通讯协议

1. 引言

本通讯协议详细描述了本机串行口通讯的读写命令格式及内部信息数据的定义，以便第三方开发使用。

MODBUS 通讯规约允许本装置与施耐德、西门子、Modicon 等多个国际知名品牌的可编程顺序装置(PLC)、RTU、SCADA 系统、DCS 或第三方具有 MODBUS 兼容的监控系统之间进行信息和数据的有效传递。只要增加一套基于 PC(或工控机)的中央通讯主控显示软件(如：组态王，Intouch、FIX、synall 等)就可建立一套监控系统。

2. ModBus 基本规则：

- 所有 RS232 通讯回路都应遵照主、从方式。依照这种方式，数据可以在一个主站(如：PC)和 32 个子站之间传递。
- 主站将初始化的装置在 RS232 通讯回路上传递的所有信息。
- 任何一次通讯都不能从子站开始。
- 在 RS232 回路上的所有通讯都以“信息帧”方式传递。
- 如果主站或子站接收到含有未知命令的信息帧，则不予响应。

3. 数据帧格式：

通讯传输为异步方式，并以字节(数据帧)为单位。在主站和子站之间传递的每一个数据帧都是以 11 位的串行数据流。

数据帧格式：

起始位	1 位
数据位	8 位
奇偶校验位	无
停止位	2 位

4. 通信规约：

当通信命令发送至仪器时，符合相应的地址码的设备接收通信命令，并除去地址码，读取信息，如果没有出错，则执行相应的任务，然后把执行结果返送给发送者。返送的信息中包括地址码、执行动作的功能码、执行动作后的数据以及错误校验码(CRC)。如果出错就不发送任何信息。

●信息帧格式：

初始结构	地址码	功能码	数据区	错误校验	结束结构
延时(相当于 4 个字节的时间)	1 字节 8 位	1 字节 8 位	N 字节 N*8 位	2 字节 16 位	延时(相当于 4 个字节的时间)

●地址码(ADDRESS)：

地址码为每次通信传送的信息帧中的第一个数据帧(8 位)，从 0 到 255。单个设备的地址范围是 1-247，这个字节表明由用户设定的地址码的子机将接收由主机发送来的信息，并且每个子机都有唯一的地址码，并且响应回送均以各自的地址码开始。主机发送来的地址码表明将发送到的子机地址，而子机发送的地址码表明回送的子机地址。

●功能码 (FUNCTION CODE)

功能码是每次通信传送的第二个数据。ModBus 通讯规约定义功能码为 1-255(01H-0FFH)。本机利用其中的一部分功能码。作为主机请求发送，通过功能码告诉子机执行什么动作。作为子机响应，子机发送的功能码与主机发送来的功能码一样，并表明子机已响应主机进行操作。如果子机发送的功能码的最高位是 1(功能码>127)，则表明子机没有响应或出错。

下表列出功能码具体的含义及操作。

ModBus 部分功能码

功能码	定义	操作
01H	读开关量	读取单个或多个开关量
03H	读寄存器	读取一个或多个寄存器数据
05H	置单个开关量	置单个开关量

1. 01H 读开关量

主机可以利用功能码为 01 的通讯命令，读取装置内的各种开关量(如开关合闸、分闸、故障，自动或手动状态等)。

2. 03H 读寄存器

主机利用功能码为 03H 的通讯命令，读取装置内的数值寄存器，数值寄存器内保存的是采集到的各种模拟量和参数的设定值)。功能码 03H 映射的数据区的输入寄存器值都是 16 位(2 字节)。这样从装置读取的寄存器值都是 2 字节。一次最多可读取的寄存器数是 125 个。

子机响应的命令格式是子机地址、功能码、数据区及 CRC 码。数据区的数据

都是每二个字节为一组的双字节数，且高字节在前。

3. 05H 置单个开关量

主机利用这条命令把单个开关量数据保存到装置内的位存储器(如控制 ATS 转换的开关量)。子机也用这个功能码向主机返送信息。

●数据区(DATA)：

数据区随功能码不同而不同。

●错误校验码(CRC)：

主机或子机可用校验码进行判别接收信息是否出错。有时，由于电子噪声或其它一些干扰，信息在传输过程中会发生细微的变化，错误校验码保证了主机或子机对在传送过程中出错的信息不起作用。这样增加了系统的安全和效率。错误校验码采用 CRC-16 校验方法。
二字节的错误校验码，低字节在前，高字节在后。

***注意：**

信息帧的格式都是相同的：地址码、功能码、数据区及错误校验码。

冗余循环码(CRC)包含 2 个字节，即 16 位二进制。CRC 码由发送端计算，放置于发送信息的尾部。接收端的设备再重新计算接收信息的 CRC 码是否与接收到的相同，如果二者不同，则表明出错。

CRC 码的计算方法是，先预置 16 位寄存器全为 1。再逐渐把每 8 位数据信息进行处理。在进行 CRC 码计算时只用 8 位数据位，起始位及停止位都不参与 CRC 码计算。

在计算 CRC 码时，8 位数据与寄存器的数据相异或，得到的结果向低位位移一位，用 0 填补最高位。再检查最低位，如果最低位为 1，把寄存器的内容与预置数异或，如果最低位为 0，不进行异或运算。
这个过程一直重复次。第 8 次移位后，下一个 8 位再与现在的寄存器的内容相异或，这个过程与上次一样重复 8 次。当所有的数据信息处理完后，最后寄存器的内容即为 CRC 码值。

CRC-16 码的计算步骤为：

1、 置 16 位 CRC 寄存器为十六进制 FFFF；
2、 把一个 8 位数据与 CRC 寄存器的低 8 位相异或，把结果放于 CRC 寄存器；
3、 把 CRC 寄存器的内容右移一位，用 0 填补最高位，检查移出位。
4、 如果最低位为 0：重复第 3 步（再次移位）。
如果最低位为 1：CRC 寄存器与十六进制数 A001 进行异或。
5、 重复步骤 3 和 4，直到右移 8 次，这样整个 8 位数据全部进行了处理。

6、 重复步骤 2 到 5，进行下一个数据处理。

7、 最后得到的 CRC 寄存器值即为 CRC 码，传送时将低 8 位先发送，高 8 位最后发送。

注：CRC 码的计算从＜子机地址＞开始，除＜CRC 码＞的所有字节。

●信息帧格式举例

◎功能码 01H

子机地址为 00，读取起始地址为 0000H 的 20H(十进制 32)个开关量

主机发送	字节数	举例（十六进制）
子机地址	1	01　　送至子机 01
功能码	1	01　　读取开关量
起始地址	2	00　　起始地址为 0000 00
读取个数	2	00　　读取 28 个开关量 1C
CRC 码	2	3D　　由主机计算得到的 CRC 码 C3

子机响应	字节数	举例（十六进制）
子机地址	1	01　　送至子机 01
功能码	1	01　　读取点
读取字节数	1	04　　28 个开关量（共 4 个字节）
数据 1	1	30　　地址为 07－00 内的内容
数据 2	1	00　　地址为 0F－08 内的内容
数据 3	1	93　　地址为 17－10 内的内容
数据 4	1	0A　　地址为 1C－18 内的内容
CRC 码	2	18　　由子机计算得到的 CRC 码 26

开关量07–00的值用十六进制表示为30H，用二进制表示为00110000，开关量07是字节的高位，00 是低位，开关量07－00的状态是：OFF－OFF－ON－ON－OFF－OFF－OFF－OFF。

◎功能码 03H

子机地址为 01，起始地址为 0026H 的 3 个点

此例中点数据地址为：

地址	数据（十六进制）
0026	0014
0028	0014
002A	0005

主机发送	字节数	举例（十六进制）
子机地址	1	01　　送至子机 01
功能码	1	03　　读取点
起始地址	2	00　　起始地址为 0032 26
读取个数	2	00　　读取 3 个点（共 6 个字节） 03
CRC 码	2	E4　　由主机计算得到的 CRC 码 00

子机响应	字节数	举例（十六进制）
子机地址	1	01　　送至子机 01
功能码	1	03　　读取点
读取字节数	1	06　　3 个点（共 6 个字节）
点 1 数据	2	00　　地址为 0026 内的内容 14
点 2 数据	2	00　　地址为 0028 内的内容 14
点 3 数据	2	00　　地址为 002A 内的内容 05
CRC 码	2	91　　由子机计算得到的 CRC 码 71

◎功能码 05H

子机地址为 01，起始地址为 0002H 的 1 个开关量，置 0002 单元为 1

此例中开关量数据地址为：

地址	数据（十六进制）
0000	0
0001	1
0002	0

说明：十六进制值FF00强制开关量为1，0000H强制为0，其它值则为非法且不影响开关量的状态

主机发送	字节数	举例（十六进制）
子机地址	1	01　　送至子机 01
功能码	1	05　　强制开关量
起始地址	2	00　　起始地址为 0002 04
数据	2	FF　　开关量置 1 00
CRC 码	2	CD　　由主机计算得到的 CRC 码 FB

子机响应	字节数	举例（十六进制）
子机地址	1	01　　送至子机 01
功能码	1	05　　强制开关量
起始地址	2	00　　起始地址为 0002 04
数据	2	FF　　开关量置 1 00
CRC 码	2	CD　　由主机计算得到的 CRC 码 FB

●出错处理

当装置检测到了 CRC 码出错以外的错误时，必须向主机返送信息，功能码的最高位置 1，即子机返送的功能码是在主机发送的功能码的基础上加 128。以下的这些代码表明有意外的错误发生。

从主机接收到的信息如有 CRC 错误，则被装置忽略。

子机返送的错误码的格式如下（CRC 除外）：

地址码	1 字节
功能码	1 字节（最高位是 1）
错误码	1 字节
CRC 码	2 字节

错误功能码：

01　非法的功能码
　　接收到的功能码不支持
02　非法的数据地址
　　指定的地址超出子机的范围
03　非法的数据值
　　接收到主机发送的数据值超出相应地址的数据范围。

附录：地址和数据

表 1：功能码 01H 所映射的开关量区

开关量		
地址（Address）	项目(Item)	说明
0000H	1#开关合闸／分闸	为 1 合闸，为 0 分闸
0001H	1#电压正常	为 1 正常，为 0 异常
0002H	2#开关合闸／分闸	为 1 合闸，为 0 分闸
0003H	2#电压正常	为 1 正常，为 0 异常
0004H	自动／手动	为 1 自动，为 0 手动
0005H	1＃主用／备用	为 1 主用，为 0 备用
0006H	2＃主用／备用	为 1 主用，为 0 备用
0007H	油机起动输出	1：开机输出 0：停机输出
0008H	严重故障*1	为 1 故障，为 0 无故障
0009H	1＃开关故障（过流或短路）	为 1 故障，为 0 无故障分闸
000AH	2＃开关故障（过流或短路）	为 1 故障，为 0 无故障分闸
000BH	1＃合闸失败	为 1 有效
000CH	2＃合闸失败	为 1 有效
000DH	1＃分闸失败	为 1 有效
000EH	2＃分闸失败	为 1 有效
000FH	保留	未定义
0010H	一般告警 *3	为 1 告警，为 0 无告警
0011H	1＃电压过高	为 1 过高，为 0 正常
0012H	1＃电压过低	为 1 过低，为 0 正常
0013H	1＃缺相	为 1 缺相，为 0 不缺相
0014H	2＃电压过高	为 1 过高，为 0 正常
0015H	2＃电压过低	为 1 过低，为 0 正常
0016H	2＃缺相	为 1 缺相，为 0 不缺相
0017H	负载过流	为 1 过流，为 0 正常
0018H	1＃过频	为 1 过频，为 0 正常
0019H	1＃欠频	为 1 欠频，为 0 正常
001AH	2＃过频	为 1 过频，为 0 正常
001BH	2＃欠频	为 1 欠频，为 0 正常
001CH	历史记录	为 1 有记录，为 0 无记录
001DH	保留	未定义

001EH	保留	未定义
001FH	保留	未定义

*1：严重故障包括 1#,2#开关合闸失败，开关分闸失败，开关故障。

*2：当严重故障为 1 时，报警输出有效，输出延时可通过编程实现。

*3：一般告警包括 1#,2#电压过高，电压过低，缺相，过频，欠频和负载过流。

表 2：功能码 03H 所映射的数据区

数据区(DATA)		
地址（Address）	项目(Item)	说明
0000H	Uab1	1# AB 相电压
0002H	Ubc1	1# BC 相电压
0004H	Uca1	1# CA 相电压
0006H	Uab2	2# AB 相电压
0008H	Ubc2	2# BC 相电压
000AH	Uca2	2# CA 相电压
000CH	Ua1	1# A 相电压
000EH	Ub1	1# B 相电压
0010H	UcI	1# C 相电压
0012H	Ua2	2# A 相电压
0014H	Ub2	2# B 相电压
0016H	Uc2	2# C 相电压
0018H	Ia	A 相负载电流
001AH	Ib	B 相负载电流
001CH	Ic	C 相负载电流
001EH	F1	1#频率
0020H	F2	2#频率
0022H	保留	保留
0024H	P	视在功率(Kva)

表 2：功能码 05H 所映射的开关量区

开关量		
地址（Address）	项目(Item)	说明
0000H	遥控 1#开关合闸	置 1 合闸，置 0 无效
0001H	遥控 1#开关分闸	置 1 合闸，置 0 无效
0002H	遥控 2#开关合闸	置 1 合闸，置 0 无效
0003H	遥控 2#开关分闸	置 1 合闸，置 0 无效
0004H	自动／手动设置	1：自动　0：手动
0005H	1＃主用状态设置	0：备用 1：主用 *
0006H	2＃主用状态设置	0：备用 1：主用 *

*：如果 1#已设为主用状态，当 2#设为主用状态时，自动将 1#设为备用状态。如果 2#已设为主用状态，当 1#设为主用状态时，自动将 2#设为备用状态。