

NFC 读卡接口

成都鱼住未来科技有限公司

二次开发: <http://www.yzfuture.cn>

技术支持: faq@yzfuture.cn

售后: sales@yzfuture.cn

电话: [028-82880293](tel:028-82880293)

日期	版本	说明	作者
2018/12/09	V1.0.0	NFC&OTG 证件读取，支持身份证/港澳居民居住证	TygerZH

1. 概述

本 SDK 支持二代身份证、护照及 EID 的读取。

二代身份证接口添加了身份鉴权操作，只有当鉴权通过的用户才可以使用公司提供的解码服务器进行身份证解码。

护照及 EID 功能开通需要单独申请。

可兼容目前市场大多数读卡器。

用本 SDK 读证件需要配套我公司专用证件读卡器。

2. 硬件购买

读卡器购买：

<https://item.taobao.com/item.htm?id=610814936998>

<https://item.taobao.com/item.htm?id=611082485024>

服务器购买：

<https://item.taobao.com/item.htm?id=611230762637>

3. 接口概要

接口文件 ReadCardInfoInterface.h 中，同时需要包含 Type.h 文件。

4. 回调

- `typedef void(__stdcall *cardReadProgress)(unsigned int nProgress, YZWLHandle nhandle);`
返回身份证读卡进度，一共 20 步。

5. 结构体

- 卡片类型

```
typedef enum cardType
{
    unkwonType = -1,
    ACardType = 0, // A 卡
    BCardType = 1  // B 卡
};
```
- 身份证解码结果

```
typedef struct TwoIdInfoStructEx
{
```

```

char arrTwoIdName[30];           //姓名 UNICODE
char arrTwoIdSex[2];             //性别 UNICODE
char arrTwoIdNation[4];          //民族 UNICODE
char arrTwoIdBirthDay[16];       //出生日期 UNICODE YYYYMMDD
char arrTwoIdAddress[70];        //住址 UNICODE
char arrTwoIdNo[36];             //身份证号码 UNICODE
char arrTwoIdSignedDepartment[30]; //签发机关 UNICODE
char arrTwoIdValidityPeriodBegin[16]; //有效期起始日期 UNICODE YYYYMMDD
char arrTwoIdValidityPeriodEnd[16]; //有效期截止日期 UNICODE YYYYMMDD 有效期为长期时存储“长期”

char arrTwoOtherNO[18]; // 通行证类号码
char arrTwoSignNum[4];    // 签发次数
char arrTwoRemark1[6]; // 预留区
char arrTwoType[2];       // 证件类型标识
char arrTwoRemark2[6]; // 预留区

char arrTwoIdNewAddress[70]; //最新住址 UNICODE
char arrReserve[2];          //保留字节 字节对齐用
unsigned char arrTwoIdPhoto[1024]; //照片信息
unsigned char arrTwoIdFingerprint[1024]; //指纹信息
unsigned char arrTwoIdPhotoJpeg[4096]; //照片信息 JPEG 格式
unsigned int unTwoIdPhotoJpegLength; //照片信息长度 JPEG 格式
};

```

● 护照解码结果

```

typedef struct _EPassportInfoStruct
{
    // 护照信息
    char szPaperType[100]; // 证件类型(缩写)
    char szTypeFullName[100]; // 证件类型(全称)
    char szSignedDepartment[100]; // 签发国家或签发机构
    char szENName[100]; // 英文名
    char szCNName[100]; // 中文名
    char szIdNo[100]; // 证件号码
    char szDocumentID[100]; // 护照 ID
    char szCountry[100]; // 国籍
    char szBirthDay[6]; // 出生日期 UNICODE YYMMDD
    char szSex[1]; // 性别 UNICODE
    char szValidityPeriodEnd[6]; // 有效期截止日期 UNICODE YYMMDD
    char faceImage[1024 * 40]; // 照片信息
}EPassportInfoStruct;

```

6. 注册接口

为支持不同语言开发，接口统一为标准 C 接口。

- **regeditGetVerificationCode**
通过手机号请求注册短信验证码。
szServerIp: 服务器 IP 地址 (默认为 id.yzfuture.cn)
nServerPort: 服务器端口 (默认为 8848)
szPhoto: 需要注册的手机号
szError: 返回的错误信息
nError: 返回的错误信息长度
返回值:
成功 - 0
失败 - <=0
-100 传入的错误信息内存不足
- **regeditGetLogonAppKey**
通过短信验证码获取 appKey。
szServerIp: 服务器 IP 地址 (默认为 id.yzfuture.cn)
nServerPort: 服务器端口 (默认为 8848)
szPhoto: 需要注册的手机号
szCode: 需手机收到的短信验证码
szAppKey: 返回的 appKey
nAppkey: 返回的 appKey 长度
szError: 返回的错误信息
nError: 返回的错误信息长度
返回值:
成功 - 0
失败 - <=0
-100 传入的错误信息内存不足
- **regeditGetAppKeyUseNum**
获取 appKey 剩余可用次数。
szServerIp: 服务器 IP 地址 (默认为 id.yzfuture.cn)
nServerPort: 服务器端口 (默认为 8848)
szAppKey: 请求的 appKey
szError: 返回的错误信息
nError: 返回的错误信息长度
返回值:
剩余可用次数

7. 解码接口

为支持不同语言开发，接口统一为标准 C 接口。

- **setDeviceType**
设置鱼住标准读卡器还是离线读卡器，默认为鱼住标准读卡器
typedef enum _eDeviceType
{

```

        yzwlType = 0, // 鱼住读卡器
        sdtapiType = 1 // 离线读卡器
    }eDeviceType;

```

- **cardOpenDevice**

打开读卡器硬件设备。

nouttime:超时时间（秒）

nDeviceNo:读卡器序号，默认为 0（为解决同时有多个相同读卡器的情况，可以选其中某一个为读卡设备）（如果 **setDeviceType** 设置为离线读卡器的话，**nDeviceNo** 应该为 1001~1016 中的某一值）

返回值:

成功 - 读卡器句柄

失败 <= 0

- **setCardType**

设置卡片类型

nDeviceHandle:打开的设备句柄

ctype:卡片类型，身份证为 B 卡

返回值:

True - 成功

False - 失败

- **cardFindCard**

寻卡

nDeviceHandle:打开的设备句柄

bmove:固定写 true

返回值:

True - 成功

False - 失败

- **cardGetCardSN**

获取卡片 SN 码（离线读卡器有可能返回数据为空，视读卡器品牌而定）

nDeviceHandle:打开的设备句柄

szsn:SN 码

nlen:返回的 SN 码字符串长度，字节为单位（传入时需要赋值 szsn 的字节空间大小，如果小于实际 SN 码长度，则返回失败）

返回值:

True - 成功

False - 失败

- **cardGetCardDN**

获取身份证 DN 码（离线读卡器返回数据为空）

nDeviceHandle:打开的设备句柄

szsn:DN 码

nlen:返回的 DN 码字符串长度，字节为单位（传入时需要赋值 szsn 的字节空间

大小，如果小于实际 DN 码长度，则返回失败)

返回值:

True - 成功
False - 失败

- **cardGetDeviceNO**

获取读卡器芯片唯一序列号 (离线读卡器有可能返回数据为空，视读卡器品牌而定)

nDeviceHandle: 打开的设备句柄

szsn: 芯片唯一序列号

nlen: 返回的序列号字符串长度，字节为单位 (传入时需要赋值 szsn 的字节空间大小，如果小于实际序列号长度，则返回失败)

返回值:

True - 成功
False - 失败

- **cardGetDeviceSN**

获取读卡器出厂序列号 (离线读卡器有可能返回数据为空，视读卡器品牌而定)

nDeviceHandle: 打开的设备句柄

szsn: 读卡器出厂序列号

nlen: 返回的序列号字符串长度，字节为单位 (传入时需要赋值 szsn 的字节空间大小，如果小于实际读卡器出厂序列号长度，则返回失败)

返回值:

True - 成功
False - 失败

- **cardReadTwoCard**

读身份证、港澳居住证或 EID 信息，集成了寻卡、选卡和读身份证的操作。

nDeviceHandle: 打开的设备句柄

cardCB: 回调的读卡进度

szFactoryFlag: 用户申请的 appkey, 可以一个公司用一个 appkey, 也可以公司每种 apk 用一个 appkey (申请过程请见下方 “appkey 申请流程”)

szServerIp: 服务器 IP 地址 (申请 appkey 时会提供)

nServerPort: 服务器端口 (申请 appkey 时会提供)

cardinfo: 解码后的结果保存在这个结构体里 (unicode 格式)

bTest: 固定填 false

返回值:

True - 成功
False - 失败

- **cardGetSerialNumber**

获取当次读卡的流水号，可用于后期对账 (离线读卡器返回数据为空)

返回值:

字符串形式的流水号，如果解码在鉴权前就失败，则流水号可能为空

- **cardBeep**
读卡器蜂鸣一次（10ms）（离线读卡器有可能无效）
nDeviceHandle: 打开的设备句柄
返回值:
True - 成功
False - 失败
- **cardGetLastErrorCode**
获取最后一次出错代码
nDeviceHandle: 打开的设备句柄
返回值:
True - 成功
False - 失败
- **cardGetLastError**
获取最后一次出错信息
nDeviceHandle: 打开的设备句柄
nlen: 返回错误信息的长度
返回值:
最后一次出错信息
- **cardGetErrorInfo**
根据传入的错误代码查找对应的错误信息
nDeviceHandle: 打开的设备句柄
nlen: 返回错误信息的长度
nErrorCode: 需要查找的错误代码
返回值:
查找出错信息
- **cardCloseDevice**
关闭读卡器设备
nDeviceHandle: 打开的设备句柄
返回值:
无
- **decodeCardImage**
将读出来的身份证信息解码成 bmp 格式
srcimage: 读出来的头像信息原数据
outimage: 解码出来的图片数据，内存空间由用户自己管理，不得小于 40K
outlen: 传入时为 outimage 实际大小，传出时为实际图片大小
返回值:
True - 成功
False - 失败
- **readEPassportInfo**

读护照信息（此接口功能开通需要单独向本公司申请）

nDeviceHandle: 打开的设备句柄

szFactoryFlag: 用户申请的 appkey (申请过程请和公司商务联系)

szServerIp: 服务器 IP 地址 (申请 appkey 时会提供)

nServerPort: 服务器端口 (申请 appkey 时会提供)

szN0: 护照号

szBirth: 出生日期

szEndtime: 护照有效期

cardinfo: 解码后的结果保存在这个结构体里

bTest: 固定填 false

返回值:

True - 成功

False - 失败

8. 错误码

0	成功
-1	未知错误
-2	参数无效
-3	格式出错
-4	获取卡号失败
-999	获取 SN 失败
-998	获取随机码失败
-997	获取 DN 失败
-996	跳转 2 号扇区数据失败
-995	读 11 号扇区数据失败
-994	读 12 号扇区数据失败
-993	读 13 号扇区数据失败
-992	读 21 号扇区数据失败
-991	上层回调执行数据失败
-1999	网络创建失败
-1998	发送 SN 失败
-1997	接收 SN 回应失败
-1996	发送 DN 失败
-1995	接收第一次认证 APDU 失败
-1994	执行第一次认证 APDU 失败
-1993	发送第一次认证结果失败
-1992	发送第二次认证失败
-1991	接收第二次认证 APDU 失败
-1990	执行第二次 APDU 失败
-1989	接收不到身份证结果

-1988	接收结果超时
-19999	设备初始化失败
-19998	读卡器设备找不到
-19997	打开设备失败
-19996	设备没有打开
-19995	设置卡片类型失败
-19994	寻卡失败
-19993	选卡失败
-19992	防冲突失败
-19991	重置卡片失败
-19990	不支持的卡片类型
-19989	执行 Apdu 指令失败
-19988	设备已关闭
-25000	首次鉴权出错
-24999	设备鉴权出错
-24998	获取服务器信息时失败
-24997	此设备没有解码权限
-24996	获取护照信息时失败
-24995	获取卡片类型失败
-29999	网络出错的开始码
-29998	网络连接失败，请检查网络
-29997	解析用户信息失败
-29996	解析身份证服务器信息失败
-29995	解析身份证信息失败
-29994	解析身份证信息异常
-29993	解析护照服务器信息异常
-29992	解析护照信息失败
-29991	解析护照信息异常
-29990	保存卡片信息失败
-29989	返回包命令 ID 错误
-29988	返回错误 ID 失败
-29987	返回包解析数据出错
-29986	返回包格式出错
-39999	厂商信息为空
-39998	创建设备失败
-39997	读卡器没有打开
-39996	寻卡选卡失败
-39995	获取 SN 或 DN 失败

-39994	获取手机权限失败
-39993	获取读卡器信息失败
-39992	获取身份证信息失败
-39991	解码身份证信息失败
-39990	身份证解码服务器地址异常
-39989	获取读卡器序列号失败
-39988	获取读卡器芯片序列号失败
-39987	输入空间不足
-39986	输出空间不足
-39985	加载动态库失败
-39984	导入动态函数失败
-39983	图片解码失败
-39982	回调函数为空
-39981	连接服务器失败
-39980	数据发送失败
-39979	与服务器交互失败
-49999	获取身份证姓名失败
-49998	获取身份证性别失败
-49997	获取身份证民族失败
-49996	获取身份证生日失败
-49995	获取身份证地址失败
-49994	获取身份证号失败
-49993	获取身份证发证机关失败
-49992	获取身份证开始日期失败
-49991	获取身份证结束日期失败
-49990	获取身份证新地址失败
-49989	获取身份证其它号码失败
-49988	获取身份证发证次数失败
-49987	获取身份证备注失败
-49986	获取身份证子类型失败
-49985	获取身份证头像失败
-49984	获取身份证指纹失败

9. 接口调用样例

```
TwoIdInfoStructEx cardinfo;  
if ( “离线读卡器” ) // 设备型号 YNR201 离线版  
{  
    setDeviceType(1); // 切换成离线读卡器  
    nIndex = 1001;  
}  
else // 设备型号 YNR101 标准版 YNR101N 共享版
```

```

{
    setDeviceType(0); // 切换成鱼住标准读卡器
    nindex = 0;
}
YZWLHandle hlHandle = cardOpenDevice(2, nindex);
if (hlHandle>0)
{
    if (setCardType(hlHandle, BCardType))
    {
        if (cardFindCard(hlHandle, true))
        {
            bool bret = cardReadTwoCard(hlHandle, nullptr,
            szAppKey, szIP, nport, cardinfo, false);
            if (bret)
            {
                // 解码成功
            }
        }
    }
}
cardCloseDevice(hlHandle);
}

```

10. appkey 申请流程

- Windows 版
 - ✓ 调用 regeditGetVerificationCode 获取短信验证码
 - ✓ 调用 regeditGetAppKeyUseNum 获取 appkey
- 安卓版
 - ✓ 下载安卓版 demo
 - <https://lv-bucket.oss-cn-qingdao.aliyuncs.com/%E9%B1%BC%E4%BD%8F%E6%9C%AA%E6%9D%A5%E8%AF%81%E4%BB%B6%E8%AF%86%E5%88%ABdemo.apk>
 - ✓ 安装后如下

17:03 ... 4G 100%

鱼住未来证件识别测试

姓名
性别
国家
民族
出生日期
英文名
证件类型
证件号码
地址
签发机关
有效期
签发次数

是否通知:

通行证号码

OTG刷卡 跳转 重置

剩余次数: 0
AppKey: appkey

注册 APPKEY 修改 APPKEY 找回 APPKEY

二次开发请联系我们: <http://www.yzfuture.cn>
技术支持: faq@yzfuture.cn
售后: sales@yzfuture.cn
电话: 028-82860293
读卡器: <https://item.taobao.com/item.htm?spm=686.10014936998>
二次开发提供安卓平台NFC和OTG证件识别及Windows平台USB读卡器证件识别, 包括: 身份证识别、港澳居住证识别、护照识别、EID识别、人脸识别、人脸对比功能接口, 方便快速集成

✓ 点击“注册 appKey”按钮

17:03 ... 4G 100%

鱼住未来证件识别测试

姓名
性别
国家
民族
出生日期
英文名
证件类型
证件号码
地址
签发机关
有效期
签发次数

是否通知:

通行证号码

OTG刷卡 跳转 重置

剩余次数: 0
AppKey: appkey

注册 APPKEY 修改 APPKEY 找回 APPKEY

二次开发请联系我们: <http://www.yzfuture.cn>
技术支持: faq@yzfuture.cn
售后: sales@yzfuture.cn
电话: 028-82860293
读卡器: <https://item.taobao.com/item.htm?spm=686.10014936998>
二次开发提供安卓平台NFC和OTG证件识别及Windows平台USB读卡器证件识别, 包括: 身份证识别、港澳居住证识别、护照识别、EID识别、人脸识别、人脸对比功能接口, 方便快速集成

✓ 填写注册信息，并完成注册



- ✓ 注册成功，系统返回 appKey
- ✓ 点击 appkey 字符位置即可复制



11.