



深圳四博智联科技有限公司

---

利用 **Arduino** 和 **wifi** 将温度传送至云端



二〇一五年十二月

### 3.1 问题描述：如何利用 Arduino 和 wifi 将温度传感器的数据上传到服务器端

在这一讲中，我们将重点阐述如何基于 Arduino 控制板利用 wifi 模块上传您周边的温度至服务器端，然后在服务器端有个客服端。这样，就算您远在大西洋彼岸的任何国家，都能查阅到自己亲人身边的温度。当然，您也可以利用多种类型的传感器，查阅您需要的信息。

### 3.2 硬件与软件

本次实验所需要的装备如表 3-1 所示。

表 3-1：所需材料

序号	名称	数量	作用	备注
1	Arduino 软件平台	1	提供平台	
2	服务器端	1	收集数据	
3	Wifi 模块	1 块	无线通信	
4	Arduino 扩展板	1 块	连接 wifi	各种版本均可
5	USB 转串口 RS232	1 条	转换连接	
6	5V1A 电压适配器	1 个	提供电压	
7	天线	1 个	发送 wifi 信号	可选
8	HTC 手机	1 部	提供 wifi 热点	
9	客服端	1	查看数据	

本次实验所需的硬件材料如后图 3-1 所示。可在深圳四博智联淘宝商城购的：

<https://szdoit.taobao.com/>

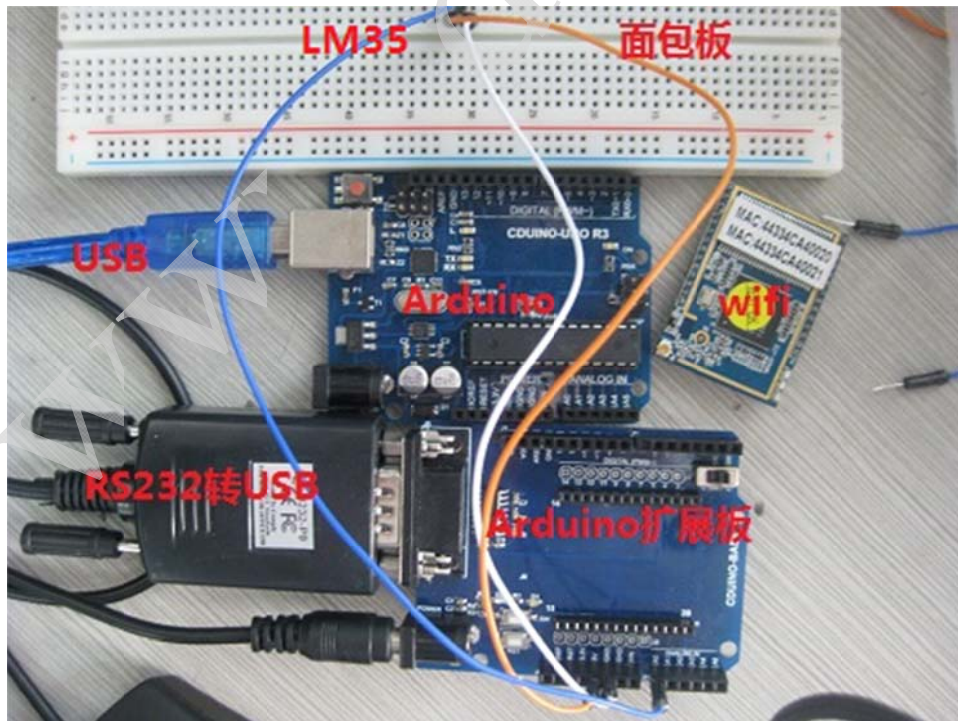


图 3-1 硬件实物

### 3.3 工作原理

本次实验的主要工作原理如图 3-2 所示。

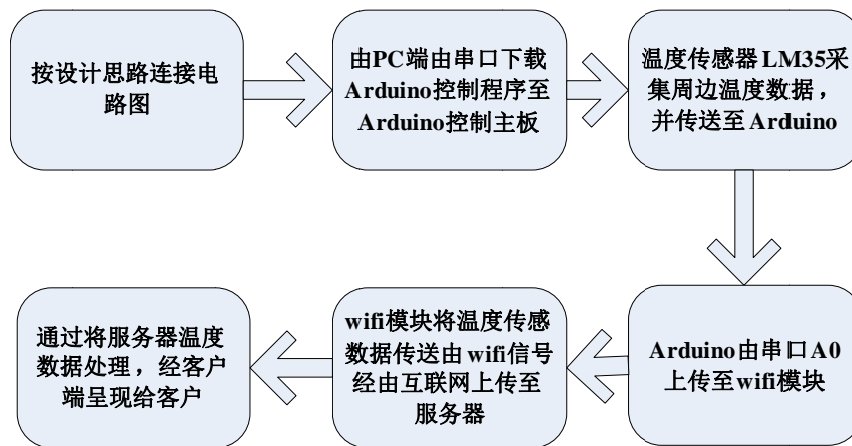


图 3-2 工作原理

### 3.4 实验分析

(1) **画实验原理图：**按照设计思路，画出实验原理图，如图 3-3 所示。这是从前一篇的第 8 讲中抽取的一个图，为了读者查阅的方便，在这再次给出这个图。但这个原理图中，由于画法的原因，还缺少 Arduino 的扩展板和 wifi 模板。但利用传感器 LM35 来采集数据的原理是一样的。

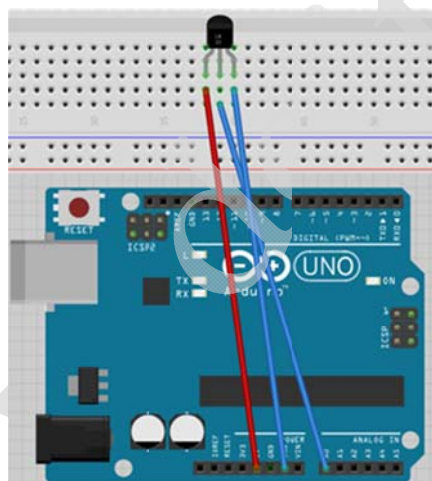
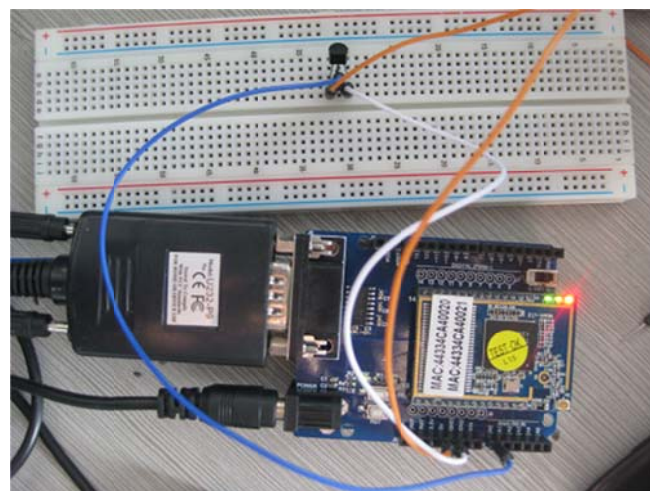


图 3-3 实验原理

(2) **实物连接：**按照原理图，连接好电路，如图 3-4 所示。



3-4 实验连接图

(3) **wifi 网络连接**: 当连接 Arduino 主板时, 就已经可以收到 wifi 信号了, 但是为了稳定, 最好用专用充电器连接 Arduino 扩展板。你的电脑将会搜索到有 wifi 模块发出的 wifi 信号 HI\_LINK\_0021 (或者 HI\_LINK\_XXXX 可参考第 2 讲得内容)。双击连接 HI\_LINK\_0021wifi 信号 (默认密码 12345678)。若不是第一次使用的, 为了确保数据配置的正确性, 需要将 wifi 模块恢复出厂默认设置 (按住 Arduino 扩展板上的 RST 按钮 6s, 然后断电重启即可, 具体也可参考第 2 讲的内容)。这一步相当于咱们用打开手机的 wifi 信号上网一样。此时的 wifi 模块打开 wifi 信号, 连接手机的 wifi 热点信号上网。

(4) **建立 wifi 热点**: 由于本次实验环境没有其他 wifi, 所以采用的是 HTC 手机建立 wifi 热点, 用 wifi 路由器也是一样的。打开手机的“WLAN 热点”, 记住此时记得打开手机的 3G 网络信号, 以便由温度传感器 LM35 采集的数据上传至云端的服务器。本次实验 WLAN 热点名称“HTCTest”, 密码“12345678”。也可以用手机默认的用户名和密码, 不过, 为了输入的方便, 可以更改手机默认的用户名和密码。

(5) **网络数据配置**: 当成功连接 wifi HI\_LINK\_0021 网络后, 打开浏览器, 输入地址 192.168.16.254, 输入用户名和密码 (均为 admin) 进入数据配置, 如图 3-5 所示。

The screenshot shows a web-based configuration interface for a device. It has two main sections: 'NetMode' and 'Serial Configure'. The 'NetMode' section includes fields for SSID, Encrypt Type, Password, and IP Type. The 'Serial Configure' section includes fields for Baud rate, Serial Framing Length, Serial Framing Timeout, Network Mode, Remote Server Domain/IP, Locale/Remote Port Number, Network Protocol, and Network Timeout. Red boxes and arrows highlight specific settings with Chinese annotations: 'WIFI (CLIENT)-SERIAL' is selected for NetMode; SSID is 'HTCTest'; Password is '12345678'; Baud rate is updated to '115200, 8, n, 1'; Network Mode is changed from 'server' to 'Client'; Remote Server Domain/IP is changed from '192.168.11.245' to '60.10.1.205'; and the Remote Port Number is changed from '8080' to '80'.

Field	Current Value	Updated Value	Annotation
NetMode	WIFI (CLIENT)-SERIAL	WIFI (CLIENT)-SERIAL	选择wifi (client)
SSID	HTCTest	HTCTest	手机上的SSID
Encrypt Type	WPA2 AES	WPA2 AES	
Password	12345678	12345678	手机热点的密码
IP Type	DHCP	DHCP	
Serial Configure: Baud	115200, 8, n, 1	115200, 8, n, 1	Baud为115200
Serial Framing Lenth	64	64	
Serial Framing Timeout	10 milliseconds	10 milliseconds (< 256, 0 for no timeout)	
Network Mode	server	Client	选择Client
Remote Server Domain/IP	192.168.11.245	60.10.1.205	输入服务器端IP
Locale/Remote Port Number	8080	80	服务器端口
Network Protocol	tcp	TCP	
Network Timeout	0 seconds	0 seconds (< 256, 0 for no timeout)	

图 3-5 网络数据配置  
坦克配置



NetMode:	WIFI (CLIENT) - SERIAL ▾	
SSID:	S4test	Scan
Encrypt Type:	WPA2 AES ▾	
Password:	12345678	
IP Type:	DHCP ▾	

	Current	Updated
Serial Configure:	115200,8,n,1	9600, 8, n, 1
Serial Framing Lenth:	64	64
Serial Framing Timeout:	10 milliseconds	10 milliseconds (< 256, 0 for no timeout)
Network Mode:	server	Client ▾
Remote Server Domain/IP:	192.168.11.245	182.92.178.210
Locale/Remote Port Number:	8080	6005
Network Protocol:	tcp	TCP ▾
Network Timeout:	0 seconds	0 seconds (< 256, 0 for no timeout)

在配置数据时注意红色框框标记部分，其他按默认设置即可。几点说明

- 1) 此时的 wifi 模块相当于一个采集输送单元，将采集到的数据由 wifi 模块送至服务器端；
- 2) SSID 和 Password 指的是手机（或 wifi 路由器）的 SSID 和 Password；
- 3) Remote ServerDomain/IP: 指的是咱们要将采集到的温度数据要存放的服务器 IP 地址；
- 4) Locale/Remote Port: 是访问服务器的地址。

配置完成后，点击“Apply”。这时，再检查手机的“WLAN 热点界面”最下方的“管理用户处”，有“1 连接的用户”。此时，表示 wifi 模块已经连接上了手机的热点，也即连接上了 internet。

#### (6) 编写 Arduino 代码:

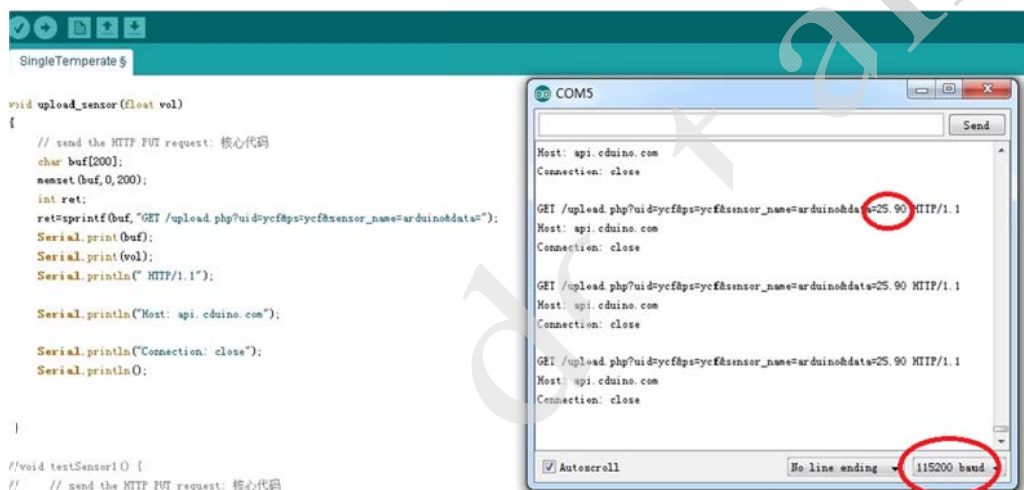
```
void setup()
{
    Serial.begin(115200); // 设置串口波特率
}

void loop()
{
    int n = analogRead(A0); // 定义 A0 口为接收电压信号数据
    float vol = n * (5.0 / 1023 * 100); // 电压信号数据温度转换
    upload_sensor(vol); // 调用子函数
    delay(5000);
}

void upload_sensor(float vol)
{
    // send the HTTP PUT request: 核心代码
    char buf[200];
    memset(buf, 0, 200);
```

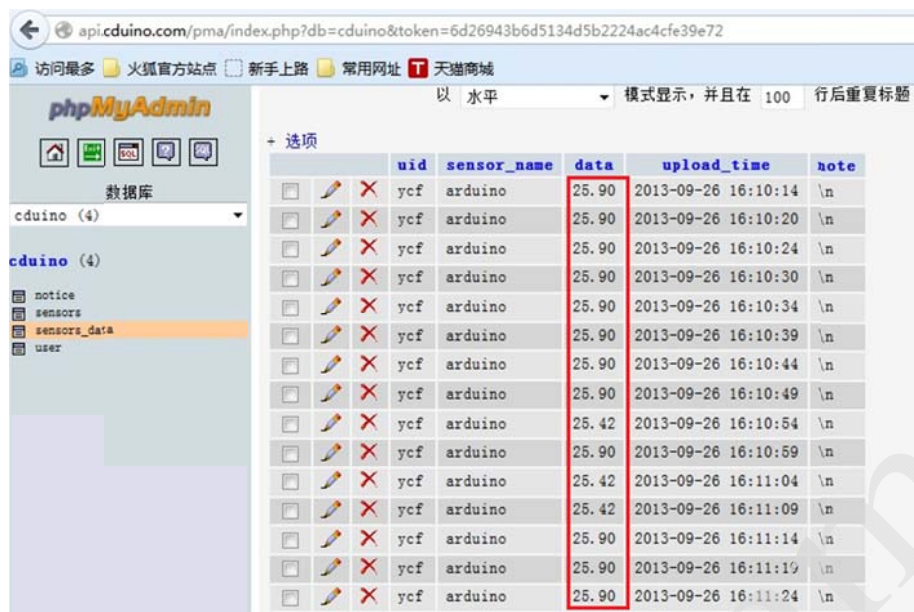
```
int ret;
ret=sprintf(buf,"GET /upload.php?uid=ycf&ps=ycf&sensor_name=arduino&data=");
//设置协议
Serial.print(buf); //暂存至 buf
Serial.print(vol); //调用温度数据 vol
Serial.println(" HTTP/1.1"); //HTTP 协议名称
Serial.println("Host: api.cduino.com"); //设置服务器地址
Serial.println("Connection: close"); //数据传输完毕，连接关闭
Serial.println();
}
```

将上述代码下载到 Arduino 主板上，注意，在 Arduino 的扩展板上有个开关，在下载数据时，请将拨至外侧（O）。这是因为连接串口时，有可能会影响数据的下载。同时打开串口，如图 3-6 注意红色的标记。此时的环境温度为 25.90。然后这个温度将通过 wifi 模块连接手机热点上的 wifi 信号上传至服务器端。



3-6 Arduino 串口监视器数据显示

(7) 终端数据采集：若前面 6 步全部成功，恭喜你！成功啦，由温度传感器 LM35 采集到的温度数据已经通过 wifi 模块经由手机 wifi 联网成功，一路过关斩将，历经千辛万苦，终于将温度传感器 LM35 采集到的温度数据上传至服务器端，如图 3-7 所示。



	uid	sensor_name	data	upload_time	note
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:14	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:20	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:24	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:30	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:34	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:39	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:44	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:49	\n
<input type="checkbox"/>	ycf	arduino	25.42	2013-09-26 16:10:54	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:10:59	\n
<input type="checkbox"/>	ycf	arduino	25.42	2013-09-26 16:11:04	\n
<input type="checkbox"/>	ycf	arduino	25.42	2013-09-26 16:11:09	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:11:14	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:11:19	\n
<input type="checkbox"/>	ycf	arduino	25.90	2013-09-26 16:11:24	\n

图 3-7 服务器端数据显示

比较图 3-6（用 Arduino 自带的串口显示器）和图 3-7（云端服务器），会发现，用 Arduino 自带串口监视器采集的温度数据和云端服务器温度数据是一致的，说明数据传送成功。在下一讲的内容，咱们将接着下一步工作，将云端服务器的数据很好地展现给别人看。

### 3.5 要点与总结

- （1）配置数据前，为确保数据能配置成功，请将 wifi 模块恢复出厂设置；
- （2）注意配置数据的准确性，尤其是关于手机和服务器端的 IP 地址设置；