

# 视频解码

---

文件标识: RK-SYS1-MPI-VDEC

发布版本: V0.2.0

日期: 2021.1

文件密级: ☐绝密 ☐秘密 ☐内部资料 ☒公开

## 免责声明

本文档按“现状”提供, 瑞芯微电子股份有限公司 (“本公司”, 下同) 不对本文档的任何陈述、信息和内容的准确性、可靠性、完整性、适销性、特定目的性和非侵权性提供任何明示或暗示的声明或保证。本文档仅作为使用指导的参考。

由于产品版本升级或其他原因, 本文档将可能在未经任何通知的情况下, 不定期进行更新或修改。

## 商标声明

“Rockchip”、“瑞芯微”、“瑞芯”均为本公司的注册商标, 归本公司所有。

本文档可能提及的其他所有注册商标或商标, 由其各自拥有者所有。

## 版权所有 © 2021 瑞芯微电子股份有限公司

超越合理使用范畴, 非经本公司书面许可, 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。

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd.

地址: 福建省福州市铜盘路软件园A区18号

网址: [www.rock-chips.com](http://www.rock-chips.com)

客户服务电话: +86-4007-700-590

客户服务传真: +86-591-83951833

客户服务邮箱: [fae@rock-chips.com](mailto:fae@rock-chips.com)

---

## 前言

### 概述

VDEC 模块提供驱动视频解码硬件工作的 MPI 接口, 实现视频解码功能。

### 产品版本

芯片名称	内核版本
RK356X	4.19

### 读者对象

本文档 (本指南) 主要适用于以下工程师:

技术支持工程师

软件开发工程师

修订记录

版本号	作者	修改日期	修改说明
V0.1.0	杨文杰	2020-01-09	初始版本
V0.2.0	杨文杰	2020-01-23	完善数据结构定义

目录

视频解码

目录

功能描述

基本概念

模块参数

API 参考

- RK\_MPI\_VDEC\_CreateChn
- RK\_MPI\_VDEC\_DestroyChn
- RK\_MPI\_VDEC\_ResetChn
- RK\_MPI\_VDEC\_GetChnAttr
- RK\_MPI\_VDEC\_SetChnAttr
- RK\_MPI\_VDEC\_StartRecvStream
- RK\_MPI\_VDEC\_StopRecvStream
- RK\_MPI\_VDEC\_QueryStatus
- RK\_MPI\_VDEC\_SendStream
- RK\_MPI\_VDEC\_GetFrame
- RK\_MPI\_VDEC\_ReleaseFrame
- RK\_MPI\_VDEC\_SetChnParam
- RK\_MPI\_VDEC\_GetChnParam
- RK\_MPI\_VDEC\_SetDisplayMode
- RK\_MPI\_VDEC\_GetChnParam

数据类型

- VDEC\_MAX\_CHN\_NUM
- VDEC\_CHN\_ATTR\_S
- VDEC\_CHN\_PARAM\_S
- VDEC\_STREAM\_S
- VIDEO\_FRAME\_INFO\_S
- VDEC\_CHN\_STATUS\_S

错误码

功能描述

基本概念

- 码流发送方式  
VDEC 解码器提供两种码流发送方式：

1. 流式发送 (VIDEO\_MODE\_STREAM)：用户每次可发送任意长度码流到解码器，由解码器内部完成一帧码流的识别过程。须注意，对 H.264/H.265 而言，在收到下一帧码流才能识别当前帧码流的结束，所以在该发送模式下，输入一帧 H.264/H.265 码流，不能希望马上开始解码图像。
  2. 按帧发送 (VIDEO\_MODE\_FRAME)：用户每次发送完整一帧码流到解码器，每调用一次发送接口，解码器就认为该帧码流已经结束，开始解码图像，因此需保证每次调用发送接口发送的码流必须为一帧，否则会出现解码错误。通过该发送方式可以达到快速解码的目的。  
码流发送方式 mode 可在接口 RK\_MPI\_VDEC\_CreateChn 中设置。
- 时间戳 (PTS) 处理  
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(RK\_MPI\_VENC\_SendFrame)中用户送入的 PTS，解码器不会更改此值；如果用户配置的 PTS 值为 0，则表示用户不进行帧率控制，而是由视频输出模块 (VO) 进行帧率控制；  
注意：不能出现 PTS 值为 0 和非 0 混合的情况。  
在模式 RK\_MPI\_VENC\_SendFrame 下发送码流时，解码输出图像的 PTS 统一设为 0，表示用户不进行帧率控制，而是由视频输出模块 (VO) 进行帧率控制。

## 模块参数

- vdec\_max\_chn\_num  
VDEC 模块支持的最大通道数。用户可根据业务场景需求配置一个合理的最大解码通道数，以减少通道上下文相关的内存占用。取值范围是[1, VDEC\_MAX\_CHN\_NUM]。

## API 参考

该功能模块为用户提供以下 API：

- RK\_MPI\_VDEC\_CreateChn：创建视频解码通道。
- RK\_MPI\_VDEC\_DestroyChn：销毁视频解码通道。
- RK\_MPI\_VDEC\_ResetChn：复位解码通道。
- RK\_MPI\_VDEC\_GetChnAttr：获取视频解码通道属性。
- RK\_MPI\_VDEC\_SetChnAttr：设置视频解码通道属性。
- RK\_MPI\_VDEC\_StartRecvStream：解码器开始接收用户发送的码流。
- RK\_MPI\_VDEC\_StopRecvStream：解码器停止接收用户发送的码流。
- RK\_MPI\_VDEC\_QueryStatus：查询解码通道状态。
- RK\_MPI\_VDEC\_SendStream：向视频解码通道发送码流数据。
- RK\_MPI\_VDEC\_GetFrame：获取视频解码通道的解码图像。
- RK\_MPI\_VDEC\_ReleaseFrame：释放视频解码通道的解码图像。
- RK\_MPI\_VDEC\_SetChnParam：设置解码通道参数。
- RK\_MPI\_VDEC\_GetChnParam：获取解码通道参数。
- RK\_MPI\_VDEC\_SetDisplayMode：设置显示模式。
- RK\_MPI\_VDEC\_GetDisplayMode：获取显示模式。

## RK\_MPI\_VDEC\_CreateChn

### 【描述】

创建视频解码通道。

### 【语法】

```
RK_S32 RK_MPI_VDEC_CreateChn(VDEC_CHN VdChn, const VDEC_CHN_ATTR_S *pstAttr)
```

### 【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstAttr	解码通道属性指针。	输入

### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

### 【注意】

- 不支持重复创建。

## RK\_MPI\_VDEC\_DestroyChn

### 【描述】

销毁视频解码通道。

### 【语法】

```
RK_S32 RK_MPI_VDEC_DestroyChn(VDEC_CHN VdChn)
```

### 【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入

### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

### 【注意】

- 通道必须已创建。
- 调用此接口之前，必须先调用 RK\_MPI\_VDEC\_StopRecvStream 禁用此通道。
- 调用此接口时，会一直等待此通道当前任务处理结束才会真正销毁。

## RK\_MPI\_VDEC\_ResetChn

### 【描述】

复位视频解码通道。

### 【语法】

RK\_S32 RK\_MPI\_VDEC\_ResetChn(VDEC\_CHN VdChn)

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。

## RK\_MPI\_VDEC\_GetChnAttr

【描述】

获取视频解码通道属性

【语法】

RK\_S32 RK\_MPI\_VDEC\_GetChnAttr(VDEC\_CHN VdChn, VDEC\_CHN\_ATTR\_S \*pstAttr)

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstAttr	解码通道属性指针。	输出

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。

## RK\_MPI\_VDEC\_SetChnAttr

【描述】

设置视频解码通道属性

【语法】

```
RK_S32 RK_MPI_VDEC_SetChnAttr(VDEC_CHN VdChn, const VDEC_CHN_ATTR_S *pstAttr)
```

#### 【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstAttr	解码通道属性指针。	输入

#### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

#### 【注意】

- 通道必须已创建。
- 必须在调用RK\_MPI\_VDEC\_StartRecvStream之前设置，不支持动态切换属性

## RK\_MPI\_VDEC\_StartRecvStream

#### 【描述】

解码器开始接收用户发送的码流。

#### 【语法】

```
RK_S32 RK_MPI_VDEC_StartRecvStream(VDEC_CHN VdChn)
```

#### 【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入

#### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

#### 【注意】

- 通道必须已创建。

## RK\_MPI\_VDEC\_StopRecvStream

#### 【描述】

解码器停止接收用户发送的码流。

#### 【语法】

RK\_S32 RK\_MPI\_VDEC\_StopRecvStream(VDEC\_CHN VdChn)

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。

## RK\_MPI\_VDEC\_QueryStatus

【描述】

查询解码通道状态。

【语法】

RK\_S32 RK\_MPI\_VDEC\_QueryStatus(VDEC\_CHN VdChn, VDEC\_CHN\_STATUS\_S \*pstStatus)

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstStatus	视频解码通道状态结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。

## RK\_MPI\_VDEC\_SendStream

【描述】

解码器停止接收用户发送的码流。

【语法】

```
RK_S32 RK_MPI_VDEC_SendStream(VDEC_CHN VdChn, const VDEC_STREAM_S *pstStream,
RK_S32 s32MilliSec)
```

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstStream	解码码流数据指针。	输入
s32MilliSec	送码流方式标志。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
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。
- 必须在RK\_MPI\_VDEC\_StartRecvStream之后才能调用
- 以非阻塞方式发送码流，如果码流缓冲区已满，会立刻返回错误码 RK\_ERR\_VDEC\_BUF\_FULL。

## RK\_MPI\_VDEC\_GetFrame

【描述】

获取视频解码通道的解码图像。

【语法】

```
RK_S32 RK_MPI_VDEC_GetFrame(VDEC_CHN VdChn, VIDEO_FRAME_INFO_S *pstFrameInfo,
RK_S32 s32MilliSec)
```

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstFrameInfo	获取的解码图像信息。	输出
s32MilliSec	获取图像方式标志。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
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。
- 通过 RK\_MPI\_VDEC\_GetFrame 获取解码图像数据后，需要通过 RK\_MPI\_VDEC\_ReleaseFrame 来释放。
- 以非阻塞方式获取解码图像，如果缓冲区内无图像，会立刻返回错误码 RK\_ERR\_VDEC\_BUF\_EMPTY。

## RK\_MPI\_VDEC\_ReleaseFrame

【描述】

释放视频解码通道的图像。

【语法】

```
RK_S32 RK_MPI_VDEC_ReleaseFrame(VDEC_CHN VdChn, const VIDEO_FRAME_INFO_S *pstFrameInfo)
```

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstFrameInfo	解码后的图像信息指针，由 RK_MPI_VDEC_GetFrame 接口获取。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。
- 通过 RK\_MPI\_VDEC\_GetFrame 获取解码图像数据后，需要通过 RK\_MPI\_VDEC\_ReleaseFrame 来释放。
- 允许用户非顺序释放，即不按照获取图像的顺序释放图像。

## RK\_MPI\_VDEC\_SetChnParam

【描述】

设置解码通道参数。

【语法】

```
RK_S32 RK_MPI_VDEC_SetChnParam(VDEC_CHN VdChn, const VDEC_CHN_PARAM_S *pstParam)
```

#### 【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstParam	通道参数。	输入

#### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

#### 【注意】

- 通道必须已创建。
- 必须在调用RK\_MPI\_VDEC\_StartRecvStream之前设置，不支持动态切换属性

## RK\_MPI\_VDEC\_GetChnParam

#### 【描述】

获取解码通道参数。

#### 【语法】

```
RK_S32 RK_MPI_VDEC_GetChnParam(VDEC_CHN VdChn, VDEC_CHN_PARAM_S *pstParam)
```

#### 【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstParam	通道参数。	输出

#### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

#### 【注意】

- 通道必须已创建。

## RK\_MPI\_VDEC\_SetDisplayMode

【描述】

设置显示模式。

【语法】

```
RK_S32 RK_MPI_VDEC_SetDisplayMode(VDEC_CHN VdChn, VIDEO_DISPLAY_MODE_E enDisplayMode)
```

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
enDisplayMode	显示模式枚举。	输入

【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

【注意】

- 通道必须已创建。
- 预览模式（VIDEO\_DISPLAY\_MODE\_PREVIEW）：预览模式下 VDEC 绑定的直接后级模块（比如 VPSS）以非阻塞方式接收解码图像，即当 VPSS 的图像 Buffer 满时（解码帧存个数比 VPSS 缓存队列个数多），VPSS 丢弃 VDEC 发送过来的图像，以达到不反压 VDEC 解码的目的，实现实时预览。需要注意的是，当解码帧存个数比 VPSS 缓存队列个数少时，即使开启预览模式，VPSS 还是会反压解码。
- 回放模式（VIDEO\_DISPLAY\_MODE\_PLAYBACK）：回放模式下 VDEC 绑定的直接后级模块（比如 VPSS）以阻塞方式接收解码图像，即当 VPSS 的图像 Buffer 满时，拒绝接收 VDEC 发送过来的图像，VDEC 发现当前图像发送失败后启动图像重新发送机制，直到图像发送成功为止。回放模式下 VDEC 绑定的直接后级模块能够反压 VDEC 解码，以达到不丢弃任何一帧解码图像的回放效果。

RK\_MPI\_VDEC\_GetChnParam

【描述】

获取显示模式。

【语法】

```
RK_S32 RK_MPI_VDEC_GetChnParam(VDEC_CHN VdChn, VDEC_CHN_PARAM_S *pstParam)
```

【参数】

参数名	描述	输入/输出
VdChn	视频解码通道号。取值范围：(0, VDEC_MAX_CHN_NUM)。	输入
pstParam	显示模式枚举指针。	输出

#### 【返回值】

返回值	描述
0	成功。
非0	失败，请参见错误码。

#### 【注意】

- 通道必须已创建。

## 数据类型

视频解码相关数据类型、数据结构定义如下：

- VDEC\_CHN\_ATTR\_S：定义解码通道属性。
- VDEC\_CHN\_PARAM\_S：定义通道参数结构体。
- VDEC\_STREAM\_S：定义视频解码的码流结构体。
- VIDEO\_FRAME\_INFO\_S：定义视频图像帧信息结构体。。
- VDEC\_CHN\_STATUS\_S：定义通道状态结构体。

## VDEC\_MAX\_CHN\_NUM

#### 【说明】

定义解码通道最大个数。

#### 【定义】

```
#define VDEC_MAX_CHN_NUM          64
```

#### 【注意事项】

无

## VDEC\_CHN\_ATTR\_S

#### 【说明】

定义解码通道属性

#### 【定义】

```
typedef struct rkVDEC_CHN_ATTR_S {
    PAYLOAD_TYPE_E enType;
    VIDEO_MODE_E enMode;
    RK_U32 u32Picwidth;
    RK_U32 u32PicHeight;
    RK_U32 u32PicVirwidth;
    RK_U32 u32PicVirHeight;
    RK_U32 u32StreamBufSize;
    RK_U32 u32FrameBufSize;
    RK_U32 u32FrameBufCnt;
    RK_U32 u32StreamBufCnt;
    union {
        VDEC_ATTR_VIDEO_S stVdecVideoAttr;
    };
} VDEC_CHN_ATTR_S;
```

### 【成员】

成员名称	描述	属性
enType	解码协议类型枚举值。	静态属性。
enMode	码流发送方式。	静态属性。
u32PicWidth	视频的宽度（以像素为单位）。	静态属性。
u32PicHeight	视频的高度（以像素为单位）。	静态属性。
u32PicVirWidth	未使用。	静态属性。
u32PicVirHeight	未使用。	静态属性。
u32StreamBufSize	未使用。	静态属性。
u32FrameBufSize	未使用。	静态属性。
u32FrameBufCnt	解码图像帧存个数。	静态属性。
u32StreamBufCnt	码流包存储个数。	静态属性。

### 【注意事项】

## VDEC\_CHN\_PARAM\_S

### 【说明】

定义通道参数结构体。

### 【定义】

```
typedef struct rkVDEC_PARAM_VIDEO_S {
    RK_S32 s32ErrThreshold;
    VIDEO_DEC_MODE_E enDecMode;
    VIDEO_OUTPUT_ORDER_E enOutputOrder;
    COMPRESS_MODE_E enCompressMode;
    VIDEO_FORMAT_E enVideoFormat;
} VDEC_PARAM_VIDEO_S;

typedef struct rkVDEC_CHN_PARAM_S {
    PAYLOAD_TYPE_E enType;
    RK_U32 u32DisplayFrameNum;
    union {
        VDEC_PARAM_VIDEO_S stVdecVideoParam;
        VDEC_PARAM_PICTURE_S stVdecPictureParam;
    };
} VDEC_CHN_PARAM_S;
```

### 【成员】

成员名称	描述	属性
enCompressMode	图像输出方式，压缩或者非压缩	静态属性。

### 【注意事项】

- 如果需要获取可以正常显示的解码数据，需要设置图像输出为非压缩：COMPRESS\_MODE\_NONE

## VDEC\_STREAM\_S

### 【说明】

定义视频解码的码流结构体

### 【定义】

```
typedef struct rkVDEC_STREAM_S {
    MB_BLK  pMbBlk;
    RK_U32  u32Len;
    RK_U64  u64PTS;
    RK_BOOL bEndOfStream;
    RK_BOOL bEndOfFrame;
    RK_BOOL bBypassMbBlk;
} VDEC_STREAM_S;
```

### 【成员】

成员名称	描述
pMbBlk	码流包的地址。
u32Len	码流包的长度。以 byte 为单位。
u64PTS	码流包的时间戳。以 $\mu\text{s}$ 为单位。
bEndOfStream	是否发完所有码流。
bEndOfFrame	未使用。
bBypassMbBlk	pMbBlk 是否需要拷贝

### 【注意事项】

- 用户分配的buffer需要统一封装成MB\_BLK类型，可通过RK\_MPI\_SYS\_CreateMB来封装
- 拷贝模式，即设置bBypassMbBlk为RK\_FALSE。在该模式下，用户data送入解码器后，解码器内部会做数据拷贝，用户data内存空间交由用户控制，用户可以循环复用此内存空间。
- 零拷贝模式，即设置bBypassMbBlk为RK\_TRUE，并设置释放回调函数user\_data\_callback，推荐使用此模式。在该模式下，用户data内存空间由解码器来释放。

```
pstMbExtConfig.pFreeCB = user_data_callback;
pstMbExtConfig.pOpaque = userdata;
pstMbExtConfig.pu8VirAddr = userdata;
pstMbExtConfig.u64Size = usersize;

RK_MPI_SYS_CreateMB(&buffer, &pstMbExtConfig);

pstStream.u64PTS = userpts;
pstStream.pMbBlk = buffer;
pstStream.u32Len = usersize;
pstStream.bEndOfStream = RK_FALSE;
pstStream.bBypassMbBlk = RK_TRUE;
```

## VIDEO\_FRAME\_INFO\_S

### 【说明】

定义视频图像帧信息结构体。

### 【定义】

```
typedef struct rkVIDEO_FRAME_S {
    MB_BLK          pMbBlk;
    RK_U32           u32Width;
    RK_U32           u32Height;
    VIDEO_FIELD_E    enField;
    PIXEL_FORMAT_E   enPixelFormat;
    VIDEO_FORMAT_E   enVideoFormat;
    COMPRESS_MODE_E  enCompressMode;
    DYNAMIC_RANGE_E  enDynamicRange;
    COLOR_GAMUT_E    enColorGamut;

    RK_S16           s16OffsetTop;
    RK_S16           s16OffsetBottom;
    RK_S16           s16OffsetLeft;
    RK_S16           s16OffsetRight;

    RK_U32           u32MaxLuminance;
    RK_U32           u32MinLuminance;

    RK_U32           u32TimeRef;
    RK_U64           u64PTS;

    RK_U64           u64PrivateData;
    RK_U32           u32FrameFlag;
    RK_BOOL          bBypassMbBlk;
} VIDEO_FRAME_S;

typedef struct rkVIDEO_FRAME_INFO_S {
    VIDEO_FRAME_S  stVFrame;
} VIDEO_FRAME_INFO_S;
```

### 【成员】

成员名称	描述
pMbBlk	用户数据
u32Width	图像宽度。
u32Height	图像高度。
enVideoFormat	目标图像视频格式。
enPixelFormat	目标图像像素格式。
enDynamicRange	目标图像动态范围。
enCompressMode	目标图像压缩模式。
u64PTS	图像时间戳。

### 【注意事项】

- 用户获取图像数据，可以通过RK\_MPI\_MB\_Handle2VirAddr转换pMbBlk成虚拟地址来使用

```
data = RK_MPI_MB_Handle2VirAddr(sFrame.stVFrame.pMbBlk);
fwrite(data, 1, sFrame.stVFrame.u32Width * sFrame.stVFrame.u32Height * 3 / 2,
fp);
fflush(fp);
```

```
data = RK_MPI_MB_Handle2VirAddr(sFrame.stVFrame.pMbBlk);
```

VDEC\_CHN\_STATUS\_S

【说明】  
定义通道状态  
【定义】

```
typedef struct rkVDEC_CHN_STATUS_S {
    PAYLOAD_TYPE_E enType;
    RK_U32 u32LeftStreamBytes;
    RK_U32 u32LeftPics;
    RK_BOOL bStartRecvStream;
    VDEC_DECODE_ERROR_S stVdecDecErr;
} VDEC_CHN_STATUS_S;
```

【成员】

成员名称	描述
enType	解码协议类型枚举值。
u32LeftStreamBytes	码流 buffer 中待解码的 byte 数，包括正在解码的当前帧中未解码的 byte 数。
u32LeftPics	图像 buffer 中剩余的 pic 数目。
bStartRecvStream	解码器是否已经启动接收码流。
stVdecDecErr	未使用。

【注意事项】  
无

错误码

视频解码 API 错误码如下所示：



错误代码	宏定义	描述
0xA0058002	RK_ERR_VDEC_INVALID_CHNID	VDEC 通道号无效
0xA0058003	RK_ERR_VDEC_ILLEGAL_PARAM	VDEC 参数设置无效
0xA0058004	RK_ERR_VDEC_EXIST	VDEC 通道已创建
0xA0058005	RK_ERR_VDEC_UNEXIST	VDEC 通道未创建
0xA0058006	RK_ERR_VDEC_NULL_PTR	输入参数空指针错误
0xA0058008	RK_ERR_VDEC_NOT_SUPPORT	操作不支持
0xA0058009	RK_ERR_VDEC_NOT_PERM	操作不允许
0xA005800C	RK_ERR_VDEC_NOMEM	分配内存失败
0xA005800D	RK_ERR_VDEC_NOBUF	分配 BUF 池失败
0xA005800E	RK_ERR_VDEC_BUF_EMPTY	图像队列为空
0xA005800F	RK_ERR_VDEC_BUF_FULL	图像队列满状态
0xA0058010	RK_ERR_VDEC_NOTREADY	VDEC 系统未初始化
0xA0058012	RK_ERR_VDEC_BUSY	VDEC 系统忙
0xA0058013	RK_ERR_VDEC_BADADDR	错误的地址