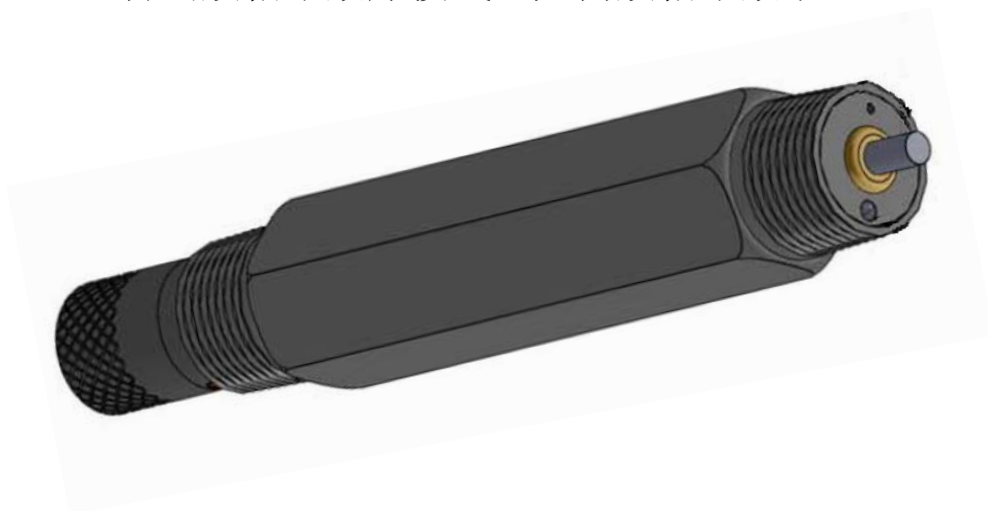
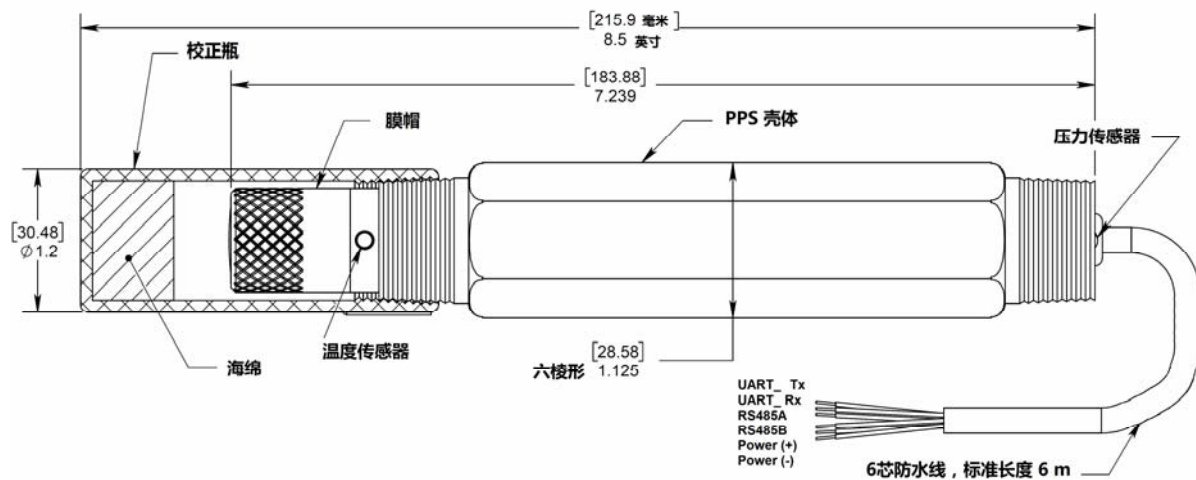


ASI 第二代工业式光学溶氧探头（RS485 输出）简介

- 光学技术（氧气淬灭荧光）原理
- 精确测量，容易维护
- 符合工业标准，两端都有 $\frac{3}{4}$ " NPT螺纹
- 各功能整合于探头，无需分体转换器
- RS485-Modbus 输出
- 防水压力传感器
- 温度压力自动补偿
- 自动盐度补偿当用户输入盐度值
- 首创膜帽系数库模式，无需膜帽系数卡



外观特征

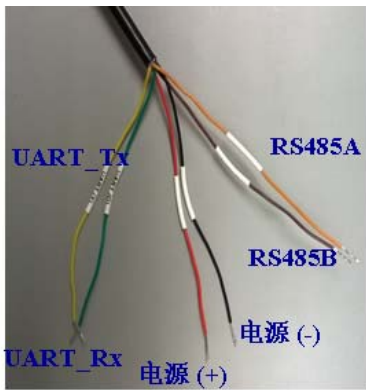


式制品（ prototypes）采用 3m 线长，末端接好了 2 个 USB 用于进行电脑测试。此外，探头末端的 LED 指示灯是用于模拟输出的探头，对于数字输出的产品，无需在意。

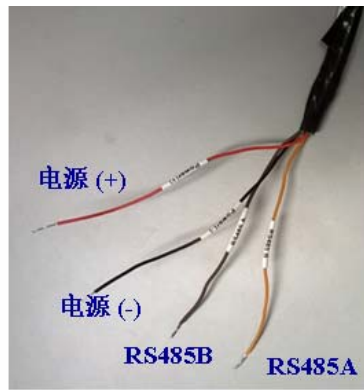


固定式接线; 防水级别: IP 68

可拆卸式接线; 防水级别: IP 67



6 线



4 线 (无升级功能)



6 线带USB用于电脑测试

输出线颜色	功能和接法
红	电源(+) 该样品需要 5 V 直流电
黑	电源 (-)
绿	UART_RX 用于探头升级
黄	UART_TX 用于探头升级
棕	RS485B
桔	RS485A

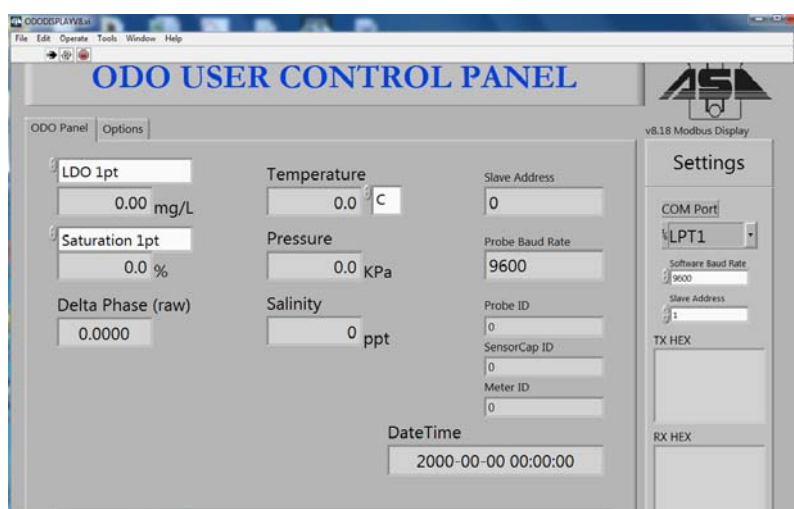
请先用厂家提供的测试软件在电脑上进行各项测试，以确定连接和各器件都正常。然后用户可去掉USB，和仪表进行连接测试。本品的Modbus 协议内容，地址分布附在说明书末页，用户可以此设置仪表和探头的通讯。如用户方的仪表有特别定义的内容，可能导致无法通讯。用户也可以将仪表端已有的协议，地址分布告知ASI, 我方可以重新定义Modbus 输出细节，以匹配用户已有的设置。如有必要，双方可以协商决定最佳通讯协议。

技术指标:

测量范围: 溶氧浓度: 0 -20 mg/L (ppm); 饱和度 %: 0 - 200% 工作温度: 0 - 50°C ; 储存温度: -20 - 70°C 工作大气压力: 40 - 115 kPa; 最大承受压力: 1000 kPa	溶氧补偿: 1) 全量程内温度自动补偿。 2) 0 -55 ppt 范围内得到盐度值后自动补偿。 3) 当压力传感器在水面上或者水下 20cm 以内, 通过实测压力值自动补偿。 4) 当压力传感器超过在水下 20cm, 通过缺省压力值自动补偿。缺省压力值是上一次一点校正时的当地气压。
精度: 溶氧: 0-100% < $\pm 1\%$; 100-200 % < $\pm 1.5\%$ 温度: $\pm 0.2\text{ }^{\circ}\text{C}$ 压力: $\pm 0.2\text{ kPa}$	一点校正 (100% 点): 空气饱和水 水饱和空气 (或校正瓶) 两点校正 (零点和 100% 点): 空气饱和水或者水饱和空气 零氧水
分辨率: (1 ppb = 0.001 mg/L) ~ 1 ppb, 当测试范围 <1 mg/L; 4 ~ 8 ppb, 当测试范围 < 10 mg/L; ~ 10 ppb, 当测试范围为 10-20 mg/L	响应时间: 溶氧: $T_{90} \sim 35\text{ s}$ (从 100 % 到 10 %) 温度: $T_{90} \sim 45\text{ s}$ (从 5 °C 到 45 °C, 如果搅拌)
防护等级: 固定式为 IP-68, 50m 水深一星期无漏水。 证书认证: RoHs, CE, C-Tick, 探头材料: PPS body 导线长度: 标准式为 6 m , 其他长度可选 膜帽寿命: 在正确操作和实验室环境下可使用 2 年	输入/输出 / 协议: 电源输入: 4.5 - 32 V DC 能耗: 在 5 V 供电下, 平均 60 mA 。 输出: 数字信号 协议: RS485-Modbus 或 UART

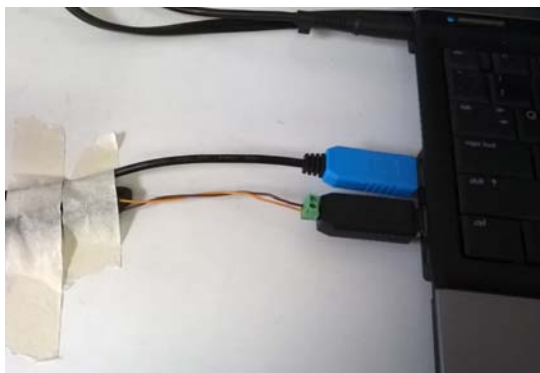
安装测试软件

- 1) 测试软件“ODO for RS485.exe”保存在附带的 SD 卡中，将其复制到电脑，点击即可使用。如果打不开软件，表明该电脑需安装一些支持程序。
- 2) 该测试软件需要的支持程序可在 SD 卡中可以找到，分别是“LVRTE90std.exe”，“Setup”和“LVRTE2013std”。一般安装好第一个支持程序就可打开测试软件，如果不能请将其其他两个都安装。
- 3) 当支持程序安装后，双击“ODO for RS485.exe”就可打开软件如下所示。如果不能请与 ASI 联系。



连接和测试

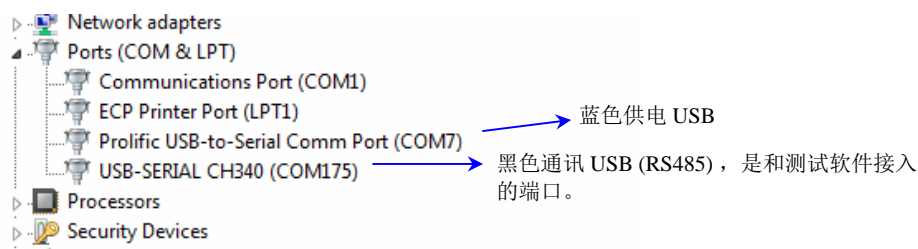
- 1) 如图所示，将探头与电脑连接。



- i) 将两个 USB 插入电脑。

- ii) 用胶带纸将导线固定，以免因导线的扯动导致 USB 接口的接触不良。

注: 这两个 USB 应该能被 Windows 7 系统识别，并自动安装相应的驱动。对于 Windows XP 操作系统，用户需要手动安装 RS485 USB 的驱动。如有需要，请联系厂家。

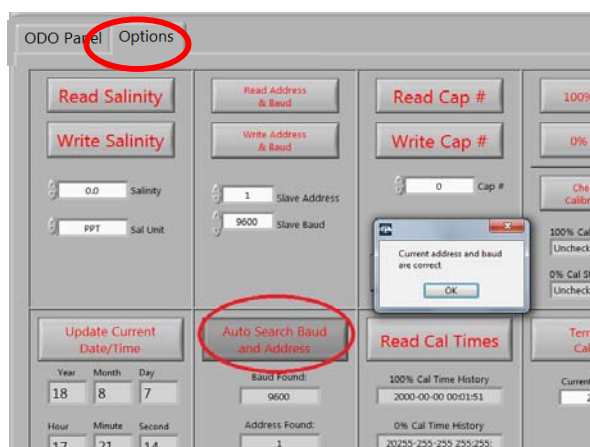


iii) 电脑装好驱动后，会显示相应的端口。用户也可以在 电脑属性中的设备管理中看到这些端口，如上图所示。



2a) 打开测试软件 “ODO for RS485” 选择正确的端口。一般几秒内会得到信号，如不能，请尝试步骤 2b。测试软件有两个显示界面，即 *ODO Panel* 和 *Options*。前者给出了在给定温度，压力和盐度条件下的溶氧浓度及饱和度，Modbus 的从站地址和波特率，电极和膜帽的系列

号。此外，探头的原始信号（与溶氧关联）显示在“Delta Phase (raw)”中。



2b) 如果软件中没有读数显示，请检查一下电脑和探头之间的连接。用户可通过拔下再插上两个 USB 刷新连接。对于第一次使用的情况，如左图所示，可能需要做 “Auto Search Baud and Address”，即在 *Options* 界面点击“Auto Search Baud and Address”。如果接通，软件会显示左图显示的信息，点击 ok 后，再回到 *ODO Panel* 中观察读数是否出现。

测试和校准

1) 在 *Options* 窗口中进行溶氧校正

(1) 一点校正

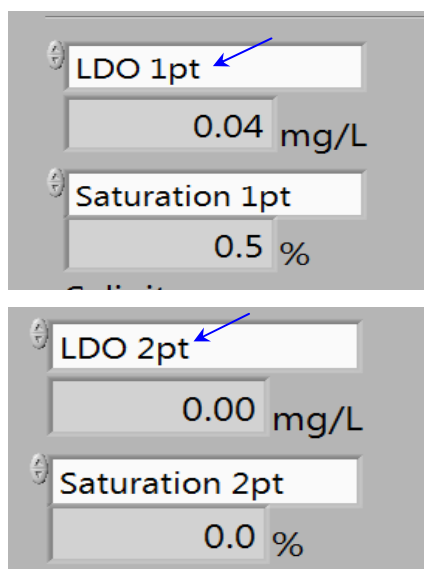
一点校正就是在空气饱和水中或者水饱和空气中（100 % 点）校准探头使其读数为



100%。当 ODO 在上述条件下 读数稳后，点击 “100% Point Cal” 当按键由灰变正常即完成一点校正。如果最终显示读数不在 $100 \pm 0.5 \%$ 范围内，可再

点击校正一次。

(2) 两点校正(零点和 100% 点)

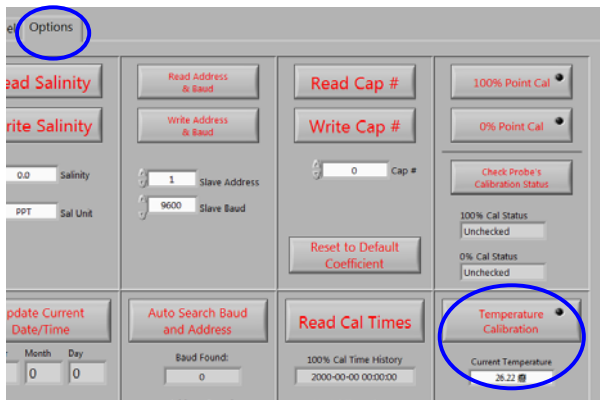


- 将 ODO 放入空气饱和水中，读数稳定后，点击 “100% Point Cal” 。
- 当溶氧读数变成 $\sim 100\%$ ，将探头放入零氧水中。
- 当读数稳定后（至少需要 2 分钟），点击 “0% Point Cal”。按键由灰变正常表明校准结束。
- 在 ODO Panel, 用户可选择显示一点或者两点校正下的溶氧读数。

注：1) 大多数使用场合中，两点校正是不必要的，除非用户需要一个更为精确结果在低浓度范围($< 0.5\text{ppm}$)。

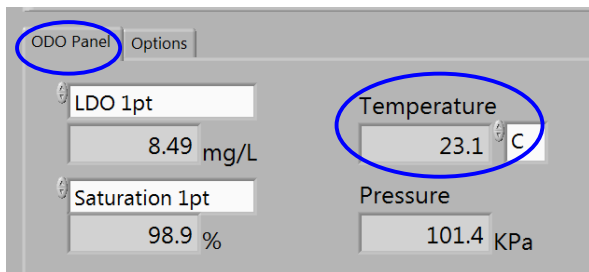
2) 两点校正的顺序必须是先 100% 然后再零点，不可以单一执行零点校正。

2) 温度的一点校正



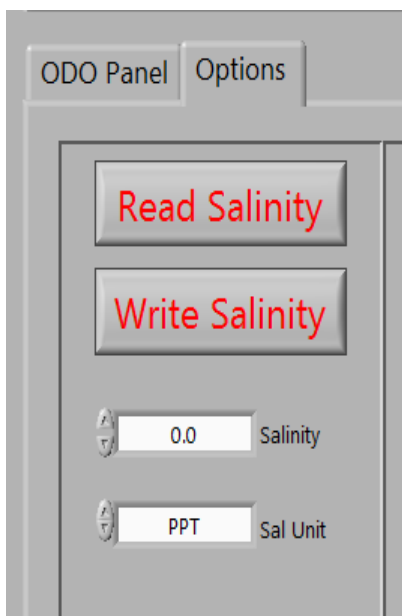
i) 在 *Options* 窗口, 在“Current Temperature”下输入 ODO 温度传感器所在环境的温度。

ii) 点击 “Temperature Cal”, 按键由灰变正常表明校准结束。



iii) 在 ODO Panel 窗口检查温度显示是否变成指定温度, 如果不是可再做一次校准。

3) 盐度补偿



(i) 在 *Options* 窗口, 点击 “READ SAL” 可在“Salinity”显示当前探头内保存的上一次输入的盐度值。盐度值的输入/更改在“Salinity”处进行。

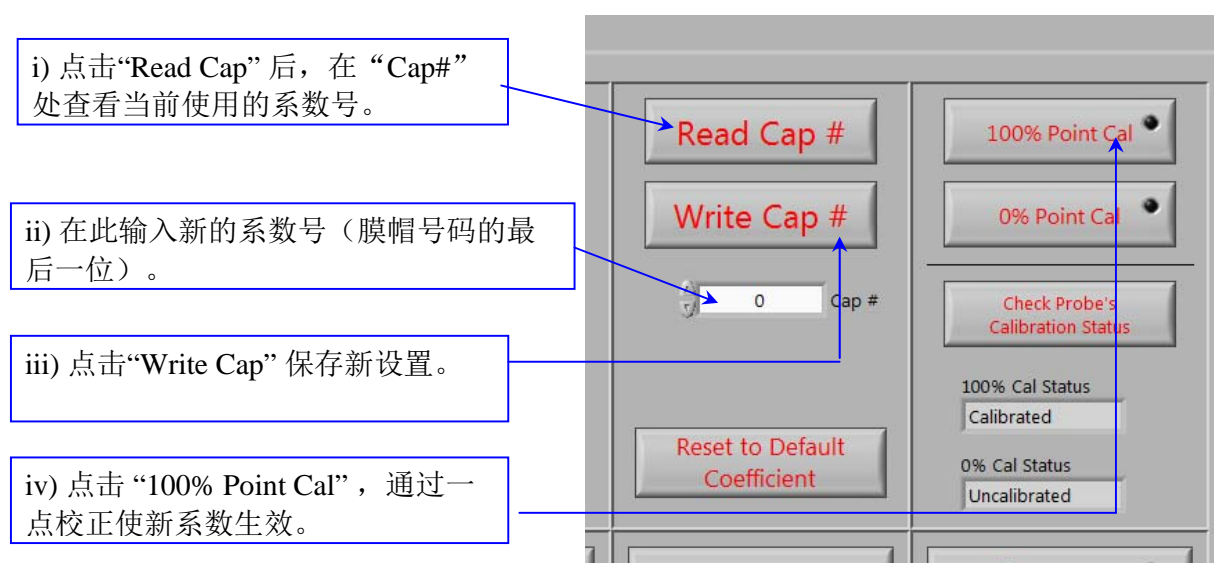
(ii) 当“Salinity”不是零 或者盐度单位不是 off 时。盐度补偿会自动进行。

(iii) 新输入的盐度值要等到点击 “Write Salinity” 后才生效。

(iv) 盐度补偿不会自动停止, 所以当 ODO 返回到淡水环境后, 用户要么将盐度值清零, 要么在 “Sal Unit” 中选择 “off”, 无论哪种操作, 都需要再点击“Write Salinity”使之生效。

更换膜帽之后，选择膜帽对应的系数

每个膜帽侧面都有 4 位数的号码 (SN)，该号码的最后一位为系数号。对于没有号码的膜帽（特别是不锈钢膜帽），表明是系数号为 0 的膜帽。日后刚换同样的无号码膜帽，无需特殊处理。如果更换的是尾数不同的膜帽，需要在探头的内存中选择一套与原有膜帽不同的系数。步骤如下：



注：出厂前探头的内存，已经储存了 6~8 套系数，ASI 经过多年的摸索发现，在相当一段时间内（1~2 年），各批次的膜帽差异不大，2 年后的膜帽可能与以前的膜帽会有一定差异，虽然不能精确控制差异的走向，但是能预测差异一般就在几种可能范围内。所以，6~8 套系数已经能够应付几年的膜帽变化。此外，本品的系数库是可以更新的。

附录：Gen 2 ODO 的 RS485-Modbus 通讯协议

定义和术语（Terms）：

- 主站：用户的仪表
- 从站：溶氧仪探头
- TX：从站 RS485 传输（输出）
- RX：从站 RS485 接收（输入）
- Char：8 比特大端（big endian）
- Int：16 比特大端（big endian）
- Float：32 比特大端（big endian）

单方控制（Unilateral controls）：

- 主站能够单方通过将位于第 5 脚的 I0 拉到低电平，实现从站重启。
 - 在其他时刻，第 5 脚必须被拉到高电平或者保持悬浮。
 - 除了固件升级，主站在启动从站后 5 秒内不得通讯，因为将 Pin 5 拉到高位需要大致 5 秒。

命令结构（Command structure）：

- 在 buildMSG.c 和 ProcessCMD.c 中详细说明了从站如何建立处理命令和数据包。
- 完成上一个应答后，新的命令应在 50 毫秒后发送。
- 如果期待的从站应答在 25 毫秒后还没出现，就显示通讯错误。
 - 注：必须只有一个主站和一个从站在线
 - 注：没有从站被使用
- 所有发送到从站的 Hex 必须有起始标头“1001”和结束页尾“1003”和 CRC16。
- 从站发出的 Hex 没有标头，页尾和 CRC16。

传输系列结构（Serial Transmission structure）：

- Each RS485 transmission will have: one start bit, 8 data bits, no parity bit, and two stop bit.（每个 RS485 传输包含一个起始符，8 个数据比特，两个终止符，不包含校验位。
- RS485 will run at a default baud rate of 9600（缺省波特率是 9600）
- The 8 data bits transmitted after the start bit are most significant bit first（跟随起始位，接着发送 8 个数据位，首先是最高位）
- Transmission Mode must be Standard RTU。（传输模式必须是标准 RTU）

位序

Start bit	1	2	3	4	5	6	7	8	Stop bit
-----------	---	---	---	---	---	---	---	---	----------

定时（Timing）：

- 如要运行启动程序（Bootloader），必须要在通电或 reset 后 5 秒内进行。
- 除了固件升级，通电或者 reset 后 5 秒内不执行任何一个命令，5 秒后再运行第一个命令。
- 发送命令后 200 毫秒后无应答，可断定为发生了超时。

命令表（Command List）：

命令和应答总是开始于：

- 第一个字节：探头地址
- 第二个字节：0x14 = 读取包，0x15 = 写入包
- 如果一个写入值是---，表明它必须在所有描述的情况下被写入
- 如果一个应答值是---，表明无意义，可以丢掉

CMD: 读取探头数据（Read Probe Data）

原始 Hex: 0114 0009 81DA

地址	读	Sub CMD	CRC
0x01	0x14	0x0009	0x81DA
1	20	9	

示例：探头的应答

Raw Hex: 0114 0009 0036 A273 0B41 65FC CF42 5228 CC40 F03E 9842 F503 F200 7C5E 7243 0000 0000 0100 A000 2200 0200 0000 0000 0000 0100 0000 0000 0000 0100 0400

地址	读	Sub CMD	# of Bytes*	溶氧浓度	溶氧饱和度	----	----	探头压力
0x01	0x14	0x0009	0x0036	0xA273 0B41	0x65FC CF42	0x5228 CC40	0xF03E 9842	0xF503
1	20	9	54	8.71573	103.993	----	----	101.3
----	探头温度	时间 (y/m/d h:m:s)	错误提示	index	膜帽 ID	表头 ID	探头 ID	盐度 PPT
0x0xF200	0x7C5E 7243	0x0000 0000 0100	0xA000	0x2200	0x0200 0000	0x0000 0000	0x0100 0000	0x0000 0000
----	24.2369	0/0/0 0:1:0	0b10100000	34	2	0	1	0
从站地址	波特率							
0x0100	0x0100							
1	9600							

示例：探头的应答

原始 Hex: 0114 0009 0036 FE67 D340 5AA1 D742 0000 0000 0000 0000 F103 F900 7088 7943 1208 0A0B 152B A000 6F00 66D6 1200 0000 0000 0100 0000 0000 5042 0100 0400

地址	读	Sub CMD	# of Bytes	溶氧浓度	溶氧饱和度	----	----	探头压力
0x01	0x14	0x0009	0x0036	0xFE67 D340	0x5AA1 D742	0x0000 0000	0x0000 0000	0xF103
1	20	9	54	6.606	107.8	----	----	100.9
----	探头温度	时间 (y/m/d h:m:s)	Error	index	膜帽 ID	表头 ID	探头 ID	盐度 PPT
0x0xF900	0x7088 7943	0x1208 0A0B 152B	0xA000	0x6F00	0x66D6 1200	0x0000 0000	0x0100 0000	0x0000 5042
----	24.95	18/8/10 11:21:43	0b10100000	111	1234534	0	1	52.0
地址	Baud							
0x0100	0x0100							
1	9600							

*注：“# of Bytes”是被发送数据的字节数。（包括从上表中的“溶氧浓度”一直到“波特率”）。

“探头压力”是被乘以 10 的压力读数, 因为数据传输过程中，需要是一个整数，不是整数的都会被取整。

“探头温度”是被乘以 10 的温度读数。

“Error”是一个 16 位字节的错误提示。

“Index” 是探头被启动后发送的样品数。

CMD: 写入盐度值（Write Salinity）

值 = 0, 单位 = off

原始 Hex: 0115 0001 0000 0000 0200 0000 0000 2937

盐度单位: 0=ppt, 1=ms/cm, 2=off

地址	写	Sub CMD	----	盐度值/10	单位	----	CRC
0x01	0x15	0x0001	0x0000	0x0000	0x0200	0x0000 0000	0x2937
1	21	1	----	0	off	----	----

Response:

地址	写	Sub CMD	----	盐度值/10	单位	----	
0x01	0x15	0x0001	0x0008	0x0000	0x0200	0x0000 0000	
1	21	1	----	0	off	----	

值 = 1234.5, 单位 = PPT

原始 Hex: 0115 0001 0000 3930 0000 0000 0000 2C3F

盐度单位: 0=ppt, 1=ms/cm, 2=off

地址	写	Sub CMD	----	盐度值/10	单位	----	CRC
0x01	0x15	0x0001	0x0000	0x3930	0x0000	0x0000 0000	0x2C3F
1	21	1	----	1234.5	PPT	----	----

Response:

原始 Hex: 0115 0001 0008 3930 0000 0000 0000

地址	写	Sub CMD	----	盐度值/10	单位	----	
0x01	0x15	0x0001	0x0008	0x3930	0x0000	0x0000 0000	
1	21	1	----	1234.5	PPT	----	

CMD: 读取输入的盐度值（读 Salinity）

原始 Hex: 0114 0001 0000 6009

盐度单位: 0=PPT, 1=ms/cm, 2=off

地址	读	Sub CMD	----	盐度值/10	单位	----	CRC
0x01	0x14	0x0001	0x0000	0x0000	0x0200	0x0000 0000	0x2937
1	21	1	----	0	off	----	----

示例应答: 值 = 0, 单位 = off

原始 Hex: 0114 0001 0008 0000 0200 0000 0000

地址	读	Sub CMD	----	盐度值/10	单位	----	
0x01	0x14	0x0001	0x0008	0x0000	0x0200	0x0000 0000	
1	20	1	----	0	off	----	

示例应答: 值 = 5433.2, 单位 = PPT

原始 Hex: 0114 0001 0008 3CD4 0000 9AC9 A945

地址	读	Sub CMD	----	盐度值/10	单位	----	
0x01	0x14	0x0001	0x0008	0x3CD4	0x0000	0x9AC9 A945	

1	20	1	----	5433.2	PPT	----
---	----	---	------	--------	-----	------

CMD: 写入日期时间 (Write Date/Time)**值: 2018/8/15 16:39:57**

原始 Hex: 0115 0006 0000 1208 0F10 2739 CCED

地址	写	Sub CMD	----	YY/MM/DD hh:mm:ss	CRC
0x01	0x15	0x0006	0x0000	0x1208 0F10 2739	0xCCED
1	21	6	----	12/8/15 16:39:57	----

值: 2001/2/3 4:5:7

原始 Hex: 0115 0006 0000 0102 0304 0507 8C6B

地址	写	Sub CMD	----	YY/MM/DD hh:mm:ss	CRC
0x01	0x15	0x0006	0x0000	0x0102 0304 0507	0x8C6B
1	21	6	----	1/2/3 4:5:7	----

Response for all:

原始 Hex: 0115 0006 0000

地址	写	Sub CMD	----
0x01	0x15	0x0006	0x0000

CMD: 写入从站地址和波特率 (Write Address/Baud)

从站地址不能为 0, 缺省值为 1

波特率缺省值为 9600

地址 = 1, Baud = 9600

原始 Hex: 0115 0003 0000 0100 0400 E36E

波特率 : 0=300, 1=2400, 2=2400, 3=4800, 4=9600, 5=119200, 6=38400, 7=115200

从站地址	写	Sub CMD	----	地址	波特率	CRC
0x01	0x15	0x0003	0x0000	0x0100	0x0400	0xE36E
1	21	3	----	1	9600	----

响应 @ 初始地址/波特率:

原始 Hex: 0115 0003 0000 0100 0400

地址	写	Sub CMD	----	地址	波特率
0x01	0x15	0x0003	0x0000	0x0100	0x0400
1	21	3	----	1	9600

地址 = 123, 波特率 = 115200

原始 Hex: 0115 0003 0000 7B00 0700 FA86

波特率 rate 值 s: 0=300, 1=2400, 2=2400, 3=4800, 4=9600, 5=119200, 6=38400, 7=115200

地址	写	Sub CMD	----	地址	波特率	CRC
0x01	0x15	0x0003	0x0000	0x7B00	0x0700	0xFA86
1	21	3	----	123	115200	----

应答 @ 初始 地址/波特率:

原始 Hex: 0115 0003 0000 7B00 0700

地址	写	Sub CMD	----	地址	波特率
0x01	0x15	0x0003	0x0000	0x7B00	0x0700
1	21	3	----	123	115200

CMD: 写入膜帽的 ID (Write Sensor Cap #)

Cap # = 0

原始 Hex: 0115 0002 0004 0000 0000 0152

地址	写	Sub CMD	----	Cap #	----	CRC
0x01	0x15	0x0002	0x0004	0x0000	0x0000	0x0152
1	21	2	----	0	----	----

应答 = 成功

原始 Hex: 0115 0002 0004 0000 0500

地址	写	Sub CMD	----	Cap #	----
0x01	0x15	0x0002	0x0004	0x0000	0x0500
1	21	2	----	0	----

Cap # = 1234 or 4 (如果第 4 号系数不在探头的数据库里)

原始 Hex: 0115 0002 0004 D204 0000 79EB

地址	写	Sub CMD	----	Cap #	----	CRC
0x01	0x15	0x0002	0x0004	0xD204	0x0000	0x79EB
1	21	2	----	1234	----	----

应答 = 数据库没有找到相应的系数, 请使用缺省系数

原始 Hex: 0115 0002 0004 FFFF FFFF

地址	写	Sub CMD	----	Cap #	----
0x01	0x15	0x0002	0x0004	0xFFFF	0xFFFF
1	21	2	----	-1	----

CMD: 读取膜帽 ID (Read Sensor Cap #)

Cap # = 0

原始 Hex: 0114 0002 C01D

地址	写	Sub CMD	CRC
0x01	0x14	0x0002	0xC01D
1	20	2	----

示例应答 = 9785

原始 Hex: 0114 0002 0004 3926 0A00

地址	写	Sub CMD	----	地址	----
0x01	0x14	0x0002	0x0004	0x3926	0x0A00
1	20	2	----	9785	----

示例应答 = 使用了缺省系数

原始 Hex: 0114 0002 0004 FFFF FFFF

地址	写	Sub CMD	----	地址	----
0x01	0x14	0x0002	0x0004	0xFFFF	0xFFFF
1	20	2	----	-1	----

CMD: Check Calibration Status

原始 Hex: 0114 0005 0000 21C8

地址	写	Sub CMD	----	CRC
0x01	0x14	0x0005	0x0000	0xC01D
1	20	5	----	----

示例应答: 1 校正正在执行

原始 Hex: 0114 0005 0002

地址	写	Sub CMD	Status
0x01	0x14	0x0005	0x0002
1	20	5	1 pt

示例应答: 温度校正正在执行

原始 Hex: 0114 0005 0004

地址	写	Sub CMD	Status
0x01	0x14	0x0005	0x0004
1	20	5	temperature

示例应答: 没有正在执行的校正

原始 Hex: 0114 0005 0000

地址	写	Sub CMD	Status
0x01	0x14	0x0005	0x0000
1	20	5	none

CMD: 执行校正

溶氧的一点校正

原始 Hex: 0115 0005 0002 9DC9

地址	写	Sub CMD	执行	CRC
0x01	0x15	0x0005	0x0002	0x9DC9
1	21	5	1 pt	----

Response:

原始 Hex: 0114 0005 0002

地址	写	Sub CMD	执行
0x01	0x15	0x0005	0x0002
1	21	5	1 pt

Run temperature calibration = 26.22 °C

原始 Hex: 0115 0005 0004 0A3E CFB7

地址	写	Sub CMD	执行	Temperature/100°C	CRC
0x01	0x15	0x0005	0x0002	0x0A3E	0x CFB7
1	21	5	temp	26.22°C	----

Response:

原始 Hex: 0115 0005 0004

地址	写	Sub CMD	执行
0x01	0x15	0x0005	0x0004
1	21	5	成功

数据包定义:

- 所有只读的值可以被改写为任何数值，但是这些改写后的值会被探头丢掉。

探头数据包: (Sub CMD 0x09)

(float) LDO mg/L , 溶氧浓度 , mg/L
(float) saturation % , 溶氧饱和度 , %
(float) Filler Value , 填充值
(float) Filler Value , 填充值
(int16) pressure *10 , 压力×10
(int16) Filler Value , 填充值
(float) temperature *10 , 温度×10
(int8) year , 年
(int8) month , 月
(int8) day , 日
(int8) hour , 时
(int8) minute , 分
(int8) second , 秒
(uint16) 错误提示 (error bits) : 1= 一点校正不成功, 3 = 无温度信号, 4= 溶氧浓度超过量程, 5= 没有压力信号, 6= 压力信号超过量程
(int16) 在探头重置 (reset) 后, 记录的数目
(uint32) Sensor cap ID, 膜帽 ID
(uint32) Meter ID, 户表头 ID
(uint32) Probe ID, 探头 ID
(float) 盐度值 (PPT)
(int16) 地址
(int16) 波特率
(float) 相差 (度)

盐度包: (Sub CMD 0x01)

(unsigned int) 盐度值×10, 单位 如下
(unsigned int) 盐度单位: 0=PPT, 1=mS/cm, 2= off (就是不进行盐度补偿)
(float) 被转化成 PPT 的盐度值 (只能读)

校正状态包 : (Sub CMD 0x05)

(int) 校正状态: 1 = 无任何校正正在运行, 2 = DO 一点校正正在运行, 4 = 温度校正正在运行

膜帽 ID 包: (Sub CMD 0x02)

(int) 膜帽 IDCap #
(int) 保留值 (只读)

CRC 部分:

// CRC.cpp :定义控制台应用程序的入口

```
unsigned short CRC Update1(unsigned char data1, unsigned short crc);
```

```
const unsigned short crctab[256] = {  
    0x0000, 0xc0c1, 0xc181, 0x0140, 0xc301, 0x03c0, 0x0280, 0xc241,  
    0xc601, 0x06c0, 0x0780, 0xc741, 0x0500, 0xc5c1, 0xc481, 0x0440,  
    0xcc01, 0x0cc0, 0x0d80, 0xcd41, 0x0f00, 0xcfc1, 0xce81, 0x0e40,  
    0x0a00, 0xcac1, 0xcb81, 0x0b40, 0xc901, 0x09c0, 0x0880, 0xc841,  
    0xd801, 0x18c0, 0x1980, 0xd941, 0x1b00, 0xdbc1, 0xda81, 0x1a40,  
    0x1e00, 0xdec1, 0xdf81, 0x1f40, 0xdd01, 0x1dc0, 0x1c80, 0xdc41,  
    0x1400, 0xd4c1, 0xd581, 0x1540, 0xd701, 0x17c0, 0x1680, 0xd641,  
    0xd201, 0x12c0, 0x1380, 0xd341, 0x1100, 0xd1c1, 0xd081, 0x1040,  
    0xf001, 0x30c0, 0x3180, 0xf141, 0x3300, 0xf3c1, 0xf281, 0x3240,  
    0x3600, 0xf6c1, 0xf781, 0x3740, 0xf501, 0x35c0, 0x3480, 0xf441,  
    0x3c00, 0xfcc1, 0xfd81, 0x3d40, 0xff01, 0x3fc0, 0x3e80, 0xfe41,  
    0xfa01, 0x3ac0, 0x3b80, 0xfb41, 0x3900, 0xf9c1, 0xf881, 0x3840,  
    0x2800, 0xe8c1, 0xe981, 0x2940, 0xeb01, 0x2bc0, 0x2a80, 0xea41,  
    0xee01, 0x2ec0, 0x2f80, 0xef41, 0x2d00, 0xedc1, 0xec81, 0x2c40,  
    0xe401, 0x24c0, 0x2580, 0xe541, 0x2700, 0xe7c1, 0xe681, 0x2640,  
    0x2200, 0xe2c1, 0xe381, 0x2340, 0xe101, 0x21c0, 0x2080, 0xe041,  
    0xa001, 0x60c0, 0x6180, 0xa141, 0x6300, 0xa3c1, 0xa281, 0x6240,  
    0x6600, 0xa6c1, 0xa781, 0x6740, 0xa501, 0xa5c0, 0xa480, 0xa441,  
    0x6c00, 0xacc1, 0xad81, 0x6d40, 0xaf01, 0x6fc0, 0x6e80, 0xae41,  
    0xaa01, 0x6ac0, 0x6b80, 0xab41, 0x6900, 0xa9c1, 0xa881, 0x6840,  
    0x7800, 0xb8c1, 0xb981, 0x7940, 0xbb01, 0x7bc0, 0x7a80, 0xba41,  
    0xbe01, 0x7ec0, 0x7f80, 0xbf41, 0x7d00, 0xbdc1, 0xbc81, 0x7c40,  
    0xb401, 0x74c0, 0x7580, 0xb541, 0x7700, 0xb7c1, 0xb681, 0x7640,  
    0x7200, 0xb2c1, 0xb381, 0x7340, 0xb101, 0x71c0, 0x7080, 0xb041,  
    0x5000, 0x90c1, 0x9181, 0x5140, 0x9301, 0x93c0, 0x9280, 0x9241,  
    0x9601, 0x56c0, 0x5780, 0x9741, 0x5500, 0x95c1, 0x9481, 0x5440,  
    0x9c01, 0x5cc0, 0x5d80, 0x9d41, 0x5f00, 0x9fc1, 0x9e81, 0x5e40,  
    0x5a00, 0x9ac1, 0x9b81, 0x5b40, 0x9901, 0x99c0, 0x9880, 0x9841,  
    0x8801, 0x48c0, 0x4980, 0x8941, 0x4b00, 0x8bc1, 0x8a81, 0x4a40,  
    0x4e00, 0x8ec1, 0x8f81, 0x4f40, 0x8d01, 0x4dc0, 0x4c80, 0x8c41,  
    0x4400, 0x84c1, 0x8581, 0x4540, 0x8701, 0x47c0, 0x4680, 0x8641,  
    0x8201, 0x42c0, 0x4380, 0x8341, 0x4100, 0x81c1, 0x8081, 0x4040  
};
```

```
unsigned short CRCUpdate1(unsigned char data1, unsigned short crc){  
    return((crc >> 8) ^ crctab[(unsigned char) (crc ^ data1)]); // >>是右移, ^ 是异或
```