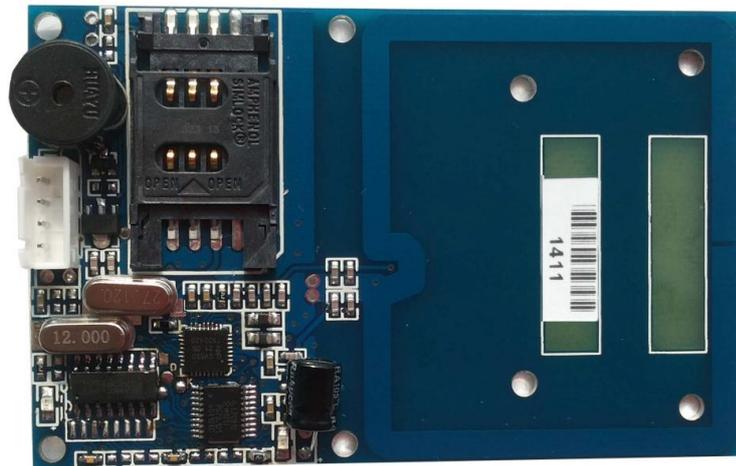




北京圆志科信 读写卡模块

应用手册



M209CX

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	2014.3.8
V1.1	更新 PSAM 指令, 支持 2 个 PSAM 卡	2014.11.24
V1.2	增加新指令:0x12,0x04,0x17,0x16,0x14,0x13,0x2a,0x49	2015.8.17

目 录

0.1 声明	2
0.2 更改历史记录	3
1. 概述	6
1.1 M209CX 读写模块简介:	6
1.2 产品型号及之间的区别:	6
1.3 技术指标	6
1.4 外观尺寸	7
1.4.1 外观示意图	7
1.4.2 尺寸图	8
2. 硬件连接	8
2.1 引脚定义	8
2.2 RS232 串口的连接 (适用 M209CA)	9
2.3 UART 口的连接 (适用于 M209CT)	9
2.4 I ² C 口的连接 (适用于 M209CI)	9
2.5 USB HID 接口的连接 (适用于 M209CH)	9
2.6 上电状态	9
3. 通过模块操作卡片步骤简述	10
3.1 通过模块操作 Mifare One 卡步骤简述:	10
3.2 通过模块操作 ISO14443 TYPE A CPU 卡和 SAM 卡步骤简述:	10
4. 与电脑连接 (可调用圆志 DLL 动态库)	10
4.1 库函数说明	11
通用函数功能: 获取动态库版本号	11
通用函数功能: DES 算法加密函数	11
通用函数功能: DES 算法解密算法函数	11
通用函数功能: 初始化串口	11

通用函数功能：关闭串口	12
通用函数功能：蜂鸣器控制	12
通用函数功能：控制指示灯	12
接触式 CPU 卡专用功能：设置读写卡器 PSAM 卡通讯波特率	12
接触式 CPU 卡专用功能：复位 PSAM 卡	13
接触式 CPU 卡专用功能：向 PSAM 发送 COS 命令	13
非接触卡类型选择专用功能：设置读写卡器非接触工作方式	13
非接触卡通用函数功能：设置读写卡器天线状态	13
M1 卡/Ultralight 专用功能：寻 TYPE_A 卡	14
M1 卡/Ultralight 专用功能：命令已激活的 TYPE_A 卡进入 HALT 状态	14
M1 卡/Ultralight 专用功能：读取 Ultralight 卡或者 MifareOne 卡一块数据	14
Ultralight 专用功能：向 Ultra light 卡中写入一块数据	14
M1 卡专用功能：防冲突	15
M1 卡专用功能：选卡	15
M1 卡专用功能：验证 MifareOne 卡密钥	15
M1 卡专用功能：写入 MifareOne 卡一块数据	15
M1 卡专用功能：将 Mifare One 卡某一扇区初始化为钱包	16
M1 卡专用功能：读取 Mifare One 卡钱包值	16
M1 卡专用功能：Mifare One 卡扣款	16
M1 卡专用功能：Mifare One 卡充值	16
M1 卡专用功能：Mifare One 卡数据回传	17
M1 卡专用功能：Mifare One 卡数据传送	17
TYPE A CPU 卡专用功能：寻感应区内符合 ISO14443 TYPE_A 标准的 CPU 卡并复位	17
TYPE A CPU 卡专用功能：向符合 ISO14443-4 标准的非接触 CPU 卡发送 COS 命令	17
5. 底层数据通讯协议：	18
5.1 通讯协议	18
5.2 命令列表	19
附录 1：指令汇总表	31

1. 概述

1.1 M209CX 读写模块简介:

M209CX 读写模块采用 13.56MHz 非接触射频技术, 内嵌恩智浦射频基站。用户不必关心射频基站的复杂控制方法, 只需通过简单的选定接口发送命令或操作函数就可以实现对卡片完全的操作。该系列读写模块支持 ISO14443-A Ultralight, Mifare One S50,S70, FM11RF08,FM1208 ,PSAM9600(T=0&T=1),PSAM38400 及其兼容卡片。

1.2 产品型号及之间的区别:

型号	主要区别	支持卡型
M209CA	RS232 接口	ISO14443-A: Mifare One S50,S70,Ultra Light,FM11RF08, FM1208; ISO7816: PSAM9600,PSAM38400,PSAM115200 (T=0& T=1) 及其兼容卡片。
M209CT	UART 接口 (TTL)	
M209CI	I ² C 接口 (可定制)	
M209CH	USB HID 接口 (免驱)	
备注: 可选配带 2 个 PSAM。		

1.3 技术指标

- 射频基站: 恩智浦射频基站
- 工作频率: 13.56MHz
- 支持的标准: ISO14443A & ISO7816
- 可读卡型 :
 - ISO14443-A: Mifare One S50,S70,Ultra Light,FM11RF08, FM1208及其兼容卡片;
 - ISO7816: PSAM9600,PSAM38400, PSAM115200 (T=0&T=1) 及其兼容卡片;
- 供电电压: DC3.3~5V
- 接口: RS232, UART, I²C, USB H ID 可选。
- 通讯速率: M209CA(RS232), M209CT(UART)默认 19.2Kbps; M209CH 为 USB 免驱。
- 最大功耗: 100mA
- 读卡距离: 40-60mm

- 工作温度: -10°C ~ +70°C
- 相对湿度: 35% ~ 95%
- 外形尺寸: 86 * 55 * 15 (mm)

1.4 外观尺寸

1.4.1 外观示意图

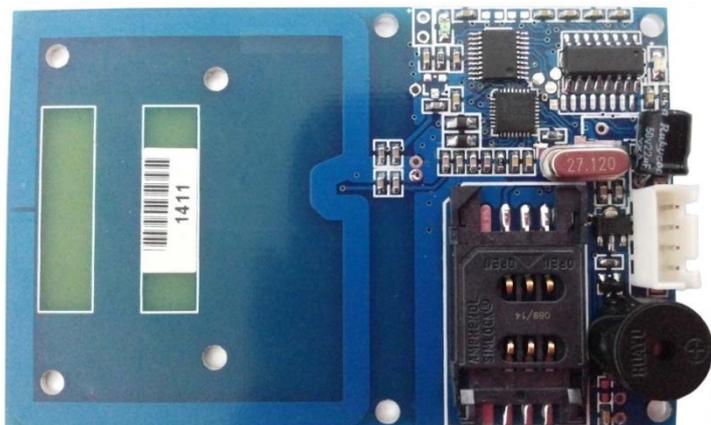


图 1: M209CA 外观示意图

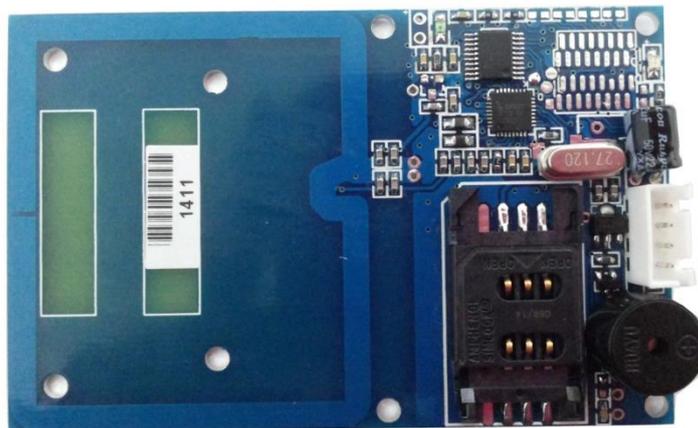


图 2: M209CT/CI 外观示意图

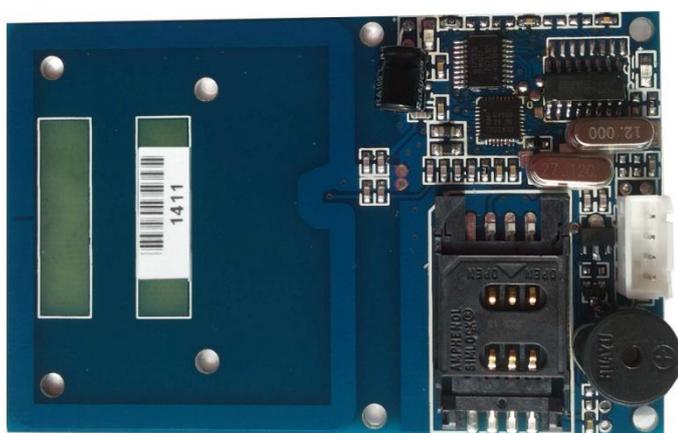


图 3: M209CH 外观示意图

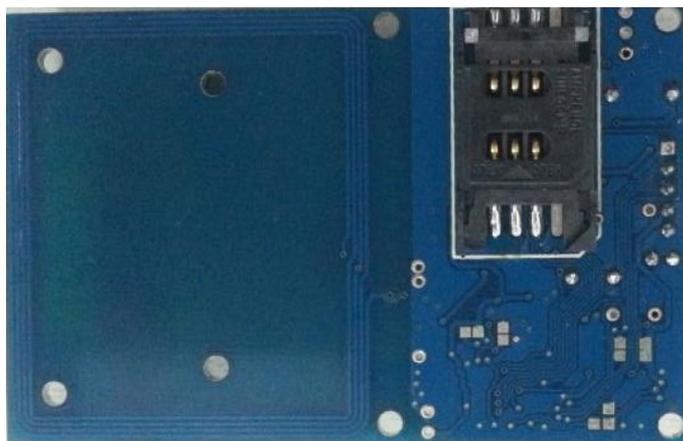
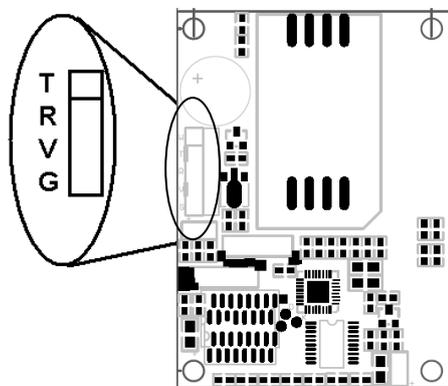
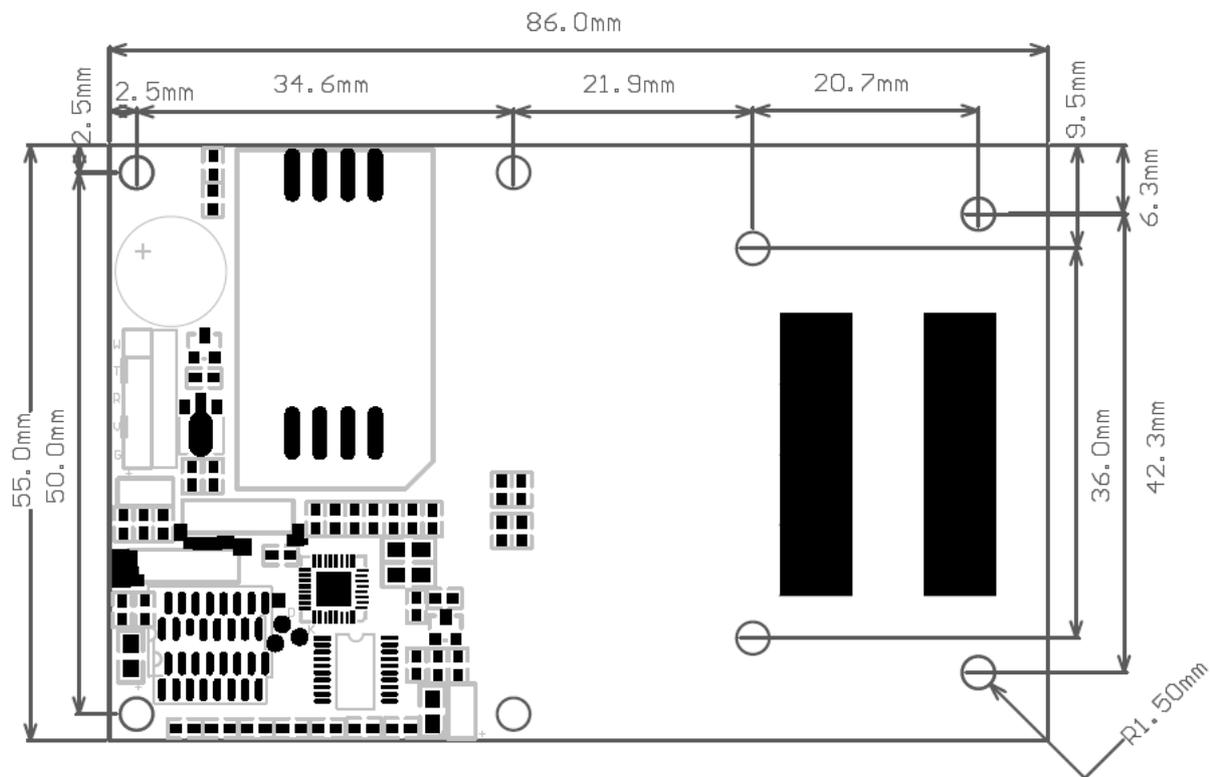


图 4: 背面图 (2#PSAM 卡)

1.4.2 尺寸图

外形尺寸： 86 mm* 55mm； 安装孔半径： 1.5mm。

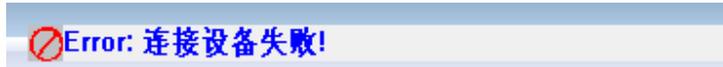


管脚	名称	管脚功能			
		M209CA (RS232)	M209CT (UART)	M209CI (I ² C)	M209CH (USB HID)
1	T	RS232 串口发送	UART 串口发送	数据线 SDA	D+
2	R	RS232 串口接收	UART 串口接收	时钟线 SCL	D-
3	V	5V 直流电源正	5V 直流电源正	5V 直流电源正	5V 直流电源正
4	G	直流电源地	直流电源地	直流电源地	直流电源地

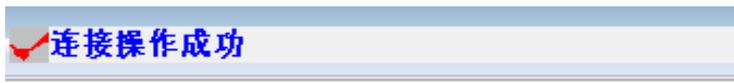
2.2 RS232 串口的连接 (适用 M209CA)

2.2.1 公司自制 9 针 D 型串口线适用于 M209CA(产品出厂默认不带线), 将 9 针 D 型串口线的串口座插到计算机的 COM 口, 线上的 USB 接口(取电口)插到计算机 USB 口上, 使读卡器取到+5V 电源。(注: 客户也可根据管脚定义自己将读卡模块连接到计算机上)

2.2.2 打开 PC_DEMO 软件,选择对应端口及 19200 波特率, 点击连接, 如果端口设置错误右上角会出现:



如果端口设置正确, 右上角会显示



2.3 UART 口的连接 (适用于 M209CT)

2.3.1 根据管脚定义, 将读卡模块的 T(UART 串口发送)接到客户 MCU 上的 RXD(UART 串口接收), 读卡模块的 R(UART 串口接收)接客户 MCU 上的 TXD(UART 串口发送), 并给 5V 电源通电。

2.3.2 如使用计算机 PC_DEMO 软件进行辅助开发, 可将 UART 接口转成 RS232 接口并按“2.2 RS232 串口的连接”接计算机。

2.4 I²C 口的连接 (适用于 M209CI)

2.4.1 根据管脚定义, 将读卡模块的 T(数据线 SDA)接到客户 MCU 上的数据线 SDA, 读卡模块的 R(时钟线 SCL)接客户 MCU 上的时钟线 SCL, 并给 5V 电源通电。

2.4.2 如使用计算机 PC_DEMO 软件进行辅助开发, 可将 I²C 接口转成 RS232 接口并按“2.2 RS232 串口的连接”接计算机。

2.5 USB HID 接口的连接 (适用于 M209CH)

2.5.1 公司自制 USB 线适用于 M209CH(产品出厂默认不带线), 将 USB 线一端接读卡模块, 一端插到计算机的 USB 口, (注: 客户也可根据管脚定义自己将读卡模块连接到计算机上)。

2.5.2 USB HID 为免驱的, 可直接打开公司附带的 PC_DEMO_HID 软件, 点击“设置”→“端口设置”→“连接”。

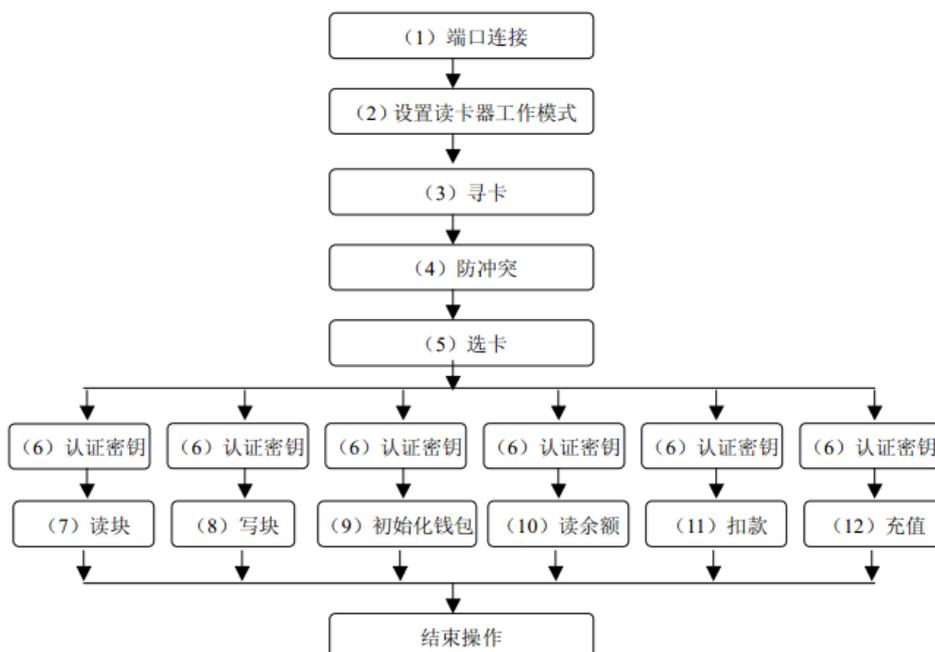
如果端口连接成功则会出现: , 否则显示错误。

2.6 上电状态

读卡器上电后的默认波特率为 19200, 绿发光二极管(读卡指示灯)闪烁几次后熄灭, 红发光二极管(电源指示灯)长亮。

3. 通过模块操作卡片步骤简述

3.1 通过模块操作 Mifare One 卡步骤简述:



注意事项:

- (1) 寻卡, 防冲突, 选卡成功之后才可以进行块的读写以及钱包功能等操作;
- (2) 在进行块的读写, 钱包等相关操作之前还需要进行密钥认证, 只有通过才可以进行相应操作;
- (3) 想将某块作为钱包功能时, 第一次必须用初始化钱包指令将该块进行初始化;
- (4) 在做钱包备份时, 必须在同一扇区内进行操作;

3.2 通过模块操作 ISO14443 TYPE A CPU 卡和 SAM 卡步骤简述:



4. 与电脑连接 (可调用圆志 DLL 动态库)

M209CAM209CT\M209CI: 调用下面 DLL 函数之前, 需将该对应动态库 MasterRDnew.dll 和 MasterCom.dll 复制到 C:\Windows\system32 文件夹下。

M209CH: 在调用下面 DLL 函数之前, 需将该对应动态库 MasterRDHID.dll 和 MasterHID.dll 复制到 C:\Windows\system32 文件夹下。

4.1 库函数说明

通用函数功能: 获取动态库版本号

原型: int WINAPI lib_ver(unsigned int *nVer)

参数: *nVer: 2 字节动态库版本号

返回: 成功返回 0

通用函数功能: DES 算法加密函数

原型: int (WINAPI* des_encrypt)(unsigned char *szOut, unsigned char *szIn, unsigned int inlen, unsigned char *key, unsigned int keylen);

参数: szOut: 输出的 DES 值, 长度等于明文长度

szIn: 明文

inlen: 明文长度, 8 字节的整数倍

key: 密钥

keylen: 密钥长度, 如果大于 8 字节, 是 3des, 如果小于等于 8 字节单 des. 不足补零

返回: 成功返回 0

通用函数功能: DES 算法解密算法函数

原型: int (WINAPI* des_decrypt)(unsigned char *szOut, unsigned char *szIn, unsigned int inlen, unsigned char *key, unsigned int keylen);

参数: szOut: 输出的 DES 值, 长度等于密文长度

szIn: 密文

inlen: 密文长度, 8 字节的整数倍

key: 密钥

keylen: 密钥长度, 如果大于 8 字节, 是 3des, 如果小于等于 8 字节单 des. 不足补零

返回: 成功返回 0

通用函数功能: 初始化串口

M209CA\CTCI:

原型: int WINAPI rf_init_com (unsigned short icdev, int port, long baud)

参数: icdev: 通讯设备标识符, 0-65536

port: 串口号, 取值为 1~9

baud: 为通讯波特率 4800 ~ 115200

返回: 成功返回 0

M209CH:

原型: int WINAPI rf_init_com (unsigned short icdev,int nVID,int nPID)

参数: icdev: 通讯设备标识符, 0-65535

nVID: 6790 (0x1A86)

nPID: 57360 (0xE010)

返回: 成功返回 0

通用函数功能: 关闭串口

原型: int WINAPI rf_ClosePort ()

参数: 空

返回: 成功返回 0

通用函数功能: 蜂鸣器控制

原型: int WINAPI rf_beep (unsigned short icdev, unsigned char msec)

参数: icdev: 通讯设备标识符, 0-65536

msec: 蜂鸣时限, 单位是 10 毫秒

返回: 成功返回 0

通用函数功能: 控制指示灯

原型: int WINAPI rf_light(unsigned short icdev, unsigned char color)

参数: icdev: 通讯设备标识符, 0-65536

color: 1 = 熄灭黄灯

2 = 点亮黄灯

接触式 CPU 卡专用功能: 设置读写卡器 PSAM 卡通讯波特率

原型: int WINAPI rf_init_sam (unsigned short icdev, unsigned char bound)

参数: icdev: 通讯设备标识符, 0-65536

bound: 1 字节波特率选择及字节卡片序号:

bit1,bit0(字节波特率选择)

00: 9600; 01: 38400;

返回: 成功返回 0

接触式 CPU 卡专用功能：复位 PSAM 卡

原型：int WINAPI rf_sam_rst(unsigned short icdev, unsigned char *pData, unsigned char *pMsgLg)

参数：icdev： 通讯设备标识符， 0-65536

 pData： 返回的复位信息内容

 pMsgLg： 返回复位信息的长度

返回：成功返回 0

接触式 CPU 卡专用功能：向 PSAM 发送 COS 命令

原型：int WINAPI rf_sam_cos(unsigned short icdev, unsigned char *command, unsigned char cmdLen, unsigned char *pData, unsigned char* Length)

参数：icdev： 通讯设备标识符， 0-65536

 command： COS 命令

 cmdLen： COS 命令长度

 pData： 卡片返回的数据， 含 SW1、SW2

 pMsgLg： 返回数据长度

返回：成功返回 0

非接触卡类型选择专用功能：设置读写卡器非接触工作方式

原型：int WINAPI rf_init_type (unsigned short icdev, unsigned char type)

参数：icdev： 通讯设备标识符， 0-65536

 type： 读写卡器工作方式

返回：成功返回 0

说明：type='A'： 设置为 TYPE_A 方式

非接触卡通用函数功能：设置读写卡器天线状态

原型：int WINAPI rf_antenna_sta (unsigned short icdev, unsigned char model)

参数：icdev： 通讯设备标识符， 0-65536

 model： 天线状态

返回：成功返回 0

说明：model=0： 关闭天线 model=1： 开启天线

M1 卡/Ultralight 专用功能：寻 TYPE_A 卡

原型：int WINAPI rf_request (unsigned short icdev, unsigned char model, unsigned short *TagType)

参数：icdev： 通讯设备标识符， 0-65536

model： 寻卡模式

TagType： 返回卡类型值

返回： 成功返回 0

说明： mode=0x26： 寻未进入休眠状态的卡

mode=0x52： 寻所有状态的卡

M1 卡/Ultralight 专用功能：命令已激活的 TYPE_A 卡进入 HALT 状态

原型：int WINAPI rf_halt(unsigned short icdev)

参数：icdev： 通讯设备标识符， 0-65536

返回： 成功返回 0

M1 卡/Ultralight 专用功能：读取 Ultralight 卡或者 MifareOne 卡一块数据

原型：int WINAPI rf_M1_read (unsigned short icdev, unsigned char block, unsigned char *pData, unsigned char *pLen)

参数：icdev： 通讯设备标识符， 0-65536

block： M1 卡绝对块号

pData： 读出数据

pLen： 读出数据的长度

返回： 成功返回 0

Ultralight 专用功能：向 Ultra light 卡中写入一块数据

原型：int WINAPI int rf_ul_write (word icdev, unsigned char page, unsigned char *pData)

参数：icdev： [IN] 通讯设备标识符

page： [IN] ultra light 卡页地址 (0 ~ 0x0f)

pData： [IN] 写入的数据， 4 字节

返回： 成功返回 0

M1 卡专用功能：防冲突

原型: int WINAPI rf_anticoll(word icdev, unsigned char bcnt, unsigned char *pSnr, unsigned char *pLen)

参数: icdev: 通讯设备标识符.(0-65536)

bcnt: 值=4

pSnr: 返回卡号

pLen: 返回卡号长度

返回: 成功返回 0

M1 卡专用功能：选卡

原型: int WINAPI rf_select(word icdev, unsigned char *pSnr, unsigned char snrLen, unsigned char *pSize)

参数: icdev: 通讯设备标识符. (0-65536)

pSnr: 卡号

snrLen: 卡号长度

pSize: 卡的类型

返回: 成功返回 0

说明: 当选卡指令执行完毕后, 卡片进入到激活状态, 同时在同样的天线区域, 仅有一张卡片处于激活状态。

M1 卡专用功能：验证 MifareOne 卡密钥

原型: int WINAPI rf_M1_authentication2(unsigned short icdev, unsigned char model, unsigned char block, unsigned char *key)

参数: icdev: 通讯设备标识符, 0-65536

model: 密码验证模式

block: 要验证密码的绝对块号

key: 密钥内容, 6 字节

返回: 成功返回 0

说明: model=0x60: 验证 A 密钥

model=0x61: 验证 B 密钥

M1 卡专用功能：写入 MifareOne 卡一块数据

原型: int WINAPI rf_M1_write (unsigned short icdev, unsigned char block, unsigned char *data)

参数: icdev: 通讯设备标识符, 0-65536

block: M1 卡绝对块号
data: 写入的数据, 16 字节
返回: 成功返回 0

M1 卡专用功能: 将 Mifare One 卡某一扇区初始化为钱包

原型: int WINAPI rf_M1_initval (unsigned short icdev, unsigned char block, long value)
参数: icdev: 通讯设备标识符, 0-65536
block: M1 卡绝对块号
value: 初始值, 16 进制, 低字节在前
返回: 成功返回 0

M1 卡专用功能: 读取 Mifare One 卡钱包值

原型: int WINAPI rf_M1_readval(WORD icdev, unsigned char block, long* pValue)
参数: icdev: 通讯设备标识符, 0-65536
block: M1 卡绝对块号
pValue: 返回的值, 16 进制, 低字节在前
返回: 成功返回 0

M1 卡专用功能: Mifare One 卡扣款

原型: int WINAPI rf_M1_decrement (unsigned short icdev, unsigned char block, long value)
参数: icdev: 通讯设备标识符, 0-65536
block: M1 卡绝对块号
value: 要扣的值, 16 进制, 低字节在前
返回: 成功返回 0

M1 卡专用功能: Mifare One 卡充值

原型: int WINAPI rf_M1_increment (unsigned short icdev, unsigned char block, long value)
参数: icdev: 通讯设备标识符, 0-65536
block: M1 卡绝对块号
value: 要增加的值, 16 进制, 低字节在前
返回: 成功返回 0

M1 卡专用功能: Mifare One 卡数据回传

原型: int WINAPI rf_M1_restore (unsigned short icdev, unsigned char block)

参数: icdev: 通讯设备标识符, 0-65536

block: M1 卡绝对块号

返回: 成功返回 0

说明: 用此函数将指定的块内容传入卡的 buffer, 然后可用 rf_M1transfer()函数将 buffer 中数据再传送到另一块中去

M1 卡专用功能: Mifare One 卡数据传送

原型: int WINAPI rf_M1_transfer (unsigned short icdev, unsigned char block)

参数: icdev: 通讯设备标识符, 0-65536

block: M1 卡绝对块号

返回: 成功返回 0

说明: 该函数仅在 increment、decrement 和 restore 命令之后调用

TYPE A CPU 卡专用功能: 寻感应区内符合 ISO14443 TYPE_A 标准的 CPU 卡并复位

原型: int WINAPI rf_typea_rst(word icdev, unsigned char model, unsigned char *pData, unsigned char *pMsgLg)

参数: icdev: [IN] 通讯设备标识符

model: [IN] 寻卡方式

pDate: [OUT]返回的数据

pMsgLg: [OUT]返回数据的长度

返回: 成功返回 0

说明: mode = 0x26: 寻未进入休眠状态的卡

mode = 0x52: 寻所有状态的卡

pDate: 4 字节 CSN + 复位信息内容

TYPE A CPU 卡专用功能: 向符合 ISO14443-4 标准的非接触 CPU 卡发送 COS 命令

原型: int WINAPI rf_cos_command(word icdev, unsigned char *pCommand, unsigned char cmdLen, unsigned char *pData, unsigned char *pMsg)

参数: icdev: [IN] 通讯设备标识符

pCommand: [IN] cos 命令

cmdLen: [IN] cos 命令长度

pDate: [OUT]卡片返回的数据, 含 SW1、SW2

pMsgLg: [OUT]返回数据长度
返回: 成功则返回 0

5. 底层数据通讯协议:

5.1 通讯协议

- ✚ 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: 默认 19200; (注: M209AH 为 USB 接口波特率)
- ✚ 发送数据封包格式:

数据包帧头 02	数据包内容	数据包帧尾 03
----------	-------	----------

注:0x02、0x03 被使用为起始字符、结束字符, 0x10 被使用为 0x02,0x03 的辨识字符。因此在此在通讯的传输数据之中 (起始字符 0x02, 至结束字符 0x03 之中) 的 0x02、0x03、0x10 字符之前, 皆必须补插入 0x10 做为数据辨识之用。例如起始字符 0x02, 至结束字符 0x03 之中有一原始数据为 0x020310, 补插入辨识字符之后, 将变更为 0x100210031010。

数据包内容:

模块地址	长度字	命令字	数据域	校验字
------	-----	-----	-----	-----

模块地址: 对于单独使用的模块来说固定为 0x0000;

对网络版模块来说为 0x0001~0xFFFFE;

0xFFFF 为广播。

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 (最后一字节)。

返回数据封包格式: 同发送数据封包格式相同

数据包内容:

模块地址	长度字	接收到的命令字	执行结果	数据域	校验字
------	-----	---------	------	-----	-----

模块地址：对与单独使用的模块来说固定为 0x0000；

对网络版模块来说为本身的地址；

长度字： 指明从长度字到**数据域最后一字节**的字节数

命令字： 本条命令的含义

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（最后一字节）。

5.2 命令列表

序号	命令名称	发送	命令字	数据域说明
1	设置模块工作模式	发送	0x3A	1 字节寻卡模式： 发送数据=“A”表示使模块工作于 ISO14443 TYPE A 模式，对应 ASCII 码为 0x41；
		返回	0x3A	无
2	Mifare one/Ultralight 卡寻卡	发送	0x46	1 字节模式控制字： 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；
		返回	0x46	返回 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 Ultralight 卡；
3	Ultralight 卡选卡	发送	0x33	空

序号	命令名称	发送	命令字	数据域说明
		返回	0x33	7 字节返回的卡号
4	Ultralight 卡读卡	发送	0x4B	1 字节的起始页的页号
		返回	0x4B	返回从起始页开始的连续 4 页内容
5	Ultralight 卡写卡	发送	0x35	1 字节页号+4 字节写入数据
		返回	0x35	空
6	Mifare one 卡防冲突	发送	0x47	数据部分为 1 字节卡序列号字节数; 发送数据= "0x04" Mifare S50,S70,FM11RF08 卡序列号为 4 字节, 故数据为 0x04;
		返回	0x47	返回 4 字节卡序列号
7	Mifare one 卡选卡	发送	0x48	4 字节卡号
		返回	0x48	返回数据为卡容量 数据 0x08 表示 Mifare one S50; 数据 0x20 表示 Mifare one S70;
8	Mifare one 卡密钥验证	发送	0x4A	发送数据部分; 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 "FFFFFFFFFFFFFF";
		返回	0x4A	空

序号	命令名称	发送	命令字	数据域说明
9	Mifare one 卡读块	发送	0x4B	发送数据部分; 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);
		返回	0x4B	返回数据为该块 16 字节内容
10	Mifare one 卡写块	发送	0x4C	发送数据部分; 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。
		返回	0x4C	空
11	Mifare one 卡初始化钱包	发送	0x4D	发送数据部分; 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。
		返回	0x4D	空
12	Mifare one 卡读钱包	发送	0x4E	发送数据部分; 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);

序号	命令名称	发送	命令字	数据域说明
		返回	0x4E	返回数据为 4 字节 16 进制钱包值, 低字节在前;
13	Mifare one 卡充值	发送	0x50	发送数据部分; 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 16 进制增加值: 为充值的金额, 低字节在前; 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。
		返回	0x50	空
14	Mifare one 卡扣款	发送	0x4F	发送数据部分; 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 16 进制增加值: 为扣款的金额, 低字节在前; 密钥块 (每个扇区的最后一块) 不能进行扣款操作, 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。
		返回		空
15	Mifare one 卡钱包备份步骤 1	发送	0x51	发送数据部分; 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。
		返回	0x51	空

序号	命令名称	发送	命令字	数据域说明
16	Mifare one 卡钱包备份步骤 2	发送	0x52	发送数据部分; 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);
		返回	0x52	空
17	Mifare one/Ultralight 卡休眠	发送	0x29	空
		返回	0x29	空
18	ISO14443 TYPE A CPU 卡复位	发送	0x53	发送数据=0x26 表示寻天线范围内的未休眠卡; 发送数据=0x52 表示寻天线范围内的所有卡;
		返回	0x53	4 字节 CSN+返回信息
19	ISO14443 TYPE A CPU 卡发送 COS 指令	发送	0x54	COS 指令
		返回	0x54	针对 COS 指令的返回数据
20	设置 PSAM 卡通讯波特率	发送	0x36	发送数据=0x00 表示选择普通 PSAM 卡通讯波特率 9600; 发送数据=0x01 表示选择高速 PSAM 卡通讯波特率 38400;
		返回	0x36	空
21	PSAM 卡复位	发送	0x37	空
		返回	0x37	复位信息
21-1	PSAM 卡复位	发送	0x19	详见下文

序号	命令名称	发送	命令字	数据域说明
		返回	0x19	详见下文
22	发送 PSAM 卡 COS 指令	发送	0x38	COS 指令
		返回	0x38	针对 COS 指令的返回数据
22-1	发送 PSAM 卡 COS 指令	发送	0x1A	详见下文
		返回	0x1A	详见下文
23	设置波特率	发送	0x15	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: 模块默认的波特率是 19200, 模块不会保存此设置, 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;
		返回	0x15	空
24	设置模块天线状态	发送	0x05	详见下文
		返回	0x05	详见下文

序号	命令名称	发送	命令字	数据域说明
25	控制 LED 引脚状态	发送	0x6A	数据=0x00 LED 引脚输出高电平, 如果外接 LED 指示灯, 则灯熄灭; 数据=0x03 LED 引脚输出低电平, 如果外接 LED 指示灯, 则灯点亮;
		返回	0x6A	空
26	设置蜂鸣器时间	发送	0x1D	数据部分为 1 字节蜂鸣时间, 范围 0x00~0xFF, 单位为 ms
		返回	0x1D	空
27	设置射频基站与卡通信指令等待时间	发送	0x12	0x80+2 字节参数 (高字节在前低字节在后) 上电默认等待时间数据为 50, 掉电丢失该参数
		返回	0x12	空
28	读卡器休眠	发送	0x04	空
		返回	0x04	空
29	获得读卡器序列号	发送	0x17	空
		返回	0x17	8 字节产品序列号
30	获得读卡器版本号	发送	0x16	空
		返回	0x16	读卡器 2 字节硬件版本号
31	获得读卡器设备标识号	发送	0x14	空

序号	命令名称	发送	命令字	数据域说明
		返回	0x14	2 字节读卡器标识号
32	设置读卡器设备标识号	发送	0x13	2 字节读卡器标识号
		返回	0x13	空
33	下载密钥到读卡器存储:(目前不支持,可定制)	发送	0x2a	1 字节密钥序号+6 字节密钥
		返回	0x2a	空
34	使用已下载的密钥认证卡片:(目前不支持,可定制)	发送	0x49	1 字节密钥模式+1 字节绝对块+1 字节密钥序号(0-9) 发送数据部分; 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); 1 字节密钥序号范围是 0-9 注:新出厂的卡片默认密钥模式为 A 密钥,6 字节密钥为“FFFFFFFFFFFF”;
		返回	0x49	空

PSAM 卡复位: (新增加指令, 逐渐替代 37 指令, 功能更强大)

功能描述: 用于多张 PSAM 卡的复位及修改波特率操作。

发送数据序列:

帧头	发送数据包内容						帧尾
	模块地址	插入	长度	命令	发送数据	校验	
0x02	0x00,0x00	0x10	0x03	0x19	见注 2	0x3A	0x03

注1: 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；

注2：发送数据定义：

- 1) 复位：1字节卡片序号（高4位）及复位指令（低2位bit3,bit2）波特率选择(低2位bit1,bit0)；
- 2) 修改波特率(前提是SAM卡要支持)：1字节卡片序号(高4位)及修改波特率指令(低2位bit3,bit2)+2字节波特率选择。

2字节修改速率指令如下：

0x10, 0x11 表示把通信速率修改为9600；

0x10, 0x13 表示把通信速率修改为38400；

0x10, 0x18 表示把通信速率修改为115200。

- 3) 修改SAM卡工作频率：1字节修改指令（高4位）及频率选择（低4位）。

数据定义表：

定义		1 字节								2 字节	3 字节
		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	十六进制	
执行 复位	复位 9600 波特率卡					0	0	0	0	无	
	复位 38400 波特率卡					0	0	0	1		
	复位 115200 波特率卡					0	0	1	0		
改变 SAM 卡的波特 速率	修改为 9600					1	1	X	X	10	11
	修改为 38400					1	1	X	X	10	13
	修改为 115200					1	1	X	X	10	18
bit7,bit6,b it5,bit4 (卡序号选 择)	操作 1#PSAM 卡	0	0	0	0						
	操作 2#PSAM 卡	0	0	0	1						
	操作 3#PSAM 卡	0	0	1	0						
	操作 4#PSAM 卡	0	0	1	1						
	操作 5#PSAM 卡	0	1	0	0						
	操作 6#PSAM 卡	0	1	0	1						
	操作 7#PSAM 卡	0	1	1	0						
修改 SAM 卡	修改为 1MHz	1	1	1	1	0	0	0	1	无	
	修改为 2MHz	1	1	1	1	0	0	1	0		

工作频率	修改为 3MHz	1	1	1	1	0	0	1	1		
	修改为 4MHz	1	1	1	1	0	1	0	0		
	修改为 6MHz	1	1	1	1	0	1	1	0		
	...										
	修改为 12MHz	1	1	1	1	1	1	0	0		
修改 SAM 指令响应等待时间		1	1	1	0	X	X	X	X	A:0-127 实际等待时间 {A*300ms+300ms}	无

正确返回数据序列:

帧头	正确返回数据包内容						帧尾
	模块地址	长度	命令	执行结果	返回数据	校验	
0x02	0x00,0x00	0"X"XX	0x19	0x00	复位信息	0"X"XX	0x03

错误返回数据序列:

帧头	错误返回数据包内容							帧尾
	模块地址	插入	长度	命令	执行结果	返回数据	校验	
0x02	0x00,0x00	0x10	0x03	0x19	非零	空	0"X"XX	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:

1>1#9600 卡的复位, 和以前的 37 指令相同功能

【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

2>复位后发 PPS 指令将 1#卡变为 38400 波特率

【发送数据:】 02 00 00 06 19 0C 10 10 13 4E 03

【接收数据:】 02 00 00 10 03 19 00 1C 03

3>复位后发 PPS 指令将 1#卡变为 115200 波特率

【发送数据:】 02 00 00 06 19 0C 10 10 18 53 03

【接收数据:】 02 00 00 10 03 19 00 1C 03

4>修改 SAM 卡工作频率为 1MHZ

【发送数据:】 02 00 00 04 19 F1 0E 03

【接收数据:】 02 00 00 10 03 19 00 1C 03

5>修改 SAM 卡工作频率为 2MHZ
 【发送数据:】 02 00 00 04 19 F2 0F 03
 【接收数据:】 02 00 00 10 03 19 00 1C 03

6>修改 SAM 卡工作频率为 3MHZ
 【发送数据:】 02 00 00 04 19 F3 10 10 03
 【接收数据:】 02 00 00 10 03 19 00 1C 03

7>修改 SAM 卡工作频率为 4MHZ
 【发送数据:】 02 00 00 04 19 F4 11 03
 【接收数据:】 02 00 00 10 03 19 00 1C 03

8>修改 SAM 卡工作频率为 6MHZ
 【发送数据:】 02 00 00 04 19 F6 13 03
 【接收数据:】 02 00 00 10 03 19 00 1C 03

9>修改 SAM 卡工作频率为 12MHZ
 【发送数据:】 02 00 00 04 19 FC 19 03
 【接收数据:】 02 00 00 10 03 19 00 1C 03

发送 PSAM 卡 COS 指令: (新增加指令, 逐渐替代 38 指令, 功能更强大)

功能描述: 用于对多张 PSAM 卡发送 COS 指令。

发送数据序列:

帧头	发送数据包内容					帧尾
	模块地址	长度	命令	发送数据	校验	
0x02	0x00,0x00	0"X"XX	0x1A	1 字节 SAM 卡编号 (0-1) +n 字节 COS 指令	0"X"XX	0x03

注: 发 COS 指令 1 字节卡编号定义表:

定义	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
操作 1#PSAM 卡	0	0	0	0	0	0	0	0
操作 2#PSAM 卡	0	0	0	0	0	0	0	1
操作 3#PSAM 卡	0	0	0	0	0	0	1	0
操作 4#PSAM 卡	0	0	0	0	0	0	1	1
操作 5#PSAM 卡	0	0	0	0	0	1	0	0
操作 6#PSAM 卡	0	0	0	0	0	1	0	1
操作 7#PSAM 卡	0	0	0	0	0	1	1	0

正确返回数据序列:

帧头	正确返回数据包内容						帧尾
	模块地址	长度	命令	执行结果	返回数据	校验	
0x02	0x00,0x00	0"X"XX	0x1A	0x00	针对 COS 指令的返回数据	0"X"XX	0x03

错误返回数据序列:

帧头	错误返回数据包内容							帧尾
	模块地址	插入	长度	命令	执行结果	返回数据	校验	
0x02	0x00,0x00	0x10	0x03	0x1A	非零	空	0"X"XX	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:

对 2 号卡发送 COS 指令 "0084000008"

【send data:】 02 00 00 09 1A 01 00 84 00 00 08 B0 03

【Receive data :】 02 00 00 0D 1A 00 06 06 BC 11 57 6A F1 58 90 00 9A 03

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:

帧头	发送数据包内容					帧尾
	模块地址	长度	命令	发送数据	校验	
0x02	0x00,0x00	0x04	0x05	A...	0x09+A...	0x03

A 所在数据域可以存在多字节, 数据意义为下表:

定义	1 字节								2 字节	3 字节
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	十六进制	
关闭天线								0	无	
打开天线								1		
射频基站全面复位							1			
关闭 CPU 射频卡执行复位指令 同时执行快速复位						1				

打开 CPU 射频卡执行复位指令 同时执行快速复位						0				
射频基站快速复位				1						
设置射频场发射功率与接收灵敏度 (掉电记忆)						1				接收灵敏度 (0x1f-0x5f) 发射功率 (0x1f-0xff)

正确返回数据序列:

帧头	正确返回数据包内容							帧尾
	模块地址	插入	长度	命令	执行结果	返回数据	校验	
0x02	0x00,0x00	0x10	0x03	0x05	0x00	空	0x08	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:

帧头	错误返回数据包内容							帧尾
	模块地址	插入	长度	命令	执行结果	返回数据	校验	
0x02	0x00,0x00	0x10	0x03	0x05	非零	空	0"X"XX	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

附录 1: 指令汇总表

序号	指令	描述	执行正确返回	执行错误返回
1	0x3A	设置模块工作在 ISO14443 TYPE A 状态	0x00	非 0
2	0x46	Mifare one/Ultralight 寻卡	0x00	非 0

序号	指令	描述	执行正确返回	执行错误返回
3	0x33	Ultralight 选卡	0x00	非 0
4	0x4B	Ultralight 读块	0x00	非 0
5	0x35	Ultralight 写块	0x00	非 0
6	0x47	Mifare one 防冲突	0x00	非 0
7	0x48	Mifare one 选卡	0x00	非 0
8	0x4A	Mifare one 密钥验证	0x00	非 0
9	0x4B	Mifare one 读块	0x00	非 0
10	0x4C	Mifare one 写块	0x00	非 0
11	0x4D	Mifare one 初始化钱包	0x00	非 0
12	0x4E	Mifare one 读钱包	0x00	非 0
13	0x50	Mifare one 钱包充值	0x00	非 0
14	0x4F	Mifare one 钱包扣款	0x00	非 0
15	0x51	Mifare one 钱包备份步骤 1	0x00	非 0
16	0x52	Mifare one 钱包备份步骤 2	0x00	非 0
17	0x29	Mifare one/Ultralight 卡休眠	0x00	非 0
18	0x53	ISO14443 TYPE A CPU 卡复位	0x00	非 0
19	0x54	ISO14443 TYPE A CPU 卡发送 COS 指令	0x00	非 0
20	0x36	设置 PSAM 卡通讯波特率	0x00	非 0
21	0x19 (0X37)	PSAM 卡复位	0x00	非 0
22	0x1A (0X38)	向 PSAM 卡发送 COS 指令	0x00	非 0
23	0x15	设置波特率	0x00	非 0
24	0x05	设置模块天线状态	0x00	非 0
25	0x6A	控制 LED 引脚状态	0x00	非 0
26	0x1D	设置蜂鸣时间	0x00	非 0
27	0x12	设置射频基站与卡通信指令等待时间	0x00	非 0
28	0x04	读卡器休眠	0x00	非 0
29	0x17	获得读卡器序列号	0x00	非 0
30	0x16	获得读卡器版本号	0x00	非 0
31	0x14	获得读卡器设备标识号	0x00	非 0
32	0x13	设置读卡器设备标识号	0x00	非 0
33	0x2a	下载密钥到读卡器存储 (目前不支持, 可定制)	0x00	非 0

序号	指令	描述	执行正确返回	执行错误返回
34	0x49	使用已下载的密钥认证卡片 (目前不支持, 可定制)	0x00	非 0