

# ALS20光照度传感器

## 用户手册



# 目 录

1	技术支持 .....	3
2	产品介绍与背景知识 .....	4
2.1	背景知识 .....	4
2.1.1	测量光照度的意义 .....	错误！未定义书签。
2.1.2	光照度对植物的影响 .....	错误！未定义书签。
2.2	产品介绍 .....	4
3	传感器接线 .....	5
4	外型尺寸、选型订购 .....	6
4.1	外型尺寸 .....	6
4.2	选型订购 .....	8
5	安装与维护 .....	9
5.1	安装方式 .....	9
5.2	注意事项 .....	9
5.3	维护保养 .....	9
6	光照度值与输出的换算 .....	10
7	RS485 通信与协议 .....	11
7.1	Modbus 通信协议 .....	11
7.2	Modbus 寄存器 .....	11
7.3	Modbus 寄存器参数说明 .....	13
7.4	Modbus 协议通信样例 .....	15
7.4.1	功能号 3 通信样例 .....	15
7.4.2	功能号 4 通信样例 .....	16
7.4.3	功能号 6 通信样例 .....	17
7.4.4	功能号 16 通信样例 .....	18
7.4.5	CRC16 校验算法及例程 .....	19
7.5	使用串口调试软件通信 .....	22
8	用户设置软件 .....	22
8.1	软件安装与启动 .....	22
8.2	运行设置软件 .....	23

# 1 技术支持

感谢您选择并使用大连哲勤科技有限公司的产品，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
2016-06-02	V1.0	创建	fg49597

## 2 产品介绍与背景知识

### 2.1 产品介绍

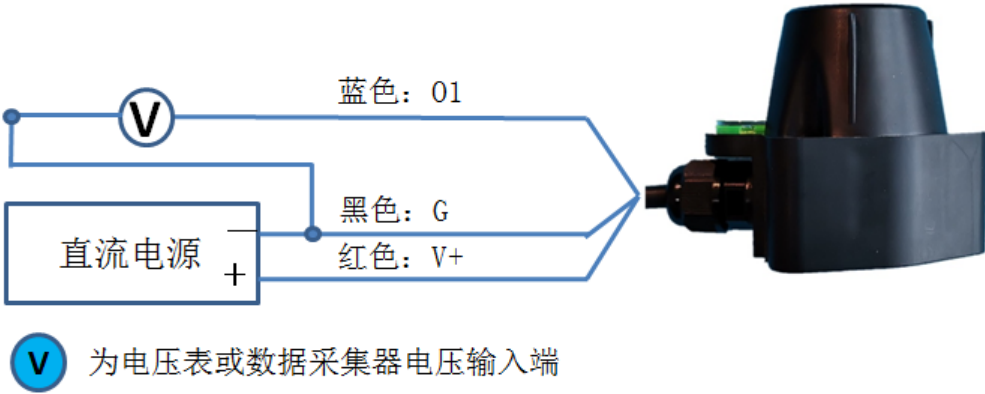
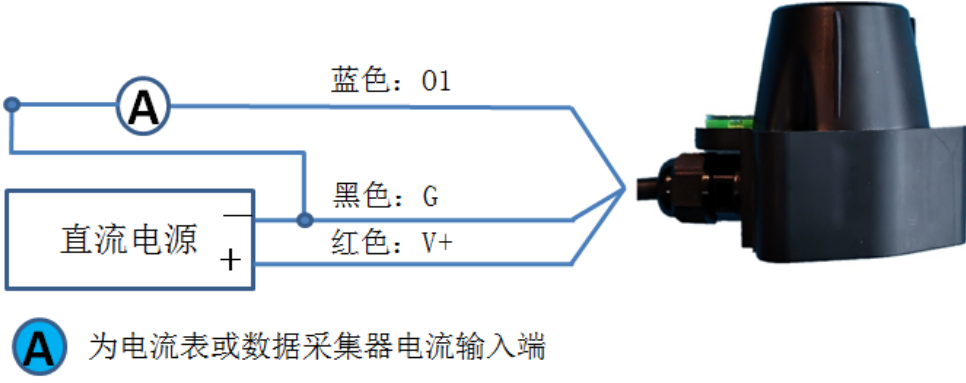
ALS20 光照度传感器采用进口高精度光照度传感器，具有测量范围宽、线形度好、防水性能好、使用方便、便于安装、传输距离远等特点，根据不同的测量场所，配合不同的量程，具有可靠性高、结构美观、安装使用方便、抗干扰能力强等特点。适用于农业大棚、城市照明等各种需要对光照度进行测量的场合。

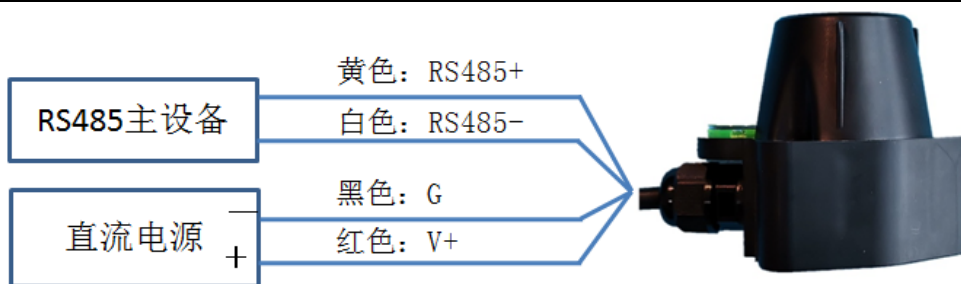
- ☑ 实现光照度的在线实时监测。
- ☑ 精密的光学滤镜以及检测电路确保稳定可靠。
- ☑ 带有水平调节机构与水平泡，方便调平。
- ☑ 防水密封，集成度高、体积小、功耗低、携带方便，可室外使用。
- ☑ 精度高，响应快，互换性好，性能可靠。
- ☑ 完善的保护电路与多种信号输出接口可选。

技术参数（备注 1）			
信号输出类型	电压输出 0~2V (输出阻抗约 0 欧)	电流输出 4~20mA (负载电阻<500ohm)	RS485接口 Modbus协议
供电电压	3.9~30V/DC 直流	12~30V/DC 直流	3.9~30V/DC 直流
静态功耗	7mA@24V DC 直流	30mA@24V DC 直流 (电流输出通道为 20mA)	7mA@24V DC 直流
量程与精度	0~2000 Lux / ±5% 0~5000 Lux / ±5% 0~10000 Lux / ±5% 0~20000 Lux / ±5% 0~100000 Lux / ±5% 0~200000 Lux / ±6%		
波长范围	400~1100nm		
防护等级	IP66		
运行环境	-40~85℃		
默认线缆长度	2 米，长度可定制		
外形尺寸	75*55*58mm		

备注 1：可按客户需求进行定制

### 3 传感器接线

型号	接线图
电压输出型	<p>红色 (V+): 电源正 黑色 (G): 电源地 蓝色 (01): 输出信号</p>  <p>蓝色: 01</p> <p>黑色: G</p> <p>红色: V+</p> <p>直流电源</p> <p>V 为电压表或数据采集器电压输入端</p>
电流输出型	<p>红色 (V+): 电源正 黑色 (G): 电源地 蓝色 (01): 输出信号</p>  <p>蓝色: 01</p> <p>黑色: G</p> <p>红色: V+</p> <p>直流电源</p> <p>A 为电流表或数据采集器电流输入端</p>
RS485 接口型 Modbus 协议	<p>红色 (V+): 电源正 黑色 (G): 电源地 黄色 (T+): RS485+/A/T+ 白色 (T-): RS485-/B/T-</p>



**RS485主设备** 为RS485主机（电脑或其他具有RS485接口的嵌入式设备）

传感器的配置参数如 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，模块有一按键，按下三秒后模块内部指示灯熄灭，松开按键，则模块所有参数恢复以下出厂设置：

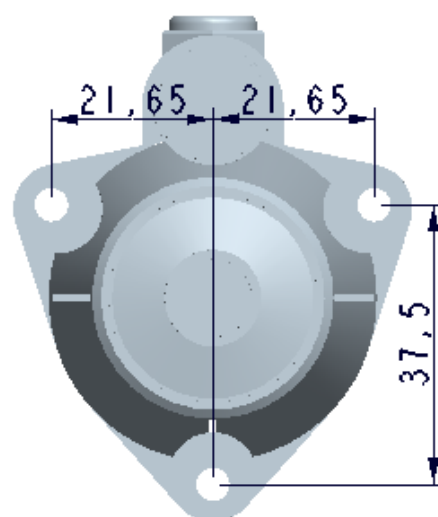
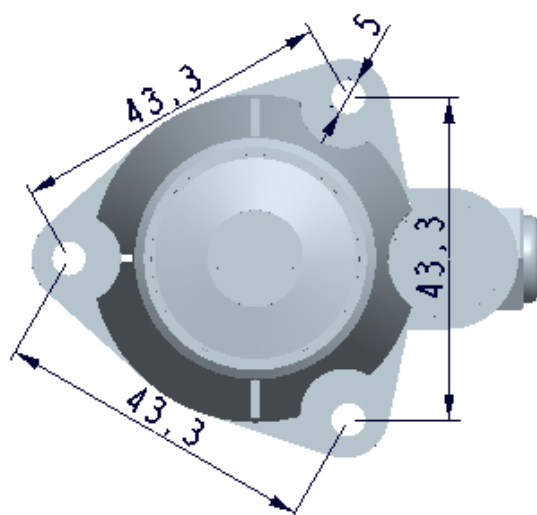
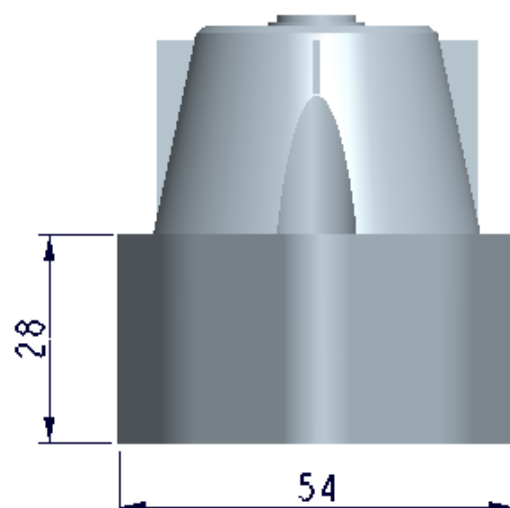
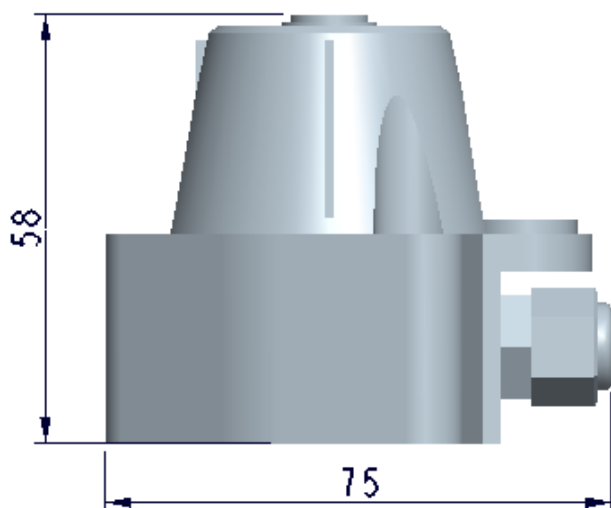
1. Modbus 地址为 1
2. 通信配置为 9600, N, 8, 1（9600bps，无校验位，8 个数据位，一个停止位）

通信协议为 Modbus-RTU

## 4 外型尺寸、选型订购

### 4.1 外型尺寸

单位：mm



## 4.2 选型订购

代码编号	代码	代码说明
代码 1: 产品系列	ALS20	ALS20 光照度传感器
代码 2: 产品系列	A	0-2000 Lux
	B	0-5000 Lux
	C	0-10000 Lux
	D	0-20000 Lux
	E	0-100000 Lux
	F	0-200000 Lux
	G	客户订制
代码 3: 供电电压	A	3.9-30V 直流
	B	2.7-16V 直流
代码 4: 输出信号	A	电压输出0-2V
	B	电流输出4-20mA
	C	RS485接口, Modbus协议
	D	RS485接口, Modbus协议 & 电压0-2V输出
	E	RS485接口, Modbus协议 & 电流4-20mA输出
	F	SDI-12接口
	G	客户订制
代码 5: 线长	002	线缆2米
	XXX	XXX为任意线长 (单位: 米)
型号举例: ALS20光照度传感器, 0-200000Lux量程, 3.9-30V 直流, RS485接口, Modbus协议, 线缆5米。选型代码为: ALS20-FAC005		



## 5 安装与维护

### 5.1 安装方式

安装时请根据水平泡把变送器安装调平。

### 5.2 注意事项

保持变送器顶端的白色滤光片清洁，避免滤光片与硬物接触划伤，影响测量。

### 5.3 维护保养

定期用软布轻轻擦净变送器顶端的白色滤光片，调整安装支架并观察水平泡使其达到水平。

## 6 光照度值与输出的换算

型号	参数范围	换算关系
电压输出 0-2V	0-2000 Lux 0-5000 Lux 0-10000 Lux 0-20000 Lux 0-100000 Lux 0-200000 Lux	线性 线性 线性 线性 线性 线性
电流输出 4-20mA	0-2000 Lux 0-5000 Lux 0-10000 Lux 0-20000 Lux 0-100000 Lux 0-200000 Lux	线性 线性 线性 线性 线性 线性
RS485 接口 Modbus 协议		参见 Modbus 协议章节介绍
客户订制	订制型号的输出请联系技术支持。	

注：公式中电压单位为伏(V)，电流单位为毫安(mA)

## 7 RS485 通信与协议

### 7.1 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。

### 7.2 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
光照度值高 16 位 (LUXHI16)	0x0000	INT16, 只 读	3/4	光照度值为 32 位无符号数 据 由 2 个 16 位寄存器组成。 每个 16 位寄存器可单独读 取。读出的数值乘以“光照 度值分辨率”即为实际的光 照度值。	N/A
光照度值低 16 位 (LUXLO16)	0x0001	INT16, 只 读	3/4		
变送器运行状态 (LUXSTATUS)	0x0002	INT16, 只 读	3/4	BIT15-BIT2: 保留 BIT1: 传感器错误时置 1 BIT0: 超量程时置 1	无
保留	0x0003	UINT16 只读	3/4	N/A	1
保留	0x0004 /4	UINT16 只读	3/4	N/A	0
保留	0x0005 /5	UINT16 只读	3/4	N/A	0
Modbus 从机地址 (ADDRESS)	0x0200 /512	UINT16 读写	3/6/16	0-255	1
串行通信波特率	0x0201 /513	UINT16	3/6/16	0-6	3:9600bps

(BAUDRATE)		读写		0:1200bps 1:2400bps 2:4800bps 3:9600bps 4:19200bps 5:38400bps	
串行通信协议 (PROTOCOL)	0x0202 /514	UINT16 读写	3/6/16	0~1 0:Modbus RTU 1:Modbus ASCII	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	0-250对应0-2500毫秒 传感器接受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时此功能禁用。	0
串行通信主动输出时间间隔 (ACTIVEOUTPUTINTERVAL)	0x0207 /519	UINT16 读写	3/6/16	0-250对应0-250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时此功能禁用。	0

UINT16:16 位无符号整数寄存器

INT16:16 位有符号整数寄存器

## 7.3 Modbus 寄存器参数说明

LUXHI16-光照度值高 16 位 LUXLO16-光照度值低 16 位		
参数范围	0-200000	默认值:无
参数存储	无	

意义：光照度值。

举例：如果 LUXHI16 寄存器值为 0001H，LUXLO16 寄存器值为 86A0H，则组成 32 位数值后为 000186A0H；转换为 10 进制数值为 100000；如果 LUXRES 寄存器（光照度值分辨率）为 1，则 100000 的每个数字代表 1Lux；光照度=100000\*1Lux=100000Lux。

LUXSTATUS-变送器运行状态		
参数范围	BIT15-BIT2：保留 BIT1：传感器错误时置1 BIT0：超量程时置1	默认值：无
参数存储	无	

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	0~1 0:Modbus RTU 1:Modbus ASCII	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	0-255	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。

## 7.4 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。

### 7.4.1 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

## 7.4.2 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：以读寄存器 0x0000，0x0001，即读光照度值的高低 16 位寄存器为例：

请求：01 04 00 00 00 02 71 CB

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	0x0002
校验	2 字节	0x71CB

响应：01 04 04 00 00 00 4D 3B B1

设备地址	1 字节	0x01
功能号	1 字节	0x04
有效字节数	1 字节	0x04
光照度值高 16 位	2 字节	0x00
		0x00
光照度值低 16 位	2 字节	0x00
		0x4D
校验	2 字节	0x3BB1

### 7.4.3 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，即修改从机地址为例

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
寄存器值	2 字节	0x0001
校验	2 字节	0x49B2

响应：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
寄存器值	2 字节	0x0001
校验	2 字节	0x49B2

#### 7.4.4 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

## 7.4.5 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 0x0000，0x0001，即光照度值为例**

**主机请求：01 04 00 00 02 71 CB**

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	0x0002
校验	2 字节	0x71CB

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 02 共 6 个字节），其中 num=6

伪代码如下：

```

unsigned char request[8]={01, 04, 00, 00, 02, 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 04 0000 004D 3BB1 (共9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	0x04
光照度值高 16 位	2 字节	0x00
		0x00
光照度值低 16 位	2 字节	0x00
		0x4D
校验	2 字节	0x3BB1

当主机接收到传感器返回的 9 个字节数据后, 进行以下 crc 计算操作, 其中 num=9  
伪代码如下:

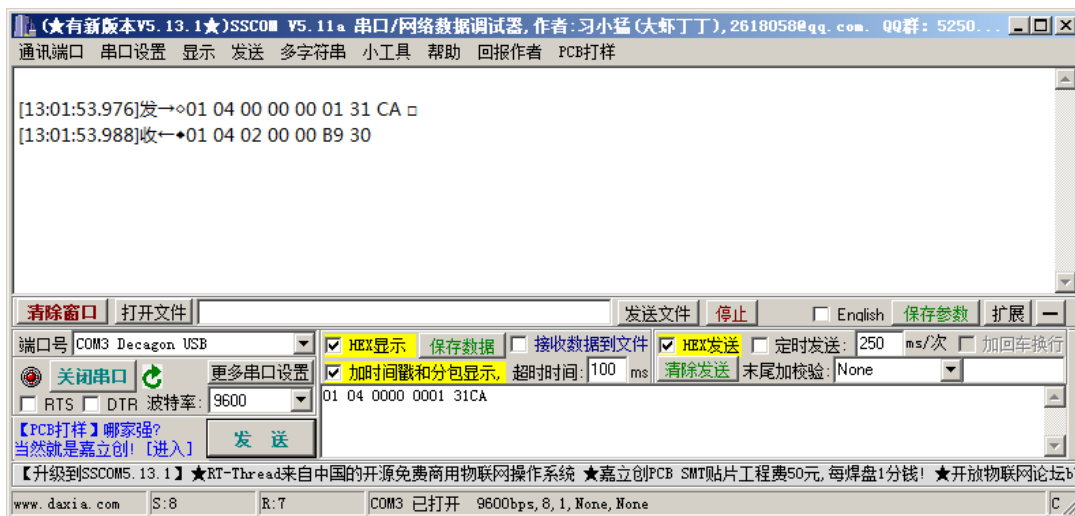
```
unsigned char response[9]={ 01 04 04 00 00 00 4D 3B B1 };//最后两个字节是传感器返回的 CRC 校验
unsigned char num=7;//计算整个返回的 9 个字节的 CRC 校验
unsigned int crc16=0;
crc16= calc_crc16 (response, num);
if(crc16==0)
{
    //crc 校验正确, 可以使用返回的数据
}
else
{
    //crc 校验错误, 不能使用返回的数据
}
```

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, 使用以下公式计算光照度值, H结尾的为16进制数据:  
光照度寄存器值=0000004DH=77。

## 7.5 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。



## 8 用户设置软件

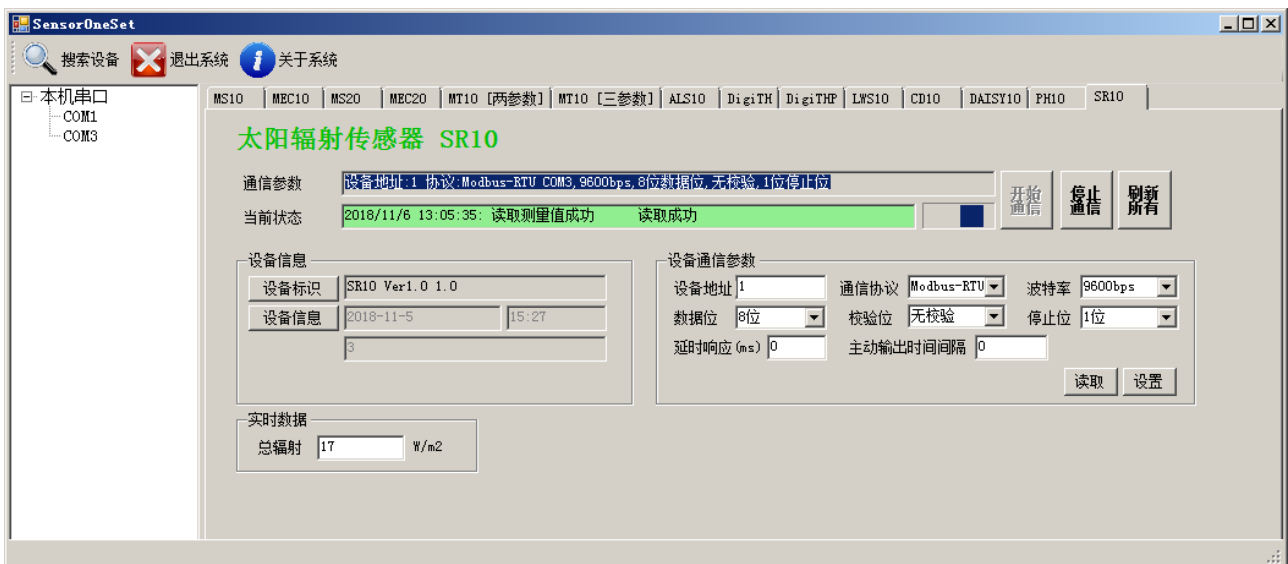
### 8.1 软件安装与启动

(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。

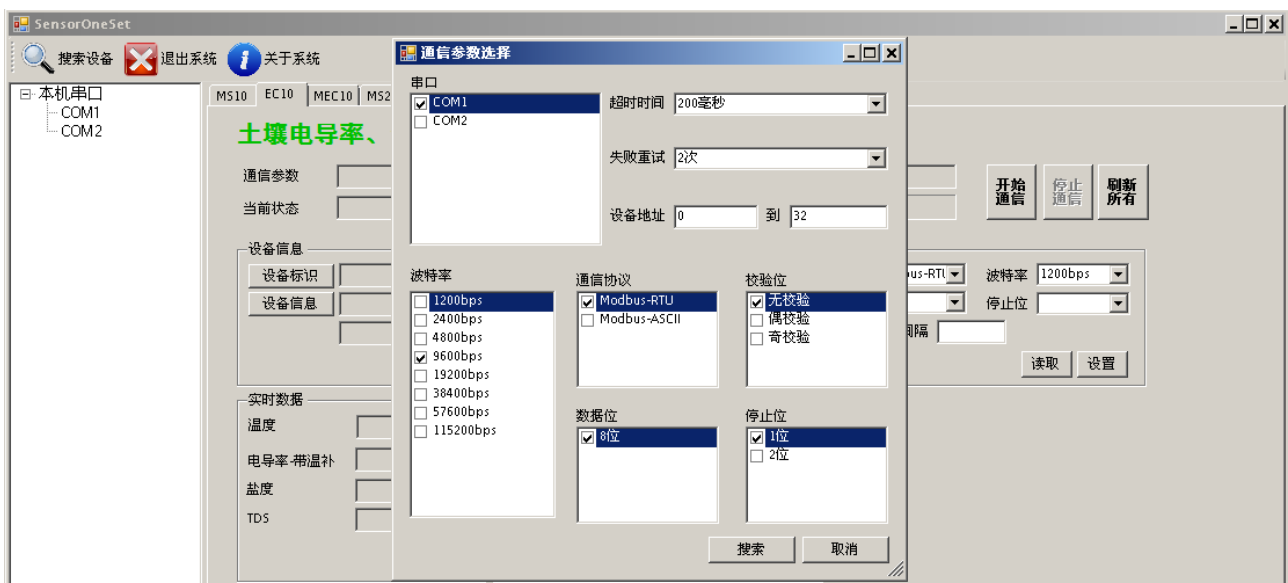
名称 ▲	修改日期	类型	大小
Install.SensorOneSet.msi	2016-04-23 12:59	Windows Installer ...	976 KB
setup.exe	2016-04-23 12:59	应用程序	483 KB

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

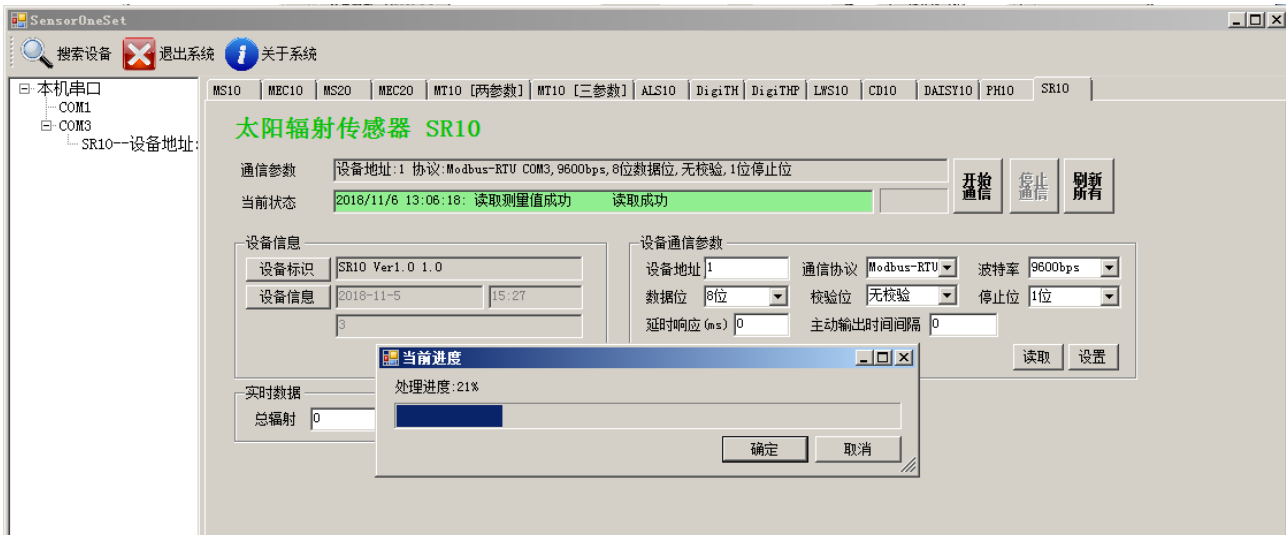
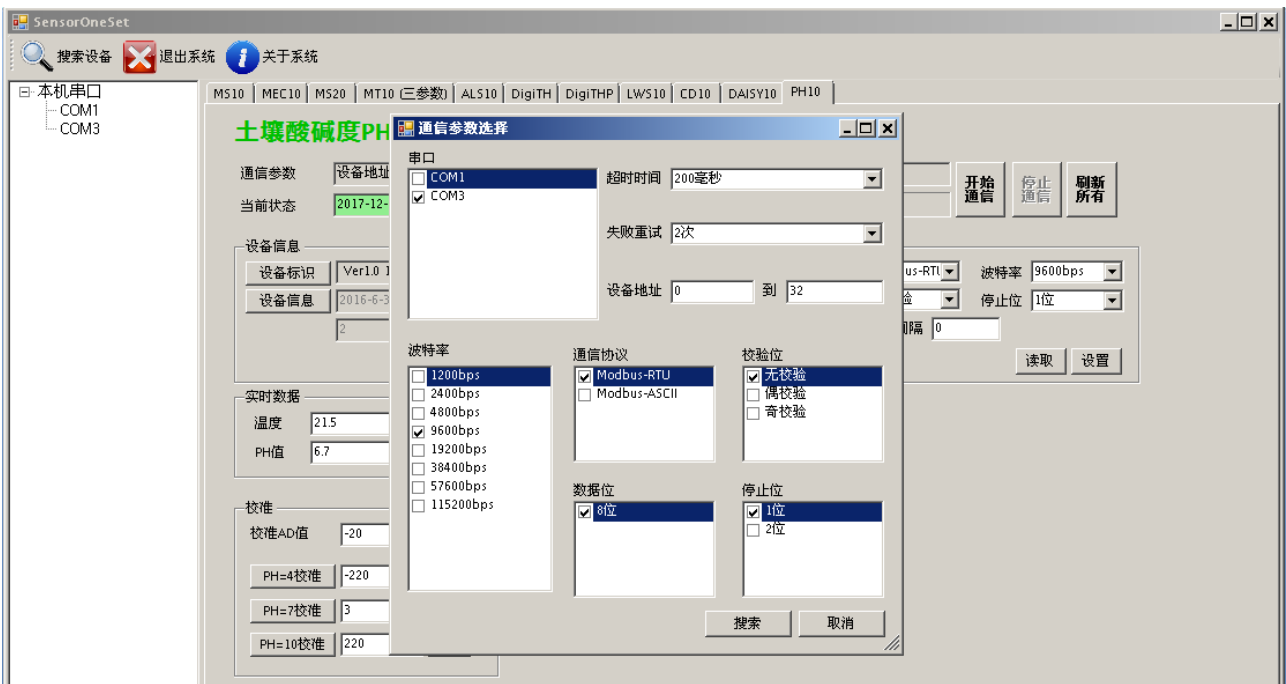


## 8.2 运行设置软件

(1) 点击工具栏中的“搜索设备”按钮，弹出“搜索在线设备-选择搜索参数”对话框。

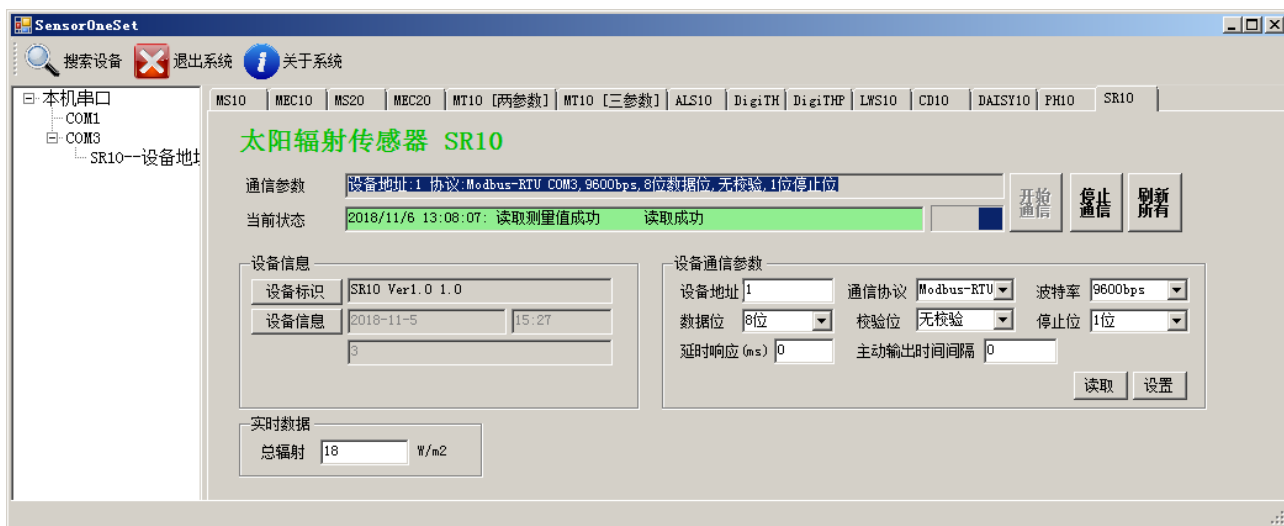


(2) 在“通信参数选择”对话框中选择合适的通讯参数。并点击“搜索”按钮。搜索到的设备会列在相应的串口下。如下图。



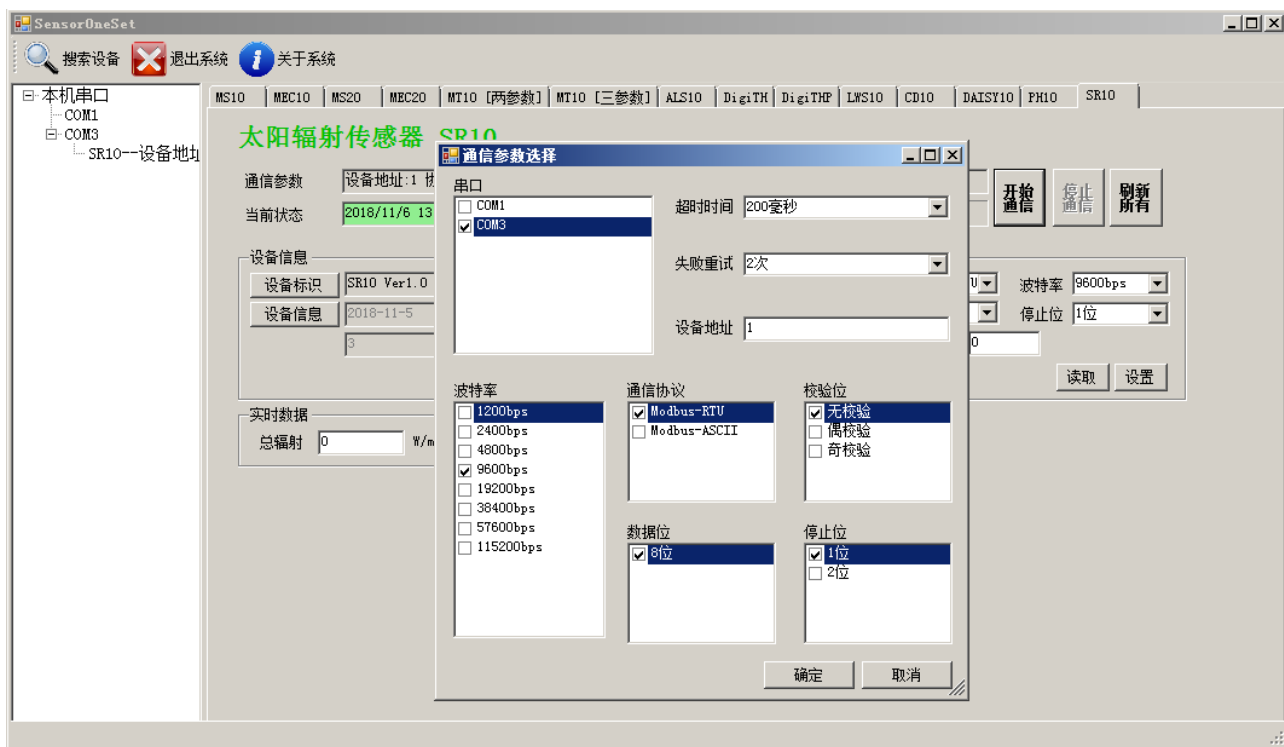
(3) 双击窗口左侧串口下列出的设备“ALS20-设备地址……”，其通讯参数会自动列到右侧的“电脑通讯设置”中。点击右侧的“开始”按钮，软件开始于模块进行通讯。





(4) 如需再次搜索模块, 请先点击“停止”按钮, 然后再次进行搜索。

(5) 如知道传感器的具体通信参数, 也可以在 ALS20 页面直接点击“开始通信”, 弹出“通信参数选择”对话框, 设置通信参数等信息后, 点“确定”关闭对话框后软件将与传感器进行通信。



(6) 通信成功后, 用户可修改各参数的值。

