

TLINK 串口指纹 SDK API 说明 V1.0

TLINK UART Fingerprint SDK API Documentation V1.0

一、SDK概览

I.SDK Overview

二、详细API说明

II. Detailed API Documentation

2.1 init()

2.2 free()

2.3 fingerGETImage()

2.4 fingerStoreCharToBuffer(FingerprintWithTLK1NC.BufferEnum)

2.5 fingerUPTemplateFromBuffer(FingerprintWithTLK1NC.BufferEnum)

2.6 fingerStoreCharToFlashPage(int)

2.7 fingerUPTemplateFromFlashPage(int)

2.8 fingerDOWNTemplateToFlashPage(int, byte[])

2.9 fingerDOWNTemplateToBuffer(FingerprintWithTLK1NC.BufferEnum, byte[])

2.10 fingerPKTemplateBuffer(FingerprintWithTLK1NC.BufferEnum,
FingerprintWithTLK1NC.BufferEnum)

2.11 fingerPKTemplatePage(int, int)

2.12 fingerPKTemplate(int, FingerprintWithTLK1NC.BufferEnum)

2.13 fingerPKTemplate2(FingerprintWithTLK1NC.BufferEnum, int)

2.14 fingerSearchTemplate(FingerprintWithTLK1NC.BufferEnum, int, int)

2.15 fingerDELTemplateBufferID(FingerprintWithTLK1NC.BufferEnum)

2.16 fingerDELTemplatePageID(int)

2.17 fingerCLEARTemplate()

2.18 fingerCLEARTemplateBuffer()

2.19 fingerGETTemplateCount()

2.20 generateImg(byte[], String)

2.21 startUPIImage(FingerprintWithTLK1NC.IUPIImageCallback)

2.22 BufferEnum Enumeration

一、SDK 概览

I. SDK Overview

`com.rscja.deviceapi.FingerprintWithTLK1NC` 是基于 Android 平台的 TLK1NC 指纹模块操作类，采用单例模式设计，核心作用是为上层应用提供标准化的 API，实现指纹模块的初始化与资源释放、指纹图像采集、特征值生成与存储（缓冲区 / Flash）、指纹特征上传下载、指纹比对（精确比对 / 库内搜索）、模板管理（删除 / 清空）及指纹图像回调获取等全流程指纹识别功能。

`com.rscja.deviceapi.FingerprintWithTLK1NC` is a TLK1NC fingerprint module operation class based on the Android platform, designed with the singleton pattern. Its core function is to provide a standardized API for upper-layer applications, enabling the full-process fingerprint recognition functions of the fingerprint module including initialization and resource release, fingerprint image collection, feature generation and storage (buffer/Flash), fingerprint feature upload and download, fingerprint matching (exact matching/database search), template management (deletion/clearing), and callback acquisition of fingerprint images.

1. 简单易用的单例设计

1. Simple and Easy-to-Use Singleton Design

您无需手动创建该类的实例，只需通过 `getInstance()` 方法即可获取全局唯一的管理对象，确保调用过程简单且资源高效。

You do not need to manually create an instance of this class. You can obtain the globally unique management object only through the `getInstance()` method, ensuring that the calling process is simple and resource-efficient.

2. 使用步骤

2. Usage Steps

第一步: 通过 `init()` 初始化指纹相关资源。

Step 1: Initialize fingerprint resources via `init()`.

第二步: 初始化成功后，根据需求执行以下操作：

Step 2: After successful initialization, perform the following operations as required:

- `fingerStoreCharToFlashPage(int)` 注册指纹
- `fingerStoreCharToFlashPage(int)` Register fingerprint
- `fingerSearchTemplate(FingerprintWithTLK1NC.BufferEnum, int, int)` 识别指纹
- `fingerSearchTemplate(FingerprintWithTLK1NC.BufferEnum, int, int)` Identify fingerprint
- `startUPIImage(IUPIImageCallback)` 生成指纹图片
- `startUPIImage(IUPIImageCallback)` Generate fingerprint image
- `fingerUPTemplateFromFlashPage(int)` 上传模板
- `fingerUPTemplateFromFlashPage(int)` Upload template
- `fingerDOWNTemplateToFlashPage(int, byte[])` 下载模板
- `fingerDOWNTemplateToFlashPage(int, byte[])` Download template

第三步: 退出app调用 `free()` 释放资源。

Step 3: Call `free()` to release resources on app exit.

3.使用示例

3.Usage Examples

```
public class FingerprintDemo {
    public void test(){
        int result = mFingerprintWithTLK1NC.init();
        if(!result){
            //Initialization failed
            return;
        }
        //*****Register fingerprint*****
        while (true) {
            if (mFingerprintWithTLK1NC.fingerGETImage() > 0) {
                if (mActivity.mFingerprintWithTLK1NC.fingerStoreCharToFlashPage(pageID)) {
                    //Success
                } else {
                    //Failed
                }
                break;
            }
        }

        //*****Identify fingerprint*****
        while (true) {
            if (mFingerprintWithTLK1NC.fingerGETImage() > 0) {
                if (mFingerprintWithTLK1NC.fingerStoreCharToBuffer(FingerprintWithTLK1NC.BufferEnum.B0))
            {
                int pageId =
mActivity.mFingerprintWithTLK1NC.fingerSearchTemplate(FingerprintWithTLK1NC.BufferEnum.B0, 0, 128);
                if (pageId >= 0) {
                    //Success
                } else {
                    //Failed
                }
                break;
            }
        }
    }

    //*****Acquire fingerprint image*****
    mFingerprintWithTLK1NC.startUPIImage(new FingerprintWithTLK1NC.IUPIImageCallback() {
        @Override
        public void messageInfo(String s) {
            //Real-time acquisition message
        }
        @Override
        public void onComplete(boolean result, byte[] bytes,int failureCode) {
            Log.i(TAG, "failureCode=" + failureCode + " result=" + result);
            if (result) {
                //-----Generate BMP image-----
                String path="sdcard/fingerprint.bmp";
                if (mFingerprintWithTLK1NC.generateImg(bytes, path)){
                    //BMP image generated successfully, path: ${path}
                }
            } else {
                //Image acquisition failed
            }
        }
    })
    @Override
```

```

        public void progress(int i) {
            //Image acquisition progress (0-100)
        }
    });

    //*****Get fingerprint template*****
    byte[] bytes = mActivity.mFingerprintWithTLK1NC.fingerUPTemplateFromFlashPage(pageID);
    if (bytes != null && bytes.length > 0) {
        //Get fingerprint template successfully, bytes: fingerprint template data
    } else {
        //Failed to get fingerprint template
    }

    //*****Fingerprint template comparison*****
    while (true){
        if(mActivity.mFingerprintWithTLK1NC.fingerGETImage()>0) {
            boolean result =
mActivity.mFingerprintWithTLK1NC.fingerStoreCharToBuffer(FingerprintWithTLK1NC.BufferEnum.B1);
            if(result) {
                byte[] fingerprintTemplate = readFile(fingerprintTemplateFilePath);//Fingerprint
template data
                result =
mActivity.mFingerprintWithTLK1NC.fingerDOWNTemplateToBuffer(FingerprintWithTLK1NC.BufferEnum.B2, bytes);
                if (result) {
                    result =
mActivity.mFingerprintWithTLK1NC.fingerPKTemplateBuffer(FingerprintWithTLK1NC.BufferEnum.B1,
FingerprintWithTLK1NC.BufferEnum.B2);
                    if (result) {
                        //Fingerprint template matching success
                    } else {
                        //Fingerprint template matching failed
                    }
                    break;
                }
            }
        }
    }

    //*****Download fingerprint features to specified Flash
Page*****
    byte[] fingerprintTemplate = readFile(fingerprintTemplateFilePath);//Fingerprint template data
    if (mActivity.mFingerprintWithTLK1NC.fingerDOWNTemplateToFlashPage(pageID,
fingerprintTemplate)) {
        mHandler.sendEmptyMessage(DOWNLOAD_SUCC);
    } else {
        mHandler.sendEmptyMessage(DOWNLOAD_FAIL);
    }
    //.....
    //.....
    mFingerprintWithTLK1NC.free();
}
}

```

二、详细API说明

II.Detailed API Documentation

2.1 init()

项目 Item	说明 Description
原型 Prototype	boolean init()
描述 Description	初始化指纹读写器的相关资源 Initialize the relevant resources of the fingerprint reader
参数 Parameters	无 N/A
返回 Return	true:成功 false:失败 true:Success false:Failure

2.2 free()

项目 Item	说明 Description
原型 Prototype	boolean free()
描述 Description	释放指纹读写器已初始化的相关资源 Release the initialized relevant resources of the fingerprint reader
参数 Parameters	无 N/A
返回 Return	true:成功 false:失败 true:Success false:Failure

2.3 fingerGETImage()

项目 Item	说明 Description
原型 Prototype	int fingerGETImage()
描述 Description	从指纹采集器采集指纹图像 Collect fingerprint images from the fingerprint collector
参数 Parameters	无 N/A
返回 Return	-1:采集错误/失败; 非负数 (0~100) : 指纹图像质量评分 (数值越高表示图像质量越好) -1: Collection error/failure; Non-negative number (0~100): Fingerprint image quality score (higher value indicates better image quality)
备注 Remark	用于检测指纹传感器上是否有指纹 Used to detect whether there is a fingerprint on the sensor

2.4 fingerStoreCharToBuffer(FingerprintWithTLK1NC.BufferEnum)

项目 Item	说明 Description
原型 Prototype	boolean fingerStoreCharToBuffer(FingerprintWithTLK1NC.BufferEnum BufferID)
描述 Description	生成指纹特征，存储在指定特征缓冲区 Generate fingerprint features and store them in the specified feature buffer
参数 Parameters	BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.5 fingerUPTemplateFromBuffer(FingerprintWithTLK1NC.BufferEnum)

项目 Item	说明 Description
原型 Prototype	byte[] fingerUPTemplateFromBuffer(FingerprintWithTLK1NC.BufferEnum BufferID)
描述 Description	上传指纹特征值，将指定缓冲区中的特征值上传到上位机 Upload fingerprint features from the specified buffer to the host computer.
参数 Parameters	BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2)
返回 Return	字节数组：指纹模板数据（失败返回空数组） Byte array: Fingerprint template data (empty array if failed)

2.6 fingerStoreCharToFlashPage(int)

项目 Item	说明 Description
原型 Prototype	boolean fingerStoreCharToFlashPage(int PageID)
描述 Description	生成指纹特征，存储在Flash Page Generate fingerprint features and store them in the Flash Page
参数 Parameters	PageID: Flash Page ID (0~127) PageID: Flash Page ID (0~127)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.7 fingerUPTemplateFromFlashPage(int)

项目 Item	说明 Description
原型 Prototype	byte[] fingerUPTemplateFromFlashPage(int PageID)
描述 Description	上传指纹特征值，将指定Flash Page中的特征值上传到上位机 Upload fingerprint features from the specified Flash Page to the host computer.

项目 Item	说明 Description
参数 Parameters	PageID: Flash Page ID (0~127) PageID: Flash Page ID (0~127)
返回 Return	字节数组: 指纹模板数据 (失败返回空数组) Byte array: Fingerprint template data (empty array if failed)

2.8 fingerDOWNTemplateToFlashPage(int, byte[])

项目 Item	说明 Description
原型 Prototype	boolean fingerDOWNTemplateToFlashPage(int PageID, byte[] templateData)
描述 Description	下载指纹特征到指定的Flash Page中 Download the fingerprint features to the specified Flash Page
参数 Parameters	PageID: Flash Page ID (0~127) PageID: Flash Page ID (0~127) templateData: 模板数据 templateData: Fingerprint template data
返回 Return	true:成功 false:失败 true:Success false:Failure

2.9 fingerDOWNTemplateToBuffer(FingerprintWithTLK1NC.BufferEnum, byte[])

项目 Item	说明 Description
原型 Prototype	boolean fingerDOWNTemplateToBuffer(FingerprintWithTLK1NC.BufferEnum BufferID, byte[] templateData)
描述 Description	下载指纹特征到指定的特征缓冲区中 Download the fingerprint features to the specified feature buffer
参数 Parameters	BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2) templateData: 模板数据 templateData: Fingerprint template data
返回 Return	true:成功 false:失败 true:Success false:Failure

2.10 fingerPKTemplateBuffer(FingerprintWithTLK1NC.BufferEnum, FingerprintWithTLK1NC.BufferEnum)

项目 Item	说明 Description
原型 Prototype	boolean fingerPKTemplateBuffer(FingerprintWithTLK1NC.BufferEnum BufferID1, FingerprintWithTLK1NC.BufferEnum BufferID2)
描述 Description	精确比对指定特征缓冲区中特征 Accurately compare features in the specified feature buffers

项目 Item	说明 Description
参数 Parameters	BufferID1: 第一个缓冲区枚举值 BufferID1: First buffer zone enumeration value BufferID2: 第二个缓冲区枚举值 BufferID2: Second buffer zone enumeration value
返回 Return	true:成功 false:失败 true:Success false:Failure

2.11 fingerPKTemplatePage(int, int)

项目 Item	说明 Description
原型 Prototype	boolean fingerPKTemplatePage(int PageID1, int PageID2)
描述 Description	精确比对指定Flash Page中特征 Accurately compare features in the specified Flash Pages
参数 Parameters	PageID1: 第一个Flash Page ID (0~127) PageID1: First Flash Page ID (0~127) PageID2: 第二个Flash Page ID (0~127) PageID2: Second Flash Page ID (0~127)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.12 fingerPKTemplate(int, FingerprintWithTLK1NC.BufferEnum)

项目 Item	说明 Description
原型 Prototype	boolean fingerPKTemplate(int PageID, FingerprintWithTLK1NC.BufferEnum BufferID)
描述 Description	精确比对指定特征缓冲区和指定的Flash Page中特征 Accurately compare features in the specified feature buffer and Flash Page
参数 Parameters	PageID: Flash Page ID (0~127) PageID: Flash Page ID (0~127) BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.13 fingerPKTemplate2(FingerprintWithTLK1NC.BufferEnum, int)

项目 Item	说明 Description
原型 Prototype	boolean fingerPKTemplate2(FingerprintWithTLK1NC.BufferEnum BufferID , int PageID)
描述 Description	精确比对指定特征缓冲区和指定的Flash Page中特征 Accurately compare features in the specified feature buffer and Flash Page

项目 Item	说明 Description
参数 Parameters	BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2) PageID: Flash Page ID (0~127) PageID: Flash Page ID (0~127)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.14 fingerSearchTemplate(FingerprintWithTLK1NC.BufferEnum, int, int)

项目 Item	说明 Description
原型 Prototype	int fingerSearchTemplate(FingerprintWithTLK1NC.BufferEnum BufferID, int templateIdStart, int templateIdEnd)
描述 Description	下载一个特征到指定特征缓冲区，然后用此特征搜索指纹库中的模板 Download a feature to the specified buffer and search for matching templates in the fingerprint library.
参数 Parameters	BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2) templateIdStart: 待搜索的起始Template编号 templateIdStart: Start Template ID for search templateIdEnd: 待搜索的结束Template编号 templateIdEnd: End Template ID for search
返回 Return	整型: 匹配的模板编号 (无匹配返回-1) Integer: Matched template ID (return -1 if no match)

2.15 fingerDELTTemplateBufferID(FingerprintWithTLK1NC.BufferEnum)

项目 Item	说明 Description
原型 Prototype	boolean fingerDELTTemplateBufferID(FingerprintWithTLK1NC.BufferEnum BufferID)
描述 Description	删除内存中指定的模板 Delete the specified template in memory
参数 Parameters	BufferID: 缓冲区枚举值 (B0/B1/B2) BufferID: Buffer zone enumeration value (B0/B1/B2)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.16 fingerDELTTemplatePageID(int)

项目 Item	说明 Description
原型 Prototype	boolean fingerDELTTemplatePageID(int PageID)

项目 Item	说明 Description
描述 Description	删除Flash中指定的模板 Delete the specified template from Flash.
参数 Parameters	PageID: Flash Page ID (0~127) PageID: Flash Page ID (0~127)
返回 Return	true:成功 false:失败 true:Success false:Failure

2.17 fingerCLEARTemplate()

项目 Item	说明 Description
原型 Prototype	boolean fingerCLEARTemplate()
描述 Description	清空Flash中指纹模板 Clear all fingerprint templates from Flash.
参数 Parameters	无 N/A
返回 Return	true:成功 false:失败 true:Success false:Failure

2.18 fingerCLEARTemplateBuffer()

项目 Item	说明 Description
原型 Prototype	boolean fingerCLEARTemplateBuffer()
描述 Description	清空指纹特征缓冲区 Clear all fingerprint feature buffers
参数 Parameters	无 N/A
返回 Return	true:成功 false:失败 true:Success false:Failure

2.19 fingerGETTemplateCount()

项目 Item	说明 Description
原型 Prototype	int fingerGETTemplateCount()
描述 Description	获取可存储的指纹模板数 Get the number of storable fingerprint templates
参数 Parameters	无 N/A

项目 Item	说明 Description
返回 Return	整型：可存储的指纹模板总数 Integer: Total number of storable fingerprint templates

2.20 generateImg(byte[], String)

项目 Item	说明 Description
原型 Prototype	boolean generateImg(byte[] data, String filePath)
描述 Description	根据指纹数据生成指纹图片并保存到指定路径 Generate a fingerprint image from fingerprint data and save it to the specified path
参数 Parameters	data：指纹数据字节数组 data: Fingerprint data byte array filePath：图片保存路径 filePath: Image save path
返回 Return	true:成功 false:失败 true:Success false:Failure

2.21 startUPImage(FingerprintWithTLK1NC.IUImageCallback)

项目 Item	说明 Description
原型 Prototype	void startUPImage(FingerprintWithTLK1NC.IUImageCallback callBack)
描述 Description	开始获取指纹图像，通过回调反馈采集状态 Start acquiring fingerprint image and report acquisition status via callback.
参数 Parameters	callBack：图像采集回调接口实例 callBack: Image acquisition callback interface instance
返回 Return	无 N/A

2.22 BufferEnum 枚举

2.22 BufferEnum Enumeration

指纹读写器缓冲区枚举类，定义设备支持的3个缓冲区标识，每个枚举值映射唯一的整型数值。

Enumeration class for fingerprint reader buffer zones, defines 3 buffer zone identifiers supported by the device, each enumeration value maps to a unique integer value.

方法/枚举值 Method/Enumeration Value	描述 Description	参数 Parameters	返回值 Return
B0(0x00)	缓冲区0标识 Buffer zone 0 identifier	-	-

方法/枚举值 Method/Enumeration Value	描述 Description	参数 Parameters	返回值 Return
B1(0x01)	缓冲区1标识 Buffer zone 1 identifier	-	-
B2(0x02)	缓冲区2标识 Buffer zone 2 identifier	-	-
getValue()	获取枚举值对应的整型数值 Acquire the integer value corresponding to the enumeration value	无 N/A	枚举值对应的整型数值（如B0返回0） Integer value corresponding to the enumeration value (e.g., B0 returns 0)

2.23 IUPIImageCallback 接口

2.23 IUPIImageCallback Interface

指纹图像采集回调接口，用于实时反馈图像采集过程中的状态消息、完成结果及采集进度。

Callback interface for fingerprint image acquisition, used to real-time feed back status messages, completion results and acquisition progress during image acquisition.

方法 Method	描述 Description	参数 Parameters	返回值 Return
messageInfo(String msg)	接收图像采集的实时状态消息 Receive real-time status messages of image acquisition	msg: 采集图像的实时描述信息 msg: Real-time description information of image acquisition	无 N/A
onComplete(boolean b, byte[] data, int Param)	图像采集完成后的回调方法，返回 采集结果及相关数据 Callback method after image acquisition is completed, returns acquisition result and related data	b: true=采集成功，false=采集失败 b: true=Acquisition success, false=Acquisition failure data: 采集到的图像数据（失败时为 空） data: Acquired image data (empty if failed) Param: 附加参数，返回采集失败原因 （成功时无意义） Param: Additional parameter, returns the reason for acquisition failure (meaningless if successful)	无 N/A
progress(int progress)	返回指纹图像采集的实时进度 Returns the real-time progress of fingerprint image acquisition	progress: 采集进度值（0-100） progress: Acquisition progress value (0-100)	无 N/A