

# IPC 播放库 SDK 用户使用说明

**Version 1.0**

# 目 录

<b>1 概述.....</b>	<b>3</b>
<b>2 功能接口介绍 .....</b>	<b>3</b>
<b>3 流程介绍 .....</b>	<b>6</b>
3.1 实时流播放 .....	6
3.2 本地文件播放 .....	7
<b>4 结构与宏定义 .....</b>	<b>8</b>
4.1 数据结构 .....	8
4.2 宏定义 .....	17
<b>5 接口定义 .....</b>	<b>26</b>
5.1 初始化 .....	26
5.2 实时流播放 .....	28
5.3 本地文件播放 .....	61

# 1 概述

播放库 SDK 分为二部分：实时流播放、本地文件播放。它是专为播放数据流设计的接口程序，以动态库的形式提供给应用软件开发人员，并同时附有演示程序及其源码，能有效地缩短应用软件开发周期。SDK 包含的组件说明如表 1 所示。

表 1. SDK 包含的组件

播放库功能接口		
播放库功能接口头文件	IPCPlayer.h	播放库 API 接口
播放库运行模块		
公共基础库	IPCPlayer.dll	
	IPCQuartz.dll	
Sample	IPCFilePlayDemo	文件播放演示程序

# 2 功能接口介绍

播放库功能接口按照下列顺序进行阐述：初始化、视频播放操作、音频播放操作、文件播放操作。

功能	接口	相关参数
<b>初始化</b>		
全局初始化	<a href="#">ipcPlayerInitialize</a>	
全局析构	<a href="#">ipcPlayerUnInitialize</a>	
注册 YUV 数据回调函数	<a href="#">ipcRegisterYUVCallBackEx</a>	
注册 AV 数据回调函数	<a href="#">ipcRegisterAVCallBackEx</a>	
注册画图回调函数	<a href="#">ipcRegisterDrawCallBack</a>	
创建实时流播放句柄	<a href="#">ipcCreatePlayerEx</a>	
创建实时流播放句柄扩展	<a href="#">ipcCreatePlayerEx2</a>	
释放实时流播放句柄	<a href="#">ipcDestroyPlayerEx</a>	
<b>实时流播放控制</b>		
开始播放	<a href="#">ipcStartEx</a>	
暂停播放	<a href="#">ipcStopEx</a>	
设置视频显示窗口句柄及区域坐标	<a href="#">ipcSetDisplayRectEx</a>	
获取视频显示窗口句柄及区域坐标	<a href="#">ipcGetDisplayRectEx</a>	
设置视频是否显示	<a href="#">ipcSetDisplayVisibleEx</a>	
获取视频是否显示状态	<a href="#">ipcGetDisplayVisibleEx</a>	
抓拍并保存到图片	<a href="#">ipcCapturePictureEx</a>	
抓拍并保存到内存	<a href="#">ipcCapturePicToBuffEx</a>	
抓拍并保存到内存扩展	<a href="#">ipcCapturePicToBuffEx2</a>	
设置 OSD	<a href="#">ipcSetOSDEx</a>	
获取实时流参数信息	<a href="#">ipcGetPlayInfoEx</a>	
输入帧数据	<a href="#">ipcInputDataEx</a>	
输入 IPC 私有帧数据	<a href="#">ipcInputDataEx2</a>	
设置音量	<a href="#">ipcSetAudioVolume</a>	
获取音量	<a href="#">ipcGetAudioVolume</a>	
设置区域显示	<a href="#">ipcSetDisplayRegion</a>	
刷新多区域显示的窗口	<a href="#">ipcRefreshDisplayRegion</a>	
设置流播放模式	<a href="#">ipcSetPlayerMode</a>	
获取当前播放器静音状态	<a href="#">ipcGetMuteEx</a>	
设置当前播放器静音状态	<a href="#">ipcSetMuteEx</a>	
获取当前解码播放的视频图像宽度	<a href="#">ipcGetImageWidthEx</a>	
获取当前解码播放的视频图像高度	<a href="#">ipcGetImageHeightEx</a>	
设置播放缓冲区缓冲时间	<a href="#">ipcSetBufferTimeEx</a>	
获取播放缓冲区缓冲时间	<a href="#">ipcGetBufferTimeEx</a>	
开启图像锐化	<a href="#">ipcStartImageSharpen</a>	
改变图像的锐化程度	<a href="#">ipcChangeSharpenDegree</a>	
停止图像锐化	<a href="#">ipcStopImageSharpen</a>	

<b>本地文件播放控制</b>		
初始化并创建文件播放句柄	<a href="#">ipcAvFileDec_Init</a>	
反初始化并销毁文件播放句柄	<a href="#">ipcAvFileDec_Done</a>	
注册解码后数据回调函数	<a href="#">ipcAvFileDec_RegisterAVFrameCallBack</a>	
注册播放状态回调函数	<a href="#">ipcAvFileDec_RegisterAVMsgCallBack</a>	
注册音视频数据解码后回调函数	<a href="#">ipcAvFileDec_RegisterAVFrameCallBackEx</a>	
打开文件	<a href="#">ipcAvFileDec_SetFileName</a>	
关闭文件	<a href="#">ipcAvFileDec_CloseFile</a>	
开始播放	<a href="#">ipcAvFileDec_Play</a>	
重复播放	<a href="#">ipcAvFileDec_RePlay</a>	
停止播放	<a href="#">ipcAvFileDec_Stop</a>	
暂停播放	<a href="#">ipcAvFileDec_Pause</a>	
快放	<a href="#">ipcAvFileDec_Fast</a>	
慢放	<a href="#">ipcAvFileDec_Slow</a>	
向前跳帧	<a href="#">ipcAvFileDec_Forward</a>	
向后跳帧	<a href="#">ipcAvFileDec_Backward</a>	
单帧播放	<a href="#">ipcAvFileDec_StepFrame</a>	
获取显示位置	<a href="#">ipcAvFileDec_GetDisplayPos</a>	
设置显示位置	<a href="#">ipcAvFileDec_SetDisplayPos</a>	
设置局部放大	<a href="#">ipcAvFileDec_SetZoom</a>	
获取局部放大	<a href="#">ipcAvFileDec_GetZoom</a>	
设置区域显示	<a href="#">ipcAvFileDec_SetDisplayRegion</a>	
刷新多区域显示的窗口	<a href="#">ipcAvFileDec_RefreshDisplayRegion</a>	
获取图像宽	<a href="#">ipcAvFileDec_GetImageWidth</a>	
获取图像高	<a href="#">ipcAvFileDec_GetImageHeight</a>	
获取总帧数	<a href="#">ipcAvFileDec_GetTotalFrames</a>	
获取当前文件播放的帧位置	<a href="#">ipcAvFileDec_GetCurrentPosition</a>	
设置当前文件播放的帧位置	<a href="#">ipcAvFileDec_SetCurrentPosition</a>	
获取总时间	<a href="#">ipcAvFileDec_GetTotalTime</a>	
获取当前播放时间点	<a href="#">ipcAvFileDec_GetCurrentTime</a>	
设置当前播放时间点	<a href="#">ipcAvFileDec_SetCurrentTime</a>	
获取当前播放状态	<a href="#">ipcAvFileDec_GetCurrentState</a>	
抓屏	<a href="#">ipcAvFileDec_CapturePicture</a>	
抓屏到内存	<a href="#">ipcAvFileDec_CapturePicToBuff</a>	
抓屏到内存扩展	<a href="#">ipcAvFileDec_CapturePicToBuffEX</a>	
文件剪切	<a href="#">ipcAvFileDec_Cut</a>	
播放器声音设置	<a href="#">ipcAvFileDec_SetVolume</a>	
播放器声音获取	<a href="#">ipcAvFileDec_GetVolume</a>	
获取音量静音标志	<a href="#">ipcAvFileDec_GetMute</a>	
设置音量静音标志	<a href="#">ipcAvFileDec_SetMute</a>	

# 3 流程介绍

## 3.1 实时流播放

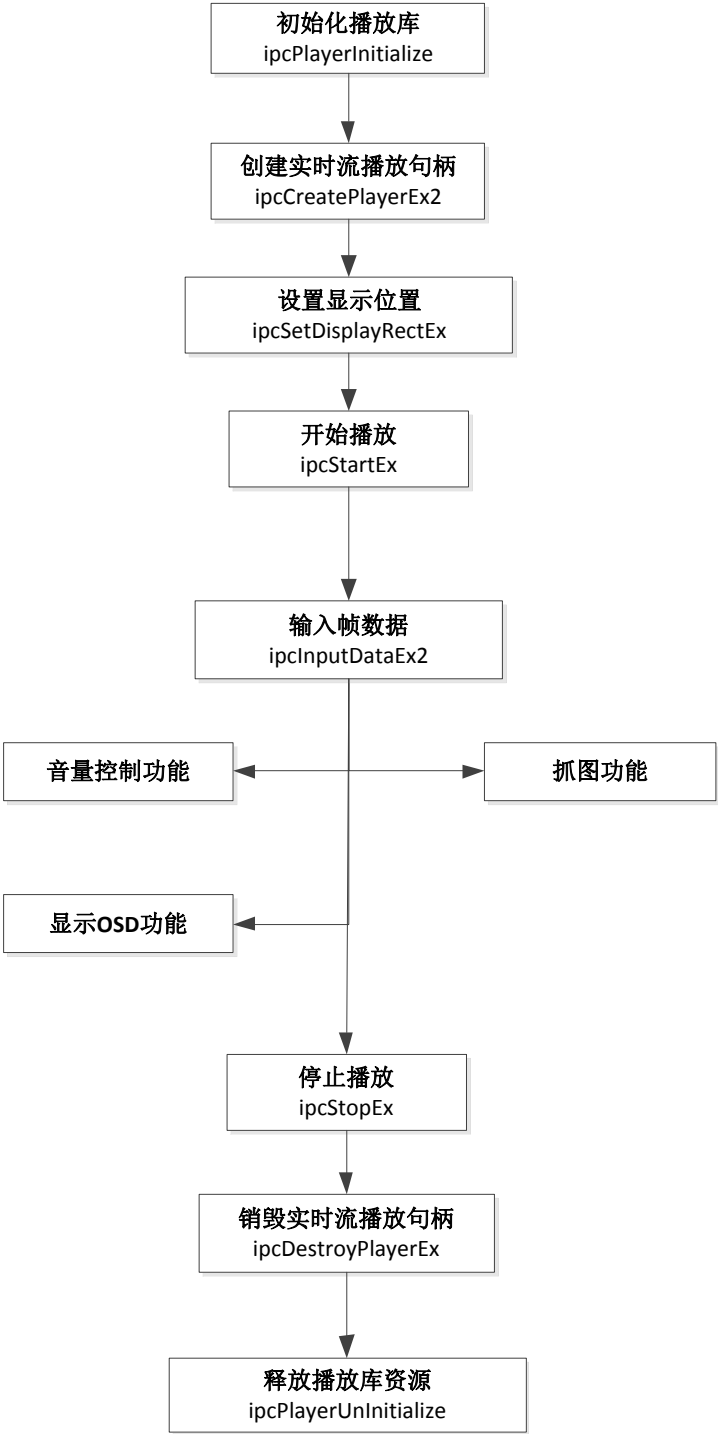


图 3-1

### 3.2 本地文件播放

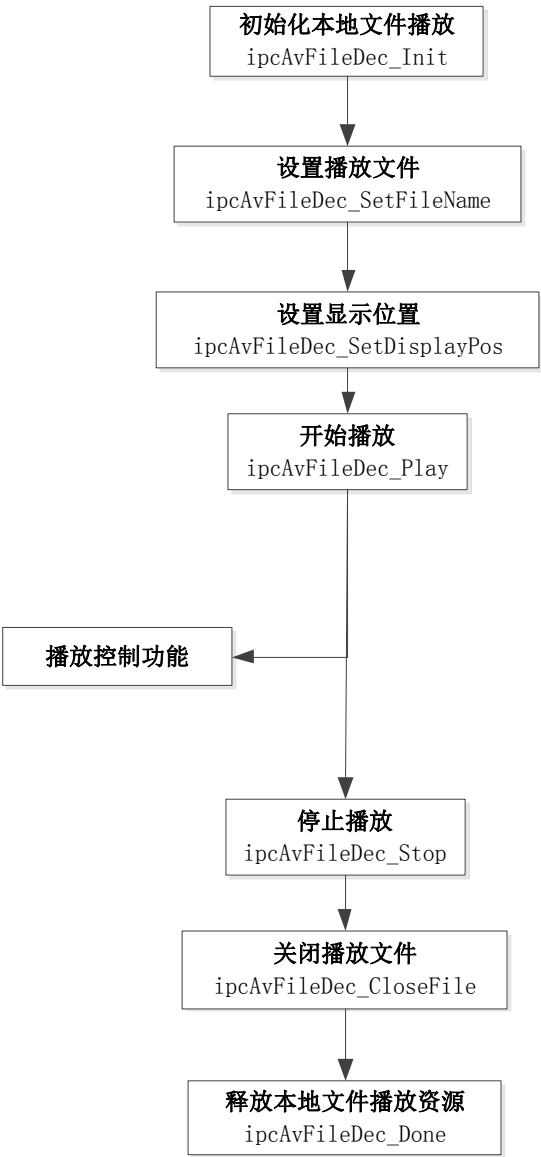


图 3-2

## 4 结构与宏定义

### 4.1 数据结构

#### IPC\_VIDEO\_INFO

视频参数信息

```
typedef struct
{
    long    IWidth;
    long    IHeight;
    long    IFouCC;
    long    IFrameRate;
} IPC_VIDEO_INFO,*PIPC_VIDEO_INFO;
```

#### Members

##### IWidth

视频图像的宽度

##### IHeight

视频图像的高度

##### IFouCC

视频编码格式，值 0 为 H264 格式，值 1 为 MJPEG 格式

##### IFrameRate

视频帧率

#### Remarks

在创建播放句柄时，需要正确输入 **IPC\_VIDEO\_INFO** 参数。否则可能会出现花屏或者不能显示的现象。

#### See Also

[IPC\\_PLAYER\\_INFO](#)



## IPC\_AUDIO\_INFO

音频参数信息

```
typedef struct
{
    long    IAudioFormat;
    long    ISamplesPerSec;
    long    IBitsPerSample;
    long    IChannels;
    long    IAvgBytesPerSec;
} IPC_AUDIO_INFO,*PIPC_AUDIO_INFO;
```

### Members

#### IAudioFormat

音频编码格式，值 0 为 AMR 格式，值 1 为 G711 格式，目前仅支持 G711

#### ISamplesPerSec

采样率，目前仅支持 8000 值

#### IBitsPerSample

采样精度，目前仅支持 8 值

#### IChannels

声道，目前仅支持 1 值

#### IAvgBytesPerSec

平均速率，目前仅支持 8000 值

### Remarks

目前仅支持播放 G711 格式音频数据

### See Also

[IPC\\_PLAYER\\_INFO](#)

## IPC\_PLAYER\_INFO

音视频参数信息

```
typedef struct
{
    IPC_VIDEO_INFO      videoInfo;
    IPC_AUDIO_INFO      audioInfo;
} IPC_PLAYER_INFO, *PIPC_PLAYER_INFO;
```

### Members

videoInfo

视频参数信息，可参考 [IPC\\_VIDEO\\_INFO](#) 数据结构

audioInfo

音频参数信息，可参考 [IPC\\_AUDIO\\_INFO](#) 数据结构

### Remarks

需正确填写 **IPC\_PLAYER\_INFO** 信息，否则会导致播放不正常。如果只播放视频，音频参数信息可都置为 0，如果只播放音频，视频参数信息可都置为 0

### See Also

[ipcCreatePlayerEx](#)

[ipcGetPlayInfoEx](#)

## IPC\_PLAYER\_FRAME

帧信息

```
typedef struct
{
    long                lType;
    unsigned char *    pBuf;
    long                lBufSize;
    ULONGLONG           ullStartTime;
    ULONGLONG           ullEndTime;
} IPC_PLAYER_FRAME, *PIPC_PLAYER_FRAME;
```

### Members

#### lType

帧类型，值'I'为 I 帧，值'P'为 P 帧，值'B'为 B 帧，值'A'为 A 帧

#### pBuf

帧数据缓冲区

#### lBufSize

帧数据缓冲区大小

#### ullStartTime

保留

#### ullEndTime

帧的时间戳，单位为纳秒

### Remarks

I 帧、P 帧、B 帧为视频帧，A 帧为音频帧。ullStartTime 请都置为 0

### See Also

[ipcVideoYUV\\_Callback](#)

[ipcInputDataEx](#)

## IPC\_PLAYER\_AV\_FRAME

实时流播放和文件播放时回调音视频解码后数据中的图像或声音信息

```
typedef struct
{
    DWORD    nType;
    DWORD    nWidth;
    DWORD    nHeight;
    DWORD    nFrameRate;
    DWORD    nStamp;
} IPC_PLAYER_AV_FRAME, *LPIPC_PLAYER_AV_FRAME;
```

### Members

#### nType

数据类型，0：YUY2，1：YUV420，2：PCM

#### nWidth

如果数据类型为视频则表示图像宽度，如果数据类型为音频则表示声音通道数

#### nHeight

如果数据类型为视频则表示图像高度，如果数据类型为音频则表示声音位率

#### nFrameRate

如果数据类型为视频则表示视频帧率，如果数据类型为音频则表示声音采样率

如果文件播放时，在数据类型为视频的情况下，该值一直为 0，则表示暂无法获取视频帧率

#### nStamp

保留

### Remarks

实时流播放和文件播放时音视频解码回调返回数据中图像或声音信息

### See Also

[ipcAVFrame\\_Callback](#)

[IPCPLAYER\\_AVFRAME\\_CALLBACKEX](#)

## PLAYOSDPARAMA

OSD 信息，unicode 版本

```
typedef struct tagPLAYOSDParamA
{
    bool        text ;
    long        text_x;
    long        text_y;
    long        text_lWidth;
    long        text_lHeight;
    LOGFONTA    text_font ;
    COLORREF    text_crColor ;
    wchar_t     text_context[256] ;

    bool        picture;                //是否显示图片
    long        pic_x;                  //图片显示区域
    long        pic_y;
    long        pic_lWidth;
    long        pic_lHeight;

    COLORREF    pic_crMask ;            //图片底色
    long        pic_data_len ;          //图片数据长度
    BYTE*       pic_data ;              //图片数据
} PLAYOSDPARAMA,*LPPLAYOSDPARAMA;
```

## Members

text

视频图像是否叠加文字信息，值 true 为叠加，值 false 为不叠加

text\_x

文字显示区域起始 x 轴坐标

text\_y

文字显示区域起始 y 轴坐标

text\_lWidth

文字显示区域宽度

text\_lHeight

文字显示区域高度

**text\_font**

文字字体结构，采用 window 字体结构 **LOGFONTA**, unicode 版本

**text\_crColor**

字体颜色

**text\_context[256]**

文字内容

**picture**

视频图像是否叠加图片信息，值 true 为叠加，值 false 为不叠加

**pic\_x**

图片显示区域起始 x 轴坐标

**pic\_y**

图片显示区域起始 y 轴坐标

**pic\_lWidth**

图片显示区域宽度

**pic\_lHeight**

图片显示区域高度

**pic\_crMask**

图片底色，可默认设置为 RGB(255, 255, 255)

**pic\_data\_len**

图片数据长度

**pic\_data**

图片数据

#### **Remarks**

如果不显示文字，除了 text 置为 false 外其他项请置为 0。图片亦然  
如遇编译环境使用 unicode 字符集，则使用本结构体

#### **See Also**

[ipcSetOSDEx](#)

## PLAYOSDPARAMW

OSD 信息，多字节字符集版本

```
typedef struct tagPLAYOSDParamW
{
    bool        text ;
    long        text_x;
    long        text_y;
    long        text_lWidth;
    long        text_lHeight;
    LOGFONTW    text_font ;
    COLORREF    text_crColor ;
    wchar_t     text_context[256] ;

    bool        picture;                //是否显示图片
    long        pic_x;                  //图片显示区域
    long        pic_y;
    long        pic_lWidth;
    long        pic_lHeight;

    COLORREF    pic_crMask ;            //图片底色
    long        pic_data_len ;          //图片数据长度
    BYTE*       pic_data ;              //图片数据
} PLAYOSDPARAMW,*LPPLAYOSDPARAMW;
```

## Members

text

视频图像是否叠加文字信息，值 true 为叠加，值 false 为不叠加

text\_x

文字显示区域起始 x 轴坐标

text\_y

文字显示区域起始 y 轴坐标

text\_lWidth

文字显示区域宽度

text\_lHeight

文字显示区域高度

**text\_font**

文字字体结构，采用 window 字体结构 **LOGFONTW**, 多字节字符集版本

**text\_crColor**

字体颜色

**text\_context[256]**

文字内容

**picture**

视频图像是否叠加图片信息，值 true 为叠加，值 false 为不叠加

**pic\_x**

图片显示区域起始 x 轴坐标

**pic\_y**

图片显示区域起始 y 轴坐标

**pic\_lWidth**

图片显示区域宽度

**pic\_lHeight**

图片显示区域高度

**pic\_crMask**

图片底色，可默认设置为 RGB(255, 255, 255)

**pic\_data\_len**

图片数据长度

**pic\_data**

图片数据

## Remarks

如果不显示文字，除了 **text** 置为 false 外其他项请置为 0。图片亦然如遇编辑环境为多字节字符集，则使用本结构

## See Also

[ipcSetOSDEx](#)



## 4.2 宏定义

### ipcVideoYUV\_CallBack

视频解码后 YUV 回调函数

```
typedef void (WINAPI *ipcVideoYUV_CallBack)
(
    PLAYERHANDLE          hPlayer,
    IPC_PLAYER_FRAME *    pFrame,
    CHAR*                  pData,
    INT                    nLen,
    LPVOID                 pContext
);
```

### Members

hPlayer

实时流播放句柄

pFrame

YUV 帧数据详细信息，详细信息可参考 [IPC\\_PLAYER\\_FRAME](#) 定义

pData

YUV 数据

nLen

YUV 数据长度

pContext

返回用户注册 YUV 回调函数时所输入的自定义数据

### Remarks

返回的 YUV 数据支持 YUY2 和 YUV420 格式

### See Also

[ipcRegisterYUVCallBackEx](#)

## ipcAVFrame\_CallBack

实时流播放时音频视频解码回调

```
typedef void (WINAPI * ipcAVFrame_CallBack)
(
    PLAYERHANDLE          hPlayer,
    IPC_PLAYER_AV_FRAME * pFrame,
    CHAR*                 pData,
    INT                   nLen,
    LPVOID                 pContext
);
```

### Members

hPlayer

实时流播放句柄

pFrame

图像或声音信息，详细信息可参考 [IPC\\_PLAYER\\_AV\\_FRAME](#) 定义

pData

解码后数据

nLen

数据长度

pContext

返回用户注册回调函数时所输入的自定义数据

### Remarks

通过该回调函数获得解码后图像或声音数据信息。

### See Also

[ipcRegisterAVCallBackEx](#)

## ipcDraw\_CallBack

实时流播放时画图回调函数

```
typedef void (WINAPI * ipcDraw_CallBack)
(
    PLAYERHANDLE          hPlayer,
    HDC                    hdc
    LPVOID                  pContext
);
```

### Members

hPlayer

实时流播放句柄

hdc

表面设备上下文，可以像操作显示窗口客户区 DC 那样操作它

pContext

返回用户注册回调函数时所输入的自定义数据

### Remarks

通过该回调函数可以在视频图像上进行绘画。每次返回 HDC 都是空白透明背景的，透明色为 RGB(255, 255, 255)

### See Also

[ipcRegisterDrawCallBack](#)

## IPCPLAYER\_MESSAGE\_CALLBACK

播放状态回调函数

```
typedef void (CALLBACK* IPCPLAYER_MESSAGE_CALLBACK)
(
    DWORD    dwCode,
    VOID *    context
);
```

### Members

dwCode

参数是一个 32 位数，低 16 位为改变的状态 见[播放状态](#)宏定义

context

返回用户注册回调函数时所输入的自定义数据

### Remarks

### See Also

[ipcAvFileDec\\_RegisterAVMsgCallBack](#)

## IPCPLAYER\_AVFRAME\_CALLBACK

音频视频解码回调

```
typedef void (CALLBACK* IPCPLAYER_AVFRAME_CALLBACK)
(
    LPBYTE    data,
    DWORD     size,
    DWORD     type,
    DWORD     width,
    DWORD     height,
    LPVOID     pContext
);
```

### Members

data

解码后 **YUY2** 数据

size

数据长度

type

解码后的数据格式

width

图像宽度

height

图像高度

pContext

返回用户注册回调函数时所输入的自定义数据

### Remarks

### See Also

[ipcAvFileDec\\_RegisterAVFrameCallBack](#)

## IPCPLAYER\_AVFRAME\_CALLBACKEX

音频视频解码回调的扩展接口

```
typedef void (CALLBACK* IPCPLAYER_AVFRAME_CALLBACKEX)
(
    LPBYTE          data,
    DWORD           size,
    LPIPC_PLAYER_AV_FRAME pInfo,
    LPVOID           pContext
);
```

### Members

data

解码后数据

size

数据长度

pInfo

码流信息， 详细可参考 [IPC\\_PLAYER\\_AV\\_FRAME](#)

pContext

返回用户注册回调函数时所输入的自定义数据

### Remarks

### See Also

[ipcAvFileDec\\_RegisterAVFrameCallBackEx](#)

## 播放状态

宏定义	宏定义值	含义
IPC_FILEPLAY_BASE	0x00008000	
IPC_PLAY_STATE_NONE	IPC_FILEPLAY_BASE +1	无文件打开
IPC_PLAY_STATE_STOP	IPC_FILEPLAY_BASE +2	停止播放
IPC_PLAY_STATE_PAUSE	IPC_FILEPLAY_BASE +3	暂停播放
IPC_PLAY_STATE_PLAY	IPC_FILEPLAY_BASE +4	正在播放
IPC_PLAY_STATE_FAST	IPC_FILEPLAY_BASE +5	快速播放
IPC_PLAY_STATE_SLOW	IPC_FILEPLAY_BASE +6	慢速播放
IPC_PLAY_STATE_BACK_NORMAL	IPC_FILEPLAY_BASE +7	倒序常速
IPC_PLAY_STATE_NORMAL	IPC_FILEPLAY_BASE +8	正序常速
IPC_PLAY_STATE_BACK_FAST	IPC_FILEPLAY_BASE +9	倒序快速
IPC_PLAY_STATE_BACK_SLOW	IPC_FILEPLAY_BASE +10	倒序慢速
IPC_PLAY_STATE_LOAD	IPC_FILEPLAY_BASE +11	正在打开文件
IPC_PLAY_STATE_OPEN	IPC_FILEPLAY_BASE +12	文件打开完毕，可以开始播放了
IPC_PLAY_STATE_CLOSE	IPC_FILEPLAY_BASE +13	关闭文件
IPC_PLAY_STATE_FILEEND	IPC_FILEPLAY_BASE +14	文件播放完毕
IPC_PLAY_STATE_ERROR	IPC_FILEPLAY_BASE +15	有错误发生

## See Also

## **MAX\_DISPLAY\_WND**

支持多区域显示的最多窗口数，目前支持最多 4 个窗口

### **See Also**

[ipcSetDisplayRegion](#)

[ipcRefreshDisplayRegion](#)



## 错误代码及说明

宏定义	宏定义值	含义
IPCPLAYER_NOERROR	0	没有错误
IPCPLAYER_GENERAL_ERROR	-1	一般性错误,待细化
IPCPLAYER_CREATE_DSHOW_ERROR	-2	创建 DSHOW 失败
IPCPLAYER_BUF_OVER	-3	缓冲区满，输入流失败

## See Also

## 5 接口定义

### 5.1 初始化

#### **ipcPlayerInitialize**

播放库全局初始化

```
INT ipcPlayerInitialize (  
    void  
);
```

#### **Parameters**

无

#### **Return Values**

正确返回 0，错误返回非 0

#### **Remarks**

函数的功能为初始化，注册所需要的组件。

## ipcPlayerUnInitialize

播放库全局析构

```
INT ipcPlayerUnInitialize (  
    void  
);
```

### Parameters

无

### Return Values

正确返回 0，错误返回非 0

### Remarks

## 5.2 实时流播放

### ipcCreatePlayerEx

创建实时流播放句柄

```
PLAYERHANDLE ipcCreatePlayerEx (  
    PIPC_PLAYER_INFO    pInfo  
);
```

#### Parameters

pInfo

[in] 音视频流参数信息，详细参数说明可参考 [IPC\\_PLAYER\\_INFO](#) 结构说明

#### Return Values

返回播放句柄，如果为 null 则创建失败

#### Remarks

函数的功能为创建播放实时流句柄，并初始化，开辟内存空间

若使用 IPC 智能高清网络摄像机 SDK 可在 **ipcSetAVCallbackEx** 混合流回调函数中获取音视频信息

#### See Also

[IPC\\_PLAYER\\_INFO](#)

## ipcCreatePlayerEx2

创建实时流播放句柄的扩展接口

```
PLAYERHANDLE ipcCreatePlayerEx2 (  
);
```

### Parameters

### Return Values

返回播放句柄，如果为 null 则创建失败

### Remarks

函数的功能为创建播放实时流句柄的扩展接口，为播放 IPC 智能高清网络摄像机 自定义帧数据  
若使用 IPC 智能高清网络摄像机 SDK 可在 **ipcSetAVCallbackEx2** 混合流回调函数中获取音视频信息

## ipcDestroyPlayerEx

释放实时流播放句柄

```
INT ipcDestroyPlayerEx (  
    PLAYERHANDLE    hPlayer  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为释放与此句柄相关的回调等信息，并释放内存空间

## **ipcStartEx**

开始播放

```
INT ipcStartEx (  
    PLAYERHANDLE hPlayer  
);
```

### **Parameters**

hPlayer

[in] 实时流播放句柄

### **Return Values**

正确返回 0，错误返回非 0

### **Remarks**

函数的功能为开始播放并等待实时流数据

## **ipcStopEx**

停止播放

```
INT ipcStopEx (  
    PLAYERHANDLE hPlayer  
);
```

### **Parameters**

hPlayer

[in] 实时流播放句柄

### **Return Values**

正确返回 0，错误返回非 0

### **Remarks**

函数的功能为停止播放



## ipcRegisterYUVCallBackEx

注册解码后的帧回调

```
INT ipcRegisterYUVCallBackEx (  
    PLAYERHANDLE      hPlayer,  
    ipcVideoYUV_CallBack pCallBack,  
    LPVOID            context  
    DWORD             dwType = 0  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

pCallBack

[in] 回调函数指针，函数信息定义可参考 [ipcVideoYUV\\_CallBack](#) 定义

context

[in] 用户自定义数据

dwType

[in] 视频解码后数据格式，0: YUY2，1: YUV420

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为通过该回调函数获得解码后 YUV 的实时视频数据。

### See Also

[ipcVideoYUV\\_CallBack](#)

## ipcRegisterAVCallbackEx

当实时流解码播放时，通过回调输出解码后的数据，其中视频为 YUY2/YUV420 格式，音频为 PCM 格式

```
INT ipcRegisterAVCallbackEx (  
    PLAYERHANDLE      hPlayer,  
    ipcAVFrame_Callback pCallBack,  
    LPVOID             context,  
    DWORD              dwType = 0  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

pCallBack

[in] 回调函数指针，函数信息定义可参考 [ipcAVFrame\\_Callback](#) 定义

context

[in] 用户自定义数据

dwType

[in] 视频解码后数据格式，0: YUY2，1: YUV420

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为通过注册回调函数获得解码图像或声音数据。

### See Also

[ipcAVFrame\\_Callback](#)

## ipcRegisterDrawCallBack

注册一个回调函数，当实时流播放时，通过该注册的回调函数进行绘画

```
INT ipcRegisterDrawCallBack (  
    PLAYERHANDLE      hPlayer,  
    ipcDraw_CallBack  pCallBack,  
    LPVOID             context  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

pCallBack

[in] 画图回调函数，函数信息定义可参考 [ipcDraw\\_CallBack](#) 定义

context

[in] 用户自定义数据

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为通过注册画图回调函数使用户可在视频图像上绘画。

### See Also

## ipcSetDisplayRectEx

设置视频显示窗口句柄及区域坐标

```
INT ipcSetDisplayRectEx (  
    PLAYERHANDLE    hPlayer,  
    HWND            hWnd,  
    IN RECT          rect  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

hWnd

[in] 显示窗口句柄

rect

[in] 显示区域坐标

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为设置显示窗口以及显示区域，如果 HWND 为 NULL，则播放器只回调解码数据不负责显示部分。对于控制显示部分的 API 也将失效

## ipcGetDisplayRectEx

获取视频显示窗口句柄及区域坐标

```
INT ipcGetDisplayRectEx (  
    PLAYERHANDLE    hPlayer,  
    HWND*           hWnd,  
    OUT RECT*        rect  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

hWnd

[out] 显示窗口句柄

rect

[out] 显示区域坐标

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为获取显示窗口句柄以及显示区域坐标

## ipcSetDisplayVisibleEx

设置视频是否显示

```
INT ipcSetDisplayVisibleEx (  
    PLAYERHANDLE    hPlayer,  
    IN BOOL          bVal  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

bVal

[in] TRUE 为显示，FALSE 为不显示

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为设置是否显示视频

## ipcGetDisplayVisibleEx

获取视频是否显示状态

```
INT ipcGetDisplayVisibleEx (  
    PLAYERHANDLE    hPlayer,  
    OUT BOOL*       bVal  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

bVal

[out] 显示状态，TRUE 为显示，FALSE 为不显示

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为获取视频显示状态

## ipcCapturePictureEx

捕获当前显示帧数据到文件

```
INT ipcCapturePictureEx (  
    PLAYERHANDLE    hPlayer,  
    int             picType,  
    WCHAR*          file_name  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

picType

[in] 保存图片的类型，值 1 为 BMP 格式图片，值 2 为 JPEG 格式图片

file\_name

[in] 保存图片文件的全路径

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为截取帧数据保存为文件



## ipcCapturePicToBuffEx

捕获当前显示帧数据到内存

```
INT ipcCapturePicToBuffEx (  
    PLAYERHANDLE    hPlayer,  
    int             picType,  
    BYTE*           buff,  
    LONG*           len  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

picType

[in] 保留，暂无作用

buff

[out] 返回图像数据，该参数传递 NULL，只返回需要的缓存长度

len

[in,out] buff 为空时返回图像数据长度，buff 不为空时需要传递 buff 的长度，函数返回的时候输出实际拷贝的内存块大小。

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为截取一帧视频数据输出保存到一块指定内存区域

## ipcCapturePicToBuffEx2

捕获当前显示帧数据到内存

```
INT ipcCapturePicToBuffEx2 (  
    PLAYERHANDLE    hPlayer,  
    int             picType,  
    BYTE*           buff,  
    LONG*           len  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

picType

[in] 保留，暂无作用

buff

[out] 返回图像数据，该参数传递 NULL，只返回需要的缓存长度

len

[in,out] buff 为空时返回图像数据长度，buff 不为空时需要传递 buff 的长度，函数返回的时候输出实际拷贝的内存块大小。

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为截取一帧视频数据输出保存到一块指定内存区域

## ipcSetOSDEx

设置 OSD

```
INT ipcSetOSDEx (  
    PLAYERHANDLE      hPlayer,  
    LPPLAYOSDPARAM    osd_info  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

osd\_info

[in] OSD 数据信息，详细可参数 [PLAYOSDPARAMW](#) 或 [PLAYOSDPARAMA](#) 结构说明

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为设置 OSD 信息，本结构有 unicode 字符集版本 **PLAYOSDPARAMA** 和多字节字符集版本 **PLAYOSDPARAMW**

## ipcGetPlayInfoEx

获取实时流参数信息

```
INT ipcGetPlayInfoEx (  
    PLAYERHANDLE      hPlayer,  
    PIPC_PLAYER_INFO  pInfo  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

pInfo

[out] 实时流参数信息，详细参数信息说明可参考 [IPC\\_PLAYER\\_INFO](#) 结构说明

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为获取实时流参数信息

## ipcInputDataEx

输入帧数据

```
INT ipcInputDataEx (  
    PLAYERHANDLE          hPlayer,  
    PIPC_PLAYER_FRAME     pFrame  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

pFrame

[in] 帧数据，详细参数信息说明可参考 [IPC\\_PLAYER\\_FRAME](#) 数据结构

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为传入所需播放的帧数据包括视频帧，音频帧

## ipcInputDataEx2

为播放私有帧数据的扩展接口

```
INT ipcInputDataEx2 (
    PLAYERHANDLE      hPlayer,
    BYTE*             pFrameBuff,
    LONG              len
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

pFrameBuff

[in] 帧数据，详细可参考 IPC 智能高清网络摄像机 SDK 用户手册

len

[in] 帧数据长度

### Return Values

正确返回 0，错误返回非 0

如返回 [IPCPLAYER\\_BUF\\_OVER](#) 表示数据满无法送入播放库

### Remarks

函数的功能为播放 IPC 智能高清网络摄像机 自定义帧数据

若使用 IPC 智能高清网络摄像机 SDK 可在 **ipcSetAVCallbackEx2** 混合流回调函数中获取帧数据

在文件流播放模式下我们为了更加突出播放的流畅性，同时减小延时性，在数据满情况下可能会进行丢帧处理。

而在实时流播放模式下，数据满而导致无法送入播放库，我们则返回 [IPCPLAYER\\_BUF\\_OVER](#) 错误码。用户可以自行选择丢弃，后果是画面可能会跳帧或花屏，如果无法接受，则 sleep 几毫秒后，再尝试多送几次。

## ipcSetAudioVolume

设置音量

```
INT ipcSetAudioVolume(  
    PLAYERHANDLE hPlayer,  
    long val  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

val

[in] 音量，值范围为：[0,100]

0 表示静音，100 表示全音量

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为设置播放音频帧时候的音量

## ipcGetAudioVolume

获取音量

```
INT ipcGetAudioVolume(  
    PLAYERHANDLE    hPlayer,  
    Long*           val  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

val

[out] 音量，值范围为：[0,100]

0 表示静音，100 表示全音量

### Return Values

正确返回 0，错误返回非 0

### Remarks

函数的功能为获取播放音频帧时候的音量



## ipcSetDisplayRegion

设置或增加显示区域

```
INT ipcSetDisplayRegion (  
    PLAYERHANDLE    hPlayer,  
    DWORD           nRegionNum,  
    RECT             rcSrc,  
    HWND            hDestWnd,  
    BOOL             bEnable  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

nRegionNum

[in] 显示区域序号,0~([MAX\\_DISPLAY\\_WND](#)-1), 如果为 0, 则将设置的区域显示在主窗口中  
将忽略 hDestWnd

rcSrc

[in] 局部显示区域

hDestWnd

[in] 显示窗口句柄

bEnable

[in] 打开(设置)或关闭显示区域

### Return Values

正确返回 0, 错误返回非 0

### Remarks

设置或增加显示区域,可以做局部放大显示

## ipcRefreshDisplayRegion

刷新多区域显示的窗口

```
INT ipcRefreshDisplayRegion (  
    PLAYERHANDLE    hPlayer,  
    DWORD            nRegionNum,  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

nRegionNum

[in] 显示区域序号,0~([MAX\\_DISPLAY\\_WND](#)-1), 如果为 0, 则将设置的区域显示在主窗口中

### Return Values

正确返回 0, 错误返回非 0

### Remarks

刷新多区域显示的窗口

## ipcSetPlayerMode

设置流播放模式

```
INT ipcSetPlayerMode (  
    PLAYERHANDLE    hPlayer,  
    DWORD            nMode  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

nMode

[in] 流播放模式，0 为实时流播放，1 为文件流播放

### Return Values

正确返回 0，错误返回非 0

### Remarks

文件流播放模式也就是按时间戳播放，而实时流播放时忽略帧数据的时间戳

文件流播放模式更加突显出播放的流畅性，但有一定的延时性。可有 [ipcSetBufferSize](#) 和 [ipcSetBufferTimeEx](#) 调整平衡不同网络环境下流畅性和延时性。

实时流播放模式则更加突显出播放的实时性，在网络抖动比较大的情况下流畅性方面比较差。

### See Also

## ipcGetMuteEx

获取当前播放器静音状态

```
INT ipcGetMuteEx (  
    PLAYERHANDLE    hPlayer,  
    BOOL*           bMute  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

bMute

[out] 静音标志位, TRUE: 打开静音, FALSE: 关闭静音

### Return Values

正确返回 0, 错误返回非 0

### Remarks

### See Also

[ipcSetMuteEx](#)

## ipcSetMuteEx

设置当前播放器静音状态

```
BOOL ipcGetMuteEx (  
    PLAYERHANDLE hPlayer,  
    BOOL bMute  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

bMute

[in] 静音标志位，TRUE：打开静音，FALSE：关闭静音

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

[ipcGetMuteEx](#)

## ipcGetImageWidthEx

获取当前解码播放的视频图像宽度

```
INT ipcGetImageWidthEx (  
    PLAYERHANDLE    hPlayer,  
    LONG*           width  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

width

[out] 实时流图像的宽度

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

## ipcGetImageHeightEx

获取当前解码播放的视频图像高度

```
INT ipcGetImageWidthEx (  
    PLAYERHANDLE    hPlayer,  
    LONG*           height  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

height

[out] 实时流图像的高度

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

## ipcSetBufferTimeEx

设置播放缓冲区缓冲时间

```
INT ipcSetBufferTimeEx (  
    PLAYERHANDLE    hPlayer,  
    DWORD           time  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

time

[in] 播放缓冲区缓冲的时间，单位为毫秒，取值范围 0 到 1000，默认值为 135

### Return Values

正确返回 0，错误返回非 0

### Remarks

设置播放缓冲区缓冲时间的大小，这个缓冲时间比较重要，它可直接影响播放的流畅性和延时性。在一定范围内缓冲时间越大越流畅，同时延时越大。在播放实时流时我们的默认缓冲时间 135 毫秒，在帧率为 30 时，缓冲 4 帧。如果用户追求最大延时最小，可以考虑适当减小这个值。

注意在播放模式设置为实时流时，设置的缓冲时间此时无效。

### See Also



## ipcGetBufferTimeEx

获取播放缓冲区缓冲时间

```
INT ipcGetBufferTimeEx (  
    PLAYERHANDLE    hPlayer,  
    DWORD*          time  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

time

[out] 播放缓冲区缓冲的时间，单位为毫秒

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

## ipcStartImageSharpen

开启图像锐化

```
INT ipcStartImageSharpen (  
    PLAYERHANDLE  hPlayer,  
    float          amount    = 1.2f  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

amount

[in] 图像锐化程度，默认为 1.2f，范围 0.0f-5.0f

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

## ipcChangeSharpenDegree

改变图像的锐化程度

```
INT ipcChangeSharpenDegree (  
    PLAYERHANDLE    hPlayer,  
    float            amount  
);
```

### Parameters

hPlayer

[in] 实时流播放句柄

amount

[in] 图像锐化程度，范围 0.0f-5.0f

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

## ipcStopImageSharpen

停止图像锐化

```
INT ipcStopImageSharpen (  
    PLAYERHANDLE    hPlayer  
);
```

## Parameters

hPlayer

[in] 实时流播放句柄

## Return Values

正确返回 0，错误返回非 0

## Remarks

## See Also

## 5.3 本地文件播放

### ipcAvFileDec\_Init

初始化播放器，每次调用初始化一个播放器

```
HANDLE ipcAvFileDec_Init (  
    HWND    hWnd,  
    DWORD   dwFlags,  
    DWORD   dwDisFmt,  
    DWORD   dwAudFmt  
);
```

#### Parameters

hWnd

[in] 播放器显示窗口句柄

dwFlags

[in] 解码器类型<只能为 H264\_DECODER\_ID\_BKASF>  
忽略，默认填 0

dwDisFmt

[in] 显示 Surface 类型<该参数暂被忽略>  
忽略，默认填 0

dwAudFmt

[in] 播放声音方式<该参数暂被忽略>  
忽略，默认填 0

#### Return Values

初始化正确返回该播放通道控制句柄，错误返回 NULL

#### Remarks

## ipcAvFileDec\_Done

释放播放器资源

```
INT ipcAvFileDec_Done (  
    HANDLE    hAmdle  
);
```

### Parameters

hAmdle

[in] 通道控制句柄

### Return Values

正确返回 0，错误返回非 0

### Remarks

## ipcAvFileDec\_RegisterAVMsgCallBack

当播放器中播放状态改变时会调用注册的回调函数

```
INT ipcAvFileDec_RegisterAVMsgCallBack (  
    HANDLE                                hAndle,  
    IPCPLAYER_MESSAGE_CALLBACK           pCallBack,  
    VOID*                                context  
);
```

### Parameters

hAndle

[in] 通道控制句柄

pCallBack

[in] 回调函数指针

context

[in] 应用程序提供的参数

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

[IPCPLAYER\\_MESSAGE\\_CALLBACK](#)

## ipcAvFileDec\_RegisterAVFrameCallBack

当解码播放时，通过回调输出解码后的数据，其中视频为 YUY2 格式

```
INT ipcAvFileDec_RegisterAVFrameCallBack (  
    HANDLE                                hAmdle,  
    IPCPLAYER_AVFRAME_CALLBACK           pCallBack,  
    VOID*                                context  
);
```

### Parameters

hAmdle

[in] 通道控制句柄

pCallBack

[in] 回调函数指针

context

[in] 应用程序提供的参数

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

[IPCPLAYER\\_AVFRAME\\_CALLBACK](#)



## ipcAvFileDec\_RegisterAVFrameCallBackEx

当解码播放时，通过回调输出解码后的数据，其中视频为 YUV420 格式，音频为 PCM 格式

```
INT ipcAvFileDec_RegisterAVFrameCallBackEx (  
    HANDLE                                hAndle,  
    IPCPLAYER_AVFRAME_CALLBACKEX         pCallBack,  
    VOID*                                context  
);
```

### Parameters

hAndle

[in] 通道控制句柄

pCallBack

[in] 回调函数指针

context

[in] 应用程序提供的参数

### Return Values

正确返回 0，错误返回非 0

### Remarks

### See Also

[IPCPLAYER\\_AVFRAME\\_CALLBACKEX](#)

## ipcAvFileDec\_SetFileName

设置播放文件

```
INT ipcAvFileDec_SetFileName (  
    HANDLE          hAandle,  
    LPCTSTR         lpFileName,  
    LPCTSTR         lpAudioFile,  
    BOOL            bPlay  
);
```

### Parameters

hAandle

[in] 通道控制句柄

lpFileName

[in] 文件名

lpAudioFile

[in] 附加音频文件名,该参数保留，传递 NULL

bPlay

[in] 自动播放标志

### Return Values

正确返回 0，错误返回非 0

### Remarks

设置播放文件，lpAudioFile 参数忽略；如果 bPlay 为 TRUE 时文件初始化成功后将自动播放，FALSE 将等待用户的控制

## **ipcAvFileDec\_CloseFile**

关闭播放器中当前打开的文件

```
INT ipcAvFileDec_CloseFile (  
    HANDLE    hAandle  
);
```

### **Parameters**

hAandle

[in] 通道控制句柄

### **Return Values**

正确返回 0，错误返回非 0

### **Remarks**

## ipcAvFileDec\_Play

设置播放器为播放状态

```
INT ipcAvFileDec_Play (  
    HANDLE    hAndle  
);
```

### Parameters

hAndle

[in] 通道控制句柄

### Return Values

正确返回 0，错误返回非 0

### Remarks

## **ipcAvFileDec\_RePlay**

设置播放器为重新播放状态

```
INT ipcAvFileDec_RePlay (  
    HANDLE    hAmdle  
);
```

### **Parameters**

hAmdle

[in] 通道控制句柄

### **Return Values**

正确返回 0，错误返回非 0

### **Remarks**

当为快放或慢放时恢复为正常播放状态，播放将从头播放

## **ipcAvFileDec\_Stop**

设置播放器为停止状态

```
INT ipcAvFileDec_Stop(  
    HANDLE    hAndle  
);
```

### **Parameters**

hAndle

[in] 通道控制句柄

### **Return Values**

正确返回 0，错误返回非 0

### **Remarks**

设置为停止播放状态

## ipcAvFileDec\_Pause

设置播放器为暂停状态

```
INT ipcAvFileDec_Pause(  
    HANDLE    hAmdle  
);
```

### Parameters

hAmdle

[in] 通道控制句柄

### Return Values

正确返回 0，错误返回非 0

### Remarks

暂停后调用 **ipcAvFileDec\_RePlay** 恢复为播放状态

## ipcAvFileDec\_Fast

设置播放器为快放状态

```
INT ipcAvFileDec_Fast (  
    HANDLE    hAndle  
    float     fRate  
);
```

### Parameters

hAndle

[in] 通道控制句柄

fRate

[in] 播放速率，为负数时为倒放

### Return Values

正确返回 0，错误返回非 0

### Remarks

设置播放器为快放状态，fRate 绝对值必须大于 1.0 ，为负数时为倒放



## ipcAvFileDec\_Slow

设置播放器为慢放状态

```
INT ipcAvFileDec_Slow (  
    HANDLE    hAndle  
    float     fRate  
);
```

### Parameters

hAndle

[in] 通道控制句柄

fRate

[in] 播放速率，负数时为倒放

### Return Values

正确返回 0，错误返回非 0

### Remarks

设置播放器为慢放状态，fRate 绝对值必须小于 1.0 ，负数时为倒放

## ipcAvFileDec\_StepFrame

设置播放器为单帧播放状态

```
INT ipcAvFileDec_StepFrame (  
    HANDLE    hAndle  
    bool      bForward  
);
```

### Parameters

hAndle

[in] 通道控制句柄

bForward

[in] 相对于文件前后帧标志

### Return Values

正确返回 0，错误返回非 0

### Remarks

暂停状态下该函数调用有效

调用一次播放一帧，bForward 为 TRUE 时文件前进一帧，FALSE 时文件后退一帧

## ipcAvFileDec\_Forward

文件前跳一段

```
INT ipcAvFileDec_Forward (  
    HANDLE    hAandle  
    DWORD     dwOffset  
);
```

### Parameters

hAandle

[in] 通道控制句柄

dwOffset

[in] 偏移量

### Return Values

正确返回 0，错误返回非 0

### Remarks

暂停状态下该函数调用有效

文件前跳一段，dwOffset 为相对于当前的帧位置(单位为帧)

## ipcAvFileDec\_Backward

文件后跳一段

```
INT ipcAvFileDec_Backward (  
    HANDLE    hAandle  
    DWORD     dwOffset  
);
```

### Parameters

hAandle

[in] 通道控制句柄

dwOffset

[in] 偏移量

### Return Values

正确返回 0，错误返回非 0

### Remarks

暂停状态下该函数调用有效

文件后跳一段，dwOffset 为相对于当前的帧位置(单位为帧)

## ipcAvFileDec\_GetDisplayPos

获取播放显示矩形框

```
INT ipcAvFileDec_GetDisplayPos (  
    HANDLE    hAndle  
    RECT*     lpRect  
);
```

### Parameters

hAndle

[in] 通道控制句柄

lpRect

[out] 显示矩形框

### Return Values

正确返回 0，错误返回非 0

### Remarks

调用 [ipcAvFileDec\\_SetDisplayPos](#) 函数设置的显示矩形框

## ipcAvFileDec\_SetDisplayPos

设置播放显示矩形框

```
INT ipcAvFileDec_SetDisplayPos (  
    HANDLE    hAndle,  
    RECT *    lpRect  
);
```

### Parameters

hAndle

[in] 通道控制句柄

lpRect

[in] 显示矩形框

### Return Values

正确返回 0，错误返回非 0

### Remarks

设置播放显示位置，lpRect 是相对于初始化时给的窗口的矩形位置

## ipcAvFileDec\_SetZoom

设置局部放大

```
INT ipcAvFileDec_SetZoom (  
    HANDLE    hAandle,  
    BOOL      bZoom,  
    RECT *    lpRect  
);
```

### Parameters

hAandle

[in] 通道控制句柄

bZoom

[in] 是否局部放大

lpRect

[in] 局部放大矩形框，坐标为被截取用于放大部分的视频图像分辨率坐标值。

### Return Values

正确返回 0，错误返回非 0

### Remarks

## ipcAvFileDec\_GetZoom

获取局部放大

```
BOOL ipcAvFileDec_GetZoom (  
    HANDLE    hAndle,  
    RECT *    lpRect  
);
```

### Parameters

hAndle

[in] 通道控制句柄

lpRect

[out] 存放局部放大矩形框的指针，坐标为被截取用于放大部分的视频图像分辨率坐标值

### Return Values

返回当前是否处于局部放大状态

### Remarks



## ipcAvFileDec\_SetDisplayRegion

设置或增加显示区域

```
INT ipcAvFileDec_SetDisplayRegion (  
    HANDLE          hAndle,  
    DWORD           nRegionNum,  
    RECT            rcSrc,  
    HWND            hDestWnd,  
    BOOL            bEnable  
);
```

### Parameters

hAndle

[in] 通道控制句柄

nRegionNum

[in] 显示区域序号,0~([MAX\\_DISPLAY\\_WND](#)-1), 如果为 0, 则将设置的区域显示在主窗口中  
将忽略 hDestWnd

rcSrc

[in] 局部显示区域

hDestWnd

[in] 显示窗口句柄

bEnable

[in] 打开(设置)或关闭显示区域

### Return Values

正确返回 0, 错误返回非 0

### Remarks

设置或增加显示区域,可以做局部放大显示

## ipcAvFileDec\_RefreshDisplayRegion

刷新多区域显示的窗口

```
INT ipcAvFileDec_RefreshDisplayRegion (  
    HANDLE          hAndle,  
    DWORD           nRegionNum,  
);
```

### Parameters

hAndle

[in] 通道控制句柄

nRegionNum

[in] 显示区域序号,0~([MAX\\_DISPLAY\\_WND](#)-1), 如果为 0, 则将设置的区域显示在主窗口中

### Return Values

正确返回 0, 错误返回非 0

### Remarks

刷新多区域显示的窗口

## **ipcAvFileDec\_GetImageWidth**

获取当前播放文件视频图像宽度

```
INT ipcAvFileDec_GetImageWidth (  
    HANDLE    hAandle  
);
```

### **Parameters**

hAandle

[in] 通道控制句柄

### **Return Values**

正确返回当前播放文件视频图像宽度

### **Remarks**

## **ipcAvFileDec\_GetImageHeight**

获取当前播放文件视频图像高度

```
INT ipcAvFileDec_GetImageHeight (  
    HANDLE    hAandle  
);
```

### **Parameters**

hAandle

[in] 通道控制句柄

### **Return Values**

正确返回当前播放文件视频图像高度

### **Remarks**

## **ipcAvFileDec\_GetTotalFrames**

获取当前播放文件的总帧数

```
LONG ipcAvFileDec_GetTotalFrames (  
    HANDLE    hAmdle  
);
```

### **Parameters**

hAmdle

[in] 通道控制句柄

### **Return Values**

正确返回打开文件的总帧数

### **Remarks**

当返回大于 0 时才表示正确得到总帧数，小于 0 则失败。

## **ipcAvFileDec\_GetCurrentPosition**

得到当前文件播放的帧位置

```
LONG ipcAvFileDec_GetCurrentPosition (  
    HANDLE    hAmdle  
);
```

### **Parameters**

hAmdle

[in] 通道控制句柄

### **Return Values**

当前视频播放的帧位置

### **Remarks**

当返回大于 0 时才表示正确得到帧位置，小于 0 则失败。

## ipcAvFileDec\_SetCurrentPosition

设置当前视频播放的帧位置

```
INT ipcAvFileDec_SetCurrentPosition (  
    HANDLE    hAndle,  
    LONG      IPosition  
);
```

### Parameters

hAndle

[in] 通道控制句柄

IPosition

[in] 视频帧位置

### Return Values

正确返回 0，错误返回非 0

### Remarks

暂停状态下该函数调用有效

设置文件播放起点位置，IPosition 单位为帧

## **ipcAvFileDec\_GetTotalTime**

得到当前播放文件的总时间

```
Float ipcAvFileDec_GetTotalTime (  
    HANDLE    hAmdle,  
);
```

### **Parameters**

hAmdle

[in] 通道控制句柄

### **Return Values**

返回打开的文件总时间(单位秒)

### **Remarks**



## **ipcAvFileDec\_GetCurrentTime**

得到当前文件播放时间

```
Float ipcAvFileDec_GetCurrentTime (  
    HANDLE    hAmdle,  
);
```

### **Parameters**

hAmdle

[in] 通道控制句柄

### **Return Values**

返回当前文件播放时间(单位秒)

### **Remarks**

## ipcAvFileDec\_SetCurrentTime

设置播放的起点时间

```
INT ipcAvFileDec_SetCurrentTime (  
    HANDLE    hAndle,  
    FLOAT     dwTime  
);
```

### Parameters

hAndle

[in] 通道控制句柄

dwTime

[in] 播放起点时间

### Return Values

正确返回 0，错误返回非 0

### Remarks

设置播放的起点时间，dwTime 文件起点播放时间，单位为秒

## ipcAvFileDec\_GetCurrentState

得到播放器当前状态

```
INT ipcAvFileDec_GetCurrentState (  
    HANDLE    hAmdle,  
);
```

### Parameters

hAmdle

[in] 通道控制句柄

### Return Values

返回播放状态，见宏定义中[播放状态](#)

### Remarks

See Also

[播放状态](#)

## ipcAvFileDec\_CapturePicture

捕获当前显示帧

```
INT ipcAvFileDec_CapturePicture (  
    HANDLE    hAndle,  
    LPCTSTR   lpFileName,  
    LPCTSTR   lpPicFormat  
);
```

### Parameters

hAndle

[in] 通道控制句柄

lpFileName

[in] 文件名

lpPicFormat

[in] 图像格式，当前为传递固定值 bmp

### Return Values

正确返回 0，错误返回非 0

### Remarks

捕获当前显示帧；图片大小为解码原始大小

## ipcAvFileDec\_CapturePictToBuff

捕获当前显示帧数据到内存

```
INT ipcAvFileDec_CapturePicture (  
    HANDLE    hAmdle,  
    BYTE*     buff,  
    LONG*     len  
);
```

### Parameters

hAmdle

[in] 通道控制句柄

buff

[out] 返回图像数据，该参数传递 NULL，只返回需要的缓存长度；  
数据为包含 BITMAPINFOHEADER 的位图数据，无 BITMAPFILEHEADER

len

[out] buff 为空时返回图像数据长度，buff 不为空时需要传递 buff 的长度，函数返回的时候输出实际拷贝的内存块大小。

### Return Values

正确返回 0，错误返回非 0

### Remarks

捕获当前显示帧数据到内存，格式为 bmp，图片大小为解码原始大小

## ipcAvFileDec\_CapturePictToBuffEx

捕获当前显示帧数据到内存

```
INT ipcAvFileDec_CapturePictureEx (  
    HANDLE    hAndle,  
    BYTE*     buff,  
    LONG*     len  
);
```

### Parameters

hAndle

[in] 通道控制句柄

buff

[out] 返回图像数据，该参数传递 NULL，只返回需要的缓存长度；数据为包含 BITMAPFILEHEADER，BITMAPINFOHEADER 的位图数据

len

[in,out] buff 为空时返回图像数据长度，buff 不为空时需要传递 buff 的长度，函数返回的时候输出实际拷贝的内存块大小。

### Return Values

正确返回 0，错误返回非 0

### Remarks

捕获当前显示帧数据到内存，格式为 bmp，图片大小为解码原始大小

## ipcAvFileDec\_Cut

文件剪切

```
INT ipcAvFileDec_Cut (  
    HANDLE    hAandle,  
    LPCTSTR   pDestFileName,  
    INT        iStartTime,  
    INT        iEndTime  
);
```

### Parameters

hAandle

[in] 通道控制句柄

pDestFileName

[in] 目标文件路径

iStartTime

[in] 起始时间点，单位 s，不得大于文件总时间长度

iEndTime

[in] 结束时间点，单位 s,不得大于文件总时间长度

### Return Values

正确返回 0，错误返回非 0

### Remarks

## ipcAvFileDec\_SetVolume

设置音量百分比

```
INT ipcAvFileDec_SetVolume(  
    HANDLE hAmdle,  
    LONG lVolume  
);
```

### Parameters

hAmdle

[in] 通道控制句柄

lVolume

[in] 0~100 ， 0 为静音,100 为最大

### Return Values

正确返回 0， 错误返回非 0

### Remarks



## ipcAvFileDec\_GetVolume

获取音量百分比

```
INT ipcAvFileDec_GetVolume(  
    HANDLE    hAandle,  
    LONG*     lVolume  
);
```

### Parameters

hAandle

[in] 通道控制句柄

lVolume

[out] 音量百分比，值范围为 0~100 ， 0 为静音,100 为最大

### Return Values

正确返回 0，错误返回非 0

### Remarks

## ipcAvFileDec\_GetMute

获取音量静音标志

```
INT ipcAvFileDec_GetMute (  
    HANDLE    hAandle,  
    BOOL*     bMute  
);
```

### Parameters

hAandle

[in] 通道控制句柄

bMute

[out] 静音标志

### Return Values

正确返回 0，错误返回非 0

### Remarks

## ipcAvFileDec\_SetMute

设置音量静音标志

```
INT ipcAvFileDec_SetMute (  
    HANDLE    hAndle,  
    BOOL      bMute  
);
```

### Parameters

hAndle

[in] 通道控制句柄

bMute

[in] 静音标志

### Return Values

正确返回 0，错误返回非 0

### Remarks