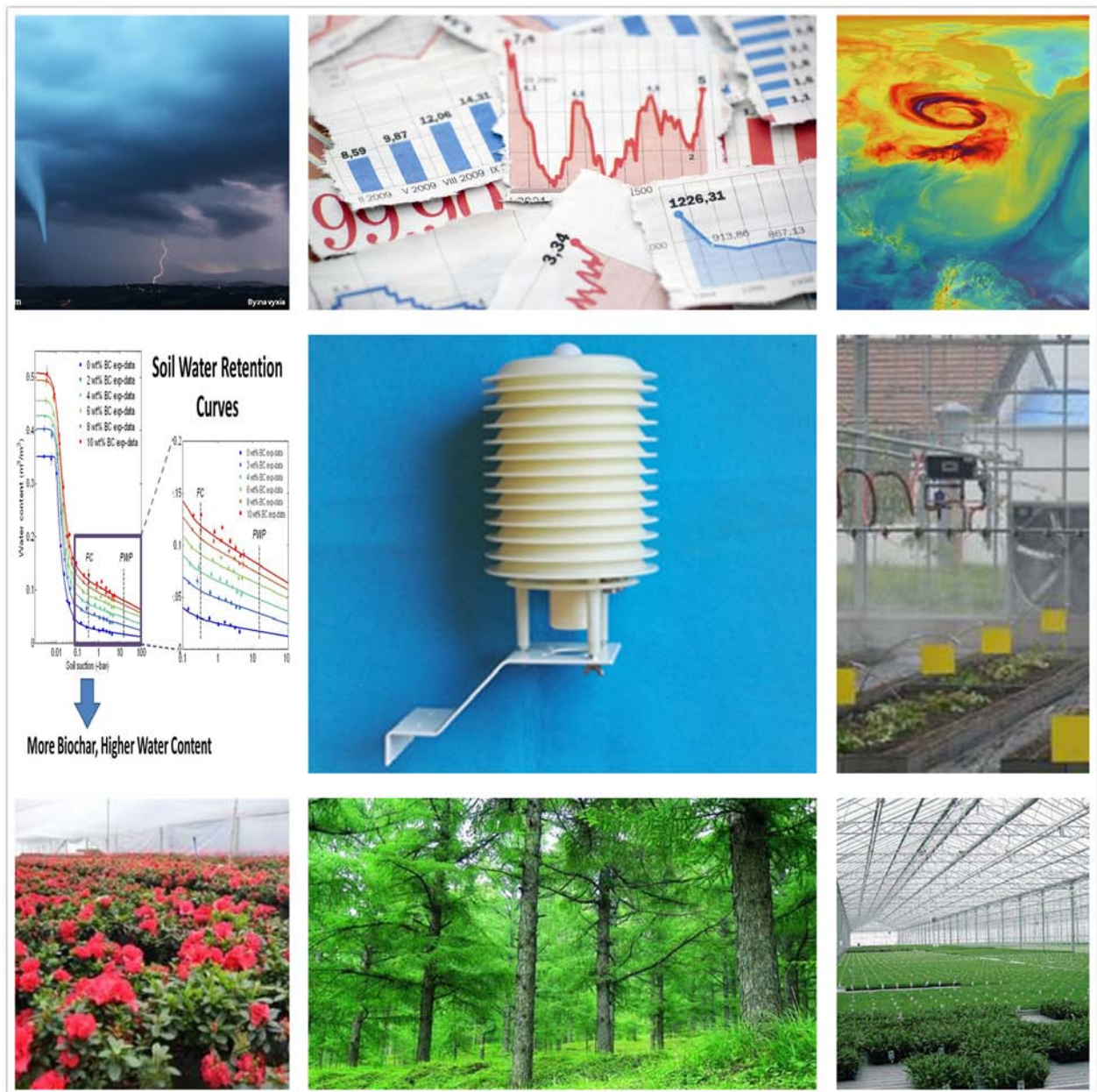


# Daisy20多参数气象监测站 用户手册



# 目 录

1	技术支持 .....	3
2	产品介绍 .....	4
2.1	产品介绍 .....	4
3	传感器接线 .....	6
4	外型尺寸、选型订购 .....	7
4.1	外型尺寸 .....	7
4.2	选型订购 .....	7
5	安装与测量 .....	9
6	RS485 通信与协议 .....	10
6.1	Modbus 通信协议 .....	10
6.2	Modbus 寄存器 .....	10
6.3	Modbus 寄存器参数说明 .....	12
6.4	Modbus 协议通信样例 .....	15
6.4.1	功能号 3 通信样例 .....	15
6.4.2	功能号 4 通信样例 .....	16
6.4.3	功能号 6 通信样例 .....	18
6.4.4	功能号 16 通信样例 .....	19
6.4.5	CRC16 校验算法及例程 .....	20
6.5	使用串口调试软件通信 .....	23
7	用户设置软件 .....	24
7.1	软件安装与启动 .....	24

# 1 技术支持

感谢您选择并使用大连哲勤科技有限公司产品，此用户手册协助您了解并正确使用传感器。如需订购产品、技术支持、以及产品信息反馈，请通过以下方式联系我们。请在联系时附注设备的购买时间，购买方式，联系人信息，地址以及电话等相关信息，便于我们为您服务。

## 网址

<http://www.infwin.com>

## E-Mail

[infwin@163.com](mailto:infwin@163.com)

## 电话

+86-411-66831953, 4000-511-521

## 传真

+86-411-82388125

## 版本控制

日期	版本号	说明	完成人
2017-01-21	V1.0	创建	fg49597

## 2 产品介绍

### 2.1 产品介绍

Daisy20 多参数气象监测站是多要素自动观测站。可监测温度、湿度、露点、气压、光照度、二氧化碳等常规气象要素，广泛应用于气象、环保、机场、农林、水文、军事、仓储、科学研究等领域。

- (1) 多种参数一体测量，可选参数包含：空气温度、空气湿度、露点、大气压、光照度、二氧化碳。
- (2) 传感器精度高，响应快，互换性好，性能可靠。
- (3) 直流宽范围供电，RS485接口，标准的Modbus协议。
- (4) 多参数一体，易于系统集成。
- (5) 完善的保护电路。

技术参数	
信号输出类型	RS485接口 Modbus协议
供电电压	9-30V/DC直流
预热时间	60秒
响应时间	30秒(90%)
静态功耗	20mA@24V DC直流
运行环境	工作温度:-40~80 ℃ 工作湿度:0~95% RH (无凝结)
安装方式	支架安装(带有支架安装板)
默认线缆长度	电源与通信电缆: 2米, 线缆长度可定制。
连接方式	接线安装
外形尺寸	直径*高=140*300mm
空气温度	量程: -40~80℃ 精度: +/-0.3℃ 分辨率: 0.01℃
空气湿度	量程: 0~100% 精度: +/-4% 分辨率: 0.01%
露点	量程: -40~80℃ 精度: +/-0.5℃ 分辨率: 0.01℃
气压	量程: 300~1100mbar 精度: +/-1mbar 分辨率: 0.1mbar

光照度	量程：0~20万Lux 精度：+/-5% 分辨率：0.1Lux
二氧化碳	量程：0~5000ppm 精度：+/- (50ppm+5%读数值) 分辨率：1ppm

### 3 传感器接线

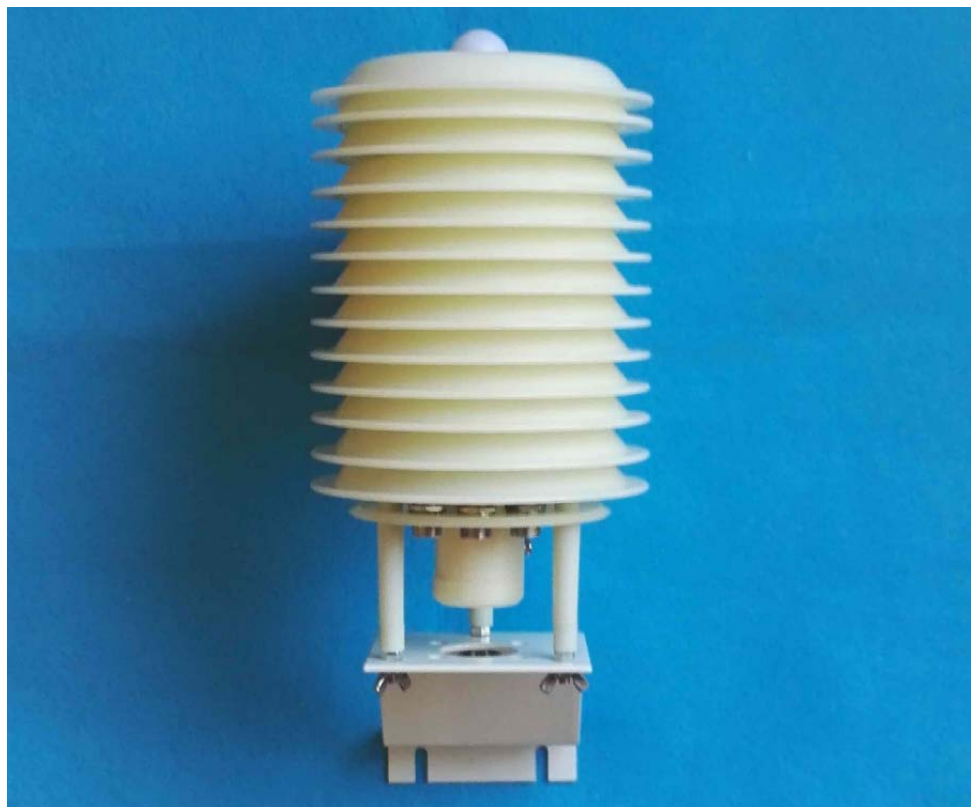
型号	接线图
RS485 接口型 Modbus 协议	<p>4芯航空插头定义：</p> <p>第1芯）红色 (V+)：电源正</p> <p>第2芯）黑色 (G)：电源地</p> <p>第3芯）黄色 (T+)：RS485+/A/T+</p> <p>第4芯）白色 (T-)：RS485-/B/T-</p>
	  <p><b>RS485主设备</b> 为RS485主机（电脑或其他具有RS485接口的嵌入式设备）</p> <p>模块的配置参数如 Modbus 地址，波特率，校验位，通讯协议等是由模块内部的 EEPROM（掉电存储设备）内存储的。有时会忘记这些参数的具体配置而导致不能与模块进行通讯。为了防止这个问题，气象站内部电路板有一按键，按下三秒后松开按键（SW1），则模块所有参数恢复以下出厂设置：</p> <ol style="list-style-type: none"> <li>1. Modbus 地址为 1</li> <li>2. 通信配置为 9600, N, 8, 1（9600bps，无校验位，8 个数据位，一个停止位）</li> </ol> <p>通信协议为 Modbus-RTU</p> 



## 4 外型尺寸、选型订购

### 4.1 外型尺寸

直径：140mm， 高：300mm



### 4.2 选型订购

代码编号	代码	代码说明
代码 1: 产品系列	Daisy20	Daisy20 多参数气象监测站
代码 2: 供电电压	A	9-30V直流
代码 3: 输出信号	A	RS485接口, Modbus协议
代码 4: 通信与电源线缆长度	002	2米线长
	XXX	客户定制, XXX为任意线长(单位: 米)
代码 5: 安装方式	A	支架安装
	X	不需要此项目
代码 6: 空气温度	A	量程: -40~80℃
	X	不需要此项目

代码 7: 空气湿度	A	量程: 0~100%
	X	不需要此项目
代码 8: 露点	A	量程: -40~80℃
	X	不需要此项目
代码 9: 大气压	A	量程: 300~1100mbar
	X	不需要此项目
代码 10: 光照度	A	量程: 0~20万Lux
	X	不需要此项目
代码 11: 二氧化碳	A	量程: 0-1000ppm
	B	量程: 0-2000ppm
	C	量程: 0-5000ppm
	D	量程: 0-10000ppm
	E	量程: 0-30000ppm
	F	量程: 0-50000ppm
	X	不需要此项目
型号举例: Daisy20 多参数气象监测站, 9-30V 直流供电, RS485 接口, Modbus 协议, 电源与通信线缆 2 米, 支架安装, 空气温度, 空气湿度, 露点, 大气压测量。选型代码为: Daisy20-AA002AAAAAXX		

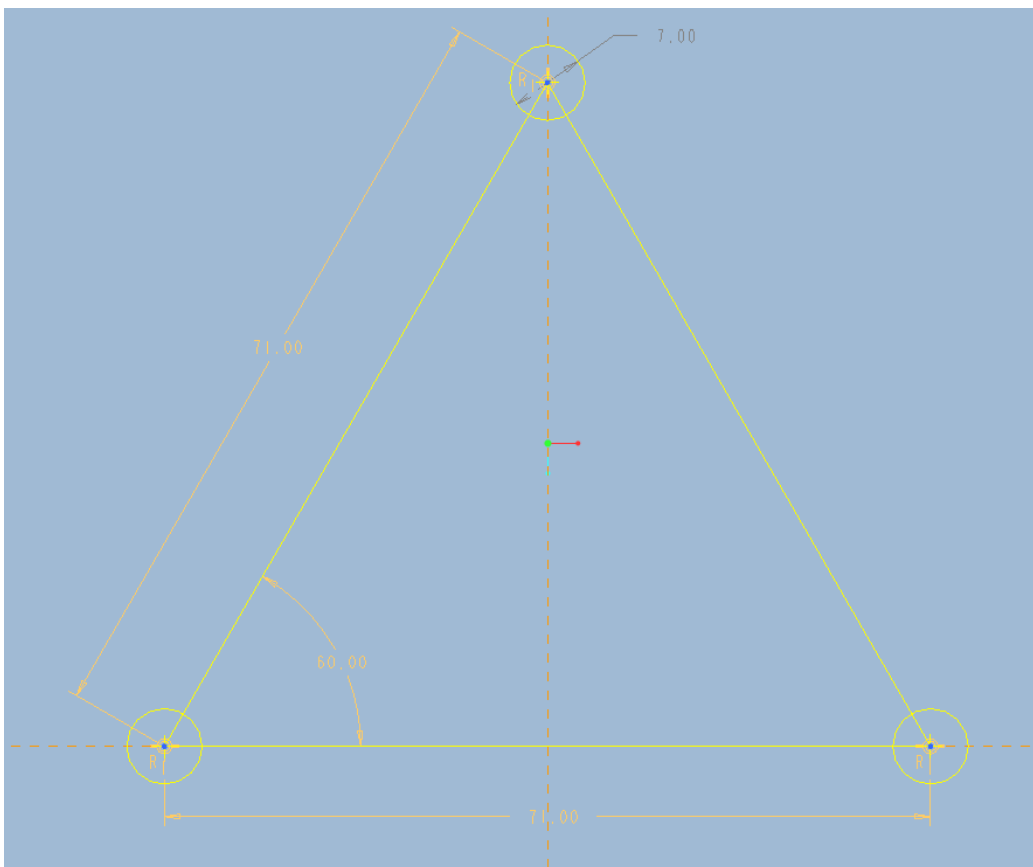


## 5 安装与测量

将安装支架固定于墙面或气象桅杆上即可。如图。



底部安装板钻孔图（单位：mm）



## 6 RS485 通信与协议

### 6.1 Modbus 通信协议

Modbus 是一种串行通信协议，是多种仪器仪表以及智能传感器在通信接口方面的标准，在智能传感器中有着广泛的应用。Modbus 协议是一个主从架构的协议。有一个主节点，其他使用 Modbus 协议参与通信的节点是从节点。每一个从设备都有一个唯一的设备地址。

传感器具有 RS485 接口，支持 Modbus 协议。通讯参数出厂默认值为：波特率 9600bps，一个起始位，8 个数据位，无校验，一个停止位。通讯协议为 Modbus RTU 协议。通讯参数可由设置程序或者 Modbus 命令改变，通信参数改变后需要重新对传感器进行上电方可生效。

### 6.2 Modbus 寄存器

参数名称	寄存器地址 (16进制/10进制)	参数 类型	Modbus 功能号	参数范围及说明	默认值
空气温度 AIRTEMP	0x0000 /0	INT16 只读	3/4	-4000~8000 对应 -40.00~80.00℃	N/A
空气湿度 AIRHUMI	0x0001 /1	UINT16 只读	3/4	0~10000 对应 0.00~100.00%	N/A
露点 AIRDEW	0x0002 /2	INT16 只读	3/4	-4000~8000 对应 -40.00~80.00℃	N/A
大气压 AIRBARO	0x0003 /3	UINT16 只读	3/4	3000~11000 对应 300.0~1100.0mbar	N/A
光照度高 16 位 SUNLUXHI16(高 8 位 在前，低 8 位在后)	0x0004 /4	UINT16 只读	3/4	0~200000 对应 0~200000Lux	N/A
光照度低 16 位 SUNLUXLO16(高 8 位 在前，低 8 位在后)	0x0005 /5	UINT16 只读	3/4		N/A
二氧化碳 AIRC02	0x0006 /6	UINT16 只读	3/4	0~50000 对应 0~50000ppm	N/A
保留 RESERVED	0x0007 /7	INT16 只读	3/4	N/A	-32768
保留 RESERVED	0x0008 /8	UINT16 只读	3/4	N/A	-32768

保留 RESERVED	0x0009 /9	UINT16 只读	3/4	N/A	-32768
保留 RESERVED	0x000A /10	INT16 只读	3/4	N/A	-32768
保留 RESERVED	0x000B /11	UINT16 只读	3/4	N/A	-32768
保留 RESERVED	0x000C /12	UINT16 只读	3/4	N/A	-32768
Modbus 从机地址 (ADDRESS)	0x0200 /512	UINT16 读写	3/6/16	0-255	1
串行通信波特率 (BAUDRATE)	0x0201 /513	UINT16 读写	3/6/16	0-5 0:1200bps 1:2400bps 2:4800bps 3:9600bps 4:19200bps 5:38400bps	3:9600bps
串行通信协议 (PROTOCOL)	0x0202 /514	UINT16 读写	3/6/16	0 0:Modbus RTU	0:Modbus RTU
串行通信校验位 (PARITY)	0x0203 /515	UINT16 读写	3/6/16	0-2 0:无校验 1:偶校验 2:奇校验	0:无校验
串行通信数据位 (DATABITS)	0x0204 /516	UINT16 读写	3/6/16	1 1:8个数据位	1:8个数据位
串行通信停止位 (STOPBITS)	0x0205 /517	UINT16 读写	3/6/16	0-1 0:1个停止位 1:2个停止位	0:1个停止位
串行通信延时响应 (RESPONSEDELAY)	0x0206 /518	UINT16 读写	3/6/16	0-255对应0-2550毫秒 传感器接收到主机请求命令后延时一段时间然后响应。延时时间为设置值*10毫秒。设置为0	0

				时不延时。	
串行通信主动输出时间间隔 (ACTIVEOUTPUTINTERVAL)	0x0207 /519	UINT16 读写	3/6/16	0-255对应0-255秒 不需要主机进行请求， 传感器以固定的时间间隔自动发送数据。时间间隔为设置值*1秒。设置为0时禁止主动输出功能。	0

UINT16:16 位无符号整数寄存器

INT16:16 位有符号整数寄存器

## 6.3 Modbus 寄存器参数说明

AIRTEMP —— 空气温度		
参数范围	-4000-8000 对应 -40.00~80.00℃	默认值: 无
参数存储	无	

意义: 温度测量值, 负数用补码表示。

举例: 如果返回的值是 0702H (16 进制, 原码), 则第一字节高字节为 07H, 第二字节低字节为 02H, 那么温度测量值为  $(07H * 256 + 02H) / 100 = 17.94$  摄氏度。

如果返回的值是 FF05H (16 进制, 补码), 则第一字节高字节为 FFH, 第二字节低字节为 05H, 那么温度测量值为  $((FFH * 256 + 05H) - FFFFH - 1H) / 100 = (FF05H - FFFFH - 1H) / 100 = -2.51$  摄氏度。

AIRHUMI —— 空气湿度		
参数范围	0-10000 对应 0-100%	默认值: 无
参数存储	无	

意义: 湿度测量值。

举例: 如果返回的值是 071DH (16 进制), 则第一字节高字节为 07H, 第二字节低字节为 1DH, 那么测量值为  $(07H * 256 + 1DH) / 10000 = (7 * 256 + 29) / 10000 = 18.21\%$  代表空气湿度为 18.21%。

AIRDEW —— 露点		
参数范围	-4000-8000 对应 -40.00~80.00℃	默认值: 无
参数存储	无	

意义: 露点温度值, 负数用补码表示。

举例：如果返回的值是 0702H（16 进制，原码），则第一字节高字节为 07H，第二字节低字节为 02H，那么温度测量值为  $(07H \times 256 + 02H) / 100 = 17.94$  摄氏度。

如果返回的值是 FF05H（16 进制，补码），则第一字节高字节为 FFH，第二字节低字节为 05H，那么温度测量值为  $((FFH \times 256 + 05H) - FFFFH - 1H) / 100 = (FF05H - FFFFH - 1H) / 100 = -2.51$  摄氏度。

AIRBARO——大气压		
参数范围	3000-11000对应300.0-1100.0mbar	默认值：无
参数存储	无	

意义：露点温度值，负数用补码表示。

举例：如果返回的值是 2827H（16 进制，原码），则第一字节高字节为 28H，第二字节低字节为 27H，那么测量值为  $(28H \times 256 + 27H) / 10 = 1027.9$ mbar。

SUNLUX——光照度		
参数范围	0-200000对应0-200000Lux	默认值：无
参数存储	无	

意义：光照强度。由光照度高 16 位 SUNLUXHI16 和光照度低 16 位 SUNLUXLO16 组成 32 位无符号数值。

举例：如果返回的值是 SUNLUXHI16 = 0x0001，SUNLUXLO16 = 8D38H（16 进制，原码），那么测量值为  $(0001H \times 65536 + 8D38H) = 101688$ Lux。

CO2PPM —— 二氧化碳浓度值		
参数范围	0-50000 对应 0~50000ppm	默认值：无
参数存储	无	

意义：二氧化碳浓度测量值。

举例：如果返回的值是 0702H（16 进制，原码），则第一字节高字节为 07H，第二字节低字节为 02H，那么二氧化碳浓度测量值为  $(07H \times 256 + 02H) = 1794$ ppm。

SLAVEADDR —— Modbus 从机地址		
参数范围	0-255	默认值：1
参数存储	立即存储	

Modbus 地址，可设置为 0-255。当模块外部的地址拨码开关设置为地址 0 时，使用此寄存器的内容作为从机地址。设置后需要重新上电或者使用 RST 命令重新启动模块，使此地址生效。使用此命令修改模块地址不需要打开机壳即可设置。

BAUDRATE --- 串行通信波特率		
参数范围	<b>0-5</b> <b>0:</b> 1200bps <b>1:</b> 2400bps <b>2:</b> 4800bps <b>3:</b> 9600bps <b>4:</b> 19200bps <b>5:</b> 38400bps	默认值:3
参数存储	立即存储	

PROTOCOL --- 串行通信协议		
参数范围	0 0:Modbus RTU	默认值:0
参数存储	立即存储	

PARITY --- 串行通信校验位		
参数范围	0-2 0:无校验 1:偶校验 2:奇校验	默认值:0
参数存储	立即存储	

DATABITS --- 串行通信数据位		
参数范围	1 1:8个数据位	默认值:1, 只支持 8 个数据位, 其他无效
参数存储	立即存储	

STOPBITS --- 串行通信停止位		
参数范围	0-1 0:1个停止位 1:2个停止位	默认值:0
参数存储	立即存储	

RESPONSEDELAY --- 串行通信延时响应		
参数范围	<b>0-255</b>	默认值:0

参数存储	立即存储	
------	------	--

串行通信延时响应在以下情况下使用:当主机发送请求命令后,模块延时 (RESPONSEDELAY\*10) 毫秒,然后将响应数据返回给主机。比如设置 RESPONSEDELAY=5,那么模块延时 5\*10=50 毫秒后响应主机请求。设置为 0 时为无延时立即响应。此命令主要应用于主机从 RS485 发送状态切换为接收状态时速度比较慢的场合。

ACTIVEOUTPUTINTERVAL —— 串行通信主动输出时间间隔		
参数范围	0-255	默认值:0
参数存储	立即存储	

串行通信主动输出时间间隔在以下情况下使用:主机不需要发送请求命令,模块主动输出响应数据,输出间隔为 ACTIVEOUTPUTINTERVAL 秒,比如设置 ACTIVEOUTPUTINTERVAL=5,那么模块每 5 秒按照设置的通信协议输出数据。设置为 0 时主动输出无效,需主机请求后方可响应。此命令主要应用于 GPRS 等无线传输时,需要终端节点主动发送数据的场合。

注意:当设置为主动输出数据时,RS485 总线上只能连接一个模块,以避免总线数据冲突。

## 6.4 Modbus 协议通信样例

以下说明中,0x 开头或者 H 结尾的数据为 16 进制数据。Modbus 协议有两种常用寄存器类型:

- (1) 保持寄存器,存储数据掉电不丢失,是可读可写的。通常用功能号 3 (0x03) 读取,用功能号 6 (0x06) 或者 16 (0x10) 写入。
- (2) 输入寄存器,用来存储一些只读的物理量,比如温度值,是只读的。通常用功能号 4 (0x04) 读取。

### 6.4.1 功能号 3 通信样例

通用请求格式: AA 03 RRRR NNNN CCCC

AA	1 字节	设备地址,范围 0-255
03	1 字节	功能号为 3
RRRR	2 字节	起始寄存器地址,高字节在前
NNNN	2 字节	要读取的寄存器数量 N,高字节在前
CCCC	2 字节	CRC 校验



通用响应格式：AA 03 MM VV0 VV1 VV2 VV3... CCCC

AA	1 字节	设备地址，范围 0-255
03	1 字节	功能号为 3
MM	1 字节	返回寄存器值的数据字节数量
VV0, VV1	2 字节	返回的第一个寄存器值
VV2, VV3	2 字节	返回的第二个寄存器值
...	...	返回的第 N 个寄存器值 (N=MM/2)
CCCC	2 字节	CRC 校验

举例：以读寄存器 0x0200-0x0201，即从机地址以及波特率为例

请求：01 03 0200 0002 C5B3

设备地址	1 字节	0x01
功能号	1 字节	0x03
起始寄存器地址	2 字节	0x0200
寄存器数量	2 字节	0x0002
校验	2 字节	0xC5B3

响应：01 03 04 00 01 00 03 EB F2

设备地址	1 字节	0x01
功能号	1 字节	0x03
有效字节数	1 字节	0x04
从机地址寄存器值	2 字节	0x00 (从机地址高字节)
		0x01 (从机地址低字节)
波特率寄存器值	2 字节	0x00 (波特率高字节)
		0x03 (波特率低字节)
校验	2 字节	0xEBF2

## 6.4.2 功能号 4 通信样例

通用请求格式：AA 04 RRRR NNNN CCCC

AA	1 字节	设备地址，范围 0-255
04	1 字节	功能号为 4
RRRR	2 字节	起始寄存器地址，高字节在前
NNNN	2 字节	要读取的寄存器数量 N，高字节在前

CCCC	2 字节	CRC 校验
------	------	--------

通用响应格式: AA 04 MM VV0 VV1 VV2 VV3... CCCC

AA	1 字节	设备地址, 范围 0-255
04	1 字节	功能号为 4
MM	1 字节	返回寄存器值的数据字节数量
VV0, VV1	2 字节	返回的第一个寄存器值
VV2, VV3	2 字节	返回的第二个寄存器值
...	...	返回的第 N 个寄存器值 (N=MM/2)
CCCC	2 字节	CRC 校验

举例 1: 以读寄存器 0x0000, 即读取空气温度测量值

请求: 01 04 0000 0001 31CA

设备地址	1 字节	0x01
功能号	1 字节	0x04
起始寄存器地址	2 字节	0x0000
寄存器数量	2 字节	0x0001
校验	2 字节	0x31CA

响应: 01 04 02 05 A3 FA 19

设备地址	1 字节	0x01
功能号	1 字节	0x04
有效字节数	1 字节	0x02
空气温度寄存器 值	2 字节	0x05
		0xA3
校验	2 字节	0xFA19

举例 2: 以读寄存器 0x0000-0x0006, 即读取空气温度、空气湿度、露点、大气压、光照度、二氧化碳为例。

请求: 01 04 0000 0007 B1C8

设备地址	1 字节	0x01
功能号	1 字节	0x04
起始寄存器地址	2 字节	0x0000
寄存器数量	2 字节	0x0007
校验	2 字节	0xB1C8

响应: 01 04 0E 09F2 1B96 07B4 26EF 0000 00C1 02A6 DEF4

设备地址	1 字节	0x01
功能号	1 字节	0x04
有效字节数	1 字节	0x0E
空气温度寄存器 值	2 字节	0x09
		0xF2
空气湿度寄存器	2 字节	0x1B
		0x96
露点寄存器	2 字节	0x07
		0xB4
大气压寄存器	2 字节	0x26
		0xEF
光照度寄存器- 高 16 位	2 字节	0x00
		0x00
光照度寄存器- 低 16 位	2 字节	0x00
		0xC1
二氧化碳浓度寄 存器	2 字节	0x02
		0xA6
校验	2 字节	0xDEF4

### 6.4.3 功能号 6 通信样例

通用请求格式: AA 06 RRRR VVVV CCCC

AA	1 字节	设备地址, 范围 0-255
06	1 字节	功能号为 6
RRRR	2 字节	寄存器地址, 高字节在前
VVVV	2 字节	要写入寄存器的数值, 高字节在前
CCCC	2 字节	CRC 校验

通用响应格式: AA 06 RRRR VVVV CCCC

AA	1 字节	设备地址, 范围 0-255
06	1 字节	功能号为 6

RRRR	2 字节	寄存器地址，高字节在前
VVVV	2 字节	要写入寄存器的数值，高字节在前
CCCC	2 字节	CRC 校验

举例：以写寄存器 0x0200，即从机地址设置为 1 为例

请求：01 06 0200 0001 49B2

设备地址	1 字节	0x01
功能号	1 字节	0x06
起始寄存器地址	2 字节	0x0200
要写入寄存器的数值，高字节在前	2 字节	0x0001
校验	2 字节	0x49B2

响应：01 06 0021 0001 1800

设备地址	1 字节	0x01
功能号	1 字节	0x06
起始寄存器地址	2 字节	0x0200
要写入寄存器的数值，高字节在前	2 字节	0x0001
校验	2 字节	0x49B2

#### 6.4.4 功能号 16 通信样例

通用请求格式：AA 10 RRRR NNNN MM VVVV1 VVVV2 ...CCCC

AA	1 字节	设备地址，范围 0-255
10 (16 进制)	1 字节	功能号为 16 (十进制)
RRRR	2 字节	起始寄存器地址，高字节在前
NNNN	2 字节	要读取的寄存器数量 N，高字节在前
MM	1 字节	要写入寄存器的数值的字节个数
VVVV1	2 字节	要写入第一个寄存器的数值，高字节在前
VVVV2	2 字节	要写入第二个寄存器的数值，高字节在前
...	...	要写入第 N 个寄存器的数值，高字节在前

		$N=MM/2$
CCCC	2 字节	CRC 校验

通用响应格式: AA 10 RRRR NNNN CCCC

AA	1 字节	设备地址, 范围 0-255
10 (16 进制)	1 字节	功能号为 16 (十进制)
RRRR	2 字节	起始寄存器地址, 高字节在前
NNNN	2 字节	要读取的寄存器数量 N, 高字节在前
CCCC	2 字节	CRC 校验

举例: 以写寄存器 0x0200-0x0201, 即设置从机地址为 1, 波特率为 19200bps 为例

请求: 01 10 0200 0002 04 0001 0004 BACC

0x01	1 字节	设备地址
0x10 (16 进制)	1 字节	功能号为 16 (十进制)
0x0200	2 字节	起始寄存器地址, 高字节在前
0x0002	2 字节	要读取的寄存器数量 N, 高字节在前
0x04	1 字节	要写入寄存器的数值的字节个数
0x0001	2 字节	要写如的从站地址寄存器值为 1
0x0004	2 字节	要写如的波特率寄存器值为 4
0xBACC	2 字节	CRC 校验

响应: 01 10 0200 0002 4070

0x01	1 字节	设备地址
0x10 (16 进制)	1 字节	功能号为 16 (十进制)
0x0200	2 字节	起始寄存器地址, 高字节在前
0x0002	2 字节	要读取的寄存器数量 N, 高字节在前
0x4070	2 字节	CRC 校验

## 6.4.5 CRC16 校验算法及例程

例程:

```
//-----
//CRC 计算 C51 语言函数如下
//输入参数 1: snd, 待校验的字节数组名
//输入参数 2: num, 待校验的字节总数
```

//函数返回值：校验和

```
//-----
unsigned int calc_crc16 (unsigned char *snd, unsigned char num)
{
    unsigned char i, j;
    unsigned int c, crc=0xFFFF;
    for(i = 0; i < num; i ++)
    {
        c = snd[i] & 0x00FF;
        crc ^= c;
        for(j = 0; j < 8; j ++)
        {
            if (crc & 0x0001)
            {
                crc>>=1;
                crc ^= 0xA001;
            }
            else
            {
                crc>>=1;
            }
        }
    }
    return(crc);
}
```

**举例：以读寄存器 0x0000，即读取空气温度测量值**

**主机请求：01 04 0000 0001 31CA（8 个字节）**

设备地址	1 字节	0x01
功能号	1 字节	0x04
起始寄存器地址	2 字节	0x0000
寄存器数量	2 字节	0x0001
校验	2 字节	0x31CA

当主机需要发送数据给传感器以前，将需要进行发送校验的数据存储到 snd 数组中（01 04 00 00 01 共 6 个字节），其中 num=6

伪代码如下：

```
unsigned char request[8]={01, 04, 00, 00, 00, 01, 00, 00}; //最后两个 00, 00 是 CRC 校验
unsigned char num=6; //计算数组前 6 个字节的 CRC 校验
unsigned int crc16=0;
crc16= calc_crc16 (request, num);
request[6]= crc16%256; //把 crc 校验存储到要发送的数组中
request[7]= crc16/256;
CommPort.Send(request, 8); //通过串口发送数据
```

**传感器响应：01 04 02 05A3 FA19 （7个字节）**

设备地址	1 字节	0x01
功能号	1 字节	0x04
有效字节数	1 字节	0x02
空气温度寄存器 值	2 字节	0x05
		0xA3
校验	2 字节	0xFA19

当主机接收到传感器返回的 7 个字节数据后，进行以下 crc 计算操作，其中 num=7

伪代码如下：

```
unsigned char response[7]={ 01 04 02 05 A3 FA 19}; //最后两个字节是传感器返回的 CRC 校验
unsigned char num=7; //计算整个返回的 7 个字节的 CRC 校验
unsigned int crc16=0;
crc16= calc_crc16 (response, num);
if(crc16==0)
{
    //crc 校验正确，可以使用返回的数据
}
else
{
    //crc 校验错误，不能使用返回的数据
}
```

得到返回结果为0时那么校验成功，如果校验失败返回为非零值。如果校验不成功，说明



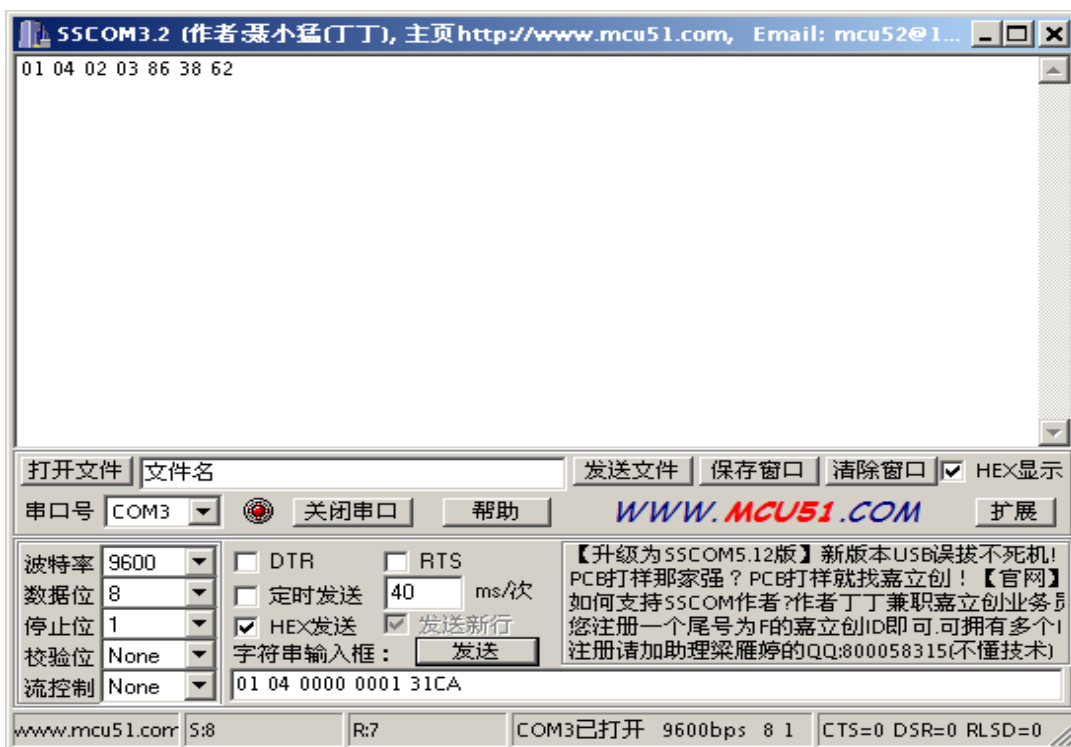
传输过程发生错误，应放弃此次采集到的数据，重新采集。

校验成功后，使用以下公式计算，H结尾的为16进制数据：

空气温度值 =  $(05H * 256 + A3H) / 100 = 14.43^{\circ}\text{C}$

## 6.5 使用串口调试软件通信

用户可使用任意一款串口调试软件与传感器进行通信，通信时需注意，选择正确的串口，波特率，以及其他串口通信参数，需要发送和接收的数据均要以16进制进行传输以及显示。



## 7 用户设置软件

### 7.1 软件安装与启动

(1) SensorOneSet 设置软件基于 Dotnet Framework 开发，安装前需先安装 Dotnet Framework 3.5 以上版本。方可运行。如果电脑没有安装微软 DotNetFramework3.5SP1 的,请先下载完整安装包: <http://www.microsoft.com/zh-cn/download/details.aspx?id=25150>

(2) 安装 Dotnet Framework 完成后，可点击“Install.SensorOneSet.msi”进行程序安装。

名称	修改日期	类型	大小
Install.SensorOneSet.msi	2016-04-23 12:59	Windows Installer ...	976 KB
setup.exe	2016-04-23 12:59	应用程序	483 KB

(3) 从开始菜单中启动“SensorOneSet 用户设置程序”，启动如下画面。

