

# 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](mailto:faq@yzfuture.cn)

售后: [sales@yzfuture.cn](mailto: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/护照/EID	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。

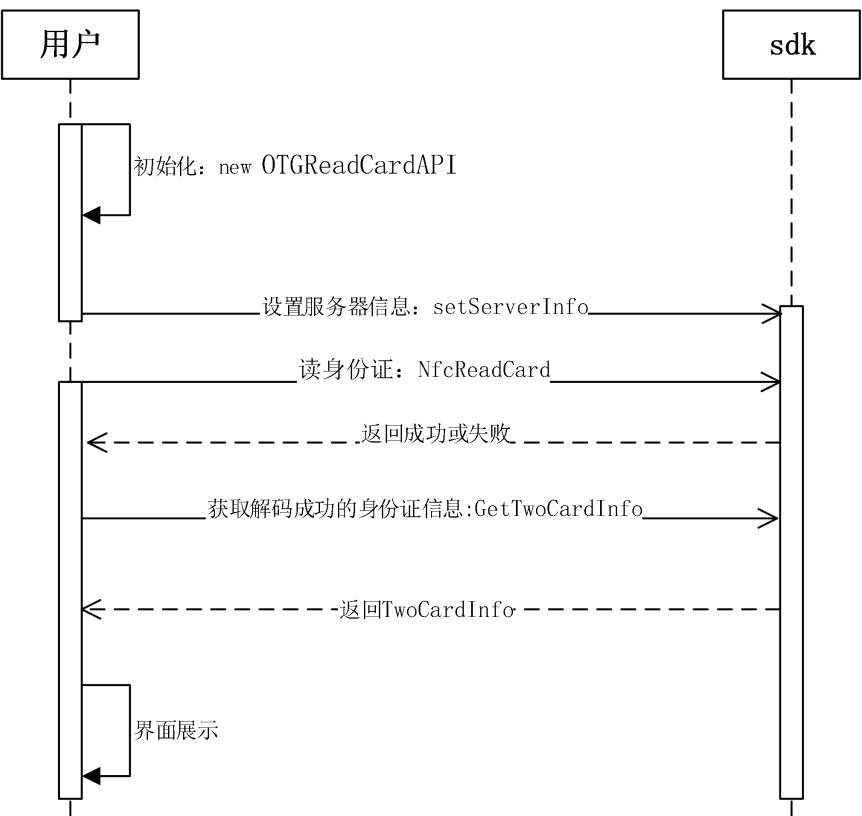
NFC 读卡用于支持 NFC 的安卓设备上。

OTG 读卡用于支持 USB 的安卓设备上 (USB 口需要配套我公司专门证件读卡器)。

## 2. 接口概要

接口文件在 com\readTwoGeneralCard\OTGReadCardAPI.java 中。

## 3. 流程图



## 4. 回调

在使用本 SDK 前必须实现 ActiveCallBack 接口中的相关函数，原型如下：

```
public interface ActiveCallBack{
    void readProgress(int npaogress);
    void setUserInfo(String sztxt);
}
```

- **void readProgress(int npaogress);**  
返回身份证读卡进度，一共 20 步。
- **void setUserInfo(String sztxt);**  
函数空实现即可，有时会返回调试信息。

## 5. 接口

- **OTGReadCardAPI**  
接口初始化操作。  
**paramContext**: android 的上下文  
**cb**: 实现回调的类  
**bNFC**: 统一写成 false
- **setDeviceType**  
设置读卡器类型，默认为标准读卡器，当读卡器环境有变化时需要调用（比如原来是标准读卡器需要切换成离线读卡器时需要调用一次）。  
**ndeviceType**: 读卡器类型（0-标准版 1-离线版）
- **GetVersion**  
获取当前版本号
- **setServerInfo**  
设置服务器列表（**初始化后最先调用，必须**）  
**twoCardServerList**: 身份证解码服务器列表  
**eidServerList**: 为 null  
**bTestServer**: 统一填 false  
返回值: 无
- **NfcReadCard**  
通过 NFC 读卡，同步操作，执行结束返回状态。  
**szAppID**: APPID，每家商户的唯一 ID（需要向本公司申请后获取）  
**userInfo**: 用户信息，暂时为空  
**intent**: NFC 句柄, OTG 时填 null  
**defaultType**: 只能填 *eTwoGeneralCard*  
**userData**: 用户透传数据，主要用来区别每个用户的身份和后期对账使用，所以需要保证这个 **userData** 在同一个 **appID** 下不能重复。重复的话，有可能会導致此 **appID** 下 **userData** 相同的其它用户无权限，切记切记！！  
**bAuth**: 填 false

返回值:

41 - 失败  
90 - 成功

- GetCardType

返回 NfcReadCard 中读取的卡片类型。

返回值:

同 NfcReadCard 中的 eCardType, 返回 *eTwoGeneralCard*

- GetTwoCardInfo

当读卡为身份证类型的时候, 获取身份证详细信息。

返回值:

```
public class TwoCardInfo {
    public String szTwoIdName; // 姓名
    public String szTwoIdSex; // 性别
    public String szTwoIdNation; // 民族
    public String szTwoIdBirthday; // 出生日期
    public String szTwoIdAddress; // 住址
    public String szTwoIdNo; // 身份证号码
    public String szTwoIdSignedDepartment; // 签发机关
    public String szTwoIdValidityPeriodBegin; // 有效期起始日期 YYYYMMDD
    public String szTwoIdValidityPeriodEnd; // 有效期截止日期 YYYYMMDD 有效期为
    长期时存储“长期”
    public String szTwoIdNewAddress; // 最新住址
    public byte[] arrTwoIdPhoto; // 照片信息
    public byte[] arrTwoIdFingerprint; // 指纹信息

    public String szSNID;
    public String szDNID;

    public String szTwoOtherNO; // 通行证类号码
    public String szTwoSignNum; // 签发次数
    public String szTwoRemark1; // 预留区
    public String szTwoType; // 证件类型标识
    public String szTwoRemark2; // 预留区
}
```

- GetErrorInfo

获取执行过程中的出错信息。

## 6. 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

## 7. 调用样例

```

OTGReadCardAPI  ReadCardAPI;
ReadCardAPI = new OTGReadCardAPI(getApplicationContext(), this, false);
ArrayList<Serverinfo> twoCardServerlist = new ArrayList<Serverinfo>();
twoCardServerlist.add(new Serverinfo("xx.xxxx.xx", 0000));
szFactory = "xxxxxxxxxxxxxxxxxxxx";
if (离线读卡器)
{
    ReadCardAPI.setDeviceType(1);
}
else
{
    ReadCardAPI.setDeviceType(0);
}
ReadCardAPI.setServerInfo(twoCardServerlist, null, false);
if (ReadCardAPI.NfcReadCard(szFactory, null, getIntent(), eCardType.eTwoGeneralCard, "", false)
== 90)
{
    // 解码成功
}
else
{
    // 解码失败
}

```



## 8. App 界面及使用说明



四幅图从左到右分别是权限申请、注册、修改 appKey、主界面

### ● 权限申请

程序第一次打开会申请需要使用到的相关权限，有且只有一次，请务必同意，否则只有清除缓存或者卸载重装后才会再次申请(或者去设置中手动打开所有权限)；如果申请失败，会影响读身份证操作。

### ● 注册

如果没有 appKey 需要注册，系统默认第一次使用会自动弹出注册框，如果取消在以后每次刷身份证或 otg 读卡前会弹出注册框，不注册成功不能使用。注册的时候需要填入昵称、手机号和短信验证码，昵称为了以后激活便于查找和管理，所以建议填自己真实姓名。填入手机号后会有 60 秒钟等待验证码接收时间，60 秒后可以重复申请验证码，**但同一个手机号一天最多只能接受 5 次验证码**，请一定注意。

注册成功后自动绑定一个免费的试用 appKey，该 appKey 可免费使用 100 次，如果到期后可点击主界面上“注册”按钮，使用其它手机号重新申请。

### ● 修改 appKey

如果已有一个 appKey，可以点击主界面上“修改”按钮进行替换，替换后使用新 appKey 的权限。系统替换新的 appKey 后，原来旧的无法通过系统找回，如果旧 appKey 还有用，请自行备份。

### ● 主界面

主要分三个功能区：

**上：**负责身份证、EID 及港澳通行证的证件识别；NFC 功能直接将证件靠近 NFC 感应芯片即可刷卡；**OTG 功能需要配套我公司专门证件读卡器才可使用**；重置可清除界面上证件信息。NFC 功能主要用于带 NFC 功能的安卓设备；OTG 功能主要用于不带 NFC 功能但带 USB 功能的安卓设备。

**中：**介绍公司联系方式及二次开发联系方式。

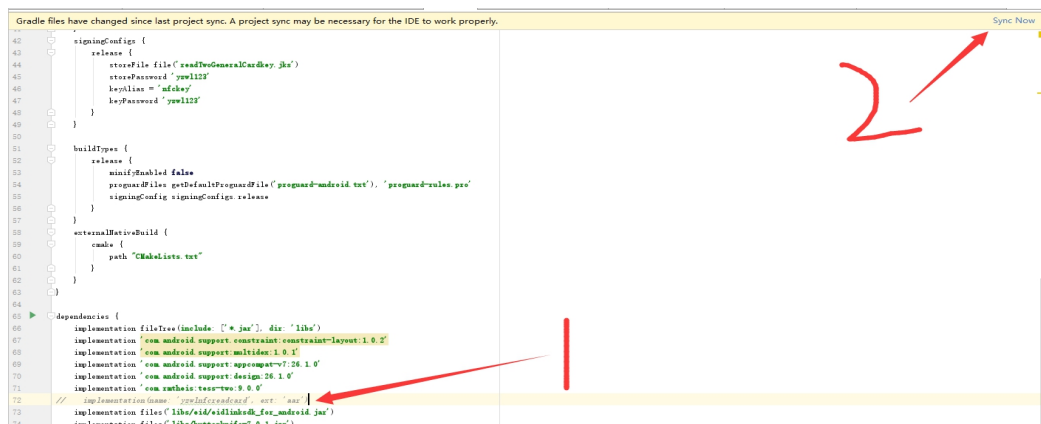
**下：**负责 appKey 的注册和替换。

## 9. 使用出错及解决方法

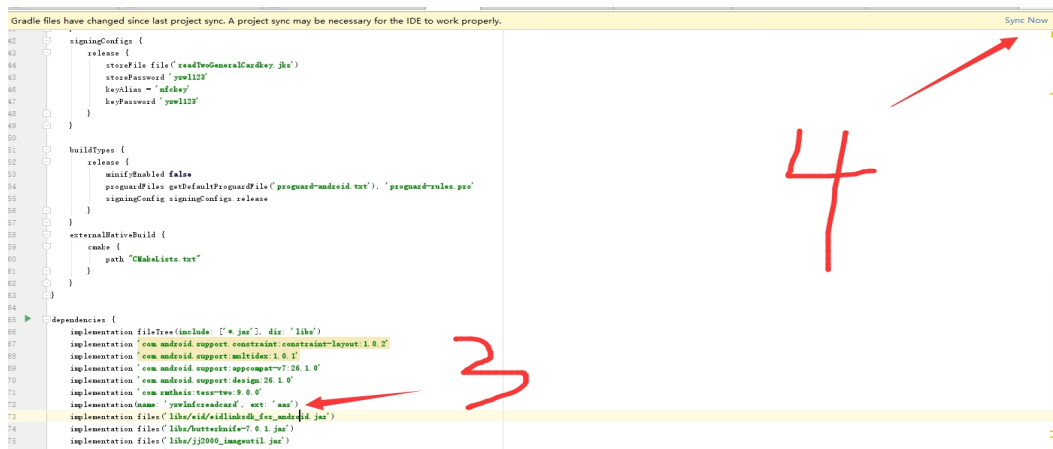
### ● 方法调用不起

可能是 aar 没有加载起，或者替换了 aar 后没有同步工程，可按照下列图示进行：

先注释掉 build.gradle 文件中的 implementation(name: 'yzwlncfreadcard', ext: 'aar') 这一行，然后点” Sync Now”



去掉注释 build.gradle 文件中的 implementation(name: 'yzwlncfreadcard', ext: 'aar') 注释，然后点” Sync Now”



编译生成新的 apk 即可

### ● FileProvider 冲突

由于 aar 包中需要操作本地文件，所以使用了 FileProvider，如果外面也用的话，有可能会报冲突，造成 App 读取文件自己的 FileProvider 失败，解决方法如下：

添加如下代码：tools:replace="android:authorities"

### ● Apk 更新失败

请查看是否是写错了下载地址，导致你们的 apk 更新失败，我们 aar 对更新没有影响

- **Demo 源码编译出来不可用**

Demo 只是提供调用样例，里面的 appkey 是用的我们统一的 appkey，这个 key 是没有试用权限的。如果需要调用的话，请用我们官网上的 demo 程序注册一个新的 appkey，然后把新 appkey 替换到代码中重新生成即可使用。

10.