IC 读卡器

DLL 函数库参考手册

(ISO14443 TypeA)
(ISO14443 TypeB)
(ISO15693)
(CPU)
(UltraLight)

2013-01-15

目 录

1. IC 读卡器 API 函数集

1.1 System Function

1.1.1 API_GetSysComm

功能	从注册表中获取串口 COM 信息
函数原型	int API_GetSysComm(unsigned char *Buffer)
描述	输入参数： 无 输出参数： Buffer[0]：系统注册表中已连接串口数 Buffer[1]：若 Buffer[0] > 0, Buffer[1]保存为串口号 Buffer[2]：若 Buffer[0] > 1, Buffer[2]保存为串口号 … Buffer[N]：若 Buffer[0] > N-1, Buffer[2]保存为串口号
返回值	成功返回 0，否则返回 1

1.1.2 API_OpenComm

功能	打开指定串口，并设置波特率
函数原型	HANDLE API_OpenComm(int nCom, int nBaudrate)
描述	输入参数： nCom：指定的串口名 nBaudrate：波特率（9600，19200,38400,57600,115200） 输出参数： 无
返回值	成功返回串口句柄，否则为 0

1.1.3 API_CloseComm

功能	关闭指定串口
函数原型	BOOL API_CloseComm(HANDLE commHandle)
描述	输入参数： commHandle：指定串口句柄 输出参数： 无
返回值	成功返回 TRUE，否则为 FALSE

1.1.4 API_SetDeviceAddress

功能	设置读卡器地址
函数原型	int API_SetDeviceAddress(HANDLE commHandle, int DeviceAddress, unsigned char NewAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 NewAddress：新地址，范围 0~255 输出参数： Buffer 如果成功，Buffer[0]为设置后的新地址 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.1.5 API_SetBaudrate

功能	设置读卡器通讯波特率
函数原型	int API_SetBaudrate(HANDLE commHandle, int DeviceAddress, unsigned char NewBaud, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 NewBaud：新的波特率代号 0x00 – 9600 bps 0x01 – 19200 bps 0x02 – 38400 bps 0x03 – 57600 bps 0x04 – 115200 bps > 0x04—9600 bps 输出参数： Buffer 如果成功，Buffer[0]为设置后的新波特率代号 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.1.6 API_SetSerNum

功能	设置读卡器序列号
函数原型	int API_SetSerNum(HANDLE commHandle, int DeviceAddress, unsigned char *NewValue, unsigned char *Buffer)
描述	输入参数：

	commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 NewValue：新的序列号（8byte） 输出参数： Buffer 如果成功，Buffer[0]=0x80 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.1.7 API_GetSerNum

功能	获得读卡器序列号
函数原型	int API_GetSerNum(HANDLE commHandle, int DeviceAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 输出参数： Buffer 如果成功，Buffer[0]为读卡器地址，Buffer[1-8]为序列号 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.1.8 API_GetVersionNum

功能	获得读卡器版本号
函数原型	int API_GetVersionNum(HANDLE commHandle, int DeviceAddress, char *VersionNum)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 输出参数： VersionNum 如果成功，VersionNum[0-7]为版本号 如果失败，VersionNum[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.1.9 API_ControlLED

功能	控制读卡器 LED
函数原型	int API_ControlLED(HANDLE commHandle, int DeviceAddress, unsigned

	char freq, unsigned char duration, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 freq：在一次循环中灯亮的周期数（一个周期为 20ms） duration：LED 状态循环的次数(一个循环一秒） 输出参数： Buffer 如果成功，Buffer[0]=0x80 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.1.10 API_ControlBuzzer

功能	控制读卡器蜂鸣器
函数原型	int API_ControlBuzzer(HANDLE commHandle, int DeviceAddress, unsigned char freq, unsigned char duration, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0~255 freq：在一次循环中，蜂鸣器鸣叫的周期数(一个周期为 100ms) duration：蜂鸣器状态循环的次数(一个循环一秒) 输出参数： Buffer 如果成功，Buffer[0]=0x80 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0，否则为非 0, 具体含义见附表。

1.2 ISO14443 Type-A Function

1.2.1 API_MF_Request

功能	寻卡操作
函数原型	int API_MF_Request(HANDLE commHandle, int DeviceAddress, unsigned char inf_mode, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF mode：寻卡模式 0x26：Idle 模式（一次只对一张卡操作） 0x52：All 模式（一次可对多张卡操作） 输出参数： Buffer 如果成功，Buffer[0-1]为两字节的卡类型 如果失败，Buffer[0]返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.2.2 API_MF_Anticoll

功能	卡片防冲突操作
函数原型	int API_MF_Anticoll(HANDLE commHandle, int DeviceAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF 输出参数： Buffer[0]：如果成功，返回数据长度，否则返回错误状态，具体含义见附表 Buffer[1]：单卡多卡标志. 0x00 – 检测到一张卡. 0x01 – 检测到多张卡. Buffer[2-N]：返回卡片 UID（4Byte or 7Byte）
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.2.3 API_MF_Select

功能	选卡操作，若有多张卡则选择其中一张
函数原型	int API_MF_Select(HANDLE commHandle, int DeviceAddress, unsigned char UIDLen, unsigned char *Uid, unsigned char *Buffer)

描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF UIDLen：UID 长度 Uid：卡芯片号（4byte or 7byte） 输出参数： Buffer[0]：如果成功，返回数据长度，否则返回错误状态，具体含义见附表 Buffer[1-N]：返回卡片 UID（4byte or 7byte）
返回值	成功返回 0 否则为非 0, 具体含义见附表。

1.2.4 API_MF_Halt

功能	使读卡器休眠操作
函数原型	int API_MF_Halt(HANDLE commHandle, int DeviceAddress)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF 输出参数： 无
返回值	成功返回 0 否则为非 0, 具体含义见附表。

1.2.5 API_MF_Read

功能	集成寻卡，防冲突，选卡，验证密码，读卡等操作，一个命令完成读卡操作
函数原型	int API_MF_Read(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Blk_add, unsigned char Num_blk, unsigned char *Key, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，要通讯的设备地址号，范围 0x00~0xFF Mode：读取模式控制 0x00：Idle 模式 + Key A 0x01：All 模式 + Key A 0x02：Idle 模式 + Key B 0x03：All 模式 + Key B Blk_add：要读的块的起点地址, 范围 0x00~0xFE Num_blk：要读的块数长度值, 范围从 1~4（m1 卡） Key[0-5]：6 个字节的密钥 输出参数： Buffer： 如果成功，Buffer[0]：返回数据长度

	Buffer[1-4]：4 字节卡序列号（从低到高） Buffer[5-N]：卡片数据 如果失败，Buffer[0]：返回状态，具体含义见附表。 注：只能读取本扇区的块，因为每个扇区有自己独立的密钥
返回值	成功返回 0 否则为非 0，具体含义见附表

1.2.6 API_MF_Write

功能	集成寻卡，防冲突，选卡，验证密码，写卡等操作，一个命令完成写卡操作
函数原型	int API_MF_Write(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Blk_add, unsigned char Num_blk, unsigned char *Key, unsigned char *Senddata, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，要通讯的设备地址号，范围 0x00~0xFF Mode：读取模式控制 0x00：Idle 模式 + Key A 0x01：All 模式 + Key A 0x02：Idle 模式 + Key B 0x03：All 模式 + Key B Blk_add：要读的块的起点地址，范围 0x00~0xFE Num_blk：要读的块数长度值，范围从 1~4（m1 卡） Key[0-5]：6 个字节的密钥 Senddata[0-M]：要写的数据（16 *Num_blk byte） 输出参数： Buffer： 如果成功，Buffer[0]：返回数据长度 Buffer[1-4]：4 字节卡序列号（从低到高） 如果失败，Buffer[0]：返回状态，具体含义见附表。 注：只能写入本扇区的块，因为每个扇区有自己独立的密钥
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.2.7 API_MF_InitVal

功能	电子钱包初始化
函数原型	int API_MF_InitVal(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Sec_num, unsigned char *Key, unsigned char *Value, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄

	DeviceAddress：设备地址，要通讯的设备地址号，范围 0x00~0xFF Mode：初始化模式 Sec_num：初始化块地址，范围 0x00~0xFF Key[0-5]：6 个字节的密钥 Value[0-3]：初始值 输出参数： Buffer： 如果成功，Buffer[0]:返回数据长度 Buffer[1-N]:返回数据 如果失败，Buffer[0]：返回状态，具体含义见附表。
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.2.8 API_MF_Dec

功能	电子钱包减值
函数原型	int API_MF_Dec(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Sec_num, unsigned char *Key, unsigned char *Value, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，要通讯的设备地址号，范围 0x00~0xFF Mode：减值模式 Sec_num：减值块地址，范围 0x00~0xFF Key[0-5]：6 个字节的密钥 Value[0-3]：待减的值 输出参数： Buffer： 如果成功，Buffer[0]:返回数据长度 Buffer[1-N]:返回数据 如果失败，Buffer[0]：返回状态，具体含义见附表。
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.2.9 API_MF_Inc

功能	电子钱包增值
函数原型	int API_MF_Inc(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Sec_num, unsigned char *Key, unsigned char *Value, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，要通讯的设备地址号，范围 0x00~0xFF Mode：增值模式

	Sec_num：增值块地址，范围 0x00~0xFF Key[0-5]：6 个字节的密钥 Value[0-3]：待增的值 输出参数： Buffer： 如果成功，Buffer[0]:返回数据长度 Buffer[1-N]:返回数据 如果失败，Buffer[0]：返回状态，具体含义见附表。
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.3 ISO14443 Type-B Function

1.3.1 API_Request_B

功能	ISO14443 TypeB 卡寻卡操作
函数原型	int API_Request_B(HANDLE commHandle, int DeviceAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：输出数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.3.2 API_Anticoll_B

功能	ISO14443 TypeB 卡防冲突操作
函数原型	int API_Anticoll_B(HANDLE commHandle, int DeviceAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：输出数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.3.3 API_Attrib_B

功能	ISO14443 TypeB 卡 Attrib 操作
函数原型	int API_Attrib_B(HANDLE commHandle, int DeviceAddress, unsigned char *SerialNum, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF SerialNum[0-3]：卡序列号（4byte）

	输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.3.4 API_RESET_B

功能	ISO14443 TypeB 卡 Reset 操作
函数原型	int API_RESET_B(HANDLE commHandle, int DeviceAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：输出数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.3.5 API_TransferCMD_B

功能	ISO14443 TypeB 数据传输操作
函数原型	int API_TransferCMD_B(HANDLE commHandle, int DeviceAddress, unsigned char CmdSize, unsigned char *Cmd, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF CmdSize：输入数据的长度 Cmd[0-M]：待输入数据（CmdSize byte） 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：输出数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.4 CPU Card Function

1.4.1 API_MF_PowerOn

功能	CPU 卡上电操作
函数原型	int API_MF_PowerOn(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Cmd, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Mode：CPU 卡上电模式 Idle：0x26 All：0x53 Cmd：上电命令参数，固定为 0x00 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：卡号数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.4.2 API_MF_TransferCMD

功能	CPU 卡数据传输操作
函数原型	int API_MF_TransferCMD(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Cmdlength, unsigned char *Cmd, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Mode：CPU 卡数据传输模式，Mode=0x00 Cmdlength：待传输的数据长度 Cmd[0-M]：待传输的数据 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：返回的数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.4.3 API_MF_RST_Antenna

功能	CPU 卡数据传输操作
函数原型	int API_MF_RST_Antenna(HANDLE commHandle, int DeviceAddress, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5 UltraLight Card Function

1.5.1 API_MF_GET_SNR

功能	UltraLight 寻卡操作命令
函数原型	int API_MF_GET_SNR(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Cmd, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Mode：寻卡模式 Idle：0x26 All：0x53 Cmd：寻卡参数数据，Cmd=0x00 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：卡序列号数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.2 API_MF_Halt

功能	读卡器休眠操作
函数原型	int API_MF_Halt(HANDLE commHandle, int DeviceAddress)
描述	输入参数： commHandle：串口句柄

	DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF 输出参数： 无
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.5.3 API_MF_Read

功能	UltraLight 卡读页操作
函数原型	int API_MF_Read(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Blk_add, unsigned char Num_blk, unsigned char *Key, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF Mode：读页模式，Mode=0x00 Blk_add：页号（0x00~0x0F） Num_blk：页数，Num_blk=0x01 Key[0-5]：Key=NULL 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-3]：卡片页数据（4byte）
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.5.4 API_MF_Write

功能	UltraLight 卡写页操作
函数原型	int API_MF_Write(HANDLE commHandle, int DeviceAddress, unsigned char Mode, unsigned char Blk_add, unsigned char Num_blk, unsigned char *Key, unsigned char *Senddata, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，即要通讯的设备地址号，范围 0x00~0xFF Mode：读页模式，Mode=0x00 Blk_add：页号（0x00~0x0F） Num_blk：页数，Num_blk=0x01 Key[0-5]：Key=NULL Senddata[0-3]：待写的数据（4byte） 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0，具体含义见附表。

1.6 ISO15693 Function

1.5.1 API_ISO15693_Inventory

功能	ISO15693 寻卡操作命令
函数原型	int API_ISO15693_Inventory(HANDLE commHandle, int DviceAddress, unsigned char Flag, unsigned char Afi, unsigned char Datalen, const unsigned char *pData, unsigned char *pBuffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flag：flags 标志 Afi：AFI Datalen：发送数据长度 pData：发送数据 输出参数： pBuffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 pBuffer[1-N]：卡片数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.2 API_ISO15693_Read

功能	ISO15693 读卡操作命令
函数原型	int API_ISO15693_Read(HANDLE commHandle, int DeviceAddress, unsigned char Flag, unsigned char Blk_add, unsigned char Num_blk, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flag：flags 标志 Blk_add：读取块号 Num_blk：读取块数 uid：卡片 UID 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：读取卡片数据

返回值	成功返回 0 否则为非 0，具体含义见附表

1.5.3 API_ISO15693_Write

功能	ISO15693 写卡操作命令
函数原型	int API_ISO15693_Write(HANDLE commHandle, int DeviceAddress, unsigned char Flag, unsigned char Blk_add, unsigned char Num_blk, unsigned char *uid, unsigned char *data)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flag：flags 标志 Blk_add：要写的块号 Num_blk：写的块数 uid：卡片 UID data：要写的数据 输出参数： 无
返回值	成功返回 0 否则为非 0，具体含义见附表

1.5.4 API_ISO15693_Lock

功能	ISO15693 锁卡操作命令
函数原型	int API_ISO15693_Lock(HANDLE commHandle, int DeviceAddress, unsigned char Flag, unsigned char Addr_blk, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flag：flags 标志 Addr_blk：锁块号 uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0，具体含义见附表

1.5.5 API_ISO15693_StayQuiet

功能	ISO15693 将卡置为静止状态
函数原型	int API_ISO15693_StayQuiet(HANDLE commHandle, int DeviceAddress, unsigned char Flag, unsigned char *uid, unsigned char *Buffer)

描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flag：flags 标志 uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.6 API_ISO15693_Select

功能	ISO15693 选卡操作
函数原型	int API_ISO15693_Select(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.7 API_ISO15693_ResetToReady

功能	ISO15693 使卡进入 READY 状态
函数原型	int API_ISO15693_ResetToReady(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.8 API_ISO15693_WriteAFI

功能	ISO15693 写 AFI
函数原型	int API_ISO15693_WriteAFI(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char Afi, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 Afi：AFI uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.9 API_ISO15693_LockAFI

功能	ISO15693 锁 AFI
函数原型	int API_ISO15693_LockAFI(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.10 API_ISO15693_WriteDSFID

功能	ISO15693 写 DSFID
函数原型	int API_ISO15693_WriteDSFID(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char DSFID, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志

	DSFID：DSFID uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.11 API_ISO15693_LockDSFID

功能	ISO15693 锁 DSFID
函数原型	int API_ISO15693_LockDSFID(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 uid：卡片 UID 输出参数： Buffer[0]：如果成功，Buffer[0]=0x80 如果失败，返回错误状态，具体含义见附表
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.12 API_ISO15693_GetSysInfo

功能	ISO15693 获得卡片系统信息
函数原型	int API_ISO15693_GetSysInfo(HANDLE commHandle, int deviceAddress, unsigned char Flag, unsigned char *uid, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 uid：卡片 UID 输出参数： Buffer[0]：如果成功，返回信息长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：返回信息
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.13 API_ISO15693_GetMulSecurity

功能	ISO15693 获得卡片安全信息
函数原型	int API_ISO15693_GetMulSecurity(HANDLE commHandle, int DeviceAddress, unsigned char Flags, unsigned char BlkAddr, unsigned char

	BlkNum, const unsigned char *uid, unsigned char *pBuffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 BlkAddr：块号 BlkNum：块数 uid：卡片 UID 输出参数： Buffer[0]：如果成功，返回信息长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：返回信息
返回值	成功返回 0 否则为非 0, 具体含义见附表

1.5.14 API_ISO15693_TransCmd

功能	ISO15693 向卡片发任何数据和命令操作
函数原型	int API_ISO15693_TransCmd(HANDLE commHandle, int DeviceAddress, int CmdSize, unsigned char *Cmd, unsigned char *Buffer)
描述	输入参数： commHandle：串口句柄 DeviceAddress：设备地址，范围 0x00~0xFF Flags：flags 标志 CmdSize：命令长度 Cmd[0-M]：命令 uid：卡片 UID 输出参数： Buffer[0]：如果成功，返回数据长度 如果失败，返回错误状态，具体含义见附表 Buffer[1-N]：返回数据
返回值	成功返回 0 否则为非 0, 具体含义见附表

附录

API 函数返回码（return value）

Return	描述
0x00	表示命令执行成功
0x02	表示接收数据长度不匹配
0x03	表示串口发送失败
0x04	表示串口未接到任何数据
0x05	表示设备地址不匹配
0x07	表示校验和不正确
0x0A	表示输入参数有误，请参见不具体的函数说明

读写器状态返回码（S）

Status	描述
0x00	表示命令执行成功
0x01	表示命令操作失败（具体说明参见函数）
0x80	表示参数设置成功
0x81	表示参数设置失败
0x82	表示通讯超时
0x83	表示卡不存在
0x84	表示接收卡数据出错
0x85	表示输入参数或者输入命令格式错误
0x87	表示未知的错误
0x89	The parameter of the command or the Format of the command Erro
0x8A	在块初始化中出现错误
0x8B	在防冲突过程中得到错误的序列号
0x8C	密码认证没通过
0x8f	表示输入的指令代码不存在
0x90	表示卡不支持这个命令
0x91	表示命令格式有错误
0x92	表示在命令的 FLAG 参数中，不支持 OPTION 模式
0x93	表示要操作的 BLOCK 不存在
0x94	表示要操作的对象已经别锁定，不能进行修改
0x95	表示锁定操作不成功
0x96	表示写操作不成功
0x89	在 15693 卡中 flag 不正确