

【八路控制器】协议通讯

修订历史

版本	修订内容	修订日期
1.0	初稿	2020-11-23

一，RS232 接口通讯协议

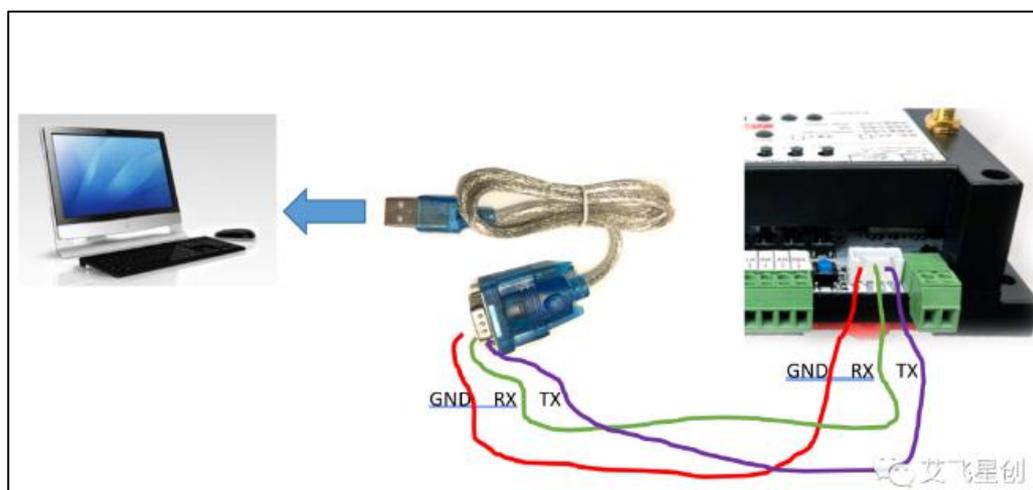
1. 需用 RS232 转 TTL 电平后才能其他 MCU 通讯。
2. 用电脑调试需要用 USB 转 RS232 模块转换才可以通讯。
3. 控制器先单独供电，然后只接 RX、TX、GND 三个接口即可。
4. 波特率：9600，8 位数据位，1 位停止位，无校验。不可以修改

1.1 准备工具（以 PC 为例）



准备一根 USB 转 RS232 的转换线。

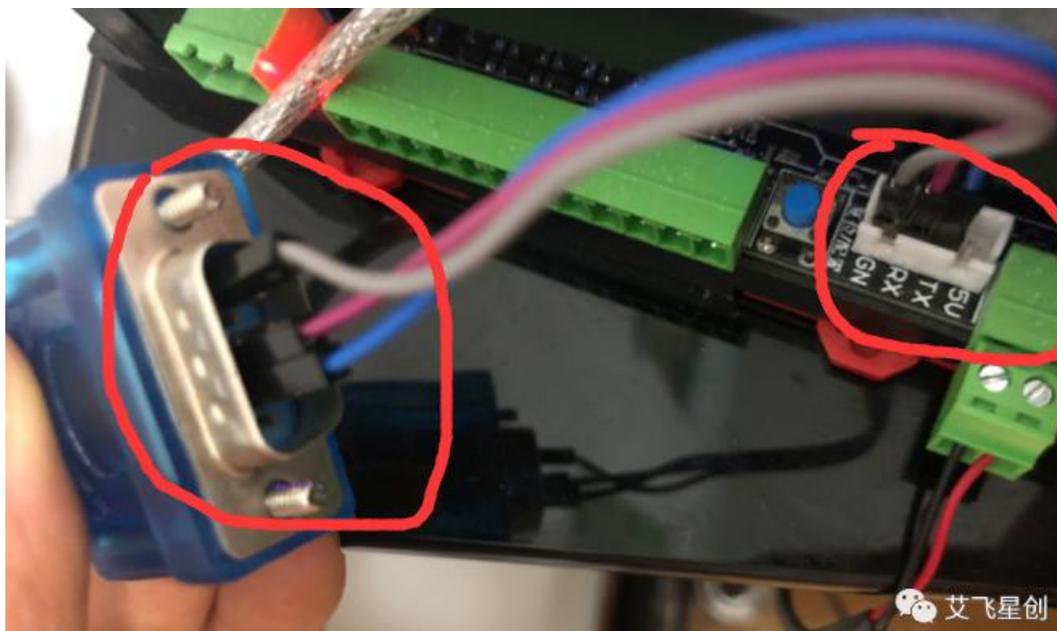
1.2 连接主机



以 PC 为例。以上连接主机的示意图

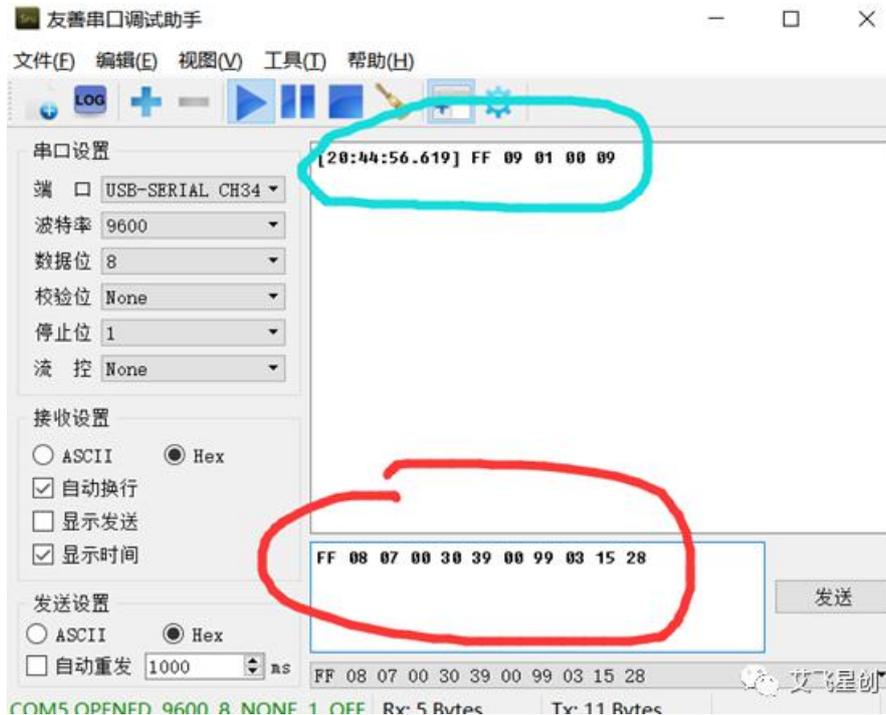


DB9 接口，我们只用到 2-RXD、3-TXD、5-GND。



注，主机需单独先上电运行！
然后如上图连接到主机，另外一边 USB 插到电脑上面。

1.3 准备串口工具软件



工具就不提供了，自己网上找一下。串口通讯工具。仅用于调试使用！

1.4 协议内容

- 1, 为方便理解，分主机与从机，那么电脑为主机（以下统称主机），八路控制器为从机（以下统称从机）；
- 2, 只能有一个主机与一个从机！

1.4.1 从机开关状态上报（命令 03）

帧头	命令	数据长度（固定）	数据内容	校验和（前面所有数据之和）
0XFF	0X03	0X01	0X00	0X03

数据内容：开关数据 8 位，每一位对应一个开关，“1”启动“0”关闭

注：

- 1, 从机上电会主动上报一次数据！然后每次切换开关，从机都会主动上报一次数据！
- 2, 不需要主动去访问开关状态，从机是自动上报，须知须知！

1.4.2 主机切换从机开关-全部（命令 02）

主机发送控制设备开关-全部

帧头	命令	数据长度（固定）	数据内容	校验和（前面所有数据之和）
0XFF	0X02	0X01	0X00	0X02

数据内容：开关数据 8 位，每一位对应一个开关，“1”启动“0”关闭

从机回复开关状态

帧头	命令	数据长度（固定）	数据内容	校验和（前面所有数据之和）
0XFF	0X03	0X01	0X00	0X03

数据内容：开关数据 8 位，每一位对应一个开关，“1”启动“0”关闭

注：

- 1, 命令 02 是一次性发送所有开关的切换状态，可以实现全开全关！
- 2, 如果发送开关指令是重复的，那么从机不会回复开关状态。

1.4.3 主机切换从机开关-单独（命令 0A）

主机发送控制设备开关-单独

帧头	命令	数据长度（固定）	数据地址	数据内容	校验和（前面所有数据之和）
0XFF	0X0A	0X02	0X00	0X00	0X0B

数据地址：0X00-开关 1
0X01-开关 2
0X02-开关 3
0X03-开关 4
0X04-开关 5
0X05-开关 6
0X06-开关 7
0X07-开关 8
0X08-蜂鸣器

数据内容：0X00-关闭
0XFF-打开

从机回复开关状态

帧头	命令	数据长度（固定）	数据内容	校验和（前面所有数据之和）
0XFF	0X03	0X01	0X00	0X03

数据内容：开关数据 8 位，每一位对应一个开关，“1”启动“0”关闭

注：

- 1, 命令 03 是单独发送某一个开关切换状态的切换，如多个开关需多次发送。
- 2, 如果发送开关指令是重复的，那么从机不会回复开关状态。

1.4.4 主机读取从机地址数据-单独（命令 04）

地址分布：

地址	字节	注释
00	2	8 个开关状态
01	2	A1 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
02	2	A2 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
03	2	A3 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
04	2	A4 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
05	2	A5 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
06	2	D1 口第一个原始数字数据
07	2	D1 口第二个原始数字数据
08	2	D1 口第三个原始数字数据
09	2	D1 口第四个原始数字数据
0A	2	D1 口第五个原始数字数据
0B	2	D2 口第一个原始数字数据
0C	2	D2 口第二个原始数字数据
0D	2	D2 口第三个原始数字数据
0E	2	D2 口第四个原始数字数据
0F	2	D2 口第五个原始数字数据
10	2	D3 口第一个原始数字数据
11	2	D3 口第二个原始数字数据
12	2	D3 口第三个原始数字数据
13	2	D3 口第四个原始数字数据
14	2	D3 口第五个原始数字数据
15	6	系统时钟 字节 1：年 字节 2：月 字节 3：日 字节 4：时 字节 5：分 字节 6：秒
5A	2	<ul style="list-style-type: none"> ●bit0: 是否开启了SoftAP模式，0为关，1为开 ●bit1: 是否开启了Station模式，0为关，1为开 ●bit2: 保留 ●bit3: 是否开启了绑定模式，0为关，1为开 ●bit4: WiFi模组是否已成功连接上了无线路由器，0为未连接，1为已连接 ●bit5: WiFi模组是否已成功连接上了M2M服务器，0为未连接，1为已连接 ●bit6~bit7: 保留 ●bit8~bit10: 三个位合起来表示一个整型值，值范围 为0~7，表示WiFi模组当前连接无线路由器的信号强度(RSSI)，0为最低，7为最高 ●bit11: 是否有App在线，0为否，1为是

	bit12~bit15: 保留
--	-----------------

主机查询从机地址读值（一次只能访问一个地址）

帧头	命令	数据长度（固定）	数据地址	校验和（前面所有数据之和）
0XFF	0X04	0X01	0X00	0X04

注：

多次连续采集请务必加延时。

模拟口最快刷新频率 150ms 一次，采集速度快于 150ms 没意义。

数字口最快刷新频率 2 秒一次，采集数据快于 2 秒没意义。

从机返回地址数据

帧头	命令	数据长度	数据地址	数据字节 1	数据字节 2	校验和（前面所有数据之和）
0XFF	0X05	0X03	0X00	0X00	0X00	0X07

1.4.5 主机读取从机地址数据-全部（命令 06）

主机访问所有读值

帧头	命令	数据长度（固定）	数据（固定）	校验和（前面所有数据之和）
0XFF	0X06	0X01	0X55	0X5B

从机返回所有传感器读值

帧头	命令	数据长度	数据字节 1	数据（50 字节）	校验和（前面所有数据之和）
0XFF	0X07	0X32	0X00	0X00	0XXX

注：

一次性返回全部，并不与单独读值地址一致！

字节解析：（按照返回的字节排列，十进制）

字节	注释
3	8 个开关状态
4	时间年
5	时间月
6	时间日
7	时间时
8	时间分
9	时间秒
10-11	A1 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
12-13	A2 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
14-15	A3 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
16-17	A4 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
18-19	A5 口原始模拟数据，0-3300，对应 0-3.3V，输出值为实际采集电压。
20-21	D1 口第一个原始数字数据
22-23	D1 口第二个原始数字数据

24-25	D1 口第三个原始数字数据
26-27	D1 口第四个原始数字数据
28-29	D1 口第五个原始数字数据
30-31	D2 口第一个原始数字数据
32-33	D2 口第二个原始数字数据
34-35	D2 口第三个原始数字数据
36-37	D2 口第四个原始数字数据
38-39	D2 口第五个原始数字数据
40-41	D3 口第一个原始数字数据
42-43	D3 口第二个原始数字数据
44-45	D3 口第三个原始数字数据
46-47	D3 口第四个原始数字数据
48-49	D3 口第五个原始数字数据
50-51	<ul style="list-style-type: none"> ● bit0: 是否开启了SoftAP模式, 0为关, 1为开 ● bit1: 是否开启了Station模式, 0为关, 1为开 ● bit2: 保留 ● bit3: 是否开启了绑定模式, 0为关, 1为开 ● bit4: WiFi模组是否已成功连接上了无线路由器, 0为未连接, 1为已连接 ● bit5: WiFi模组是否已成功连接上了M2M服务器, 0为未连接, 1为已连接 ● bit6~bit7: 保留 ● bit8~bit10: 三个位合起来表示一个整型值, 值范围为0~7, 表示WiFi模组当前连接无线路由器的信号强度(RSSI), 0为最低, 7为最高 ● bit11: 是否有App在线, 0为否, 1为是 bit12~bit15: 保留
52	预留

1.4.6 主机上报数据到传感器显示（命令 08）

帧头	命令	输入传感器的数据长度	传感器类别	数据 1	数据 2	校验和（前面所有数据之和）
0XFF	0X08	0X03	0X00	0X00	0X00	0XA

注：

1. 传感器类别仅支持 0X00(D1)、0X01(D2)、0X02(D3)。
2. 输入传感器的长度最大 10+1 字节，必须奇数。

从机返回是否成功

帧头	命令	数据长度	数据	校验和（前面所有数据之和）
0XFF	0X09	0X01	0X00(成功) 0X01(失败)	0X09

教程可以参考 <https://mp.weixin.qq.com/s/B40FjU4lrn7hdu7M0Hd6cA>

二，RS485 接口通讯协议

1. 物理层协议：

物理层采用 UART 转 RS485 通讯接口。

波特率：9600，8 位数据位，1 位停止位，无校验。不可以修改

2. 应用层协议：

应用层协议采用 Modbus-RTU 协议进行通讯，目前只支持 0x01(读继电器状态)、0x04(读取传感器寄存器)、0x05(控制单个继电器)、0x06(写从机参数)、0x10(写自定义传感器寄存器)、0x0f(控制多个继电器)等功能码。

3. 从机地址：

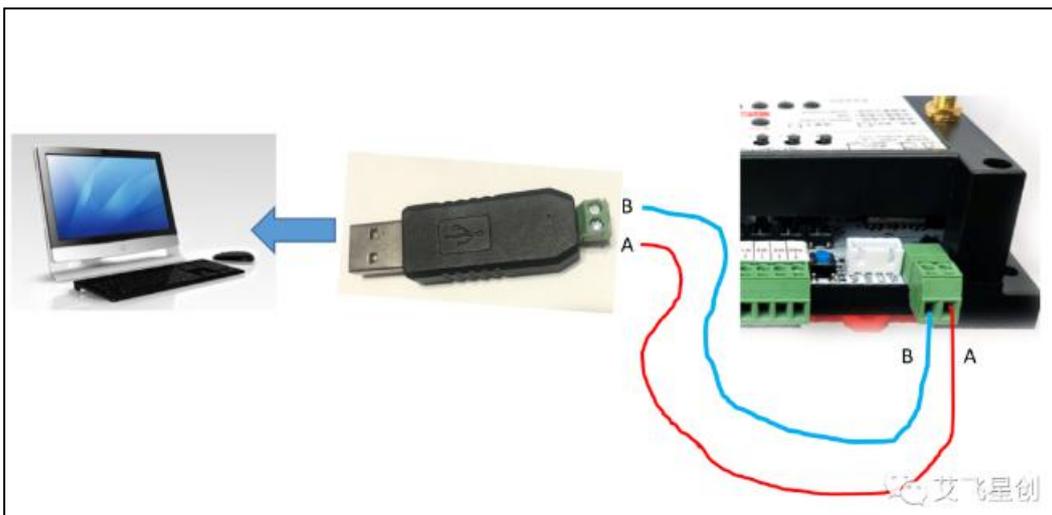
出厂地址码默认 01，00 为万能地址，可以通过 00 来得到原设备地址。

2.1 准备工具（以 PC 为例）



准备一根 USB 转 RS485 的转换线。

2.2 连接主机



以 PC 为例。

注，主机需单独先上电运行！

然后如上图连接到主机，另外一边 USB 插到电脑上面。

2.3 协议内容

- 1, 为方便理解, 分主机与从机, 那么电脑为主机 (以下统称主机), 八路控制器为从机 (以下统称从机);
- 2, 一般主机只有 1 个, 从机可以有多个, 多个从机下地址不可以重复。

2.3.1 功能码 01

主机读取 1 路或多路继电器输出状态

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	CRC 校验低字节	CRC 校验高字节
0X01	0X01	0X00	0X00	0X00	0X08	0X3D	0XCC

注: 地址只支持 0x0000-0x0007 对应 8 个继电器, 数量+地址不可以大于 8。

从机回复

从机地址	功能码	字节数	继电器状态	CRC 校验低字节	CRC 校验高字节
0X01	0X01	0X01	0X00	0X88	0X51

注: 继电器状态 8 位对应 8 个继电器, 0 没启动 1 启动。

2.3.2 功能码 04

主机读值传感器寄存器

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	CRC 校验低字节	CRC 校验高字节
0X01	0X04	0X00	0X00	0X00	0X01	0X31	0XCA

寄存器地址:

地址	类型	注释
0000	U8	8 个开关状态
0001	U8*6	系统时钟 字节 1: 年 字节 2: 月 字节 3: 日 字节 4: 时 字节 5: 分 字节 6: 秒
0002	U16	A1 口原始模拟数据, 0-3300, 对应 0-3.3V, 输出值为实际采集电压。
0003	U16	A2 口原始模拟数据, 0-3300, 对应 0-3.3V, 输出值为实际采集电压。
0004	U16	A3 口原始模拟数据, 0-3300, 对应 0-3.3V, 输出值为实际采集电压。
0005	U16	A4 口原始模拟数据, 0-3300, 对应 0-3.3V, 输出值为实际采集电压。
0006	U16	A5 口原始模拟数据, 0-3300, 对应 0-3.3V, 输出值为实际采集电压。
0007	U16	D1 口第一个原始数字数据

0008	U16	D1 口第二个原始数字数据
0009	U16	D1 口第三个原始数字数据
000A	U16	D1 口第四个原始数字数据
000B	U16	D1 口第五个原始数字数据
000C	U16	D2 口第一个原始数字数据
000D	U16	D2 口第二个原始数字数据
000E	U16	D2 口第三个原始数字数据
000F	U16	D2 口第四个原始数字数据
0010	U16	D2 口第五个原始数字数据
0011	U16	D3 口第一个原始数字数据
0012	U16	D3 口第二个原始数字数据
0013	U16	D3 口第三个原始数字数据
0014	U16	D3 口第四个原始数字数据
0015	U16	D3 口第五个原始数字数据
0029	U16	<ul style="list-style-type: none"> ●bit0: 是否开启了SoftAP模式, 0为关, 1为开 ●bit1: 是否开启了Station模式, 0为关, 1为开 ●bit2: 保留 ●bit3: 是否开启了绑定模式, 0为关, 1为开 ●bit4: WiFi模组是否已成功连接上了无线路由器, 0为未连接, 1为已连接 ●bit5: WiFi模组是否已成功连接上了M2M服务器, 0为未连接, 1为已连接 ●bit6~bit7: 保留 ●bit8~bit10: 三个位合起来表示一个整型值, 值范围为0~7, 表示WiFi模组当前连接无线路由器的信号强度(RSSI), 0为最低, 7为最高 ●bit11: 是否有App在线, 0为否, 1为是 bit12~bit15: 保留

从机回复

从机地址	功能码	字节数	数据 1	CRC 校验低字节	CRC 校验高字节
0X01	0X04	0X01	0X00	0X89	0X41

2.3.3 功能码 05

主机控制单个继电器的动作

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	CRC 校验低字节	CRC 校验高字节
0X01	0X05	0X00	0X00	0XFF	0X00	0X8C	0X3A

注: 寄存器地址只支持 0x0000-0x0007 对应 8 个继电器, 数据 0xff00 打开, 0x0000 关闭。

从机回复

与主机一致

2.3.4 功能码 06

主机写从机寄存器

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	CRC 校验低字节	CRC 校验高字节
0X01	0X06	0X00	0X00	0X00	0X01	0X48	0X0A

寄存器地址:

地址	注释	范围
0000	从机地址	0x0001-0x00ff

从机回复
与主机一致

2.3.5 功能码 0F

主机一次写入多个继电器动作

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	字节数	数据
0X01	0X0F	0X00	0X00	0X00	0X08	0X01	0X0F

CRC 校验低字节	CRC 校验高字节
0XBE	0X91

注：寄存器地址只支持 0x0000-0x0007 对应 8 个继电器，数量+地址不可以大于 8。

数据 BIT0 开关 1-BIT7 开关 8， 1 打开，0 关闭。

从机回复

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	字节数	CRC 校验低字节	CRC 校验高字节
0X01	0X0F	0X00	0X00	0X00	0X08	0X01	0XCD	0X3F

2.3.6 功能码 10

主机写自定义传感器值到主机显示

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	字节数
0X01	0X10	0X00	0X00	0X00	0X02	0X04
数据 1 高字节	数据 1 低字节	数据 2 高字节	数据 2 低字节	CRC 校验低字节	CRC 校验高字节	
0X00	0X00	0X00	0X00	0XF3	0XAF	

寄存器地址：

地址	类型	注释
0000	U16	D1 口第一个原始数字数据
0001	U16	D1 口第二个原始数字数据
0002	U16	D1 口第三个原始数字数据
0003	U16	D1 口第四个原始数字数据
0004	U16	D1 口第五个原始数字数据
0005	U16	D2 口第一个原始数字数据
0006	U16	D2 口第二个原始数字数据
0007	U16	D2 口第三个原始数字数据
0008	U16	D2 口第四个原始数字数据
0009	U16	D2 口第五个原始数字数据
000A	U16	D3 口第一个原始数字数据
000B	U16	D3 口第二个原始数字数据
000C	U16	D3 口第三个原始数字数据
000D	U16	D3 口第四个原始数字数据
000E	U16	D3 口第五个原始数字数据

从机回复

从机地址	功能码	寄存器地址高字节	寄存器地址低字节	寄存器数量高字节	寄存器数量低字节	CRC 校验低字节	CRC 校验高字节
0X01	0X10	0X00	0X00	0X02	0X00	0X41	0XC8

教程可以参考 <https://mp.weixin.qq.com/s/B40FjU4lrn7hdu7M0Hd6cA>

注意事项【上报数据的间隔时间不要小于 1 秒，尽量 2 秒上报一次数据！】

2.3.7 错误码

错误码解读：

从机地址	功能码	错误码	CRC 校验低字节	CRC 校验高字节
0X01	0X41	0X01	0XD1	0X90

0x01：功能码不存在

0x02：地址不存在

0x03：长度超过最大范围

0x04：读值范围超出