

一、动态库播放控件.....	4
1、开发目的.....	4
二、API 定义及作用.....	4
IP_TPS_OpenStream 打开流.....	4
IP_TPS_CloseStream 关闭流.....	5
IP_TPS_DeleteStream 删除并释放一路播放相关资源.....	5
IP_TPS_Play 开始播放(或解码).....	5
IP_TPS_Stop 停止播放(或解码).....	6
IP_TPS_PlaySound 播放声音.....	6
IP_TPS_InputVideoData 传入视频数据.....	7
IP_TPS_InputAudioData 传入音频数据.....	7
IP_TPS_StopSound 停止声音.....	7
IP_TPS_CatchPic 抓图保存到.....	8
IP_TPS_CatchPicByFileName 抓图保存到指定文件.....	8
IP_TPS_CloseAll 关闭所有.....	9
IP_TPS_ReleaseAll 释放所有.....	9
视频解码回调函数定义(OnMediaDataRecv).....	10
事件通知回调函数定义(OnPlayerEventRecv).....	11
IP_TPS_SetDecCallBack 设置解码数据回调.....	12
IP_TPS_GetVersion 取当前发布版本.....	12
IP_TPS_SetStatusEventCallBack 设置事件回调.....	13
IP_TPS_SetZoomRectOn 设置电子回放模式.....	13
IP_TPS_GetZoomRectStatus 取得电子缩放模式.....	13
IP_TPS_InputMouseEvent 传入鼠标事件控制电子缩放.....	14
IP_TPS_SetFullFillStatus 设置全屏播放状态.....	14
IP_TPS_SetVideoOn 设置是否播放视频(仅用于播放模式).....	15
IP_TPS_PlayLocFile 回放本地文件.....	15
IP_TPS_StopPlayLocFile 停止本地播放.....	15
IP_TPS_GetPlayTime 取得当前正在播放的时间.....	16
IP_TPS_GetFileTime 取得当前回放文件的总时间长度.....	16
IP_TPS_ControlPlay 回放控制.....	16
IP_TPS_GetBufferCount 取得当前缓冲里有多少帧.....	17
IP_TPS_ClearBuffer 清除缓冲区里的数据.....	17
IP_TPS_BufferIsFull 判断当前缓冲区是否已满.....	18
IP_TPS_SetShowTitle 视频叠加标题信息设置.....	18
IP_TPS_CreateAuidoStreamPlayer 创建音频播放器.....	19
IP_TPS_InputAuidoStreamPlayer 传入要播放的压缩音频.....	19
IP_TPS_DestroyAuidoStreamPlayer 释放音频播放器.....	19
IP_TPS_SetColor 调节图像参数.....	20
IP_TPS_GetColor 获取调节图像参数.....	20

IP_TPS_OpenFile 打开本地文件.....	20
IP_TPS_StartPlayFile 开始播放文件.....	21
IP_TPS_RefreshSurface 刷新指定通道图像.....	21
IP_TPS_SetVolume 设置指定通道音频音量.....	22
IP_TPS_GetVolume 获取指定通道音频音量.....	22
IP_TPS_SetPlayDirection 设置回放方向.....	22
IP_TPS_SetDisplayFinishCallBack 设置显示图片结束回调函数.....	23
IP_TPS_PlaySoundShare 共享声卡方式播放声音.....	23
IP_TPS_StopSoundShare 停止播放音频.....	23
IP_TPS_SetBufferTick 设置播放缓冲时间.....	24
三、播放开发流程.....	24
使用 DllPlayer.dll 作为播放库.....	24
四、解码开发流程.....	25
使用 DllPlayer.dll 作为解码库大概流程.....	25
五、开发注意事项.....	27

一、动态库播放控件

1、开发目的

由于很多客户希望提供动态库DLL调用形式播放库，所以临时开发一个使用提供API函数调用。

二、API 定义及作用

IP_TPS_OpenStream 打开流

API定义原型

```
DLLPLAYER_API int __stdcall IP_TPS_OpenStream(LONG nPort, PBYTE pParam, DWORD pSize, int isAudioParam, DWORD nBufPoolSize);
```

此函数将打开一个流，其中：

nPort:

不同的值标记视频流。

pParam:

相关参数，当为视频流时（isAudioParam=0），则表示视频解压参数，否则为音频参数

pSize:

参数长度

isAudioParam:

标记当前是否为音频流，0表示视频，非0表示音频

nBufPoolSize:

表示最大有多少个缓存，如果是实时流且网络相对稳定时此值建议为40，网络差时可以考虑适当加大。

0: 成功返回

ERR_PLY_AUDIOPARAM_ERROR: 音频参数错误

ERR_PLY_VIDECPARAM_ERROR: 视频参数错误

IP_TPS_CloseStream 关闭流

API定义原型

```
DLLPLAYER_API int __stdcall IP_TPS_CloseStream(LONG nPort);
```

功能:

关闭一路解码

参数:

nPort: 不同的值标记视频流。

0: 成功

ERR_PLY_VIDECCHANNELID_ERROR: 视频通道端口nPort错误

注意:

调用此函数后, 将会停止解码和停止播放。但是nPort对应的资源未释放, 方便再次播放同一路视频。

IP_TPS_DeleteStream 删除并释放一路播放相关资源

定义原型:

```
DLLPLAYER_API int __stdcall IP_TPS_DeleteStream(LONG nPort);
```

功能:

停止播放并释放一路播放相关资源

参数:

nPort: 标记视频流编号

0: 成功

ERR_PLY_VIDECCHANNELID_ERROR: 视频通道端口nPort错误

注意:

此API与IP_TPS_CloseStream不同之处在于, 此函数会释放占用资源, 而IP_TPS_CloseStream并不会释放占用资源。建议使用IP_TPS_CloseStream, 退出程序时再调用IP_TPS_ReleaseAll

IP_TPS_Play 开始播放(或解码)

API定义原型

```
DLLPLAYER_API int __stdcall IP_TPS_Play(LONG nPort, HWND hWnd);
```

此函数表示在某窗口上开始播放某一路视频 其中:

nPort:

不同的值标记视频流。

hWnd:

表示要在哪个窗口上播放视频, 如果只是使用外部收流, 则此值必须为空

注意:

如果不同的视频流(即nPort不同)在相同的窗口句柄上播放时, 一定要先将之前的视频流停掉, 否则会出现交替显示错乱的情况。

如果在外部播放, 则要先调用IP_TPS_SetDecCallBack进行设置回调函数, 然后hWnd必须为NULL(即0)

0:成功

ERR_PLY_NOT_DECODER_MODE:hWnd为无效句柄且不是当作解码模式

IP_TPS_Stop 停止播放(或解码)

API定义原型

```
DLLPLAYER_API int __stdcall IP_TPS_Stop(LONG nPort);
```

功能:

此函数等同于IP_TPS_DeleteStream, 仅为保持接口

参数, 返回值, 见IP_TPS_DeleteStream函数.

IP_TPS_PlaySound 播放声音

API定义原型

```
DLLPLAYER_API int __stdcall IP_TPS_PlaySound(LONG nPort);
```

功能:

表示开始播放音频

参数:

nPort: 不同的值标记视频流。

0:成功

ERR_PLY_VIDECCHANNELID_ERROR: 视频通道端口nPort错误

注意:

调用此函数时, 会停掉其它正在播放的声音, 即同一时间只会播放一路声音。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/825221102033011103>