# iOS LCOpenSDK开发指南

**修订历史（Revision history）**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 编号 | 修订内容描述 | 修订人 | 修订日期 | 修订后版本号 | 批准人 |
| 1 | 初稿 | 39890 | 2019/08/01 | 3.X |  |
| 2 | 修订 | 119607 | 2019/12/25 | 3.X.1 |  |
| 3 | 修订 | 119607 | 2020/08/17 | 3.X.2 |  |
| 4 | 修订 | 119607 | 2020/10/20 | 3.10.1 |  |

# 目录

[iOS LCOpenSDK开发指南 1](#_Toc54771783)

[目录 2](#_Toc54771784)

[1 概述 6](#_Toc54771785)

[2 SDK安装 7](#_Toc54771786)

[2.1 CocoaPods 安装 7](#_Toc54771787)

[2.2 官网下载 8](#_Toc54771788)

[3 主要流程 9](#_Toc54771789)

[3.1无线配网 9](#_Toc54771790)

[3.2实时预览 12](#_Toc54771791)

[3.3录像回放 13](#_Toc54771792)

[3.4语音对讲 15](#_Toc54771793)

[3.5录像下载 16](#_Toc54771794)

[4 SDK接口说明 17](#_Toc54771795)

[4.1 概述 18](#_Toc54771796)

[4.2 LCOpenSDK\_Api 18](#_Toc54771797)

[4.2.1 shareMyInstance 18](#_Toc54771798)

[4.2.2 initOpenApi 18](#_Toc54771799)

[4.2.3 uninitOpenApi 19](#_Toc54771800)

[4.3 LCOpenSDK\_DeviceInit 19](#_Toc54771801)

[4.3.1 searchDeviceInitInfo 19](#_Toc54771802)

[4.3.2 initDevice 19](#_Toc54771803)

[4.4 LCOpenSDK\_PlayWindow 20](#_Toc54771804)

[4.4.1 initPlayWindow 20](#_Toc54771805)

[4.4.2 uninitPlayWindow 20](#_Toc54771806)

[4.4.3 openTouchListener 20](#_Toc54771807)

[4.4.4 getWindowView 20](#_Toc54771808)

[4.4.5 setWindowFrame 21](#_Toc54771809)

[4.4.6 setWindowListener 21](#_Toc54771810)

[4.4.7 getWindowListener 21](#_Toc54771811)

[4.4.8 setSurfaceBGColor 21](#_Toc54771812)

[4.4.9 playRtspReal 21](#_Toc54771813)

[4.4.10 stopRtspReal 22](#_Toc54771814)

[4.4.11 playDeviceRecordByFileName 22](#_Toc54771815)

[4.4.12 playDeviceRecordByUtcTime 23](#_Toc54771816)

[4.4.13 stopDeviceRecord 23](#_Toc54771817)

[4.4.14 playCloudRecord 23](#_Toc54771818)

[4.4.15 stopCloud 24](#_Toc54771819)

[4.4.16 playFile 24](#_Toc54771820)

[4.4.17 stopFile 24](#_Toc54771821)

[4.4.18 playAudio 25](#_Toc54771822)

[4.4.19 stopAudio 25](#_Toc54771823)

[4.4.20 seek 25](#_Toc54771824)

[4.4.21 setPlaySpeed 25](#_Toc54771825)

[4.4.22 pause 26](#_Toc54771826)

[4.4.23 resume 26](#_Toc54771827)

[4.4.24 snapShot 26](#_Toc54771828)

[4.4.25 startRecord 26](#_Toc54771829)

[4.4.26 stopRecord 27](#_Toc54771830)

[4.4.27 setStreamCallback 27](#_Toc54771831)

[4.4.28 doScale 27](#_Toc54771832)

[4.4.29 getScale 27](#_Toc54771833)

[4.4.30 doTranslateX 27](#_Toc54771834)

[4.5 LCOpenSDK\_AudioTalk 28](#_Toc54771835)

[4.5.1 setListener 28](#_Toc54771836)

[4.5.2 getListener 28](#_Toc54771837)

[4.5.3 playTalk 28](#_Toc54771838)

[4.5.4 stopTalk 29](#_Toc54771839)

[4.6 LCOpenSDK\_ConfigWifi 29](#_Toc54771840)

[4.6.1 configWifiStart 29](#_Toc54771841)

[4.6.2 configWifiStop 29](#_Toc54771842)

[4.7 LCOpenSDK\_SoftAP 30](#_Toc54771843)

[4.7.1 startSoftApConfig 30](#_Toc54771844)

[4.8 LCOpenSDK\_ Download 30](#_Toc54771845)

[4.8.1 shareMyInstance 30](#_Toc54771846)

[4.8.2 setListener 30](#_Toc54771847)

[4.8.3 getListener 31](#_Toc54771848)

[4.8.4 startDownload 31](#_Toc54771849)

[4.8.5 3.8.5 startDownload 32](#_Toc54771850)

[4.8.6 stopDownload 32](#_Toc54771851)

[4.9 LCOpenSDK\_Utils 33](#_Toc54771852)

[4.9.1 decryptPic 33](#_Toc54771853)

[3.9.2 decryptPic 33](#_Toc54771854)

[3.10 LCOpenSDK\_Log 34](#_Toc54771855)

[3.10.1 setLogInfo 34](#_Toc54771856)

[3.10.2 closeLog 34](#_Toc54771857)

[3.11 LCOpenSDK\_EventListener（回调接口） 34](#_Toc54771858)

[3.11.1 onControlClick 35](#_Toc54771859)

[3.11.2 onWindowDBClick 35](#_Toc54771860)

[3.11.3 onZoomBegin 35](#_Toc54771861)

[3.11.4 onZooming 35](#_Toc54771862)

[3.11.5 onZoomEnd 36](#_Toc54771863)

[3.11.6 onPlayerResult 36](#_Toc54771864)

[3.11.7 onResolutionChanged 38](#_Toc54771865)

[3.11.8 onReceiveData 38](#_Toc54771866)

[3.11.9 onStreamCallback 38](#_Toc54771867)

[3.11.10 onPlayBegan 38](#_Toc54771868)

[3.11.11 onPlayFinished 39](#_Toc54771869)

[3.11.12 onPlayerTime 39](#_Toc54771870)

[3.11.13 onWindowLongPressBegin 39](#_Toc54771871)

[3.11.14 onWindowLongPressEnd 39](#_Toc54771872)

[3.11.15 onSlipBegin 39](#_Toc54771873)

[3.11.16 onSlipping 40](#_Toc54771874)

[3.11.17 onSlipEnd 40](#_Toc54771875)

[3.11.18 onIVSInfo 40](#_Toc54771876)

[3.12 LCOpenSDK\_DownloadListener（回调接口） 41](#_Toc54771877)

[3.12.1 onDownloadReceiveData 41](#_Toc54771878)

[3.12.2 onDownloadState 41](#_Toc54771879)

[3.13 LCOpenSDK\_TalkerListener（回调接口） 42](#_Toc54771880)

[3.13.1 onTalkResult 42](#_Toc54771881)

# 概述

乐橙是一家专注视频智能硬件、视频云和视频智能技术，为视频应用提供云视频能力的服务商。iOS LCOpenSDK作为一整套接入乐橙云的客户端接入框架，可以使开发者在自身iOS应用中快速集成乐橙云视频能力，本SDK只含真机调试的功能，不支持任何模拟器调试，支持Xcode 9.0以上。

本文档主要从以下几方面对如何集成 SDK进行说明：

|  |  |
| --- | --- |
| **SDK安装** | 介绍如何引入SDK，以及引入后的相关配置。 |
| **主要流程** | 介绍初始化SDK、直播、录像播放、对讲、录像下载、配网主要流程。 |
| [**SDK接口说明**](#_SDK接口说明) | 介绍LCOpenSDK各个接口的功能、参数、返回值以及注意事项。 |

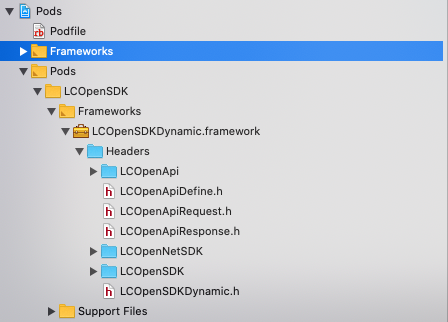
# SDK安装

SDK的安装分为两种方式：cocoapods安装和官网下载（https://open.imoulife.com/developDoc/19）。

## CocoaPods 安装

1.通过CocoaPods引入比较简单，只需要在Podfile文件中添加如下语句

pod 'LCOpenSDK'， 然后执行pod install，如果不能引入则可能是需要更新本地仓库，需执行pod repo update。安装成功后如下图：



2.路径配置

1） Framework路径：Build Setting->Search Paths->Framework Search Paths

添加：

$(PROJECT\_DIR)/Pods/LCOpenSDK/Framework.

2） 头文件路径：Build Setting->Search Paths->Header Search Paths.

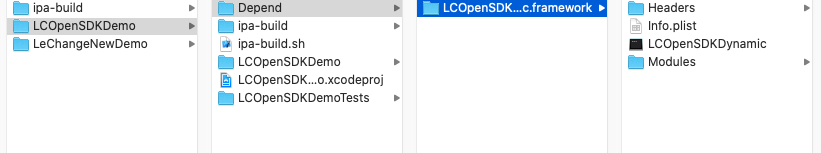
添加：

$(PROJECT\_DIR)/Pods/LCOpenSDK/Framework/LCOpenSDKDynamic.framework/Headers/LCOpenSDK $(PROJECT\_DIR)/Pods/LCOpenSDK/Framework/LCOpenSDKDynamic.framework/Headers/LCOpenNetSDK $(PROJECT\_DIR)/Pods/LCOpenSDK/Framework/LCOpenSDKDynamic.framework/Headers/LCOpenApi

3）Build Settings->Build Options ->Enable Bitcode 设置为NO

## 官网下载

官网下载是通过乐橙云开放平台官网，下载到本地进行手动引入，此时引入时要十分注意SDK的路径，以demo为例，SDK位于工程目录下的Depend文件夹下面。



路径配置：

1. Framework路径：Build Setting->Search Paths->Framework Search Paths

添加：

$(PROJECT\_DIR)/Depend

1. 头文件路径：Build Setting->Search Paths->Framework Search Paths

添加：

$(PROJECT\_DIR)/Depend/LCOpenSDKDynamic.framework/Headers

$(PROJECT\_DIR)/Depend/LCOpenSDKDynamic.framework/Headers/LCOpenSDK

$(PROJECT\_DIR)/Depend/LCOpenSDKDynamic.framework/Headers/LCOpenNetSDK

$(PROJECT\_DIR)/Depend/LCOpenSDKDynamic.framework/Headers/LCOpenApi

1. 路径：Build Phases ->Embed Frameworks

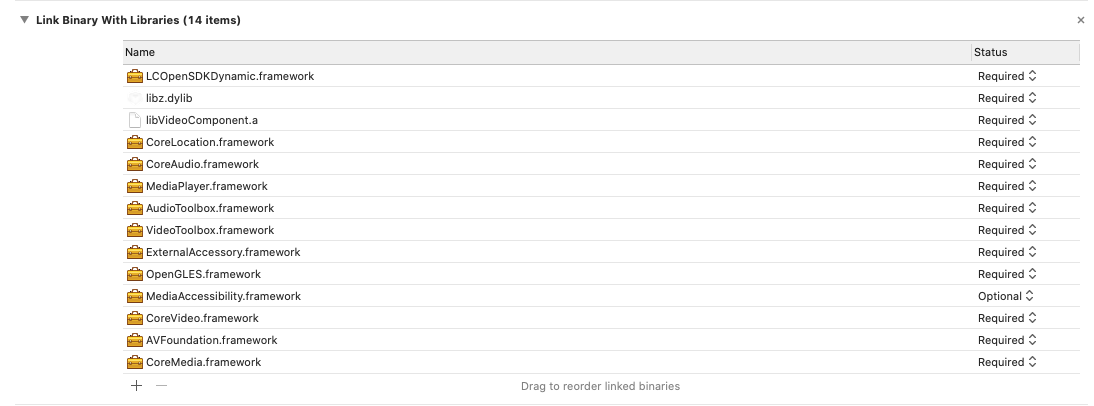
添加：LCOpenSDKDynamic.framework，Destinations选择Frameworks。

注：$(PROJECT\_DIR)为Demo工程所在目录，按要求添加LCOpenSDKDemo和Depend文件夹（对应iOS\_demo和iOS\_sdk文件夹）；Build Phases ->Embed Frameworks路径如果不存在选择左上角加号，然后选择NEW Copy Files Phase进行添加， 最后双击修改名称为Embed Frameworks。

系统库配置：

LCOpenSDK的使用还依赖系统库，所以在使用时我们还需要添加系统库，在工程TARGETS->Build Phases->Link Binary With Libraries下添加如下系统库：

CoreAudio.framework、MediaPlayer.framework、AudioToolbox.framework VideoToolbox.framework、OpenGLES.framework、MediaAccessibility.framework CoreVideo.framework、AVFoundation.framework、 CoreMedia .framework、 libz.tbd（libz.dylib），配置后如下图：



# 主要流程

SDK提供的主要功能包括：实时预览、录像回放（包括设备录像回放和云录像回放）、语音对讲、录像下载（包括设备录像回放和云录像回放）、无线配网。

在使用SDK开发之前首先要前往乐橙云开放平台官网注册开发者账号，获取到AppId和AppSecret。在进行下列功能开发前我们必须要初始化平台信息，这个平台信息就是我们需要对接的服务，国内乐橙云开放平台地址为openapi.lechange.cn:443，海外地址为openapi.easy4ip.com:443，如果开发者需要自己的服务对接乐橙云开放平台，则需要初始化自己的服务，具体代码实现可参考下面：



## 3.1无线配网

无线配网是SDK基础功能之一，和无线配网结合一起使用的还有设备搜索，设备初始化等，无线配网目前SDK支持SmartConfig和软AP配网两种方式，不同设置支持的配网方式不同，具体采用哪种配网方式，需要根据设备型号进行确定。

**一：**SmartConfig配网及添加基本流程

**第一步**：调用开放平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）[查询 设备绑定情况](https://open.imoulife.com/book/zh/http/opendevicemanage/checkDeviceBindOrNot.html)，防止在已经被绑定的情况下进行重新绑定。

**第二步**：调用SDK接口初始化ConfigWifi对象进行SmartConfig配网。

**第三步**：调用SDK接口初始化设备初始化对象进行设备搜索，搜索完成停止配网。

**第四步**：调用SDK接口对于设备初始化状态为1切非SC码设备的设备进行出实话， 上面三步的具体代码实现如下：



**第五步**：调用平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）进行设备绑定。

**二：**软Ap配网

**第一步：**调用开放平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）[查询 设备绑定情况](https://open.imoulife.com/book/zh/http/opendevicemanage/checkDeviceBindOrNot.html)，防止在已经被绑定的情况下进行重新绑定。

**第二步：**调用SDK接口初始化设备初始化对象，搜索设备，获取设备初始化状态。

**第三步：**初始化软Ap配网接口，进行软AP配网， 代码实现如下面：



**第四步：**调用平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）进行设备绑定。

## 3.2实时预览

实时预览是SDK的主要功能之一，开发者可以开放平台接口和SDK接口相结合进行设备实时预览功能的开发，其主要步骤分为以下几步：

**第一步：**通过乐橙云平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）获取设备信息。

**第二步**：调用SDK接口进行播放窗口初始化设置以及监听设置，具体代码实现参考下面：



**第三步**：调用SDK接口进行播放参数设置，然后开始播放，具体代码实现参考下面：



**第四步：**实现监听回调，具体代码实现参考下面：



## 3.3录像回放

录像回放是开对设备存储在SD卡以及硬盘上录像和云存储上录像进行播放的的功能，针对设备录像和云存储录像，SDK提供了不同的接口，使用时要注意。对于设备录像播放SDK提供了两个接口，分别为按文件名播放和按时间播放，使用过程可根据不同场景需要进行使用，推荐使用按文件名播放。

录像回放功能的开发主要分为以下几步：

**第一步：**通过乐橙云平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）获取设备信息。

**第二步：**通过乐橙云平台接口（<https://open.imoulife.com/book/zh/http/cloudstorage/opencloud.html>）获取云存储录像信息 、（https://open.imoulife.com/book/zh/http/openLocalRecord/openLocalRecord.html）获取设备录像信息。

**第三步：**调用SDK接口进行播放窗口初始化设置以及监听设置，具体代码实现参考实时预览第二步。

**第四步**：调用SDK接口进行播放参数设置，然后开始播放，具体代码实现参考下面：

1.设备录像按文件名播放：



2.设备录像按时间播放：



3.云存储录像播放：



**第五步**：实现监听回调，具体代码实现参考下面：



## 3.4语音对讲

语音对讲是设备和客户端进行语音对讲的功能，在设备进行实时预览的同时，进行语音对讲。

语音对讲功能开发主要分为以下几步：

**第一步**：进行实时预览功能开发。

**第二步**：调用SDK接口初始化语音对讲对象，初始化播放参数对象并设置参数开始对讲，具体代码实现参考下面：



**第三步**：实现对讲结果回调，具体代码实现参考下面：



## 3.5录像下载

SDK提供了接口供开发者开发设备录像和云存储录像的下载功能，目前接口仅提供MP4格式的录像文件下载，暂不支持其他格式。

语音对讲功能开发主要分为以下几步：

**第一步：**通过乐橙云平台接口（https://open.imoulife.com/book/zh/http/opendevicemanage/opendevicemanage.html）获取设备信息。

**第二步：**通过乐橙云平台接口（<https://open.imoulife.com/book/zh/http/cloudstorage/opencloud.html>）获取云存储录像信息 、（https://open.imoulife.com/book/zh/http/openLocalRecord/openLocalRecord.html）获取设备录像信息。

**第三步**：调用SDK接口进行初始化下载对象，设置监听对象， 进行录像下载，具体代码实现参考下面：



**第四步**：实现下载回调，具体代码实现参考下面：



# SDK接口说明

## 概述

开放平台SDK以动态Framework的形式发布，提供多个方法类：LCOpenSDK\_Api（平台接口类，结合LCOpenApiClientSDK具体方法头文件使用），LCOpenSDK\_PlayWindow（视频播放类），LCOpenSDK\_AudioTalk（语音对讲类），LCOpenSDK\_ConfigWifi（设备无线配网类），LCOpenSDK\_SoftAp（设备软AP配网类）， LCOpenSDK\_Download（录像下载类），LCOpenSDK\_DeviceInit （设备初始化类），LCOpenSDK\_Utils（公用方法类），LCOpenSDK\_Log（日志接口类），iOS应用只需集成上述方法类，即可快速实现乐橙开放平台所支持的绝大部分业务。

此外，提供LCOpenSDK\_EventListener、LCOpenSDK\_TalkerListener和LCOpenSDK\_DownloadListener三个监听接口类，分别配合LCOpenSDK\_PlayWindow、LCOpenSDK\_AudioTalk和LCOpenSDK\_Download使用，iOS应用可继承并设置相应的监听接口类来接收播放、对讲和下载过程中的业务回调，业务回调包括但不限于：播放窗口事件、视频播放过程中的状态码、语音对讲过程中的状态码、云录像下载过程中的状态码、解码时间戳，回调接口请参考后续说明。

## LCOpenSDK\_Api

### shareMyInstance

|  |  |
| --- | --- |
| 方法 | + (LCOpenSDK\_Api\*) shareMyInstance; |
| 简介 | 获取单例对象 |
| 参数 | 无 |
| 返回值 | 单例指针 |
| 备注 | 无 |

### initOpenApi

|  |  |
| --- | --- |
| 方法 | - (id \_Nonnull)initOpenApi:(LCOpenSDK\_ApiParam \*\_Nonnull)apiParam; |
| 简介 | 接口初始化 |
| 参数 | apiParam LCOpenSDK\_ApiParam对象  LCOpenSDK\_ApiParam 属性：  procotol 协议类型  addr 平台  port 端口  token 授权token |
| 返回值 | self 可不必关注 |
| 备注 | LCOpenSDK\_Api初始化接口，在LCOpenSDK\_PlayWindow、LCOpenSDK\_AudioTalk或LCOpenSDK\_Download类接口之前必须进行调用初始化。 |

### uninitOpenApi

|  |  |
| --- | --- |
| 方法 | - (void)uninitOpenApi; |
| 简介 | initOpenApi反向接口，释放内部资源 |
| 参数 | 无 |
| 返回值 | 无 |
| 备注 | 无 |

## LCOpenSDK\_DeviceInit

### searchDeviceInitInfo

|  |  |
| --- | --- |
| 方法 | - (void)searchDeviceInitInfo:(NSString\*)deviceID  timeOut:(int)timeOut  success:(void(^)(LCOPENSDK\_DEVICE\_INIT\_INFO info))success; |
| 简介 | 搜索设备初始化信息 |
| 参数 | deviceID 待搜索的设备序列号  timeOut 搜索超时时长  success 语句块，返回设备信息，包括设备Mac地址、设备初始化状态、大华私有协议端口和设备IP |
| 返回值 | 无 |
| 备注 | 该接口必须与设备在同一局域网内调用 |

### ­initDevice

|  |  |
| --- | --- |
| 方法 | -(int)initDevice:(NSString\*)mac  password:(NSString\*)password  ip:(NSString\*)ip; |
| 简介 | 初始化设备 |
| 参数 | mac 设备Mac地址  password 设备初始化密码  ip 设备IP |
| 返回值 | 0表示成功，-1表示参数错误，-2表示失败（具体错误原因需进一步分析SDK日志） |
| 备注 | 该接口必须与设备在同一局域网内调用 |

## LCOpenSDK\_PlayWindow

### initPlayWindow

|  |  |
| --- | --- |
| 方法 | - (id) initPlayWindow:(CGRect) frame  Index:(int) index; |
| 简介 | 初始化播放窗口 |
| 参数 | frame 指定播放窗口坐标  index 播放窗口的唯一标识ID，由外层传入 |
| 返回值 | self |
| 备注 | 初始化播放窗口，需叠加于父窗口之上 |

### uninitPlayWindow

|  |  |
| --- | --- |
| 方法 | - (void) uninitPlayWindow; |
| 简介 | 反初始化接口，释放内部资源 |
| 参数 | 无 |
| 返回值 | 无 |
| 备注 | 无 |

### openTouchListener

|  |  |
| --- | --- |
| 方法 | - (void) openTouchListener:(BOOL)flag; |
| 简介 | 设置播放窗口监听 |
| 参数 | flag YES 开启监听  NO 关闭监听 |
| 返回值 | 无 |
| 备注 | 开启手势监听（单击、双击、缩放、滑动），也可自行实现监听父窗口事件 |

### getWindowView

|  |  |
| --- | --- |
| 方法 | - (UIView\*)getWindowView; |
| 简介 | 获取播放控件 |
| 参数 | 无 |
| 返回值 | 播放控件指针 |
| 备注 | 叠加于父窗口上，用于播放渲染 |

### setWindowFrame

|  |  |
| --- | --- |
| 方法 | - (void)setWindowFrame:(CGRect)rect; |
| 简介 | 设置播放窗口的位置和大小 |
| 参数 | rect 表示播放窗口的位置和大小 |
| 返回值 | 无 |
| 备注 | 无 |

### setWindowListener

|  |  |
| --- | --- |
| 方法 | - (void) setWindowListener:(id< LCOpenSDK\_EventListener>) lis; |
| 简介 | 为播放窗口设置监听 |
| 参数 | lis 监听指针 |
| 返回值 | 无 |
| 备注 | 设置监听，监听者需继承LCOpenSDK\_EventListener，可接收播放业务回调及窗口事件回调 |

### getWindowListener

|  |  |
| --- | --- |
| 方法 | - (id< LCOpenSDK\_EventListener>) getWindowListener; |
| 简介 | 获取播放窗口监听者 |
| 参数 | 无 |
| 返回值 | 监听指针 |
| 备注 |  |

### setSurfaceBGColor

|  |  |
| --- | --- |
| 方法 | - (void) setSurfaceBGColor:(UIColor\*)normalColor; |
| 简介 | 设置窗口颜色 |
| 参数 | normalColor 背景颜色 |
| 返回值 | 无 |
| 备注 | 无 |

### playRtspReal

|  |  |
| --- | --- |
| 方法 | - (NSInteger)playRtspReal:(LCOpenSDK\_ParamReal \*)paramReal; |
| 简介 | 实时播放 |
| 参数 | paramReal LCOpenSDK\_ParamReal对象  LCOpenSDK\_ParamReal 属性：  accessToken 授权token  deviceID 设备序列号  channel 通道号  psk 解密密钥  playToken 播放token  defiMode 分辨率模式  isOpt 是否使用长连接优化拉流，YES-使用， NO-不使用 |
| 返回值 | -1 表示参数错误  0 表示调用成功 |
| 备注 | 1、异步播放接口，实际播放结果需根据onPlayerResult回调状态码判断，可参考Demo；  2、psk参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义秘钥，该字段必传。 |

### stopRtspReal

|  |  |
| --- | --- |
| 方法 | - (NSInteger)stopRtspReal:(BOOL)isKeepLastFrame |
| 简介 | 停止实时播放 |
| 参数 | isKeepLastFrame 是否保留最后一帧画面，YES-保存， NO-不保存 |
| 返回值 | 该接口始终返回0 |
| 备注 | 1、如无特殊需求该字段传NO，传YES不会停止渲染，会增加内存占用。 |

### playDeviceRecordByFileName

|  |  |
| --- | --- |
| 方法 | - (NSInteger) playDeviceRecordByFileName:(LCOpenSDK\_ParamDeviceRecordFileName \*)paramDevRecord; |
| 简介 | 按文件名设备录像回放 |
| 参数 | paramDevRecord LCOpenSDK\_ParamDeviceRecordFileName 对象  LCOpenSDK\_ParamDeviceRecordFileName 属性：  accessToken 授权token  deviceID 设备序列号  channel 通道号  psk 解密密钥  playToken 播放token  fileName 设备录像文件名（包含路径）  offsetTime 录像回放起始偏移时间，以秒为单位，传0代表从开始回放录像  isOpt 是否使用长连接优化拉流，YES-使用， NO-不使用 |
| 返回值 | -1 表示参数错误  0 表示调用成功 |
| 备注 | 1、异步播放接口，实际播放结果需根据onPlayerResult回调状态码判断，可参考Demo；  2、psk参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义秘钥，该字段必传。 |

### playDeviceRecordByUtcTime

|  |  |
| --- | --- |
| 方法 | - (NSInteger) playDeviceRecordByUtcTime:(LCOpenSDK\_ParamDeviceRecordUTCTime \*)paramDevRecord; |
| 简介 | 按时间设备录像回放 |
| 参数 | paramDevRecord LCOpenSDK\_ParamDeviceRecordUTCTime 对象  LCOpenSDK\_ParamDeviceRecordUTCTime 属性：  accessToken 授权token  deviceID 设备序列号  channel 通道号  psk 解密密钥  playToken 播放token  defiMode 分辨率模式  beginTime 设备录像开始时间  endTime 设备录像结束时间  isOpt 是否使用长连接优化拉流，YES-使用， NO-不使用 |
| 返回值 | -1 表示参数错误  0 表示调用成功 |
| 备注 | 1、异步播放接口，实际播放结果需根据onPlayerResult回调状态码判断，可参考Demo；  2、psk参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义秘钥，该字段必传。  3、playDeviceRecordByFileName和 playDeviceRecordByUtcTime，推荐使用playDeviceRecordByFileName接口，easy4ip设备和p2p设备只能调用playDeviceRecordByUtcTime。 |

### stopDeviceRecord

|  |  |
| --- | --- |
| 方法 | - (NSInteger)stopDeviceRecord:(BOOL)isKeepLastFrame; |
| 简介 | 停止设备录像回放 |
| 参数 | isKeepLastFrame 是否保留最后一帧画面，YES-保存， NO-不保存 |
| 返回值 | 该接口始终返回0 |
| 备注 | 1、如无特殊需求该字段传NO，传YES不会停止渲染，会增加内存占用。 |

### playCloudRecord

|  |  |
| --- | --- |
| 方法 | - (NSInteger)playCloudRecord:(LCOpenSDK\_ParamCloudRecord \*)paramCloud; |
| 简介 | 云录像回放 |
| 参数 | paramCloud LCOpenSDK\_ParamCloudRecord 对象  LCOpenSDK\_ParamCloudRecord 属性：  accessToken 授权token  deviceID 设备序列号  channel 通道号  psk 解密密钥  playToken 播放token  recordRegionID 分辨率模式  offsetTime 设备录像开始时间  recordType 云录像类型  offsetTime 录像回放起始偏移时间，以秒为单位，传0代表从开始回放录像  timeOut 下载云录像切片的超时时长，单位秒 |
| 返回值 | -1 表示参数错误  0 表示调用成功 |
| 备注 | 1、异步播放接口，实际播放结果需根据onPlayerResult回调状态码判断，可参考Demo；  2、psk参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义秘钥，该字段必传。 |

### stopCloud

|  |  |
| --- | --- |
| 方法 | -(void) stopCloud; |
| 简介 | 停止云录像回放 |
| 参数 | isKeepLastFrame 是否保留最后一帧画面，YES-保存， NO-不保存 |
| 返回值 | 该接口始终返回0 |
| 备注 | 1、如无特殊需求该字段传NO，传YES不会停止渲染，会增加内存占用。 |

### playFile

|  |  |
| --- | --- |
| 方法 | - (NSInteger)playFile:(NSString\*)fileName; |
| 简介 | 播放本地文件 |
| 参数 | filename 文件名称， 包含路径 |
| 返回值 | -1 表示参数错误  0 表示调用成功 |
| 备注 |  |

### stopFile

|  |  |
| --- | --- |
| 方法 | - (NSInteger)stopFile:(BOOL)isKeepLastFrame; |
| 简介 | 停止本地文件播放 |
| 参数 | isKeepLastFrame 是否保留最后一帧画面，YES-保存， NO-不保存 |
| 返回值 | 该接口始终返回0 |
| 备注 | 1、如无特殊需求该字段传NO，传YES不会停止渲染，会增加内存占用。 |

### playAudio

|  |  |
| --- | --- |
| 方法 | - (NSInteger) playAudio; |
| 简介 | 播放声音 |
| 参数 | 无 |
| 返回值 | -1 表示失败  0 表示成功 |
| 备注 | 无 |

### stopAudio

|  |  |
| --- | --- |
| 方法 | - (NSInteger) stopAudio; |
| 简介 | 关闭声音 |
| 参数 | 无 |
| 返回值 | -1 表示失败  0 表示成功 |
| 备注 | 无 |

### seek

|  |  |
| --- | --- |
| 方法 | - (NSInteger) seek:(NSInteger)timeInfo; |
| 简介 | 录像回放定位功能 |
| 参数 | timeInfo 相对录像起始时间的偏移秒数 |
| 返回值 | -1 表示内部错误  0 表示调用成功 |
| 备注 | 异步接口，实际是否定位成功需根据onPlayerResult回调状态码判断，参考RTSP（设备录像）和HLS（云录像）业务码定义。 |

### setPlaySpeed

|  |  |
| --- | --- |
| 方法 | - (NSInteger)setPlaySpeed:(float)speed; |
| 简介 | 设置录像播放倍速 |
| 参数 | speed 倍速 |
| 返回值 | -1 表示内部错误  0 表示调用成功 |
| 备注 | 1.本地录像有些设备不支持，设置后可能没有效果。 |

### pause

|  |  |
| --- | --- |
| 方法 | - (NSInteger) pause; |
| 简介 | 暂停录像回放 |
| 参数 | 无 |
| 返回值 | -1 表示内部错误  0 表示调用成功 |
| 备注 | 异步接口，实际是否定位成功需根据onPlayerResult回调状态码判断，参考RTSP（设备录像）和HLS（云录像）业务码定义。 |

### resume

|  |  |
| --- | --- |
| 方法 | - (NSInteger) resume; |
| 简介 | 恢复录像回放（pause反向接口） |
| 参数 | 无 |
| 返回值 | -1 表示内部错误  0 表示调用成功 |
| 备注 | 异步接口，实际是否定位成功需根据onPlayerResult回调状态码判断，参考RTSP（设备录像）和HLS（云录像）业务码定义。 |

### snapShot

|  |  |
| --- | --- |
| 方法 | - (NSInteger) snapShot:(NSString \*)filePath; |
| 简介 | 本地抓图 |
| 参数 | filePath 图片文件名（包含路径） |
| 返回值 | -1 表示失败  0 表示成功 |
| 备注 | 播放过程中可调用该接口将一帧视频保存到本地JPG图片 |

### startRecord

|  |  |
| --- | --- |
| 方法 | - (NSInteger) startRecord:(NSString \*)filePath  recordType: (NSInteger)nRecordType; |
| 简介 | 开始本地录制 |
| 参数 | filePath 录像文件名（包含路径）  nRecordType 0：Dav格式  1：MP4格式 |
| 返回值 | -1 表示失败  0 表示成功 |
| 备注 | 播放过程中可调用该接口将视频录制到本地。 |

### stopRecord

|  |  |
| --- | --- |
| 方法 | - (NSInteger) stopRecord; |
| 简介 | 停止本地录制 |
| 参数 | 无 |
| 返回值 | -1 表示失败  0 表示成功 |
| 备注 | 与startRecord结合使用 |

### setStreamCallback

|  |  |
| --- | --- |
| 方法 | - (void)setStreamCallback:(LC\_OUTPUT\_STREAM\_FORMAT)streamFormat; |
| 简介 | 设置标准流回调 |
| 参数 | streamFormat 输出的标准流格式，目前支持PS和TS标准流 |
| 返回值 | 无 |
| 备注 | 1、可在播放前或播放过程中调用该接口开启标准流回调的功能；  2、标准流通过[LCOpenSDK\_EventListener::onStreamCallback](#_onStreamCallback)接口导出，每次导出一帧视频或音频，第一帧为视频I帧；  3、当不需要标准流回调功能时，可通过设置输出的标准流格式为LC\_OUTPUT\_STREAM\_FORMAT\_NULL来关闭之，以节省性能。 |

### doScale

|  |  |
| --- | --- |
| 方法 | - (void) doScale:(float)scale; |
| 简介 | 设置视频显示缩放比例 |
| 参数 | scale 相对缩放比例 |
| 返回值 | 无 |
| 备注 | 可使用onZooming回调的缩放比例 |

### getScale

|  |  |
| --- | --- |
| 方法 | - (CGFloat) getScale; |
| 简介 | 获取缩放比例 |
| 参数 | 无 |
| 返回值 | 相对正常视频画面的缩放比例 |
| 备注 |  |

### doTranslateX

|  |  |
| --- | --- |
| 方法 | - (void) doTranslateX:(float)x Y:(float)y; |
| 简介 | 窗口移动 |
| 参数 | x 目的位置点距离屏幕左上角的绝对横坐标  y 目的位置点距离屏幕左上角的绝对纵坐标 |
| 返回值 | 无 |
| 备注 | 视频画面被放大后，可调用此接口移动显示窗口 |

## LCOpenSDK\_AudioTalk

### setListener

|  |  |
| --- | --- |
| 方法 | - (void) setListener:(id<LCOpenSDK\_TalkerListener>) lis; |
| 简介 | 为语音对讲设置监听 |
| 参数 | lis 监听指针 |
| 返回值 | 无 |
| 备注 | 监听者需继承LCOpenSDK\_TalkerListener类，可接收语音对讲的状态码回调 |

### getListener

|  |  |
| --- | --- |
| 方法 | - (id<LCOpenSDK\_TalkerListener>) getListener; |
| 简介 | 获取语音对讲监听者 |
| 参数 | 无 |
| 返回值 | 监听指针 |
| 备注 | 无 |

### playTalk

|  |  |
| --- | --- |
| 方法 | - (NSInteger) playTalk:(LCOpenSDK\_ParamTalk \*)paramTalk; |
| 简介 | 开启语音对讲 |
| 参数 | paramTalk LCOpenSDK\_ParamTalk 对象  LCOpenSDK\_ParamTalk 属性  accessToken 授权token  deviceID 设备序列号  channel 通道号  psk 解密密钥  playToken 播放token  isOpt 是否使用长连接优化拉流，YES-使用， NO-不使用 |
| 返回值 | -1 表示参数错误  0 表示调用成功 |
| 备注 | 1、异步接口，实际是否对讲成功需根据onTalkResult回调状态码判断，可参考Demo；  2、psk参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义秘钥，该字段必传。 |

### stopTalk

|  |  |
| --- | --- |
| 方法 | - (NSInteger) stopTalk; |
| 简介 | 停止语音对讲 |
| 参数 | 无 |
| 返回值 | 该接口始终返回0 |
| 备注 | 无 |

## LCOpenSDK\_ConfigWifi

### configWifiStart

|  |  |
| --- | --- |
| 方法 | -(NSInteger)configWifiStart:(NSString\*)devId  ssid:(NSString\*)ssid  password:(NSString\*)pwd  secure:(NSString\*)security  vciceFreq:(NSInteger)vciceFreq; |
| 简介 | 开启无线配对 |
| 参数 | devId 设备序列号  ssid 热点名称  pwd 热点密码  security 加密类型（wpa,wpa-psk,wpa2,wpa2-psk等），可填空  voiceFreq 声波配网声波的频率(11000-17000) |
| 返回值 | -1 表示失败  0 表示成功 |
| 备注 | 1、该接口用于设备无线配网，配对成功后待设备连上开放平台才能进行绑定或其他操作；  2、该接口返回值仅表示接口调用成功或失败，实际配网结果需要根据搜索设备是否在同一局域网下来判定。 |

### configWifiStop

|  |  |
| --- | --- |
| 方法 | -(NSInteger)configWifiStop; |
| 简介 | 停止无线配对 |
| 参数 | 无 |
| 返回值 | 该接口始终返回0 |
| 备注 | 收到配网事件回调后，不论成功或失败，都需调用此接口停止无线配对 |

## LCOpenSDK\_SoftAP

### startSoftApConfig

|  |  |
| --- | --- |
| 方法 | -(NSInteger) startSoftAPConfig:(NSString\*)wifiName  wifiPwd:(NSString\*)wifiPwd  deviceId:(NSString\*)deviceId  devicePwd:(NSString\*)devicePwd  isSC:(BOOL)isSC; |
| 简介 | 开始软AP配网 |
| 参数 | wifiName WiFi名称  wifiPwd WiFi 登录密码  deviceId 设备序列号  devicePwd 设备密码  isSC 是否为安全码设备 |
| 返回值 | 0 开启成功  -1 必传参数为空  -2 搜索设备失败  -3 设备被未搜索到参数中的WiFi信息 |
| 备注 | 1. 是否为安全码设备，可通过设备背面信息中是否有8位安全码确认，有isSC为YES， 否则为NO。   2、devicePwd 安全码设备可传空字符串，K5锁需要传admin， 其他传设备登录密码。 |

## LCOpenSDK\_ Download

### shareMyInstance

|  |  |
| --- | --- |
| 方法 | + (LCOpenSDK\_Download\* )shareMyInstance |
| 简介 | 获取单例指针 |
| 参数 | 无 |
| 返回值 | 单例指针 |
| 备注 | 无 |

### setListener

|  |  |
| --- | --- |
| 方法 | -(void) setListener:(id<LCOpenSDK\_DownloadListener>) listener |
| 简介 | 为下载设置监听 |
| 参数 | listener 监听指针 |
| 返回值 | 无 |
| 备注 | 监听者需继承LCOpensdk\_DownloadListener类，可接收云录像和设备录像下载的状态码回调 |

### getListener

|  |  |
| --- | --- |
| 方法 | - (id<LCOpenSDK\_DownloadListener>)getListener; |
| 简介 | 获取下载监听者 |
| 参数 | 无 |
| 返回值 | 监听指针 |
| 备注 | 无 |

### startDownload

|  |  |
| --- | --- |
| 方法 | - (NSInteger)startDownload:(NSInteger)index filepath:(NSString\*)filepath  token:(NSString\*)accessTok devID:(NSString\*)deviceID  channelID:(NSInteger)channelID  psk:(NSString\*)psk  recordRegionId:(NSString \*)recordRegionId  Type:(NSInteger)type  Timeout:(NSInteger)timeout; |
| 简介 | 开启（云录像）下载 |
| 参数 | index 该路下载的唯一标示ID  filepath 下载文件名（包含路径）  accessTok 授权token  deviceID 设备序列号  channelID 设备通道号  psk 解密密钥  recordRegionId 云录像ID  type 云录像类型，报警云录像传1000，定时云录像传2000  timeout 下载云录像切片的超时时长，单位秒 |
| 返回值 | -1 接口调用失败  0 接口调用成功 |
| 备注 | 1、异步接口，实际下载状态需根据监听接口onDownloadState回调状态码判断，可参考demo；  2、psk参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义密钥，此字段必传。  3、报警云录像ID和定时云录像ID分别通过<https://open.imoulife.com/developDoc/8>中QueryCloudRecords和QueryCloudPlanRecords接口请求查询。 |

### 3.8.5 startDownload

|  |  |
| --- | --- |
| 方法 | - (NSInteger)startDownload:(NSInteger)index filepath:(NSString\*)filepath  token:(NSString\*)token  devID:(NSString\*)deviceID decryptKey:(NSString\*)decryptKey  fileID:(NSString\*)fileID  speed:(double)speed; |
| 简介 | 开启（设备录像）下载 |
| 参数 | index 该路下载的唯一标示ID  filepath 下载文件名（包含路径）  token 授权token  deviceID 设备序列号  decryptKey 解密密钥  psk 解密密钥  fileID 录像文件ID  speed 下载速度，一般设为16. |
| 返回值 | -1 接口调用失败  0 接口调用成功 |
| 备注 | 1、异步接口，实际下载状态需根据监听接口onDownloadState回调状态码判断，可参考demo；  2、decryptKey 参数传空指针或空字符串时使用deviceID作为解密秘钥，用户自定义密钥，此字段必传。  3、报警录像ID通过<https://open.imoulife.com/developDoc/9>中QueryLocalRecords接口请求查询。 |

### stopDownload

|  |  |
| --- | --- |
| 方法 | -(BOOL) stopDownload:(NSInteger)index; |
| 简介 | 停止下载（云录像或者设备录像） |
| 参数 | index 该路下载的唯一标示ID |
| 返回值 | YES 接口调用成功  NO 接口调用失败 |
| 备注 | 下载完成或失败，sdk内部会自动调用该接口，只有在终止正在进行的下载任务时才需调用该接口。 |

## LCOpenSDK\_Utils

### decryptPic

|  |  |
| --- | --- |
| 方法 | -(NSInteger)decryptPic:(NSData \*)pSrcBufIn  deviceID:(NSString\*)deviceID  key:(NSString\*)key  bufOut:(NSData\*\*)pDestBufOut; |
| 简介 | 图片解密 |
| 参数 | pSrcBufIn 解密前图片内容  deviceID 设备序列号  key 解密密钥  pDestBufOut 解密后图片数据 |
| 返回值 | 0 表示解密成功  1 表示完整性校验失败  2 表示密钥错误  3 表示图片非加密   1. 不支持的加密方式 2. 缓冲区长度不够   99 内部错误 |
| 备注 | 1、解密密钥默认使用设备序列号；  2、报警图片、云录像缩略图均为加密图片，设备通道缩略图为非加密图片；  3、具体使用方法可参见Demo。 |

### 3.9.2 decryptPic

|  |  |
| --- | --- |
| 方法 | -(NSInteger)decryptPic:(NSData \*)pSrcBufIn  deviceID:(NSString\*)deviceID  key:(NSString\*)key  token:(NSString\*)token  bufOut:(NSData\*\*)pDestBufOut; |
| 简介 | 图片解密（TCM设备） |
| 参数 | pSrcBufIn 解密前图片内容  deviceID 设备序列号  key 解密密钥  token 授权token  pDestBufOut 解密后图片数据 |
| 返回值 | 0 表示解密成功  1 表示完整性校验失败  2 表示密钥错误  3 表示图片非加密   1. 不支持的加密方式 2. 缓冲区长度不够   99 内部错误 |
| 备注 | 1、解密密钥默认使用设备序列号；  2、报警图片、云录像缩略图均为加密图片，设备通道缩略图为非加密图片；  3、具体使用方法可参见Demo。 |

## LCOpenSDK\_Log

### 3.10.1 setLogInfo

|  |  |
| --- | --- |
| 方法 | - (void)setLogInfo:(LCOpenSDK\_LogInfo \*)logInfo; |
| 简介 | 设置日志信息 |
| 参数 | logInfo LCOpenSDK\_LogInfo 对象  LCOpenSDK\_LogInfo 属性：  levelType 日志等级类型 |
| 返回值 | 无 |
| 备注 | 设置日志等级为Fatal不能关闭播放过程中和播放库调用系统硬件的相关debug日志打印。 |

### 3.10.2 closeLog

|  |  |
| --- | --- |
| 方法 | - (void) closeLog; |
| 简介 | 关闭日志打印 |
| 参数 | 无 |
| 返回值 | 无 |
| 备注 | 此接口不能关闭播放过程中和播放库调用系统硬件的相关debug日志打印。 |

## LCOpenSDK\_EventListener（回调接口）

上层如需接收视频播放类的状态回调或窗口事件回调，需遵守LCOpenSDK\_EventListener协议，不关心的事件可空实现，通过LCOpenSDK\_PlayWindow::setWindowListener接口设置监听者即可。

### onControlClick

|  |  |
| --- | --- |
| 方法 | - (void) onControlClick:(CGFloat)dx dy:(CGFloat)dy  Index:(NSInteger)index; |
| 简介 | 播放窗口单击事件 |
| 参数 | index 播放窗口的唯一标识ID  dx 单击点相对屏幕左上角的绝对横坐标  dy 单击点相对屏幕左上角的绝对纵坐标 |
| 返回值 | 无 |
| 备注 | 实现即可接收播放窗口点击事件。 |

### onWindowDBClick

|  |  |
| --- | --- |
| 方法 | - (void) onWindowDBClick:(CGFloat)dx dy:(CGFloat)dy  Index:(NSInteger)index; |
| 简介 | 播放窗口双击事件 |
| 参数 | index 播放窗口的唯一标识ID  dx 单击点相对屏幕左上角的绝对横坐标  dy 单击点相对屏幕左上角的绝对纵坐标 |
| 返回值 | 无 |
| 备注 | 实现即可接收播放窗口双击事件。 |

### onZoomBegin

|  |  |
| --- | --- |
| 方法 | - (void) onZoomBegin:(NSInteger) index; |
| 简介 | 手势缩放事件开始的回调 |
| 参数 | index 播放窗口的唯一标识ID |
| 返回值 | 无 |
| 备注 | 不关注此事件，可不实现。 |

### onZooming

|  |  |
| --- | --- |
| 方法 | - (void) onZooming:(CGFloat)scale Index:(NSInteger)index; |
| 简介 | 手势缩放事件中的回调 |
| 参数 | index 播放窗口的唯一标识ID  scale 相对缩放比例 |
| 返回值 | 无 |
| 备注 | 不关注此事件，可不实现。 |

### onZoomEnd

|  |  |
| --- | --- |
| 方法 | - (void) onZoomEnd:(ZoomType) zoom Index:(NSInteger)index; |
| 简介 | 手势缩放事件结束的回调 |
| 参数 | index 播放窗口的唯一标识ID  zoom 放大or缩小 |
| 返回值 | 无 |
| 备注 | 不关注此事件，可不实现。 |

### onPlayerResult

|  |  |
| --- | --- |
| 方法 | - (void) onPlayerResult:(NSString \*)code Type:(NSInteger)type  Index:(NSInteger)index; |
| 简介 | 视频播放的状态码回调 |
| 参数 | index 播放窗口的唯一标识ID  code 状态码  type 业务类型 |
| 返回值 | 无 |
| 备注 | type: 0 RESULT\_PROTO\_TYPE\_RTSP;  1 RESULT\_PROTO\_TYPE\_HLS;  3 RESULT\_PROTO\_TYPE\_NETSDK;  5 私有协议优化拉流;  99 OPENAPI;  **实时播放和设备录像回放（type=0）的状态码code:**  STATE\_PACKET\_FRAME\_ERROR = 0; 组帧失败，错误状态  STATE\_RTSP\_TEARDOWN\_ERROR = 1; 内部要求关闭,如连接断开等，错误状态  STATE\_RTSP\_DESCRIBE\_READY = 2; 会话已经收到Describe响应，连接建立中  STATE\_RTSP\_AUTHORIZATION\_FAIL = 3; RTSP鉴权失败，错误状态  STATE\_RTSP\_PLAY\_READY = 4; 收到PLAY响应，连接成功  STATE\_RTSP\_FILE\_PLAY\_OVER = 5; 录像文件回放正常结束  STATE\_RTSP\_PAUSE\_READY = 6; 收到PAUSE响应  STATE\_RTSP\_KEY\_MISMATCH = 7; 解密秘钥错误  STATE\_RTSP\_SERVICE\_UNAVAILABLE = 99; 基于503错误码的连接最大数错误，错误状态  STATE\_RTSP\_USER\_INFO\_BASE\_START = 100; 用户信息起始码，服务端上层传过来的信息码会在该起始码基础上累加，错误状态  **云录像回放（type=1）的状态码code:**  HLS\_DOWNLOAD\_FAILD = 0; 下载失败  HLS\_DOWNLOAD\_BEGIN = 1; 开始下载  HLS\_DOWNLOAD\_END = 2; 下载结束  HLS\_SEEK\_SUCCESS = 3; 定位成功  HLS\_SEEK\_FAILD = 4; 定位失败  HLS\_ABORT\_DONE = 5; 停止下载  HLS\_RESUME\_DONE = 6; 恢复下载成功  HLS\_DOWNLOAD\_TIMEOUT = 7; 下载超时  HLS\_KEY\_ERROR = 11; 解密秘钥错误  **优化拉流 (type=5)的状态吗code:**  STATE\_DHHTTP\_COMPONENT\_ERROR = -1, 调流媒体失败  STATE\_DHHTTP\_START = 0, 开始拉流  STATE\_DHHTTP\_OK = 1000, 开启播放成功  STATE\_DHHTTP\_PLAY\_FILE\_OVER = 2000, 回放当前文件播放完毕  STATE\_DHHTTP\_PAUSE\_OK = 4000, 子链路正常关闭  STATE\_DHHTTP\_BAD\_REQUEST = 400000, 非法请求,关闭客户端  STATE\_DHHTTP\_UNAUTHORIZED=401000, 未授权,用户名密码错误  STATE\_DHHTTP\_FORBIDDEN=403000, 禁止访问，关闭客户端  STATE\_DHHTTP\_NOTFOUND = 404000, 未找到，关闭客户端  STATE\_DHHTTP\_REQ\_TIMEOUT = 408000, 请求超时，指拉流成功，但后续网络异常，导致拉流断开  STATE\_DHHTTP\_SERVER\_ERROR = 500000, 服务器内部错误，关闭客户端  STATE\_DHHTTP\_SERVER\_UNVALILABLE = 503000, 服务不可用  STATE\_DHHTTP\_FLOWLIMIT = 503006, mts限流  STATE\_DHHTTP\_P2P\_MAXCONNECT = 503007, p2p达到最大链接数  STATE\_DHHTTP\_CHECK\_FAILED = 503008, 对讲错误  STATE\_DHHTTP\_BUSY\_LINE = 503009, 对讲忙线  STATE\_DHHTTP\_GATEWAY\_TIMEOUT = 504000, 网络不通  STATE\_DHHTTP\_CLIENT\_ERROR = 1000000, 客户端内部错误，一 般代码逻辑错误  STATE\_DHHTTP\_KEY\_ERROR = 1000005, 客户端密钥和服务端密钥不一致  **OpenApi交互（type = 99）的状态码code:**  -1000; HTTP交互出错或超时  其他参考<https://open.lechange.com/developDoc/26>  本地录像暂停会回调4000，这是代表成功。  接收到错误回调后需关闭会话，具体判断和处理可参考Demo |

### onResolutionChanged

|  |  |
| --- | --- |
| 方法 | - (void) onResolutionChanged:(NSInteger)width  Height:(NSInteger)height Index:(NSInteger)index; |
| 简介 | 分辨率改变的回调 |
| 参数 | Index 播放窗口的唯一标识ID  width 宽  height 高 |
| 返回值 | 无 |
| 备注 | 无 |

### onReceiveData

|  |  |
| --- | --- |
| 方法 | - (void) onReceiveData:(NSInteger)len Index:(NSInteger)index; |
| 简介 | 接收码流数据长度的回调 |
| 参数 | index 播放窗口的唯一标识ID  len 数据长度 |
| 返回值 | 无 |
| 备注 | 可根据len计算码流速率。 |

### onStreamCallback

|  |  |
| --- | --- |
| 方法 | - (void) onStreamCallback:(NSData\*)data Index:(NSInteger)index; |
| 简介 | 标准流导出的回调 |
| 参数 | data PS/TS标准流数据  index 播放窗口的唯一标识ID |
| 返回值 | 无 |
| 备注 |  |

### onPlayBegan

|  |  |
| --- | --- |
| 方法 | - (void) onPlayBegan:(NSInteger)index; |
| 简介 | 码流开始播放的回调 |
| 参数 | index 播放窗口的唯一标识ID |
| 返回值 | 无 |
| 备注 |  |

### onPlayFinished

|  |  |
| --- | --- |
| 方法 | - (void) onPlayFinished:(NSInteger)index; |
| 简介 | 录像回放结束的回调 |
| 参数 | index 播放窗口的唯一标识ID |
| 返回值 | 无 |
| 备注 | 设备录像和云录像回放结束时都会触发此回调。 |

### onPlayerTime

|  |  |
| --- | --- |
| 方法 | - (void) onPlayerTime:(long)time Index:(NSInteger)index; |
| 简介 | 解码时间戳的回调 |
| 参数 | index 播放窗口的唯一标识ID  time UTC时间 |
| 返回值 | 无 |
| 备注 |  |

### onWindowLongPressBegin

|  |  |
| --- | --- |
| 方法 | - (void) onWindowLongPressBegin:(Direction)dir dx:(CGFloat)dx  dy:(CGFloat)dy Index:(NSInteger)index; |
| 简介 | 播放窗口长按事件开始的回调 |
| 参数 | index 播放窗口的唯一标识ID  dir 方向，参考Direction结构，相对于播放窗口的中心点  dx 所在位置相对屏幕左上角的绝对横坐标  dy 所在位置相对屏幕左上角的绝对纵坐标 |
| 返回值 | 无 |
| 备注 |  |

### onWindowLongPressEnd

|  |  |
| --- | --- |
| 方法 | - (void) onWindowLongPressEnd:(NSInteger)index; |
| 简介 | 播放窗口长按事件结束的回调 |
| 参数 | index 播放窗口的唯一标识ID |
| 返回值 | 无 |
| 备注 |  |

### onSlipBegin

|  |  |
| --- | --- |
| 方法 | - (bool) onSlipBegin:(Direction)dir dx:(CGFloat)dx dy:(CGFloat)dy Index:(NSInteger)index; |
| 简介 | 滑动开始的回调 |
| 参数 | index 播放窗口的唯一标识ID  dir 方向，参考Direction结构  dx 滑动开始点相对屏幕左上角绝对横坐标  dy 滑动开始点相对屏幕左上角绝对纵坐标 |
| 返回值 | true/false 不需要关注 |
| 备注 | 可根据此接口实现云台控制功能或播放窗口放大后的拖动，参考demo。 |

### onSlipping

|  |  |
| --- | --- |
| 方法 | - (void) onSlipping:(Direction)dir preX:(CGFloat)preX  preY:(CGFloat)preY dx:(CGFloat)dx dy:(CGFloat)dy  Index:(NSInteger)index; |
| 简介 | 滑动中的回调 |
| 参数 | index 播放窗口的唯一标识ID  dir 方向，参考Direction结构  preX 滑动开始点相对屏幕左上角绝对横坐标  preY 滑动开始点相对屏幕左上角绝对纵坐标  dx 滑动过程中位置点相对屏幕左上角绝对横坐标  dy 滑动过程中位置点相对屏幕左上角绝对纵坐标 |
| 返回值 | 无 |
| 备注 | 可根据此接口实现云台控制功能或播放窗口放大后的拖动，参考demo。 |

### onSlipEnd

|  |  |
| --- | --- |
| 方法 | - (void) onSlipEnd:(Direction)dir dx:(CGFloat)dx dy:(CGFloat)dy  Index:(NSInteger)index; |
| 简介 | 滑动结束的回调 |
| 参数 | index 播放窗口的唯一标识ID  dir 方向，参考Direction结构  dx 滑动结束点相对屏幕左上角绝对横坐标  dy 滑动结束点相对屏幕左上角绝对纵坐标 |
| 返回值 | 无 |
| 备注 | 可根据此接口实现云台控制功能或播放窗口放大后的拖动，参考demo。 |

### onIVSInfo

|  |  |
| --- | --- |
| 方法 | - (void)onIVSInfo:(NSString\*)pIVSBuf  nIVSType:(long)nIVSType  nIVSBufLen:(long)nIVSBufLen  nFrameSeq:(long)nFrameSeq; |
| 简介 | 滑动结束的回调 |
| 参数 | pIVSBuf 辅助帧数据  nIVSType 辅助帧数据类型  nIVSBufLen 助帧数据长度(json或智能帧结构体数据)  nFrameSeq 辅助帧id |
| 返回值 | 无 |
| 备注 | 可根据此接口实现判断云台操作是否到达边界，具体判断逻辑参考demo。 |

## LCOpenSDK\_DownloadListener（回调接口）

### onDownloadReceiveData

|  |  |
| --- | --- |
| 方法 | - (void) onDownloadReceiveData:(NSInteger)index datalen:(NSInteger)datalen; |
| 简介 | 下载数据长度的回调 |
| 参数 | Index 该路下载的唯一标示ID  datalen 下载的数据长度（增量， 单位字节） |
| 返回值 | 无 |
| 备注 | 云录像/设备录像的文件长度可通过<https://open.imoulife.com/developDoc/9> 中[QueryCloudRecords](#_LCOpenApi.QueryCloudRecords_1)接口 和<https://open.imoulife.com/developDoc/8> 中[QueryLocalRecords](#_LCOpenApi.QueryLocalRecords)接口请求）接口查询，datalen的累加值可以反映下载进度。 |

### onDownloadState

|  |  |
| --- | --- |
| 方法 | -(void) onDownloadState:(NSInteger)index code:(NSString \*)code type:(NSInteger)type; |
| 简介 | 下载状态的回调 |
| 参数 | index 该路下载的唯一标示ID  code 状态码  type 业务类型 |
| 返回值 | 无 |
| 备注 | type: 0 RESULT\_PROTO\_TYPE\_RTSP;  1 RESULT\_PROTO\_TYPE\_HLS;  99 OPENAPI;  **设备录像下载（type=0）的状态码code:**  STATE\_PACKET\_FRAME\_ERROR = 0; 组帧失败，错误状态  STATE\_RTSP\_TEARDOWN\_ERROR = 1; 内部要求关闭,如连接断开等，错误状态  STATE\_RTSP\_DESCRIBE\_READY = 2; 会话已经收到Describe响应，连接建立中  STATE\_RTSP\_AUTHORIZATION\_FAIL = 3; RTSP鉴权失败，错误状态  STATE\_RTSP\_PLAY\_READY = 4; 收到PLAY响应，连接成功  STATE\_RTSP\_FILE\_PLAY\_OVER = 5; 录像文件回放正常结束  STATE\_RTSP\_PAUSE\_READY = 6; 收到PAUSE响应  STATE\_RTSP\_KEY\_MISMATCH = 7; 解密秘钥错误  STATE\_RTSP\_SERVICE\_UNAVAILABLE = 99; 基于503错误码的连接最大数错误，错误状态  STATE\_RTSP\_USER\_INFO\_BASE\_START = 100; 用户信息起始码，服务端上层传过来的信息码会在该起始码基础上累加，错误状态  **云录像下载（type=1）的状态码code:**  HLS\_DOWNLOAD\_FAILD = 0; 下载失败  HLS\_DOWNLOAD\_BEGIN = 1; 开始下载  HLS\_DOWNLOAD\_END = 2; 下载结束  HLS\_SEEK\_SUCCESS = 3; 定位成功  HLS\_SEEK\_FAILD = 4; 定位失败  HLS\_ABORT\_DONE = 5; 停止下载  HLS\_RESUME\_DONE = 6; 恢复下载成功  HLS\_DOWNLOAD\_TIMEOUT = 7; 下载超时  HLS\_KEY\_ERROR = 11; 解密秘钥错  **OpenApi交互（type = 99）的状态码code:**  -1000; HTTP交互出错或超时  其他参考<https://open.lechange.com/developDoc/26>  出错后sdk内部会停止下载；状态码的回调处理可参见Demo |

## LCOpenSDK\_TalkerListener（回调接口）

### onTalkResult

|  |  |
| --- | --- |
| 方法 | -(void) onTalkResult:(NSString \*) error TYPE:(NSInteger) type; |
| 简介 | 语音对讲的状态码回调 |
| 参数 | error 状态码  type 业务类型 |
| 返回值 | 无 |
| 备注 | type: 0 RESULT\_PROTO\_TYPE\_RTSP;  99 OPENAPI;  **语音对讲（type=0）的状态码code:**  STATE\_PACKET\_FRAME\_ERROR = 0; 组帧失败，错误状态  STATE\_RTSP\_TEARDOWN\_ERROR = 1; 内部要求关闭,如连接断开等，错误状态  STATE\_RTSP\_DESCRIBE\_READY = 2; 会话已经收到Describe响应，连接建立中  STATE\_RTSP\_AUTHORIZATION\_FAIL = 3; RTSP鉴权失败，错误状态  STATE\_RTSP\_PLAY\_READY = 4; 收到PLAY响应，连接成功  STATE\_RTSP\_FILE\_PLAY\_OVER = 5; 录像文件回放正常结束  STATE\_RTSP\_PAUSE\_READY = 6; 收到PAUSE响应  STATE\_RTSP\_KEY\_MISMATCH = 7; 解密秘钥错误  STATE\_RTSP\_SERVICE\_UNAVAILABLE = 99; 基于503错误码的连接最大数错误，错误状态  STATE\_RTSP\_USER\_INFO\_BASE\_START = 100; 用户信息起始码，服务端上层传过来的信息码会在该起始码基础上累加，错误状态  **OpenApi交互（type = 99）的状态码code:**  -1000; HTTP交互出错或超时  其他参考<https://open.lechange.com/developDoc/26>  接收到错误回调后需关闭会话，具体判断和处理可参考Demo |

# 