

基于物联网的温湿度盒子

The Third Team

¹ 物理系 西安理工大学

² applied physics
Xi'an University of Technology

2016 年 7 月 15 日



Contents

- ① 概述
- ② 系统硬件设计与实现
- ③ 系统软件方案

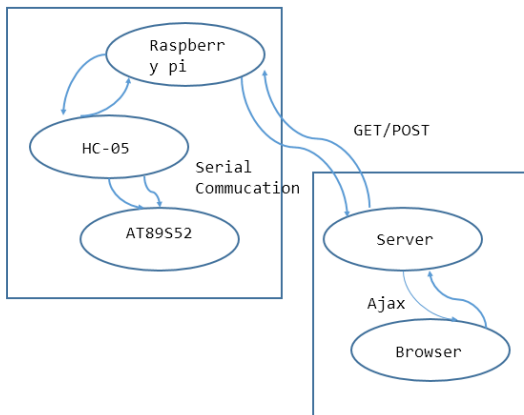


总体设计

系统采用的结构是：AT89S52+bluetooth+Raspberry
Pi+JSON+Python+yeelink



框架图示



系统结构及原理

主控模块

⊛ AT89S52（最小单片机系统）



系统结构及原理

主控模块

- ⊗ AT89S52（最小单片机系统）

传感器模块

- ⊗ DHT11 温湿度传感器
- ⊗ MQ-2 型烟雾传感器



系统结构及原理

主控模块

- ⊛ AT89S52（最小单片机系统）

传感器模块

- ⊛ DHT11 温湿度传感器
- ⊛ MQ-2 型烟雾传感器

串行通信模块

- ⊛ HC-05 蓝牙模块



系统结构及原理

主控模块

- ⊛ AT89S52（最小单片机系统）

传感器模块

- ⊛ DHT11 温湿度传感器
- ⊛ MQ-2 型烟雾传感器

串行通信模块

- ⊛ HC-05 蓝牙模块

数据通信

- ⊛ Raspberry Pi 开发板



系统原理图

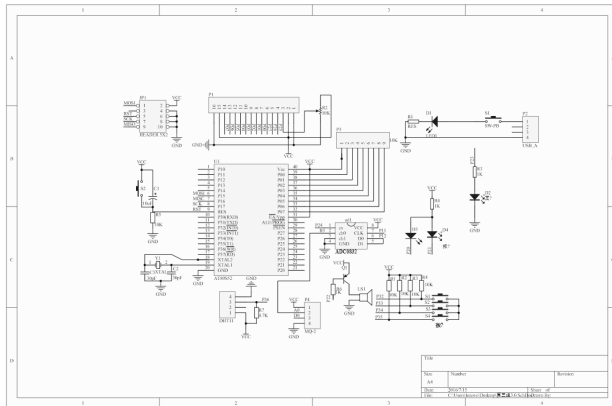
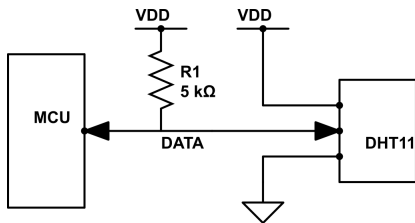


Figure: 原理图

传感器模块

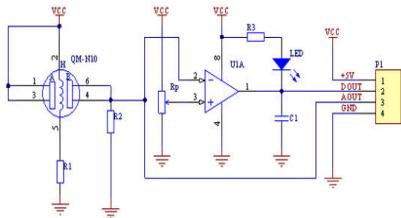


DHT11

- ① 测量范围湿度 20-90 温度 0 – 50°C
- ② 供电电压 3.3-5.5V DC
- ③ 全部校准，数字输出
- ④ 卓越的长期稳定性
- ⑤ 单总线串行接口的通信方式



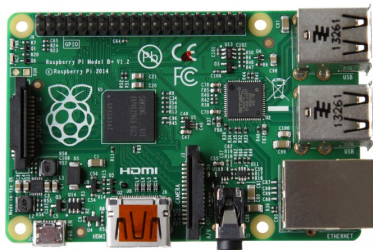
传感器



MQ-2 烟雾传感器

- ※ 采用的气敏材料是在清洁空气中电导率较低的二氧化锡 (SnO_2)。当它置于敏感气体环境中时，其电导率随空气中敏感气体浓度的增大而增大。该传感器对于液化气、丙烷和氢气的灵敏度非常高

Raspberry Pi



Raspberry Pi

- ① Linux 系统
- ② 体积仅信用卡大小，搭载 ARM 架构处理器，运算性能和智能手机相仿。
- ③ 可以用诸如 Python、Ruby 或 Bash 来执行脚本



蓝牙模块



HC-05

- ① 主从一体蓝牙模块
- ② 快闪表示没有蓝牙连接，慢闪表示进入 AT 模式，双闪表示蓝牙已连接并打开了端口
- ③ 可通过 AT 命令切换为主机或者从机模式，可通过 AT 命令连接指定设备

方案选择

- ① 数据通信格式选择 JSON。



方案选择

- ① 数据通信格式选择 JSON。
- ② Raspberry 开发环境为 Linux。



方案选择

- ① 数据通信格式选择 JSON。
- ② Raspberry 开发环境为 Linux。
- ③ 开发语言为 Python, 使用到 pySerial。



软件设计

在本地我们需要解决的问题可以如下描述，通过蓝牙模块（从机）不断将读取的温湿度数据发送到树莓派上的蓝牙模块（主机）上，我们可以使用下面的伪 C 描述。

```
//*****中断服务函数块*****
```

```
void ComSer() interrupt 4 //串口中断服务函数（串口数据接收）
```

```
{
```

```
if(RI)
```

```
{
```



软件设计

```
RI=0;
}
}
//****串口字符（字符串）发送块*****
void putchar(unsigned char n)
{
    SBUF=n;
    while(!TI);
    TI=0;
}
void puts(unsigned char *q)
```



软件设计

```
{  
    while(*q)  
        putchar(*q++);  
}  
void Enter()  //换行函数  
{  
    putchar(0x0d);  
    putchar(0x0a);  
}  
void DispNum(unsigned char n)  
{
```



软件设计

```
unsigned char t[8]={0};  
t[0]=n/10+'0';  
t[1]=n%10+'0';  
//t[2]=0;  
putchar(t[0]);  
    putchar(t[1]);  
        // putchar(t[2]);  
// puts(t);  
    Enter();  
}
```



软件设计

```
//*****初始化函数块*****
```

```
void init_com() //寄存器初始化函数
```

```
{
```

```
    SCON=0x50; //串行口工作方式1，多机通信，允许接收
```

```
    IE|=0X90; //定时器T1和总中断允许
```

```
    PCON=0X80; //SMOD为1
```

```
    TMOD|=0X21; //T1工作在方式2，T0工作在方式1
```

```
    TH1=250; //T1高八位赋值
```

```
    TR1=1; //T1打开
```

```
    TR0=1; //T0关闭
```



软件设计

```
void main() {  
While(1)  
{  
read_TRH();  
    // puts("Wendu:");  
    DispNum(TH_data);  
    // puts("Shidu:");  
    // DispNum(RH_data);  
    // Enter();  
delay_ms(500);  
}
```



python 串口通信

```
# -*- coding: utf-8 -*-  
import serial  
import time  
import os  
# 打开串口  
ser = serial.Serial("/dev/ttyAMA0",baudrate=9600,timeout=8.0)  
def main():  
    #true=false  
    while True:  
        # 获得接收缓冲区字符  
        count = ser.inWaiting()
```



python 串口通信

```
if count != 0:
    # 读取内容并回显
    recv = ser.read(count)
    ser.write(recv)
    print("tem",recv)
# 清空接收缓冲区
ser.flushInput()
# 必要的软件延时
time.sleep(0.1)
#数据转换成JSON格式
#res = '{"value":%s}' %recv
```



python 串口通信

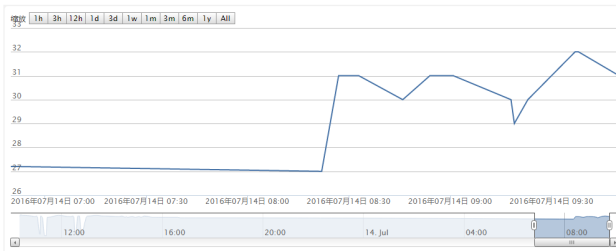
```
#output = open('/home/pi/python/mytest/tem_data.txt', 'w')
#写数据到文本文件中
#output.write(res)
#关闭文件
#output.close

if __name__ == '__main__':
    try:
        main()
    except KeyboardInterrupt:
        if ser != None:
            ser.close()
```



yeelink

temtest



最新数据

31

摄氏度(°C)

更新时间: 09:50:04 2016-07-14



IOT

系统采用的结构是:AT89S52+HC-05+Raspberry Pi+Ajax+Python+Yeelink, AT89S52 与 Raspberry Pi 通过蓝牙串行通信, 相互传输所需要的数据。Raspberry Pi 将温湿度数据上传于 Yeelink 对应的接口, 接口可以在互联网上被访问, 互联网上的其他设备便可以通过网络来访问这些设备。移动端 UWP 应用也可连入本地终端通过蓝牙进行数据的读取。



总结

One More Thing



Thanks

Thanks for all the people who helped me



Q&A

