



# AX VDEC API 文档

文档版本: V1.2

发布日期: 2021/10/21

AXERA CONFIDENTIAL FOR Sipeed

前 言 .....	4
修订历史 .....	5
1 概述 .....	6
1.1 概述 .....	7
1.2 功能描述 .....	7
1.2.1 重要概念 .....	7
1.2.2 软件流程 .....	9
2 API 简介 .....	11
3 API 定义 .....	12
AX_VDEC_Init .....	12
AX_VDEC_Deinit .....	13
AX_VDEC_CreateGrp .....	14
AX_VDEC_DestroyGrp .....	16
AX_VDEC_StartRecvStream .....	17
AX_VDEC_StopRecvStream .....	18
AX_VDEC_SendStream .....	19
AX_VDEC_SetDisplayMode .....	21
AX_VDEC_GetDisplayMode .....	22
AX_VDEC_SetGrpAttr .....	23
AX_VDEC_GetGrpAttr .....	24
AX_VDEC_SetProtocolParam .....	25
AX_VDEC_GetProtocolParam .....	26

AX_VDEC_GetFrame.....	27
AX_VDEC_ReleaseFrame.....	29
AX_VDEC_AttachPool.....	30
AX_VDEC_DetachPool.....	32
<b>4 数据结构.....</b>	<b>33</b>
AX_VDEC_GRP_ATTR_S.....	33
AX_VDEC_STREAM_S.....	35
AX_VDEC_DISPLAY_MODE_E.....	37
AX_VDEC_PARAM_VIDEO_S.....	38
AX_VDEC_DEC_MODE_E.....	39
AX_VDEC_OUTPUT_ORDER_E.....	40
AX_VDEC_MODE_E.....	41
AX_VIDEO_FRAME_INFO_S.....	42
AX_MOD_ID_E.....	43
AX_VIDEO_FRAME_S.....	45
AX_VDEC_PRTCL_PARAM_S.....	48
AX_VDEC_H264_PRTCL_PARAM_S.....	49
AX_VDEC_H265_PRTCL_PARAM_S.....	50
AX_PAYLOAD_TYPE_E.....	51
<b>5 错误码.....</b>	<b>55</b>
<b>6 调试信息.....</b>	<b>57</b>

## 权利声明

爱芯元智半导体（上海）有限公司或其许可人保留一切权利。

非经权利人书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

## 注意

您购买的产品、服务或特性等应受商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非商业合同另有约定，本公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

AXERA CONFIDENTIAL FOR Speed

## 适用产品



爱芯 AX620A

## 前言

## 适读人群

- 软件开发工程师
- 技术支持工程师

## 符号与格式定义

符号/格式	说明
xxx	表示您可以执行的命令行。
斜体	表示变量。如，“安装目录/AX620A_SDK_Vx.x.x/build 目录”中的“安装目录”是一个变量，由您的实际环境决定。
 说明/备注：	表示您在使用产品的过程中，我们向您说明的事项。
 注意：	表示您在使用产品的过程中，需要您特别注意的事项。

## 修订历史

文档版本	发布时间	修订说明
V1.0	2021/08/30	文档初版
V1.1	2021/10/15	<ol style="list-style-type: none"><li>1. 更新了流程描述</li><li>2. destroychn api 增加了 timeout 参数</li><li>3. 部分结构体的 padding 参数改名为 frameheight</li></ol>
V1.2	2021/10/21	<ol style="list-style-type: none"><li>1. 修改 API 命名及参数命名</li><li>2. 修改参数结构体命名，加入 AX_前缀</li></ol>
V1.3	2022/04/09	<ol style="list-style-type: none"><li>1. 更新结构体 AX_VDEC_GRP_ATTR_S 信息</li><li>2. 增加 AX_VDEC_AttachPool 和 AX_VDEC_DetachPool API 描述</li><li>3. 更新 H264 最小分辨率支持</li><li>4. 增加 VDEC 模块缓存 VB 使用描述</li><li>5. 增加调试信息说明</li></ol>
V1.4	2022/05/12	<ol style="list-style-type: none"><li>1. 修改表述符号错误</li><li>2. 更新送流方式支持情况</li></ol>

---

本章节包含：

[1.1 概述](#)

[1.2 功能描述](#)

## 1 概述

AXERA CONFIDENTIAL FOR Sipeed

## 1.1 概述

视频解码（video decoder）模块简称 VDEC，负责对 H.264/JPEG 格式的数字压缩视频进行解压缩处理，从而得到 YUV 格式的图像。该模块支持多路解码。本文档主要介绍 VDEC 模块的工作流程，以及相关 API 接口的使用方法。

不同型号的芯片支持不同的解码规格。芯片支持的解码规格如下表所示。

表1-1 芯片解码规格

芯片型号	H.264				JPEG		
AX620A	Baseline Profile	Main Profile	High Profile	最大分辨率	JPEG Baseline	Motion JPEG	最大分辨率
	支持 Levels 1-5.1	支持 Levels 1-5.1	支持 Levels 1-5.1	1920x1088	支持	支持	16384x16384

## 1.2 功能描述

### 1.2.1 重要概念

#### 码流提交方式

VDEC 模块目前支持一种码流发送方式：

- 按帧发送方式（VIDEO\_MODE\_FRAME）：用户每次调用 API 向解码器发送一帧压缩码流。该方式所需的帧缓存少，解码延迟小，较适合实时解码的场景。如无特别需求，一般建议使用该模式解码。



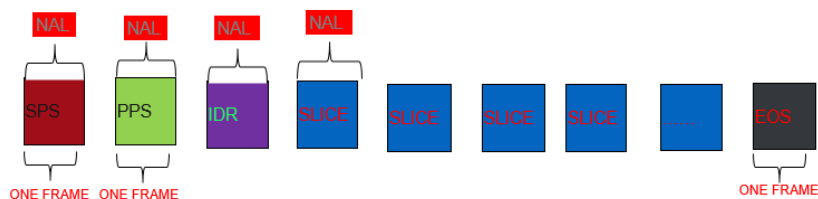


图1-1 按帧发送示意图

用户在创建解码通道的同时设置码流提交方式，详情可参考 `AX_VDEC_CreateChn()` 的接口说明。

## 图像输出顺序

当码流中存在 B 帧时，码流的解码顺序与图像的显示顺序可能是不一样的，所以 VDEC 模块支持两种图像输出方式：

- 按解码的先后顺序输出
- 按图像的正常显示顺序输出

## 播放时间戳（Presentation Timestamp, PTS）

在模式 `VIDEO_MODE_FRAME` 下发送码流时，解码输出的图像时间戳 PTS 为发送码流接口（`AX_VDEC_SendStream`）中用户送入的 PTS，解码器不会更改此值。

### ！ 注意：

开始解码一个序列时，首先送入解码器的一般都是 SPS/PPS/SEI 等非数据帧，这些数据将被解码器内部消化，不会有对应的图像帧输出，自然也不会有 PTS 输出。如果用户对送帧和收帧进行计数，则可能出现收帧的次数少于送帧次数的情况。

## 取帧方式

解码器的后级模块可选阻塞或非阻塞的方式接收解码图像。阻塞方式的典型用法是用户为每路码流创建一个取帧线程，在线程中调用 `GetFrame()` 接口取帧。当解码路数很多时，阻塞方式线程开销较大，此时效率更高的方法是用一个线程处理所有解码通道的数据，此方法需要使用（待发布的）`Select()` 接口配合 `GetFrame()` 接口使用，其中 `Select()` 接口承担阻塞功能，并监控所有解码通道的活动情况。当一个或多个解码通道有数据可用时，`Select()` 立即退出阻塞

并返回待处理的通道列表，然后 GetFrame()接口以非阻塞的方式遍历通道列表，完成取帧。

当解码器内部没有申请到可用的输出 buffer 时，解码过程会暂停，当有空闲输出 buffer 可用时会自动恢复解码。如果用户为解码器配置的 buffer 总数不满足解码所需的最小数量要求，或者用户取帧后没有及时归还 buffer，都可能导致解码器暂停工作。

## 解码通道 (Channel)

每一路解码输入仅支持一路解码后的输出,即一个输入对应一个输出，H264 解码可以支持的最大分辨率为 1080P，JPEG 解码支持的最大分辨率为 16K。

## 解码能力 (Specification)

AX620A 解码器的输出图像只支持一种 YUV 格式，即 NV12。

H264 最大分辨率支持 1080p FHD (1920x1088)，最小分辨率支持 48x48。

JPEG 支持最大分辨率 16K (16384x16384)，最小分辨率支持 48x48。

### ！ 注意：

AX620A 芯片的编码能力和解码能力不对称，使用时应注意。

## 1.2.2 软件流程

VDEC 的软件解码处理流程图如下所示。

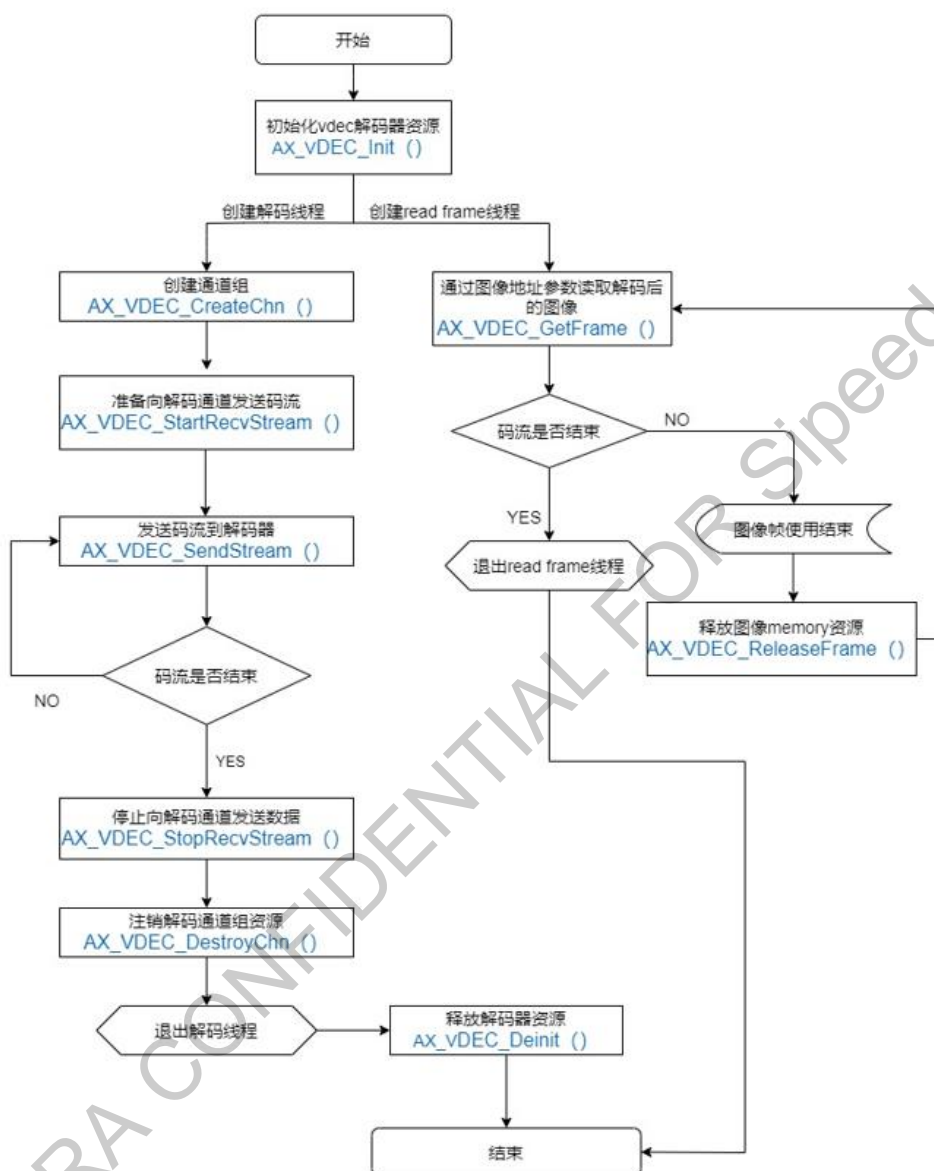


图1-2 解码流程图

典型解码流程包括：初始化解码器硬件及模块公共软件资源，创建解码实例，配置解码通道参数，配置协议参数，设置通道模式，开始发送码流解码，解码完成停止发送码流，注销解码实例。

解码模块提供的 API 接口如下：

## 2 API 简介

- AX\_VDEC\_Init：初始化解码硬件资源。
- AX\_VDEC\_Deinit：解码结束释放解码硬件资源。
- AX\_VDEC\_CreateGrp：创建解码通道。
- AX\_VDEC\_DestroyGrp：销毁解码通道。
- AX\_VDEC\_StartRecvStream：触发解码通道开始接收码流数据。
- AX\_VDEC\_SendStream：发送码流数据到解码通道。
- AX\_VDEC\_StopRecvStream：解码通道停止接收码流数据。
- AX\_VDEC\_SetDisplayMode：设置显示模式参数。
- AX\_VDEC\_GetDisplayMode：获取显示模式参数。
- AX\_VDEC\_SetGrpAttr：设置解码通道参数。
- AX\_VDEC\_GetGrpAttr：获取解码通道参数。
- AX\_VDEC\_SetProtocolParam：设置协议相关的参数。
- AX\_VDEC\_GetProtocolParam：获取协议相关的参数。
- AX\_VDEC\_GetFrame：获取解码通道解码后的图像。
- AX\_VDEC\_ReleaseFrame：释放解码图像缓存。
- AX\_VDEC\_AttachPool：将解码通道绑定到某个缓存 VB 池中。
- AX\_VDEC\_DetachPool：将解码通道从某个缓存 VB 池中解绑定。

AX\_VDEC\_Init

3 API 定义

【描述】

初始化解码硬件资源及模块公共资源。

【语法】

```
AX_S32 AX_VDEC_Init()
```

【参数】

无

【返回值】

返回值	描述
非 0	失败
0	成功

【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

【注意】

- 如果硬件初始化失败或未预期的错误，返回 AX\_ERR\_VDEC\_UNKNOWN。
- 如果模块公共资源初始化失败，返回 AX\_ERR\_VDEC\_RUN\_ERROR。

## AX\_VDEC\_Deinit

### 【描述】

解码结束释放解码硬件及模块公共资源。

### 【语法】

```
AX_S32 AX_VDEC_Deinit( )
```

### 【参数】

无

### 【返回值】

返回值	描述
非 0	失败
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 如果还有通道没有销毁，返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 如果释放资源失败，返回错误码 AX\_ERR\_VDEC\_UNKNOWN。

## AX\_VDEC\_CreateGrp

### 【描述】

创建解码通道。

### 【语法】

```
AX_S32 AX_VDEC_CreateGrp(AX_VDEC_GRP VdGrp, const AX_VDEC_GRP_ATTR_S
*pstAttr)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	视频解码通道号	输入
pstAttr	解码通道属性参数	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号须保证在最大通道号范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 如果参数 pstAttr 为空，返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 如果模块软件初始化失败返回错误码 AX\_ERR\_VDEC\_RUN\_ERROR。
- 创建视频解码通道前必须保证通道未被创建，否则返回错误码 AX\_ERR\_VDEC\_EXIST。

- 通道的 type 必须为 H264/JPEG，否则返回错误码 AX\_ERR\_VDEC\_NOT\_SUPPORT。
- 如果分配内存失败，返回错误码 AX\_ERR\_VDEC\_NOMEM。
- 创建通道前必须保证系统资源已经初始化，否则返回错误码 AX\_ERR\_VDEC\_SYS\_NOTREADY。
- 创建解码通道时通过设置 u32FrameBufCnt 值设置解码图像 buffer 的个数，buffer 个数至少为参考帧+2。通过调用 AX\_POOL\_CreatePool API 创建专属于 VDEC 模块的 UserVB 池，VB 块的个数不小于 u32FrameBufCnt 值，VB 块的大小可以通过 AX\_VDEC\_GetPicBufferSize（）接口获得，具体计算方法可参见 ax\_buffer\_tool.h。

AXERA CONFIDENTIAL FOR CIPED



## AX\_VDEC\_DestroyGrp

### 【描述】

销毁解码通道。

### 【语法】

```
AX_S32 AX_VDEC_DestroyGrp(AX_VDEC_GRP VdGrp)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号在最大通道号范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_StartRecvStream

### 【描述】

触发解码通道开始接收码流数据，并开始解码。

### 【语法】

```
AX_S32 AX_VDEC_StartRecvStream(AX_VDEC_GRP VdGrp)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_StopRecvStream

### 【描述】

解码通道停止接收码流数据，并停止解码。

### 【语法】

```
AX_S32 AX_VDEC_StopRecvStream(AX_VDEC_GRP VdGrp)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_SendStream

### 【描述】

发送码流数据到解码通道。

### 【语法】

```
AX_S32 AX_VDEC_SendStream(AX_VDEC_GRP VdGrp, const AX_VDEC_STREAM_S
*pstStream, AX_S32 s32MilliSec)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
pstStream	编码码流结构体指针	输入
s32MilliSec	超时选项，取值范围： -1：阻塞方式 0：非阻塞方式 正值：超时时间，没有上限值，以 ms 为单位	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 `AX_ERR_VDEC_INVALID_CHNID`。
- 码流结构体指针不能为空，否则返回错误码 `AX_ERR_VDEC_NULL_PTR`。
- 硬件资源应已经初始化，否则返回错误码 `AX_ERR_VDEC_NOT_PERM`。
- 发送数据前应已经调用 `AX_VDEC_StartRecvStream()` 接口启动接收码流，否则返回错误码 `AX_ERR_VDEC_NOT_PERM`。
- 发送数据前应保证通道已经创建成功，否则返回错误码 `AX_ERR_VDEC_UNEXIST`。
- 如果发送码流长度为非法值，返回错误码 `AX_ERR_VDEC_NOT_PERM`。
- 对于 H264 协议可能返回以下错误码：
  - 如果没有从码流中解析到必要的信息，导致无法配置解码参数，则返回错误码 `AX_ERR_VDEC_NOT_CONFIG`；
  - 如果是不支持的码流会返回 `AX_ERR_VDEC_NOT_SUPPORT`；
  - 如果发生解码超时则会返回错误码 `AX_ERR_VDEC_TIMED_OUT`；
  - 如果送入的码流有错误，则会返回错误码 `AX_ERR_VDEC_STRM_ERROR`。
- 在发送完所有码流后，用户可以发送一个 `bEndFrame` 为 1 的空码流包，表示当前码流文件已结束，解码器会把所有码流全部解完并输出全部图像。除此之外，其他情况应该把 `bEndStream` 置为 0。

## AX\_VDEC\_SetDisplayMode

### 【描述】

设置显示模式参数。

### 【语法】

```
AX_S32 AX_VDEC_SetDisplayMode(AX_VDEC_GRP VdGrp, AX_VDEC_DISPLAY_MODE_E  
enDisplayMode)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
enDisplayMode	显示模式	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_GetDisplayMode

### 【描述】

获取显示模式参数。

### 【语法】

```
AX_S32 AX_VDEC_GetDisplayMode(AX_VDEC_GRP VdGrp, AX_VDEC_DISPLAY_MODE_E  
*penDisplayMode)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
penDisplayMode	显示模式	输出

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

AX\_VDEC\_SetGrpAttr

【描述】

设置通道组参数。

【语法】

```
AX_S32 AX_VDEC_SetGrpAttr(AX_VDEC_GRP VdGrp, AX_VDEC_GRP_ATTR_S *pstAttr)
```

【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
pstAttr	通道属性参数	输入

【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道属性结构体指针 pstAttr 不能为空，否则返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。



## AX\_VDEC\_GetGrpAttr

### 【描述】

获取通道组参数。

### 【语法】

```
AX_S32 AX_VDEC_GetGrpAttr(AX_VDEC_GRP VdGrp, AX_VDEC_GRP_ATTR_S *pstAttr)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
pstAttr	通道属性参数	输出

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道属性结构体指针 pstAttr 不能为空，否则返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_SetProtocolParam

### 【描述】

设置协议相关的参数。

### 【语法】

```
AX_S32 AX_VDEC_SetProtocolParam(AX_VDEC_GRP VdGrp, const  
AX_VDEC_PRTCL_PARAM_S *pstParam)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
pstParam	协议相关参数	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALIDL\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 结构体指针 pstParam 不能为空，否则返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_GetProtocolParam

### 【描述】

获取协议相关的参数。

### 【语法】

```
AX_S32 AX_VDEC_GetProtocolParam(AX_VDEC_GRP VdGrp, AX_VDEC_PRTCL_PARAM_S
*pstParam)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道组号	输入
pstParam	协议相关参数	输出

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 结构体指针 pstParam 不能为空，否则返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_GetFrame

### 【描述】

获取解码通道解码后的图像。

### 【语法】

```
AX_S32 AX_VDEC_GetFrame(AX_VDEC_GRP VdGrp, AX_VIDEO_FRAME_INFO_S  
*pstFrameInfo, AX_S32 s32MilliSec)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
pstFrameInfo	获取解码图像信息	输出
s32MilliSec	超时选项，取值范围： -1：阻塞 0：非阻塞 正值：超时时间，没有上限值，以 ms 为单位	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 结构体指针 pstFrameInfo 不能为空，否则返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。如果在获取图像的过程中销毁通道，会立刻返回错误码 AX\_ERR\_VDEC\_UNEXIST。
- 在以非阻塞方式获取解码图像时，如果缓冲区内没有图像，会立刻返回错误码 AX\_ERR\_VDEC\_BUF\_EMPTY。

AXERA CONFIDENTIAL FOR Sipeed

## AX\_VDEC\_ReleaseFrame

### 【描述】

释放视频图像缓存。

### 【语法】

```
AX_S32 AX_VDEC_ReleaseFrame(AX_VDEC_GRP VdGrp, const AX_VIDEO_FRAME_INFO_S
*pstFrameInfo)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
pstFrameInfo	解码图像信息	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 硬件资源应已经初始化，否则返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 结构体指针 pstFrameInfo 不能为空，否则返回错误码 AX\_ERR\_VDEC\_NULL\_PTR。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_AttachPool

### 【描述】

将解码通道绑定到某个缓存 VB 池中。

### 【语法】

```
AX_S32 AX_VDEC_AttachPool(AX_VDEC_GRP VdGrp, AX_POOL Pool)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入
Pool	缓存 VB 池 ID	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 必须保证已创建 VB 池的 Pool Id 有效，如果 VB 池 ID 为 AX\_INVALID\_POOLID，则返回错误码 AX\_ERR\_VDEC\_ILLEGAL\_PARAM。
- 用户必须要调用 AX\_POOL\_CreatePool 创建一个缓存 VB 池，再通过调用接口 AX\_VDEC\_AttachPool 把当前解码通道绑定到固定 Pool Id 上，不允许一个解码通道绑定多个 Pool Id。

- 通道应已经创建成功，否则返回错误码 `AX_ERR_VDEC_UNEXIST`。
- 如果通道正在销毁，返回错误码 `AX_ERR_VDEC_NOT_PERM`。

AXERA CONFIDENTIAL FOR Sipeed



## AX\_VDEC\_DetachPool

### 【描述】

将解码通道从某个缓存 VB 池中解绑定。

### 【语法】

```
AX_S32 AX_VDEC_DetachPool (AX_VDEC_GRP VdGrp)
```

### 【参数】

参数名称	描述	输入/输出
VdGrp	解码通道号	输入

### 【返回值】

返回值	描述
非 0	失败，其值为错误码
0	成功

### 【需求】

- 头文件：ax\_vdec\_api.h
- 库文件：libax\_vdec.so

### 【注意】

- 通道号应在最大通道范围内，否则返回错误码 AX\_ERR\_VDEC\_INVALID\_CHNID。
- 如果通道正在销毁，返回错误码 AX\_ERR\_VDEC\_NOT\_PERM。
- 通道应已经创建成功，否则返回错误码 AX\_ERR\_VDEC\_UNEXIST。

## AX\_VDEC\_GRP\_ATTR\_S

## 4 数据结构

### 【说明】

视频解码通道属性结构体。

### 【定义】

```
typedef struct axVDEC_GRP_ATTR_S {  
  
    AX_PAYLOAD_TYPE_E enType;  
  
    AX_VDEC_MODE_E enMode;  
  
    AX_LINK_MODE_E enLinkMode;  
  
    AX_U32 u32PicWidth;  
  
    AX_U32 u32PicHeight;  
  
    AX_U32 u32FrameHeight;  
  
    AX_U32 u32StreamBufSize;  
  
    AX_U32 u32FrameBufSize;  
  
    AX_U32 u32FrameBufCnt;  
  
    AX_S32 s32DestroyTimeout;  
  
    Union  
  
    {  
  
        AX_VDEC_ATTR_VIDEO_S stVdecVideoAttr;  
  
    }  
}
```

```
} AX_VDEC_GRP_ATTR_S;
```

**【成员】**

成员名称	描述
enType	解码协议类型
enMode	送解码数据流方式
enLinkMode	设置 link 模式
u32PicWidth	通道支持的解码图像最大宽度（以像素为单位）
u32PicHeight	通道支持的解码图像最大高度（以像素为单位）
u32FrameHeight	通道输出图像的总高度，当此值大于图像高度时，图像尾部区域可视为填充数据
u32StreamBufSize	码流缓存大小
u32FrameBufSize	用户设置的每个输出 buffer 大小
u32FrameBufCnt	用户需要设置输出 buffer 个数,每路开的 output buffer 个数最大值为 34 个，输出 buffer 个数由客户控制
s32DestroyTimeout	销毁通道时，等待输出队列为空的延时选项，单位 ms。 默认值 0，-1 表示阻塞模式。
stVdecVideoAttr	解码属性

**【相关数据类型及接口】**

[AX\\_VDEC\\_CreateGrp](#)

## AX\_VDEC\_STREAM\_S

### 【说明】

视频解码的码流结构体定义。

### 【定义】

```
typedef struct axVDEC_STREAM_S
{
    AX_U32  u32Len;

    AX_U64  u64PTS;

    AX_BOOL bEndOfFrame;

    AX_BOOL bEndOfStream;

    AX_BOOL bDisplay;

    AX_U8*  ATTRIBUTE pu8Addr;
}AX_VDEC_STREAM_S;
```

### 【成员】

成员名称	描述
pu8Addr	送入视频码流的地址
u32Len	码流长度
u64PTS	时间戳
bEndOfFrame	当前帧是否结束
bEndOfStream	是否发完所有码流
bDisplay	当前帧是否显示

### 【注意】

U64PTS 是每帧的时间戳，只有在按照帧的方式送流解码的情况下有效。

**！ 注意：**

H264 协议中如果开始送帧的 header 部分单独送入解码，是不会有解码图像输出该时间戳的。

**【相关数据类型及接口】**

[AX\\_VDEC\\_SendStream](#)

AXERA CONFIDENTIAL FOR Sipeed

# AX\_VDEC\_DISPLAY\_MODE\_E

【说明】

定义显示模式枚举。

【定义】

```
typedef enum axVIDEO_DISPLAY_MODE_E{  
  
    VIDEO_DISPLAY_MODE_PREVIEW = 0x0,  
  
    VIDEO_DISPLAY_MODE_PLAYBACK = 0x1,  
  
    VIDEO_DISPLAY_MODE_BUTT  
  
} AX_VDEC_DISPLAY_MODE_E;
```

【成员】

成员名称	描述
VIDEO_DISPLAY_MODE_PREVIEW	预览模式
VIDEO_DISPLAY_MODE_PLAYBACK	回放模式

【相关数据类型及接口】

- [AX\\_VDEC\\_SetDisplayMode](#)
- [AX\\_VDEC\\_GetDisplayMode](#)

## AX\_VDEC\_PARAM\_VIDEO\_S

### 【说明】

视频解码的属性定义。

### 【定义】

```
typedef struct axVDEC_PARAM_VIDEO_S
```

```
{  
  
    AX_S32                s32ErrThreshold;  
  
    AX_VDEC_DEC_MODE_E    enDecMode;  
  
    AX_VDEC_OUTPUT_ORDER_E enOutputOrder;  
  
    AX_COMPRESS_MODE_E    enCompressMode;  
  
    AX_VSCAN_FORMAT_E     enVideoFormat;  
  
}AX_VDEC_PARAM_VIDEO_S;
```

### 【成员】

成员名称	描述
S32ErrThreshold	通道错误率阈值[0,100]
enDecMode	解码模式（H264/JPEG）
enOutputOrder	解码图像输出顺序
enCompressMode	压缩模式
enVideoFormat	解码图像数据格式

## AX\_VDEC\_DEC\_MODE\_E

### 【说明】

解码模式定义。

### 【定义】

```
typedef enum axVIDEO_DEC_MODE_E {  
  
    VIDEO_DEC_NODE_IPB = 0,  
  
    VIDEO_DEC_NODE_IP,  
  
    VIDEO_DEC_NODE_I,  
  
    VIDEO_DEC_NODE_BUTT  
} AX_VDEC_DEC_MODE_E;
```

### 【成员】

成员名称	描述
VIDEO_DEC_NODE_IPB	包括 I/P/B 帧码流
VIDEO_DEC_NODE_IP	包含 I/P 帧码流
VIDEO_DEC_NODE_I	只有 I 帧码流



## AX\_VDEC\_OUTPUT\_ORDER\_E

### 【说明】

解码输出图像的顺序定义。

### 【定义】

```
typedef enum axVIDEO_OUTPUT_ORDER_E {  
  
    VIDEO_OUTPUT_ORDER_DISP = 0,  
  
    VIDEO_OUTPUT_ORDER_DEC,  
  
    VIDEO_OUTPUT_ORDER_BUTT  
} AX_VDEC_OUTPUT_ORDER_E;
```

### 【成员】

成员名称	描述
VIDEO_OUTPUT_ORDER_DISP	按照显示顺序输出
VIDEO_OUTPUT_ORDER_DEC	按照解码顺序输出

## AX\_VDEC\_MODE\_E

### 【说明】

解码输入图像的方式定义。

### 【定义】

```
typedef enum axVIDEO_MODE_E{  
  
    VIDEO_MODE_STREAM = 0,  
  
    VIDEO_MODE_FRAME,  
  
    VIDEO_MODE_COMPAT,  
  
    VIDEO_MODE_BUTT  
} AX_VDEC_MODE_E;
```

### 【成员】

成员名称	描述
VIDEO_MODE_STREAM	按照流方式输入码流
VIDEO_MODE_FRAME	按照单帧方式输入码流
VIDEO_MODE_COMPAT	一帧多个 slice 发送，直到 bEndOffFrame 为 TRUE 结束

## AX\_VIDEO\_FRAME\_INFO\_S

### 【说明】

定义解码图像信息。

### 【定义】

```
typedef struct axVIDEO_FRAME_INFO_S {
```

```
    AX_VIDEO_FRAME_S    stVFrame;
```

```
    AX_U32               u32PoolId;
```

```
    AX_U32               u32BlkId[2];
```

```
    AX_MOD_ID_E          enModId;
```

```
    AX_BOOL              bEof;
```

```
} AX_VIDEO_FRAME_INFO_S;
```

### 【成员】

成员名称	描述
StVFrame	视频图像帧
u32PoolId	视频缓存池 ID，暂未使用
u32BlkId[2]	视频缓存块 ID
enModId	当前帧数据是由哪一个模块写出的
bEof	是否结束标志

## AX\_MOD\_ID\_E

### 【说明】

定义各模块 ID 类型枚举。

### 【定义】

```
typedef enum {
```

```
    AX_ID_ISP      = 0x01,
```

```
    AX_ID_CE       = 0x02,
```

```
    AX_ID_DPU      = 0x03,
```

```
    AX_ID_VDSP     = 0x04,
```

```
    AX_ID_EFUSE    = 0x05,
```

```
    AX_ID_NPU      = 0x06,
```

```
    AX_ID_VENC     = 0x07,
```

```
    AX_ID_VDEC     = 0x08,
```

```
    AX_ID_JENC     = 0x09,
```

```
    AX_ID_JDEC     = 0x0a,
```

```
    AX_ID_SYS      = 0x0b,
```

```
    AX_ID_AENC     = 0x0c,
```

```
    AX_ID_IVPS     = 0x0d,
```

```
    AX_ID_MIPI     = 0x0e,
```

```
    AX_ID_ADEC     = 0x0f,
```

```
    AX_ID_DMA      = 0x10,
```

```
    AX_ID_BUTT,
```

```
} AX_MOD_ID_E;
```

AXERA CONFIDENTIAL FOR Sipeed

## AX\_VIDEO\_FRAME\_S

### 【说明】

定义每个输出通道解码图像信息。

### 【定义】

```
typedef struct axVIDEO_FRAME_S {  
  
    AX_U32          u32Width;  
  
    AX_U32          u32Height;  
  
    AX_IMG_FORMAT_E  enImgFormat;  
  
    AX_VSCAN_FORMAT_E  enVscanFormat;  
  
    AX_COMPRESS_MODE_E  enCompressMode;  
  
    AX_U32          u32HeaderStride[3];  
  
    AX_U32          u32PicStride[3];  
  
    AX_U32          u32ExtStride[3];  
  
    AX_U64          u64HeaderPhyAddr[3];  
  
    AX_U64          u64HeaderVirAddr[3];  
  
    AX_U64          u64PhyAddr[3];  
  
    AX_U64          u64VirAddr[3];  
  
    AX_U64          u64ExtPhyAddr[3];  
  
    AX_U64          u64ExtVirAddr[3];  
  
    AX_S16          s16OffsetTop;  
  
    AX_S16          s16OffsetBottom;  
  
    AX_S16          s16OffsetLeft;
```

```
AX_S16          s16OffsetRight;

AX_U32          u32TimeRef;

AX_U64          u64PTS;

AX_U64          u64SeqNum;

AX_U64          u64UserData;

AX_U64          u64PrivateData;

AX_U32          u32FrameFlag;

AX_U8          u8MetaDataInternal[128];

AX_U8          u8MetaDataUser[128];

AX_U32          u32FrameSize;

} AX_VIDEO_FRAME_S;
```

### 【成员】

成员名称	描述
u32Width	解码图像的宽度
u32Height	解码图像的高度
enImgFormat	解码图像输出格式
enVscanFormat	视频图像像素格式
enCompressMode	解码图像的压缩模式
u32HeaderStride[3]	图像压缩头跨距
u32PicStride[3]	图像数据跨距
u32ExtStride[3]	10bit 数据位宽的图像，有些数据格式的存储方式是前 8bit 和后 2bit 分开存储，这里指后 2bit 数据跨距
u64HeaderPhyAddr[3]	压缩头物理地址
u64HeaderVirAddr[3]	压缩头虚拟地址

成员名称	描述
u64PhyAddr[3]	物理地址
u64VirAddr[3]	虚拟地址
u64ExtPhyAddr[3]	10bit 数据位宽的图像，有些数据格式的存储方式是前 8bit 和后 2bit 分开存储，这里指后 2bit 数据的物理地址
u64ExtVirAddr[3]	10bit 数据位宽的图像，有些数据格式的存储方式是前 8bit 和后 2bit 分开存储，这里指后 2bit 数据的虚拟地址
s16OffsetTop	图像顶部裁剪宽度
s16OffsetBottom	图像底部裁剪宽度
s16OffsetLeft	图像左侧裁剪宽度
s16OffsetRight	图像右侧裁剪宽度
u32TimeRef	图像帧序列号
u64PTS	图像时间戳
u64SeqNum	图像序列号
u64UserData	用户数据
u64PrivateData	私有数据
u32FrameFlag	当前帧标记
u32FrameSize	图像帧大小，未使用



AX\_VDEC\_PRTCL\_PARAM\_S

【说明】

定义解码协议属性。

【定义】

```
typedef struct axVDEC_PRTCL_PARAM_S {  
  
    AX_PAYLOAD_TYPE_E enType;  
  
    union {  
  
        AX_VDEC_H264_PRTCL_PARAM_S stH264PrtclParam;  
  
        AX_VDEC_H265_PRTCL_PARAM_S stH265PrtclParam;  
  
    } ;  
  
} AX_VDEC_PRTCL_PARAM_S;
```

【成员】

成员名称	描述
enType	协议类型
stH264PrtclParam	H264 协议参数
stH265PrtclParam	H265 协议参数（AX620A 不适用）

# AX\_VDEC\_H264\_PRTCL\_PARAM\_S

【说明】

定义 H264 协议属性。

【定义】

```
typedef struct axH264_PRTCL_PARAM_S{  
  
    AX_S32  s32MaxSliceNum;  
  
    AX_S32  s32MaxSpsNum;  
  
    AX_S32  s32MaxPpsNum;  
  
}AX_VDEC_H264_PRTCL_PARAM_S;
```

【成员】

成员名称	描述
s32MaxSliceNum	该通道支持的最大 slice 个数
S32MaxSpsNum	该通道支持的最大 SPS 个数
S32MaxPpsNum	该通道支持的最大 PPS 个数

## AX\_VDEC\_H265\_PRTCL\_PARAM\_S

### 【说明】

定义 H265 协议属性（AX620A 不适用）。

### 【定义】

```
typedef struct axH265_PRTCL_PARAM_S{  
  
    AX_S32  s32MaxSliceSegmentNum;  
  
    AX_S32  s32MaxVpsNum;  
  
    AX_S32  s32MaxSpsNum;  
  
    AX_S32  s32MaxPpsNum;  
  
}AX_VDEC_H265_PRTCL_PARAM_S;
```

### 【成员】

成员名称	描述
s32MaxSliceSegmentNum	该解码通道支持的最大 SliceSegment 个数
s32MaxVpsNum	该解码通道支持的最大 VPS 个数
s32MaxSpsNum	该解码通道支持的最大 SPS 个数
s32MaxPpsNum	该解码通道支持的最大 PPS 个数

## AX\_PAYLOAD\_TYPE\_E

### 【说明】

定义解码通道类型枚举。

### 【定义】

```
typedef enum
```

```
{
```

```
    PT_PCMU = 0,
```

```
    PT_1016 = 1,
```

```
    PT_G721 = 2,
```

```
    PT_GSM = 3,
```

```
    PT_G723 = 4,
```

```
    PT_DVI4_8K = 5,
```

```
    PT_DVI4_16K = 6,
```

```
    PT_LPC = 7,
```

```
    PT_PCMA = 8,
```

```
    PT_G722 = 9,
```

```
    PT_S16BE_STEREO = 10,
```

```
    PT_S16BE_MONO = 11,
```

```
    PT_QCELP = 12,
```

```
    PT_CN = 13,
```

```
    PT_MPEGAUDIO = 14,
```

```
    PT_G728 = 15,
```

PT\_DVI4\_3 = 16,  
PT\_DVI4\_4 = 17,  
PT\_G729 = 18,  
PT\_G711A = 19,  
PT\_G711U = 20,  
PT\_G726 = 21,  
PT\_G729A = 22,  
PT\_LPCM = 23,  
PT\_CelB = 25,  
PT\_JPEG = 26,  
PT\_CUSM = 27,  
PT\_NV = 28,  
PT\_PICW = 29,  
PT\_CPV = 30,  
PT\_H261 = 31,  
PT\_MPEGVIDEO = 32,  
PT\_MPEG2TS = 33,  
PT\_H263 = 34,  
PT\_SPEG = 35,  
PT\_MPEG2VIDEO = 36,  
PT\_AAC = 37,  
PT\_WMA9STD = 38,  
PT\_HEAAC = 39,

PT\_PCM\_VOICE = 40,

PT\_PCM\_AUDIO = 41,

PT\_MP3 = 43,

PT\_ADPCMA = 49,

PT\_AEC = 50,

PT\_X\_LD = 95,

PT\_H264 = 96,

PT\_D\_GSM\_HR = 200,

PT\_D\_GSM\_EFR = 201,

PT\_D\_L8 = 202,

PT\_D\_RED = 203,

PT\_D\_VDVI = 204,

PT\_D\_BT656 = 220,

PT\_D\_H263\_1998 = 221,

PT\_D\_MP1S = 222,

PT\_D\_MP2P = 223,

PT\_D\_BMPEG = 224,

PT\_MP4VIDEO = 230,

PT\_MP4AUDIO = 237,

PT\_VC1 = 238,

PT\_JVC\_ASF = 255,

PT\_D\_AVI = 256,

PT\_DIVX3 = 257,

```
PT_AVS = 258,  
  
PT_REAL8 = 259,  
  
PT_REAL9 = 260,  
  
PT_VP6 = 261,  
  
PT_VP6F = 262,  
  
PT_VP6A = 263,  
  
PT_SORENSEN = 264,  
  
PT_H265 = 265,  
  
PT_VP8 = 266,  
  
PT_MVC = 267,  
  
PT_PNG = 268,  
  
PT_AVS2 = 269,  
  
PT_VP7 = 270,  
  
PT_VP9 = 271  
} AX_PAYLOAD_TYPE_E;
```

VDEC API 错误码如下表所示。

## 5 错误码

表5-1 VDEC API 错误码

错误代码	宏定义	描述
0x80080104	AX_ERR_VDEC_INVALID_CHNID	无效通道 ID
0x8008010A	AX_ERR_VDEC_ILLEGAL_PARAM	输入参数非法
0x8008010B	AX_ERR_VDEC_NOT_PTR	空指针
0x8008010C	AX_ERR_VDEC_BAD_ADDR	无效地址
0x80080110	AX_ERR_VDEC_SYS_NOTREADY	系统未就绪
0x80080111	AX_ERR_VDEC_BUSY	目标对象正忙
0x80080112	AX_ERR_VDEC_NOT_INIT	模块未初始化
0x80080113	AX_ERR_VDEC_NOT_CONFIG	参数未配置
0x80080114	AX_ERR_VDEC_NOT_SUPPORT	不支持该功能
0x80080115	AX_ERR_VDEC_NOT_PERM	操作不允许
0x80080116	AX_ERR_VDEC_EXIST	目标对象已存在
0x80080117	AX_ERR_VDEC_UNEXIST	目标对象不存在
0x80080118	AX_ERR_VDEC_NOMEM	申请堆内存失败
0x80080119	AX_ERR_VDEC_NOBUF	申请帧缓存失败
0x80080120	AX_ERR_VDEC_BUF_EMPTY	缓存中无数据
0x80080121	AX_ERR_VDEC_BUF_FULL	缓存区已满
0x80080127	AX_ERR_VDEC_TIMED_OUT	操作超时
0x80080128	AX_ERR_VDEC_FLOW_END	流程终止
0x80080129	AX_ERR_VDEC_UNKNOWN	未分类的异常



错误代码	宏定义	描述
0x80080180	AX_ERR_VDEC_RUN_ERROR	运行出错
0x80080181	AX_ERR_VDEC_STRM_ERROR	解析码流时出错

AXERA CONFIDENTIAL FOR Sipeed

## 【调试信息】

## 6 调试信息

# cat /proc/ax\_proc/vdec

```

/ # cat /proc/ax_proc/vdec
----- VDEC VERSION -----
version:v0.31.0 build:Apr 7 2022 15:27:47

----- MODULE PARAM -----
MaxGroupNum  H264MaxPicwid  H264MaxPichei  H264MinPicwid  H264MinPichei  JPEGMaxPicwid  JPEGMaxPichei  JPEGMinPicwid  JPEGMinPichei
16            1920            1088            48              48              16384          16384          48              48
----- MODULE GROUP STATUS -----
workingGroupNum
1
GroupID      Fps      DecType      StreamBufSize  GrpStatus      FrameBufCnt
0           187.06    96           8388608        2              8
SendFrameNum GetFrameNum ReleaseFrameNum FreeOutBufNum
316          309          309          2

```

## 【调试信息分析】

记录当前视频解码通道属性配置及状态信息

## 【参数说明】

参数		描述
VDEC VERSION	Version	模块版本信息
MODULE PARAM	MaxGroupNum	最大解码通道组数
	H264MaxPicWid	H264 协议支持图像的最大宽度
	H264MaxPicHei	H264 协议支持图像的最大高度
	H264MinPicWid	H264 协议支持图像的最小宽度
	H264MinPicHei	H264 协议支持图像的最小高度

	JPEGMaxPicWid	JPEG 协议支持图像的最大宽度
	JPEGMaxPicHei	JPEG 协议支持图像的最大高度
	JPEGMinPicWid	JPEG 协议支持图像的最小宽度
	JPEGMinPicHei	JPEG 协议支持图像的最小高度
MODULE	WorkingGroupNum	正在工作的通道组数
GROUP	GroupID	通道号
STATUS	Fps	解码帧率
	DecType	码流协议
	StreamBufSize	输入码流 <b>buffer</b> 大小
	GrpStatus	解码通道处于哪个状态阶段
	FrameBufCnt	输出 <b>buffer</b> 个数
	SendFrameNum	发送解码帧数
	GetFrameNum	用户取走图像帧数
	ReleaseFrameNum	用户释放 <b>buffer</b> 个数
	FreeOutBufNum	空闲输出 <b>buffer</b> 个数