```
/// Create a communication connection - 创建一个通讯连接
public bool Connect(string hostOrCom, int baudOrPort)

/// DisConnect communication - 断开通讯
public void DisConnect()

/// Get base information - 获取基本信息
public int Information(ref int oAddress, ref string oVer, ref string oMode);

/// Set Basic Parameters - 设置基本参数
public int pSetConfig(byte[] iData);

/// Get Basic Parameters - 获取基本参数
public int pGetConfig(ref byte[] oData);

/// Set Address - 设置读卡器通讯地址
public int pSetAddress(int iData);

/// Reset Reader - 重启读卡器
public int pReset();

/// Init Syris SN And ID(nonstandard) - 初始化读卡器序列号和 ID(非标准命令)
public int pInitSyris(byte[] iData);

/// Identify card from ISO18000-6C(EPC) tag - 识别单张 ISO18000-6C(EPC)标签卡号
public int pIdentify6C(ref byte[] oData);

/// Identify cards from mult ISO18000-6C(EPC) tag - 识别多张 ISO18000-6C(EPC)标签卡号
public int pIdentify6CMult(ref byte[] oData);

/// Read data from ISO18000-6C(EPC) tag - 读取 ISO18000-6C(EPC)标签数据
public int pRead6C(int iMem, int iStartWord, int iLengthWord, ref byte[] oData);

/// Write Data to ISO18000-6C(EPC) tag - 写入 ISO18000-6C(EPC)标签数据
public int pWrite6C(int iMem, int iStartWord, int iLengthWord, string iData);

/// Write Data to ISO18000-6C(EPC) tag - 写入 ISO18000-6C(EPC)标签数据
public int pWrite6C(int iMem, int iStartWord, int iLengthWord, byte[] iData);

/// Identify card from ISO18000-6B tag - 识别 ISO18000-6B 标签卡号
public int pIdentify6B(ref byte[] oData);

/// Read data from ISO18000-6B tag - 读取 ISO18000-6B 标签数据
```

```
public int pRead6B(int iStart, int iLength, ref byte[] oData);

/// Write Data to ISO18000-6B tag - 写入 ISO18000-6B 标签数据
public int pWrite6B(int iStart, int iLength, string iData);

/// Write Data to ISO18000-6B tag - 写入 ISO18000-6B 标签数据
public int pWrite6B(int iStart, int iLength, byte[] iData);

/// Get Tcpip Parameters(nonstandard) - 获取 TCPIP 参数(非标准命令)
public int pGetTcpip(ref byte[] oData);

/// Set Tcpip Parameters(nonstandard) - 设置 TCPIP 参数(非标准命令)
public int pSetTcpip(byte[] iData);

/// Exit Progarm Mode(only usb reader) - 将 USB 读卡器退出编程模式(仅用于 USB 通讯版读卡器)
public int pExitProgarm();

/// Set Progarm Mode(only usb reader) - 设置 USB 读卡器为编程模式(仅用于 USB 通讯版读卡器)
public int pSetProgarm();

/// Set Virtual Keyboard Mode(only usb reader) - 设置 USB 读卡器为虚拟键盘模式(仅用于 USB 通讯版读卡器)
public int pSetKeyboard();

/// Set USB OutCard Parameters(only usb reader) - 设置 USB 读卡器输出卡号参数(仅用于 USB 通讯版读卡器)
public int pSetOutCard(byte[] iData);

/// Get USB OutCard Parameters(only usb reader) - 获取 USB 读卡器输出卡号参数(仅用于 USB 通讯版读卡器)
public int pGetOutCard(ref byte[] oData);

/// Set IO Out[nonstandard] - 设置 IO 口输出(非标准命令)
public int pRemote(byte iPosition, byte iAction);

/// Get Datetime Parameters(nonstandard) - 获取读卡器实时时间(非标准命令)
public int pGetTime(ref byte[] oData);

/// Set Datetime Parameters(nonstandard) - 设置读卡器实时时间(非标准命令)
public int pSetTime(byte[] iData);
```