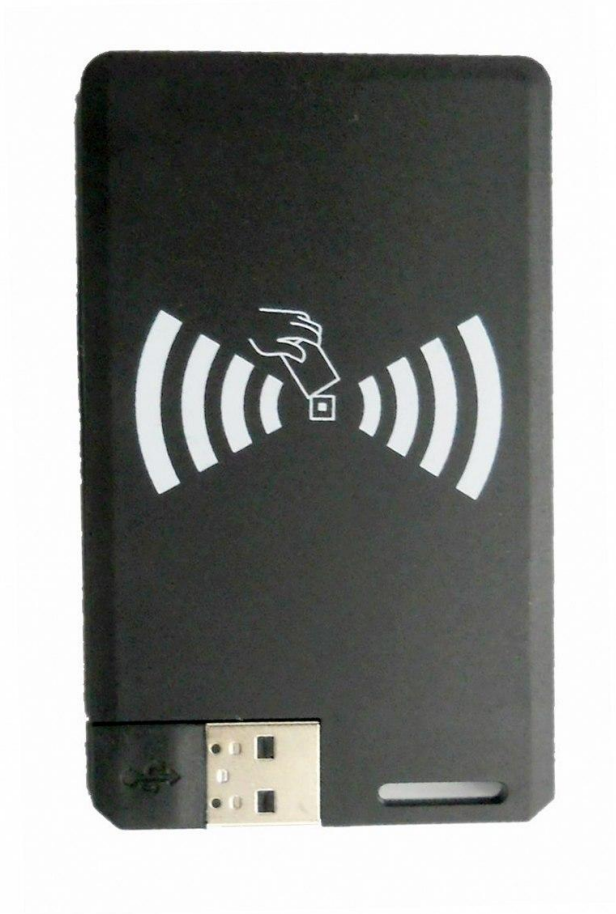
W2386 读写器

用户手册

(V1.1)



北京握奇数据系统有限公司

二〇一一年十二月

修订记录

时间	版本	修订内容
2009-9	1.0	初稿
2012-5-11	1.1	增加产品图片

目录

一. 读卡器功能和性能介绍

W2386 桌面读卡器是我公司为城市通卡客户开发的个人非接触式读卡器，产品可便于北京市政公交一卡通持卡人及其他城市通持卡人在互联网终端轻松实现余额查询、自动充值、小额支付、小额网购等服务，使用户充分体会到越来越便利、轻松的城市一卡通服务。

1. 主要功能

- 支持符合 ISO/IEC 14443 Type A 的非接触式智能卡
- 支持 Mifare one 系列非接触式存储卡
- 上位机的通讯接口为标准 USB 接口，采用的协议为 HID
- 内透式高亮 LED 指示灯，指示电源状态及通信状态
- MCU 芯片内具有 RSA1024 及 3DES 硬算法，可存储证书及密钥，保证交易安全。
- 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. 符合标准

- ISO/IEC 14443-1/2/3/4 Type A 标准
- 符合 HID 协议
- GB/T 18239-2000:集成电路(IC)卡读写机通用规范。
- 符合 USB2.0 标准

二. 主要技术指标

参数	指标
CPU	采用同方 8 位 8051 架构智能卡芯片，主频 40MHz
支持非接触卡	CPU 卡：符合 ISO/IEC 14443 Type A 卡片 其他：Mifare one 系列非接触式存储卡（S50/S70）
非接触卡最大读卡距离	CPU 卡(TYPE A) 2.5 cm Mifare 卡 5.0 cm Simpass 卡 2.5 cm
通讯协议	符合 HID 的通讯协议
安全	采用 KEY 安全芯片硬件加密，确保交易数据安全
接口	采用 USB2.0,全速接口，可支持 12Mbps 速率。
电源	工作电压：DC5V 工作电流：≤150mA
工艺	采用先进的 SMT 贴片机和回流焊机，确保焊接可靠，故障率低； 选用器件和焊锡等辅材确保无铅，符合 RoHS 规范
非接触卡通讯速率	支持 106Kbps
外型尺寸（H×W×D）	85.4*52*5.2（mm）
工作温度	0℃-40℃
工作湿度	-20℃ - +70℃

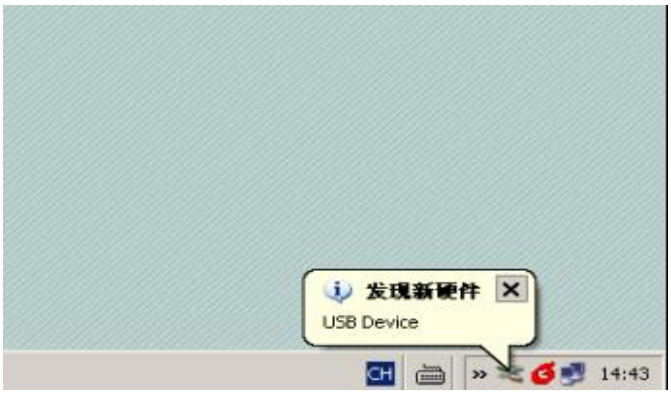
4. 配套软件

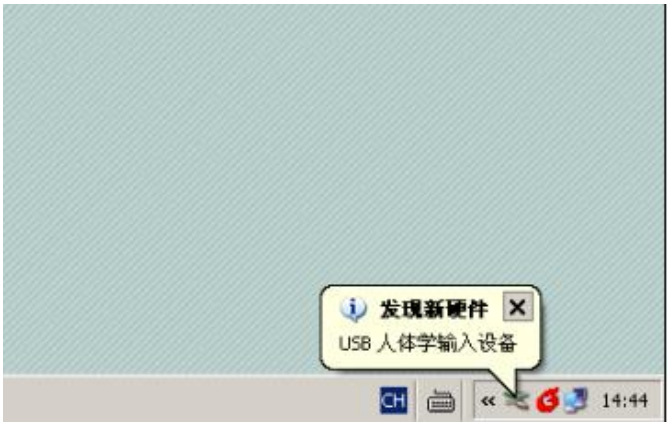
- VC++6.0 的演示源程序
- TimeCOS 2.9.1 用户工具

三.安装说明

1.W2386 读写器的连接

本产品无需用户安装驱动程序，只需将W238读卡器USB接口与计算机USB接口相连接后，计算机右下角出现发现新硬件提示，先后出现“USB Device”及“USB人体学输入设备”提示，如下图：





接着显示新硬件已经安装并可以使用，读卡器连接及枚举完成，即可正常使用了。

打开计算机设备管理器，会显示“人体学输入设备”，点其‘+’会显示“HID-compliant device”及“USB人体学输入设备”信息，证明枚举正常。

四.对读写器设备命令介绍

Mifare One 卡自定义命令：

CLA	INS	P1	P2	LC	DATA	功能说明
80	11	09	00	00	无	对Mifare卡进行复位
80	11	02	00	08	DATA	命令说明：验证块密码 data 数据说明： Mode:1 Byte 认证方式 Block:1 Byte 块地址 Password: 6 Byte 块密码 例如： 8011 0200 08 0018FFFFFFFFFFFF
80	11	03	00	01	DATA	命令说明：读块数据 data数据说明： Block:1 Byte 块地址 例如： 8011 0300 01 18
80	11	04	00	11	DATA	命令说明：向块写数据 data数据说明： block:1 Byte 块地址 Data:16 Byte 向块内写入的数据 例如： 8011 0400 11 1811223344556677889900AABBCCDDEEFF
80	11	05	00	05	DATA	命令说明：钱包初始化 data数据说明： Block: 1Byte 块地址 Money:4 Byte 初始化钱包金额 例如： 8011 0500 05 1810000000
80	11	06	00	05	DATA	命令说明：存款 data数据说明：

						Block: 1 Byte 块地址 Money: 4 Byte存款金额 例如： 8011 0600 05 1810000000
80	11	07	00	05	DATA	命令说明：扣款 data数据说明： Block: 1 Byte 块地址 Money: 4 Byte 扣款金额 例如： 8011 0700 05 1801000000
80	11	08	00	00	……	去活Mifare One卡，使其进入HALT状态，无指令数据
80	11	03	00	01	BlockAdr	命令说明：读取所在扇区A密钥 BlockAdr：该扇区密钥所在块地址，每个扇区密钥在最末块存储，如扇区0的A密钥存储在03块 例如： 8011 0300 01 03 读取0扇区A密钥
80	11	04	00	11	BlockAdr+KAY A+Access Bits+FFFFFFFFFFFF	命令说明：修改所在扇区A密钥 BlockAdr：该扇区密钥所在块地址，同上 KAY A+Access Bits+ FFFFFFFFFFFF KAY A:6 bytes 新KAY A密钥 Access Bits:状态字(参考Mifare1数据手册) 例如： 8011040011 03 111111111111 FF078069ffffffffffff 将0扇区A密钥修改为111111111111

ESAM 模块支持指令：（指令详细注释请见附表一）

编号	命令名称	CLA	INS	功能描述	是否支持
1	Append Record	00/04	E2	增加记录	是
2	Verify PIN	00/04	20	验证口令	是
3	Verify&Change PIN	84	24	验证并修改口令	是
4	External Authentication	00	82	外部认证	是
5	Get Challenge	00	84	取随机数	是
6	Internal Authentication	00	88	内部认证	是
7	Select File	00	A4	选择文件	是
8	Read Binary	00/04	B0	读二进制文件	是
9	Read Record	00/04	B2	读记录文件	是
10	Get Response	00	C0	取响应数据	是
11	Update Binary	00/04	D6	写二进制文件	是
12	Update Record	00/04	DC	写记录文件	是
13	Erase MF	80	0E	P1P2=‘00 00’，擦除MF	是
14	Erase EF/DF	00	E4	擦除DF/EF P1P2为某一DF/EF的File ID	是
15	Signatures Verify	80	C4	签名认证	是
16	Data Encrypt	80	C6	数据加密	是
17	Data Decrypt	80	C8	数据解密	是
18	Compress	80	CC	数据压缩（SHA-1）	是
19	Compress	80	7B	数据压缩（MD5）	是
20	Generate RSA Key	80	CE	生成RSA密钥对	是
21	Write Key	80/84	D4	增加或修改密钥	是
22	Create File	80	E0	建立文件	是
23	restart	00	12	Key复位	是
24	Erase SK File	80	7A	擦空私钥文件内容	是
25	Digital Signatures	80	C2	签名	是
26	SetserNo	80	7C	序列号设置/读取指令	是

五.读写器的通讯协议

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。

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 12 指令发送给读卡器 13 指令发送给 ESAM 模块 15 指令发送给非接触卡
	PCB	1	需设为 00
	LEN	1	数据长度，包括 CLA INS P1 P2 Lc DATA
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 卡选择主文件（3F00）命令

15	00	07	00	A4	00	00	02	3F	00
↓	↓	↓	↓	↓	↓	↓	↓	↓	↓
NAD	PCB	LEN	CLA	INS	P1	P2	Lc	(D	A T A)

例 2：CPU 卡复位命令

15	00	05	00	12	00	00	00
↓	↓	↓	↓	↓	↓	↓	↓
NAD	PCB	LEN	CLA	INS	P1	P2	Lc

2.从读写器返回信息的格式

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=15，返回 NAD=51
PCB	1	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：

51	00	02	61	XX
↓	↓	↓	↓	↓
NAD	PCB	LEN	SW1	SW2

例 2 返回信息如下

```
51    00    13    3B6D0000574446224A864341301F131C12    90    00
↓     ↓     ↓     ↓                  ↓                 ↓     ↓     ↓
NAD   PCB   LEN   (←-------DATA-------→)               SW1   SW2
```

3 . 动态库的处理

为了方便用户的使用，读写器发送指令和接收信息格式中的 NAD PCB LEN 这几个字节由提供的动态库进行了添加处理，用户只需根据所操作的正确设置 NAD，然后仅发送相应的 APDU 指令即可。即可获得 DATA 和 SW（SW1+SW2）返回信息。如无特殊说明，本手册说明发送指令和返回信息时，不涉及 NAD PCB LEN 这些字节。

返回状态值说明：

SW1SW2：

1.0x9000：正确

2.0x6200：无卡

3.0x6ff0：通信失败，或其他未知错误

4.0x6700：数据长度错误

5.0x6d00：不支持的命令

6.0x6901：按键取消

7. 0x6902：按键超时

8.0x63CX：密码校验失败，还可重试的次数。X，表示还剩余的可试次数

9.0x6982：安全状态不满足

10.0x6985：该地址数据已写保护，不能更新。或磁条卡卡通道使用条件不满足

11.0x6a80：写入数据与该地址已有数据不一致，不能对该地址写保护

12.0x6a86：参数 P1,P2 错误

13. 0x6983：锁死

14. 0x6204：卡片处于下电状态

15. 0x6205：卡片上电失败

六.指示灯说明

W2386 非接触读写器正面有一个 LED 指示灯，接入电源后，灯会亮。使用卡片时，把卡片放于读写器正面外壳上读卡区域。使用过程中，灯的指示说明如下：

- 无灯亮，表示设备未上电
- 灯亮，表示设备上电
- 灯闪烁，表示读写器正在进行数据传输，此时请不要将卡片移出读卡器的感应区

七.W2386 非接触读写器动态库函数

适应操作系统：

- Windows 2K/XP/2003/Vista/7 系统

适用的 IC 卡：

- 符合 14443 协议的 CPU 卡(包括 TYPE A)

1、读写器及 CPU 卡读写接口函数

此部分是针对 CPU 卡的操作函数，推荐的函数调用顺序

CT_open　　打开设备获得设备句柄
ICC_set_NAD　　设置卡座 NAD，默认值为 00
ICC_reset　　对 IC 卡复位
…
ICC_tsi_api　　对 IC 卡或读写器进行操作
...
CT_close　　关闭打开设备的连接

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:

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

示例：

```
//以下的程序为打开无驱读写器
HANDLE fDev;              //定义句柄，用于保存端口句柄
char devName[5];          //用于保存端口名称
strcpy(devName, " hid1");      //获得端口名称
```

```
fDev = CT_open(devName ,1, 0); //以相应的格式打开端口，并得到端口句柄
if(fDev == INVALID_HANDLE_VALUE)      //判断端口是否正确打开
{
  MessageBox("Open Device Error!");
  return;
 }
```

2）关闭与 IC 卡读写器相连的端口（必须用 CT_open 打开的端口）

```
int WINAPI CT_close(
    HANDLE fd
    );
```

参数：

fd ： 为函数 CT_open 所返回的句柄

返回值：

-1：失败 0：成功

示例：

```
//以下示例为关闭以 CT_open()函数打开的读写器
int ret;      //定义 int 变量用于保存关闭函数返回值
ret = CT_close( fDev );       //关闭端口，并获得结果
if(ret == -1)      //判断端口是否正确关闭
{
  MessageBox("Close Deivce Error");
}
```

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

0x6FF0 卡通讯失败或其它未知的错误

0xFFFF 通迅失败

说明：

此函数不受 ICC_set_NAD 函数影响，默认的 NAD 为 0x13

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
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）设置 CPU 卡读写地址 NAD

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

参数：

fd ： 已打开的端口描述符

nad ： 读写地址

返回值：

无

说明：

13 选择 ESAM 卡

15 选择非接触卡座

示例：

```
见函数5）中的以下语句
ICC_set_NAD(fDev,0x13);    //设置NAD为0x13
```

5）从外设向 CPU 卡或读写器发送 APDU 命令并接收应答

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

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

示例：

```
//以下示例为发送取版本号指令并显示
unsigned char lens;        //定义发送的数据长度变量
unsigned char lenr;        //定义保存返回数据长度变量
unsigned char comm[300];   //定义发送指令数组
unsigned char resp[300];       //定义接收数据数组
unsigned int sw;           //定义变量用于保存返回状态值
char tmpbuf[300];      //定义变量用于保存转化为字符型值的返回值
CString strDisplay;        //定义变量用于显示
ICC_set_NAD(fDev,0x13);    //设置NAD为0x13
memcpy(comm,"\x00\x19\x00\x00\x00",5);     //设置十六进制的指令
lens=5;                //设置指令长度为5
sw=ICC_tsi_api(fDev,lens,comm,&lenr,resp);     //发送指令并取得返回值
if(sw!=0x9000)     //对指令执行是否成功进行判断
{
  MessageBox("Get Reader Firmware Version Error!");
}else
{
  BinToCHex((unsigned char *)tmpbuf,resp,lenr);        //将返回值转换为字符以供显示
  tmpbuf[lenr*2]=0;
```

```
        strDisplay=CString(tmpbuf);
        MessageBox("Firmware Version is "+strDisplay);
    }
```

7）检查读写器的主卡座否插入 IC 卡

```
unsigned WINAPI ICC_present(
    HANDLE fd
    );
```

参数：

fd ： 已打开的端口描述符

返回值：

0x9000 已插入 IC 卡

0x6200 没有插入卡或卡没插到位

示例：

```
//此例程为检查卡片是否存在
unsigned int sw;          //定义返回状态变量
sw = ICC_present(fDev); //检查是否插入卡操作
if(sw != 0x9000)          //判断是否执行成功
{
    … //操作失败后，用户的处理
}
```

8）写 CPU 卡的二进制文件

```
unsigned WINAPI ICC_write_file(
    HANDLE fd,
    unsigned int offset,
    unsigned int len,
    unsigned char *data
    );
```

参数：

fd ： 已打开的端口描述符

offset ： 二进制文件的偏移量

len ： 要写入卡上的数据长度

data ： 要写入卡上的数据

返回值：

0xFFFF 通迅失败（发送命令或接收返回的数据失败）

0x6FF0 卡通讯失败或其它未知的错误

其它为从卡上返回的状态 SW1 SW2

说明：

该函数适用于所有 CPU 卡操作，用户必须先选择要操作的二进制文件

示例：

```
//此示例为向 CPU 卡的二进制文件写入 3 个数据，请先对 CPU 卡片复位后，选择相应二进制文件
unsigned int offset = 0;          //定义写入数据地址的偏移量变量，并赋初值
unsigned int len;              //定义写入数据长度的变量
unsigned char data[3]={0, 1, 2};     //定义写入数据变量，并赋初值
len = 3;               //写入数据的长度为 3 字节
sw = ICC_write_file(fDev, offset, len, data);    //写入二进制文件 3 字节数据操作
if(sw != 0x9000)          //判断是否执行成功
{
    … //操作失败后，用户的处理
```

```
}
```

9）读 CPU 卡的二进制文件

```
unsigned WINAPI ICC_read_file(
    HANDLE fd,
    unsigned int offset,
    unsigned int len,
    unsigned char *data
    );
```

参数：

```
    fd      :   已打开的端口描述符
    offset  :   二进制文件的偏移量
    len     :   要读卡上的数据长度
    data    :   要读卡上的数据
```

返回值：

```
    0xFFFF  通迅失败（发送命令或接收返回的数据失败）
    0x6FF0  卡通讯失败或其它未知的错误
    其它为从卡上返回的状态 SW1 SW2
```

说明：

```
    该函数适用于所有 CPU 卡操作，用户必须先选择要操作的二进制文件
```

示例：

```
    //此示例为向 CPU 卡的二进制文件读取 3 个数据，请先对 CPU 卡片复位后，选择相应二进制文件
    unsigned int offset = 0;        //定义读取数据地址的偏移量变量，并赋初值
    unsigned int len;           //定义读取数据长度的变量
    unsigned char data[256];    //定义读取数据变量
    len = 3;            //读取数据的长度为 3 字节
    sw = ICC_read_file (fDev, offset, len, data);   //读取二进制文件 3 字节数据操作
    if(sw != 0x9000)        //判断是否执行成功
    {
        … //操作失败后，用户的处理
    }

    10)Mifare 1 卡复位
    unsigned WINAPI Mifare1_Reset(
        HANDLE fd,
        unsigned char *len,
        unsigned char *resp
        );
     参数:
     fd: 已打开的端口描述符
     len: 返回数据的长度
     resp: 返回的数据
```

示例：

```
    unsigned char resp[128];
    unsigned char  m_temp[128];
    unsigned char len=0;
    unsigned res=0;
    res=Mifare1_Reset(hDevice,&len,resp);
```

2、Mifare One 专用函数

1）Mifare1 密钥认证

```
unsigned WINAPI Mifare1_Verify(
  HANDLE        fd,
  BOOL          bAuthType,
  unsigned int  BlockNum,
  BOOL          bKeyType,
  unsigned char *uKey
  );
```

参数:

fd: 已打开的端口描述符

bAuthType: 认证类型。1, 利用指令中给定的密钥进行认证;
0, 利用 RC531 存储密钥区（EEPROM）中对应的密钥进行认证

BlockNum: 待认证的块号(0-0x80)

bKeyType: 密钥类型: 0, KeyA; 1, KeyB

uKey: 6 字节的密钥值(只在 bAuthType = 1 时有效)

示例：

```
unsigned int res=0;
unsigned char uKey[8];
bool bKeyType=0;
unsigned bAuthType=0;
unsigned int  BlockNum=4;
unsigned int uAmount=9;
memset(uKey,0,sizeof(uKey));
memcpy(uKey,"\xff\xff\xff\xff\xff\xff",6);
BlockNum=4;
res2=Mifare1_Verify(hDevice,bAuthType,BlockNum,bKeyType,uKey);
```

2） 读 Mifare 1 卡

```
unsigned WINAPI Mifare1_Read(
  HANDLE        fd,
  unsigned int  BlockNum,
  unsigned char *resp
```

参数:

fd : 已打开的端口描述符

BlockNum: 欲读的已经过认证的块号(0-0x80)

resp : 读出的数据

示例：

```
unsigned int res=0;
unsigned int  BlockNum=4;
unsigned char resp[128];
res=Mifare1_Read(hDevice,BlockNum,resp);
```

3）写 Mifare 1 卡

```
unsigned WINAPI Mifare1_Write(
  HANDLE        fd,
  unsigned int  BlockNum,
  unsigned char *uData
 );
```

参数:

fd: 已打开的端口描述符

BlockNum: 欲写入的已经过认证的块号(0-0x80)

uData : 欲写入的数据（16 字节）

示例：

```
unsigned res=0;
unsigned int  BlockNum=4;
unsigned char uData[128];
memcpy(uData ,"\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF\xFF",16);
res=Mifare1_Write(hDevice,BlockNum, uData);
```

4） 将 Mifare 1 卡的块初始化为其规定的钱包形式。

```
unsigned WINAPI Mifare1_InitAmount(
   HANDLE        fd,
   unsigned int  BlockNum,
   ULONG         uAmount
);
```

参数:

fd : 已打开的端口描述符

BlockNum: 经过认证的块号(0-0x80)

uAmount : 初始金额

示例：

```
unsigned res=0;
unsigned int  BlockNum=4;
unsigned long uAmount=9999;
res=Mifare1_InitAmount(hDevice,BlockNum,uAmount);
if(res==0x9000)
 {
   …
 }
```

5)向 mifare 1 卡的钱包块内添加金额。

```
unsigned WINAPI Mifare1_IncreaseAmount(
   HANDLE        fd,
   unsigned int  BlockNum,
   ULONG         uAmount
   );
```

参数:

fd : 已打开的端口描述符

BlockNum: 经过认证的块号(0-0x80)

uAmount : 欲添加的金额

示例：

```
unsigned res=0;
unsigned int  BlockNum=4;
unsigned long uAmount=1;
res= Mifare1_IncreaseAmount(hDevice,BlockNum,uAmount);
if(res==0x9000)
 {
   …
 }
```

6） 在 Mifare 1 卡的钱包块中扣除给定的金额。

```
unsigned WINAPI Mifare1_DecreaseAmount(
   HANDLE        fd,
   unsigned int  BlockNum,
   ULONG         uAmount
  );
```

参数:

fd : 已打开的端口描述符

BlockNum: 经过认证的块号(0-0x80)

uAmount : 欲扣除的金额

示例：

```
unsigned res=0;
unsigned int  BlockNum=4;
unsigned long uAmount=1;
res= Mifare1_DecreaseAmount(hDevice,BlockNum,uAmount);
if(res==0x9000)
{
  …
}
```

3、常用算法及辅助函数

1）十六进制字符串转化为二进制数

```
unsigned char * WINAPI CHexToBin(
    unsigned char *bin,
    unsigned char *asc,
    unsigned int len
    );
```

参数：

bin : 二进制结果串：0x12，0x34，0xE1，0xFA

asc : 十六进制字符串，如“1234E1FA”

len : 十六进制字符串长度

返回值：

二进制结果串的指针

2）二进制数转化为十六进制字符串

```
unsigned char * WINAPI BinToCHex(
    unsigned char *asc,
    unsigned char *bin,
    unsigned int len
    );
```

参数：

asc : 十六进制字符串，如“1234E1FA”

bin : 二进制结果串：0x12，0x34，0xE1，0xFA

len : 二进制串长

返回值：

十六进制字符串的指针

3）哈什摘要

```
void WINAPI SHA1(
```

```
    unsigned char *sour,
    unsigned len,
    unsigned char *digest
    );
```

参数：

```
    len     :   要压缩的原文长度
    sour    :   要压缩的原文
    digest  :   20字节的摘要数据
```

4 ）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
    SourDsata   :   源数据
    DestData    :   目标数据
```

返回值：

目标数据的长度

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

5 ）3DES 加密

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

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
```

返回值：

目标数据的长度

说明：

加密时，当明文长度不是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)))

6）DES 认证码

unsigned WINAPI SingleMAC(
unsigned char * SingleMACKey,
unsigned char *InitData,
unsigned int SourDataLen,
unsigned char *SourData,
unsigned char *MACData
);

参数：

SingleMACKey： 8字节密钥
InitData ： 8字节的初始值
SourDataLen ： 用来产生mac码的原文长度
SourData ： 用来产生mac码的原文
MactData ： 计算出的认证码
mac码的计算方法参见有关标准

返回值：

认证码的长度为8

7）3DES 认证码

unsigned WINAPI TripleMAC(
unsigned char * SingleMACKey,
unsigned char *InitData,
unsigned int SourDataLen,
unsigned char *SourData,
unsigned char *MACData
);

参数：

SingleMACKey： 16字节密钥
InitData ： 8字节的初始值
SourDataLen ： 用来产生mac码的原文长度
SourData ： 用来产生mac码的原文
MactData： 计算出的认证码

返回值：

认证码的长度

八.注意事项

1） 因为读卡器工作频率为 13.56MHz，所以在读卡器安装现场不得有 13MHz～15MHz 之间强电磁场
2）为了防止读卡器发射磁场的相互影响，2 台读卡器的安装距离应大于 10CM。
3） 金属平面对电磁波有反射和屏蔽作用，因此读卡器周围应尽量避免放置或安装在金属平面上。

附表一　ESAM模块支持的指令详解

1.Append Record(增加记录）

定义与范围

Append Record 命令用于对变长记录文件、循环文件追加记录。

注意事项

- Append Record命令适用于变长记录文件和循环文件。
- 访问记录文件的命令如下：
  - 建立文件（Create File）
  - 选择文件（Select File）
  - 读记录文件（Read Record）
  - 写记录文件（Update Record）
  - 增加记录（Append Record）
- 只有满足记录文件写权限时才能执行此命令。
- 若循环文件记录已满则覆盖最早写入的记录，且新增加记录的记录号总为1。

命令报文

表 1.1 Append Record 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00/04	–
INS	1	E2	–
P1	1	00	–
P2	1	XX	见说明
Lc	1	XX	–
DATA	XX	XX...XX	写入的数据
Le	-	-	不存在

说明：

- 参数 P2 的含义：

b8	b7	b6	b5	b4	b3	b2	b1	含义
X	X	X	X	X	0	0	0	b4-b8 为短文件标识符
0	0	0	0	0	0	0	0	当前文件

- Lc 表示要写入的字节数。

1. 若为线路保护，Lc 为写入数据的长度+4 字节 MAC。
2. 若为加密线路保护，Lc 为加密后数据的长度+4 字节 MAC。

命令报文数据域

命令报文数据域由追加记录组成。
若为线路保护则由追加记录附上 4 字节 MAC 码组成。
若为线路加密保护则由被加过密的记录数据附上 4 字节 MAC 码组成。
用维护密钥加密数据和计算 MAC，方法见“4.安全报文传送”。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下：

表 1.2Append Record 命令响应状态码

SW1	SW2	意义
90	00	正确执行
65	81	写 EEPROM 失败
67	00	长度错误(Lc 域为空)
69	81	当前文件不是循环文件或变长记录文件
69	82	不满足安全状态
69	84	没有取随机数
69	86	没有选择当前可操作的文件
6A	81	不支持此功能（无 MF 或 MF 已锁定）
6A	82	未找到文件
6A	83	未找到记录
6A	84	文件中存储空间不够（对变长记录文件）

应用举例

[1] 条件：文件类型：变长记录文件；
文件标识符=0001；

建立时不采用线路保护。

操作：往变长记录文件中增加 1 条记录标识为 AA 的记录，不进行线路保护。

命令：00 E2 00 08 0E AA 0C 11 22 33 44 55 66 77 88 99 AA BB CC

响应：9000

[2] 条件：文件类型：循环文件；
文件标识符=0001；
记录数=02；
记录长度=06；
建立时不采用线路保护；
设该文件为当前文件。

操作：往循环文件中追加 1 条记录，不进行线路保护

命令：00 E2 00 00 06 11 22 33 44 55 66

响应：9000

2.External Authentication（外部认证）

定义与范围

External Authentication 命令要求 IC 卡中的应用验证密码。

注意事项

- 在满足该外部认证密钥的使用权限且该密钥未被锁死时才可执行该命令。

命令报文

表 2.1 External Authentication 命令报文编码

代码	长度 (byte)	值 (Hex)	描述 （Hex）
CLA	1	00	-
INS	1	82	-
P1	1	00	-
P2	1	XX	外部认证密钥标识号
Lc	1	8	-
DATA	8	XX...XX	8 字节加密后的随机数
Le	-	-	-

说明：

将命令中的数据用指定外部认证密钥解密，然后与先前产生的随机数进行比较，

- 若一致则表示认证通过，置安全状态寄存器为该密钥规定的后续状态值，错误计数器恢复成初始值；
- 若不一致则认证失败，可再试错误数减一，且不改变安全状态寄存器的值。

命令报文数据域

命令报文数据域包括8字节加密后的随机数。

响应报文数据域

响应报文数据不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 2.2 External Authentication 命令响应状态码

SW1	SW2	意义
90	00	正确执行
63	CX	还剩 x 次可试机会
67	00	错误的长度
69	81	不是外部认证密钥
69	82	密钥使用条件不满足
69	83	认证方法（外部认证密钥）锁死
69	84	没有取随机数
6A	82	KEY 文件未找到
93	02	安全信息不正确
94	03	密钥未找到

外部认证过程

外部认证是卡片对机具的认证，认证过程如下图所示：

终端	方向	卡片
取 8 字节随机数	⇒	卡片内部产生随机数 RND_{ICC}
	⇐	送随机数 RND_{ICC}
用与卡片认证密钥相同的密钥 Cardkey 对 RND_{ICC} 进行加密得鉴别数据 D1。即： D1=DES(Cardkey, RND_{ICC})；		
送鉴别数据 D1 作外部认证。	⇒	卡片用指定的外部认证密钥对 D1 进行解密运算，产生鉴别数据 D2，后比较 D2 和 RND_{ICC}。即： 1)D2=DES^{-1}(KID,D1) 2)D2?=RND_{ICC}
	⇐	送比较结果(即 SW1SW2)，若比较正确，

		则置安全状态寄存器值为该密钥后续状态。

说明：

1．终端从卡片取随机数 RND_{ICC}；

2．终端用相应的密钥对 RND_{ICC} 进行 DES 加密运算，产生鉴别数据 D1；

4．终端向卡片发出外部认证命令，送入 D1 到卡片内；

00 82 00 kid 08 D1

5．卡片收到 D1 后，用卡内的相应密钥对 D1 进行 DES 解密运算，产生 8 字节鉴别数据 D2；

卡片比较 RND_{ICC} 和 D2，

- 若一致则表示认证通过，置安全状态寄存器为该密钥规定的后续状态值，错误计数器恢复成初始值；
- 若不一致则认证失败，可再试错误数减一，且不改变安全状态寄存器的值。

应用举例

[1] 条件：外部密钥标识号=01；

使用权限=0xF0；

更改权限=0xEF；

错误计数器=0x33；

后续状态=01；

操作：外部认证。

[步骤 1] 取 8 字节随机数。

命令：00 84 00 00 08

响应：D3 89 BF 67 45 B9 35 50 9000

[步骤 2] 卡终端用与外部认证密钥相同的密钥‘ ’对随机数进行加密，加密后的结果为 C1 8A 5B 4B 13 40 25 21.

[步骤 3] 卡终端将加密后的随机数送到卡中作外部认证。

命令：00 82 00 00 08 C1 8A 5B 4B 13 40 25 21

说明：其中 C1 8A 5B 4B 13 40 25 21 是[步骤 2]中加密后的数据。

响应：9000

说明：成功执行后置安全状态寄存器值为该外部认证密钥的后续状态 01.

3.Get Response（取响应数据）

定义与范围

当 APDU 不能用现有协议传输时，Get Response 命令提供了一种从卡片向接口设备传送 APDU（或 APDU 的一部分）的传输方法。

注意事项

- 此命令只用于T=0通讯协议。

命令报文

表 3.1 Get Response 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00	-
INS	1	C0	-
P1	1	00	-
P2	1	00	-
Lc	-	-	不存在
DATA	-	-	不存在
Le	1	XX-	期望响应数据的长度

命令报文数据域

命令报文数据不存在。

响应报文数据域

响应报文数据的长度由 Le 的值决定。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 3.2 Get Response 命令响应状态码

SW1	SW2	意义
90	00	正确执行
62	81	回送的数据可能错误
6A	86	参数 P1P2 错误
67	00	长度错误（Le 大于卡中响应数据长度）
6F	00	卡中无数据可返回

应用举例

[1] 条件：定长记录文件标识=0007；
记录数=3；
记录长度=12。
[步骤 1]：读出记录号为 03 的记录内容；
命令：00 B2 03 04 00
响应：61 0C
说明：对于 T=0 的卡片，610C 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

[步骤 2]：取响应数据
命令 ： 00 C0 00 00 0C
响应 ：11 22 33 44 55 66 77 88 99 AA BB CC 90 00
说明：11 22 33 44 55 66 77 88 99 AA BB CC 为定长记录文件记录号 03 的记录内容；

4.Get Challenge（取随机数）

定义与范围

Get Challenge 命令请求一个用于安全相关过程（如安全报文）的随机数。

命令报文

表 4.1 Get Challenge 命令报文编码

代码	长度 (byte)	值 (Hex)	描述 （Hex）
CLA	1	00	-
INS	1	84	-
P1	1	00	-
P2	1	00	-

Lc	-	-	不存在
DATA	-	-	不存在
Le	1	04,08-10	要求卡片返回的随机数长度

命令报文数据域

命令报文数据不存在。

响应报文数据域

响应报文数据包括随机数，长度为 Le 个字节。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 4.2Get Challenge 命令响应状态码

SW1	SW2	意义
90	00	正确执行
67	00	长度错误
6A	81	不支持此功能（无 MF 或卡片已锁定）
6A	86	参数 P1P2 错误

5.Internal Authentication（内部认证）

定义与范围

Internal Authentication 命令提供了利用接口设备发来的随机数和自身存储的相关密钥进行数据认证的功能。

注意事项

- 在满足该密钥的使用条件时才能执行此命令。

命令报文

表 5.1 Inernal Authentication 命令报文编码

代码	长度 (byte)	值 (Hex)	描述 （Hex）
CLA	1	00	-

INS	1	88	-
P1	1	00	加密
		01	解密
		02	计算 MAC
P2	1	XX	DES 密钥标识号
Lc	1	XX	-
DATA	XX	XX...XX	认证数据
Le	1	00	-

说明：

- P1=00，表示进行加密运算，密钥类型是DES加密密钥
- P1=01，表示进行解密运算，密钥类型是DES解密密钥
- P1=02，表示进行MAC运算，密钥类型是DES&MAC密钥

命令报文数据域

命令报文数据域的内容是应用专用的认证数据。

响应报文数据域

响应报文数据域的内容是相关认证数据，即 DES 运算的结果。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 5.2 Internal Authentication 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
67	00	错误的长度
69	81	密钥与运算方法不匹配
69	82	不满足安全状态
69	85	不满足使用条件
69	84	没有取随机数
6A	82	KEY 文件不存在
6A	86	参数 P1P2 错误
94	03	密钥未找到

内部认证过程

内部认证是机具对卡片的认证，认证过程如下图所示：

终端	方向	卡片
产生两个 8 字节随机数 RND_{IFD}		
送 RND_{IFD} 作内部认证	⇒ ⇐	卡片用指定的 DES 加密钥对随机数 RND_{IFD} 进行 DES 加密运算，产生鉴别数据 D1。即： D1=DES（KID，RND_{IFD}） 送 D1
用与卡片 DES 加密密钥相同的密钥 Cardkey 对 RND_{IFD} 进行 DES 加密运算，产生产生鉴别数据 D2，后比较 D1 和 D2。即： 1）D2=DES(CardKey，RND_{IFD}) 2）D1？=D2		

说明：

1．终端自己产生或从 PSAM 卡申请 1 个 8 字节随机数 RND_{IFD}；

2．终端向卡片发出内部认证命令，送入 RND_{IFD} 到卡片内；

00 88 00 KID 08 RND_{IFD}

3．卡片收到 RND_{IFD} 后，用卡内的相应密钥对随机数 RND_{IFD} 进行 DES 加密运算，产生 8 字节鉴别数据 D1；

4．卡片送鉴别数据 D1 到卡外；

5．终端接收到卡片送出的鉴别数据 D1 后，用相应密钥对随机数 RND_{IFD} 进行 DES 加密运算，产生 8 字节鉴别数据 D2；

终端比较 D1 和 D2，若一致则认证通过，不一致认证失败。

应用举例

[1] 条件：密钥标识号=01；

密钥类型是 DES 加密密钥；

使用权限=0xF0；

更改权限=0xEF；

算法标识=01；

密钥版本号=01；

16 字节的密钥=‘ ’；

待加密数据=‘1122334455667788’。

操作：内部认证即 DES 加密。

命令：00 88 00 01 08 11 22 33 44 55 66 77 88

响应；6108

说明：对于 T=0 的卡片，6108 表示卡片要回送的数据长度，可以通过 Get Response

命令取返回数据。对于本公司读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 08

响应：07 CB F6 15 E7 D7 2F 96 9000

说明：07 CB F6 15 E7 D7 2F 96 是内部认证即 DES 加密的结果。

[2] 条件：密钥标识号=01；

密钥类型是 DES 解密密钥；

使用权限=0xF0；

更改权限=0xEF；

算法标识=01；

密钥版本号=01；

16 字节的密钥=‘ ’；

待解密数据=‘07CBF615E7D72F96’。

操作：内部认证即 DES 解密。

命令：00 88 01 01 08 07 CB F6 15 E7 D7 2F 96

响应；6108

说明：对于 T=0 的卡片，6108 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于本公司读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 08

响应：11 22 33 44 55 66 77 88 9000

说明：11 22 33 44 55 66 77 88 是内部认证即 DES 解密的结果。

[3] 条件：密钥标识号=01；

密钥类型是 DES&MAC 解密密钥；

使用权限=0xF0；

更改权限=0xEF；

算法标识=01；

密钥版本号=01；

16 字节的密钥=‘ ’；

待计算 MAC 数据=‘1122334455667788’。

操作：内部认证即计算 MAC。

命令：00 88 02 01 08 11 22 33 44 55 66 77 88

响应；6104

说明：对于 T=0 的卡片，6104 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于本公司读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 04

响应：87 56 E2 85 9000

说明：87 56 E2 85 是内部认证即计算 MAC 的结果。

计算 MAC 的 8 字节初始值=‘0000000000000000’。

6.Read Binary（读二进制文件）

定义与范围

Read Binary 命令用于读取二进制文件的内容（或部分内容）、公钥文件 。

注意事项

- Read Binary命令只适用于二进制文件、公钥文件。
- 访问二进制文件的命令如下：
  - 建立文件（Create File）
  - 选择文件（Select File）
  - 读二进制文件（Read Binary）/写二进制文件（Update Binary）
- 只有满足二进制文件、公钥文件的读权限时才能执行此命令。

命令报文

表 6.1 Read Binary 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00/04	-
INS	1	B0	-
P1	1	XX	见说明
P2	1	XX	见说明
Lc	-	-	CLA≠04 时，不存在 CLA=04 时，Lc 存在
DATA	-	-	CLA≠04 时，不存在 CLA=04 时，应包括 MAC
Le	1	XX	‘00’或要读取的数据长度

说明：

- 若 P1 的高三位为 100，则低 5 位为短的文件标识符，P2 为读的偏移量。

P1								P2
b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
1	0	0	短文件标识符					文件的偏移量

- 若 P1 的最高位不为 1，则 P1 P2 为欲读文件的偏移量，所读的文件为当前文件。

P1								P2
b7	b6	b5	b4	b3	b2	b1	b0	
0	文件的偏移量							

命令报文数据域

一般情况下，命令报文数据域不存在。

当使用安全报文时，命令报文数据域中应包含 MAC。

用维护密钥加密数据和计算 MAC，方法见“4.安全报文传送”。

响应报文数据域

响应报文数据域由读取的数据组成。

若为线路保护则由读取的数据附上 4 字节 MAC 组成。
若为线路加密保护则由被加过密的数据附上 4 字节 MAC 码组成。
注：文件被置成线路保护/线路加密保护时也允许明文读取，设置方法见“《TimeCOS®专用技术参考手册》之 7.1 Create File”。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 6.2Read Binary 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
62	81	回送的数据可能错误
67	00	错误的长度
69	81	不是二进制文件
69	82	读的条件不满足
69	84	没有取随机数
69	86	没有选择当前可操作的文件
6A	81	不支持此功能（无 MF 或 MF 已锁定）
6A	82	未找到文件
6B	00	参数错误（偏移地址超出了 EF）
6C	XX	错误的 Le

说明：

- 若文件校验不正确，卡将送出所读的数据，并给出警告状态 SW1 SW2=6281。若下次重写该文件，卡将重新计算校验。
- 若 Le=00 或大于文件实际长度时，则送回警告状态 6Cxx 请求将 Le 置为 xx 并重发该命令。

应用举例

[1] 条件：文件类型：二进制文件；
文件标识符=0005；
文件主体空间的大小=8 个字节。
操作：读出自偏移量 00 开始到文件结束的所有数据，不进行线路保护。
命令：00 B0 85 00 00
响应：6C08
说明：6C08 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=0x08.
命令：00 B0 85 00 08
响应：11 22 33 44 55 66 77 88 9000

7.Read Record（读记录文件）

定义与范围

Read Record 命令用于读取定长记录文件、循环文件、钱包文件和变长记录文件的内容。IC 卡的响应由回送记录组成。

注意事项

Read Record 命令适用于定长记录文件、循环文件、钱包文件和变长记录文件。

访问记录文件的命令如下所示：

建立文件（Create File）

选择文件（Select File）

读记录文件（Read Record）

写记录文件（Update Record）

只有满足记录文件读权限时才能执行此命令。

命令报文

表 7.1 Read Record 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00/04	-
INS	1	B2	-
P1	1	XX	记录号或记录标识符，见说明

P2	1	XX	见说明
Lc	-	-	CLA≠04 时，不存在 CLA=04 时，Lc 存在
DATA	-	-	CLA≠04 时，不存在 CLA=04 时，应包括 MAC
Le	1	XX	‘00’或要读取的数据长度

说明：

参数 P1 的含义:

类型	P1 的含义
定长记录文件	记录号，若该文件有 N 条记录，则记录号可以是 1~N。
变长记录文件	记录号，若该文件有 N 条记录，则记录号可以是 1~N; 记录标识，参见“TimeCOS®专用技术参考手册 3.5.6 变长记录文件”
循环文件	记录号,最新写入的记录号为 01,上 1 条记录的记录号为 02，依次类推...
钱包文件	记录号,最新写入的记录号为 01,上 1 条记录的记录号为 02，依次类推...

参数 P2 的含义

b7	b6	b5	b4	b3	b2	b1	b0	描述
0	0	0	0	0	-	-	-	对当前文件进行操作
x	x	x	x	x	-	-	-	短文件标识符
-	-	-	-	-	1	0	0	按记录号，读 P1 指定的记录
-	-	-	-	-	1	0	1	按记录号，从 P1 指定的记录读到最后一条记录
-	-	-	-	-	1	1	0	按记录号，从最后一条记录读到 P1 指定的记录
-	-	-	-	-	0	0	0	读 P1 指定记录标识符的第一个记录
-	-	-	-	-	0	0	1	读 P1 指定记录标识符的最后一个记录
-	-	-	-	-	0	1	0	读 P1 指定记录标识符的下一个记录
-	-	-	-	-	0	1	1	读 P1 指定记录标识符的上一个记录

注：X X X X X 代表短文件标识符（SFI）；- - - - - 代表全 0 或短文件标识符

命令报文数据域

命令报文数据域不存在。

响应报文数据域

响应报文数据域由读取的记录组成。

响应报文状态码

IC 卡可能回送的状态码如下所示：

附 7.2Read Record 命令响应状态码

SW1	SW2	意义
90	00	正确执行
62	81	回送的数据可能有错
62	82	记录长度小于 Le 个字节
67	00	错误的长度（Lc 域不存在）
69	81	命令与文件结构不相容
69	82	不满足安全状态
69	85	使用条件不满足
69	84	没有取随机数
69	86	没有选择当前可操作的文件
6A	81	不支持此功能（无 MF 或 MF 已锁定）
6A	82	未找到文件
6A	83	未找到记录
6A	86	参数 P1、P2 不正确
6C	XX	Le 错误

说明：若 CLA =04,Le 被忽略,并返回整条记录内容；
当 Le 不等于该记录的实际长度时,则送回警告状态 6Cxx 请求将 Le 置为 xx 并重发该命令。

应用举例

条件：文件类型：定长记录文件；
文件标识符=0001；
记录数=3 条；
记录长度=12 个字节。
建立时不采用线路保护。
操作：读出定长记录文件中记录号为 02 的记录。
命令：00 B2 02 0C 00 返回状态 6C 0C
响应：6C0C
说明：对于 T=0 的卡片，6C0C 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=12。
命令：00 B2 02 0C 0C
响应：01 02 03 04 05 06 07 08 09 0A 0B 0C 9000
说明：01 02 03 04 05 06 07 08 09 0A 0B 0C 为读出的记录号为 02 的记录的内容。

条件：文件类型：循环文件
文件标识符=0003；
记录数=3 条；
记录长度=12 个字节。
建立时不采用线路保护。
操作：读出循环文件中记录号为 01 的记录，即最新写入的记录。
命令：00 B2 01 1C 00
响应：6C0C
说明：对于 T=0 的卡片，6C0C 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=12.
命令：00 B2 01 1C 0C
响应：11 22 33 44 55 66 77 88 99 AA BB CC 9000

说明：11 22 33 44 55 66 77 88 99 AA BB CC 为读出的记录号为 01 的记录的内容。

条件：文件类型：变长记录文件
文件标识符=0007；
建立时不采用线路保护。
[操作 1]：按记录标识来读，读出变长记录文件中记录标识为 AA 的第一条记录。
命令：00 B2 AA 38 00
说明：由于按记录标识来读记录标识为 AA 的第一条记录，则 P2 的低 3 位必须为‘000’。
响应：6C03
说明：对于 T=0 的卡片，6C03 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=3.
命令：00 B2 AA 38 03
响应：AA 01 11 9000
说明：读出的是 TLV 格式的记录，AA 为记录标识，01 表示记录数据的长度，11 为 1 个字的记录数据。

[操作 2]：按记录号来读，读出变长记录文件中的第 1 条记录。
命令：00 B2 01 3C 00
说明：由于按记录号来读记录文件中的第一条记录，则 P1 为记录号时，P2 的低 3 位必须为‘100’。

响应：6C03
说明：对于 T=0 的卡片，6C03 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=3.
命令：00 B2 01 3C 03
响应：AA 01 11 9000
说明：读出的是 TLV 格式的记录，AA 为记录标识，01 表示记录数据的长度，11 为 1 个字节的记录数据。

条件：文件类型：普通钱包文件
文件标识符=0004；
记录数=2 条；
记录长度=4 个字节。
建立时不采用线路保护。
操作：读出钱包文件中记录号为 01 的记录，即最新写入的记录。
命令：00 B2 01 24 00
响应：6C04
说明：对于 T=0 的卡片，6C04 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=4.
命令：00 B2 01 24 04
响应：00 00 00 01 9000
说明：00 00 00 01 为钱包的新余额。

8. Select File（选择文件）

定义与范围

Select File 命令通过文件名、文件标识符或选择下一个应用来选择 IC 卡中 MF、DDF 或 ADF。IC 卡的响应报文应由回送文件控制信息 FCI 组成。

能够选择到父 DF，同级 DF 和下级 DF、EF 以及 MF。

注意事项

- 正确选择 MF 后，MF 安全寄存器将被复位为 0.
- 正确选择 MF 下各个 DF 后，DF 安全寄存器将被复位为 0，MF 安全寄存器的值不变。

命令报文

表 8.1 Select File 命令报文编码

代码	长度 (byte)	值 (Hex)	描述 （Hex）
CLA	1	00	-
INS	1	A4	-
P1	1	00/04	见说明
P2	1	00/02	见说明
Lc	1	XX	-
DATA	XX	XX...XX	文件标识符或 DF 名称
Le	1	XX	对于 DF 而言为卡片自动返回的 FCI 的最大长度

说明：

- P1=00，表示按文件标识符选择（P2 必须等于 0），可选择
  - 当前目录（DF）下基本文件或子目录文件。
  - 同级目录文件（DF）。
- P1=04，表示用 DF 名称选择，分如下两种情况：
  - P2=00，表示第一个或仅有一个；
  - P2=02，表示下一个。

  用此方法可以选择 DF。

在任何情况下均可通过标识符‘3F00’或目录名称1PAY.SYS.DDF01选择MF。

命令报文数据域

命令报文数据域可为空或包含文件标识符或 DF 名称。

响应报文数据域

响应报文数据域应包括所选择的 DDF 或 ADF 的文件控制信息(FCI)，如表 7.21 和表 7.22 所示。

表 8.2 成功选择 DDF 后回送的文件控制信息 FCI

标志	值	存在方式
6F	文件控制信息模板	必备
84	DF 名称	必备
A5	文件控制信息专用数据	可选

88	目录基本文件的短文件标识符	可选

表 8.3 成功选择 ADF 后回送的文件控制信息 FCI

标志	值	存在方式
6F	文件控制信息模板	必备
84	DF 名称	必备
A5	文件控制信息专用数据	可选
9F0C	发卡方自定数据的文件控制信息	可选

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 8.4 Select File 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
62	83	选择文件无效，文件或密钥校验错误
62	84	FCI 格式和 P2 不匹配
64	00	状态标志未改变
67	00	错误的长度
6A	81	不支持此功能(无 MF 或卡片已锁定)
6A	82	未找到文件
6A	86	参数 P1 P2 不正确

应用举例

（可参考 TimeCOS® PK 专用技术参考手册-“附录 3　　TimeCOS®卡应用举例”）

◆　符合银行标准的应用目录的选择

[1] 条件：MF 下目录基本文件的短文件标识符=01；

操作：对主文件 MF 进行选择即对 DDF 进行选择。

命令：00 A4 00 00 02 3F 00

响应：6117

说明：对于 T=0 的卡片，6117 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于本公司读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 17

响应：6F 15 84 0E 31 50 41 59 2E 53 59 53 2E 44 44 46 30 31 A5 03 88 01 01 9000

说明：

返回的信息为嵌套的 TLV 格式的变长记录。

- ‘6F’为文件控制信息模板的记录标识。
- ‘15’为文件控制信息模板的记录数据长度（不包括 Tag、Length）。
- 84 0E 31 50 41 59 2E 53 59 2E 44 44 46 30 31 A5 03 88 01 01 为 21 字节的记录数据。
  - ‘84’为 DF 名称的记录标识。
  - ‘0E’为 DF 名称的记录数据长度（不包括 Tag、Length）。
  - 31 50 41 59 2E 53 59 53 2E 44 44 46 30 31 为 14 字节的记录数据，即 MF 的名称 1PAY.SYS.DDF01。
  - ‘A5’为文件控制信息专用模板的记录标识。
  - ‘03’ 为文件控制信息专用模板的记录数据长度（不包括 Tag、Length）
  - 88 01 01 为 3 字节的记录数据。
    - ‘88’为目录短文件标识符的记录标识。
    - ‘01’为目录短文件标识符的记录数据长度（不包括 Tag、Length）
    - ‘01’为 1 字节的记录数据，即目录基本文件（DIR）的短文件标识符。

[2] 条件：目录基本文件是一个变长记录文件。

操作：读目录基本文件（DIR）的第一条记录。

命令：00 B2 01 0C 00

响应：6C15

说明：对于 T=0 的卡片，6C15 表示要求终端向 IC 卡重发前一个命令的命令头，其中 Le=0x15.

命令：00 B2 01 0C 15

响应：70 13 61 11 4F 09 A0 00 00 00 03 86 98 07 01 50 04 50 42 4F 43 9000

说明：

返回的信息为嵌套的 TLV 格式的变长记录。

- ‘70’是变长记录的标识。
- ‘13’是变长记录数据长度。
- ‘61’是 ADF 应用目录入口封装标志。
- ‘15’是 ADF 应用目录入口封装数据长度。
- ‘4F’为银行应用目录文件 ADF 名称的记录标识。
- ‘09’为银行应用目录文件 ADF 名称的记录数据长度（不包括 Tag、Length）。
- ‘ A0 00 00 00 03 86 98 07 01’为 9 字节的记录数据，即银行应用目录文件 ADF 的名称。
- ‘50’为应用标签。
- ‘04’为应用标签长度。
- ‘50 42 4F 43’是‘PBOC’的 ASC 码。

[3] 条件：ADF 下发卡方专用数据文件的短文件标认识符=0x95(在建立银行应用目录文件 ADF 下的 KEY 文件时指定）

ADF 的名称:；‘A0 00 00 00 03 86 98 07 01’.

操作：对 ADF 进行选择。

命令：00 A4 04 00 09 A0 00 00 00 03 86 98 07 01

响应：6130

说明：对于 T=0 的卡片，6130 表示卡片要回送的数据长度，可以通过 Get Response

命令取返回数据。对于本公司读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 30

响应：6F 2E 84 09 A0 00 00 00 03 86 98 07 01 A5 21 9F 0C 1E 11 11 22 22 33 33 00 06 03 01 00 06 19 98 08 17 00 00 00 30 19 98 08 15 19 98 12 15 55 66 90 00

说明：

返回的信息为嵌套的 TLV 格式的变长记录。

- '6F'为文件控制信息模板的记录标识。
- '2E'为文件控制信息模板的记录数据长度（不包括 Tag、Length）
- 后续为'2E'个字节的记录数据。
  - '84'为 DF 名称的记录标识。
  - '09'为 DF 名称的记录数据长度（不包括 Tag、Length）。
  - A0 00 00 00 03 86 98 07 01 为 9 字节的记录数据，即 ADF 的名称。
  - 'A5'为文件控制信息专用数据的记录标识。
  - '21' 为文件控制信息专用数据的记录数据长度（不包括 Tag、Length）。
  - '9F0C'为发卡方定义的基本数据文件的文件控制信息的记录标识.
  - '1E' 为发卡方定义的文件控制信息专用数据的记录数据长度（不包括 Tag、Length），即标识符为 0015 的二进制文件的内容（见附录 2 的应用举例）。

在任何目录下选择 MF

命令格式：

CLA	INS	P1	P2	Lc	DATA
00	A4	00	00	02	3F 00

说明：成功选择 MF 后，MF 将成为当前目录，且 DF 安全状态寄存器的值自动等于 MF 安全状态寄存器的值。 当然，也可用 SELECT 命令对文件'1PAY.SYS.DDF01'直接选择。

按文件标识符选择当前目录下的文件或下级目录

命令格式：

CLA	INS	P1	P2	Lc	DATA
00	A4	00	00	02	文件标识符

说明：成功选择文件后，若选择的文件为子目录时，该目录成为当前目录，且 DF 安全状态寄存器的值变为 0；若选择的文件为 EF 时，该文件成为当前文件。

通过文件名称选择 DF

命令格式：

CLA	INS	P1	P2	Lc	DATA
00	A4	04	00	XX	DF 文件名

说明：Lc 定义了 DF 文件名的长度。

成功选择 DF 后，该目录成为当前目录，DF 安全状态寄存器的值变为 0。

9.Update Binary（写二进制文件）

定义与范围

Update Binary 命令用于写二进制文件、公钥文件。

注意事项

- Update Binary命令适用于二进制文件、公钥文件。
- 访问二进制文件的命令：
  - 建立文件（Create File）
  - 选择文件（Select File）
  - 读二进制文件（Read Binary）/写二进制文件（Update Binary）
- 只有满足二进制文件、公钥文件的写权限时才能执行此命令。

命令报文

表 9.1 Update Binary 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00/04	-
INS	1	D6	-
P1	1	XX	见说明
P2	1	XX	见说明
Lc	1	XX	-
DATA	XX	XX...XX	写入文件的数据
Le	-	-	不存在

说明：

- 若 P1 的高三位为 100，则低 5 位为短的文件标识符，P2 为欲读文件的偏移量。

P1								**P2**
b7	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b0**	
1	0	0	短文件标识符					文件的偏移量

- 若 P1 的最高位不为 1，则 P1 P2 为欲写文件的偏移量，所写的文件为当前文件。

P1								P2
b7	b6	b5	b4	b3	b2	b1	b0	
0	文件的偏移量							

- Lc 表示要写入的字节数。
  - 若为线路保护，Lc 为写入数据的长度+4 字节 MAC。
  - 若为加密线路保护，Lc 为加密后数据的长度+4 字节 MAC。

命令报文数据域

报文数据包括要写入的新数据。
若为线路保护文件数据域应包含 4 字节 MAC 码。
若为线路加密保护文件数据域应包含加密后的数据及 4 字节 MAC 码。
用维护密钥加密数据和计算 MAC，方法见“4.安全报文传送”。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 9.2 Update Binary 命令响应状态码

SW1	SW2	意义
90	00	正确执行
67	00	长度错误(Lc 域为空)
69	81	不是二进制或 FAC 密钥文件不可写
69	82	写的条件不满足
69	84	没有取随机数
69	86	没有选择当前可操作的文件
69	87	无安全报文
6A	81	不支持此功能（无 MF 或 MF 已锁定）
6A	82	未找到文件
6B	00	参数错误（偏移地址超出了 EF）

应用举例

[1] 条件：文件类型：二进制文件；
文件标识符=0005；

文件主体空间的大小=8 个字节；
建立时不采用线路保护。

操作：写二进制文件

命令：00 D6 85 00 08 11 22 33 44 55 66 77 88

响应：9000

10.Update Record（写记录文件）

定义与范围

Update Record 命令用于添加记录或更改指定的记录。

对线性结构文件来说，当指定的记录号不存在时，可按记录号顺序添加记录。按记录标识符访问的记录不存在时，也应视为添加新的记录。

对循环结构文件来说，当使用“上一个记录”命令选项时应视为添加新的记录。

注意事项

Update Record 命令适用于定长记录文件、变长记录文件和循环记录文件。

访问记录文件的命令如下所示：

建立文件（Create File）

选择文件（Select File）

读记录文件（Read Record）

写记录文件（Update Record）

只有满足记录文件写权限时才能执行此命令。

命令报文

附 10.1Update Record 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00/04	-
INS	1	DC	-
P1	1	XX	记录号或记录标识符（‘00’，表示当前记录）
P2	1	XX	见说明
Lc	1	XX	数据长度
DATA	XX	XXXX	添加的或更新原有记录的新记录
Le	-	-	不存在

说明：

参数 P2 的含义

b7	b6	b5	b4	b3	b2	b1	b0	描述
0	0	0	0	0	–	–	–	当前的 EF 文件
x	x	x	x	x	–	–	–	SFI
1	1	1	1	1	–	–	–	保留
–	–	–	–	–	1	x	x	利用 P1 中的记录号
–	–	–	–	–	1	0	0	P1 记录号
–	–	–	–	–	0	x	x	利用 P1 中的记录标识符
–	–	–	–	–	0	0	0	P1 指定标识的第一个记录
–	–	–	–	–	0	0	1	P1 指定标识的最后一个记录
–	–	–	–	–	0	1	0	P1 指定标识的下一个记录
–	–	–	–	–	0	1	1	P1 指定标识的上一个记录

注：X X X X X 代表短文件标识符（SFI）；– – – – – 代表全 0 或短文件标识符

注：1、循环记录文件只能用 P1=‘00’，P2=‘03’来添加。

2、当 P1≠ ‘00’，P2=‘04’， 若 P1 等于已有记录的最大记录号+1，则添加。

命令报文数据域

命令报文数据域由添加的或更新原有记录的新记录组成。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

附 10.2 Update Recod 命令响应状态码

SW1	SW2	意义
90	00	命令成功执行
65	81	写 EEPROM 不成功
67	00	长度错误
69	81	当前文件不是定长或变长记录文件
69	82	写的条件不满足
69	84	没有取随机数
69	86	没有选择当前可操作的文件
6A	81	不支持此功能（无 MF 或 MF 已锁定）
6A	82	未找到文件
6A	83	未找到记录
6A	84	文件无足够空间

应用举例

条件：文件类型：定长记录文件；
文件标识符=0002；
记录数=3 条；
记录长度=12 个字节；
建立时不采用线路保护。
操作：写定长记录文件，不进行线路保护。
命令：00 DC 01 14 0C 01 02 03 04 05 06 07 08 09 0A 0B 0C
说明：01 02 03 04 05 06 07 08 09 0A 0B 0C 为写入的数据。

条件：文件类型：变长记录文件；
文件标识符=0001；
建立时不采用线路保护。
[操作 1]：在变长记录文件中建立 1 条记录标识为 AA 的新记录，不进行线路保护。
命令：00 DC 01 0A 04 AA 02 11 22
响应：9000
[操作 2]：修改记录标识为 AA 的记录，同时将记录标识改为 CC，不进行线路保护。
命令：00 DC AA 08 04 CC 02 33 44
响应：9000

条件：文件类型：循环文件
文件标识符=0003；
记录数=3 条；
记录长度=12 个字节；
建立时不采用线路保护。
操作：往循环文件中追加 1 条记录，不进行线路保护。
命令：00 DC 00 03 0C 11 22 33 44 55 66 77 88 99 AA BB CC
响应：9000

11.Verify PIN（验证口令）

定义与范围

Verify PIN 命令用于校验命令数据域的口令密钥正确性。

注意事项

- 在满足该口令密钥的使用权限时才可执行该命令。
- 若PIN值的后面字节为连续的FF,校验时可以忽略该段字节,但若PIN值为全FF,则最少应输入两个FF值。

命令报文

表 11.1Verify PIN 命令报文编码

代码	长度 (byte)	值 (Hex)	描述 （Hex）
CLA	1	00	-
INS	1	20	-
P1	1	00	-
P2	1	XX	口令密钥标识号
Lc	1	XX	口令密钥长度（最大 32 字节）
DATA	XX	XX...XX	外部输入的口令密钥
Le	-	-	不存在

说明：

- 若口令验证成功，则安全状态寄存器的值被置成该密钥的后续状态，同时口令错误计数器被置成初始值。
- 若验证错误，则口令可试次数减一，若口令已被锁死，则不能再执行该命令。
- 出于对安全的考虑，解锁口令指令已经封掉，口令被锁定后无法进行解锁。
- 当长度不足32字节时，由TimeCOS®自动填充FF至32字节 。

命令报文数据域

命令报文数据域由持卡者输入的口令密钥组成。
用口令解锁密钥加密口令密钥和计算MAC，方法见“4. 安全报文传送”。

响应报文数据域

响应报文数据不存在。

响应报文状态码

当命令数据域中外部输入的口令密钥与卡中存放的口令密钥校验失败时，

- IC卡将回送SW2=CX，X表示个人密码允许重试的次数；
- 当卡片回送SW2=C0时，表示不能重试口令密钥，此时再使用Verify PIN命令时，将回送失败状态码 SW1 SW2= ‘6983’。

IC 卡可能回送的状态码如下所示：

表 11.2Verify PIN 命令响应状态码

SW1	SW2	意义
90	00	正确执行
63	CX	还剩 x 次可试机会
62	83	口令密钥校验错误
67	00	错误的长度
69	81	不是口令密钥
69	82	密钥使用条件不满足
69	83	认证方法（口令密钥）锁死
69	84	没有取随机数
6A	82	KEY 文件未找到
6A	86	参数 P1P2 错误
93	02	密钥线路保护错误
94	03	密钥未找到

12.Verify & Change PIN（验证并修改口令）

定义与范围

Verify&Change PIN 命令用于核对并修改长度在 32 字节以内口令。

注意事项

在满足口令使用权限时，可以使用 Verify&Change 命令用于核对并修改长度为 32 字节的口令。
必须使用 DES&MAC 方式来验证，并且输入 Pin 长度必须为 32 字节。（不足的以 0xFF 补足 32 字节）
Verify&Change 命令**必须**使用口令解锁密钥进行线路保护。

命令报文

表 12.1Verify & Change PIN 命令报文编码

代码	长度	值	描述

	(byte)	(Hex)	
CLA	1	84	-
INS	1	24	-
P1	1	00	-
P2	1	XX	口令密钥标识
Lc	1	40	
DATA	40	XX…XX	旧口令+新口令
Le	-	-	不存在

说明：

- 若核对成功，则安全状态寄存器被置为该口令密钥的后续状态，并用新口令取代旧口令，错误计数器被恢复；
- 若核对不成功，则可再试次数减一，且不修改口令值。
- 当输入的数据不足 0x40 字节时，由上层应用自动填充 FF。

命令报文数据域

命令报文数据域由旧口令和新口令组成。

用口令解锁密钥加密口令密钥和计算 MAC，方法见“4.安全报文传送”。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 12.2Verify & Change PIN 命令响应状态码

SW1	SW2	意义
90	00	正确执行
63	CX	还剩 x 次可试机会
6A	82	KEY 文件未找到
6A	86	参数 P1 P2 不正确
93	02	安全报文数据项不正确
93	03	应用永久锁定
94	03	密钥未找到

13.复位

定义与范围

Restart 用来使 KEY 重新复位，回到 MF 目录下，返回复位信息。

注意事项

复位后，所有安全状态寄存器清 0。进行操作需要再次外部认证或验证口令。

命令报文

表 12.3　restart 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00	-
INS	1	12	-
P1	1	00	-
P2	1	00	
Lc	-	-	
DATA	-	-	
Le	1	-00	不存在

响应报文数据域

复位信息。

14.数据压缩（Data Compress-SHA-1）

定义与范围

Data Compress 命令用安全散列算法 SHA1 将数据压缩为 20 个字节，用 RSA 签名或验证时使用。也可以由上层应用计算出部分摘要，通过级联的方式对分段数据进行摘要。（请参见 5.1 SHA 数据压缩算法）

命令报文

表 14.1 Data Compress 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	CC	-
P1P2	2	XXXX	见表 8.3 和表 8.4
Lc	1	XX	-
DATA	XX	XX...XX	待压缩数据
Le	1	14	-

说明：对数据总长度小于 256 字节的数据直接进行压缩。
否则以 64 字节为一块重复执行此命令进行分块压缩，压缩结果由最后一块数据产生，见表 8.4。
若为级联方式计算摘要，需讲上层计算的中间结果当作首块传入，参数设置见表 8.4

表 14.2 Data Compress 命令的 P1P2 设置（单块数据压缩）

P1								**P2**								
b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	含　义
0	0	0	0	0	0	0	0	数据总长度(最大 255 字节)								表示单块数据
	RUF															

表 14.3Data Compress 命令的 P1P2 设置（分块数据压缩）

b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	Lc	含　义
1	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	20	级联，送相应算法的上一级计算结果
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-	首块数据
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-	中间块数据
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Len(data)+4	最后一块数据
	RUF																

命令报文数据域

命令报文数据域包括待压缩的数据。

响应报文数据域

当进行单块数据压缩时，响应报文数据域是 20 字节压缩的结果数据。
当进行分块数据压缩时，除对最后一块数据压缩，卡片只回送 SW1SW2=9000，并不返回压缩结果。只有对最后一块数据进行压缩时，卡片才返回压缩结果。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 14.4 Data Compress 命令响应状态码

SW1	**SW2**	**意义**
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）

67	00	错误的长度
6A	81	不支持此命令（卡片锁死或无 MF）

应用举例

[1] 操作：对长度为 64 字节的数据进行压缩，压缩结果由 DATA 产生。

命令：80 CC 00 40 40 DATA（64 个字节的 0）

响应；6114

说明：对于 T=0 的卡片，6114 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于握奇读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 14

响应：C8 D7 D0 EF 0E ED FA 82 D2 EA 1A A5 92 84 5B 9A 6D 4B 02 B7 9000

说明：C8 D7 D0 EF 0E ED FA 82 D2 EA 1A A5 92 84 5B 9A 6D 4B 02 B7 为 20 字节的压缩结果。

注:对单组的数据进行压缩可参考[1]。

[2] 操作：对长度为 160 字节的数据进行压缩，压缩结果由 DATA3 产生。

[步骤 1] 对第一块数据进行压缩。

命令：80 CC 3F FF 40 DATA1（64 个字节的 0）

响应：9000

[步骤 2] 对中间数据块进行压缩。

命令：80 CC BF FF 40 DATA2（64 个字节的 0）

响应：9000

[步骤 3] 对最后一块数据进行压缩。

命令：80 CC 80 A0 20 DATA3（32 个字节的 0）

响应；6114

说明：对于 T=0 的卡片，6114 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于握奇读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 14

响应：97 97 ED F8 D0 EE D3 6B 1C F9 25 47 81 60 51 C8 AF 4E 45 EE 9000

说明：97 97 ED F8 D0 EE D3 6B 1C F9 25 47 81 60 51 C8 AF 4E 45 EE 为 20 字节的压缩结果。

[3]操作：对长度 544 字节的数据进行级联方式摘要（上一级对前 256 字节进行摘要的中间结果传下。）

[步骤 1]上层传送中间结果：

命令：80 CC C0 FF 14 DATA1（20 个字节的中间结果）

响应：9000

[步骤 2] 对中间数据块进行压缩。

命令：80 CC BF FF 40 DATA2（64 个字节数据）

响应：9000

[步骤 3]重复步骤 2

[步骤 4]重复步骤 2

[步骤 5] 对最后一块数据进行压缩。

命令：80 CC 80 00 24　DATA3（32 个字节的 0）+DATALC(数据总长度 544 字节)

响应；9000 +压缩结果

说明：此部分要将数据的总长度输入，得到最终的摘要结果。LC=DATA 长度+数据总长度（4 字节）

注:对分组的数据进行压缩可参考[2]。

15.数据压缩（Data Compress-MD5）

定义与范围

此命令用于使用 MD5 算法计算数据的摘要，用来对数据进行签名等操作。

命令报文

表 15.1MD5 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	7B	-
P1P2	2	XXXX	见表 8.3 和表 8.4
Lc	1	XX	-
DATA	XX	XX...XX	待压缩数据
Le	1	10	-

说明：对数据总长度小于 256 字节的数据直接进行压缩。
否则以 64 字节为一块重复执行此命令进行分块压缩，压缩结果由最后一块数据产生，见表 8.4。
若为级联方式计算摘要，需讲上层计算的中间结果当作首块传入，参数设置见表 8.4

表 15.2 Data Compress 命令的 P1P2 设置（单块数据压缩）

P1								**P2**								
b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	含　义
0	0	0	0	0	0	0	0	数据总长度(最大 255 字节)								表示单块数据
	RUF															

表 15.3Data Compress 命令的 P1P2 设置（分块数据压缩）

b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1	Lc	含　义
1	1	0	0	0	0	0	0	1	1	1	1	1	1	1	1	16	级联，送上一级计算结果
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-	首块数据
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-	中间块数据
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Len(data)+4	最后一块数据
	RUF																

命令报文数据域

命令报文数据域包括待压缩的数据。

响应报文数据域

当进行单块数据压缩时，响应报文数据域是 20 字节压缩的结果数据。
当进行分块数据压缩时，除对最后一块数据压缩，卡片只回送 SW1SW2=9000，并不返回压缩结果。只有对最后一块数据进行压缩时，卡片才返回压缩结果。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 15.4 Data Compress 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
67	00	错误的长度
6A	81	不支持此命令（卡片锁死或无 MF）

应用举例

[1] 操作：对长度为 64 字节的数据进行压缩，压缩结果由 DATA 产生。
命令：80 CC 00 40 40 DATA（64 个字节的 0）
响应；6114

说明：对于 T=0 的卡片，6114 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于握奇读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 14

响应：C8 D7 D0 EF 0E ED FA 82 D2 EA 1A A5 92 84 5B 9A 6D 4B 02 B7 9000

说明：C8 D7 D0 EF 0E ED FA 82 D2 EA 1A A5 92 84 5B 9A 6D 4B 02 B7 为 20 字节的压缩结果。

注:对单组的数据进行压缩可参考[1]。

[2] 操作：对长度为 160 字节的数据进行压缩，压缩结果由 DATA3 产生。

[步骤 1] 对第一块数据进行压缩。

命令：80 CC 3F FF 40 DATA1（64 个字节的 0）

响应：9000

[步骤 2] 对中间数据块进行压缩。

命令：80 CC BF FF 40 DATA2（64 个字节的 0）

响应：9000

[步骤 3] 对最后一块数据进行压缩。

命令：80 CC 80 A0 20 DATA3（32 个字节的 0）

响应；6114

说明：对于 T=0 的卡片，6114 表示卡片要回送的数据长度，可以通过 Get Response 命令取返回数据。对于握奇读写机具，产品默认设置为可自动读取卡片响应数据，所以无需通过 Get Response 命令取返回数据。

命令：00 C0 00 00 14

响应：97 97 ED F8 D0 EE D3 6B 1C F9 25 47 81 60 51 C8 AF 4E 45 EE 9000

说明：97 97 ED F8 D0 EE D3 6B 1C F9 25 47 81 60 51 C8 AF 4E 45 EE 为 20 字节的压缩结果。

[3]操作：对长度 544 字节的数据进行级联方式摘要（上一级对前 256 字节进行摘要的中间结果传下。）

[步骤 1]上层传送中间结果：

命令：80 CC C0 FF 14 DATA1（20 个字节的中间结果）

响应：9000

[步骤 2] 对中间数据块进行压缩。

命令：80 CC BF FF 40 DATA2（64 个字节的 0）

响应：9000

[步骤 3]重复步骤 2

[步骤 4]重复步骤 2

[步骤 5] 对最后一块数据进行压缩。

命令：80 CC 80 00 18 DATA3（32 个字节的 0）+DATALC(数据总长度 544 字节)

响应；9000 +压缩结果

说明：此部分要将数据的总长度输入，得到最终的摘要结果。LC=DATA 长度+数据总长度（4 字节）

16.数字签名（Digital Signatures）

定义与范围

Digital Signatures 命令用非对称密码算法 RSA 的私钥对数据进行签字。（请参见 5.4 RSA 签名/验证）

命令报文

表 16.1 Digital Signatures 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	C2	-
P1P2	2	XXXX	私钥文件标识符
Lc	1	80/00	-
DATA	XX	XX...XX	需要签字的数字
Le	1	80	-

说明：***在满足该私钥文件的使用权限时才能执行此命令。***

命令报文数据域

必须为 128 字节或者 0 字节数据，当 LC=0X00 时，对卡内压缩数据进行签名，对于外送数据摘要只支持 128 字节长度，不足的需由卡外填充至 128 字节。不支持符合 PKCS#1 标准的 SHA-256 填充方式的数据签名。

响应报文数据域

响应报文数据域由签名后的数据组成。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 16.2 Digital Signatures 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
67	00	错误的长度

69	81	P_1 P_2 指定的标识符不是相应的私钥文件
69	82	使用条件不满足
6A	82	私钥文件未找到

17.数据解密（Data Decrypt）

定义与范围

Data Decrypt 命令用非对称密码算法 RSA 的私钥对加密数据进行解密。（请参见 5.5 RSA 加密/解密）。

命令报文

表 17.1 Data Decrypt 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	C8	-
P1P2	2	XXXX	私钥文件标识符
Lc	1	80	-
DATA	128	XX...XX	需解密的密文
Le	1	00	-

说明：***在满足该私钥文件的使用权限时才能执行此命令。***

命令报文数据域

命令报文数据域由需解密的密文组成。当为 RSA 数据解密时，Lc = 80H。

响应报文数据域

响应报文数据域由解密后的明文组成。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 17.2 Data Decrypt 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
67	00	错误的长度
69	81	P_1 P_2 指定的标识符不是相应的私钥文件
69	82	使用条件不满足
6A	82	私钥文件未找到

18.数据加密（Data Encrypt）

定义与范围

Data Encrypt 命令用非对称密码算法 RSA 的公钥对数据进行加密。（请参见 5.5 RSA 加密/解密）

命令报文

表 18.1Data Encrypt 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	C6	-
P1P2	2	XXXX	公钥文件标识符
Lc	1	XX	-
DATA	XX	XX...XX	需要加密的数据
Le	1	80	-

说明：**在满足该公钥文件的使用权限时才能执行此命令。**

P1P2=0000 时，表示用卡内临时公钥对数据进行加密。该临时公钥由 Data Compress 命令分批送入。Data Compress 命令除对临时公钥进行压缩以便卡中公钥对该临时公钥作认证外,还保存该临时公钥用于对其他签名进行验证以及对数据进行加密。

命令报文数据域

命令报文数据域由需加密的数据组成。Lc 域有如下两种情况：

1. Lc = 80H，此时，数据域的数据必须小于 RSA 密钥模数 n，且数据域高 2 字节不能为‘0002’。

2. Lc ≠ 80H，此时，Lc 的最大长度请参见“5.5.1 加密”。

响应报文数据域

响应报文数据域由加密后的密文组成。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 18.2 Data Encrypt 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
67	00	错误的长度
69	81	P_1 P_2 指定的标识符不是相应的公钥文件
69	82	使用条件不满足
6A	82	公钥文件未找到

19.生成 RSA 密钥对（Generate RSA Key）

定义与范围

Generate RSA Key 命令用于卡片自动生成模长为 1024 位的 RSA 密钥对,通过此指令的数据域指定密钥对所存入的公钥和私钥文件的标示符，密钥对不能回送终端，私钥在任何情况下不可读写。公钥只能通过读公钥的专用指令才能读取，可以选择明文读和密文读方式。（请参见 5.3 RSA 密钥生成）

命令报文

表 19.1 Generate RSA Key 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	CE	-
P1	1	00/01	-
P2	1	00	
Lc	1	08	
DATA	8	XXXXXXXX	私钥文件的标示符+公钥文件的标示符
Le	1	00	-

说明：私钥和公钥存到数据域中指定的公钥和私钥标示符中。

注：具体操作步骤见“8.5.6 应用举例”。

命令报文数据域

命令报文数据域不存在。

响应报文数据域

响应报文数据域为公开钥(包括模数 n 和公开指数 e)。

若 P2 为 00 时,响应报文还包括秘密钥{秘密钥包括 q、d mod (q-1)、p、d mod (p-1)、(inverse of q) mod p }

响应报文将以 TLV 格式送出, 其中 L 值指的是字节长度。

具体操作步骤见“8.5.6 应用举例”。

响应报文的TLV格式中的T值有如下意义:

表 19.2 RSA 密钥对各个元素值的含义

字符	十六进制	含义
N	6E	模数 n
E	65	公开指数 e
P	70	素数 p
q	71	素数 q
P	50	秘密钥指数 d 模 (p-1)
Q	51	秘密钥指数 d 模 (q-1)
I	49	q 模 p 的逆

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 19.3 Generate RSA Key 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。(仅用于 T=0)
67	00	长度错误(Lc 域为空)
69	82	写的条件不满足
6A	81	不支持此功能（无 MF 或 MF 已锁定）
6A	82	P1、P2 所指定的秘密钥文件不存在

69	81	P1、P2 所指的文件不是秘密钥文件

应用举例

操作：产生一模长为 1024 位的 RSA 密钥对

[步骤 1] 向卡片发送生成 RSA 密钥命令 Generate RSA Key。

命令：80 CE 00 00 08 00010002
响应：9000
说明： KEY 计算出公钥和私钥存入 0002 和 0001 文件中。

响应：06Eh 080h ⟶ n

09Bh 09Eh 0C6h 074h 065h 05Ah 0BCh 0BAh
0C6h 0B0h 05Dh 03Ch 024h 03Ch 090h 059h
07Dh 05Bh 06Dh 003h 07Bh 0ADh 09Ah 0B4h
058h 0FEh 080h 022h 0ECh 0F0h 0ABh 084h
0F9h 007h 084h 091h 0E2h 008h 0A6h 09Bh
0EDh 0D7h 045h 0C9h 0DDh 0BFh 0F8h 07Ah
08Bh 0E1h 017h 0CDh 03Dh 0D3h 06Fh 0B2h
059h 00Ch 00Eh 047h 00Bh 08Eh 083h 0C5h
00Fh 0AFh 0D8h 021h 00Ch 0D1h 00Bh 0B4h
024h 08Dh 066h 0ACh 093h 0A3h 0E4h 061h
0EFh 026h 050h 01Ch 030h 0EDh 073h 0F1h
092h 0D8h 02Ch 0C6h 038h 0D4h 06Dh 081h
048h 02Bh 0CCh 042h 0F8h 060h 061h 0ADh
08Dh 07Fh 08Dh 06Dh 087h 0BFh 07Dh 01Ch
061h 02Bh 0C0h 042h 047h 0DBh 0DDh 0C9h
03Fh 007h 023h 0E1h 03Dh 0DAh 0DBh 085h
9000

[步骤 3] 取模数 e。

命令：80 CE 00 00 00
响应：6105
命令：00 C0 00 00 85

响应：065h 003h ⟶ e

001h 000h 001h
9000

[步骤 4] 取模数 q。

命令：80 CE 00 00 00
响应：6142
命令：00 C0 00 00 42

响应：071h 040h ⟶ q

0C5h 0D7h 020h 028h 0FAh 0A7h 091h 055h
023h 0E2h 00Dh 0E4h 028h 07Ch 065h 0B7h
018h 059h 0D9h 00Dh 0BAh 0E7h 0CFh 06Ah

0F1h 0E3h 010h 0C3h 07Eh 048h 00Dh 0BCh
076h 07Bh 004h 086h 0B6h 07Fh 0CDh 06Ch
084h 01Ah 00Bh 086h 0B9h 096h 0AAh 083h
068h 063h 03Ch 04Dh 043h 084h 0B8h 06Dh
048h 0AAh 0C4h 0C9h 01Bh 050h 047h 049h
9000

[步骤 5] 取模数 Q -- (d mod (q - 1))。

命令：80 CE 00 00 00

响应：6142

命令：00 C0 00 00 42

响应：051h 040h ⟶ Q -- (d mod (q - 1))

0A3h 08Dh 0A3h 0EDh 09Ch 0C2h 018h 0B8h
09Dh 010h 08Dh 051h 058h 052h 0F6h 0B7h
0B5h 0EEh 0D9h 02Ch 0ABh 09Eh 065h 0EFh
0D0h 086h 059h 0DEh 073h 0B0h 057h 082h
0BDh 024h 017h 0EAh 0D2h 046h 0B7h 069h
085h 090h 00Eh 085h 053h 03Ah 006h 03Eh
0DAh 076h 067h 06Ch 0ACh 06Bh 0B5h 017h
0CBh 062h 039h 08Ah 0D4h 004h 0BAh 0D9h
9000

[步骤 6] 取模数 p。

命令：80 CE 00 00 00

响应：6142

命令：00 C0 00 00 42

响应：070h 040h ⟶ p

0C9h 05Eh 04Ah 008h 00Eh 0DCh 0A5h 045h
090h 01Ch 052h 0F8h 03Eh 0B0h 06Bh 08Fh
0CFh 0EAh 0A8h 0F5h 01Eh 0ADh 0D3h 082h
052h 030h 043h 004h 09Ch 025h 063h 087h
037h 020h 064h 03Fh 016h 0B0h 050h 0CCh
038h 006h 01Bh 0DFh 0E6h 078h 0C3h 099h
0F7h 0C4h 03Fh 095h 081h 0E0h 077h 05Ch
0A3h 078h 0B8h 09Ch 08Dh 09Bh 046h 05Dh
9000

[步骤 7] 取模数 P -- (d mod (p - 1))。

命令：80 CE 00 00 00

响应：6142

命令：00 C0 00 00 42

响应：050h 040h ⟶ P -- (d mod (p - 1))

044h 003h 003h 0B0h 01Bh 00Ch 0EDh 009h
044h 0B6h 03Ch 053h 0BAh 020h 0AEh 003h
0A1h 0AEh 0D9h 028h 009h 017h 09Eh 0C3h
07Ah 06Ch 0F0h 085h 0C3h 013h 061h 0BDh
04Eh 0A2h 033h 019h 097h 0D9h 02Fh 040h
0FAh 07Fh 01Dh 0B5h 00Eh 0CBh 0A5h 00Dh
000h 0C1h 018h 0D4h 0AFh 04Ch 018h 024h
082h 0D6h 008h 04Ch 060h 00Bh 09Ch 0C5h

9000

[步骤 8] 取模数 I -- (inverse of q) mod p。

命令：80 CE 00 00 00

响应：6142

命令：00 C0 00 00 42

响应：049h 040h ——————————————————→I -- (inverse of q) mod p

07Fh 0DEh 017h 036h 0DAh 018h 01Eh 0EFh

0D0h 050h 0BDh 02Ch 057h 0BEh 010h 07Bh

008h 07Dh 0C6h 0C7h 0F5h 076h 0A2h 059h

035h 07Dh 0EAh 0B0h 073h 0E6h 051h 0EBh

0D3h 007h 0DDh 073h 056h 08Bh 076h 046h

03Fh 0AEh 00Ah 0B9h 0A3h 0E3h 00Dh 071h

0FBh 0FEh 055h 002h 0F6h 0B6h 04Dh 0C2h

079h 064h 0FDh 028h 0BBh 013h 023h 0B2h

9000

20.签名验证（Signatures Verify）

定义与范围

Signatures Verify 命令用非对称密码算法 RSA 的公钥对签名数据进行认证。（请参见 5.4 RSA 签名/验证）

命令报文

表 20.1 Signatures Verify 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	C4	-
P1P2	2	XXXX	公钥文件标识符
Lc	1	80	-
DATA	128	XX...XX	本公钥对应的私钥所产生的数字签名
Le	1	80	-

说明：**在满足该公钥文件的使用权限时才能执行此命令。**

若 P1P2=0000 时，表示使用卡内临时公钥进行验证。该临时公钥由 Data Compress 命令分批送入。Data Compress 命令除对临时公钥进行压缩以便卡中公钥对该临时公钥作认证外，还保存该临时公钥用于对其他签名进行验证以及对数据进行加密。

命令报文数据域

命令报文数据域由本公钥对应的私钥所产生的数字签名数据组成。

当为 RSA 签名验证时，Lc = 80H。

响应报文数据域

验证被签名的压缩数据时无响应报文数据域，其他情况响应报文数据域由认证后的数据组成。
响应报文数据域说明：

被签名数据不大于 109 字节时，签名验证响应报文为 128 字节卡内填充后的数据和被签名数据（详见 5.4.1 签名）；被签名数据大于 109 字节时，签名验证响应报文为被签名数据和 RAM 内补足的共 128 字节数据。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 20.2Digital Signatures 命令响应状态码

SW1	SW2	意义
90	00	正确执行
61	XX	正确执行 XX 表示响应数据长度。可用 Get Response 命令取回响应数据。（仅用于 T=0）
67	00	错误的长度
69	81	P_1 P_2 指定的标识符不是相应的公钥文件
69	82	使用条件不满足
6A	82	公钥文件未找到

21．私钥文件清空（Erase SK File）

定义与范围

私钥文件清空指令用于清空私钥文件的内容，使私钥文件内容恢复为全为 FF。

命令报文

表 21.1 Erase SK File 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	7A	-
P1P2	2	XXXX	私钥文件标识符
Lc	1	–	–
DATA	128	–	–

Le	1	00	-

说明：**在满足该私钥文件所在 ADF 的擦除权限时才能执行此命令。**

命令报文数据域

无

响应报文数据域

无

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 21.2 Erase SK File 命令响应状态码

SW1	SW2	意义
90	00	正确执行
69	81	P_1 P_2 指定的标识符不是相应的公钥文件
69	82	使用条件不满足
6A	82	私钥文件未找到

22.序列号设置/读取指令(SetSerNo)

定义与范围

序列号设置/读取指令可以设置/读取 usbKey 的唯一序列号，注意一个 usbKey 只能设置一次序列号，序列号一旦设置成功就不能再进行设置了。

命令报文

表 22.1 序列号设置/读取指令命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	
INS	1	7C	
P1	1	XX	
P2	1	00	
Lc	1	xx	

DATA	xx	xx	
Le	不存在	不存在	

参数说明

P1=00 Lc=00	读序列号
P1=01 Lc=序列号长度<=32byte	写序列号

命令报文数据域

说明	长度(字节)
序列号数值	Lc

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 22.2 序列号设置/读取指令命令响应状态码

SW1	SW2	意义
90	00	正确执行
64	01	序列号不存在
64	02	序列号已设置

23.Erase MF（擦除主文件 MF）

定义与范围

Erase MF 命令用于擦除 MF，该指令仅对 MF 有效。

注意事项

在满足 MF 的擦除权限时，可以用此命令擦除 MF 下的所有文件(DF、EF)，但 MF 当前的访问权限、空间等信息并没有改变（即不能擦除当前 MF 的文件头信息），且 MF 的文件名称也不能被擦除。

当前 MF 下无任何文件时，则在该目录下可任意擦空 MF 而不受擦除权限控制。

当前 MF 被擦除后，则在该目录下可任意建立文件和读写文件而不受文件访问权限的限制（私钥文件除外），一旦离开 MF 再进入 MF 时，将遵循文件的访问权限。

命令报文

23.1 Erase MF 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	0E	-
P1	1	00	-
P2	1	00	-
Lc	1	00	-
DATA	-	-	不存在
Le	-	-	不存在

命令报文数据域

命令报文数据域不存在。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 23.2 Erase MF 命令响应状态码

SW1	SW2	意义
90	00	命令成功执行
65	81	写 EEPROM 不成功
69	82	擦除权限不满足
6A	81	用此命令擦除 DF 时回送的错误信息

24.Erase EF/DF（擦除目录文件）

定义与范围

Erase EF/DF 命令用于擦除 MF 下与 P1P2 匹配的 EF/DF，并且该文件的访问权限、空间等信息（即头文件，包括文件名称）也都被擦除。擦除所带来的剩余空间也重新任意分配。(P1P2 指 MF 下某一 EF/DF 的 文件标识)

注意事项

- 擦除任何文件（EF/DF）均必须在其父 DF 下进行，且必须满足父 DF 的擦除权限。
- 在满足其父 DF 的擦除权限时，可以用此命令擦除 MF 下与 P1P2 匹配的 EF/DF，并且该文件的访问权限、空间等信息（即头文件，包括文件名称）也都被擦除。擦除所带来的剩余空间也重新任意分配。
- 擦除与P1P2匹配EF/DF时，不对别的EF/DF产生任何影响。

命令报文

表 24.1 Erase EF/DF 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	00	-
INS	1	E4	-
P1	1	00	-
P2	1	00	-
Lc	1	02	-
DATA	-	-	见说明
Le	-	-	不存在

说明：数据域中为 MF 下的某一 DF/EF 的文件标识号，长度 2 字节。

命令报文数据域

命令报文数据域不存在。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 24.2　Erase EF/DF 命令响应状态码

SW1	SW2	意义
90	00	命令成功执行
65	81	写 EEPROM 不成功
69	82	擦除权限不满足

25.Write Key（增加或修改密钥）

定义与范围

Write Key 命令可向卡中装载密钥（向 KEY 文件写入密钥），或更新卡片已存在密钥（口令密钥除外）。请参见“3.6 内部基本文件”。

注意事项

- 在满足当前 DF 下 KEY 文件的增加权限时时，可用 Write Key 命令向 KEY 文件中写入密钥。
- 当满足密钥的修改权限时，可以对密钥值进行修改(口令密钥除外)。

命令报文

表 25.1 Write Key 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80/84	-
INS	1	D4	-
P1	1	01	用于密钥装载
		3X	密钥类型，用于密钥更新
P2	1	XX	密钥标识
Lc	1	XX	数据长度
DATA	Lc	XXXX	密钥头+密钥值
Le	-	-	不存在

命令报文数据域

命令数据域中的所有权限设置请参见“5. TimeCOS®/PK 的安全体系”。

密钥装载

命令报文数据=密钥头（5 字节）+密钥值，
若为线路加密保护则由被加过密的数据附上 4 字节 MAC 码组成。（见说明[2]）

表 25.2 Write Key 之密钥装载命令报文数据域

数据域 / 密钥类型	Byte 1	Byte2	Byte3	Byte 4	Byte 5	密钥值长度（byte）
DES 加密密钥	‘30’	使用权限	更改权限	密钥版本号	算法标识	8/16
DES 解密密钥	‘31’	使用权限	更改权限	密钥版本号	算法标识	8/16
DESMAC 密钥	‘32’	使用权限	更改权限	密钥版本号	算法标识	8/16
维护密钥	‘36’	使用权限	更改权限	‘FF’	错误计数器	8/16
主控密钥	即密钥标识为 00 的外部认证密钥，其命令报文数据域同外部认证密钥。					
外部认证密钥	‘39’	使用权限	更改权限	后续状态	错误计数器	8/16
口令密钥	‘3A’	使用权限	‘EF’	后续状态	错误计数器	最大 32 字节
连接 MF 下的密钥	‘3X’	此密钥数据域只有密钥类型 1 字节。采用此种方法装载的密钥，其真正的密钥属性和内容为 MF 下相对应的密钥类型和标识的密钥属性和内容（见 3.6.5 全局密钥）				

说明：因为当装载口令密钥时，如果长度不足 32 字节，COS 内部会自动填满 32 字节，所以当使用连接在 MF 下的密钥方式装载时，如果 MF 下不存在该条密钥，COS 会自己建立一条内容全为 FF 的口令密钥。

注：表中密钥版本号、后续状态等见说明[4]。

说明：

[1] 对于密钥可以采用安全报文传送。

如对密钥进行安全报文传送（使用 Write Key、Verify 等），只需在安装密钥时改变 Byte1(密钥类型)字节高两位即可。

密钥数据域 Byte1（密钥类型）字节定义如下：

b7	b6	b5	b4	b3	b2	b1	b0	线路保护方式
0	0	密钥类型						无
1	1	密钥类型						DES&MAC

例：对密钥若需进行线路加密保护（DES&MAC）则将密钥类型最高位及次高位均置 1，如外部认证密钥类型由‘39’变为‘F9’.

◆ ***注：具有线路保护属性的密钥，必须用相应的线路保护模式装载与修改，但 MF 下的主控密钥装载除外。***

[2] 以安全报文方式装载或修改密钥时所使用的密钥如下：

- 当装载 MF 下的主控密钥时，分以下两种情况：
  i. 由厂家在卡片 MF 的 KEY 文件中已预先装入一条主控密钥（即卡片传输密钥，见“4. 卡片初始化设置”），其密钥带有线路保护属性。用户可以在发卡中先认证或替换此密钥，后继续对卡片进行发卡操作。
  ii. 在用户擦除 MF 后，MF 下的主控密钥必须以明文方式装入，但可以设置密钥类型为线路保护方式，此后可以使用线路保护方式更新此密钥。
  当修改 MF 下的主控密钥时，用 MF 下的主控密钥加密数据和计算 MAC。
- 当装载应用目录（MF 除外）下的主控密钥时，用上一级应用目录下的主控密钥加密数据和计算 MAC。
  当修改应用目录（MF 除外）下的主控密钥时，用当前应用目录下的主控密钥加密数据和计算 MAC。
- 当装载/更新应用目录（MF 或 DF）下的密钥（主控密钥除外）时，用当前应用下的主控密钥加密数据和计算 MAC。

MAC 计算方法见“《TimeCOS®/PK 通用技术参考手册》之 4.安全报文传送”。

[3] 若应用目录下某类型密钥只有一个，则其密钥标识是‘00’，否则，应从‘01’顺序开始。在一个应用下：

- 只能有一个主控密钥,它的密钥标识必须是 00。
- 维护密钥最多可以有 4 个，密钥标识为 00～03。
- 密钥标识不能是‘FF’。

[1] 术语解释：

- 使用权限
  指该密钥在使用时如核对、认证、运算时所需满足的条件。
  例如：使用权为 41 表示在使用该密钥时当前目录安全状态寄存器值必须大于 等于 1 且小于等于 4。
- 更改权限
  指用 WRITE KEY 更改密钥内容的权限,在满足该条件时可使用 WRITE KEY 更改密钥内容，但不能改变错误计数器的值。
- 错误计数器

  高半字节指出密钥可以连续错误的最多次数，低半字节指出还可以再 试的次数。如果连续错误错超过规定的次数，密钥自动被锁死。

  例如：错误计数器的值为 33，表示该密钥最多可以连续错误 3 次，若输错一次则其值变为 32，再错一次之后变为 31，若下次核对或认证正确则该值变为 33。使用解锁口令时，解锁口令正确后错误次数低半字节被设置成高半字节值，同时口令被修改。解锁口令若错误，解锁口令允许再试次数减一，解锁口令和外部认证密钥锁死后无法被解锁。

- 后续状态
  当口令核对成功或外部认证成功后，置安全状态寄存器值为后续状态的低半字。
- 密钥版本号和算法标识由用户自己定义。

26.Create File（建立文件）

定义与范围

Create File 命令用于建立文件系统。请参见“3.4 专用文件、3.5 工作基本文件和 3.6 内部基本文件”。

注意事项

- 在满足当前 DF 的建立权限时，可用此命令建立 DF 或 EF。
- 每个 DF 下只能有一个 KEY 文件，且必须最先被建立。
- 当前 DF 被擦除后，则在该 DF 下可任意建立文件和读写文件而不受文件访问权限的限制，一旦离开当前 DF 再进入 DF 时，将遵循文件的访问权限。
- 目录文件建立后不能自动被选择（MF 除外），需使用 Select File 命令选择。
- 私钥文件的权限在任何时候都不可读写。

命令报文

方式一　表 26.1 Create File 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	E0	-
P1P2	2	XXXX	文件标识（FileID）
Lc	1	XX	-
DATA	XX	XX…XX	文件控制信息（和 DF 名称）
Le	-	-	不存在

注：MF 的文件标识符必须是‘3F00’；KEY 文件的文件标识符必须是‘0000’；

方式二 表 26.2 Create File 命令报文编码

代码	长度 (byte)	值 (Hex)	描述
CLA	1	80	-
INS	1	E0	-
P1P2	2	0000	-
Lc	1	XX	数据长度
DATA	XX	XX…XX	文件控制信息（和 DF 名称）
Le	-	-	不存在

注：MF 的文件标识符必须是‘3F00’； KEY 文件的文件标识符必须是‘0000’；

命令报文数据域

命令数据域中的所有权限设置请参见“5. TimeCOS®/PK 的安全体系”。

主文件（MF）

- P1 P2 参数固定为‘3F 00’

表 26.3 MF 的文件控制信息

数据域	文件类型	文件空间	建立权限	擦除权限	8 字节传输代码
长度（byte）	1	2	1	1	8
值（HEX）	38	FFFF	XX	XX	FFFFFFFFFFFFFFFF

专用文件（DF）

表 26.4 DF 的文件控制信息

DATA	文件类型	文件空间	建立权限	擦除权限	保留字	DF 名称（可选）
长度(byte)	1	2	1	1	3	5~16
值（HEX）	38	XXXX	XX	XX	FFFFFF	DF 名称

基本文件（EF）

基本文件控制信息内容如下表所示。

表 26.5 EF 的文件控制信息

数据域 / 文件类型	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
二进制文件	‘28’	文件空间		读权限	写权限	‘FF’	KID，见说明[2]
定长记录文件	‘2A’	2≤记录数≤254	记录长度≤255	读权限	写权限	‘FF’	KID，见说明[2]
循环记录文件	‘2E’	2≤记录数≤254	记录长度≤255	读权限	写权限	‘FF’	KID，见说明[2]
变长记录文件	‘2C’	文件空间=所有记录长度和 每条记录=记录长度+1 字节校验码（由 COS 计算）		读权限	写权限	‘FF’	KID，见说明[2]
普通钱包文件	‘2F’	2≤记录数≤254	记录长度≤7	读/扣款权限	存款权限	‘FF’	KID，见说明[2
私钥文件	‘3D’	文件空间≥330		使用权限	任意无用字节	任意无用字节	KID，见说明[2]
公钥文件	‘3E’	文件空间≥135		使用权限	更改权限	读权限	KID，见说明[2]

密钥文件	‘3F’	文件空间=所有密钥记录长度之和+5 字节保留空间 每条记录的计算方法见说明[4]	当前 DF 文件短标识符 见说明[4]	增加权限	‘FF’	‘FF’

表 26.6 二进制、定长记录、变长记录和循环记录文件控制信息-Lc=0x0C （方式二）

长度	代码含义	备注
2	文件标识	
1	保留 ”0x00”	
1	初始值	
1	文件类别 28、2A、2C、2E	
2	文件空间(个数*长度)	
1	SFI	注 1
1	KID	注 2
1	读权	
1	改权	
1	保留 ”0x00”	

说明：

[1] 二进制文件、定长记录文件、变长记录文件、循环文件、普通钱包文件、公钥文件（密钥文件除外）都可以采用安全报文传送。

如对上述文件进行安全报文传送，只需在建立文件时改变文件类型字节高两位即可。

基本文件数据域 Byte1（文件类型）定义如下：

b7	b6	b5	b4	b3	b2	b1	b0	线路保护方式
0	**0**	**文件类型**						**无**
1	**0**	**文件类型**						**MAC**
1	**1**	**文件类型**						**DES&MAC**

[例] 建立文件时若需进行加密线路保护则将文件类型高两位置 1，如二进制类型由 28 变为 E8。

◆ ***注意：具有线路保护属性的文件，在进行写操作时必须使用相应的线路保护模式***

[2] 对 KID 的说明

表 26.7 基本文件 KID 说明

B7	保留为：1
B6	文件写位置　　1：文件处于基本 EEPROM 中，即前 32K 0：文件在扩充 EEPROM 中，即除前 32K 以外的空间中
b5	断电保护机制　1：写操作时无断电保护机制 0：写操作时有断电保护机制
b4	读文件方式　　1：返回的数据为明文 0：返回的数据为密文
b3	读文件返回数据加密时使用的密钥标识:
b2	KID 取反
b1	写文件发送数据加密时使用的密钥标识:
b0	KID 取反

注：对于 TimeCOS®而言，前 32K 和后面的空间是独立管理的。每个文件都由文件头和文件体组成，卡片中，所有文件的文件头都在前 32K 空间。而文件体要么全部位于前 32K，要么全部位于后面的独立空间中，不能跨越。后面的空间中能建立二进制文件、定长记录文件，且文件大小不超过 32K。

[3] KEY 文件

注：KEY 文件标识符必须是‘0000’。

1）每条记录长度=1 字节 TAG+1 字节的长度+5 字节的密钥头+密钥值的长度。

记录中的 T、L 字节由 COS 维护。

注：对于连接 MF 下密钥的 KEY 记录，则

记录长度=1 字节 TAG+1 字节的长度+1 字节密钥类型。

记录中的 T、L 字节由 COS 维护。

DF 文件短标识符如下表所示。

表 26.8　DF 短文件标识符

b7	b6	b5	b4	b3	b2	b1	b0	描述
0	0	0	X	X	X	X	X	当前 DF 为 DDF，低 5 位为 DDF 下目录基本文件的短文件标识符。
1	0	0	X	X	X	X	X	当前 DF 为 ADF，低 5 位为发卡方专用数据文件的短文件标识符。
1	1	0	X	X	X	X	X	包含当前 DF 的 A5 模板的短文件标识符
1	1	1	1	1	1	1	1	保留值

注：‘A5’为文件控制信息专用模板的记录标识。

[4]：数据文件初始值

二进制文件、定长记录文件、变长记录文件、循环文件（密钥文件除外）建立后，文件的初始值为 0x00 或 0xFF。

响应报文数据域

响应报文数据域不存在。

响应报文状态码

IC 卡可能回送的状态码如下所示：

表 26.9 Create File 命令响应状态码

SW1	SW2	意义
90	00	命令成功执行
67	00	错误的长度
69	82	建立权限不满足
6A	80	记录个数小于 2 或目录级数超过三级
6A	84	文件无足够空间
6A	86	文件已存在

应用举例

例一：建 0015 文件，数据域长度为 8 字节，随意读写，读写位置在前 32K，带保护写，密文读，读取密钥为 00 的，写密钥为 00 的 36 密钥

指令：80E0001507280008F0F0FFCF

返回：9000

例二：建立钱包文件，标识 0002，类别 2F，允许消费写明细，存权 F0、扣权 F0，限额 FFFFFF

指令：80E000001E000200002F020842FFF0F000F0F000FFFFFFFF00000000000000000000000000

交易记录文件默认为 0018 文件

返回：9000