

4. AS200通信协议

4.1 通信协议

Modbus-RTU使用RS-485接口作为硬件载体，详细的通信参数请查阅表3。

表 3. 串口通信协议参数

通讯参数	描述
协议格式	Modbus-RTU
协议CRC多项式	0xA001
通讯速率	2400/4800/9600/19200bps
起始位	1位
数据位	8位
停止位	1位
奇偶校验	无

通信协议是标准的Modbus RTU。主机可以是计算机、RS485接收器和MCU控制器等。AS200作为从机，默认的地址为0x01，支持地址修改，可接入多个从机的总线上。

4.2 功能码

Modbus消息帧的设备地址域包含8bit（RTU模式下），功能代码长度也为8bit。消息帧里的功能码可以实现主单元对从单元的数字量和模拟量的读写操作，AS200中最常用的两个功能码是0x03及0x06，0x03及0x06功能码对应的数据类型及作用见表4。

表 4. 03 和 06 功能码

功能码	名称	数据类型	作用
03	读保持寄存器	整型	读取一个的保持寄存器的值
06	预置单寄存器	整型	把具体的十六进制数值写入一个保持寄存器

5. 寄存器

AS200内部有多个寄存器，可以通过读取寄存器的值来获取瞬时流量、累计流量、设置流量等信息，各参数对应的寄存器值请查阅表5。

表 5. Modbus RTU 通信寄存器定义

寄存器地址	读写属性	功能码	倍数	寄存器信息	备注
0	只读	03	1	瞬时流量 [31:16]	瞬时流量为32bit float 类型
1	只读	03	1	瞬时流量 [15:0]	
2	读写	03/06	1	设置流量 [31:16]	设置流量为32bit float 类型
3	读写	03/06	1	设置流量 [15:0]	
4	读写	03/06	1	累积流量 [63:48]	累计流量为 64bit double类型，写1 清零累积流量
5	读写	03/06	1	累积流量 [47:32]	
6	读写	03/06	1	累积流量 [31:16]	
7	读写	03/06	1	累积流量 [15:0]	
8	读写	03/06	1	从机地址	Modbus从机地址，默认 地址：1
9	读写	03/06	1	波特率	1200, 2400, 4800, 9600, 1 9200, 38400，默认波特 率：19200
11	读写	03/06	1	模拟/数字流量 控制模式	1：数字控制；（出厂默 认） 2：模拟电压控制； 默认控制模式：数字控 制模式

注意：可写入寄存器写入数据后，会立即生效。

主机读取AS200的瞬时流量的数值（设备默认从机地址为0x01）时，主机对AS200系列发送命令为01 03 00 00 00 02 C4 0B，数据格式如表6所示。

表 6. 主机发送读寄存器命令的格式表

主机发送信息	字节数	发送信息举例（Hex）	信息含义说明
从机地址	1	01	通信从机地址
功能码	1	03	读多路寄存器
寄存器起始地址	2	0000	寄存器0x0000存放着瞬时流量数值
读取寄存器个数	2	0002	读取2个寄存器
CRC码	2	C40B	CRC码用于校验

当主机接收AS200系列返回的数据为01 03 04 46 1C 40 CD DE E8，数据格式如表7所示。

表 7. 主机接收 AS200 系列寄存器数据的格式表

从机发送信息	字节数	接收信息举例（Hex）	信息含义说明	按上述举例主机发送接收的数据含义
从机地址	1	01	通信从机地址	-
功能码	1	03	读多路寄存器	-
接收数据字节数	1	04	接收数据字节数=读取寄存器数×2	-
寄存器数据	4	461C40CD	读取的2个寄存器	转换成10进制float型数据，最终得10000.2mL/min
CRC码	2	DEE8	CRC码用于校验	-

主机对AS200设置流量为10000mL/min时，主机对AS200系列发送命令为01 10 00 02 00 02 04 46 1C 40 00 97 38，数据格式如表8所示。

表 8. 主机发送写寄存器命令的格式表

主机发送信息	字节数	发送信息举例（Hex）	信息含义说明
从机地址	1	01	通信从机地址
功能码	1	10	写多个寄存器
寄存器起始地址	2	0002	从第2个寄存器开始写入
写入寄存器的数量	2	0002	写入2个寄存器
写入寄存器的字节数	1	04	写入数据字节数=写入寄存器数×2
写入寄存器的数据	4	461C4000	0x461C4000转换成10进制float型数

			据，最终得10000.0mL/min
CRC码	2	9738	CRC码用于校验

异常应答返回，数据格式如表9所示。

表 9. 主机接收异常数据格式表

从机发送信息	字节数	接收信息举例 (Hex)	信息含义说明	按上述举例主机发送接收的数据含义
从机地址	1	01	通信从机地址	-
功能码	1	80		-
异常码	1	01（非法功能） 02（非法数据地址） 03（非法数据）		-

异常码举例：

本机仅支持03和06功能码，如功能码错误的设为07，则返回：01 87 01 82 30。

Modbus寄存器数据和浮点型数据互相转换代码参考：

1. float型：

```
//float型数据为32位数据，存放于2个16位数组元素里；
//需要注意具体使用的MCU的大小端类型，大端和小端MCU存放数据高低位相反；
//本例以小端MCU STM32举例：

#include <stdio.h>
union
{
    float ffloat;//单精度浮点型数据
    unsigned short int farray[2];//无符号16位整型数组
}utemp;

int main(void)
{
    #if 0
        //浮点型数据转整型数据
        utemp.ffloat = 10000.200195;
        printf("farray[0] = 0x%04X\n",utemp.farray[0]);
        printf("farray[1] = 0x%04X\n",utemp.farray[1]);

        //打印结果为：
        //farray[0] = 0x40CD
        //farray[1] = 0x461C
    #else
        //整型数组转浮点型数据
        utemp.farray[0] = 0x40CD;
        utemp.farray[1] = 0x461C;
        printf("float = %f\n",utemp.ffloat);
        //打印结果为： float = 10000.200195
    #endif
    return 0;
}
```

2. double型:

//double型数据为64位数据，存放于4个16位数组元素里；
//需要注意具体使用的MCU的大小端类型，大端和小端MCU存放数据高低位相反；
//本例以小端MCU STM32举例：

```
#include <stdio.h>
union
{
    double ffloat;//双精度浮点型数据
    unsigned short int farray[4];//无符号16位整型数组
}utemp;

int main(void)
{
    #if 0
        //浮点型数据转整型数据
        utemp.ffloat = 10000.200195;
        printf("farray[0] = 0x%04X\n",utemp.farray[0]);
        printf("farray[1] = 0x%04X\n",utemp.farray[1]);
        printf("farray[2] = 0x%04X\n",utemp.farray[2]);
        printf("farray[3] = 0x%04X\n",utemp.farray[3]);

        //打印结果为:
        //farray[0] = 0x60E9
        //farray[1] = 0x9FFD
        //farray[2] = 0x8819
        //farray[3] = 0x40C3
    #else
        //整型数组转浮点型数据
        utemp.farray[0] = 0x60E9;
        utemp.farray[1] = 0x9FFD;
        utemp.farray[2] = 0x8819;
        utemp.farray[3] = 0x40C3;
        printf("double = %f\n",utemp.ffloat);

        //打印结果为: double = 10000.200195
    #endif
    return 0;
}
```