**温湿度采集器说明书\_V1.1**

修订历史

|  |  |  |  |
| --- | --- | --- | --- |
| 版本 | 日期 | 作者 | 描述 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

目录

[1 设备框架 3](#_Toc17873)

[1.1 接线说明 3](#_Toc25977)

[1.2 温湿度采集器效果图 3](#_Toc21079)

[1.3 操作说明 4](#_Toc28784)

[1.3.1 设备上电 4](#_Toc20347)

[1.3.2 配置**联网信息** 4](#_Toc4102)

[1.3.3 指示灯说明 6](#_Toc22610)

[2 功能描述 6](#_Toc14553)

[3 ModBus通信协议 7](#_Toc9585)

[3.1 ModBus基本规则 7](#_Toc17876)

[3.1.1 Modbus命令格式 7](#_Toc16409)

[3.1.1.1 地址码(ADD) 7](#_Toc21803)

[3.1.1.2 功能码(CS) 7](#_Toc1987)

[3.1.1.3 数据区(DATA) 8](#_Toc25152)

[3.1.1.4 校验码(CRC) 9](#_Toc25835)

[3.2 设备实时采样数据寄存器地址（MODBUS功能码03） 10](#_Toc11194)

[4 附：CRC校验代码 11](#_Toc6985)

# 设备框架

## 接线说明

|  |  |  |
| --- | --- | --- |
| **说明** | **线色** | **备注** |
| VCC | 红 | DC 24V |
| GND | 黑 |
| RS485-A | 黄 |  |
| RS485-B | 蓝 |  |

## 温湿度采集器效果图



## 操作说明

### 设备上电

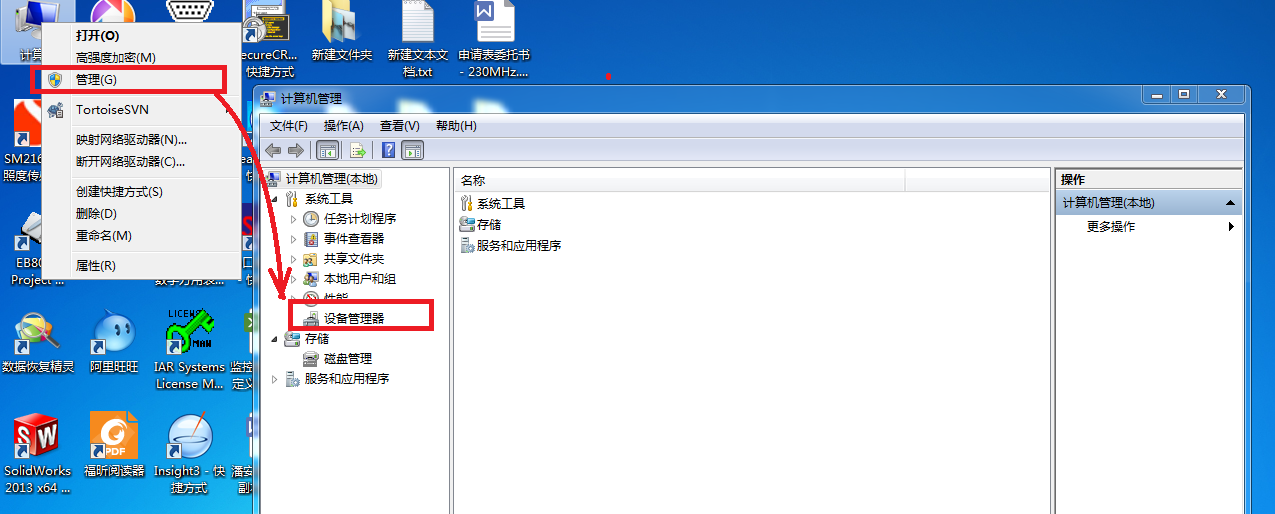
使用者手里拿到温湿度采集器，把天线装上。把SIM卡插入设备。然后接上电源，注意正负极。

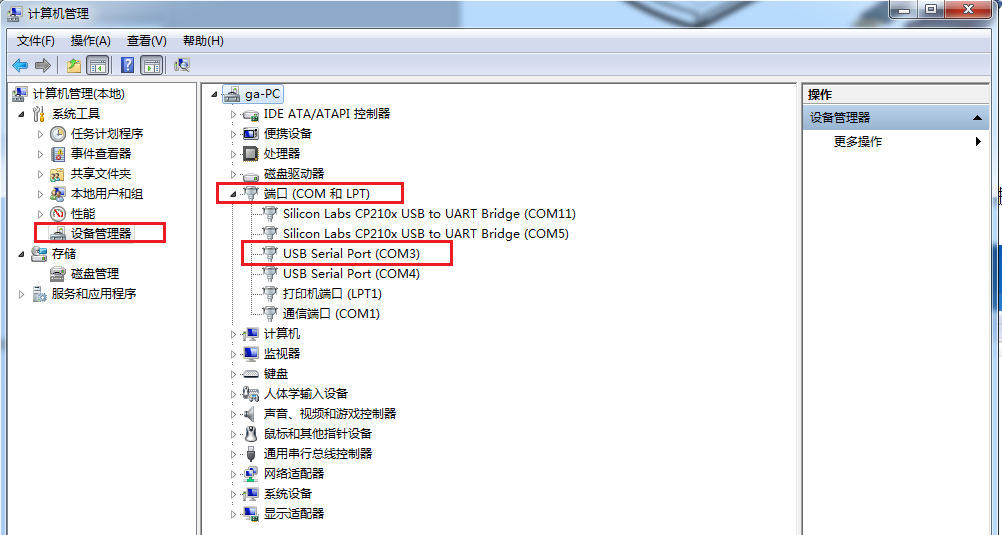
### 配置联网信息

使用温湿度的GPRS方式传输数据，则需要进行联网参数设置。电脑上通过usb转RS485线引出串口。把设备的RS485串口接到电脑端的RS485口上。通过配置工具配置参数。



在电脑中找的使用的串口：假设使用win7系统，在我的电脑右键点击“管理”，然后点击“设备管理”，在端口找到使用的USB转RS485串口，找到当前使用的USB转RS485为com3。





打开配置工具：安装好《恒星物联科技调试工具》，在电脑桌面上双击工具图标打开工具。



打开调试工具后，选择“温湿度传感器系列”：



选择点击“温湿度传感器系列”后进入“无线温度参数配置工具”配置界面。选择刚才找到的USB转485串口COM3.打开串口。设置gprs联网参数。

通信协议选择TCP,APN接入点“CMENT”(移动卡)，“uintnet”（联通卡），不支持电信卡。填入服务器后台的IP地址和端口，参数如下图所示：然后点击设置，设备断电重启就完成参数设置，设备可以正常运行了。



### 指示灯说明

采集器上有3个指示灯。

设备上电后运行指示灯会闪烁（一秒亮一秒灭循环）。Gprs联网指示灯，GPRS联网未成功时，联网指示灯快闪，联网成功后慢闪。

数据收发指示灯为双色灯，采集器通过GPRS往外发数据时绿灯闪烁，采集器通过GPRS收到数据时蓝灯闪烁；

# 功能描述

采集器同时支持温度和湿度采集，温度范围-40~120℃，精度0.5℃，湿度0~100%，精度2%。

采集器支持GPRS传输到服务后台。同时支持RS485输出数据。

采集器数据传输使用modbus RTU通信协议。

# ModBus通信协议

温湿度采集器通讯规约详细描述了本机射频通信的读、写命令格式及内部信息数据的定义，以便第三方开发使用。

## ModBus基本规则

所有RF射频通信回路都应遵照主/从方式。依照这种方式，数据可以在一个主站(如：PC)和子站（如：温湿度采集器）之间传递。

任何一次通讯都不能从子站开始。

在RF射频通信上的所有通讯都以“信息帧”方式传递。

如果主站或子站接收到含有未知命令的信息帧，则不予以响应。

### Modbus命令格式

|  |  |  |  |
| --- | --- | --- | --- |
| ADD | CS | DATA | CRC |
| 地址码 | 功能码 | 数据区 | 错误校验 |
|  |  |  |  |
| 1字节  8位 | 1字节  8位 | N字节  N×8位 | 2字节  16位 |

#### 地址码(ADD)

地址码为每次通讯传送的信息帧中的第一个数据帧(8位)，从1到255。这个字节表明由用户设定地址码的子机将接收由主机发送来的信息。并且每个子机都有唯一的地址码，并且响应回送均以各自的地址码开始。主机发送的地址码表明将发送到的子机地址，而子机发送的地址码表明回送的子机地址。

#### 功能码(CS)

功能码是每次通讯传送的信息帧中的第二个数据帧。ModBus通讯规约定义功能码为1～127(01H～7FH)。温湿度传感器利用其中的一部分功能码。作为主机请求发送，通过功能码告诉子机执行什么动作。作为子机响应，子机发送的功能码与主机发送来的功能码一样，并表明子机已响应主机进行操作。如果子机发送的功能码的最高位是1(功能码＞127)，则表明子机没有响应或出错。

下表列出的功能码都具体的含义及操作。

MODBUS部分功能码

|  |  |  |
| --- | --- | --- |
| 功能码 | 定义 | 操作 |
| 03H | 读寄存器 | 读取一个或多个寄存器的数据 |
| 10 H | 写寄存器 | 把一个16位二进制数写入寄存器 |

03，读寄存器

温湿度传感器采用ModBus通讯规约，利用通讯命令，可以进行读取点寄存器)。功能码03H映射的数据区的保持和输入寄存器值都是16位(2字节)。这样从温湿度传感器读取的寄存器值都是2字节。一次最多可读取寄存器数根据后续寄存器表给出。子机响应的命令格式是子机地址、功能码、数据区及CRC码。数据区的数据都是每2个字节为一组的双字节数，且高字节在前。

10，写寄存器：

主机利用这条命令把多点数据保存到温湿度传感器的存储器。Modbus通讯规约中的寄存器指的是16位(即2字节)，并且高位在前。这样温湿度传感器用一条命令保存的最大寄存器数根据后续寄存器表给出。

#### 数据区(DATA)

数据区随功能码不同而不同。由主机发送的读命令(03H)数据区与子机应答信息帧的数据区是不同的，由主机发送的写命令(10H)数据区与子机应答信息帧的数据区是完全相同。数据区包含需要子机执行什么动作或由子机采集的需要回送的信息。例如，功能码告诉子机读取寄存器的数值，则数据区必须包含要读取寄存器的起始地址及读取长度(寄存器个数)。

与功能码03对应的数据区格式：

主机发送

|  |  |  |
| --- | --- | --- |
| 数据顺序 | 1 | 2 |
| 数据含义 | 起始地址 | 读寄存器个数 |
| 字节数 | 2 | 2 |

子机应答

|  |  |  |
| --- | --- | --- |
| 数据顺序 | 1 | 2 |
| 数据含义 | 回送字节数 | N个寄存器的数据 |
| 字节数 | 1 | 2×N |

与功能码10对应的数据区格式：

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据顺序 | 1 | 2 | … | N |
| 数据含义 | 起始地址 | 写入数据1 | … | 写入数据N |
| 字节数 | 2 | 2 | … | 2 |

#### 校验码(CRC)

主机或子机可用校验码进行判别接收信息是否出错。有时，由于电子噪声或其他一些干扰，信息在传输过程中会发生细微的变化，校验码保证了主机或子机对在传送过程中出错的信息不起作用。这样增加了系统的安全和效率。校验码采用CRC-16校验方法。

二字节的校验码，低字节在前，高字节在后。

|  |  |
| --- | --- |
| 注意： | 命令的格式都是相同的：地址码、功能码、数据区和校验码。 |

冗余循环码(CRC)包含2个字节，即16位二进制。CRC码由发送端计算，放置于发送信息的尾部。接收端的设备再重新计算接收到信息的CRC码，比较计算得到的CRC码是否与接收到的相符，如果二者不相符，则表明出错。

CRC码的计算方法是，先预置16位寄存器全为。再逐渐把每8位数据信息进行处理。

在计算CRC码时，8位数据与寄存器的数据相异或，得到的结果向低位移一位，用0填补最高位。再检查最低位，如果最低位为1，把寄存器的内容与预置数相异或，如果最低位为0，不进行异或运算。

这个过程一直重复8次。第8次移位后，下一个8位再与现在寄存器的内容相异或，这个过程与上以上一样重复8次。当所有的数据信息处理完后，最后寄存器的内容即为CRC码值。

CRC-16码的计算步骤

（1）置16位寄存器为十六进制FFFF(即全为1)。称此寄存器为CRC寄存器。

（2）把一个8位数据与16位CRC寄存器的低位相异或，把结果放于CRC寄存器。

（3）把寄存器的内容右移一位(朝低位)，用0填补最高位，检查最低位(移出位)。

（4）如果最低位为0：复第3步(再次移位)。

如果最低位为1：CRC寄存器与多项式A001(1010 0000 0000 0001)进行异或。

（5）重复步骤3和4，直到右移8次，这样整个8位数据全部进行了处理。

（6）重复步骤2到步骤5，进行下一个8位的处理。

（7）最后得到的CRC寄存器即为CRC码，低字节在前，高字节在后。

## 设备实时采样数据寄存器地址（MODBUS功能码03）

每个寄存器存储2个字节数据。

采集数据寄存器表，支持0x03功能码

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **寄存器** | **项目说明** |  |  |
| 1 | 0x0000 | 设备电压 | hex转成10进制数除以10，单位：v |  |
| 2 | 0x0001 | 温度 | hex转成10进制数除以10，单位：℃ |  |
| 3 | 0x0002 | 湿度 | hex转成10进制数除以10，单位：% |  |
| 4 | 0x0003 |  |  |  |
| 5 | 0x0004 |  |  |  |

设备信息寄存器表，支持0x03、0x10功能码

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **寄存器** | **项目说明** |  |  |
| 1 | 0x1200 | 软件版本 | BCD码 |  |
| 2 | 0x1201 | Modbus地址 | 值范围：1~255 |  |

# 附：CRC校验代码

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//函数名：

//函数功能：CRC16计算

//输入参数：\*Array 需要进行校验的数组指针

// Len 需要进行校验的数组长度

// \*Rcvbuf 计算结果//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void CRC16(uint8\_t \*Array, uint8\_t \*Rcvbuf,uint8\_t Len)

{

uint8\_t IX,IY;

uint16\_t CRC1;

CRC1=0xFFFF;//set all 1

if (Len<=0)

CRC1 = 0;

else

{

for (IX=0;IX<Len;IX++)

{

CRC1=CRC1^(unsigned int)(Array[IX]);

for(IY=0;IY<=7;IY++)

if ((CRC1&1)!=0 )

CRC1=(CRC1>>1)^0xA001;

else

CRC1=CRC1>>1;

}

}

Rcvbuf[0] = (CRC1 & 0x00ff); //低位置

Rcvbuf[1] = (CRC1 & 0xff00)>>8;//高位置

}