

# 13.56M射频卡读写器YW-60x系列

## 使用手册

(version 1.5)

ISO14443B

● 简体中文版

繁體中文版

English Edition

网址:	<a href="http://www.youwokeji.com.cn">http://www.youwokeji.com.cn</a>
EMAIL:	coodor@126.com
电话:	010-59395668
手机:	013671114914
联系人:	周先生

## 目录

1	概述.....	4
2	YW-605 系列型号.....	4
3	功能特点.....	5
4	技术指标.....	5
5	二次开发指南.....	6
	动态库及读写器相关函数.....	6
1.	读取库函数内部版本号.....	6
2.	DES加解密函数.....	6
3.	3DES加解密函数.....	7
4.	带向量的 3DES加解密函数.....	7
5.	初始化串口.....	8
6.	释放串口.....	8
7.	USB无驱读写器，初始化USB.....	9
8.	USB无驱读写器，释放USB.....	9
9.	修改读写器串口波特率.....	9
10.	设置设备标识.....	10
11.	查询设备标识.....	10
12.	读取读卡器内部版本号.....	10
13.	查询读写器产品序列号.....	10
14.	蜂鸣器控制函数.....	11
15.	LED指示灯控制.....	11
16.	设置LED显示器显示的内容.....	12
17.	设置天线的状态.....	13
18.	设置寻卡模式.....	13
	ISO14443B相关函数.....	14
19.	Type B 卡复位.....	14
20.	Type B 卡执行 C O S 命令.....	14
	AT88RF020 卡片操作函数.....	15
21.	AT88RF020 秘钥认证.....	15
22.	AT88RF020 读数据块.....	15
23.	AT88RF020 写数据块.....	16
24.	AT88RF020 锁数据块.....	16
25.	AT88RF020 计数函数.....	17
26.	AT88RF020 不选择当前卡片.....	17
27.	AT88RF020 读取多块.....	17
28.	AT88RF020 写多块.....	18
	ST系列卡操作函数，SR176, SR512, SR1X4K等.....	18
29.	激活 S T 系列卡.....	18
30.	使激活 S T 卡休眠.....	19
31.	获得SR176 卡UID.....	19

---

32.	SR176 卡读数据块.....	20
33.	SR176 卡写数据块.....	20
34.	SR176 卡锁数据块.....	20
35.	SR176 卡锁块状态.....	21
36.	SR176 卡读取多个数据块.....	21
37.	SR176 卡写多个数据块.....	22
38.	获取SR512 卡的UID.....	22
39.	SR512 卡读数据块.....	23
40.	SR512 卡写数据块.....	23
41.	SR512 卡锁数据块.....	24
42.	SR512 卡获取锁状态.....	24
43.	SR512 卡读多块.....	24
44.	SR512 卡写多块.....	25
45.	SRIX4K卡密钥认证.....	26
	SAM卡操作函数 .....	26
46.	SAM卡波特率设置.....	26
47.	SAM卡复位.....	27
48.	SAM卡执行COS命令.....	27
49.	SAM卡PPS波特率设置.....	28
6	关于射频卡读写模块.....	29
7	订购方式.....	29

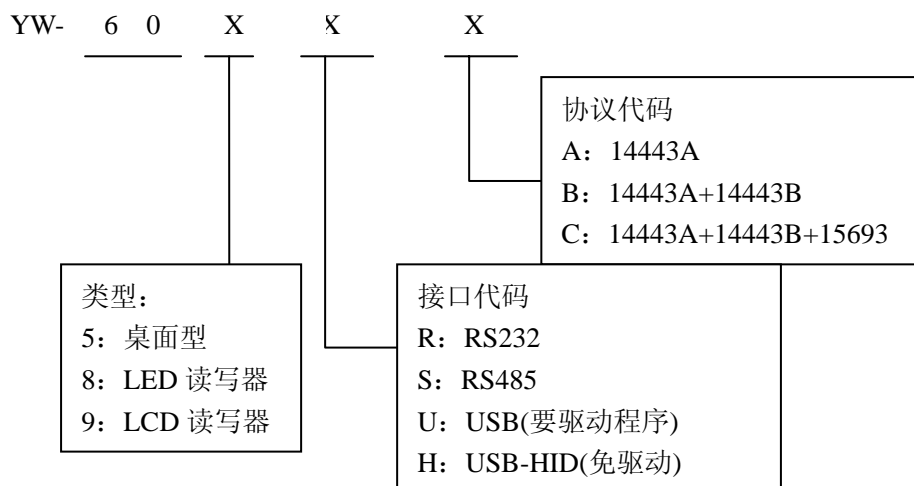
# 1 概述

YW-605系列射频卡读卡器是采用13.56M非接触射频技术设计而成的通用型读卡器，内嵌MF RC530、RC531和CL RC632等一些列原装芯片，读写性能稳定可靠。天线和主板一体化设计，使芯片的读写性能更出色的发挥出来。该系列读卡器采用USB/ RS232 等数据传输方式与计算机相连接，使计算机二次开发变得更为简单。

该系列读卡器支持ISO14443-A Mifare One S50, S70, ISO14443B和ISO15693及其兼容卡片。广泛应用于非接触智能水、电、气三表、交通一卡通读写器，桌面发卡器，门禁考勤读写卡器，汽车电子感应锁配套，办公/商场/洗浴中心储物箱的安全控制，各种防伪系统及生产过程控制，数据采集等。各种型号支持的卡片请参见第二节。

## 2 YW-605 系列型号

13.56M 射频卡读写器系列有以下型号，请选择合适的射频卡读写器。



### 3 功能特点

- ☞ 13.56M高频射频卡读写器，具有主动读卡模式和被动读卡模式，让您的程序设计更轻松。
- ☞ 读卡距离可达5到10cm。
- ☞ 提供DLL API支持，使二次开发省时省力。（同时提供VC, VB, Delphi, C++Builder, C#.net, WEB客户端等各种开发语言的例程）
- ☞ 读卡器上有可自编程控制的3个红黄蓝指示灯和1个蜂鸣器。
- ☞ 指示灯和蜂鸣器可随意控制。
- ☞ 结构简单紧凑，外观协调大方。
- ☞ 极高的性价比。

### 4 技术指标

- ☞ 串口波特率：19200BPS

- ☞ 电源：DC5V ± 10%
- ☞ 最大功耗：1.5W
- ☞ 环境温度：-30℃ ~ +70℃
- ☞ 相对湿度：35% ~ 95%
- ☞ 外形尺寸：120 \* 84 \* 25 (mm)
- ☞ 重量：约100g

## 5 二次开发指南

YW-605系列读卡器提供二次开发功能，用户可以在我们的DLL的基础上调用相应的函数开发应用程序，我们提供Delphi, C++Builder, VB, VC等的调用例程和相关函数声明单元，或者按照读卡器的通信协议直接开发应用程序。

库函数，C++语言版，其它语言见相应的函数声明文件。

### 动态库及读写器相关函数

#### 1. 读取库函数内部版本号

**函数原形：** `int stdcall YW_GetDLLVersion(void);`

**参数列表：** 无

**返回值：** 大于0为版本号，小于0为错误

#### 2. DES加解密函数

**函数原形：** `int stdcall DES(unsigned char cModel, unsigned char *pkey, unsigned char *in, unsigned char *out);`

**参数列表：**

参数	类型	含义
cModel	unsigned	加解密方向，0为加密，1为解密

	char	
pkey	unsigned char*	加解密密钥，8个字节
in	unsigned char*	原始数据，8个字节
out	unsigned char*	加解密后的数据，8个字节

**返回值：**无意义

### 3. 3DES加解密函数

**函数原形：**`int stdcall DES3(unsigned char cModel, unsigned char *pKey, unsigned char *In, unsigned char *Out);`

**参数列表：**

参数	类型	含义
cModel	unsigned char	加解密方向，0为加密，1为解密
pkey	unsigned char*	加解密密钥，16个字节
in	unsigned char*	原始数据，8个字节
out	unsigned char*	加解密后的数据，8个字节

**返回值：**无意义

### 4. 带向量的3DES加解密函数

**函数原形：**`int stdcall DES3_CBC(unsigned char cModel, unsigned char *pKey, unsigned char *In, unsigned char *Out, unsigned char`

\*pIV);

参数列表:

参数	类型	含义
cModel	unsigned char	加解密方向, 0为加密, 1为解密
pkey	unsigned char*	加解密密钥, 16个字节
in	unsigned char*	原始数据, 8个字节
out	unsigned char*	加解密后的数据, 8个字节
pIV	unsigned char*	加解密向量, 8个字节

返回值: 无意义

## 5. 初始化串口

函数原形: `int stdcall YW_ComInitial(int PortIndex, int Bound);`

参数列表:

参数	类型	含义
PortIndex	int	串口号, 1--255
Bound	int	通信波特率, 2400—115200, 默认为19200

返回值: 1成功, 0失败

## 6. 释放串口

函数原形: `int stdcall YW_ComFree(void);`

参数列表: 无

返回值: 1成功, 0失败



## 7. USB无驱读写器，初始化USB

函数原形: `int stdcall YW_USBHIDInitial(void);`

参数列表: 无

返回值: 1成功, 0失败

## 8. USB无驱读写器，释放USB

函数原形: `int stdcall YW_USBHIDFree(void);`

参数列表: 无

返回值: 1成功, 0失败

## 9. 修改读写器串口波特率

函数原形: `int stdcall YW_ComNewBound(int ReaderID ,int NewBound);`

参数列表:

参数	类型	含义
ReaderID	<code>int</code>	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
NewBound	<code>int</code>	新的波特率 0x01->9600bps 0x02->14400bps 0x03->19200bps 0x04->28800bps 0x05->38400bps 0x06->57600bps 0x07->115200bps

返回值: 1成功, 0失败

## 10. 设置设备标识

函数原形: `int stdcall YW_SetReaderID(int OldID, int NewID);`

参数列表:

参数	类型	含义
OldID	<code>int</code>	老的设备标示ID, 范围 0x0000-0xFFFF
NewID	<code>int</code>	修改成新的设备标示ID, 范围 0x0000-0xFFFF

返回值: 1成功, 0失败

## 11. 查询设备标识

函数原形: `int stdcall YW_GetReaderID(int ReaderID);`

参数列表:

参数	类型	含义
ReaderID	<code>int</code>	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0

返回值: >=0成功, 并且为所获取的设备标示, <0失败

## 12. 读取读卡器内部版本号

函数原形: `int stdcall YW_GetReaderVersion(int ReaderID);`

参数列表:

参数	类型	含义
ReaderID	<code>int</code>	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0

返回值: 大于0为版本号, 小于0为错误

## 13. 查询读写器产品序列号

**函数原形:** `int stdcall YW_GetReaderSerial(int ReaderID, char *ReaderSerial);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
ReaderSerial	Char *	读取的产品序列号, 长度为8个字节

**返回值:** 大于0为成功, 小于0为失败

#### 14. 蜂鸣器控制函数

**函数原形:** `int stdcall YW_Buzzer(int ReaderID, int Time_ON, int Time_OFF, int Cycle);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
Time_ON	int	蜂鸣器鸣叫时间, 单位: 秒
Time_OFF	int	蜂鸣器静音时间, 单位: 秒
Cycle	int	把Time_ON和Time_OFF作为一个周期, 则此参数为执行此周期的次数。

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

#### 15. LED指示灯控制

**函数原形:** `int stdcall YW_Led(int ReaderID, int LEDIndex, int Time_ON, int Time_OFF, int Cycle, int LedIndexOn);`

## 参数列表:

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
LEDIndex	int	LED灯序号 01: 红灯 02: 绿灯 04: 黄灯
Time_ON	int	LED灯亮时间, 单位: 秒
Time_OFF	int	LED灯灭时间, 单位: 秒
Cycle	int	把Time_ON和Time_OFF作为一个周期, 则此参数为执行此周期的次数。
LedIndexOn	int	最后要亮的灯: 00: 全灭 01: 红灯 02: 绿灯 04: 黄灯

返回值: 大于0为命令发送成功, 小于0为命令发送失败

## 16. 设置LED显示器显示的内容

函数原形: `int stdcall YW_LEDDisplay(int ReaderID, int Alignment, char *LEDText);`

## 参数列表:

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0

Alignment	int	显示时的对齐方式： 1: 左对齐 2: 居中对齐 3: 右对齐
LEDText	Char *	要显示的字符串。 可显示的字符如下： 0123456789AbCdEF.-

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

## 17. 设置天线的状态

**函数原形:** int stdcall YW\_AntennaStatus(int ReaderID, bool Status);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
Status	bool	True: 开天线 False: 关天线

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

## 18. 设置寻卡模式

**函数原形:** int stdcall YW\_SearchCardMode(int ReaderID, int

SearchMode);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0

SearchMode	char	卡类型 0x41-----ISO14443A 0x42----- ISO14443B 0x31----- ISO15693 0x53-----ST系列卡 0x52-----AT88RF020等
------------	------	---

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### ISO14443B相关函数

#### 19. Type B 卡复位

**函数原形:** `int stdcall YW_TypeB_Reset(int ReaderID, unsigned char Mode, int *rtLen, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
Mode	unsigned char	寻卡的模式 0x52----- 所有卡 0x26----- 激活卡
rtLen	int *	Type B卡复位信息的长度
pData	Char *	Type B卡复位信息

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

#### 20. Type B 卡执行C O S命令

**函数原形:** `int stdcall YW_TypeB_COS(int ReaderID, int LenCOS, unsigned char *Com_COS, int *DataLen, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
Mode	unsigned char	寻卡的模式 0x52----- 所有卡 0x26----- 激活卡
LenCOS	int	执行的C O S命令的长度
Com_COS	unsigned char*	要执行的COS命令
rtLen	int *	执行COS后返回的数据的长度
pData	Char *	执行COS后返回的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

## AT88RF020卡片操作函数

### 21. AT88RF020密钥认证

**函数原形:** int stdcall YW\_AT88RF020\_Check(int ReaderID, unsigned char \*Key);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
Key	Char *	密钥, 8 字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 22. AT88RF020读数据块

**函数原形:** int stdcall YW\_AT88RF020\_Read(int ReaderID, int BlockID, unsigned char \*pData);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	要读取的数据块
pData	Char *	读取的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

**23. AT88RF020写数据块**

**函数原形:** int stdcall YW\_AT88RF020\_Write(int ReaderID, int BlockID, unsigned char \*pData);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	要写入的数据块号
pData	Char *	要写入的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

**24. AT88RF020锁数据块**

**函数原形:** int stdcall YW\_AT88RF020\_Lock(int ReaderID, unsigned char \*LockByte);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0



LockByte	Char *	锁块数据，4字节
----------	--------	----------

**返回值：** 大于0为命令发送成功，小于0为命令发送失败

## 25. AT88RF020计数函数

**函数原形：** `int stdcall YW_AT88RF020_Count(int ReaderID, unsigned char *Signature);`

**参数列表：**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID，范围0x0000-0xFFFF，如果未知，则ReaderID=0
Signature	Char *	签名，6字节

**返回值：** 大于0为命令发送成功，小于0为命令发送失败

## 26. AT88RF020不选择当前卡片

**函数原形：** `int stdcall YW_AT88RF020_DeSel(int ReaderID);`

**参数列表：**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID，范围0x0000-0xFFFF，如果未知，则ReaderID=0

**返回值：** 大于0为命令发送成功，小于0为命令发送失败

## 27. AT88RF020读取多块

**函数原形：** `int stdcall YW_AT88RF020_ReadMulti(int ReaderID, int BlockID, int BlockNums, int *LenData, unsigned char *pData);`

**参数列表：**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID，范围

		0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	要读取的开始块号
BlockNums	int	块数量
LenData	Int*	返回的数据长度
pData	unsigned char *	返回的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

## 28. AT88RF020写多块

**函数原形:** `int stdcall YW_AT88RF020_WriteMulti(int ReaderID, int BlockID, int BlockNums, int LenData, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	要读取的开始块号
BlockNums	int	块数量
LenData	Int	要写入的数据长度
pData	unsigned char *	要写入的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

ST系列卡操作函数, SR176, SR512, SRIX4K等

## 29. 激活 S T 系列卡

**函数原形:** `int stdcall YW_ST_Active(int ReaderID, unsigned char *ChipID);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
ChipID	unsigned char *	卡的ChipID, 一个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

**30. 使激活 S T 卡休眠**

**函数原形:** int stdcall YW\_ST\_DeActive(int ReaderID);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

**31. 获得SR176卡UID**

**函数原形:** int stdcall YW\_SR176\_GetUID(int ReaderID, unsigned char \*UID);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
UID	unsigned char *	SR176卡的UID, 8个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 32. SR176卡读数据块

**函数原形:** `int stdcall YW_SR176_Read(int ReaderID, int BlockID, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
BlockID	int	块号
pData	unsigned char *	返回读取的数据, 2 个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 33. SR176卡写数据块

**函数原形:** `int stdcall YW_SR176_Write(int ReaderID, int BlockID, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
BlockID	int	块号
pData	unsigned char *	要写入的数据, 2 个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 34. SR176卡锁数据块

**函数原形:** `int stdcall YW_SR176_Lock(int ReaderID, unsigned char LockByte);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
LockByte	unsigned char	锁数据, 一个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 35. SR176卡锁块状态

**函数原形:** `int stdcall YW_SR176_LockStatus(int ReaderID, unsigned char *LockByte);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
LockByte	unsigned char*	返回锁数据, 一个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 36. SR176卡读取多个数据块

**函数原形:** `int stdcall YW_SR176_ReadMulti(int ReaderID, int BlockID, int BlockNums, int *LenData, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则

		ReaderID=0
BlockID	int	读取的开始块号
BlockNums	int	块数量
LenData	Int*	返回读取的数据长度
pData	unsigned char*	返回读取的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 37. SR176卡写多个数据块

**函数原形:** int stdcall YW\_SR176\_WriteMulti(int ReaderID, int BlockID, int BlockNums, int LenData, unsigned char \*pData);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	读取的开始块号
BlockNums	int	块数量
LenData	Int	要写入的数据长度
pData	unsigned char*	要写入的数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 38. 获取SR512卡的UID

**函数原形:** int stdcall YW\_SR512\_GetUID(int ReaderID, unsigned char \*UID);

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围

		0x0000-0xFFFF, 如果未知, 则 ReaderID=0
UID	unsigned char*	UID, 8 个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 39. SR512卡读数据块

**函数原形:** `int stdcall YW_SR512_Read(int ReaderID, int BlockID, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	块号
pData	unsigned char*	返回读取的数据, 4 个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

### 40. SR512卡写数据块

**函数原形:** `int stdcall YW_SR512_Write(int ReaderID, int BlockID, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	块号
pData	unsigned	要写入的数据, 4 个字节

	char*	
--	-------	--

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

#### 41. SR512卡锁数据块

**函数原形:** `int stdcall YW_SR512_Lock(int ReaderID, unsigned short LockByte);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
LockByte	unsigned short	锁数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

#### 42. SR512卡获取锁状态

**函数原形:** `int stdcall YW_SR512_LockStatus(int ReaderID, unsigned short *LockByte)`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
LockByte	unsigned short*	锁数据

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

#### 43. SR512卡读多块

**函数原形:** `int stdcall YW_SR512_ReadMulti(int ReaderID, int`



BlockID, int BlockNums, int \*LenData, unsigned char \*pData)

参数列表:

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	开始块号
BlockNums	int	块数量
LenData	Int*	返回的数据长度
pData	unsigned char *	返回的数据

返回值: 大于0为命令发送成功, 小于0为命令发送失败

#### 44. SR512卡写多块

函数原形: int stdcall YW\_SR512\_WriteMulti(int ReaderID, int BlockID, int BlockNums, int LenData, unsigned char \*pData)

参数列表:

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围 0x0000-0xFFFF, 如果未知, 则 ReaderID=0
BlockID	int	开始块号
BlockNums	int	块数量
LenData	Int	写入数据长度
pData	unsigned char *	要写入的数据

返回值: 大于0为命令发送成功, 小于0为命令发送失败

#### 45. SRIX4K卡密钥认证

**函数原形:** `int stdcall YW_SRIX4K_Check(int ReaderID, unsigned char *Key, unsigned char *Signature)`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
Key	unsigned char*	密钥, 6 个字节
Signature	unsigned char*	输出签名, 3 个字节

**返回值:** 大于0为命令发送成功, 小于0为命令发送失败

#### SAM卡操作函数

#### 46. SAM卡波特率设置

**函数原形:** `int __stdcall YW_SAM_ResetBaud(int ReaderID, int SAMIndex, int BaudIndex);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
SAMIndex	int	SAM卡序号
BaudIndex	int	0x00->9600 (默认复位波特率) 0x01->19200 0x02->38400 0x03->55800 0x04->57600 0x05->115200

**返回值:** 大于0为成功, 小于0为失败

#### 47. SAM卡复位

**函数原形:** `int __stdcall YW_SAM_Reset(int ReaderID, int SAMIndex, int *rtLen, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
SAMIndex	int	SAM卡序号
rtLen	int *	SAM卡复位返回的数据pData的长度
pData	unsigned char *	SAM卡复位返回的数据

**返回值:** 大于0为成功, 小于0为失败

#### 48. SAM卡执行COS命令

**函数原形:** `int __stdcall YW_SAM_COS(int ReaderID, int SAMIndex, int LenCOS, unsigned char *Com_COS, int *rtLen, unsigned char *pData);`

**参数列表:**

参数	类型	含义
ReaderID	int	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
SAMIndex	int	SAM卡序号
LenCOS	int	向SAM卡要发送的COS命令的长度
Com_COS	unsigned	向SAM卡要发送的COS命令

	<code>char *</code>	
rtLen	<code>unsigned char *</code>	SAM执行COS命令后返回的数据的长度
pData	<code>unsigned char *</code>	SAM执行COS命令后返回的数据

**返回值:** 大于0为成功, 小于0为失败

#### 49. SAM卡PPS波特率设置

**函数原形:** `int __stdcall YW_SAM_PPSBaud(int ReaderID, int SAMIndex, int BaudIndex);`

**参数列表:**

参数	类型	含义
ReaderID	<code>int</code>	所要获取的设备标示ID, 范围0x0000-0xFFFF, 如果未知, 则ReaderID=0
SAMIndex	<code>int</code>	SAM卡序号
BaudIndex	<code>int</code>	0x00->9600 (默认复位波特率) 0x01->19200 0x02->38400 0x03->55800 0x04->57600 0x05->115200

**返回值:** 大于0为成功, 小于0为失败

## 6 关于射频卡读写模块

如果不在计算机下使用，如果想将读写卡的功能嵌入到你的系统中，可以选择射频卡读写模块YW201, YW202, YW203, YW204等。关于YW-200系列射频卡读写模块请登录网站<http://www.youwokeji.com.cn>查看。

## 7 订购方式

可以通过我们的网站或电话订购。或者联系当地的经销商。

北京友我科技有限公司

网站: <http://www.youwokeji.com.cn>

电话: 010-59395668

24小时手机: 13671114914

Email: [coodor@126.com](mailto:coodor@126.com)