# W7282 读写器

# 用户手册

# （V2.0）


北京握奇智能科技有限公司

# 重要声明：

随着读写器产品的升级，本手册内容将会做相应的修改。握奇智能科技有限公司保留对本手册内容进行修改的权利。



# 声明

此产品为 A 级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，可能需要用户对其干扰采取切实可行的措施。

# 目录

# 1. 概述

W7282 读写器系列是握奇公司自主研发、生产的一款符合 ISO 7816、磁条卡符合 ISO 7810、ISO 7811、ISO 7812、ISO 7813 标准、射频接口符合 ISO14443 标准，能实现对 CPU 卡(typeA /TypeB)以及 Mifare 卡(S50/S70) 的操作，并且符合读写器 GB18239 的标准，具有一个用户卡座和 4 个 SAM 卡座，用户卡座能实现对 CPU 卡(T=0/T=1)以及逻辑加密卡(包括 4428/4442/1608)卡片的操作，SAM 卡座能实现对 CPU 卡进行操作。该读写器支持 USB 通讯接口，拥有多种功能，外形美观，性能稳定的特点。同时使用该设备无需安装驱动即可以在 Windows 操作系统中正常使用该设备，使用户操作更方便快捷。

## 1.1 功能概述

- 支持符合ISO 7816协议智能卡
- 支持接触4442、4428、1608存储卡
- 磁条卡支持符合ISO 7810、ISO 7811、ISO 7812、ISO 7813标准
- 支持符合ISO 14443 typeA /TypeB标准
- 符合HID协议
- 支持全速USB通讯
- LED指示灯，指示电源或通讯状态
- 内置用户可控的蜂鸣器
- 支持外接串口设备
- 提供通用接口函数库，可支持多种操作系统和语言开发平台

- 读卡器底层可在线升级

## 1.2 符合标准

- 符合ISO/IEC 7816-1/2/3/4标准
- 符合ISO 7810、ISO 7811、ISO 7812、ISO 7813标准
- 符合ISO 14443 typeA /TypeB标准
- 符合读卡器GB18239的标准
- 符合USB2.0 标准

## 1.3 主要技术指标

| 参数 | 指标 |
|---|---|
| 支持接触卡 | 逻辑加密卡：4428、4442、1608<br>CPU 卡：符合 ISO7816<br>磁条卡：符合 ISO 7810、ISO 7811、ISO 7812、ISO 7813 标准的磁条卡 |
| 支持非接触卡 | CPU 卡：符合 ISO14443-1/2/3/4 标准智能片<br>存储卡：Mifare One 卡（S50、S70） |
| 卡座 | 12：用户卡座,可以支持 ISO7816 标准的 cpu 卡和 4442/4428/1608 的逻辑加密卡 |

| | |
|---|---|
| | 15： 支持 CPU 卡（TypeA/ TypeB）和 Mifare（S50/S70）<br>16： SAM 卡座，符合 ISO7816 标准的 cpu 卡<br>17： SAM 卡座，符合 ISO7816 标准的 cpu 卡<br>18： SAM 卡座，符合 ISO7816 标准的 cpu 卡<br>19： SAM 卡座，符合 ISO7816 标准的 cpu 卡<br>1A：磁条卡 |
| 射频卡读卡距离 | 0~2.5cm |
| 射频场强 | 0cm:：2.7A/m |
| 读写器材质 | ABS |
| 磁条卡模块刷卡速度 | 7－120cm/s |
| 电源 | USB 供电（两根线可以同时接入 Pc 机，以保证稳定供电） |
| USB 通讯 | USB2.0 全速 12Mbps |
| 通讯协议 | 符合 HID 的通讯协议（即插即用） |

| | |
|---|---|
| 接触 CPU 卡通讯速率 | 默认 9600bps 支持自动 PPS |
| 工作电流 | ≤300mA |
| 外型尺寸（H×W×D） | 166*99*64 |
| 工作温度 | 0℃ ～ 50℃ |
| 工作湿度 | 20% ～ 90% |
| LED 灯 | 红绿指示灯 |
| MTBF | 5000 小时 |
| 操作系统 | Windows XP/2003/VISTA/WIN7/WIN8/WIN10 Ubutu/中标麒麟 |

# 2. 通讯协议

本通讯协议指的是 IC 卡读写器与上位机之间数据传输的格式，用户也可以按照此格式，通过不同的系统与 IC 卡读写器进行通讯连接。总体来说，该通讯协议就是在 ISO7816 协议的 APDU 指令基础上，在头尾各增加相应的数据，以保证通信数据的完整和正确性。

特别说明：本手册里与命令相关的的数字默认为十六进制。

## 2.1 发送到读写器的命令格式

| 信息域 | 标识 | 字节长度 | 含义 |
|---|---|---|---|
| 通信数据头 | NAD | 1 | NAD 卡座选择：<br>0x00 DFU 更新<br>0x12 用户卡座<br>0x15 非接触卡座<br>0x16 SAM1 卡座<br>0x17 SAM2 卡座<br>0x18 SAM3 卡座<br>0x19 SAM4 卡座<br>0x1A 磁条卡 |
| | PCB | 1 | 包编号，默认为 00 |
| | LEN | 1 | 数据长度，包括 CLA INS P1 P2 Lc DATA 的长度 |
| APDU 指令 | CLA | 1 | 指令类型 |
| | INS | 1 | 指令码 |
| | P1 | 1 | 指令参数 1 |
| | P2 | 1 | 指令参数 2 |
| | Lc | 1 | 输入数据长度或期望返回数据长度 |
| | DATA | 0-FF | 输入数据 |

例 1：CPU 卡复位命令

12 00 05 00 12 00 00 00

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

NAD PCB LEN CLA INS P1 P2 Lc

例 2：CPU 卡写二进制命令

12 00 0A 00 D6 00 00 05 1122334455

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

NAD PCB LEN CLA INS P1 P2 Lc DATA

例 3：CPU 卡读二进制命令

12 00 05 00 B0 00 00 05

↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓

NAD PCB LEN CLA INS P1 P2 Lc

## 2.2 从读写器返回的信息格式

| 标识 | 字节长度 | 含义 |
|---|---|---|
| NAD | 1 | 发送命令 NAD 的半字节互换 |

| | | |
|---|---|---|
| | | 例如发送 NAD=12，返回 NAD=21 |
| PCB | 1 | 包编号，默认为 00 |
| LEN | 1 | 数据长度，包括 DATA SW1 SW2 |
| DATA | 0-FF | 返回数据 |
| SW1 | 1 | 状态字节 1 |
| SW2 | 1 | 状态字节 2 |

例 1 返回信息如下：

21 00 13 3B6D0000574446224A864341301F131C12 90 00

↓ ↓ ↓ ↓ ↓ ↓

NAD PCB LEN D A T A SW1 SW2

其中 90 00 是读写器自动补上的状态字节 SW1 SW2

例 2：返回信息如下：

21 00 02 90 00

↓ ↓ ↓ ↓ ↓

NAD PCB LEN SW1 SW2

例 3：返回信息如下：

21 00 07 1122334455 90 00

↓ ↓ ↓ ↓ ↓ ↓

NAD PCB LEN DATA SW1 SW2

# 3. 应用命令概述

W7282读写器提供8个通道对相应的设备进行设置或对卡片进行操作。

（1）NAD=0x15：非接触指令透传通道，通过该通道实现对非接触卡片的操作以及该通道支持的自定义命令，命令参见4.1。

（2）NAD=0x12：接触卡座指令透传通道，该通道可以实现对接触卡的操作，接触卡包括，命令参见4.2。

（3）NAD=/0x16/0x17/0x18/0x19：4个SAM卡座指令透传通道，该通道可以实现对接触的SAM卡的操作，可以支持标准的接触式CPU卡（T=0/T=1）的APDU命令。

（4）NAD=0x1A：磁条卡通道，该通道的命令参见4.3。

（5）NAD=0x00：固件版本更新通道，自定义命令的执行通道参见4.4。

# 3.1 NAD=0x15 通道设备参数配置说明

此通道是非接触通道。通过该通道实现 PC 机与非接触卡片之间的通讯。该通道可以自动识别 TypeA 和 TypeB 两类非接触卡片，以及标准的 Mifare one(S50/S70)卡片。

## 3.1.1 标准非接触卡的复位命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 00 | 冷复位，关场后再复位 |
| **INS** | **1** | 12 | |
| **P1** | **1** | 00 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 00 | |

响应报文数据域：

| Data | 说明 |
|---|---|
| XX | 返回非接触卡片的复位信息 |

应用举例：
命令：0012000000
响应：0800C150961A1D200777F7A0024792 9000
说明：对非接触卡片进行复位操作，设备返回了正确的复位信息。

## 3.1.2 Mifare One 卡复位命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 09 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 00 | |

应用举例：

命令：8011090000
响应：0400DEB097FB0208 9000
说明：设备对 Mifare One 卡成功复位并返回复位信息。

### 3.1.3 MIfare One 密钥认证命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 02 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 08 | |
| **Data** | **8** | Data | |

命令报文数据域：

| Data | 说明 |
|---|---|
| 1 字节 | 密钥类型：00：keyA　02：keyB |
| 2 字节 | 认证块号 |
| 3 字节～6 字节 | 密钥值 |

应用举例：
命令：80110200080018FFFFFFFFFFFF
响应：9000
说明：设备用 keyA 密钥对 Mifare One 卡的 24(十进制数据)块进行密钥验证。

### 3.1.4 MIfare One 读块命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 03 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 01 | |
| **Data** | **8** | Data | |

命令报文数据域：

| Data | 说明 |
|---|---|
| 1 字节 | 经过认证的块号 |

应用举例：
命令：801103000118
响应：A0A1A2A3A4A5FF078069B0B1B2B3B4B5 9000
说明：密钥验证通过后，成功读取 Mifare One 卡的 24(十进制数据)块的数据。

### 3.1.5 MIfare One 写块命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 04 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 11 | |
| **Data** | **17** | Data | |

命令报文数据域：

| Data | 说明 |
|---|---|
| 1 字节 | 经过认证的块号 |
| 2 字节～17 字节 | 写入的数据字节 |

应用举例：
命令：80110400111811223344556677889900AABBCCDDEEFF
响应：9000
说明：密钥验证通过后，成功将数据写入 Mifare One 卡的 24(十进制数据)块。

### 3.1.6 MIfare One 初始化钱包文件命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 05 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 05 | |
| **Data** | **5** | Data | |

命令报文数据域：

| **Data** | **说明** |
|---|---|
| 1 字节 | 经过认证的块号 |
| 2 字节～5 字节 | 4 字节的金额 |

应用举例：
命令：80110500051810000000
响应：9000
说明：密钥验证通过后，成功将 Mifare One 卡的 24(十进制数据)块初始化成钱包文件。

### 3.1.7 MIfare One 钱包存款命令

| **代码** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 06 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 05 | |
| **Data** | **5** | Data | |

命令报文数据域：

| **Data** | **说明** |
|---|---|
| 1 字节 | 经过认证的块号 |
| 2 字节～5 字节 | 4 字节的金额 |

应用举例：
命令：80110600051810000000
响应：9000
说明：密钥验证通过后，将 Mifare One 卡的 24(十进制数据)块钱包存款成功。

### 3.1.8 MIfare One 钱包扣款命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 11 | |
| **P1** | **1** | 07 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 05 | |
| **Data** | **5** | Data | |

命令报文数据域：

| Data | 说明 |
|---|---|
| 1 字节 | 经过认证的块号 |
| 2 字节～5 字节 | 4 字节的金额 |

应用举例：
命令：80110700051801000000
响应：9000
说明：密钥验证通过后，将 Mifare One 卡的 24(十进制数据)块钱包扣款成功。

## 3.2 NAD=0x12 通道设备参数配置说明

此通道是通过大用户卡座与读写器进行通讯的通道。

### 3.2.1 读卡状态命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|

| | | | |
|---|---|---|---|
| **CLA** | 1 | 80 | |
| **INS** | 1 | 17 | |
| **P1** | 1 | 01 | |
| **P2** | 1 | 01 | |
| **Le** | 1 | 00 | |

应用举例：
命令：8017010100
响应：9000
说明：读写器的接触卡槽内有卡。

### 3.2.2 上电命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | CLA | |
| **INS** | **1** | 12 | |
| **P1** | **1** | 00 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 00 | |

命令报文数据域_CLA：

| CLA | **说明** |
|---|---|
| 00 | 默认复位方式，第一次对卡是冷复位，以后是热复位 |
| 80 | 冷复位 |

应用举例：
命令：8012000000
响应：3B6D0000574446224A864341301F131C12 9000
说明：设备对接触卡片成功复位。

### 3.2.3 卡类型命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 17 | |
| **P1** | **1** | 00 | |
| **P2** | **1** | 59 | |
| **Lc** | **1** | 01 | |

响应报文数据域：

| XX | 说明 |
|---|---|
| 00 | CPU 卡 |
| 09 | SLE4418/4428 |
| 0A | SLE4432/4442 |
| 15 | AT88SC1608 |

应用举例：
命令：8017005901
响应：09 9000
说明：用户卡槽内有接触卡，并且接触卡的类型是 SLE4428 卡。

### 3.2.4 逻辑加密卡读命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | B0 | |
| **P1** | 1 | P1（高位地址） | |
| **P2** | 1 | P2（低位地址） | |
| **Le** | 1 | Len | |

命令报文数据域_ P1P2：

| P1 | P2 | P1 P2 起始地址取值 |
|---|---|---|
| 0 | XX | SLE4442：使用 P2，P1 置为 0 |
| XX | XX | SLE4428：需共同使用 P1 和 P2 |

命令报文数据域_ Len：

| Len | 说明 |
|---|---|
| =00 | 表示读 256 字节命令 |
| ≠00 | 读 P1P2 指定的地址开始的 Len 字节数据 |

应用举例：
命令：00b0008204
响应：11223344 9000
说明：该命令表示对逻辑加密卡进行读数据操作，起始地址从 0x82 开始读取 4 个字节长度的数据，返回 0x9000 表示成功读出的数据为 11223344。

### 3.2.5 校验密码命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 20 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | 00 | |
| **Lc** | 1 | Len | |
| **Data** | 1 | Pin | |

命令报文数据域_ Len：

| Len | 说明 |
|---|---|
| 03 | SLE4442 卡口令的长度 |
| 02 | SLE4428 卡口令的长度 |

命令报文数据域_ Pin：

| Pin | 说明 |
|---|---|
| FFFFFF | SLE4442 卡口令默认值是 FFFFFF |
| FFFF | SLE4428 卡口令默认值是 FFFF |

应用举例：
命令：0020000003FFFFFF
响应：9000
说明：该命令执行完成后，返回 0x9000 表示对 SLE4442 卡片的口令校验通过。

### 3.2.6 逻辑加密卡写命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | D0 | |
| **P1** | 1 | P1（高位地址） | |
| **P2** | 1 | P2（低位地址） | |
| **Lc** | 1 | Len | |
| **Data** | xx | XX | |

命令报文数据域_P1 P2：

| **P1** | **P2** | P1 P2 起始地址取值 |
|---|---|---|
| 0 | XX | SLE4442：使用 P2，P1 置为 0 |
| XX | XX | SLE4428：需共同使用 P1 和 P2 |

命令报文数据域_ Len：

| Len | **说明** |
|---|---|
| XX | 写入数据长度 |

应用举例：
命令：00d000820411223344
响应：9000
说明：该命令表示对 SLE4442 逻辑加密卡进行写数据操作，起始地址从 0x82 开始写入 4 个字节长度的数据 11223344，返回 0x9000 表示成功写入卡片。
说明：写命令之前需要验证口令，验证口令成功才可以进行写入数据的操作。

### 3.2.7 修改密码命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 24 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | 00 | |
| **Lc** | 1 | Len | |
| **Data** | 1 | 新 Pin | |

命令报文数据域_ Len：

| Len | 说明 |
|---|---|
| 03 | SLE4442 卡口令的长度是 3 个字节 |
| 02 | SLE4428 卡口令的长度是 2 个字节 |

命令报文数据域_ Pin：

| 新 Pin | 说明 |
|---|---|
| XXXXXX | 设置 SLE4442 卡新口令 |
| XXXX | 设置 SLE4428 卡新口令 |

应用举例：
命令：0024000003112233
响应：9000
说明：该命令执行完成后，返回 0x9000 表示对 SLE4442 卡片的口令修改成了 112233。

### 3.2.8 SLE4428 带保护位读命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | |
| **INS** | 1 | B0 | |
| **P1** | 1 | P1（高位地址） | |
| **P2** | 1 | P2（低位地址） | |

| | | | |
|---|---|---|---|
| **Le** | 1 | Len | |

命令报文数据域_ Len：

| Len | **说明** |
|---|---|
| =00 | 表示读 256 字节命令 |
| ≠00 | 读 P1P2 指定的地址开始的 Len 字节数据 |

响应报文数据域：

| **返回** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **Data** | Len/2 | 相应地址数据+保护位字节+下一地址数据+保护位字节…… | 因保护位单独占 byte，发指令时，LEN 为实际需要长度的 2 倍。<br>保护位说明：<br>0x80:该地址未写保护，可写入<br>0x00:该地址已写保护，不可再写入 |

应用举例：

命令：80B0000A04

响应：FF80 FF80 9000

说明：该命令表示对逻辑加密卡进行带保护位读操作，起始地址从 0x0A 开始读取 4 个字节长度的数据，返回 0x9000 表示成功读出的数据为 FF80FF80，该数据的含义是从 0x0A 位置读出的两个数据是 FFFF，并且 0x0A 位置的数据没有被保护，0x0B 位置的数据也没被保护。

### 3.2.9 SLE4428 带保护位写命令

| **代码** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **CLA** | 1 | 80 | 无需判断写入数据和原来该位置的数据是否一致 |
| **INS** | 1 | D0 | |
| **P1** | 1 | P1（高位地址） | |
| **P2** | 1 | P2（低位地址） | |
| **Lc** | 1 | Len | |
| **Data** | Lc | Data | |

应用举例：
命令：80D0000A02FF00
响应：9000
说明：该命令表示对 4428 卡进行带保护位写操作，起始地址从 0x0A 开始写入 2 个字节的数据，返回 0x9000 表示成功写入数据为 FF00，该数据的含义是从 0x0A 位置写入的两个数据是 FF00，并且 0x0A 位置的数据被保护，0x0B 位置的数据也被保护。如果发送读命令 80B0000A04，则读出的数据为 FF000000。

### 3.2.10 SLE4428 带保护位写命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 20 | 需要判断写入数据和原来该位置的数据是否一致 |
| **INS** | 1 | D0 | |
| **P1** | 1 | P1（高位地址） | |
| **P2** | 1 | P2（低位地址） | |
| **Lc** | 1 | Len | |
| **Data** | Lc | Data | |

应用举例 1：
命令：20D0000A02FF00
响应：6A80
说明：该命令表示对 4428 卡进行带保护位写操作，起始地址从 0x0A 开始写入 2 个字节的数据，返回 0x6A80 表示从 0x0A 位置开始写入的两个数据是 FF00，因为 0x0A 位置的数据原来就是 FF，0x0B 位置的数据原来也是 FF 与目前要写入的数据 00 不一致。所以返回 6A80。

应用举例 2：
命令：20D0000A02FFFF
响应：9000
说明：写入的数据和原来该位置的数据一致，都是 FFFF，故成功写入，返回 9000。

### 3.2.11 SLE4442 读保护存储器命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | |

| | | | |
|---|---|---|---|
| **INS** | 1 | B0 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | 00 | |
| **Le** | 1 | Len | |

Len 说明：
LEN 小于 4 时，返回长度为 LEN 长度数据，
当 LEN 大于等于 4 时，返回 4bytes 保护存储器数据。因保护存储区共 4bytes

应用举例：
命令：80B0000008
响应：30FF1FF8 9000
说明：该命令表示对 SLE4442 卡片进行读保护存储器操作。读取的保护存储器的数据是 30FF1FF8。

### 3.2.12 SLE4442 读安全存储区命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | B0 | |
| **P1** | 1 | 01 | |
| **P2** | 1 | 00 | |
| **Le** | 1 | 04 | |

应用举例：
命令：00B0010004
响应：00000000 9000
说明：该命令表示对 SLE4442 卡片进行读取安全存储区数据的操作。读取的安全存储区的数据是 00000000。

### 3.2.13 SLE4442 写保护存储器命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | 需要判断写入数据和 |
| **INS** | 1 | D0 | |
| **P1** | 1 | 00 | |

| | | | |
|---|---|---|---|
| **P2** | 1 | P2（保护存储器地址） | 原来该位置的数据是否一致 |
| **Lc** | 1 | Len | |
| **Data** | Lc | Data | |

命令报文数据域_ Len：

| P2 | **说明** |
|---|---|
| 0x00-0x1F | 保护存储器地址取值在 0x00-0x1F 之间，一共 32 个保护位。 |

应用举例：
命令：80D6000A01FF
响应：9000
说明：该命令表示对 SLE4442 卡片的 0x0A 位置的数据进行写保护操作。该位置的目前保护数据为 FF，与该位置读出的数据一致，故保护成功。

### 3.2.14 AT88SC1608 初始化认证命令

| **代码** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 20 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | 00 | |
| **Lc** | 1 | 08 | |
| **Data** | Lc | Q0（初始化的数据） | |

命令：00200000081122334455667788
响应：9000
说明：该命令表示对 AT88SC1608 卡片进行 Q0 的初始化认证，并且认证成功。

### 3.2.15 AT88SC1608 校验认证命令

| **代码** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 20 | |
| **P1** | 1 | 01 | |

| | | | |
|---|---|---|---|
| **P2** | 1 | 00 | |
| **Lc** | 1 | 08 | |
| **Data** | Lc | Q1（加密数据） | |

Q1 是按照客户要求的固定算法进行计算的数据。
命令：0020010008 aa4408e1cda15b85
响应：9000
说明：该命令表示 AT88SC1608 卡片对 Q1 进行校验认证，并且认证成功。

### 3.2.16 AT88SC1608 校验密码命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 20 | |
| **P1** | 1 | P1 | |
| **P2** | 1 | P2 | |
| **Lc** | 1 | 03 | |
| **Data** | 3 | Pin | |

命令报文数据域_ P1：

| P1 | 说明 |
|---|---|
| 0 | 写密码 |
| 1 | 读密码 |

命令报文数据域_ P2：

| P2 | 说明 |
|---|---|
| 1～8 | 密码组号，1 表示 0 区，2 表示 1 区…… |

命令：0020010103343434
响应：9000
说明：该命令表示对 AT88SC1608 卡片的 0 区进行密码校验。返回 9000，密码校验成功。

### 3.2.17 AT88SC1608 修改密码命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 24 | |
| **P1** | 1 | P1 | |
| **P2** | 1 | P2 | |
| **Lc** | 1 | 03 | |
| **Data** | 3 | Pin | |

命令报文数据域_ P1：

| P1 | 说明 |
|---|---|
| 0 | 写密码 |
| 1 | 读密码 |

命令报文数据域_ P2：

| P2 | 说明 |
|---|---|
| 0～7 | 密码组号 |

命令：0024000003FFFFFF
响应：9000
说明：该命令表示对 AT88SC1608 卡片的 0 区进行密码修改。返回 9000，密码修改成功。

### 3.2.18 AT88SC1608 读用户区数据命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | B0 | |
| **P1** | 1 | P1 | |
| **P2** | 1 | P2（起始地址） | |
| **Le** | 1 | Len | |

命令报文数据域_ P1：

| P1 | 说明 |
|---|---|
| 0~7 | 用户区号 |

命令报文数据域_ Len：

| Len | 说明 |
|---|---|
| =00 | 表示读 256 字节命令 |
| ≠00 | 读 P1P2 指定的地址开始的 Len 字节数据 |

应用举例：
命令：00B0010004
响应：FFFFFFFF 9000
说明：该命令表示对 AT88SC1608 卡片的 01 区 0x00 的起始地址进行读 4 个字节的用户区数据的操作。读取的数据为 FFFFFFFF。

### 3.2.19 AT88SC1608 读配置区命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | |
| **INS** | 1 | B0 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | P2 | |
| **Le** | 1 | Len | |

命令报文数据域_ P2：

| P2 | 说明 |
|---|---|
| 0x00~0x7F | 用户区号 |

应用举例：
命令：80B0008001
响应：00 9000
说明：该命令表示对 AT88SC1608 卡片的 0x80 的配置区地址进行读 1 个字节的操作。读取的数据为 00。

### 3.2.20 AT88SC1608 写用户区数据命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | D0 | |
| **P1** | 1 | P1 | |
| **P2** | 1 | P2(起始地址) | |
| **Lc** | 1 | Len | |
| **Data** | 3 | Data | |

命令报文数据域_ P1：

| P2 | 说明 |
|---|---|
| 0～7 | 用户区号 |

命令：00D001010111
响应：9000
说明：该命令表示对 AT88SC1608 卡片的 1 用户区，起始地址为 01 的地址进行写入一个字节 0x11 的操作，返回 9000，写入成功。

### 3.2.21 AT88SC1608 写配置区命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | |
| **INS** | 1 | D0 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | P2(起始地址) | |
| **Lc** | 1 | Len | |

命令：80D0008000
响应：9000
说明：该命令表示对 AT88SC1608 卡片的 0x80 的起始地址进行写配置操作，返回 9000，写入成功。

### 3.2.22 AT88SC1608 写熔丝设置命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | |
| **INS** | 1 | D0 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | 80 | |
| **Lc** | 1 | 00 | |

命令：80D0008000
响应：9000
说明：该命令表示对 AT88SC1608 卡片的 0x80 的起始地址进行写熔丝操作，返回 9000，写入成功。

## 3.3 NAD=0x1A 通道设备参数配置说明

此通道主要是对磁条卡进行操作的通道。默认通道为模拟键盘通道，每次清除卡信息，或等待刷卡超时，都回到默认通普通模式下的通道。在普通通道下，完成刷卡操作后必须执行一次清除信息命令。

### 3.3.1 设置虚拟键盘上传通道命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | 默认第二磁道 |
| **INS** | **1** | 17 | |
| **P1** | **1** | 03 | |
| **P2** | **1** | P2 | |
| **Lc** | **1** | 00 | |

命令报文数据域_ P2：

| P2 | **说明** |
|---|---|
| 01 | 一磁道 |

| | |
|---|---|
| 02 | 二磁道 |
| 03 | 三磁道 |
| 04 | 一二磁道 |
| 05 | 一三磁道 |
| 06 | 二三磁道 |
| 07 | 一二三道 |

该命令执行完后，掉电后设置仍有效。

应用举例：
命令：8017030100
响应：9000
说明：该命令执行完后，只上传第一个磁道的数据。

### 3.3.2 读取设备当前磁道号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 80 | |
| **INS** | **1** | 17 | |
| **P1** | **1** | 03 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 01 | |

响应报文数据域：

| P2 | **说明** |
|---|---|
| 01 | 一磁道 |
| 02 | 二磁道 |
| 03 | 三磁道 |
| 04 | 一二磁道 |
| 05 | 一三磁道 |
| 06 | 二三磁道 |
| 07 | 一二三道 |

应用举例：
命令：8017030001

响应：04 9000
说明：当前设备上传的磁道号为 03，即只上传一二磁道的数据.

### 3.3.3 读取设备当前磁道号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 00 | |
| **INS** | **1** | B0 | |
| **P1** | **1** | P1 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 00 | |

命令报文数据域_ P1：

| P1 | **说明** |
|---|---|
| 00 | 代表所有磁道信息 |
| 01 | 代表第一磁道 |
| 02 | 代表第二磁道 |
| 03 | 代表第三磁道 |

应用举例：
命令：00B0000001
响应：06180996000279549492643 6 9000
说明：当前设备上传的磁道号为 01，即只上传一磁道的数据信息为
06180996000279549492643 6。

### 3.3.4 磁条卡通道转换命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 00 | |
| **INS** | **1** | 19 | |
| **P1** | **1** | 06 | |
| **P2** | **1** | P2 | |

| | | | |
|---|---|---|---|
| **Lc** | **1** | 00 | |

命令报文数据域_ P2：

| P2 | **说明** |
|---|---|
| 00 | 模拟键盘通道 |
| 01 | 普通卡通道（受控模式） |

应用举例：
命令：0019060000
响应：9000
说明：当前设备的刷卡模式采用了模拟键盘的通道方式，即表示无需通过给设备发送命令，就可以直接进行刷卡操作，设备就会将相应通道的数据上传到输出端。

### 3.3.5 等待刷卡命令

| **代码** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **CLA** | **1** | 00 | |
| **INS** | **1** | 19 | |
| **P1** | **1** | 20 | |
| **P2** | **1** | P2（刷卡等待时间） | |
| **Lc** | **1** | 00 | |

命令报文数据域_ P2：

| P2 | **说明** |
|---|---|
| 刷卡等待时间 | 最长时间 3 分钟（单位：秒） |

应用举例：
命令：0019200300
响应：9000
说明：该设备在 3 秒内刷卡成功。

### 3.3.6 清除刷卡信息命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | **1** | 00 | 备注：读取完刷卡信息后，必须清刷卡信息 |
| **INS** | **1** | 19 | |
| **P1** | **1** | 21 | |
| **P2** | **1** | 00 | |
| **Lc** | **1** | 00 | |

应用举例：
命令：0019210000
响应：9000
说明：清除刷卡信息成功。。

## 3.4 NAD=0x00 通道设备参数配置说明

该通道可以完成一些对设备的参数进行获取和设置的命令。

### 3.4.1 取版本号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 19 | |
| **P1** | 1 | 00 | |
| **P2** | 1 | 00 | |
| **Lc** | 1 | 00 | |

应用举例：
命令： 0019000000
响应：576174636820435257373238322031 2E46363330203234303137(Watch CRW7282 1.F630 24017)
说明：获取的设备的版本号。。

### 3.4.2 取读写器的 ID 号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 19 | |
| **P1** | 1 | 01 | |
| **P2** | 1 | 00 | |
| **Le** | 1 | 00 | |

应用举例：
命令：0019010000
响应：57FF760649835748513024879000
说明：获取的设备的 ID 号为 57FF760649835748513024879000。

### 3.4.3 读设备序列号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 19 | |
| **P1** | 1 | 01 | |
| **P2** | 1 | 01 | |
| **Le** | 1 | Len | |

命令报文数据域_ Len：

| Len | 说明 |
|---|---|
| 0x01~0x14 | 设备的序列号的长度不超过 20 个字节 |

应用举例：
命令：0019010104
响应：FFFFFFF 9000
说明：读出的设备的序列号是 FFFFFFF。

### 3.4.4 擦除序列号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 19 | |
| **P1** | 1 | 01 | |
| **P2** | 1 | ee | |
| **Lc** | 1 | 02 | |
| **Data** | 2 | ee ee | |

应用举例：
命令：001901ee02eeee
响应： 9000
说明：返回 9000 表示擦除成功，其它值表示擦除失败。

### 3.4.5 写序列号命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | 19 | |
| **P1** | 1 | 01 | |
| **P2** | 1 | 02 | |
| **Lc** | 1 | N | |
| **Data** | N | **Data** | |

应用举例：写入序列号“WD000001”
命令：00 19 01 02 05 57 44 00 00 01
响应： 9000

**说明：执行写序列号命令之前必须执行“擦除序列号命令”。**

### 3.4.6 蜂鸣器声音命令

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 80 | |
| **INS** | 1 | F2 | |
| **P1** | 1 | Th | |
| **P2** | 1 | Tl | |
| **Lc** | 1 | 00 | |

命令报文数据域_Th：

| Th | 说明 |
|---|---|
| 高位 | 蜂鸣器持续时间参数的高 8 位 |

命令报文数据域_Tl：

| Tl | 说明 |
|---|---|
| 低位 | 蜂鸣器持续时间参数的低 8 位 |

应用举例：
命令：80F2001000
响应：9000
说明：设备蜂鸣器的鸣叫时间持续 16ms。

### 3.4.7 串口工作参数设置命令

串口工作参数
初始上电串口工作速率：115200 波特率
串口通讯速率范围：2400~115200 波特率
数据位长度：固定为 8 为，不可设置
奇偶校验：可设置 3 中方式（无，奇校验，偶校验）
停止位长度：可设置为 1 位或 2 位停止位
一次最大传输数据个数为 255 个字节

| 代码 | 长度（byte） | 值 | 描述信息 |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | F8 | |

| | | | |
|---|---|---|---|
| **P1** | 1 | P1 | |
| **P2** | 1 | P2 | |
| **Lc** | 1 | 03 | |
| **Data** | 3 | Baud(波特率) | |

命令报文数据域_ P1：

| P1 | **说明** |
|---|---|
| 奇偶校验标志位 | 0：无奇偶校验位<br>2：偶校验<br>3：奇校验 |

命令报文数据域_ P2：

| P2 | **说明** |
|---|---|
| 停止位个数标识 | 1：停止位长度为 1 位<br>2：停止位长度为 2 位 |

命令报文数据域_ Baud：

| P2 | **说明** |
|---|---|
| 波特率 | 2400~115200 波特率，需要将该数值转换成十六进制数，长度为 3 个字节。 |

应用举例：
命令：00F800010001C200
响应：9000
说明：串口通讯方式是无校验，停止位为 1 位，通讯速率是 115200 波特率。

### 3.4.8 串口收发数据命令

该命令需要串口调试助手配合完成串口的发送工作。

| **代码** | **长度（byte）** | **值** | **描述信息** |
|---|---|---|---|
| **CLA** | 1 | 00 | |
| **INS** | 1 | F9 | |
| **P1** | 1 | P1 | |

| | | | |
|---|---|---|---|
| **P2** | 1 | P2 | |
| **Lc** | 1 | Len | |
| **Data** | XX | Data(接口数据) | |

命令报文数据域_ P1：

| P1 | **说明** |
|---|---|
| ＝0 | 期望收到的数据长度未知，可为 0~256 之间的任意长度 |
| ≠00 | 期望收到的数据长度 |

命令报文数据域_ P2：

| P2 | **说明** |
|---|---|
| ＝0 | 无返回数据 |
| ≠00 | 等待返回的数据的定时时间，最大等待时间 180 秒 |

应用举例：
命令：00F909081122334455667788 99（PC 机发给设备的 APDU 命令）
此时，设备规定在 8 秒内，通过串口调试助手发送数据 112233445566778899 给设备。
响应：112233445566778899 9000 （设备反馈给 PC 机）
说明：通过串口接收 09 个长度的数据，接收等待时间为 8 秒，接收的数据为 112233445566778899。

## 3.5 APDU 返回状态字说明

在 NAD=0x00 情况下响应状态码，在 NAD=0x12 及 0x22 情况下,返回状态取决于卡片。

| **SW1** | **SW2** | **说明** |
|---|---|---|
| **90** | **00** | 执行成功 |
| **62** | **00** | 无卡 |
| **62** | **05** | 上电失败 |
| **62** | **83** | 认证密钥错误 |
| **69** | **82** | 不满足安全状态 |
| **69** | **83** | 认证被锁定 |
| **69** | **85** | 位已写保护/磁条卡通道错误 |
| **67** | **00** | 长度错误 |

| | | |
|---|---|---|
| **69** | **02** | 超时 |
| **67** | **00** | 长度错误 |
| **6A** | **86** | 地址范围错误 |
| **6A** | **80** | 写保护数据与相应地址数据不同，写失败 |
| **63** | **Cx** | 密码错误，还有 x 次重试机会 |
| **6F** | **F1** | 通讯时编解码格式错误 |
| **6F** | **F2** | 响应位长度错误 |
| **6F** | **F3** | 写卡错误 |
| **6F** | **F4** | 钱包金额错误 |
| **6F** | **F5** | 通讯超时 |
| **6F** | **F6** | 记录未找到 |
| **6F** | **F0** | 卡通讯失败或其它未知的错误 |
| **6D** | **00** | 命令无效 |
| **FF** | **FF** | PC 与读写器通讯失败 |

# 4. 软件函数说明

W2160-HID 读写器系统的工作流程首先是通过 USB 线实现上位机与读写器之间的数据通讯，然后通过操作系统以调用 wdcrwv.dll 动态库的方式，以读写器为传输媒介，将 APDU 指令的透传给卡片，最终实现对卡片的操作。(说明：具体的函数传入传出参数参见《附录》)
动态库接口函数如下：

1) 打开 IC 卡终端端口

HANDLE WINAPI CT_open(char *name,unsigned int param1,unsigned char param2);

2)关闭与读写器相连的端口(必需用 CT_open 打开的端口)

int WINAPI CT_close(HANDLE fd);

3)对设备当前激活插槽中的 IC 卡进行复位

unsigned WINAPI ICC_reset(HANDLE fd,unsigned char *lenr,unsigned char *resp);

4)从外设向 CPU 卡或读写器发送命令 APDU 并接收应答 APDU ， 数据长度小于 255 字节,

unsigned WINAPI ICC_tsi_api( HANDLE fd, unsigned char len, unsigned char *comm, unsigned char *lenr, unsigned char *resp );

5)从外设向 CPU 卡发送命令 APDU 并接收应答 APDU，数据长度大于 255 字节。

UINT WINAPI ICC_TransmitAPDU32( HANDLE hDev, DWORD len, PBYTE comm, PDWORD lenr, PBYTE resp );

6)设置 CPU 卡读写地址 NAD

void WINAPI ICC_set_NAD(HANDLE fd,unsigned char nad);

7)DES 加密

unsigned WINAPI SingleDES(char DESType, unsigned char * SingleDESKey, unsigned int SourDataLen, unsigned char *SourData, unsigned char *DestData );

8)3DES 加密

unsigned WINAPI TripleDES( char DESType, unsigned char * TripleDESKey, unsigned int SourDataLen, unsigned char *SourData, unsigned char *DestData);
unsigned WINAPI TripleDESVB( char DESType, unsigned char * TripleDESKey, unsigned int SourDataLen, unsigned char *SourData, unsigned char *DestData );

# 附录

1) 打开 IC 卡终端端口

HANDLE WINAPI CT_open(
char *name,
unsigned int param1,
unsigned char param2
);

入口参数:

(A) 串行口读写器

name: 可取"COM1", "COM2", "COM3", "COM4"等,字母大小写无关(以下 name 也一样)

param1: 串口读写器为波特率,9600、38400 等

param2: 串口读写器为奇偶校验,可为'O' 奇校验,'E' 偶校验, 'N' 无校验
无特殊要求,一般使用偶校验即可

(B) USB 读写器

name: 可取"USB1", "USB2", "USB3", "USB4"等。

USB 序号分配原则

1. 从当前系统已由设备插入的顺序决定,第一次插入的设备为 USB1,第二次插入设备为 USB2 依次类推

2. 当新插入 USB 设备时驱动从 1 开始查询当前未使用的序号,直到找到为止

3. 序号分配和系统其他 USB 设备无关

param1: 打开设备的模式 1：共享模式 2：独占模式

param2: 未使用,设为 0

(C) U 盘类(仅支持 Watchdata 产品)

name: 可取"FlashDisk1", "FlashDisk2","flashdiskN"(1<=N<=9)等,
序号分配原则，有如下情况:

i) 若系统没有插入任何设备，则按照此时插入顺序依次为"flashdisk1","flashdisk2"等打开。

ii) 若系统之前已插入 N 个设备，此时新插入的顺序号为 N+1，而后依次往后递增。

iii) 对于单个设备，"flashdisk1"既能打开 U 盘设备，也支持 CDROM 类设备。

param1: 未使用,设为 0

param2: 未使用,设为 0

(D) CDROM 类设备(仅支持 Watchdata 产品)

name: 可取"cdrom1","cdrom2","cdromN"(1<=N<=9)等顺序号同(C)

param1: 未使用，设为 0

param2: 未使用，设为 0

(E) 人体学输入设备(HID Key)(仅支持 Watchdata 产品)

name: 可取"hid1","hid2","hidN"等,1<=N<=9,序号分配原则，有以下情况:

i) 系统上所有 N 个设备均处于断开状态，则调用"hid1","hid2",...,"hidN"依次打开设备，跟插入顺序无关。

ii) 系统上存在 N 个已建立连接的设备，插上一新设备并打开此设备，则需从"hid1"～"hid(N+1)"依次调用，

已打开过的设备将返回 INVALID_HANDLE_VALUE,由此定位到新插入的设备。

param1: 未使用,设为 0

param2: 未使用,设为 0

返回值: INVALID_HANDLE_VALUE(-1) 表示打开端口失败;

其他值(大于 0)为打开的端口句柄,用于卡操作函数的 fd

说明：如需 USB 端口绑定功能，则打开设备时调用 CT_openUSB_ByPort()函数，该函数参数说明如下：

name: "usb1","usb2",...,"usbN",N 的取值由 UsbCurrentDevice.ini 文件中的[PortN]决定；

Param1: 打开设备的模式 1：共享模式 2：独占模式

Param2: 设备类型,由 UsbCurrentDevice.ini 文件中的[DeviceInfo]－>DeviceDesc 字符串决定，

具体取值如下：

| [DeviceDesc] | Param2 |
|---|---|
| Smartcard Reader | 1 |
| USB Key | 1 |
| USB Mass Storage(Disk) | 2 |
| 人体学输入设备 | 3 |
| USB Mass Storage(CDROM) | 4 |

```
HANDLE WINAPI CT_open(
char              *name,
unsigned int      param1,
unsigned char     param2
);
```

HANDLE WINAPI CT_openUSB_ByPort(char *name,unsigned int Param1,unsigned char Param2);

2)关闭与读写器相连的端口(必需用 CT_open 打开的端口)

int WINAPI CT_close(HANDLE fd);

入口参数:

fd 为函数 CT_open 所返回的句柄

返回值:

-1 为失败 0 成功.

2)关闭与读写器相连的端口(必需用 CT_open 打开的端口)

int WINAPI CT_close(HANDLE fd);
入口参数:
fd 为函数 CT_open 所返回的句柄
返回值:
-1 为失败 0 成功.

3)对设备当前激活插槽中的 IC 卡进行复位
unsigned WINAPI ICC_reset(
HANDLE fd,
unsigned char *lenr,
unsigned char *resp
);
参数：
fd : 已打开的端口描述符
lenr : 为对 IC 卡复位所返回的复位数据的长度
resp : 复位的数据结果
返回值：
0x9000 成功
0x6200 无卡
0x6201 协议不认识
0xffff 通迅失败
示例：

```
//以下示例，为对 CPU 卡片进行复位
unsigned int sw;            //定义变量分别用于保存返回状态值
unsigned int lenr, resp[256];  //定义变量分别用于保存返回数据长度及数据
ICC_set_NAD(fDev,0x12);   //设置 NAD 为 0x12
sw = ICC_reset(fDev, &lenr, resp);            //执行复位指令
if(sw != 0x9000)          //判断是否执行成功
{
    …  //操作失败后，用户的处理
}
```

4)从外设向 CPU 卡或读写器发送命令 APDU 并接收应答 APDU ，数据长度小于 255 字节,可以兼容 CRW 系列读写器。
unsigned WINAPI ICC_tsi_api(
HANDLE fd,
unsigned char len,
unsigned char *comm,
unsigned char *lenr,
unsigned char *resp
);
comm 的结构: CLA INS P1 P2 Lc DATA [Le] 其中 DATA 长度为 Lc 字节

resp 的结构: DATA 其中 DATA 长度为 Le 字节
参数:
fd: 已打开的端口描述符.
len: 命令 comm 的长度
comm: 发向卡上的命令
lenr: 从卡上接收到的数据长度
resp: 从卡上接收到的数据
返回值:
0xffff 通迅失败(发送命令或接收返回的数据失败.)
其它为从读写器或卡上返回的状态 SW1 SW2
说明:
该函数适用于所有 CPU 卡操作 或读写器命令

5)从外设向 CPU 卡发送命令 APDU 并接收应答 APDU，数据长度大于 255 字节，CRW-X 读写器专用命令。

UINT WINAPI ICC_TransmitAPDU32(
HANDLE hDev,
DWORD len,
PBYTE comm,
PDWORD lenr,
PBYTE resp
);
comm 的结构: CLA INS P1 P2 Lc DATA [Le] 其中 DATA 长度为 Lc 字节
resp 的结构: DATA 其中 DATA 长度为 Le 字节
参数:
fd: 已打开的端口描述符.
len: 命令 comm 的长度
comm: 发向读写器的命令
lenr: 从卡上接收到的数据长度
resp: 从卡上接收到的数据
返回值:
0xffff 通迅失败(发送命令或接收返回的数据失败.
其它为从卡上返回的状态 SW1 SW2

6)设置 CPU 卡读写地址 NAD

void WINAPI ICC_set_NAD(HANDLE fd,unsigned char nad);
参数:
NAD 读写地址
返回值:
无
说明:
系统缺省值为 00
12 对 SAM1 卡操作,
13 对 SAM2 卡操作,

15 对非接触卡操作

其它，保留以后使用

7)DES 加密

unsigned WINAPI SingleDES(char DESType,
unsigned char * SingleDESKey,
unsigned int SourDataLen,
unsigned char *SourData,
unsigned char *DestData
);

参数:

DESType: =1 加密，=2 解密

SingleDESKey: 8 字节密钥

SourDataLen: 源数据长度

SourData: 源数据

DestData: 目标数据

说明:

加密时,当明文长度不时 8 的倍数时,该函数在明文数据的后面加上 16 进制数字串"80 00 00...",使其为 8 的倍数后加密.

返回值:

目标数据的长度

8)3DES 加密

参数:

DESType: =1 加密，=2 解密

TripleDESKey: 16 字节密钥 K1K2

SourDataLen: 源数据长度

SourData: 源数据

DestData: 目标数据

说明:

加密时,当明文长度不时 8 的倍数时,该函数在明文数据的后面加上 16 进制数字串"80 00 00...", 使其为 8 的倍数后加密

加解密过程如下:

DES3-E({K1,K2},P)=E(K1,D(K2,E(K1,P)))

DES3-D({K1,K2},C)=D(K1,E(K2,D(K1,P)))

返回值：

目标数据的长度

unsigned WINAPI TripleDES(
char DESType,
unsigned char * TripleDESKey,
unsigned int SourDataLen,
unsigned char *SourData,
unsigned char *DestData
);

```
unsigned WINAPI TripleDESVB(
char DESType,
unsigned char * TripleDESKey,
unsigned int SourDataLen,
unsigned char *SourData,
unsigned char *DestData
);
```