

# 中国科学技术大学

# 工程硕士学位论文



**基于 arm2410**  
**的远程视频采集与传输**

# 项目开发文档

--基于 arm2410 的远程视频采集与传输

# 目 录

第一章 范围.....	3
第二章 可行性分析.....	3
2.1 引言.....	3
2.2 可行性研究的前提.....	3
2.3 对现有系统的分析.....	3
2.4 对所建系统的分析.....	4
第三章 需求分析.....	4
3.1 外部接口需求.....	5
3.2 系统功能需求.....	5
3.3 软硬件需求.....	6
3.4 性能需求.....	7
3.5 安全需求.....	7
第四章 概要设计.....	7
4.1 编写目的.....	7
4.2 总体设计.....	7
第五章 详细设计.....	9
5.1 系统组织结构.....	9
5.2 视频采集算法设计.....	10
5.3 监控界面的设计.....	11
第六章 用户操作手册.....	12
第七章 测试计划.....	12
7.1 编写目的：在发布产品之前检查出系统的错误，方便以后用户的使用。12	
7.2 测试方案：.....	12

# 第 1 章 范围

本开发文档用于指导开发者为中国科技大学软件学院工程实践项目（基于 arm2410 的远程视频采集与传输）的开发过程，通过规范项目开发过程提高开发质量，降低开发和维护成本。开发者应根据本文档进行开发和编写文档。

## 第 2 章 可行性分析

### 2.1 引言

背景：随着计算机技术及网络技术的迅猛发展, 公安、 安防行业的发展趋势必然是全面数字化、 网络化。传统的模拟闭路电视监控系统有很多局限性：传输距离有限、无法联网，而且模拟视频信号数据的存储会耗费大量的存储介质（如录像带），查询取证时十分烦琐。基于个人计算机的视频监控系统终端功能较强, 但稳定性不好, 视频前端（如电压耦合元件等视频信号的采集、 压缩、 通讯）较为复杂，可靠性不高。基于嵌入式 L i n u x 视频的网络监控系统不需要用于处理模拟视频信号的个人计算机，而是把视频服务器内置一个嵌入式 We b 服务器, 采用嵌入式实时多任务操作系统。由于把视频压缩和 We b 功能集中到一个体积很小的设备内，可以直接连入局域网，即插即看，省掉复杂的电缆，安装方便（仅需设置一个 I P 地址），用户也无需安装任何硬件设备，仅用浏览器即可观看。

开发的系统名称：基于 arm2410 的远程视频采集与传输

开发者：闫家胜、左文涛、殷超、孙青、周健杏

### 2.2 可行性研究的前提

要求

功能：这是一个远程视频采集与传输系统，主要负责对监控区域的监控，视频采集与传输，并在客户端对视频进行保存方便日后查看。

目标

克服传统视频监控系统的传输距离近, 反应慢, 故障率高和成本高的缺点, 开发出新一代的基于网络的远程视频采集和传输，使之具有传输距离远, 反应快, 故障率低, 成本低, 易于安装和维护的特点。

限制条件

1. 经费不足，无法购得高品质的摄像头。
2. 硬件配备不齐。
3. 竞争激烈，市场上有许多性能优越的同类产品。

### 2.3 对现有系统的分析

#### 1. 模拟监控系统：

主要有摄像头、视频矩阵监视器、录像机等组成，有视频电缆、控制电缆等连接。监视系统以模拟方式传输，一般传输距离不能太远，主要用

于小范围内的监控，监控图像一般只能在控制中心查看。模拟监控系统的缺点是：通常只适合小范围的区域监控，系统的扩展能力差，无法形成有效的报警联动，由于各部分独立运作，相互之间的控制协议很难互通。

## **2. 数字监控系统：**

在这一代监控系统中，将各个摄像机获得的视频信号利用专门的压缩算法进行压缩，然后通过内部网络联到一个 PC 机或工业计算机上，在这台作为核心的计算机上进行视频解压，提供视频监控服务。这一代的视频监控系统较模拟视频监控系统相比，无论在图像质量、保存时间以及可靠性商均有了大幅度提高。但是视频数据的传输距离任然不远，还是达不到现在人们对视频监控的需求。所以说开发新一代远程视频采集与传输系统是相当有必要的。

## **3. 处理流程和数据流程**

派专人在监控室内进行监控，要时时刻刻地看着监控器，没有主动报警功能。

## **4. 工作负荷**

要时时刻刻看着监视器，工作量大。

## **5. 费用开支**

费用开支较大，因为通常要派好几个人轮班监视设备。

## **6. 局限性**

人难免有疲倦的时候，这样就很容易出现事故，而且只能实现近距离监控。

## **2.4 对所建系统的分析**

在监控区域进行视频采集，压缩后用网络进行传输到监控端，经过解压缩后显示视频图像。

### **1.工作负荷：**

只需一人负责即可，有异常情况会触动警报器，工作量小。

### **2. 改进之处：**

提高了监控的距离和反应时间，智能化，无需时时刻刻观看监视器。

### **3. 技术条件方面的可行性**

互联网技术都已经很成熟，基于 V4L 的视频采集技术也已经很成熟。

### **4. 成本**

摄像头：几十元

ARM2410 开发板的精简版：500 元左右

监控软件：200 元左右

总成本：加上网络传输设备共一千元左右。

### **5.受益**

现在许多领域都要用的远程视频的采集与传输，所以市场很好。我们的系统成本控制在 1000 元左右，成本相对低廉，而且性能出色稳定，所以很有竞争力。

## 第 3 章 需求分析

### 3.1 外部接口需求

#### 用户界面

提供可视化的监控界面：一个视频窗口，一个 IP 地址输入框，一个连接按钮，一个开始按钮，一个停止按钮和一个保存按钮。

#### 硬件接口

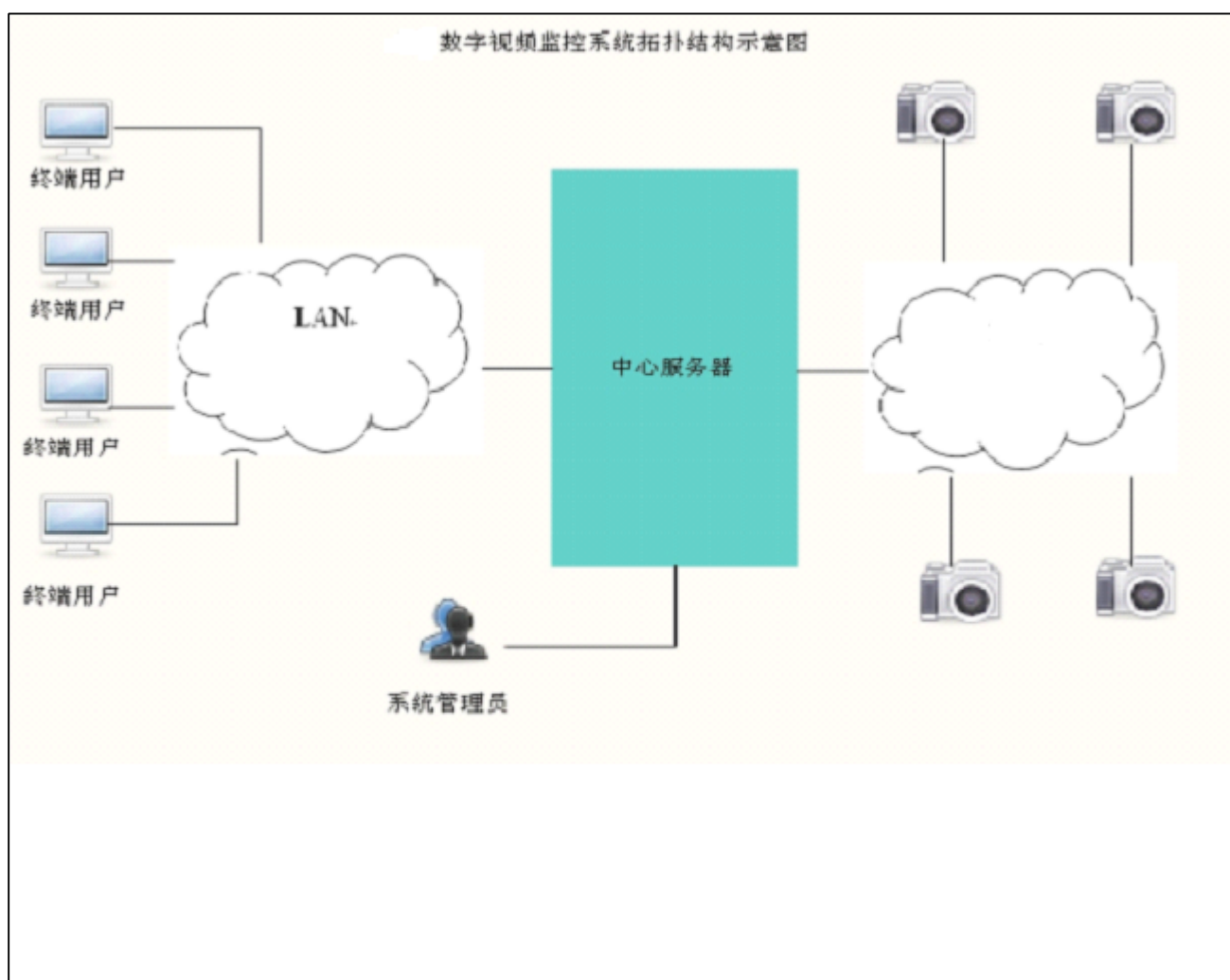
支持各种常用摄像头和常用开发板。

#### 软件接口

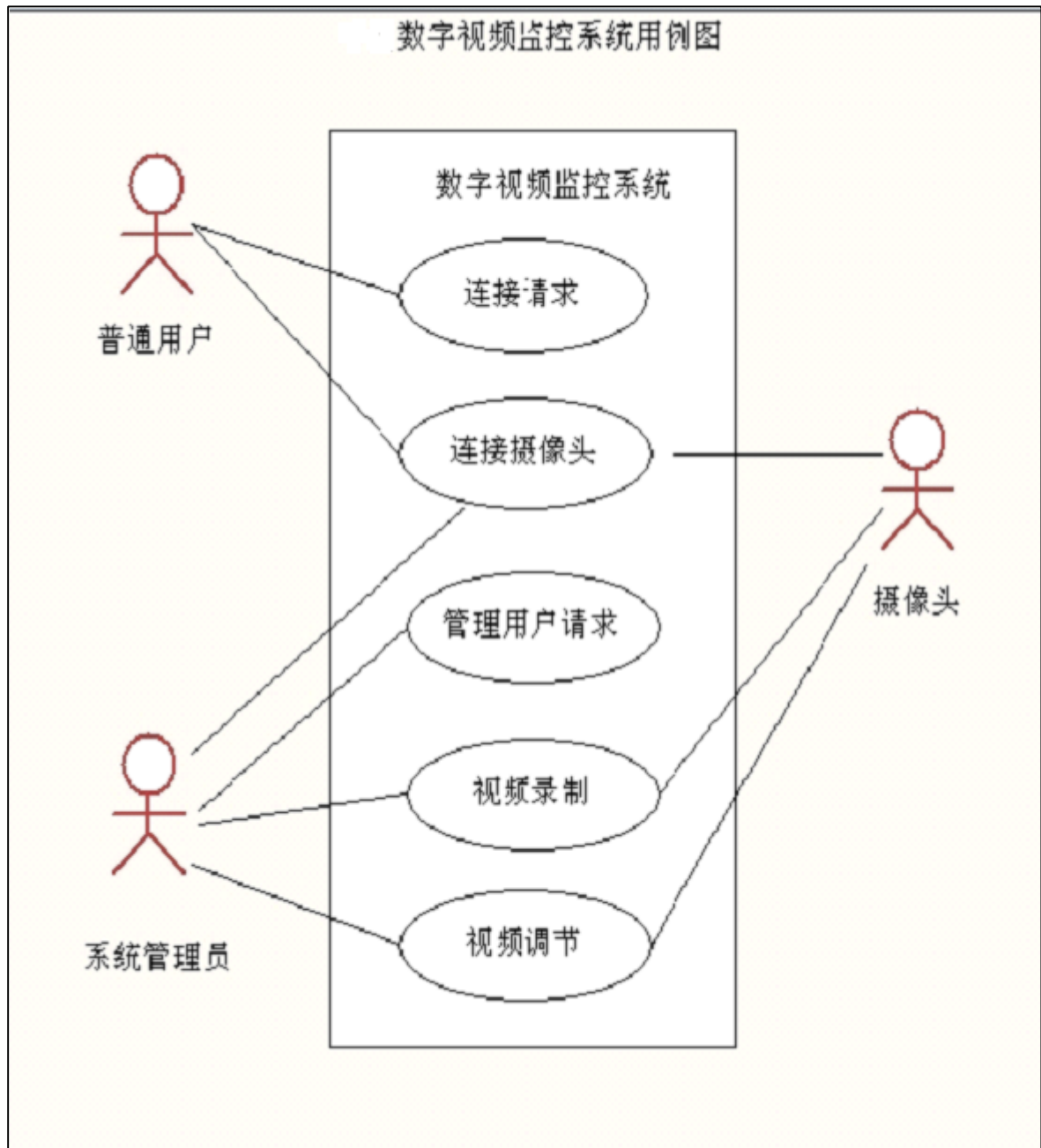
支持 Windows 和 Linux 操作系统，支持 gcc 和 c 编译器。

### 3.2 系统功能需求

#### 系统组成图



### 3.2.2 系统用例图



### 3.3 软硬件需求

#### 硬件系统

客户端：普通 PC

中心服务器:处理能力较强的服务器

监控端：ARM S3C2410 开发板+USB ov511 芯片摄像头

网络传输:双绞线

网络连接设备:二层交换机

#### 软件系统

客户端：运行在 linux 操作系统下的基于 QT3.3.8 版本开发的客户端程序  
中心服务器：运行在 linux 操作系统下的基于 QT3.3.8 版本开发的服务端程序  
ARM 摄像头服务端：运行在 linux2.6.26 内核下的 C 语言开发的服务端程序

### 3.4 性能需求

根据软件工程领域的技术水平和开发者的经验技术水平，可达到以下要求：  
监控距离：利用有线网络可实现远程和超远程监控，利用无线网络可实现几百米的健康  
响应时间：3 秒左右  
容量需求：存储器：至少 1G  
视频清晰度：可清楚地区分不同的人 and 物体

### 3.5 安全需求

可在监控端设置访问权限

## 第 4 章 概要设计

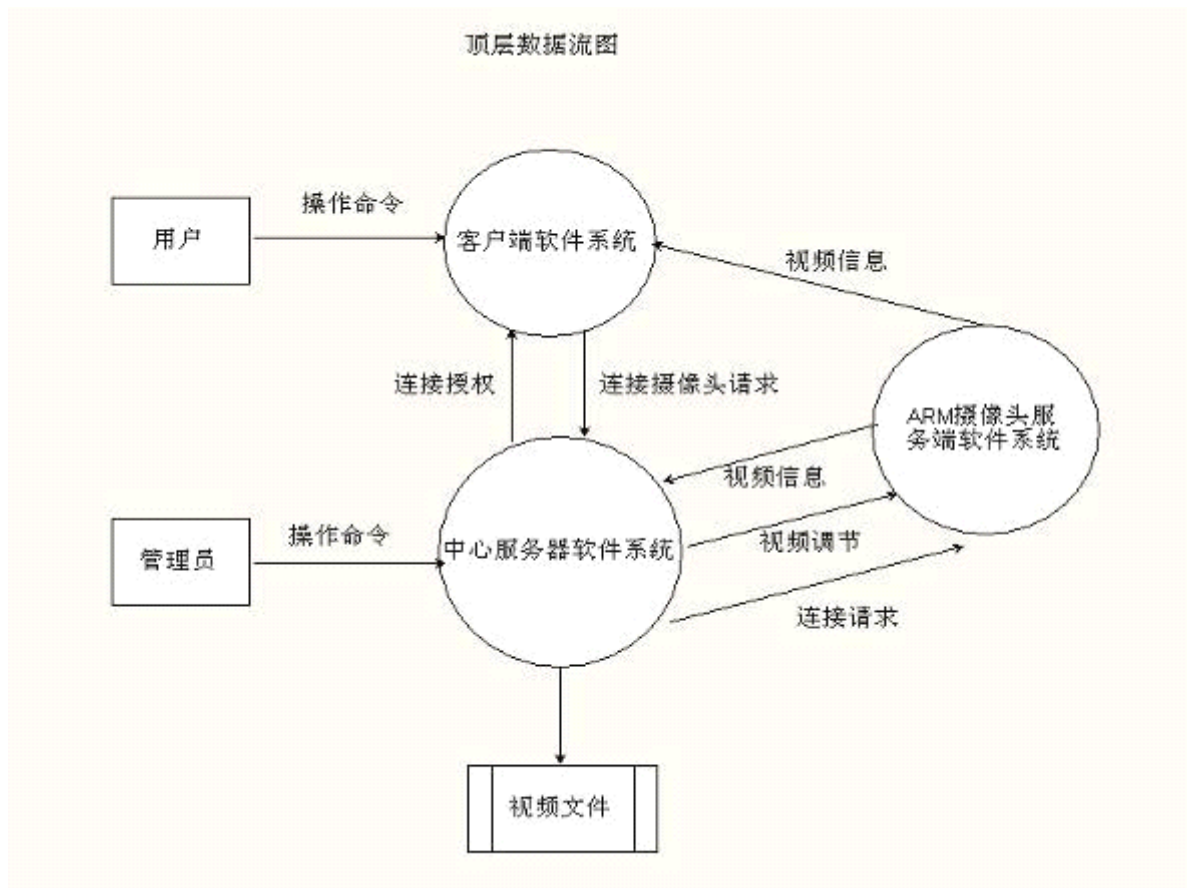
### 4.1 编写目的

从该阶段开发正式进入软件的实际开发节点，本阶段完成系统的大致设计并明确系统的数据结构。在软件设计阶段主要是把一个软件学期转化为软件表示过程，这种表示至少描绘出软件的总的概貌。本概要设计说明书的目的就是进一步细化软件设计阶段得出的软件总体概貌，把它加工成程序喜剧诌上非常接近于源程序的软件表示

### 4.2 总体设计

项目数据流图

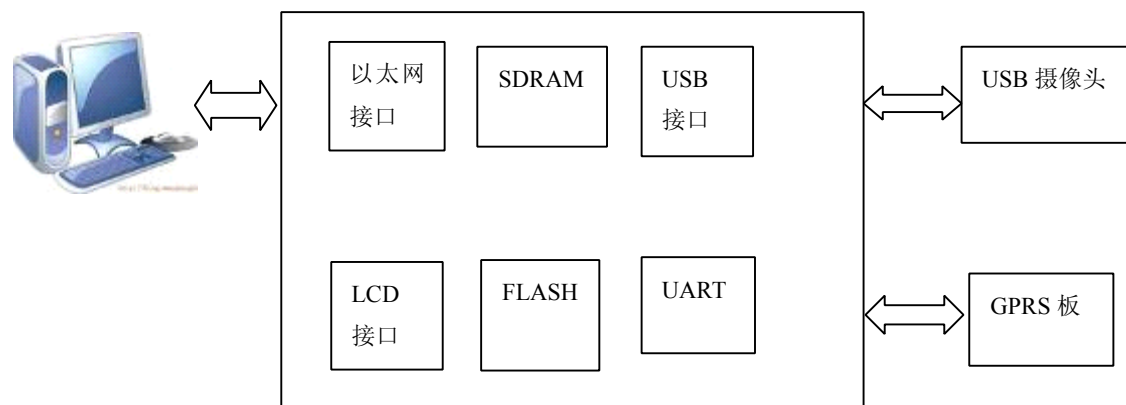




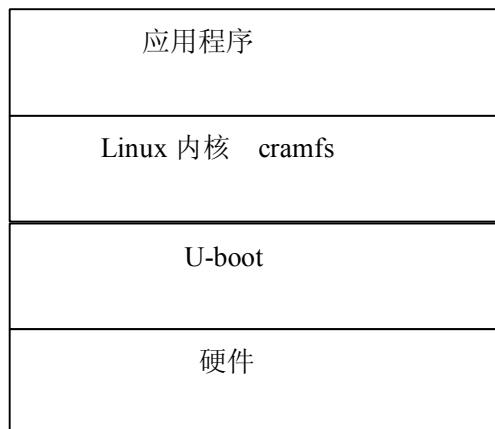
#### 4. 2. 2 项目总体框架



#### 4.2.3 硬件结构框图

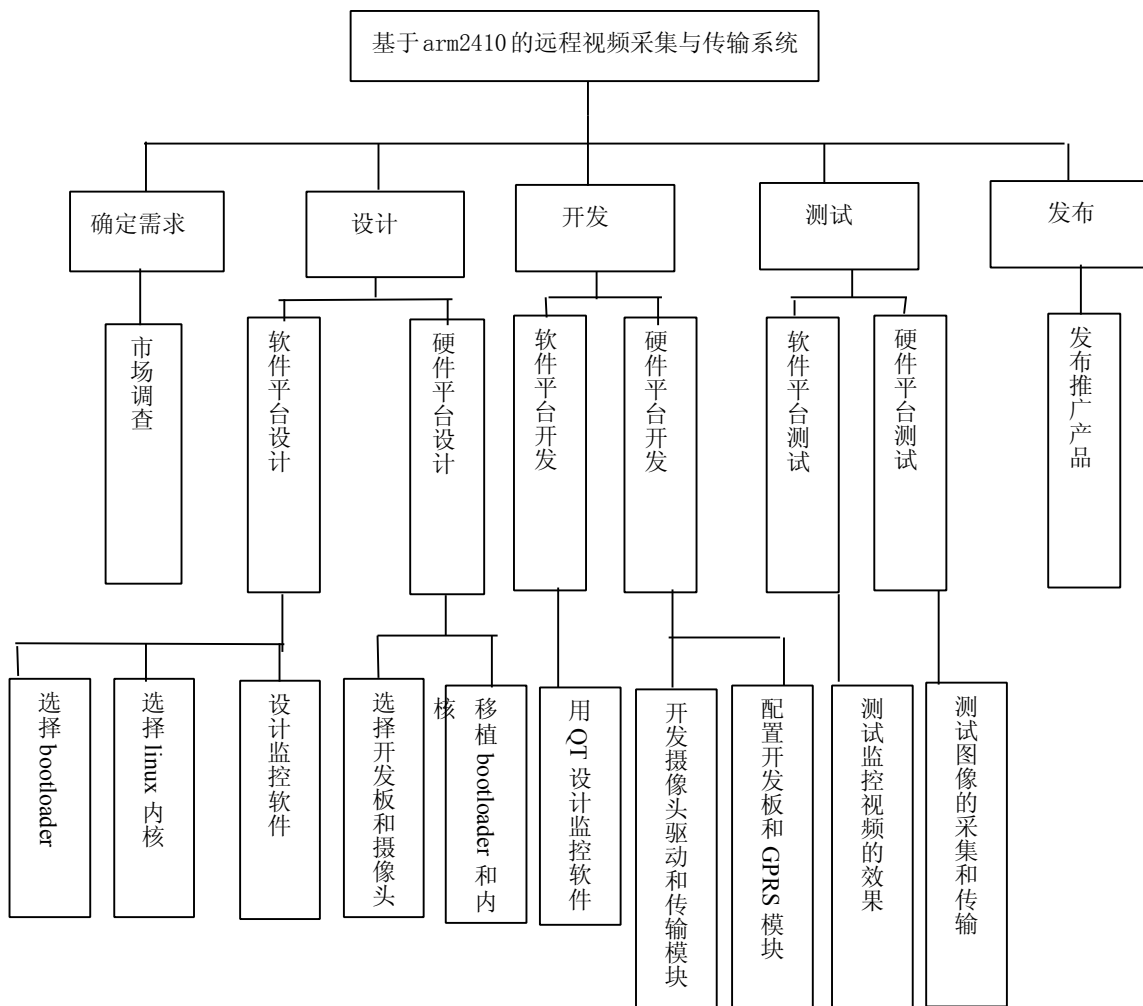


#### 4.2.4 软件系统框图



## 第5章 详细设计

### 5.1 系统组织结构



### 5.2 视频采集算法设计

采用 V4L 进行视频采集。

Video4Linux 是 Linux 下用于获取视频和音频数据的 API 接口，可以实现获取摄像头数据，以实现连续影像的播放。

#### 1. 摄像头的安装

在 Linux 下常用的摄像头驱动是 spca5xx，这是一个通用驱动，读者可以在以下网站下到这个驱动

<http://mxhaard.free.fr/download.html>。这个网站还给出了这款驱动支持的摄

像头的种类。另外，ov511 芯片直接就支持 Linux，使用者款芯片的摄像头有网眼 V2000。我使用的是网眼 V2000 的摄像头，和 Z-Star

301p+现代 7131R 芯片的摄像头。后一种需要 spca5xx 的驱动。关于 spca5xx 的安装方法，网上有很多介绍，这里就不说了。

## 2. 摄像头的调试

安装好摄像头后，为了测试摄像头能否正常工作，可以用一下软件。比较著名的是 xawtv，在网上搜以下可以下载到。安装好后，打开 xawtv 则可以调试摄像头。

## 3. Video4Linux 编程获取数据

现有的 video4linux 有两个版本，v4l 和 v4l2。本文主要是关于 v4l 的编程。利用 v4l API 获取视频图像一般有以下几步：

打开设备

设置设备的属性，比如图像的亮度，对比度等等

设定传输格式和传输方式

开始传输数据，一般是一个循环，用以连续的传输数据

关闭设备

下面具体介绍 v4l 编程的过程。首先指出，在 video4linux 编程时要包含头文件，其中包含了 video4linux 的数据结构和函数定义。

### 1) v4l 的数据结构

在 video4linux API 中定义了如下数据结构，详细的数据结构定义可以参考 v4l API 的文档，这里就编程中经常使用的数据结构作出说明。

首先我们定义一个描述设备的数据结构，它包含了 v4l 中定义的所有数据结构：

```
typedef struct _v4ldevice
{
    int fd;//设备号
    struct video_capability capability;
    struct video_channel channel[10];
    struct video_picture picture;
    struct video_clip clip;
    struct video_window window;
    struct video_capture capture;
    struct video_buffer buffer;
    struct video_mmap mmap;
    struct video_mbuf mbuf;
    struct video_unit unit;
    unsigned char *map;//mmap 方式获取数据时，数据的首地址
    pthread_mutex_t mutex;
    int frame;
    int framestat[2];
    int overlay;
}v4ldevice;
```

下面解释上面这个数据结构中包含的数据结构，这些结构的定义都在中。

```

struct video_capability
name[32] Canonical name for this interface
type Type of interface
channels Number of radio/tv channels if appropriate
audios Number of audio devices if appropriate
maxwidth Maximum capture width in pixels
maxheight Maximum capture height in pixels
minwidth Minimum capture width in pixels
minheight Minimum capture height in pixels

```

这一个数据结构是包含了摄像头的属性，name 是摄像头的名字，maxwidth maxheight 是摄像头所能获取的最大图像大小，用像素作单位。

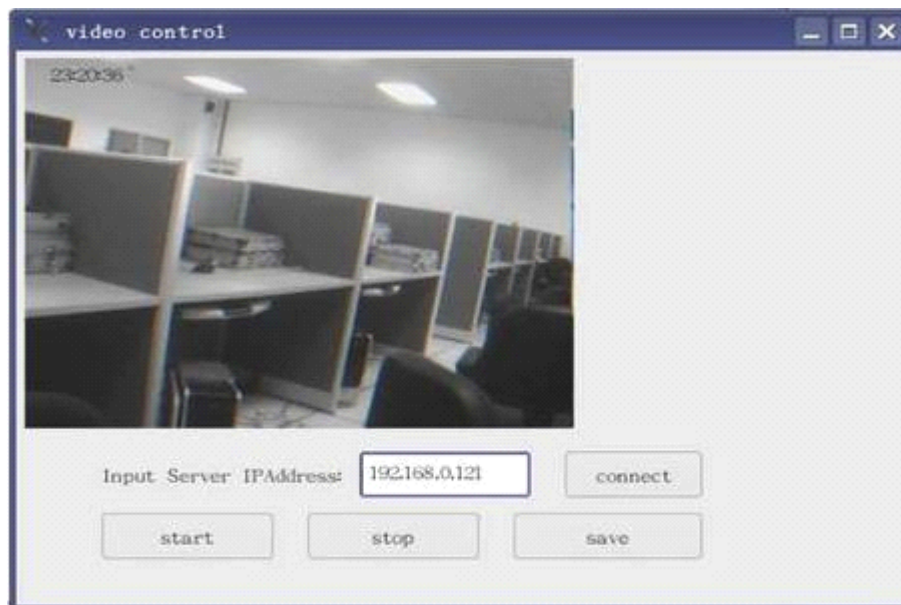
在程序中，通过 ioctl 函数的 VIDIOCGCAP 控制命令读写设备通道已获取这个结构，有关 ioctl 的使用，比较复杂，这里就不说了。下面列出获取这一数据结构的代码：

```

int v4lgetcapability(v4ldevice *vd)
{
    if(ioctl(vd->fd, VIDIOCGCAP, &(vd->capability)) fd, VIDIOCGMBUF,
    &(vd->mbuf))map + vd->mbuf.offsets[vd->frame]);
}

```

## 5.3 监控界面的设计



# 第 6 章 用户操作手册

## 6.1 编写目的

为用户提供使用该系统的方法，规范用户的操作，使用户可以很容易地使用该系统。

## 6.2 软件使用概述

功能：提供监控界面及对视频图像的调整

操作方式：在 IP 地址输入框中输入所以监控区域摄像头的地址，然后单击链接和开始 按钮就可以看到视频图像。

## 6.3 运行环境

硬件：ARM2410 开发板，PC 机（有至少 1G 硬盘），摄像头

软件：windows 和 Linux 操作系统，QT 开发工具

## 6.4 使用说明

安装摄像头：将符合需要的摄像头安装在监控区域

配置：配置摄像头设备的地址等基本配置

运行监控软件：打开 QT 界面，输入 IP 地址，进行链接，观看视频

# 第 7 章 测试计划

## 7.1 编写目的：

在发布产品之前检查出系统的错误，方便以后用户的使用。

## 7.2 测试方案：

测试方法：采用黑盒测试

测试手段：采用手工测试

功能、数据流及边界测试

测试用例如下：

测试点	前提条件	输入与操作	期望结果	测试结果
IP 地址	4 个数字和点号	172.168.130.11	观察到视频	观察到视频
IP 地址	4 个数字和点号	172.168.130.15	观察到视频	未观察到视频

反应速度	输入合法	172.168.130.11, 快速移动摄像头	图像流畅	有轻微抖动和暂停
环境测试	输入合法	172.168.130.11, 将摄像头置于明亮环境中	图像清晰	图像清晰
环境测试	输入合法	172.168.130.11, 将摄像头置于昏暗环境中	图像清晰	图像昏暗
存储测试	输入合法	172.168.130.11, 保存监控视频, 然后查看保存的视频	能清晰观看到之前监控的视频	成功完成视频保存于重放