

称重控制系统 modbus 通信协议规范

协议格式如下

起始位	设备地址	功能代码	数据	CRC 校验	结束位
T	1 byte	1 byte	N byte	2 byte	T

说明

- 1: 起始位 至少 3.5 个字符时间
- 2: 设备地址 设备地址, 0~247 个地址, 其中 0 号地址作为广播地址
- 3: 功能码 使用 0x03,0x06,0x10,其中 0x03 是读操作, 0x06 是写操作,0x10 是批量写
- 4: 数据 操作地址 + 数据
- 5: CRC 校验 CRC16 校验
- 6: 结束位 至少 3.5 个字符时间

功能码 03 举例:
主机发送: 01 03 00 00 00 01 84 0A
从机回复: 01 03 02 01 02 35 15

/*主机发送解析*/
01-地址
03-功能码,代表查询功能,其他功能后面再说
00 00-代表查询的起始寄存器地址.说明从 0x0000 开始查询.
00 01-代表查询了一个寄存器.结合前面的 00 00,意思就是查询从 0 开始的 1 个寄存器值;
84 0A-循环冗余校验,是 modbus 的校验公式,从首个字节开始到 84 前面为止;
/*从机回复解析*/
01-地址
03-功能码
02-代表后面数据的字节数,因为上面说到,一个寄存器有 2 个字节,所以后面的字节数肯定是 2*查询的寄存器个数;
01 02-寄存器的值是 0x0102,结合发送的数据看出,01 这个寄存器的值为 0x0102
35 15-循环冗余校验

功能码 06 举例:
主机发送: 01 06 00 00 12 34 84 BD
从机回复: 01 06 00 00 12 34 84 BD
/* 主机发送解析 */
01-主机要查的地址
06-功能码,代表修改单个寄存器功能
00 00-代表修改的起始寄存器地址.说明从 0x0000 开始.
12 34-代表修改的值为 0x12 34.结合前面的 00 00,意思就是修改 0 号寄存器值为 0x1234;

84 BD -循环冗余校验,是 Modbus 的校验公式,从首个字节开始到 84 前面为止;

/*从机回复解析*/

01-从机返回的地址,说明这就是主机查的从机

06-功能码,代表修改单个寄存器功能;

00 00-代表修改的起始寄存器地址.说明是 0x0000.

12 34-代表修改的值为 0x1234.结合前面的 00 00,意思就是修改 0 号寄存器值为 0x1234;

84 BD -循环冗余校验,是 Modbus 的校验公式,从首个字节开始到 84 前面为止;

功能码 16 举例:

主机发送: 01 10 00 00 00 02 04 00 11 22 33 FB 1F

从机回复: 01 10 00 00 00 02 41 C8

/*主机发送解析*/

01-主机要查的地址

10-功能码,0x10=16D 代表修改多个寄存器功能;

00 00-代表修改的起始寄存器地址.说明从 0x0000 开始.

00 02-代表修改的寄存器数量,这里开始于 0x06 的修改不同;

04 -表示修改的总字节数,由于修改了 2 个寄存器,所以数据要有 4 个字节;

00 11-表示修改的值,结合上面,就是从第 0000 寄存器开始修改第一个寄存器值为 0x0011,
就是把 0000 寄存器改为 0x0011;

22 33-表示修改的值,结合上面,就是从第 0000 寄存器开始修改第二个寄存器值为 0x22 33,
就是把 0001 寄存器改为 0x2233;

FB 1F -循环冗余校验,是 Modbus 的校验公式,从首个字节开始到 22 前面为止;

/*从机回复解析*/

01-从机返回的地址,说明这就是主机查的从机

10-功能码

00 00-代表修改的起始寄存器地址.说明是 0x0000.

00 02-代表修改的寄存器数量,只需要回复这么多久足够了,从机告诉主机,你修改了哪几个寄存器就足够了;

41 C8-循环冗余校验;

1、零点校准

一通道校准:

发送: **addr** 06 00 01 00 00 CRC_L CRC_H

回应: **addr** 06 00 01 00 00 CRC_L CRC_H

二通道校准:

发送: **addr** 06 00 02 00 00 CRC_L CRC_H

回应: **addr** 06 00 02 00 00 CRC_L CRC_H

双通道同时校准:

发送: **addr** 06 00 03 00 00 CRC_L CRC_H

回应: **addr** 06 00 03 00 00 CRC_L CRC_H

2、重量标定

一通道重量标定:

发送: **addr** 06 00 05 weight_H weight_L(砝码重量) CRC_L CRC_H

回应: **addr** 06 00 05 00 00 CRC_L CRC_H

二通道重量标定:

发送: **addr** 06 00 06 weight_H weight_L(砝码重量) CRC_L CRC_H

回应: **addr** 06 00 06 00 00 CRC_L CRC_H

3、重量读取

一通道重量读取

发送: **addr** 03 00 09 00 02 CRC_L CRC_H

回应: **addr** 03 04 (**weight1***1000>>24) (**weight1***1000>>16) (**weight1***1000>>8)
(**weight1***1000) CRC_L CRC_H

二通道重量读取

发送: **addr** 03 00 0A 00 02 CRC_L CRC_H

回应: **addr** 03 04 (**weight2***1000>>24) (**weight2***1000>>16) (**weight2***1000>>8)
(**weight2***1000) CRC_L CRC_H

双通道同时重量读取

发送: **addr** 03 00 0B 00 04 CRC_L CRC_H

回应: **addr** 03 08 (**weight1***1000>>24) (**weight1***1000>>16) (**weight1***1000>>8)
(**weight1***1000) (**weight2***1000>>24) (**weight2***1000>>16) (**weight2***1000>>8)
(**weight2***1000) CRC_L CRC_H

4、AD 读取

一通道 AD 读取:

发送: **addr** 03 00 11 00 02 CRC_L CRC_H

回应: **addr** 03 04 (**AD1**>>24) (**AD1**>>16) (**AD1**>>8) (**AD1**) CRC_L CRC_H

二通道 AD 读取:

发送: **addr** 03 00 13 00 02 CRC_L CRC_H

回应: **addr** 03 04 (**AD2**>>24) (**AD2**>>16) (**AD2**>>8) (**AD2**) CRC_L CRC_H

双通道同时 AD 读取:

发送: **addr** 03 00 14 00 04 CRC_L CRC_H

回应: **addr** 03 08 (**AD1**>>24) (**AD1**>>16) (**AD1**>>8) (**AD1**) (**AD2**>>24) (**AD2**>>16)
(**AD2**>>8) (**AD2**) CRC_L CRC_H

5、通信地址

地址修改:

发送: **addr** 06 00 12 00 address CRC_L CRC_H

回应: **addr** 06 00 12 00 address CRC_L CRC_H

地址查询：

发送：addr 03 00 80 00 01 CRC_L CRC_H

回应：addr 03 02 00 address CRC_L CRC_H

波特率修改：

发送：addr 06 00 81 baudrate_H baudrate_L CRC_L CRC_H

回应：addr 06 00 81 baudrate_H baudrate_L CRC_L CRC_H

波特率查询：

发送：addr 03 00 82 00 01 CRC_L CRC_H

回应：addr 03 02 baudrate_H baudrate_L CRC_L CRC_H

6、去皮：

发送：addr 06 00 40 00 00 CRC_L CRC_H

回应：addr 06 00 40 00 00 CRC_L CRC_H

7、版本号查询

发送：addr 03 00 90 00 02 CRC_L CRC_H

回应：addr 03 04 (version>>24) (version>>16) (version>>8) (version) CRC_L CRC_H