

# 基于 S3C2440 的图像采集与显示系统<sup>①</sup>

郑凯华, 章旭东, 俞立, 鲍荣, 蒋国华

(浙江工业大学 信息工程学院, 杭州 310023)

**摘要:** 设计一种基于 S3C2440 硬件平台和嵌入式 Linux 操作系统的图像采集与显示系统。该系统通过 USB 摄像头采集图像数据, 利用 Video4Linux 模块提供的编程接口, 实现图像的采集设计, 基于 Qt/Embedded 实现 LCD 实时显示, 并经 JPEG 压缩算法对图像进行压缩编码用于图像处理或网络传输。整个系统建立在嵌入式架构上, 能独立完成实时图像采集、显示及压缩功能, 可广泛应用于智能交通、计算机视觉等领域。

**关键词:** S3C2440; 嵌入式 Linux; 图像采集; JPEG 压缩; Qt/Embedded

## Image Acquisition and Display System Based on S3C2440

ZHENG Kai-Hua, ZHANG Xu-Dong, YU Li, BAO Rong, JIANG Guo-Hua

(College of Information Engineering, Zhejiang University of Technology, Hangzhou 310023, China)

**Abstract:** An image acquisition and display system based on S3C2440 and embedded Linux is designed. The image data is captured through USB camera, while the programming interfaces provided by the Video4Linux module are used to implement the image collection. Based on Qt/Embedded, the collected image is real-time displayed on LCD. Further, the images are compressed according to JPEG compression algorithm for subsequent process or transmission. The proposed system which is based on the embedded framework performs image data collection, real-time display and image compression. It can be widely used in the fields of intelligent transportation, computer vision and so on.

**Key words:** S3C2440; embedded linux; image acquisition; JPEG compress; Qt/Embedded

## 1 引言

以实时信息采集、处理为核心的图像采集处理系统被广泛地应用于工业控制、工业检测、通讯网络、机器视觉、医学、公安等领域。实时图像的采集方法有很多, 传统的视觉图像采集大多是基于“摄像头—图像采集卡—通用 PC 机”三者为一体的系统, 这种视觉系统不能满足实际应用中对于图像实时处理的需求。目前也有很多基于单片机的数据采集系统, 虽然它们具有高可靠性、集成化等优点, 但是在实时性、远程监控、人机接口等方面不尽人意。很多图像采集系统使用 DSP 作为数据处理单元, 存在成本较高的弊端<sup>[1]</sup>。此外, 某些基于 ARM 的图像采集系统<sup>[2]</sup>并未将采集的图像进行保存, 无法满足用户对图像数据进行历史查询的需求。

随着嵌入式技术的迅猛发展, 以及 Linux 系统的

开源性、可定制性、可移植性等优势, 利用嵌入式 Linux 技术实现图像采集成为可能。本文在嵌入式 Linux 操作系统和以 S3C2440 为核心的硬件系统的基础上, 以 USB 摄像头作为采集模块, 利用 Video4Linux 内核 API 接口函数, 编写摄像头采集程序; 基于 Qt/Embedded 实现图像在 LCD 上的实时显示, 同时利用 JPEG 对采集的图像进行压缩并保存用于后续的网络传输。本系统具有体积小、成本低、稳定性高等优点, 在智能交通、计算机视觉、通信等领域得到广泛应用。

## 2 系统组成

### 2.1 系统的硬件构成

本系统的硬件构成如图 1 所示。系统采用 S3C2440A 作为核心处理器。S3C2440A 是三星公司设计的一款基于 ARM920T 内核的 32 位嵌入式 RISC 微

① 基金项目: 2009 年浙江省大学生科技创新活动计划(2009R403064)

收稿时间: 2010-12-13; 收到修改稿时间: 2011-01-24

处理器，它的最高工作频率可达 533 MHz，内部集成了通用的 LCD 控制器、8 通道 10 位 ADC 和触摸屏接口，且具备高性能、低功耗的优点，适用于智能手机、便携式媒体播放器、手持导航仪等领域<sup>[3]</sup>。另外，系统配置了 64MB 的 FLASH 和 64MB 32 位的 SDRAM 以满足系统存储操作系统、应用程序和数据的需求；采用 OV511 芯片的 USB 摄像头作为视频采集模块；配备分辨率为 320\*240 的 LCD 显示器，以便实时显示采集到的图像。

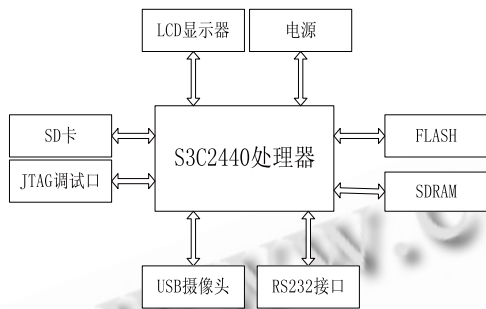


图 1 系统硬件构成

### 2.2 系统的软件构成

系统的软件构成由引导程序(Bootloader)、嵌入式 Linux 内核 (Kernel)、文件系统 (Cramfs) 和基于 QT/Embedded 的用户图形界面显示程序和其他应用程序组成，如图 2 所示。

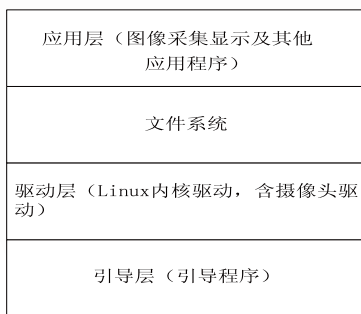


图 2 系统软件平台框图

引导程序在系统启动时进行内核重映射，将保存在 FLASH 中的操作系统和应用程序映射到 SDRAM 中；驱动层是包含摄像头驱动的 Linux 内核驱动，其中 Video4Linux(简称 V4L)是 Linux 中关于视频设备的内核驱动，它是针对视频设备的应用程序编程提供的一系列接口函数，这些视频设备包括现今市场上流行的 TV 卡、视频捕捉卡和 USB 摄像头等。文件系统采用的是 Yaffs，它包括应用程序、模块、配置文件和库。

系统应用程序建立在 Linux 内核、摄像头驱动和文件系统上。

特别地，由于采用 USB 摄像头进行图像数据采集，内核应支持 USB 设备并包含相应的摄像头驱动，因此，在编译内核时，在 usb support->目录下用(\*)选择 support for usb 和 usb camera ov511 support。这就实现内核中加载以 OV511 为接口芯片的 USB 数字摄像头的驱动支持。

### 3 系统实现

#### 3.1 基于 Video4Linux 的视频图像采集

因为在嵌入式 Linux 的内核定制过程中，已经加入了对 V4L 模块及 USB 摄像头的支持，可以调用 V4L 提供的编程接口(API)从 USB 摄像头中获取图像数据，也可以保存图像数据以备后续开发应用。V4L 模块提供的主要 API 函数有：①VIDIOC\_GCAP 函数：返回视频采集设备的基本信息，包括设备名称、设备类型、信道数、最大及最小像素高度和宽度等；②VIDIOC\_GPICT 函数：返回图像的基本信息，包括了亮度，对比度，色深，调色板等；③VIDIOC\_SFBUF 函数：设置采集设备的帧缓存参数，包括缓存区大小、同时采集的帧数、偏移量等；④VIDIOC\_GWIN 函数：设置采集窗口的大小参数；⑤VIDIOC\_SPICT 函数：获取和设置采集图像帧的属性，如亮度、色调和对比度等；⑥VIDIOC\_MCAPTURE 函数：实现视频信号采集<sup>[4]</sup>。利用这些接口函数，视频采集流程图如图 3 所示。

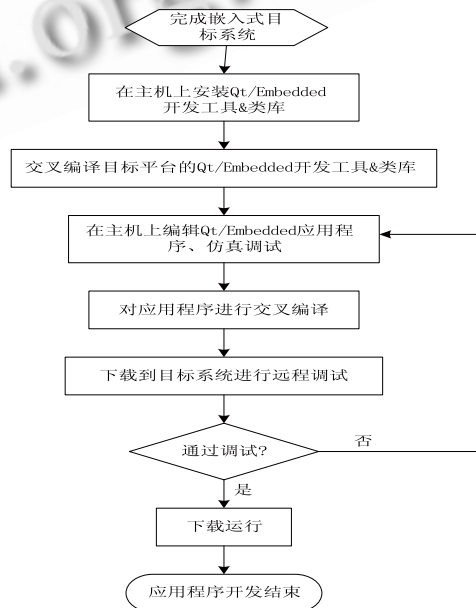


图 3 视频采集流程图

特别地, 读入图像数据的两种方法: 一是直接读设备, 即调用 `read` 函数; 二是通过内存映射(`mmap`)得到视频驱动的数据缓冲, 然后直接对映射后的缓冲进行读操作。下面分别介绍:

(1) `read` 系统调用方式比较简单, 只需将摄像头参数传入 `read` 函数中, 得到图像数据指针后返回即可: `read(vd->fd, picture_p, width*height)`; 其中 `vd->fd` 为设备文件描述符, `picture_p` 为得到图像数据指针, `width*height` 为图像的大小;

(2) 内存映射使用函数 `ioctl(vd->fd, VIDIOC GMBUF, &(vd->mbuf))` 获得摄像头存储缓冲区的帧信息, 然后使用函数 `((vd->map=mmap (0,vd->mbuf.size, PROT_READ|PROT_WRITE,MAP_SHARED, vd->fd, 0))` 把摄像头对应的设备文件映射到内存区, 该映射内容区可读可写且不同进程间可共享。该函数成功时返回图像数据的指针, 失败时返回值为-1。

`mmap` 系统调用使进程之间通过映射同一个普通文件实现共享内存, 普通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。采用共享内存通信的一个显而易见的好处是效率高, 因为进程可以直接读写内存, 而不需要任何数据的拷贝。因此我们使用 `mmap` 系统调用来读入图像数据。

### 3.2 基于 Qt/Embedded 的图像显示

Qt 是一个跨平台的 C++ 图形用户界面库, 由 Troll Tech 公司出品, Qt 具有优良的跨平台特性, 面向对象的程序设计, 丰富的 API、XML 支持等优点, 越来越多地应用在 PDA、掌上设备、网络设备、工业自动化仪表等高端设备中<sup>[5]</sup>。

Qt/Embedded 是面向嵌入式系统的 Qt 版本。Qt/Embedded 的底层图形引擎基于 `framebuffer`。`Framebuffer` 是在 Linux 内核架构版本 2.2 以后推出的标准的显示设备驱动接口。采用 `mmap` 系统调用, 可以将 `framebuffer` 的显示缓存映射为可连续访问的一段内存指针<sup>[6]</sup>。

为了方便开发, 需要建立两套环境系统, 一套在 PC 主机上, 我们称之为宿主机, 而另一套在硬件系统上, 称之为目标机。我们要做的就是先要在宿主机上开发出理想中的界面, 再通过交叉编译生成在目标机上可执行的二进制代码, 然后移植到目标机上进行调试。

在 Qt4 版本中提供这样一个函数 `QImage(const`

`uchar *data, int width, int height, Format format)`; `*data` 表示图像的内存地址, `width` 表示图像的宽, `height` 表示图像的高, `format?` 表示图像格式。所以只要得到图像的内存地址, 要求显示图片的宽, 高, 以及采集图像的格式, 就能创建 `QImage` 类型的对象。然后创建 `PaintEvent` 的操作对象, 使用 `drawImage` 函数将 `QImage` 类型的对象显示出来。

调用以上函数, 宿主机图像显示画面如图 4 所示。



图 4 宿主机 Qt 图像显示

在宿主机上仿真显示成功之后, 我们将程序进行交叉编译, 然后将程序下载到硬件系统上, 运行结果如图 5 所示。



图 5 硬件系统 LCD 上的图像显示

### 3.3 基于 JPEG 的图像压缩

由于原始图像数据比较大, 不便于后续的网络传输和保存, 本系统采用 JPEG 压缩方式对采集到的原始数据进行压缩处理。

JPEG 静止图像压缩标准是一种被广泛认可的标准, 一般的操作系统和应用都支持。JPEG 标准是在变换编码的基础上, 综合应用了 DCT 和哈夫曼编码两种手段达到很好的图像压缩效果, 基于 DCT 的编码方法是 JPEG 算法的核心内容。

在 JPEG 算法中通常是先将图像分成一个个 8\*8

的图像子块，对每一个图像子块进行离散余弦变换，所使用的二维离散余弦变换和逆变换公式如下：

$$F(u, v) = \frac{1}{4} k(u)k(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \quad (1)$$

$$f(x, y) = \frac{1}{4} k(u)k(v) \sum_{x=0}^7 \sum_{y=0}^7 F(u, v) \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \quad (2)$$

$$k(u) = k(v) = \begin{cases} 1 & \text{当 } u, v = 0 \\ \sqrt{2} & \\ 0 & \text{当 } u, v \neq 0 \end{cases} \quad (3)$$

其中  $x, y$  代表图像数据矩阵内某个数值的坐标位置； $f(x, y)$  代表图像数据矩阵内的某个数值； $u, v$  代表 DCT 变换后矩阵内某个数值的坐标位置； $F(x, y)$  代表 DCT 变换后矩阵内的某个数值。(2)式中，当  $u = v = 0$  时， $k(u) = k(v) = 1/\sqrt{2}$ ；当  $u, v$  为其他值时， $k(u) = k(v) = 0$ 。 $f(x, y)$  经 DCT 变换后， $F(0,0)$  是直流系数，其他为交流系数。在离散余弦变换中，设法使变换后的图像子块的能量下降速度加快，减少块效应并使细节部分清晰。正是基于这一点，通过收敛性良好的 Chebychev 多项式来拟合离散余弦变换，使得变换后的图像子块的能量下降速度加快，从

而提高图像数据的压缩比，同时还能够改善图像的块效应。压缩实现过程如图 6 所示。

图 7 为同一背景下大小均为 320\*240 的照片，其中图 7a 是未压缩的图像，数据量为 225K，图 7b 为嵌入式 Linux 下经过 JPEG 压缩后的图像，大小为 7.69K，结果表明图像数据量大大减少，压缩效果良好。



(a)未压缩的图像 (b)压缩后的图像

图 7 图像压缩前后比较

#### 4 结束语

本系统是以 S3C2440 为硬件核心的图像采集系统。32 位的微处理器 S3C2440 中丰富的硬件资源方便了系统设计。在软件设计中，采用嵌入式 Linux 操作系统运行稳定、效率高，在图像采集及处理方面，具有开发过程简捷高效、灵活多变等优势。本系统可作为环境监测、安全监控、远程操作等应用，在工业控制、监控系统、视频电话等应用领域中有广阔的发展空间。

#### 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.自动化仪表,2010,31(3):18-22.
- 2 查婧,刘波,曹剑中.嵌入式视频采集与网络传输系统.电子器件,2009,32(3):646-648.
- 3 徐英慧,马忠梅,王磊.ARM9 嵌入式系统设计—基于 S3C2410 与 Linux.北京:北京航空航天大学出版社,2007.
- 4 汤霄峰,郑善贤.基于  $\mu$  Clinux 的远程监控系统的设计与实现.计算机工程与科学,2009,31(10):141-143.
- 5 任善全,吕强,钱培德,杨季文.一个基于 Qt/Embedded 的嵌入式 Linux 应用程序的实现.计算机应用与软件,2006, 23(2):105-107.
- 6 倪继利.Qt 及 Linux 操作系统窗口设计.北京:电子工业出版社,2006.101-103.

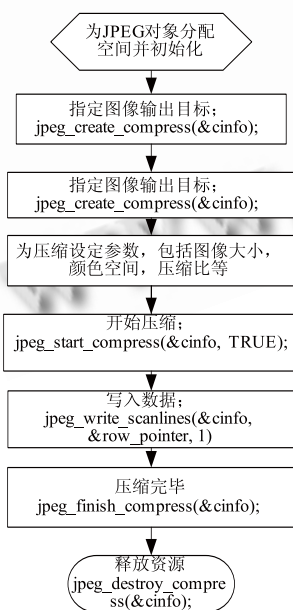


图 6 压缩实现过程