Advanced Card Systems Ltd.
Card & Reader Technologies

ACR1281U-C1

双界面读写器（USB 接口）

应用程序编程接口 V1.11

如有任何更改，恕不另行通知

info@acs.com.hk
www.acs.com.hk

版本历史

发布日期	修订说明	版本号
2011-08-19	• 初始发布	1.00
2011-12-21	• 增加 ACR128 手动 PICC 轮询 • 删除蜂鸣器控制 OFF:00 • 删除以下功能： ○ 蜂鸣器状态 ○ 读取/初始化注册表设置 ○ 读取/更新注册表 ○ 所有 Atmel 存储卡功能 ○ 读取接口状态 • 更新 5.3.5 & 5.3.6 节: 设置/读取默认 LED&蜂鸣器操作 • 更新 5.3.10 & 5.3.11 节: 设置/读取自动 PICC 轮询 • 更新关于设置/读取天线场的说明	1.01
2012-07-18	• 更新格式	1.02
2013-01-23	• 更新格式 • 更新 2.0 节：特性	1.03
2013-07-11	• 更新产品图片 • 在介绍特性的 2.0 节增加 FIPS201 • 更新 5.3.5 & 5.3.6 节: 设置/读取默认 LED&蜂鸣器操作 • 更新 5.3.10 & 5.3.11 节: 设置/读取自动 PICC 轮询	1.04
2014-01-09	• 更新格式 • 在关于接触式智能卡协议的 5.1 节增加存储卡 - ATMEL AT88SC15 和存储卡 - AT88SC101 / AT88SC102 / AT88SC1003 • 在关于接触式智能卡协议的 5.1 节增加存储卡- ATMEL AT88SC1608 ○ 选择卡片类型 ○ 读取存储卡 ○ 写入存储卡 ○ 校验密码 ○ 初始化认证 ○ 校验认证 • 删除 5.3 节的以下命令 ○ 设置自动 PPS ○ 读取自动 PPS ○ 设置天线场 ○ 读取天线场状态 ○ 设定用户额外保护时间设置 ○ 读取用户额外保护时间设置 ○ 设定“616C”自动处理选项设置 ○ 读取“616C”自动处理选项设置 • 刷新接口状态	1.05

发布日期	修订说明	版本号
2015-10-14	• 更新 5.3.10 节：设置自动 PICC 轮询	1.06
2017-01-18	• 更新 2.0 节：特性 • 增加 5.3.17 节：设置自动 PPS 和 5.3.18 节：读取自动 PPS • 在附录 D.直接命令（Escape Command）示例中增加备注 • 删除 2.0 节中的 FIPS201	1.07
2018-07-18	• 更新 2.0 节：特性	1.08
2020-04-30	• 更新 2.0 节：特性 • 附录 B（访问 Mifare DESFire 标签）转为 5.2.6 节 • 增加 5.3.19 节：读取序列号	1.09
2020-05-27	• 更新 5.2.1.1 节：ATR 信息格式（适用于 ISO 14443-3 PICC） • 更新 5.2.3 节：MIFARE 1K/4K 存储卡的 PICC 命令（T=CL 模拟）	1.10
2020-07-03	• 更新 5.3 节：外设控制	1.11

目录

图目录

表目录

1.0. 简介

ACR1281U-C1 DualBoost II 是 ACS 的 ACR128 DualBoost 读写器的第二代产品，属于功能强大而具有成本效益的双界面智能卡读写器，既支持符合 ISO 7816 标准的 MCU 卡和 MIFARE® 卡，也支持符合 ISO 14443 标准的 A 类和 B 类非接触式卡。它通过 USB CCID 类驱动程序以及 USB 接口与电脑连接，接受来自计算机应用的卡命令。

做为电脑与卡片之间的中间设备，ACR1281U-C1 会执行来自于电脑的命令，专门与非接触式标签、MCU 卡、SAM 卡及外围设备（LED 或蜂鸣器）进行通信。它的三种界面（PICC 界面、ICC 界面和 SAM 界面）均符合 PC/SC 标准，其中接触式界面使用 ISO 7816 标准定义的 APDU 命令。有关接触式 MCU 卡的操作，请参阅有关卡片的文档以及 PC/SC 标准。

本 API 文件会详细介绍如何执行 PC/SC APDU 命令来支持非接触式界面和接触式存储卡，以及控制 ACR1281U-C1 的外设。

2.0. 特性

- USB 全速接口
- 符合CCID标准
- 智能卡读写器：
  - 非接触接口：
    - 读写速度达 848 Kbps
    - 内置天线用于访问非接触式标签，卡片读取距离可达 50 mm（视标签的类型而定）
    - 支持 ISO 14443 第 4 部分的 A 类和 B 类卡，以及 MIFARE Classic® 系列卡片
    - 内建防冲突特性（任何时候都只能访问 1 张标签）
    - 支持扩展的 APDU（最大 64 KB）
  - 接触式接口：
    - 支持 ISO 7816 A 类、B 类和 C 类卡（5 V、3 V 和 1.8 V）
    - 支持 CAC（通用权限卡)
    - 支持 PIV （个人身份验证卡）
    - 支持符合 T=0 或 T=1 协议的微处理器卡
    - 支持各类存储卡
    - 支持 PPS（协议和参数选择）
    - 具有短路保护功能
    - 支持扩展的 APDU (T=1：最多 64 K 字节；T=0：最多 512+10 字节)
  - SAM 接口：
    - 一个 SAM 卡槽
    - 符合 ISO 7816 的 SAM 卡槽（A 类）
- 应用程序编程接口：
  - 支持 PC/SC
  - 支持 CT-API（通过 PC/SC 上一层的封装）
- 内置外围设备：
  - 2 个用户可控的 LED 指示灯
  - 1 个用户可控的蜂鸣器
- 具有USB固件升级能力
- 支持Android™ 3.1及以上版本[1]
- 符合下列标准：
  - ISO 14443
  - ISO 7816
  - PC/SC
  - CCID
  - CE
  - FCC
  - RoHS
  - REACH
  - Microsoft® WHQL

[1] *使用已定义的 Android 库*

3.0. ACR1281U-C1 的结构

3.1. 读写器功能框图

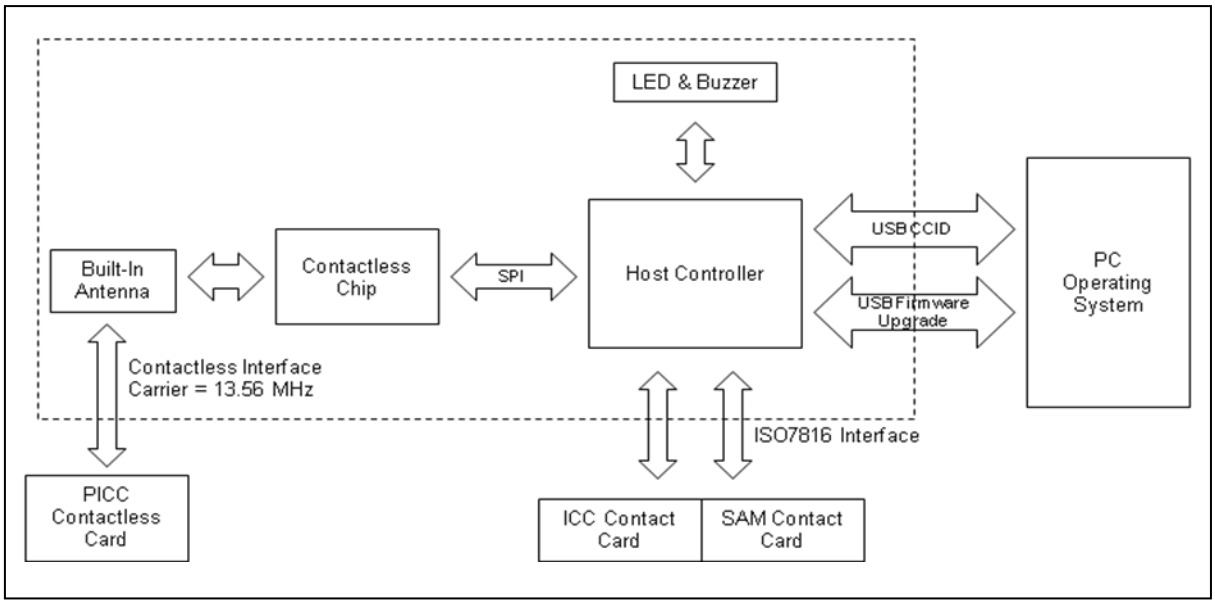

图1 : ACR1281U-C1 读写器功能框图

3.2. PC/SC 驱动与 ICC、PICC 和 SAM 间的通信

ACR1281U-C1 与计算机使用 CCID 协议进行通讯，而 ICC、PICC 和 SAM 间的通信则完全符合 PC/SC 标准。

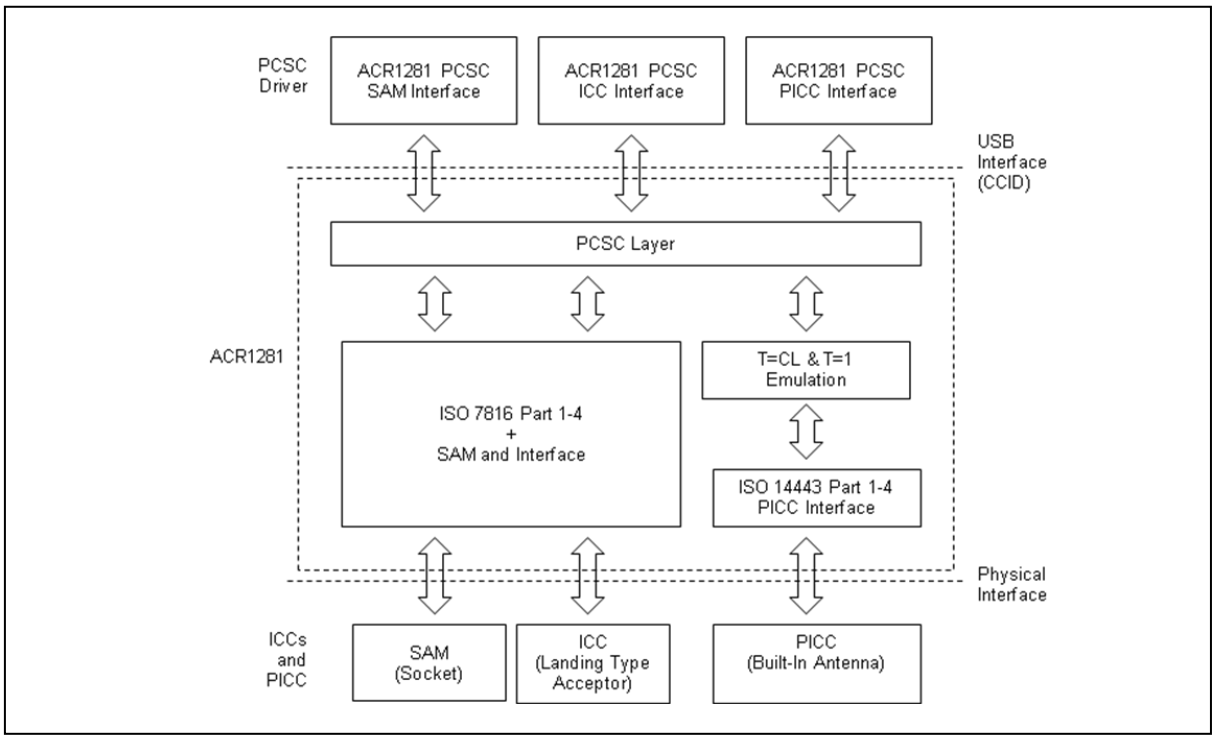

图2 : ACR1281U-C1 的结构

4.0. 硬件描述

4.1. USB

ACR1281U-C1 通过符合 USB 标准的 USB 接口与计算机连接。

4.1.1. 通信参数

ACR1281U-C1 按照 USB 2.0 规范的要求通过 USB 接口与计算机连接，支持 USB 全速模式，速率为 12 Mbps。

Pin	Signal	Function
1	V_{BUS}	为读写器提供+5 V 的电源
2	D-	ACR1281U-C1 和 PC 间以差分信号传输数据
3	D+	ACR1281U-C1 和 PC 间以差分信号传输数据
4	GND	参考电压等级

表1 : USB 接口配线

***注：**为了使 ACR1281U-C1 通过 USB 接口正常运行，应该先安装设备驱动程序。*

4.1.2. 端点

ACR1281U-C1 通过如下的端点与主计算机进行通信。

Control Endpoint – 用于设置和控制

Bulk OUT – 用于从主计算机发送至 ACR1281U-C1 的命令（数据包大小为 64 字节）

Bulk IN – 用于从 ACR1281U-C1 发送至主计算机的响应（数据包大小为 64 字节）

Interrupt IN – 用于从 ACR1281U-C1 发送至主计算机的卡片状态报文（数据包大小为 8 字节）

4.2. 接触式智能卡接口

ACR1281U-C1 与插入的智能卡之间的接口遵循 ISO 7816-3 标准，并进行了某些限制或提升来增强 ACR1281U-C1 的实用功能。

4.2.1. 智能卡电源 VCC (C1)

插入的智能卡电流消耗不得大于 50 mA。

4.2.2. 卡片类型选择

激活插入的卡片前，处于控制地位的电脑总是要向 ACR1281U-C1 发送正确的命令来选择卡片类型。这些卡片包括存储卡和基于 MCU 的卡。

对于基于 MCU 的卡来说，读写器允许从 T=0 或 T=1 中选择首选的协议。但是，只有当插入读写器的卡片对这两种协议类型都支持时，读写器才可以通过协议与参数选择（PPS）接受并执行这样的选择。当基于 MCU 的卡仅支持一种协议类型—T=0 或 T=1 时，读写器会自动采用该协议类型，而不管应用程序选择了哪个协议类型。

4.2.3. 微控制器卡接口

基于微控制器的智能卡只使用触点 C1（VCC）、C2（RST)、C3（CLK）、C5（GND）和 C7（I/O）。时钟信号（C3）的频率为 4.8 MHz。

4.3. 非接触式智能卡接口

ACR1281U-C1 与非接触卡之间的接口遵循 ISO 14443 标准，并进行了某些限制或提升来增强 ACR1281U-C1 的实用功能。

4.3.1. 载波频率

ACR1281U-C1 的载波频率为 13.56 MHz。

4.3.2. 卡片轮询

ACR1281U-C1 会自动检测进入工作场的非接触卡。此功能支持符合 ISO 14443-4 的 A 类和 B 类卡，以及 MIFARE 卡。

4.4. 用户接口

4.4.1. 蜂鸣器

ACR1281U-C1 配有一个单音蜂鸣器，用于指示“卡片插入”和“卡片移出”事件。

事件	蜂鸣器
1. 读写器上电，初始化成功。	1 次
2. 卡片插入（ICC 或 PICC）	1 次
3. 卡片移出（ICC 或 PICC）	1 次

表2：蜂鸣器事件说明

4.4.2. LED

ACR1281U-C1 配有 LED 指示灯，用于显示接触式和非接触式接口的状态。红色 LED 是 PICC 状态指示灯，绿色 LED 是 ICC 状态指示灯。

读写器状态	红色 LED PICC 指示器	绿色 LED ICC 指示器
1. 无法找到 PICC，或者 PICC 存在但无激活。	5 秒闪烁一次	
2. PICC 存在并激活	开	
3. PICC 操作中	闪烁	
4. ICC 存在并激活		开
5. ICC 存在但无激活		关
6. ICC 操作中		闪烁

表3：LED 指示灯

5.0. 软件设计

5.1. 接触式智能卡协议

5.1.1. 存储卡 – 1/2/4/8/16 kilobits I2C 卡

5.1.1.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Command	Class	INS	P1	P2	Lc	Card Type
Select Card Type	FFh	A4h	00h	00h	01h	01h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.1.2. Select page size

此命令会选择用于智能卡读取的页面大小。默认值是 8 字节页写。在卡片移出，读写器下电时会重置为默认值。

命令

Command	Class	INS	P1	P2	Lc	Page Size
Select Page Size	FFh	01h	00h	00h	01h	

其中：

Page Size （1 个字节）

03h = 8 字节页写

04h = 16 字节页写

05h = 32 字节页写

06h = 64 字节页写

07h = 128 字节页写

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.1.3. Read memory card

此命令会从指定的地址位置读取存储卡的内容。

命令

Command	Class	INS	Byte Address		MEM_L
			MSB	LSB	
Read Memory Card	FFh	B0h			

其中：

Byte Address 2 个字节）

存储卡的内存地址位置。

MEM_L （1 个字节）

要从存储卡读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡读取的数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.1.4. Write memory card

此命令用于将存储卡的内容写入指定地址。

命令

Command	Class	INS	Byte Address		MEM_L	Byte 1	…	…	Byte N
			MSB	LSB					
Write Memory Card	FFh	D0h							

其中：

Byte Address （2 个字节）

存储卡的内存地址位置。

MEM_L （1 个字节）

要从存储卡读取的数据的长度。

Byte (1…N) 要写入存储卡的数据。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.2. 存储卡 – 32/64/128/256/512/1024 kilobits I2C 卡

5.1.2.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Command	Class	INS	P1	P2	Lc	Card Type
Select Card Type	FFh	A4h	00h	00h	01h	02h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.2.2. Select page size

此命令用于选择读取智能卡的页面大小。默认值是 8 字节页写。在卡片移出，读写器下电时会重置为默认值。

命令

Command	Class	INS	P1	P2	Lc	Page Size
Select Page Size	FFh	01h	00h	00h	01h	

其中：

Page Size （1 个字节）

03h = 8 字节页写

04h = 16 字节页写

05h = 32 字节页写

06h = 64 字节页写

07h = 128 字节页写

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.2.3. Read memory card

此命令会从指定的地址位置读取存储卡的内容。

命令

Command	Class	INS	Byte Address		MEM_L
			MSB	LSB	
Read Memory Card	FFh				

其中：

INS （1 个字节）

B0h = 32, 64, 128, 256, 512 kilobit I2C 卡

1011 000*b；其中*为 17 位寻址的 MSB = 1024 kilobit I2C 卡

Byte Address （2 个字节）

存储卡的内存地址位置

MEM_L （1 个字节）

要从智能卡读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡中读取的数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.2.4. Write memory card

此命令用于将存储卡的内容写入指定地址。

命令

Command	Class	INS	Byte Address		MEM_L	Byte 1	…	…	Byte N
			MSB	LSB					
Write Memory Card	FFh								

其中：

INS （1 个字节）

D0h = 32, 64, 128, 256, 512 kilobit I2C 卡

1101 000*b；其中*为 17 位寻址的 MSB = 1024 kilobit I2C 卡：

Byte Address （2 个字节）

存储卡的内存地址位置。

MEM_L	（1 个字节） 要从存储卡读取的数据的长度。
Byte (1…N)	要写入存储卡的数据。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.3. 存储卡 – ATMEL AT88SC153

5.1.3.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。另外它还将选择页面大小为 8 字节页写。

命令

Pseudo-APDU						
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Card Type**
Select Card Type	FFh	A4h	00h	00h	01h	03h

响应

Response	**Data Out**	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.3.2. Read memory card

此命令会从指定的地址位置读取存储卡的内容。

命令

Pseudo-APDU					
Command	**CLA**	**INS**	**P1**	**Byte Address**	**MEM_L**
Read Memory Card	FFh		00h		

其中：

INS （1 个字节）

读取分区 00b, INS = B0h

读取分区 01b, INS = B1h

读取分区 10b, INS = B2h

读取分区 11b, INS = B3h

读取熔丝标志, INS = B4h

Byte Address （1 个字节）

存储卡的内存地址位置。

MEM_L （1 个字节）

要从存储卡读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡读取的数据

SW1 SW2 = 90 00h（若操作成功完成）

5.1.3.3. Write memory card

此命令会从指定地址位置开始向存储卡写入内容。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**Byte Address**	**MEM_L**	**Byte 1**	**…**	**…**	**Byte N**
Write Memory Card	FFh		00h						

其中：

INS （1 个字节）

读取分区 00b, INS = D0h

读取分区 01b, INS = D1h

读取分区 10b, INS = D2h

读取分区 11b, INS = D3h

读取熔丝标志, INS = D4h

Byte Address （1 个字节）

存储卡的内存地址位置。

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要写入存储卡的数据

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.3.4. Verify password

此命令用于校验用户输入的密码是否与存储卡的密码相同。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**RP**	**PW (0)**	**PW (1)**	**PW (2)**
Verify Password	FFh	20h	00h		03h				

其中：

PW (0), PW (1), PW (2) = 要发送给存储卡的密码

P2 （1 个字节）

= 0000 00r pb

其中“rp”两个位标识待比较的密码

r = 0：写密码

r = 1：读密码

p = 密码集编号

r p = 01b：安全密码

响应

Response	Data Out	
Result	SW1	ErrorCnt

其中：

SW1 = 90h

ErrorCnt （1 个字节）

= 错误计数器

FFh 表示验证正确，00h 表示密码被锁定（超过最大重试次数）。其它值表示当前验证失败。

5.1.3.5. Initialize authentication

此命令用于初始化存储卡认证。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Q (0)**	**Q (1)**	**…**	**Q (7)**
Initialize Authentication	FFh	84h	00h	00h	08h				

其中：

Q (0…7) （8 个字节）

= 主机随机数

响应

Response	**Data Out**	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.3.6. Verify authentication

此命令用于校验存储卡认证。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Ch (0)**	**Ch (1)**	**…**	**Ch (7)**
Verify Authentication	FFh	82h	00h	00h	08h				

其中：

Ch (0…7) （8 个字节）

= 主机挑战数

响应

Response	**Data Out**	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.4. 存储卡 – ATMEL AT88SC1608

5.1.4.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。另外它还将选择页面大小为 16 字节页写。

命令

Pseudo-APDU						
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Card Type**
Select Card Type	FFh	A4h	00h	00h	01h	04h

响应

Response	**Data Out**	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.4.2. Read memory card

此命令会从指定的地址位置读取存储卡的内容。

命令

Pseudo-APDU					
Command	**CLA**	**INS**	**Zone Address**	**Byte Address**	**MEM_L**
Read Memory Card	FFh				

其中：

INS （1 个字节）

读取用户区, INS = B0h

读取配置区或读取熔丝标志，INS = B1h

Zone Address （1 个字节）

= 00000 A10 A9 A8b，其中 A10 是分区地址的 MSB

** 读熔丝标志时无关

Byte Address （1 个字节）

= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

读熔丝标志时，Byte Address = 1000 0000b

MEM_L （1 个字节）

要从存储卡内读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡读取的数据

SW1 SW2 = 90 00h（若操作成功完成）

5.1.4.3. Write to memory card

此命令会从指定地址位置开始向存储卡写入内容。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**Zone Address**	**Byte Address**	**MEM_L**	**Byte 1**	**…**	**…**	**Byte N**
Write Memory Card	FFh								

其中：

INS （1 个字节）

读取用户区, **INS = D0h**

读取配置区或读取熔丝标志, **INS = D1h**

Zone Address （1 个字节）

= 00000 A10 A9 A8b，其中 A10 是分区地址的 MSB

** 读熔丝标志时无关

Byte Address （1 个字节）

= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

读熔丝标志时，Byte Address = 1000 0000b

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要写入存储卡的数据

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.4.4. Verify password

此命令用于校验用户输入的密码是否与存储卡的密码相同。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**RP**	**PW (0)**	**PW (1)**	**PW (2)**
Verify Password	FFh	20h	00h	00h	04h				

其中：

PW (0), PW (1), PW (2) = 要发送给存储卡的密码

RP （1 个字节）

= 0000 r p2 p1 p0b

其中“r p2 p1 p0”两个位标识待比较的密码

r = 0：写密码

r = 1：读密码

p2 p1 p0 = 密码集编号

r p2 p1 p0 = 0111b：安全密码。

响应

Response	**Data Out**	
Result	SW1	ErrorCnt

其中：

SW1 = 90h

ErrorCnt （1 个字节）

= 错误计数器

FFh 表示验证正确，00h 表示密码被锁定（超过最大重试次数）。其它值表示当前验证失败。

5.1.4.5. Initialize authentication

此命令用于初始化存储卡认证。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Q (0)**	**Q (1)**	**…**	**Q (7)**
Initialize Authentication	FFh	84h	00h	00h	08h				

其中：

Q (0…7) （8 个字节）

= 主机随机数

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.4.6. Verify authentication

此命令用于校验存储卡认证。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Ch (0)**	**Ch (1)**	**…**	**Ch (7)**
Verify Authentication	FFh	82h	00h	00h	08h				

其中：

Ch (0…7) （8 个字节）

= 主机挑战数

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.5. 存储卡 – SLE4418/SLE4428/SLE5518/SLE5528

5.1.5.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Command	Class	INS	P1	P2	Lc	Card Type
Select Card Type	FFh	A4h	00h	00h	01h	05h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.5.2. Read memory card

此命令会通过指定地址读取存储卡的内容。

命令

Command	Class	INS	Byte Address		MEM_L
			MSB	LSB	
Read Memory Card	FFh	B0h			

其中：

MSB Byte Address （1 个字节）
= 0000 00 A9 A8b 是存储卡的内存地址位置

LSB Byte Address （1 个字节）
= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

MEM_L （1 个字节）
要从存储卡读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡中读取的数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.5.3. Read presentation error counter memory card (for SLE4428 and SLE5528 only)

此命令用于读取密码输入错误计数器。

命令

Command	Class	INS	P1	P2	MEM_L
Read Presentation Error Counter	FFh	B1h	00h	00h	03h

响应

Response	ErrCnt	Dummy 1	Dummy 2	SW1	SW2
Result					

其中：

ErrCnt （1 个字节）

密码输入错误计数器的值。

FFh =表示验证正确

00h =表示密码被锁定（超过最大重试次数）

其它值表示验证失败

Dummy 1, Dummy 2 （2 个字节）

从卡片读取的虚拟数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.5.4. Read protection bit

此命令用于读取保护位。

命令

Command	Class	INS	Byte Address		MEM_L
			MSB	LSB	
Read Protection Bit	FFh	B2h			

其中：

MSB Byte Address （1 个字节）

存储卡的内存地址位置。

= 0000 00 A9 A8b

LSB Byte Address （1 个字节）

存储卡的内存地址位置。

= A7 A6 A5 A4 A3 A2 A1 A0b

MEM_L （1 个字节）

从卡片中读取的保护位的长度，位数是 8 的倍数，最大值为 32。

MEM_L = 1 + INT ((number of bits – 1)/8)

例如，要读取始于内存 0010h 的 8 个保护位，应当运行下面的私有 APDU：

FF B1 00 10 01h

响应

Response	PROT 1	…	…	PROT L	SW1	SW2
Result						

其中：

PROT (1..L) 含有保护位的字节

SW1 SW2 = 90 00h （若操作成功完成）

在 PROT 字节中，保护位的排列如下：

PROT 1								PROT 2								….							
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	P18	P17

其中：

Px 是响应数据中字节 *x* 的保护位：

0 = 字节写保护

1 = 字节可以被写入

5.1.5.5. Write memory card

此命令会将存储卡的内容写入指定地址。

命令

Command	Class	INS	Byte Address		MEM_L	Byte 1	…	…	Byte N
			MSB	LSB					
Write Memory Card	FFh	D0h							

其中：

MSB Byte Address （1 个字节）

= 0000 00 A9 A8b 是存储卡的内存地址位置

LSB Byte Address （1 个字节）

= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要写入存储卡的数据。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.5.6. Write protection memory card

此命令将命令中指定的每一个字节与存储在特定地址中的字节进行对比，若数据相符，则相应的保护位就会被不可逆转的设定为“0”。

命令

Command	Class	INS	Byte Address		MEM_L	Byte 1	…	…	Byte N
			MSB	LSB					
Write Protection Memory Card	FFh	D1h							

其中：

MSB Byte Address （1 个字节）

= 0000 00 A9 A8b 是存储卡的内存地址位置

LSB Byte Address （1 个字节）

= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要与卡片内始于 Byte Address 的数据做比较的 Byte 值。Byte 1 与在 Byte Address 的数据比较；Byte N 与在 Byte Address + N -1 的数据比较。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.5.7. Present code memory card (for SLE44428 and SLE5528 only)

此命令用于向存储卡提交密码，从而启用对 SLE4428 卡和 SLE5528 卡的写操作。执行的操作如下：

- 搜索密码输入错误计数器中值为“1”的位，然后将该位写为“0”
- 向卡片提交指定的密码
- 擦除密码错误计数器

命令

Command	Class	INS	P1	P2	MEM_L	Code	
						Byte 1	Byte 2
Present Code Memory Card	FFh	20h	00h	00h	02h		

其中：

Code （3 个字节）
密码（PIN）

响应

Response	Data Out	
Result	90h	ErrorCnt

其中：

ErrorCnt （1 个字节）
错误计数器。
FFh = 表示验证正确。
00h = 表示密码被锁定（超过最大重试次数）。
其它值表示验证失败。

5.1.6. 存储卡 – SLE4432/SLE4442/SLE5532/SLE5542

5.1.6.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Command	Class	INS	P1	P2	Lc	Card Type
Select Card Type	FFh	A4h	00h	00h	01h	06h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.6.2. Read Memory Card

此命令会通过指定地址读取存储卡的内容。

命令

Command	Class	INS	P1	Byte Address	MEM_L
Read Memory Card	FFh	B0h	00h		

其中：

Byte Address 1 个字节）

= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

MEM_L （1 个字节）

要从存储卡内读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	PROT 1	PROT 2	PROT 3	PROT 4	SW1	SW2
Result										

其中：

Byte (1…N) 从存储卡中读取的数据

PROT (1…4) 含有保护位的字节

SW1 SW2 = 90 00h （若操作成功完成）

在 PROT 字节中，保护位的排列如下：

PROT 1								PROT 2								....							
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	P18	P17

其中：

Px 是响应数据中字节 *x* 的保护位：

0 = 字节写保护

1 = 字节可以被写入

5.1.6.3. Read presentation error counter memory card (for SLE4442 and SLE5542 only)

此命令用于读取密码输入错误计数器。

命令

Command	Class	INS	P1	P2	MEM_L
Read Presentation Error Counter	FFh	B1h	00h	00h	04h

响应

Response	ErrCnt	Dummy 1	Dummy 2	Dummy 3	SW1	SW2
Result						

其中：

ErrCnt （1 个字节）

密码输入错误计数器的值。

07h = 表示验证正确

00h = 表示密码被锁定（超过最大重试次数）

其它值表示验证失败

Dummy 1, Dummy 2, Dummy 3 （3 个字节）

从卡片读取的虚拟数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.6.4. Read protection bit

此命令用于读取前 32 个字节的保护位。

命令

Command	Class	INS	P1	P2	MEM_L
Read Protection Bit	FFh	B2h	00h	00h	04h

响应

Response	PROT 1	PROT 2	PROT 3	PROT 4	SW1	SW2
Result						

其中：

PROT (1..4) 含有保护位的字节

SW1 SW2 = 90 00h （若操作成功完成）

在 PROT 字节中，保护位的排列如下：

PROT 1								PROT 2								….							
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	P18	P17

其中：

Px 响应数据中字节的保护位：

0 = 字节写保护

1 = 字节可以被写入

5.1.6.5. Write memory card

此命令会将存储卡的内容写入指定地址。

命令

Command	Class	INS	P1	Byte Address	MEM_L	Byte 1	…	…	Byte N
Write Memory Card	FFh	D0h	00h						

其中：

Byte Address （1 个字节）

= A7 A6 A5 A4 A3 A2 A1 A0b 是存储卡的内存地址位置

MEM_L （1 个字节）

要写入存储卡的数据的长度

Byte (1…N) 要写入存储卡的数据

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.6.6. Write protection memory card

此命令将命令中指定的每一个字节与存储在特定地址中的字节进行对比，若数据相符，则相应的保护位就会被不可逆转的设定为“0”。

命令

Command	Class	INS	P1	Byte Address	MEM_L	Byte 1	…	…	Byte N
Write Protection Memory Card	FFh	D1h	00h						

其中：

Byte Address （1 个字节）

= 000A4 A3 A2 A1b (00h - 1Fh) 是存储卡的保护内存地址位置

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要与卡片内始于 Byte Address 的数据做比较的 Byte 值。Byte 1 与在 Byte Address 的数据比较；Byte N 与在 Byte Address + N -1 的数据比较。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.6.7. Present code memory card (for SLE4442 and SLE5542 only)

此命令用于向存储卡提交密码，从而启用对 SLE4442 卡和 SLE5542 卡的写操作。执行的操作如下：

1. 搜索密码输入错误计数器中值为“1”的位，然后将该位写为“0”
2. 向卡片提交指定的密码
3. 擦除密码错误计数器

命令

Command	Class	INS	P1	P2	MEM_L	Code		
						Byte 1	Byte 2	Byte 3
Present Code Memory Card	FFh	20h	00h	00h	03h			

其中：

Code 密码（PIN）（3 个字节）

响应

Response	Data Out	
Result	SW1	ErrorCnt

其中：

ErrorCnt （1 个字节）

错误计数器。

07h = 表示验证正确

00h = 表示密码被锁定（超过最大重试次数）。

其它值表示验证失败。

5.1.6.8. Change code memory card (for SLE4442 and SLE5542 only)

此命令用于将特定数据作为新密码写入卡片。执行此命令之前，需要先使用“Present Code”命令向卡片提交当前密码。

命令

Command	Class	INS	P1	P2	MEM_L	Code		
						Byte 1	Byte 2	Byte 3
Change Code Memory Card	FFh	D2h	00h	01h	03h			

其中：

Code 密码（PIN）（3 个字节）

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.7. 存储卡 – SLE4406/SLE4436/SLE5536/SLE6636

5.1.7.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Command	Class	INS	P1	P2	Lc	Card Type
Select Card Type	FFh	A4h	00h	00h	01h	07h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.7.2. Read memory card

此命令会通过指定地址读取存储卡的内容。

命令

Command	Class	INS	P1	Byte Address	MEM_L
Read Memory Card	FFh	B0h	00h		

其中：

Byte Address （1 个字节）
存储卡的内存地址位置

MEM_L （1 个字节）
要从存储卡内读取的数据的长度。

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡中读取的数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.7.3. Write one byte memory card

此命令用于向所插入卡片的特定地址写一个字节。该字节从 LSB 开始写入卡片，也就是说，卡片地址 bit 0 被视为 byte 0 的 LSB。

此类卡片有四种不同的写入模式，通过命令数据域内的标志加以区分。

a. Write

命令中指定的字节值被写入特定的地址，可用于向卡片写入个人化信息和计数器值。

b. Write with carry

命令中指定的字节值被写入特定的地址，且命令被送至卡片来擦除下一个低位计数器。因此， 该模式仅适用于卡内计数器的值的更新。

c. Write with backup enabled (for SLE4436, SLE5536 and SLE6636 only)

命令中指定的字节值被写入特定的地址，可用于向卡片写入个人化信息和计数器值。同时启用 备份位，保护数据免受卡片插拔导致的损失。

d. Write with carry and backup enabled (SLE4436, SLE5536 and SLE6636 only)

命令中指定的字节值被写入特定的地址，且命令被送至卡片来擦除下一个低位计数器。因此， 该模式仅适用于卡内计数器的值的更新。同时启用备份位，保护数据免受卡片插拔导致的损 失。

在这四种模式下，指定地址上的字节在写操作前不会被擦除，所以存储位只能由“1”设为“0”。

SLE4436 卡和 SLE5536 卡的备份模式可以在写操作中被启用或禁用。

命令

Command	Class	INS	P1	Byte Address	MEM_L	Mode	Byte
Read Memory Card	FFh	D0h	00h		02h		

Where:

Byte Address （1 个字节）

存储卡的内存地址位置

Mode （1 个字节）

指定写入模式和备份选项

00h = write

01h = write with carry

02h = write with backup enabled (for SLE4436, SLE5536 and SLE6636 only)

03h = write with carry and with backup enabled (for SLE4436, SLE5536 and SLE6636 only)

Byte （1 个字节）

要写入卡片的字节值

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.7.4. Present code memory card

此命令用于向存储卡提交密码，从而启用卡片个人化模式。执行的操作如下：

- 搜索密码输入错误计数器中值为“1”的位，然后将该位写为“0”
- 向卡片提交指定的密码

命令

Command	Class	INS	P1	P2	MEM_L	Code			
						Addr	Byte 1	Byte 2	Byte 3
Present Code Memory Card	FFh	20h	00h	00h	04h	09h			

其中：

Addr （1 个字节）

输入错误计数器的字节地址

Code （3 个字节）

密码（PIN）

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.7.5. Authenticate memory card (for SLE4436, SLE5536 and SLE6636 only)

此命令用于从卡片中读取认证证书。执行的操作如下：

- 根据命令在卡片中选择 Key 1 或 Key 2
- 将命令中指定的随机数提交给卡片
- 为卡片计算出的每位认证数据生成指定数量的时钟脉冲
- 从卡片中读取 16 位的认证数据
- 将卡片复位回正常的操作模式

认证的过程分为 2 个步骤：步骤 1 是将认证证书发送至卡片。步骤 2 是取回卡片计算出的 2 个字节的认证数据。

步骤 1：向卡片发送认证证书

命令

Command	Class	INS	P1	P2	MEM_L	Code				
						Key	CLK_CNT	Byte 1	…	Byte 6
Send Authentication Certificate	FFh	84h	00h	00h	08h					

其中：

Key （1 个字节）

用于计算认证证书的密钥。

00h = key1，不带密码块链接

01h = key2，不带密码块链接

80h = key1，带密码块链接（仅适用于 SLL5536 和 SLE6636）

81h = key2，带密码块链接（仅适用于 SLL5536 和 SLE6636）

CLK_CNT （1 个字节）

待提供给卡片的时钟脉冲的个数，卡片将该脉冲用于计算认证证书的每个位。通常为 160（A0h）。

Byte (1...6) 卡片随机数据.

响应

Response	SW1	SW2
Result	61h	02h

步骤 2：取认证数据（Get_Response）

命令

Command	Class	INS	P1	P2	MEM_L
Get Authentication Data	FFh	C0h	00h	00h	02h

响应

Response	Cert		SW1	SW2
Result				

其中：

Cert （2 个字节）

卡片计算出的 16 位的认证数据。Byte 1 的 LSB 是从卡片中读取的第一个认证位。

SW1 SW2 = 90 00h （若操作成功完成）

5.1.8. 存储卡 – SLE4404

5.1.8.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Command	Class	INS	P1	P2	Lc	Card Type
Select Card Type	FFh	A4h	00h	00h	01h	08h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.8.2. Read memory card

此命令会从指定的地址位置读取存储卡的内容。

命令

Command	Class	INS	P1	Byte Address	MEM_L
Read Memory Card	FFh	B0h	00h		

其中：

Byte Address （1 个字节）

存储卡的内存地址位置

MEM_L （1 个字节）

要从存储卡内读取的数据的长度。.

响应

Response	Byte 1	…	…	Byte N	SW1	SW2
Result						

其中：

Byte (1…N) 从存储卡中读取的数据

SW1 SW2 = 90 00h （若操作成功完成）

5.1.8.3. Write memory card

此命令会将存储卡的内容写入指定地址。字节从 LSB 开始写入卡片，也就是说，卡片地址 bit 0 被视为 byte 0 的 LSB。

指定地址上的字节在写操作前不会被擦除，所以存储位只能由“1”设为“0”。

命令

Command	Class	INS	P1	Byte Address	MEM_L	Byte 1	…	…	Byte N
Write Memory Card	FFh	D0h	00h						

其中：

Byte Address （1 个字节）

存储卡的内存地址位置

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要写入存储卡的数据。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.8.4. Erase scratch pad memory card

此命令用于擦除所插入卡片的暂存存储器的数据。暂存存储器内所有的存储位都会被设定为状态“1”。

命令

Command	Class	INS	P1	Byte Address	MEM_L
Erase Scratch Pad	FFh	D2h	00h		00h

其中：

Byte Address （1 个字节）

暂存存储器的内存字节地址位置。（典型值为 02h）

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

5.1.8.5. Verify user code

此命令用于向插入的卡片提交用户密码（2 个字节）。该密码允许用户访问卡片的内存。

执行的操作如下：

1. 向卡片提交指定的密码
2. 搜索密码输入错误计数器中值为“1”的位，然后将该位写为“0”
3. 擦除密码输入错误计数器。提交的密码验证正确后，用户错误计数器可被擦除。

命令

Command	Class	INS	Error Counter LEN	Byte Address	MEM_L	Code	
						Byte 1	Byte 2
Verify User Code	FFh	20h	04h	08h	02h		

其中：

Error Counter LEN （1 个字节）

密码输入错误计数器的长度，单位为比特。

Byte Address （1 个字节）

卡片中密钥的字节地址。

Code （1 个字节）

用户密码。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

= 63 00h （若剩余重试次数为 0）

注：*收到响应 SW1 SW2 = 90 00h 后，应当再次读取用户错误计数器，检查 Verify_User_Code 是否正确。 如果用户错误计数器被擦除并且等于‘FFh’，证明先前的验证成功。*

5.1.8.6. Verify memory code

此命令用于向插入的卡片提交存储密码（4 个字节）。该存储密码可授权用户重新载入用户内存及用户密码。

执行的操作如下：

1. 向卡片提交指定的密码
2. 搜索密码输入错误计数器中值为“1”的位，然后将该位写为“0”
3. 擦除密码输入错误计数器。

注：*存储错误计数器的内容不能被擦除。*

命令

Command	Class	INS	Error Counter LEN	Byte Address	MEM_L	Code			
						Byte 1	Byte 2	Byte 3	Byte 4
Verify Memory Code	FFh	20h	40h	28h	04h				

其中：

Error Counter LEN （1 个字节）

密码输入错误计数器的长度，单位为比特。

Byte Address （1 个字节）

卡片中密钥的字节地址。

Code （4 个字节）

存储密码。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h （若操作成功完成）

= 63 00h （若剩余重试次数为 0）

注：*收到响应 SW1 SW2 = 90 00h 后，应当再次读取用户错误计数器，检查 Verify_User_Code 是否正确。如果应用区域的全部数据都被擦除并且等于‘FFh’，证明先前的验证成功。*

5.1.9. 存储卡 – AT88SC101/AT88SC102/AT88SC1003

5.1.9.1. Select card type

此命令用于对选定的插入读写器的卡片进行上电/下电，同时进行卡片复位操作。

命令

Pseudo-APDU						
Command	**CLA**	**INS**	**P1**	**P2**	**Lc**	**Card Type**
Select Card Type	FFh	A4h	00h	00h	01h	09h

响应

Response	**Data Out**	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.9.2. Read memory card

此命令会从指定的地址位置读取存储卡的内容。

命令

Pseudo-APDU					
Command	**CLA**	**INS**	**P1**	**Byte Address**	**MEM_L**
Read Memory Card	FFh	B0h	00h		

其中：

Byte Address （1 个字节）
存储卡的内存地址位置。

MEM_L （1 个字节）
要从存储卡读取的数据的长度。

响应

Response	**Byte 1**	**…**	**…**	**Byte N**	**SW1**	**SW2**
结果						

其中：

Byte (1…N) 从存储卡读取的数据

SW1 SW2 = 90 00h（若操作成功完成）

5.1.9.3. Write memory card

此命令向所插入卡片的特定地址写入数据。该字节从 LSB 开始写入卡片，也就是说，卡片地址 bit 0 被视为 byte 0 的 LSB。

指定地址上的字节在写操作前不会被擦除，所以存储位只能由‘1’设为‘0’。

命令

Pseudo-APDU									
Command	**CLA**	**INS**	**P1**	**Byte Address**	**MEM_L**	**Byte 1**	**…**	**…**	**Byte N**
Write Memory Card	FFh	D0h	00h						

其中：

Byte Address （1 个字节）

存储卡的内存地址位置。

MEM_L （1 个字节）

要写入存储卡的数据的长度。

Byte (1…N) 要写入卡片的字节值。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.9.4. Erase Non-Application Zone

此命令用于擦除存储在非应用区的数据。EEPROM 内存由 16 位字构成。即使只擦除单独的一个位，内存中的整个字都会被 ERASE 操作所清除。因此对某个字中的任何位执行 Erase 操作，都会将该字的全部 16 位清除为状态‘1’。

要擦除错误计数器或是在应用区域存储的数据，请参考：

- 指定的 Erase Application Zone With Erase 命令
- 指定的 Erase Application Zone With Write and Erase 命令
- 指定的 Verify Security Code 命令

命令

Pseudo-APDU					
Command	**CLA**	**INS**	**P1**	**Byte Address**	**MEM_L**
Erase Non-Application Zone	FFh	D2h	00h		00h

其中：

Byte Address （1 个字节）

要擦除的字的内存字节地址位置

响应

Response	**Data Out**	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

5.1.9.5. Erase Application Zone with Erase

此命令可用于下列情况：

- AT88SC101：擦除应用区域中的数据，EC 功能禁用
- AT88SC102：擦除应用区域 1 中的数据
- AT88SC102：擦除应用区域 2 中的数据，EC2 功能禁用
- AT88SC1003：擦除应用区域 1 中的数据
- AT88SC1003：擦除应用区域 2 中的数据，EC2 功能禁用
- AT88SC1003：擦除应用区域 3 中的数据

此命令执行以下操作：

1. 向卡片提交指定的密码
2. 擦除密码输入错误计数器。提交的密码验证正确后，相应的应用区域中的数据可以被擦除。

命令

Pseudo-APDU										
Command	**CLA**	**INS**	**Error Counter LEN**	**Byte Address**	**MEM_L**	**CODE**				
						Byte 1	**Byte 2**	**…**	**…**	**Byte N**
Erase Application Zone with Erase	FFh	20h	00h							

其中：

Error Counter LEN （1 个字节）

= 密码输入错误计数器的长度，单位为比特。值始终是 00h。

Byte Address （1 个字节）

= 卡片中应用区域密钥的字节地址。正确值请参阅下表：

MEM_L （1 个字节）

= “擦除”密钥的长度。正确值请参阅下表：

CODE (1…N) = “擦除”密钥

Case	Byte Address	LEN
AT88SC101：擦除应用区域，EC 功能禁用	96h	04h
AT88SC102：擦除应用区域 1	56h	06h
AT88SC102：擦除应用区域 2，EC2 功能禁用	9Ch	04h
AT88SC1003：擦除应用区域 1	36h	06h
AT88SC1003：擦除应用区域 2，EC2 功能禁用	5Ch	04h
AT88SC1003：擦除应用区域 3	C0h	06h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

注：*收到状态字 SW1SW2 = 90 00 后，可重新读取应用区域内的数据来检查 Erase Application Zone with Erase 命令是否正确。如果应用区域的全部数据都被擦除并且等于“FF”，证明先前的验证成功。*

5.1.9.6. Erase Application Zone with Write and Erase

此命令可用于下列情况：

- AT88SC101：擦除应用区域中的数据，EC 功能启用
- AT88SC102：擦除应用区域 2 中的数据，EC2 功能启用
- AT88SC1003：擦除应用区域 2 中的数据，EC2 功能启用

EC 或 EC2 功能启用后（即：ECEN 或 EC2EN 标识位没有被更改并处于“1”状态），会执行以下操作：

1. 向卡片提交指定的密码
2. 搜索密码输入错误计数器中值为‘1’的位，然后将该位写为‘0’
3. 擦除密码输入错误计数器。提交的密码验证正确后，相应的应用区域中的数据可以被擦除。

命令

Pseudo-APDU									
Command	CLA	INS	Error Counter LEN	Byte Address	MEM_L	CODE			
						Byte 1	Byte 2	Byte 3	Byte 4
Erase Application Zone with Write and Erase	FFh	20h	80h		04h				

其中：

Error Counter LEN （1 个字节）

= 密码输入错误计数器的长度，单位为比特。值始终是 80h。

Byte Address （1 个字节）

= 卡片中应用区域密钥的字节地址。正确值请参阅下表：

CODE （4 个字节）

= “擦除”密钥

Cases	Byte Address
AT88SC101	96h
AT88SC102	9Ch
AT88SC1003	5Ch

响应

Response	Data Out	
结果	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

= 63 00（若不再有重试机会）

注：*收到状态字 SW1SW2 = 90 00 后，可重新读取应用区域内的数据来检查 Erase Application Zone with Write and Erase 命令是否正确。如果应用区域的全部数据都被擦除并且等于“FF”，证明 先 前 的验证成功。*

5.1.9.7. Verify Security Code

此命令用于向插入的卡片提交安全密码（2 个字节）。安全密码旨在使卡的内存能够被访问。

执行的操作如下：

1. 向卡片提交指定的密码
2. 搜索密码输入错误计数器中值为‘1’的位，然后将该位写为‘0’
3. 擦除密码输入错误计数器。提交的密码验证正确后，安全密码尝试计数器可被擦除。

命令

Pseudo-APDU							
Command	CLA	INS	Error Counter LEN	Byte Address	MEM_L	CODE	
						Byte 1	Byte 2
Verify Security Code	FFh	20h	08h	0Ah	02h		

其中：

Error Counter LEN （1 个字节）

= 密码输入错误计数器的长度，单位为比特。

Byte Address （1 个字节）

= 卡片中密钥的字节地址。

CODE （2 个字节）

= 安全密码

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

= 63 00（若不再有重试机会）

注：*收到响应 SW1SW2 = 90 00 后，应当再次读取安全密码尝试计数器（SCAC），检查 Verify User Code 是否正确。如果 SCAC 已经被擦除并且等于“FF”，证明先前的验证成功。*

5.1.9.8. Blown Fuse

此命令用于更改所插入卡片的标识位。标识位可以是 EC_EN 标识位、EC2EN 标识位、发行商标识位或生产商标识位。

注：*更改标识位的过程是不可逆的。*

命令

Pseudo-APDU									
Command	CLA	INS	Error Counter LEN	Byte Address	MEM_L	CODE			
						Fuse Bit Addr (High)	Fuse Bit Addr (Low)	State of FUS Pin	State of RST Pin
Blown Fuse	FFh	05h	00h	00h	04h			01h	00h 01h

其中：

Fuse Bit Addr （2 个字节）

= 标识位的位地址。正确值请参阅下表：

State of FUS Pin （1 个字节）

= FUS Pin 的状态，始终应该是 01h。

State of RST Pin （1 个字节）

= RST Pin 的状态，正确值请参阅下表。

		Fuse Bit Addr (High)	Fuse Bit Addr (Low)	State of RST Pin
AT88SC101	生产商标识位	05h	80h	01h
	EC_EN 标识位	05h	C9h	01h
	发行商标识位	05h	E0h	01h
AT88SC102	生产商标识位	05h	B0h	01h
	EC2EN 标识位	05h	F9h	01h
	发行商标识位	06h	10h	01h
AT88SC1003	生产商标识位	03h	F8h	00h
	EC2EN 标识位	03h	FCh	00h
	发行商标识位	03h	E0h	00h

表4：更改标识位代码值

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h（若操作成功完成）

= 63 00（若不再有重试机会）

5.2. 非接触式智能卡协议

5.2.1. ATR 的生成

读写器检测到 PICC 后，一个 ATR 会被发送至 PC/SC 驱动来识别 PICC。

5.2.1.1. ATR 信息格式（适用于 ISO 14443-3 PICC）

字节	值（十六进制）	标记	说明
0	3B	初始字符	
1	8N	T0	高半字节 8 表示：后续不存在 TA1、TB1 和 TC1，只存在 TD1 低半字节 N 指出历史字符的个数（HistByte 0 - HistByte N-1)
2	80	TD1	高半字节 8 表示：后续不存在 TA2、TB2 和 TC2，只存在 TD2 低半字节 0 表示协议类型为 T=0
3	01	TD2	高半字节 0 表示后续不存在 TA3、TB3、TC3 和 TD3 低半字节 1 表示协议类型为 T=1
4 To 3+N	80	T1	类别指示字节，80 表示在可选的 COMPACT-TLV 数据对象中可能存在状态指示
	4F	Tk	应用标识符存在标识
	0C		长度
	RID		注册的应用供应商标识(RID) # A0 00 00 03 06h
	SS		标准字节
	C0.. C1		卡片名称字节
	00 00 00 00	RFU	RFU # 00 00 00 00h
4+N	UU	TCK	T0 至 Tk 的所有字符按位异或

表5 : ISO 14443 第 3 部分规定的 ATR 格式

例如：

MIFARE 1K 卡的 ATR = {3B 8F 80 01 80 4F 0C A0 00 00 03 06 03 00 01 00 00 00 00 6Ah}

ATR											
初始字符	T0	TD1	TD2	T1	Tk	长度	RID	标准	卡片名称	RFU	TCK
3Bh	8Fh	80h	01h	80h	4Fh	0Ch	A0 00 00 03 06h	03h	00h 01h	00 00 00 00h	6Ah

其中：

长度 (YY)	= 0Ch
RID	= A0 00 00 03 06h (PC/SC Workgroup)
标准 (SS)	= 03h (ISO 14443A, Part 3)
卡片名称 (C0 ... C1)	[00 01h] (MIFARE 1K)
	[00 02h] (MIFARE 4K)
	[00 03h] (MIFARE Ultralight)
	[00 26h] (MIFARE Mini)
	[FF 28h] JCOP 30
	FF SAK 未定义标签
	[00 36h] (MIFARE PLUS SL1_2K)
	[00 37h] (MIFARE PLUS SL1_4K)
	[00 38h] (MIFARE PLUS SL2_2K)
	[00 39h] (MIFARE PLUS SL2_4K)
	[00 3Ah] (MIFARE Ultralight C)

FW532 及更高版本的其他 ATR 支持

5.2.1.2. ATR 信息格式（适用于 ISO 14443-4 PICC）

<table>
<tr><th>字节</th><th>值（十六进制）</th><th>标记</th><th>说明</th></tr>
<tr><td>0</td><td>3B</td><td>初始字符</td><td></td></tr>
<tr><td>1</td><td>8N</td><td>T0</td><td>高半字节 8 表示：后续不存在 TA1、TB1 和 TC1，只存在 TD1
低半字节 N 指出历史字符的个数（HistByte 0 - HistByte N-1)</td></tr>
<tr><td>2</td><td>80</td><td>TD1</td><td>高半字节 8 表示：后续不存在 TA2、TB2 和 TC2，只存在 TD2
低半字节 0 表示协议类型为 T=0</td></tr>
<tr><td>3</td><td>01</td><td>TD2</td><td>高半字节 0 表示后续不存在 TA3、TB3、TC3 和 TD3
低半字节 1 表示协议类型为 T=1</td></tr>
<tr><td rowspan="2">4
-
3 + N</td><td>XX</td><td>T1</td><td rowspan="2">历史字节：
ISO 14443A：
来自 ATS 响应的历史字节。参考 ISO 14443-4 标准。

ISO 14443B：
<table><tr><th>Byte1-4</th><th>Byte5-7</th><th>Byte8</th></tr><tr><td>ATQB 的应用数据</td><td>ATQB 的协议信息字符</td><td>高半字节 =ATTRIB 命令的 MBLI；低半字节 (RFU)=0</td></tr></table></td></tr>
<tr><td>XX
XX
XX</td><td>Tk</td></tr>
<tr><td>4+N</td><td>UU</td><td>TCK</td><td>T0 至 Tk 的所有字符按位异或</td></tr>
</table>

表6 : ISO 14443 第 4 部分规定的 ATR 格式

例 1：考虑来自 MIFARE DESFire 的 ATR 如下：

DESFire (ATR) = 3B 81 80 01 80 80h (6 个字节的 ATR)

注：*使用 APDU "FF CA 01 00 00h"来区分是符合 ISO 14443A-4 的 PICC 还是符合 ISO 14443B-4 的 PICC，并且如果有的话，取回完整的 ATS。符合 ISO 14443A-3 或 ISO 14443B-3/4 类的 PICC 会返回 ATS。*

APDU 命令 = FF CA 01 00 00h

APDU 响应 = 06 75 77 81 02 90 00h

ATS = {06 75 77 81 02 80h}

例 2：考虑来自 EZ-Link 的 ATR 如下：

EZ-Link (ATR) = 3B 88 80 01 1C 2D 94 11 F7 71 85 00 BEh

ATQB 的应用数据 = 1C 2D 94 11h

ATQB 的协议信息 = F7 71 85h

ATTRIB 的 MBLI = 00h

5.2.2. 非接触接口的私有 APDU

5.2.2.1. Get data

此命令用于返回“已建立连接的卡片”的序列号或 ATS。

命令

Command	Class	INS	P1	P2	Le
Get Data	FFh	CAh	00h 01h	00h	00h (Full Length)

如果 P1 = 00h，执行 Get UID Response

Response	UID	…	…	UID	SW1	SW2
Result	LSB			MSB		

如果 P1 = 01h，执行 Get ATS Response (仅适用于符合 ISO 14443 的 A 类卡)

Response	Data Out		
Result	ATS	SW1	SW2

响应码

Results	SW1 SW2	含义
成功	90 00h	操作成功完成。
警告	62 82h	UID/ATS 的末尾先于 Le 字符到达（Le 大于 UID 的长度）
错误	6C XXh	长度错误（错误的 Le 数：‘XX’表示确切的数字）如果 Le 小于 UID 的长度
错误	63 00h	操作失败
错误	6A 81h	不支持此功能

例 1: 获取已建立连接的 PICC 的序列号

UINT8 GET_UID[5] = {FF CA 00 00 00h};

例 2: 获取已建立连接的 ISO14443-A PICC 的 ATS

UINT8 GET_ATS[5] = {FF CA 01 00 00h};

5.2.3. MIFARE 1K/4K 存储卡的 PICC 命令（T=CL 模拟）

5.2.3.1. Load authentication keys

此命令用于向读写器加载认证密钥。该认证密钥用于验证 MIFARE 1K/4K 存储卡的特定扇区。读写器提供了两种认证密钥位置：易失密钥位置和非易失密钥位置。

命令

Command	Class	INS	P1	P2	Le	Data In
Load Authentication Keys	FFh	82h	Key Structure	Key Number	06h	Key

其中：

Key Structure （1 个字节）

00h = 密钥被载入读写器的易失存储器

20h = 密钥被载入读写器的非易失存储器

其它 = 保留的

Key Number （1 个字节）

00h – 1Fh = 用于存储密钥的非易失存储器。密钥被永久地存在读写器中，即使读写器与电脑断开连接也不会被擦除。读写器的非易失存储器最多可以存储 32 个密钥。

20h (过程密钥) = 用于临时存储密钥的易失存储器。读写器与电脑断开连接的时候，密钥会被擦除。易失存储器只有一个。易失密钥可以用作不同会话的过程密钥。默认值 = FF FF FF FF FF FFh.

Key （6 个字节）

载入读写器的密钥值

例如： {FF FF FF FF FF FFh}

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

例 1:

向非易失存储器位置 05h 加载密钥{ FF FF FF FF FF FFh}

APDU = {FF 82 20 05 06 FF FF FF FF FF FFh}

向易失存储器位置 20h 加载密钥{ FF FF FF FF FF FFh }

APDU = {FF 82 00 20 06 FF FF FF FF FF FFh}

注意：

1. *基本上，应用程序需要了解所有正在被使用的密钥。出于安全方面的考虑，建议将所有需要的密钥存储在非易失存储器内。任何应用都无法读取易失性和非易失性存储器的内容。*
2. *直到读写器复位或下电，易失存储器的内容"过程密钥 20h"才会失效。过程密钥适于存储经常变化的密钥值，被存储在"内部 RAM"中，而非易失密钥存储在"EEPROM"中。EEPROM 相对于内部RAM 速度稍慢。*
3. *不建议使用"非易失密钥位置 00-1Fh"来存储任何经常变化的"临时密钥"。"非易失密钥"主要是用于存储不经常变化的"密钥值"。如果"密钥值"会不时的变化，则可以将其存储在"易失密钥位置 20h"*

5.2.3.2. Authentication for MIFARE 1K/4K

此命令用于通过存储在读写器内的密钥来验证 MIFARE 1K/4K 卡（PICC）。其中用到两种认证密钥，Type_A 和 Type_B。

命令

Command	Class	INS	P1	P2	P3	Data In
Authentication 6 Bytes (Obsolete)	FFh	88h	00h	Block Number	Key Type	Key Number

Command	Class	INS	P1	P2	Lc	Data In
Authentication 10 Bytes	FFh	86h	00h	00h	05h	Authenticate Data Bytes

其中：

Authenticate Data Bytes（5 个字节）

Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
Version 01h	00h	Block Number	Key Type	Key Number

其中：

Block Number （1 个字节）

待验证的存储块。

注： *一张 MIFARE 1K 卡分为 16 个扇区，每个扇区包含 4 个连续的块。例如：扇区 00h 包含块{00, 01, 02 和 03h}; 扇区 01h 包含块{04, 05, 06 和 07h}; 最后一个扇区 0Fh 包含块{3C, 3D, 3E 和 3Fh}。*

验证通过后，读取同一个扇区内的其他块不需要要再进行验证。详情请参考 MIFARE 1K/4K 卡标准。

Key Type （1 个字节）

60h = 该密钥被用作 Key A 密钥进行验证

61h = 该密钥被用作 Key B 密钥进行验证

Key Number （1 个字节）

00h – 1Fh =用于存储密钥的非易失存储器。密钥被永久地存在读写器中，即使读写器与电脑断开连接也不会被擦除。读写器的非易失存储器内最多可以存储 32 个密钥。.

20h（过程密钥） = 用于临时存储密钥的易失存储器。读写器与电脑断开连接的时候，密钥会被擦除。易失存储器只有一个。易失密钥可以用作不同会话的过程密钥。默认值 = FF FF FF FF FF FFh.

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

扇区 (共 16 个扇区，每个扇区包含 4 个连续的块)	数据块 (3 个块，每块 16 个字节)	尾部块 (1 个块，16 个字节)
Sector 0	00h ~ 02h	03h
Sector 1	04h ~ 06h	07h
..		
..		
Sector 14	38h ~ 0Ah	3Bh
Sector 15	3Ch ~ 3Eh	3Fh

1 KB

表7 : MIFARE 1K 卡的内存结构

扇区 (共 32 个扇区，每个扇区包含 4 个连续的块)	数据块 (3 个块，每块 16 个字节)	尾部块 (1 个块，16 个字节)
Sector 0	00h ~ 02h	03h
Sector 1	04h ~ 06h	07h
...		
...		
Sector 30	78h ~ 7Ah	7Bh
Sector 31	7Ch ~ 7Eh	7Fh

2 KB

扇区 (共 32 个扇区，每个扇区包含 4 个连续的块)	数据块 (3 个块，每块 16 个字节)	尾部块 (1 个块，16 个字节)
Sector 32	80h ~ 8Eh	8Fh
Sector 33	90h ~ 9Eh	9Fh
...		
...		
Sector 38	E0h ~ EEh	EFh
Sector 39	F0h ~ FEh	FFh

2 KB

表8 : MIFARE 4K 卡的内存结构

例 1：

通过下列特征验证 Block 04h：Key A, key number 00h, PC/SC V2.01（弃用）。

APDU = { FF 88 00 04 60 00h }

例 2：

类似于前面的例子，通过下列特征验证 Block 04h：Key A, key number 00h, PC/SC V2.07。

APDU = { FF 86 00 00 05 01 00 04 60 00h }

注: *由于 MIFARE® Ultralight 的用户数据区域可以自由访问，所以 MIFARE® Ultralight 不需要通过验证。*

字节号	0	1	2	3	页
序列号	SN0	SN1	SN2	BCC0	0
序列号	SN3	SN4	SN5	SN6	1
内部/锁	BCC1	Internal	Lock0	Lock1	2
OTP	OPT0	OPT1	OTP2	OTP3	3
数据读/写	Data0	Data1	Data2	Data3	4
数据读/写	Data4	Data5	Data6	Data7	5
数据读/写	Data8	Data9	Data10	Data11	6
数据读/写	Data12	Data13	Data14	Data15	7
数据读/写	Data16	Data17	Data18	Data19	8
数据读/写	Data20	Data21	Data22	Data23	9
数据读/写	Data24	Data25	Data26	Data27	10
数据读/写	Data28	Data29	Data30	Data31	11
数据读/写	Data32	Data33	Data34	Data35	12
数据读/写	Data36	Data37	Data38	Data39	13
数据读/写	Data40	Data41	Data42	Data43	14
数据读/写	Data44	Data45	Data46	Data47	15

512 位
或
64 字节

表9 : MIFARE Ultralight 卡的内存结构

5.2.3.3. Read binary blocks

此命令用于从 PICC 卡片中获取多个"数据块"。执行 Read Binary Blocks 命令前，必须先对数据块/尾部块进行验证.

命令

Command	Class	INS	P1	P2	Le
Read Binary Blocks	FFh	B0h	00h	Block Number	Number of Bytes to Read

其中：

Block Number （1 个字节）

起始块

Number of Bytes to Read （1 个字节）

MIFARE 1K/4K 卡的待读字节的长度是 16 字节的倍数；MIFARE Ultralight 卡的待读字节的长度是 4 字节的倍数

MIFARE Ultralight 卡的待读字节数最大为 16。

MIFARE 1K 卡的待读字节数最大为 48。（多块模式：3 个连续的块）

MIFARE 4K 卡的待读字节数最大为 240。（多块模式：15 个连续的块）

例 1：10h（16 个字节）；仅起始块。（单块模式）

例 2： 40h（64 个字节）；从起始块至起始+3 块。（多块模式）

注：*出于安全因素考虑，多块模式仅用于访问数据块。尾部块不能在多块模式下被访问，请使用单块模式对其进行访问。*

响应

Response	Data Out		
Result	Data (Multiple of 4 or 16 bytes)	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

例 1： 从二进制块 04h 中读取 16 个字节（MIFARE 1K 或 4K）

APDU = { FF B0 00 04 10h }

例 2： 从二进制块 80h 开始读取 240 个字节（MIFARE 4K）。块 80h 至块 8Eh（15 个块）

APDU = { FF B0 00 80 F0 }

5.2.3.4. Update binary blocks

此命令用于向 PICC 卡写入多个数据块。执行 Update Binary Blocks 命令前，必须先对数据块/尾部块进行验证。

命令

Command	Class	INS	P1	P2	Le	Data In
Update Binary Blocks	FFh	D6h	00h	Block Number	Number of Bytes to Update	Block Data (Multiple of 16 Bytes)

其中：

Block Number （1 个字节）

起始块

Block Data 16 字节的整数倍 + 2 个字节或 6 个字节；要写入二进制块的数据

Number of Bytes to Read （1 个字节）MIFARE 1K/4K 卡的待读字节的长度是 16 字节的倍数；MIFARE Ultralight 卡的待读字节的长度是 4 字节的倍数

MIFARE Ultralight 卡的待读字节数最大为 16。

MIFARE 1K 卡的待读字节数最大为 48。（多块模式：3 个连续的块）

MIFARE 4K 卡的待读字节数最大为 240。（多块模式：15 个连续的块）

例 1：10h（16 个字节）；仅起始块。（单块模式）

例 2：30h（48 个字节）；从起始块至起始+2 块。（多块模式）

注：*出于安全因素考虑，多块模式仅用于访问数据块。尾部块不能在多块模式下被访问，请使用单块模式对其进行访问。*

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

例 1：将 MIFARE 1K/4K 卡中二进制块 04h 的数据更新为{00 01 .. 0Fh}

APDU = { FF D6 00 04 10 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0Fh }

例 2：将 MIFARE Ultralight 卡中二进制块 04h 的数据更新为{ 00 01 02 03h }

APDU = {FF D6 00 04 04 00 01 02 03h }

5.2.3.5. Value block operation (Increment, Decrement, Store)

此命令用于对基于数值的交易进行操作（例如：增加值块的值等）。

命令

Command	Class	INS	P1	P2	Lc	Data In	
Value Block Operation	FFh	D7h	00h	Block Number	05h	VB_OP	VB_Value (4 Bytes) {MSB…LSB}

其中：

Block Number （1 个字节）
待操作的值块

VB_OP （1 个字节）
值块操作.
00h =将 VB_Value 存入该块，然后该块变为值块。
01h =使值块的值增加 VB_Value。此命令仅适用于对值块的操作。
02h =使值块的值减少 VB_Value。此命令仅适用于对值块的操作。

VB_Value 4 个字节；进行操作的数值，是一个有符号长整数。

例 1：Decimal - 4 = { FF FF FF FCh }

VB_Value			
MSB			LSB
FFh	FFh	FFh	FCh

例 2：Decimal 1 = { 00 00 00 01h }

VB_Value			
MSB			LSB
00h	00h	00h	01h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功
= 63 00h 表示操作失败

5.2.3.6. Read value block

此命令用于从值块中获取数值，仅适用于对值块的操作。

命令

Command	Class	INS	P1	P2	Le
Read Value Block	FFh	B1h	00h	Block Number	00h

其中：

Block Number （1 个字节）

待访问的值块

响应

Response	Data Out		
Result	Value {MSB … LSB}	SW1	SW2

其中：

Value （4 个字节）

卡片返回的值，是一个有符号长整数。

例 1：Decimal - 4 = { FF FF FF FCh }

VB_Value			
MSB			LSB
FFh	FFh	FFh	FCh

例 2：Decimal 1 = { 00 00 00 01h }

VB_Value			
MSB			LSB
00h	00h	00h	01h

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

5.2.3.7. Copy value block

此命令用于将一个值块中的值复制到另外一个值块。

命令

Command	Class	INS	P1	P2	Lc	Data In	
Copy Value Block	FFh	D7h	00h	Source Block Number	02h	03h	Target Block Number

其中：

Source Block Number （1 个字节）

源值块的编号，源值块中的值会被复制到目标值块。

Target Block Number （1 个字节）

目标值块的编号，来自源值块的数值会被复制到该值块。源值块和目标值块必须位于同一个扇区。

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

例 1： 将数值 “1”存入块 05h

APDU = {FF D7 00 05 05 00 00 00 00 01h }

例 2：读取值块 05h

APDU = {FF B1 00 05 00h }

例 3：复制值块 05h 的值到值块 06h

APDU = {FF D7 00 05 02 03 06h }

例 4：使值块 05h 的值增加 “5”

APDU = {FF D7 00 05 05 01 00 00 00 05h }

5.2.4. 访问符合 PC/SC 标准的标签（ISO 14443-4)

基本上，所有符合 ISO 14443-4 标准的卡片（PICC 卡）都可以理解 ISO 7816-4 规定的 APDU。ACR1281U-C1 读写器与符合 ISO 14443-4 标准的卡片进行通信时，需要对 ISO 7816-4 规定的 APDU 和响应进行转换。ACR1281U-C1 会在内部处理 ISO 14443 第 1-4 部分协议。

MIFARE 1K, 4K, MINI 和 Ultralight 标签是通过 T=CL 模拟进行支持的。只要简单地将 MIFARE 标签视作标准的 ISO 14443-4 标签。更多相关信息，请参阅 **MIFARE 1K/4K 存储卡的 PICC 命令（T=CL 模拟）** – MIFARE1K/4K 存储卡的 PICC 命令。

命令

Command	Class	INS	P1	P2	Lc	Data In	Le
ISO 7816 Part 4 Command					Length of the Data In		Expected Length of the Response Data

响应

Response	Data Out	
Result	SW1	SW2

其中：

SW1 SW2 = 90 00h 表示操作成功

= 63 00h 表示操作失败

典型的操作顺序为：

1. 出示标签，并与 PICC 接口建立连接
2. 读取/更新标签的存储内容

步骤 1： 连接标签

标签的 ATR 为 3B 88 80 01 00 00 00 00 33 81 81 00 3Ah

其中，

ATQB 应用数据 = 00 00 00 00h，ATQB 协议信息 = 33 81 81h。这是一个 ISO 14443-4 Type B 标签。

步骤 2：发送 APDU，取随机数

<< 00 84 00 00 08h

>> 1A F7 F3 1B CD 2B A9 58 [90 00h]

注：*对于 ISO 14443-4 Type A 标签来说，ATS 可以通过 APDU "FF CA 01 00 00h"来获取。*

例如：ISO 7816-4 APDU

从 ISO 14443-4 Type B PICC（ST19XR08E）中读取 8 个字节

APDU = { 80 B2 80 00 08h }

Class = 80h; INS = B2h; P1 = 80h; P2 = 00h;

Lc = None; Data In = None; Le = 08h

Answer: 00 01 02 03 04 05 06 07 [$90 00h]

5.2.5. 访问 MIFARE DESFire 标签（ISO 14443-4）

MIFARE® DESFire 支持 ISO7816-4 APDU 包模式和本地模式。一旦 DESFire 标签被激活，发送至 DESFire 标签的第一个 APDU 就会确定“命令的模式”。如果第一个 APDU 采用“本地模式”，则其余的 APDU 都必须是“本地模式”。同样地，如果第一个 APDU 采用“ISO 7816-4 APDU 包模式”，则其余的 APUDU 都必须是“ISO 7816-4 APDU 包模式”。

例 1：MIFARE DESFire ISO 7816-4 APDU 包。

从 ISO 14443-4 Type A PICC (DESFire)中读取八个字节的随机数

APDU = {90 0A 00 00 01 00 00h}

Class = 90h; INS = 0Ah (DESFire Instruction); P1 = 00h; P2 = 00h

Lc = 01h; Data In = 00h; Le = 00h (Le = 00h for maximum length)

应答： 7B 18 92 9D 9A 25 05 21h [$91AFh]

注：*状态码{91 AFh}由 MIFARE DESFire 标准定义，详情请参阅 MIFARE DESFire 标准。*

例 2：MIFARE DESFire 分页链接 (ISO 7816 APDU 包模式)

在本例中，应用涉及到"分页链接"。

要获得 DESFire 卡的版本号：

步骤 1：发送 APDU {90 60 00 00 00h}来获取第一个数据页。 INS=60h

应答：04 01 01 00 02 18 05 91 AFh [$91AFh]

步骤 2：发送 APDU {90 AF 00 00 00h}来获取第二个数据页。 INS=AFh
应答：04 01 01 00 06 18 05 91 AFh [$91AFh]

步骤 3：发送 APDU {90 AF 00 00 00h} 来获取最后一个数据页。INS=AFh

应答：04 52 5A 19 B2 1B 80 8E 36 54 4D 40 26 04 91 00h [$9100h]

例 3：MIFARE DESFire 本地命令。

若本地 DESFire 命令更易于操作，则我们可以向读写器发送不带 ISO7816 包的本地 DESFire 命令。

从 ISO 14443-4 Type A PICC (DESFire)中读取八个字节的随机数

APDU = {0A 00h}

应答：AF 25 9C 65 0C 87 65 1D D7h [$1DD7h]

其中，第一个字节“AF”是 DESFire 卡片返回的状态码。

应用程序可以对括号中的数据[$1DD7h]予以忽略。

例 4：MIFARE DESFire 分页链接 (本地模式)

在本例中，应用涉及到"分页链接"。

要获得 DESFire 卡的版本号：

步骤 1：发送 APDU {60h} 来获取第一个数据页。INS=60h

应答：AF 04 01 01 00 02 18 05h [$1805h]

步骤 2：发送 APDU {AFh} 来获取第二个数据页。INS=AFh
应答：AF 04 01 01 00 06 18 05h [$1805h]

步骤 3：发送 APDU {AFh} 来获取最后一个数据页。INS=AFh

应答：00 04 52 5A 19 B2 1B 80 8E 36 54 4D 40 26 04h [$2604h]

注：*在 DESFire 本地模式下，如果响应报文的长度大于 1，则不会在响应中增加状态码[90 00h]。但是如果响应报文的长度小于 2，则会根据 PC/SC 的要求在响应中增加状态码[90 00h]。最短的响应长度为 2。*

5.3. 外设控制

读写器的外设控制命令通过使用 PC_to_RDR_Escape 来实现。

注：*驱动程序自动增加 Class、INS 和 P1。*

5.3.1. 获取固件版本（Get Firmware Version）

此命令用于获取读写器的固件信息。

Get Firmware Version 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
Get Firmware Version	E0h	00h	00h	18h	00h

Get Firmware Version 的响应结构

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	Number of bytes to be received	Firmware Version

例如：

响应 = E1 00 00 00 0F 41 43 52 31 32 38 31 55 5F 56 35 30 33 2E 31

固件版本 (HEX) = 41 43 52 31 32 38 31 55 5F 56 35 30 33 2E 31

固件版本 (ASCII) = “ACR1281U_V503.1”

5.3.2. LED 控制（LED Control）

此命令用于控制 LED 的输出。

LED Control 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
LED Control	E0h	00h	00h	29h	01h	LED Status

其中：

LED 状态 （1 个字节）

LED 状态	描述	说明
Bit 0	红色 LED	1 = 开 0 = 关
Bit 1	绿色 LED	1 = 开 0 = 关
Bit 2 – 7	RFU	RFU

LED Control 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	LED Status

5.3.3. LED 状态（LED Status）

此命令用于检查当前 LED 的状态。

LED Status 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
LED Status	E0h	00h	00h	29h	00h

LED Status 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	LED Status

其中：

LED 状态 （1 个字节）

LED 状态	描述	说明
Bit 0	红色 LED	1 = 开 0 = 关
Bit 1	绿色 LED	1 = 开 0 = 关
Bit 2 – 7	RFU	RFU

5.3.4. 蜂鸣器控制（Buzzer Control）

此命令用于控制蜂鸣器的输出。

Buzzer Control 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
Buzzer Control	E0h	00h	00h	28h	01h	Buzzer On Duration

其中：

蜂鸣器鸣响时间 （1 个字节）

00h = 关闭

01 – FFh = 持续时间（单位：10 ms）

Buzzer Control 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	00h

5.3.5. 设置 LED 和蜂鸣器默认操作（Set Default LED and Buzzer Behaviors）

此命令用于设置 LED 和蜂鸣器的默认操作。

Set LED and Buzzer Behaviors 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
Set Default LED and Buzzer Behaviors	E0h	00h	00h	21h	01h	Default Behaviors

其中：

默认操作: （1 个字节）

LED Status	描述	说明
Bit 0	ICC 激活状态 LED	显示 ICC 界面的激活状态 1 = 启用 0 = 停用
Bit 1	PICC 轮询状态 LED	显示 PICC 轮询状态 1 = 启用 0 = 停用
Bit 2	RFU	RFU
Bit 3	RFU	RFU
Bit 4	卡片插拔事件蜂鸣器	每次检测到卡片插入或者卡片移出就会发出哔的一声（适用于 ICC 和 PICC） 1 = 启用 0 = 停用
Bit 5	非接触芯片复位指示蜂鸣器	非接触芯片复位时发出哔的一声。 1 = 启用 0 = 停用
Bit 6	独享模式状态蜂鸣器。ICC 或 PICC 界面只有一个可以被激活。	独享模式被激活时会发出哔的一声。 1 = 启用 0 = 停用
Bit 7	卡片操作闪烁 LED	LED 在卡片（PICC 或 ICC）被访问时会闪烁。

注：*默认的操作值 = FBh*

Set LED and Buzzer Behaviors 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Default Behaviors

5.3.6. 读取 LED 和蜂鸣器默认操作 (Read Default LED and Buzzer Behaviors)

此命令用于读取当前 LED 和蜂鸣器的默认操作。

Read LED and Buzzer Behaviors 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
Read Default LED and Buzzer Behaviors	E0h	00h	00h	21h	00h

Read LED and Buzzer Behaviors 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Default Behaviors

其中：

默认操作:（1 个字节）

LED 状态	描述	说明
Bit 0	ICC 激活状态 LED	显示 ICC 界面的激活状态 1 = 启用 0 = 停用
Bit 1	PICC 轮询状态 LED	显示 PICC 轮询状态 1 = 启用 0 = 停用
Bit 2	RFU	RFU
Bit 3	RFU	RFU
Bit 4	卡片插拔事件蜂鸣器	每次检测到卡片插入或者卡片移出就会发出哔的一声（适用于 ICC 和 PICC）。 1 = 启用 0 = 停用
Bit 5	非接触芯片复位指示蜂鸣器	非接触芯片复位时发出哔的一声。 1 = 启用 0 = 停用
Bit 6	独享模式状态蜂鸣器。ICC 或 PICC 界面只有一个可以被激活。	独享模式被激活时会发出哔的一声。 1 = 启用 0 = 停用
Bit 7	卡片操作闪烁 LED	LED 在卡片（PICC 或 ICC）被访问时会闪烁。

注：*默认的操作值 = FBh*

5.3.7. 初始化卡片插入计数器（Initialize Card Insertion Counter）

此命令用于初始化卡片插入/检测计数器。

Initialize Card Insertion Counter 的命令结构（9 字节）

Command	Class	INS	P1	P2	Lc	Data In			
Initialize Cards Insertion Counter	E0h	00h	00h	09h	04h	ICC Cnt (LSB)	ICC Cnt (MSB)	PICC Cnt (LSB)	PICC Cnt (MSB)

其中：

ICC Cnt (LSB) （1 个字节）
ICC 插入计数器 (LSB)

ICC Cnt (MSB) （1 个字节）
ICC 插入计数器 (MSB)

PICC Cnt (LSB) （1 个字节）
PICC 插入计数器 (LSB)

PICC Cnt (MSB) （1 个字节）
PICC 插入计数器 (MSB)

Initialize Card Insertion Counter 的响应结构（5 字节）

Response	Class	INS	P1	P2	Le
Result	E1h	00h	00h	00h	00h

5.3.8. 读取卡片插入计数器（Read Card Insertion Counter）

此命令用于检查卡片插入/检测计数器的值。

Read Card Insertion Counter 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
Read Cards Insertion Counter	E0h	00h	00h	09h	00h

Read Card Insertion Counter 的响应结构（9 字节）

Response	Class	INS	P1	P2	Le	Data Out			
Result	E1h	00h	00h	00h	04h	ICC Cnt (LSB)	ICC Cnt (MSB)	PICC Cnt (LSB)	PICC Cnt (MSB)

其中：

ICC Cnt (LSB) （1 个字节）
ICC 插入计数器 (LSB)

ICC Cnt (MSB) （1 个字节）
ICC 插入计数器 (MSB)

PICC Cnt (LSB) （1 个字节）
PICC 插入计数器 (LSB)

PICC Cnt (MSB) （1 个字节）
PICC 插入计数器 (MSB)

5.3.9. 更新卡片插入计数器（Update Card Insertion Counter）

此命令用于更新卡片插入/检测计数器的值。

Update Card Insertion Counter 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
Update Card Insertion Counter	E0h	00h	00h	0Ah	00h

Update Card Insertion Counter 的响应结构（9 字节）

Response	Class	INS	P1	P2	Le	Data Out			
Result	E1h	00h	00h	00h	04h	ICC Cnt (LSB)	ICC Cnt (MSB)	PICC Cnt (LSB)	PICC Cnt (MSB)

其中：

ICC Cnt (LSB) （1 个字节）
ICC 插入计数器 (LSB)

ICC Cnt (MSB) （1 个字节）
ICC 插入计数器 (MSB)

PICC Cnt (LSB) （1 个字节）
PICC 插入计数器 (LSB)

PICC Cnt (MSB) （1 个字节）
PICC 插入计数器 (MSB)

5.3.10. 设置自动 PICC 轮询（Set Automatic PICC Polling）

此命令用于设置读写器的轮询模式。

每当读写器连接到电脑的时候，读写器的 PICC 轮询功能就会启动 PICC 扫描，以确定 PICC 是否被放置于/移出了内置天线的范围。

我们可以发送一个命令来停用 PICC 轮询功能。该命令通过 PC/SC Escape Command 接口发送。为了满足节能要求，当 PICC 闲置，或者没有找到 PICC 的时候，提供了几种关闭天线场的特殊模式。在省电模式下，读写器会消耗更低的电能。

Set Automatic PICC Polling 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
Set Automatic PICC Polling	E0h	00h	00h	23h	01h	Polling Setting

其中：

轮询设置　　（1 个字节）

轮询设置	描述	说明
Bit 0	自动 PICC 轮询	1 = 启用 0 = 停用
Bit 1	如果没有找到 PICC，关闭天线场	1 = 启用 0 = 停用
Bit 2	如果 PICC 闲置，关闭天线场	1 = 启用 0 = 停用
Bit 3	RFU	RFU
Bit 5 – 4	PICC 轮询间隔	Bit 5 – Bit 4： 0 – 0 = 250 毫秒 0 – 1 = 500 毫秒 1 – 0 = 1000 毫秒 1 – 1 = 2500 毫秒
Bit 6	RFU	RFU
Bit 7	执行 ISO14443A 第 4 部分	1 = 启用 0 = 停用

注：*轮询设置的默认值 = 8Fh。*

Set Automatic PICC Polling 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Polling Setting

注：

1. *建议启用"如果 PICC 闲置，关闭天线场"选项，这样闲置的 PICC 就不会一直暴露在天线场中，防止 PICC 发热。*
2. *PICC 轮询间隔时间越长，节能效果越好。然而，PICC 轮询的响应时间也会增加。在节能状态下，空闲时的电流消耗约为 60 mA；而在非节能状态下，空闲时的电流消耗约为 130 mA。空闲时的电流消耗=PICC 处于闲置状态。*
3. *读写器会自动激活"ISO 14443A-4 PICC"的 ISO 14443A-4 模式。B 类的 PICC 不受此选项影响。*
4. *JCOP30 卡片有两种模式：ISO 14443A-3 (MIFARE 1K)和 ISO 14443A-4 模式。一旦 PICC 被激活，应用必须要决定选择哪一种模式。*

5.3.11. 读取自动 PICC 轮询（Read Automatic PICC Polling）

此命令用于检查当前的自动 PICC 轮询设置。

Read Automatic PICC Polling 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
Read Automatic PICC Polling	E0h	00h	00h	23h	00h

Read Automatic PICC Polling 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Polling Setting

其中：

轮询设置 (1 个字节)

轮询设置	描述	说明
Bit 0	自动 PICC 轮询	1 = 启用 0 = 停用
Bit 1	如果没有找到 PICC，关闭天线场	1 = 启用 0 = 停用
Bit 2	如果 PICC 闲置，关闭天线场	1 = 启用 0 = 停用
Bit 3	RFU	RFU
Bit 5 – 4	PICC 轮询间隔	Bit 5 – Bit 4： 0 – 0 = 250 毫秒 0 – 1 = 500 毫秒 1 – 0 = 1000 毫秒 1 – 1 = 2500 毫秒
Bit 6	RFU	RFU
Bit 7	执行 ISO14443A 第 4 部分	1 = 启用 0 = 停用

注：*轮询设置的默认值 = 8Fh。*

5.3.12. 手动 PICC 轮询（Manual PICC Polling）

此命令用于确定是否有 PICC 处于读写器的检测范围内。在自动 PICC 轮询功能停用时，可以使用此命令。

Manual PICC Polling 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
Manual PICC Polling	E0h	00h	00h	22h	01h	0Ah

Manual PICC Polling 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Status

其中：

状态 （个字节）

00h = 检测到 PICC

FFh = 未检测到 PICC

5.3.13. 设置 PICC 操作参数（Set PICC Operating Parameter）

此命令用于设置 PICC 的操作参数。

Set PICC Operating Parameter 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
Set the PICC Operating Parameter	E0h	00h	00h	20h	01h	Operating Parameter

其中：

操作参数 (1 个字节)

Operating Parameter	参数	说明	选项
Bit 0	ISO 14443 Type A	PICC 轮询中待检测的标签的类别	1 = 检测 0 = 跳过
Bit 1	ISO 14443 Type B		1 = 检测 0 = 跳过
Bit 2 – 7	RFU	RFU	RFU

注：*操作参数的默认值 = 03h。*

Set PICC Operating Parameter 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Operating Parameter

5.3.14. 读取 PICC 操作参数（Read PICC Operating Parameter）

此命令用于检查当前 PICC 的操作参数。

Read PICC Operating Parameter 的命令结构（5 字节）

Command	Class	INS	P1	P2	Lc
Read the PICC Operating Parameter	E0h	00h	00h	20h	00h

Read PICC Operating Parameter 的响应结构（6 字节）

Response	Class	INS	P1	P2	Le	Data Out
Result	E1h	00h	00h	00h	01h	Operating Parameter

其中：

操作参数 (1 个字节)

Operating Parameter	参数	说明	选项
Bit 0	ISO14443 Type A	PICC 轮询中待检测标签的类别	1 = 检测 0 = 跳过
Bit 1	ISO14443 Type B		1 = 检测 0 = 跳过
Bit 2 – 7	RFU	RFU	RFU

注：*操作参数的默认值 = 03h。*

5.3.15. 设置独享模式（Set Exclusive Mode）

此命令用于设置读写器进入/离开独享模式。

Set Exclusive Mode 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc	Data In
Set Exclusive Mode	E0h	00h	00h	2Bh	01h	New Mode Configuration

Set Exclusive Mode 的响应结构（7 字节）

Response	Class	INS	P1	P2	Le	Data Out	
Result	E1h	00h	00h	00h	02h	Mode Configuration	Current Mode Configuration

其中：

独享模式 （1 个字节）

00h = 共享模式：ICC 和 PICC 接口可以同时工作。

01h = 独享模式：插入 ICC 卡时自动轮询功能和天线关闭，PICC 停用（默认）

5.3.16. 读取独享模式（Read Exclusive Mode）

此命令用于检查当前独享模式的设置。

Read Exclusive Mode 的命令结构（6 字节）

Command	Class	INS	P1	P2	Lc
Read Exclusive Mode	E0h	00h	00h	2Bh	00h

Read Exclusive Mode 的响应结构（7 字节）

Response	Class	INS	P1	P2	Le	Data Out	
Result	E1h	00h	00h	00h	02h	Mode Configuration	Current Mode Configuration

其中：

独享模式 （1 个字节）

00h = 共享模式：ICC 和 PICC 接口可以同时工作。

01h = 独享模式：插入 ICC 卡时自动轮询功能和天线关闭，PICC 停用（默认）

5.3.17. 设置自动 PPS（Set Auto PPS）

每次识别出 PICC，读写器都会尝试更改由最大连接速度定义的 PCD 和 PICC 间的通信速率。若卡片不支持建议的连接速度，读写器会尝试以较慢的速度与卡片建立连接。

Set Auto PPS 的命令结构（6 个字节）

Command	Class	INS	P1	P2	Lc	Data In
Set Auto PPS	E0h	00h	00h	24h	01h	Max Speed

Set Auto PPS 命令的响应结构（7 个字节）

Response	Class	INS	P1	P2	Le	Data Out	
Result	E1h	00h	00h	00h	02h	Max Speed	Current Speed

其中：

Max Speed 最大速度（1 字节）

Current Speed 当前速度（1 字节）

00h = 106 Kbps; 默认设置，相当于没有设置自动 PPS

01h = 212 Kbps

02h = 424 Kbps

03h = 848 Kbps

注：

1. *通常来讲，应用程序应当知道正在使用的 PICC 的最大连接速率，周围环境也会对最大可达速率有所影响。读写器只以建议的通信速率与 PICC 进行通信。如果 PICC 或周围环境不能满足建议的通信速率的要求，PICC 将不可访问。*
2. *如果较高的速率设置会影响读写器的性能，请切换回较低的速率设置。*

5.3.18. 读取自动 PPS（Read Auto PPS）

此命令用于查看当前的自动 PPS 设置。

Read Auto PPS 的命令结构（5 个字节）

Command	Class	INS	P1	P2	Lc
Read Auto PPS	E0h	00h	00h	24h	00h

Read Auto PPS 命令的响应结构（7 个字节）

Response	Class	INS	P1	P2	Le	Data Out	
Result	E1h	00h	00h	00h	02h	Max Speed	Current Speed

其中：

Max Speed 最大速度（1 字节）

Current Speed 当前速度（1 字节）

00h = 106 Kbps; 默认设置，相当于没有设置自动 PPS

01h = 212 Kbps

02h = 424 Kbps

03h = 848 Kbps

5.3.19. 读取序列号（Read Serial Number）

此命令用于读取读写器的序列号。

注：*仅适用于 533.00 及以上版本的固件。*

Read Serial Number 的命令格式（5 字节）

Command	Class	INS	P1	P2	Lc
Read Serial Number	E0h	00h	00h	33h	00h

Read Serial Number 的响应格式

Response	Class	INS	P1	P2	Le	Data Out
Result	E1	00h	00h	00h	Len	Serial Number (N bytes)

附录A. 非接触式应用基本程序流

步骤 0：启动应用，读写器会不断地进行 PICC 轮询和标签扫描。一旦找到并检测到标签，相应的 ATR 会被发送到 PC。

步骤 1：通过 T=1 协议连接“ACR1281U PICC 接口”。

步骤 2：通过 APDU 交换访问 PICC。

..

步骤 N：断开“ACR1281U PICC 接口”的连接，关闭应用。

附录B. 扩展的 APDU 示例

卡片：ACOS7 (支持扩展的 APDU，回应响应）

写 CMD：80 D2 00 00 XX XX XXh

CLA = 80h

INS = D2h

P1 = 00h

P2 = 00h

Data Len = XX XX XXh

例 1：APDU 长度 = 263 字节

APDU 命令：

80D2000000010000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
202122232425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F40414243444546
4748494A4B4C4D4E4F505152535455565758595A5B5C5D5E5F606162636465666768696A6B6C6
D6E6F707172737475767778797A7B7C7D7E7F808182838485868788898A8B8C8D8E8F90919293
9495969798999A9B9C9D9E9FA0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B
8B9BABBBCBDBEBFC0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9D
ADBDCDDDEDFE0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFD
FEFFh

响应:

000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F20212223242526
2728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F404142434445464748494A4B4C4
D4E4F505152535455565758595A5B5C5D5E5F606162636465666768696A6B6C6D6E6F70717273
7475767778797A7B7C7D7E7F808182838485868788898A8B8C8D8E8F909192939495969798999A
9B9C9D9E9FA0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDB
EBFC0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDFE
0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF9000

例 2：APDU 长度 = 775 字节

APDU 命令：

80D2000000030000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F
202122232425262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F40414243444546
4748494A4B4C4D4E4F505152535455565758595A5B5C5D5E5F606162636465666768696A6B6C6
D6E6F707172737475767778797A7B7C7D7E7F808182838485868788898A8B8C8D8E8F90919293
9495969798999A9B9C9D9E9FA0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B
8B9BABBBCBDBEBFC0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9D
ADBDCDDDEDFE0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFD
FEFF000102030405060708090A0B0C0D0E0F101112131415161718191A1B1C1D1E1F2021222324
25262728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F404142434445464748494A4B
4C4D4E4F505152535455565758595A5B5C5D5E5F606162636465666768696A6B6C6D6E6F70717
27374757677787978 7A7B7C7D7E7F808182838485868788898A8B8C8D8E8F909192939495969798
99A9B9C9D9E9FA0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBC
BDBEBFC0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDE
DFE0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF0001020
30405060708090A0B0C0D0E0F10111213141516171819 1A1B1C1D1E1F202122232425262728292
A2B2C2D2E2F303132333435363738393A3B3C3D3E3F404142434445464748494A4B4C4D4E4F50

515253545565758595A5B5C5D5E5F606162636465666768696A6B6C6D6E6F7071727374757677
78797A7B7C7D7E7F80818283848586878889 8A8B8C8D8E8F909192939495969798999A9B9C9D9
E9FA0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDBEBFC0C1
C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDFE0E1E2E3
E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFFh

响应:

000102030405060708090A0B0C0D0E0F10111213141516171819 1A1B1C1D1E1F20212223242526
2728292A2B2C2D2E2F303132333435363738393A3B3C3D3E3F404142434445464748494A4B4C4
D4E4F505152535455565758595A5B5C5D5E5F606162636465666768696A6B6C6D6E6F70717273
7475767778797A7B7C7D7E7F80818283848586878889 8A8B8C8D8E8F909192939495969798999A
9B9C9D9E9FA0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDB
EBFC0C1C2C3C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDFE
0E1E2E3E4E5E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF00010203040
5060708090A0B0C0D0E0F10111213141516171819 1A1B1C1D1E1F202122232425262728292A2B2
C2D2E2F303132333435363738393A3B3C3D3E3F404142434445464748494A4B4C4D4E4F505152
535455565758595A5B5C5D5E5F606162636465666768696A6B6C6D6E6F70717273747576777879
7A7B7C7D7E7F80818283848586878889 8A8B8C8D8E8F909192939495969798999A9B9C9D9E9FA
0A1A2A3A4A5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDBEBFC0C1C2C3
C4C5C6C7C8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDFE0E1E2E3E4E5
E6E7E8E9EAEBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF000102030405060708090A
0B0C0D0E0F10111213141516171819 1A1B1C1D1E1F202122232425262728292A2B2C2D2E2F303
132333435363738393A3B3C3D3E3F404142434445464748494A4B4C4D4E4F50515253545556575
8595A5B5C5D5E5F606162636465666768696A6B6C6D6E6F707172737475767778797A7B7C7D7E
7F80818283848586878889 8A8B8C8D8E8F909192939495969798999A9B9C9D9E9FA0A1A2A3A4A
5A6A7A8A9AAABACADAEAFB0B1B2B3B4B5B6B7B8B9BABBBCBDBEBFC0C1C2C3C4C5C6C7C
8C9CACBCCCDCECFD0D1D2D3D4D5D6D7D8D9DADBDCDDDEDFE0E1E2E3E4E5E6E7E8E9EA
EBECEDEEEFF0F1F2F3F4F5F6F7F8F9FAFBFCFDFEFF9000h

附录C.　直接命令(Escape Command) 示例

例如：获取固件版本号（使用 PCSCDirectCommand.exe）

步骤 1：将 ACR1281 读写器连接到电脑上

步骤 2：打开 PCSCDirectCommand.exe

步骤 3：选择读写器以“Direct”模式连接，会显示出 ATR（如果存在卡片）或“No ATR retrieved (ATRLen = 0)”（如果没有卡片）

步骤 4：　输入命令：“20 79”

输入数据：“18 00”（用于获取固件版本的 APDU）

单击 Enter 将命令发送至读写器，然后检查 Response。

注：*软件开发工具包（SDK）中并未提供 PCSCDirectCommand.exe。如需了解更多信息，请与 ACS 联系。*

附录D. 支持的卡片类型

下面的表格总结了 GET_READER_INFORMATION 命令返回的卡片类型数据以及相对应的卡片类别。

卡片类型代码	卡片类型
00h	自动选择 T=0 或 T=1 通信协议
01h	I2C 存储卡 (1k, 2k, 4k, 8k and 16k bits)
02h	I2C 存储卡(32k, 64k, 128k, 256k, 512k 和 1024k bits)
03h	RFU
04h	RFU
05h	Infineon SLE4418 和 SLE4428
06h	Infineon SLE4432 和 SLE4442
07h	Infineon SLE4406, SLE4436 和 SLE5536
08h	Infineon SLE4404
09h	RFU

附录E. ACR128 兼容性

以下是 ACR128 读写器的一些功能，ACR1281U-C1 读写器对这些功能的实现方式不同或不支持这些功能。

功能	ACR128	ACR1281U-C1
1. 修改已激活的 PICC 的默认 FWI 和传输帧大小	1F 03 [数据：3 个字节]	不支持。
2. 收发器设置	20 04 06 [数据：3 个字节]	不支持。
3. PICC 设置	2A 0C [数据：12 个字节]	不支持。
4. PICC T=CL 数据交换错误处理	2C 02 [数据：1 个字节]	不支持。
5. 读取寄存器	19 01 [寄存器编号]	不支持。
6. 更新寄存器	1A 02 [寄存器编号] [值]	不支持。
7. 特定类型的 PICC 轮询	20 02 [数据：1 个字节] FF	20 01 [数据：1 个字节]
8. 蜂鸣器控制	28 01 [鸣响持续时间] 鸣响持续时间： 00 = 关闭 01 – FE = 鸣响持续时间 x 10 ms FF = 开启	28 01 [鸣响持续时间] 鸣响持续时间： 01 – FF = 鸣响持续时间 x 10 ms

功能	ACR128	ACR1281U-C1
9. 设置/读取默认的 LED 和蜂鸣器操作	设置：21 01 [数据：1 个字节] 读取：21 00 数据： Bit 0 = ICC 激活状态 Bit 1 = PICC 轮询状态 LED Bit 2 = PICC 激活状态蜂鸣器 Bit 3 = PICC PPS 状态蜂鸣器 Bit 4 = 卡片插拔时间蜂鸣器 Bit 5 = 非接触芯片复位指示蜂鸣器 Bit 6 = 独占模式状态蜂鸣器 Bit 7 = 卡片操作闪烁 LED	设置：21 01 [数据：1 个字节] 读取：21 00 数据： Bit 0 = ICC 激活状态 Bit 1 = PICC 轮询状态 LED Bit 2 = RFU Bit 3 = RFU Bit 4 =卡片插拔事件蜂鸣器 Bit 5 =非接触芯片复位指示蜂鸣器 Bit 6 =独占模式状态蜂鸣器 Bit 7 =卡片操作闪烁 LED
10. 设置/读取自动 PICC 轮询	设置：23 01 [数据：1 个字节] 读取：23 00 数据： Bit 0 = 自动 PICC 轮询 Bit 1 =如果没有找到 PICC，关闭天线场 Bit 2 =如果 PICC 闲置，关闭天线场 Bit 3 = 检测到 PICC 后激活 PICC Bit 4..5 = PICC 轮询间隔 Bit 6 = 测试模式 Bit 7 = 执行 ISO 14443A 第 4 部分	设置：23 01 [数据：1 个字节] 读取：23 00 数据： Bit 0 =自动 PICC 轮询 Bit 1 =如果没有找到 PICC，关闭天线场 Bit 2 =如果 PICC 闲置，关闭天线场 Bit 3 = RFU Bit 4..5 = PICC 轮询间隔 Bit 6 = RFU Bit 7 = 执行 ISO 14443A 第 4 部分

MIFARE、MIFARE Classic、MIFARE DESFire 和 MIFARE Ultralight 是 NXP B.V.的商标。
Windows 和 Windows Vista 是微软公司的商标。