全志科技
Allwinner Technology

# 音视频解码库项目

## 视频输出 API 说明

# 文档履历

| 版本号 | 日期 | 制/修订人 | 内容描述 |
| --- | --- | --- | --- |
| V1.0 | 2016-05-13 | 王喜望 | 初稿 |
| V1.1 | 2016-11-03 | | 修改 api 说明和数据结构 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

# 目 录

# 1. 概述

## 1.1. 编写目的

设计一套用于渲染视频帧数据的显示接口，简化中间件对显示模块的调用。指导显示模块的使用和后续维护

## 1.2. 适用范围

A80/A83/H3/H5/H8 等各个芯片平台的 Android 系统 SDK 和 Linux SDK。

## 1.3. 相关人员

开发和维护显示模块的相关人员。

# 2. 模块介绍

## 2.1.功能介绍

视频解码器解码出来的视频帧数据，最终会作为显示设备的输入数据进行显示，而在android 系统中，显示设备的操作句柄为 nativeWindow，本模块的作用是对 nativeWindow 提供的接口进行二次封装，将细节处理和定制化的处理封装在接口函数里，简化中间件对 nativeWindow 的调用。

## 2.2.相关术语介绍

layerControl：显示接口模块，对 surface 的接口调用进行封装。

nativeWindow：显示设备 surface 的具体操作句柄。

# 3. 接口说明

## 3.1. 视频输出 API

| 视频输出 API | |
|---|---|
| LayerCreate | 创建 layercontrol |
| LayerRelease | 释放 layer buffer 信息 |
| LayerSetSecureFlag | 设置是否是安全视频 |
| LayerSetDisplayBufferSize | 设置显示 buffer 的大小，即 width 、height |
| LayerSetDisplayBufferCount | 设置显示 buffer 的个数 |
| LayerSetDisplayRegion | 设置 buffer 的显示区域 |
| LayerSetDisplayPixelFormat | 设置 buffer 的显示格式 |
| LayerSetVideoWithTwoStreamFlag | 设置是否是 3D 双流的视频 |
| LayerSetIsSoftDecoderFlag | 设置 decoder 是否是软件解码 |
| LayerSetBufferTimeStamp | 设置每一帧 buffer 的 pts |
| LayerResetNativeWindow | 重新设置 nativeWindow 句柄 |
| LayerGetBufferOwnedByGpu | 获取被 gpu 占用的 buffer |
| LayerGetDisplayFPS | 获取显示设备的显示帧率 |
| LayerGetBufferNumHoldByGpu | 获取被 nativeWindow 占用的 buffer 个数 |
| LayerCtrlShowVideo | 显示视频画面 |
| LayerCtrlHideVideo | 隐藏视频画面不进行显示 |
| LayerCtrlIsVideoShow | 查询显示状态 |
| LayerCtrlHoldLastPicture | 保持最后一帧 |
| LayerDequeueBuffer | 从 nativeWindow 获取一帧 buffer |
| LayerQueueBuffer | 把一帧 buffer 传递到 nativeWindow |
| LayerReleaseBuffer | 释放一帧 buffer |
| LayerReset | 重新设置 layer，清空 buffer 信息 |
| LayerDestroy | 销毁 layercontrol |
| LayerControl | Layer 控制操作，用于扩展 |

下面对各个接口分别进行介绍。

### 3.1.1. LayerCreate

| 函数原型 | LayerCtrl* LayerCreate(void* pNativeWindow) |
|---|---|
| 功能 | 创建 layercontrol |
| 参数 | pNativeWindow：显示设备的操作句柄 |
| 返回值 | 成功：LayerCtrl 指针<br>失败：NULL |
| 调用说明 | |

### 3.1.2. LayerRelease

| 函数原型 | void LayerRelease(LayerCtrl* l) |
|---|---|
| 功能 | 释放 layer buffer 信息 |
| 参数 | l: layercontrol 指针 |
| 返回值 | 无 |
| 调用说明 | 无 |

### 3.1.3. LayerSetSecure

| 函数原型 | int LayerSetSecureFlag(LayerCtrl* l, int f) |
|---|---|
| 功能 | 设置是否是安全视频 |
| 参数 | l: layercontrol 指针, f: 安全视频标志 |
| 返回值 | 0：成功 |
| 调用说明 | 无 |

### 3.1.4. LayerSetDisplayBufferSize

| 函数原型 | int LayerSetDisplayBufferSize(LayerCtrl* l, int w, int h) |
|---|---|
| 功能 | 设置显示 buffer 的大小，即 width 、height |
| 参数 | L: layercontrol 指针<br>w：buffer 的宽度<br>h：buffer 的高度 |
| 返回值 | 0：设置成功 |
| 调用说明 | |

### 3.1.5. LayerSetDisplayBufferCount

| 函数原型 | int LayerSetDisplayBufferCount(LayerCtrl* l, int n) |
|---|---|
| 功能 | 设置显示 buffer 的个数 |
| 参数 | L: layercontrol 指针<br>n：buffer 个数 |
| 返回值 | 返回设置的显示 buffer 个数 |
| 调用说明 | |

### 3.1.6. LayerSetDisplayRegion

| 函数原型 | int LayerSetDisplayRegion(LayerCtrl* l, int left, int top, int w, int h) |
|---|---|
| 功能 | 设置 buffer 的显示区域 |

| | |
|---|---|
| 参数 | L：layercontrol 指针<br>left：显示区域的左边距<br>top：显示区域的上边距<br>w：显示区域的宽度<br>h：显示区域的高度 |
| 返回值 | 0：设置成功<br>-1：设置失败 |
| 调用说明 | |

### 3.1.7. LayerSetDisplayPixelFormat

| 函数原型 | int LayerSetDisplayPixelFormat(LayerCtrl* l, enum EPIXELFORMAT p) |
|---|---|
| 功能 | 设置 buffer 的显示格式 |
| 参数 | L：layercontrol 指针<br>p：显示格式 |
| 返回值 | 0：设置成功<br>-1：设置失败 |
| 调用说明 | |

### 3.1.8. LayerSetVideoWithTwoStreamFlag

| 函数原型 | int LayerSetVideoWithTwoStreamFlag(LayerCtrl* l, int f) |
|---|---|
| 功能 | 设置标志位，表明视频是否是 3D 双流 |
| 参数 | l：layercontrol 指针<br>f：标志位的设置值 |
| 返回值 | 0：设置成功 |
| 调用说明 | |

### 3.1.9. LayerSetIsSoftDecoderFlag

| 函数原型 | int LayerSetIsSoftDecoderFlag(LayerCtrl* l, int flag) |
|---|---|
| 功能 | 设置标志位，表明 decoder 是否是软件解码器 |
| 参数 | l：layercontrol 指针<br>flag：标志位的设置值 |
| 返回值 | 0：设置成功 |
| 调用说明 | |

### 3.1.10. LayerSetBufferTimeStamp

| 函数原型 | int LayerSetBufferTimeStamp(LayerCtrl* l, int64_t pts) |
|---|---|
| 功能 | 设置 buffer 数据携带的 pts |

| | |
|---|---|
| 参数 | l: layercontrol 指针<br>pts：显示时间戳 pts |
| 返回值 | 0：设置成功 |
| 调用说明 | |

### 3.1.11. LayerResetNativeWindow

| 函数原型 | void LayerResetNativeWindow(LayerCtrl* l,void* p) |
|---|---|
| 功能 | 重新设置 nativeWindow 的操作句柄 |
| 参数 | l: layercontrol 指针<br>p：nativeWindow 操作句柄 |
| 返回值 | 无 |
| 调用说明 | |

### 3.1.12. LayerGetBufferOwnedByGpu

| 函数原型 | VideoPicture* LayerGetBufferOwnedByGpu(LayerCtrl* l) |
|---|---|
| 功能 | 获取当前被 nativeWindow 占用的 buffer |
| 参数 | l: layercontrol 指针 |
| 返回值 | 成功：VideoPicture（描述 buffer 的结构体）指针 |
| 调用说明 | |

### 3.1.13. LayerGetDisplayFPS

| 函数原型 | int LayerGetDisplayFPS(LayerCtrl* l) |
|---|---|
| 功能 | 获取显示设备的显示帧率 |
| 参数 | l: layercontrol 指针 |
| 返回值 | 显示帧率的值 |
| 调用说明 | |

### 3.1.14. LayerGetBufferNumHoldByGpu

| 函数原型 | int LayerGetBufferNumHoldByGpu(LayerCtrl* l) |
|---|---|
| 功能 | 获取 nativeWindow 占用的 buffer 个数 |
| 参数 | l: layercontrol 指针 |
| 返回值 | 占用个数 |
| 调用说明 | |

### 3.1.15. LayerCtrlShowVideo

| 函数原型 | int LayerCtrlShowVideo(LayerCtrl* l) |
|---|---|
| 功能 | 控制 nativeWindow 的显示状态，使其正常显示视频画面 |
| 参数 | l: layercontrol 指针 |
| 返回值 | 0：调用成功<br>-1：调用失败 |
| 调用说明 | |

### 3.1.16. LayerCtrlHideVideo

| 函数原型 | int LayerCtrlHideVideo(LayerCtrl* l) |
|---|---|
| 功能 | 控制 nativeWindow 的显示状态，使其不显示视频画面 |
| 参数 | l: layercontrol 指针 |
| 返回值 | 0：调用成功<br>-1：调用失败 |
| 调用说明 | |

### 3.1.17. LayerCtrlIsVideoShow

| 函数原型 | int LayerCtrlIsVideoShow(LayerCtrl* l) |
|---|---|
| 功能 | 查询 nativeWindow 的显示状态 |
| 参数 | l: layercontrol 指针 |
| 返回值 | 1：正常显示<br>0：不显示 |
| 调用说明 | |

### 3.1.18. LayerCtrlHoldLastPicture

| 函数原型 | int LayerCtrlHoldLastPicture(LayerCtrl* l, int h) |
|---|---|
| 功能 | 退出播放时，使 nativeWindow 保持显示最后一帧的视频画面 |
| 参数 | l: layercontrol 指针<br>h：值为 1 则保持最后一帧，值为 0 时不保持 |
| 返回值 | 0：调用成功 |
| 调用说明 | |

### 3.1.19. LayerDequeueBuffer

| 函数原型 | int LayerDequeueBuffer(LayerCtrl* l, VideoPicture** p, int f) |
|---|---|
| 功能 | 从 nativeWindow 获取一帧 buffer |

| | |
|---|---|
| 参数 | l：layercontrol 指针<br>p：用于存放 buffer 地址<br>f：是否需要初始化当前 buffer |
| 返回值 | 0：获取成功<br>-1：获取失败 |
| 调用说明 | |

### 3.1.20. LayerQueueBuffer

| 函数原型 | int LayerQueueBuffer(LayerCtrl* l, VideoPicture* p, int f) |
|---|---|
| 功能 | 把一帧 buffer 还回到 nativeWindow |
| 参数 | l：layercontrol 指针<br>p：描述 buffer 的结构体指针<br>f：值为 1 则表明当前 buffer 用于显示，值为 0 则不显示 |
| 返回值 | 0：调用成功<br>-1：调用失败 |
| 调用说明 | |

### 3.1.21. LayerReleaseBuffer

| 函数原型 | int LayerReleaseBuffer(LayerCtrl* l,VideoPicture* p) |
|---|---|
| 功能 | 通过 ion 接口释放当前 buffer，重新设置过 nativeWindow 时才会调用到此接口 |
| 参数 | l：layercontrol 指针<br>p：描述 buffer 的结构体指针 |
| 返回值 | 0：调用成功 |
| 调用说明 | |

### 3.1.22. LayerReset

| 函数原型 | int LayerReset(LayerCtrl* l) |
|---|---|
| 功能 | 重新设置 layer，清空 buffer 信息 |
| 参数 | l：layercontrol 指针 |
| 返回值 | 0：调用成功 |
| 调用说明 | |

### 3.1.23. LayerDestroy

| 函数原型 | int LayerDestroy(LayerCtrl* l) |
|---|---|
| 功能 | 销毁 layercontrol |
| 参数 | l：layercontrol 指针 |

| | |
|---|---|
| 返回值 | 无 |
| 调用说明 | |

### 3.1.24. LayerControl

| 函数原型 | int LayerControl(LayerCtrl* l, int cmd, void *para) |
|---|---|
| 功能 | Layer 控制操作，用于扩展 |
| 参数 | l: layercontrol 指针<br>cmd：命令<br>para：参数 |
| 返回值 | 返回 0 |
| 调用说明 | |

# 4. 数据结构设计

## 4.1. 数据结构

| 数据结构 | |
|---|---|
| struct LayerCtrl | 表示视频输出控制句柄 |
| enum CdxLayerCommandE | 视频输出控制命令 |
| struct LayerCtrlContext | 表示视频输出控制环境 |

### 4.1.1. LayerCtrl

| 名称 | struct LayerCtrl |
|---|---|
| 功能描述 | 表示视频输出控制句柄，成员包括各个接口，具体见 3.1 视频输出 API 说明章节。 |

### 4.1.2. CdxLayerCommandE

| 名称 | enum CdxLayerCommandE |
|---|---|
| 功能描述 | 视频输出控制命令，用于扩展其他操作。 |
| 取值 | 描述 |
| CDX_LAYER_CMD_RESTART_SCHEDULER | 重启视频调度，更新参数。 |

### 4.1.3. LayerCtrlContext

| 名称 | LayerCtrlContext | |
|---|---|---|
| 功能描述 | 表示视频输出控制环境。 | |
| 属性 | 类型 | 描述 |
| base | LayerCtrl | 视频输出控制句柄 |
| pNativeWindow | ANativeWindow* | 显示设备 surface 的操作句柄 |
| eDisplayPixelFormat | enum EPIXELFORMAT | Buffer 的显示格式 |
| nWidth | int | Buffer 的宽度值 |
| nHeight | int | Buffer 的高度值 |
| nLeftOff | int | Buffer 的显示区域的左边距 |
| nTopOff | int | Buffer 的显示区域的上边距 |
| nDisplayWidth | int | Buffer 的显示区域的宽度值 |
| nDisplayHeight | int | Buffer 的显示区域的高度值 |

| | | |
|---|---|---|
| bLayerInitialized | int | 是否已经对 nativeWindow 进行初始化的标志位 |
| bLayerShowed | int | 值为 1 表明显示视频帧，值为 0 表明不显示视频帧 |
| bProtectFlag | int | 安全视频标志位，即支持 widevine level 1 的安全视频播放场景 |
| picNodes | VPictureNode | 描述 nativeWindow 占用的 buffer 队列 |
| mGpuBufferInfo | GpuBufferInfoT | 用于 nativeWindow 与 decoder 关于描述 buffer 的结构体之间的转换，nativeWindow 定义 ANativeWindowBuffer 用于描述 buffer，decoder 定义 VideoPicture 用于描述 buffer |
| nGpuBufferCount | int | Buffer 个数 |
| ionFd | int | 操作 ion 设备的描述符 |
| b4KAlignFlag | int | Buffer 地址是否需要 4K 对齐的标志位 |
| bHoldLastPictureFlag | int | 退出播放时是否保持显示最后一帧画面 |
| bVideoWithTwoStreamFlag | int | 值为 1 表明视频属于 3D 双流场景，在这种场景下 buffer 需要多申请一倍 |
| bIsSoftDecoderFlag | int | 值为 1 表明 decoder 属于软件解码器 |
| nUsage | unsigned int | 用于描述 buffer 的属性，如 buffer 是否带 cache |
| mVideoScheduler | CdxVideoScheduler | 视频帧调度句柄 |

# 5. Declaration