

# 基于嵌入式的语音识别智能家居控制系统设计

胡益峰，丁岩岩

湖北工程学院新技术学院，湖北 孝感 432000

**摘要：**随着科学技术的发展，传统的智能家居系统已经无法满足人们对智能家居系统的需求，而近些年发展迅速的语音识别技术可以很好地解决这一问题。基于此，文章设计了一个基于嵌入式的语音识别智能家居控制系统。以 STM32F103 单片机为核心，通过 DHT11 获取温、湿度数据，并将采集的数据显示在 OLED 上。同时对 MQ-135 的输出进行实时监控，当检测到有害气体时，系统会立刻发出报警信号并开启风扇。语音控制部分采用了 ESP8266 接入互联网，并连接到云平台，对手机 AI 发出语音控制指令后，手机 AI 会将指令转发到云平台，再由云平台将数据发送到 ESP8266，最后 ESP8266 通过串口将控制指令下发到单片机上，实现了使用语音控制家用电器设备。智能家居与语音控制的结合可以使现代的家庭生活更加轻松、便捷。

**关键词：**语音控制；智能家居；物联网

中图分类号：F724

## 1 语音识别智能家居控制系统总体设计

本系统主要分为 stm32 主控芯片部分、ESP8266 模块部分、DHT11 温湿度检测部分、MQ-135 气体检测部分、蜂鸣器报警部分、LED 灯部分以及风扇驱动部分。系统总体框架如图 1 所示。

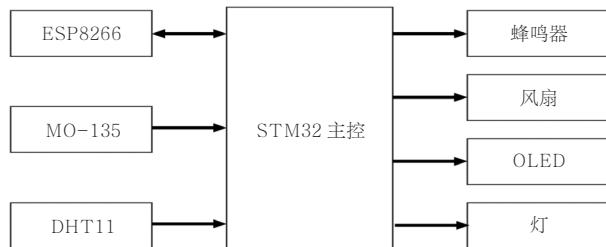


图 1 系统总体框架

## 2 硬件系统设计

### 2.1 电源电路设计

本文采用的电源有 5 V 与 3.3 V 两种，USB 接口提供 5 V 电压，ME6211 芯片提供 3.3 V 电压。ME6211 系列芯片是以 CMOS 工艺制造的低压差线性稳压器，有着输出噪声低、稳定性好等优点<sup>[1-2]</sup>。ME6211 系列的稳压器内置了固定的参考电压源，经过稳压后的电源可以直接给单片机以及其他需要 3.3 V 电压供电的设备供电，而其他 5 V 设备均由 USB 供电。USB 与 ME6211 电路如图 2 所示。

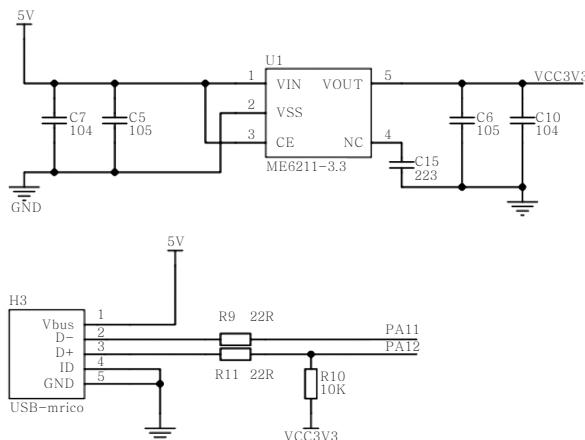


图 2 USB 与 ME6211 电源电路

### 2.2 温湿度检测电路

本文采用 DHT11 温湿度模块采集室内的温湿度。该传感器由电阻式感湿元件和测温元件组成，与 8 位单片机相连，传输距离可达 20 m 以上。DHT11 供电电压为 3~5.5 V，数据传输使用串行接口，采用单总线的数据格式，只用一根线就能实现双向传输，一次数据通信时间在 4 ms 左右。DHT11 发送数据时一共发送 5 个字节，首先发送 2 个字节的温度数据，前一个字节是整数部分，后一个字节是小数部分。然后再发送 2 个字节的湿度数据，和温度数据相同，前一个是整数，后一个是小数部分，最后再发送一个字节的校验和，用来检测数据传输是否正确<sup>[3]</sup>。DHT11 原理如图 3 所示。

**基金项目：**湖北工程学院新技术学院教学研究项目（项目编号：2022JY04）；湖北工程学院新技术学院科研项目基金资助（项目编号：2022KY08）。

**作者简介：**胡益峰（1998—），男，汉族，湖北襄阳人，本科，研究方向：嵌入式开发；丁岩岩（1989—），通讯作者，女，汉族，山东济宁人，硕士研究生，讲师，研究方向：嵌入式控制及通信技术。

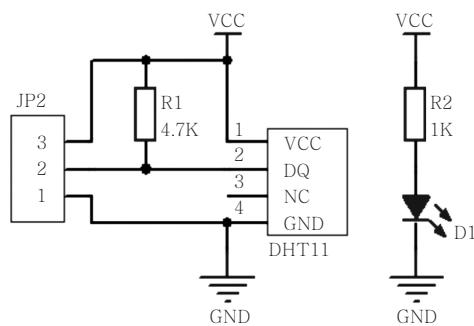


图 3 DHT11 原理

### 2.3 有害气体检测电路

有害气体检测电路用来检测室内的有害气体，如煤气、燃烧产生的烟雾，当检测到有害气体时，系统会报警并打开风扇迅速通风，保障室内人员的安全。

有害气体检测采用 MQ-135 传感器，该传感器的工作原理是当传感器所处的环境中存在污染气体时，传感器的电导率会随着空气中污染气体浓度的增加而增大。MQ-135 传感器可以检测多种有害气体，很适合放在室内作为家庭用空气污染报警器。由于 MQ-135 的工作电压为 5V，因此模拟口 AOUT 的输出电压范围在 0~5V，然而 STM32 单片机的 A/D 检测口只能测量 0~3.3V 的电压，超过这个电压就会烧坏 IO 口，因此在 IO 口加入了分压电路，将输出电压限制在 0~3.3V 之间。根据电阻分压公式  $V_o = \frac{R_1}{R_1 + R_2} \times V$  可知，要想使输入电压的 0~5V 转换成输出的 0~3.3V，需要满足  $R_1 : R_2 = 1 : 2$  的关系，因此电阻 R1 取 1.8k、R2 取 3.6k。气体检测电路，如图 4 所示。

系统报警功能采用蜂鸣器。蜂鸣器分为有源蜂鸣器以及无源蜂鸣器，有源蜂鸣器只要有电流通过时就会报警，而无源蜂鸣器则需要给它施加一个方波信号才会报警。本次设计使用有源蜂鸣器，并将蜂鸣器的正极引脚接在了 VCC 上，负极引脚接在单片机的 IO 口上。当检测到有害气体时，蜂鸣器就会发出报警声。

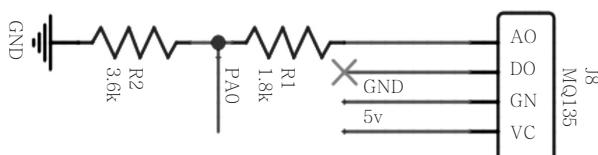


图 4 气体检测电路

### 2.4 风扇控制电路

本文使用 ADDA 风扇，采用继电器控制，使用三极管开关电路驱动继电器。三极管是模拟电子技术中最常用的器件之一，分为 NPN 三极管和 PNP 三极管。

当 PNP 三极管的基极电压低于 0.6V 时，三极管导通，而 NPN 三极管则需要基极电压高于集电极电压才可以导通。本次设计将三极管作为开关电路，让三极

管工作于饱和区与截至区。三极管基极电阻 R8 为基极提供电流，集电极上的电阻 R7 是 LED 灯的限流电阻，R7 与 LED 串联于继电器，保证电压相同。当基极电压为 5V 时，三极管导通 LED 灯亮，指示继电器已开，同时继电器闭合 5V 电压，接入风扇，风扇开始工作，此时输入 PWM 信号就可以控制风扇的转速。风扇控制电路，如图 5 所示。

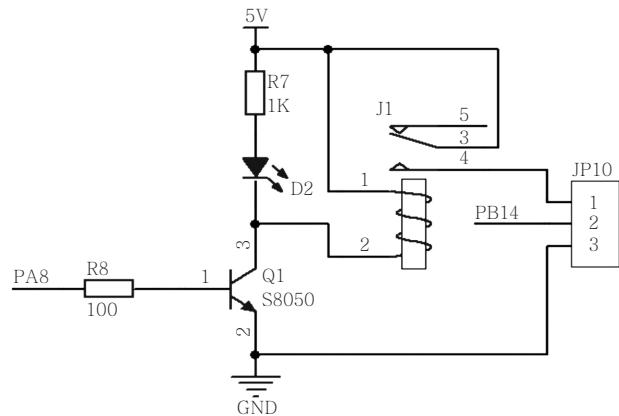


图 5 风扇控制电路

### 2.5 Wi-Fi 模块电路

Wi-Fi 模块选用乐鑫公司生产的 ESP8266-01S，具有性能稳定、集成度高、低功耗、内置处理器等优点。ESP8266 既可以当作热点使用，也可以作为从机接入别的网络。

ESP8266 支持 softAP 模式、station 模式、softAP+station 3 种模式。softAP 模式是将 ESP8266 当作一个无限接入点，让其他无线终端接入。station 模式则是作为无线终端接入其他的无线接入点。在本次设计中，主要使用 station 模式，将 ESP8266 作为无线终端，通过路由器接入互联网，通过阿里云服务器向 ESP8266 模块发送控制指令。

ESP8266 模块引脚，如图 6 所示。

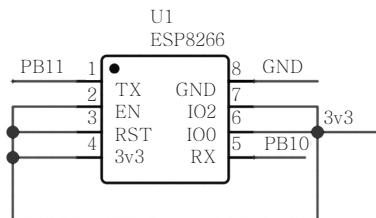
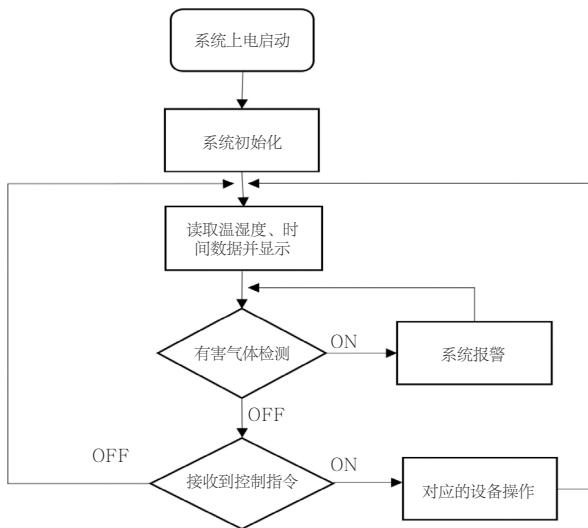


图 6 ESP8266 模块引脚

### 3 系统软件设计

软件设计主要包含显示模块设计、温湿度模块设计、Wi-Fi 模块设计、报警模块设计等。具体工作流程是系统上电后首先完成一系列外设的初始化，等待 ESP8266 联网，网络连接成功后发送指令，读取当前时间，开启 STM32 的 RTC 功能，实时读取显示在 OLED 显示屏的 RTC 时间和 DHT11 的温湿度数据。然后系统通过

串口等待 ESP8266 发送的控制指令。在接收到指令后，系统控制各模块做出对应的动作。系统在读取时间、温度和湿度数据的同时，还会通过 STM32 的 ADC 功能检测空气质量模块的输出电压，一旦空气质量模块的输出电压发生突变且电压数值超过阈值，就意味着产生了有害气体，系统会立刻发出报警声并开启风扇，排出室内的有害气体。系统总体程序设计如图 7 所示。



### 3.1 报警模块程序设计

报警模块由 MQ-135 空气质量检测模块、蜂鸣器及风扇组成。本系统通过 AD 转换获取 MQ-135 数据，判断是否存在有害气体。AD 转换由软件开启，软件每给一个信号，ADC 就启动一次转换。由于 AD 采集会存在波动，因此选用算数的平均滤波以提高数据的稳定性。当检测到电压突变时，说明存在有害气体，系统将会报警提示，并自动打开风扇全速运行，以最快的速度将室内的有害气体排出。

### 3.2 Wi-Fi 模块程序设计

本系统的 Wi-Fi 模块采用 ESP8266，它的主要作用是接入互联网，并与 STM32 主控进行通信，采用 Arduino 上传数据。

ESP8266 的程序逻辑为初始化，包括 Serial.begin(115200) 设置串口波特率为 115200，Blinker.begin(auth, ssid, pswd) 设置用户名、密码与设备码接入 Wi-Fi，Blinker.setTimezone(8.0) 设置时区为东八区。初始化完成后，等待 STM32 发送指令，当 ESP8266 接收到指令后，就会读取当前时间。ESP8266 除了会等待 STM32 发送来的数据外，还会接收来自阿里云平台发送的控制指令，并将这些指令转发给 STM32。

## 4 系统测试

将系统的全部模块安装完成，对系统上电，除

MQ-135 (MQ-135 发热是由于它内部的发热芯在加热) 以外，其他器件均无明显发热现象。系统启动后，OLED 上显示正在联网，等待网络连接完成后，OLED 显示当前时间和温湿度，达到预期的要求。接着在 MQ-135 前放出一点打火机的气体，系统报警并打开风扇以全速运行，待打火机的气体大部分消散完后，蜂鸣器停止报警，风扇停止工作，即这一部分也达到了预期要求。最后，测试语音控制的功能，通过手机 AI 说出语音控制指令，系统会做出对应的动作。本次设计的控制指令有开灯、关灯、打开风扇、关掉风扇这几个命令，经过测试，这些命令都可以实现。

指令识别测试表如表 1 所示。获取时间测试表，如表 2 所示。

表 1 指令识别测试表

次数	开灯	关灯	打开风扇	关掉风扇
测试次数	10	10	10	10
成功次数	10	10	10	10

表 2 获取时间测试表

当前时间	2020/11/23	2020/11/23	2020/11/23	2020/11/23
	12:31:20	12:36:46	16:22:02	17:31:12
获取时间	2020/11/23	2020/11/23	2020/11/23	2020/11/23
	12:31:42	12:37:12	16:22:26	17:30:48

由表 1 和表 2 可以看出，整个系统的识别率很高，几乎不会出现错误，但是在获取时间上大概会有 30 s 左右的误差。这是由于它获取的是云服务器的时间，因此会存在时间误差。

## 5 结语

智能家居的未来发展前景非常广阔、市场空间也很大，随着 5G、物联网和人工智能等相关技术的发展，整个行业将会继续获得新的发展机会。本文以嵌入式微处理器为控制核心，利用 ESP8266、手机 AI、MQ-135、DHT11、OLED 及报警系统电路构建了智能家居控制系统。通过 MQ-135 和 DHT11 传感器采集的数据，实现了自动监测并及时反馈和处理，语音通过手机 AI 控制 ESP8266，实现了家居的智能化。实验结果表明，设计的控制系统能够很好地实现智能家居的功能。

## 参考文献

- [1] 英钊, 买永峰, 王艳敏. 智能家居语音控制系统发展展望 [J]. 智能建筑电气技术, 2019, 13(1):30–33.
- [2] 张瑞英, 荆学海. 基于物联网的智能家居设计与实现 [J]. 中国建材技, 2018, 27(1):128–129.
- [3] 盛振涛, 颜铿洋. 基于语音识别的智能家居助手设计与制作 [J]. 数码界, 2019(8):134.