

# UHF 超高频RFID读写器YW-602H

## 开发手册

(version 1.6)

● 简体中文版

繁體中文版

English Edition

网址: <https://www.youwokeji.com.cn>

EMAIL: coodor@126.com

手机: 13671114914

联系人: 周先生

### 目录

1 概述 .....	3
2 功能特点 .....	3
3 二次开发指南 .....	3
1. 读取库函数内部版本号 .....	3
2. USB 无驱读写器，初始化 USB .....	3
3. USB 无驱读写器，释放 USB .....	3
4. 设置设备标识 .....	4
5. 查询设备标识 .....	4
6. 读取读卡器内部版本号 .....	4
7. 蜂鸣器控制函数 .....	5
8. LED 指示灯控制 .....	5
9. 获取读写器的功率 .....	6
10. 设置读写器的功率 .....	6
11. 获取读写器寻卡阈值 .....	6
12. 设置读写器寻卡阈值 .....	7
13. G2 标签寻卡 .....	7
14. G2 标签读内容 .....	8
15. G2 标签写内容 .....	9
16. G2 标签写 EPC，此函数必须单标签操作 .....	10
17. G2 销毁标签 .....	10
18. G2 标签设置保护 .....	11
19. G2 标签内容擦除 .....	12
20. G2 标签设定某个标签读保护 .....	13
21. G2 标签设定标签读保护 .....	13
22. G2 标签解锁读保护 .....	14
23. G2 标签锁定用户区 .....	14
4 订购方式 .....	15

# 1 概述

YW-602H 是一款高性能的 UHF 超高频电子标签一体机，完全自主知识产权设计，结合专有的高效信号处理算法，在保持高识读率的同时，实现对电子标签的快速读写处理，可广泛应用于物流、车辆管理、防伪系统及生产过程控制等多种无线射频识别（RFID）系统。

## 2 功能特点

- 充分支持符合 EPC CLASS1 G2 标准的电子标签；
- 工作频率 902~928MHz(可以按不同国家或地区要求调整)
- 输出功率达至 14-20dBm（可调）；
- USB 免驱动设计，读写距离<1 米；
- 支持 windows，arm64 linux（银河麒麟，通信 UOS）。

## 3 二次开发指南

YW-602H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，linux QT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。

### 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

返回值: 大于0为成功，小于0为错误

### 2. 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

### 3. 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

#### 4. 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 0x00-0xFF, 0xFF为广播地址
NewID	<code>int</code>	修改成新的设备标示ID, 范围 0x00-0xFF

返回值: >=0成功, 并且为所获取的设备标示, <0失败

#### 5. 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>	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址

返回值: >=0成功, 并且为所获取的设备标示, <0失败

#### 6. 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>	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址

返回值: 大于0为版本号, 小于0为错误

## 7. 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

## 8. 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

## 9. 获取读写器的功率

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

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址

**返回值：**大于0为成功，小于0为错误

## 10. 设置读写器的功率

**函数原形：**int stdcall YW\_GetPower (int ReaderID, int Power);

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
Power	int	读写器的功率，从0到30

**返回值：**大于0为成功，小于0为错误

## 11. 获取读写器寻卡阈值

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

**参数列表:**

参数	类型	含义
ReaderID	int	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址

**返回值:** 大于0为成功, 小于0为错误

## 12. 设置读写器寻卡阈值

**函数原形:** `int stdcall YW_SetInventoryThreshold (int ReaderID, int Threshold);`

**参数列表:**

参数	类型	含义
ReaderID	int	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址
Threshold	int	读写器寻卡阈值, 可以通过此参数 来调节读卡距离, 标准卡参考有效 范围190到210, 值越大距离越近

**返回值:** 大于0为成功, 小于0为错误

## 13. G2标签寻卡

**函数原形:** `int stdcall YW_G2_Inventory (int ReaderID, bool GetEPC, int AddrTID, int LenTID, int *TagsNum, int *DataLen, unsigned char *EPC);`

**参数列表:**

参数	类型	含义
ReaderID	int	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址
GetEPC	bool	是否获得EPC
AddrTID	int	如果获得TID, 则为首地址
LenTID	int	要读取的TID长度, 偶数字节

TagsNum	int *	寻到的标签数量
DataLen	int *	读到的数据
EPC	unsigned char *	读到的EPC，结构为EPC长度+EPC 码

**返回值：**大于0为成功，小于0为错误

#### 14. G2标签读内容

**函数原形：**`int stdcall YW_G2_Read (int ReaderID, int EPCLen, unsigned char *EPC, unsigned char MemType, unsigned char StartPos, unsigned char ReadLen, unsigned int Password, int *G2DataLen, unsigned char *G2Data);`

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
EPCLen	int	要读写的EPC标签EPC长度，字节 数
EPC	unsigned char *	EPC内容
MemType	unsigned char	要读的区块 G2_BLOCK_RESERVE=0;保留区 G2_BLOCK_EPC =1;EPC区 G2_BLOCK_TID =2;TID区 G2_BLOCK_USER =3;USER区
StartPos	unsigned char	开始读取的地址（必须为偶数地 址）
ReadLen	unsigned char	要读取的字节数（必须为偶数）



Password	unsigned int	标签操作的密码，默认为0
G2DataLen	int *	读到的字节数
G2Data	unsigned char	读到的内容

**返回值：**大于0为成功，小于0为错误

## 15. G2标签写内容

**函数原形：**int stdcall YW\_G2\_Write (int ReaderID, int EPCLen, unsigned char \*EPC, unsigned char MemType, unsigned char StartPos, unsigned int Password, int G2DataLen, unsigned char \*G2Data);

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
EPCLen	int	要读写的EPC标签EPC长度，字节数
EPC	unsigned char *	EPC内容
MemType	unsigned char	要写的区块 G2_BLOCK_RESERVE=0;保留区 G2_BLOCK_EPC =1;EPC区 G2_BLOCK_TID =2;TID区 G2_BLOCK_USER =3;USER区
StartPos	unsigned char	开始写的地址（必须为偶数地址）
Password	unsigned int	标签操作的密码，默认为0

G2DataLen	int	要写的字节数
G2Data	unsigned char	写的内容

**返回值：**大于0为成功，小于0为错误

## 16. G2标签写EPC，此函数必须单标签操作

**函数原形：**int stdcall YW\_G2\_WriteEPC (int ReaderID, int EPCLen, unsigned char \*EPC, unsigned int Password);

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
EPCLen	int	要读写的EPC标签EPC长度，字节 数，必须为偶数
EPC	unsigned char *	EPC内容
Password	unsigned int	标签操作的密码，默认为0

**返回值：**大于0为成功，小于0为错误

## 17. G2销毁标签

**函数原形：**int stdcall YW\_G2\_KillTag (int ReaderID, int EPCLen, unsigned char \*EPC, unsigned int Password);

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
EPCLen	int	要操作的EPC标签EPC长度，字节 数，必须为偶数

EPC	<code>unsigned char *</code>	要操作的EPC
Password	<code>unsigned int</code>	标签操作的密码，密码为0时，不可销毁

**返回值：**大于0为成功，小于0为错误

## 18. G2标签设置保护

**函数原形：**`int stdcall YW_G2_SetProtected (int ReaderID, int EPCLen, unsigned char *EPC, unsigned char Protect, unsigned char ProtectMode, unsigned int Password);`

**参数列表：**

参数	类型	含义
ReaderID	<code>int</code>	老的设备标示ID，范围0x00-0xFF, 0xFF为广播地址
EPCLen	<code>int</code>	要操作的EPC标签EPC长度，字节数，必须为偶数
EPC	<code>unsigned char *</code>	要操作的EPC
Protect	<code>unsigned char</code>	0x00 时，控制销毁密码读写保护设定。 0x01 时，控制访问密码读写保护设定。 0x02 时，控制 EPC 存储区读写保护设定。 0x03 时，控制 TID 存储区读写保护设定。 0x04 时，控制用户存储区读写保护设定。
ProtectMode	<code>unsigned char</code>	当 Protect 为 0x00 或 0x01，即当设置 Kill 密码区或访问密码区的时候，ProtectMode 的值代表的意义如下： 0x00：设置为无保护

		<p>下的可读可写</p> <p>0x01: 设置为永远可读可写</p> <p>0x02: 设置为带密码可读可写</p> <p>0x03: 设置为永远不可读不可写</p> <p>当 Protect 为 0x02、0x03、0x04 的时候，即当设置 EPC 区、TID 区及用户区的时候，ProtectMode 的值代表的意义如下：</p> <p>0x00: 设置为无保护下的可写</p> <p>0x01: 设置为永远可写</p> <p>0x02: 设置为带密码可写</p> <p>0x03: 设置为永远不可写</p>
Password	unsigned int	标签操作的密码，默认为0

**返回值：** 大于0为成功，小于0为错误

## 19. G2标签内容擦除

**函数原形：** `int stdcall YW_G2_Earse (int ReaderID, int EPCLen, unsigned char *EPC, unsigned char MemType, unsigned char StartPos, unsigned char Nums, unsigned int Password);`

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
EPCLen	int	要读写的EPC标签EPC长度，字节数
EPC	unsigned char *	EPC内容
MemType	unsigned	要写的区块

	char	G2_BLOCK_RESERVE=0;保留区 G2_BLOCK_EPC =1;EPC区 G2_BLOCK_TID =2;TID区 G2_BLOCK_USER =3;USER区
StartPos	unsigned char	开始写的地址（必须为偶数地址）
Nums	unsigned char	要擦除的字节数
Password	unsigned int	标签操作的密码，默认为0

**返回值：**大于0为成功，小于0为错误

## 20. G2标签设定某个标签读保护

**函数原形：**int stdcall YW\_G2\_SetReadProtectedWithEPC (int ReaderID, int EPCLen, unsigned char \*EPC, unsigned int Password);

**参数列表：**

参数	类型	含义
ReaderID	int	老的设备标示ID，范围 0x00-0xFF, 0xFF为广播地址
EPCLen	int	要读写的EPC标签EPC长度，字节 数
EPC	unsigned char *	EPC内容
Password	unsigned int	标签操作的密码，默认为0

**返回值：**大于0为成功，小于0为错误

## 21. G2标签设定标签读保护

**函数原形:** `int stdcall YW_G2_SetReadProtected (int ReaderID, unsigned int Password);`

**参数列表:**

参数	类型	含义
ReaderID	int	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址
Password	unsigned int	标签操作的密码, 默认为0

**返回值:** 大于0为成功, 小于0为错误

## 22. G2标签解锁读保护

**函数原形:** `int stdcall YW_G2_SetUnlockReadProtected (int ReaderID, unsigned int Password);`

**参数列表:**

参数	类型	含义
ReaderID	int	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址
Password	unsigned int	标签操作的密码, 默认为0

**返回值:** 大于0为成功, 小于0为错误

## 23. G2标签锁定用户区

**函数原形:** `int stdcall YW_G2_LockUser(int ReaderID, int EPCLen, unsigned char *EPC, unsigned int Password, unsigned char UserAddr);`

**参数列表:**

参数	类型	含义
ReaderID	int	老的设备标示ID, 范围 0x00-0xFF, 0xFF为广播地址

EPCLen	int	要读写的EPC标签EPC长度，字节数
EPC	unsigned char *	EPC内容
Password	unsigned int	标签操作的密码，默认为0
UserAddr	unsigned char	要锁定的用户地址，偶数

**返回值：**大于0为成功，小于0为错误

## 4 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: <https://www.youwokeji.com.cn>

24小时手机: 13671114914

Email: coodor@126.com