

LABVIEW 应用动态链接库串口访问 DK2XXX 系列仪表的方法

一. ModbusRTU.dll 的函数介绍:

1. **int EXAPI OpenPort_Modbus(int Port,int Baud,int crc);**

本函数调用成功返回值 0, 失败返回-1.

int Port: 访问的串口端口号, 就是串口 COM1, 2, 3,.....。如果测试串口 6, 数值就是十进制的 6。

int Baud: 通讯速度波特率。

0-4800

1-9600

2-19200

3-38400

4-57600

5-115200

6-230400

int crc:

0-无校验 N

1-奇校验 ODD

2-偶校验 EVEN

2. **int EXAPI Get_3X_DX(int Port,int Addr,int MAddr,int PN,int *DReg);** //连续读 MODBUS 寄存器地址是 30000X(功能码 04)开始的 PN 个有符号数据。返回有符号的 PN 个有符号数据。

int Port: 访问的串口端口号, 就是串口 COM1, 2, 3,.....。如果测试串口 6, 数值就是十进制的 6。

int Addr: 仪表的通讯地址, 1-247

int MAddr: 访问的 MODBUS 寄存器起始地址。比如地址 1, 就输入 1, 实际对应 MODBUS 寄存器地址 300001。注意 MODBUS 地址规范是地址加 1 处理。函数已经做加 1 处理。

int PN: 访问从 MAddr 寄存器地址开始的数据个数, 即读回多少个参数。

int *Dreg: 返回读回的参数数组。

3. **int EXAPI Get_3X_UnDX(int Port,int Addr,int MAddr,int PN,int *DReg);** //连续读 MODBUS 寄存器地址是 30000X(功能码 04)开始的 PN 个无符号数据。返回无符号的 PN 个有符号数据。

int Port: 访问的串口端口号, 就是串口 COM1, 2, 3,.....。如果测试串口 6, 数值就是十进制的 6。

int Addr: 仪表的通讯地址, 1-247

int MAddr: 访问的 MODBUS 寄存器起始地址。比如地址 1, 就输入 1, 实际对应 MODBUS 寄存器地址 300001。注意 MODBUS 地址规范是地址加 1 处理。函数已经做加 1 处理。

int PN: 访问从 MAddr 寄存器地址开始的数据个数, 即读回多少个参数。

int *Dreg: 返回读回的参数数组。

3. **int EXAPI Get_4X_UnDX(int Port,int Addr,int MAddr,int PN,int *DReg);** //连续读 MODBUS 寄存器地址是 40000X(功能码 03)开始的 PN 个无符号数据。返回无符号的 PN 个有符号数据。

int Port: 访问的串口端口号, 就是串口 COM1, 2, 3,.....。如果测试串口 6, 数值就是十进制的 6。

int Addr: 仪表的通讯地址, 1-247
int MAddr: 访问的 MODBUS 寄存器起始地址。比如地址 1 , 就输入 1, 实际对应 MODBUS 寄存器地址 40001。注意 MODBUS 地址规范是地址加 1 处理。函数已经做加 1 处理。
int PN: 访问从 MAddr 寄存器地址开始的数据个数, 即读回多少个参数。
int *Dreg: 返回读回的参数数组。

4. int EXAPI Get_4X_DX(int Port, int Addr, int MAddr, int PN, int *DReg); //连续读 MODBUS 寄存器地址是 4000X(功能码 03)开始的 PN 个有符号数据。返回有符号的 PN 个有符号数据。
int Port: 访问的串口端口号, 就是串口 COM1, 2, 3,。如果测试串口 6, 数值就是十进制的 6。
int Addr: 仪表的通讯地址, 1-247
int MAddr: 访问的 MODBUS 寄存器起始地址。比如地址 1 , 就输入 1, 实际对应 MODBUS 寄存器地址 40001。注意 MODBUS 地址规范是地址加 1 处理。函数已经做加 1 处理。
int PN: 访问从 MAddr 寄存器地址开始的数据个数, 即读回多少个参数。
int *Dreg: 返回读回的参数数组。

5. float EXAPI Get_3X_Float(int Port, int Addr, int MAddr, int PN, float *DReg) ; // 连读 PB 个浮点数 3XXXXXX, 32 位浮点数, 字节顺序为 22 11 44 33

6. float EXAPI Get_4X_Float(int Port, int Addr, int MAddr, int PN, float *DReg) ; // 连读 PB 个浮点数 3XXXXXX, 32 位浮点数, 字节顺序为 22 11 44 33

7. long EXAPI Set_4X_DX(int Port, int Addr, int MAddr, int PN , long *DReg) ; //PN 个数据, DREG 数组写入

8. long EXAPI Set_4X_D1(int Port, int Addr, int MAddr, long Value); //设定一个参数 VALUE 写到 MADDR 地址。

9. long EXAPI Get_0X_B16(int Port, int Addr, int MAddr, long *DReg); //连续读 16 个位输出线圈 (2 个字节), 返回两个数据, 分别代表低 8 位和高 8 位)

10. long EXAPI Get_1X_B16(int Port, int Addr, int MAddr, long *DReg); //连续度 16 个位输出线圈 (2 个字节), 返回两个数据, 分别代表低 8 位和高 8 为)

11. long EXAPI Get_0X_Bit1(int Port, int Addr, int MAddr); //地址为 0000X 的读写位区

12. long EXAPI Get_1X_Bit1(int Port, int Addr, int MAddr); //地址为 1000X 的只读位区

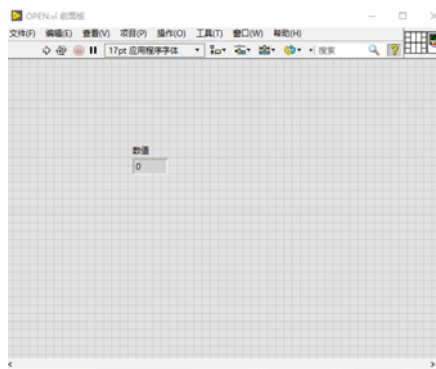
```
13. long EXAPI Set_0X_Bit1(int Port,int Addr,int MAddr); //  
地址为 0000X 的读写区置位
```

```
14. long EXAPI Rst_0X_Bit1(int Port,int Addr,int MAddr); //  
地址为 0000X 的读写区复位
```

```
15. int EXAPI ClosePort_Modbus(int Port);
```

二. LABVIEW 调用动态链接库访问仪表示例

1. 首先打开 LABVIEW , 新建 VI

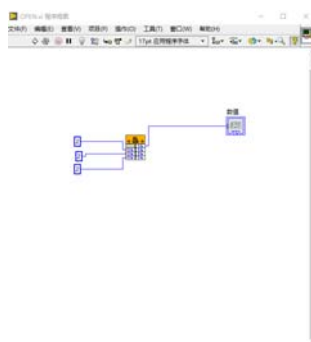


2. 打开本 VI 程序框图:

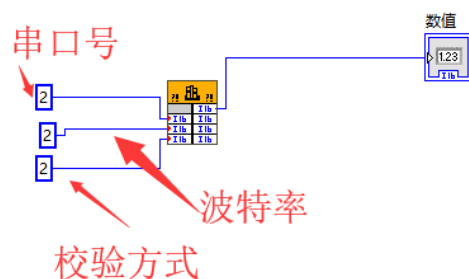
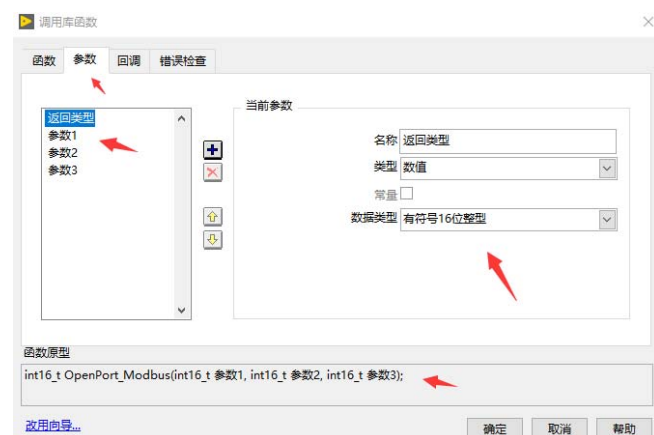
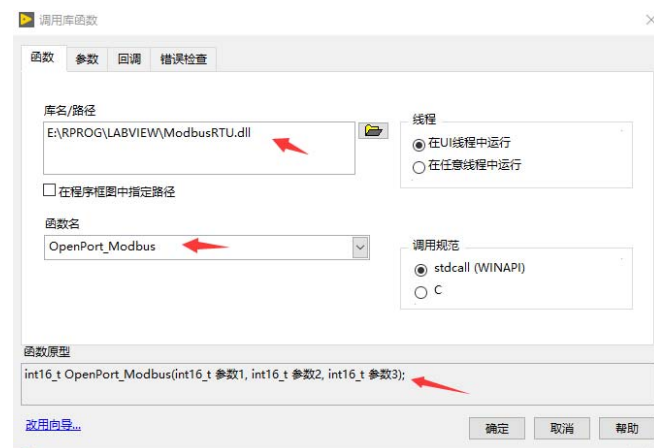


在后面板选件添加

在程序框图中放置库函数选件



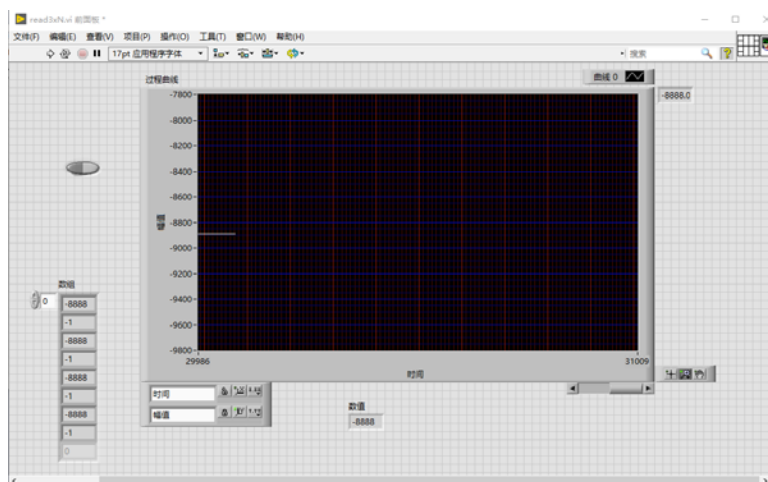
对库函数功能进行设置：



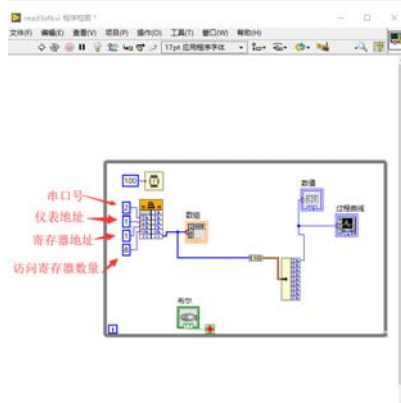
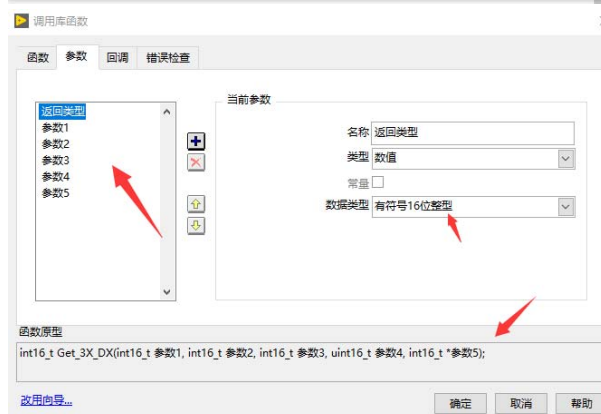
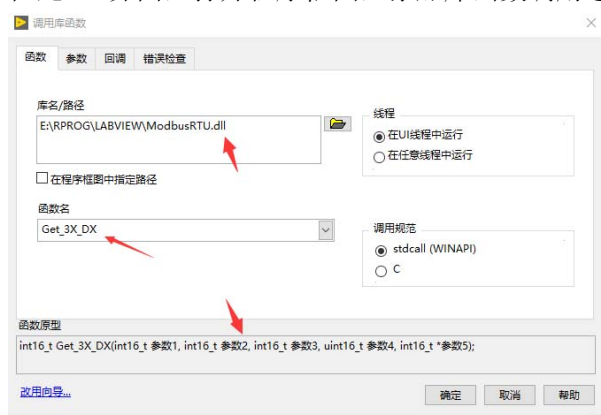
到此为止，我们已经建立了一个和串口通讯设置的函数： OPENPORT_MODBUS，运行此 VI ，与串口通讯的端口就被打开。 注意函数返回-1 为失败。

3. 用函数 Get_3X_DX 测试访问仪表的数据

首先我们在新建一个 VI 如下：



在此 VI 界面，打开程序框图，添加库函数调用选项



运行此 VI，仪表数据被连续读回 8 个寄存器值。

