

密级：二级

深圳宇川智能系统 消费节能通用卡操作 API

用 户 手 册

修订历史				
版本	说明	作者	批准	生效日期
V1.0	基本使用说明(初稿)	胥 申 林		2011 年 07 月 25 日

目 录

4. API 函数说明.....	3
4.1 读写器 API 函数说明.....	3
4.1.1 rf_init.....	3
4.1.2 rf_exit.....	3
4.1.3 int rf_card.....	3
4.1.4 int rf_request.....	4
4.1.5 int rf_anticoll.....	4
4.1.6 int rf_select.....	4
4.1.7 int rf_load_key.....	4
4.1.8 int rf_load_key_hex.....	5
4.1.9 int rf_authentication.....	5
4.1.10 int rf_read.....	5
4.1.11 int rf_write.....	5
4.1.12 int rf_halt.....	6
4.1.13 int rf_beep.....	6
4.1.14 int rf_disp8.....	6
4.1.15 int rf_disp_m.....	6
4.1.16 int rf_settime.....	6
4.1.17 int rf_initval.....	6
4.1.18 int rf_increment.....	7
4.1.19 int rf_decrement.....	7
4.1.20 int rf_readval.....	7
4.2.1 HANDLE OpenComm.....	7
4.2.2 int CloseComm.....	7
4.2.3 int Check_Reader.....	7
4.2.4 int ReadCard_ID.....	7
4.2.5 int Query_Card_Type.....	8
4.2.6 int Init_SysCard12_NewCreat.....	8
4.2.7 int Init_SysCard12_NewReWrite.....	8
4.2.8 int WRT_UserCard_Term.....	8
4.2.9 int Make_UserPassword_NEW.....	9
4.2.10 int Change_Pos_UserType12_NEW.....	9
4.3 消费专用 API 函数说明.....	9
4.3.1 int Init_Pos_UserCard12.....	9
4.3.2 int Query_Pos_UserCard12.....	9
4.3.3 int WRT_Pos_UserCard_AddCount12.....	10
4.3.4 int RST_Pos_UserCard12.....	10
4.3.5 int Init_Pos_OPTCard12.....	10
4.3.6 int RST_Pos_OPTCard12.....	11

4.4 节能专用 API 函数说明.....	11
4.4.1 int QueryJsCard.....	11
4.4.2 int Init_Js_UserCard.....	11
4.4.3 int RSTJsUserCard.....	12
4.4.4 int WRT_Js_UserCard_AddCount.....	12
4.4.5 int Set_Js_DateTimeCard.....	12
4.4.6 int Set_Js_AddreCard.....	12
4.4.7 int Init_JS_OPTCard12.....	12
4.4.8 int MakeJSSubSysCard12.....	12
4.4.9 int RSTJSSubSysCard.....	13
4.4.10 int InitCollectCard.....	13
4.4.11 int RSTCollectCard.....	13
4.4.12 int ReadRecNum_NEW.....	13
4.4.13 int ReadAllRec_NEW.....	13
4.4.14 int ReadDetailedRec_NEW.....	14
4.4.15 int InitBlackCard.....	14
4.4.16 int WriteBlackCard.....	14
4.5 函数操作流程圖.....	15

4. API 函数说明

4.1 读写器 API 函数说明

4.1.1 rf_init(__int16 port, long baud);

功 能: 打开串口

参 数: __int16 port: 串口号 (例如串口1 port = 1)

long baud: 波特率

返 回: 串口句柄

4.1.2 rf_exit(HANDLE icdev);

功 能: 关闭串口

参 数: HANDLE icdev: 打开串口句柄

返 回: 0正确关闭, 非0关闭失败

4.1.3 int rf_card(HANDLE icdev, unsigned char _Mode, unsigned long *_Snr);

功 能: 寻卡, 能返回在工作区域内某张卡的序列号

参 数: HANDLE icdev: 通讯设备标识符

unsigned char _Mode: 寻卡模式

unsigned long *_Snr: 返回的卡序列号

返 回: 成功则返回 0

例: int st;

```
unsigned long *_Snr;  
st=rf_card(icdev,0,&snr);
```

注：选择IDLE模式，在对卡进行读写操作后，必须执行rf_halt()指令中止卡操作。只有当该卡离开并再次进入操作区时，读写器才能够再次对它进行操作。

4.1.4 int rf_request(HANDLE icdev, unsigned char _Mode, unsigned __int16 *TagType);

功 能：寻卡请求

参 数：HANDLE icdev: 通讯设备标识符

unsigned char _Mode: 寻卡模式

unsigned __int16 *TagType: 卡类型值，0x0004 为 M1 卡，0x0010 为 ML 卡

返 回：成功则返回 0

例：int st;

```
unsigned __int16 *TagType;  
st=rf_request(icdev, IDLE, tagtype);
```

4.1.5 int rf_anticoll(HANDLE icdev, unsigned char _Bcnt, unsigned long *_Snr);

功 能：防止卡冲突，返回卡的序列号

参 数：HANDLE icdev: 通讯设备标识符

unsigned char _Bcnt: 预选卡所用的位数，标准值为 0（不考虑序列号）

unsigned long *_Snr: 返回的卡序列号地址

返 回：成功则返回 0

例：int st;

```
unsigned long *_Snr;  
st=rf_anticoll(icdev,0,snr);
```

注：request 指令之后应立即调用 anticoll，除非卡的序列号已知。

4.1.6 int rf_select(HANDLE icdev,unsigned long _Snr, unsigned char *_Size);

功 能：从多个卡中选取一个给定序列号的卡

参 数：HANDLE icdev: 通讯设备标识符

unsigned long _Snr: 卡序列号

unsigned char *_Size: 指向返回的卡容量的数据

返 回：成功则返回 0

例：int st;

```
unsigned long _Snr =239474;  
unsigned char *_Size;  
st=rf_select(icdev,snr,size);
```

4.1.7 int rf_load_key(HANDLE icdev,unsigned char _Mode, unsigned char _SecNr, unsigned char *_NKey);

功 能：将密码装入读写模块 RAM 中

参 数：HANDLE icdev: 通讯设备标识符

unsigned char _Mode: 装入密码模式，同密码验证模式

unsigned char _SecNr: 扇区号（M1 卡：0~15； ML 卡：0）

unsigned char *_NKey: 写入读写器中的卡密码

返回：成功则返回 0

例：//key A and key B

```
unsigned char tk[6]={0xa0,0xa1,0xa2,0xa3,0xa4,0xa5};
/* 装入 1 扇区的 0 套 A 密码 */
if((rf_load_key(icdev,0,1,tk))!=0)
{
    printf("Load key error!");
    rf_exit()
    exit(1);
}
```

4.1.8 int rf_load_key_hex(HANDLE icdev,unsigned char _Mode,unsigned char _SecNr,char *_NKey);

功能：将密码装入读写模块 RAM 中

参数：HANDLE icdev：通讯设备标识符

unsigned char _Mode：装入密码模式，同密码验证模式

unsigned char _SecNr：扇区号（M1卡：0~15； ML卡：0）

unsigned char *_NKey：写入读写器中的卡密码

返回：成功则返回 0

4.1.9 int rf_authentication(HANDLE icdev,unsigned char _Mode,unsigned char _SecNr);

功能：验证某一扇区密码

参数：HANDLE icdev：通讯设备标识符

unsigned char _Mode：密码验证模式

unsigned char _SecNr：要验证密码的扇区号（0~15）

返回：成功则返回 0

例：int st;

```
st=rf_authentication(icdev,0,0);
```

注：卡上每个扇区有 A 密码和 B 密码，可根据实际需要确定是否使用 B 密码，这由该扇区的存取控制位来决定。此外，读写器中可以存放三套密码，可用 rf_load_key() 来分别装入，只有装入后才能使用验证密码函数验证。

4.1.10 int rf_read(HANDLE icdev,unsigned char _Adr,unsigned char *_Data);

功能：读取卡中数据

对于 M1 卡，一次读一个块的数据，为 16 个字节；

参数：HANDLE icdev：通讯设备标识符

unsigned char _Adr：M1 卡块地址（0~63）

unsigned char *_Data：读出数据

返回：成功则返回 0

例：int st;

```
unsigned char *_Data [16];
```

```
st=rf_read(icdev,4,data); //读 M1 卡块 4 的数据
```

4.1.11 int rf_write(HANDLE icdev,unsigned char _Adr,unsigned char *_Data);

功 能：向卡中写入数据

对于 M1 卡，一次必须写一个块，为 16 个字节；

参 数：HANDLE icdev：通讯设备标识符

unsigned char _Adr：M1 卡块地址（1~63）；

unsigned char *_Data：要写入的数据

返 回：成功则返回 0

例：int st；

```
unsigned char *_Data =1234567890123456?
```

```
st=rf_write(icdev,4,data);
```

4.1.12 int rf_halt(HANDLE icdev);

功 能：中止卡操作

参 数：HANDLE icdev：通讯设备标识符

返 回：成功则返回 0

例：st=rf_halt(icdev);

4.1.13 int rf_beep(HANDLE icdev,unsigned short _Msec)

功能：蜂鸣

参数：HANDLE icdev : OpenComm() 返回的设备号

unsigned short _Msec : 蜂鸣时间的长短，单位是 10 毫秒

返回：成功则为 0，小于 0 见错误代码

4.1.14 int rf_disp8(HANDLE icdev,__int16 len,unsigned char* digit)

功能：使读写器的数码管显示数字

参数：HANDLE icdev : OpenComm() 返回的设备号

__int16 len : 显示位数，其值为 1-8，表示设置他总共使用几位数码管

unsigned char* digit : 显示数据（BCD 码，最高位控制小数点）

返回：成功则为 0，小于 0 见错误代码

4.1.15 int rf_disp_m(HANDLE icdev,unsigned char mode)

功能：设置读写器显示时间还是显示日期(读写器控制，仅适用于新版读卡器)

参数：HANDLE icdev : OpenComm() 返回的设备号

unsigned char mode : 显示内容的类型，0 表示显示日期 1 表示显示时间

返回：成功则为 0，小于 0 见错误代码

注：调用此函数前必须先调用函数 rf_ctl_m 将显示设为由计算机控制

4.1.16 int rf_settime(HANDLE icdev,unsigned char *time)

功能：设置读写器时间

参数：HANDLE icdev：OpenComm() 返回的设备号

unsigned char *time : 时间数组（7 个字节，年、周、月、日、时、分、秒）。

返回：成功则为 0，小于 0 见错误代码

4.1.17 int rf_initval(HANDLE icdev,unsigned char _Adr,unsigned long _Value);

功能：初始化块值

参数: icdev : 通讯设备标识符
_Adr : 块地址
_Value: 初始值
返回: 成功则返回 0

4.1.18 int rf_increment(HANDLE icdev, unsigned char block, unsigned long Value)

功能: 块加值

参数: icdev : 通讯设备标识符
Value : 要增加的值
Block : 块地址
返回: 成功则返回 0

4.1.19 int rf_decrement(HANDLE icdev, unsigned char block, unsigned long Value)

功能: 块减值

参数: icdev : 通讯设备标识符
Value : 要减少的值
Block : 块地址
返回: 成功则返回 0

4.1.20 int rf_readval(HANDLE icdev, unsigned char _Adr, unsigned long *_Value)

参数: icdev: 通讯设备标识符
_Adr: 块地址
_Value: 读出值的地址
返回: 成功则返回 0

4.2 通用 API 函数

4.2.1 HANDLE OpenComm (int CommPort)

功能: 打开串口

参数: int CommPort: 串口标志(取值如下: 0 — Com1, 1 — Com2 最大到 COM20)

返回: 设备句柄, 类似于 4 字节整数, 大于 0 为串口设备句柄, 小于 0 表示打开串口错误。

4.2.2 int CloseComm (HANDLE icdev)

功能: 关闭串口

参数: HANDLE icdev: OpenComm() 返回的设备句柄
返回: 成功则返回 0

4.2.3 int Check_Reader(HANDLE icdev)

功能: 测试发卡器

参数: HANDLE icdev: OpenComm() 返回的设备号
返回: 成功则返回 0, 小于 0 见错误代码

4.2.4 int ReadCard_ID(HANDLE icdev, unsigned char *TagType, unsigned long *Snr)

功能: 读卡片物理 ID

参数: HANDLE icdev: OpenComm() 返回的设备句柄

unsigned char *TagType : 卡类型: 04 为 MF1 卡

unsigned long *Snr : 卡片的物理 ID

返回: 成功则返回 0, 小于 0 见错误代码

4.2.5 int Query_Card_Type(HANDLE icdev, LPINT SysType, LPINT CardType, unsigned long *CardSerno, unsigned int *OPT_Num, int WaitTime, unsigned char *user_code_new)

功能: 更新系统卡 (存在其他系统的情况下, 添加其他系统的信息)

参数: HANDLE icdev : OpenComm() 返回的设备句柄

LPINT SysType : 系统的类型 0 系统或工具卡、1 门禁、2 消费、3 节水, 如果是两种以上的系统卡, 显示 12, 123, 23, 13

LPINT CardType : 卡片的类型, 0 用户卡、2 系统卡、1 操作员卡、3 初始化卡、4 白卡、5 节水设置卡、6 采集卡、7 加密卡、8 查询卡、9 机号设置卡、10 时间设置卡

unsigned long * CardSern : 卡片的物理 ID

unsigned int *OPT_Num : 营业员卡编号

int WaitTime : 等卡时间, 单位为 ms

unsigned char *user_code_new : 配置参数组 (9 字节详见[附录 1](#))

返回: 成功则返回 0, 小于 0 见错误代码

4.2.6 int Init_SysCard12_NewCreat(HANDLE icdev, LPCSTR UserPassword, int SysType, int UseSector, LPSTR CommPassword)

功能: 初始化系统卡 (无其他系统的情况下)

参数: HANDLE icdev : OpenComm() 返回的设备句柄

LPCSTR UserPassword: 用户密码 (8 位 ASCII 字符, 此密码用户必须保存数据库中)

int SysType : 系统类别, 1 为门禁, 2 为售饭

int UseSector : 用户卡使用扇区。

LPSTR CommPassword : 返回此系统的通讯密码。

注: 此密码用于通讯握手中, 只要用此系统卡的机子, 必须用此密码才能通讯成功。

返回: 成功则返回 0, 小于 0 见错误代码

注意: 当初始化系统卡成功时, 在调用此过程的执行文件的当前目录中会生一个 LICENCECARD.DAT 的文件, 以后的发卡和通讯函数均要用到此文件, 用户需把此文件备份, 不能丢失。此文件不能修改。

4.2.7 int Init_SysCard12_NewReWrite (HANDLE icdev, int SysType, int UseSector)

功能: 更新系统卡 (存在其他系统的情况下, 添加其他系统的信息)

参数: HANDLE icdev: OpenComm() 返回的设备句柄

int SysType : 系统类别, 1 为门禁, 2 为售饭

int UseSector: 用户卡使用扇区。

返回: 成功则返回 0, 小于 0 见错误代码

注意: 当初始化系统卡成功时, 在调用此过程的执行文件的当前目录必须有一个 LICENCECARD.DAT 的文件, 他会将其他信息, 用户需把此文件备份, 不能丢失。此文件不能修改。

4.2.8 int WRT_UserCard_Term(HANDLE icdev, int Systype, unsigned long useterm, unsigned int CardSerno, int WaitTime)

功能：修改用户卡有效期

参数：HANDLE icdev : OpenComm() 返回的设备号

int Systype : 用户卡类型，1 为门禁、2 为消费、3 为节水

unsigned long useterm : 卡片有效期（十进制，例如 2009 年 1 月 1 号 20090101）

unsigned int CardSerno: 卡片物理 ID

int WaitTime : 等卡时间，单位为 ms

返回：成功则返回 0，小于 0 见错误代码

4.2.9 int Make_UserPassword_NEW(HANDLE icdev, int Systype)

功能：恢复用户卡密码（默认为 0000）

参数：HANDLE icdev : OpenComm() 返回的设备号

int Systype : 用户卡类型，1 为门禁、2 为消费、3 为节水

返回：成功则返回 0，小于 0 见错误代码

4.2.10 int Change_Pos_UserType12_NEW(HANDLE icdev, int * Old_UserType, int UserType ,int Systype)

功能：修改用户卡类

参数：HANDLE icdev : OpenComm() 返回的设备号

int *Old_UserType : 原用户卡卡类（1-8 类）

int UserType : 待修改的卡类（1-8 类）

int Systype : 用户卡类型，1 为门禁、2 为消费、3 为节水

返回：成功则返回 0，小于 0 见错误代码

4.3 消费专用 API 函数说明

4.3.1 int Init_Pos_UserCard12 (HANDLE icdev, int Serno, LPCSTR Cardno, int UserType, int WaitTime, unsigned long * CardSerno, unsigned long use_term , unsigned char * user_code_new)

功能：初始化消费用户卡

参数：HANDLE icdev : OpenComm() 返回的设备号

int Serno : 用户卡流水号，不可重复（1—100000）

LPCSTR Cardno : 用户卡号（五位 ASCII 字符）

int UserType : 用户卡类型（1-8 类）

int WaitTime : 等卡时间，单位为 ms

unsigned long * CardSerno: 返回用户卡固有 ID 号

int use_term : 卡使用期限

unsigned char * user_code_new: 配置参数组，前 9 个字节参考密码说明（[附录 1](#)），第 10 字节开始动态添加考勤参数，依次为人员编号长度（1B）+ 人员编号（长度 B<GB2312-80>）+ 人员姓名长度（1B）+ 人员姓名（长度 B<ASCII>）+ 人员类型长度（1B）+ 人员类型（长度 B<GB2312-80>）

返回：成功则返回 0，小于 0 见错误代码

4.3.2 int Query_Pos_UserCard12(HANDLE icdev, LPINT CardType, LPINT OPT_Num, LPINT Serno, LPSTR Cardno, LPINT UserType, unsigned long * CardSerno, LPINT ChkSum1, LPINT Value1, LPINT LastPay1,

LPINT Count1, LPINT Consume_Add1, LPINT ChkSum2, LPINT Value2, LPINT LastPay2, LPINT Count2, LPINT Consume_Add2, unsigned long *use_term, LPINT AddCount, int WaitTime, unsigned char *user_code_new)

功能：消费机用户卡块查询

参数：HANDLE icdev : OpenComm() 返回的设备号
LPINT CardType : 卡片的类型，0 用户卡、2 系统卡、1 操作员卡、3 初始化卡、4 白卡、5 节水设置卡、6 采集卡、7 加密卡、8 查询卡、9 机号设置卡、10 时间设置卡
LPINT OPT_Num : 营业员编号
LPINT Serno : 返回用户卡流水号
LPSTR Cardno : 返回用户卡卡编号
LPINT UserType : 返回用户类型
unsigned long * CardSerno: 返回用户卡固有 ID 号
LPINT ChkSum1 : 返回用户卡使用块的校验状态，0 为正确，1 为错误。
LPINT Value1 : 返回用户卡使用块的金额，为实际金额 * 100
LPINT LastPay1 : 返回用户卡使用块的余额，为实际余额 * 100
LPINT Count 1 : 返回用户卡使用块的消费次数
LPINT Consume_Add1: 返回用户卡使用块的消费累加额，为实际余额 * 100
LPINT ChkSum 2 : 返回用户卡备份块的校验状态，0 为正确，1 为错误
LPINT Value 2 : 返回用户卡备份块的金额，为实际金额 * 100
LPINT LastPay 2 : 返回用户卡备份块的余额，为实际余额 * 100
LPINT Count 2 : 返回用户卡备份块的消费次数
LPINT Consume_Add2: 返回用户卡备份块的消费次数
LPINT use_term : 返回用户卡的使用期限
LPINT AddCount : 返回充值次数增加 1
int WaitTime : 等卡时间，单位为 ms
unsigned char * user_code_new: 配置参数组（9 字节详细见[附录 1](#)）
返回：成功则为 0，小于 0 见错误代码

4.3.3 int WRT_Pos_UserCard_AddCount12 (HANDLE icdev, int Value, unsigned int CardSerno, int WaitTime)

功能：用户卡写值（充值次数增加 1）

参数：HANDLE icdev : OpenComm() 返回的设备号
int Value : 需要写入的金额，为实际金额 * 100
unsigned int CardSerno: 需要写值的用户卡固有 ID 号
int WaitTime : 等卡时间，单位为 ms

返回：成功则返回 0，小于 0 见错误代码

4.3.4 int RST_Pos_UserCard12 (HANDLE icdev, unsigned int CardSerno, int WaitTime, unsigned char *user_code_new)

功能：回收消费机用户卡

参数：HANDLE icdev : OpenComm() 返回的设备号
int CardSerno : 要回收的用户卡固有 ID 号
int WaitTime : 等卡时间，单位为 ms
unsigned char *user_code_new: 配置参数组（9 字节详细见[附录 1](#)）

返回：成功则返回 0，小于 0 见错误代码

4.3.5 int Init_Pos_OPTCard12 (HANDLE icdev, int OPT_Num, int WaitTime, unsigned long * CardSerno, unsigned char *user_code_new)

功能：初始化消费机操作员卡

参数：HANDLE icdev : OpenComm() 返回的设备号

int Opt_Num : 操作员编号, (1 - 255)

int WaitTime : 等卡时间, 单位为 ms

unsigned long * CardSerno: 返回操作员卡固有 ID 号

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回：成功则返回0, 小于0见错误代码

4.3.6 int RST_Pos_OPTCard12 (HANDLE icdev, unsigned int CardSerno, int WaitTime, unsigned char *user_code_new)

功能：回收消费机操作员卡

参数：HANDLE icdev : OpenComm() 返回的设备号

unsigned int CardSerno: 要回收的操作员卡固有 ID 号

int WaitTime : 等卡时间, 单位为 ms

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回：成功则返回 0, 小于 0 见错误代码

4.4 节能专用 API 函数说明

4.4.1 int QueryJsCard(HANDLE icdev, LPINT CardType, LPINT OPT_Num, LPINT Serno, LPSTR Cardno, unsigned long * CardSerno, LPINT Value, LPINT Count, LPINT UserType, int WaitTime, unsigned char * user_code_new)

功能：读节能用户卡

参数：HANDLE icdev : OpenComm() 返回的设备号

LPINT CardType : 卡类型 (0 - 用户卡, 1 - 操作员卡, 2 - 系统卡, 4 - 白卡)

LPINT OPT_Num : 操作员编号

LPINT Serno : 用户卡流水号

LPSTR Cardno : 用户卡编号

unsigned long * CardSerno : 用户卡物理 ID 号

LPINT Value : 用户卡余额

LPINT Count : 用户卡使用次数

LPINT UserType : 用户卡卡类 (1-8 类)

int WaitTime : 等卡时间, 单位为 ms

unsigned char * user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回：成功返回 0, 其它错误请参见错误代码表

4.4.2 int Init_Js_UserCard (HANDLE icdev, int Serno, LPCSTR Cardno, int UserType, unsigned long * CardSerno, unsigned char * user_code_new)

功能：发节能用户卡

参数：HANDLE icdev : OpenComm() 返回的设备号

int Serno : 用户卡流水号
LPSTR Cardno : 用户卡编号
int UserType : 用户卡卡类 (1-8 类)
unsigned long * CardSerno : 用户卡物理 ID 号
unsigned char * user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回: 成功返回 0, 其它错误请参见错误代码表

4.4.3 int RSTJsUserCard (HANDLE icdev, unsigned int CardSerno, unsigned char *user_code_new)

功能: 回收节能用户卡

参数: HANDLE icdev : OpenComm() 返回的设备号

unsigned int CardSerno : 要回收的用户卡固有 ID 号

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回: 成功则返回 0, 小于 0 见错误代码

4.4.4 int WRT_Js_UserCard_AddCount (HANDLE icdev, int Blance, char *ChargeDateTime, unsigned long CardSerno)

功能: 修改节水系统用户卡上的金额并将充值次数加 1

参数: HANDLE icdev : OpenComm() 返回的设备号

int Blance : 卡金额

char *ChargeDateTime: 充值时间 YYMMDDHHMMSS 格式

unsigned long CardSerno : 卡物理 ID 号

返回: 成功则返回 0, 小于 0 见错误代码

4.4.5 int Set_Js_DateTimeCard (HANDLE icdev, char *js_date, char *js_time, unsigned char *user_code_new)

功能: 制一体化节水时间设置卡

参数: HANDLE icdev : 串口设备句柄

char *js_date : 设置的日期 YYMMDD 格式, 即只存放年月日

char *js_time : 设置的时间 HHMMSS 格式, 即只存放时分秒

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回: 成功返回 0, 其它错误请参见错误代码表

4.4.6 int Set_Js_AddreCard (HANDLE icdev, int Addr, unsigned char *user_code_new)

功能: 制一体化节水机号设置卡

参数: HANDLE icdev : 串口设备句柄

int Addr : 设置机号的起始地址

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录 1](#))

返回: 成功返回 0, 其它错误请参见错误代码表

4.4.7 int Init_JS_OPTCard12 (HANDLE icdev, int OPT_Num, int WaitTime, unsigned long * CardSerno, unsigned char *user_code_new)

功能: 初始化节能营业员卡

参数: HANDLE icdev : OpenComm() 返回的设备号

int Opt_Num : 营业员编号, (1 - 254)

int WaitTime : 等卡时间, 单位为 ms

unsigned long * CardSerno: 返回操作员卡物理 ID 号

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录1](#))

返回: 成功则返回0, 小于0见错误代码

4.4.8 int MakeJSSubSysCard12(HANDLE icdev, JS_FEE_RATE&fee_rate, JS_AdvanPara&AdvanPara , int Addr, unsigned char *user_code_new)

功能: 发节能参数设置卡

参数: HANDLE icdev : OpenComm() 返回的设备号

JS_FEE_RATE &fee_rate : 节能费率参数, 见附录一 JS_FEE_RATE 参数结构体

JS_AdvanPara & AdvanPara: 节能高级参数 , 见附录二 JS_AdvanPara 参数结构体

int Addr : 设备机号设置卡起始机号 (供节能设备设置机号用)

unsigned char *user_code_new: 配置参数组 (9 字节详细见[附录1](#))

返回: 成功则返回 0, 小于 0 见错误代码

4.4.9 int RSTJSSubSysCard(HANDLE icdev , unsigned char *user_code_new)

功能: 回收节能参数设置卡

参数: HANDLE icdev : OpenComm() 返回的设备号

unsigned char *user_code_new

返回: 成功则返回 0, 小于 0 见错误代码

4.4.10 int InitCollectCard(HANDLE icdev,int OperatorNo, int Mode, unsigned long * CardSerno , unsigned char * user_code_new)

功能: 发节能数据采集卡

参数: HANDLE icdev : OpenComm() 返回的设备号

int OperatorNo : 采集卡操作员编号

unsigned long CardSerno: 返回采集卡物理 ID 号

unsigned char * user_code_new: 配置参数组 (9 字节详细见[附录1](#))

返回: 成功则返回 0, 小于 0 见错误代码

4.4.11 int RSTCollectCard(HANDLE icdev,unsigned int CardSerno , unsigned char *user_code_new)

功能: 回收节能采集卡

参数: HANDLE icdev : OpenComm() 返回的设备号

unsigned int CardSerno: 采集卡物理 ID 号

unsigned char * user_code_new: 配置参数组 (9 字节详细见[附录1](#))

返回: 成功则返回 0, 小于 0 见错误代码

4.4.12 int ReadRecNum_NEW(HANDLE icdev, unsigned int *MODE, unsigned long *CardSerno, unsigned int *RecNum)

功能: 读采集卡基本信息

参数: HANDLE icdev: OpenComm() 返回的设备句柄

unsigned int *MODE 返回采集卡的模式, 1 为明细采集卡 , 0 为总额采集卡

unsigned long *CardSerno 返回采集卡卡 ID 号

unsigned int *RecNum 返回已采集记录条数 如果为明细采集卡则为 0

返回：成功则返回 0，小于 0 见错误代码

4.4.13 int ReadAllRec_NEW(HANDLE icdev, RecordAllData *RecAllData, unsigned long CardSerno)

功能：如果 ReadRecNum_NEW 返回采集卡模式为 0 可采用此函数采集各设备总额

参数：HANDLE icdev: OpenComm() 返回的设备句柄

RecordAllData *RecAllData 返回每台设备总额情况（注意结构体定义）

unsigned long CardSerno 采集卡卡 ID 号

返回：成功则返回 0，小于 0 见错误代码

4.4.14 int ReadDetailedRec_NEW(HANDLE icdev, RecordDetailedData *RecDetailedData, unsigned long CardSerno, unsigned long *All_Money, unsigned long *All_Times, unsigned long *All_Amount, unsigned long *Address)

功能：如果 ReadRecNum_NEW 返回采集卡模式为 1 可采用此函数采集设备明细

参数：HANDLE icdev: OpenComm() 返回的设备句柄

RecordDetailedData *RecAllData 返回每台设备消费明细（注意结构体定义）

unsigned long CardSerno 采集卡卡 ID 号

unsigned long *All_Money 返回此设备消费总金额

unsigned long *All_Times 返回此设备消费总使用次数

unsigned long *All_Amount 返回此设备消费总使用量（计时单位为 S，计量单位为 L）

unsigned long *Address 返回设备地址

返回：成功则返回 0，小于 0 见错误代码

4.4.15 int InitBlackCard(HANDLE icdev, unsigned char Block, unsigned long *CardSerno, unsigned char * user_code_new)

功能：初始化黑名单卡

参数：HANDLE icdev: OpenComm() 返回的设备句柄

unsigned char Block 黑名单块号（每块设置 4096 个黑名单成员）

unsigned long *CardSerno 返回黑名单卡卡 ID 号

unsigned char * user_code_new 配置参数组（9 字节详细见[附录 1](#)）

返回：成功则返回 0，小于 0 见错误代码

4.4.16 int WriteBlackCard(HANDLE icdev, unsigned long CardSerno, unsigned char *Black_Data)

功能：下黑名单数据

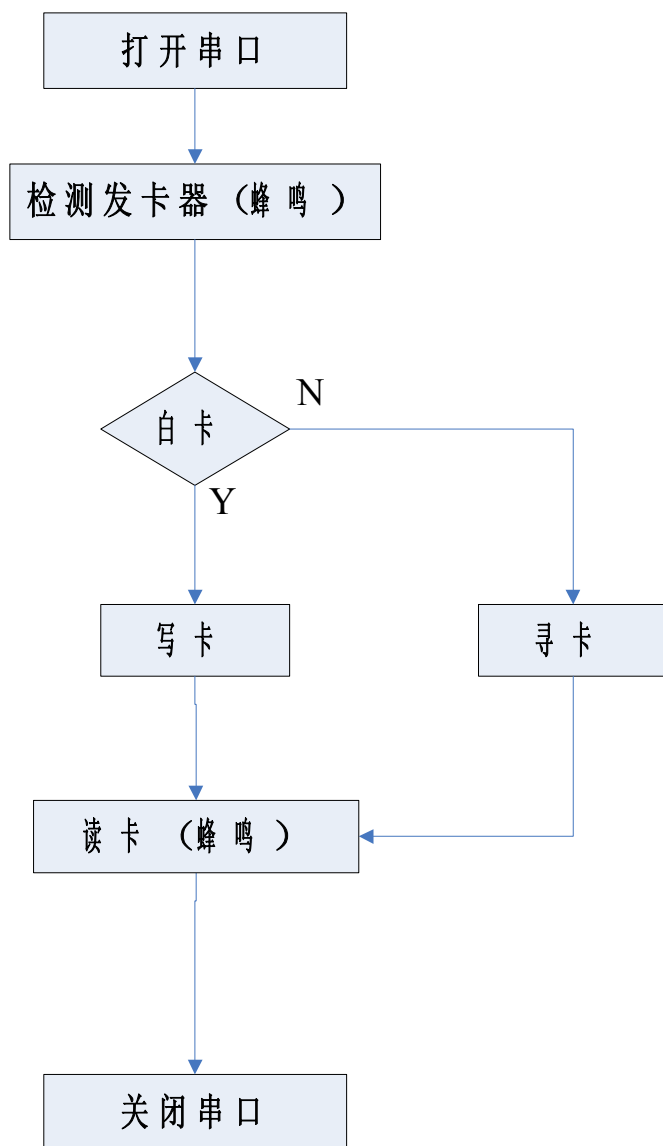
参数：HANDLE icdev: OpenComm() 返回的设备句柄

unsigned long CardSerno 核对黑名单卡 ID 号

unsigned char *Black_Data 黑名单数据组（每张卡设置 512 个字节，如果大于 4096 用户可以用多张黑名单卡在初始化黑名单卡时候配置他的块号来实现）

返回：成功则返回 0，小于 0 见错误代码

4.5 函数操作流程图



附录 1

传参的内容包括：

动态加密的密码（如果非动态加密传送 8 个 0），长度固定为 8 个字节。

第 9 字节表示版本方式：1 为非加密版本，2 为固定密码版本，3 为动态加密版本

后面字节在特定的函数有特定用途，详细见相关函数详细说明。

例如：`unsigned char buf[]={0,0,0,0,0,0,0,0,1}`；表示不加密