****

**嵌入式应用系统综合实训**

**期末设计**

|  |  |
| --- | --- |
| **任务名称：** | **树莓派采集存储展示DHT22数据** |
| **学 院：** | **数学与信息学院** |
| **专 业：** | **2017级物联网工程** |
| **学 号：** | **116042017058** |
| **姓 名：** | **赵镇峰** |
| **任课老师：** | **张美平** |

**2020年6月**

1. **嵌入式实训大作业任务：**  
   代码组成  
   1）编写嵌入式Linux端程序  
   2）设计web服务器与web端程序  
   具体要求  
   1、结合分阶段任务的mqtt、http、mysql数据库、flask服务器的相关知识设计课程作业；  
   2、使用openwrt或者直接在ubuntu linux虚拟机上运行程序模拟嵌入式设备；  
   3、使用虚拟串口软件以及串口调试助手模拟传感器设备，实现传感器数据的上传与传感器控制指令的接收，控制指令与传感器数据上报格式自定义；  
   4、使用mqtt或http实现终端与服务器双向通讯，  
   0）、搭建mqtt、flask、MySQL服务器；  
   1）、用mqtt实现对终端控制指令的订阅；  
   2）、用mqtt或http实现终端采集到的数据向服务器上报；  
   3）、传感器数据能存入数据库；  
   4）、能在web页面查看实时数据，利用websocket mqtt显示最新实时数据；  
   5）、能web页面查看历史数据，  
   6）、能通过web页面发布mqtt对硬件的控制指令，利用websocket mqtt实时发布控制指令；  
     
   另外如果用树莓派，树莓派GPIO上接传感器模拟传感器设备，GPIO接口外接简易控制器（如led二极管）模拟设备控制功能
2. **设计功能：**

树莓派驱动DHT22温湿度传感器

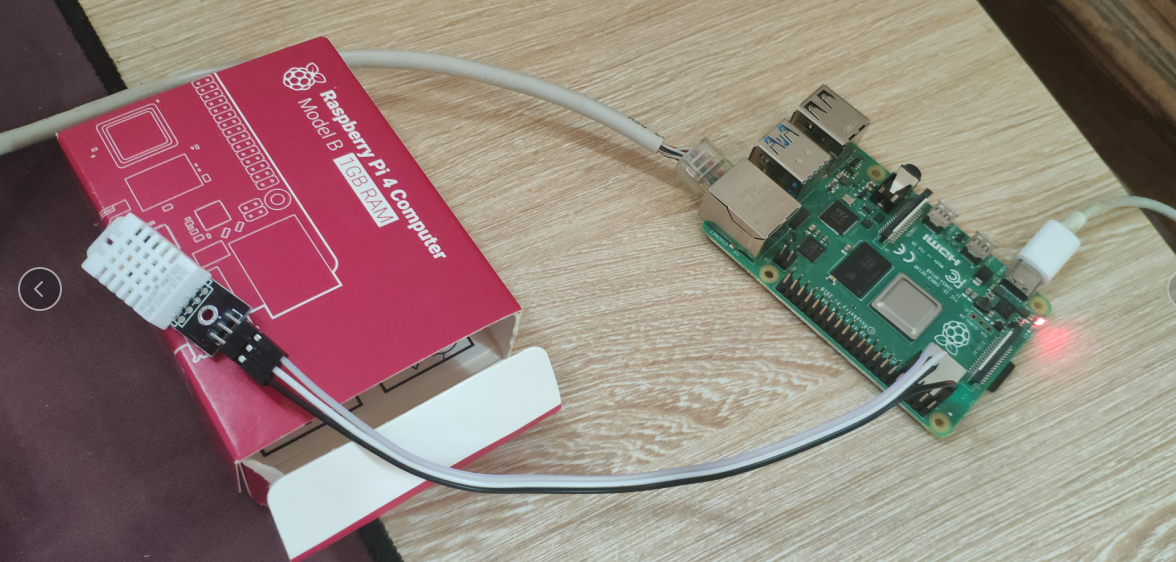
树莓派GPIO上接dht22传感器设备，将采集到的信息存入mysql数据库，并且用http协议实现终端采集到的数据向服务器上报onenet云平台，并设计web界面，使用flask提供后台服务，页面使用echarts展示存入mysql的温湿度数据和实时数据。

VCC接上3V3，可以选择1口或者17口

DATA接上GPIO口，本次选GPIO4，第7口

GND接上接地口，本次选第14口

接线完成后，此时灯亮。



1. **具体实现：**
2. 采集温湿度数据：

通过import Adafruit\_DHT库来实现对gpio口传感器数据的采集。

def rasp\_get(channel):

sensor = Adafruit\_DHT.DHT22

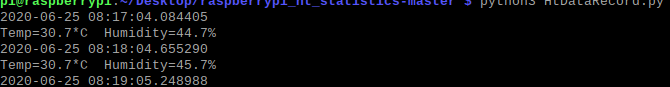
pin = 4 #GPIO4

humidity, temperature = Adafruit\_DHT.read\_retry(sensor, pin)

print("温度：", temperature, "摄氏度，湿度：", humidity, "%")

data = [0, temperature, humidity]

通过此函数将采集的数据存储于data中。



1. 将历史采集数据存储于mysql中：python3 HtDataRecord.py

通过import pymysql库来实现温湿度数据的存储。

Ht.sql：

CREATE TABLE `ht` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`create\_time` datetime DEFAULT NULL,

`site` varchar(20) DEFAULT NULL COMMENT '位置',

`temperature` float(3,1) DEFAULT NULL,

`humidity` float(3,1) DEFAULT NULL,

PRIMARY KEY (`id`) USING BTREE,

KEY `ht\_create\_time\_index` (`create\_time`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=1246 DEFAULT CHARSET=utf8 COMMENT='温湿度表';

创建一个sql库ht

con = pymysql.connect(

host = 'localhost',

port = 3306,

user = 'pi',

password = '123456',

db = 'ht',

charset = 'utf8'

在在树莓派中访问ht库，用以读写数据

if humidity is not None and temperature is not None:

cur = con.cursor()

print(datetime.datetime.now())

print('Temp={0:0.1f}\*C Humidity={1:0.1f}%'.format(temperature, humidity))

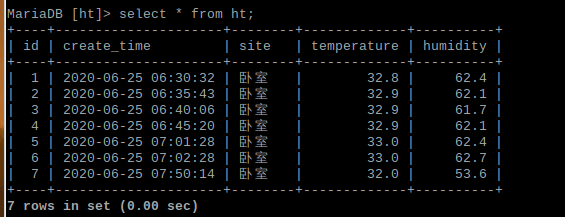
cur.execute("insert into ht(create\_time,site,temperature,humidity) value(now(),'%s','%s','%s')" %(site,temperature,humidity))

con.commit()

cur.close()

time.sleep(60)

将数据存入mysql中，每隔60s采集一次。



1. 使用http协议将实时温度数据上传至onenet云平台：python3 DTH-inoput.py

apiurl = ["http://api.heclouds.com/devices/606846327/datapoints",

# 用户密码

apiheaders = {'api-key' : 'vhl4ll0Ekjow59REecUKsEAv=8o='}

输入云平台的API地址与APIkey

json\_data\_turn1 = json.dumps(json\_data1)

json\_data\_turn2 = json.dumps(json\_data2)

print("开始上传数据")

time.sleep(12)

r1 = requests.post(apiurl[0], data=jdata, headers=apiheaders)

time.sleep(12)

r2=requests.post(apiurl[1], data=json\_data\_turn2.encode(), headers=apiheaders)

print(r1)

print(r2)

print("完成上传数据")

对云平台上传数据。

1. 使用flask提供后台服务，在web（statistics\_template.html）页面展示当前温湿度数据以及历史温湿度数据。Python3 Server.py

def get\_jsondata(range=1,n=20):

con = conn\_db()

cur = con.cursor()

result = cur.execute("SELECT id,temperature,humidity,create\_time FROM ht WHERE DATE\_FORMAT(create\_time,'%%i')%%%s = 0 AND create\_time >= DATE\_SUB(NOW(), INTERVAL %s MINUTE) order by create\_time " %(range,n))

u=cur.fetchall()

con.close()

print(u)

从数据库中访问历史数据。

$(document).ready(function () {

setNowHT();

getData(5,1440);

//console.log(app.xTime);

//console.log(app.yTvalues);

//console.log(app.yHvalues)

var t2 = window.setInterval(function() {

setNowHT();

},20000)

//window.clearInterval(t2)  // 去除定时器

});

function setNowHT() {

$.ajax({

url:'/getNowHT',

type:'POST',

dataType:'json',

success:function(data) {

console.log(data);

$("#now\_temp").text(data.temperature + " ℃");

$("#now\_hum").text(data.humidity + " %");

$("#now\_time1").text(new Date().Format("yyyy-MM-dd hh:mm:ss"));

$("#now\_time2").text(new Date().Format("yyyy-MM-dd hh:mm:ss"));

}

});



发送ajax请求，从后台获取json数据并展示实时时间与数据。

function getData(range\_args,time\_args) {

$.ajax({

url:'/getData',

data:{range:range\_args,time:time\_args}, //range 每多少分钟取一个点， time：最近多少范围内的。

type:'POST',

//async:false,

dataType:'json',

success:function(data) {

console.log(data);

//app.xTime = data.xTime;

//app.yTvalues = data.yTvalues;

//合成二维数组

var array1 = data.xTime;

var array2 = data.yTvalues;

var array3 = data.yHvalues;

var arrayAll=[]; //温度

var arrayAll2=[]; //湿度

for(var i=0;i<array1.length;i++){

var time = parseInt(array1[i]\*100000/100);

arrayAll.push([time,array2[i]]);

arrayAll2.push([time,array3[i]]);

}

myChartT.clear();

myChartH.clear();

myChartT.setOption({

title: {

text: '温度图'

},

xAxis: {

name:'时间',

type: 'time'

//data: data.xTime time类型时，无需数据

},

yAxis: {

type: 'value',

axisLabel: {

formatter:'{value} ℃'

}

},

//用于添加框选缩放功能

/\*

toolbox: {

show: true,

feature:{

dataZoom:{

realtime: false,

yAxisIndex: 'none',

},

restore: {},

}

}, \*/

dataZoom:

[

{type: 'inside'}, //用于添加滚轮缩放

{type:'slider' }, //用于添加滑动条缩放，

],

series: [{

name:"温度",

symbol:'none', //设置不显示数据的小圆点

smooth:true,

data: arrayAll,

type: 'line'

}],

tooltip : {

trigger: 'axis',

axisPointer: {

type: 'cross'

},

formatter:function(parmas){

//console.log(parmas[0].value[0] + "," + parmas[0].seriesName + "," + parmas[0].value[1]);

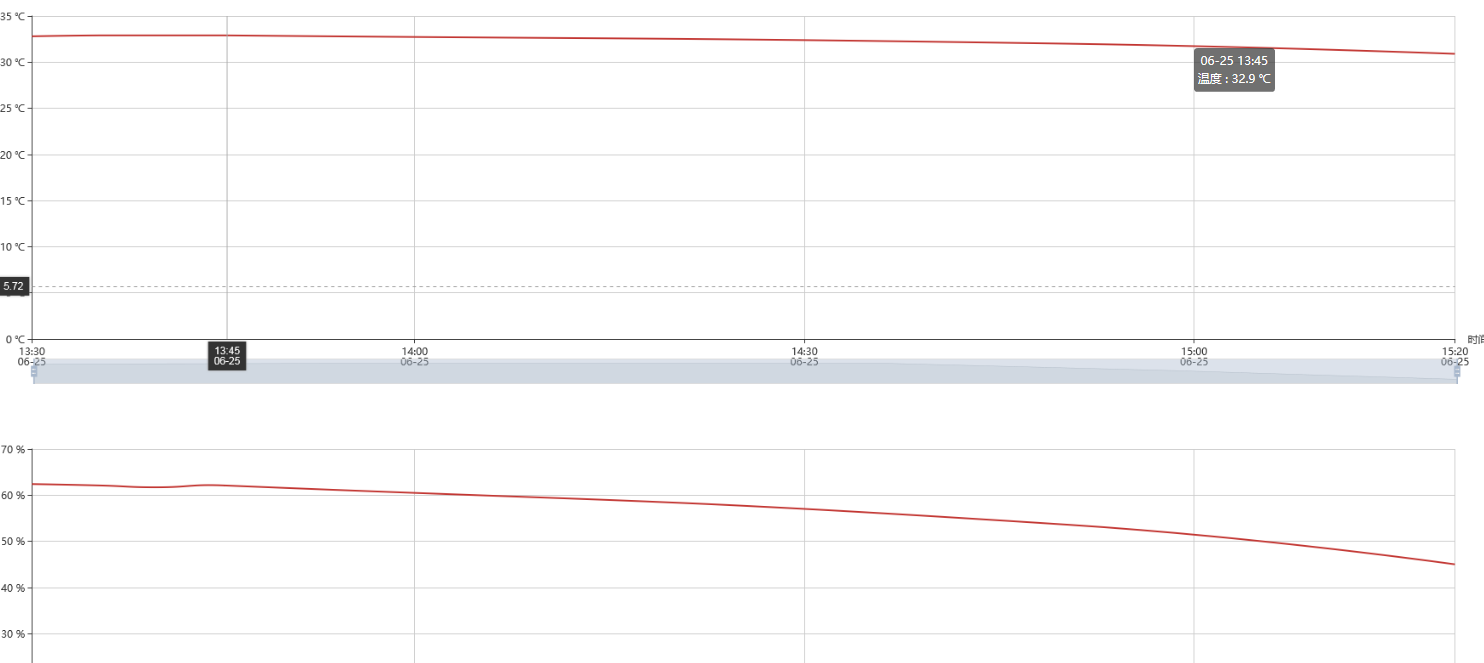
return echarts.format.formatTime('MM-dd hh:mm', parmas[0].value[0])+ "<br>" + parmas[0].seriesName + " : " + parmas[0].value[1] + " ℃";

}

}

},true);

在web页面展示数据曲线图



5.设置开机自己启动：`vi /etc/rc.local`

ntpdate ntp.aliyun.com &

/etc/init.d/mysql start &

sudo python3 /home/pi/raspberrypi\_ht/Server.py &

sleep 10

sudo python3 /home/pi/raspberrypi\_ht/HtDataRecord.py &

exit 0