
数字式 LVDT 传感器编程手册

(Modbus 版 3.1)

2020 年 06 月 01 日

数字式 LVDT 传感器编程手册

一、数字式 LVDT 传感器介绍

1、使用说明

(1)、数字式 LVDT 传感器额定供电电压为 DC9V~DC28V，红线接电源正极，黑线接电源负极，蓝线接 RS485 引脚 A，白线接 RS485 引脚 B；

(2)、传感器上电后，需要经过 1.5 秒左右时间才能正常通信，其中启动后 1 秒左右的时间用来确定是否需要系统升级，之后程序初始化，然后采样并准备通信接收发数据，因此上电后最好等待片刻再操传感器；

(3)、传感器出厂默认通信参数为：9600bps、偶校验、RTU 通信模式，此参数也可根据用户需求更改，用户也可自行更改，设置方式下文会有具体说明；

(4)、传感器的通信方式 RS485 为半双工方式。只有在接收到上位机有效的指令消息后才向上位机发送回馈消息。空闲状态一直处于接收上位机指令消息状态；

(5)、同一批次传感器出厂都经过统一编号(传感器地址)，每个传感器都有唯一编号，出厂时编号即为传感器外壳上印制的编号，此编号用户可通过下文介绍的指令进行更改(不建议自主更改)，同一网络中不允许有相同编号的传感器接入，否则通信不能正常进行；

(6)、传感器参数如传感器地址、波特率和奇偶校验位等参数设定后，无需重启传感器即能立即生效，只有恢复出厂默认设置指令，恢复成功后，传感器会自动重启，需要等待 1.5 秒左右时间恢复正常运行。

2、功能参数

- (1)、使用标准 Modbus 通讯协议，同一个网络中支持最多接入 247 路传感器；
- (2)、支持 RTU 和 ASCII 两种通讯模式，可自主设置进行切换(默认 RTU 模式)；
- (3)、位移 ADC 为 16 位，分辨率高达 1 微米，自动更新且已进行数字滤波；
- (4)、温度 ADC 为 12 位，分辨率为 0.1 摄氏度，自动更新且已进行数字滤波；
- (5)、位移数字量支持补偿标定，采用高阶多项式拟合修正，误差更小；
- (6)、位移和温度支持采样频率设定，范围为 1~50Hz，采样值已进行数字滤波；
- (7)、上位机软件连接支持自动匹配功能，在通信参数设置混淆的情况下，可通过上位机软件一键匹配完成连接(一键匹配连接时只支持接入 1 路传感器)；
- (8)、支持在线升级，方便更新版本，使用最新版的功能。

3、注意事项

(1)、传感器波特率设置最好不要低于 9600bps，波特率为 2400bps 时，RS485 每发送 1 个字节的时间约为 5ms 左右，若在 ASCII 通信模式下，一帧最简短的消息接收发时间就在 200ms 以上，会降低实时性；

(2)、读位移和温度数据一般频率不高于 10Hz，最快支持 25Hz 的频率(25Hz 时波特率要设置成 115200bps，RTU 通信模式，传感器默认位移和温度采样频率为 10Hz，读位移和温度数据超过 10Hz 时，需要设置位移和温度采样频率为 20Hz 或 50Hz，否则读数没意义，超频到 50Hz，数据传输能进行，但有一机率传输丢失数据)，写入传感器地址、波特率、奇偶校验等参数时频率最高不要超过 10Hz，因写 Flash 需要花费一定时间，且 Flash 有写入有次数限制，一般为 10 万次左右，所以写入频率不要太高；

二、数字式 LVDT 传感器参数

1、数字式 LVDT 传感器的参数

传感器参数有：传感器地址、波特率、奇偶校验、输出模式、位移、温度、位移补偿使能、位移和温度采样频率、位移清零/取消清零、恢复出厂设置。传感器地址是传感器在 Modbus 网络中的地址，具有唯一性，波特率、奇偶校验和输出模式是 Modbus 网络通信参数的设置，位移是由传感器输入的模拟信号变化而来的位移值，温度为环境摄氏温度值，位移补偿使能通过内部保存的标定值对位移偏差进行补偿，逻辑清零/取消清零可以在使用过程中设置任意位置为位移零点，恢复出厂设置能还原以上所有参数默认值，确保在参数设置混淆下恢复原状。

2、数字式 LVDT 传感器参数对应寄存器属性

表 1 数字式 LVDT 传感器寄存器表

寄存器地址	描述	符号	类型	操作功能码(Hex)
0x0002	温度高 16 位	TempH	R	04
0x0003	温度低 16 位	TempL	R	04
0x0004	位移高 16 位	PosH	R	04
0x0005	位移低 16 位	PosL	R	04
0x0030	设备地址	Addr	R/W	03/10
0x0031	波特率	Baud	R/W	03/10
0x0032	奇偶校验	Parity	R/W	03/10
0x0034	补偿使能	Compensation	R/W	03/10

0x003F	位移和温度采样频率	Frequency	R/W	03/10
0x0050	位移清零/取消清零	SetLogicalZero	W	05
0x0051	恢复出厂设置	Reset	W	05
0x0060	输出模式	OutPutMod	R/W	03/10

注：1、寄存器皆为 16 位，每个寄存器都有唯一的地址编号，且每个寄存器操作功能码都是确定的；

2、位移和温度数据均占用两个寄存器，读取数据要读两个寄存器的内容；

3、类型：R 表示只读，W 表示只写，R/W 表示该寄存器即可以读又可以写操作。

3、数字式 LVDT 传感器默认参数

数字式 LVDT 传感器在出厂时对参数进行了默认设置，如果传感器参数进行了修改，也可以通过操作“恢复出厂设置”寄存器恢复默认值，默认值如表 2 所示。

表 2 数字式 LVDT 传感器默认参数表

寄存器地址	描述	符号	默认值	含义
0x0002	温度高 16 位	TempH	0x0000	温度高 16 位(保留)
0x0003	温度低 16 位	TempL	随机	温度数据，实时更新
0x0004	位移高 16 位	PosH	随机	位移整数，实时更新
0x0005	位移低 16 位	PosL	随机	位移小数，实时更新
0x0030	传感器地址	Addr	1~247	传感器外壳编号一致
0x0031	波特率	Baud	0x0003	波特率为 9600bps
0x0032	奇偶校验	Parity	0x0002	偶校验
0x0034	补偿使能	Compensation	0x0002	补偿失能
0x003F	位移和温度采样频率	Frequency	0x0004	采样频率为 10Hz
0x0050	位移清零/取消清零	SetLogicalZero	-	只写
0x0051	恢复出厂设置	Reset	-	只写
0x0060	输出模式	OutPutMod	0x0001	通信为 RTU 模式

注：补偿使能默认值为 0x0002，补偿即为失能状态(未生效)，补偿是出厂时位移已经通过多点校准标定，并对位移进行了高次多项式拟合，只有在使能状态(写入数值 0x0001)，位移输出值会自动拟合修正，此时位移输出值误差会更小。

传感器地址有效范围为 1-247，0 为广播地址，以广播地址发送查询指令时，所有传感器均能接收并回复，因此主广播地址时只允许 1 路传感器接入，否则通信不

能正常进行。传感器地址默认值为传感器产品外壳上注明的传感器编号值，原则上每个传感器有唯一编号，也可以通过指令修改传感器地址值，但要注意同一个网络中不允许接入相同地址值的传感器，否则通信不能正常进行。

4、数字式 LVDT 传感器参数设置区间

数字式 LVDT 传感器寄存器类型具有 W(可写)属性的，可通过指令写入数值更改传感器参数，写入的参数值需要符合规范才能正常写入，否则写入失败。传感器寄存器可写入数值区间及含义如表 3 所示。

表 3 数字式 LVDT 传感器参数取值区间

参数名称	取值及含义
传感器地址	1~247, 表示传感器编号, 每个传感器取值不同
波特率	1-2400bps, 2-4800bps, 3-9600bps, 4-19200bps, 5-38400bps、 6-57600bps, 7-115200bps
奇偶校验	1-奇校验, 2-偶校验, 3-无校验
补偿使能	1-补偿使能, 2-补偿失能
位移和温度采样频率	1-1Hz, 2-2Hz, 3-5Hz, 4-10Hz, 5-20Hz, 6-50Hz, 数字量 已进行数字滤波, 稳定性较好
位移清零/取消清零	0x0000-取消清零, 0xFF00-位移清零
恢复出厂设置	0xFF00-恢复出厂设置, 设置成功后, 传感器会自动重启, 等 待 1.5s 左右传感器恢复正常运行
输出模式	1-RTU 模式, 2-ASCII 模式

注：1、位移和温度采样频率越低时，数据稳定性越好，但实时性会越低(采样值为周期内的平均值)，频率越高时，数据稳定性越低，但实时性越高，用户需要根据自身的需要设定该频率值；

2、默认波特率为 9600bps，即如果用 0x03 功能码读取 0x31 地址寄存器的值时则为 0x0003(寄存器为 16 位)，当把波特率由 9600bps 更改为 115200bps 时，就用 0x10 功能码向 0x31 地址寄存器内写入数据 0x0007；

3、上表中不存在的值不要写入，写入的值不存在时，传感器会返回 0x03 错误码，提示用户写入数据错误。

5、数字式 LVDT 传感器参数数据格式

(1)、传感器地址占 2 个字节，高 8 位在前(高 8 位为 0x00)，低 8 位在后。有效地址范围为 1-247，0 为广播地址(注：传感器地址以参数读写时传输数据占 2 字节，但 Modbus 指令第 1 字节为传感器地址时仅占用 1 个字节，需要注意)；

(2)、传感器位移数据占 4 个字节 32 位，第 1，2 个字节表示符号与整数部分，第 3，4 个表示小数部分的 1000 倍。

表 4 传感器位移数据格式

字节位	DataH16(31~16)	DataL16(15~0)
说明	整数部分(最高位为符号位)	小数部分

举例 1：传感器位移值 311.328mm

符号为正，整数部分为 311，小数部分为 $0.328 \times 1000 = 328$ ，转换为 16 进制后为

0x0137(整数 16 位带符号位) 0x0148(小数 16 位)

因此 311.328 在数字式 LVDT 传感器表示为 0x01 0x37 0x01 0x48 (0x1370148)；若传感器位移值-311.328m，则符号为负，整数部分为-311，小数部分为 $0.328 \times 1000 = 328$ ，转换为 16 进制后

0x8137(整数 16 位带符号位) 0x0148(小数 16 位)

在数字式 LVDT 传感器表示为 0x81 0x37 0x01 0x48 (0x81370148)。

举例 2：若接收到数字式 LVDT 传感器传送过来位移数据为 0x81 0x37 0x01 0x48 (0x81370148)，则位移值解析过程如下：

- a、 $\text{DataH16} = (0x81 \times 256) + 0x37 = 0x8137$ ， $\text{DataL16} = (0x01 \times 256) + 0x48 = 0x0148$ ；
- b、 $\text{DataH16} > 0x8000$ ，判定此数据为负数，否则为正数；
- c、整数数值 $\text{Si} = 0x8000 - \text{DataH16} = -311$ ，小数数值 $\text{Sd} = 0x0148 / 1000 = 0.148$ (为正数时整数数值 $\text{Si} = \text{DataH16}$)；
- d、位移值 $\text{S} = \text{Si} - \text{Sd} = -311.148$ (为正数时位移值 $\text{S} = \text{DataH16} + \text{DataL16}$)。

(3)、传感器温度数据占 4 个字节 32 位，第 1，2 字节保留，读取值为 0x0000，第 3，4 字节为温度值的 10 倍。

表 5 传感器温度数据格式

字节位	DataH16(31~16)	DataL16(15~0)
说明	保留，读取值为 0x0000	温度值的 10 倍(最高位为符号位)

举例 3：若接收到数字式 LVDT 传感器传送过来温度数据为 0x00 0x00 0x00 0xFF (0x000000FF)，则温度值解析过程如下：

- a、 $\text{DataL16} = 0x00 \times 256 + 0xFF = 255$ ；
- b、 $\text{DataL16} < 0x8000$ ，判定此数据为正数，否则为负数；
- c、温度值 $\text{Temp} = \text{DataL16} / 10 = 25.5^\circ\text{C}$ (为负数时，温度值 $\text{Temp} = (\text{DataL16} - 65536) / 10$)；

注：MODBUS 使用 ‘big-Endian’ 表示地址和数据。当发送多个字节时，首先发送最高有效位。

三、数字式 LVDT 传感器指令

数字式 LVDT 传感器提供的指令有：广播读地址，读位移或温度，读写传感器参数，位移清零/取消清零，恢复出厂设置五大类。

1、广播读地址

传感器地址是在同一个网络中识别每个传感器的“身份证”只有知道了传感器地址，才能和传感器正常进行通信，广播读地址指令就是在忘记传感器地址的情况下，去获取传感器的地址。广播读地址以地址 0x00 发送指令，因为传感器地址编号范围为 0x01~0xF7，0x00 不属于任何一个传感器的地址，所有的传感器都能接收此地址消息并上传自身的地址，因此广播读地址时，必须只有 1 路传感器接在网络中，有多路时，每一路都会上传自己的地址，就会造成通信碰撞，通信不能正常进行。

广播读地址使用功能码 0x03 读地址保持寄存器，消息帧格式如下(以下所有指令如无特殊说明，指令数据均为 16 进制格式，如 01、30 表示 0x01、0x30，转换成 10 进制数则表示 1、48，因为通信方式为 RS485，串口数据发送一般用 16 进制格式表示)。

上位机请求消息为：00 03 00 30 00 01 85 D4。

表 6 广播读地址上位机指令说明

广播地址	00
功能码	03
寄存器地址高 8 位	00
寄存器地址低 8 位	30
数据数量高 8 位	00
数据数量低 8 位	01
CRC 校验低 8 位	85
CRC 校验高 8 位	D4

传感器正常回馈消息为：00 03 02 AddrH AddrL CRCL CRCH。

表 7 广播读地址传感器回复说明

广播地址	00
功能码	03
字节数	02
传感器地址高 8 位	AddrH

传感器地址低 8 位	AddrL
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

注：AddrH 是传感器地址高 8 位，因为地址取值范围为 1~247，所以高 8 位为 00，16 位 CRC 校验结果是 CRC 校验低 8 位前所有字节的 CRC 校验结果。

2、读位移或温度

位移和温度是传感器最重要的输出数据，也是唯一只能读不能写入的参数，且位移和温度均为 32 位，占用两个寄存器，因此在读取参数时，必须以两个寄存器为单位进行读取，如一次读取一个寄存器的数据，传感器会返回 0x02 错误码(地址错误)。读位移或温度使用功能码 0x04 读输入寄存器，输入寄存器起始地址由表 1 可知为 0x0002 或 0x0004，一次仅读取一个参数时寄存器数量为 2，同时读取两个参数时寄存器数量为 4。

上位机请求消息为：Addr 04 RegAddrH RegAddrL NumH NumL CRCL CRCH。

表 8 读位移或温度上位机指令说明

传感器地址	Addr
功能码	04
寄存器地址高 8 位	RegAddrH
寄存器地址低 8 位	RegAddrL
数据数量高 8 位	NumH
数据数量低 8 位	NumL
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

注：Addr 是传感器地址，为 8 位数据，RegAddrH 和 RegAddrL 是读参数的起始地址，为 0x0002 或 0x0004，NumH 和 NumL 是读寄存器的个数，为 0x0002 或 0x0004。

传感器正常回馈消息为：Addr 04 ByteNum Data CRCL CRCH。

表 9 广播读地址传感器回复说明

传感器地址	Addr
功能码	04
数据字节数	ByteNum
位移或温度数据(N 字节)	Data

CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

注：当只读位移或温度中的一个参数时 Data 为 32 位 4 字节，当同时读两个参数时，Data 为 64 位 8 字节。

举例 1：寄存器内容为表 10 所示时，上位机发送指令格式为：Addr 04 00 02 00 04 CRCL CRCH 时(读温度和位移数据)，则下位机正常回复消息格式为：Addr 04 08 00 00 01 1D 00 7B 01 C8 CRCL CRCH。

表 10 寄存寄存器数据表

寄存器地址	描述	符号	寄存器数据
0x0002	温度高 16 位	TempH	0x0000
0x0003	温度低 16 位	TempL	0x011D
0x0004	位移高 16 位	PosH	0x007B
0x0005	位移低 16 位	PosL	0x01C8

由回复消息计算出位移和温度数据：

1、位移计算

- a、位移整数数值 $S_i = 0x00 * 256 + 0x78 = 123$;
- b、位移小数数值 $S_d = (0x01 * 256 + 0xC8) / 1000 = 0.456$;
- c、因为 $S_i < 0x8000$ ，位移值为正数，因此位移值 $S = S_i + S_d = 123.456(\text{mm})$ 。

2、温度计算

- a、温度数据 $\text{Data} = 0x01 * 256 + 0x1D = 285$;
- b、因为 $\text{Data} < 0x8000$ ，温度值为正数，因此温度值
 $\text{Temp} = \text{Data} / 10 = 28.5(^{\circ}\text{C})$ 。

3、读写传感器参数

读写传感器参数包括传感器地址，波特率，奇偶校验，补偿使能，位移和温度采样频率和输出模式，这 6 个参数即可用使用功能码 0x03 读保持寄存器，也可以使用功能码 0x10 写保持寄存器。

读传感器参数上位机请求消息为：Addr 03 RegAddrH RegAddrL NumH NumL CRCL CRCH。

表 11 读传感器参数指令说明

传感器地址	Addr
功能码	03

寄存器地址高 8 位	RegAddrH
寄存器地址低 8 位	RegAddrL
数据数量高 8 位	NumH
数据数量低 8 位	NumL
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

注：读传感器参数支持连续读取，当参数地址连续时，就能使用一条指令读取连续参数值，参数地址不连续时不能连续读取，否则传感器会返回错误码 0x02(地址错误)

读参数传感器正常回馈消息为：Addr 03 ByteNum Data CRCL CRCH。

表 12 读参数传感器回复说明

传感器地址	Addr
功能码	03
数据字节数	ByteNum
参数数据(N 字节)	Data
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

举例 1：地址为 1 的传感器读取波特率参数，则上位机发送指令为：01 03 00 31 00 01 D5 C5，若地址为 1 的传感器当前波特率为 115200bps，则传感器回复指令为：01 03 02 00 07 F9 86。

写参数上位机请求消息为：Addr 10 RegAddrH RegAddrL NumH NumL ByteNum Data CRCL CRCH。

表 13 写传感器参数指令说明

传感器地址	Addr
功能码	10
寄存器地址高 8 位	RegAddrH
寄存器地址低 8 位	RegAddrL
数据数量高 8 位	NumH
数据数量低 8 位	NumL
写入数据字节数	ByteNum

写入数据(N 字节)	Data
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

写参数传感器正常回馈消息为：Addr 10 RegAddrH RegAddrL NumH NumL CRCL CRCH。

表 14 写参数传感器回复说明

传感器地址	Addr
功能码	10
寄存器地址高 8 位	RegAddrH
寄存器地址低 8 位	RegAddrL
数据数量高 8 位	NumH
数据数量低 8 位	NumL
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

举例 2：修改地址为 1 的传感器奇偶校验方式为偶校验，则上位机发送指令为：01 10 00 32 00 01 02 00 02 23 83，若地址为 1 的传感器奇偶校验修改成功，则传感器回复指令为：01 10 00 32 00 01 A0 06。

4、位移清零/取消清零

位移清零/取消清零是在使用过程中，位移零点发生漂移或者将任意位置点设置为位移零点或者取消设置，方便用户操作和数据读取，位移清零/取消清零寄存地址为 0x50，使用功能码 0x05 写寄存器，写入数据只能为 0x0000 和 0xFF00(见表 3 所示)，写入数据为 0x0000 表示取消清零，写入数据为 0xFF00 表示位移清零，上位机只需要发送指令，传感器就会跟根据指令执行对应的操作。

位移清零/取消清零上位机请求消息为：Addr 05 00 50 FF/00 00 CRCL CRCH。

表 15 位移清零/取消清零上位机指令说明

传感器地址	Addr
功能码	05
寄存器地址高 8 位	00
寄存器地址低 8 位	50
数据数量高 8 位	FF/00

数据数量低 8 位	00
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

位移清零/取消清零传感器正常回馈消息为：Addr 05 00 50 FF/00 00 CRCL CRCH，回复消息和上位指令数据和格式均一致，即发送什么消息就回复什么消息。
 举例 1：把地址为 1 的传感器位移清零，则上位机发送指令为：01 05 00 50 FF 00 8C 2B，若地址为 1 的传感器位移清零成功，则传感器回复指令为：01 05 00 50 FF 00 8C 2B。

5、恢复出厂设置

恢复出厂设置是用户设置传感器各参数后，想恢复出厂时设定的各参数值，恢复出厂设置寄存器地址为 0x51，使用功能码 0x05 写寄存器，由表 3 可知，寄存器写入数据为 0xFF00，写入成功时，传感器会回复相同格式指令，然后便自动重启，等待 1.5s 后传感器便恢复正常工作，此时参数恢复到出厂时设定值。

恢复出厂设置上位机请求消息为：Addr 05 00 51 FF 00 CRCH CRCL。

表 16 恢复出厂调协上位机指令说明

传感器地址	Addr
功能码	05
寄存器地址高 8 位	00
寄存器地址低 8 位	51
数据数量高 8 位	FF
数据数量低 8 位	00
CRC 校验低 8 位	CRCL
CRC 校验高 8 位	CRCH

恢复出厂设置传感器正常回馈消息为：Addr 05 00 51 FF 00 CRCH CRCL，回复消息和上位指令数据和格式均一致，即发送什么消息就回复什么消息。

举例 1：把地址为 1 的传感器恢复出厂设置，则上位机发送指令为：01 05 00 51 FF 00 DD EB，若地址为 1 的传感器位移清零成功，则传感器回复指令为：01 05 00 51 FF 00 DD EB。

6、传感器异常回复

上位机发送指令不符合规范时，传感器会不做应答或回复错误码，若 CRC 校验码不正确，传感器不会做出应答，若使用了表 1 所示操作功能码以外的功能码时

传感器返回错误码 0x01，写参数或读参数寄存器地址不正确时返回错误码 0x02，写入参数不符合表 3 所示的数值时返回错误码 0x03。

表 17 异常码含义

错误码	含义
0x01	非支持的功能码
0x02	寄存器地址不正确
0x03	寄存器值超出范围

传感器异常回复长度均为 5 个字节，第 1 个字节为传感器地址或者广播地址，第 2 个字节为上位机请求功能码或上 0x80 的值，如上位机请求功能码是 0x03，则导常回复功能码为 0x03|0x80=0x83，第 3 个字节为错误码，即为表 17 其中的一个数值，最后两位为 CRC 校验位。

四、RTU 模式和 ASCII 模式

传感器通信模式默认为 RTU 模式，上文中介绍的指令格式均为 RTU 模式。RTU 模式下，采用 8bit 二进制字符，ASCII 模式下采用 7bit ASC 字符。将 RTU 模式下的一个字节的 4 位和高 4 位和低 4 位分开，变成 2 个字节，这样就是 ASCII 模式下传输的字节。比如 RTU 模式下的数据 0x1A，那么 ASCII 模式下就是 0x31 0x41 两个字节，所以 ASCII 模式下的帧长度为 RTU 模式下的 1 倍。

RTU 传输模式的数据帧采用 CRC 校验，ASCII 模式采用 LRC 校验，两种传输模式的差别如表 18 所示。

表 18 RTU 模式和 ASCII 模式区别

传输模式	ASCII (7/8 bit)	RTU (8 bit)
编码格式	ASCII 码 ('0' - '9' 'A' - 'F')	8bit 二进制 (0x00 - 0xff)
起始位	1	1
数据位	7、8	8
校验位	无、奇、偶	无、奇、偶
停止位	1、2	1、2
帧校验	LRC	CRC16

上位机读取地址为 1 的传感器波特率参数两种模式下的指令格式如表 19 所示。

表 19 RTU 模式和 ASCII 模式区别

名称	RTU 示例数据(HEX)	ASC 示例数据(HEX)
起始符	NONE	3A
传感器地址	01	30 31
功能码	03	30 33
寄存器地址高 8 位	00	30 30
寄存器地址低 8 位	31	33 31
数据数量	00 01	30 30 30 31
帧校验	D5 C5 (CRC 低位/高位)	43 31 (LRC 高位/低位)
结束符	NONE	0D 0A

五、常见问题与解决方法

1、读不到传感器的响应消息

a、可能原因

- (1)、上位机设置的传感器地址与当前通信的传感器的地址不一致；
- (2)、通信的参数设置不匹配；
- (3)、通信的数据线断开/串口异常；
- (4)、通信代码解析错误。

b、解决方法

(1)、一般情况下，传感器出厂时的默认从机地址都为传感器外壳上的编号，当您修改过或工程师遗忘了此传感器的从机地址，主机设置又不对时，可以发送读地址指令获取传感器的地址：

上位机发送命令：00 03 00 00 00 01 85 DB

传感器响应：00 03 02 00 01 44 44

响应的代码译为 00 地址，03 功能码，02 字节数，00 01 即为传感器地址。

注意，使用此指令时，只能有一路传感器接入，其他的必须先断开。

(2)、通信参数不匹配，请查阅传感器手册确定此传感器的通信参数如波特率 9600bps 奇偶校验位为无校验等，当您修改过或工程师遗忘了此传感器的通信参数时，只能把波特率和奇偶校验做一个排列组合，主机逐个配置的发送指令测试，接收到响应值时即为正确的参数。

(3)、检查您的线路是否断开或者串口 COM 是否被其他软件占用或者计算机的 COM 端口异常，重启计算机即可。

(4)、上位机指令 CRC 校验码或 LRC 校验码不正确，此时传感器接收到了指令但不会做出回复，检查指令 CRC 或 LRC 校验是否正确，校验方法可直接在网上查询下载一个 CRC 和 LRC 计算器。

2、位移或温度数据不正常

读到传感器位移或温度响应值，发现数值误差大、偶尔数值错误、数值明显不对。

a、可能原因

(1)、计算传感器返回的数据位时不正确，比如代码中对十六进制数转换成十进制数不正确，显示打印位移值时的字符缓存等等的操作不正确；

(2)、上位机可能没有做返回数据的 CRC 校验或 LRC 校验，通信过程中由于强磁干扰或线路原因的数据冲突/突变，导致数据在传输过程中的丢失或变异，您需要在代码中添加相应的 CRC 或 LRC 检查函数，字符帧的数量检查等操作，确定收到的返回值是正确的，再进行计算，否则主机应该重新发起查询指令。

当您的系统相当复杂时，建议先把传感器单独取出，进行代码调试正确后再加入到整体系统中，以便减少代码与测试出错的发生。

3、当批量查询多个传感器时，发现偶尔有些传感器无响应

在轮询代码中，主机进行查询指令收到响应值后，切换到下一个传感器时，应当在代码中加入一点延时，如主机在查询 1 号传感器成功后，等待 100ms，再发送下一号传感器的查询指令，注意串口缓冲区的字符清空和串口等待超时设置。