C2440 微处理器集成的 ARM920T 处理器内核不具备 DSP 处理能力,因此,对后续进行图像处理有一定的局限性。

上海丰宝公司和桂林电子科技大学联合开发了 LPC3250 平台,其 CPU 内核提供了 DSP 处理能力,可以从一定程度上提高后续图像处理的速度。因此,本文基于 LPC3250 硬件平台与嵌入式 Linux 软件平台,重点介绍了在 LPC3250 平台上利用 Video4Linux (V4L) 编写应用程序,进而进行图像采集的图像处理方案。

修改稿件到日期:2010-11-03。

第一作者冯欢,女,1985 年生,现为桂林电子科技大学信号与信息处理专业在读硕士研究生;主要从事嵌入式系统方面的研究。

## 1 系统简介

### 1.1 系统硬件结构

LPC3250 平台采用了恩智浦公司基于 ARM926EJS 内核的 LPC3250 微控制器。该控制器内带 MMU、32 kB 指令高速缓存、32 kB 数据高速缓存、高达 256 kB 的内部 SRAM 和矢量浮点协处理器,CPU 内核还包含了带有单周期 MAC 操作的一系列 DSP 指令扩展以及 Jazelle Java 字节代码执行,可在高于 200 MHz 的 CPU 频率下工作。在存储设备上,LPC3250 平台具备 32 MB 的 NAND Flash、4 MB 的 NOR Flash 和 64 MB 的 SDRAM。此外,LPC3250 平台还拥有 3 个串口、1 个网络接口、3 个 USB 接口、1 个存储卡接口和 LCD 及触摸屏接口等<sup>[5]</sup>。系统硬件结构如图 1 所示。

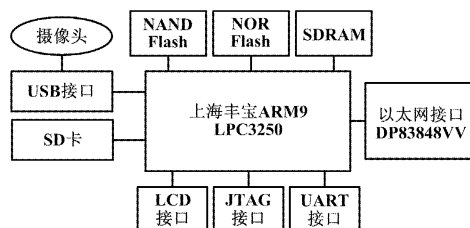


图 1 LPC3250 平台硬件结构图

Fig. 1 Structure of LPC3250 hardware platform

为实现图像采集,在 USB 接口上外接了一个 USB 摄像头。在摄像头的选取上,本文采用了市场上应用最为广泛的中星微公司生产的 ZC301B 芯片摄像头,摄像头采集到的图像为 JPEG 编码格式。

## 1.2 系统软件结构

图像采集系统软件结构如图 2 所示,整个系统软件基于嵌入式 Linux。

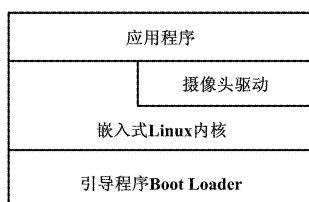


图 2 系统软件结构图

Fig. 2 Structure of the systematic software

Linux 是目前较为流行的一款源代码开放的操作系统。目前,正在开发的嵌入式系统中 70% 以上的项目选择 Linux 作为嵌入式操作系统。

嵌入式 Linux 具有内核精简、性能高且稳定、支持多任务、可伸缩、外设接口统一、源代码开放、软件资源丰富和适用面广等特点,适用于不同的 CPU 架构。为此,本文操作系统采用嵌入式 Linux 操作系统。嵌入式 Linux 内核为 linux-2.6.27 内核,文件系统使用 jffs2,图像采集和其他操作均建立在嵌入式 Linux 内核之上。

## 2 基于 V4L 的图像采集

V4L 是 Linux 内核中支持影像设备的一组 APIs,它配合适当的视频采集设备和相应的驱动程序,可以实现影像采集、AM/FM 无线广播、影像 CODEC 和频道切换等功能。目前,V4L 主要应用于影像串流系统与嵌入式影像系统。因此,其在远程会议、可视电话和视频监控系统中有着广泛的应用<sup>[6-7]</sup>。

在嵌入式 Linux 下应用 USB 摄像头采集图像主要分为成功驱动 USB 摄像头和图像采集两步。

### 2.1 USB 摄像头的驱动

成功驱动 ZC301B 芯片的 USB 摄像头的关键在于保证在内核中成功加载 USB 设备驱动、V4L 标准和 ZC301B 芯片驱动三部分(三者缺一不可)<sup>[8]</sup>。因此,需重新配置并编译内核。具体操作方法为:在 Linux2.6.27.8 源文件目录下依次运行 make phy3250\_defconfig、make menuconfig,进入内核系统配置选项,然后分别对 USB 设备驱动、V4L 标准和 ZC301B 驱动进行添加。

#### 2.1.1 USB 设备驱动的添加

运行 make menuconfig 后,在系统配置中进行如下

的选择即可添加 USB 设备驱动。

```

System Type - - - >
  LPC32xx chip components - - - >
    I2C interfaces - - - >
      [ * ] Enable the USB OTG I2C peripheral
Device Drivers - - - >
  [ * ] - > USB support - - - >
    < * > OHCI HCD support
    < * > USB Mass Storage support
  SCSI device support - - - >
    < * > SCSI device support
    < * > SCSI disk support
  
```

#### 2.1.2 V4L 和 ZC301B 驱动的添加

加载 ZC301B DSP 控制芯片的驱动方式有静态加载和动态加载两种。

静态加载是指直接把驱动程序编译进内核,在系统启动后可直接调用,不需要任何加载或卸载命令。其优点是使用方便,缺点是将模块编译进内核后,生成的内核很大,占用宝贵的系统资源,而且一旦要在现有的内核中新增或者去除某项功能,必须重新编译内核,效率较低。

动态加载是指利用 Linux 的模块特性,在系统启动后通过 rmmod 或 modprobe 命令挂载 .ko 内核目标文件,对模块进行加载,加载成功后可通过 mkmod 指令进行节点挂载,并在不需要时通过 rmmod 命令来卸载模块。这种加载方式因模块本身没有被编译进内核,从而在一定程度上控制了内核的大小,节约了系统资源;此外,模块一经加载,其和内核中其他部分完全一样,使用灵活。不足之处在于每次调用时都需要通过命令进行模块的加载与卸载,操作麻烦。

鉴于开发平台硬件资源充足,为方便使用,本文的芯片驱动加载采用静态加载,即将 ZC301B 芯片摄像头驱动编译进内核。具体配置选项如下。

```

Device Drivers - - - >
  ( * ) Multimedia device - - - >
    < * > Video For Linux
      [ * ] Video capture adapters - - - >
        [ * ] V4L USB devices - - - >
          [ * ] USB GSPCA driver
  
```

其中,GSPCA 是摄像头驱动程序,当此驱动被加载以后,将会调用“usb\_register”函数来注册驱动,usb\_register 结构中的“device\_table”包含驱动支持的设备类型。当摄像头插入后,Linux 内核会在所有的驱动中寻找一个适合该设备的驱动<sup>[9]</sup>。

完成以上添加后,退出内核配置,保存配置菜单,

在终端依次运行 make、make uImage, 编译出内核镜像文件 uImage; 将生成的 uImage 烧写到开发板中, 并用新内核重新启动开发板。此时, 可以在启动信息中看到摄像头的驱动已经成功编译到内核, 信息如下。

```
gspca: main v2.2.0 registered
usbcore: registered new interface driver zc3xx
zc3xx: registered
```

插入 USB 摄像头, 系统将正确识别 USB 摄像头。串口显示信息如下。

```
[root@linpo /]# usb 1-1: new full speed USB device using
usb-ohci and address 2
usb 1-1: configuration #1 chosen from 1 choice
gspca: probing 0ac8:301b
zc3xx: probe 2wr ov vga 0x0000
zc3xx: probe sensor -> 11
zc3xx: Find Sensor HV7131R(c)
gspca: probe ok
```

摄像头被成功识别的同时, 在文件系统的“/dev”目录下会产生名为“video0”的摄像头设备文件。此时, 摄像头即可以在开发板上正常使用。

## 2.2 图像采集的具体实现

开发板成功识别摄像头后, 即可在嵌入式 Linux 下进行图像采集。基于 V4L 的图像采集流程如图 3 所示。

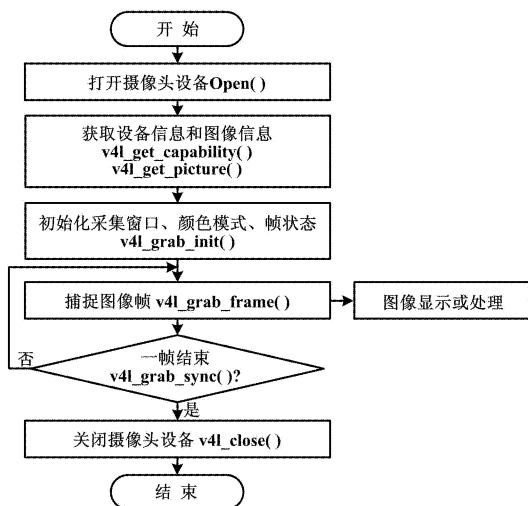


图 3 图像采集流程图

Fig. 3 Flowchart of image acquisition

图像采集的具体实现过程如下。

### ① 打开摄像头设备

在 Linux 下, 可以通过系统的设备文件来访问摄像头。摄像头在系统中所对应的设备文件为/dev/video0, 调用的函数为: `vd->fd = open(dev, O_RDWR)`。

函数调用成功则打开设备并返回相应的文件描述符 `vd->fd`, 以后的系统调用函数就可使用它来对设备文件进行操作; 函数调用失败则返回错误信息。

### ② 获取设备信息

/usr/include/linux/videodev.h 中定义了编写应用程序的所有数据结构、宏和函数。获取摄像头信息调用的函数为: `ioctl(vd->fd, VIDIOCGCAP, &(vd->capability))`。

通过访问结构体 `vd->capability`, 即可获得摄像头的基本信息, 如摄像头名称、支持的最大和最小宽度与高度、信号源等信息。

### ③ 获取图像属性信息

获取图像属性信息调用的函数为: `ioctl(vd->fd, VIDIOCGPICT, &(vd->picture))`。

通过访问结构体 `vd->picture`, 即可获得图像的基本信息, 包括对比度、色深和调色板等。

### ④ 图像采集

图像采集方式有 `read()` 直接读取方式和 `mmap()` 内存映射方式两种。`read()` 方式是从内核缓存区中读取数据到内存当中; 而在 `mmap()` 方式下, 进程之间通过映射同一个文件实现内存共享, 即摄像头的图像缓冲区与用户能访问的图像数据区共享同一段内存区域, 从而绕过内核缓冲区。文件被映射到进程地址空间后, 进程可以像访问普通内存一样对文件进行访问, 不必再调用 `read()`、`write()` 等操作。采用共享内存方式加速了 I/O 访问, 提高了效率。这是因为进程可以直接读写内存, 而不需要任何数据的拷贝。鉴于以上优点, 本文采用 `mmap()` 方式获取图像。

在使用 `mmap()` 方式之前, 必须先通过 `ioctl(vd->fd, VIDIOCGMBUF, &(vd->mbuf))` 获得摄像头设备可映射的摄像头内存大小、摄像头可同时存储的帧数以及每帧图像的偏移量。

使用 `mmap()` 内存映射获取图像时, 主要涉及到以下 4 个函数。

① `int v4l_mmap_init(v4l_device * vd), vd->map = (unsigned char *)mmap(0, vd->mbuf.size, PROT_READ | PROT_WRITE, MAP_SHARED, vd->fd, 0)`。该函数将摄像头图像数据映射到进程中, 则此时使用 `vd->map` 指针就可以使用采集到的图像数据。

② `int v4l_grab_init(v4l_device *, int, int)` 完成图像采集前的初始化。

③ `int v4l_grab_frame(v4l_device *, int), ioctl(vd->fd, VIDIOCMCAPTURE, &(vd->mmap))`。该函数为 `mmap()` 方式下真正的图像采集函数, 若调用成功, 则

开始一帧图像的截取,是非阻塞的。而图像截取是否完毕由 VIDIOCSYNC 判断。

④ `int v4l_grab_sync (v4l_device *), ioctl (vd->fd, VIDIOCSYNC, &(vd->frame_current))` 完成截取图像的同步工作,它在截取一帧图像后调用,返回表明一帧图像截取结束。

利用 V4L 编写图像采集程序与 Makefile 文件,在宿主机上交叉编译图像采集程序后查看编译好的可执行文件如下。

```
[hh@localhost image] $ file image
image: ELF 32-bit LSB executable, ARM, version 1, dynamically linked (uses shared libs), for GNU/Linux 2.6.27, not stripped
```

从上面的文件属性可以看出,得到的可执行文件是针对本文的嵌入式操作系统,将该可执行文件复制到开发板上运行,可以看到如下信息。

```
[root@linpo /]# ./image
open success!
set_norm success!
init success!
size = 475136
memory map success!
zc3xx: probe 2wr ov vga 0x0000
get image success!
syn success!
img address 0x402d8000
```

运行图像采集程序的可执行文件,则在开发板的指定目录下得到了像素为  $320 \times 240$  的图像。通过 Linux 下程序运行时间算法计算可知,整个程序运行时间约为 0.07 s,图像采集部分运行时间约为 0.04 s,系统具有较好的实时性。

### 3 结束语

本文在 LPC3250 嵌入式 Linux 平台上利用 V4L 编程实现了基于中星微 ZC301B 芯片的 USB 摄像头的图像采集。实验证明,本文采用的图像采集方案具有较好的实时性。在实际应用中,用户也可以根据实际需求更改应用程序,通过 FrameBuffer 设备在 LCD 上实时显示图像;或将采集到的图像数据传入图像处理算法中,并借助 OpenCV 计算机视觉库对采集到的图像进行进一步的处理。

#### 参考文献

- [1] 常慕,洪健,李钟慎. 嵌入式机器视觉系统的图像采集及显示技术[J]. 自动化仪表,2010,31(3):16-18.
- [2] 汪庆年,张广栋,李桂勇. 嵌入式 Linux 视频采集系统的设计与实现[J]. 微计算机信息,2009,25(11):39-40.
- [3] 吴晴,周健. 嵌入式图像采集系统的设计与实现[J]. 电子测量技术,2007,30(6):89-92.
- [4] 郭剑,赵建. 嵌入式 Linux 的图像采集与显示[J]. 现代电子技术,2006,29(7):129-131.
- [5] 桂电-丰宝联合实验室. 基于 LPC3250 的嵌入式 Linux 系统开发[M]. 北京:电子工业出版社,2010:8-10.
- [6] 电脑编程技巧与维护杂志社. Linux 编程典型实例解析[M]. 北京:清华大学出版社,2009:316-320.
- [7] 陈俊宏. Embedded Linux 嵌入式系统原理与实务[M]. 北京:中国铁道出版社,2004:184-224.
- [8] Li Quanxi, Liu Peiqian, Li Changyou. Research on embedded video monitoring system based on Linux[C] // Proceedings of the 2009 International Conference on Computer Engineering and Technology, Singapore, 2009:478-481.
- [9] Tian Huan, Li Hui, An Huiyao. Special effect drivers for camera on an open source platform[C] // Proceedings of the 2009 Second International Conference on Information and Computing Science, Manchester, England, UK, 2009:339-334.

#### (上接第 82 页)

在系统设计中引入  $\mu\text{C}/\text{OS}-\text{II}$  是 DSP 应用复杂化后编程方式的一种改变趋势,可处理多任务的复杂系统,大大缩短了系统开发周期,便于程序的编写和调试。 $\mu\text{C}/\text{OS}-\text{II}$  较高的可移植性也有利于系统的进一步改进和升级。目前,系统正在测试当中,任务调度正常,运行效果良好。

#### 参考文献

- [1] 姜士林,赵长汉. 感应加热原理与应用[M]. 天津:天津科技翻译出版公司,1993:10-11.
- [2] 赵刘强,林永君,马良玉,等. 基于  $\mu\text{C}/\text{OS}-\text{II}$  嵌入式系统的数据采集系统[J]. 微计算机信息,2010(5):62-66.

- [3] 潘超军,陈刚.  $\mu\text{C}/\text{OS}-\text{II}$  在 TMS320F2812 上的移植[J]. 江苏电机工程,2005,24(4):33-34.
- [4] 高阳,潘宏侠,吴升,等. 基于 DSP 的电力设备远程监测分析系统[J]. 电力自动化设备,2010,30(1):127-131.
- [5] 左丽霞,卢山. 基于 DSP 和  $\mu\text{C}/\text{OS}-\text{II}$  的馈线终端装置设计方案[J]. 华东交通大学学报:自然科学版,2009,26(5):64-69.
- [6] 韩丰田. TMS320F281xDSP 原理及应用技术[M]. 北京:清华大学出版社,2009:279-280.
- [7] 王忠凯,赵磊.  $\mu\text{C}/\text{OS}-\text{II}$  任务调度研究[J]. 山东理工大学学报:自然科学版,2009,23(2):30-35.
- [8] 孙丽明. TMS320F2812 原理及其 C 语言程序开发[M]. 北京:清华大学出版社,2008:462-465.