RS232_RS485 自由协议使用说明

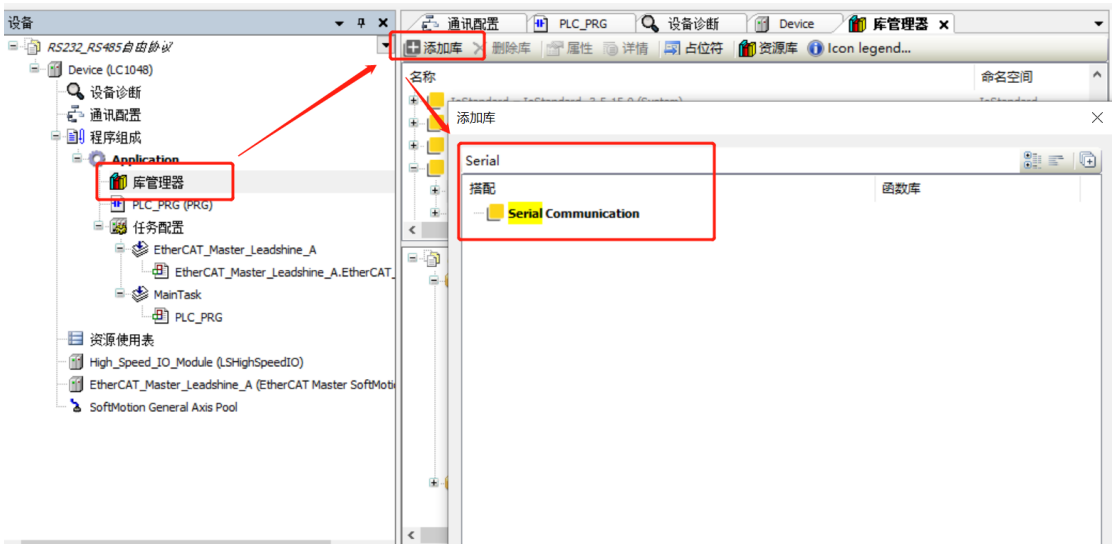
1、概要

自由协议通讯也称无协议通讯，需要根据对方设备的通讯数据格式编写一个 临时协议，虽然随着标准协议（Modbus，USS 等）的普及，自由协议通讯应用 越来越少，但是对于一些小的设备如扫码枪、LCD 显示屏等，由于没有集成标 准通讯协议，所以只能选用自由协议通讯。以下例程是基于 LC3048 的 RS232 及 RS485 硬件端口的自由协议通信例程说明。1）功能运行前的准备：1 个控制器，1 台电脑，电脑安装 Commix 串口调试软件 （其余串口调试软件均可）

2、LC 端设置及程序编写

1) 创建如图所示的变量

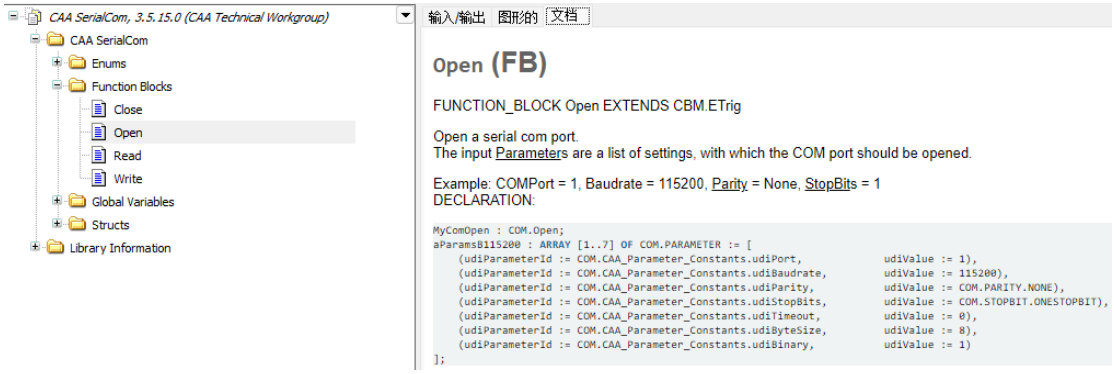
左键双击【库管理器】—左键单击【添加库】，搜索【Serial Communication】，单 击【确定】即可添加完成。



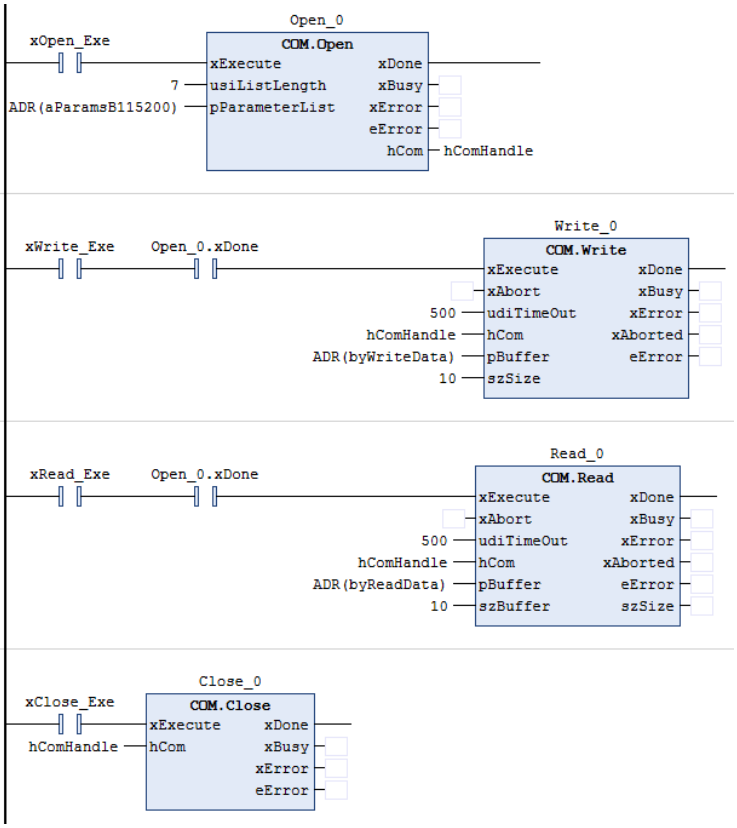
2) 创建如图所示的变量

```
PROGRAM PLC_PRG
VAR
    Open_0: COM.Open;
    Write_0: COM.Write;
    Read_0: COM.Read;
    Close_0: COM.Close;
    xOpen_Exe:BOOL;
    hComHandle:CAA.HANDLE;
    byWriteData:BYTE;
    xWrite_Exe:BOOL;
    xRead_Exe:BOOL;
    byReadData:BYTE;
     xClose_Exe:BOOL;
    aParamsB115200 : ARRAY [1..7] OF COM.PARAMETER := [
    (udiParameterId := COM.CAA_Parameter_Constants.udiPort,udiValue := 2),//COM口,对于LC2000/5000产品，COM1 为485，COM2是232
    (udiParameterId := COM.CAA_Parameter_Constants.udiBaudrate,udiValue := 115200),//波特率，设置为115200
    (udiParameterId := COM.CAA_Parameter_Constants.udiParity,udiValue := INT_TO_UDINT(COM.PARITY.EVEN)),//校验，设置为偶校验
    (udiParameterId := COM.CAA_Parameter_Constants.udiStopBits,udiValue := INT_TO_UDINT(COM.STOPBIT.ONESTOPBIT)),//停止位，1位
    (udiParameterId := COM.CAA_Parameter_Constants.udiTimeout,udiValue := 500),//超时时间，设置成500ms
    (udiParameterId := COM.CAA_Parameter_Constants.udiByteSize,udiValue := 8),//数据位
    (udiParameterId := COM.CAA_Parameter_Constants.udiBinary, udiValue := 1)//二进制参数，启用二进制模式
];
END_VAR
```

3)调用如图所示的功能块



4) 编写如图所示程序



① 调用 COM.Open，此功能块为打开串口；变量 usiListLength 为变量 pParameterList

的数据长度（根据后续介绍可知，数据长度为 7）；变量 pParameterList 为指针变量，指针地址指向结构体 COM.PARAMETER 的数组 aParamsB115200，数组包含了 udiPort,udiValue(波特率)、udiBaudrate,udiValue (端口号)、udiParity,udiValue (校验)、udiStopBits,udiValue (停止位)、udiTimeout,udiValue (超时时间)、udiByteSize,udiValue（数据位）及 udiBinary,udiValue（二进制参数，启用二进制模式）7 种关于串口的参数，相关的变量创建直接参照下列库文件中的例程，再对参数进行设置；变量 hCom 为结构体 CAA.HANDLE 数据类型，获取通讯句柄；

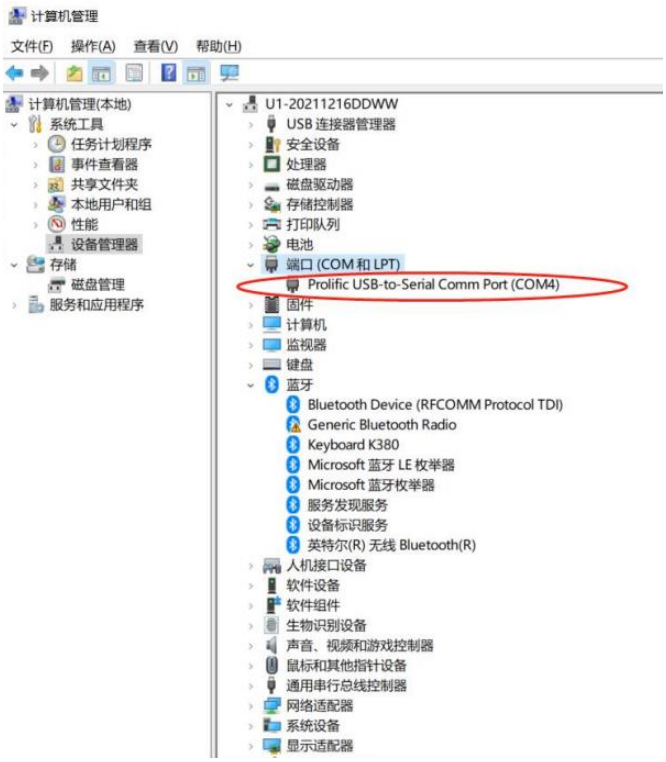
② udiPort,udiValue: 端口号，根据使用不同硬件端口进行赋值；LC3000 系列 RS232 值为 1，第一路 RS485 值为 7，第二路 RS485 值为 9。udiBaudrate,udiValue:波特率，此次例程设置波特率为 115200; udiParity,udiValue：校验类型（奇校验、偶校验或无校验），此次例程设置偶校验，赋值 COM.PARITY.EVEN；udiStopBits,udiValue: 停止位（ONESTOPBIT / ONE5STOPBIT / TWOSTOPBIT），此次例程设置为 ONESTOPBIT；udiTimeout,udiValue：超时时间，此次例程设置成 500ms；udiByteSize,udiValue：数据位，此次例程设置成 8；udiBinary,udiValue：二进制参数，采取默认值设置成 1 即可；设置完成后由于 pParameterList 为指针类型，则填入 ADR(aParamsB115200)即可；

③ 调用 COM.Write，此功能块为串口发送数据；变量 udiTimeout 超时时间，此次例程设置为 500ms；变量 hCom 为结构体 CAA.HANDLE 数据类型，为通讯句柄，可从 Com.Open 中的输出变量 hCom 获取；pBuffer 为指针类型，数据指向串口发送数据缓存的地址，本例程设置写入数值变量为 BYTE 类型的 byWriteData，因此本例程填入 ADR（byWriteData）；szSize 为写入数据长度，此次例程设置为 10 个 BYTE 数据长度，因此填入 10；

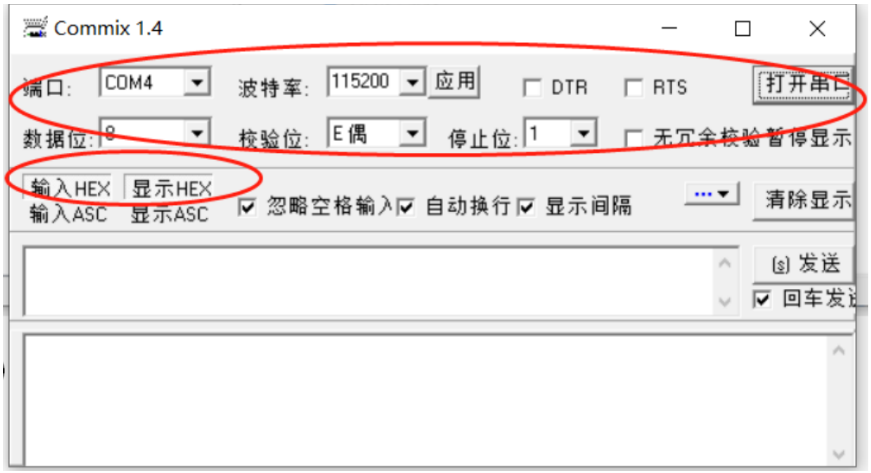
④ 调用 COM.Read，此功能块为串口读取数据；变量 udiTimeout 超时时间，此次例程设置为 500ms；变量 hCom 为结构体 CAA.HANDLE 数据类型，为通讯句柄，可从 Com.Open 中的输出变量 hCom 获取；pBuffer 为指针类型，数据指向串口读取数据缓存的地址，本例程设置写入数值变量为 BYTE 类型的 byReadData，因此本例程填入 ADR（byReadData）；szBuffer 为从串口缓存读取数据的大学，此次例程设置为 10 个 BYTE 数据长度，因此填入 10；⑤ 调用 COM.Close，此功能块为关闭串口；变量 hCom 为结构体 CAA.HANDLE 数据类型，为通讯句柄，可从 Com.Open 中的输出变量 hCom 获取；

3、打开串口助手设置相关参数

1) 打开计算机管理，查看计算机串口端口号（本次例程为 USB 转串口，端口号为 COM4）

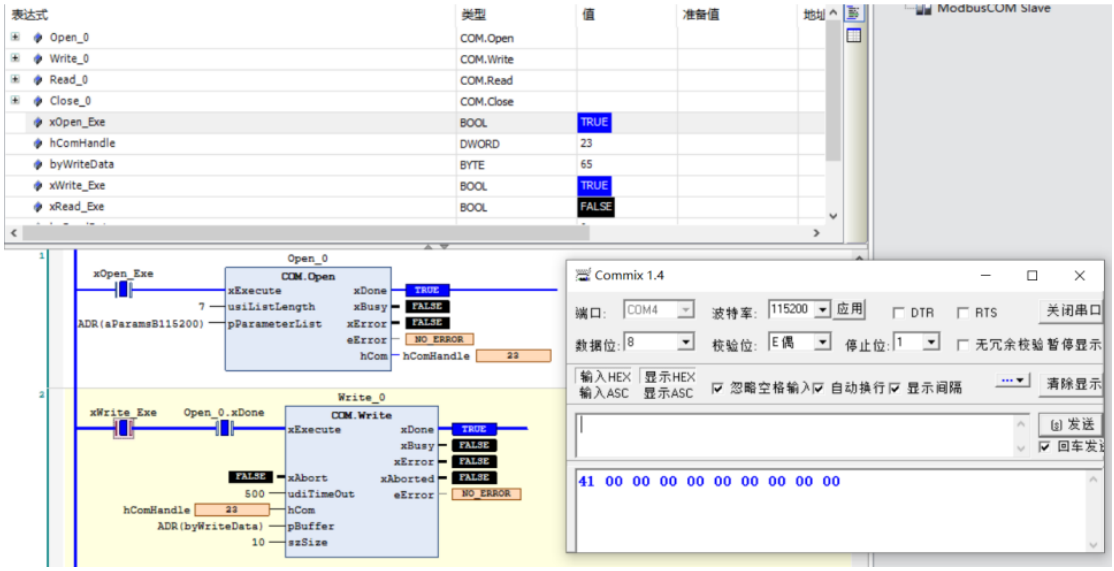


2) 打开串口助手软件 Commix 并设置相关参数 【端口】COM4（计算机端口号）;【波特率】115200;【数据位】8;【校验位】E 偶;【停止位】1; 左键单击选择【输入 HEX】及【显示 HEX】，设置以 16 进制 数据类型进行写入及显示; 左键单击【打开串口】即可。



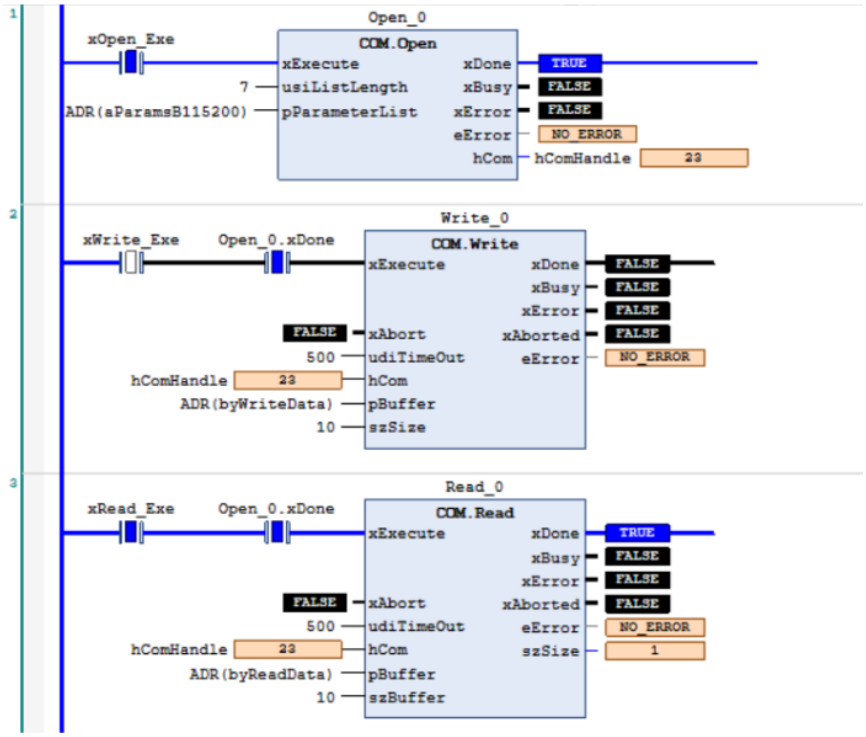
4、PLC 与电脑端做数据发送及读取

1) PLC 发送数据，电脑端读取数据 触发 COM.Open 打开串口端口，对 byWriteData 赋值十进制数 65（16 进制为 0x41）后触发 COM.Write,可见电脑端可读取 PLC 发送的数值。



2) 电脑端发送数据，PLC 端读取数据 通过串口助手发送 16 进制数值 0x86；触发 COM.Open 打开串口端口，关闭串口 数据写入功能块，触发 COM.Read 功能块,可成功读取 16 进制数值 0x86（十进制 为 134）





表达式	类型	值	准备值	地址
Open_0	COM.Open			
Write_0	COM.Write			
Read_0	COM.Read			
Close_0	COM.Close			
xOpen_Exe	BOOL	TRUE		
hComHandle	DWORD	23		
byWriteData	BYTE	65		
xWrite_Exe	BOOL	FALSE		
xRead_Exe	BOOL	TRUE		
byReadData	BYTE	134		
xClose_Exe	BOOL	FALSE		
aParamsB115200	ARRAY [1..7] OF C...			