

中控指纹仪 SDK 开发指南 C API

版本：2.0

日期：2016 年 9 月

中控指纹仪 SDK 开发指南

Copyright ©ZKTeco Inc.2016 All rights reserved.

Release history

日期	版本	备注
2016 年 5 月 21 日	1.0	基础版
2016 年 6 月 1 日	1.1	增加外部图像接口
2016 年 9 月 17 日	2.0	更新为统一接口(保留 1.0 接口)

目录

1	概述.....	4
2	隐私权说明.....	4
3	系统需求.....	4
4	安装部署.....	4
5	SDK 接口说明.....	4
5.1	类型定义.....	4
5.1.1	常量	5
5.2	接口说明.....	5
5.2.1	ZKFPM_Init.....	5
5.2.2	ZKFPM_Terminate	5
5.2.3	ZKFPM_GetDeviceCount.....	5
5.2.4	ZKFPM_OpenDevice	6
5.2.5	ZKFPM_CloseDevice	6
5.2.6	ZKFPM_SetParameters.....	6
5.2.7	ZKFPM_GetParameters.....	7
5.2.8	ZKFPM_AcquireFingerprint	7
5.2.9	ZKFPM_AcquireFingerprintImage	8
5.2.10	ZKFPM_DBInit.....	8
5.2.11	ZKFPM_DBFree	9
5.2.12	ZKFPM_DBMerge	9
5.2.13	ZKFPM_DBAdd.....	10
5.2.14	ZKFPM_DBDel.....	10
5.2.15	ZKFPM_DBClear.....	10
5.2.16	ZKFPM_DBCount.....	11
5.2.17	ZKFPM_DBIdentify	11
5.2.18	ZKFPM_DBMatch	12
5.2.19	ZKFPM_ExtractFromImage.....	12
5.2.20	ZKFPM_AcquireFingerprintImage	13
5.2.21	ZKFPM_Base64ToBlob	13
5.2.22	ZKFPM_BlobToBase64	14
5.2.23	ZKFPM_BlobToBase64	14
6	附录.....	15
6.1	附录 1.....	15
6.2	附录 2.....	16

1 概述

欢迎使用中控指纹仪 SDK, 在使用前请您先仔细阅读本手册, 以便您能更快地掌握并使用中控指纹仪 SDK。

2 隐私权说明

本公司将本软件程序的使用权授权予您, 但您必须向本公司作出如下保证: 不在本协议规定的条款之外, 使用、拷贝、修改、租赁或转让本系统获取其中的任一部分。

3 系统需求

- 1) 操作系统: Ubuntu
- 2) 适用开发语言 C、C++

4 安装部署

- 1) 拷贝 lib-x86/lib-x64 所有动态库到/usr/lib 下。

5 SDK 接口说明

5.1 类型定义

详见 *libzkfptype.h*

本 SDK 接口使用 `__stdcall`。

```
#ifdef _WIN32
#ifdef APICALL
#define APICALL __stdcall
#endif
```

5.1.1 常量

- 1) 模板最大长度
[定义] `#define MAX_TEMPLATE_SIZE 2048`
- 2) 指纹 1:1 阈值参数代码
[定义] `#define FP_THRESHOLD_CODE` 1
- 3) 指纹 1:N 阈值参数代码
[定义] `#define FP_MTHRESHOLD_CODE` 2

5.2 接口说明

5.2.1 ZKFPM_Init

[函数]

```
int APICALL ZKFPM_Init();
```

[功能]

初始化资源，首先调用此接口

[参数]

无

[返回值]

0 表示成功

其他表示失败(见附录)

5.2.2 ZKFPM_Terminate

[函数]

```
int APICALL ZKFPM_Terminate();
```

[功能]

释放资源

[参数]

无

[返回值]

0 表示成功

其他表示失败(见附录)

5.2.3 ZKFPM_GetDeviceCount

[函数]

```
int APICALL ZKFPM_GetDeviceCount();
```

[功能]

获取设备数

[参数]

无

[返回值]

>=0 表示设备数

<0 表示调用失败(见附录)

5.2.4 ZKFPM_OpenDevice

[函数]

```
HANDLE APICALL ZKFPM_OpenDevice (int index);
```

[功能]

打开设备

[参数]

index

设备索引

[返回]

设备操作实例句柄

5.2.5 ZKFPM_CloseDevice

[函数]

```
int APICALL ZKFPM_CloseDevice (HANDLE hDevice);
```

[函数]

关闭设备

[参数]

hDevice

设备操作实例句柄

[返回值]

0 表示成功

其他表示失败(见附录)

5.2.6 ZKFPM_SetParameters

[函数]

```
int APICALL ZKFPM_SetParameters (HANDLE hDevice, int nParamCode, unsigned char* paramValue, unsigned int cbParamValue);
```

[功能]

设置采集器参数

[参数]

hDevice

设备操作实例句柄

nParamCode
参数代码(详见参数代码表)

paramValue
参数值

cbParamValue
参数数据长度

[返回值]
0 表示成功
其他表示失败(见附录)

[其他]

5.2.7 ZKFPM_GetParameters

[函数]
`int APICALL ZKFPM_GetParameters(HANDLE hDevice, int nParamCode, unsigned char* paramValue, unsigned int* cbParamValue);`

[功能]
获取采集器参数

[参数]
hDevice
设备操作实例句柄
nParamCode
参数代码(详见参数代码表)
paramValue [out]
返回参数值
cbParamValue [in/out]
[in] paramValue 分配内存大小
[out] 返回参数值数据大小

[返回值]
0 表示成功
其他表示失败(见附录)

[其他]

5.2.8 ZKFPM_AcquireFingerprint

[函数]
`int APICALL ZKFPM_AcquireFingerprint(HANDLE hDevice, unsigned char* fpImage, unsigned int cbFPImage, unsigned char* fpTemplate, unsigned int* cbTemplate);`

[功能]
采集指纹, 指纹模板

[参数]
hDevice

设备操作实例句柄

fpImage [out]
返回指纹图像

fpImage
fpTemplate 内存大小

fpTemplate [out]
返回指纹模板

cbfpTemplate [in/out]
[in], 预分配 fpTemplate 内存大小, 建议值 MAX_TEMPLATE_SIZE (2048)
[out], 实际返回指纹模板数据大小

[返回值]
0 表示成功
其他表示失败(见附录)

[其他]

5.2.9 ZKFPM_AcquireFingerprintImage

[函数]
int APICALL ZKFPM_AcquireFingerprintImage (HANDLE hDevice, unsigned char* fpImage, unsigned int cbFPImage);

[功能]
采集指纹图像

[参数]
hDevice
设备操作实例句柄

fpImage [out]
返回指纹图像

cbFPImage
fpImage 内存大小

[返回值]
0 表示成功
其他表示失败(见附录)

[其他]

5.2.10 ZKFPM_DBInit

[函数]
HANDLE APICALL ZKFPM_DBInit ();

[功能]
创建算法缓冲区

[参数]
无
[返回值]
缓冲区句柄

5.2.11 ZKFPM_DBFree

[函数]
`int` APICALL ZKFPM_DBFree(HANDLE hDBCACHE);
[功能]
释放算法缓冲区
[参数]
缓冲区句柄
[返回值]
0 表示成功
其他表示失败(见附录)

5.2.12 ZKFPM_DBMerge

[函数]
`int` APICALL ZKFPM_DBMerge(HANDLE hDBCACHE, `unsigned char*` temp1, `unsigned char*` temp2, `unsigned char*` temp3, `unsigned char*` regTemp, `unsigned int*` cbRegTemp);
[功能]
将 3 枚预登记指纹模板合并为一枚登记指纹
[参数]
hDBCACHE
缓冲区句柄
temp1
预登记指纹模板 1
temp2
预登记指纹模板 2
temp3
预登记指纹模板 3
regTemp[out]
登记模板
cbRegTemp[in/out]
[in], 预分配 fpTemplate 内存大小, 建议值 MAX_TEMPLATE_SIZE(2048)
[out], 实际返回指纹模板数据大小
[返回值]
0 表示成功
其他表示失败(见附录)

5.2.13 ZKFPM_DBAdd

[函数]

```
int APICALL ZKFPM_DBAdd(HANDLE hDBCach, unsigned int fid, unsigned char* fpTemplate,
unsigned int cbTemplate);
```

[功能]

添加登记指纹模板到缓冲区

[参数]

hDBCach

缓冲区句柄

fid

指纹 ID (>0 的 32 位无符号整数)

fpTemplate

登记模板

cbTemplate

模板长度

[返回值]

0 表示成功

其他表示失败(见附录)

5.2.14 ZKFPM_DBDeI

[函数]

```
int APICALL ZKFPM_DBDeI (HANDLE hDBCach, unsigned int fid);
```

[功能]

从缓冲区删除指定指纹 ID 的登记模板

[参数]

hDBCach

缓冲区句柄

fid

指纹 ID

[返回值]

0 表示成功

其他表示失败(见附录)

5.2.15 ZKFPM_DBClear

[函数]

```
int APICALL ZKFPM_DBClear (HANDLE hDBCach);
```

[功能]

清空缓冲区

[参数]

hDBCACHE
缓冲区句柄

[返回值]

0 表示成功
其他表示失败(见附录)

5.2.16 ZKFPM_DBCount

[函数]

```
int APICALL ZKFPM_DBCount(HANDLE hDBCACHE, unsigned int* fpCount);
```

[功能]

获取缓冲区指纹数

[参数]

hDBCACHE
缓冲区句柄
fpCount [out]
指纹数

[返回值]

0 表示成功
其他表示失败(见附录)

[其他]

5.2.17 ZKFPM_DBIdentify

[函数]

```
int APICALL ZKFPM_DBIdentify(HANDLE hDBCACHE, unsigned char* fpTemplate, unsigned int cbTemplate, unsigned int* FID, unsigned int* score);
```

[功能]

指纹 1:N 识别

[参数]

hDBCACHE
缓冲区句柄
fpTemplate
指纹模板
cbfpTemplate
指纹模板数据长度
FID [out]
返回指纹 ID
Score [out]
返回比对分数

[返回值]

0 表示成功
其他表示失败(见附录)

5.2.18 ZKFPM_DBMatch

[函数]

```
int APICALL ZKFPM_DBMatch(HANDLE hDBCach, unsigned char* fpTemplate1, unsigned int  
cbfpTemplate1, unsigned char* fpTemplate2, unsigned int cbfpTemplate2);
```

[功能]

比对两枚指纹是否匹配

[参数]

hDBCach

缓冲区句柄

fpTemplate1

指纹模板 1

cbfpTemplate1

指纹模板 1 数据长度

fpTemplate2

指纹模板 2

cbfpTemplate2

指纹模板 2 数据长度

[返回值]

>=0 比对分数

<0 错误 (见附录)

5.2.19 ZKFPM_ExtractFromImage

[函数]

```
ZKINTERFACE int APICALL ZKFPM_ExtractFromImage(HANDLE hDBCach, const char*  
lpFilePathName, unsigned int DPI, unsigned char* fpTemplate, unsigned int *cbTemplate);
```

[功能]

从 BMP、JPG 文件提取模板

[参数]

hDBCach

缓冲区句柄

lpFilePathName

文件全路径

DPI

图像 DPI

fpTemplate

指纹模板

cbfpTemplate

指纹模板 1 数据长度

[返回值]

0 表示成功

其他表示失败(见附录)

[其他]

仅标准版支持该功能

5.2.20 ZKFPM_AcquireFingerprintImage

[函数]

```
ZKINTERFACE int APICALL ZKFPM_AcquireFingerprintImage(HANDLE hDevice, unsigned char* fpImage, unsigned int cbFPImage);
```

[功能]

采集指纹图像

[参数]

hDevice

设备句柄

fpImage

指纹图像

cbFPImage

指纹图像数据长度

[返回值]

0 表示成功

其他表示失败(见附录)

[其他]

5.2.21 ZKFPM_Base64ToBlob

[函数]

```
ZKINTERFACE int APICALL ZKFPM_Base64ToBlob(const char* src, unsigned char* blob, unsigned int cbBlob);
```

[功能]

Base64 字符串转字节流

[参数]

src

Base64 字符串

blob

字节流指针

cbBlob

字节流长度

[返回值]

0 表示成功
其他表示失败(见附录)
[其他]

5.2.22 ZKFPM_BlobToBase64

[函数]

```
ZKINTERFACE int APICALL ZKFPM_BlobToBase64(const unsigned char* src, unsigned int  
cbSrc, char* base64Str, unsigned int cbBase64str);
```

[功能]

字节流转 Base64 字符串

[参数]

src

字节流

blob

字节流数据长度

base64Str

字符串缓冲区

cbBase64str

字符串缓冲区长度

[返回值]

0 表示成功

其他表示失败(见附录)

[其他]

5.2.23 ZKFPM_BlobToBase64

[函数]

```
ZKINTERFACE int APICALL ZKFPM_BlobToBase64(const unsigned char* src, unsigned int  
cbSrc, char* base64Str, unsigned int cbBase64str);
```

[功能]

字节流转 Base64 字符串

[参数]

src

字节流

blob

字节流数据长度

base64Str

字符串缓冲区

cbBase64str

字符串缓冲区长度
[返回值]
0 表示成功
其他表示失败(见附录)

6 附录

6.1附录 1

常用参数代码表

参数代码	属性	数据类型	描述
1	只读	Int	图像宽
2	只读	Int	图像高
3	读写(目前只有 LIVEID20R 可写)	Int	图像 DPI(儿童建议设置 750/1000)
106	只读	Int	图像数据大小
1015	只读	4 字节 Byte 数组	VID&PID(前 2 字节 VID,后 2 字节 PID)
2002	读写(目前只有 LIVEID20R 支持)	Int	防假开关(1 打开/0 关闭)
2004	只读	Int	低五位全为 1 表示真手指(value&31==31)
1101	只读	String	厂商信息
1102	只读	String	产品名
1103	只读	String	设备序列号
101	只写(非 LIVE20R 需调用 关闭)	Int	1 表示闪白灯;0 表示 关闭
102	只写(非 LIVE20R 需调用 关闭)	Int	1 表示闪绿灯;0 表示 关闭
103	只写(非 LIVE20R 需调用 关闭)	Int	1 表示闪红灯;0 表示 关闭
104	只写(LIVE20R 不支持)	Int	1 表示开启蜂鸣;0 表 示关闭
10001	读写(仅 ISO/ANSI 版本支 持)	Int	0 表示 ANSI;1 表示 ISO

6.2附录 2

错误返回值描述

0	操作成功
1	已经初始化
-1	初始化算法库失败
-2	初始化采集库失败
-3	未连接设备
-4	接口暂不支持
-5	无效参数
-6	打开设备失败
-7	无效句柄
-8	取像失败
-9	提取指纹模板失败
-10	中断操作
-11	内存不足
-12	当前正在采集(设备繁忙)
-13	添加指纹模板到内存失败
-14	删除指纹模板失败
-17	操作失败(其他错误)
-18	取消采集
-20	比对指纹失败(登记过程中按不同的手指，或者按压不合理导致差异太大)
-22	合并登记指纹模板失败
-23	文件打开失败
-24	处理图像失败