

W5182 接触式 EMV 读卡器

用户手册

(V2.0)



北京握奇数据系统有限公司

修订记录

| 时间 | 版本 | 修订内容 |
|-----------|-------------|---------------------------------|
| 2010-08 | V1.0 | 初稿 |
| 2012-4-26 | V1.1 | 增加产品图片，更正产品尺寸，更正通讯速率，驱动安装说明进行更新 |
| 2016-3-28 | V2.0 | 参数更新 |
| | | |

目录

| | |
|----------------------------------|---|
| 修订记录 | 1 |
| 一、读写器功能和性能介绍 | 0 |
| 1. W5182 接触式 EMV 读卡器主要功能 | 0 |
| 2. W5182 接触式 EMV 读卡器主要技术指标 | 0 |
| 3. 配套软件 | 1 |
| 4. 符合标准 | 1 |
| 5. 产品型号 | 1 |
| 二、读写器通讯协议 | 1 |
| 1. 发送到读写器的命令格式: | 2 |
| 2. 从读写器返回信息的格式 | 2 |
| 3. 给读写器上电的命令格式: | 2 |
| 4. 上电后从读写器返回信息的格式: | 3 |
| 5. 给读写器下电的命令格式: | 3 |
| 6. 下电后从读写器返回信息的格式: | 3 |
| 三、W5182 接触式 EMV 读卡器操作函数说明 | 4 |
| 1. 与终端建立连接的函数: | 4 |
| 2. 与终端断开连接的函数: | 4 |
| 3. 向终端发送 APDU 指令的函数: | 4 |
| 4. 为访问智能卡数据库建立描述表 | 5 |
| 5. 关闭已经建立的描述表 | 5 |
| 6. 获取读卡器列表 | 5 |
| 7. 分配内存 | 5 |
| 8. 释放内存 | 5 |

一、读写器功能和性能介绍

W5182接触式EMV读卡器系列是握奇公司的新一代产品，它支持符合ISO/IEC 7816协议的接触式智能卡，功能强大，通用性强。W5182的一大特点是：它支持三种封装的卡片：标准大卡、SIM卡以及半卡，使用同一个接口进行通信。这使得用户不用更换应用程序即可对三种封装的卡进行操作，非常方便。

注：一次操作只允许插入一种封装的卡片！

1. W5182接触式EMV读卡器主要功能

- 支持 ISO/IEC 7816 系列接触式智能卡
- 符合 CCID 协议
- 符合 EMV2000 标准
- 支持全速 USB 通讯，在 Vista/WIN7/WIN8/WIN10 下无需安装驱动
- LED 指示灯，指示电源或通讯状态
- 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

2. W5182接触式EMV读卡器主要技术指标

| 参数 | 指标 |
|-------------|--|
| 取电方式 | USB 自取电 |
| 支持全速 USB 接口 | USB2.0 全速 12Mbps |
| 支持 CCID 协议 | CCID 协议 |
| 支持 EMV 标准 | EMV2000 标准 |
| 通讯速率 | 默认 9600bps，支持自动 PPS（9600-224Kbps） |
| 支持操作系统 | Windows2000/XP/Vista/WIN7/WIN8/WIN10 /Linux/Android4.0 以上/IOS |

| | |
|--------------|-----------------|
| 工作电流 | ≤300mA |
| 工作电压 | DC 5V |
| 外型尺寸 (H×W×D) | 101*64*19 (mm) |
| 工作温度 | 0℃ ~ 50℃ |
| 工作湿度 | 20% ~ 90% |
| 平均无故障时间 | 5000 小时 |
| API 函数 | PC/SC 规范 |
| LED 指示灯 | 双色指示灯，指示电源和操作状态 |

3. 配套软件

DEMO 软件；

4. 符合标准

- ISO/IEC 7816-1/2/3
- USB2.0 标准
- EMV2000 标准

5. 产品型号

W5182接触式EMV读卡器，支持接触卡，三种封装形式：标准大卡、SIM卡，以及半卡。

二. 读写器通讯协议

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

就是将 APDU 命令在头尾各增加相应的数据，以保证通信数据符合 CCID 协议。

特别说明：本手册里与命令相关的数字默认为十六进制。

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 | Type | 1 | CCID 指令 |
| | Length | 4 | Abdata 的长度 |
| | Slot | 1 | 卡槽号 |
| | Bseq | 1 | 结果号 |
| | bBwi | 1 | 块等待时间 |
| | Level Param | 2 | 选择通讯方式 |
| 指令 | Abdata | 1 | 发送给 CCID 的数据 |

例 1：CPU 卡取随机数命令

```

6f 05000000 00 f1 00 0000 0084000008
  ↓         ↓         ↓   ↓   ↓   ↓         ↓
  Type Length  Slot Bseq bBwi Level  Abdata
    
```

2. 从读写器返回信息的格式

| 信息域 | 标识 | 字节长度 | 含义 |
|-------|-------------|------|------------|
| 通信数据头 | Type | 1 | CCID 指令 |
| | Length | 4 | Abdata 的长度 |
| | Slot | 1 | 卡槽号 |
| | Bseq | 1 | 结果号 |
| | Bstatus | 1 | 卡槽状态 |
| | bError | 1 | 卡槽错误信息 |
| | BchainParam | 1 | 依据通讯方式返回参数 |
| 指令 | Abdata | 1 | 从读卡器返回的数据 |

例：取随机数返回信息如下：

```

80 0a000000 00 1b 00 00 00 c3f5bae6e9487cd99000
  ↓         ↓   ↓   ↓   ↓   ↓   ↓         ↓
  Type  Len  Slot Bseq Bstatus bError BchainParam  Abdata
    
```

3. 给读写器上电的命令格式：

| 信息域 | 标识 | 字节长度 | 含义 |
|------|--------|------|-----------------|
| 通信数据 | Type | 1 | CCID 指令 |
| | Length | 4 | 长度，默认 00000000h |

| | | | |
|----|--------------|---|------|
| 头 | Slot | 1 | 卡槽号 |
| | Bseq | 1 | 结果号 |
| | Power Select | 1 | 电压选择 |
| 指令 | AbRFU | 2 | 保留 |

例：上电命令

```

62 00000000 00 02 01 0000
  ↓      ↓      ↓  ↓      ↓      ↓
  Type Length Slot Bseq Power Select AbRFU

```

4. 上电后从读写器返回信息的格式：

| 信息域 | 标识 | 字节长度 | 含义 |
|-------|-------------|------|------------|
| 通信数据头 | Type | 1 | CCID 指令 |
| | Length | 4 | Abdata 的长度 |
| | Slot | 1 | 卡槽号 |
| | Bseq | 1 | 结果号 |
| | Bstatus | 1 | 卡槽状态 |
| | bError | 1 | 卡槽错误信息 |
| | BchainParam | 1 | 依据通讯方式返回参数 |
| 指令 | Abdata | 1 | 从读卡器返回的数据 |

例：上电后返回信息如下：

```

80 11000000 00 02 01 00 00 3b6d000057443778878693011edf010a1a
  ↓      ↓      ↓  ↓      ↓      ↓      ↓      ↓
  Type Len Slot Bseq Bstatus bError BchainParam Abdata

```

5. 给读写器下电的命令格式：

| 信息域 | 标识 | 字节长度 | 含义 |
|-------|--------|------|-----------------|
| 通信数据头 | Type | 1 | CCID 指令 |
| | Length | 4 | 长度，默认 00000000h |
| | Slot | 1 | 卡槽号 |
| | Bseq | 1 | 结果号 |
| 指令 | AbRFU | 2 | 保留 |

例：下电命令

```

63 00000000 00 01 000000
  ↓      ↓      ↓  ↓      ↓
  Type Length Slot Bseq AbRFU

```

6. 下电后从读写器返回信息的格式：

| 信息域 | 标识 | 字节长度 | 含义 |
|-------|--------------|------|------------|
| 通信数据头 | Type | 1 | CCID 指令 |
| | Length | 4 | Abdata 的长度 |
| | Slot | 1 | 卡槽号 |
| | Bseq | 1 | 结果号 |
| | Bstatus | 1 | 卡槽状态 |
| | bError | 1 | 卡槽错误信息 |
| | BClockstatus | 1 | 时钟运行状态 |

例：下电后返回信息如下

```

81 00000000 00 01 01 00 00
↓ ↓ ↓ ↓ ↓ ↓ ↓
Type Len Slot Bseq Bstatus bError BClockstatus

```

三. W5182接触式EMV读卡器操作函数说明

适应操作系统：

- Windows 2000/XP/Vista 系统

适用的 IC 卡：

- ISO7816 协议的 CPU 卡（包括 T=0 和 T=1）

函数动态库名称：

- Wincard.DLL（Windows 自带的 API 函数，以下是主要的几个函数介绍）

1. 与终端建立连接的函数：

hContext=hSC;

```

IReturn = SCardConnect(hContext,
                        Readers,
                        SCARD_SHARE_SHARED,
                        SCARD_PROTOCOL_T0|SCARD_PROTOCOL_T1,//SCARD_PROTOCOL_DEFAULT,
                        &hCardHandle[index],
                        &dwAP[index]);

```

2. 与终端断开连接的函数：

```

IReturn = SCardDisconnect(hCardHandle[index],
                          SCARD_UNPOWER_CARD);

```

3. 向终端发送 APDU 指令的函数：

```

IReturn = SCardTransmit(hCardHandle[index],
                        (dwAP[index]==SCARD_PROTOCOL_T0?SCARD_PCI_T0:SCARD_PCI_T1),
                        inBuf,

```



```
inBufLen,  
NULL,  
outBuf,  
&dwstatusLength );
```

4. 为访问智能卡数据库建立描述表

```
IReturn = SCardEstablishContext(SCARD_SCOPE_SYSTEM,  
                                NULL,  
                                NULL,  
                                &hSC);
```

5. 关闭已经建立的描述表

```
IReturn = SCardReleaseContext(hSC);
```

6. 获取读卡器列表

```
IReturn = SCardListReaders(hSC, NULL,(LPTSTR)&pmszReaders, &cch );
```

7. 分配内存

```
IReturn = SCardGetAttrib(hCardHandle[index],  
                        SCARD_ATTR_CHANNEL_ID,  
                        (LPBYTE)&pbAttr,  
                        &cByte);
```

8. 释放内存

```
IReturn = SCardFreeMemory( hContext, pbAttr );
```