

此文档共分为四章

第一章使用上位机软件联机控制 DPM8600

第二章使用串口助手调试 DPM8600

第三章 简易通信协议

第四章 MODBUS 通讯协议

注：此协议适用于 DPM8605、DPM8608、DPM8616、DPM8624、等所有 DPM8600 系列产品

~~~~~

## 第一章使用上位机软件联机控制 DPM8600

一、先用的“USB 转 TTL 数据线”把 DPM8600 和电脑连接起来，给电源模块供电（10-75V 直流电），并把波动开关切换到 ON，使电源通电；DPM8600-485 系列的按照 A-A,B-B 进行接线。



## 二、打开资料包双击运行

install



### 三、选择中文



### 四、按照顺序依次安装 (USB 驱动) - (Ni-Visa) - (应用程序), 注意每一步都不能少,

安装完成后再桌面找到这个图标并打开  DPM8600;

### 五、选中“我的电脑”-设备管理器-查看分配的端口号;



### 六、打开上位机软件如下图, 注意选择相对应的“端口”;

“通讯波特率”、“从机地址”要和 DPM8600 电源内部设置的一致, 机器默认通讯波特

然后点一下联机，显示相对应的机器型号，表示通讯成功：



可以根据我们提供的 “简易通讯协议 ”和 “MODBUS 通讯协议 ”二次开发控制 DPM8600，两种通讯协议可以在 DPM8600 功能 “5-CS ”中自由选择，0 表示 “自定义通讯协议 ”，1 表示 “MODBUS 通讯协议 ”

## 使用通讯协议编程前可以先用串口助手调试命令

例如: :01w10=1234,\r\n, 表示电压设定值为: 12.34V



## 综述

控制指令总体结构采用命令行方式，通讯速率可以在七种波特率（2400、4800、9600、19200、38400、57600、115200）之间选择，机器地址码可以设定范围在 01-99 共计 99 个，具体设定操作请看说明书。

由 PC 机发送命令, 本机解析执行, 在地址码一致的情况下, 把结果返回给 PC 机, 地址码不同时不返回任何信息, 这个非常适合用于多机集中控

制的情况。

协议中的电流统一表示为 0.001A 精度，DPM-8616 和 DPM-8624 产品显示精度 0.01A 自动忽略小数点后第三位。

发送指令格式如下：

| 起始符 | 地址码   | 功能码 | 功能号   | 连接符 | 操作数     | 间隔符 | 结束符  |
|-----|-------|-----|-------|-----|---------|-----|------|
| :   | 01~99 | w,r | 00~99 | =   | 0~65535 | ,   | \r\n |

- 1、起始符是 “:”
- 2、地址码是本机地址，设定范围是 01~99。
- 3、功能码是小写字母 ‘w’ 或 ‘r’，表示写入或者读出。
- 4、功能号是给不同功能的编号，数值不同代表不同的参数设置
- 5、操作数是命令的操作数
- 6、间隔符：不同数据之间用 “,” 间隔。
- 7、结束符，一个命令的的结尾用\r\n 表示结束，这在 ASCII 代码里面是回车和换行，十六进制表示为 0x0d,0x0a。

-----

一.w 指令

(1) 10 命令 写入电压设定值

格式为 :01w10=\*\*\*\*,

其中 “\*\*\*\*” 表示一个数值，表示设定电压值

例如: :01w10=1234,\r\n, 表示电压设定值为: 12.34V

(2) 11 命令 写入电流设定值

格式为 :01w11=\*\*\*\*,

其中 “\*\*\*\*” 表示一个数值，表示电流值

例如 :01w11=12345, 表示电流设定值为: 12.345A

### (3) 12 命令 输出状态设置

格式为 :01w12=\*,

其中 “\*” 表示 1 个数字, 代表输出状态

例如: :01w12=0, 表示关闭输出

:01w12=1, 表示开启输出

### (4) 20 命令 同时设定电压和电流设定值

格式为 :01w20=\*\*\*\*,####, \r\n,

其中 “\*\*\*\*” 表示电压设定值, “####” 表示电流设定值,

例如: :01w20=1234,2345, 表示电压和电流设定值为: 12.34V、

2.345A

二.r 指令

### (1) 00 命令 读取最大输出电压

格式为 :01r00=0,

例如: 发送:01r00=0, 返回:01r00=6000, 表示最大输出电压 60V

### (2) 01 命令 读取最大调节电流

格式为 :01r01=0,

例如: 发送:01r01=0,

如果返回:01r01=24000, 表示最大输出电流 24A, 对应的型号是

DPM-8624

如果返回:01r01=16000, 表示最大输出电流 16A, 对应的型号是

DPM-8616

如果返回:01r01=8000, 表示最大输出电流 8A, 对应的型号是

DPM-8608

如果返回:01r01=5000, 表示最大输出电流 5A, 对应的型号是

DPM-8605

### (3) 10 命令 读取电压设定值

格式为 :01r10=0,

例如: 发送 :01r10=0, 返回 :01r10=1234, 表示预设电压值为

12.34V

(4) 11 命令 读取电流设定值

格式为 :01r11=0,

例如：发送:01r11=0, 返回 :01r11=12345, 表示电流设定值为 12.345A

(5) 12 命令 读取输出状态

格式为 :01r12=0,

例如：发送:01r12=0, 返回:01r12=0, 表示目前为关闭输出状态

例如：发送:01r12=0, 返回:01r12=1, 表示目前为开启输出状态

(6) 30 命令 读取输出电压测量值

格式为 :01r30=0,

例如：发送:01r30=0, 返回:01r30=2345, 表示当前输出电压测量值为 23.45V

(7) 31 命令 读取输出电流测量值

格式为 :01r31=0,

例如：发送:01w31=0, 返回:01r31=12345, 表示当前输出电流测量值为 12.345A

(8) 32 命令 读取 CC CV 状态

格式为 :01r32=0,

例如：发送:01r32=0,

如果返回 :01r32=0, 表示恒压输出 CV

如果返回 :01r32=1, 表示恒流输出 CC

(9) 33 命令 读取温度

格式为 :01r33=0,



### 1.1 地址码：

地址码是每次通讯信息帧的第一字节（8 位），从 1 到 255。这个字节表明由用户设置地址的从机将接收由主机发送来的信息。每个从机都必须有唯一的地址码，并且只有符合地址码的从机才能响应回送信息。当从机回送信息时，回送数据均以各自的地址码开始。主机发送的地址码表明将发送到的从机地址，而从机返回的地址码表明回送的从机地址。相应的地址码表明该信息来自于何处。

### 1.2 功能码：

功能码为每次通讯信息帧传送的第二个字节，ModBus 通讯规约可定义的功能码为 1 到 127。作为主机请求发送，通过功能码告诉从机应执行什么动作。作为从机响应，从机返回的功能码与从主机发送来的功能码一样，并表明从机已响应主机并且已进行相关的操作。本机仅支持 0x03、0x06、0x10 功能码。

| 功能码  | 定 义    | 操 作（二进制）        |
|------|--------|-----------------|
| 0x03 | 读寄存器数据 | 读取一个或多个寄存器的数据   |
| 0x06 | 写单个寄存器 | 把一组二进制数据写入单个寄存器 |
| 0x10 | 写多个寄存器 | 把多组二进制数据写入多个寄存器 |

### 1.3 数据区：

数据区包括需要由从机返送何种信息或执行什么动作，这些信息可以是数据（如：开量输入/输出、模拟量输入/输出、寄存器等等）、参考地址等。例如，主机通过功能码 03 告诉从机返回寄存器的值（包含要读取寄存器的起始地址及读取寄存器的长度），则返回的数据包括寄存器的数据长度及数据内容。

#### 0x03 读取功能主机格式

| 地址码  | 功能码  | 寄存器起始地址 | 寄存器地址数量 n | CRC 校验码 |
|------|------|---------|-----------|---------|
| 1 字节 | 1 字节 | 2 字节    | 2 字节      | 2 字节    |

#### 0x03 读取功能从机返回格式

| 地址码  | 功能码  | 返回字节数 2*n | 寄存器数据   | CRC 校验码 |
|------|------|-----------|---------|---------|
| 1 字节 | 1 字节 | 1 字节      | 2*n 个字节 | 2 字节    |

#### 0x06 写单个寄存器功能主机格式

| 地址码  | 功能码  | 寄存器地址 | 寄存器数据 | CRC 校验码 |
|------|------|-------|-------|---------|
| 1 字节 | 1 字节 | 2 字节  | 2 字节  | 2 字节    |

#### 0x06 写单个寄存器功能从机返回格式

| 地址码  | 功能码  | 寄存器地址 | 寄存器数据 | CRC 校验码 |
|------|------|-------|-------|---------|
| 1 字节 | 1 字节 | 2 字节  | 2 字节  | 2 字节    |

#### 0x10 写功能主机格式

| 地址码  | 功能码  | 寄存器起始地址 | 寄存器地址数量 n | 写入字节数 2*n | 寄存器数据  | CRC 校验码 |
|------|------|---------|-----------|-----------|--------|---------|
| 1 字节 | 1 字节 | 2 字节    | 2 字节      | 1 字节      | 2*n 字节 | 2 字节    |

#### 0x10 写功能从机返回格式



| 地址码  | 功能码  | 寄存器起始地址 | 寄存器地址数量 n | CRC 校验码 |
|------|------|---------|-----------|---------|
| 1 字节 | 1 字节 | 2 字节    | 2 字节      | 2 字节    |

### 协议寄存器介绍(单个寄存器地址内的数据为双字节型数据)

| 序号 | 名称    | 说明    | 字节数 | 范围     | 小数点 | 单位 | 读写  | 通讯地址  | 备注  |
|----|-------|-------|-----|--------|-----|----|-----|-------|-----|
| 1  | Set-U | 电压设定值 | 2   | 0-Umax | 2   | V  | r/w | 0000H | 注 1 |
| 2  | Set-I | 电流设定值 | 2   | 0-Imax | 3   | A  | r/w | 0001H | 注 1 |
| 3  | SW    | 输出开关  | 2   | 0, 1   | 0   | -  | r/w | 0002H | 注 2 |
| 4  | CCCV  | 输出状态  | 2   | 0,1,2  | 0   | -  | r   | 1000H | 注 3 |
| 5  | U     | 电压显示值 | 2   | 实际值    | 2   | V  | r   | 1001H | 注 4 |
| 6  | I     | 电流显示值 | 2   | 实际值    | 3   | A  | r   | 1002H | 注 4 |
| 7  | T     | 温度显示值 | 2   | 实际值    | 0   | ℃  | r   | 1003H | 注 5 |

#### 注释:

1. Set-U, Set-I为两个设定值，用于控制电源的电压和电流设定值。
2. 控制输出是否开启，写入0关闭，写入1开启，同时也可以读取状态。
3. 这个是只读寄存器，读出0表示无输出，1表示CV状态，2表示CC状态。
4. U, I为两个实时测量值，用于回读电源的电压和电流数值。
5. T为温度测量值，用于回读电源的内部温度。
6. 协议中的电流精度统一为0.001A。

#### 1.4 错误校验码 (CRC 校验):

主机或从机可用校验码进行判别接收信息是否正确。由于电子噪声或一些其它干扰，信息在传输过程中有时会发生错误，错误校验码 (CRC) 可以检验主机或从机在通讯数据传送过程中的信息是否有误，错误的数据可以放弃 (无论是发送还是接收)，这样增加了系统的安全和效率。MODBUS 通讯协议的 CRC (冗余循环码) 包含 2 个字节，即 16 位二进制数。

CRC 码由发送设备 (主机) 计算，放置于发送信息帧的尾部。接收信息的设备 (从机) 再重新计算接收到信息的 CRC，比较计算得到的 CRC 是否与接收到的相符，如果两者不相符，则表明出错。CRC 校验码发送时低位在前，高位在后。

#### CRC 码的计算方法：

- (1) 预置 1 个 16 位的寄存器为十六进制 FFFF (即全为 1)；称此寄存器为 CRC 寄存器；
- (2) 把第一个 8 位二进制数据 (既通讯信息帧的第一个字节) 与 16 位的 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 (朝低位) 用 0 填补最高位，并检查右移后的移出位；
- (4) 如果移出位为 0：重复第 3 步 (再次右移一位)；如果移出位为 1：CRC 寄存器与多项式 A001 (1010 0000 0000 0001) 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 到步骤 5，进行通讯信息帧下一个字节的处理；
- (7) 将该通讯信息帧所有字节按上述步骤计算完成后，得到的 16 位 CRC 寄存器的高、低字节进行交换；
- (8) 最后得到的 CRC 寄存器内容即为 CRC 码。

## 三、通讯实例

**例 1：主机读取输出电压和输出电流显示值**

主机发送的报文格式：

| 主机发送    | 字节数 | 发送的信息 | 备 注           |
|---------|-----|-------|---------------|
| 从机地址    | 1   | 01    | 发送至地址为 01 的从机 |
| 功能码     | 1   | 03    | 读寄存器          |
| 寄存器起始地址 | 2   | 0000H | 寄存器起始地址       |
| 寄存器地址数量 | 2   | 0002H | 共 2 个字节       |
| CRC 码   | 2   | C40BH | 由主机计算得到 CRC 码 |

例如如当前显示值是 05.00V，5.000A，则从机响应返回的报文格式：

| 从机响应             | 字节数 | 返回的信息 | 备 注           |
|------------------|-----|-------|---------------|
| 从机地址             | 1   | 01    | 来自从机 01       |
| 功能码              | 1   | 03    | 读寄存器          |
| 读取字节数            | 1   | 04    | 共 1 个字节       |
| 地址为 0002H 寄存器的内容 | 2   | 01F4H | 输出电压显示值       |
| 地址为 0003H 寄存器的内容 | 2   | 1388H | 输出电流显示值       |
| CRC 码            | 2   | B76BH | 由从机计算得到 CRC 码 |

**例 2：主机要设定电压 24.00V**

主机发送的报文格式：

| 主机发送             | 字节数 | 发送的信息 | 备 注           |
|------------------|-----|-------|---------------|
| 从机地址             | 1   | 01H   | 来自从机 01       |
| 功能码              | 1   | 06H   | 写单个寄存器        |
| 寄存器地址            | 2   | 0000H | 寄存器地址         |
| 地址为 0000H 寄存器的内容 | 2   | 0960H | 设定输出电压值       |
| CRC 码            | 2   | 8FB2H | 由主机计算得到 CRC 码 |

从机接收后响应返回的报文格式：

| 从机响应             | 字节数 | 返回的信息 | 备 注           |
|------------------|-----|-------|---------------|
| 从机地址             | 1   | 01H   | 发送至地址为 01 的从机 |
| 功能码              | 1   | 06H   | 写单个寄存器        |
| 寄存器地址            | 2   | 0000H | 寄存器起始地址       |
| 地址为 0000H 寄存器的内容 | 2   | 0960H | 设定输出电压值       |
| CRC 码            | 2   | 8FB2H | 由从机计算得到 CRC 码 |

**例 3：主机要设定电压为 24.00V，电流 1.500A。**

主机发送的报文格式：

| 主机发送    | 字节数 | 发送的信息 | 备 注     |
|---------|-----|-------|---------|
| 从机地址    | 1   | 01H   | 来自从机 01 |
| 功能码     | 1   | 10H   | 写寄存器    |
| 寄存器起始地址 | 2   | 0000H | 寄存器起始地址 |
| 寄存器地址数量 | 2   | 0002H | 共 2 个字节 |

|                  |   |       |               |
|------------------|---|-------|---------------|
| 写入字节数            | 1 | 04H   | 共 1 个字节       |
| 地址为 0000H 寄存器的内容 | 2 | 0960H | 设定输出电压值       |
| 地址为 0001H 寄存器的内容 | 2 | 05DCH | 设定输出电流值       |
| CRC 码            | 2 | F2E4H | 由主机计算得到 CRC 码 |

从机接收后响应返回的报文格式：

| 从机响应    | 字节数 | 返回的信息 | 备 注           |
|---------|-----|-------|---------------|
| 从机地址    | 1   | 01H   | 发送至地址为 01 的从机 |
| 功能码     | 1   | 10H   | 写寄存器          |
| 寄存器起始地址 | 2   | 0000H | 寄存器起始地址       |
| 寄存器地址数量 | 2   | 0002H | 共 2 个字节       |
| CRC 码   | 2   | 41C8H | 由从机计算得到 CRC 码 |

