# 2相关技术简介

本章节主要介绍本系统所涉及的主要技术，如Python、SQLite3、Redis、HTML、Django框架以及Raspberry Pi的GPIO编程、Ubuntu系统以及Lirc。

# 2.1 Python介绍

### 2.1.1 Python简介

Python是一种简洁并且能简单上手但其实不简单的解释性语言，大部分人也叫它为脚本语言，它没有编译这个环节，类似于PHP和Perl语言。

使用Python语言对一个函数的设计，或者是作为面向对象的设计，它都具有非常强的可读性，相比于类似Java、C++等语言中一些英文关键字的设计，Python语言以其独特的标点符号、命名方式、对齐方式都为人所赞美。尽管Python语言运算速度比不上C，但是其依靠各种强大的内置库和丰富的第三方库都让Python变得越来越火热，尤其是现在机器学习库的发布，更是让Python隐隐约约有超过Java使用量的趋势。

### 2.1.2 Python特点

Python的主要特点是：上手简单、方便维护和阅读、丰富的三方库以及跨平台兼容、各种数据库支持。

# 2.2 SQLite3介绍

SQLite3是一种非常轻的但是支持标准SQL语句的数据库。Sqlite3与Mysql等主流数据库一样，也支持事务操作，但是却没有像其他数据库一样连接繁琐，它是不需要依赖服务器存在的、不需要进行繁琐配置的一种实现了自给自足的轻量级数据库。

SQLite3不需要配置意味着它根本不需要安装，无论是Linux系统还是Windows系统，都自带了SQLite3的环境，并且如果要用Python使用SQLite3，只需要安装SQLite3的Python库，即可简单使用API进行事务的ACID操作。

# 2.3 Redis介绍

Redis是一种非关系性的数据库，但是Redis的主要目的并不是作为数据库使用。Redis是基于内存的，读写速度非常快，因此Redis一般用于做缓存处理，并且Redis是key-value的，所以不管是在线程中或者是不同的会话中，都能获取到设置的值，这样就可以用来做在线和离线的一种交互。

Redis支持设置一个key的过期时间，当时间到了之后，Redis会自动删除该key-value对，这样可以用来做一种验证码的功能。

同时，Redis的CRDU是原子性的，不会发生线程安全问题。

Redis拥有丰富的数据类型操作，如String、List、Set、Hash、Sorted Set这五种，其中Sorted Set是一个没有重复并且每个元素带有一个Score的数据类型，可以很方便的做一个热榜排行功能。

# 2.4 HTML介绍

Hyper Text Markup Language，简称HTML，它指的是超文本标记语言。HTML是一种标记语言，而不是一种编程语言，它使用一种标记标签来描述网页。

Cascading Style Sheets，简称CSS，它值的是层叠样式表，当我们编写了HTML代码时，我们会使用CSS来美化我们的页面。

JavaScript，简称js，它是一种可以嵌入HTML代码中的轻量级编程语言，由用户的浏览器代为执行。

# 2.5 Django框架介绍

Django是一个高级Python Web框架，鼓励快速开发和干净实用的设计。由经验丰富的开发人员构建，它可以处理Web开发的大部分麻烦，因此您可以专注于编写应用程序，而无需重新发明轮子。它是免费且开源的。、

**快速：**Django旨在快速的为开发人员搭建一个WEB应用程序。

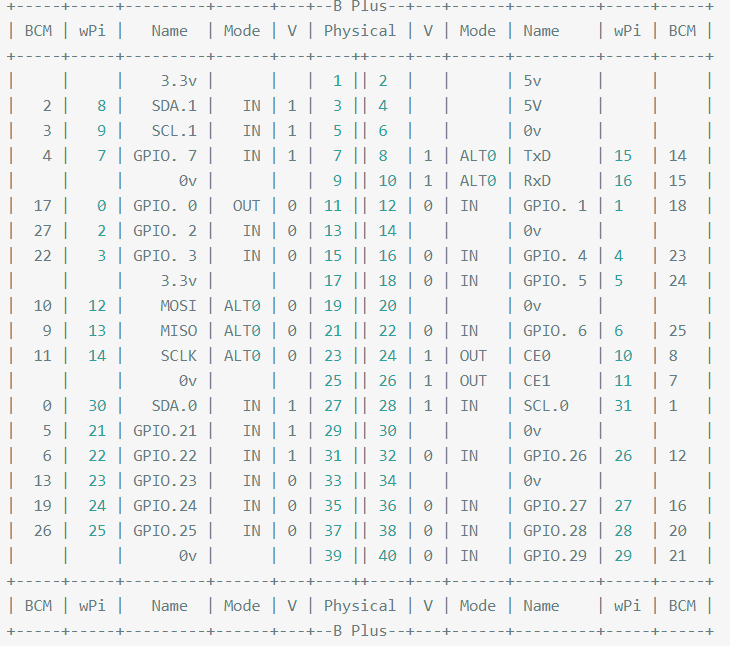
**安全：**防止SQL注入，帮助开发人员避免很多常见的安全错误。

**实时性：**可以在服务运行时修改代码，并自动热部署。

# 2.6 Raspberry Pi的GPIO编程

Raspberry Pi是一种卡片式电脑，虽然小，但是功能非常多。Raspberry Pi有40个引脚，其中有两个5V引脚、两个3.3V引脚、8个GND引脚以及28个可以控制的GPIO引脚。树莓派实物图如图2-5-1所示，引脚对照表如图2-5-2所示。

**图2-5-1 树莓派实物图**



**图2-5-2 树莓派引脚对照**

Raspberry Pi的引脚工作方式有两种，一种是Board，这种工作方式将使用板子编号将引脚接入电路，不用考虑不同Raspberry Pi不同版本中每个引脚代表不同功能，只与板子中的引脚编号有关；一种是采用BCM工作方式，当采用BCM方式时，接入的引脚必须是BCM编号对应的Board编号。

程序可以对可使用的GPIO口设置成输入引脚或者输出引脚。当一个引脚被设置成输入引脚时，可以通过硬件上拉电阻来解决输入为float类型，也可以通过软件设置上拉状态。当一个引脚被设置成输出时，可以通过代码控制输出高低电平，同时也可以通过软件来读取此时输出状态是什么。

Raspberry Pi的引脚设置为输入时，可以在代码中为该引脚注册一个中断，当电平变化时触发中断，执行相应函数。

# 2.7 强大的Lirc

LIRC是一个包，允许用户解码和发送许多（但不是所有）常用遥控器的红外信号。

最近的Linux内核可以将一些红外遥控器用作常规输入设备。有时候这会让LIRC变得多余。但是，LIRC提供了更多的灵活性和功能，并且在许多场景中仍然是正确的工具。

LIRC最重要的部分是lircd守护进程，用于解码设备驱动程序接收到的IR信号，并在套接字上提供信息。如果硬件支持，它还接受要发送的IR信号的命令。

用户空间应用程序允许您使用遥控器控制电脑。只需按一下按钮，即可将X11事件发送到应用程序，启动程序等等。可能的应用是显而易见的：红外鼠标，电视调谐卡或CD-ROM遥控器，远程关机，用计算机编程VCR和/或卫星调谐器等。在Raspberry Pi上使用lirc很受欢迎。

### 2.7.1 安装Lirc

在Raspberry Pi下安装Lirc是非常简单的，只需要在终端执行一下命令就好了。

sudo apt-get install lirc

安装完之后执行命令：

sudo leafpad /etc/lirc/hardware.conf

修改下面几处：

LIRCD\_ATGS=""

DRIVER="default"

DEVICE="/dev/lirc0"

MODULES="lirc-rpi"

### 2.7.2 配置Lirc引脚

Lirc需要配合红外接收器和红外发射模块使用的，因此我们需要在Raspberry Pi中配置红外拓展板中红外接收和发送引脚。

首先打开树莓派模块文件：

sudo leafpad /etc/modules

添加如下内容：

lirc-dev

lirc-rpi gpio\_in\_pin=18 gpio\_out\_pin=17

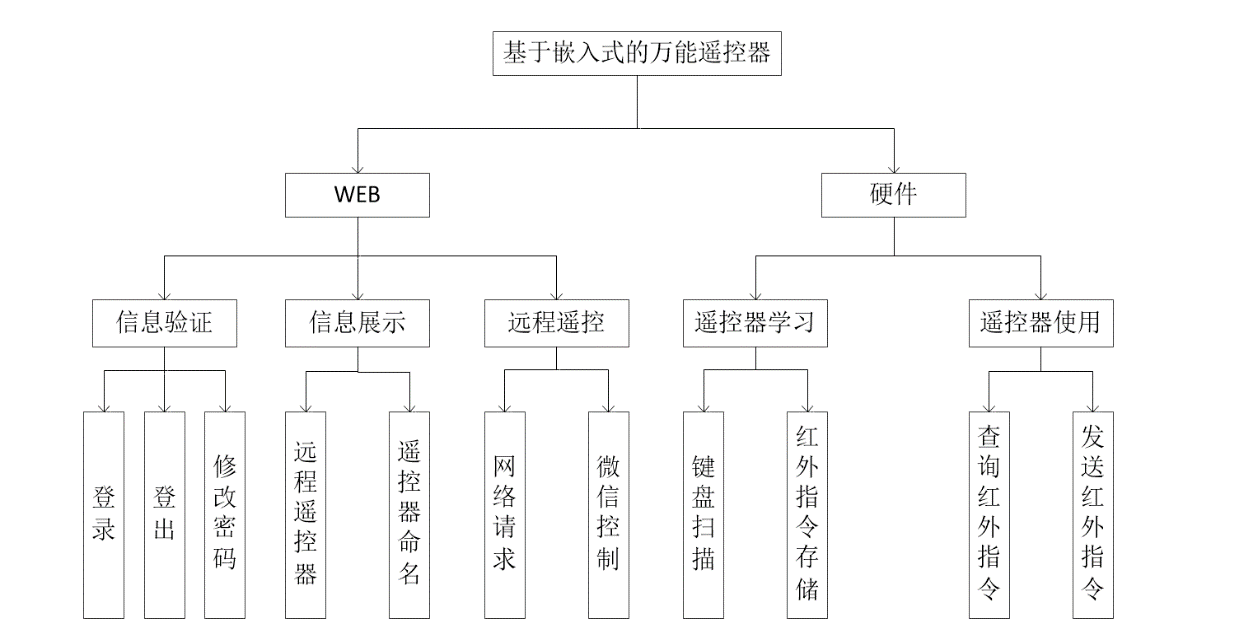
# 采用的是BCM 模式

保存配置后，重启Raspberry Pi使得配置生效，至此Lirc就可以使用了。

# 3设计方案与功能介绍

本章节主要是从大范围的讲述整个系统的基本情况，具体的内容将在后面章节中讲述。

# 3.1 整体架构

基于嵌入式的万能遥控器分为硬件部分和软件部分，其中软件部分有：信息验证模块（登录模块、登出模块、修改密码模块）、信息展示模块（远程遥控器模块、学习完后的遥控器在WEB端命名模块）以及远程遥控模块（通过RESTful API进行网络请求来远程遥控模块、微信接入发送指令遥控模块），硬件部分有：遥控器学习模块（键盘扫描模块、红外指令存储模块）以及遥控器的使用模块（查询当前遥控器属于哪个设备模块、发送红外指令模块）。其整体架构如图3-1所示。

**图3-1 基于嵌入式的万能遥控器整体架构图**

远程遥控系统通过登录模块进行用户名密码验证，验证通过后将可进入系统，进入系统后有两种方式可以控制远程设备。一是通过网页样式设计的一个虚拟遥控器，每一个按键会触发一个RESTful API请求，后台会将对应的红外编码发出去；二是接入微信，在微信中根据提示回复相应的命令，后台将根据不同命令来对不同的设备发送自己想要的指令。在开发板中，编写了键盘扫描程序，不同的按键代表了不同的信息，用户按下学习键即可开始学习一台新的设备。

# 3.2 整体功能

基于嵌入式的万能遥控器的难点有一下几个：1、如何接收红外信号和发送红外信号？这个问题将使用lirc和红外接收发送扩展板解决；2、如何保存学习到的按键？后面将使用SQLite3作为整个系统的数据库，以用来存储；3、如何远程遥控？这个问题将使用Django来搭建一个web服务。更为具体的功能如下。

### 3.2.1 登录功能

用户填写用户名和密码，点击登录后台接收到用户名和密码后进行判断，如果存在并密码正确即登录成功进入系统，否则登录失败。

### 3.2.2 忘记密码功能

当用户忘记密码时，可以进入忘记密码页面，输入邮箱并获取验证码，当输入验证码正确时可以进行密码重置。

### 3.2.3 我的家电功能

在这里将可以查看所有学习过的家电，并可以删除和使用家电。

### 3.2.4 查看未命名家电功能

当在硬件部分学习完一个新的设备后，这个新的设备将在这里出现，但是设备名字是一串特殊的字符，我们需要为这个设备重新命名才能使用它。

### 3.2.5 微信功能

登录微信后可以进行机器人对话也可以远程遥控设备。

### 3.2.6 学习功能

通过矩阵键盘作为按键，当按下学习键后可以开始学习，并通过lirc进行红外的解码。

### 3.2.7 遥控器功能

开发板在每一次学习结束后都是一个新的遥控器，可以直接使用来遥控刚学习的设备。

# 3.3 设计原则

**低功耗设计：**整个系统基于Raspberry Pi开发板，仅需USB供电，并且其余器件也是选用的低且可控功耗的器件。

**模块化设计：**整个系统是分模块设计的，每个模块单独实现功能再整合成完整系统。每个模块下面又分了很多小功能，每个小功能也尽量是实现高内聚低耦合方式，所以不会产生重复开发的情况，让得系统更简洁、更高效。

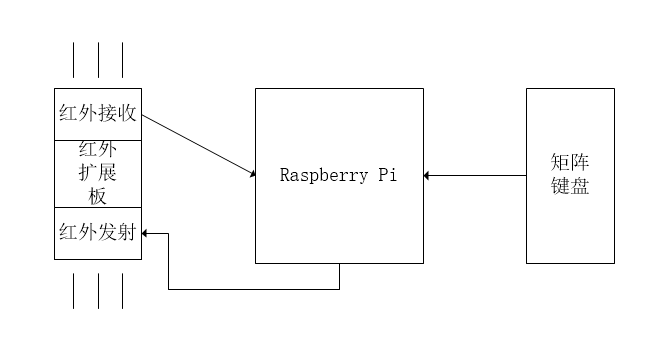
**能优化则优化：**整个系统开发过程中，不管是软件还是硬件，对于能优化的部分都进行了优化，比如说开发中硬件的绝大部分都是采用杜邦线连接，而不是在PCB板子中焊接，减少了电路与电路之间的干扰，提高了抗干扰能力。

# 4 硬件设计

此章节将详细的描述整个硬件的部分，包括整个硬件电路的设计、红外模块电路的设计、按键模块的设计以及指示灯模块电路设计。

# 4.1 硬件整体方案

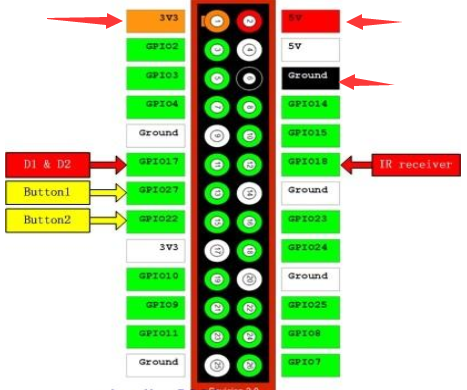
硬件是分模块设计的，但是每个模块之间也有自己的关联。系统以Raspberry Pi为主控，用2个4\*4的矩阵键盘作为遥控键值的输入，以红外拓展板作为红外信号的接收和发射，Raspberry Pi的GPIO编程作为整个业务的逻辑判断，设计图如图4-1所示。

**图4-1 硬件设计图**

# 4.2 红外拓展板与Raspberry Pi

这个红外拓展板是针对Raspberry Pi的专用红外控制拓展板，用户可以使用该拓展板配合Raspberry Pi完成各种遥控和被遥控的功能。红外拓展板使用5v及3.3v配合供电，红外接收引脚连Raspberry Pi 的GPIO18（BCM），红外发射引脚接Raspberry Pi的GPIO17（BCN）。红外拓展实物图和红外拓展与Raspberry Pi引脚对应图分别如图4-2-1、图4-2-2所示。

**图****4-2-1红外拓展实物图**

**图4-2-2 红外拓展板与树莓派对应的引脚**

# 4.3 Raspberry Pi和2个4\*4键盘

基于嵌入式的万能遥控器一共设计了25个按键，如果直接焊接25个按钮并使用25个引脚作为输入的话，那么不仅代码工作量要增加很多，并且大多数还是一些重复性的代码，还会导致Raspberry Pi的GPIO也将占用多达25个，这样既不好管理，也让代码冗余，所以选择了2个4\*4的矩阵键盘。

