

文章编号: 2095-2163(2019)01-0065-04

中图分类号: TP391

文献标志码: A

# 基于 Cortex-A53 的嵌入式图像采集系统设计

许 晓, 毕远伟

(烟台大学 计算机与控制工程学院, 山东 烟台 264005)

**摘 要:** 为实现具有集成本地图像处理功能的嵌入式图像采集系统, 采用嵌入式 Linux, 选用 Cortex-A53 架构的处理器 S5P6818 和具有 MIPI 接口的 OV5645 CMOS 图像传感器, 设计了系统的硬件连接电路, 研究了在 Linux 下基于 V4L2 的摄像头的驱动框架并实现了 OV5645 的设备驱动, 编写嵌入式 Linux 应用程序进行实验验证。成功调用 V4L2 的 API 实现图像抓取和处理, 完整实现了嵌入式图像采集系统的软硬件设计。

**关键词:** 嵌入式; Linux; S5P6818; CMOS 图像传感器; MIPI; V4L2

## Design of embedded image-capturing system based on Cortex-A53

XU Xiao, BI Yuanwei

(School of Computer and Control Engineering, Yantai University, Yantai Shandong 264005, China)

**[Abstract]** In order to achieve the embedded image-capturing system with local image processing function, the paper uses the embedded Linux, chooses the processor S5P6818 of Cortex-A53 architecture and the OV5645 CMOS image sensor with MIPI interface to design the hardware connection circuit of the system. After that, the driving frame of the camera based on V4L2 in Linux is studied, the device driver of OV5645 is realized, the embedded Linux application program is written for experimental verification. Therefore, the paper successfully invokes application programming interface of V4L2 to implement image grabbing and processing, and fully implement the software and hardware design of the image-capturing system under Linux.

**[Key words]** embedded; Linux; S5P6818; CMOS image sensor; MIPI; V4L2

## 0 引 言

计算机技术、嵌入式技术和图像处理技术的日益进步,推动了嵌入式图像采集系统朝着高速化、集成化、高分辨率、智能化<sup>[1]</sup>的方向发展。目前,嵌入式图像采集系统正广泛应用于电子及半导体、包装、汽车、交通、印刷等领域<sup>[2]</sup>。嵌入式图像采集系统<sup>[3]</sup>依托于计算机技术和图像处理技术,为机器视觉的研究和应用提供高品质的本地视频图像采集和处理平台。

传统的图像采集系统主要由摄像头模组和上位机组成。摄像头模组只负责图像采集,后续的图像处理依赖上位机,系统的采集模块和处理模块相分离,给安装和操作造成不便。随着图像采集系统的广泛应用,介绍嵌入式图像采集系统的资料越来越多,但是相关研究资料中大多采用 USB、DVP 接口<sup>[4]</sup>的摄像头,摄像头的成像质量和抗干扰性较差。在嵌入式平台上,对 MIPI 接口摄像头<sup>[5]</sup>的开

发工作缺乏相关论文指导。

本文图像采集系统的处理器选用三星公司的 S5P6818。S5P6818 是 64 位 8 核 Cortex-A53 架构的 ARM 处理器,可稳定的运行在 1.4 GHz 主频以上,同时具有 MIPI 和 DVP 图像数据输入接口。图像传感器选用 OmniVision 公司推出的 500 万像素 CMOS 感光芯片 OV5645<sup>[6]</sup>,支持 MIPI 接口。作者研究并实现了硬件电路设计,OV5645 摄像头驱动编写<sup>[7]</sup>,并在应用程序中成功通过调用 V4L2 提供的 API 实现图像采集<sup>[8,10]</sup>,完成了基于 MIPI 接口的嵌入式的图像采集系统软硬件实现的全过程。

## 1 系统硬件设计

### 1.1 系统硬件组成

嵌入式图像采集系统按照功能可以划分为图像采集模块、图像数据处理模块和通信接口模块 3 个部分。系统的硬件组成框图如图 1 所示。

系统正常工作时,由 CMOS 图像传感器 OV5645 将

**基金项目:** 山东省自然科学基金(ZR2017QF006)。

**作者简介:** 许 晓(1994-),女,硕士研究生,主要研究方向:嵌入式系统应用;毕远伟(1972-),男,硕士,副教授,主要研究方向:计算机视觉、嵌入式系统应用。

**通讯作者:** 毕远伟 Email: biyuanwei@126.com

**收稿日期:** 2018-10-24

光学信号转换成数字信号,然后利用 camera 接口将采集到的图像数据传至图像处理模块,在嵌入式系统上进行图像处理,如格式转换、压缩、保存。系统可以进一步利用通信接口模块完成和上位机的数据传输。

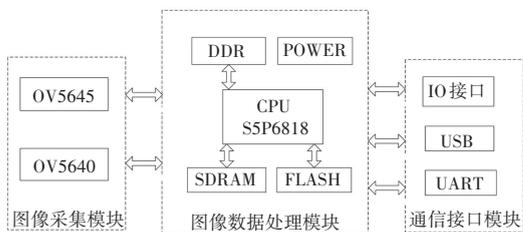


图1 系统硬件组成框图

Fig. 1 Block diagram of system hardware

### 1.2 MIPI 摄像头接口设计

MIPI ( Mobile Industry Processor Interface ) 是为

移动应用处理器制定的开放标准和一个规范。其子协议 CSI 是针对摄像头的高速串行接口应用。同 DVP 接口相比较,MIPI 接口需要的信号线很少,且信号成对存在。MIPI 接口采用一对同步的差分时钟和 1-4 对差分数据线来进行数据传输。OV5645 的 MIPI 接口电路如图 2 所示。

OV5645 支持 ( MDP0, MDN0 ) 和 ( MDP1, MDN1 ) 2 对差分数据线。在传输图像信号时,数据通道工作在高速模式下,此时通道状态为差分的 0 或 1。( MCP, MCN ) 为 OV5645 传输时钟的引脚。SDA 和 SCL 是 OV5645 的 I<sup>2</sup>C 控制信号,用来设置摄像头的时钟、图像输出格式、分辨率等。由于 MIPI 采用差分信号高速传输,因此在进行 PCB 设计时要严格按照差分对走线要求,并实现阻抗匹配。

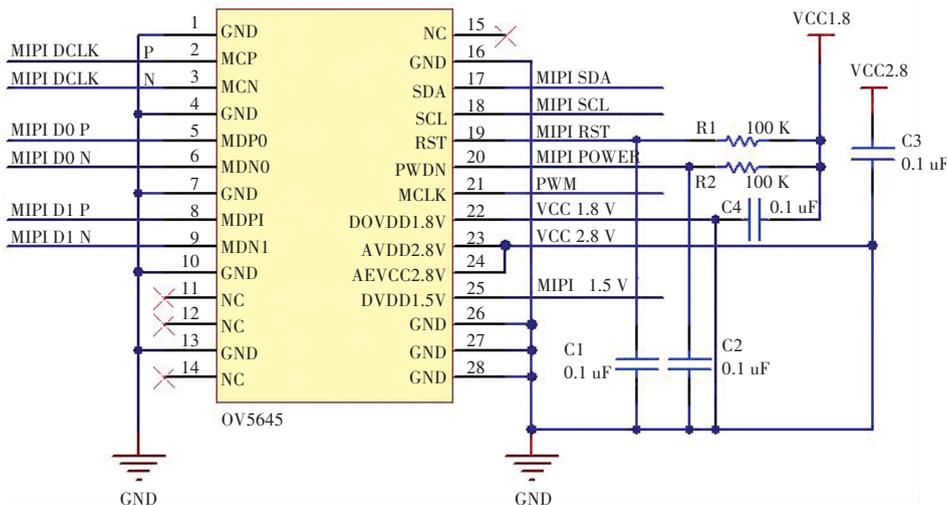


图2 MIPI 接口电路

Fig. 2 MIPI interface circuit

## 2 嵌入式 Linux 开发环境搭建

由于本文的开发环境是在 X86 平台上,而嵌入式 Linux 操作系统和应用程序最终运行在 ARM 架构处理器平台上。因此需要为开发平台搭建交叉编译环境。本次设计在虚拟机 VMware Workstation 上安装了 Ubuntu16.04 操作系统,所采用编译器版本为 arm-linux-gcc-4.5.1,下载对应版本的安装包解压,并添加系统环境变量就完成了基础开发环境搭建。嵌入式系统所需的 Linux 内核、根文件系统和应用程序,需要在该环境下编译成功,再烧写到嵌入式系统上。

## 3 MIPI 摄像头驱动设计

OV5645 属于视频输入设备,其驱动包括 2 部分

的内容:一是控制接口驱动。从 OV5645 的硬件连接电路可知,摄像头和处理器之间的通信是由 I<sup>2</sup>C 总线实现。OV5645 作为 I<sup>2</sup>C 从设备,在这一部分主要完成 Linux I<sup>2</sup>C 子系统的搭建,为 S5P6818 与 OV5645 摄像头之间的数据交互提供管道。二是实现摄像头自身功能的驱动部分。这一部分的工作主要围绕 V4L2 驱动框架展开,完成 video 设备驱动,为应用程序提供控制接口。OV5645 摄像头在 Linux 内核中的驱动框架如图 3 所示。

### 3.1 I<sup>2</sup>C 子系统搭建

I<sup>2</sup>C 设备驱动是基于总线-设备-驱动模型。其实现过程可以划分为设备注册和驱动注册 2 个步骤。图 4 为总线设备驱动模型。

I<sup>2</sup>C 设备的注册就是创建和注册一个 i2c\_client 的过程。在 BSP 文件 device.c 中对 struct i2c\_board

\_info 填充从设备所需要的 id、name、addr、adapter、driver 等数据。在板级初始化时,内核通过调用 i2c\_register\_board\_info 函数将填充的 I<sup>2</sup>c 从设备 OV5645 的相关信息加入到设备链表 \_i2c\_board\_list 中,调用 i2c\_get\_adapter 函数和 i2c\_new\_device 函数来指定设备相连的适配器和注册一个新的 I<sup>2</sup>C 设备。

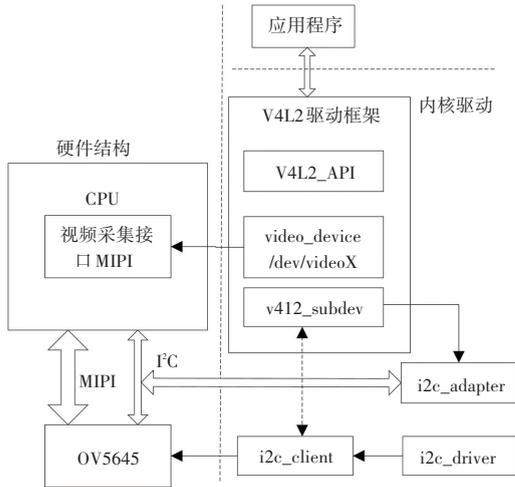


图 3 视频采集驱动框架

Fig. 3 Video capture driver framework

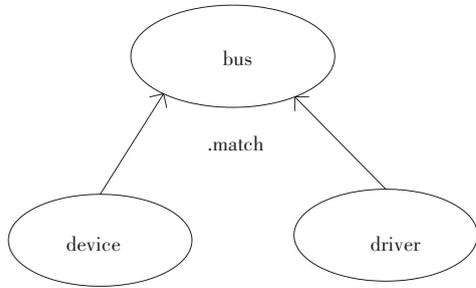


图 4 总线-设备-驱动模型

Fig. 4 Bus-dev-driv model

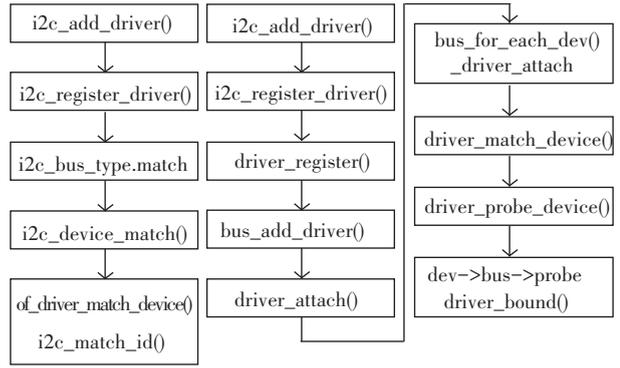
I<sup>2</sup>C 设备驱动注册和设备注册步骤类似,先分配、设置一个 i2c\_driver 的数据结构,实现其成员函数 probe\_remove、id\_table,利用 i2c\_add\_driver 函数注册 i2c\_driver,最终把驱动程序添加到驱动列表中。

I<sup>2</sup>C 设备注册和驱动注册完成后,系统调用 I<sup>2</sup>C 总线结构 i2c\_bus\_type 提供的 match 函数比较设备结构 i2c\_client 和驱动结构 i2c\_driver 结构的 name 是否相同,若相同则调用驱动程序中的 probe 函数。I<sup>2</sup>C 设备和 I<sup>2</sup>C 设备驱动注册不分前后,二者的注册函数都会尝试进行驱动和设备的绑定。图 5 为 I<sup>2</sup>C 设备和驱动匹配的函数调用关系图。

### 3.2 V4L2 驱动模型

V4L2 是 Linux 下视频类设备处理模型,为访问视频设备提供了通用接口。V4L2 驱动中有 3 个核

心结构体 v4l2\_device、v4l2\_subdev、video\_device。v4l2\_device 是所有 v4l2\_subdev 的父设备,负责管理注册在其下面的子设备,v4l2\_device 通常被嵌入到一个特定的结构体中,在 S5P6818 中被嵌入到 nxp\_v4l2 中。v4l2\_subdev 代表子设备,描述了子设备的相关属性和操作。video\_device 结构体用于生成设备节点。



(a) match 过程 (b) probe 过程  
(a) match process (b) probe process

图 5 I<sup>2</sup>C 设备和驱动匹配过程

Fig. 5 I<sup>2</sup>C device and driver matching process

驱动程序首先分配设置一个 video\_device 结构体,并重点实现 2 个操作集: v4l2\_file\_operations 和 v4l2\_ioctl\_ops,然后调用 video\_register\_device 函数注册 video 设备,最终 OV5645 以节点/dev/video1 的形式暴露给应用层。

由图 3 Linux 内核的视频采集驱动框架可知,在实现摄像头 OV5645 的驱动过程中,其既作为 I<sup>2</sup>C 子系统下的从设备 i2c\_client,又作为 V4L2 驱动模型中的子设备 v4l2\_sbudev。二者之间通过初始化函数 v4l2\_i2c\_subdev\_init(sd, client, &ov5645\_subdev\_ops) 建立 v4l2\_subdev 和 i2c\_client 联系,使得 video 通过用户传入的 ioctl 命令来对设备进行控制。

## 4 应用程序设计和测试结果

### 4.1 应用程序设计

在 Linux 下,摄像头 OV5645 硬件已经被映射为设备文件"/dev/video1",直接利用 open() 打开对应的设备文件,通过 ioctl 函数来控制摄像头,如设置图像分辨率、视频数据格式、开始/结束视频显示等。调用 V4L2 接口进行视频采集的流程如图 6 所示。

在数据采集过程中,驱动程序将采集到的视频数据存放在内核空间中,此时用户无法直接访问。为了获取相机采集到的视频数据,V4L2 提供了 2 种

方法:一种是用直接 read 和 write 方式。虽然这种直接把视频数据从内核空间拷贝到用户空间的方法比较简单,但是视频数据过多,造成了拷贝效率过低以及内核空间的过度占用等问题。另一种是 mmap 的方式。mmap 是 v4l2\_file\_operation 结构体的成员函数,相机驱动文件通过对 v4l2\_file\_operation 数据结构中填充 mmap 函数,就可以利用 mmap 函数建立缓冲区和用户空间的映射,直接在用户空间读取到视频数据。同直接读写方式相比,利用 mmap 的方式采集视频数据不需要从内核拷贝大量的视频数据,工作效率更高,因此本次设计采用内存映射的方式采集数据。

部分代码如下:

```
Buffers [ i ]. start = mmap ( NULL, buffers [ i ].
length, PROT _ READ | PROT _ WRITE, MAP _
SHARED, fd_v4l, buffers. offset );
```



图7 OV5645采集的图像

Fig. 7 The result of capture image

## 5 结束语

本文提出了在 Cortex-A53 平台上构建嵌入式图像采集系统的设计方案。完成了 MIPI 接口摄像头同 S5P6818 的硬件电路设计,介绍了嵌入式系统开发环境的搭建方法,并且详细介绍基于 V4L2 驱动框架摄像头驱动设计和相关工作,实现了调用 V4L2 提供的接口抓取图像过程。对在 Linux 下进行摄像头驱动开发以及相关应用研究有一定指导作用。

## 参考文献

- [1] 陈洋. 基于 ARM 和嵌入式 Linux 系统的多功能工业相机的设计和实现[D]. 重庆:重庆邮电大学, 2016.
- [2] 马牙川. 基于 FPGA 的智能工业相机系统的研究[D]. 杭州:浙江大学, 2016.
- [3] 蔡金. 嵌入式高清工业相机研究与实现[D]. 杭州:浙江大学, 2013.
- [4] Khan T, Wahid K. Universal bridge interface for DVP-compatible image sensors[J]. Microprocessors & Microsystems, 2011, 35(6):547-556.
- [5] 李宇成,葛正中. Linux 下高清 MIPI 差分摄像头的软硬件设计[J]. 单片机与嵌入式系统应用, 2016, 16(9): 35-40.
- [6] OmniVision Technologies. OV5645 color CMOS QSXGA image sensor with OmniBSI technology, 2012.
- [7] 宋宝华. Linux 设备驱动开发详解[M]. 北京:人民邮电大学出版社, 2010.
- [8] 王飞,孔聪. 基于 V4L2 的 Linux 摄像头驱动的实现[J]. 电子科技, 2012, 25(2):86-87,92.
- [9] 曹占中,周余,王自强,等. 基于 s3c2440 的 Linux 摄像头驱动开发[J]. 电子测量技术, 2009, 32(2):108-111.
- [10] 杨东琼,云利军,王坤. Linux 和 OK6410 环境下视频图像的采集与显示[J]. 云南师范大学学报(自然科学版), 2016, 36(2): 48-53.
- [11] 段东波,靳天玉. 基于 ARM11 的视频采集系统及二维码识别[J]. 电子测试, 2013(20):81-82.

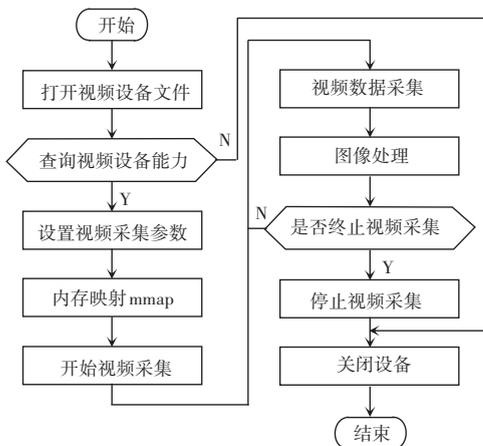


图6 视频采集流程图

Fig. 6 Video capture flow chart

## 4.2 实验结果

按照上述视频采集流程,编写应用程序,利用 arm-linux-gcc-4.5.1 编译器交叉编译后,通过 FileZilla 下载到飞凌嵌入式提供的开发板上进行实验验证。最终将采集到的图像数据压缩成 jpeg 格式的图片并保存为 IMGresult.jpg,存放到同级目录下。将实验结果上传到 Ubuntu 环境下进行查看结果。图 7 为 OV5645 抓取的图片效果图。