



# 北京圆志科信 射频读写器

## 应用手册



# RW210Ax/Cx

地址：北京市通州区通胡大街 78 号京贸中心 1004D

电话：010-64389905

传真：010-89524306

Web: <http://www.yzrfid.com>

## 0.1 声明

本说明书是为了让用户更好的选择北京圆志科信电子科技有限公司的产品而提供的开发资料，不转让属于北京圆志科信电子科技有限公司或者第三者所有的知识产权，用户在确定使用本产品前，请根据自己实际需求对产品性能及其使用安全性等方面进行相应评估，北京圆志科信电子科技有限公司不承担因评估不当而造成的直接或间接损失，也不承担因此而带来的任何法律或经济责任。

北京圆志科信电子科技有限公司致力于为用户提供不断完善的服务与产品，保有对产品及其相应说明书更新的权利，如有变动，恕不另行通知，在确定购买此产品时，请预先联系北京圆志科信电子科技有限公司以确认是否为最新版本。

本说明书所有权归北京圆志科信电子科技有限公司所有，未经许可，不得翻印或复制全部或部分本资料内容。

## 0.2 更改历史记录

版本	描述	日期
V1.0	PDF 版本第一版发布	2017.12.29
V1.1	文档格式进行了调整，转化为简介+chm 方式	2018.03.05
V1.2	在原有基础上增加对 Ntag,Ultralight 的支持	2018.07.25
V1.3	增加了对 2 字节长度的描述，适用那些每帧数据超过 255 字节的特殊应用，需要注意的是 2 字节长度与 1 字节长度模块的固件不同，故型号不同，2 字节长度的模块型号在原默认 1 字节的型号上增加 L2	2018.12.25
V1.4	DII 描述增补对 RW210AH/CH 读卡器端口连接的描述	2021.02.05
V1.5	增加读卡器外观图片	2021.03.12

## 目 录

0.1 声明 .....	2
0.2 更改历史记录 .....	3
1 概述 .....	7
1.1 RW210Ax/Cx IC 卡读写器简介: .....	7
1.2 产品型号及之间的区别: .....	7
1.3 主要特点 .....	8
1.4 技术指标 .....	9
2 硬件连接 .....	10
2.1 读卡器外观 .....	10
2.1.1 RS232 串口读卡器 .....	10
2.1.2 USB 读卡器 .....	10
2.1.3 RS485 读卡器/TTL232 读卡器 .....	11
2.1.4 外壳尺寸图 .....	11
2.2 读卡器硬件连接 .....	12
2.2.1 RS232 串口连接 .....	12
2.2.2 RS485 连接 .....	12
2.2.3 TTL232 连接 .....	13
2.2.4 USB 非免驱读卡器连接 .....	13
2.2.5 USB 免驱读卡器连接 .....	14
2.3 上电状态 .....	15
3 通过读卡器操作卡片步骤简述 .....	15
3.1 MifareOne 卡操作流程 .....	15
3.2 Mifare Ultralight / Ntag 卡操作流程 .....	15
3.3 ISO14443 TYPE A T=CL CPU 卡操作流程 .....	16
3.4 Mifare Desfire EV0/EV1/EV2 卡操作流程 .....	16
3.5 ISO7816-4 卡操作流程 (可支持多个卡) .....	17
3.6 ISO7816-4 卡操作流程 (只支持 1 个卡) .....	17
4 底层数据通讯协议 .....	17
4.1 RS232/TTL232/RS485 协议 .....	17
4.2 系统函数指令 .....	19
通用系统函数: 设置波特率 .....	19
通用系统函数: 设置模块工作在 ISO14443 TYPE A 模式 .....	20
通用系统函数: 设置模块天线状态 .....	21
通用系统函数: 控制 LED 指示灯状态 .....	22
通用系统函数: 控制蜂鸣器蜂鸣时间 .....	23
通用系统函数: 读取软硬件版本号 .....	24
通用系统函数: 读取读卡器序列号 .....	24
通用系统函数: 设置读卡器的设备地址 .....	25
通用系统函数: 读取读卡器设备地址 .....	26

4.3 MifareOne 卡函数指令 .....	27
MF1 专用指令: MF1 卡寻卡 .....	27
MF1 专用指令: Mifare one 卡防冲突 .....	27
MF1 专用指令: Mifare one 卡选卡 .....	28
MF1 专用指令: Mifare one 卡密钥验证 .....	29
MF1 专用指令: Mifare one 卡读块 .....	30
MF1 专用指令: Mifare one 卡写块 .....	31
MF1 专用指令: Mifare one 卡初始化钱包 .....	32
MF1 专用指令: Mifare one 卡读钱包 .....	32
MF1 专用指令: Mifare one 卡充值 .....	33
MF1 专用指令: Mifare one 卡扣款 .....	34
MF1 专用指令: Mifare one 卡钱包备份步骤 1 .....	35
MF1 专用指令: Mifare one 卡钱包备份步骤 2 .....	36
MF1 专用指令: Mifare one 卡读连续 3 块 .....	37
MF1 专用指令: Mifare one 卡写连续 3 块 .....	37
MF1 专用指令: Mifare one 卡休眠 .....	38
4.4 Ultralight/Ntag 卡函数指令 .....	39
UL/Ntag 专用指令: Ultralight/Ntag 卡寻卡 .....	39
UL/Ntag 专用指令: Ultralight/Ntag 卡选卡 .....	40
UL/Ntag 专用指令: Ultralight/Ntag 卡读卡 .....	41
UL/Ntag 专用指令: Ultralight/Ntag 卡写卡 .....	42
Ntag21x 专用指令: Ntag21x 系列卡快速读取 .....	42
Ntag21x 专用指令: 读取 Ntag21x 系列卡片版本号 .....	43
Ntag21x 专用指令: Ntag21x 系列密钥认证 .....	44
Ntag21x 专用指令: 读取 Ntag21x 系列卡片读取 CNT .....	44
Ntag21x 专用指令: 读取 Ntag21x 系列卡片读取数字签名 .....	45
NFC 卡专用指令: NFC 类型 2 标签选择扇区 .....	46
UL/Ntag 专用指令: Ultralight/Ntag 卡休眠 .....	47
4.5 ISO14443A-4 CPU 卡函数指令 .....	47
TYPE A CPU 卡专用功能: ISO14443A-4 CPU 卡复位 .....	47
TYPE A CPU 卡专用功能: ISO14443A-4 CPU 卡发送 APDU(COS)指令 .....	48
TYPE A CPU 卡专用功能: ISO14443A-4 CPU 卡 PPS 升频 .....	49
TYPE A CPU 卡专用功能: ISO14443A-4 CPU 卡退出激活状态 .....	50
4.6 ISO7816-4 SAM 卡函数指令 (支持多个卡) .....	51
ISO7816-4 SAM/SIM 卡专用功能: 第 N 号 SAM 卡复位 .....	51
ISO7816-4 SAM/SIM 卡专用功能: 向第 N 号 SAM 卡发送 APDU(COS)指令 .....	52
ISO7816-4 SAM/SIM 卡专用功能: 向第 N 号 SAM 卡发送 PPS 请求 .....	52
4.7 ISO7816-4 SAM 卡函数指令 (只支持 1 个卡) .....	53
ISO7816-4 SAM/SIM 卡专用功能: SAM 卡初始化波特率 (只支持 1 个卡) .....	53
ISO7816-4 SAM/SIM 卡专用功能: SAM 卡复位 (只支持 1 个卡) .....	54

ISO7816-4 SAM/SIM 卡专用功能: SAM 卡发送 APDU(COS)指令 (只支持 1 个卡) .....	55
5 数据发送接收举例: .....	56
5.1 读卡器通用命令发送接收举例: .....	56
5.2 M1 卡发送接收举例: .....	57
5.3 Ultralight 卡发送接收举例: .....	58
5.4 ISO14443A-4 CPU 卡发送接收举例: .....	59
5.5 ISO7816-4 CPU/SAM/SIM 卡新指令发送接收举例: .....	59
5.6 ISO7816-4 CPU/SAM/SIM 卡旧指令发送接收举例: .....	59

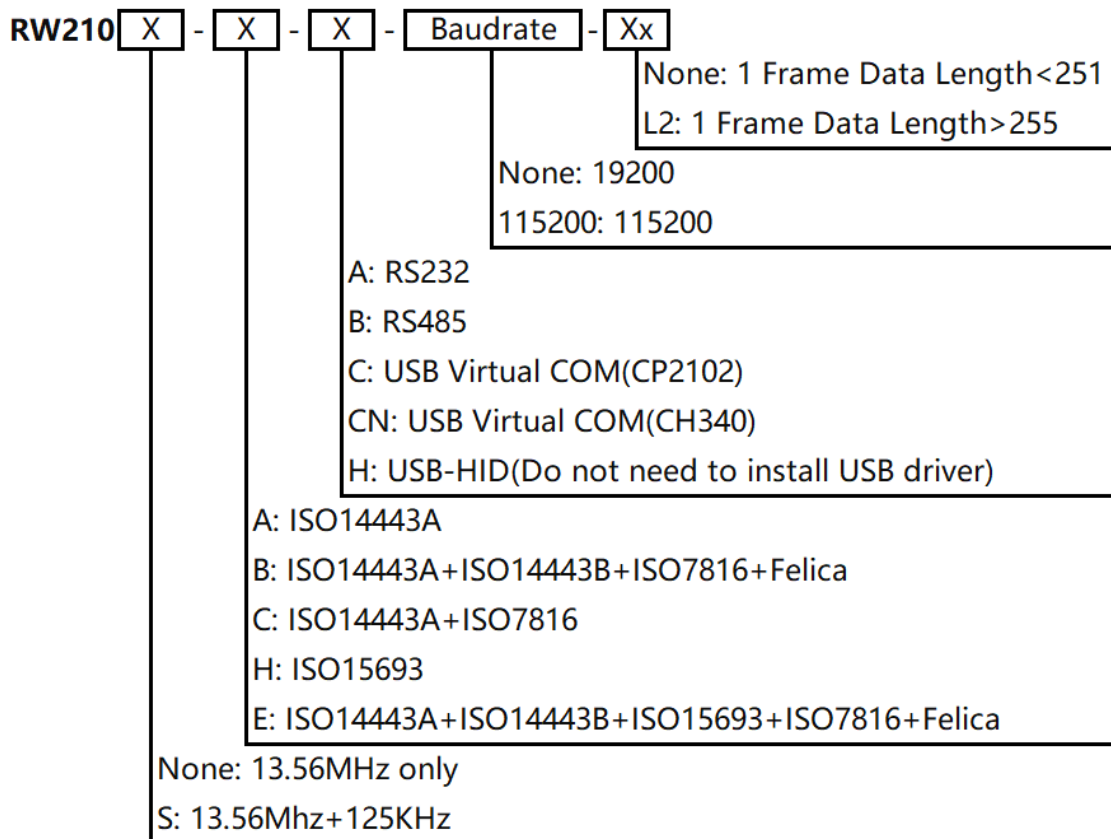
## 1 概述

### 1.1 RW210Ax/Cx IC 卡读写器简介:

RW210Ax/Cx 读写器采用 13.56MHz 非接触射频技术, 内嵌飞利浦射频基站。用户不必关心射频基站的复杂控制方法, 只需通过简单的接口发送命令或操作函数就可以实现对卡片完全的操作。该系列读写卡器支持 ISO14443-A Mifare One S50, S70, FM11RF08, Ultralight, Ntag203, Ntag213, 215, 216, FM1208, Desfire EV1/EV2 D21, D41, D81,T=CL CPU 卡, ISO7816-4 SAM,SIM 卡及其兼容卡片。

如果需要支持 Ultralight C 卡的密钥功能, 需要联系圆志升级为新的固件版本, 默认不支持 Ultraligh C 密钥功能部分。

### 1.2 产品型号及之间的区别:



型号	主要区别	备注
RW210AA	RS232 接口, 无 SAM 卡座	
RW210AB	RS485 接口, 无 SAM 卡座	
RW210AC	USB 接口 (CP2102), 无 SAM 卡座	需安装驱动
RW210ACN	USB 接口 (CH340), 无 SAM 卡座	需安装驱动
RW210AH	USB (HID) 接口, 无 SAM 卡座	免驱动

RW210AT	TTL232 接口, 无 SAM 卡座	
RW210CA	RS232 接口, 带 1 个 SAM 卡座	
RW210CB	RS485 接口, 带 1 个 SAM 卡座	
RW210CC	USB 接口 (CP2102), 带 1 个 SAM 卡座	需安装驱动
RW210CCN	USB 接口 (CH340), 带 1 个 SAM 卡座	需安装驱动
RW210CH	USB (HID) 接口, 带 1 个 SAM 卡座	免驱动
RW210CT	TTL232 接口, 带 1 个 SAM 卡座	
<p>说明: 以上型号均为标准默认型号, 特殊需增加尾缀。</p> <p>(1) 如果出厂默认 115200 波特率时, 需增加尾缀 115200。</p> <p>(2) RW210Cx 系列默认带 1 个 SAM 卡座, 需支持 2 个时请特殊说明, 型号尾缀增加 “-2SAM”。</p> <p>(3) 模块默认长度字为 1 字节, 即每帧数据不能超过 251 字节, 如果每帧数据长度大于 255 字节, 则需要在型号之后再增加尾缀 L2, 此应用主要针对部分特殊的 CPU 卡指令。</p> <p>(4) 如果需要支持 Ultralight C 卡的密钥功能, 需要联系圆志升级为新的固件版本, 默认不支持 Ultraligh C 密钥功能部分。</p>		

### 1.3 主要特点

- 全新一代读卡器, 内嵌 NXP 32 位 ARM 处理器, 处理速度更快
- 采用 NXP 高度集成读写卡芯片射频基站, 性能更优, 更稳定
- 壳体外观专业美学设计, 更美观
- 工作频率: 13.56MHZ
- 支持 ISO14443 TYPE A / MIFARE 标准/ISO7816 国家标准
- 可支持 IC 卡:
  - ISO14443A: Mifare 1K/4K, FM11RF08, Ultralight, Ntag203, Nta213, Ntag215, Ntag216, 符合 T=CL 协议的 CPU 卡 (ISO14443A) FM1208, Desfire EV1/EV2 D21,D41,D81 及其兼容卡片
  - 如果需要支持 Ultralight C 卡的密钥功能, 需要联系圆志升级为新的固件版本, 默认不支持 Ultraligh C 密钥功能部分。
  - ISO17816-4: SAM/SIM9600(T=0 & T=1), SAM/SIM38400(T=0 & T=1), SAM/SIM115200(T=0 & T=1)
  - 及其兼容卡
- 可支持 SAM 卡, SIM 卡操作, 可选装 1-2 个 PSAM 卡座或者 1-2 个 SOP8 ESAM 芯片或者 1-2 个 DIP8 ESAM 芯片
- 内嵌两个可控指示灯



- 内嵌蜂鸣器
- 自带看门狗, 防止死机
- 上电默认自动寻卡, 有卡靠近, 绿色指示灯点亮, 但是不会主动输出卡号, 如有此需求, 请联系圆志索取新的固件版本
- 对外接口: RS232/RS485/USB/TTL232 接口可选
- 用户不必关心射频基站的复杂控制方法, 只需通过简单的命令集可以实现对卡的完全操作
- 为了便于开发者的应用, 我们提供了 PCDMO 测试软件, 以及配套的各种开发环境例程: 比如适合 Windows 的 java, javascript, VC, VB, C++ Builder, DELPHI, Power Builder, C#, CVI 等, 同时还有适合 Android, Linux 等例程
- 基于读卡器的扩展功能很强可**根据用户要求修改软件定制**个性化功能, 不用改变硬件
- 丰富、完善的接口函数, 可方便地应用于充电桩, 交通一卡通, 医疗, 门禁考勤、汽车电子, 感应锁, 密集柜等智能卡管理应用系统

#### 1.4 技术指标

- 工作频率: 13.56MHz
- 支持的标准: ISO14443A/Mifare/ISO7816
- 支持 IC 卡:
  - ISO14443A: Mifare 1K/4K, FM11RF08, Ultralight, Ntag203, Nta213, Ntag215, Ntag216, 符合 T=CL 协议的 CPU 卡 FM1208, Desfire EV1 D21, D41, D81, Desfire EV2 及其兼容卡片
  - ISO7816: SAM/SIM9600(T=0 & T=1), SAM/SIM38400(T=0 & T=1), SAM/SIM115200(T=0 & T=1)

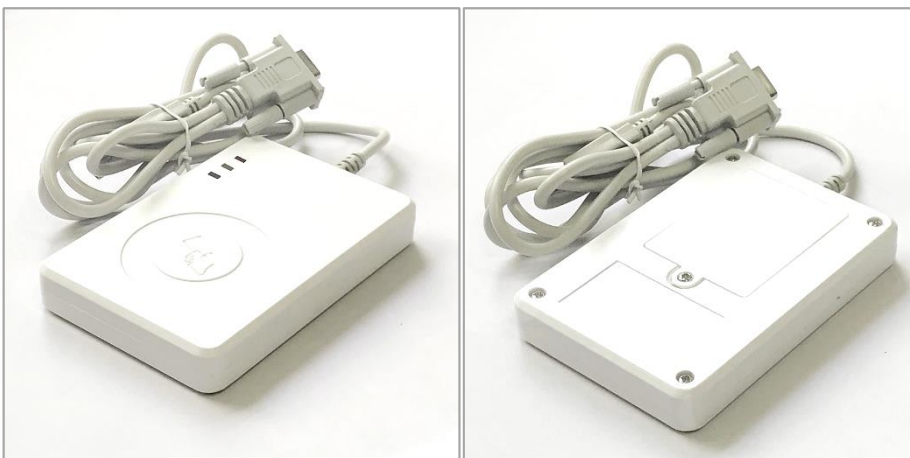
#### 及其兼容卡

- 供电电压: DC5V  $\pm$  10%
- 接口: RS232/RS485/USB/TTL232 接口可选
- 通讯速率: 默认 19.2Kbp 或者 115200
- 最大功耗: 150mA
- 读卡距离: 40-60mm (依据标准 M1 卡测试所得)
- 工作温度: -20°C ~ +70°C
- 外形尺寸: 125mm\*85mm\*19mm
- 重量: 130~160g (内部 USB 线约 45g, RS232 线约 75g)

## 2 硬件连接

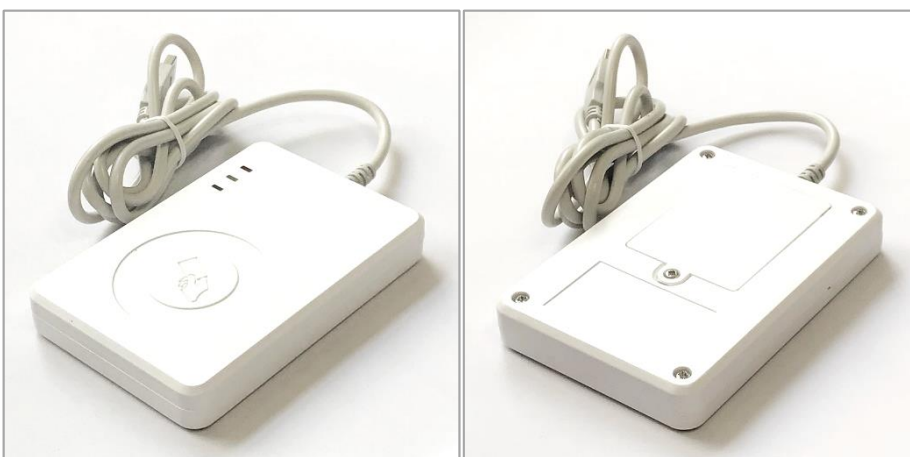
### 2.1 读卡器外观

#### 2.1.1 RS232 串口读卡器

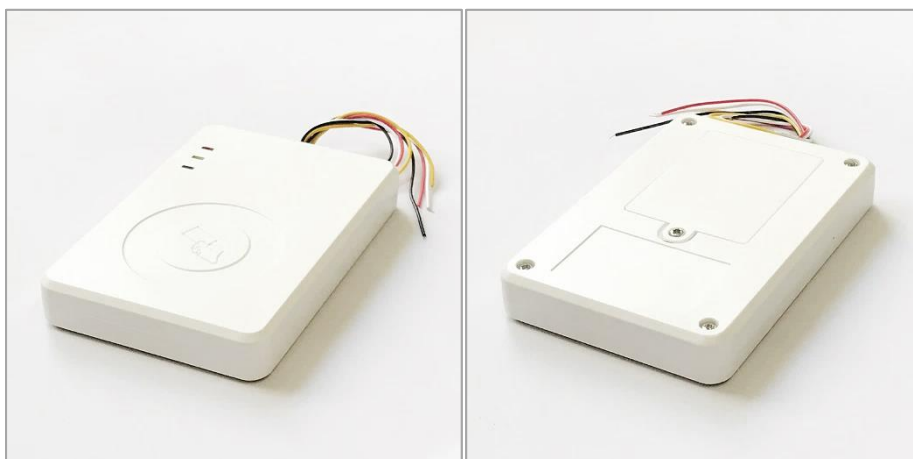


注：USB 插头仅用于取电，不用于通讯。

#### 2.1.2 USB 读卡器



### 2.1.3 RS485 读卡器/TTL232 读卡器



### 2.1.4 外壳尺寸图



## 2.2 读卡器硬件连接

### 2.2.1 RS232 串口连接

适用读卡器型号：RW210AA, RW210CA



如图方式，USB 为取电用途，不具有通讯功能，插入 USB 口取电 9 针串口头为通讯功能，插入主机 9 针串口即可

### 2.2.2 RS485 连接

适用读卡器型号：RW210AB, RW210CB



如图读卡器出线 4 根：

线色	含义
红色	电源正 5V
黑色	电源地
黄色	RS485-A
白色	RS485-B

### 2.2.3 TTL232 连接

适用读卡器型号：RW210AT, RW210CT

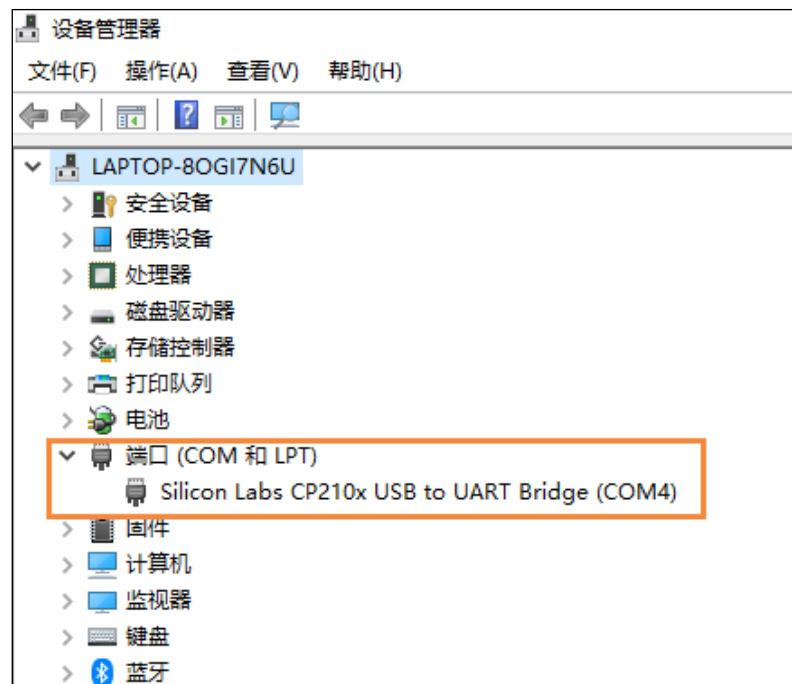


如图读卡器出线 4 根：

线色	含义
红色	电源正 5V
黑色	电源地
黄色	TTL232: TXD
白色	TTL232: RXD

### 2.2.4 USB 非免驱读卡器连接

(1)RW210AC/RW210CC 需要安装驱动程序 (CP210x)，以模拟出一个串口，驱动可从附带资料中找到，驱动安装成功之后，在设备管理器中会显示出虚拟的串口，并且没有黄色叹号表示驱动安装成功，如下图：



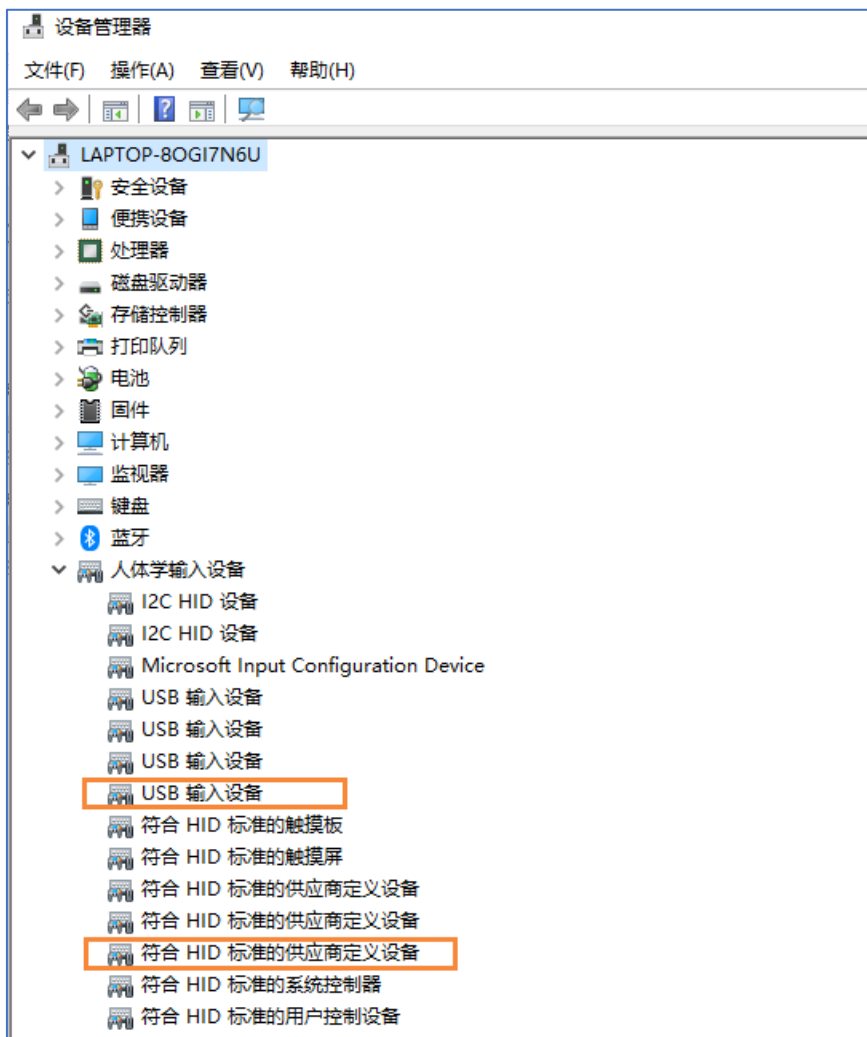
(2)RW210ACN/RW210CCN 需要安装驱动程序 (CH340)，以模拟出一个串口，驱动可从附带

资料中找到，驱动安装成功之后，在设备管理器中会显示出虚拟的串口，并且没有黄色叹号表示驱动安装成功，如下图



### 2.2.5 USB 免驱读卡器连接

RW210AH/RW210CH 为 USB HID 接口，系统会自动安装驱动程序，无需用户单独安装，系统自动安装成功之后，会在设备管理器中增加 USB 描述符，如下图（不同的 windows 版本可能显示不同）：



**备注：硬件连接成功之后，可通过我公司附带的 PC\_DEMO 测试软件进行常规功能测试，以协助您进行二次开发，具体演示软件说明，请参阅附带的 PC\_DEMO 说明。**

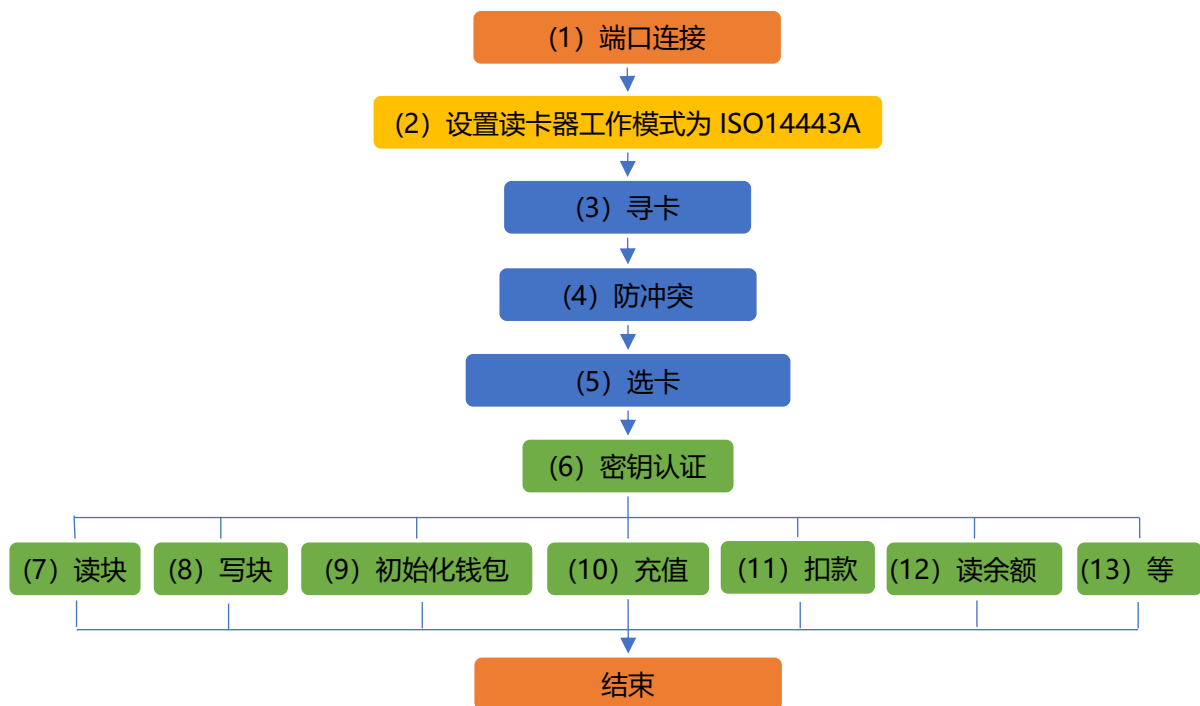
### 2.3 上电状态

读卡器上电后黄绿发光二极管闪烁几次后熄灭，红发光二极管(电源指示灯)长亮。读卡器默认上电有自动检测卡功能，当卡片靠近读卡器时，绿色指示灯会亮起，该功能在主机发送命令之后自动取消。

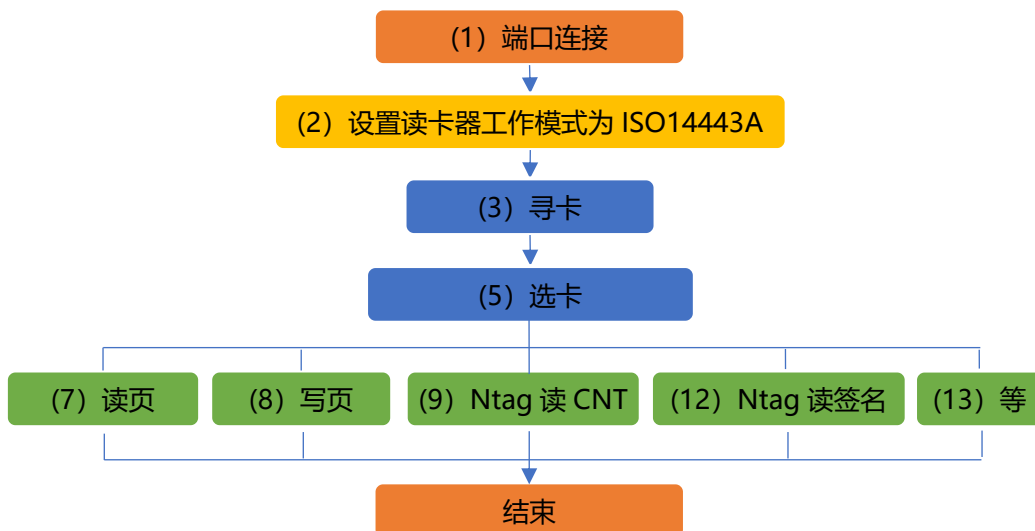
**注：读卡器也可以实现上电默认自动寻卡，并主动输出卡号功能，如有需求请联系圆志。**

## 3 通过读卡器操作卡片步骤简述

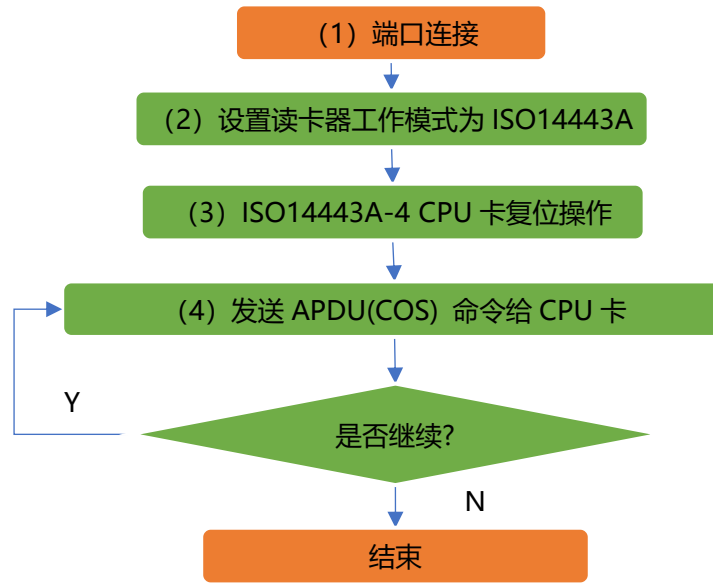
### 3.1 MifareOne 卡操作流程



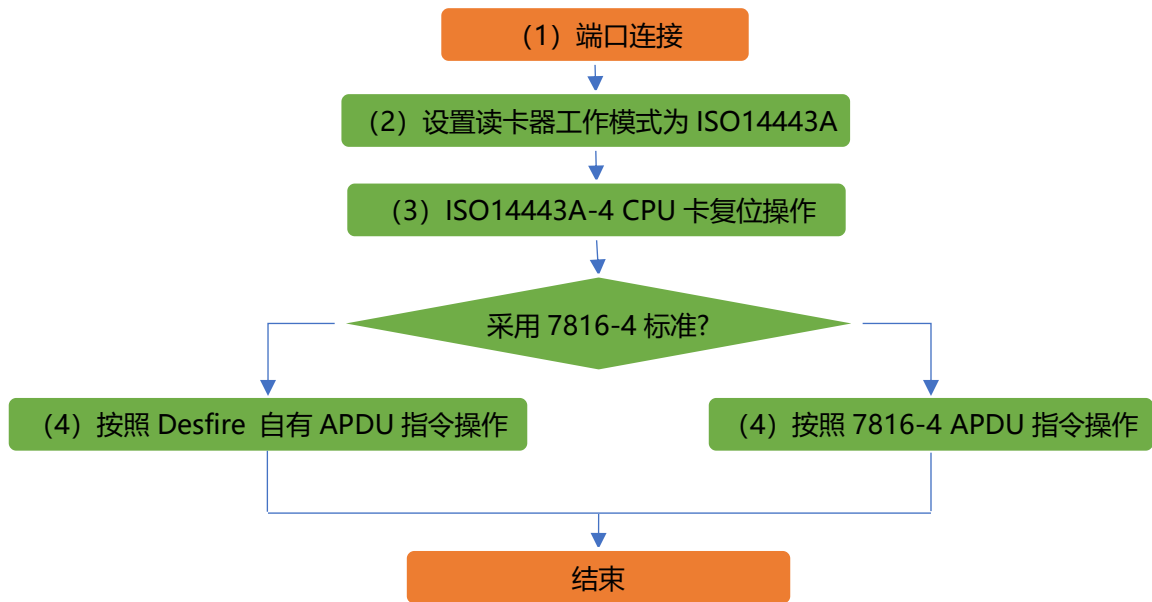
### 3.2 Mifare Ultralight / Ntag 卡操作流程



### 3.3 ISO14443 TYPE A T=CL CPU 卡操作流程

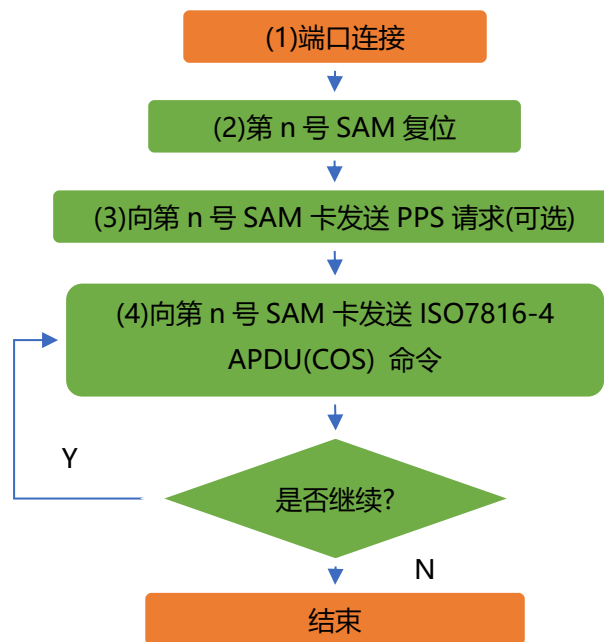


### 3.4 Mifare Desfire EV0/EV1/EV2 卡操作流程



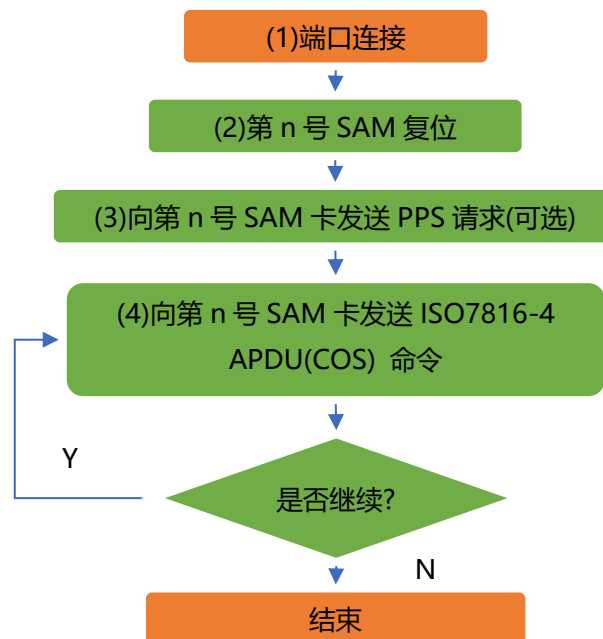


### 3.5 ISO7816-4 卡操作流程 (可支持多个卡)



新的用户建议采用此操作流程，可以扩展对多个 PSAM 卡片的支持。

### 3.6 ISO7816-4 卡操作流程 (只支持 1 个卡)



此操作流程是为了兼容老的型号读卡器，只能支持一个 PSAM 卡，对应读卡器内部的 1 号卡座位置。

## 4 底层数据通讯协议

### 4.1 RS232/TTL232/RS485 协议

- ✚ RS232/TTL232/RS485 接口一帧的数据格式为 1 个起始位，8 个数据位，无奇偶校验位，1 个停止位。

✚ 波特率：19200

✚ 发送接收数据封包格式：



注：(1) 帧头 0x02，帧尾 0x03 被用于起始字符、结束字符。

(2) 为了区分《发送/接收数据包内容》中存在的 0x02,0x03 数据，特增加 0x10 作为 0x02、0x03、0x10 的辨识字符。

(3) 0x10 辨识符用法：在《发送/接收数据包内容》中如果碰到数据为 0x02、0x03、0x10 则需在该数据前插入 0x10 做为数据辨识之用。

(4) 对于增加的 0x10，不影响长度字与校验字的值

发送数据包内容：

发送数据包内容				
模块地址 (2字节)	长度字 (1字节)	命令字 (1字节)	数据域 (N字节)	校验字 (1字节)

模块地址：2 字节的地址

=0x0000：读卡器地址自身地址不管是不是 0x0000，都会执行命令并返回执行结果。

=0x0001~0xFFFF：只有读卡器自身地址与发送指令中相同的才会执行命令并返回执行结果。

长度字：指明从长度字到**校验字**的字节数

命令字：本条命令的含义

数据域：该条命令的内容,此项可以为空

校验字：从模块地址到数据域最后一字节的逐字节累加值（保留最后一字节）

接收数据包内容：

接收数据包内容					
模块地址 (2字节)	长度字 (1字节)	接收到的命令字 (1字节)	执行结果 (1字节)	数据域 (N字节)	校验字 (1字节)

模块地址：读卡器内部自身地址，出厂不一定是 0x0000

长度字：指明从长度字到**数据域最后一字节**的字节数，与发送不同，请特别注意

命令字：返回的命令字应该与发送的命令字相同

如果返回 0x00，则表示发送数据的校验不正确

执行结果: 0x00 执行正确

0x01---0xFF 执行错误

数据域: 该条命令的内容,返回执行状态和命令内容

校验字: 从模块地址到数据域最后一字节的逐字节累加值 (保留最后一字节)

#### 举例说明:

主机到从机指令: 02 00 00 04 15 10 03 1C 03

16 进制数据	缩写	缩写含义
02	Fr-S	帧头
00 00	Rd-addr	模块地址
04	Len	长度字
15	CMD	命令字
10		插入的标识字符
03	DF	数据域
1C	ChkS	校验字
03	Fr-E	帧尾

校验字  $0x1C = 0x00 + 0x00 + 0x04 + 0x15 + 0x03$

长度字  $0x04 = 1(0x04) + 1(0x15) + 1(0x03) + 1(0x1C)$

从机到主机返回: 02 00 00 10 03 15 00 18 03

16 进制数据	缩写	缩写含义
02	Fr-S	帧头
00 00	Rd-addr	模块地址
10		插入的标识字符
03	Len	长度字
15	CMD	命令字
00		执行结果
18	ChkS	校验字
03	Fr-E	帧尾

校验字  $0x18 = 0x00 + 0x00 + 0x03 + 0x15 + 0x00$

长度字  $0x03 = 1(0x03) + 1(0x15) + 1(0x00)$

## 4.2 系统函数指令

### 通用系统函数: 设置波特率

**功能描述:** 用于修改读卡器内部的通讯波特率, 需注意的是, 读卡器重新上电后, 会恢复成默认的出厂波特率, 此次修改会失效, 故建议直接采用出厂默认波特率, 如果需要修改出厂默认波特率, 请联系圆志进行固件升级。

#### 发送数据序列:

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x15	发送数据	0x1C	0x03

发送数据：1 字节设置模式

- 发送数据=0x01 表示设置模块波特率为 9600;
- 发送数据=0x02 表示设置模块波特率为 14400;
- 发送数据=0x03 表示设置模块波特率为 19200;
- 发送数据=0x04 表示设置模块波特率为 28800;
- 发送数据=0x05 表示设置模块波特率为 38400;
- 发送数据=0x06 表示设置模块波特率为 57600;
- 发送数据=0x07 表示设置模块波特率为 115200;

**特别注意：**读卡器默认的波特率是 19200，读卡器不会保存此设置，当重新上电后，模块又恢复到默认的 19200，故当更改波特率后，因中间掉电有可能造成通讯故障；

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x15	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x15	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 15 10 03 1C 03

【接收数据:】 02 00 00 10 03 15 00 18 03

**通用系统函数：设置模块工作在 ISO14443 TYPE A 模式**

**功能描述：**用于设置模块工作于 ISO14443 TYPE A 工作模式，对于单协议读卡器可省略此条指令。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x3A	发送数据	0x7F	0x03

发送数据：1 字节设置模式：

发送数据=0x41 ( 'A' ) 设置为 ISO14443A 模式 (MF1,Ultralight,Ntag,TypeA CPU 等)；

发送数据=0x42 ('B') 设置为 ISO14443B 模式 (Type B CPU 卡, 二代证等);

发送数据=0x43 ('C') 设置为 Felica 方式 (Felica);

发送数据=0x73 ('s') 设置为 ST 卡方式 (Type B SR176,SRI512,SRI4K 等);

发送数据=0x72 ('r') 设置为 Atmel 卡方式 (Type B At88rf020 等);

发送数据=0x31 ('1') 设置为 ISO15693 方式 (I Code SL2,Tag it 等);

#### 正确返回数据序列:

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x3A	0x00	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的, 故在操作接收数据时需过滤掉;

#### 错误返回数据序列:

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x3A	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的, 故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】 02 00 00 04 3A 41 7F 03

【接收数据:】 02 00 00 10 03 3A 00 3D 03

#### 通用系统函数: 设置模块天线状态

**功能描述:** 用于设置模块的天线工作状态。

**发送数据序列:**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x05	发送数据	0xXX	0x03

发送数据: 1 字节控制模式, 按位定义:

发送数据 bit0=0 关闭天线;

=1 开启天线;

发送数据 bit1=0 关闭自动寻卡;

=1 开启自动寻卡;

发送数据 bit2=0 无意义;

=1 读卡芯片硬件复位;

Bit3-bit7: 保留, 固定为 00000b

**说明: bit1 和 bit2 功能需要读卡器支持才可以**

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x05	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x05	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 04 05 00 09 03

【接收数据：】 02 00 00 10 03 05 00 08 03

**通用系统函数：控制 LED 指示灯状态**

**功能描述：**用于控制读卡器 LED 指示灯的状态。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x6A	发送数据	0xXX	0x03

发送数据：1 字节指示灯状态：

数据=0x00 黄灯熄灭，LED1 引脚输出高电平；

数据=0x01 绿灯熄灭，LED2 引脚输出高电平；

数据=0x02 绿灯点亮，LED2 引脚输出低电平；

数据=0x03 黄灯点亮，LED1 引脚输出低电平；

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x6A	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x6A	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 6A 00 6E 03

【接收数据:】 02 00 00 10 03 6A 00 6D 03

#### 通用系统函数：控制蜂鸣器蜂鸣时间

**功能描述：**用于控制蜂鸣器蜂鸣时间。

#### 发送数据序列：

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x1D	发送数据	0xXX	0x03

发送数据：数据部分为 1 字节蜂鸣时间，取值 0x00~0xFF，单位为 10ms

#### 正确返回数据序列：

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x1D	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x1D	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例:

【发送数据:】 02 00 00 04 1D 10 03 71 03

【接收数据:】 02 00 00 10 03 1D 00 21 03

### 通用系统函数: 读取软硬件版本号

**功能描述:** 用于读取读卡器的软硬件版本号。

**发送数据序列:**

发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x16	空	0xXX	0x03

白色部分为正常的指令序列, 因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03, 根据通讯协议需在其前面增加 0x10, 即黄色的部分;

**正确返回数据序列:**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x16	0x00	返回数据	0xXX	0x03

返回数据: 2 字节软硬件版本号

**错误返回数据序列:**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x16	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】 02 00 00 10 03 16 19 03

【接收数据:】 02 00 00 05 16 00 01 01 1D 03

### 通用系统函数: 读取读卡器序列号

**功能描述:** 用于读取读卡器的序列号。

**发送数据序列:**

发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x17	空	0xXX	0x03



白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

#### 正确返回数据序列：

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x17	0x00	返回数据	0xXX	0x03

返回数据：N 字节读卡器序列号

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x17	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 10 03 17 1A 03

【接收数据:】 02 00 00 0B 17 00 16 10 03 24 14 55 40 01 01 0A 03

#### 通用系统函数：设置读卡器的设备地址

**功能描述：**用于设置读卡器的设备地址，**该功能需要读卡器支持才可以。**

#### 发送数据序列：

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x13	发送数据	0xXX	0x03

发送数据：数据部分为 2 字节设备地址

#### 正确返回数据序列：

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x13	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x13	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 05 13 12 34 5E 03

【接收数据：】 02 12 34 10 03 13 00 5C 03

### 通用系统函数：读取读卡器设备地址

**功能描述：**用于读取读卡器的设备地址。

**发送数据序列：**

发送数据序列

帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x14	空	0xXX	0x03

白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

**正确返回数据序列：**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x14	0x00	返回数据	0xXX	0x03

返回数据：N 字节读卡器序列号

**错误返回数据序列：**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x14	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 10 03 14 17 03

【接收数据：】 02 00 00 05 14 00 00 00 19 03

### 4.3 MifareOne 卡函数指令

#### MF1 专用指令：MF1 卡寻卡

**功能描述：**用于 Mifare one/Ultralight/Ntag 等满足 ISO14443A 卡的寻卡，返回卡片类型。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x46	发送数据	0xXX	0x03

发送数据部分为 1 字节寻卡模式；

发送数据=“0x26” 寻未进入睡眠状态的卡；

发送数据=“0x52” 寻天线范围内的所有状态的卡；

**正确返回数据序列：**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x05	0x46	0x00	返回数据	0xXX	0x03

返回数据：2 字节卡类型 Tagtype：

Tagtype[0]的 Bit7-bit6：

=b00 此卡为 4 字节卡号

=b01 此卡为 7 字节卡号

=b10 此卡为 10 字节卡号

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x46	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 46 52 9C 03

【接收数据:】 02 00 00 05 46 00 04 00 4F 03

#### MF1 专用指令：Mifare one 卡防冲突

**功能描述：**用于 Mifare one 卡的防冲突指令，返回卡片唯一序列号，注该指令发送之前必须先发送**寻卡指令**，并且如果需要对卡进行读写等操作时，在该条指令之后还要发送**选卡指令**。

**发送数据序列：**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x47	0x04	0x4F	0x03

发送数据= "0x04"

**正确返回数据序列:**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x47	0x00	返回数据	0xXX	0x03

返回数据: 4/7/10 字节卡号, 与卡片有关

**错误返回数据序列:**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x47	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】 02 00 00 04 47 04 4F 03

【接收数据:】 02 00 00 07 47 00 42 0B C2 08 65 03 (返回卡号: 42 0B C2 08)

**MF1 专用指令: Mifare one 卡选卡**

**功能描述:** 用于 Mifare one 卡的选卡操作, 此指令的目的是选定一张卡, 然后就可以实现对这张选定卡的读写等操作了。有一些读卡器此条指令可以省略。

**发送数据序列:**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x07	0x48	发送数据	0xXX	0x03

发送数据: 4 字节卡序列号字节数

**正确返回数据序列:**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x48	0x00	返回数据	0xXX	0x03

返回数据：1 字节选择确认信息

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x48	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 07 48 42 0B C2 08 66 03

【接收数据:】 02 00 00 04 48 00 08 54 03

#### MF1 专用指令：Mifare one 卡密钥验证

**功能描述：**用于 Mifare one 卡的读写卡钱包操作之前的**密钥验证**，**只有密钥正确之后**才可以对卡进行读写或钱包操作。

#### 发送数据序列：

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x0B	0x4A	发送数据	0xXX	0x03

发送数据：1 字节密钥模式+1 字节绝对块+6 字节密钥

1 字节密钥模式=0x60 表示采用 A 密钥认证；

1 字节密钥模式=0x61 表示采用 B 密钥认证；

1 字节绝对块号=0x00~0x3F 之间的任意块号（Mifare One S50，共计 64 块）；

1 字节绝对块号=0x00~0xFF 之间的任意块号（Mifare One S70，共计 256 块）；

6 字节密钥为要操作块所在扇区的密钥；

注：新出厂的卡片默认密钥模式为 A 密钥，6 字节密钥为“FFFFFFFFFFFF”；

#### 正确返回数据序列：

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4A	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4A	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 0B 4A 60 00 FF FF FF FF FF AF 03 (验证 A 密钥, 0 块, 密码: "FFFFFFFFFFFF" )

【接收数据:】 02 00 00 10 03 4A 00 4D 03

### MF1 专用指令: Mifare one 卡读块

**功能描述:** 用于 Mifare one 卡的读块内容操作，注在读块前需进行**密钥验证**。

**发送数据序列:**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x4B	发送数据	0xXX	0x03

发送数据: 1 字节块号

1 字节绝对块号=0x00~0x3F 之间的任意块号 (Mifare One S50, 共计 64 块);

1 字节绝对块号=0x00~0xFF 之间的任意块号 (Mifare One S70, 共计 256 块);

**正确返回数据序列:**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x13	0x4B	0x00	返回数据	0xXX	0x03

返回数据: 读取块 16 字节内容

**错误返回数据序列:**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4B	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 4B 00 4F 03 (读 0 块数据。)

【接收数据:】 02 00 00 13 4B 00 42 0B C2 08 83 08 04 00 62 63 64 65 66 67 68 69 30 03

(0 块数据: 42 0B C2 08 83 08 04 00 62 63 64 65 66 67 68 69)

### MF1 专用指令: Mifare one 卡写块

**功能描述:** 用于 Mifare one 卡的写块内容操作, 注在写块前需进行**密钥验证**。

**发送数据序列:**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x14	0x4C	发送数据	0xXX	0x03

发送数据: 1 字节绝对块号+16 字节写入内容

1 字节绝对块号=0x00~0x3F 之间的任意块号 (Mifare One S50, 共计 64 块);

1 字节绝对块号=0x00~0xFF 之间的任意块号 (Mifare One S70, 共计 256 块);

16 字节要写入的内容;

**对于密钥块 (每个扇区的最后一块) 的写操作一定要谨慎, 否则有可能造成该扇区的失效, 具体使用注意事项请参阅卡片说明书。**

**正确返回数据序列:**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4C	0x00	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的, 故在操作接收数据时需过滤掉;

**错误返回数据序列:**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4C	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】 02 00 00 14 4C 01 11 11 11 11 11 11 11 11 11 11 11 11 11 11 71 03 (将第一块写成 "11111111111111111111111111111111")

【接收数据:】 02 00 00 10 03 4C 00 4F 03

**MF1 专用指令：Mifare one 卡初始化钱包**

**功能描述：**用于 Mifare one 卡的钱包初始化操作，注在钱包初始化前需进行**密钥验证**。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x08	0x4D	发送数据	0xXX	0x03

发送数据：1 字节绝对块号+4 字节初始化金额

1 字节绝对块号=0x00~0x3F 之间的任意块号（Mifare One S50，共计 64 块）；

1 字节绝对块号=0x00~0xFF 之间的任意块号（Mifare One S70，共计 256 块）；

4 字节初始化金额；4 字节 16 进制初始化金额，低字节在前；

**密钥块（每个扇区的最后一块）不能初始化为钱包，否则会造成该扇区的失效，具体使用注意事项请参阅卡片说明书。**

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4D	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4D	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 08 4D 01 64 00 00 00 BA 03 （01 块初始化为钱包，金额 16 进制 “64”）

【接收数据:】 02 00 00 10 03 4D 00 50 03

**MF1 专用指令：Mifare one 卡读钱包**

**功能描述：**用于 Mifare one 卡的钱包读取操作，注在钱包读取前需进行**密钥验证**。

**发送数据序列：**



发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x4E	发送数据	0xXX	0x03

发送数据：1 字节绝对块号

1 字节绝对块号=0x00~0x3F 之间的任意块号 (Mifare One S50, 共计 64 块);

1 字节绝对块号=0x00~0xFF 之间的任意块号 (Mifare One S70, 共计 256 块);

#### 正确返回数据序列:

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x07	0x4E	0x00	返回数据	0xXX	0x03

返回数据：4 字节 16 进制钱包值，低字节在前；

#### 错误返回数据序列:

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4E	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 4E 01 53 03

【接收数据:】 02 00 00 07 4E 00 96 00 00 00 EB 03 （读出金额 16 进制 “96”）

#### MF1 专用指令：Mifare one 卡充值

**功能描述：**用于 Mifare one 卡的钱包充值操作，注在钱包充值前需进行**密钥验证**。

**发送数据序列：**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x08	0x50	发送数据	0xXX	0x03

发送数据：1 字节绝对块号 + 4 字节 16 进制要增加的金额值（低字节在前）

1 字节绝对块号=0x00~0x3F 之间的任意块号 (Mifare One S50, 共计 64 块);

1 字节绝对块号=0x00~0xFF 之间的任意块号 (Mifare One S70, 共计 256 块);

**密钥块（每个扇区的最后一块）不能进行充值操作，否则会造成该扇区的失效，具体使用注意事项请参阅卡片说明书。**

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x50	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x50	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 08 50 01 64 00 00 00 BD 03 (01 块充值 16 进制 “64” )

【接收数据:】 02 00 00 10 03 50 00 53 03

**MF1 专用指令：Mifare one 卡扣款**

**功能描述：**用于 Mifare one 卡的钱包扣款操作，注在钱包扣款前需进行**密钥验证**。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x08	0x4F	发送数据	0xXX	0x03

发送数据：1 字节绝对块号 + 4 字节 16 进制要减少的金额值（低字节在前）

1 字节绝对块号=0x00~0x3F 之间的任意块号（Mifare One S50，共计 64 块）；

1 字节绝对块号=0x00~0xFF 之间的任意块号（Mifare One S70，共计 256 块）；

**密钥块（每个扇区的最后一块）不能进行充值操作，否则会造成该扇区的失效，具体使用注意事项请参阅卡片说明书。**

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4F	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动

增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4F	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 08 4F 01 32 00 00 00 8A 03 (01 块扣款 16 进制 “32” )

【接收数据:】 02 00 00 10 03 4F 00 52 03

#### MF1 专用指令：Mifare one 卡钱包备份步骤 1

**功能描述：**用于将 Mifare one 卡的指定块的钱包内容回传至卡的 Buffer，然后通过钱包备份步骤 2 将钱包值备份到同一扇区的另一指定块，注在此操作前需进行**密钥验证**，**同时这两步操作需要在同一扇区内操作。**

#### 发送数据序列：

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x51	发送数据	0xXX	0x03

发送数据：1 字节绝对块号

1 字节绝对块号=0x00~0x3F 之间的任意块号 (Mifare One S50, 共计 64 块)；

1 字节绝对块号=0x00~0xFF 之间的任意块号 (Mifare One S70, 共计 256 块)；

**注：要备份的块必须为钱包格式。**

#### 正确返回数据序列：

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x51	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x51	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 51 01 56 03 （将 01 块内容传入卡的 BUFFER 区）

【接收数据:】 02 00 00 10 03 51 00 54 03

### MF1 专用指令：Mifare one 卡钱包备份步骤 2

**功能描述:**用于将 Mifare one 卡通过钱包备份步骤 1 传入 Buffer 中的钱包值备份到**同一扇区**的指定块，注在此操作前需进行 **Mifare one 卡钱包备份步骤 1 的操作**。

**发送数据序列:**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x52	发送数据	0xXX	0x03

发送数据：1 字节绝对块号

1 字节绝对块号=0x00~0x3F 之间的任意块号（Mifare One S50，共计 64 块）；

1 字节绝对块号=0x00~0xFF 之间的任意块号（Mifare One S70，共计 256 块）；

**正确返回数据序列:**

正确返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x52	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列:**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x52	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例:

【发送数据:】 02 00 00 04 52 10 02 58 03 (将 BUFFER 区中的内容传入 02 块中进行备份)

【接收数据:】 02 00 00 10 03 52 00 55 03

### MF1 专用指令: Mifare one 卡读连续 3 块

**功能描述:** 用于 Mifare one 卡的读连续 3 块内容操作, 注在读块前需进行**密钥验证**。

**发送数据序列:**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x22	发送数据	0xXX	0x03

发送数据: 1 字节块号

1 字节绝对块号=0x00~0x3F 之间的任意块号 (Mifare One S50, 共计 64 块);

1 字节绝对块号=0x00~0xFF 之间的任意块号 (Mifare One S70, 共计 256 块);

**正确返回数据序列:**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x33	0x22	0x00	返回数据	0xXX	0x03

返回数据: 读取连续 3 块 48 字节内容

**错误返回数据序列:**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x22	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】 02 00 00 04 22 00 26 03

【接收数据:】 02 FF FF 33 22 00 93 42 7A 0A A1 28 04 00 90 10 10 15 00 00 00 00 00 11 11  
11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 00 00 00 00 00 00 00 00 00 00 00  
00 00 00 00 3E 03

### MF1 专用指令: Mifare one 卡写连续 3 块

**功能描述:** 用于 Mifare one 卡的写连续 3 块内容操作, 注在写块前需进行**密钥验证**。

**发送数据序列:**



发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x29	空	0x2C	0x03

白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

#### 正确返回数据序列：

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x29	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x29	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 10 03 29 2C 03

【接收数据：】 02 00 00 10 03 29 00 2C 03

## 4.4 Ultralight/Ntag 卡函数指令

### UL/Ntag 专用指令：Ultralight/Ntag 卡寻卡

**功能描述：**用于 Mifare one/Ultralight/Ntag 卡的寻卡，返回卡片类型。

#### 发送数据序列：

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x46	发送数据	0xXX	0x03

发送数据部分为 1 字节寻卡模式；

发送数据=“0x26” 寻未进入睡眠状态的卡；

发送数据=“0x52” 寻天线范围内的所有状态的卡；

#### 正确返回数据序列：

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x05	0x46	0x00	返回数据	0xXX	0x03

返回数据：2 字节卡类型：

返回数据=0x04 0x00 表示 Mifare one S50 卡；

返回数据=0x02 0x00 表示 Mifare one S70 卡；

返回数据=0x44 0x00 表示 Utralight 卡；

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x46	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 46 52 9C 03

【接收数据:】 02 00 00 05 46 00 04 00 4F 03 （通过返回数据可判断为 S50 卡）

#### UL/Ntag 专用指令：Ultralight/Ntag 卡选卡

**功能描述：**用于选择 UltraLight/Ntag 卡片，此条指令需要在 0x46 寻卡指令之后操作。

**发送数据序列：**

发送数据序列

帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x33	空	0x36	0x03

白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

#### 正确返回数据序列：

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x0A	0x33	0x00	返回数据	0xXX	0x03

返回数据：7 字节 UID 卡号。

#### 错误返回数据序列：



错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x33	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 10 03 33 36 03

【接收数据:】 02 00 00 0A 33 00 04 6E F0 BA E1 22 80 DC 03

### UL/Ntag 专用指令：Ultralight/Ntag 卡读卡

**功能描述：**用于 Ultralight/Ntag 卡读写操作，此指令与 mifare one 读块相同，此指令读出的为连续 4 页的内容。

**发送数据序列：**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x4B	发送数据	0xXX	0x03

发送数据：为 1 字节起始页页号；

**正确返回数据序列：**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x13	0x4B	0x00	返回数据	0xXX	0x03

返回数据：从起始页开始的连续 4 页内容，16 字节

**错误返回数据序列：**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x4B	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例（此例读出从页 0 到页 3 的连续 4 页的内容）：

【发送数据:】 02 00 00 04 4B 00 4F 03

【接收数据:】 02 00 00 13 4B 00 04 6E F0 12 BA E1 22 80 F9 48 00 00 00 00 00 00 50 03

**UL/Ntag 专用指令：Ultralight/Ntag 卡写卡**

**功能描述：**用于写 UltraLight/Ntag 卡片 1 页数据。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x08	0x35	发送数据	0xXX	0x03

发送数据部分为 1 字节页号+4 字节写入数据；

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x35	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x35	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 08 35 04 11 11 11 11 85 03

【接收数据:】 02 00 00 10 03 35 00 38 03

**Ntag21x 专用指令：Ntag21x 系列卡快速读取**

**功能描述：**用于快速读取 Ntag 卡片 N 页数据。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x08	0x88	发送数据	0xXX	0x03

发送数据：1 字节 cStartAdr + 1 字节 cEndAdr，一次最多读取 15 块；

cStartAdr：起始页号

cEndAdr：结束页号

每次最多读取 15 页内容

**正确返回数据序列:**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x88	0x00	返回数据	0xXX	0x03

返回数据: (终止页 cEndAdr -起始页 cStartAdr +1) \*4 字节数据

**错误返回数据序列:**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x88	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉; s

发送与返回正确举例:

【发送数据:】 02 00 00 05 88 00 0E 9B 03

【接收数据:】 02 00 00 3F 88 00 04 4A 3C FA 72 9A 3D 80 55 48 00 00 E1 10 10 12 00 10 03  
1B D1 10 02 16 53 70 91 00 00 00 00 22 22 22 22 11 22 33 44 00 00 00 00 11  
11 11 11 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 85 03

**Ntag21x 专用指令: 读取 Ntag21x 系列卡片版本号**

**功能描述:** 用于 Ntag21x 系列卡片版本号读取。

**发送数据序列:**

发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x87	空	0x2C	0x03

白色部分为正常的指令序列, 因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03, 根据通讯协议需在其前面增加 0x10, 即黄色的部分;

**正确返回数据序列:**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x87	0x00	返回数据	0xXX	0x03

返回数据: 8 字节卡片版本号。

**错误返回数据序列:**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x87	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 10 03 87 8A 03

【接收数据：】 02 00 00 0B 87 00 00 04 04 10 02 01 00 0F 10 03 AF 03

### Ntag21x 专用指令：Ntag21x 系列密钥认证

**功能描述：**用于 Ntag21x 系列的卡片密钥认证，**卡片需开启密钥功能。**

**发送数据序列：**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x07	0x8A	发送数据	0xXX	0x03

发送数据：4 字节的密钥；

**正确返回数据序列：**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x8A	0x00	返回数据	0xXX	0x03

返回数据：2 字节 PACK。

**错误返回数据序列：**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x8A	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 07 8A FF FF FF FF 8D 03

【接收数据：】 02 00 00 05 8A 00 12 34 D5 03

### Ntag21x 专用指令：读取 Ntag21x 系列卡片读取 CNT

**功能描述：**用于 Ntag21x 系列卡片读取 CNT。

**发送数据序列:**

发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x89	空	0x2C	0x03

白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

**正确返回数据序列:**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x89	0x00	返回数据	0xXX	0x03

返回数据：3 字节 CNT。

**错误返回数据序列:**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x89	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 10 03 89 8C 03

【接收数据:】 02 00 00 06 89 00 14 00 00 A3 03 03

**Ntag21x 专用指令：读取 Ntag21x 系列卡片读取数字签名**

**功能描述：**用于 Ntag21x 系列卡片读取数字签名。

**发送数据序列:**

发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x8B	空	0x2C	0x03

白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

**正确返回数据序列:**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x8B	0x00	返回数据	0xXX	0x03

返回数据：32 字节数字签名。

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x8B	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 10 03 8B 8E 03

【接收数据:】 02 00 00 23 8B 00 D1 38 F2 2C 7C C3 AD FE D0 50 AC D4 5A 10 03 98 C8 22  
AD 21 BC 75 BA 3A 1E C2 7C 60 46 A5 CC 67 19 24 03

#### NFC 卡专用指令：NFC 类型 2 标签选择扇区

功能描述：选择 NFC 类型 2 标签扇区。

#### 发送数据序列：

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x86	发送数据	0xXX	0x03

发送数据：1 字节扇区号，取值 0x00-0xFE；

#### 正确返回数据序列：

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x86	0x00	空	0xXX	0x03

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x86	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例:

【发送数据:】

【接收数据:】

#### UL/Ntag 专用指令: Ultralight/Ntag 卡休眠

**功能描述:** 用于将 Mifare one/Ultralight/Ntag 卡休眠操作。

**发送数据序列:**

发送数据序列							
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x29	空	0x2C	0x03

白色部分为正常的指令序列,因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03,根据通讯协议需在其前面增加 0x10,即黄色的部分;

**正确返回数据序列:**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x29	0x00	空	0xXX	0x03

注:黄色部分为模块在返回数据时,在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03 后自动增加的,故在操作接收数据时需过滤掉;

**错误返回数据序列:**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x29	非零	空	0xXX	0x03

注:黄色部分为模块在返回数据时,在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的,故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】 02 00 00 10 03 29 2C 03

【接收数据:】 02 00 00 10 03 29 00 2C 03

#### 4.5 ISO14443A-4 CPU 卡函数指令

此为 CPU 卡通用函数,适合满足 ISO14443A-4 标准的 CPU 卡,比如:FM1208,Desfire 等

**TYPE A CPU 卡专用功能: ISO14443A-4 CPU 卡复位**

**功能描述:** 用于进行满足 ISO14443A-4 CPU 卡复位操作,正确返回卡片复位信息。

**发送数据序列:**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x53	发送数据	0xXX	0x03

发送数据：1 字节寻卡模式

1 字节寻卡模式=0x26 表示寻天线范围内的未休眠卡；

1 字节寻卡模式=0x52 表示寻天线范围内的所有卡；

#### 正确返回数据序列：

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x53	0x00	返回数据	0xXX	0x03

返回数据：4/7/10 字节 CSN+CPU 卡复位信息；

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x53	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 53 52 A9 03

【接收数据:】 02 00 00 0F 53 00 16 61 1B 82 10 10 78 80 90 10 02 20 90 00 C0 03

#### TYPE A CPU 卡专用功能：ISO14443A-4 CPU 卡发送 APDU(COS)指令

**功能描述：**用于发送 ISO14443A-4 CPU 卡 APDU(COS)指令。

#### 发送数据序列：

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x54	发送数据	0xXX	0x03

发送数据：APDU(COS)命令

具体 APDU(COS)指令请参阅 CPU 卡片厂家数据手册；

#### 正确返回数据序列：



正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x54	0x00	返回数据	0xXX	0x03

返回数据：对应发送 APDU(COS)指令的返回数据；

#### 错误返回数据序列：

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x54	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 08 54 00 84 00 00 04 E4 03      (取 4 字节随机数指令)

【接收数据：】 02 00 00 09 54 00 7B A3 5F 28 90 00 92 03      (4 字节随机数 7B A3 5F 28, 90 00 表示执行正确)

#### TYPE A CPU 卡专用功能：ISO14443A-4 CPU 卡 PPS 升频

**功能描述：**用于进行满足 ISO14443A-4 CPU 卡 PPS 请求操作。PPS 请求必须在复位指令执行后，发送第一条 COS 指令之前进行

**注意：**此功能需要读卡器支持

#### 发送数据序列：

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0xC0	发送数据	0xXX	0x03

发送数据：1 字节 cCID+1 字节 cSetting

1 字节 cCID:固定为 0；

1 字节 cSetting: =0: 106K; =1: 212K; =2: 424K; =3: 848K;

PPS 请求必须在复位指令执行后，发送第一条 COS 指令之前进行。

#### 正确返回数据序列：

正确返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0xC0	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0xC0	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】

【接收数据:】

**TYPE A CPU 卡专用功能：ISO14443A-4 CPU 卡退出激活状态**

**功能描述：**用于进行满足 ISO14443A-4 CPU 卡退出激活状态

**发送数据序列：**

发送数据序列								
帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾	
0x02	0x00,0x00	0x10	0x03	0x6B	空	0xXX	0x03	

白色部分为正常的指令序列，因为在帧头 0x02 帧尾 0x03 之间出现了 0x02 或 0x10 或 0x03，根据通讯协议需在其前面增加 0x10，即黄色的部分；

**正确返回数据序列：**

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x6B	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x6B	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】

【接收数据:】

#### 4.6 ISO7816-4 SAM 卡函数指令（支持多个卡）

新的用户建议采用此操作流程，可以扩展对多个 PSAM 卡片的支持。

##### ISO7816-4 SAM/SIM 卡专用功能：第 N 号 SAM 卡复位

**功能描述：**用于第 N 号 ISO7816-4 CPU/SAM/SIM 卡复位操作，正确返回卡片复位信息。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x19	发送数据	0xXX	0x03

发送数据：1 字节 mode

- bit1-0: =b00 (复位 9600 的卡片)
- =b01 (复位 38400 的卡)
- =b10 (复位 115200 的卡)
- bit3-2: =b00 (复位操作)
- bit7-4: =b0000 (SAM 卡编号 1)
- =b0001 (SAM 卡编号 2)
- =b0010 (SAM 卡编号 3)
- .....

**正确返回数据序列：**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x19	0x00	返回数据	0xXX	0x03

返回数据：CPU/SAM/SIM 卡复位信息；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x19	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 19 00 1D 03

【接收数据:】 02 00 00 16 19 00 3B 7D 94 00 00 4C 31 76 68 10 03 4C 4B 12 10 02 16 4F 95  
2D 00 AB 03

**ISO7816-4 SAM/SIM 卡专用功能：向第 N 号 SAM 卡发送 APDU(COS)指令**

**功能描述：**用于向第 N 号 ISO7816-4 CPU/SAM/SIM 卡发送 APDU(COS)指令。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x1A	发送数据	0xXX	0x03

发送数据：1 字节 SAM 卡编号+N 字节 APDU(COS)命令

具体 APDU(COS)指令请参阅 CPU/SAM/SIM 卡片厂家数据手册；

**正确返回数据序列：**

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x1A	0x00	返回数据	0xXX	0x03

返回数据：对应发送 APDU(COS)指令的返回数据；

**错误返回数据序列：**

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x1A	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 09 1A 01 00 84 00 00 08 B0 03

【接收数据：】 02 00 00 0D 1A 00 06 06 BC 11 57 6A F1 58 90 00 9A 03

**ISO7816-4 SAM/SIM 卡专用功能：向第 N 号 SAM 卡发送 PPS 请求**

**功能描述：**用于向第 N 号 ISO7816-4 CPU/SAM/SIM 卡发送 PPS 请求。

**发送数据序列：**

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04/0x06	0x19	发送数据	0xXX	0x03

发送数据：长度为 0x04 时，只有 1 字节 mode

1 字节 mode:

bit1-0: =bXX (备用)

bit3-2: =b10 (升频到 38400)

bit7-4: =b0000 (SAM 卡编号 1)

=b0001 (SAM 卡编号 2)

=b0010 (SAM 卡编号 3)

.....

长度为 0x06 时, 1 字节 mode+2 字节 setting

1 字节 mode:

bit1-0: =bXX (备用)

bit3-2: =b11 (升频到 38400 或者 115200)

bit7-4: =b0000 (SAM 卡编号 1)

=b0001 (SAM 卡编号 2)

=b0010 (SAM 卡编号 3)

.....

setting[0]: =0x10 (固定值)

setting[1]: =0x13 (升频到 38400)

=0x18 (升频到 115200)

#### 正确返回数据序列:

正确返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x19	0x00	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉;

#### 错误返回数据序列:

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x19	非零	空	0xXX	0x03

注: 黄色部分为模块在返回数据时, 在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的, 故在操作接收数据时需过滤掉;

发送与返回正确举例:

【发送数据:】

【接收数据:】

#### 4.7 ISO7816-4 SAM 卡函数指令 (只支持 1 个卡)

此操作流程是为了兼容老的型号读卡器, 只能支持一个 PSAM 卡, 对应读卡器内部的 1 号卡座位置

#### ISO7816-4 SAM/SIM 卡专用功能: SAM 卡初始化波特率 (只支持 1 个卡)

**功能描述:** 用于进行满足 ISO7816-4 CPU/SAM/SIM 卡初始化波特率。

**发送数据序列:**

发送数据序列

帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x04	0x36	发送数据	0xXX	0x03

发送数据：1 字节 mode

=0x00 复位 9600 的卡片

=0x01 复位 38400 的卡

=0x02 复位 115200 的卡

**正确返回数据序列：**

正确返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x36	0x00	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

**错误返回数据序列：**

错误返回数据序列

帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x36	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据:】 02 00 00 04 36 00 3A 03

【接收数据:】 02 00 00 10 03 36 00 39 03

**ISO7816-4 SAM/SIM 卡专用功能： SAM 卡复位（只支持 1 个卡）**

**功能描述：**用于进行满足 ISO7816-4 CPU/SAM/SIM 卡复位。

**发送数据序列：**

发送数据序列

帧头	模块地址	插入	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0x10	0x03	0x37	空	0xXX	0x03

**正确返回数据序列：**

正确返回数据序列

帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x37	0x00	返回数据	0xXX	0x03

返回数据：ISO7816-4 CPU/SAM/SIM 卡复位信息

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x37	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 10 03 37 3A 03

【接收数据：】 02 00 00 14 37 00 3B 6D 00 00 57 44 29 46 41 86 93 05 6D B0 09 41 56 19 03

#### ISO7816-4 SAM/SIM 卡专用功能： SAM 卡发送 APDU(COS)指令（只支持 1 个卡）

**功能描述：**用于发送 ISO7816-4 CPU/SAM/SIM 卡发送 APDU(COS)指令。

#### 发送数据序列：

发送数据序列						
帧头	模块地址	长度字	命令字	数据域	校验字	帧尾
0x02	0x00,0x00	0xXX	0x38	发送数据	0xXX	0x03

发送数据：N 字节 APDU(COS)命令

具体 APDU(COS)指令请参阅 CPU/SAM/SIM 卡片厂家数据手册；

#### 正确返回数据序列：

正确返回数据序列							
帧头	模块地址	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0xXX	0x38	0x00	返回数据	0xXX	0x03

返回数据：对应发送 APDU(COS)指令的返回数据；

#### 错误返回数据序列：

错误返回数据序列								
帧头	模块地址	插入	长度字	命令字	执行结果	数据域	校验字	帧尾
0x02	0xXX,0xXX	0x10	0x03	0x38	非零	空	0xXX	0x03

注：黄色部分为模块在返回数据时，在帧头0x02帧尾0x03之间出现了0x02或0x10或0x03后自动增加的，故在操作接收数据时需过滤掉；

发送与返回正确举例：

【发送数据：】 02 00 00 08 38 00 84 00 00 04 C8 03

```
【接收数据:】 02 00 00 09 38 00 D5 74 FA CD 90 00 E1 03
```

## 5 数据发送接收举例:

### 5.1 读卡器通用命令发送接收举例:

设置读卡器波特率并成功:

```
【发送数据:】 02 00 00 04 15 10 03 1C 03
```

```
【接收数据:】 02 00 00 10 03 15 00 18 03
```

设置读卡器为 ISO14443A 标准并成功:

```
【发送数据:】 02 00 00 04 3A 41 7F 03
```

```
【接收数据:】 02 00 00 10 03 3A 00 3D 03
```

设置关闭读卡器天线并成功:

```
【发送数据:】 02 00 00 04 05 00 09 03
```

```
【接收数据:】 02 00 00 10 03 05 00 08 03
```

控制指示灯状态:

```
【发送数据:】 02 00 00 04 6A 00 6E 03
```

```
【接收数据:】 02 00 00 10 03 6A 00 6D 03
```

控制蜂鸣器蜂鸣时间:

```
【发送数据:】 02 00 00 04 1D 50 71 03
```

```
【接收数据:】 02 00 00 10 03 1D 00 20 03
```

读取软硬件版本号:

```
【发送数据:】 02 00 00 10 03 16 19 03
```

```
【接收数据:】 02 00 00 05 16 00 01 01 1D 03
```

读取读卡器序列号:

```
【发送数据:】 02 00 00 10 03 17 1A 03
```

```
【接收数据:】 02 00 00 0B 17 00 16 10 03 24 14 55 40 01 01 0A 03
```

设置读卡器设备地址:

```
【发送数据:】 02 00 00 05 13 12 34 5E 03
```

```
【接收数据:】 02 12 34 10 03 13 00 5C 03
```

读取读卡器设备地址:



【发送数据:】 02 00 00 10 03 14 17 03

【接收数据:】 02 00 00 05 14 00 00 00 19 03

## 5.2 M1 卡发送接收举例:

Mifare S50 卡寻卡并成功

【发送数据:】 02 00 00 04 46 52 9C 03

【接收数据:】 02 00 00 05 46 00 04 00 4F 03

【发送数据:】 02 00 00 04 47 04 4F 03

【接收数据:】 02 00 00 07 47 00 42 0B C2 08 65 03

【发送数据:】 02 00 00 07 48 42 0B C2 08 66 03

【接收数据:】 02 00 00 04 48 00 08 54 03

读扇区 0 并成功

【发送数据:】 02 00 00 0B 4A 60 00 FF FF FF FF FF AF 03

【接收数据:】 02 00 00 10 03 4A 00 4D 03

【发送数据:】 02 00 00 04 4B 00 4F 03

【接收数据:】 02 00 00 13 4B 00 42 0B C2 08 83 08 04 00 62 63 64 65 66 67 68 69 30 03

【发送数据:】 02 00 00 04 4B 01 50 03

【接收数据:】 02 00 00 13 4B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 5E 03

【发送数据:】 02 00 00 04 4B 10 02 51 03

【接收数据:】 02 00 00 13 4B 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 5E 03

【发送数据:】 02 00 00 04 4B 10 03 52 03

【接收数据:】 02 00 00 13 4B 00 00 00 00 00 00 00 FF 07 80 69 FF FF FF FF FF 47 03

写扇区 0 块 1, 将块 1 全写成 1:

【发送数据:】 02 00 00 0B 4A 60 01 FF FF FF FF FF B0 03

【接收数据:】 02 00 00 10 03 4A 00 4D 03

【发送数据:】 02 00 00 14 4C 01 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 71 03

【接收数据:】 02 00 00 10 03 4C 00 4F 03

将扇区 0 块 1 初始化为钱包, 初始值为 100:

【发送数据:】 02 00 00 0B 4A 60 01 FF FF FF FF FF B0 03

【接收数据:】 02 00 00 10 03 4A 00 4D 03

【发送数据:】 02 00 00 08 4D 01 64 00 00 00 BA 03

【接收数据:】 02 00 00 10 03 4D 00 50 03

将扇区 0 块 1 充值 100:

【发送数据:】 02 00 00 0B 4A 60 01 FF FF FF FF FF B0 03

【接收数据:】 02 00 00 10 03 4A 00 4D 03

【发送数据:】 02 00 00 08 50 01 64 00 00 00 BD 03

【接收数据:】 02 00 00 10 03 50 00 53 03

将扇区 0 块 1 扣款 50:

【发送数据:】 02 00 00 0B 4A 60 01 FF FF FF FF FF FF B0 03

【接收数据:】 02 00 00 10 03 4A 00 4D 03

【发送数据:】 02 00 00 08 4F 01 32 00 00 00 8A 03

【接收数据:】 02 00 00 10 03 4F 00 52 03

读扇区 0 块 1 余额为 150:

【发送数据:】 02 00 00 0B 4A 60 01 FF FF FF FF FF FF B0 03

【接收数据:】 02 00 00 10 03 4A 00 4D 03

【发送数据:】 02 00 00 04 4E 01 53 03

【接收数据:】 02 00 00 07 4E 00 96 00 00 00 EB 03

### 5.3 Ultralight 卡发送接收举例:

Ultralight 卡寻卡并成功

【发送数据:】 02 00 00 04 46 52 9C 03

【接收数据:】 02 00 00 05 46 00 44 00 8F 03

【发送数据:】 02 00 00 10 03 33 36 03

【接收数据:】 02 00 00 0A 33 00 04 DB CF 5 E3 25 80 C4 03

读卡并成功

【发送数据:】 02 00 00 04 4B 00 4F 03

【接收数据:】 02 00 00 13 4B 00 04 DB CF 98 51 E3 25 80 17 48 00 00 00 91 53 E5 A5 03

【发送数据:】 02 00 00 04 4B 04 53 03

【接收数据:】 02 00 00 13 4B 00 04 DB CF 98 51 E3 25 80 17 48 00 00 00 91 53 E5 A5 03

【发送数据:】 02 00 00 04 4B 08 57 03

【接收数据:】 02 00 00 13 4B 00 04 DB CF 98 51 E3 25 80 17 48 00 00 00 91 53 E5 A5 03

【发送数据:】 02 00 00 04 4B 0C 5B 03

【接收数据:】 02 00 00 13 4B 00 04 DB CF 98 51 E3 25 80 17 48 00 00 00 91 53 E5 A5 03

写页, 将页 4 全写成 1:

【发送数据:】 02 00 00 08 35 04 11 11 11 11 85 03

【接收数据:】 02 00 00 10 03 35 00 38 03

Ultralight 卡休眠

【发送数据:】 02 00 00 10 03 29 2C 03

【接收数据:】 02 00 00 10 03 29 00 2C 03

#### 5.4 ISO14443A-4 CPU 卡发送接收举例:

ISO14443A-4 CPU 卡复位并成功:

【发送数据:】 02 00 00 04 53 52 A9 03

【接收数据:】 02 00 00 0F 53 00 16 61 1B 82 10 10 78 80 90 10 02 20 90 00 C0 03

FM1208 发送 COS 指令(0084000004)并成功返回:

【发送数据:】 02 00 00 08 54 00 84 00 00 04 E4 03

【接收数据:】 02 00 00 09 54 00 7B A3 5F 28 90 00 92 03

#### 5.5 ISO7816-4 CPU/SAM/SIM 卡新指令发送接收举例:

SAM 卡复位并成功:

【发送数据:】 02 00 00 04 19 00 1D 03

【接收数据:】 02 00 00 16 19 00 3B 7D 94 00 00 4C 31 76 68 10 03 4C 4B 12 10 02 16 4F 95 2D  
00 AB 03

SAM 卡发送 COS 命令并成功:

【发送数据:】 02 00 00 09 1A 01 00 84 00 00 08 B0 03

【接收数据:】 02 00 00 0D 1A 00 06 06 BC 11 57 6A F1 58 90 00 9A 03

#### 5.6 ISO7816-4 CPU/SAM/SIM 卡旧指令发送接收举例:

SAM 卡初始化波特率并成功:

【发送数据:】 02 00 00 04 36 00 3A 03

【接收数据:】 02 00 00 10 03 36 00 39 03

SAM 卡复位并成功:

【发送数据:】 02 00 00 10 03 37 3A 03

【接收数据:】 02 00 00 14 37 00 3B 6D 00 00 57 44 29 46 41 86 93 05 6D B0 09 41 56 19 03

SAM 卡发送 COS 命令并成功:

【发送数据:】 02 00 00 08 38 00 84 00 00 04 C8 03

【接收数据:】 02 00 00 09 38 00 D5 74 FA CD 90 00 E1 03