

# 电动读卡机

## ACT-A6

REV: A1    2013 年 12 月

Advance Card Technology Co., Ltd.  
©深圳市驰卡技术有限公司 2011—2013  
版权所有

# 说明

该文档包含 ACT-A6 读卡器 API 的说明。

ACT-A6            开发包提供以下相关文件供开发使用：  
A6CRTAPI.h        头文件  
A6CRTAPI.lib       A6CRTAPI.DLL 的导入库  
A6CRTAPI.dll       32 位动态链接库。

## API 参考

### 基本操作函数

#### A6\_Connect

A6\_Connect 在调用程序和读卡器间建立一个连接。

```
LONG  
WINAPI  
A6_Connect  
(  
    __in    DWORD    dwPort,  
    __in    DWORD    dwSpeed,  
    __out   PREADERHANDLE phReader  
);
```

#### 参数

dwPort            COM 端口号。可用值：1 ~ 256.

dwSpeed      波特率。可用值：  
1200  
2400  
4800  
9600  
19200  
38400

phReader      返回一个标识读卡器连接的句柄。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Disconnect

断开调用程序和读卡器间的连接。

```
LONG  
WINAPI  
A6_Disconnect  
(  
    __in    READERHANDLE hReader  
);
```

## 参数

hReader      引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Initialize

复位读卡器。

LONG

```
WINAPI
A6_Initialize
(
    __in    READERHANDLE hReader,
    __in    BYTE         bResetMode,
    __out   PBYTE        pbVerBuff,
    __inout PDWORD       pcbVerLength
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bResetMode	复位模式。可用值： RESET_ONLY            仅复位，不移动卡片。 RESET_AND_EJECT      复位并移动读卡器内的卡片到前端持卡位。 RESET_AND_CAPTURE    复位并移动读卡器内的卡片到后端持卡位。
pbVerBuff	接收固件版本信息的缓冲区。
pcbVerLength	提供 pbVerBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## 示例

```
BYTE VerBuff[128];
DWORD VerLength = sizeof(VerBuff);

LONG lResult = A6_Initialize(hReader, RESET_ONLY, VerBuff, &VerLength);
```

## A6\_SetCardIn

设置读卡器前端和后端的进卡方式。

```
LONG
WINAPI
A6_SetCardIn
(
```

```
__in    READERHANDLE hReader,  
__in    BYTE         bFrontSet,  
__in    BYTE         bRearSet  
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。	
bFrontSet	前端进卡方式。可用值：	
	FCI_PROHIBITED	禁止进卡。
	FCI_MAGCARD_ONLY	仅允许进磁卡。
	FCI_ALLOWED	允许进卡。
bRearSet	后端进卡方式。可用值：	
	RCI_ALLOWED	允许进卡。
	RCI_PROHIBITED	禁止进卡。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SetDockedPos

设置持卡位置。

```
LONG  
WINAPI  
A6_SetDockedPos  
(  
    __in    READERHANDLE hReader,  
    __in    BYTE         bDockedPos  
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。	
bDockedPos	持卡位置。可用值：	
	DPOS_FRONT_NH	前端不持卡位。
	DPOS_FRONT	前端持卡位。
	DPOS_INTERNAL	读卡器内部。

DPOS_IC_POS	IC 卡位。
DPOS_REAR	后端持卡位。
DPOS_REAR_NH	后端不持卡位。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SetBaudRate

设置波特率。

```
LONG
WINAPI
A6_SetBaudRate
(
    __in  READERHANDLE hReader,
    __in  DWORD dwBaudRate
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
dwBaudRate	波特率。可用值： 1200 2400 4800 9600 19200 38400

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_GetCRCondition

获取读卡器的基本状态。

```
LONG
WINAPI
A6_GetCRCondition
(
    __in    READERHANDLE hReader,
    __out   PCRSTATUS    pStatus
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pStatus 指向 CRSTATUS 结构的指针。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_GetSensorStatus

获取传感器的基本状态。

```
LONG
WINAPI
A6_GetSensorStatus
(
    __in    READERHANDLE hReader,
    __out   BYTE          (&bStatus) [NUM_SENSORS]
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bStatus 接收状态信息的字节数组的引用，长度为 NUM\_SENSORS 个字节。  
数组中依次存放 PSS0 至 PSS5、CTSW、KSW 共 8 个传感器的识别状态。

对于 PSS0 至 PSS5，若对应的值为 0x31，表示有卡；值为 0x30 则表示无卡。

CTSW 的值为 0x31，表示闸门已打开；值为 0x30，表示闸门已关闭。

KSW 的值为 0x31，表示检测到卡片插入信号；值为 0x30，表示未检测到。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_GetSensorVoltages



获取传感器的电压信息。

```
LONG
WINAPI
A6_GetSensorVoltages
(
    __in    READERHANDLE hReader,
    __out   float (&fVoltages) [NUM_SENSORS]
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

fVoltages 接收传感器电压信息数组的引用。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_DetectIccType

检测读卡器内的IC卡的类型。

```
LONG
WINAPI
A6_DetectIccType
(
    __in    READERHANDLE hReader,
    __out   PBYTE    pbType
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbType 指向 BYTE 变量的指针，用于返回 IC 卡的类型。不可以为 NULL。  
可用值：  
ICCTYPE\_UNKNOWN

```
ICCTYPE_MIFARE_1K
ICCTYPE_T0_CPU
ICCTYPE_T1_CPU
ICCTYPE_AT24C01
ICCTYPE_AT24C02
ICCTYPE_AT24C04
ICCTYPE_AT24C08
ICCTYPE_AT24C16
ICCTYPE_AT24C32
ICCTYPE_AT24C64
ICCTYPE_SLE4442
ICCTYPE_SLE4428
ICCTYPE_AT88SC102
ICCTYPE_AT88SC1604
ICCTYPE_AT88SC1608
ICCTYPE_AT45DB041
```

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_MoveCard

移动读卡器内的卡片。

```
LONG
WINAPI
A6_MoveCard
(
    __in    READERHANDLE hReader,
    __in    BYTE         bMoveMethod
);
```

## 参数

hReader            引用 A6\_Connect 返回的句柄值。

bMoveMethod	移动方式。可用值：	
	MOVE_TO_FRONT_NH	移动到前端不持卡位。
	MOVE_TO_FRONT	移动到前端持卡位。
	MOVE_TO_RF_POS	移动到射频位。
	MOVE_TO_IC_POS	移动到IC卡位。
	MOVE_TO_REAR	移动到后端持卡位。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_LedOn

点亮 LED。

```
LONG
WINAPI
A6_LedOn
(
    __in    READERHANDLE hReader
);
```

## 参数

hReader                    引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_LedOff

熄灭 LED。

```
LONG
WINAPI
A6_LedOff
(
    __in    READERHANDLE hReader
);
```

## 参数

hReader                    引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_LedBlink

闪烁 LED。

```
LONG
WINAPI
A6_LedBlink
(
    __in    READERHANDLE hReader,
    __in    BYTE    bOnTime,
    __in    BYTE    bOffTime
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bOnTime 亮指示灯的时间值（实际的时间为 bOnTime X 0.25 秒）。

bOffTime 灰指示灯的时间值（实际的时间为 bOffTime X 0.25 秒）。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_IccPowerOn

IC卡上电。

```
LONG
WINAPI
A6_IccPowerOn
(
    __in    READERHANDLE    hReader
);
```

## 参数

hReader                    引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_IccPowerOff

IC卡下电。

```
LONG
WINAPI
A6_IccPowerOff
(
    __in    READERHANDLE    hReader
);
```

## 参数

hReader                    引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## 接触式 CPU 卡操作函数

## A6\_CpuColdReset

CPU 卡冷复位。

```
LONG
WINAPI
A6_CpuColdReset
(
    __in    READERHANDLE hReader,
```

```
__out  PBYTE  pbATRBuff,  
__inout PDWORD pcbATRLength  
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbATRBuff 指向返回的复位信息。如果不为 NULL，pcbATRLength 也不可以为 NULL。

pcbATRLength 提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_CpuWarmReset

CPU 卡热复位。

```
LONG  
WINAPI  
A6_CpuWarmReset  
(  
__in  READERHANDLE hReader,  
__out PBYTE  pbATRBuff,  
__inout PDWORD pcbATRLength  
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbATRBuff 指向返回的复位信息。如果不为 NULL，pcbATRLength 也不可以为 NULL。

pcbATRLength 提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_CpuTransmit

CPU 卡数据传输。

LONG

```
WINAPI  
A6_CpuTransmit  
(  
    __in    READERHANDLE hReader,  
    __in    BYTE        bProtocol,  
    __in    PBYTE        pbSendBuff,  
    __in    USHORT       cbSendLength,  
    __out   PBYTE        pbRecvBuff,  
    __inout PDWORD       pcbRecvLength  
);
```

### 参数

hReader	引用 A6_Connect 返回的句柄值。
bProtocol	CPU 卡的通信协议类型。可用值： ICC_PROTOCOL_T0      T = 0 协议 ICC_PROTOCOL_T1      T = 1 协议
pbSendBuff	指向要写入到卡片的数据。不可以为 NULL。
cbSendLength	提供 pbSendBuff 参数的长度（字节数）。
pbRecvBuff	指向返回的数据。不可以为 NULL。
pcbRecvLength	提供 pbRecvBuff 参数的长度（字节数）并接收读卡器实际返回的长度。不可以为 NULL。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## 磁卡操作函数

### A6\_ReadTracks

读磁卡。

```
LONG
WINAPI
A6_ReadTracks
(
    __in    READERHANDLE hReader,
    __in    BYTE         bMode,
    __in    UINT         iTrackID,
    __out   PTRACKINFO  pTrackInfo
);
```

#### 参数

hReader	引用 A6_Connect 返回的句柄值。
bMode	读卡模式。可用值： READ_ASCII READ_BINARY
iTrackID	要读的磁道号。下列值的一个或多个： TRACKID_IS01      磁道1 TRACKID_IS02      磁道2 TRACKID_IS03      磁道3 TRACKID_ALL        所有磁道
pTrackInfo	指向 TRACKINFO 变量的指针，用于返回读取的磁道信息。不可以为 NULL。

#### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。



## SAM 卡操作函数

### A6\_SamActivate

激活SAM卡。

LONG

```
WINAPI
A6_SamActivate
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSAMNumber,
    __in    BYTE         bVoltage,
    __out   PBYTE        pbATRBuff,
    __inout PDWORD       pcbATRLength
);
```

### 参数

- hReader            引用 A6\_Connect 返回的句柄值。
- bSAMNumber        SAM 卡号。可用值：0~5。
- bVoltage           SAM 卡上电时使用的电压。可用值：  
VOLTAGE\_1\_8       1.8V  
VOLTAGE\_3          3.0V  
VOLTAGE\_5          5.0V
- pbATRBuff         指向返回的复位信息。如果不为 NULL，pcbATRLength 也不可以为 NULL。
- pcbATRLength      提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

### A6\_SamDeactivate

SAM 卡下电。

```
LONG
WINAPI
A6_SamDeactivate
(
    __in    READERHANDLE hReader
);
```

## 参数

hReader                      引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SamTransmit

SAM卡数据传输。

```
LONG
WINAPI
A6_SamTransmit
(
    __in    READERHANDLE hReader,
    __in    BYTE         bProtocol,
    __in    BYTE         bSAMNumber,
    __in    PBYTE        pbSendBuff,
    __in    USHORT       cbSendLength,
    __out   PBYTE        pbRecvBuff,
    __inout PDWORD       pcbRecvLength
);
```

## 参数

hReader                      引用 A6\_Connect 返回的句柄值。

bSAMNumber                  SAM 卡号。可用值：0~5。

bProtocol                   SAM 卡的通信协议类型。可用值：  
ICC\_PROTOCOL\_T0            T = 0 协议  
ICC\_PROTOCOL\_T1            T = 1 协议

pbSendBuff                  指向要写入到卡片的数据。不可以为 NULL。

cbSendLength	提供 pbSendBuff 参数的长度（字节数）。
pbRecvBuff	指向返回的数据。不可以为 NULL。
pcbRecvLength	提供 pbRecvBuff 参数的长度（字节数）并接收读卡器实际返回的长度。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## I2C 内存卡操作

### A6\_I2cSelect

选择当前 I2C 卡的类型。

```
LONG
WINAPI
A6_I2cSelect
(
    __in    READERHANDLE hReader,
    __in    BYTE         bCardType
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bCardType	I2C 卡的类型。可用值： ICCTYPE_AT24C01 ICCTYPE_AT24C02 ICCTYPE_AT24C04 ICCTYPE_AT24C08 ICCTYPE_AT24C16 ICCTYPE_AT24C32 ICCTYPE_AT24C64

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_I2cRead

读I2C内存卡。

```
LONG
WINAPI
A6_I2cRead
(
    __in    READERHANDLE hReader,
    __in    BYTE        bCardType,
    __in    WORD         wAddress,
    __in    BYTE         bBytesToRead,
    __out    PBYTE       pbBuffer,
    __inout  PDWORD      pcbLength
);
```

### 参数

hReader	引用 A6_Connect 返回的句柄值。
bCardType	I2C 卡的类型。可用值： ICCTYPE_AT24C01 ICCTYPE_AT24C02 ICCTYPE_AT24C04 ICCTYPE_AT24C08 ICCTYPE_AT24C16 ICCTYPE_AT24C32 ICCTYPE_AT24C64
wAddress	要操作的地址。
bBytesToRead	要读取的字节数。
pbBuffer	指向返回的数据。不可以为 NULL。
pcbLength	提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。不可以为 NULL。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_I2cWriteWithoutVerification

不带校验写卡。

```
LONG
WINAPI
A6_I2cWriteWithoutVerification
(
    __in    READERHANDLE hReader,
    __in    BYTE        bCardType,
    __in    WORD        wAddress,
    __in    BYTE        bBytesToWrite,
    __in    PBYTE       pbBuffer
);
```

### 参数

hReader	引用 A6_Connect 返回的句柄值。
bCardType	I2C 卡的类型。可用值： ICCTYPE_AT24C01 ICCTYPE_AT24C02 ICCTYPE_AT24C04 ICCTYPE_AT24C08 ICCTYPE_AT24C16 ICCTYPE_AT24C32 ICCTYPE_AT24C64
wAddress	要操作的地址。
bBytesToWrite	要写的字节数。
pbBuffer	指向要写入到卡片的数据。不可以为 NULL。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_I2cWriteWithVerification

带校验写卡。

LONG

```
WINAPI
A6_I2cWriteWithVerification
(
    __in    READERHANDLE hReader,
    __in    BYTE    bCardType,
    __in    WORD     wAddress,
    __in    BYTE     bBytesToWrite,
    __in    PBYTE    pbBuffer
);
```

### 参数

hReader	引用 A6_Connect 返回的句柄值。
bCardType	I2C 卡的类型。可用值： ICCTYPE_AT24C01 ICCTYPE_AT24C02 ICCTYPE_AT24C04 ICCTYPE_AT24C08 ICCTYPE_AT24C16 ICCTYPE_AT24C32 ICCTYPE_AT24C64
wAddress	要操作的地址。
bBytesToWrite	要写的字节数。
pbBuffer	指向要写入到卡片的数据。不可以为 NULL。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## SLE4442 卡操作函数

### A6\_Sle4442Reset

复位 SLE4442 卡。

```
LONG
WINAPI
A6_Sle4442Reset
(
    __in    READERHANDLE hReader,
    __out   PBYTE        pbATRBuff,
    __inout PDWORD       pcbATRLength
);
```

#### 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbATRBuff 指向返回的复位信息。如果不为 NULL，pcbATRLength 也不可以为 NULL。

pcbATRLength 提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

#### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

### A6\_Sle4442ReadMainMemory

读主存储区。

```
LONG
WINAPI
A6_Sle4442ReadMainMemory
(
    __in    READERHANDLE hReader,
    __in    BYTE         bAddress,
    __in    BYTE         bBytesToRead,
    __out   PBYTE        pbBuffer,
    __inout PDWORD       pcbLength
);
```

);

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bAddress	要操作的地址。
bBytesToRead	要读取的字节数。
pbBuffer	指向返回的数据。不可以为 NULL。
pcbLength	提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4442ReadProtectionMemory

读保护存储区。

```
LONG
WINAPI
A6_Sle4442ReadProtectionMemory
(
    __in    READERHANDLE hReader,
    __out   BYTE          (&bBuffer)[32]
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bBuffer	引用接收返回数据的字节数组。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。



## A6\_Sle4442ReadSecurityMemory

读安全存储区。

```
LONG
WINAPI
A6_Sle4442ReadSecurityMemory
(
    __in    READERHANDLE hReader,
    __out   BYTE         (&bBuffer) [4]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bBuffer 引用接收返回数据的字节数组。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4442WriteMainMemory

写主存储区。

```
LONG
WINAPI
A6_Sle4442WriteMainMemory
(
    __in    READERHANDLE hReader,
    __in    BYTE         bAddress,
    __in    BYTE         bBytesToWrite,
    __in    PBYTE        pbBuffer
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bAddress 要操作的地址。

bBytesToWrite 要写的字节数。

pbBuffer 指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4442WriteProtectionMemory

写保护存储区。

```
LONG
WINAPI
A6_Sle4442WriteProtectionMemory
(
    __in    READERHANDLE hReader,
    __in    BYTE         bAddress,
    __in    BYTE         bBytesToWrite,
    __in    PBYTE        pbBuffer
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bAddress 要操作的地址。

bBytesToWrite 要写的字节数。

pbBuffer 指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4442VerifyPSC

校验安全码。

```
LONG
WINAPI
A6_Sle4442VerifyPSC
(
    __in    READERHANDLE hReader,
    __in    BYTE         bPSCByte1,
    __in    BYTE         bPSCByte2,
    __in    BYTE         bPSCByte3
);
```

## 参数

hReader            引用 A6\_Connect 返回的句柄值。

bPSCByte1        安全码字节1。

bPSCByte2        安全码字节2。

bPSCByte3        安全码字节3。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4442UpdatePSC

更新安全码。

```
LONG
WINAPI
A6_Sle4442UpdatePSC
(
    __in    READERHANDLE hReader,
    __in    BYTE         bPSCByte1,
    __in    BYTE         bPSCByte2,
    __in    BYTE         bPSCByte3
);
```

## 参数

hReader            引用 A6\_Connect 返回的句柄值。

---

bPSCByte1	安全码字节1。
bPSCByte2	安全码字节2。
bPSCByte3	安全码字节3。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## SLE4428 卡操作函数

### A6\_Sle4428Reset

复位 SLE4428 卡。

```
LONG
WINAPI
A6_Sle4428Reset
(
    __in    READERHANDLE hReader,
    __out   PBYTE        pbATRBuff,
    __inout PDWORD        pcbATRLength
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
pbATRBuff	指向返回的复位信息。如果不为 NULL，pcbATRLength 也不可以为 NULL。
pcbATRLength	提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4428ReadWithoutPB

不带保护位读卡。

```
LONG
WINAPI
A6_Sle4428ReadWithoutPB
(
    __in    READERHANDLE hReader,
    __in    WORD          wAddress,
    __in    BYTE          bBytesToRead,
    __out   PBYTE         pbBuffer,
    __inout PDWORD        pcbLength
);
```

### 参数

hReader	引用 A6_Connect 返回的句柄值。
wAddress	要操作的地址。
bBytesToRead	要读取的字节数。
pbBuffer	指向返回的数据。不可以为 NULL。
pcbLength	提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4428ReadProtectionBits

读保护位。

```
LONG
WINAPI
A6_Sle4428ReadProtectionBits
(
    __in    READERHANDLE hReader,
    __in    WORD          wAddress,
    __in    BYTE          bBytesToRead,
```

```
__out  PBYTE  pbBuffer,  
__inout PDWORD pcbLength  
);
```

## 参数

**hReader**                引用 A6\_Connect 返回的句柄值。

**wAddress**             要操作的地址。

**bBytesToRead**        要读取的字节数。

**pbBuffer**             指向返回的数据。不可以为 NULL。

**pcbLength**           提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4428WriteWithoutPB

不带保护位写卡。

```
LONG  
WINAPI  
A6_Sle4428WriteWithoutPB  
(  
__in  READERHANDLE hReader,  
__in  WORD          wAddress,  
__in  BYTE          bBytesToWrite,  
__in  PBYTE         pbBuffer  
);
```

## 参数

**hReader**                引用 A6\_Connect 返回的句柄值。

**wAddress**             要操作的地址。

**bBytesToWrite**        要写的字节数。

**pbBuffer**             指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4428WriteWithPB

带写保护位写卡。

```
LONG
WINAPI
A6_Sle4428WriteWithPB
(
    __in    READERHANDLE hReader,
    __in    WORD          wAddress,
    __in    BYTE          bBytesToWrite,
    __in    PBYTE         pbBuffer
);
```

## 参数

**hReader**            引用 A6\_Connect 返回的句柄值。

**wAddress**           要操作的地址。

**bBytesToWrite**      要写的字节数。

**pbBuffer**           指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4428VerifyPSC

校验安全码。

```
LONG
WINAPI
A6_Sle4428VerifyPSC
```

```
(  
    __in    READERHANDLE hReader,  
    __in    BYTE        bPSCByte1,  
    __in    BYTE        bPSCByte2  
);
```

## 参数

hReader            引用 A6\_Connect 返回的句柄值。

bPSCByte1         安全码字节1。

bPSCByte2         安全码字节2。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sle4428UpdatePSC

更新安全码。

```
LONG  
WINAPI  
A6_Sle4428UpdatePSC  
(  
    __in READERHANDLE hReader,  
    __in  BYTE        bOldPSCByte1,  
    __in  BYTE        bOldPSCByte2,  
    __in  BYTE        bNewPSCByte1,  
    __in  BYTE        bNewPSCByte2  
);
```

## 参数

hReader            引用 A6\_Connect 返回的句柄值。

bOldPSCByte1       原安全码字节1。

bOldPSCByte2       原安全码字节2。

bNewPSCByte1       新安全码字节1。



bNewPSCByte2      新安全码字节2。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## AT45DB041 卡操作函数

### A6\_Db041Reset

复位 AT45DB041 卡。

```
LONG  
WINAPI  
A6_Db041Reset  
(  
    __in    READERHANDLE hReader  
);
```

## 参数

hReader      引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

### A6\_Db041ReadOnePage

读卡。

```
LONG  
WINAPI  
A6_Db041ReadOnePage  
(  
    __in    READERHANDLE hReader,  
    __in    WORD        wAddress,  
    __out   BYTE        (&bBuffer)[264])
```

);

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bBuffer 引用接收返回数据的字节数组。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Db041WriteOnePage

写卡。

```
LONG
WINAPI
A6_Db041WriteOnePage
(
    __in    READERHANDLE hReader,
    __in    WORD          wAddress,
    __in    BYTE          (&bBuffer)[264]
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

wAddress 要操作的地址。

bBuffer 要写入到卡片的数据。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## AT88SC102 卡操作函数

### A6\_Sc102Reset

复位 AT88SC102 卡。

```
LONG  
WINAPI  
A6_Sc102Reset  
(  
    __in    READERHANDLE hReader  
);
```

#### 参数

hReader            引用 A6\_Connect 返回的句柄值。

#### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

### A6\_Sc102VerifySC

校验主安全码。

```
LONG  
WINAPI  
A6_Sc102VerifySC  
(  
    __in    READERHANDLE hReader,  
    __in    BYTE    bSCByte1,  
    __in    BYTE    bSCByte2  
);
```

#### 参数

hReader            引用 A6\_Connect 返回的句柄值。

bSCByte1           安全码字节1。

bSCByte2          安全码字节2。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102UpdateSC

更新主安全码。

```
LONG
WINAPI
A6_Sc102UpdateSC
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSCByte1,
    __in    BYTE         bSCByte2
);
```

## 参数

hReader          引用 A6\_Connect 返回的句柄值。

bSCByte1          安全码字节1。

bSCByte2          安全码字节2。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102ReadMemory

读存储区。

```
LONG
WINAPI
A6_Sc102ReadMemory
(
    __in    READERHANDLE hReader,
```

```
__in    BYTE    bAddress,  
__in    BYTE    bBytesToRead,  
__out   PBYTE    pbBuffer,  
__inout PDWORD   pcbLength  
);
```

## 参数

**hReader**            引用 A6\_Connect 返回的句柄值。

**bAddress**           要操作的地址。

**bBytesToRead**       要读取的字节数。

**pbBuffer**           指向返回的数据。不可以为 NULL。

**pcbLength**          提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102WriteMemory

写存储区。

```
LONG  
WINAPI  
A6_Sc102WriteMemory  
(  
__in    READERHANDLE hReader,  
__in    BYTE    bAddress,  
__in    BYTE    bBytesToWrite,  
__in    PBYTE    pbBuffer  
);
```

## 参数

**hReader**            引用 A6\_Connect 返回的句柄值。

**bAddress**           要操作的地址。

**bBytesToWrite**      要写的字节数。

pbBuffer           指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102EraseMemory

擦除存储区。

```
LONG
WINAPI
A6_Sc102EraseMemory
(
    __in    READERHANDLE hReader,
    __in    BYTE         bAddress,
    __in    BYTE         bBytesToErase
);
```

## 参数

hReader           引用 A6\_Connect 返回的句柄值。

bAddress         要操作的地址。

bBytesToErase    要擦除的字节数。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102EraseAZ1

擦除应用区一。

```
LONG
WINAPI
A6_Sc102EraseAZ1
(
```

```
__in    READERHANDLE hReader,  
__in    BYTE          (&bKeyBytes) [6]  
);
```

## 参数

hReader                引用 A6\_Connect 返回的句柄值。

bKeyBytes             应用区一的擦除键。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102EraseAZ2

擦除应用区二。

```
LONG  
WINAPI  
A6_Sc102EraseAZ2  
(  
__in    READERHANDLE hReader,  
__in    BOOL          fEC2Enabled,  
__in    BYTE          (&bKeyBytes) [4]  
);
```

## 参数

hReader                引用 A6\_Connect 返回的句柄值。

fEC2Enabled           指示 EC2 熔丝是否已熔断。

bKeyBytes             应用区二的擦除键。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102UpdateEZ1

更新应用区一的擦除键。

```
LONG
WINAPI
A6_Sc102UpdateEZ1
(
    __in    READERHANDLE hReader,
    __in    BYTE          (&bKeyBytes) [6]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bKeyBytes 应用区一的新擦除键。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102UpdateEZ2

更新应用区二的擦除键。

```
LONG
WINAPI
A6_Sc102UpdateEZ2
(
    __in    READERHANDLE hReader,
    __in    BYTE          (&bKeyBytes) [4]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bKeyBytes 应用区二的新擦除键。

### 返回值



如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc102Personalize

个人化。

```
LONG
WINAPI
A6_Sc102Personalize
(
    __in READERHANDLE hReader,
    __in BYTE bMode
);
```

### 参数

hReader	引用 A6_Connect 返回的句柄值。
bMode	个人化模式。可用值： PERSONALIZATION_TEST      伪个人化，可恢复 PERSONALIZATION_LOGOUT    退出伪个人化 PERSONALIZATION_REAL      真个人化，不可恢复

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## AT88SC1604 卡操作函数

### A6\_Sc1604Reset

复位 AT88SC1604 卡。

```
LONG
WINAPI
A6_Sc1604Reset
(
    __in READERHANDLE hReader
```

);

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1604VerifyPassword

校验密码。

```
LONG
WINAPI
A6_Sc1604VerifyPassword
(
    __in    READERHANDLE hReader,
    __in    BYTE          bPwdType,
    __in    BYTE          bPwdByte1,
    __in    BYTE          bPwdByte2
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bPwdType 密码类型。可用值：

PWDTYPE_SC	主安全码
PWDTYPE_SC1	应用区一安全码
PWDTYPE_EZ1	应用区一擦除键
PWDTYPE_SC2	应用区二安全码
PWDTYPE_EZ2	应用区二擦除键
PWDTYPE_SC3	应用区三安全码
PWDTYPE_EZ3	应用区三擦除键
PWDTYPE_SC4	应用区四安全码
PWDTYPE_EZ4	应用区四擦除键

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1604UpdatePassword

更新密码。

```
LONG
WINAPI
A6_Sc1604UpdatePassword
(
    __in    READERHANDLE hReader,
    __in    BYTE        bPwdType,
    __in    BYTE        bPwdByte1,
    __in    BYTE        bPwdByte2
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bPwdType 密码类型。可用值：

PWDTYPE_SC	主安全码
PWDTYPE_SC1	应用区一安全码
PWDTYPE_EZ1	应用区一擦除键
PWDTYPE_SC2	应用区二安全码
PWDTYPE_EZ2	应用区二擦除键
PWDTYPE_SC3	应用区三安全码
PWDTYPE_EZ3	应用区三擦除键
PWDTYPE_SC4	应用区四安全码
PWDTYPE_EZ4	应用区四擦除键

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1604ReadMemory

读存储区。

```
LONG
WINAPI
A6_Sc1604ReadMemory
```

```
(  
    __in    READERHANDLE hReader,  
    __in    WORD        wAddress,  
    __in    BYTE        bBytesToRead,  
    __out   PBYTE       pbBuffer,  
    __inout PDWORD      pcbLength  
);
```

## 参数

**hReader**                    引用 A6\_Connect 返回的句柄值。

**wAddress**                要操作的地址。

**bBytesToRead**            要读取的字节数。

**pbBuffer**                指向返回的数据。不可以为 NULL。

**pcbLength**              提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1604WriteMemory

写存储区。

```
LONG  
WINAPI  
A6_Sc1604WriteMemory  
(  
    __in    READERHANDLE hReader,  
    __in    WORD        wAddress,  
    __in    BYTE        bBytesToWrite,  
    __in    PBYTE       pbBuffer  
);
```

## 参数

**hReader**                    引用 A6\_Connect 返回的句柄值。

**wAddress**                要操作的地址。

bBytesToWrite 要写入的字节数。

pbBuffer 指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1604EraseMemory

擦除存储区。

```
LONG
WINAPI
A6_Sc1604EraseMemory
(
    __in    READERHANDLE hReader,
    __in    WORD          wAddress,
    __in    BYTE          bBytesToErase
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

wAddress 要操作的地址。

bBytesToErase 要擦除的字节数。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1604Personalize

个人化。

```
LONG
WINAPI
```

```
A6_Sc1604Personalize  
(  
    __in READERHANDLE hReader,  
    __in BYTE bMode  
);
```

## 参数

hReader                    引用 A6\_Connect 返回的句柄值。

bMode                    个人化模式。可用值：

PERSONALIZATION_TEST	伪个人化，可恢复
PERSONALIZATION_LOGOUT	退出伪个人化
PERSONALIZATION_REAL	真个人化，不可恢复

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## AT88SC1608 卡操作函数

### A6\_Sc1608Reset

复位 AT88SC1608 卡。

```
LONG  
WINAPI  
A6_Sc1608Reset  
(  
    __in READERHANDLE hReader  
);
```

## 参数

hReader                    引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608VerifyReadPassword

校验读密码。

LONG

WINAPI

A6\_Sc1608VerifyReadPassword

(

\_\_in READERHANDLE hReader,

\_\_in BYTE bZoneID,

\_\_in BYTE bPwdByte1,

\_\_in BYTE bPwdByte2,

\_\_in BYTE bPwdByte3

);

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bZoneID 用户区 ID。可用值：

ZONEID\_USER0 用户区0

ZONEID\_USER1 用户区1

ZONEID\_USER2 用户区2

ZONEID\_USER3 用户区3

ZONEID\_USER4 用户区4

ZONEID\_USER5 用户区5

ZONEID\_USER6 用户区6

ZONEID\_USER7 用户区7

bPwdByte1 密码字节1。

bPwdByte2 密码字节2。

bPwdByte3 密码字节3。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608VerifyWritePassword

校验写密码。

LONG

WINAPI

A6\_Sc1608VerifyWritePassword

(

\_\_in READERHANDLE hReader,

\_\_in BYTE bZoneID,

\_\_in BYTE bPwdByte1,

\_\_in BYTE bPwdByte2,

\_\_in BYTE bPwdByte3

);

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bZoneID 用户区 ID。可用值：

ZONEID_USER0	用户区0
ZONEID_USER1	用户区1
ZONEID_USER2	用户区2
ZONEID_USER3	用户区3
ZONEID_USER4	用户区4
ZONEID_USER5	用户区5
ZONEID_USER6	用户区6
ZONEID_USER7	用户区7

bPwdByte1 密码字节1。

bPwdByte2 密码字节2。

bPwdByte3 密码字节3。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608ReadMemory

读存储区。

LONG

WINAPI

A6\_Sc1608ReadMemory



```
(  
    __in    READERHANDLE hReader,  
    __in    BYTE        bZoneID,  
    __in    BYTE        bAddress,  
    __in    BYTE        bBytesToRead,  
    __out   PBYTE        pbBuffer,  
    __inout PDWORD       pcbLength  
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bZoneID	存储区 ID。可用值： ZONEID_USER0      用户区0 ZONEID_USER1      用户区1 ZONEID_USER2      用户区2 ZONEID_USER3      用户区3 ZONEID_USER4      用户区4 ZONEID_USER5      用户区5 ZONEID_USER6      用户区6 ZONEID_USER7      用户区7 ZONEID_CONFIG      配置区
bAddress	要操作的地址。
bBytesToRead	要读取的字节数。
pbBuffer	指向返回的数据。不可以为 NULL。
pcbLength	提供 pbBuffer 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608WriteMemory

写存储区。

```
LONG  
WINAPI  
A6_Sc1608WriteMemory
```

```
(  
    __in    READERHANDLE hReader,  
    __in    BYTE        bZoneID,  
    __in    BYTE        bAddress,  
    __in    BYTE        bBytesToWrite,  
    __in    PBYTE       pbBuffer  
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bZoneID	存储区 ID。可用值： ZONEID_USER0      用户区0 ZONEID_USER1      用户区1 ZONEID_USER2      用户区2 ZONEID_USER3      用户区3 ZONEID_USER4      用户区4 ZONEID_USER5      用户区5 ZONEID_USER6      用户区6 ZONEID_USER7      用户区7 ZONEID_CONFIG      配置区
bAddress	要操作的地址。
bBytesToWrite	要写入的字节数。
pbBuffer	指向要写入到卡片的数据。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608ReadFuses

读熔丝状态。

```
LONG  
WINAPI  
A6_Sc1608ReadFuses  
(  
    __in    READERHANDLE hReader,  
    __out    PBYTE       pbFAB,
```

```
__out PBYTE pbCMA,  
__out PBYTE pbPER  
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbFAB 接收返回的 FAB 熔丝的状态。可用值：  
FUSE\_BURNED 已熔断  
FUSE\_UNBURED 未熔断

pbCMA 接收返回的 CMA 熔丝的状态。

pbPER 接收返回的 PER 熔丝的状态。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608WriteFuses

写熔丝。

```
LONG  
WINAPI  
A6_Sc1608WriteFuses  
(  
__in READERHANDLE hReader  
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608InitAuth

初始化认证。

```
LONG
WINAPI
A6_Sc1608InitAuth
(
    __in    READERHANDLE hReader,
    __in    BYTE          (&bRandomNumberBytes)[8]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bRandomNumberBytes 认证字节数组。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_Sc1608VerifyAuth

校验认证。

```
LONG
WINAPI
A6_Sc1608VerifyAuth
(
    __in    READERHANDLE hReader,
    __in    BYTE          (&bChallengeBytes)[8]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bChallengeBytes 校验字节数组。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## Mifare S50/S70 操作函数

### A6\_SxxSelect

寻 Mifare 1K 卡。

```
LONG
WINAPI
A6_SxxSelect
(
    __in    READERHANDLE    hReader
);
```

#### 参数

hReader                    引用 A6\_Connect 返回的句柄值。

#### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

### A6\_SxxGetUID

读卡序列号。

```
LONG
WINAPI
A6_SxxGetUID
(
    __in    READERHANDLE hReader,
    __out   PBYTE        pbUIDBuff,
    __inout PDWORD       pcbUIDLength
);
```

#### 参数

**hReader**                引用 A6\_Connect 返回的句柄值。

**pbUIDBuff**            指向返回的序列号信息。不可以为 NULL。

**pcbUIDLength**        提供 pbUIDBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SxxVerifyPassword

校验密钥。

```
LONG
WINAPI
A6_SxxVerifyPassword
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BOOL         bWithKeyA,
    __in    BYTE         (&bKeyBytes)[6]
);
```

## 参数

**hReader**                引用 A6\_Connect 返回的句柄值。

**bSectorNumber**        要认证的扇区号。

**bWithKeyA**            是否使用 Key A。

**bKeyBytes**            密钥字节数组。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SxxUpdatePassword

更新密钥。

```
LONG
WINAPI
A6_SxxUpdatePassword
(
    __in    READERHANDLE hReader,
    __in    BYTE        bSectorNumber,
    __in    BYTE        (&bKeyBytes) [6]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 扇区号。

bKeyBytes 密钥字节数组。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SxxReadBlock

读某个块。

```
LONG
WINAPI
A6_SxxReadBlock
(
    __in    READERHANDLE hReader,
    __in    BYTE        bSectorNumber,
    __in    BYTE        bBlockNumber,
    __out   BYTE        (&bBuffer) [16]
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 要读的扇区号。

bBlockNumber 要读的块号。

bBuffer 接收返回数据的缓冲区。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SxxWriteBlock

写某个块。

```
LONG
WINAPI
A6_SxxWriteBlock
(
    __in    READERHANDLE hReader,
    __in    BYTE          bSectorNumber,
    __in    BYTE          bBlockNumber,
    __in    BYTE          (&bBuffer)[16]
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 要读的扇区号。

bBlockNumber 要读的块号。

bBuffer 要写入到卡片的数据。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。



## A6\_S50InitializeValue

初始化S50卡的某个值块。

```
LONG
WINAPI
A6_S50InitializeValue
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BYTE         bBlockNumber,
    __in    UINT          uValue
);
```

### 参数

hReader            引用 A6\_Connect 返回的句柄值。

bSectorNumber    要读的扇区号。

bBlockNumber     要读的块号。

uValue            要写入的值。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_S70InitializeValue

初始化S50卡的某个值块。

```
LONG
WINAPI
A6_S70InitializeValue
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BYTE         bBlockNumber,
    __in    UINT          uValue
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bSectorNumber	要读的扇区号。
bBlockNumber	要读的块号。
uValue	要写入的值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_S50ReadValue

读S50卡的值块。

```
LONG
WINAPI
A6_S50ReadValue
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BYTE         bBlockNumber,
    __out   PUINT        puValue
);
```

## 参数

hReader	引用 A6_Connect 返回的句柄值。
bSectorNumber	要读的扇区号。
bBlockNumber	要读的块号。
puValue	指向返回的值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_S70ReadValue

读S70卡的值块。

```
LONG
WINAPI
A6_S70ReadValue
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BYTE         bBlockNumber,
    __out   PUINT        puValue
);
```

### 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 要读的扇区号。

bBlockNumber 要读的块号。

puValue 指向返回的值。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SxxIncrement

增值某个块。

```
LONG
WINAPI
A6_SxxIncrement
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BYTE         bBlockNumber,
    __in    UINT         uValue
);
```

);

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 要读的扇区号。

bBlockNumber 要读的块号。

uValue 要增加到目标块的值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_SxxDecrement

减值某个块。

```
LONG
WINAPI
A6_SxxDecrement
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __in    BYTE         bBlockNumber,
    __in    UINT         uValue
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 要读的扇区号。

bBlockNumber 要读的块号。

uValue 要从目标块减掉的值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## Mifare Ultralight 卡操作函数

### A6\_U1Select

寻 Ultralight 卡。

```
LONG  
WINAPI  
A6_U1Select  
(  
    __in    READERHANDLE hReader  
);
```

#### 参数

hReader                      引用 A6\_Connect 返回的句柄值。

#### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

### A6\_U1GetUID

获取卡序列号。

```
LONG  
WINAPI  
A6_U1GetUID  
(  
    __in    READERHANDLE hReader,  
    __out   PBYTE      pbUIDBuff,  
    __inout PDWORD      pcbUIDLength  
);
```

#### 参数

hReader                      引用 A6\_Connect 返回的句柄值。

pbUIDBuff 指向返回的序列号信息。不可以为 NULL。

pcbUIDLength 提供 pbUIDBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_U1ReadSector

读扇区。

```
LONG
WINAPI
A6_U1ReadSector
(
    __in    READERHANDLE hReader,
    __in    BYTE         bSectorNumber,
    __out   BYTE         (&bBuffer)[4]
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

bSectorNumber 要操作的扇区号。

bBuffer 接收返回数据的缓冲区。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_U1WriteSector

写扇区。

```
LONG
WINAPI
A6_U1WriteSector
```

```
(  
    __in    READERHANDLE hReader,  
    __in    BYTE        bSectorNumber,  
    __in    BYTE        (&bBuffer) [4]  
);
```

## 参数

hReader            引用 A6\_Connect 返回的句柄值。

bSectorNumber    要操作的扇区号。

bBuffer           要写入到卡片的数据。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## 非接触 CPU 卡操作函数

### A6\_TypeACpuSelect

寻 TypeA CPU 卡。

```
LONG  
WINAPI  
A6_TypeACpuSelect  
(  
    __in    READERHANDLE hReader,  
    __out    PBYTE        pbATRBuff,  
    __inout PDWORD        pcbATRLength  
);
```

## 参数

hReader           引用 A6\_Connect 返回的句柄值。

pbATRBuff        指向返回的复位信息。不可以为 NULL。

pcbATRLength 提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_TypeBCpuSelect

寻 TypeB CPU 卡。

```
LONG
WINAPI
A6_TypeBCpuSelect
(
    __in    READERHANDLE hReader,
    __out   PBYTE        pbATRBuff,
    __inout PDWORD        pcbATRLength
);
```

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbATRBuff 指向返回的复位信息。不可以为 NULL。

pcbATRLength 提供 pbATRBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_TypeABCpuDeselect

取消选定 TypeA/B CPU卡。

```
LONG
WINAPI
A6_TypeABCpuDeselect
(
    __in    READERHANDLE hReader
```



);

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_TypeABCpuTransmit

TypeA/B CPU 卡数据传输。

LONG

WINAPI

A6\_TypeABCpuTransmit

(

\_\_in READERHANDLE hReader,

\_\_in PBYTE pbSendBuff,

\_\_in USHORT cbSendLength,

\_\_out PBYTE pbRecvBuff,

\_\_inout PDWORD pcbRecvLength

);

## 参数

hReader 引用 A6\_Connect 返回的句柄值。

pbSendBuff 指向要写入到卡片的数据。不可以为 NULL。

cbSendLength 提供 pbSendBuff 参数的长度（字节数）。

pbRecvBuff 指向返回的数据。不可以为 NULL。

pcbRecvLength 提供 pbRecvBuff 参数的长度（字节数）并接收读卡器实际返回的长度。不可以为 NULL。

## 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## A6\_TypeACpuGetUID

获取TypeA CPU卡序列号。

```
LONG
WINAPI
A6_TypeACpuGetUID
(
    __in    READERHANDLE hReader,
    __out   PBYTE    pbUIDBuff,
    __inout PDWORD    pcbUIDLength
);
```

### 参数

**hReader**            引用 A6\_Connect 返回的句柄值。

**pbUIDBuff**        指向返回的序列号信息。不可以为 NULL。

**pcbUIDLength**    提供 pbUIDBuff 参数的长度（字节数）并接收读卡器实际返回的长度。

### 返回值

如果函数调用成功，返回值为 0，其它值为错误码。

## 数据类型

### CRSTATUS

CRSTATUS 结构包含读卡器的基本状态信息。

```
typedef struct _CRSTATUS
{
    BYTE    bLaneStatus;
    BYTE    bFCI;
    BYTE    bRCI;
} CRSTATUS, *PCRSTATUS;
```

## 成员

bLaneStatus	通道状态。可用值：	
	LS_LONG_CARD_IN	有长卡
	LS_SHORT_CARD_IN	有短卡
	LS_CARD_IN_FRONT_NH	卡在前端不持卡位
	LS_CARD_IN_FRONT	卡在前端持卡位
	LS_CARD_IN_RF_POS	卡在射频位
	LS_CARD_IN_IC_POS	卡在IC位
	LS_CARD_IN_REAR	卡在后端持卡位
	LS_NO_CARD_IN	读卡器内无卡
bFrontSet	前端进卡方式。可用值：	
	FCI_PROHIBITED	禁止进卡。
	FCI_MAGCARD_ONLY	仅允许进磁卡。
	FCI_ALLOWED	允许进卡。
	FCI_THINCARD_ONLY	仅允许进薄磁卡。
bRearSet	后端进卡方式。可用值：	
	RCI_ALLOWED	允许进卡。
	RCI_PROHIBITED	禁止进卡。

## TRACKINFO

磁道信息结构。

```
typedef struct _TRACKINFO
{
    char    Contents[3][128];
    int     Lengths[3];
    BYTE    Status[3];
} TRACKINFO, *PTRACKINFO;
```

## 成员

Contents	各磁道的内容。	
Lengths	各磁道内容的长度。	
Status	各磁道的状态。可用值：	
	TRACKST_NORMAL	正常

TRACKST_NO_SS	无起始字符
TRACKST_NO_ES	无结束字符
TRACKST_PARITY_ERROR	校验错误
TRACKST_LRC_ERROR	LRC 错误
TRACKST_NO_DATA	空白轨

## 错误码

A6_E_DEV_NOT_RECOGNIZED	未检测到设备。（仅在调用 A6_Connect 函数时才有可能返回该错误。可能的原因：指定的COM端号或波特率不正确，或串口线未正确连接到计算机和设备，或电压异常，或串口线有问题）
A6_E_PORT_UNAVAILABLE	指定的COM端口不存在或者被其他程序占用。
A6_E_UNKNOWN_ERROR	检测到一个内部错误，但原因未知。
A6_E_COMM_TIMEOUT	内部通信超时。
A6_E_COMM_ERROR	内部通信错误。（可能的原因：在执行命令时，返回的数据在通信协议上未定义；或接到NAK；或BCC校验错误等）
A6_E_INVALID_HANDLE	提供的句柄无效。
A6_E_INVALID_PARAMETER	提供的一个或多个参数无效或者为NULL值。
A6_E_NO_MEMORY	没有足够的可用内存来完成当前的操作。
A6_E_BUFFER_TOO_SMALL	接收数据的缓冲区太小，不足以存放返回的数据。
A6_E_UNDEFINED_COMMAND	未定义的命令。
A6_E_DISABLED_COMMAND	当前的命令不能执行。
A6_E_COMMAND_DATA_ERROR	命令数据包错误。
A6_E_VOLTAGE_ABNORMALITY	电源电压异常，高于14.5V或小于10.8V。
A6_E_LENGTH_ABNORMALITY	读卡器内的卡片长度异常。
A6_E_POWER_DOWN	主供电电源掉电。

---

A6_E_COMMAND_FAILURE	命令执行失败。
A6_E_NO_CARD_IN	读卡器内无卡。
A6_E_CARD_UNRECOGNIZED	未能识别读卡器内的卡片的类型。
A6_E_NOT_IN_STD_POS	读卡器内的卡片不在可操作的位置。
A6_E_CARD_SCRAPPED	读卡器内的卡片已经报废。
A6_E_NO_RF_CARD	读卡器内无射频位。
A6_E_SECTOR_NOT_CERTIFIED	要操作的扇区未认证。
A6_E_SN_ERROR	卡序列号错误。
A6_E_INVALID_PASSWORD	校验卡密码时提供的密码不正确。
A6_E_DATA_FORMAT_ERROR	值格式不正确。
A6_E_VALUE_OVERFLOW	增值或减值时值溢出。