

**毕 业 设 计（论 文）**

基于单片机的太阳能面板自动转向

控制器设计

学生姓名： 王睿

学 号： 20201014135

所在系部： 机械工程系

专业班级： 20gb机电一班

指导教师： 张敏

日 期： 二O二四年五月

**Microcontroller-based solar panel**

**auto-steering**

**Controller Design**

By

Rui Wang

May 2024

# 

**毕业设计（论文）任务书**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 系 部 | 机械工程系 | 指导教师 | 张敏 | 职 称 | 讲师 |
| 学生姓名 | 王睿 | 专业班级 | 20gb机电1班 | 学 号 | 20201014135 |
| 论文题目 | 基于单片机的太阳能面板自动转向控制器设计 | | | | |
| 论  文  内  容  目  标  及  进  度  要  求 | **一．主要内容与目标**  结合所学专业，查阅相关资料，深入了解单片机控制技术。完成太阳能自动转向控制系统的设计，解决主流控制器自由度低、控制困难、体积大等问题。完成系统的方案设计、器件选型、和实际模拟。  在进行上述学习、研究、设计的基础上，依据《湖北工业大学工程技术学院毕业设计(论文)规范(理工类专业适用)》完成毕业论文撰写。  **二.课题的进度安排**  第一阶段（2022年2月20日—2023年2月30日）：撰写并完成开题报告，初步确定论文的框架和大致研究内容。  第二阶段（2023年2月25日—2023年3月1日）：完成中、外资料的查找和翻译工作，并记录和收集相关系统设计的资料。  第三阶段（2023年3月1日—2023年4月10日）：完成系统分析、设计、以及开发，并调试完成。  第四阶段（2023年4月11日—2023年5月5日）：撰写、修订毕业论文，准备论文答辩。  指导教师签名：  2024年2月18日 | | | | |
| 系 部  审 核 | 同意 | | | | |

此表由指导教师填写，由所在系部审核。

**毕业设计（论文）学生开题报告**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 课题名称 | 基于单片机的太阳能面板自动转向控制器设计 | | | | |
| 课题类型 | 实践应用型 | | | 指导教师 | 张敏 |
| 学生姓名 | 王睿 | 学 号 | 20201014135 | 专业班级 | 20gb机电1班 |
| **本课题的研究现状、研究目的及意义**  随着不可再生能用的日益短缺，越来越多的人开始使用可再生能源进行发电，多数人对于新能源类别选择也有一定的要求。根据我查阅资料，和对身边使用新能源的人的观察和询问，发现大多数人都是使用太阳能发电板进行发电的，但是传统的太阳能发电板存在着体积巨大光纤利用收集率低的问题。其中便有部分人为了解决这个问题，选择了可以移动的太阳能收集面板，方便的同时也提高了能源利用效率。但我觉得还能让太阳能发电更“高效”一点——自动转向。  目前常见太阳能发电装置以下几种：固定式太阳能发电装置和单轴追踪、双轴追踪、光电追踪。传统固定式太阳能收集器安装简单但是占用空间巨大，且太阳辐射收集利用效率低，渐被市场淘汰；双轴追踪就是指在二维表面上旋转运动的追踪系统，它可以同时在两个相互垂直的方向上追踪太阳的位置，目前市面常见的太阳能手机产品产品基本上都是双轴追踪，但是都是通过视日运动或者 GPS去预判太阳轨迹从而提前转向跟踪。在我看来它们存在以下缺点：  1.单轴跟踪方式虽然结构简单易于控制，但由于其追踪自由度低，导致在单位时间内与大多数光线都无法达到最大吸收的角度，太阳辐射的利用率仍然低下  2.目前的双轴跟踪大多需要配合以GPS以定位太阳运动或者进行视日轨迹预测，程序结构都叫较为复杂，且对于跟踪装置的安装位置有一定要求，否则会因为累计误差而影响到精确度，稳定性和精度较差。  所以在此提出一款基于单片机的太阳能面板自动转向控制器设计，此装置以stm32单片机为核心，在太阳能板四角使用四个光敏电阻测量光照强度，通过伺服电机进行驱动使转向的稳定性和自由度得到保障。并且添加光照平均值对比的功能，进而判断最强光源方向，使太阳能板时刻保持面向最强太阳辐射方向。 | | | | | |

课题类型： A-理论探究型；B-实践应用型。

|  |
| --- |
| **本课题的研究内容**  基于单片机的太阳能面板自动转向控制器具体设计实现内容描述如下：  （1）系统的组成情况：STM32F401ccu6控制模块，伺服电机，光电传感器，转向云台  （2）系统实现的主要功能：1通过读读取光传感器的数据，计算太阳的方位角，通过云台调整太阳能板位置，使其正对太阳方位，达到最大发电功率。  （3）预计主要性能指标：   太阳能板输出电压： 12V   太阳能板输出功率： 30W   设备自身功耗（静态）1.5W   设备自身功耗（动态）9W   设备输出：5V1A; 220v,0.5A   1. 开发平台：标准嵌入式开发环境，KEIL+STM32CUBE, 搭配ST官方代码自动生产软件，可极大提高代码编写效率 2. 主要的功能模块包括：云台控制板，太阳能充电控制板，太阳能追光系统，云台控制系统； |
| **本课题研究的实施方案、进度安排**  第一阶段（1月1日-3月20日）：完成开题报告，初步确定论文的框架和大致研究内容。  第二阶段（3月20日-5月30日）：完成中、外资料的查找和翻译工作，并记录  第三阶段（5月30日-6月10日：完成系统分析、设计、以及开发，并调试完成。  第四阶段（6月10日-6月15日）：撰写、修订毕业论文，准备论文答辩。 |

|  |
| --- |
| 已查阅的主要参考文献（参照《信息与文献参考文献著录规则》（GB/T 7714-2015））   1. 姜楠. 基于单片机的太阳能双轴追踪系统开发与研究[D]. 景德镇陶瓷大学,2023. 2. 王博林. 太阳能电池板双轴追踪控制系统的研究[D]. 黑龙江:东北农业大学,2016. DOI:10.7666/d.Y3022348. 3. 许方斌. 双轴追踪太阳能光热发电系统镜架结构选型及受力性能研究[D]. 北京:北京交通大学,2012. DOI:10.7666/d.Y2222620. 4. 曾利霞. 基于视日运动轨迹的双轴太阳跟踪系统的研究[D]. 湖北:湖北工业大学,2012. 5. 王海军. 基于阴晴判断的混合双轴太阳跟踪控制系统[D]. 湖北:武汉理工大学,2012. DOI:10.7666/d.y2098424. 6. 尚凯林. 海上移动光伏太阳能追光与功率跟踪控制研究[D]. 湖北:武汉理工大学,2020. 7. 孙晓宁. 基于单片机的太阳光自动追踪系统研究[D]. 河北:河北大学,2015. DOI:10.7666/d.D723762. 8. Hayat, Muhammad Badar,Ali, Danish,Monyake, Keitumetse Cathrine,等.Solar energy-A look into power generation, challenges, and a solar-powered future[J].International journal of energy research.2019,43(3).1049-1067.DOI:10.1002/er.4252 . 9. Lv, Yuexia,Si, Pengfei,Rong, Xiangyang,等.Determination of optimum tilt angle and orientation for solar collectors based on effective solar heat collection.[J].Applied energy.2018.21911-19.DOI:10.1016/j.apenergy.2018.03.014 . 10. LI TINGTING, GUARNIERI MICHAEL T., et al. Synthetic microbial communities of heterotrophs and phototrophs facilitate sustainable growth[J]. Nature Communications,2020,11(1). DOI:10.1038/s41467-020-17612-8. |
| **指导教师意见**  审核通过，注意存档，按照任务书及开题报告，抓紧进行论文写作。  指导教师签名：  2023年2月24日 |

**学位论文原创性声明**

本人郑重声明：所呈交的学位论文是本人在导师的指导下独立进行研究所取得的研究成果。除了文中特别加以标注引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写的成果作品。本人完全意识到本声明的法律后果由本人承担。

作者签名： 2023年05月31日

**学位论文版权使用授权书**

本学位论文作者完全了解学院有关保管、使用学位论文的规定，同意学院保留并向有关学位论文管理部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅。本人授权省级优秀学士学位论文评选机构将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

本学位论文属于

1、保密 □，在 年解密后适用本授权书。

2、不保密 ☑。

（请在以上相应方框内打“√”）

作者签名： 2024年05月31日

导师签名： 年 月 日

# 摘 要

太阳能面板常用于光伏发电，以满足传统化石能源短缺、环境保护的需求。当前市场上的太阳能面板主要面向私人民用，存在价格高、功能过于复杂、体积大的缺点，本文基于对个人用户对太阳能发电实际需求，研究设计了一种基于基于单片机的太阳能面板自动转向控制器。

该系统通过单片机读取安装在太阳能板的四角光电传感器数据，经过数据的处理和分析，预测太阳所处的方位角，进而分析计算装置的太阳能板应该到达的位置，反推计算每个电机应该转动的角度，最后向电机驱动器发送指令，让电机转动到对应位置，从而实现对太阳位置的跟踪。该系统可以使得太阳能板一直处于最佳位置，最大限度地利用太阳能，相较于传统太阳能发电装置更加高效，增加了太阳能发电效益，光电追踪的形式降低了硬件成本，具有很高的实用性。

关键词： 光电追踪，发电效率，自动转向

# ABSTRACT

Remote power-on cards are usually used to remotely power on computers, servers, or other network devices to meet the needs of remote management, monitoring, and maintenance. Current remote power-on cards on the market are mainly aimed at small and medium-sized enterprises, with the disadvantages of high prices, overly complex functions, and large sizes. This article introduces a design and development method for a computer remote power-on and environmental temperature and humidity measurement system based on ESP8266 to meet the actual needs of remote power on and off for personal users.

This system uses the WiFi module of the ESP8266 module to achieve data transmission, and the mobile App can conveniently display and control it. The paper elaborates on the component selection, circuit design, lower computer control software, cloud platform server-side program, and mobile App design implementation methods, and produces a test prototype. The system can not only remotely power on and off, but also remotely monitor the operating environment and internal temperature and humidity of the computer case, improving the safety of computer use in unmanned environments after remote power-on, and achieving the design goal of reducing space, low power consumption, and low cost. It compensates for the shortcomings of mainstream power-on cards in the personal user usage process and has practicality and promotion value.

**Key words：**remote boot; ESP8266; temperature and humidity monitori;DHT11

**目 录**

[摘 要 Ⅰ](#_Toc4927)

[ABSTRACT Ⅱ](#_Toc494)

[1 引言 1](#_Toc16284)

[1.1 前言 1](#_Toc28181)

[1.2 市面主流太阳能支架的缺点 1](#_Toc24078)

[1.3 本课题的研究内容 2](#_Toc17956)

[2 系统的总体设计 3](#_Toc10063)

[2.1 系统的总体架构 3](#_Toc13115)

[2.2 系统实现的主要功能 4](#_Toc19997)

[2.3 系统运作的流程 4](#_Toc6840)

[3 系统机械结构与硬件设计 6](#_Toc4495)

[3.1 结构设计 6](#_Toc14784)

[3.1.1 二轴云台 6](#_Toc29438)

[3.1.2 太阳能板固定夹 8](#_Toc27139)

[3.1.3 底座 9](#_Toc13351)

[3.2 硬件设计 10](#_Toc16851)

[3.2.1 电机驱动 10](#_Toc10102)

[3.2.2 主控板 12](#_Toc4551)

[3.2.3 储能方案 13](#_Toc27548)

[3.3.4 光敏元件 14](#_Toc5234)

[4 系统软件设计 17](#_Toc6582)

[4.1 平台与开发环境 17](#_Toc30682)

[4.1.1 控制流程 17](#_Toc17864)

[4.1.2 keil 2](#_Toc172)6

[5 结论 3](#_Toc27919)2

[参考文献 3](#_Toc3579)4

[致 谢 3](#_Toc17485)6

# 1 引言

## 1.1 前言

随着电脑圈DIY风潮的到来，越来越多的人选择组装台属于自己的电脑，除了性能之外，多数人对于机箱内的美观也有一定的要求，导致选择的机箱体积过大。根据对身边的人，和拥有自己台式电脑的人的观察和询问，发现大多数人的台式电脑都是放在桌子底下的，这样会导致开机不便。其中便有部分人为了解决这个问题，选择了网吧同款的桌面开机按键，方便的同时也增加了美观。但有能让电脑开机更“科技”一点的方式——远程开机。

目前常见的开机方案有以下几种WOL唤醒、路由器唤醒、远程开机棒、智能插座、物联网模块。WOL唤醒常用与局域网，如果需要连接互联网，则需要公网IP；路由器唤醒需要刷入相应固件，但是每隔一段时间需要更新固件，过于繁琐；远程开机棒价格昂贵，逐渐被市场淘汰；智能插座通过控制电脑是否接入电路，以及主板设置里的通电自动开机选项来达到远程开机的效果，但无法远程关机。目前市面常见的远程开机产品基本上都是物联网模块，通过网络将物联网设备接入网络，从而达到远程控制开关机。

## 1.2 市场主流开机卡的缺点

（1）占用了PCIE插槽资源。现在个人计算机主板追求小型化、集成化，多数采用M-ATX或者ITX规格，此类主板PCIE插槽有限，如果用来插开机卡可能会面临PCIE插槽槽位资源紧张的情况，给计算机后期扩展、升级带来麻烦；

（2）需要较长的跳线。PICE插槽与主板的机箱前置面板插针之间，存在着不同规格的独立显卡，使用跳线的方式，存在与显卡风扇缠绕的风险，而且严重影响美观。 所以在此提出一款从usb接口或者电源D形接口供电的远程开机卡，距离机箱前置面板插针更近，且不占用插槽，使主板的拓展和机箱内的美观性得到保障。并且添加环境温湿度测量的功能，可以放在机箱内监测内部环境，也可以放置在机箱外部监测环境温湿度。并且将其接入小爱同学，实现远程语音开机等功能。

## 1.3 本课题的研究内容

（1）系统的组成情况：STM32F401CCU6最小系统，蜗杆电机和驱动器，4个光敏电阻，太阳能充放电系统；

（2）系统实现的主要功能：

* 自动将太阳能板调整到与太阳方位角相同的位置
* 发电后储能，稳压输出

（3）主要性能指标：

* 太阳能板电压： 12V 
* 太阳能板额定功率：30W
* 系统消耗（转向时）9W
* 系统消耗（静止时）1.5W
* 额定输出功率：100W 
* 额定输出电压：220V
* 电池容量：110W.h

开发平台：

Keil uVision5 IDE：

keil5是一款集成了嵌入式系统软件编程，仿真与调试的一体化IDE软件，相较于keil其他版本，keil5拥有更好的语法提示功能和更强的仿真功能。

Keil5支持STM32全系列的开发，在keil官网可下载对应芯片型号的DFP支持包。开发流程简单，软件易用。

STM32CUBEMX:

STM32CubeMX是STM32Cube工具家族中的一员，从MCU/MPU选型，引脚配置，系统时钟以及外设时钟设置，到外设参数配置，中间件参数配置，它给STM32开发者们提供了一种简单，方便，并且直观的方式来完成这些工作。所有的配置完成后，它还可以根据所选的IDE生成对应的工程和初始化C代码。

立创EDA

国产电路EDA软件，该软件集成了上万电子元件库，大部分元件可直接使用，无需绘制封装，可自动生成BOM表，GERbER, 实现电路设计生产与元件购买一体化服务

Solidworks:

三维绘图软件

# 2 系统的总体设计

## 2.1 系统的总体架构

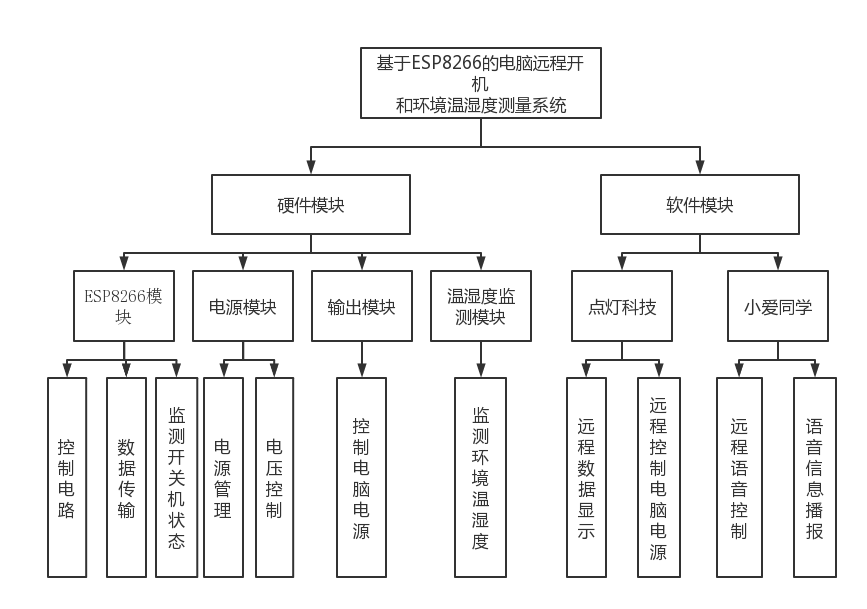
本设计以ESP8266作为控制器，需要将ESP8266MCU和电源模块、输出模块进行连接，用来控制电脑的电源开关。能够让用户使用手机操作端通过Wi-Fi向ESP8266发送指令，用来触发开关机命令，并执行。这种系统架构可以通过下面这种方式实现：

图 2-1 系统总体框图

1. ESP8266模块：本设计以ESP8266为微控制器，使用PCB板将ESP8266连接到电源模块和输出模块。从而达到控制总体电路、数据传输、监测开关机状态的功能。
2. 电源模块：电脑主板的9针USB接口获取5V电压，经过导线与ESP8266连接起来。接下来在电源模块中添加稳压模块，将5V电源电压降至3.3V，使ESP8266可以正常工作。
3. 输出模块：使用光电隔离器，与ESP8266连接，同时将电脑的开关机跳线与机箱开关跳线分离开来，来稳定的控制电脑电源开关。
4. 温湿度监测模块：通过温湿度传感器，将监测数据传给微控制器，并由微控制器上传到软件端。
5. 点灯科技：提供了物联网接入平台，可在APP上设置虚拟按键，及其功能，并将物联网设备的数据显示在仪表盘上。
6. 小爱同学：中国目前最流行的语音助手之一，提供了小米多种设备或者其他物联网平台的接入，支持智能家居和智能手机等设备之间的连接和控制，让用户在硬件控制方面更加方便。

## 2.2 系统实现的主要功能

1. 网页智能配网：上电后连上对应热点，按照提示配网
2. 检测温湿度：利用温湿度传感器监测设备周围温湿度
3. 远程数据传输和显示：将电脑状态，温湿度传输至物联网平台，并显示在APP仪表盘上
4. 远程控制开关机：利用手机APP可以实现远程开关电脑
5. 语音控制：通过训练小爱同学，可以实现语音控制电脑开关

## 2.3 系统运作的流程

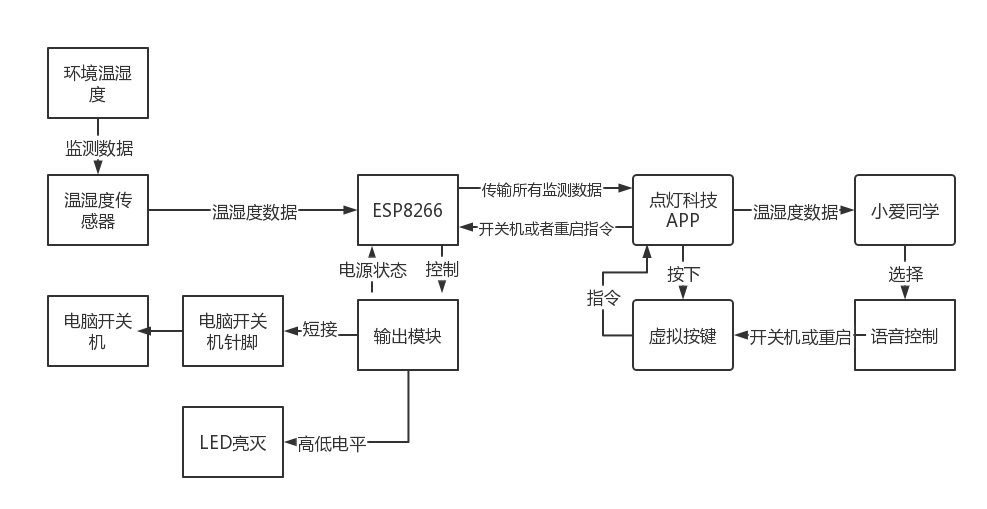


图 2-2 系统运行的基本过程

由温湿度传感器将环境温湿度数据传输至ESP8266,同时输出模块的电源状态也传输至ESP8266，然后再由ESP8266将数据打包发送至点灯科技APP，显示在仪表盘上。小爱同学从点灯科技APP获取温湿度数据。可以手机端进行语音控制，等同于在点灯科技APP按下虚拟按键，再将指令发送回ESP8266，然后由ESP8266控制输出模块，使相应功能的针脚短接[[1]](#footnote-0)，从而达到远程控制的效果。

# 3 系统硬件设计

本章节主要介绍本设计中的电路设计，电路原理，元件选型等硬件设计相关内容

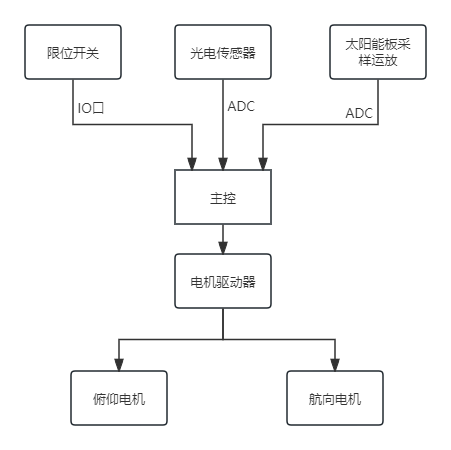


图3-1 控制系统框图

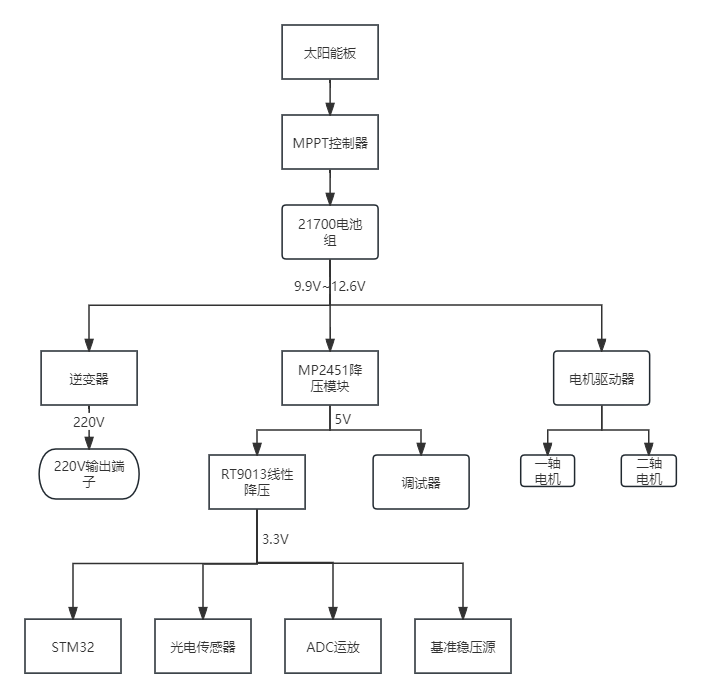


图3-2 电源系统框图

## 3.1 硬件选型

### 3.1.1 处理器的选型

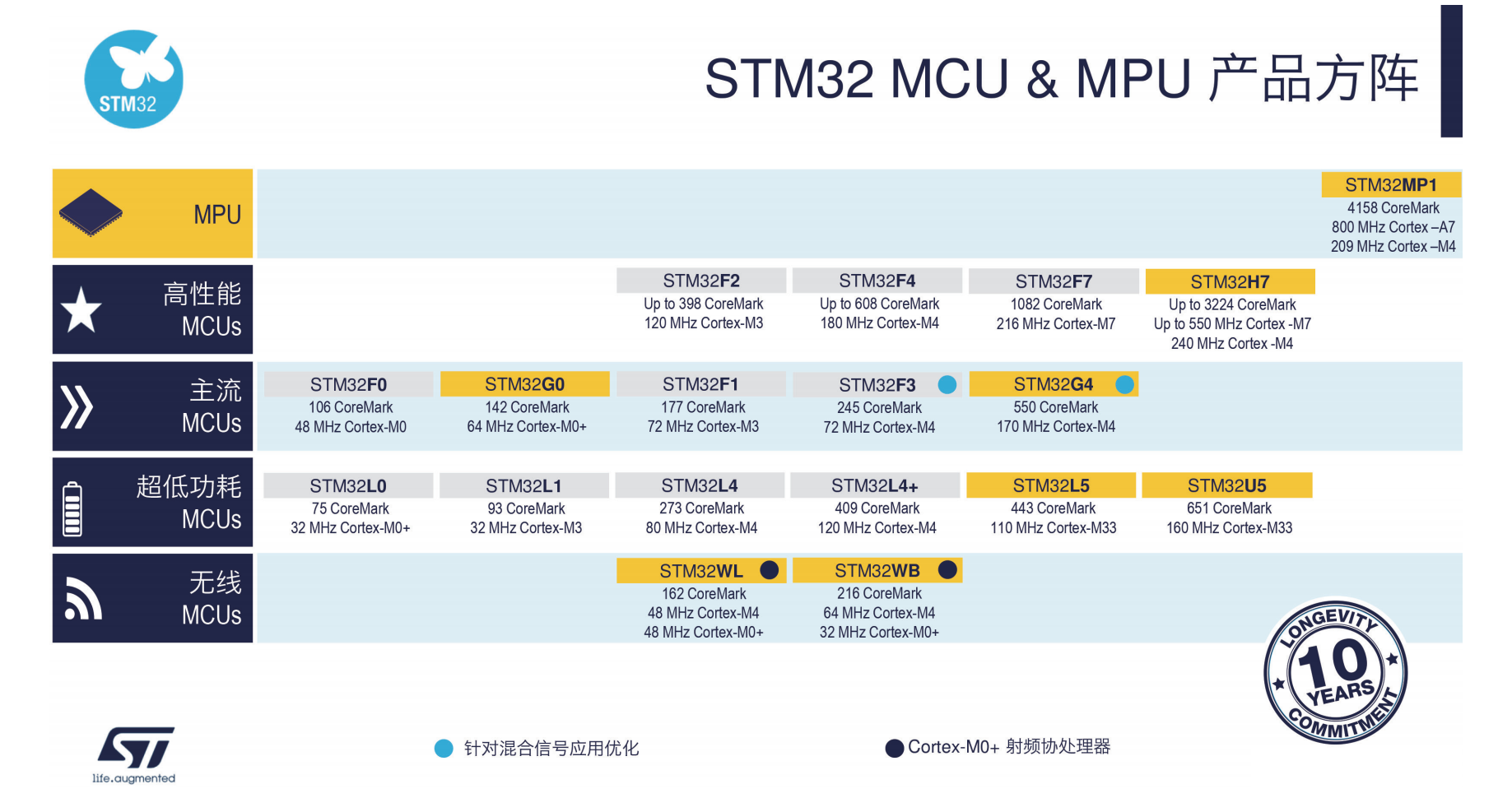
本设计中需要用到ADC, PWM和串口等外设，还最好有较多IO口，以应对多个限位开关，模块功能引脚等应用，同时可能会涉及到部分浮点运算如PID, ADC数据滤波，下面列出3种类型(大类)单片机进行比较

表3-1 ESP8266、ESP32、Ai-WB2的数据比较

|  |  |  |  |
| --- | --- | --- | --- |
| 对比内容 | 89C51 | STM32 | ESP32 |
| ADC | 无 | 最少8路ADC | 最少8路ADC |
| PWM | 定时器模拟PWM | 专用PWM输出通道，最多24路 | 专用PWM输出通道，最多8路 |
| 串口 | 一路 | 1~6路，部分支持485,232 | 1~3路 |
| 无线 | 无 | 部分型号支持 | 带蓝牙WIFI |
| 主频 | 最大36Mhz | 最大480MHZ | 最大240MHZ |
| 浮点运算FPU | 无 | 有 | 有 |
| 价格 | 5元左右 | 0.5~90元左右 | 10~15元左右 |

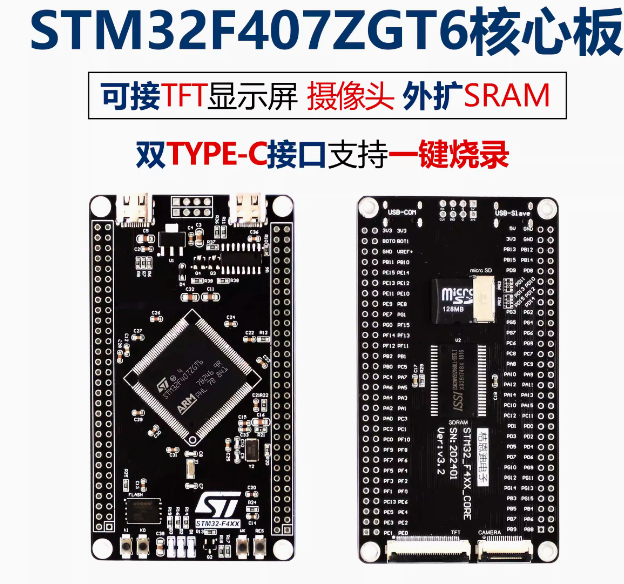
本设计所需IO接口较多，同时需要用到ADC,PWM等模块，如果使用51则需要额外购买ADC芯片，所以排除51，本设计中不会用到无线部分，所以排除ESP32，综合价格选择STM32系列主控。

STM32系列有F0,F1,F3,F4,F7,H7.G0,MP1等系列，每一个系列下又有不同的型号，其中F3以后的系列均带有FPU运算单元详情如下

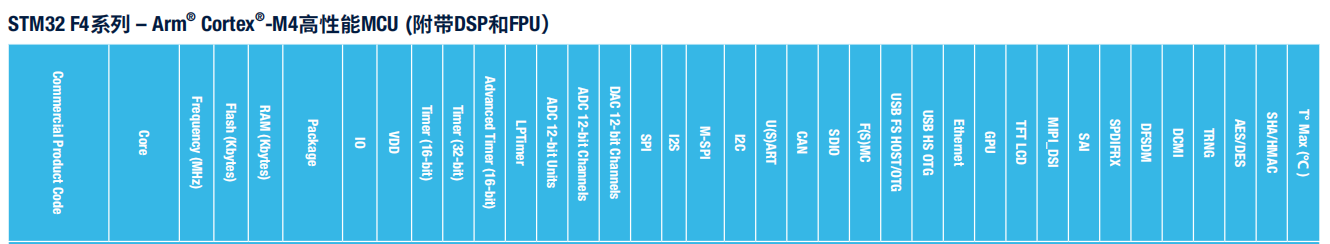
本设计中选用F4系列，F4系列是带FPU性价比最高的，同时功能丰富。本设计中选用F4系列的最小系统板，市面上F4系列最小系统板有如下几种



STM32F411核心板 STM32F401核心板

综上，市面上主要有F401 F411 F407 和F429系列主控的核心板，下面是这几款主控的参数



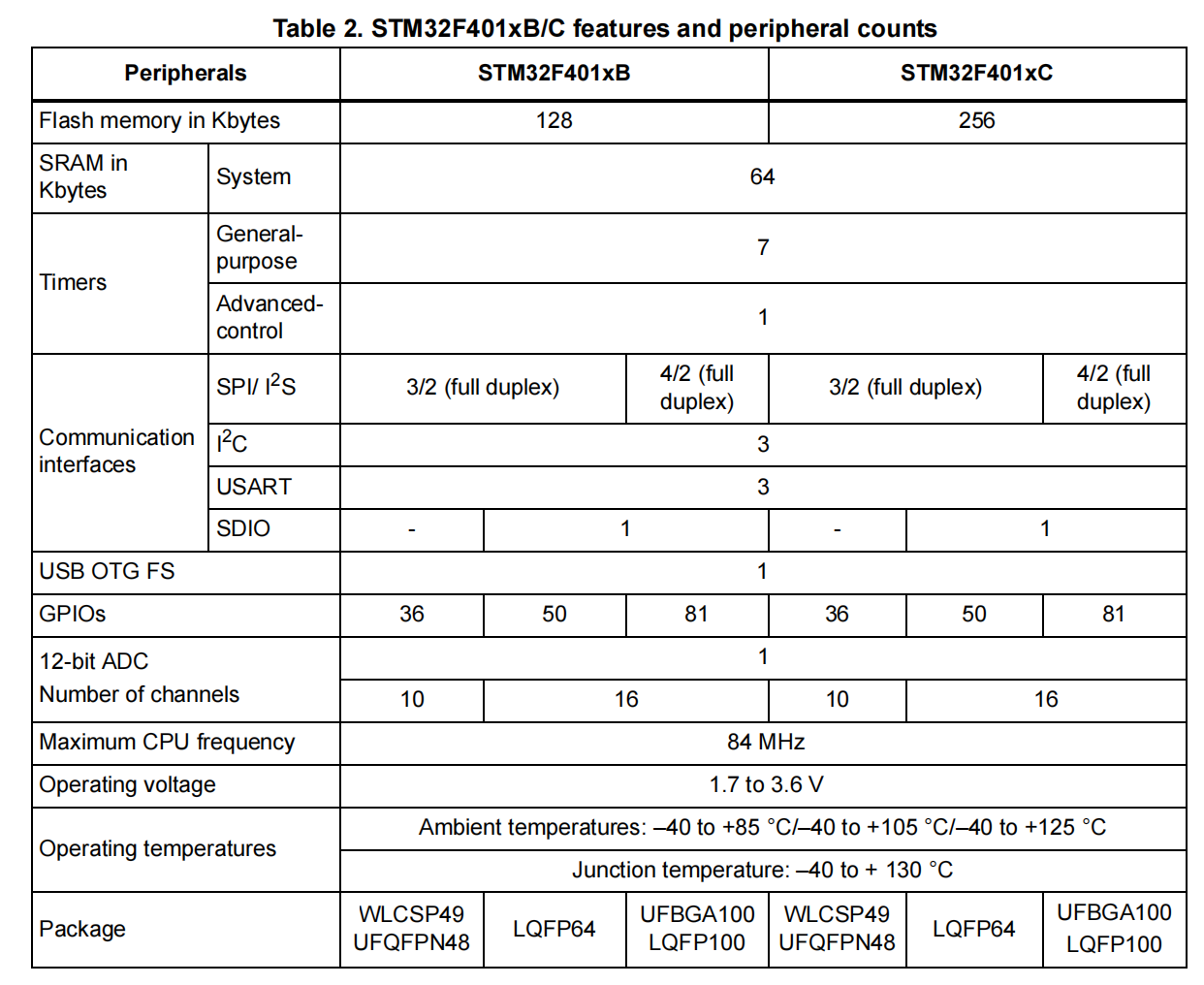


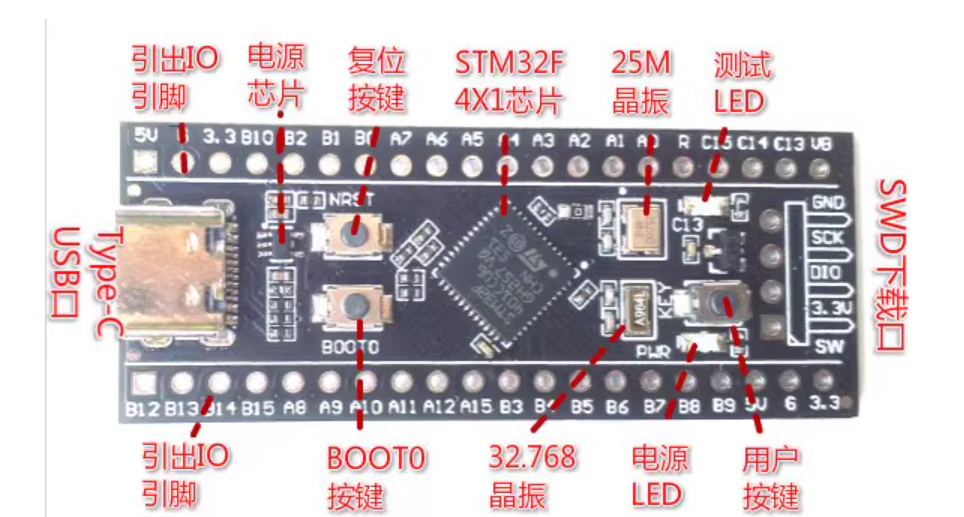


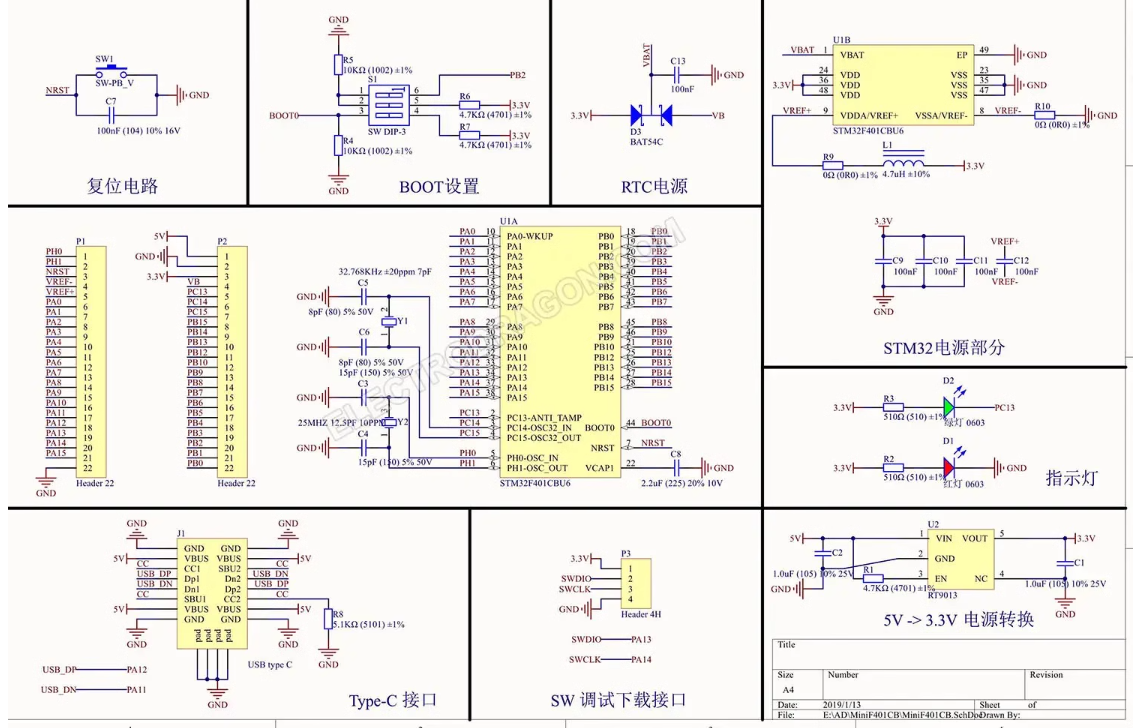




本设计中选用STM32F401,具体参数如下



* 型号：STM32F401CCU6核心板
* 价格：14.5
* FLASH容量256KB
* SRAM容量：64KB
* 主频：最大84MHz
* 定时器：7个普通定时器，1个高级定时器。
* ADC: 1个12位ADC模块，10ADC通道
* 通信外设：3个全双工SPI，3个I2C外设，3个USART（串口）外设一个SDIO接口
* 1个USB OTG FS接口
* 板载3.3V降压芯片，晶振等最小系统基本元件  
    
   最小系统板板载元件介绍



最小系统板原理图

### 3.1.2 稳压器的选择

LDO（Low Dropout Voltage Regulator）是一种线性稳压器，也叫低压差线性稳压器。是一种微型片上系统，功耗极低。

LDO有多个功能模块集成在同一芯片上，具有极低导通电阻MOSFET调整元件、基准电压、误差放大器和各种保护电路等。

LDO的主要作用是将输入电源的电压稳定到所需的输出电压。经过MOSFET调整元件的导通电阻，让输出电压始终保持在稳定的水平。相比传统的线性稳压器，LDO有较低的压降，改善了传统线性稳压器效率低的缺点。

LDO广泛应用于各种电子设备和系统中，特别是对电源稳定性要求较高的应用场景。可以用于移动设备、通信设备、计算机、工控系统等，提供稳定可靠的电源输出。

本设备由于ESP8266[[2]](#footnote-1)工作电压为3.3V，电源输入电压为5V所以选择固定输出3.3V的AMS1117-3.3为本次电源电路使用的硬件。

图 3-3 AMS1117-3.3的原理图

### 3.1.3 光耦隔离器的选择

机箱跳线开关机与远程控制开关机各有1对引脚，所以需要2个独立的电路分别控制远程开关机和机箱开关机。光耦隔离器是一种电子元器件，它利用光电转换的原理将输入端和输出端完全隔离，以达到电气隔离和信号传输的双重功能。光电耦合器是光电隔离电路的核心器件，它具有单向信号传输、输入输出端完全电气隔离、抗干扰能力强、工作稳定、无触点、寿命长、传输效率高等优点。已经运用于许多领域，在电气绝缘、电平转换、级间耦合、驱动电路、开关电路、斩波器、振荡器、级间隔离、数字仪表、远距离传输、通信设备等方面都有广泛的应用。但光电耦合器虽然是一种电流控制的电流转移器件，具有和双极性晶体管类似的传输特性，但由于它的线性工作区比较窄，电流传输比受温度影响比较明显，不适合作为模拟电路的输入输出隔离。

结合价格，体积，选用TLP293为光耦隔离器。

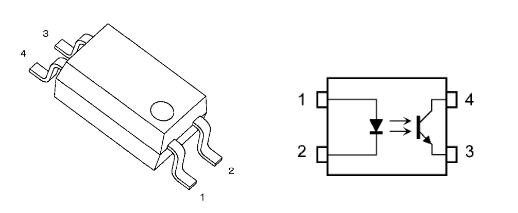


图 3-4 TLP293的封装及引脚图

## 3.2 硬件设计

立创EDA是一款国产的PCB设计软件，提供完整的电路设计和制造解决方案。它的界面简洁易用，提供了丰富的元件库、封装库和板子库，支持多种导入导出格式，能够快速设计出符合要求的电路板。

### 

图 3-5 立创EDA页面

### 3.2.1 ESP8266[[3]](#footnote-2)单片机的最小系统

单片机最小系统说的通熟易懂的话就是以最少的元器件组成能让单片机工作起来的系统，接下来开始介绍ESP8266单片机最小系统必备的器件及其作用。完整的ESP8266单片机最小系统电路图如图3-6所示。

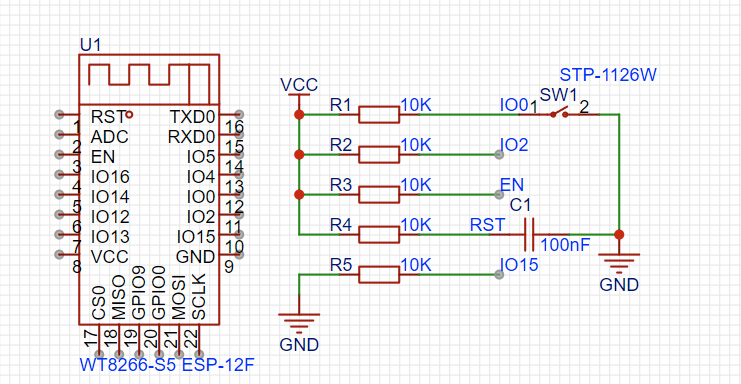


图3-6 ESP8266单片机最小系统

首先电源这对于一个电子产品的话是必不可少，它提供能源给系统运作，在本设计中由于ESP8266单片机的工作电压是2.5V到3.6V，通常使用3.3V电源。如果使用5V电源，则需要通过稳压芯片或者电平转换器将电压转换为3.3V，以避免对ESP8266的损坏。ESP8266的电源稳定性对于其正常工作非常重要，因此需要使用高质量的电源，以避免电压波动或噪声对ESP8266的影响。由于电脑上的USB提供的是5V电压所以需要一个电源模块来进行稳压。

ESP8266的晶振电路采用外部晶体振荡器，主要作用是提供时钟信号，使ESP8266正常工作。本设计的ESP8266最小系统中，有晶振电路，其中包括一个晶体振荡器和两个电容。不过已经被集成在了ESP12F之中，且符合使用要求，无须单独设计。

晶体振荡器一般来说是一个石英晶体振荡器，有高稳定性、准确性。电容通常放在晶振的两端，用来稳定晶振的频率。根据晶振的频率，可以达到用不同的晶振电路来满足不同的需求。

晶振电路的稳定性部分是非常重要的，尤其是对整个电路，会对系统的时钟精度和稳定性产生影响。所以，设计ESP8266电路时，要注意晶振电路的设计和选择，来确保系统的稳定性、可靠性。

然后复位电路，ESP8266中的一个重要硬件就是复位电路，用来保证芯片的稳定运行。在ESP8266设计中，复位电路主要有复位电路器件、复位电路电源两个部分。复位电路器件一般包括一个复位电路芯片以及一个电容器，它们连接到ESP8266芯片的RST引脚上实现复位功能。在RST引脚上的电压低于一定值时，复位电路器件向ESP8266芯片发送信号，用来实现芯片的重启。复位电路电源主要是复位电路和ESP8266的电源保证。为了保证复位电路的可靠，需要为它提供稳定的电源。所以，在设计时要考虑到复位电路和ESP8266的电源合理行，用来保证复位电路的正常工作。

### 3.2.2 主板USB9针接口

主板上的9针USB接口通常被称为“针脚排”，作用是连接电脑主板和USB接口。一般情况有9个针脚，用于连接USB接口。您提到的九针串口是指RS-232接口的九针串口，它是个人电脑上的通信接口之一，也是由电子工业协会（EIA）制定的异步传输标准接口。九针串口具有以下功能：

* 载波检测（DCD）：检测通信线路上是否有载波信号。
* 接收数据（RXD）：接收从外部设备发送过来的数据。
* 发送数据（TXD）：将数据发送到外部设备。
* 数据终端就绪（DTR）：表示电脑准备好进行通信。
* 信号地（GND）：用作信号的共地引线，建立信号间的参考电位。
* 数据就绪（DSR）：表示外部设备准备好进行通信。
* 发送请求（RTS）：电脑向外部设备发送的请求信号，表示准备发送数据。
* 发送清除（CTS）：外部设备向电脑发送的清除信号，表示准备接收数据。
* 振铃指示（RI）：表示外部设备的电话线路上有振铃信号。

九针串口通过这些功能线路进行数据的接收和发送，实现计算机与外部设备之间的数据通信。

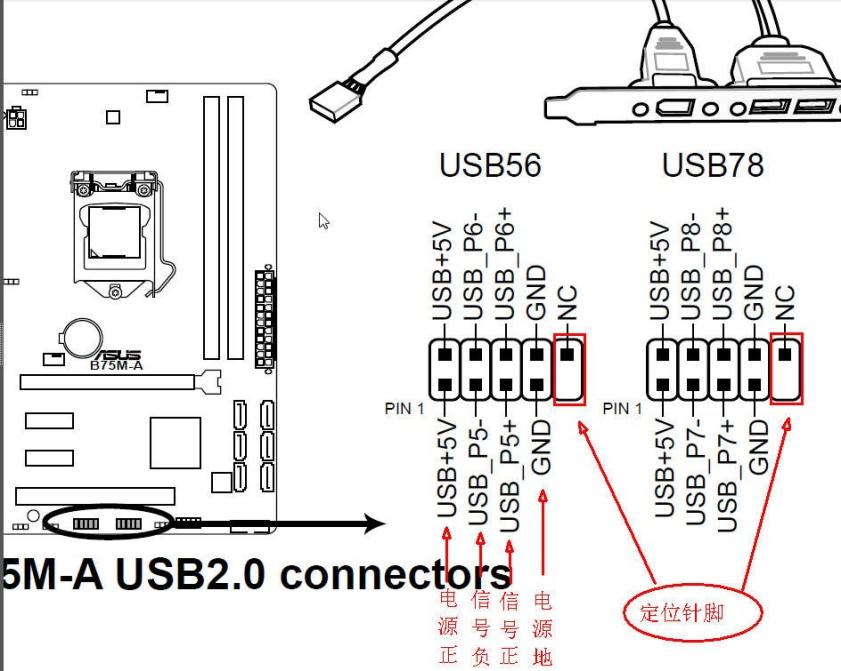


图3-7 主板9针USB接口图

### 3.2.3 稳压模块AMS1117

AMS1117 是款正电压输出的低压降三端线性电路元器件，1A输出电流下的压降为1.2V。AMS1117 有为两个版本，包括固定电压输出版本和可调电压版本。

AMS1117，提供了不同的输出电压选项，包括1.2V、3.3V和5.0V等，和可调版本，一般可调版本的电压精度为百分之一，只有输出电压为1.2V的产品不同，它的精度为2％。电压稳压器内部集成了过热保护和限流电路，适用于各类电子产品。本次采用的ESP8266通常使用3.3V电源。如果使用5V电源，则需要通过稳压芯片或者电平转换器将电压转换为3.3V，以避免对ESP8266的损坏。所以选择固定输出电压3.3V的1117-3.3一般AMS1117内部框图如图3-8所示。

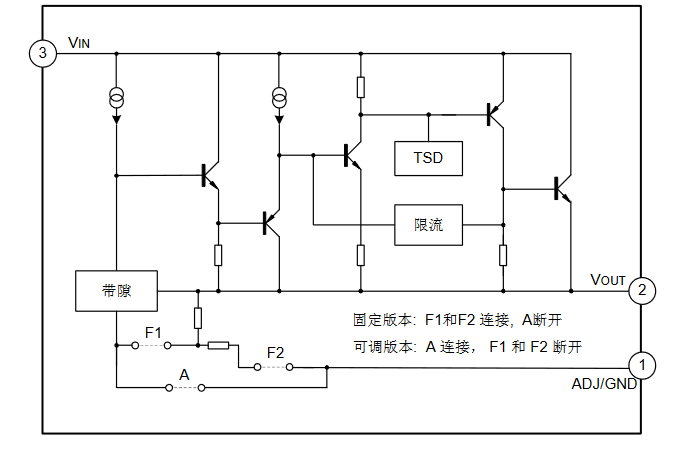


图3-8 AMS1117内部框图

### 3.3.4 光耦隔离器件TLP293

TLP293是一种光耦隔离器件，利用光学传感器将输入的电信号在内部使其转换为光信号，然后传递到输出端，途径隔离区域，最后把光信号转换为电信号，从而输出。因为采用了光电耦合的工作方式，使得输入和输出之间可以完全隔离，实现了电气隔离和信号传输的双重功能。TLP293可用于需要隔离高电压和低电压信号的场合，如隔离控制系统中的驱动信号和传感器信号。除此之外，TLP293还有高共模抑制比、高速开关、低电流驱动等特点，在自动化、通信、医疗等领域有广泛应用。

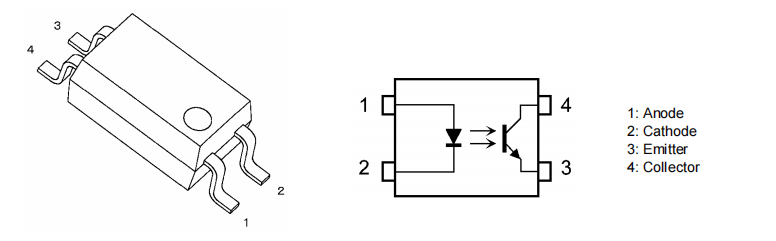


图3-9 TPL293的封装图

### 3.2.5 TLP293的使用及电路

TLP293是光耦隔离元器件，一般用于隔离高压和低压电路之间的信号传输。TLP293使用时需要把输入信号连接到端口1、2，输出信号连接到端口4、5。当控制端口电压高于设定值的时候，光电耦合器会导通输出端口的电流。想要进行高速开关操作，可以把控制端口连接到处理器或者其他电路的输出端口，用来实现控制。在使用TLP293时要注意避免超过它最大额定电流以及最大额定电压，这样可以确保设备的正常运行和使用安全。

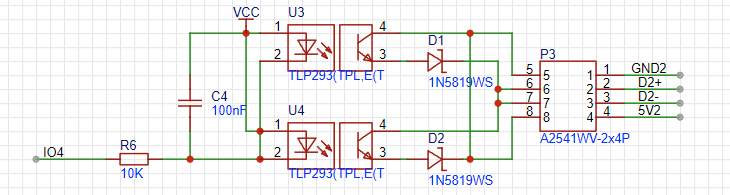


图 3-10 TPL293的输出电路

## 3.3 系统硬件测试

系统硬件电路的测试是一项重要的任务，能够保证电路的正确工作和使用稳定。

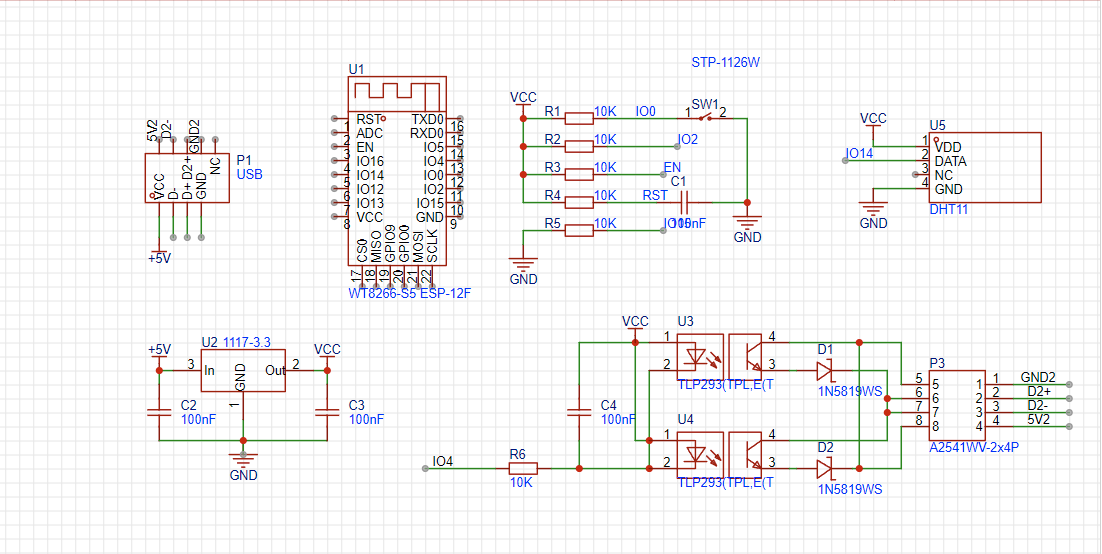
首先可以使用仿真软件进行电路理论测试，测试这个电路能不能达到理想工作状态，对整个系统进行实验、测试和分析，来提高生产效率、降低成本还能减少潜在风险。

软件Arduino烧录的步骤如下：

1. 准备工具和材料：Arduino开发板、数据线、电脑、Arduino IDE软件。
2. 把Arduino板插入电脑的USB接口，然后打开Arduino IDE。
3. 在Arduino IDE软件中选择正确的端口，来与Arduino板进行通信。
4. 点击“文件”菜单，选择“上传”选项，把程序上传到Arduino开发板。
5. 若上传成功，Arduino IDE会显示一个绿色的“上传成功”。
6. 若上传失败，Arduino IDE会显示一个红色的“上传失败”，此时要检查代码正确与否、电源连接稳定与否等。
7. 上传完成后，可以开始使用Arduino板进行实验和编程。

要注意的是，进行烧录时，务必保证电源连接稳定，避免因为电源问题导致烧录失败又或是损坏Arduino开发板。并且，也要保护好Arduino板和数据线的接口，避免损坏。

然后再在PCB上进行元器件的焊接、调试。进行系统硬件电路测试时，需要设计测试计划，使用正确的测试仪器和方法。这样能够保证测试结果的准确性，有助于发现和解决问题。系统硬件电路的测试主要关注电路是否存在漏焊、短路、断路、虚焊以及一些具有方向性的元件是否安装错误、电路设计错误等情况。

图3-11完整的电路原理图

# 4 系统软件设计

本设计由移动端设备操作，例如手机，小爱音箱等。通过移动端的操作，将指令发送给控制器，并且将控制器的响应结果返回到仪表盘。

硬件端响应，包括平台接入，按键控制等。接收来自操作端的指令，并对指令进行识别，响应。还要将实时数据传输至操作端。

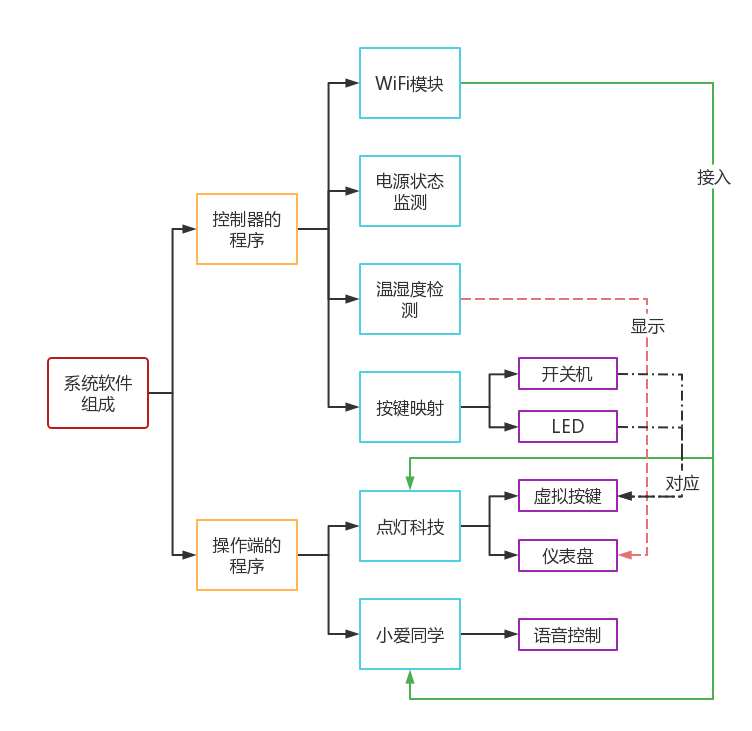


图3-12 系统的软件组成

## 4.1 平台与开发环境

### 4.1.1 接入点灯科技物联网平台

下载点灯科技的Blinker APP，在app中添加设备，获取Secret Key

1. 进入App，点击右上角的“+”号，然后选择 添加设备
2. 点击选择Arduino > WiFi接入
3. 复制申请到的Secret Key
4. 将自己的 Secret Key填进代码中

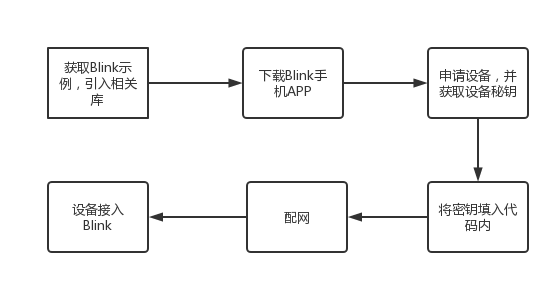


图 4-1 Blinker接入流程

Blinker是一种基于物联网的解决方案，能够轻松地将Arduino和其他硬件设备连接到云端，然后使用手机应用程序进行远程控制。下面是一个使用Blinker的示例程序：

*#include <Blinker.h>*

*char auth[] = "Your\_Auth\_Key"; // Blinker身份验证密钥*

*char ssid[] = "Your\_SSID"; // WiFi网络名称*

*char pswd[] = "Your\_WIFI\_Password"; // WiFi网络密码*

在上述代码中，我们首先需要包含Blinker库。然后，我们需要设置Blinker身份验证密钥、WiFi网络名称和密码。接着，我们创建了一个名为N1的数字数据流。在setup()函数中，我们通过Blinker.begin()函数初始化Blinker。在loop()函数中，通过N1.print()函数向数据流N1中写入了一个随机数，然后用delay()函数使其暂停1秒钟。到最后，使用Blinker.run()函数来运行Blinker。

### 4.1.2 Arduino

Arduino的软件部分包括Arduino IDE和库文件。Arduino IDE是款跨平台的集成开发环境，可用来编写和上传程序到Arduino控制器。编程语言类似C++，但是更简单易学，不需要过多的电子背景知识。库文件则提供了各种功能的实现：数字/模拟输入输出、串口通信、网络通信、传感器和执行器的控制等。

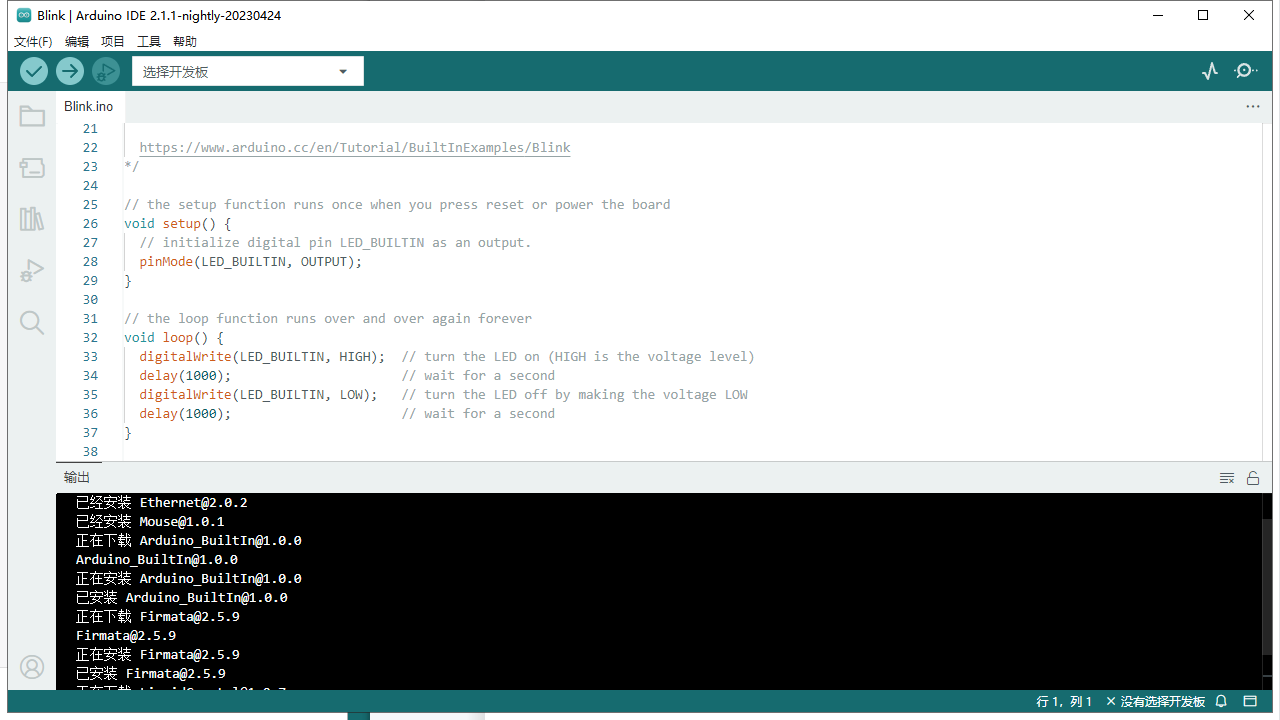


图 4-2 Arduino的Blink示例

### 4.1.3 Blinker

Blinker是款物联网解决方案，可以帮助用户快速地搭建智能家居和物联网应用。Blinker提供了完整的物联网平台，其中有硬件开发、云端服务、移动应用程序等内功。

在Blinker平台上，任何人都可以使用Arduino、ESP8266、ESP32、等硬件来接入各种传感器和设备，然后通过云端服务从而达到远程控制和监控。Blinker提供了可视化的控制面板、移动端程序和API接口等，用来访问设备和数据，方便用户进行智能家居和物联网应用的开发和管理。

## 4.2 温湿度采集模块

### 4.2.1 温湿度采集模块的主要功能

DHT11是一款湿温度一体化的数字传感器，包括一个测湿元件（电阻式）和一个测温元件（NTC），通过ARM芯片之间的电路连接，从此就可以随时随地地采集到传感器所在地区的温湿度，DHT11和ARM芯片之间会采用单总线进行通信，这个方式非常简单，因此一个IO接口便足够。同时单总线格式也是数据格式所采用的，即单个数据引脚端口完成输入输出双向传输。数据包的大小为5Byte，具体的数据格式为: 湿度整数数据（8bit） +湿度小数数据（8bit）+温度整数数据（8bit）+温度小数数据（8bit）+ 校验（8bit），所以40bit是一次完整的数据传输，其中校验和数据为前4个字节相加，传感器数据输出的是未编码的二进制数据，数据之间应该分开处理，例如，某次从 DHT11 读到的数据 byte0 为校验和，byte1 为温度的小数部分，byte2 为温度的整数部分，byte3 为湿度的小数部分，byte4 为湿度的整数部分，由以上数据就可得到湿度和温度的值，计算方法:

湿度 = byte4． byte3( % RH)

温度 = byte2． byte1( °C )

校验 = byte4 + byte3 + byte2 + byte1 ( = 湿度 + 温度) ( 校验正确)

### wps4.2.2 DHT11数据处理

图 4-3 DHT11 处理流程图

在DHT11开始工作后，会读取传感器的数据，并且将数据赋值到对应的键名，判断数据是否读取成功，如果读取成功，则将数据回传给Blinker APP，对应的键名对应其数据。

### 4.2.3 关键代码

*void heartbeat()*

*{*

*HUMI.print(humi\_read); //给blinkerapp回传湿度数据*

*TEMP.print(temp\_read); //给blinkerapp回传温度数据*

*}*

*void miotQuery(int32\_t queryCode) //小爱同学语音命令反馈*

*{*

*BLINKER\_LOG("MIOT Query codes: ", queryCode);*

*int humi\_read\_int=humi\_read; //去掉湿度浮点数*

*BlinkerMIOT.humi(humi\_read\_int); //小爱反馈湿度属性*

*BlinkerMIOT.temp(temp\_read); //小爱反馈温度属性*

*BlinkerMIOT.print();//将以上属性发送给小爱，使得小爱可以接收到温湿度的数据*

*}*

*void setup()//将设置代码放在此处，运行一次；*

*{*

*//初始化端口*

*Serial.begin(115200);*

*BLINKER\_DEBUG.stream(Serial);*

*BLINKER\_DEBUG.debugAll();*

*Blinker.begin(auth, ssid, pswd); // // 初始化blinker*

*Blinker.attachHeartbeat(heartbeat);//将传感器获取的数据传给blinker app上*

*dht.begin();//初始化DHT传感器*

*//在回调函数中反馈该控制状态*

*BlinkerMIOT.attachQuery(miotQuery);//每次呼出小爱同学，就会调用miotQuery()函数*

*}*

*//通过循环不断读取温湿度传感器获取的数据*

*void loop() //把主代码放在这里，重复运行：*

*{*

*Blinker.run();//运行Blinker*

*float h = dht.readHumidity();//读取DHT11传感器的湿度 并赋值给h*

*float t = dht.readTemperature();//读取传感器的温度 并赋值给t*

*if (isnan(h) || isnan(t))//判断是否成功读取到温湿度数据*

*BLINKER\_LOG("Failed to read from DHT sensor!");//读取温湿度失败！*

*else//成功读取到数据*

*{*

*//打印*

*BLINKER\_LOG("Humidity: ", h, " %");*

*BLINKER\_LOG("Temperature: ", t, " \*C");*

*humi\_read = h;//将读取到的湿度赋值给全局变量humi\_read*

*temp\_read = t;//将读取到的温度赋值给全局变量temp\_read*

*}*

*Blinker.delay(2000);//延时函数*

*}*

这段代码实现了一个基于DHT11温湿度传感器的物联网设备，可以通过Blinker平台与手机App进行交互和控制。在setup函数中，通过Blinker.begin()函数初始化Blinker平台，连接到Wi-Fi，并将设备授权信息等信息传入。同时，通过Blinker.attachHeartbeat()函数，将传感器获取的温湿度数据回传给Blinker平台，并在miotQuery()函数中处理小爱同学语音命令的反馈。在loop函数中，通过dht.readHumidity()和dht.readTemperature()函数从DHT11传感器读取温湿度数据，将数据赋值给全局变量humi\_read和temp\_read，并通过Blinker.delay()函数实现延时。最后，将获取到的温湿度数据通过BlinkerNumber的print()函数发送给Blinker平台，并在miotQuery()函数中将数据发送给小爱同学。

### 4.2.4 界面效果

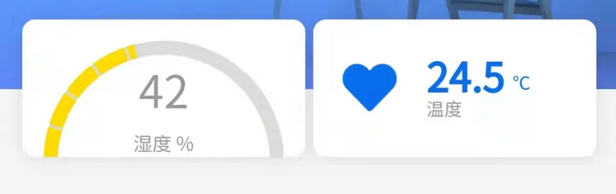


图 4-4 手机APP端的温湿度仪表盘

该仪表盘的界面效果可根据个人喜好进行更改。

## 4.3 开关机控制模块

### 4.3.1 开关机模块的主要功能

在软件操作端，我们设置了两个虚拟按键：电源、LED。这两个按键分别对应着不同的功能。为此在硬件设计里，我们需要为这两个按键分别设置程序，来达到其对应的功能。

### 4.3.2 按键数据处理

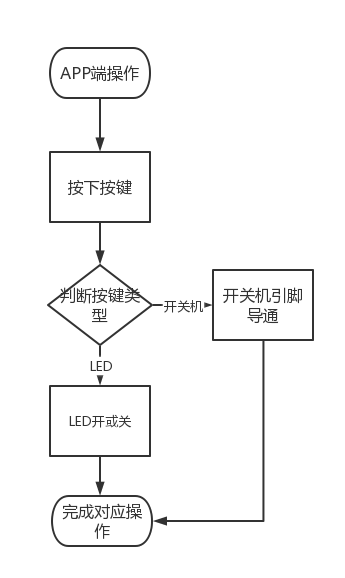


图 4-5按键数据处理流程

操作端按下对应按键，将指令数据发送到ESP-12F，并且对指令进行分析判断，从而执行对应的任务。

### 4.3.3 关键代码

*void button1\_callback(const String & state)*

*{*

*BLINKER\_LOG("get button state: ", state);*

*if (state == BLINKER\_CMD\_ON)*

*{*

*BLINKER\_LOG("Button on!");*

*//Button1.color("#00ffff");*

*Button1.print("on");*

*PC\_ON();*

*}*

*else if (state == BLINKER\_CMD\_OFF)*

*{*

*BLINKER\_LOG("Button off!");*

*//Button1.color("#555555");*

*Button1.print("off");*

*PC\_OFF();*

*}*

*}*

*void button2\_callback(const String & state)*

*{*

*BLINKER\_LOG("get button state: ", state);*

*if (state == BLINKER\_CMD\_ON)*

*{*

*BLINKER\_LOG("Button on!");*

*Button2.print("on");*

*LED\_sign = 0;*

*}*

*else if (state == BLINKER\_CMD\_OFF)*

*{*

*BLINKER\_LOG("Button off!");*

*Button2.print("off");*

*LED\_sign = 1;*

*}*

*}*

*void button3\_callback(const String & state)*

*{*

*BLINKER\_LOG("get button state: ", state);*

*if (state == BLINKER\_CMD\_BUTTON\_RELEASED)*

*{*

*BLINKER\_LOG("RST!");*

*Button3.print("RST");*

*Button1.print("off");*

*PC\_RST();*

*}}*

### 4.3.4 界面效果

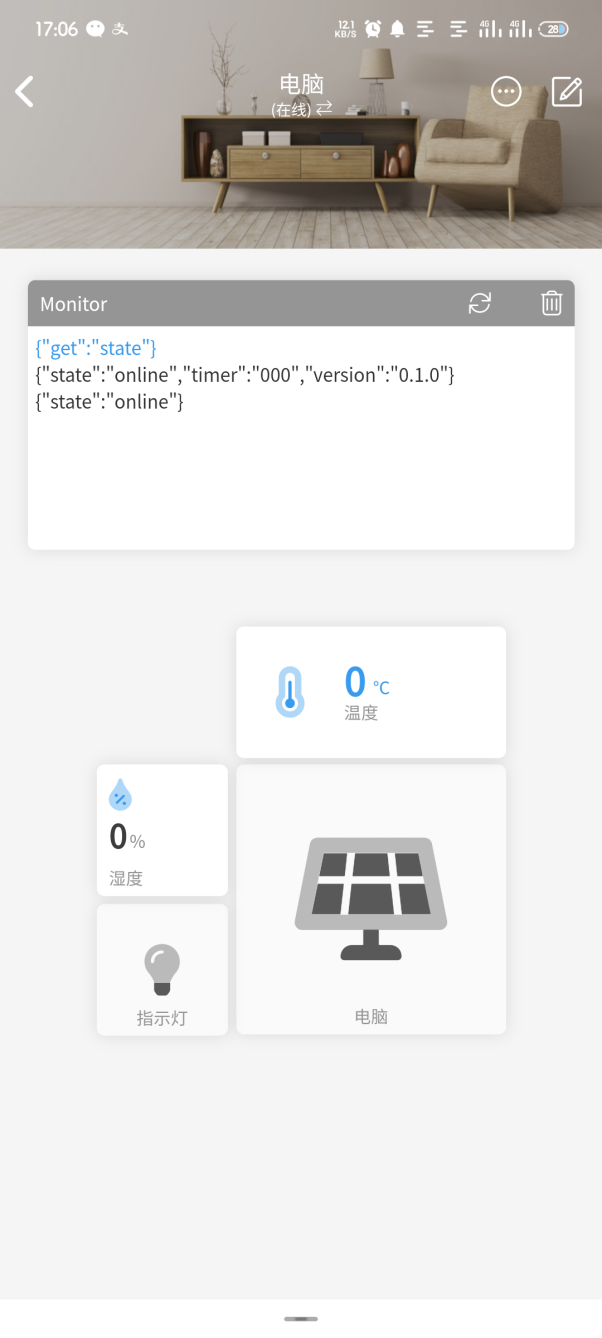


图 4-6 手机APP页面按键效果

重启、电脑、指示灯三个按键均可点击，分别对应不同功能。各个部分的图标均可替换。

# 5 系统测试

本章主要为系统能够达到的性能指标的检测，其中包括网络环境、数据准确度、指令执行程度三项测试内容。首先，网络环境是系统性能测试中不可忽视的一项指标。我们通过模拟不同的网络环境，来检测系统在不同网络环境下的性能表现。

其次，数据准确度是系统测试中至关重要的一项指标，通过测试检查数据准确度，数据响应时间，以确保系统能保持快速、高准确度的数据处理能力。

最后，指令执行程度是系统性能测试中的关键指标之一。检测系统能否快速、准确地执行各种指令。

## 5.1 网络环境测试

开启路由器，本次采用代码配网方式，将WiFi网络名称与密码烧录进去。上传成功后，程序开始自动运行。按照下一步操作即可看到输出结果。



图 5-1串口监视器

找到串口监视器并打开，在程序中设置的115200作为波特率，所以我们也要在串口监视器中选择115200。接下来重新给设备通电，随后即可看到程序复位，整体系统开始工作，也可以选择按下复位键RST。

可以看到串口收到的来自开发板的信息。

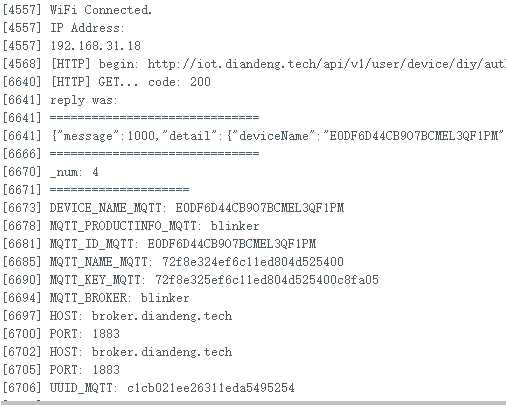


图 5-2 串口监视器显示内容

可以看到显示已经连接WIFI，且IP地址为192.168.31.18。前往路由器管理页面查看设备列表，可以成功看到设备IP及设备名。

前往手机APP页面，可以看到显示设备在线，且有温湿度数据显示

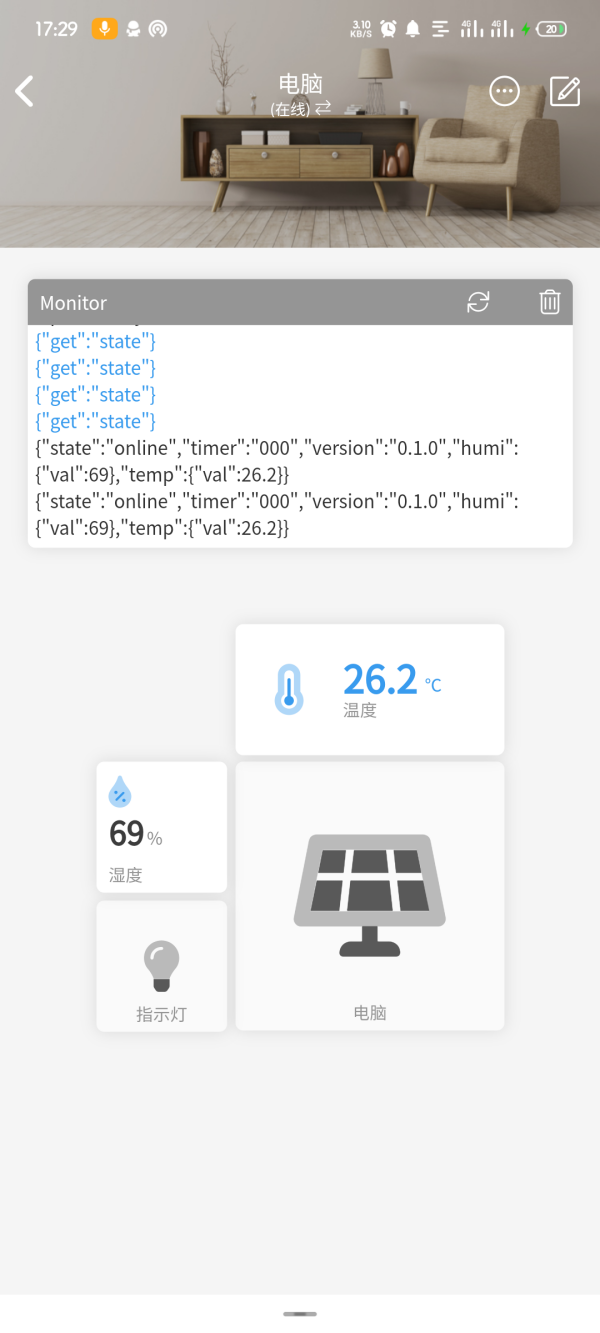


图 5-3 手机APP页面

以上操作及结果表示，网络环境测试成功。

## 5.2 温湿度测量测试

温度测试方法：将一处热源放置在传感器一定距离处，使用测温枪测量DHT11处的温度，同时查看APP端显示的数据并记录，绘制成表

湿度测试方法：准备若干个湿度不同的杯子，将DHT11放置其中，同时使用高精度湿度计测量湿度，同时查看APP端显示的数据并记录，如表5-1所示。

表5-1 温湿度实验数据

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 实验序号 | APP端温度/℃ | 实测温度/℃ | APP端湿度/% | 实测湿度/% |
| 1 | 24.3 | 24.5 | 41 | 43 |
| 2 | 29.4 | 30.5 | 50 | 55 |
| 3 | 40.1 | 43.1 | 58 | 60 |
| 4 | 45.5 | 47.3 | 65 | 65 |
| 5 | 53.9 | 58.1 | 69 | 74 |

经过测试发现，温湿度数据较实时数据偏低，待温湿度完全稳定后，数据与实测数据相差无几。可以推断出是设备发送的数据不是实时数据，是短时间之前的数据，环境温湿度稳定后则在误差范围之内。

## 5.3指令测试

测试方法：使用Blinker APP，按下APP页面的开关机和重启功能键，记录是否成功执行相关操作。

表5-2 按键测试数据

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 试验序号 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 开机 | √ | √ | √ | √ | √ | √ | × | √ |
| 关机 | √ | √ | √ | × | √ | √ | √ | √ |
| LED | √ | √ | √ | √ | √ | √ | √ | √ |

在24次实验中有2次失败，22次成功，考虑到机箱内部信号较弱，电脑距离路由器较远，导致数据传输时可能存在丢包，导致失败。

切换网络，让手机与电脑处在非同一个网络，进行多次测试，结果与上表相似，可以得出结论：远端发送的指令能够正常送达，且能正确执行

# 6 结论

完成了基于ESP8266的电脑远程开机和环境温湿度测量系统的开发。该系统采用ESP8266作为微控制器，利用DHT11温湿度传感器实现环境温湿度的实时检测，并通过连接局域网中的电脑实现了电脑远程开机的功能。用户能在远程操作界面上控制电脑的开机、关机和重启，还能实时的监测环境的温度和湿度。在实验测试中，对系统的功能进行了全面的测试和评估。测试结果显示，此系统可以稳定运行，有很好的响应速度和稳定性，可以够满足多数用户的需求。并且，此系统的硬件成本低，方便制作和维护，有较高的实用性和经济性。所以，本文设计的基于ESP8266的电脑远程开机和环境温湿度测量系统有不错的实用性和可行性，可以为用户提供一种便捷、高效、低成本的解决方案，有着不错的应用前景。

在这次的设计过程中，学到了简单原理图的绘画，以及将其转换成PCB的能力。我们接触了许多新平台，如AD、立创EDA和硬创社。通过自己的实践，增强了动手能力，并丰富了自己在物联网平台方面的知识。学会了使用之前在课上没有接触过的点灯科技和小爱同学等。通过实际工程的设计，也认识到书本知识和实际应用的差别。

在实际操作中，遇到了许多问题，比如原理图的绘画和网络的建立。我学会了查看元器件的说明书，并重温了模拟电路的知识。但在这个方面仍然存在许多不足，许多元器件的封装选择依旧不是很明白该如何选择。同时，针脚过多的元器件也难以看懂其中的复杂电路。

在制作PCB的过程中，没有考虑到机箱的限制，这导致了一些问题。我没有经验也不会使用仿真软件，这导致代码迟迟不能进行测试，仅在理论方面存在可行性。PCB设计过小，导致焊接起来难度不小，耗时耗力。并且没有设计数据烧录模块，焊接后不便取下重新烧录。

本设计的初衷是解决电脑开机不便的问题。因为寝室空间比较紧张，台式机庞大的体积迫使我只能把电脑放在桌子底下，开机键从此离我比较远，每次开机都不方便，所以萌生了使用远程开机的想法。在购买市面上的产品回来体验后，确实发现能给我的生活带来不少的便捷，但PCIE的供电方式对于我来说还是不够方便。所以产生了将其改为无人使用的USB9针供电方式，来解决接口不够用的问题。

同时，在设计过程中遇到的问题和不足也对设计者提出了新的挑战。比如，在绘制原理图和设计PCB时，遇到了元器件封装和针脚过多等问题，这些问题需要更多的学习和实践来解决。另外，由于服务器过期等原因，对物联网平台的选择也需要更多的考虑和实践。

尽管在设计过程中遇到了挺多的困难和挑战，但经过学习和实践，最终成功地完成了基于ESP8266的电脑远程开机和环境温湿度测量系统。这个设计为用户提供了种便捷、高效、低成本的解决办法，还拥有广泛的应用前景。在这个过程中，新的知识和技能不是唯一的收获，自己的动手能力以及解决问题的能力也得到了提升。

**参考文献**

1. 旷工大叔叔:超详细之电脑如何远程开机,访问远程文件[Z]，网易号，2022-10-10，https://www.163.com/dy/article/HJBNND1O0553XQGN.html
2. HIPC：产品文档，HIPC官网，https://docs.hipc.cn/gongneng.html
3. AFATAR：基于ESP8266的远程开机棒设计和实现[Z]，CSDN，2021-2-10,<https://blog.csdn.net/AFATAR/article/details/113774126>
4. 酷电玩家：开源ESP-12F电脑远程开机卡[Z]，bilibli，2023-1-11,https://www.bilibili.com/video/BV1nt4y1y7cG/?share\_source=copy\_web&vd\_source=4d272084f83383b2b8623ddcfde5465e
5. 边寒：极简远程开机卡,无需接线[Z]，立创开源平台，2023-2-6,https://oshwhub.com/ghgjkh/wakeonlan2
6. black科技社：自制远程开机棒丨原理讲解[Z]，bilibli，2022-10-22,https://www.bilibili.com/video/BV1WG4y1H7sx/?share\_source=copy\_web&vd\_source=4d272084f83383b2b8623ddcfde5465e
7. --猛禽--：ESP8266远程开机 diy远程开机器 [Z]，bilibili，2020-2-27,https://www.bilibili.com/video/BV1d7411P7UK/?share\_source=copy\_web&vd\_source=4d272084f83383b2b8623ddcfde5465e
8. black科技社：远程开机方案汇总+手机远程控制电脑[Z]。以及手机丨小度丨小爱丨天猫丨控制电脑开关机。硬核控制电源键。，bilibli，2022-1-6,https://www.bilibili.com/video/BV1MR4y1G7Vf/?share\_source=copy\_web&vd\_source=4d272084f83383b2b8623ddcfde5465e
9. 硬件茶谈：如何把电脑的软开关机接入米家用小爱远程开关机？[Z] ，bilibli，2021-7-30，https://www.bilibili.com/video/BV1Jg41177TG/?share\_source=copy\_web&vd\_source=4d272084f83383b2b8623ddcfde5465e
10. 小右sama：不超50元，将电脑接入米家网络！[Z]，bilibli，2021-3-3,https://www.bilibili.com/video/BV1h54y1h7cE/?share\_source=copy\_web&vd\_source=4d272084f83383b2b8623ddcfde5465e
11. 潘祖记：电脑远程开机pcie卡[Z]，CSDN,2021-9-30，https://blog.csdn.net/panpanloveruth/article/details/120570466
12. BlackBox：DHT11 温度湿度传感器 ESP8266 开发资料[Z]，太极创客，2020-6-17,http://www.taichi-maker.com/homepage/iot-development/iot-dev-reference/esp8266-iot-dev-ref/esp8266-sensor-ref/dht11-esp8266-ref/
13. 侯清江,张黎强,许栋刚.开关电源的基本原理及发展趋势探析[J].制造业自动化,2010,32(09):160-162+169.
14. 唐泽皓,张志博,王晋东.基于STC单片机的光耦控制研究[J].数字技术与应用,2011(12):22.DOI:10.19695/j.cnki.cn12-1369.2011.12.015.
15. 卜永波,罗小玲,陈一.基于DHT11传感器的温湿度采集系统[J].计算机与现代化,2013,No.219(11):133-135.

# 致 谢

首先，我要感谢我的导师：姜向宏副教授,他在整个论文研究过程中给予了我悉心的指导和帮助。他不仅在学术上给予我指导，还在生活上给予了我很多关心和照顾，让我感受到了温暖和支持。

其次，我要感谢对我提供帮助的所有老师和同学，他们给予了我很多帮助和支持，让我能够顺利完成此次的毕业设计。在学校里里，我学到了很多知识和技能，也结交了很多优秀的同学和朋友。

此外，我还要感谢家人和朋友，他们一直以来给予了我无私的支持和关爱，在我遇到困难和挫折时，他们总是在我身边给予我鼓励和支持。

最后，我要感谢所有为本篇论文提供数据、文献和技术支持的人员和机构，没有他们的帮助和支持，本篇论文的研究和写作将无法顺利完成。

再次感谢您的关注和支持！

1. 电脑的电源是由针脚控制，短接则进行相应操作，如通电，断电等 [↑](#footnote-ref-0)
2. 此处指ESP8266-12F [↑](#footnote-ref-1)
3. 后文ESP8266均指本次设计使用的ESP-12F [↑](#footnote-ref-2)