HiMPP 媒体处理软件

开发参考

文档版本 25

发布日期 2015-04-16

版权所有 © 深圳市海思半导体有限公司 2011-2015。保留一切权利。

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。

商标声明

、HISILICON、海思和其他海思商标均为深圳市海思半导体有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受海思公司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。除非合同另有约定，海思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。

深圳市海思半导体有限公司

地址：深圳市龙岗区坂田华为基地华为电气生产中心 邮编：518129

网址：http://www.hisilicon.com

客户服务电话：+86-755-28788858

客户服务传真：+86-755-28357515

客户服务邮箱：support@hisilicon.com

目 录

前 言.....1

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

前 言

概述

本文为使用 HiMPP 媒体处理芯片进行开发的程序员而写，目的是供您在开发过程中查阅媒体处理软件开发包的各种参考信息，包括 API、头文件、错误码等。

本文档描述 HiMPP 媒体处理软件的各个 API 的使用方法，以及相关的数据结构和错误码。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
Hi3531 芯片	V100
Hi3532 芯片	V100
Hi3521 芯片	V100
Hi3520A 芯片	V100
Hi3518 芯片	V100
Hi3516C 芯片	V100
Hi3520D 芯片	V100
Hi3515A 芯片	V100
Hi3515C 芯片	V100

读者对象

本文档主要适用于以下工程师：

- 技术支持工程师

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 软件开发工程师

约定

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
危险	表示有高度潜在危险，如果不能避免，会导致人员死亡或严重伤害。
警告	表示有中度或低度潜在危险，如果不能避免，可能导致人员轻微或中等伤害。
注意	表示有潜在风险，如果忽视这些文本，可能导致设备损坏、数据丢失、设备性能降低或不可预知的结果。
窍门	表示能帮助您解决某个问题或节省您的时间。
说明	表示是正文的附加信息，是对正文的强调和补充。

通用格式约定

格式	说明
宋体	正文采用宋体表示。
黑体	一级、二级、三级标题采用**黑体**。
楷体	警告、提示等内容一律用**楷体**，并且在内容前后增加线条与正文隔离。
“Terminal Display”格式	“Terminal Display”格式表示屏幕输出信息。此外，屏幕输出信息中夹杂的用户从终端输入的信息采用加粗字体表示。
“”	用双引号表示文件路径。如“C:\Program Files\Huawei”。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

命令行格式约定

格式	意义
粗体	命令行关键字（命令中保持不变、必须照输的部分）采用**加粗**字体表示。
斜体	命令行参数（命令中必须由实际值进行替代的部分）采用*斜体*表示。
[ ]	表示用“[ ]”括起来的部分在命令配置时是可选的。
{ x | y | ... }	表示从两个或多个选项中选取一个。
[x | y | ...]	表示从两个或多个选项中选取一个或者不选。
{ x | y | ... } *	表示从两个或多个选项中选取多个，最少选取一个，最多选取所有选项。
[x | y | ...] *	表示从两个或多个选项中选取多个或者不选。

数值单位约定

数据容量、频率、数据速率等的表达方式说明如下。

类别	符号	对应的数值
数据容量（如 RAM 容量）	1K	1024
	1M	1,048,576
	1G	1,073,741,824
频率、数据速率等	1k	1000
	1M	1,000,000
	1G	1,000,000,000

地址、数据的表达方式说明如下。

符号	举例	说明
0x	0xFE04、0x18	用 16 进制表示的数据值、地址值。
0b	0b000、0b00 00000000	表示 2 进制的数据值以及 2 进制序列（寄存器描述中除外）。
X	00X、1XX	在数据的表达方式中，X 表示 0 或 1。 例如：00X 表示 000 或 001； 1XX 表示 100、101、110 或 111。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订日期	版本	修订说明
2015-04-16	25	**第 4 章 视频输出** 4.3 新增 HI_MPI_VO_SetChnDisplayWindow 和 HI_MPI_VO_GetChnDisplayWindow **第 6 章 视频编码** 新增 6.2.13 小节的内容 6.4 节，修改 VENC_ATTR_H264_S、VENC_ATTR_MJPEG_S、VENC_ATTR_JPEG_S 和 VENC_ATTR_MPEG4_S 中的【成员】描述。 **第 13 章 Proc 调试信息** 修改 13.4、13.7、13.8、13.10、13.22 和 13.23 相关内容。
2015-03-30	24	**第 8 章 区域管理** 修改 COVER_CHN_ATTR_S 的【成员】stRect 的取值范围。 **第 9 章 音频** 9.2.1.5 小节中有修改； 9.3.1 新增 HI_MPI_AI_SetHpfAttr 和 HI_MPI_AI_GetHpfAttr； 9.3.5.1 新增 ACODEC_SET_INPUT_VOL 到 ACODEC_GET_OUTPUT_VOL 9.4.1 新增 AUDIO_HPF_FREQ_E、AI_HPF_CONFIG_S 和 AI_HPF_ATTR_S **第 10 章 视频解码** 10.2 小节有修改 10.3 HI_MPI_VDEC_SetPrtclParam 中的注意有更改 10.4 VDEC_CHN_ATTR_S 和 VDEC_PRTCL_PARAM_S 的成员描述有修改
2015-02-10	23	**第 3 章 视频输入** 3.4 新增 HI_MPI_VI_SetChnScanMode 和 HI_MPI_VI_GetChnScanMode **第 5 章 视频处理子系统** 5.3 新增 HI_MPI_VPSS_EnableBackupFrame 和 HI_MPI_VPSS_DisableBackupFrame， HI_MPI_VPSS_SetCropCfg 的【注意】有改动 **第 6 章 视频编码**

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		新增 6.2.11 和 6.2.12 小节。6.3 新增 HI_MPI_VENC_AttachVbPool、HI_MPI_VENC_DetachVbPool、HI_MPI_VENC_SetH264VideoSignal 和 HI_MPI_VENC_GetH264VideoSignal 6.4 新增 VENC_PARAM_H264_VIDEO_SIGNAL_S **第 9 章 音频** 9.3.5.1 更改 ACODEC_ENABLE_BOOSTL 并新增 ACODEC_ENABLE_BOOSTR **第 12 章 HDMI** 12.4 新增 HI_HDMI_SPD_INFOFRAME_S，HI_HDMI_MPEGSOURCE_INFOFRAME_S 和 HI_HDMI_VENDORSPEC_INFOFRAME_S **第 13 章 Proc 调试信息** 13.8，13.9，13.12 和 13.23 节，修改【调试信息】和【参数说明】
2014-09-21	22	**第 4 章 视频输出** 4.4 添加 VO_PUB_ATTR_S，VO_PUB_ATTR_S 中的【芯片差异】和 VO_VIDEO_LAYER_ATTR_S 中的【差异说明】有修改 **第 5 章 视频处理子系统** 5.3 HI_MPI_VPSS_SetGrpParam 中的【注意】有修改 5.4 VPSS_GRP_ATTR_S 中的【成员】有修改 **第 9 章 音频** 9.2.2.3 小节有修改 9.4.3 添加 AUDIO_FRAME_INFO_S **第 12 章 HDMI** 12.4 HI_HDMI_VIDEO_FMT_E 中有修改
2014-09-02	21	**第 3 章 视频输入** 3.4 API 参考，添加 HI_MPI_VI_SetPortMode，HI_MPI_VI_GetPortMode，HI_MPI_VI_EnableChn422to420 和 HI_MPI_VI_DisableChn422to420 3.5 数据类型中添加 VI_PORTMODE_E **第 9 章 音频** 9.3.4 音频解码，添加 HI_MPI_ADEC_GetFrame 和 HI_MPI_ADEC_ReleaseFrame
2014-08-15	20	**第 2 章系统控制** 2.3 API 参考中增加 HI_MPI_VB_ GetVRegAddr **第 3 章视频输入** 3.4 API 参考中 HI_MPI_VI_SetDevAttr 新增 VI 模块参数；修改

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		HI_MPI_VI_SetCSCAttr 的注意（新增“关于 CSC 有以下模块参数”） **第 5 章视频处理子系统** 5.3 API 参考，增加 HI_MPI_VPSS_CreateGrp 和 HI_MPI_VPSS_SetChnMode 中【注意】的相关内容。 **第 6 章视频编码** 添加表 6-1 中 SVC-T 的相关内容。 6.3 API 参考中增加 HI_MPI_VENC_SetRcPriority、HI_MPI_VENC_GetRcPriority、HI_MPI_VENC_SetLostFrameStrategy、HI_MPI_VENC_GetLostFrameStrategy。 修改 HI_MPI_VENC_CreateChn 的芯片差异；修改 HI_MPI_VENC_SetChnAttr 的注意；修改 HI_MPI_VENC_SetRcPara 和 HI_MPI_VENC_GetH264InterPred 的注意 6.4 数据类型中增加 H264E_REF_TYPE_E、VENC_ATTR_H264_ABR_S、VENC_RC_PRIORITY_E、VENC_PARAM_LOSTFRM_S 和 VENC_LOSTFRM_MODE_E **第 9 章 音频** 9.2.1.4 修改回声抵消的示意图及注意，新增 9.2.1.5 章节。 9.2.4.2 新增“如果模拟增益达不到要求，可开启 BOOST 功能。数字增益默认已调为最优” 9.3.1 新增 HI_MPI_AI_SetVqeAttr、HI_MPI_AI_GetVqeAttr、HI_MPI_AI_EnableVqe、HI_MPI_AI_DisableVqe、HI_MPI_AI_EnableReSmpEx、HI_MPI_AI_DisableReSmpEx、HI_MPI_AI_ClrPubAttr、HI_MPI_AI_EnableReSmpEx、HI_MPI_AI_DisableReSmpEx、HI_MPI_AI_Save_File；修改 HI_MPI_AI_GetFrame、HI_MPI_AI_SetChnParam、HI_MPI_AI_GetChnParam 的注意 9.3.2 新增 HI_MPI_AO_EnableReSmpEx、HI_MPI_AO_DisableReSmpEx、HI_MPI_AO_ ClrPubAttr。 修改 HI_MPI_AO_SendFrame 的注意。 9.3.3 新增 HI_MPI_AENC_Save_File；修改 HI_MPI_AENC_SendFrame 的注意 9.3.5 新增 ACODEC_ENABLE_BOOST 9.4.1 新增 AUDIO_RESAMPLE_ATTR_EX_S、AUDIO_AEC_MODE_E、AI_AEC_CONFIG_S、AI_ANR_CONFIG_S、AI_ALC_CONFIG_S、AI_VQE_CONFIG_S。 **第 10 章 视频解码** 10.2 增加通道解码能力。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		10.3 API 参考中修改 HI_MPI_VDEC_CreateChn 的注意及相关主题；修改 HI_MPI_VDEC_SetChnParam、HI_MPI_VDEC_GetChnParam、HI_MPI_VDEC_SetPrtclParam、HI_MPI_VDEC_GetPrtclParam、 HI_MPI_VDEC_SetDisplayFrameNum 的注意 10.4 数据类型中修改 VDEC_ATTR_VIDEO_S 的成员名称 u32RefFrameNum 描述及注意事项；修改 VDEC_PRTCL_PARAM_S 的成员名称 s32DisplayFrameNum 描述。 **第 13 章 Proc 调试信息** 13.10 RC 修改调试信息及参数说明. 13.19 AI 修改调试信息及参数说明 13.20 AO 修改调试信息及参数说明 13.21 AENC 修改调试信息及参数说明 新增 13.25 小节内容。
2014-05-09	19	**第 4 章 视频输出** 新增 HI_MPI_VO_EnableRecvFrameRateMatch 和 HI_MPI_VO_DisableRecvFrameRateMatch。 **第 6 章 视频编码** 新增 6.2.10 编码码流 buffer 配置模式。 6.3 API 参考中新增 HI_MPI_VENC_GetStreamBufInfo。 **第 9 章 音频** 9.2.1.4 回声抵消中增加当 AI-AENC 为不同绑定方式时的采样率。 **第 10 章 视频解码** 10.3 API 参考中，增加 HI_ MPI_VDEC_SetDisplayFrameNum、HI_MPI_VDEC_GetDisplayFrameNum、 HI_MPI_VDEC_SetChnPrevMode 和 HI_MPI_VDEC_GetChnPrevMode。 10.4 数据类型中，VDEC_PRTCL_PARAM_S 的【成员】s32DisplayFrameNum 的描述中删除关于“对 JPEG 通道默认为 4”的描述，修改取值范围的最小值为 0；增加【注意事项】。 **第 13 章 Proc 调试信息** 13.13 VDEC 中增加参数 PREVMODE。 13.15 VO 中增加参数 bRcvMatch。
2014-02-26	18	**第 2 章 系统控制** HI_MPI_SYS_SetMemConf 的【注意】中补充关于此接口调用的描述。 **第 3 章 视频输入**

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		LDC_ATTR_S 增加【注意事项】。 **第 4 章 视频输出** **新增** HI_MPI_VO_EnableSdTdeBypass 和 HI_MPI_VO_DisableSdTdeBypass。 **第 5 章 视频子系统** 新增 HI_MPI_VPSS_SetGrpField 和 HI_MPI_VPSS_GetGrpField。 **第 6 章 视频编码** 新增 HI_MPI_VENC_RequestIDRInst。 **第 9 章 音频** 9.2.1.4 回声抵消、重采样和噪声抑制中增加关于“回声抵消”的注意事项。 修改表 9-3 中关于采样率的描述。 **第 13 章 Proc 调试信息** 更新 H264E、JPEGE 和 VENC。
2014-01-20	17	**补充 Hi3518E 的相关描述。**
2013-11-30	16	**第 4 章 视频输出** 新增 HI_MPI_VO_SetDevFramerate 和 HI_MPI_VO_GetDevFramerate。 VO_PUB_ATTR_S 的【注意事项】中删除“用户可以自定义时序”的描述。 **第 13 章 Proc 调试信息** 更新 VO 和 VPSS。
2013-09-25	15	**第 2 章 系统控制** 增加“MPP 不支持多进程调用”的描述。 表 2-1 中，修改 VDEC 和 VO 的相关描述。 HI_MPI_SYS_Init 的【注意】中删除“只要有一个进程进行初始化即可”的描述。 **第 3 章 视频输入** VI_CHN_STAT_S 增加【成员】u32AutoDisInt 及描述。 **第 4 章 视频输出** 新增 HI_MPI_VO_EnableChnDoubleFrame 和 HI_MPI_VO_DisableChnDoubleFrame。 新增 HI_MPI_VO_SetVtth 和 HI_MPI_VO_GetVtth。 VO_PUB_ATTR_S 的【成员】enIntfSync 新增 VO_OUTPUT_1600x1200_60、VO_OUTPUT_1680x1050_60、VO_OUTPUT_1920x1200_60、VO_OUTPUT_640x480_60；【注意事项】中修改 enIntfSync 的最大值为 22。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		第 5 章 视频子系统 5.2.3 芯片差异中增加 chn1 及修改 chn0 的部分描述。 更新图 5-3。 **第 6 章 视频编码** 更新 HI_MPI_VENC_CreateGroup、HI_MPI_VENC_StartRecvPicEx 的【举例】。 HI_MPI_VENC_GetStream 的【注意】中增加码流特征信息 stMpeg4Info。 HI_MPI_VENC_SetH264eRefParam、HI_MPI_VENC_GetH264eRefParam 的【语法】更新。 HI_MPI_VENC_GetJpegSnapMode 的【注意】中修改“通道组”为“通道”。 新增 VENC_PARAM_H264_CBRV2_S。 VENC_STREAM_INFO_H264_S、VENC_STREAM_INFO_JPEG_S、VENC_STREAM_INFO_MPEG4_S、VENC_RC_ATTR_S、VENC_RC_MODE_E、VENC_PARAM_H264_CBR_S、VENC_PARAM_H264_VBR_S、VENC_PARAM_MJPEG_CBR_S、VENC_PARAM_MJPEG_VBR_S、VENC_PARAM_MPEG4_CBR_S、VENC_PARAM_MPEG4_VBR_S、VENC_RC_PARAM_S 的【语法】更新。 **第 10 章 视频解码** 增加 VDEC_DECODE_ERROR_S。 更新 VDEC_CHN_STAT_S【定义】。 **第 12 章 HDMI** 新增 HI_MPI_HDMI_SetInfoFrame、HI_MPI_HDMI_GetInfoFrame。 新增 HI_HDMI_INFOFRAME_TYPE_E、HI_HDMI_AVI_INFOFRAME_VER2_S、HI_HDMI_AUD_INFOFRAME_VER1_S、HI_HMDI_INFORFRAME_UNIT_U 和 HI_HDMI_INFOFRAME_S。 **第 13 章 Proc 调试信息** 更新 VDEC 和 VO。
2013-07-31	14	**第 9 章 音频** 9.3.2 音频输出中新增 HI_MPI_AO_SetVolume 和 HI_MPI_AO_GetVolume。
2013-07-03	13	**第 13 章 Proc 调试信息**

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		更新 Log。
2013-06-21	12	**补充 Hi3515C 的相关信息描述。**
2013-05-20	11	**补充 Hi3515A 的相关信息描述。** **第 3 章 视频输入** 3.3 功能描述中，增加 Hi3518/Hi3516C 的视频输入设备 Dev0 不支持非连续 pixel clock 时序的描述。 **第 4 章 视频输出** 表 4-1 中增加 Hi3520D/Hi3515A 对支持的显示设备的描述。 新增 HI_MPI_VO_SetChnDispThreshold 和 HI_MPI_VO_GetChnDispThreshold。 HI_MPI_VO_SetChnFrameRate 的【注意】中新增两个事项 HI_MPI_VO_SetChnField 的【描述】中增加一个应用场合；更新【举例】。 HI_MPI_VO_SetZoomInWindow 的【注意】中增加注意事项。 VO_ZOOM_ATTR_S 的【注意事项】中修改按位置和按比例两种类型的描述。 HI_MPI_VO_SetDevCSC、HI_MPI_VO_SetVgaParam 的【差异说明】的描述有更新。 修改 HI_MPI_VO_SetDispBufLen 的【参数】u32BufLen 的最小值为 3。 VO_CSC_S 和 VO_VGA_PARAM_S 的【成员】中增加各成员的取值范围。 增加三个错误码。 **第 8 章 区域管理** 增加 HI_MPI_RGN_SetAttachField 和 HI_MPI_RGN_GetAttachField。 增加 RGN_ATTACH_FIELD_E。 修改 COVER_CHN_ATTR_S 和 COVEREX_CHN_ATTR_S 的【成员】stRect 的取值范围。 **第 9 章 音频** ACODEC_SOFT_RESET_CTRL 的【注意】中增加一条事项。 ACODEC_SET_MIXER_MIC 的【注意】中增加“选择 LINEIN”的相关描述。 ACODEC_SET_GAIN_MICL 和 ACODEC_SET_GAIN_MICR 的【注意】中增加一条事项；修改【举例】中的配置值。 **第 12 章 HDMI** 新增 HI_MPI_HDMI_SetCsc、HI_MPI_HDMI_GetCsc 和 HI_HDMI_CSC_S。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		第 13 章 Proc 调试信息 更新 REGION、VI、VO。
2013-04-03	10	**第 3 章 视频输入** HI_MPI_VI_SetUserPic 的【注意】中增加“VI 通道混合采集时，如何设置图片”。 **第 4 章 视频输出** HI_MPI_VO_SetChnAttr 的【注意】中增加“HD 设备的通道属性设置”的相关描述。 HI_MPI_VO_SetChnDispPos 的【注意】中增加“设置显示位置后，通道 disable 后”的相关描述。 HI_MPI_VO_SetDevCSC 的【注意】中增加“建议选择的 CSC 矩阵”。 HI_MPI_VO_GfxLayerBindDev 的【注意】中增加“Hi3520D 中，建议用户对鼠标层的 CSC 模式设置”。 HI_MPI_VO_SetGfxLayerCSC 的【注意】中删除“CSC 转换矩阵恢复”的描述。 **第 5 章 视频子系统** 表 5-2 中，修改通道 0 为“没有缩放能力”。 VPSS_GRP_PARAM_S 的【成员】u32MotionThresh 的描述中，补充 Hi3520D 的描述；修改【成员】u32ChromaRange 的取值范围。 **第 6 章 视频编码** HI_MPI_VENC_CreateChn 的【注意】中更新设置 ViFrmRate 的相关描述。 **第 13 章 Proc 调试信息** 更新 VDEC。
2013-03-31	09	**第 5 章 视频子系统** 更新图 5-4。 **第 6 章 视频编码** 增加 6.2.5 高级跳帧参考模式。 新增 HI_MPI_VENC_SetH264eRefParam 和 HI_MPI_VENC_GetH264eRefParam。 新增 VENC_ATTR_H264_REF_PARAM_S。 **第 10 章 视频解码** 增加错误码 HI_ERR_VDEC_BADADDR。 **第 13 章 Proc 调试信息** 更新 CHNL、DSU、GROUP、H264E、JPEGE、RC、REGION、VI、VO、VPSS、AI 和 AO。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
2013-02-05	08	**第 3 章 视频输入** VI_DEV_ATTR_S 和 VI_DEV_ATTR_EX_S 的【定义】和【成员】增加 bDataRev。 **第 4 章 视频输出** HI_MPI_VO_SetVgaParam、HI_MPI_VO_GetVgaParam 的【差异说明】中增加 Hi3518/Hi3516C 暂不支持 VoDev 的说明。 VO_WBC_ATTR_S【定义】和【成员】增加 enDataSource **第 6 章 视频编码** 新增 HI_MPI_VENC_StartRecvPicEx。 HI_MPI_VENC_Query 的【注意】中增加编码通道状态结构体中的 u32LeftRecvPics 和 u32LeftEncPics 的描述。 VENC_CHN_STAT_S【定义】和【成员】中增加 u32LeftRecvPics 和 u32LeftEncPics。 VENC_ATTR_MJPEG_VBR_S 的【成员】u32StatTime 最大值由 10 改为 16。 **第 9 章 音频** 新增 9.2.4 内置 Audio Codec；9.3.5 内置 Audio Codec；9.4.4 内置 Audio Codec。 **第 13 章 Proc 调试信息** 更新 AI。
2013-01-15	07	**第 8 章 区域管理** OVERLAYEX_ATTR_S 的[成员] stSize 的宽度和高度最大值有修改，且要求宽度 x 高度小于等于 480000。 OVERLAYEX_CHN_ATTR_S 的【成员】stPoint 的垂直位置 Y 的最大值有修改。 **第 9 章 音频** 新增 HI_MPI_AO_QueryChnStat 和 AO_CHN_STATE_S **第 13 章 Proc 调试信息** 13.5 CHNL 中更新调试信息，增加参数 IntPro 和 IntCostL。 13.7 GROUP 中更新调试信息。 13.14 VI 中更新调试信息，增加参数 detece_err_frame、MaxIntT、OverCnt、LintCnt 和 ThrCnt。 13.19 AI 中更新调试信息，增加参数 MaxFrmTime、MaxIsrTime、bAnr、bResmp、PoiNum、SampR 和 ResmpType。 13.20 AO 中更新调试信息，增加参数 FrmTime、MaxFrmTime、IsrTime、MaxIsrTime、bResmp、PoiNum、SampR 和 ResmpType。
2012-12-26	06	**增加 Hi3516C 的信息。** **第 3 章 视频输入**

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		3.3.3 Hi3518/Hi3516C 的掩码配置中增加掩码配置与硬件管脚的描述。 HI_MPI_VI_SetChnAttr 的【芯片差异】中增加通道属性参数 stCapRect 中 s32Y 和 u32Height 的使用限制；【注意】中删除 s32Y 和 u32Height“隔行且采集两场时必须 4 对齐”。 VI_CHN_ATTR_S 的【芯片差异】修改 Hi3518/Hi3516C 的 stDestSize 的描述。 **第 4 章 视频输出** HI_MPI_VO_DisableChn 的【注意】中增加“当高清设备的通道绑定 VPSS 时”的描述。 HI_MPI_VO_SetChnAttr 的【注意】增加“当高清设备是隔行时序且 SPYCbCr420 显示时，高度必须 4 对齐”的描述。 VO_CHN_ATTR_S 的【成员】stRect 的描述中增加“当高清设备是隔行时序且 SPYCbCr420 显示时，高度必须 4 对齐”的描述；【注意事项】中删除“SD 设备的通道属性 stRect 需 2 对齐”。 **第 5 章 视频处理子系统** VPSS_CROP_INFO_S 的【定义】中增加参数 enCapSel 及描述。 VPSS_GRP_ATTR_S 的【成员】中修改 u32MaxH（Hi3521/Hi3520A/Hi3531/Hi3532）的最大值为 4080。 **第 6 章 视频编码** HI_MPI_VENC_SetJpegParam、HI_MPI_VENC_SetMjpegParam 的【注意】中修改高级参数 u32MCUPerECS 的最大值。 VENC_ATTR_H264_CBR_S 的【成员】u32BitRate 的最小值改为 2；更新【注意事项】中关于 ViFrmRate 的描述。 VENC_ATTR_MPEG4_CBR_S、VENC_ATTR_MJPEG_CBR_S、的【成员】u32BitRate 的最小值改为 2。 VENC_ATTR_H264_VBR_S、VENC_ATTR_MPEG4_VBR_S 和 VENC_ATTR_MJPEG_VBR_S 的【成员】u32MaxBitRate 的最小值改为 2。 **第 9 章 音频** HI_MPI_AI_EnableAec、HI_MPI_AI_EnableAnr 的【注意】中新增“禁用 AI 通道后，如果重新启用”时的说明。 新增 HI_MPI_AENC_RegeisterEncoder、HI_MPI_AENC_UnRegisterEncoder、HI_MPI_DENC_RegeisterEncoder、HI_MPI_DENC_UnRegisterEncoder、AENC_ENCODER_S、ADNC_DECODER_S。 HI_MPI_AENC_CreateChn 的【注意】中修改 buffer 的取值范围的最小值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		HI_MPI_ADEC_CreateChn【举例】更新。 删除 HI_MPI_DENC_GetFrame、HI_MPI_DENC_ReleaseFrame 和 AUDIO_FRAME_INFO_S。 AENC_CHN_ATTR_S 的【定义】和【成员】中 buffer size 的最小值改为 2。 ADEC_CHN_ATTR_SD【成员】u32BufSize 的最小值改为 2。 表 9-3 中增加“在音频码流网络传输应用中，不推荐使用 ADPCM_ORG_DVI4”的相关描述。 HI_MPI_AENC_RegeisterEncoder 的【注意】中修改“同一种编码协议不允许重复注册编码器”的举例。 **第 11 章 智能加速引擎** 11.2.1 重要概念中**跨度**增加“Hist 的输出 stride 固定为 256”的描述。 **第 13 章 Proc 调试信息** 13.24 HDMI 中参数 EDID Parse Status 的描述有更新。
2012-11-30	05	**第 5 章 视频处理子系统** 5.2.1 基本概念中增加"备份节点"的描述。 HI_MPI_VPSS_UserGetGrpFrame 的【注意】中增加关于“备份节点”的描述。 **第 13 章 Proc 调试信息** 更新 13.3 VB 的调试信息，增加参数 MinFree 的描述。 更新 13.8 H264E 的 Syntax INFO 的调试信息，修改 slcmode 的描述。 更新 13.10 RC 的调试信息。 增加 HDMI 的调试信息。
2012-11-25	04	**第 2 章 系统控制** 表 2-1 中，数据源 VPSS 和 VDEC 增加 VDA 为数据接收者。 **第 3 章 视频输入** 表 3-1 中 0/2/4/6 端口的最大图像输入能力改为 2560x1600。 HI_MPI_VI_SetFlashConfig 的【注意】中删除“该接口需要在设备使能后设置”的描述。 HI_MPI_VI_SetExtChnAttr 的【注意】中修改扩展通道支持的缩小倍数为小于 13；同时补充扩展通道的输出宽高要求为小于 4096。 HI_MPI_VI_SetLDCAttr 的【注意】中修改“该接口必须在设置通道属性后调用”。 新增 VI_DATA_TYPE_E。 VI_EXT_CHN_ATTR_S 的【注意事项】中有修改。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		第 4 章 视频输出 HI_MPI_VO_SetPubAttr 的【差异说明】中增加“CVBS 输出时支持自动负载检测”的描述；【注意】中增加“自动负载检测”的描述。 HI_MPI_VO_SetChnAttr 的【注意】中接口使用情况在“VO 设备使能”的基础上增加“且相应视频层已配置”的描述。 HI_MPI_VO_SetChnDispPos 的【注意】中修改 PIP 层节省内存时，将 PIP 层的画布大小设置为“实际显示图像需要的内存大小”。 HI_MPI_VO_SetZoomInWindow 的【注意】中调用前的限制改为“视频层已经设置”。 HI_MPI_VO_GetZoomInWindow 的【注意】中删除调用前的使用限制。 HI_MPI_VO_SetAttrBegin、HI_MPI_VO_ReleaseChnFrame 的【注意】中删除“调用前保证视频设备使能”的描述。 HI_MPI_VO_SetAttrEnd、HI_MPI_VO_ChnShow、HI_MPI_VO_QueryChnStat、的【注意】中修改调用前的使用限制。 HI_MPI_VO_GetScreenFrame 的【注意】中修改为“调用前需保证视频层已经使能”的描述。 HI_MPI_VO_ReleaseScreenFrame 的【注意】中修改使用限制。 HI_MPI_VO_SetVgaParam、HI_MPI_VO_GetVgaParam 的【定义】中数据结构更新。 HI_MPI_VO_SetGfxLayerCSC、HI_MPI_VO_GetGfxLayerCSC 的【参数】u32Layer 的最大值有修改。 VO_VIDEO_LAYER_ATTR_S 的【差异说明】中将 Hi3532 的内容单独描述；【注意】中删除“stImageSize 如果小于 stDispRect”的相关描述。 HI_MPI_VPSS_CreateGrp 的【语法】中修改 API 接口名称。 **第 5 章 视频处理子系统** HI_MPI_VPSS_SetGrpParam 的【语法】中删除参数 vpsschn HI_MPI_VPSS_SetDepth 的【语法】和【参数】中有修改。 HI_MPI_VPSS_UserGetFrame 的【描述】更新。 新增 HI_MPI_VPSS_UserGetGrpFrame、HI_MPI_VPSS_UserReleaseGrpFrame。 VPSS_GRP_PARAM_S 的【成员】u32SfStrength（Hi3518）的取值范围和默认值有修改；u32ChromaRange（Hi3518）由保留改为色度幅度及相关描述。 **第 6 章 视频编码** HI_MPI_VENC_SetH264InterPred 的【注意】中修改实际搜索窗高度的公式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		HI_MPI_VENC_SetRcPara 的【注意】中 CBR 高级参数 u32MinIprop 的默认值由 0 改为 1。 VENC_ATTR_H264_S 的【成员】bVIField 增加“目前未使用”的描述，并增加 u32Profile 及描述。 VENC_ATTR_MJPEG_S、VENC_ATTR_JPEG_S 的【成员】bVIField 增加“建议用户取值”的描述。 VENC_PARAM_JPEG_S、VENC_PARAM_MJPEG_S 的【成员】u32MCUPerECS 的描述更新。 VENC_ATTR_H264_CBR_S、VENC_ATTR_H264_VBR_S、VENC_ATTR_H264_FIXQP_S、VENC_ATTR_MPEG4_CBR_S、VENC_ATTR_MPEG4_VBR_S、VENC_ATTR_MPEG4_FIXQP_S、的【成员】u32Gop 的取值范围中最大值有修改。 VENC_PARAM_H264_CBR_S 的【成员】u32MinIprop 和 u32MaxIprop 的取值范围更改最小值；u32MinQp 的最小值改为 0；u32SuperIFrmBitsThr 和 u32SuperPFrmBitsThr 的取值范围改为“大于等于 0”。 VENC_PARAM_H264_VBR_S 的【成员】u32SuperIFrmBitsThr 和 u32SuperPFrmBitsThr 的取值范围改为“大于等于 0”。 VENC_PARAM_MJPEG_CBR_S 的【成员】u32SuperFrmBitsThr 的取值范围改为“大于等于 0”。 VENC_PARAM_MJPEG_VBR_S 的【成员】u32SuperFrmBitsThr 的取值范围改为“大于等于 0”。 VENC_PARAM_MPEG4_CBR_S 的【成员】u32SuperIFrmBitsThr 和 u32SuperPFrmBitsThr 的取值范围改为“大于等于 0”。 VENC_PARAM_MPEG4_VBR_S 的【成员】u32SuperIFrmBitsThr 和 u32SuperPFrmBitsThr 的取值范围改为“大于等于 0”。 **第 7 章 视频侦测分析** VDA_MAX_HEIGHT 的【定义】中不区分项目进行定义。 **第 9 章 音频** 图 9-7 下增加“SIO 只支持 I^2S 主模式”。 HI_MPI_AI_EnableAec 的【注意】中改为“启用回声抵消前必须先启用相对应的 AI 通道”。 HI_MPI_AI_EnableReSmp 的【注意】中删除调用接口限制中的“绑定 AI 通道前”；增加“在禁用 AI 通道之后，如果重新启用 AI 通道，并使用重采样功能，需调用此接口重新启用重采样”。 HI_MPI_AI_EnableAnr 的【注意】中修改调用此接口的限制；增加“在 AI 通道禁用后，如果重新使能 AI 通道，并使用噪声

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		抑制功能，需调用此接口重新启用噪声抑制”。 HI_MPI_AI_DisableAnr 的【注意】中增加使用接口的限制。 HI_MPI_AO_EnableReSmp 的【注意】中增加“在 AO 通道禁用后，如果重新使能 AO 通道，并使用噪声抑制功能，需调用此接口重新启用噪声抑制”。 新增 AIO_MAX_CHN_NUM。 AUDIO_BIT_WIDTH_E 的【注意事项】中增加“Hi3518 只支持 16bit 位宽”。 AIO_MODE_E 的【注意事项】中增加 Hi3518A 和 Hi3518C 的相关描述。 表 9-7 中修改 HI_ERR_AI_NOT_ENABLED 的错误代码；删除 0xA015800B。 **第 10 章 视频解码** HI_MPI_VDEC_CreateChn 的【注意】中增加“解码 YUV444 格式的 JPEG 图片”时的相关描述。 **第 11 章 智能加速引擎** 11.2.1 重要概念中增加“跨度”的描述。 **第 12 章 HDMI** 新增 HI_MPI_HDMI_GetDeepColor。
2012-10-30	03	**第 3 章 视频输入** 修改设置和获取 VI 通道次属性的 API 接口 HI_MPI_VI_SetChnMinorAttr 和 HI_MPI_VI_GetChnMinorAttr。 **第 4 章 视频输出** 新增 HI_MPI_VO_SetChnField、HI_MPI_VO_GetChnField、HI_MPI_VO_SetGfxLayerCSC 和 HI_MPI_VO_GetGfxLayerCSC。 HI_MPI_VO_ChnRefresh 的【注意】中涉及 VPSS 的 API 由 HI_MPI_VPSS_SetClipCfg 改为 HI_MPI_VPSS_SetCropCfg。 HI_MPI_VO_GfxLayerBindDev 和 HI_MPI_VO_GfxLayerUnBindDev 的【参数】enGfxLayer 的取值范围改为[0, VOU_GRAPHICS_LAYER_BUTT)。 **第 5 章 视频处理子系统** 新增 HI_MPI_VPSS_SetDelay 和 HI_MPI_VPSS_GetDelay。 HI_MPI_VPSS_SetCropCfg 的【注意】中新增 CROP 区域的相关描述。 HI_MPI_VPSS_UserGetFrame 的【参数】pstVideoFrame 输入输出属性改为“输出”。 VPSS_GRP_ATTR_S 的【成员】按照不同的芯片进行描述。 VPSS_GRP_PARAM_S 的【成员】u32SfStrength 和 u32TfStrength

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		的取值范围有修改。 更新 VPSS_CAPSEL_E 的【定义】和【成员】。 **第 6 章 视频编码** 图 6-1 下的说明增加“Hi3518 支持 JPEG 闪光灯抓拍模式时，通道组接收图像”的相关描述。 增加 6.2.7 JPEG 抓拍模式。 HI_MPI_VENC_SetRcPara 的【注意】中码率控制器的高级参数的描述中新增 u32MaxQp&u32MinQp&u32MaxStartQp。 HI_MPI_VENC_SetJpegSnapMode 的【注意】中新增“Hi3518 支持 JPEG 闪光灯抓拍模式时，通道组接收图像”的相关描述。 VENC_ATTR_MPEG4_VBR_S 的【成员】u32MaxQp 的取值范围改为(u32MinQp, 31]。 VENC_PARAM_H264_CBR_S 的【定义】和【参数】中新增 u32MaxStartQp;【成员】u32MaxQp 的取值范围改为(u32MinQp, 51]。 VENC_PARAM_MPEG4_CBR_S 的【定义】和【参数】中新增 u32MaxStartQp；【成员】u32MinIprop、u32MaxIprop 和 u32MaxQp 的最小值分别改为(0, 100]、(u32MinIprop, 100]和(u32MinQp, 31]。 **第 9 章 音频** 图 9-7 下面新增关于“Hi3518 使用内部 audio codec”的相关描述。 **第 12 章 HDMI** 新增 HI_MPI_HDMI_SetDeepColor。
2012-09-20	02	**第 3 章 视频输入** HI_MPI_VI_SetExtChnAttr 的【注意】中补充扩展通道支持的缩小倍数的描述。 HI_MPI_VI_SetCSCAttr 增加“接口要在设备属性设置后再调用”的描述。 **第 5 章 视频处理子系统** 修改图 5-2、图 5-3、图 5-4；修改表 5-1 中通道 1、通道 2、通道 3 的缩放能力描述；修改表 5-2 中通道 1、通道 2、通道 3 的缩放能力描述；修改表 5-3 中通道 1、通道 3、通道 4、通道 5、通道 6 和通道 7 的缩放能力描述；补充 Hi3518 关于扩展通道的注意事项。 更新 HI_MPI_VPSS_SetExtChnAttr 和 HI_MPI_VPSS_GetExtChnAttr 的定义。 增加“操作不支持”的错误码。 **第 6 章 视频编码** 表 6-5 下增加波动等级的相关描述。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		修改 VENC_ATTR_H264_CBR_S 中的参数 u32StatTime 的取值范围；VENC_ATTR_H264_VBR_S 中的参数 u32StatTime 的取值范围；VENC_ATTR_MPEG4_CBR_S 中的参数 u32StatTime 的取值范围；VENC_ATTR_MPEG4_VBR_S 中的参数 u32StatTime 的取值范围；VENC_ATTR_MJPEG_CBR_S 中的参数 u32StatTime 的取值范围；VENC_ATTR_MJPEG_VBR_S 中的参数 u32StatTime 的取值范围。 **第 7 章 视频侦测分析** 7.1 概述中删除绑定视频输入源的图像宽高描述。 增加 VDA_CHN、VDA_MAX_WIDTH 和 VDA_MAX_HEIGHT。 修改 VDA_OD_RGN_ATTR_S 的成员 stRect 的取值范围；VDA_CHN_ATTR_S 的成员 u32Width 和 u32Height 的取值范围 **第 13 章 Proc 调试信息** 13.18 VD 的 VDA STATE2 增加参数 VpssSd 及说明。
2012-08-30	01	**第 3 章 视频输入** 增加如下 API：HI_MPI_VI_EnableChnInterrupt、HI_MPI_VI_DisableChnInterrupt HI_MPI_VI_SetChnAttr 的【芯片差异】中增加 Hi3518 芯片对通道属性参数的限制。 修改 LDC_VIEW_TYPE_E 的【定义】。 **第 4 章 视频输出** 4.2 重要概念中删除“PIP 层总面积大小不超过 D1”的描述； HI_MPI_VO_EnableChn 的【错误码】中修改最后一个错误码及描述。 修改 HI_MPI_VO_GfxLayerBindDev 和 HI_MPI_VO_GfxLayerUnBindDev 的【描述】和【注意事项】。 VO_DEV 的【成员】中修改 Hi3520A 定义为 1 和 2 的描述。 **第 5 章 视频处理子系统** 修改表 5-2 中通道 0 到通道 3 的缩放能力。 **第 6 章 视频编码** HI_MPI_VENC_SetJpegParam 和 HI_MPI_VENC_SetMjpegParam 中的【注意】和【举例】中修改量化表空间的高级参数。 **第 8 章 区域管理** 更新 OVERLAY_INVERT_COLOR_S 的【定义】和【成员】中的“OSD 反色触发模式”参数。 **第 9 章 音频** 增加 HI_MPI_ADEC_GetFrame 和 HI_MPI_ADEC_ReleaseFrame。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		修改 HI_MPI_ADEC_CreateChn 的【举例】中“get audio frame from adec”和“release audio frame”的部分示例。 **第 11 章 智能加速引擎** 11.2.1 重要概念中，刷新“及时刷 cache”的描述，并增加输入数据类型和输出数据类型的相关描述。 在 API 的【注意】中增加“计算公式”。 **第 13 章 Proc 调试信息** 修改 RC 的调试信息；修改 VI 中 VI 物理通道基本状态 1 参数 Field 的描述。
2012-08-15	00B80	**第 5 章 视频处理子系统** 补充 Hi3518 的相关描述。
2012-07-30	00B70	**第 3 章 视频输入** 3.2 重要概念中增加“镜头畸变矫正”的描述。 3.3 功能描述中的绑定关系中增加“同一组内物理通道和设备绑定”的描述。 增加 3.3.3 Hi3518 的功能描述 增加如下 API：HI_MPI_VI_GetFrameTimeOut、HI_MPI_VI_SetFlashConfig、HI_MPI_VI_GetFlashConfig、HI_MPI_VI_FlashTrigger、HI_MPI_VI_SetExtChnAttr、HI_MPI_VI_GetExtChnAttr、HI_MPI_VI_SetLDCAttr、HI_MPI_VI_GetLDCAttr、HI_MPI_VI_SetCSCAttr、HI_MPI_VI_GetCSCAttr、HI_MPI_VI_SetRotate、HI_MPI_VI_GetRotate、HI_MPI_VI_GetChnLuma。 在各 API 的【芯片差异】中增加 Hi3518 的相关描述。 Hi3531/Hi3532 的通道号取值范围改为[0, VIU_MAX_CHN_NUM) VIU_MAX_PHYCHN_NUM 中【定义】有修改。 VI_USERPIC_MODE_E 的【定义】中有修改。 增加如下数据结构：VIU_EXT_CHN_START、VIU_MAX_EXT_CHN_NUM、VIU_MAX_EXTCHN_BIND_PER_CHNVI_DATA_PATH_E、VI_EXT_CHN_ATTR_S、LDC_VIEW_TYPE_E、LDC_ATTR_S、VI_LDC_ATTR_S、VI_CSC_TYPE_E、VI_CSC_ATTR_S、VI_FLASH_MODE_E、VI_FLASH_CONFIG_S、VI_CHN_LUM_S。 在各数据结构的【芯片差异】中增加 Hi3518 的相关描述。 **第 4 章 视频输出** 增加 Hi3518 的相应描述。 新增 HI_MPI_VO_SetChnDispPos、HI_MPI_VO_GetChnDispPos

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		增加标清设备通道宽高的限制（不小于 16 或不小于 32）。 HI_MPI_VO_SetDevCSC 的【注意事项】中 CSC 转换矩阵的取值限制。 增加 VO_CSC_MATRIX_E。 VO_CSC_S 中增加参数 enCscMatrix。 **第 5 章 视频处理子系统** 5.2.1 基本概念中增加筛选 Sizer 的描述。 增加 HI_MPI_VPSS_SetGrpSizer、HI_MPI_VPSS_GetGrpSizer、HI_MPI_VPSS_SetGrpFrameRate 和 HI_MPI_VPSS_GetGrpFrameRate。 HI_MPI_VPSS_SetPreScale 的【注意事项】中增加“Bypass 通道不支持此功能”。 增加 VPSS_SIZER_INFO_S 和 VPSS_FRAME_RATE_S。 VPSS_GRP_PARAM_S 的【成员】中，修改 u32IeStrength 和 u32IeSharp 的默认值（Hi3521/Hi3520A）；u32SfStrength 和 u32TfStrength 改为保留（Hi3521/Hi3520A）。 VPSS_CHN_MODE_S 的【注意事项】中增加 u32Width 和 u32Height 的取值范围描述。 VPSS_CHN_NR_PARAM_S 的成员 u32SfStrength 的默认值由 8 改为 16。 修改错误代码 **第 6 章 视频编码** 增加 Hi3518 的相应描述。 HI_MPI_VENC_SetGrpCrop、HI_MPI_VENC_GetGrpCrop、HI_MPI_VENC_SetJpegSnapMode、HI_MPI_VENC_GetJpegSnapMode HI_MPI_VENC_SendFrame 的【注意事项】中增加不同的编码通道接收图像格式的不同；视频输入的原始图像和编码通道图像大小的关系。 HI_MPI_VENC_SetH264InterPred 的【注意事项】中增加水平搜索窗和垂直搜索窗的限制。 HI_MPI_VENC_SetRcPara 的【注意事项】中修改低码率时 u32Thrd[12]和 u32QpDelta 的推荐配置；修改 u32MinIprop&u32MaxIprop 的默认值 新增 VENC_JPEG_SNAP_MODE_EN。 **第 8 章 区域管理** 增加 Hi3518 的相应描述。 **第 9 章 音频** 增加 Hi3518 的相应描述。 音频接口时序中 SIO 选择主模式时的相关描述。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		修改音频输入通道号的取值范围描述。 HI_MPI_AO_SetPubAttr 的【注意】中删除“在从模式下，采样率由 codec 决定，采样率的设置不起作用”。 HI_MPI_AENC_DestroyChn 的【注意】中增加“通道未创建的情况下调用此接口会返回成功”的说明；删除“先创建完编码通道，再调用此接口”的说明。 AUDIO_SOUND_MODE_E 的【注意事项】中增加双声道模式的相关描述。 **第 10 章 视频解码** VDEC_ATTR_JPEG_S 的【定义】中删除参数 u32Tmp。 **第 13 章 Proc 调试信息** 修改 SYS、CHNL、GROUP、RC、VENC、VDEC、VI、VO、VPSS、AENC。
2012-07-13	00B60	**第 4 章 视频输出** 表 4-1 中，Hi3521 芯片增加对 PIP 层的说明；Hi3521 芯片将支持的标清设备由 DSD0 改为 DSD1，支持的图形层由 G1 改为 G2。 表 4-2 中，Hi3520A 增加支持 LCD 的说明。 Hi3520A 的 VoDev 的取值范围中[0, 1]改为[0]和[2]，HI_MPI_VO_GetScreenFrame 和 HI_MPI_VO_ReleaseScreenFrame 的接口中，VoDev 的取值范围中[1]改为[2]。 涉及 PIP 层的 API 接口中，Hi3520A 增加“高清设备支持 PIP 层”的说明。 HI_MPI_VO_GfxLayerBindDev 和 HI_MPI_VO_GfxLayerUnBindDev 中，【差异说明】中修改 Hi3531 的描述，并增加参数 VoTargetDev 的取值范围。 VO_DEV 中，【差异说明】里增加 Hi3521 和 Hi3520A 标清设备的相关描述。 VO_VIDEO_LAYER_ATTR_S 中,【差异说明】里增加 Hi3520A 的相关描述。 VO_WBC_MODE_E 中，增加 WBC 回写模式下，WBC 实际回写帧率的相关描述。 **第 10 章 视频解码** 表 10-1 中，Hi3520A 增加支持 VDH；Hi3520A 支持的最大通道数由 32 改为 64；Hi3521 支持的 H.264 协议中删除 MPEG4；Hi3520A 增加支持 H.264 协议的描述；Hi3520A 增加 VDH 模块中 H.264 协议支持的最大分辨率为 1920x2048；删除 Hi3521 支持 MPEG4 的描述。 表 10-3 中，修改 Hi3520 可以创建的最大通道数为≤64；删除参数 s32MaxBitRate；删除 Hi3521 支持 MPEG4 的描述。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		修改 10.5 错误码表中的错误代码。
2012-06-08	00B50	新增 Hi3520A 芯片的相关描述。
2012-04-20	00B40	新增 Hi3521 芯片的相关描述。
2012-03-11	00B30	**第 3 章 视频输入** 更新 HI_MPI_VI_SetDevAttr 中的【举例】。 修改 HI_MPI_VI_SetChnAttr 中的【参数】pstAttr 的描述和【注意事项】中的描述。 修改 HI_MPI_VI_SetFrameDepth 中的【参数】u32Depth 的取值范围为[0, 8]。 新增 HI_MPI_VI_GetFrameDepth。 删除 HI_MPI_VI_SetUserPic、HI_MPI_VI_EnableUserPic、HI_MPI_VI_DisableUserPic 中的【参数】ViDev；修改 ViChn 的取值范围。 更新 HI_MPI_VI_EnableCascade 的【举例】。 修改 VI_CHN_ATTR_S 中的【参数】stCapRect、stDestSize 和 enCapSel 的描述；【注意事项】中增加“隔行图像不支持翻转图像”的描述。 **第 4 章 视频输出** 更新 HI_MPI_VO_SetChnFrameRate、HI_MPI_VO_GetChnFrameRate、HI_MPI_VO_ClearChnBuffer、HI_MPI_VO_GetDispBufLen、HI_MPI_VO_SetWbcDepth 的【举例】。 删除 HI_MPI_VO_ChnPause、HI_MPI_VO_ChnResume、HI_MPI_VO_ChnStep、HI_MPI_VO_ChnShow、HI_MPI_VO_ChnHide 的【注意】中的部分项。 HI_MPI_VO_SetAttrBegin 的【注意】中增加“批处理中通道操作”的相关描述；更新【举例】。 修改 HI_MPI_VO_SetDispBufLen 的【参数】u32BufLen 的取值范围；修改并新增【注意】；补充【举例】。 HI_MPI_VO_EnableWbc 中增加【注意】；更新【举例】 HI_MPI_VO_SetWbcAttr、VO_WBC_MODE_E 中的【注意】新增并修改。 修改 HI_MPI_VO_PipLayerUnBindDev 的【参数】VoTargetDev 的取值范围；【注意】中增加关于“重复解绑定和对绑定设备”的相关描述。 HI_MPI_VO_SetWbcMode 中的【注意】新增 2 条。 新增 HI_MPI_VO_GetWbcMode、HI_MPI_VO_SetPlayToleration、HI_MPI_VO_GetPlayToleration、HI_MPI_VO_SetDevCSC、HI_MPI_VO_GetDevCSC、HI_MPI_VO_SetVgaParam 和

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		HI_MPI_VO_GetVgaParam。 新增 VO_CSC_S 和 VO_VGA_PARAM_S。 VO_CHN_ATTR_S 中增加成员及描述。 **第 6 章 视频编码** 修改表 6-2 下面和 VENC_ATTR_H264_CBR_S 中关于 ViFrmRate 的相关描述。 增加 6.2.4 跳帧参考； HI_MPI_VENC_InsertUserData 的【注意】中增加关于“向 H.264 通道插入用户数据时”的描述。 HI_MPI_VENC_CreateChn 的【注意】中增加注意事项。 HI_MPI_VENC_SetChnAttr 中增加“不支持动态改变宽高”的描述。 VENC_PARAM_JPEG_S 的【参数】u32Qfactor 的默认值改为 90。 新增 VENC_PARAM_MJPEG_S VENC_PARAM_H264_INTRA_PRED_S 、VENC_PARAM_H264_TRANS_S、VENC_PARAM_H264_ENTROPY_S、VENC_PARAM_H264_VUI_S、VENC_PARAM_JPEG_S、VENC_RC_PARAM_S 补充部分参数的取值范围。 VENC_ATTR_H264_CBR_S 中增加【注意事项】 **第 7 章 视频侦测分析** 修改 VDA_CHN_NUM_MAX 的定义。 **第 8 章 区域管理** OVERLAY_CHN_ATTR_S 增加一项注意并增加图 8-2。 **第 9 章 音频** 修改图 9-4 下面的描述。 修改 HI_MPI_AI_EnableReSmp、HI_MPI_AO_EnableReSmp、HI_MPI_AO_ResumeChn、 HI_MPI_AENC_DestroyChn 的【注意】。 新增 HI_MPI_AENC_GetFd。 AUDIO_SOUND_MODE_E 增加注意事项。 HI_MPI_VDEC_GetImage 的【注意】中增加“VDEC 与其他模块绑定发送图像时”的相关描述。 表 9-5 和表 9-6 中，增加 0xA0158005 对应的宏定义，删除 0xA015800C 对应的宏定义。 **第 10 章 视频解码** VDEC_PRTCL_PARAM_S 的【成员】s32SCDBufSize 的描述增加最小取值的说明。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		第 11 章 智能加速引擎 修改所有 API 的【注意】。 **第 13 章 Proc 调试信息** 更新 GROUP、H264E、JPEGE、VENC、VDEC、VI、VO 和 MPEG4。 新增 HDMI。
2012-02-15	00B20	**第 2 章 系统控制** 新增 HI_MPI_SYS_MmzAlloc_Cached 和 HI_MPI_SYS_MmzFlushCache。 **第 3 章 视频输入** 新增 HI_MPI_VI_Query 和 VI_CHN_STAT_S。 **第 4 章 视频输出** 4.2 重要概念中增加 VGA 兼容性的说明。 新增 HI_MPI_VO_SetDispBufLen、HI_MPI_VO_GetDispBufLen 和 HI_MPI_VO_SetWbcMode。 修改参数 VoDev 的取值范围。 修改 HI_MPI_VO_SetChnAttr 和 VO_CHN_ATTR_S 中的【注意】中关于 HD、SD 设备的通道属性 stRect 的使用注意事项。 修改 HI_MPI_VO_SetZoomInWindow 和 HI_MPI_VO_EnableWbc 的【举例】。 HI_MPI_VO_SetAttrEnd 的【注意】中删除“调用前需保证所要设置的通道已经使能”的描述。 HI_MPI_VO_SetWbcAttr 的【注意】中删除“X 和 Y 的使用限制”。 修改 VO_DEV 的【成员】中的描述。 新增 VO_WBC_MODE_E。 **第 5 章 视频处理子系统** 更新图 5-1。 修改图 5-2 下面的描述。 新增 HI_MPI_VPSS_SetChnMode、HI_MPI_VPSS_GetChnMode、HI_MPI_VPSS_SetDepth、HI_MPI_VPSS_GetDepth、HI_MPI_VPSS_UserGetFrame 和 HI_MPI_VPSS_UserRlsFrame。 HI_MPI_VPSS_UserSendFrame 的【注意】中增加 2 条事项。 修改 VPSS_CLIP_INFO_S 中【注意事项】的描述。 新增 VPSS_CHN_MODE_E 和 VPSS_CHN_MODE_S。 **第 6 章 视频编码** 新增 HI_MPI_VENC_SetH264eRefMode 和 HI_MPI_VENC_GetH264eRefMode。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		新增 H264E_REFSLICE_TYPE_E。 VENC_STREAM_INFO_H264_S 新增参数 enRefSliceType 及描述，并增加注意事项。 **第 7 章 视频侦测分析** 修改 VDA_OD_RGN_ATTR_S 的成员 u32SadTh 的取值范围。 **第 8 章 区域管理** 将 Vicover 改为 Cover，VicoverEx 改为 CoverEx，Vioverlay 改为 OverlayEx，并修改相应的 API 和数据类型及相关描述等。 **第 9 章 音频** 删除目前已经支持的原“暂不支持”的相关描述。 修改 9.2.2.1 音频编解码流程的描述。 修改表 9-3。 **第 10 章 视频解码** 新增 HI_MPI_VDEC_SetPrtclParam 和 HI_MPI_VDEC_GetPrtclParam。 新增 VDEC_PRTCL_PARAM_S。 **第 12 章 Proc 调试信息** 更新 SYS、GROUP、VI 和 VPSS 的调试信息及参数说明。 新增**第 13 章 HDMI**
2012-01-15	00B10	**第 2 章 系统控制** 修改 2.4.1 基本数据类型下的一些定义和说明 **第 3 章 视频输入** 修改图 3-3 下面，“当 Dev N 设置为 X 路复合模式”时与该设备绑定的通道的描述。 优化 3.2.4 掩码配置小节的描述。 **第 4 章 视频输出** 4.2 重要概念中增加裁减和缩放的概念。 HI_MPI_VO_ChnPause、HI_MPI_VO_ChnResume 和。HI_MPI_VO_ChnStep 的【注意】中删除关于同步组的相关描述。 HI_MPI_VO_SetAttrBegin 的【注意】中对该接口的功能进行了详细的描述。 HI_MPI_VO_SetWbcDepth 的【注意】中修改了描述。 **第 5 章 视频处理子系统** VPSS_SF_WINDOW_E 的【描述】中增加了窗大小和滤波强度的关系。 VPSS_GRP_PARAM_S 的【成员】中增加了各成员影响的范围。 **第 6 章 视频编码** HI_MPI_VENC_SetH264InterPred 的【注意】中增加了水平和

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		垂直搜索窗的大小和图像的宽度存在的限制。 HI_MPI_VENC_SetRcPara 的【注意】中增加了对高级参数的详细描述。 MPEG4E_PACK_TYPE_E、VENC_STREAM_INFO_MPEG4_S、VENC_PARAM_H264_INTRA_PRED_S 的【定义】和【成员】有修改。 **第 9 章 音频** 修改图 9-1。 9.2.1.3 音频接口时序和 AI、AO 通道排列中增加不同的采样率时的主时钟相关的描述。
2011-12-07	00B04	**第 3 章 视频输入** 增加级联扩展通道的相关内容； 增加 3.2.5 级联模式章节； 增加 HI_MPI_VI_SetUserPic、HI_MPI_VI_EnableUserPic、HI_MPI_VI_DisableUserPic、HI_MPI_VI_EnableCascade、HI_MPI_VI_DisableCascade、HI_MPI_VI_EnableCascadeChn、HI_MPI_VI_DisableCascadeChn。 修改 VI 通道的取值范围为[0, VIU_MAX_CHN_NUM-2)及相关描述。 增加 VI_CAS_CHN_1、VI_CAS_CHN_2、VI_USERPIC_MODE_E、VI_USERPIC_BGC_S、VI_USERPIC_ATTR_S。 修改 VI_DEV_ATTR_S 的【注意事项】。 **第 4 章 视频输出** 增加视频级联的概念； 增加 HI_MPI_VO_SendFrame；增加级联操作的相关 API。 VO_DEV 的【成员】和【注意事项】中增加级联扩展设备的相关描述。 增加级联的数据结构。 **第 5 章 视频子系统** 增加 HI_MPI_VPSS_UserSendFrame HI_MPI_VPSS_GetChnAttr 的【注意事项】中通道 1～通道 3 支持缩写的倍数改为 16。 增加 VPSS_DISPLAY_MODE_E、VPSS_SF_WINDOW_E、RECT_S。 更新 VPSS_CLIP_INFO_S 的【注意事项】中的公式。 修改 VPSS_GRP_ATTR_S 的【成员】中部分参数的描述及修改注意事项。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

修订日期	版本	修订说明
		修改 VPSS_FRAME_S 的【注意事项】。 **第 6 章 视频编码** 删除原暂不支持 MPEG-4 的相关说明。 增加了 VENC_PARAM_MJPEG_CBR_S、VENC_PARAM_MJPEG_VBR_S、VENC_PARAM_MPEG4_CBR_S、VENC_PARAM_MPEG4_VBR_S、VENC_RC_PARAM_S。 修改了编码图形宽度和高度参数的为静态属性。 更新 VENC_ATTR_MPEG4_S、VENC_ROI_CFG_S、VENC_RC_PARAM_S 的【定义】和【参数】。 新增**第 7、9、11 章**。 **第 10 章 视频解码** 修改 Hi3531 解码器提供的两种码流发送方式。 **第 12 章 Proc 调试信息** 修改各模块的调试信息，并增加 IVE、AI、AO、VDA、AENC、ADEC 和 MPEG4。
2011-11-10	00B03	第 3 次临时版本发布。
2011-09-30	00B02	第 2 次临时版本发布。
2011-08-15	00B01	第 1 次临时版本发布。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

1 系统概述.....1-1
1.1 概述.....1-1
1.2 系统架构.....1-1
1.3 海思媒体处理平台架构.....1-2

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

1 系统概述

1.1 概述

海思提供的媒体处理软件平台(Media Process Platform,简称 MPP)，可支持应用软件快速开发。该平台对应用软件屏蔽了芯片相关的复杂的底层处理，并对应用软件直接提供 MPI（MPP Programe Interface）接口完成相应功能。该平台支持应用软件快速开发以下功能：输入视频捕获、H.264/MJPEG/JPEG/MPEG4 编码、H264/VC1/MPEG4/MPEG2/AVS 解码、视频输出显示、视频图像前处理（包括去噪、增强、锐化、Deinterlace）、编码码流叠加 OSD、视频侦测分析、智能分析、音频捕获及输出、音频编解码等功能。

1.2 系统架构

- MPP 平台支持的典型的系统层次如图 1-1 所示，主要分为以下层次：硬件层

  硬件层由 Hi35xx 芯片加上必要的外围器件构成。外围器件包括 Flash、DDR（Double Data-Rate）、视频 Sensor 或 AD、音频 AD 等。
- 操作系统层

  基于 Linux 3.0 的 OS 系统。
- 媒体处理平台

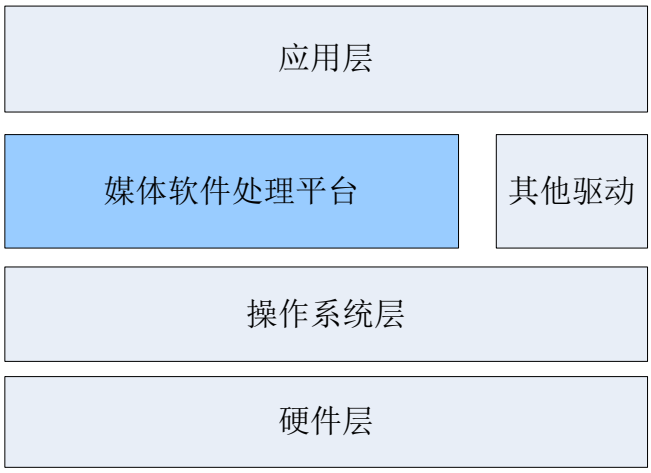
  基于操作系统层，控制芯片完成相应的媒体处理功能。它对应用层屏蔽了硬件处理细节，并为应用层提供 API 接口完成相应功能。
- 其他驱动

  除媒体处理平台外,海思为 Hi35xx 芯片的其他相关硬件处理单元提供了相应的驱动,包括 GMAC、SDIO、I2C、USB、SSP 等驱动。
- 应用层

  基于海思媒体处理平台及其他驱动，由用户开发的应用软件系统。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图1-1 Hi35xx 典型的系统层次图

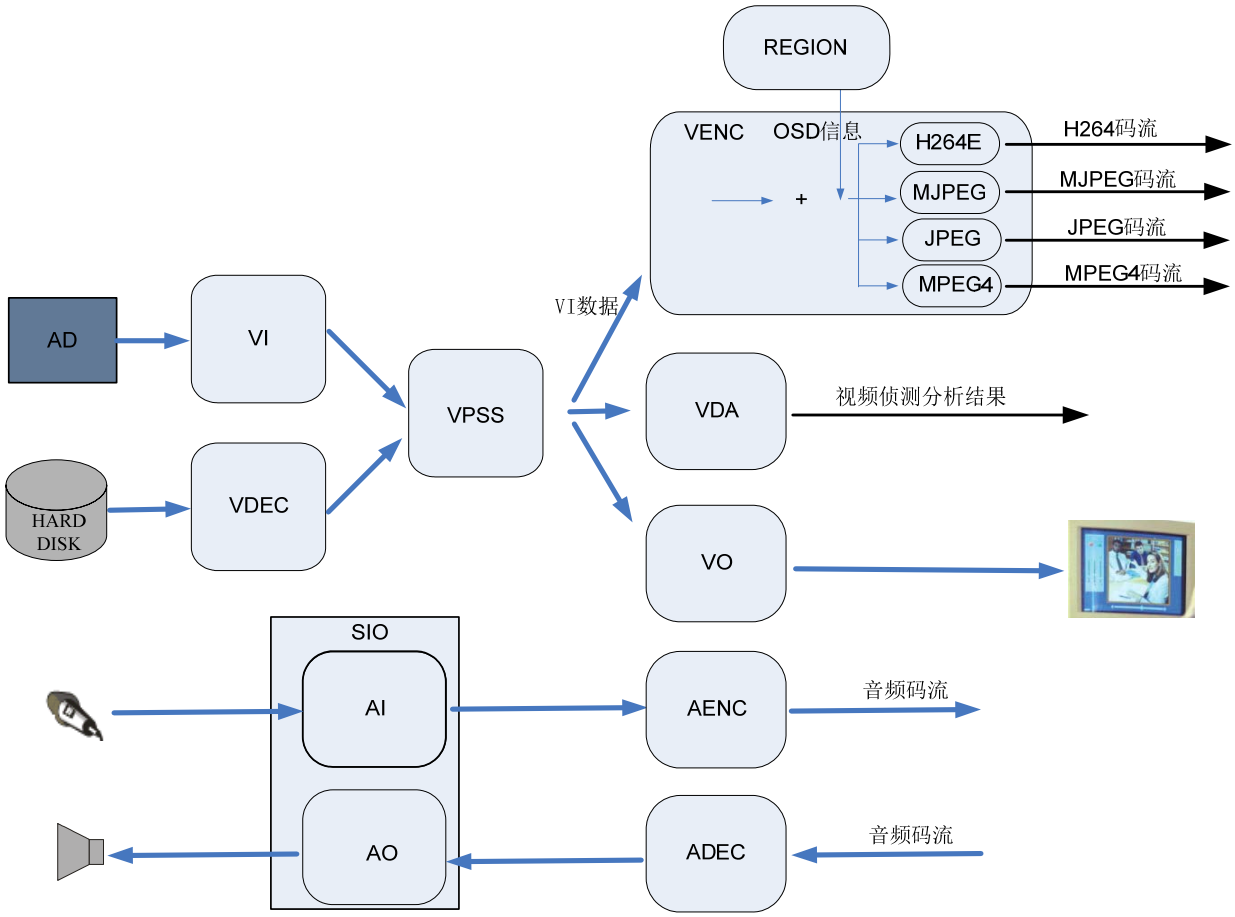


1.3 海思媒体处理平台架构

海思媒体处理平台的主要内部处理流程如图 1-2 所示，主要分为视频输入（VI）、视频处理（VPSS）、视频编码（VENC）、视频解码（VDEC）、视频输出(VO)、视频侦测分析(VDA)、音频输入(AI)、音频输出(AO)、音频编码（AENC）、音频解码（ADEC）、区域管理（REGION）等模块。主要的处理流程介绍如下：

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图1-2 海思媒体处理平台内部处理流程图



- VI 模块捕获视频图像，可对其做剪切、缩放、镜像等处理，并输出多路不同分辨率的图像数据。
- 解码模块对编码后的视频码流进行解码，并将解析后的图像数据送 VPSS 进行图像处理或直接送 VO 显示。可对 H264/VC1/MPEG4/MPEG2/AVS 格式的视频码流进行解码。
- VPSS 模块接收 VI 和解码模块发送过来的图像，可对图像进行去噪、图像增强、锐化等处理，并实现同源输出多路不同分辨率的图像数据用于编码、预览或抓拍。
- 编码模块接收 VI 捕获并经 VPSS 处理后输出的图像数据，可叠加用户通过 Region 模块设置的 OSD 图像，然后按不同协议进行编码并输出相应码流。
- VDA 模块接收 VI 的输出图像，并进行移动侦测和遮挡侦测，最后输出侦测分析结果。
- VO 模块接收 VPSS 处理后的输出图像，可进行播放控制等处理，最后按用户配置的输出协议输出给外围视频设备。
- AI 模块捕获音频数据，然后 AENC 模块支持按多种音频协议对其进行编码，最后输出音频码流。

用户从网络或外围存储设备获取的音频码流可直接送给 ADEC 模块，ADEC 支持解码多种不同的音频格式码流，解码后数据送给 AO 模块即可播放声音。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

2 系统控制.....2-1
2.1 概述.....2-1
2.2 功能描述.....2-1
2.2.1 视频缓存池.....2-1
2.2.2 系统绑定.....2-2
2.3 API 参考.....2-3
2.4 数据类型.....2-39
2.4.1 基本数据类型.....2-39
2.4.2 系统控制数据类型.....2-51
2.4.3 视频公共类型.....2-53
2.5 错误码.....2-59
2.5.1 系统控制错误码.....2-59
2.5.2 视频缓存池错误码.....2-59

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 2-1 HiMPP 支持的绑定关系.....2-2
表 2-2 系统控制 API 错误码.....2-59
表 2-3 视频缓存池 API 错误码.....2-59

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

2 系统控制

2.1 概述

系统控制根据 Hi35xx 芯片特性，完成硬件各个部件的复位、基本初始化工作，同时负责完成 MPP（Media Process Platform 媒体处理平台）系统各个业务模块的初始化、去初始化以及管理 MPP 系统各个业务模块的工作状态、提供当前 MPP 系统的版本信息、提供大块物理内存管理等功能。

应用程序启动 MPP 业务前，必须完成 MPP 系统初始化工作。同理，应用程序退出 MPP 业务后，也要完成 MPP 系统去初始化工作，释放资源。

说明
MPP 不支持多进程调用。

2.2 功能描述

2.2.1 视频缓存池

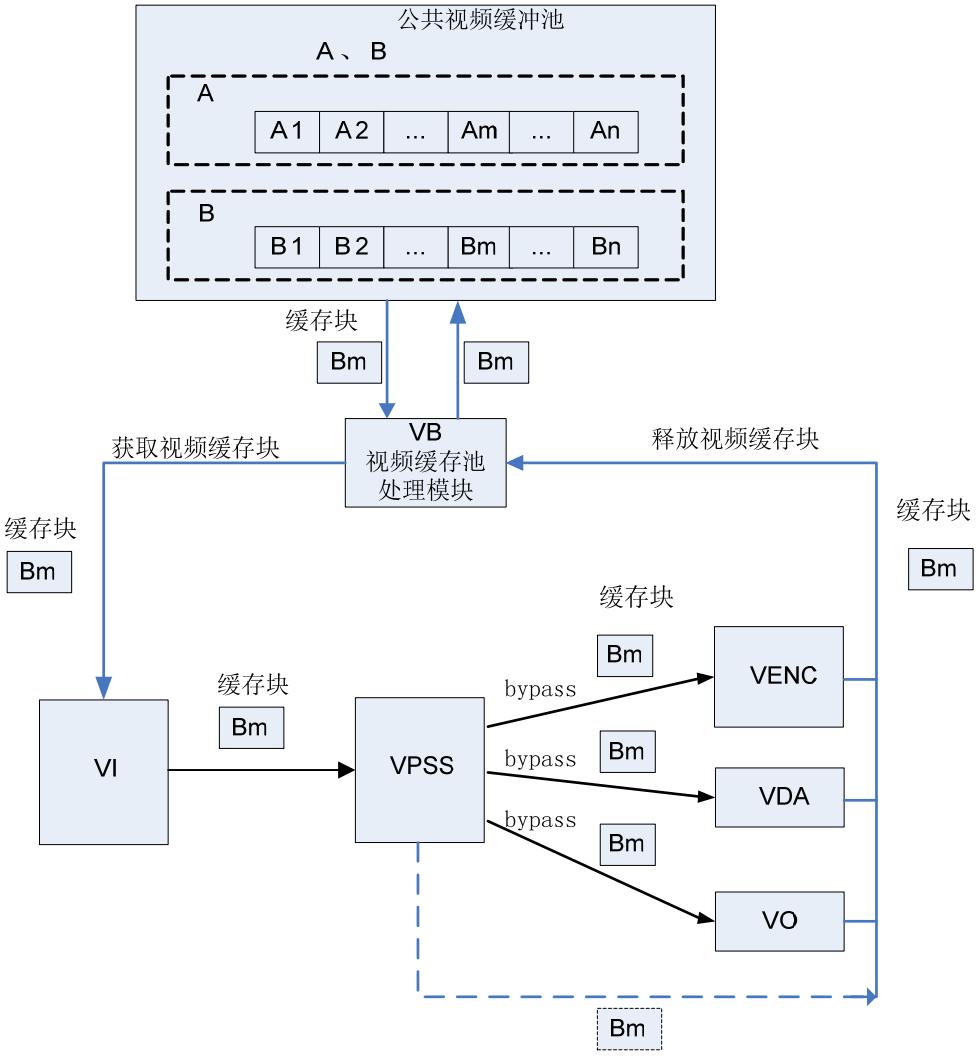
视频缓存池主要向媒体业务提供大块物理内存管理功能，负责内存的分配和回收，充分发挥内存缓存池的作用，让物理内存资源在各个媒体处理模块中合理使用。

一组大小相同、物理地址连续的缓存块组成一个视频缓存池。

对于视频输入通道，则需要使用公共视频缓存池。所有的视频输入通道都可以从公共视频缓存池中获取视频缓存块用于保存采集的图像（如图 2-1 中所示从公共视频缓存池 A 中获取视频缓存块 Bm）。由于视频输入通道不提供创建和销毁公共视频缓存池功能，因此，在系统初始化之前，必须为视频输入通道配置公共视频缓存池。根据业务的不同，公共缓存池的数量、缓存块的大小和数量会有所不同。图 2-1 中所示缓存块的生存期是指经过 VPSS 的 bypass 通道传给后续模块的情形（图 2-1 实线路径）。如果该缓存块完全没有经过 VPSS 的 bypass 通道透传给其他模块，则将在 VPSS 模块处理后被放回公共缓存池（图 2-1 虚线路径）。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图2-1 典型的公共视频缓存池数据流图



2.2.2 系统绑定

MPP 提供系统绑定接口（HI_MPI_SYS_Bind），即通过数据接收者绑定数据源来建立两者之间的关联关系（只允许数据接收者绑定数据源）。绑定后，数据源生成的数据将自动发送给接收者。目前 HiMPP 支持的绑定关系如表 2-1 所示。

表2-1 HiMPP 支持的绑定关系

数据源	数据接收者
VI	VO
	VENC
	VDA
	VPSS

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

数据源	数据接收者
VPSS	VO
	VENC
	VDA
VDEC	VPSS
	VO（只能是标清/虚拟设备）
	VDA
VO（WBC、虚拟设备）	VO
	VENC
	VPSS
AI	AENC
	AO
ADEC	AO

2.3 API 参考

系统控制实现 MPP（Media Process Platform）系统初始化、获取 MPP 版本号、视频缓存池初始化、创建视频缓存池等功能。

该功能模块提供以下 MPI：

- HI_MPI_SYS_SetConf：配置系统控制参数。
- HI_MPI_SYS_GetConf：获取系统控制参数。
- HI_MPI_SYS_Init：初始化 MPP 系统。
- HI_MPI_SYS_Exit：去初始化 MPP 系统。
- HI_MPI_SYS_Bind：数据源到数据接收者绑定。
- HI_MPI_SYS_UnBind：数据源到数据接收者解绑定。
- HI_MPI_SYS_GetBindbyDest：获取绑定的源。
- HI_MPI_SYS_GetVersion：获取 MPP 的版本号。
- HI_MPI_SYS_GetCurPts：获取当前时间戳。
- HI_MPI_SYS_InitPtsBase：初始化 MPP 时间戳。
- HI_MPI_SYS_SyncPts：同步 MPP 时间戳。
- HI_MPI_SYS_Mmap：存储映射接口。
- HI_MPI_SYS_Munmap：存储反映射接口。
- HI_MPI_SYS_SetReg：设置寄存器的值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_SYS_GetReg：获取寄存器的值。
- HI_MPI_SYS_MmzAlloc：在用户态分配 MMZ 内存。
- HI_MPI_SYS_MmzAlloc_Cached：在用户态分配 MMZ 内存，该内存支持 cache 缓存。
- HI_MPI_SYS_MmzFlushCache：清空 cache 里的内容到内存并且使 cache 里的内容无效。
- HI_MPI_SYS_MmzFree：在用户态释放 MMZ 内存。
- HI_MPI_SYS_SetMemConf：设置模块设备通道使用内存的 DDR 名。
- HI_MPI_SYS_GetMemConf：获取模块设备通道使用内存的 DDR 名。
- HI_MPI_SYS_CloseFd：关闭所有 SYS 打开的日志、系统、内存 Fd。
- HI_MPI_VB_SetConf：设置 MPP 视频缓存池属性。
- HI_MPI_VB_GetConf：获取 MPP 视频缓存池属性。
- HI_MPI_VB_Init：初始化 MPP 视频缓存池。
- HI_MPI_VB_Exit：去初始化 MPP 视频缓存池。
- HI_MPI_VB_CreatePool：创建一个视频缓存池。
- HI_MPI_VB_DestroyPool：销毁一个视频缓存池。
- HI_MPI_VB_GetBlock：获取一个缓存块。
- HI_MPI_VB_ReleaseBlock：释放一个已经获取的缓存块。
- HI_MPI_VB_Handle2PhysAddr：获取一个缓存块的物理地址。
- HI_MPI_VB_Handle2PoolId：获取一个缓存块所在缓存池的 ID。
- HI_MPI_VB_MmapPool：为一个视频缓存池映射用户态虚拟地址。
- HI_MPI_VB_MunmapPool：为一个视频缓存池解除用户态映射。
- HI_MPI_VB_GetBlkVirAddr：获取一个视频缓存池中的缓存块的用户态虚拟地址。
- HI_MPI_VB_ GetVRegAddr：获取 ISP 寄存器的物理地址。

HI_MPI_SYS_SetConf

【描述】

配置系统控制参数。

【语法】

```
HI_S32 HI_MPI_SYS_SetConf(const MPP_SYS_CONF_S *pstSysConf);
```

【参数】

参数名称	描述	输入/输出
pstSysConf	系统控制参数指针。 静态属性（指只能在系统未初始化、未启用设备或通道时，才能设置的属性）。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

只有在 MPP 整个系统处于未初始化状态，才可调用此函数配置 MPP 系统，否则会配置失败。

【举例】

```
HI_S32 s32ret;
VB_CONF_S struVbConf;
MPP_SYS_CONF_S struSysConf;

memset(&struVbConf,0,sizeof(VB_CONF_S));

struVbConf.u32MaxPoolCnt             = 64;
struVbConf.astCommPool[0].u32BlkSize  = 1920*1088*2;
struVbConf.astCommPool[0].u32BlkCnt   = 15;
memset(struVbConf.astCommPool[0].acMmzName,0,sizeof(struVbConf.astCommPoo
l[0].acMmzName));
s32ret = HI_MPI_VB_SetConf(&struVbConf);
if (HI_SUCCESS != s32ret)
{
    return s32ret;
}
s32ret = HI_MPI_VB_Init();
if (HI_SUCCESS != s32ret)
{
    return s32ret;
}
struSysConf.u32AlignWidth = 16;
/* set config of mpp system*/
s32ret = HI_MPI_SYS_SetConf(&struSysConf);
if (HI_SUCCESS != s32ret)
{
    printf("Set mpp sys config failed!\n");
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    return s32ret;
}

/* init system*/
s32ret = HI_MPI_SYS_Init();
if (HI_SUCCESS != s32ret)
{
    printf("Mpi init failed!\n");
    return s32ret;
}

/* ...... */

/* exit system*/
s32ret = HI_MPI_SYS_Exit();
if (HI_SUCCESS != s32ret)
{
    printf("Mpi exit failed!\n");
    return s32ret;
}

s32ret = HI_MPI_VB_Exit();
if (HI_SUCCESS != s32ret)
{
    return s32ret;
}
```

【相关主题】

HI_MPI_VB_GetConf

HI_MPI_SYS_GetConf

【描述】

获取系统控制参数。

【语法】

```
HI_S32 HI_MPI_SYS_GetConf(MPP_SYS_CONF_S *pstSysConf);
```

【参数】

参数名称	描述	输入/输出
pstSysConf	系统控制参数指针。 静态属性。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

必须先调用 HI_MPI_SYS_SetConf 成功后才能获取配置。

【举例】

无。

【相关主题】

HI_MPI_SYS_SetConf

HI_MPI_SYS_Init

【描述】

初始化 MPP 系统。除了音频的编解码通道外，所有的音频输入输出、视频输入输出、视频编码、视频叠加区域、视频侦测分析等都会被初始化。

【语法】

```
HI_S32 HI_MPI_SYS_Init(HI_VOID);
```

【参数】

无。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 必须先调用 HI_MPI_SYS_SetConf 配置 MPP 系统后才能初始化，否则初始化会失败。
- 由于 MPP 系统的正常运行依赖于缓存池，因此必须先调用 HI_MPI_VB_Init 初始化缓存池，再初始化 MPP 系统。
- 如果多次初始化，仍会返回成功，但实际上系统不会对 MPP 的运行状态有任何影响。
- 只要有一个进程进行初始化即可，不需要所有的进程都做系统初始化的操作。

【举例】

请参见 HI_MPI_SYS_SetConf 的举例。

【相关主题】

HI_MPI_SYS_Exit

HI_MPI_SYS_Exit

【描述】

去初始化 MPP 系统。除了音频的编解码通道外，所有的音频输入输出、视频输入输出、视频编码、视频叠加区域、视频侦测分析通道等都会被销毁或者禁用。

【语法】

```
HI_S32 HI_MPI_SYS_Exit(HI_VOID);
```

【参数】

无。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

- 去初始化时，如果有阻塞在 MPI 上的用户进程，则去初始化会失败。如果所有阻塞在 MPI 上的调用都返回，则可以成功去初始化。
- 可以反复去初始化，不返回失败。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 由于系统去初始化不会销毁音频的编解码通道，因此这些通道的销毁需要用户主动进行。如果创建这些通道的进程退出，则通道随之被销毁。

【举例】

请参见 HI_MPI_SYS_SetConf 的举例。

【相关主题】

HI_MPI_SYS_Init

HI_MPI_SYS_Bind

【描述】

数据源到数据接收者绑定接口。

【语法】

```
HI_S32  HI_MPI_SYS_Bind(const MPP_CHN_S *pstSrcChn, const MPP_CHN_S
*pstDestChn);
```

【参数】

参数名称	描述	输入/输出
pstSrcChn	源通道指针。	输入
pstDestChn	目的通道指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

- 系统目前支持的绑定关系，请参见表 2-1。
- 同一个数据接收者只能绑定一个数据源。
- 绑定是指数据源和数据接收者建立关联关系。绑定后，数据源生成的数据将自动发送给接收者。
- VI 和 VDEC 作为数据源，是以通道为发送者，向其他模块发送数据，用户将设备号置为 0，SDK 不检查输入的设备号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VO 作为数据源发送回写(WBC)数据时，是以设备为发送者，向其他模块发送数据，用户将通道号置为 0，SDK 不检查输入的通道号。
- VPSS 和 VENC 作为数据接收者时，是以设备（GROUP）为接收者，接收其他模块发来的数据，用户将通道号置为 0，SDK 不检查输入的通道号。
- 其他情况均需指定设备号和通道号。

【举例】

```
HI_S32 s32Ret;
MPP_CHN_S stBindSrc;
MPP_CHN_S stBindDest;
stBindDest.enModId = HI_ID_VPSS;
stBindDest.s32ChnId = 0;
stBindDest.s32DevId = 0;

stBindSrc.enModId = HI_ID_VIU;
stBindSrc.s32ChnId = 0;
stBindSrc.s32DevId = 0;

s32Ret = HI_MPI_SYS_Bind(&stBindSrc, &stBindDest);
if (s32Ret)
{
return HI_FAILURE;
}
```

【相关主题】

HI_MPI_SYS_UnBind

HI_MPI_SYS_UnBind

【描述】

数据源到数据接收者解绑定接口。

【语法】

```
HI_S32  HI_MPI_SYS_UnBind(const MPP_CHN_S *pstSrcChn, const MPP_CHN_S
*pstDestChn);
```

【参数】

参数名称	描述	输入/输出
pstSrcChn	源通道指针。	输入
pstDestChn	目的通道指针。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

无。

【举例】

请参见 Sample 的举例。

【相关主题】

HI_MPI_SYS_Bind

HI_MPI_SYS_GetBindbyDest

【描述】

获取此通道上绑定的源通道的信息。

【语法】

```
HI_S32  HI_MPI_SYS_GetBindbyDest(const MPP_CHN_S *pstDestChn, MPP_CHN_S *pstSrcChn);
```

【参数】

参数名称	描述	输入/输出
pstSrcChn	源通道指针。	输出
pstDestChn	目的通道指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

无。

【举例】

请参见 Sample 的举例。

【相关主题】

- HI_MPI_SYS_Bind
- HI_MPI_SYS_UnBind

HI_MPI_SYS_GetVersion

【描述】

获取 MPP 的版本号。

【语法】

```
HI_S32 HI_MPI_SYS_GetVersion(MPP_VERSION_S *pstVersion);
```

【参数】

参数名称	描述	输入/输出
pstVersion	版本号描述指针。 动态属性（指在任何时刻都可以设置的属性）。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

无。

【举例】

```
HI_S32 s32ret;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
MPP_VERSION_S stVersion;

s32ret = HI_MPI_SYS_GetVersion(&stVersion);
if (HI_SUCCESS != s32ret)
{
    return s32ret;
}
printf("mpi version is %s\n", stVersion.aVersion);
```

【相关主题】

无。

HI_MPI_SYS_GetCurPts

【描述】

获取 MPP 的当前时间戳。

【语法】

```
HI_S32 HI_MPI_SYS_GetCurPts(HI_U64 *pu64CurPts);
```

【参数】

参数名称	描述	输入/输出
pu64CurPts	当前时间戳指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

HI_MPI_SYS_InitPtsBase

【描述】

初始化 MPP 的时间戳基准。

【语法】

```
HI_S32 HI_MPI_SYS_InitPtsBase(HI_U64 u64PtsBase);
```

【参数】

参数名称	描述	输入/输出
u64PtsBase	时间戳基准。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

初始化时间戳基准会将当前系统的时间戳强制置成 u64PtsBase，与系统原有时间戳没有任何约束。因此，建议在媒体业务没有启动时（例如操作系统刚启动），调用这个接口。如果媒体业务已经启动，建议调用 HI_MPI_SYS_SyncPts 进行时间戳微调。

【举例】

无。

【相关主题】

无。

HI_MPI_SYS_SyncPts

【描述】

同步 MPP 的时间戳。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_SYS_SyncPts(HI_U64 u64PtsBase);
```

【参数】

参数名称	描述	输入/输出
u64PtsBase	时间戳基准。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

对当前系统时间戳进行微调，微调后不会出现时间戳回退现象。在多片之间做同步时，由于单板的时钟源误差可能比较大，建议一秒钟进行一次时间戳微调。

【举例】

无。

【相关主题】

无。

HI_MPI_SYS_Mmap

【描述】

存储映射接口。

【语法】

```
HI_VOID * HI_MPI_SYS_Mmap(HI_U32 u32PhyAddr, HI_U32 u32Size);
```

【参数】

参数名称	描述	输入/输出
u32PhyAddr	需映射的内存单元起始地址。	输入
u32Size	映射的字节数。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	无效地址。
非 0	有效地址

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

- 等同于 LINUX 系统 mmap 函数
- 输入的地址需为合法的物理地址

【举例】

无。

【相关主题】

HI_MPI_SYS_Munmap

HI_MPI_SYS_Munmap

【描述】

存储反映射接口。

【语法】

```
HI_S32 HI_MPI_SYS_Munmap(HI_VOID* pVirAddr, HI_U32 u32Size);
```

【参数】

参数名称	描述	输入/输出
pVirAddr	mmap 后返回的地址。	输入
u32Size	映射区的字节长度。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

等同于 LINUX 系统 munmap 函数。

【举例】

无。

【相关主题】

HI_MPI_SYS_Mmap

HI_MPI_SYS_SetReg

【描述】

设置寄存器的值

【语法】

```
HI_S32 HI_MPI_SYS_SetReg(HI_U32 u32Addr, HI_U32 u32Value);
```

【参数】

参数名称	描述	输入/输出
u32Addr	写入的地址。	输入
u32Value	写入的值。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

输入的地址必须为有效的物理地址。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

HI_MPI_SYS_GetReg

HI_MPI_SYS_GetReg

【描述】

获取寄存器的值。

【语法】

```
HI_S32 HI_MPI_SYS_GetReg(HI_U32 u32Addr, HI_U32 *pu32Value);
```

【参数】

参数名称	描述	输入/输出
u32Addr	物理地址。	输入
pu32Value	此内存地址中的值。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

输入的地址必须为有效的物理地址。

【举例】

无。

【相关主题】

HI_MPI_SYS_SetReg

HI_MPI_SYS_MmzAlloc

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

在用户态分配 MMZ 内存。

【语法】

```
HI_S32 HI_MPI_SYS_MmzAlloc(HI_U32 *pu32PhyAddr, void **ppVitAddr, const HI_CHAR *pstrMmb, const HI_CHAR *pstrZone, HI_U32 u32Len);
```

【参数】

参数名称	描述	输入/输出
pu32PhyAddr	物理地址指针。	输出
ppVirtAddr	指向虚拟地址指针的指针。	输出
pstrMmb	Mmb 名称的字符串指针。	输入
pstrZone	MMZ zone 名称的字符串指针。	输入
u32Len	内存块大小。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

MMZ 分为许多区域（Zone），每个区域下有多个 Mmb，调用此接口在 MMZ 的名为*pstrZone 的区域中分配一个名为*pstrMmb 的内存块，大小为 u32Len，并返回物理地址和用户态虚拟地址指针。如果 MMZ 中有名为 anonymous 的区域，*pstrZone 可设为 NULL。如果*pstrMmb 设为 NULL，创建的 Mmb 分块名为“<null>”。

【举例】

无。

【相关主题】

无。

HI_MPI_SYS_MmzAlloc_Cached

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

在用户态分配 MMZ 内存，该内存支持 cache 缓存。

【语法】

```
HI_S32 HI_MPI_SYS_MmzAlloc_Cached(HI_U32 *pu32PhyAddr, HI_VOID
**ppVitAddr, const HI_CHAR *pstrMmb, const HI_CHAR *pstrZone, HI_U32
u32Len);
```

【参数】

参数名称	描述	输入/输出
pu32PhyAddr	物理地址指针。	输出
ppVirtAddr	指向虚拟地址指针的指针。	输出
pstrMmb	Mmb 名称的字符串指针。	输入
pstrZone	MMZ zone 名称的字符串指针。	输入
u32Len	内存块大小。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

- 本接口与 HI_MPI_SYS_MmzAlloc 接口的区别：通过本接口分配的内存支持 cache 缓存，对于频繁使用的内存，最好使用本接口分配内存，这样可以提高 cpu 读写的效率，提升系统性能，如用户在使用 ive 算子时，就存在大量数据频繁读写，此时使用此接口来分配内存，就能很好的提高 cpu 的效率。
- 当 cpu 访问此接口分配的内存时，会将内存中的数据放在 cache 中，而硬件设备(如 ive)只能访问物理内存，不能访问 cache 的内容，对于这种 cpu 和硬件会共同操作的内存，需调用 HI_MPI_SYS_MmzFlushCache 做好数据同步。

【举例】

无。

【相关主题】

- HI_MPI_SYS_MmzAlloc

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_SYS_MmzFlushCache

HI_MPI_SYS_MmzFlushCache

【描述】

清空 cache 里的内容到内存并且使 cache 里的内容无效。

【语法】

```
HI_S32 HI_MPI_SYS_MmzFlushCache(HI_U32 u32PhyAddr, HI_VOID *pVitAddr,
HI_U32 u32Size);
```

【参数】

参数名称	描述	输入/输出
u32PhyAddr	待操作数据的起始物理地址。	输入
pVirtAddr	待操作数据的起始虚拟地址指针。	输入
u32Size	待操作数据的大小。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

- 当 cache 里的数据为最新数据时，为了保证不能直接访问 cache 的硬件(如 ive)在访问内存时能够得到正确的数据，此时需要先调用此接口将 cache 里的内容更新到内存，这样，当硬件访问内存时，保证了数据的一致性和正确性。
- 此接口应与 HI_MPI_SYS_MmzAlloc_Cached 接口配套使用。
- 若将 u32PhyAddr 设为 0，则表示操作整个 cache 区域。

【举例】

无。

【相关主题】

HI_MPI_SYS_MmzAlloc_Cached

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_SYS_MmzFree

【描述】

在用户态释放 MMZ 内存。

【语法】

```
HI_S32 HI_MPI_SYS_MmzFree(HI_U32 u32PhyAddr, HI_VOID *pVirtAddr);
```

【参数】

参数名称	描述	输入/输出
u32PhyAddr	物理地址。	输入
pVirtAddr	虚拟地址指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

输入的地址必须为有效的物理地址，虚拟地址指针可以置为 NULL。

【举例】

无。

【相关主题】

HI_MPI_SYS_MmzAlloc

HI_MPI_SYS_SetMemConf

【描述】

配置内存参数，设置模块设备通道使用内存的 DDR 名。

【语法】

```
HI_S32 HI_MPI_SYS_SetMemConf(MPP_CHN_S *pstMppChn,const HI_CHAR
*pcMmzName);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
pstMppChn	模块设备通道信息。	输入
pcMmzName	内存所在 DDR 的名字。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

- 输入的 DDR 名字必须是已经存在的 DDR 名。例如，有两块 DDR，其中一块没有命名（为无名 NULL），另一块名为 DDR1，则输入 pcMmzName 必须为 NULL 或“DDR1”，该函数常用于均分内存。
- 对 VO 模块做均分内存策略时，用户须保证同一设备号下的通道内存在同一个 DDR 上。如果不一致，SDK 默认从该设备的通道 0 设置的 DDR 分配内存。
- 此接口必须在业务正常运转之前调用。即在 sys 和 vb 初始化后，如有需要，则先调用此接口，再配置其他模块。

【举例】

```
HI_S32  i, s32Ret;
HI_CHAR   *pcMmzName;
MPP_CHN_S stMppChnVI;
/* 为16D1均分内存 */
    for(i=0;i<16;i++)
    {
        stMppChnVI.enModId = HI_ID_VIU;
        stMppChnVI.s32DevId = 0;
        stMppChnVI.s32ChnId = i;
        if(0 == (i%2))
        {
            pcMmzName = NULL;
        }
        else
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    {
        pcMmzName = "ddr1";
    }
    s32Ret = HI_MPI_SYS_SetMemConf(&stMppChnVI,pcMmzName);
    if (HI_SUCCESS != s32ret)
    {
        printf("SetMemConf ERR !\n");
        return s32ret;
    }
}
```

【相关主题】

HI_MPI_SYS_GetMemConf

HI_MPI_SYS_GetMemConf

【描述】

获取模块设备通道使用内存的 DDR 名。

【语法】

```
HI_S32 HI_MPI_SYS_GetMemConf(MPP_CHN_S *pstMppChn, HI_CHAR *pcMmzName);
```

【参数】

参数名称	描述	输入/输出
pstMppChn	模块设备通道信息。	输入
pcMmzName	内存所在 DDR 的名字。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

无。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

HI_MPI_SYS_SetMemConf

HI_MPI_SYS_CloseFd

【描述】

关闭所有 SYS 打开的日志、系统、内存 Fd。

【语法】

```
HI_S32 HI_MPI_SYS_CloseFd(HI_VOID);
```

【参数】

无.

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

该接口必须在所有模块都禁用后使用。

【举例】

无。

【相关主题】

无。

HI_MPI_VB_SetConf

【描述】

设置 MPP 视频缓存池属性。

【语法】

```
HI_S32 HI_MPI_VB_SetConf (const VB_CONF_S *pstVbConf);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstVbConf	视频缓存池属性指针。 静态属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 只能在系统处于未初始化的状态下，才可以设置缓存池属性，否则会返回失败。
- video buf 根据不同的应用场景需要不同的配置。配置规则参见“2.2.1 视频缓存池”

【举例】

```
HI_S32 s32ret;
VB_CONF_S stVbConf;
memset(&stVbConf,0,sizeof(VB_CONF_S));
stVbConf.u32MaxPoolCnt = 128;
stVbConf.astCommPool[0].u32BlkSize = 768*576*2;
stVbConf.astCommPool[0].u32BlkCnt = 20;
stVbConf.astCommPool[1].u32BlkSize = 384*288*2;
stVbConf.astCommPool[1].u32BlkCnt = 40;
s32ret = HI_MPI_VB_SetConf(&stVbConf);
if (HI_SUCCESS != s32ret)
{
    printf("set vb err:0x%x\n", s32ret);
    return s32ret;
}

s32ret = HI_MPI_VB_Init();
if (HI_SUCCESS != s32ret)
{
    printf("init vb err:0x%x\n", s32ret);
    return s32ret;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}

/* ... ... */

(void)HI_MPI_VB_Exit();
```

【相关主题】

HI_MPI_VB_GetConf

HI_MPI_VB_GetConf

【描述】

获取 MPP 视频缓存池属性。

【语法】

```
HI_S32 HI_MPI_VB_GetConf (VB_CONF_S *pstVbConf);
```

【参数】

参数名称	描述	输入/输出
pstVbConf	视频缓存池属性指针。 静态属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

必须先调用 HI_MPI_VB_SetConf 设置 MPP 视频缓存池属性，再获取属性。

【举例】

无。

【相关主题】

HI_MPI_VB_SetConf

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VB_Init

【描述】

初始化 MPP 视频缓存池。

【语法】

```
HI_S32 HI_MPI_VB_Init (HI_VOID);
```

【参数】

无。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 必须先调用 HI_MPI_VB_SetConf 配置缓存池属性，再初始化缓存池，否则会失败。
- 可反复初始化，不返回失败。

【举例】

请参见 HI_MPI_VB_SetConf 的举例。

【相关主题】

HI_MPI_VB_Exit

HI_MPI_VB_Exit

【描述】

去初始化 MPP 视频缓存池。

【语法】

```
HI_S32 HI_MPI_VB_Exit (HI_VOID);
```

【参数】

无。

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 必须先调用 HI_MPI_SYS_Exit 去初始化 MPP 系统，再去初始化缓存池，否则返回失败。
- 可以反复去初始化，不返回失败。
- 去初始化不会清除先前对缓存池的配置。

【举例】

请参见 HI_MPI_VB_SetConf 的举例。

【相关主题】

HI_MPI_VB_Init

HI_MPI_VB_CreatePool

【描述】

创建一个视频缓存池。

【语法】

```
VB_POOL HI_MPI_VB_CreatePool(HI_U32 u32BlkSize,HI_U32 u32BlkCnt, const HI_CHAR *pcMmzName);
```

【参数】

参数名称	描述	输入/输出
u32BlkSize	缓存池中每个缓存块的大小。 取值范围：（0, 2^{32}），以 byte 为单位。	输入
u32BlkCnt	缓存池中缓存块的个数。 取值范围：（0, 2^{32}）。	输入
pcMmzName	缓存池所在 DDR 的名字。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 VB_INVALID_POOLID	有效的缓存池 ID 号。
VB_INVALID_POOLID	创建缓存池失败，可能是参数非法或者保留内存不够。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 该缓存池是从保留内存中分配，一个缓存池包含若干个大小相同的缓存块。如果该缓存池的大小超过了保留内存中的空闲空间，则创建缓存池会失败。
- 用户需保证输入的 DDR 名字已经存在，如果输入不存在 DDR 的名字，会造成分不到内存。

【举例】

```
VB_POOL VbPool;
VB_BLK VbBlk;
HI_U32 u32BlkSize = 768*576*2;
HI_U32 u32BlkCnt  = 15;
HI_U32 u32Addr;

/* create a video buffer pool*/
VbPool = HI_MPI_VB_CreatePool(u32BlkSize,u32BlkCnt,"ddr1");
if ( VB_INVALID_POOLID == VbPool )
{
    printf("create vb err\n");
    return HI_FAILURE;
}

/* get a buffer block from pool*/
VbBlk = HI_MPI_VB_GetBlock(VbPool, u32BlkSize,"ddr1");
if (VB_INVALID_HANDLE == VbBlk )
{
    printf("get vb block err\n");
    (void)HI_MPI_VB_DestroyPool(VbPool);
    return HI_FAILURE;
}

/* get the physical address of buffer block*/
u32Addr = HI_MPI_VB_Handle2PhysAddr(VbBlk);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
if (HI_NULL_PTR == u32Addr)
{
    printf("blk to physaddr err\n");
    (void)HI_MPI_VB_ReleaseBlock(VbBlk);
    (void)HI_MPI_VB_DestroyPool(VbPool);
    return HI_FAILURE;
}
/* use this address do something ...*/
/* then release the buffer block*/
(void)HI_MPI_VB_ReleaseBlock(VbBlk);
/* destroy video buffer pool */
(void)HI_MPI_VB_DestroyPool(VbPool);
```

【相关主题】

HI_MPI_VB_DestroyPool

HI_MPI_VB_DestroyPool

【描述】

销毁一个视频缓存池。

【语法】

```
HI_S32 HI_MPI_VB_DestroyPool(VB_POOL Pool);
```

【参数】

参数名称	描述	输入/输出
Pool	缓存池 ID 号。 取值范围：[0, VB_MAX_POOLS)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 销毁一个不存在的缓存池，则返回 HI_ERR_VB_UNEXIST。
- 在去初始化 MPP 缓存池时，所有的缓存池都将被销毁，包括用户态的缓存池。

【举例】

请参见 HI_MPI_VB_CreatePool 的举例。

【相关主题】

HI_MPI_VB_CreatePool

HI_MPI_VB_GetBlock

【描述】

用户态获取一个缓存块。

【语法】

```
VB_BLK HI_MPI_VB_GetBlock(VB_POOL Pool, HI_U32 u32BlkSize,const HI_CHAR *pcMmzName);
```

【参数】

参数名称	描述	输入/输出
Pool	缓存池 ID 号。 取值范围：[0, VB_MAX_POOLS)。	输入
u32BlkSize	缓存块大小。 取值范围：(0, 2^{32})，以 byte 为单位。	输入
pcMmzName	缓存池所在 DDR 的名字。	输入

【返回值】

返回值	描述
非 VB_INVALID_HANDLE	有效的缓存块句柄。
VB_INVALID_HANDLE	获取缓存块失败。

【错误码】

无。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 用户可以在创建一个缓存池之后，调用本接口从该缓存池中获取一个缓存块；即将第 1 个参数 Pool 设置为创建的缓存池 ID（第 2 个参数 u32BlkSize 不用设置）；获取的缓存块的大小即为用户创建缓存池时指定的缓存块大小。从指定缓存池获取缓存块时，参数 pcMmzName 无效。
- 如果用户需要从任意一个公共缓存池中获取一块指定大小的缓存块，则可以将第 1 个参数 Pool 设置为无效 ID 号（VB_INVALID_POOLID），将第 2 个参数 u32BlkSize 设置为需要的缓存块大小，并指定要从哪个 DDR 上的公共缓存池获取缓存块。如果指定的 DDR 上并没有公共缓存池，那么将获取不到缓存块。
- 公共缓存池主要用来存放 VIU 的捕获图像，因此，对公共缓存池的不当操作（如占用过多的缓存块）会影响 MPP 系统的正常运行。

【举例】

请参见 HI_MPI_VB_CreatePool 的举例。

【相关主题】

HI_MPI_VB_ReleaseBlock

HI_MPI_VB_ReleaseBlock

【描述】

用户态释放一个已经获取的缓存块。

【语法】

```
HI_S32 HI_MPI_VB_ReleaseBlock(VB_BLK Block);
```

【参数】

参数名称	描述	输入/输出
Block	缓存块句柄。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

获取的缓存块使用完后，应该调用此接口释放缓存块。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

请参见 HI_MPI_VB_CreatePool 的举例。

【相关主题】

HI_MPI_VB_GetBlock

HI_MPI_VB_Handle2PhysAddr

【描述】

用户态获取一个缓存块的物理地址。

【语法】

```
HI_U32 HI_MPI_VB_Handle2PhysAddr(VB_BLK Block);
```

【参数】

参数名称	描述	输入/输出
Block	缓存块句柄。	输入

【返回值】

返回值	描述
0	无效返回值，缓存块句柄非法。
非 0	有效物理地址。

【错误码】

无。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

指定的缓存块应该是从 MPP 视频缓存池中获取的有效缓存块。

【举例】

请参见 HI_MPI_VB_CreatePool 的举例。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VB_Handle2PoolId

【描述】

用户态获取一个帧缓存块所在缓存池的 ID。

【语法】

```
VB_POOL HI_MPI_VB_Handle2PoolId (VB_BLK Block);
```

【参数】

参数名称	描述	输入/输出
Block	缓存块句柄。	输入

【返回值】

返回值	描述
非负数	有效的缓存池 ID 号。
负数	无效的缓存池 ID 号。

【错误码】

无。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

指定的缓存块应该是从 MPP 视频缓存池中获取的有效缓存块。

【举例】

```
VB_POOL VbPool;
VB_BLK VbBlk;/* get vb blk id from somewhere*/
/* get pool id */
VbPool = HI_MPI_VB_Handle2PoolId(VbBlk);
if ( VB_INVALID_POOLID != VbPool )
{
    printf("pool id is %d\n", VbPool);
    /* use pool id do something ...*/
}
```

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

HI_MPI_VB_MmapPool

【描述】

为一个视频缓存池映射用户态虚拟地址。

【语法】

```
HI_S32 HI_MPI_VB_MmapPool(VB_POOL Pool);
```

【参数】

参数名称	描述	输入/输出
Pool	缓存池 ID 号。 取值范围：[0, VB_MAX_POOLS)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 必须输入合法的缓存池 ID。
- 重复映射视为成功。

【举例】

无。

【相关主题】

- HI_MPI_VB_MunmapPool
- HI_MPI_VB_GetBlkVirAddr

HI_MPI_VB_MunmapPool

【描述】

为一个视频缓存池解除用户态映射。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VB_MunmapPool(VB_POOL Pool);
```

【参数】

参数名称	描述	输入/输出
Pool	缓存池 ID 号。 取值范围：[0, VB_MAX_POOLS)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 必须输入合法的缓存池 ID。
- 视频缓存池必须已经映射过。
- 缓存池中的缓存块没有被 MPI 层占用，如果被 MPI 层占用，视为正在使用映射的用户态虚拟地址，返回失败。

【举例】

无。

【相关主题】

- HI_MPI_VB_MmapPool
- HI_MPI_VB_GetBlkVirAddr

HI_MPI_VB_GetBlkVirAddr

【描述】

获取一个视频缓存池中的缓存块的用户态虚拟地址。

【语法】

```
HI_S32 HI_MPI_VB_GetBlkVirAddr(VB_POOL Pool, HI_U32 u32PhyAddr, HI_VOID
**ppVirAddr);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
Pool	缓存池 ID 号。 取值范围：[0, VB_MAX_POOLS)。	输入
u32PhyAddr	缓存块的物理地址。	输入
ppVirAddr	用户态虚拟地址。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_vb.h、mpi_vb.h
- 库文件：libmpi.a

【注意】

- 必须输入合法的缓存池 ID、合法的缓存块物理地址。
- 必须已调用 HI_MPI_VB_MmapPool。

【举例】

无。

【相关主题】

- HI_MPI_VB_MmapPool
- HI_MPI_VB_MunmapPool

HI_MPI_VB_ GetVRegAddr

【描述】

获取 ISP 寄存器的物理地址。

【语法】

```
HI_S32 HI_MPI_SYS_GetVRegAddr(HI_U32 *pu32Addr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pu32Addr	物理地址指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值参见错误码。

【需求】

- 头文件：hi_comm_sys.h、mpi_sys.h
- 库文件：libmpi.a

【注意】

此接口仅为 SDK 内部使用。

【举例】

无。

【相关主题】

无。

2.4 数据类型

2.4.1 基本数据类型

基本数据类型定义如下：

公共数据类型

```
typedef unsigned char           HI_U8;
typedef unsigned short          HI_U16;
typedef unsigned int            HI_U32;
typedef signed char             HI_S8;
typedef short                   HI_S16;
typedef int                     HI_S32;
#ifndef _M_IX86
    typedef unsigned long long  HI_U64;
    typedef long long           HI_S64;
#else
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    typedef __int64             HI_U64;
    typedef __int64             HI_S64;
#endif
typedef char                    HI_CHAR;
#define HI_VOID                 void
/*----------------------------------------------*
 * const defination                             *
 *----------------------------------------------*/
typedef enum {
    HI_FALSE = 0,
    HI_TRUE  = 1,
} HI_BOOL;

#ifndef NULL
    #define NULL    0L
#endif

#define HI_NULL     0L
#define HI_SUCCESS  0
#define HI_FAILURE  (-1)

typedef HI_S32 AI_CHN;
typedef HI_S32 AO_CHN;
typedef HI_S32 AENC_CHN;
typedef HI_S32 ADEC_CHN;
typedef HI_S32 AUDIO_DEV;
typedef HI_S32 AVENC_CHN;
typedef HI_S32 VI_DEV;
typedef HI_S32 VI_WAY;
typedef HI_S32 VI_CHN;
typedef HI_S32 VO_DEV;
typedef HI_S32 VO_CHN;
typedef HI_S32 VENC_CHN;
typedef HI_S32 VDEC_CHN;
typedef HI_S32 VENC_GRP;
typedef HI_S32 VO_GRP;
typedef HI_S32 VDA_CHN;
/* 无效的通道号、无效的WAY号、无效的设备号 、无效的句柄号*/
#define HI_INVALID_CHN (-1)
#define HI_INVALID_WAY (-1)
#define HI_INVALID_DEV (-1)
#define HI_INVALID_HANDLE (-1)
/* 最大的视频缓存池个数 */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
#define VB_MAX_POOLS                256
/* 最大的公共视频缓存池个数 */
#define VB_MAX_COMM_POOLS           16
/* MMZ名字长度 */
#define MAX_MMZ_NAME_LEN            16
```

Hi3531 的基本数据类型

```
/* 硬件视频输入单元设备最大个数*/
#define VIU_MAX_DEV_NUM             8

/* 每个硬件视频输入设备支持的最大通道数目*/
#define VIU_MAX_CHN_NUM_PER_DEV     4

/* 视频输入通道总的最大个数(软件通道和硬件通道)*/
#define VIU_MAX_CHN_NUM             34
#define VIU_MAX_PHYCHN_NUM          32

/* VO设备的最大个数 */
#define VO_MAX_DEV_NUM              14

/* VO视频层的最大个数 */
#define VO_MAX_LAYER_NUM                14

/* VO通道的最大个数 */
#define VO_MAX_CHN_NUM              64

/* VO设备支持视频层的最大个数*/
#define VO_MAX_LAYER_IN_DEV     2

/* VO级联设备的最大个数 */
#define VO_MAX_CAS_DEV_NUM      2

/* 定义VO级联设备1宏，设备号为8 */
#define VO_CAS_DEV_1            8
/* 定义VO级联设备2宏，设备号为9 */
#define VO_CAS_DEV_2            9

/* VO级联样式最大个数 */
#define VO_CAS_MAX_PAT          128

/* VO级联32区域最大个数 */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
#define VO_CAS_MAX_POS_32RGN    32
/* VO级联64区域最大个数 */
#define VO_CAS_MAX_POS_64RGN    64

/* VO最大虚拟设备个数 */
#define VO_MAX_VIRT_DEV_NUM     4

/* 定义VO虚拟设备0宏，设备号为10 */
#define VO_VIRT_DEV_0           10
/* 定义VO虚拟设备1宏，设备号为11 */
#define VO_VIRT_DEV_1           11
/* 定义VO虚拟设备2宏，设备号为12 */
#define VO_VIRT_DEV_2           12
/* 定义VO虚拟设备3宏，设备号为13 */
#define VO_VIRT_DEV_3           13

/* VO最大同步组数 */
#define VO_SYNC_MAX_GRP             16

/* VO同步组容纳的最多通道数 */
#define VO_SYNC_MAX_CHN             32

/* VO最小显示buffer数 */
#define VO_MIN_DISP_BUF             5

/* VO最大显示buffer数 */
#define VO_MAX_DISP_BUF             15

/* VO最小虚拟设备buffer数*/
#define VO_MIN_VIRT_BUF             3

/* VO最大虚拟设备buffer数*/
#define VO_MAX_VIRT_BUF             15

/* 最大的编码组个数 */
#define VENC_MAX_GRP_NUM            64

/* 最大的编码通道个数 */
#define VENC_MAX_CHN_NUM            64

/* 最大的解码通道个数 */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
#define VDEC_MAX_CHN_NUM            32

/* SIO的个数，即音频设备的个数 */
#define SIO_MAX_NUM                 6

/* 最大的AIO通道数 */
#define AIO_MAX_CHN_NUM             16

/* 最大的音频编码通道数 */
#define AENC_MAX_CHN_NUM            32

/* 最大的音频解码通道数 */
#define ADEC_MAX_CHN_NUM            32

/*  最大音频帧缓存数 */
#define MAX_AUDIO_FRAME_NUM         50
```

Hi3521 的基本数据类型

```
/* 硬件视频输入单元设备最大个数*/
#define VIU_MAX_DEV_NUM             4

/* 每个硬件视频输入设备支持的最大通道数目*/
#define VIU_MAX_CHN_NUM_PER_DEV     4

/* 视频输入通道总的最大个数*/
#define VIU_MAX_PHYCHN_NUM          16

/* VO设备的最大个数 */
#define VO_MAX_DEV_NUM              7

/* VO视频层的最大个数 */
#define VO_MAX_LAYER_NUM            7

/* VO通道的最大个数 */
#define VO_MAX_CHN_NUM              32

/* VO设备支持视频层的最大个数*/
#define VO_MAX_LAYER_IN_DEV         2

/* VO最大虚拟设备个数 */
#define VO_MAX_VIRT_DEV_NUM     4
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
/* 定义VO虚拟设备0宏，设备号为3 */
#define VO_VIRT_DEV_0          3
/* 定义VO虚拟设备1宏，设备号为4 */
#define VO_VIRT_DEV_1          4
/* 定义VO虚拟设备2宏，设备号为5 */
#define VO_VIRT_DEV_2          5
/* 定义VO虚拟设备3宏，设备号为6 */
#define VO_VIRT_DEV_3          6
/* 最大的编码组个数 */
#define VENC_MAX_GRP_NUM           64

/* 最大的编码通道个数 */
#define VENC_MAX_CHN_NUM           64

/* 最大的解码通道个数 */
#define VDEC_MAX_CHN_NUM           64

/* SIO的个数，即音频设备的个数 */
#define SIO_MAX_NUM                4

/* 最大的AIO通道数 */
#define AIO_MAX_CHN_NUM            16

/* 最大的音频编码通道数 */
#define AENC_MAX_CHN_NUM           32

/* 最大的音频解码通道数 */
#define ADEC_MAX_CHN_NUM           32

/*  最大音频帧缓存数 */
#define MAX_AUDIO_FRAME_NUM        50
```

除了上述基本数据类型外，其他基本数据类型定义如下：

- POINT_S：定义坐标信息结构体。
- SIZE_S：定义尺寸信息结构体。
- RECT_S：定义矩形区域信息结构体。
- MOD_ID_E：定义模块 ID 枚举类型。
- MPP_CHN_S：定义模块设备通道结构体。
- PAYLOAD_TYPE_E：定义音视频净荷类型枚举。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

POINT_S

【说明】

定义坐标信息结构体。

【定义】

```
typedef struct hiPOINT_S
{
    HI_S32  s32X;
    HI_S32  s32Y;
}POINT_S;
```

【成员】

成员名称	描述
s32X	横坐标。
s32Y	纵坐标。

【注意事项】

无。

【相关数据类型及接口】

REGION_CTRL_PARAM_U

SIZE_S

【说明】

定义大小信息结构体。

【定义】

```
typedef struct hiSIZE_S
{
    HI_U32 u32Width;
    HI_U32 u32Height;
} SIZE_S;
```

【成员】

成员名称	描述
u32Width	宽度。
u32Height	高度。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

无。

RECT_S

【说明】

定义矩形区域信息结构体。

【定义】

```
typedef struct hiRECT_S
{
    HI_S32  s32X;
    HI_S32  s32Y;
    HI_U32  u32Width;
    HI_U32  u32Height;
}RECT_S;
```

【成员】

成员名称	描述
s32X	横坐标。
s32Y	纵坐标。
u32Width	宽度。
u32Height	高度。

【注意事项】

无。

【相关数据类型及接口】

- REGION_ATTR_S
- OVERLAY_ATTR_S
- VI_CHM_ATTR_S
- VI_PUB_ATTR_S
- VO_CHN_ATTR_S

MOD_ID_E

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义模块 ID 枚举类型。

【定义】

```
typedef enum hiMOD_ID_E
{
    HI_ID_CMPI    = 0,
    HI_ID_VB      = 1,
    HI_ID_SYS     = 2,
    HI_ID_VALG    = 3,

    HI_ID_CHNL    = 4,
    HI_ID_GROUP   = 5,
    HI_ID_VENC    = 6,
    HI_ID_VPSS    = 7,

    HI_ID_VDA     = 8,

    HI_ID_H264E   = 9,
    HI_ID_JPEGE   = 10,
    HI_ID_MPEG4E  = 11,

    HI_ID_VDEC    = 12,
    HI_ID_H264D   = 13,
    HI_ID_JPEGD   = 14,
    HI_ID_VOU     = 15,

    HI_ID_VIU     = 16,
    HI_ID_DSU     = 17,
    HI_ID_RGN     = 18,
    HI_ID_RC      = 19,


    HI_ID_SIO     = 20,
    HI_ID_AI      = 21,
    HI_ID_AO      = 22,
    HI_ID_AENC    = 23,
    HI_ID_ADEC    = 24,

    HI_ID_AVENC   = 25,

    HI_ID_PCIV    = 26,
    HI_ID_PCIVFMW = 27,

    /* there is a hole */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_ID_DCCM    = 31,
    HI_ID_DCCS    = 32,

    HI_ID_PROC    = 33,
    HI_ID_LOG     = 34,
    HI_ID_MST_LOG = 35,
    HI_ID_VD      = 36,
   /* there is a hole */

   HI_ID_VCMP    = 38,
   HI_ID_FB      = 39,

    HI_ID_USR     = 40,

    HI_ID_BUTT,
} MOD_ID_E;
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

MPP_CHN_S

MPP_CHN_S

【说明】

定义模块设备通道结构体。

【定义】

```
typedef struct hiMPP_CHN_S
{
    MOD_ID_E    enModId;
    HI_S32      s32DevId;
    HI_S32      s32ChnId;
} MPP_CHN_S;
```

【成员】

成员名称	描述
enModId	模块号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
s32DevId	设备号。
s32ChnId	通道号。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_SYS_Bind
- HI_MPI_SYS_UnBind
- HI_MPI_SYS_GetBindbyDest

PAYLOAD_TYPE_E

【说明】

定义音视频净荷类型枚举。

【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,
    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,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    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_AACLC = 42,
    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_MAX = 257,

    PT_AMR = 1001, /* add by mpp */
    PT_MJPEG = 1002,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}PAYLOAD_TYPE_E;
```

【成员】

略。

【注意事项】

无。

【相关数据类型及接口】

- VENC_CHN_ATTR_S
- VDEC_CHN_ATTR_S
- AENC_CHN_ATTR_S
- ADEC_CHN_ATTR_S

2.4.2 系统控制数据类型

系统控制相关数据类型定义如下：

- MPP_SYS_CONF_S：定义 MPP 系统控制属性结构体。
- VB_CONF_S：定义视频缓存池属性结构体。
- MPP_VERSION_S：定义 MPP 版本描述结构体。

MPP_SYS_CONF_S

【说明】

定义 MPP 系统控制属性结构体。

【定义】

```
typedef struct hiMPP_SYS_CONF_S
{
    /* stride of picture buffer must be aligned with this value.
     * you can choose a value from 1 to 1024,
     * and it except 1 must be multiple of 16.*/
    HI_U32  u32AlignWidth;
}MPP_SYS_CONF_S;
```

【成员】

成员名称	描述
u32AlignWidth	整个系统中使用图像的 stride 字节对齐数。 取值范围：[1, 1024]，一般选择 16 字节对齐。 静态属性。

【注意事项】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关数据类型及接口】

HI_MPI_SYS_SetConf

VB_CONF_S

【说明】

定义视频缓存池属性结构体。

【定义】

```
typedef struct hiVB_CONF_S
{
    HI_U32  u32MaxPoolCnt;  /* max count of pools, (0,VB_MAX_POOLS] */
    Struct  hiVB_CPOOL_S
    {
        HI_U32  u32BlkSize;
        HI_U32  u32BlkCnt;
        HI_CHAR acMmzName[MAX_MMZ_NAME_LEN];
    }astCommPool[VB_MAX_COMM_POOLS];
} VB_CONF_S;
```

【成员】

成员名称	描述
u32MaxPoolCnt	整个系统中可容纳的缓存池个数。 取值范围：(0, VB_MAX_POOLS]。 静态属性。 目前，固定取值 VB_MAX_POOLS。
astCommPool	公共缓存池属性结构体，成员包括公共缓存池中每个缓存块的大小（以 byte 为单位）和缓存块的个数及此缓存池在所在的 MMZ 的名字。 静态属性。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_SYS_SetConf

MPP_VERSION_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义 MPI 版本描述结构体。

【定义】

```
typedef struct hiMPP_VERSION_S
{
    HI_CHAR aVersion[64];
}MPP_VERSION_S;
```

【成员】

成员名称	描述
aVersion	版本描述字符串。 比如“HI_VERSION=Hi35xx_MPP_V1.0.4.0”。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_SYS_GetVersion

2.4.3 视频公共类型

视频公共数据类型定义如下：

- VIDEO_NORM_E：定义视频输入制式类型。
- PIXEL_FORMAT_E：定义像素格式类型。
- VIDEO_FIELD_E：定义视频图像帧场类型。
- VIDEO_FRAME_S：定义视频原始图像帧结构体。
- VIDEO_FRAME_INFO_S：定义视频图像帧信息结构体。
- BITMAP_S：定义位图图像信息结构。
- VIDEO_VBI_INFO_S：定义视频 VBI 信息结构体。

VIDEO_NORM_E

【说明】

定义视频输入制式类型。

【定义】

```
typedef enum hiVIDEO_NORM_E
{
    VIDEO_ENCODING_MODE_PAL=0,
    VIDEO_ENCODING_MODE_NTSC,
    VIDEO_ENCODING_MODE_AUTO,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VIDEO_ENCODING_MODE_BUTT,
} VIDEO_NORM_E;
```

【成员】

成员名称	描述
VIDEO_ENCODING_MODE_PAL	PAL 制式。
VIDEO_ENCODING_MODE_NTSC	NTSC 制式。
VIDEO_ENCODING_MODE_AUTO	自动识别制式。

【注意事项】

自动识别制式目前暂不支持。

【相关数据类型及接口】

VI_PUB_ATTR_S

PIXEL_FORMAT_E

【说明】

定义像素格式类型。

【定义】

```
typedef enum hiPIXEL_FORMAT_E
{
    PIXEL_FORMAT_RGB_1BPP = 0,
    PIXEL_FORMAT_RGB_2BPP,
    PIXEL_FORMAT_RGB_4BPP,
    PIXEL_FORMAT_RGB_8BPP,
    PIXEL_FORMAT_RGB_444,
    PIXEL_FORMAT_RGB_4444,
    PIXEL_FORMAT_RGB_555,
    PIXEL_FORMAT_RGB_565,
    PIXEL_FORMAT_RGB_1555,

    PIXEL_FORMAT_RGB_888,
    PIXEL_FORMAT_RGB_8888,
    PIXEL_FORMAT_RGB_PLANAR_888,
    PIXEL_FORMAT_RGB_BAYER,

    PIXEL_FORMAT_YUV_A422,
    PIXEL_FORMAT_YUV_A444,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    PIXEL_FORMAT_YUV_PLANAR_422,
    PIXEL_FORMAT_YUV_PLANAR_420,
    PIXEL_FORMAT_YUV_PLANAR_444,
    PIXEL_FORMAT_YUV_SEMIPLANAR_422,
    PIXEL_FORMAT_YUV_SEMIPLANAR_420,
    PIXEL_FORMAT_YUV_SEMIPLANAR_444,

    PIXEL_FORMAT_UYVY_PACKAGE_422,
    PIXEL_FORMAT_YUYV_PACKAGE_422,
    PIXEL_FORMAT_VYUY_PACKAGE_422,
    PIXEL_FORMAT_YCbCr_PLANAR,

    PIXEL_FORMAT_BUTT
} PIXEL_FORMAT_E;
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

- VI_CHN_ATTR_S
- OVERLAY_ATTR_S

VIDEO_FIELD_E

【说明】

定义视频图像帧场类型。

【定义】

```
typedef enum hiVIDEO_FIELD_E
{
    VIDEO_FIELD_TOP        = 0x01,    /* even field */
    VIDEO_FIELD_BOTTOM     = 0x02,    /* odd field */
    VIDEO_FIELD_INTERLACED  = 0x03,    /* two interlaced fields */
    VIDEO_FIELD_FRAME      = 0x04,    /* frame */
    VIDEO_FIELD_BUTT
} VIDEO_FIELD_E;
```

【成员】

成员名称	描述
VIDEO_FIELD_TOP	顶场类型。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VIDEO_FIELD_BOTTOM	底场类型。
VIDEO_FIELD_INTERLACED	两场间插类型。
VIDEO_FIELD_FRAME	帧类型。

【注意事项】

无。

【相关数据类型及接口】

VIDEO_FRAME_S

VIDEO_FRAME_S

【说明】

定义视频原始图像帧结构。

【定义】

```
typedef struct hiVIDEO_FRAME_S
{
    PIXEL_FORMAT_E  enPixelFormat;
    HI_U32          u32Width;
    HI_U32          u32Height;
    VIDEO_FIELD_E   u32Field;
    HI_U32  u32PhyAddr[3];
    HI_VOID *pVirAddr[3];
    HI_U32  u32Stride[3];
    HI_U16  u16OffsetTop;       /* top offset of show area */
    HI_U16  u16OffsetBottom;    /* bottom offset of show area */
    HI_U16  u16OffsetLeft;      /* left offset of show area */
    HI_U16  u16OffsetRight;     /* right offset of show area */
    HI_U64  u64pts;
    HI_U32  u32TimeRef;
    HI_U32  u32PrivateData;
}VIDEO_FRAME_S;
```

【成员】

成员名称	描述
enPixelFormat	视频图像像素格式。
u32Width	图像宽度。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32Height	图像高度。
u32Field	帧场模式。
u32PhyAddr	物理地址。
pVirAddr	虚拟地址。
u32Stride	图像跨距。
u16OffsetTop	图像顶部剪裁宽度。
u16OffsetBottom	图像底部剪裁宽度。
u16OffsetLeft	图像左侧剪裁宽度。
u16OffsetRight	图像右侧剪裁宽度。
u64pts	图像时间戳。
u32TimeRef	图像帧序列号。
u32PrivateData	私有数据。

【注意事项】

无。

【相关数据类型及接口】

VI_PUB_ATTR_S

VIDEO_FRAME_INFO_S

【说明】

定义视频图像帧信息结构体。

【定义】

```
typedef struct hiVIDEO_FRAME_INFO_S
{
    VIDEO_FRAME_S   stVFrame;
    HI_U32          u32PoolId;
}VIDEO_FRAME_INFO_S;
```

【成员】

成员名称	描述
stVFrame	视频图像帧。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32PoolId	视频缓存池 ID。

【注意事项】

无。

【相关数据类型及接口】

VIDEO_FRAME_S

BITMAP_S

【说明】

定义位图图像信息结构。

【定义】

```
typedef struct hiBITMAP_S
{
    PIXEL_FORMAT_E  enPixelFormat;
    HI_U32          u32Width;
    HI_U32          u32Height;
    HI_VOID         *pData;
} BITMAP_S;
```

【成员】

成员名称	描述
enPixelFormat	位图像素格式。
u32Width	位图宽度。
u32Height	位图高度。
pData	位图数据。

【注意事项】

无。

【相关数据类型及接口】

REGION_CTRL_PARAM_U

VIDEO_VBI_INFO_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义视频 VBI 信息结构体。

【定义】

```
typedef struct hiVIDEO_VBI_INFO_S
{
    HI_U32 au32Data[VIU_MAX_VBI_LEN];
    HI_U32 u32Len;
}VIDEO_VBI_INFO_S;
```

【成员】

成员名称	描述
au32Data	VBI 数据。
u32Len	VBI 数据长度。

【注意事项】

无。

2.5 错误码

2.5.1 系统控制错误码

系统控制 API 错误码如表 2-2 所示。

表2-2 系统控制 API 错误码

错误代码	宏定义	描述
0xA0028003	HI_ERR_SYS_ILLEGAL_PARAM	参数设置无效
0xA0028006	HI_ERR_SYS_NULL_PTR	空指针错误
0xA0028009	HI_ERR_SYS_NOT_PERM	操作不允许
0xA0028010	HI_ERR_SYS_NOTREADY	系统控制属性未配置
0xA0028012	HI_ERR_SYS_BUSY	系统忙
0xA002800C	HI_ERR_SYS_NOMEM	分配内存失败，如系统内存不足

2.5.2 视频缓存池错误码

视频缓存池 API 错误码如表 2-3 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表2-3 视频缓存池 API 错误码

错误代码	宏定义	描述
0xA0018003	HI_ERR_VB_ILLEGAL_PARAM	参数设置无效
0xA0018005	HI_ERR_VB_UNEXIST	视频缓存池不存在
0xA0018006	HI_ERR_VB_NULL_PTR	参数空指针错误
0xA0018009	HI_ERR_VB_NOT_PERM	操作不允许
0xA001800C	HI_ERR_VB_NOMEM	分配内存失败
0xA001800D	HI_ERR_VB_NOBUF	分配缓存失败
0xA0018010	HI_ERR_VB_NOTREADY	系统控制属性未配置
0xA0018012	HI_ERR_VB_BUSY	系统忙
0xA0018040	HI_ERR_VB_2MPOOLS	创建缓存池多

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

3 视频输入.....3-1
3.1 概述.....3-1
3.2 重要概念.....3-1
3.3 功能描述.....3-2
3.3.1 Hi3531/Hi3532.....3-2
3.3.2 Hi3521/Hi3520A.....3-15
3.3.3 Hi3520D/Hi3515A/Hi3515C.....3-18
3.3.4 Hi3518/Hi3516C.....3-21
3.4 API 参考.....3-22
3.5 数据类型.....3-95
3.6 错误码.....3-136

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 3-1 Hi3531/Hi3532 VI 硬件单元功能框图.....3-3
图 3-2 VI 通道单元功能框图.....3-5
图 3-3 简化概念（设备、Chn）.....3-7
图 3-4 完整概念（设备、Way、Chn）.....3-8
图 3-5 16 路 D1/960H 场景（4 路复合模式）.....3-9
图 3-6 16 路 D1/960H 场景（2 路复合模式）.....3-10
图 3-7 8 路 720P 或 8 路 D1/960H 场景（1 路复合模式）.....3-11
图 3-8 4 路 720P/1080P 场景（1 路复合模式）.....3-12
图 3-9 VI 级联数据传输图.....3-15
图 3-10 Hi3521/Hi3520A VI 硬件单元功能框图.....3-16
图 3-11 Hi3520D VI 硬件单元功能框图.....3-19
图 3-12 Hi3518/Hi3516C VI 硬件单元功能框图.....3-21
图 3-13 Hi3518/Hi3516C VI 通道功能框图.....3-21

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 3-1 奇偶设备功能对比.....3-4
表 3-2 VI 物理通道规格.....3-5
表 3-3 典型应用场景描述.....3-6
表 3-4 16 路 D1 或 960H 场景下的掩码配置（第 1 种方案）.....3-13
表 3-5 16 路 D1 或 960H 场景下的掩码配置（第 2 种方案）.....3-13
表 3-6 8 路 720p 场景下的掩码配置.....3-14
表 3-7 4 路 720p 或 4 路 1080p 场景下的掩码配置.....3-14
表 3-8 16 路图像输入场景下的掩码配置.....3-17
表 3-9 8 路 720p 场景下的掩码配置.....3-18
表 3-10 2 路 720p 或 2 路 1080p 场景下的掩码配置.....3-18
表 3-11 8 路图像输入场景下的掩码配置.....3-20
表 3-12 2 路 720p 场景下的掩码配置.....3-20
表 3-13 1 路 720p 或 1 路 1080p 场景下的掩码配置.....3-20
表 3-14 1 路 720p 场景下的掩码配置（12bit）.....3-22
表 3-15 1 路图像输入场景下的掩码配置（8bit）.....3-22
表 3-16 主属性和次属性的配置关系.....3-38
表 3-17 视频输入 API 错误码.....3-136

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

3 视频输入

3.1 概述

视频输入（VI）模块实现的功能：将芯片外的视频数据通过 ITU-R BT656/601/1120 接口或 Digital Camera 接口接收，存入到指定的内存区域。在此过程中，VI 可以对接收到的原始视频图像数据进行裁剪（Crop）、水平垂直缩小（Down Scaling）和水平垂直翻转（Mirror、Flip）等处理，并实现一路原始视频图像输入，输出一路或多路视频图像功能。

3.2 重要概念

- 视频输入设备

  视频输入设备支持若干种时序输入，负责对时序进行解析。

- 视频物理通道

  视频物理通道负责将输入设备解析后得到的视频数据输出到 DDR。在真正将数据输出到 DDR 之前，它可以实现裁剪、遮挡、水平垂直缩小和水平垂直翻转等功能，具体功能见各芯片的详细描述。

- 掩码

  掩码用于指示 VI 设备的视频数据来源。

- 级联

  级联指芯片的视频输出端通过 BT1120 时序将视频数据传送到另一颗芯片的视频输入端。

- 镜头畸变校正（LDC）

  镜头畸变校正，一些低端镜头容易产生图像畸变，需要根据畸变程度对其图像进行校正。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

3.3 功能描述

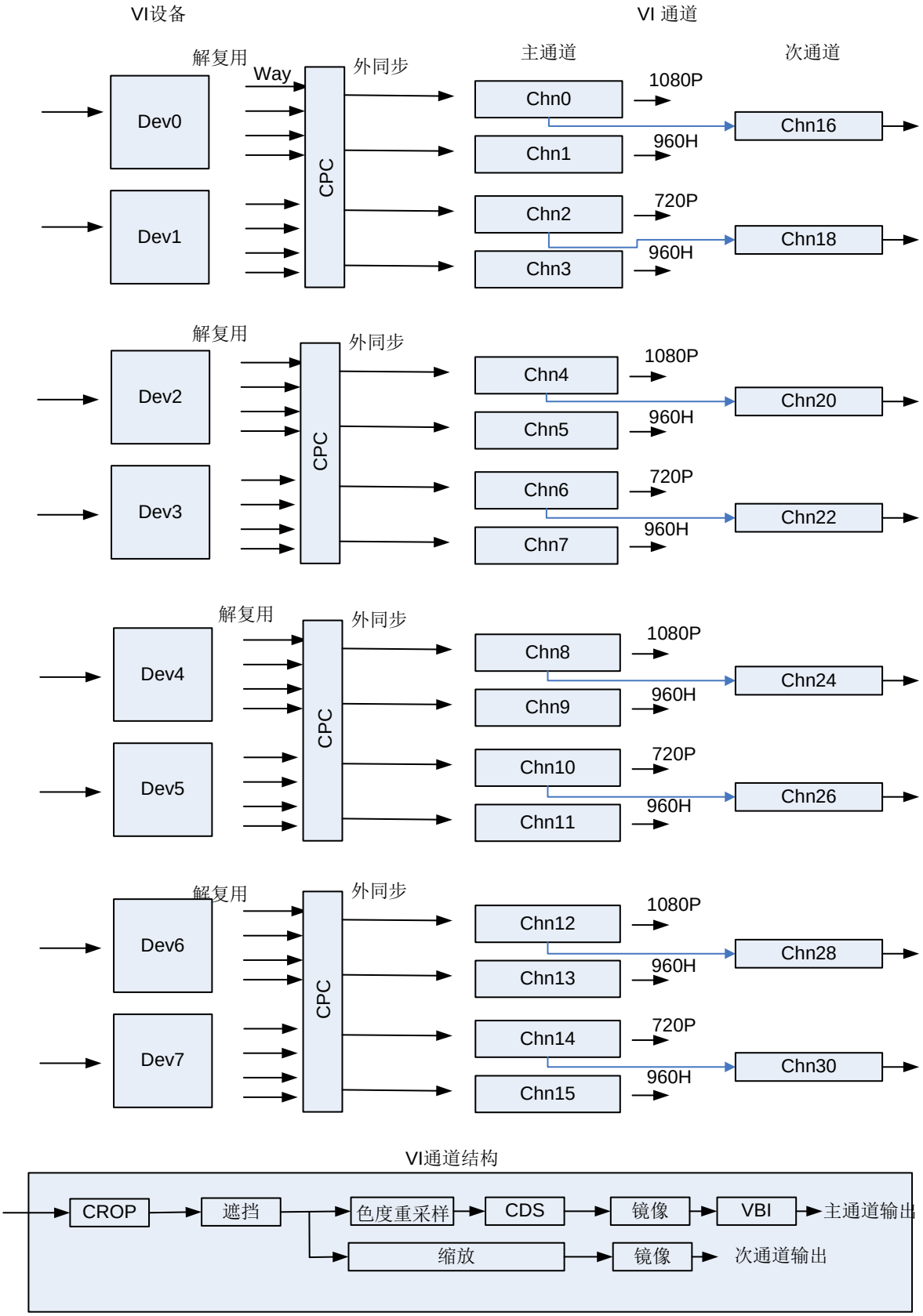
3.3.1 Hi3531/Hi3532

功能框图

Hi3531/Hi3532 视频输入硬件单元的功能如图 3-1 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-1 Hi3531/Hi3532 VI 硬件单元功能框图



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

视频输入设备

Hi3531 芯片有 4 个 BT.1120 接口（Hi3532 有 5 个），每个 BT.1120 接口依次对应两个 VI 设备（Hi3532 的第 5 个 BT.1120 口专用于级联透传，没有对应的 VI 设备），即第一个 BT.1120 口对应 VI 的 Dev0 和 Dev1，以此类推。其中，在 16 路 D1 和 16 路 960H 场景下，Hi3531 的 Dev2 可以接收第一个 BT.1120 口的低 8 位的视频数据，Dev6 可以接收第三个 BT.1120 口的低 8 位的视频数据，从而实现仅使用两个 BT.1120 接口就可以采集 16 路的视频数据。

8 个 VI 设备均支持 2 路 D1 复合模式输入（BT.656 协议），以及 1 路 720P interleave 模式。只有编号为 0/2/4/6 的 Dev 支持 4 路 D1 复合模式输入（BT.656 协议），且支持 1 路 720P/1080P 高清输入（BT.1120 协议），此时，编号为奇数 1/3/5/7 的 Dev 不可用。奇偶 Dev 的功能具体差异见表 3-1。

表3-1 奇偶设备功能对比

功能	0/2/4/6 端口	1/3/5/7 端口
最大图像输入能力	2560 x 1600 图象输入	720P 图像输入
4 路 D1/960H 复合	支持 4 路复用的 BT.656 时序 108MHz（4x4D1 输入场景）/144MHz （4x960H 输入场景）	不支持
2 路 D1/960H 复合	支持 2 路复用的 BT.656 时序 54 MHz（8x2D1 输入场景）/72MHz（8x960H 输入场景）	支持 2 路复用的 BT.656 时序 54 MHz（8x2D1 输入场景）/72MHz（8x960H 输入场景）
1 路 D1/960H 复合	支持 BT.656 时序 27MHz（8x1D1 输入场景）/36MHz（8x960H 输入场景）	支持 BT.656 时序 27MHz（8x1D1 输入场景）/36MHz（8x960H 输入场景）
对接的时序	支持 BT.656 时序(480I/576I) 支持 BT.601 时序 支持 SMPTE 296M(720P) 支持 SMPTE293M/ITU-R BT.1358 时序（480P/576P） 支持 SMPTE 274M/BT 1120 时序(1080i/1080P) 支持通用 DC 接口时序	支持 BT.656 时序(480I/576I) 支持 BT.601 时序 支持 SMPTE 296M(720P) 支持 SMPTE293M/ITU-R BT.1358 时序（480P/576P） 支持通用 DC 接口时序
Interleave 模式 8bit	支持 BT.1120 时序的 interleave 模式 148.5MHz（8x720P 输入场景）	支持 BT.1120 时序的 interleave 模式 148.5MHz（8x720P 输入场景）
对接的数据类型	YUV 数据	YUV 数据

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

视频物理通道

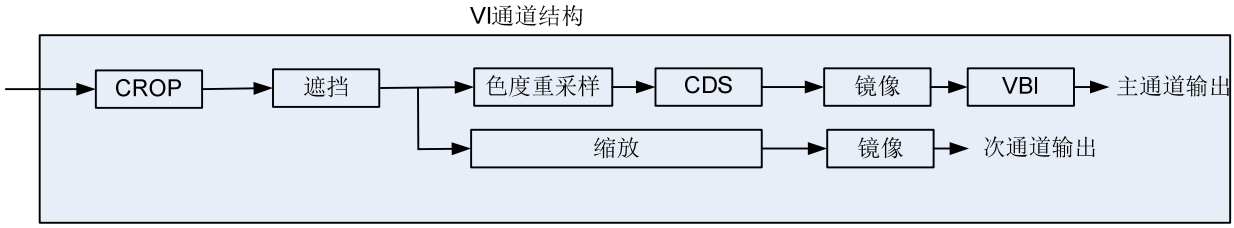
Hi3531/Hi3532VI 硬件单元包含 32 个 VI 视频物理通道。视频物理通道分为 16 个主通道和 16 个次通道，主通道编号为 Chn0～Chn15，次通道编号为 Chn16～Chn31，但只有编号为偶数的主通道对应的次通道有效（即 Chn17、Chn19、…、Chn31 奇数通道为无效通道），因此，VI 实际有效的视频物理通道只有 24 个。次通道与主通道的图像源相同，其编号可用公式（仅针对偶数通道）获得：**次通道编号 = 主通道编号 + 16，也可通过调用 API 提供的宏 SUBCHN 来获取次通道号**。

Hi3531/Hi3532 VI 的物理通道支持对接收图像按序进行以下处理：

- 主通道：裁剪→遮挡→色度重采样→色度下采样 CDS→水平垂直翻转→VBI→输出处理后的图像到 DDR。
- 次通道：裁剪→遮挡→水平垂直缩小→水平垂直翻转→输出处理后的图像到 DDR。

其中，次通道的图像来源于主通道裁剪遮挡后的图像，如图 3-2 所示。

图3-2 VI 通道单元功能框图



所有主通道均支持 D1 和 960H 的视频图像输入，所有编号为偶数的主通道都支持 GV7601 10bit multiplexed 模式输入的 720P 视频图像，只有 Chn0/4/8/12（4 的整数倍）支持标准 BT1120 720P/1080P 视频数据输入。各主通道的能力如表 3-2 所示。

表3-2 VI 物理通道规格

规格	奇数通道	Chn2/6/10/14	Chn0/4/8/12
Crop	支持	支持	支持
水平垂直缩小	不支持	支持 水平垂直最大支持 16 倍缩小	支持 水平垂直最大支持 16 倍缩小
最大输入分辨率	960x576	1280x720，另外其次通道最大输入分辨率为 640x720	2560x1600，另外其次通道最大输入分辨率为 960x1600
水平垂直翻转	支持	支持	支持
像素格式	支持 SP420 和 SP422	支持 SP420 和 SP422	支持 SP420 和 SP422

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

规格	奇数通道	Chn2/6/10/14	Chn0/4/8/12
GV7601 10bit multiplexed 模式输入	不支持	支持	支持
SMPTE 296M(720P)	不支持	不支持	支持
SMPTE 274M (1080i/1080P)	不支持	不支持	支持

典型应用场景

典型应用场景如表 3-3 所示。

表3-3 典型应用场景描述

典型场景	规格描述
16 路 D1 采集	Dev0/2/4/6 分别进 4D1（BT656/108Mhz），共 4x4 路 D1 Chn0～15 分别输出 1D1+ 8 个次通道输出
	Dev0～7 均配置为进 2D1(BT656/54Mhz)，共 8x2 路 D1 Chn0～15 分别输出 1D1 + 8 个次通道输出
16 路 960H 采集	Dev0/2/4/6 分别进 4D1（BT656/144Mhz），共 4x4 路 960H Chn0～15 分别输出 1 路 960H ＋8 个次通道输出
	Dev0～7 均配置为进 2x960H(BT656/72Mhz)，共 8x2 路 960H Chn0～15 分别输出 1 路 960H ＋8 个次通道输出
8 路 D1 采集	Dev0～7 端口均配置为进 1D1（BT656/27MHz） 偶数 Chn 输出 ＋8 个次通道输出
8 路 960H 采集	Dev0～7 端口均配置为进 1x960H（BT656/36MHz） 偶数 Chn 输出 ＋8 个次通道输出
8 路 720P/30fps 输入	8 个端口均配置为 BT656/148.5MHz（每个端口连 8bit 数据线，专用解 GV7601 10bit multiplexed 模式） 偶数 Chn 输出 ＋8 个次通道输出
4 路 720P/30fps 输入	Dev0/2/4/6 分别进 1 路 720P（每个端口连 16bit 数据线，BT1120/74.25Mhz） Chn0/4/8/12 输出（Chn=2xDev）＋4 个次通道输出
4 路 1080P/30fp 输入	Dev0/2/4/6 分别进 1 路 1080P（每个端口连 16bit 数据线，BT1120/74.25Mhz） Chn0/4/8/12 输出（Chn=2xDev） ＋4 个次通道输出

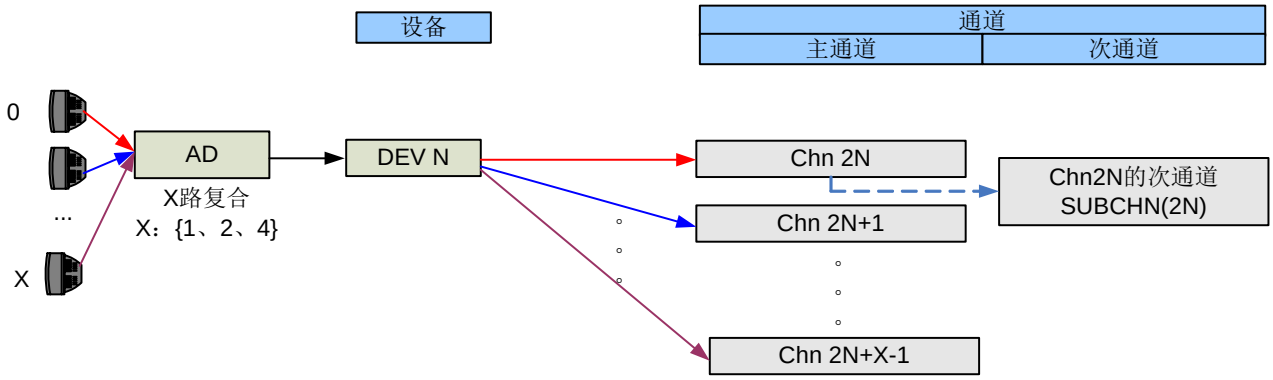
海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

绑定关系

VI 的设备和物理通道可以分为 4 组，其中 Dev0、Dev1 和 Chn0~Chn3 组成第一组，Dev2、Dev3 和 Chn4～Chn7 组成第二组，以此类推，如图 3-1 所示。同一组内前两个物理通道只能与组内第一个设备绑定，后两个物理通道可以与两个设备任意绑定，但一个物理通道同时只能绑定一个 Dev，且只能用主通道进行绑定。在设置设备属性时，系统会默认设置物理通道与设备的绑定关系，无需显式地设置绑定关系。通常，默认的绑定关系即可满足大多数应用场景的要求，特殊情况，可以调整物理通道与 VI 设备的绑定关系。

简化概念（设备、Chn）如图 3-3 所示。

图3-3 简化概念（设备、Chn）



通常，只需知道设备和 Chn 的概念，按照设备与 Chn 的默认绑定关系使用即可，无需显式地去设置 Chn 与 Dev 的绑定关系。图 3-3 中，Chn 前的彩色箭头与 AD 前的彩色箭头相对应。当 Dev N 设置为 X 路复合模式（X = 1、2、4）时，默认与之绑定的通道个数为 X，绑定的通道分别为 Chn 2N～Chn 2N+X-1。例如，如果 Dev 0 设置为 1 路复合模式，则默认与 Dev 0 绑定的通道只有 Chn0；如果 Dev 0 设置为 2 路复合模式，那么，默认与它绑定的通道有 Chn0 和 Chn1；如果 Dev 0 设置为 4 路复合模式，则默认与它绑定的通道有 Chn 0～Chn3。

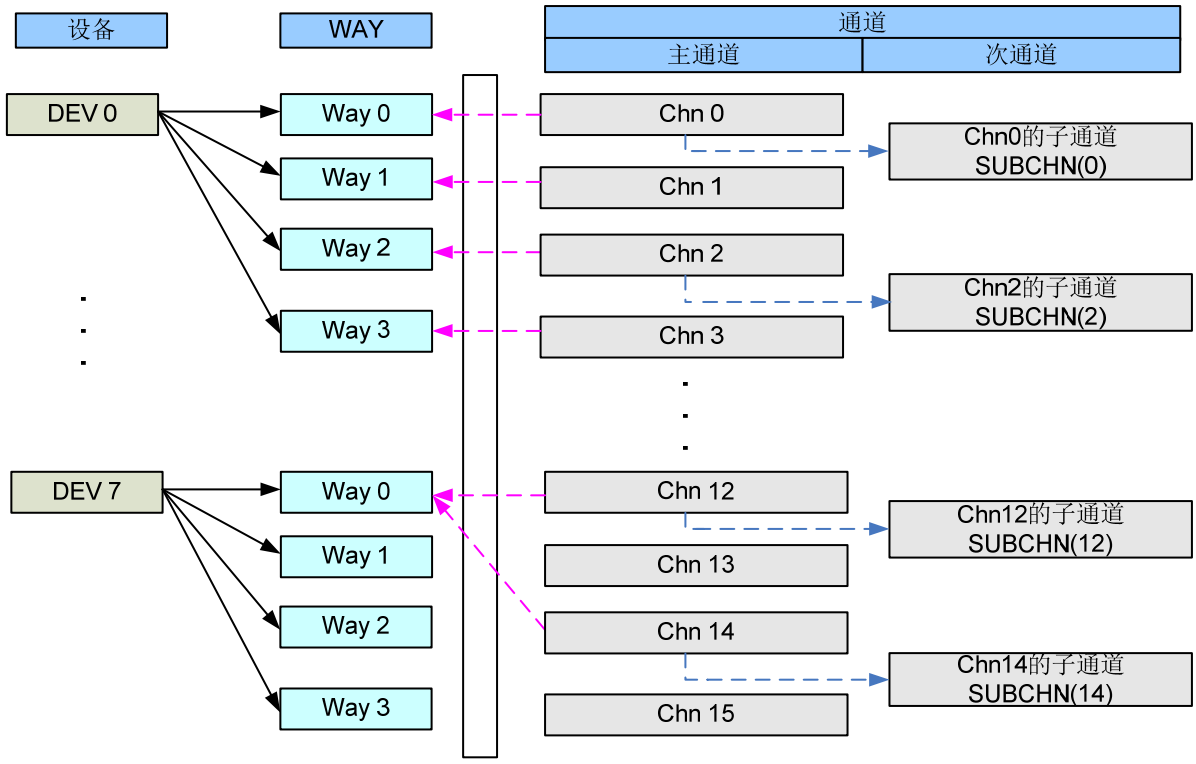
注意

当 Dev N（N=0、2、4、6）设置为 4 路复合模式，或者 720P/1080P（标准 BT1120）输入时，Dev N+1 将不可用。次通道无需绑定，与主通道共用绑定关系，即只需绑定主通道。

完整概念（设备、Way、Chn）如图 3-4 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-4 完整概念（设备、Way、Chn）



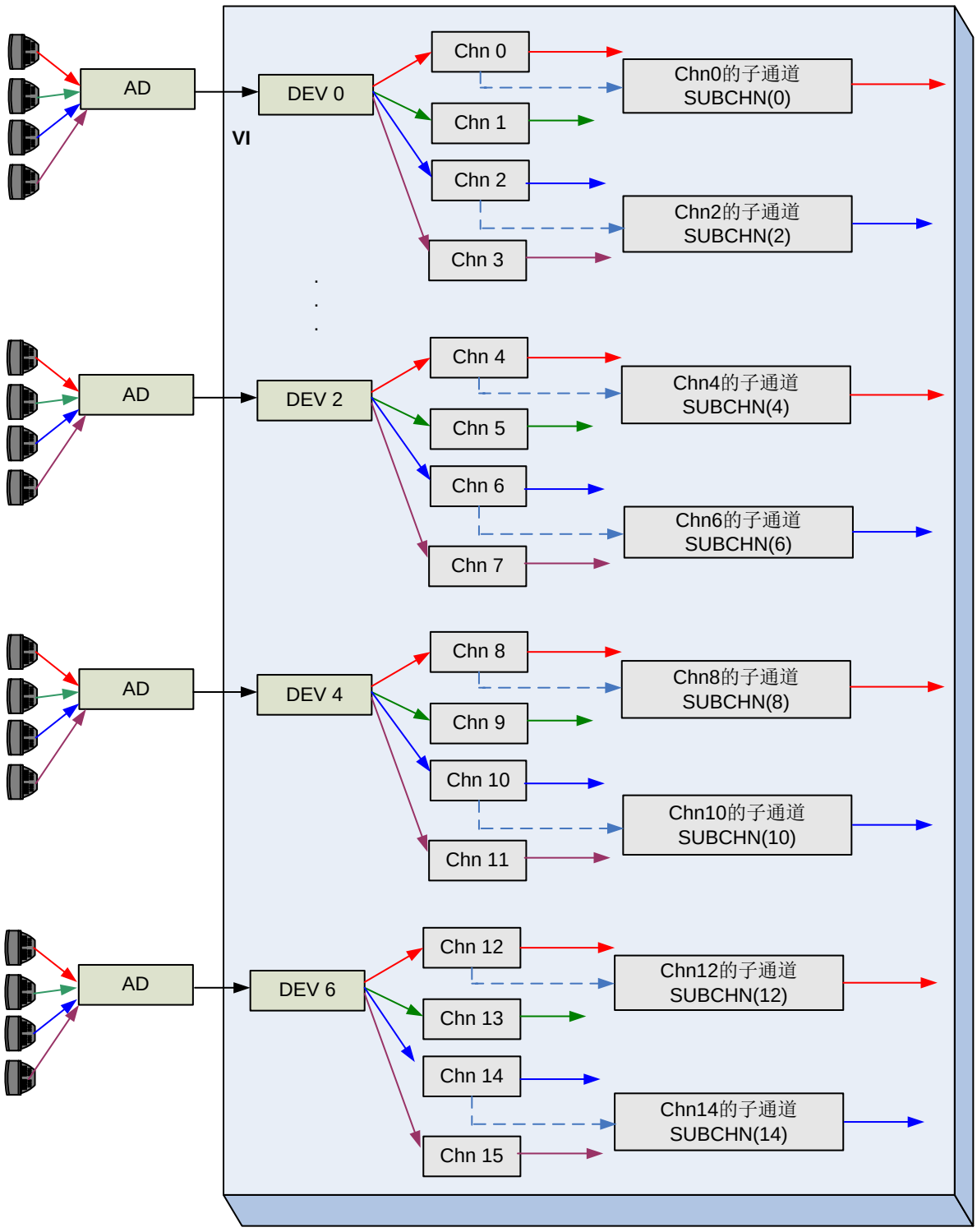
当需要使多个通道都显示同一个视频输入源的图像时，可以调整默认的绑定关系，使多个通道同时绑定到同一个{Dev，Way}，此时需要了解 Way 的概念。如图 3-4 所示，每个 Dev 拥有 4 个数据通路 Way0~Way3，一个 Way 对应一路 camera 的输入，根据设定的 X 路复合模式决定哪几个 Way 启用。在默认绑定关系下，根据 X 路复合模式，每组通道依次与启用的 Way i 绑定，例如 Dev0 和 Dev1 均设置为 2 路复合模式，那么 Chn0、Chn1 与 Dev0 下的 Way0、Way1 绑定，默认输出 camera0 和 camera1 的图像，Chn2、Chn3 与 Dev1 下的 Way0、Way1 绑定，默认输出 camera0 和 camera1 的图像；如果 Dev0 设置为 4 路复合模式，那么就 Chn0～Chn3 分别与 Way0～Way3 绑定，分别输出 camera0～camera3 的图像，如图 3-4 Dev0 所示。用户可根据需要调整绑定关系，如图 3-4 中 Dev7 所示，使 Chn12、Chn14 绑定到{Dev7，Way0}，这样两个通道的输入视频源相同，从而可以根据需要输出多路不同分辨率的视频图像。

默认绑定关系详解

默认绑定关系如图 3-5～图 3-8 所示。

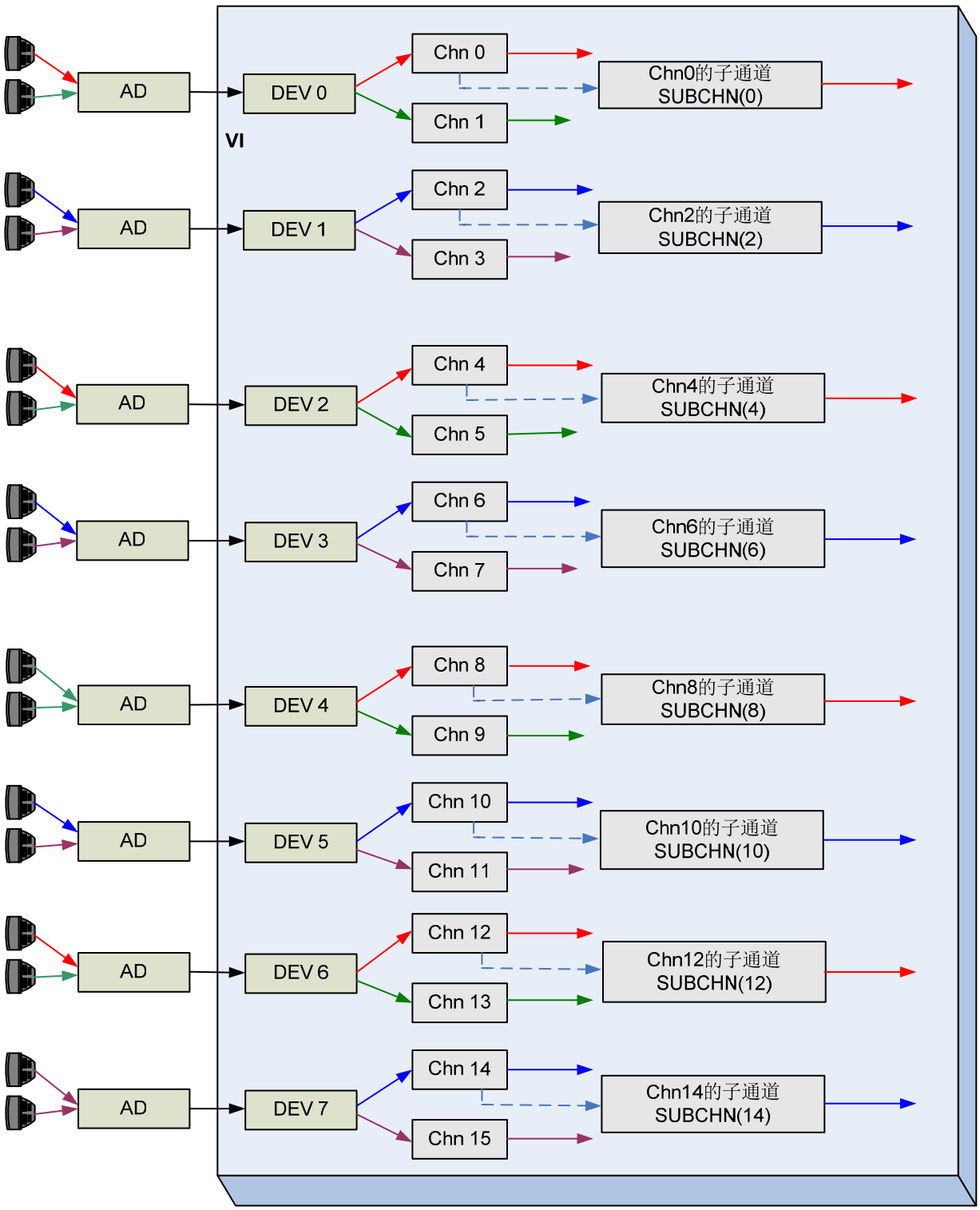
海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-5 16 路 D1/960H 场景（4 路复合模式）



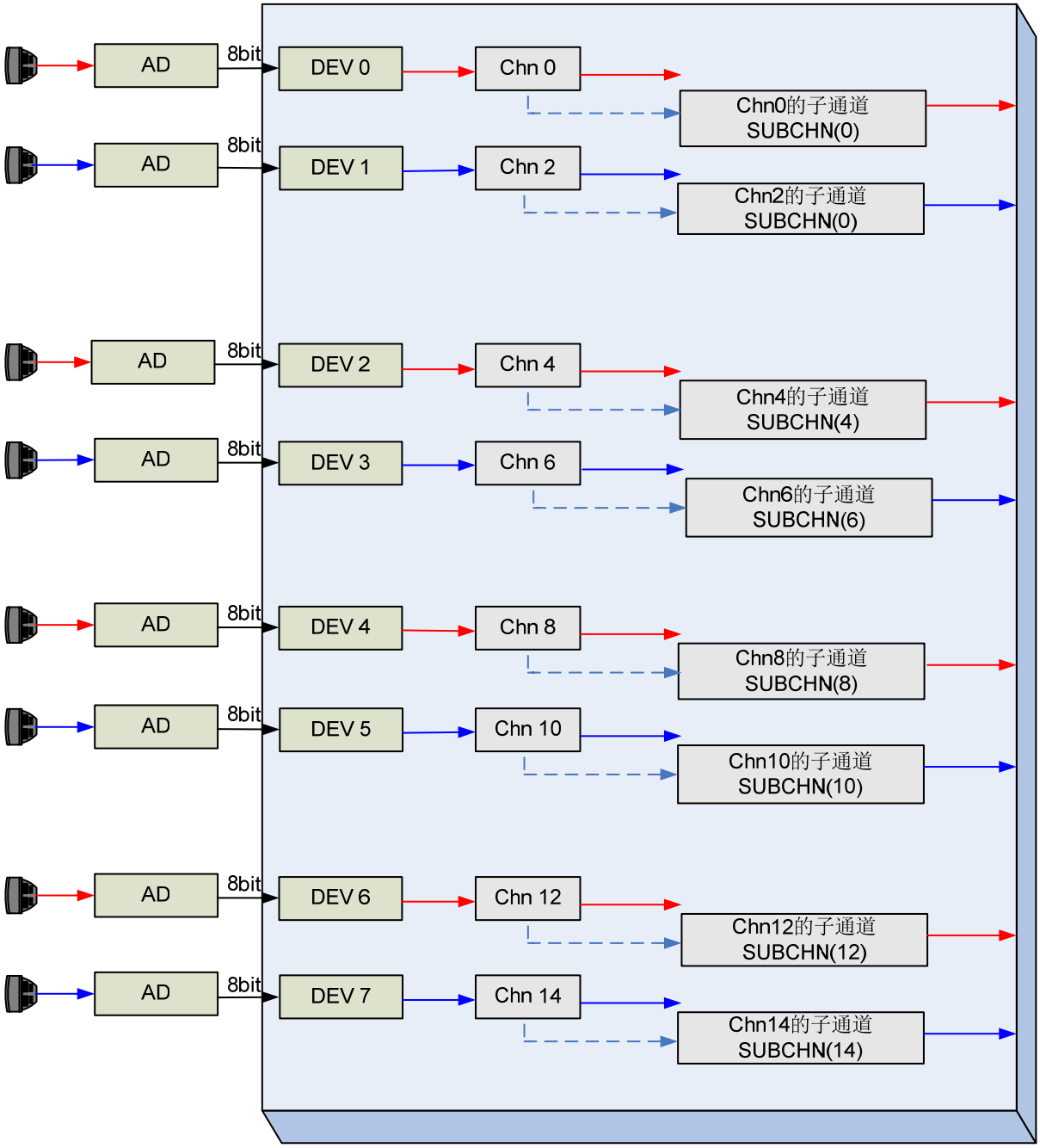
海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-6 16 路 D1/960H 场景（2 路复合模式）



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-7 8 路 720P 或 8 路 D1/960H 场景（1 路复合模式）



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-8 4 路 720P/1080P 场景（1 路复合模式）

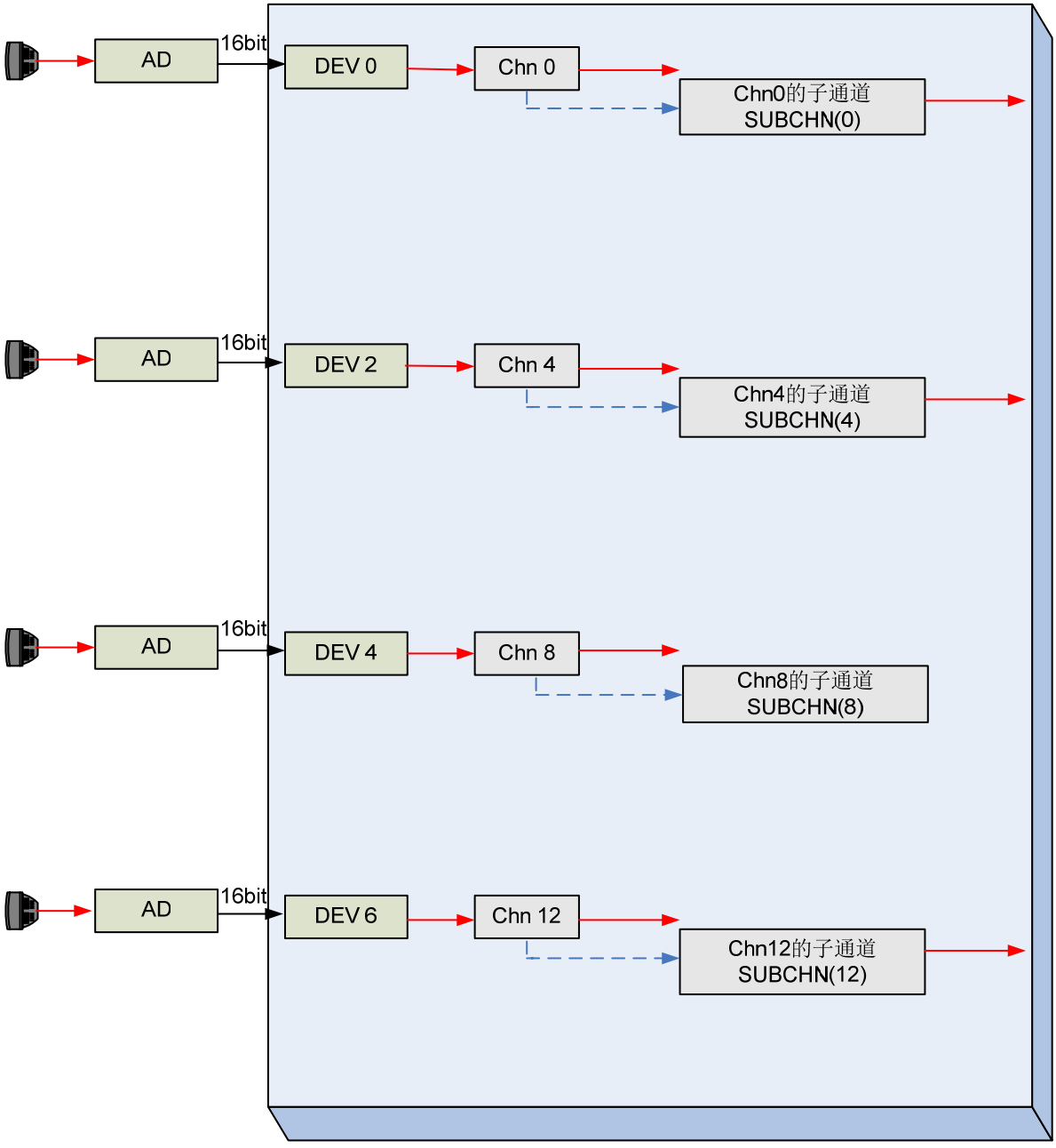


图 3-8 中，相同颜色的箭头对应同一路 camera 输入。默认的绑定关系主要基于以下两个方面：

- 每两个 Dev 与后面的 4 个 Chn 组成一组，即 Hi3531/Hi3532 芯片 VI 硬件单元可分为功能相同的 4 组{Dev，Chn}，请参见图 3-1。每组{Dev，Chn}内，只有第一个 Dev 支持 BT1120_STANDARD 模式（4 路 720P/1080P）和 4 路复合模式，只有第一个通道最大支持 1080P 视频图像输出。
- X（X=1、2 或 4）路复合模式决定了与 Dev 绑定的 Chn 的个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

掩码配置

- 16 路 D1 或 960H

  16 路 D1 或 960H 时，硬件连接芯片有两种方案：

  - 第 1 种方案：使用 4 个（VIU0～VIU3）BT.1120 接口的高 8 位接收来自 AD 的视频数据，具体配置如表 3-4 所示；
  - 第 2 种方案：使用第 1 个（VIU0）和第 3 个（VIU2）BT.1120 口高低 8 位接收来自 AD 的视频数据，具体配置如表 3-5 所示。

  16 路 D1 或 960H 场景下，设置 VI 设备属性时，可根据表 3-4 和表 3-5 配置掩码。

表3-4 16 路 D1 或 960H 场景下的掩码配置（第 1 种方案）

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
1	0x00FF0000	0x0
2	0x0000FF00	0x0
3	0x000000FF	0x0
4	0xFF000000	0x0
5	0x00FF0000	0x0
6	0x0000FF00	0x0
7	0x000000FF	0x0

注意

对于 Hi3531/Hi3532 来说，由于芯片内部支持通过两个 BT.1120 口获得 16xD1 输入(具体请参考《Hi3531 H264 编解码处理器用户指南》/《Hi3532 H.264 编码处理器用户指南》)，当芯片管脚复用方式为通过两个 BT.1120 口获得 16D1 输入时，此时掩码配置方式如表 3-5 所示。

表3-5 16 路 D1 或 960H 场景下的掩码配置（第 2 种方案）

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
2	0x00FF0000	0x0
4	0xFF000000	0x0
6	0x00FF0000	0x0

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 8 路 720p 场景

  8 路 720p 场景下，设置 VI 设备属性时，可根据表 3-6 配置掩码。

表3-6 8 路 720p 场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
1	0x00FF0000	0x0
2	0x0000FF00	0x0
3	0x000000FF	0x0
4	0xFF000000	0x0
5	0x00FF0000	0x0
6	0x0000FF00	0x0
7	0x000000FF	0x0

- 4 路 720p 或 4 路 1080p 场景

  4 路 720p 或 4 路 1080p 场景下，设置 VI 设备属性时，可根据表 3-7 配置掩码。

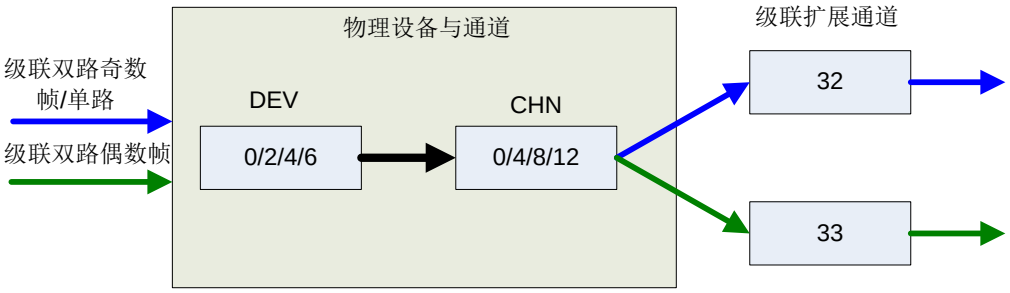
表3-7 4 路 720p 或 4 路 1080p 场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x00FF0000
2	0x0000FF00	0x000000FF
4	0xFF000000	0x00FF0000
6	0x0000FF00	0x000000FF

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

级联模式

图3-9 VI 级联数据传输图



VI 设备 0/2/4/6 及对应的物理通道 0/4/8/12 支持级联模式。级联模式使能后，VI 仍然通过原有的物理设备和物理通道采集级联图像，通过两个级联扩展通道（虚拟通道）发送并区分采集的两路级联图像，如图 3-9 所示。同时要遵守以下原则：

- 单路级联时，使用扩展通道 32 来发送采集的图像；
- 双路级联时，扩展通道 32 发送 VO 级联扩展设备 1 传输的图像（奇数帧图像），扩展通道 33 发送 VO 级联扩展设备 2 传输的图像（偶数帧图像）。

说明：图像是奇数帧还是偶数帧由 VO 端设置图像 Id 决定，双路级联时，VO 扩展级联设备 1 传输的图像 Id 固定设为奇数帧，VO 级联扩展设备 2 传输的图像 Id 固定设为偶数帧，SDK 内部确保并实现了双路数据的奇偶帧间插传输。VI 通过读取图像 VBI 信息得到当前帧是奇数帧还是偶数帧。

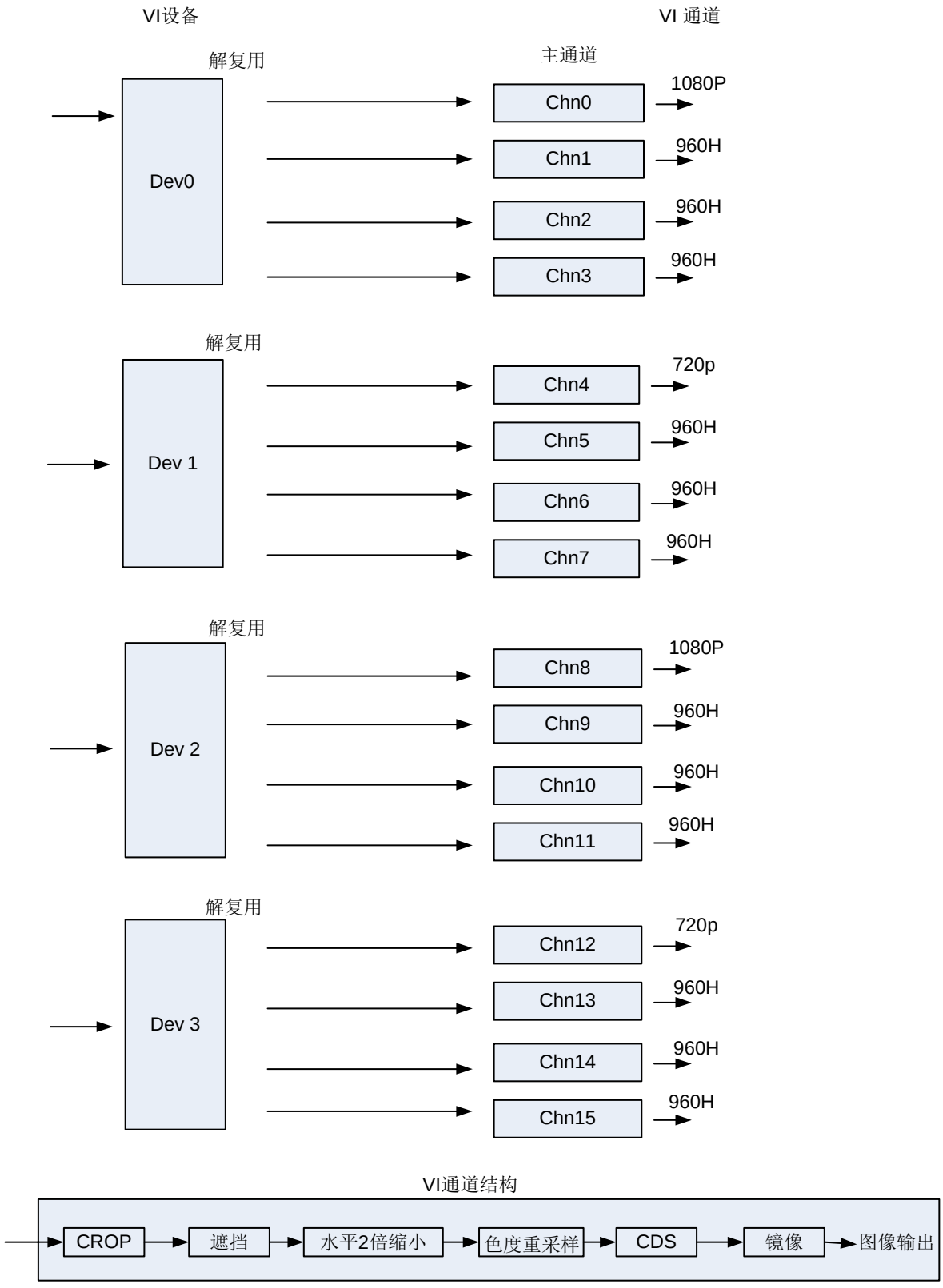
3.3.2 Hi3521/Hi3520A

功能框图

Hi3521/Hi3520A 视频输入硬件单元的功能如图 3-10 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-10 Hi3521/Hi3520A VI 硬件单元功能框图



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

视频输入设备

Hi3521/Hi3520A 芯片有 2 个 BT.1120 接口，每个 BT.1120 接口依次对应两个 VI 设备，即第一个 BT.1120 口对应 VI 的 Dev0 和 Dev1，第二个 BT.1120 口对应 VI 的 Dev2 和 Dev3。

4 个 VI 设备均支持 4 路 D1 复合模式输入（BT.656 协议），以及 1 路 720P interleave 模式。只有编号为 0/2 的 Dev 支持 1 路 720P/1080P 高清输入（BT.1120 协议），此时，编号为奇数的 Dev 不可用。

视频物理通道

Hi3521/Hi3520A VI 硬件单元包含 16 个 VI 视频物理通道，依次编号为 Chn0～Chn15。不存在次通道。

所有通道均支持 D1 和 960H 的视频图像输入，所有编号为 4 倍数的通道都支持 GV7601 10bit multiplexed 模式输入的 720P 视频图像，只有 Chn0/8（8 的整数倍）支持标准 BT1120 720P/1080P 视频数据输入。

绑定关系

VI 的设备和物理通道分为 4 组，一个设备对应 4 个通道，如 Dev0 与 Chn0～Chn3 为一组，以此类推。一个组内的物理通道与所对应的 Dev 固定绑定，不允许改变它们的绑定关系。

掩码配置

- 16 路图像输入场景

  16 路图像输入场景下，设置 VI 设备属性时，可根据表 3-8 配置掩码。

表3-8 16 路图像输入场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
1	0x00FF0000	0x0
2	0x0000FF00	0x0
3	0x000000FF	0x0

- 4 路 720p 场景

  4 路 720p 场景下，设置 VI 设备属性时，可根据表 3-9 配置掩码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表3-9 8 路 720p 场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
1	0x00FF0000	0x0
2	0x0000FF00	0x0
3	0x000000FF	0x0

- 2 路 720p 或 2 路 1080p 场景

  2 路 720p 或 2 路 1080p 场景下，设置 VI 设备属性时，可根据表 3-10 配置掩码。

表3-10 2 路 720p 或 2 路 1080p 场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x00FF0000
2	0x0000FF00	0x000000FF

级联模式

Hi3521/Hi3520A 不支持。

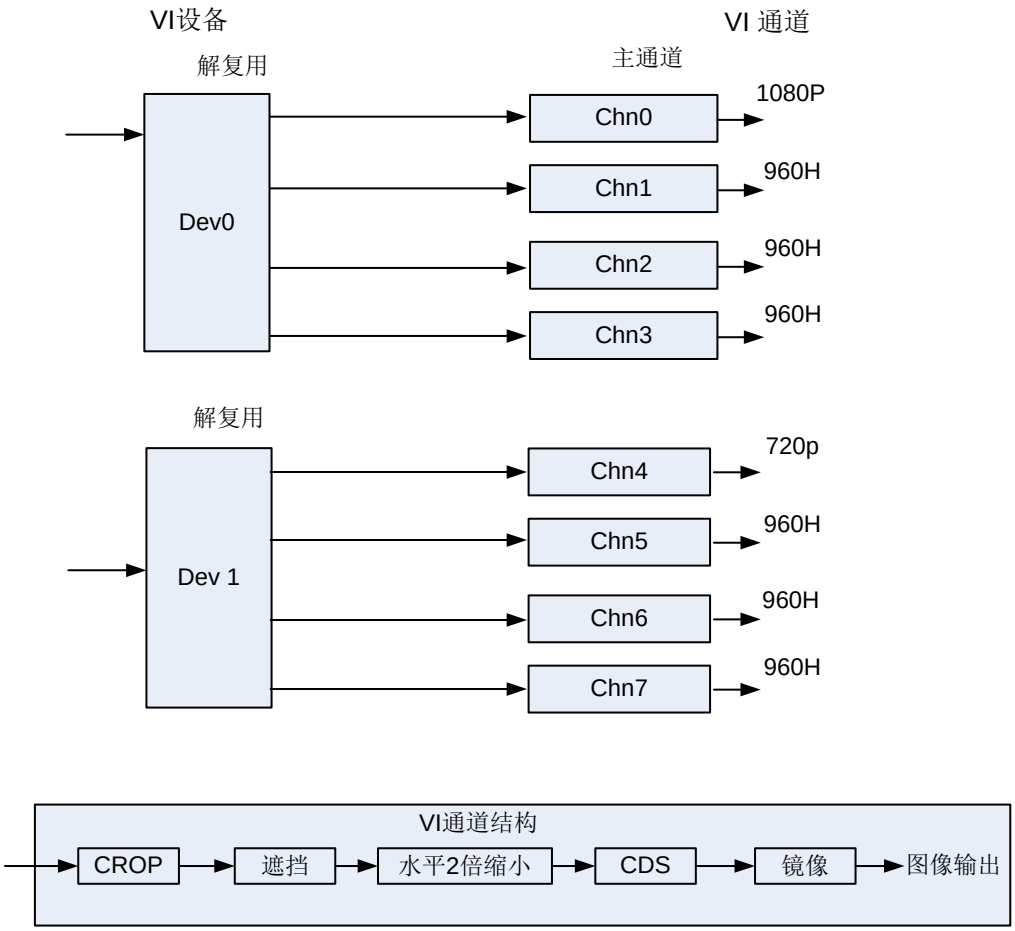
3.3.3 Hi3520D/Hi3515A/Hi3515C

功能框图

Hi3520D 视频输入硬件单元的功能如图 3-11 所示。Hi3515A/Hi3515C 只有 1 个 VI 设备，4 个 VI 物理通道，且通道 0 最大支持 720P。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图3-11 Hi3520D VI 硬件单元功能框图



视频输入设备

- Hi3520D 芯片有 1 个 BT.1120 接口，1 个 BT.1120 接口对应两个 VI 设备，即这个 BT.1120 口对应 VI 的 Dev0 和 Dev1。

  2 个 VI 设备均支持 4 路 D1 复合模式输入（BT.656 协议），以及 1 路 720P interleave 模式。只有编号为 0 的 Dev 支持 1 路 720P/1080P 高清输入（BT.1120 协议），此时，编号为奇数的 Dev 不可用。
- Hi3515A/Hi3515C 芯片有 1 个 BT.656 接口，不支持 BT.1120 接口。

  VI 设备支持 4 路 D1 复合模式输入（BT.656 协议），以及 1 路 720P interleave 模式。

视频物理通道

- Hi3520D VI 硬件单元包含 8 个 VI 视频物理通道，依次编号为 Chn0～Chn7。不存在次通道。

  所有通道均支持 D1 和 960H 的视频图像输入，所有编号为 4 倍数的通道都支持 GV7601 10bit multiplexed 模式输入的 720P 视频图像，只有 Chn0 支持标准 BT1120 720P/1080P 视频数据输入。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- Hi3515A/Hi3515C VI 硬件单元包含 4 个 VI 视频物理通道，依次编号为 Chn0～Chn3。不存在次通道。

  所有通道均支持 D1 和 960H 的视频图像输入，所有编号为 4 倍数的通道都支持 GV7601 10bit multiplexed 模式输入的 720P 视频图像，不支持标准 BT1120 720P/1080P 视频数据输入。

绑定关系

VI 的设备和物理通道分为 2 组，一个设备对应 4 个通道，如 Dev0 与 Chn0～Chn3 为一组，以此类推。一个组内的物理通道与所对应的 Dev 固定绑定，不允许改变它们的绑定关系。

掩码配置

- 8 路图像输入场景

  8 路图像输入场景下，设置 VI 设备属性时，可根据表 3-8 配置掩码。

表3-11 8 路图像输入场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
1	0x00FF0000	0x0

- 2 路 720p(INTERLEAVED)场景

  2 路 720p 场景下，设置 VI 设备属性时，可根据表 3-9 配置掩码。

表3-12 2 路 720p 场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x0
1	0x00FF0000	0x0

- 1 路 720p 或 1 路 1080p 场景

  1 路 720p 或 1 路 1080p 场景下，设置 VI 设备属性时，可根据表 3-10 配置掩码。

表3-13 1 路 720p 或 1 路 1080p 场景下的掩码配置

设备号	掩码 0	掩码 1
0	0xFF000000	0x00FF0000

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

级联模式

Hi3520D /Hi3515A/Hi3515C 不支持。

3.3.4 Hi3518/Hi3516C

功能框图

Hi3518/Hi3516C 视频输入硬件单元的功能如图 3-12 所示。

图3-12 Hi3518/Hi3516C VI 硬件单元功能框图

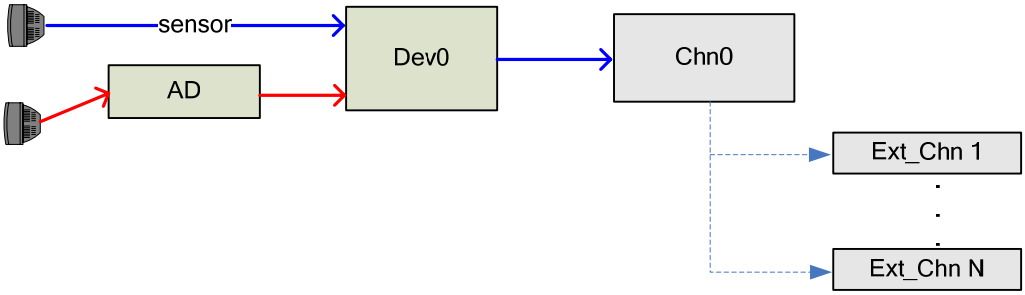
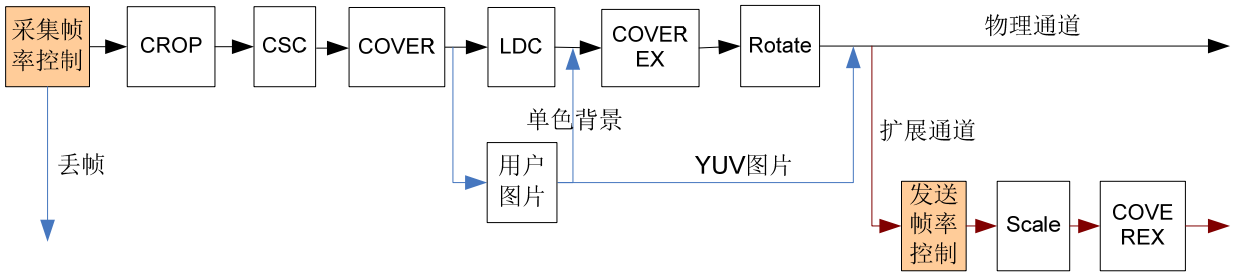


图3-13 Hi3518/Hi3516C VI 通道功能框图



视频输入设备

Hi3518/Hi3516C 芯片只有一个 VI 设备，即 Dev0。

Dev0 支持 BT.656、BT.601、DC 协议输入，但不支持 BT.1120。

Dev0 不支持非连续 pixel clock 时序。

视频物理通道

Hi3518/Hi3516C VI 硬件单元仅包含 1 个 VI 视频物理通道，即 Chn0。不存在次通道，但支持扩展通道。

物理通道支持的典型分辨率如 D1、720P@30、720P@60、1080P@30 等。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

视频扩展通道

扩展通道是物理通道的扩展，主要实现缩放功能，其数据来源于物理通道。
Hi3518/Hi3516C 最多支持 16 个扩展通道。

绑定关系

Hi3518/Hi3516C VI 的物理通道与所对应的设备固定绑定，不允许改变它们的绑定关系。

掩码配置

掩码的高 12bit 对应着硬件线路的 12 个 pin 脚连接（D0~D11），用户需要根据实际连接情况设置恰当的掩码配置，掩码的最高比特位对应的 pin 为 D11，例如 10bit 输入的 Sensor 连接的 pin 为 D2~D11，掩码配置为 0xFFC00000。

- 1 路 720p 图像输入场景（12bit 输入）

  1 路 720p 图像输入场景下，设置 VI 设备属性时，可根据表 3-14 配置掩码。

表3-14 1 路 720p 场景下的掩码配置（12bit）

设备号	掩码 0	掩码 1
0	0xFFF00000	0x0

- 1 路 D1 图像输入场景（8bit 输入）

  1 路图像输入场景下，设置 VI 设备属性时，可根据表 3-15 配置掩码。

表3-15 1 路图像输入场景下的掩码配置（8bit）

设备号	掩码 0	掩码 1
0	0xFF000000	0x0

3.4 API 参考

视频输入（VI）实现启用视频输入设备、视频输入通道、绑定视频输入通道等功能。

该功能模块提供以下 MPI：

- HI_MPI_VI_SetDevAttr：设置 VI 设备属性。
- HI_MPI_VI_GetDevAttr：获取 VI 设备属性。
- HI_MPI_VI_SetDevAttrEx：设置 VI 设备属性（特殊时序可调用该高级接口）。
- HI_MPI_VI_GetDevAttrEx：获取 VI 设备属性。
- HI_MPI_VI_EnableDev：启用 VI 设备。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VI_DisableDev：禁用 VI 设备。
- HI_MPI_VI_SetChnAttr：设置 VI 通道属性。
- HI_MPI_VI_GetChnAttr：获取 VI 通道属性。
- HI_MPI_VI_SetChnMinorAttr：设置 VI 通道次属性。
- HI_MPI_VI_GetChnMinorAttr：获取 VI 通道次属性。
- HI_MPI_VI_ClearChnMinorAttr：清除 VI 通道次属性。
- HI_MPI_VI_EnableChn：启用 VI 通道。
- HI_MPI_VI_DisableChn：禁用 VI 通道。
- HI_MPI_VI_EnableChnInterrupt：启用 VI 通道响应硬件中断处理程序。
- HI_MPI_VI_DisableChnInterrupt：禁止 VI 通道响应硬件中断处理程序。
- HI_MPI_VI_SetFrameDepth：设置可获取的 VI 图像最大深度。
- HI_MPI_VI_GetFrameDepth：获取设置的 VI 图像最大深度。
- HI_MPI_VI_GetFrame：获取 VI 图像。
- HI_MPI_VI_GetFrameTimeOut：在设定时间内获取 VI 采集的图像。
- HI_MPI_VI_ReleaseFrame：释放 VI 图像数据所占的缓存。
- HI_MPI_VI_SetUserPic：设置用户图片，作为无视频信号时的插入图片。
- HI_MPI_VI_EnableUserPic：启用插入用户图片。
- HI_MPI_VI_DisableUserPic：禁用插入用户图片。
- HI_MPI_VI_EnableCascade：启用 VI 设备的级联模式。
- HI_MPI_VI_DisableCascade：关闭 VI 设备级联模式。
- HI_MPI_VI_EnableCascadeChn：启用 VI 级联扩展通道。
- HI_MPI_VI_DisableCascadeChn：禁用 VI 级联扩展通道。
- HI_MPI_VI_ChnBind：设置 VI 通道绑定关系。
- HI_MPI_VI_ChnUnBind：解除 VI 通道绑定关系。
- HI_MPI_VI_GetChnBind：获取 VI 通道绑定关系。
- HI_MPI_VI_GetFd：获取 VI 通道的设备文件句柄。
- HI_MPI_VI_Query：查询 VI 通道的中断计数、平均帧率等信息。
- HI_MPI_VI_SetFlashConfig：设置 VI 闪光灯的配置信息。
- HI_MPI_VI_GetFlashConfig：获取 VI 闪光灯的配置信息。
- HI_MPI_VI_FlashTrigger：关闭/启用 VI 闪光灯。
- HI_MPI_VI_SetExtChnAttr：设置扩展通道属性。
- HI_MPI_VI_GetExtChnAttr：获取 VI 扩展通道属性。
- HI_MPI_VI_SetLDCAttr：设置 VI 镜头畸变校正（LDC）属性。
- HI_MPI_VI_GetLDCAttr：获取 VI 镜头畸变校正（LDC）属性。
- HI_MPI_VI_SetCSCAttr：设置 VI 设备的 CSC 属性。
- HI_MPI_VI_GetCSCAttr：获取 VI 设备的 CSC 属性。
- HI_MPI_VI_SetRotate：设置 VI 图像旋转属性。
- HI_MPI_VI_GetRotate：获取 VI 图像旋转属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VI_GetChnLuma：获取通道亮度和统计。
- HI_MPI_VI_SetPortMode：设置 VI 端口属性。
- HI_MPI_VI_GetPortMode：获取 VI 端口属性。
- HI_MPI_VI_EnableChn422to420：开启 VI 通道将采集的 422 图像转为 420 功能。
- HI_MPI_VI_DisableChn422to420：关闭 VI 通道将采集的 422 图像转为 420 功能。
- HI_MPI_VI_SetChnScanMode：设置 VI 通道扫描模式。
- HI_MPI_VI_GetChnScanMode：获取 VI 通道扫描模式。

HI_MPI_VI_SetDevAttr

【描述】

设置 VI 设备属性。基本设备属性默认了部分芯片配置，满足绝大部分的 AD 芯片对接要求，对于特殊的时序，可使用高级设备属性接口 HI_MPI_VI_SetDevAttrEx，配置 VI 设备的所有属性。

【语法】

```
HI_S32 HI_MPI_VI_SetDevAttr(VI_DEV ViDev, const VI_DEV_ATTR_S *pstDevAttr)
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)。	输入
pstDevAttr	VI 设备属性指针。 静态属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	调用该接口后，系统会根据设置的接口模式和工作模式，自动设置 Chn 与 Dev 的默认绑定关系。如需改变绑定关系，可在调用完此接口后，再调用 HI_MPI_VI_ChnUnBind 高级接口解绑定，然后设置新的绑定关系

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	差异
Hi3521/Hi3520A/ Hi3520D/Hi3515A/Hi3515C	不支持更改设备与通道的绑定关系
Hi3518/Hi3516C	不支持 BT.1120 时序，不支持更改设备与通道的绑定关系，只支持一路复合工作模式。

【VI 模块参数】

模块参数	意义及取值
detect_err_frame	该参数表示是否由用户来控制 VI 的时序异常实时检测 0：默认，表示由 SDK 来控制实时检测，只要检测到错误图像就丢帧 -1：表示不开启错误丢帧功能 >0：表示由用户控制
drop_err_frame	该参数表示当检测到图像错误时，连续丢多少帧(不管后面的图像是否正确)。0 和 1 表示丢弃当前帧。默认值为 0
int_gap_time	以 PAL 制为例，取值在 20000-40000 之间，单位 us，但如果两中断间隔超过 40000，则 VI 必然会出现图像丢失问题
stop_int_level	当 VI 内部检测到中断率过多时，自动关闭响应中断，防止因中断过多导致应用程序得不到响应；默认值为 0，单位 us。 该值不能设得过大，因为当业务量较大时，可能中断被延时响应了，此时就有两个中断间隔时间较短，但 VI 还是能处理的过来
discard_int	用于控制丢弃前几个中断，即要等几个中断确保配置生效后再去使能通道采集功能，以保证逻辑不会出错。 默认值为 0。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 在调用前要保证 VI 设备处于禁用状态。如果 VI 设备已处于使能状态，可以使用 HI_MPI_VI_DisableDev 来禁用设备。
- 参数 pstDevAttr 主要用来配置指定 VI 设备的视频接口模式，用于与外围 camera、sensor 或 codec 对接，支持的接口模式包括 BT.656、BT.601、digital camera、BT.1120 逐行以及 BT.1120 隔行模式。用户需要配置以下几类信息，具体属性意义参见 3.4 节数据类型部分的说明：

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 接口模式信息：接口模式为 BT656、BT601、digital camera、BT1120 等模式
- 工作模式信息：1 路、2 路、4 路复合模式
- 数据布局信息：有效数据的布局
- 数据信息：隔行或逐行输入、YUV 数据输入顺序
- 同步时序信息：垂直、水平同步信号的属性

【举例】

```
HI_S32 s32Ret;
VI_DEV ViDev = 0;
VI_CHN ViChn = 0;
VI_DEV_ATTR_S stDevAttr;
VI_CHN_ATTR_S stChnAttr;
stDevAttr.enIntfMode = VI_MODE_BT656;
stDevAttr.enWorkMode = VI_WORK_MODE_1Multiplex;
stDevAttr.au32CompMask[0] = 0xFF000000;
stDevAttr.au32CompMask[1] = 0x0;
stDevAttr.enScanMode = VI_SCAN_INTERLACED;
stDevAttr.s32AdChnId[0] = -1;
stDevAttr.s32AdChnId[1] = -1;
stDevAttr.s32AdChnId[2] = -1;
stDevAttr.s32AdChnId[3] = -1;
s32Ret = HI_MPI_VI_SetDevAttr(ViDev, &stDevAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set dev attributes failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
s32Ret = HI_MPI_VI_EnableDev(ViDev);
if (s32Ret != HI_SUCCESS)
{
   printf("Enable dev failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
stChnAttr.stCapRest.s32X = 0;
stChnAttr.stCapRect.s32Y = 0;
stChnAttr.stCapRect.u32Width = 720;
stChnAttr.stCapRect.u32Height = 576;
stChnAttr.stDestSize.u32Width = 720;
stChnAttr.stDestSize.u32Height = 576;
stChnAttr.enCapSel = VI_CAPSEL_BOTH;
stChnAttr.enPixFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_422;
stChnAttr.bMirror = HI_FALSE;
stChnAttr.bFlip = HI_FALSE;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
stChnAttr.bChromaResample = HI_FALSE
stChnAttr.s32SrcFrameRate = -1;
stChnAttr.s32FrameRate = -1;
s32Ret = HI_MPI_VI_SetChnAttr(ViChn,&stChnAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set chn attributes failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
s32Ret = HI_MPI_VI_EnableChn(ViChn);
if (s32Ret != HI_SUCCESS)
{
   printf("Enable chn failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
/* now, vi is capturing images, you can do something else ... */

s32Ret = HI_MPI_VI_DisableChn(ViChn);
if (s32Ret != HI_SUCCESS)
{
   printf("Disable chn failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
s32Ret = HI_MPI_VI_DisableDev(ViDev);
if (s32Ret != HI_SUCCESS)
{
   printf("Disable dev failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
```

【相关主题】

- HI_MPI_VI_GetDevAttr
- HI_MPI_VI_SetDevAttrEx

HI_MPI_VI_GetDevAttr

【描述】

获取 VI 设备属性。

【语法】

```
HI_S32 HI_MPI_VI_GetDevAttr(VI_DEV ViDev, VI_DEV_ATTR_S *pstDevAttr)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)。	输入
pstDevAttr	VI 设备属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

如果未设置 VI 设备属性，该接口将返回失败。

【举例】

无。

【相关主题】

HI_MPI_VI_SetDevAttr

HI_MPI_VI_SetDevAttrEx

【描述】

设置 VI 设备属性。一般情况下，不需要调用此接口，特殊地，当基本设备属性接口 HI_MPI_VI_SetDevAttr 不能满足特殊的时序时，可使用该高级接口，配置 VI 设备的所有属性。

【语法】

```
HI_S32 HI_MPI_VI_SetDevAttrEx(VI_DEV ViDev, const VI_DEV_ATTR_EX_S
*pstDevAttrEx)
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstDevAttrEx	VI 设备高级属性指针。 静态属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	调用该接口后，系统会根据设置的接口模式和工作模式，自动设置 Chn 与 Dev 的默认绑定关系。如需改变绑定关系，可在调用完此接口后，再调用高级接口解绑定，然后设置新的绑定关系。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	不支持更改设备与通道的绑定关系。
Hi3518/Hi3516C	不支持 BT.1120 时序，不支持更改设备与通道的绑定关系，只支持一路复合工作模式。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 在调用前要保证 VI 设备处于禁用状态。如果 VI 设备已处于使能状态，可以使用 HI_MPI_VI_DisableDev 来禁用设备。
- 参数 pstDevAttrEx 主要用来配置指定 VI 设备的视频接口模式，用于与外围 camera、sensor 或 codec 对接，支持的接口模式包括 BT.656、BT.601、digital camera、BT.1120 逐行以及 BT.1120 隔行模式。用户需要配置以下几类信息，具体属性意义参见 3.4 节数据类型部分的说明：
  - 接口模式信息：接口模式为 BT656、BT601、digital camera、interleave 模式
  - 工作模式信息：1 路、2 路、4 路复合模式
  - 数据布局信息：有效数据的布局
  - 数据信息：隔行或逐行输入、YUV 数据输入顺序、复合或分离模式、分量模式

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 时钟信息：上升沿或下降沿采样
- 同步时序信息：垂直、水平同步信号的属性

【举例】

无。

【相关主题】

HI_MPI_VI_GetDevAttrEx

HI_MPI_VI_GetDevAttrEx

【描述】

获取 VI 设备属性。

【语法】

```
HI_S32 HI_MPI_VI_GetDevAttrEx(VI_DEV ViDev, VI_DEV_ATTR_EX_S
*pstDevAttrEx)
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)。	输入
pstDevAttrEx	VI 设备属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

如果未设置 VI 设备属性，该接口将返回失败。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VI_SetDevAttrEx

HI_MPI_VI_EnableDev

【描述】

启用 VI 设备。

【语法】

```
HI_S32 HI_MPI_VI_EnableDev(VI_DEV ViDev);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 启用前必须已经设置设备属性，否则返回失败。
- 可重复启用，不返回失败。

【举例】

请参见 HI_MPI_VI_SetDevAttr 的举例。

【相关主题】

HI_MPI_VI_DisableDev

HI_MPI_VI_DisableDev

【描述】

禁用 VI 设备。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VI_DisableDev(VI_DEV ViDev);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	调用该接口后，系统会自动解开 Chn 与 Dev 的默认绑定关系。特殊地，如果调用了高级接口改变了默认的绑定关系，则需先调用相应的高级接口解绑定，否则系统默认解开所有跟该 Dev 绑定的 Chn。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	不支持更改设备与通道的绑定关系。
Hi3518/Hi3516C	不支持更改设备与通道的绑定关系。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 必须先禁用所有与该 VI 设备绑定的 VI 通道后，才能禁用 VI 设备。
- 可重复禁用，不返回失败。
- SDK 支持低功耗处理，禁用 VI 设备后将完全关闭该设备，需要重新设置属性，才能使能 VI 设备。

【举例】

请参见 HI_MPI_VI_SetDevAttr 的举例。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VI_EnableDev

HI_MPI_VI_SetChnAttr

【描述】

设置 VI 通道属性。

【语法】

```
HI_S32 HI_MPI_VI_SetChnAttr(VI_CHN ViChn,const VI_CHN_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstAttr	VI 通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	差异	
Hi3531/Hi3532	通道号范围	[0, VIU_MAX_CHN_NUM-2)，其中[VIU_MAX_CHN_NUM/2, VIU_MAX_CHN_NUM-2)内的奇数值无效
	通道属性参数限制	• 通道属性适用于主、次通道。必须先设置主通道属性后（可以不使能主通道），才可以设置次通道的属性。次通道属性可动态设置，即可在 VI 通道的工作过程中（通道已使能）调用此接口。 • stCapRect 中 s32Y 和 u32Height 隔行且采集单场时 2 对齐，隔行且采集双场时 4 对齐。 • stCapRect 仅对主通道有效，对次

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	差异	
		通道无效。对主通道来说，stCapRect 的宽与高、stDestSize 和 enCapSel 为静态属性，要改变它们必须先禁用主次两通道，且 stDestSize 的宽高必须与 stCapRect 的相同。改变主通道的 stDestSize 后，次通道的属性也应该重新设置，否则缩放系数没有更新，可能引起图像质量问题。 • 如果调用高级接口改变了默认的绑定关系，则必须先重新设置绑定关系。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	通道号范围	[0, VIU_MAX_CHN_NUM)
	通道属性参数限制	• 对于主属性，stCapRect 的宽与高为静态属性，其他项为动态属性；而对于次属性，stCapRect 以及帧率控制无意义，其他均为动态属性； • stCapRect 针对的是原始图像进行裁剪；stDestSize 的宽可为 stCapRect 宽的 1 倍或 1/2；stDestSize 的高在只采集单场时必须为 stCapRect 高的 1/2，采集两场时两者必须相同。 • stCapRect 中 s32Y 和 u32Height 隔行采集时必须 4 对齐。
Hi3518/Hi3516C	通道号范围	[0, VIU_MAX_PHYCHN_NUM)
	通道属性参数限制	• stCapRect 的宽与高为静态属性，其他项为动态属性；无次属性； • stCapRect 是针对原始图像进行裁剪；stDestSize 的宽必须等于 stCapRect 的宽；stDestSize 的高在只采集单场时必须为 stCapRect 高的 1/2，采集两场时两者必须相同。 • stCapRect 中 s32Y 和 u32Height 隔行采集时必须 4 对齐。 • 输出格式 enPixFormat 可以动态改变，但因为 Hi3518/Hi3516C 只支持 sp420 图像的 Rotate、LDC 处理，所以当 Rotate、LDC 使能时，如果动态切到 sp422，会导致

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	差异	
		无图像输出。 • 抓拍机方案中，建议不要使用帧率控制，否则会造成闪光信号触发的延时，不能及时响应抓拍。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

通道属性的各项配置限制如下：

- 采集区域 stCapRect：
  - 采集区域起始坐标用于配置需要采集的矩形图像相对于原始图像起始点的位置。起始点位置的横坐标以像素为单位，纵坐标以行为单位。
  - stCapRect 中 s32X 和 u32Width 必须 2 对齐；s32Y 和 u32Height 逐行采集时必须 2 对齐。
- 目标图像大小 stDestSize：
  - 必须配置，且大小不应该超出外围 ADC 输出图像的大小范围，否则可能导致 VI 硬件工作异常。
  - 当输入图像为隔行时，根据采单场或两场，将采集高度分别配置为单场或两场的高度，并要求两场采集时高度是 4 的整数倍。
  - 最小目标图像大小：最小采集宽高为 32x32。
- 抽场选择 enCapSel：
  - 抽场选择用于在原始图像为隔行输入时，用户可选择仅捕获其中的一场。
  - 为防止图像上下抖动现象，仅支持捕获两场或底场。
- 像素格式 enPixFormat：支持 semi-planar4:2:2 和 semi-planar4:2:0。
- 色度重采样 bChromaResample：是否进行色度重采样。
- 原始帧率 s32SrcFrameRate：如果不进行帧率控制，则该值设置为-1。
- 目标帧率 s32FrameRate：如果不进行帧率控制，则该值设置为-1，否则必须先设置原始帧率 s32SrcFrameRate 的值，且目标帧率必须小于原始帧率。
- 隔行图像不支持垂直翻转功能（因为开启该功能后顶底场采集顺序与原来的相反，后续模块处理可能会引起图像质量问题），MPP 内部不做此检查，由用户保证。

【举例】

请参见 HI_MPI_VI_SetDevAttr 的举例。

【相关主题】

HI_MPI_VI_GetChnAttr

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VI_GetChnAttr

【描述】

获取 VI 通道属性。

【语法】

```
HI_S32 HI_MPI_VI_GetChnAttr(VI_CHN ViChn,VI_CHN_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstAttr	VI 通道属性指针。 动态属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2) 内的奇数值无效
Hi3521/Hi3520A /Hi3520D/Hi3515A/Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

必须先设置通道属性再获取属性，否则将返回失败。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 s32ret;
VI_CHN ViChn = 0;
VI_CHN_ATTR_S stChnAttr;
/* first enable vi device and vi chn */
/* get channel attribute for vi chn */
s32ret = HI_MPI_VI_GetChnAttr(ViChn, &stChnAttr);
if (HI_SUCCESS != s32ret)
{
    printf("get vi chn attr err:0x%x\n", s32ret);
    return s32ret;
}
```

【相关主题】

HI_MPI_VI_SetChnAttr

HI_MPI_VI_SetChnMinorAttr

【描述】

设置 VI 通道次属性。

【语法】

```
HI_S32 HI_MPI_VI_SetChnMinorAttr(VI_CHN ViChn,const VI_CHN_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。 取值范围：[0, VIU_MAX_CHN_NUM)。	输入
pstAttr	VI 通道次属性指针。 动态属性，但 stCapRect 不起作用，由主属性的值决定。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	是否支持次属性	限制
Hi3531/Hi3532	否	不支持
Hi3521/Hi3520A /Hi3520D/Hi3515A/ Hi3515C	是	设置的 stCapRect 无意义，但受主属性 stCapRect 的限制。stDestSize 宽可为主属性 stCapRect 宽的 1 倍或 1/2，而高在只采集单场时必须为主属性 stCapRect 高的 1/2，采集两场时两者必须相同。
Hi35318/Hi3516C	否	不支持

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 默认情况下，VI 通道将按 HI_MPI_VI_SetChnAttr 接口中设置的分辨率来采集连续的视频帧图像；使用 HI_MPI_VI_SetChnMinorAttr 接口的目的是设置另外一个分辨率，VI 通道将会按照配置的帧率来轮流采集两种不同分辨率的视频图像。
- 所谓“通道次（要）属性”是相对主（要）属性而言的，主属性在函数 HI_MPI_VI_SetChnAttr 中配置；主、次属性的概念只有在对 VI 做了帧率控制时才有效，VI 采集图像时，按配置的目标帧率采集主属性大小的图像，其他帧则按次属性大小采集。

  例如：设置原始帧率为 25，目标帧率为 5，主属性大小为 D1，次属性大小为 CIF，则实际视频捕获则为 5 帧输出 D1 大小图像，剩余 20 帧输出 CIF 大小图像。主属性和次属性的配置关系如表 3-16 所示。

表3-16 主属性和次属性的配置关系

属性	AD	主属性	次属性
分辨率等	-	SetChnAttr	SetChnMinorAttr
帧率	SrcFrameRate	FrameRate	SrcFrameRate - FrameRate

- 如果已配置 VI 帧率，但未配置通道次属性，则按照目标帧率只输出主属性大小的视频图像。
- 如果已经配置了通道次属性，而中途需要关闭次属性图像采集，则可以调用 HI_MPI_VI_ClearChnMinorAttr 接口将此属性清除。
- 如果主属性源帧率大于 0，而目标帧率设为 0，表示不输出主属性图像，全部输出次属性图像。
- FrameRate 配置为与 SrcFrameRate 相等，则表示不输出次属性图像，全部输出主属性图像。
- 可以在 VI 通道启用后动态更改通道次属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

```
VI_CHN_ATTR_S stAttr;
/* set public attribute of VI device*/
/* ... ... */
/* enable VI device*/
/* ... ... */
StAttr.s32SrcFrameRate = 25;
StAttr.s32FrameRate = 5;

/* set main attribute of VI channel, size is D1 */
/* ... ... */
HI_MPI_VI_GetChnAttr(ViChn,&stAttr);
stAttr.enCapSel = VI_CAPSEL_BOTTOM;
stAttr.bDownScale = HI_TRUE;

/* set minor attribute of VI channel, size is CIF */
if (HI_MPI_VI_SetChnMinorAttr(ViChn,&stAttr))
{
    BBIT_ERR("set chn attr ex fail\n");
    return -1;
}

/* enable VI channel*/
/* ... ... */
/* if you not need minor attr capture */
if (HI_MPI_VI_ClearChnMinorAttr(ViChn))
{
    BBIT_ERR("clear chn minor attr fail\n");
    return -1;
}
```

【相关主题】

HI_MPI_VI_GetChnMinorAttr

HI_MPI_VI_GetChnMinorAttr

【描述】

获取 VI 通道次属性。

【语法】

```
HI_S32 HI_MPI_VI_GetChnMinorAttr(VI_CHN ViChn, VIDEO_FRAME_INFO_S
*pstAttr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ViChn	VI 通道号。 取值范围：[0, VIU_MAX_CHN_NUM)。	输入
pstAttr	VI 通道属性指针。 动态属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持次属性
Hi3531/Hi3532	否
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	是
Hi3518/Hi3516C	否

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

必须先设置通道属性再获取属性，否则将返回 HI_ERR_VI_FAILED_NOTCONFIG。

【举例】

请参见 HI_MPI_VI_SetChnMinorAttr 的举例。

【相关主题】

HI_MPI_VI_SetChnMinorAttr

HI_MPI_VI_ClearChnMinorAttr

【描述】

清除 VI 通道次属性。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_VI_ClearChnMinorAttr(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。 取值范围：[0, VIU_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持次属性
Hi3531/Hi3532	否
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	是
Hi3518/Hi3516C	否

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 如果已经配置通道次属性，可以调用此接口将其取消。
- 本接口可以在通道启用状态下调用。

【举例】

请参见 HI_MPI_VI_SetChnMinorAttr 的举例。

【相关主题】

- HI_MPI_VI_SetChnMinorAttr
- HI_MPI_VI_GetChnMinorAttr

HI_MPI_VI_EnableChn

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

启用 VI 通道。

【语法】

```
HI_S32 HI_MPI_VI_EnableChn(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 必须先设置通道属性，且通道所绑定的 VI 设备必须使能。
- 如果设置了 VI 与 VO/VENC 的绑定关系，成功调用该接口后即可得到视频数据。
- 可重复启用 VI 通道，不返回失败。

【举例】

请参见 HI_MPI_VI_SetDevAttr 的举例。

【相关主题】

HI_MPI_VI_DisableChn

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VI_DisableChn

【描述】

禁用 VI 通道。

【语法】

```
HI_S32 HI_MPI_VI_DisableChn(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 禁用 VI 通道后，此 VI 通道即停止采集视频输入数据，如果已经绑定 VO 或 VENC，则 VO 或 VENC 将不会再接收到视频图像。
- 可重复禁用 VI 通道，不返回失败。

【举例】

请参见 HI_MPI_VI_SetDevAttr 的举例。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VI_EnableChn

HI_MPI_VI_EnableChnInterrupt

【描述】

启用 VI 通道响应硬件中断处理程序。

【语法】

```
HI_S32 HI_MPI_VI_EnableChnInterrupt(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	描述
Hi3531/Hi3532	• 由于主次通道在硬件上公用一个中断 MASK 配置，故恢复主(次)通道响应硬件中断处理程序的同时，次(主)通道的也会恢复（如果该通道启用的话）。 • 通道号范围：[0, VIU_MAX_CHN_NUM-2)，其中[VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	通道号范围：[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	通道号范围：[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 该接口为高级接口，一般不需调用。
- 主要应用场景：当应用程序自行检测到前端 AD 时序发生异常，启用插入用户图片功能时，为了减少因时序异常导致硬件上报大量中断造成的系统资源消耗，调用 HI_MPI_VI_DisableChnInterrupt 接口，使 VI 不再响应该通道的硬件中断处理程序。当检测到前端 AD 时序恢复正常时，可调用本接口恢复 VI 响应该通道的硬件中断处理程序。
- 必须先启用 VI 通道，否则程序返回错误。
- 可重复启用 VI 通道中断，不返回失败。

【举例】

无。

【相关主题】

HI_MPI_VI_DisableChnInterrupt

HI_MPI_VI_DisableChnInterrupt

【描述】

禁止 VI 通道响应硬件中断处理程序。

【语法】

```
HI_S32 HI_MPI_VI_DisableChnInterrupt(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	描述
Hi3531/Hi3532	• 由于主次通道在硬件上公用一个中断 MASK 配置，故禁止主(次)通道响应硬件中断处理程序的同时，次(主)通道的也会被禁止（如果该通道启用的话）。 • 通道号范围：[0, VIU_MAX_CHN_NUM-2)，其中[VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	通道号范围：[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	通道号范围：[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 该接口为高级接口，一般不需调用。
- 主要应用场景：当应用程序自行检测到前端 AD 时序发生异常，启用插入用户图片功能时，为了减少因时序异常导致硬件上报大量中断造成的系统资源消耗，可调用本接口，使 VI 不再响应该通道的硬件中断处理程序。当检测到前端 AD 时序恢复正常时，可调用 HI_MPI_VI_EnableChnInterrupt 接口恢复 VI 响应该通道的硬件中断处理程序。
- 如果通道未启用，则无需禁止 VI 通道响应中断处理程序，调用本接口将直接返回成功。
- 可重复禁止，不返回失败。

【举例】

无。

【相关主题】

HI_MPI_VI_EnableChnInterrupt

HI_MPI_VI_SetFrameDepth

【描述】

设置可获取的 VI 图像最大深度。

【语法】

```
HI_S32 HI_MPI_VI_SetFrameDepth(VI_CHN ViChn, HI_U32 u32Depth);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
u32Depth	可获取的 VI 图像最大深度。 默认值：0。 取值范围：[0, 8]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM)，其中[VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 此接口用于设置某一 VI 通道缓存的视频图像帧数。当用户设置缓存多帧视频图像时，用户可以获取到一定数目的连续图像数据。
- 若指定 u32Depth 为 0，表示不需要系统为该 VI 通道缓存图像，故用户获取不到该 VI 通道图像数据。系统默认不为 VI 通道缓存图像，即 u32Depth 默认为 0。
- 若指定 u32Depth>0，系统将为该 VI 通道缓存 u32Depth 个图像，用户可通过接口 HI_MPI_VI_GetFrame 获取该 VI 通道图像数据。具体分为以下几种情况：
  - 用户一直不获取图像。

    系统将自动更新最旧的图像数据，保证用户一旦开始获取，就可获取到最近最新的 u32Depth 个连续图像。
  - 用户连续获取 u32Depth 次并一直不释放。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

系统因获取不到 VB 而自动停止缓存新的 VI 图像，用户也不能获取新的 VI 图像。故建议用户保证获取和释放接口配对使用。

- 用户获取/释放的速度比 VI 通道产生图像的速度慢。

  系统将自动更新用户仍未获取的最旧的图像数据，保证缓存的图像队列为最近的新 VI 图像。由于用户不能保证获取速度，导致获取的可能不是连续图像。

- 系统为每个 VI 通道缓存的 u32Depth 个 VI 图像数据，占用 MPP 内部的缓存块（简称 VB）。故用户应通过接口 HI_MPI_VB_SetConf 设置足够的 VB，否则可能由于系统缓存图像占用过多 VB，从而影响 VI 的正常图像采集，导致用户获取不到 VI 图像数据。支持动态调整 u32Depth。如用户在不需要获取该 VI 通道数据时，可设置 u32Depth 为 0，以减少 VI 通道占用内部 VB 的数量；在需要获取时，再设置 u32Depth 为合适的值，即可获取到设置时刻后的连续 VI 图像。

【举例】

请参见 HI_MPI_VI_GetFrame 的举例。

【相关主题】

HI_MPI_VI_GetFrameDepth

HI_MPI_VI_GetFrameDepth

【描述】

获取设置的 VI 图像最大深度。

【语法】

```
HI_S32 HI_MPI_VI_GetFrameDepth(VI_CHN ViChn, HI_U32 *pu32Depth);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pu32Depth	U32 类型的指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_GetFrame

【描述】

获取 VI 采集的图像。

【语法】

```
HI_S32 HI_MPI_VI_GetFrame(VI_CHN ViChn, VIDEO_FRAME_INFO_S *pstFrameInfo);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstFrameInfo	VI 帧信息结构指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 此接口可以获取指定 VI 通道的视频图像信息。图像信息主要包括：图像的宽度、高度、像素格式、时间戳以及 YUV 各分量的物理地址。
- 此接口需在通道已启用后才有效。
- 支持多次获取后再释放，但建议获取和释放接口配对使用。
- 获取的物理地址信息来自 MPP 内部使用的 VideoBuffer，因此使用完之后，必须要调用 HI_MPI_VI_ReleaseFrame 接口释放其内存。
- pstFrameInfo -> stVFrame .u32PhyAddr[0]和 pstFrameInfo -> stVFrame .u32PhyAddr[1]分别指向图像的亮度分量和色度分量的物理地址。
- 该接口默认超时时间为 2s，即 2s 内仍未获取到图像，则超时返回。

【举例】

```
HI_S32 s32ret;
VI_CHN ViChn = 0;
VIDEO_FRAME_INFO_S stFrame;
HI_U32 u32Depth;

/* set max depth */
u32Depth = 10;
s32ret = HI_MPI_VI_SetFrameDepth(ViChn, u32Depth);
if (HI_SUCCESS != s32ret)
{
    printf("set max depth err:0x%x\n", s32ret);
    return s32ret;
}
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
/* get video frame from vi chn */
s32ret = HI_MPI_VI_GetFrame(ViChn, &stFrame)
if (HI_SUCCESS != s32ret)
{
    printf("get vi frame err:0x%x\n", s32ret);
    return s32ret;
}

/* deal with video frame ... */

/* release video frame */
(void)HI_MPI_VI_ReleaseFrame(ViChn, &stFrame);
```

【相关主题】

- HI_MPI_VI_ReleaseFrame
- HI_MPI_VI_GetFrameTimeOut

HI_MPI_VI_GetFrameTimeOut

【描述】

在设定时间内获取 VI 采集的图像。

【语法】

```
HI_S32 HI_MPI_VI_GetFrameTimeOut(VI_CHN ViChn, VIDEO_FRAME_INFO_S
*pstFrameInfo, HI_U32 u32MilliSec);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstFrameInfo	VI 帧信息结构指针。	输出
u32MilliSec	超时时间，单位：ms	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 此接口可以获取指定 VI 通道的视频图像信息。图像信息主要包括：图像的宽度、高度、像素格式、时间戳以及 YUV 各分量的物理地址。
- 此接口需在通道已启用后才有效。
- 支持多次获取后再释放，但建议获取和释放接口配对使用。
- 获取的物理地址信息来自 MPP 内部使用的 VideoBuffer，因此使用完之后，必须要调用 HI_MPI_VI_ReleaseFrame 接口释放其内存。
- pstFrameInfo -> stVFrame .u32PhyAddr[0]和 pstFrameInfo -> stVFrame .u32PhyAddr[1]分别指向图像的亮度分量和色度分量的物理地址。
- u32MilliSec 参数的单位是毫秒，指超时时间。即在 u32MilliSec 毫秒内，如果获取不到图像，则函数返回。当 u32MilliSec 设为 0 时，表示阻塞模式，程序一直等待，直到获取到图像才返回。

【举例】

无。

【相关主题】

- HI_MPI_VI_ReleaseFrame
- HI_MPI_VI_GetFrame

HI_MPI_VI_ReleaseFrame

【描述】

释放 VI 图像数据所占的缓存。

【语法】

```
HI_S32 HI_MPI_VI_ReleaseFrame(VI_CHN ViChn, VIDEO_FRAME_INFO_S
*pstFrameInfo);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstFrameInfo	VI 帧数据存储结构指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 用户必须保证 pstFrameInfo 结构中的信息与获取时一致，否则会造成释放不成功。
- 要求用户使用完获取原始图像接口后，调用本接口释放原始图像数据（即必须与 HI_MPI_VI_GetFrame 或 HI_MPI_VI_GetFrameTimeOut 配对使用）。

【举例】

请参见 HI_MPI_VI_GetFrame 的举例。

【相关主题】

- HI_MPI_VI_GetFrame
- HI_MPI_VI_GetFrameTimeOut

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VI_SetUserPic

【描述】

设置用户图片，作为无视频信号时的插入图片。

【语法】

```
HI_S32 HI_MPI_VI_SetUserPic(VI_CHN ViChn, VI_USERPIC_ATTR_S *pstUsrPic);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstUsrPic	用户图片信息结构指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 此接口支持两种用户图片模式，一种是 YUV 图像模式，另一种是纯色背景模式。
- 设置 VI 用户图片，用于将 VI 通道输出的视频帧图像修改为用户配置的指定 YUV 图像数据或纯色背景，而不是 codec 的视频数据；此接口需要与 HI_MPI_VI_EnableUserPic、HI_MPI_VI_DisableUserPic 配合使用。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目前此接口中的参数 VI 设备和通道号未被使用，即设置用户图像针对的是所有 VI 通道，而不是具体某个 VI 通道。

- 设置完用户图片后，即可调用 HI_MPI_VI_EnableUserPic 接口对指定 VI 通道启用插入用户图片，此时 VI 通道输出的数据即为所配置的图片 YUV 数据或纯色背景；一般用于视频信号丢失时，VI 通道输出 NoVideo 图片。

  配置的用户图片的宽、高大小范围为[32，4096)，一般应该与 VI 通道大小相一致；如果不一致，VI 内部会自动将其缩放为 VI 通道大小。在 VI 通道混合采集时，建议把用户图片大小设为与高分辨率一致，其格式设为 INTERLACED，否则 VO 预览时可能会抖动。

- 用户图片的视频帧信息结构中，需要设置图片的宽度、高度、行间隔、YUV 格式以及 Y 分量和 C 分量数据的物理地址；可以从 MPP 公共缓冲池中获取一块相应大小的视频缓存块，从缓存块信息中得到存放 YUV 数据的物理地址，然后将物理地址映射到用户空间，即可对这块内存进行 YUV 数据的填充操作；**注意只支持 semi-planarYUV420、semi-planarYUV422 的格式，因此填充数据时需要遵循先存 Y 分量数据，再存 UV 分量间插数据的存储顺序（小端字节序先 V 后 U）。**

  VIU 模块在启用插入用户图片时，会直接使用设置的用户图片帧信息中的物理地址，因此设置完用户图片后，不应该释放或销毁其视频缓存块，除非确认不再使用。可以通过再次调用此接口设置另外一块 VideoBuffer 以修改图片信息。

- 启用 VI 通道或启用插入用户图片时，此接口都可以动态调用。

【举例】

```
HI_S32  s32ret;
   HI_U32  u32Width;
   HI_U32  u32Height;
   HI_U32  u32LStride;
   HI_U32  u32CStride;
   HI_U32  u32LumaSize;
   HI_U32  u32ChrmSize;
   HI_U32  u32Size;
   VB_BLK  VbBlk;
   HI_U32  u32PhyAddr;
   HI_U8   *pVirAddr;
    /* you need get width and height of pictrue */
   u32LumaSize = (u32LStride * u32Height);
   u32ChrmSize = (u32CStride * u32Height) >> 2;/* 420*/
   u32Size = u32LumaSize + (u32ChrmSize << 1);

   /* get video buffer block form common pool */
   VbBlk = HI_MPI_VB_GetBlock(VB_INVALID_POOLID, u32Size);
   if (VB_INVALID_HANDLE == VbBlk)
   {
      return -1;
   }
   /* get physical address*/
   u32PhyAddr = HI_MPI_VB_Handle2PhysAddr(VbBlk);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    if (0 == u32PhyAddr)
    {
return -1;
    }

    /* mmap physical address to virtual address*/
    /* ... ... */

    /* get pool id */
    pstVFrameInfo->u32PoolId = HI_MPI_VB_Handle2PoolId(VbBlk);
    if (VB_INVALID_POOLID == pstVFrameInfo->u32PoolId)
    {
        return -1;
    }

    pstVFrameInfo->stVFrame.u32PhyAddr[0] = u32PhyAddr;
    pstVFrameInfo->stVFrame.u32PhyAddr[1] = pstVFrameInfo-
>stVFrame.u32PhyAddr[0] + u32LumaSize;
    pstVFrameInfo->stVFrame.u32PhyAddr[2] = pstVFrameInfo-
>stVFrame.u32PhyAddr[1] + u32ChrmSize;

    pstVFrameInfo->stVFrame.pVirAddr[0] = pVirAddr;
    pstVFrameInfo->stVFrame.pVirAddr[1] = pstVFrameInfo-
>stVFrame.pVirAddr[0] + u32LumaSize;
    pstVFrameInfo->stVFrame.pVirAddr[2] = pstVFrameInfo-
>stVFrame.pVirAddr[1] + u32ChrmSize;

    pstVFrameInfo->stVFrame.u32Width  = u32Width;
    pstVFrameInfo->stVFrame.u32Height = u32Height;
    pstVFrameInfo->stVFrame.u32Stride[0] = u32LStride;
    pstVFrameInfo->stVFrame.u32Stride[1] = u32CStride;
    pstVFrameInfo->stVFrame.u32Stride[2] = u32CStride;
    pstVFrameInfo->stVFrame.enPixelFormat =
PIXEL_FORMAT_YUV_SEMIPLANAR_420;

    /* now you need get YUV Semi Palnar Data ,fill them to the virtual
address */
    /* ... ... */
    /* ... ... */

    /* enabel VI channel ... ... */

    /* first set user pic info*/
    s32ret = HI_MPI_VI_SetUserPic(0, pstVFrameInfo);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    if (s32ret)
    {
return -1;
    }

    /* ... ... */

    /* enable insert user pic if you need */
    s32ret = HI_MPI_VI_EnableUserPic(0);
    if (s32ret)
    {
        return -1;
    }

    /* ... ... */

    /* disable insert user pic if you don't need */
    s32ret = HI_MPI_VI_DisableUserPic(0, 0);
    if (s32ret)
    {
        return -1;
    }
```

【相关主题】

无。

HI_MPI_VI_EnableUserPic

【描述】

启用插入用户图片。

【语法】

```
HI_S32 HI_MPI_VI_EnableUserPic(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 启用插入用户图片之前，需要先设置用户图片帧信息。
- 启用插入用户图片之后，当前 VI 通道即输出配置的用户图片帧数据。
- 此接口可以重复调用。

【举例】

请参见 HI_MPI_VI_SetUserPic 的举例。

【相关主题】

无。

HI_MPI_VI_DisableUserPic

【描述】

禁用插入用户图片。

【语法】

```
HI_S32 HI_MPI_VI_DisableUserPic(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中[VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- VI 通道不再需要输出用户图片时，应该调用此接口以恢复输出 AD 的原始视频数据。
- 此接口可以重复调用。

【举例】

请参见 HI_MPI_VI_SetUserPic 的举例。

【相关主题】

无。

HI_MPI_VI_EnableCascade

【描述】

启用 VI 设备的级联模式。

【语法】

```
HI_S32 HI_MPI_VI_EnableCascade(VI_DEV ViDev);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)，且 ViDev 为偶数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	只有设备 0/2/4/6 支持级联模式，相应的只有物理通道 2xViDev（即 0/4/8/12/通道）支持级联模式。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	不支持级联模式，调用此接口会返回出错
Hi3518/Hi3516C	不支持级联模式，调用此接口会返回错误码

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 启用设备级联模式前必须已经设置设备属性，否则返回失败。
- 级联模式只支持 BT1120 逐行时序。
- 可重复启用，不返回失败。

【举例】

```
HI_S32 s32Ret;
    VI_DEV ViDev = 2;
    VI_CHN ViChn = 4, ViCasChn = VI_CAS_CHN_1;
    VI_DEV_ATTR_S stDevAttr;
    VI_CHN_ATTR_S stChnAttr;

    memset(&stDevAttr, 0, sizeof(VI_DEV_ATTR_S));
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    stDevAttr.enIntfMode = VI_MODE_BT1120_STANDARD;
    stDevAttr.enWorkMode = VI_WORK_MODE_1Multiplex;
    stDevAttr.au32CompMask[0] = 0xFF00;
    stDevAttr.au32CompMask[1] = 0xFF;
    stDevAttr.enScanMode = VI_SCAN_PROGRESSIVE;
    stDevAttr.s32AdChnId[0] = -1;
    stDevAttr.s32AdChnId[1] = -1;
    stDevAttr.s32AdChnId[2] = -1;
    stDevAttr.s32AdChnId[3] = -1;

    s32Ret = HI_MPI_VI_SetDevAttr(ViDev, &stDevAttr);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Set dev attributes failed with error code %#x!\n", s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VI_EnableDev(ViDev);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Enable dev failed with error code %#x!\n", s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VI_EnableCascade(ViDev);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Enable dev cascade mode failed with error code %#x!\n",
s32Ret);
        return HI_FAILURE;
    }

    stChnAttr.stCapRect.s32X = 0;
    stChnAttr.stCapRect.s32Y = 0;
    stChnAttr.stCapRect.u32Width = 1920;
    stChnAttr.stCapRect.u32Height = 1080;
    stChnAttr.stDestSize.u32Width = 1920;
    stChnAttr.stDestSize.u32Height = 1080;
    stChnAttr.enCapSel = VI_CAPSEL_BOTH;
    stChnAttr.enPixFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_422;
    stChnAttr.bMirror = HI_FALSE;
    stChnAttr.bFlip = HI_FALSE;
    stChnAttr.bChromaResample = HI_FALSE
    stChnAttr.s32SrcFrameRate = -1;
    stChnAttr.s32FrameRate = -1;
    s32Ret = HI_MPI_VI_SetChnAttr(ViChn,&stChnAttr);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    if (s32Ret != HI_SUCCESS)
    {
        printf("Set chn attributes failed with error code %#x!\n", s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VI_EnableChn(ViChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Enable chn failed with error code %#x!\n", s32Ret);
        return HI_FAILURE;
    }
    /* here, we suppose it's VO_CAS_MODE_SINGLE cascade mode */
    s32Ret = HI_MPI_VI_EnableCascadeChn(ViCasChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Enable cascade chn failed with error code %#x!\n", s32Ret);
        return HI_FAILURE;
    }
    /* now, vi is capturing images, you can do something else ... */

    s32Ret = HI_MPI_VI_DisableCascadeChn(ViCasChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Disable cascade chn failed with error code %#x!\n",
s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VI_DisableChn(ViChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Disable chn failed with error code %#x!\n", s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VI_DisableCascade(ViDev);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Disable dev cascade mode failed with error code %#x!\n",
s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VI_DisableDev(ViDev);
    if (s32Ret != HI_SUCCESS)
    {
        printf("Disable dev failed with error code %#x!\n", s32Ret);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
        return HI_FAILURE;
    }
```

【相关主题】

HI_MPI_VI_EnableCascade

HI_MPI_VI_DisableCascade

【描述】

关闭 VI 设备级联模式。

【语法】

```
HI_S32 HI_MPI_VI_DisableCascade(VI_DEV ViDev);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。 取值范围：[0, VIU_MAX_DEV_NUM)，且 ViDev 为偶数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持级联模式
Hi3531/Hi3532	是
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否
Hi3518/Hi3516C	否

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 必须先禁用所有使能的级联扩展通道及物理通道后，才能调用此接口关闭 VI 设备的级联模式。
- 可重复禁用级联模式，不返回失败。

【举例】

参见 HI_MPI_VI_EnableCascade 举例。

【相关主题】

HI_MPI_VI_EnableCascade

HI_MPI_VI_EnableCascadeChn

【描述】

启用 VI 级联扩展通道。

【语法】

```
HI_S32 HI_MPI_VI_EnableCascadeChn(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 级联扩展通道号。 取值范围：{VI_CAS_CHN_1, VI_CAS_CHN_2}。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持级联
Hi3531/Hi3532	是
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否
Hi3518/Hi3516C	否

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- 必须先调用 HI_MPI_VI_EnableCascade 接口使能设备级联模式。
- 级联扩展通道只是软件上的概念，必须先使能级联物理通道再调用此接口使能扩展通道。
- 如果用户使能了设备的级联模式，但不使能级联扩展通道，VI 将不会采集、发送图像。
- 扩展通道 VI_CAS_CHN_1 固定接收级联单路模式的图像和双路模式下的奇数帧，VI_CAS_CHN_2 固定接收级联双路模式下的偶数帧。
- 如果用户不使能级联模式，却输入双路级联视频数据，VI 将不对两路数据进行区分处理，只是简单的当成一路数据来采集、发送。
- 可重复启用 VI 级联扩展通道，不返回失败。

【举例】

请参见 HI_MPI_VI_EnableCascade 举例

【相关主题】

HI_MPI_VI_DisableCascadeChn

HI_MPI_VI_DisableCascadeChn

【描述】

禁用 VI 级联扩展通道。

【语法】

```
HI_S32 HI_MPI_VI_DisableCascadeChn(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 级联扩展通道号。 取值范围：{VI_CAS_CHN_1, VI_CAS_CHN_2}。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	是否支持级联
Hi3531/Hi3532	是
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否
Hi3518/Hi3516C	否

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 禁用 VI 级联扩展通道后，此时 VI 物理通道即停止采集该扩展通道对应的级联奇偶帧数据，如果已经绑定 VO 或 VENC，则 VO 或 VENC 将不会再接收到视频图像。
- 可重复禁用 VI 级联扩展通道，不返回失败。

【举例】

请参见 HI_MPI_VI_EnableCascade 举例

【相关主题】

HI_MPI_VI_EnableCascadeChn

HI_MPI_VI_ChnBind

【描述】

高级接口，设置 VI 通道绑定关系。

【语法】

```
HI_S32 HI_MPI_VI_ChnBind(VI_CHN ViChn, const VI_CHN_BIND_ATTR_S
*pstChnBindAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstChnBindAttr	VI 通道绑定属性指针。 静态属性。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持显式设置绑定关系	通道号范围
Hi3531/Hi3532	是	[0, (VIU_MAX_CHN_NUM-2)/2)
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 通常不需调用该接口，特殊情况，需要调整默认绑定关系，或使多个通道绑定到同一个{Dev，Way}时，可调用此接口来实现。但是，调用此接口前，必须先调用 HI_MPI_VI_ChnUnBind 接口解开默认绑定关系，再设置新的绑定关系。
- 必须先设置设备属性，再设置 VI 通道的绑定关系
- 不允许重复绑定。如果已设定绑定关系必须先解除，再重新设定新的绑定关系。
- 绑定关系为静态属性，设置绑定关系时必须先禁用主次通道。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_ChnUnBind

【描述】

高级接口，解除 VI 通道绑定关系。

【语法】

```
HI_S32 HI_MPI_VI_ChnUnBind(VI_CHN ViChn);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持显式设置绑定关系	通道号范围
Hi3531/Hi3532	是	[0, (VIU_MAX_CHN_NUM-2)/2)
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 通常不需调用该接口，如果要调整默认绑定关系，则调用此接口解绑定。
- 主次通道必需在禁止（Disable）状态下才能解绑定。
- 重复解绑定，不返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_GetChnBind

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

获取 VI 通道绑定关系。

【语法】

```
HI_S32 HI_MPI_VI_GetChnBind(VI_CHN ViChn, VI_CHN_BIND_ATTR_S
*pstChnBindAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstChnBindAttr	VI 通道绑定属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持显式设置绑定关系	通道号范围
Hi3531/Hi3532	是	[0, (VIU_MAX_CHN_NUM-2)/2)
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

必须先设置 VI 通道对应的 Dev 属性，才能获取其默认的绑定关系。

【举例】

无。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

HI_MPI_VI_GetFd

【描述】

获取 VI 通道对应的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_VI_GetFd(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【芯片差异】

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_CHN_NUM)

【错误码】

无。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

用户可以获取文件句柄实现多通道 select 获取视频帧数据。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

```
VI_CHN ViChn = 0;
HI_S32 s32ViFd = 0;

/* enable vi dev and vi chn */

/* get video frame from vi chn */
s32ViFd = HI_MPI_VI_GetFd(ViChn);
if (s32ViFd <= 0)
{
    return HI_FAILURE;
}
```

【相关主题】

无。

HI_MPI_VI_Query

【描述】

查询 VI 通道的中断计数、平均帧率等信息。

【语法】

```
HI_S32 HI_MPI_VI_Query(VI_CHN ViChn, VI_CHN_STAT_S *pstStat);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
pstStat	通道信息结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	通道号范围
Hi3531/Hi3532	[0, VIU_MAX_CHN_NUM-2)，其中 [VIU_MAX_CHN_NUM / 2, VIU_MAX_CHN_NUM-2)内的奇数值无效
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VIU_MAX_CHN_NUM)
Hi3518/Hi3516C	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 该接口可查询中断计数、通道使能状态、平均帧率、中断丢失数、获取 VB 失败次数、图像宽高等信息。
- 可以通过该接口查询通道当前的中断计数，用于无中断检测。
- 通过该接口获取到的帧率是每 10 秒钟的平均帧率，即 VI 会每隔十秒统计一次平均帧率，该值并不精确。
- 用户可通过该接口查询中断丢失数，如果该数值一直在增加，说明 VI 工作出现异常。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_SetFlashConfig

【描述】

设置 VI 闪光灯的配置信息。

【语法】

```
HI_S32 HI_MPI_VI_SetFlashConfig(VI_DEV ViDev,const VI_FLASH_CONFIG_S
*pstFlashConfig);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入
pstFlashConfig	闪光灯配置信息指针。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持闪光灯	设备号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_DEV_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 闪光灯模式有单闪、频闪两种模式，频闪模式下 u32Interval 必须大于 u32CapFrmIndex。
- 闪光灯配置的详细说明请参考 VI_FLASH_CONFIG_S。

【举例】

```
HI_S32 s32Ret;
VI_DEV ViDev = 0;
VI_FLASH_CONFIG_S stFlashConfig;

/* first enable vi device and vi chn */

/* Init flash config */
stFlashConfig.enFlashMode = VI_FLASH_ONCE;
stFlashConfig.u32StartTime = 100;
stFlashConfig.u32Duration = 400;
stFlashConfig.u32CapFrmIndex = 2;

/* Set configuration for vi flash */
s32Ret = HI_MPI_VI_SetFlashConfig(ViDev, &stFlashConfig);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
if (s32Ret != HI_SUCCESS)
{
    printf("Set flash config failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
}

/* Get configuration for vi flash */
s32Ret = HI_MPI_VI_GetFlashConfig(ViDev, &stFlashConfig);
if (s32Ret != HI_SUCCESS)
{
    printf("Get flash config failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
}
s32Ret = HI_MPI_VI_FlashTrigger(ViDev, HI_TRUE);
if (s32Ret != HI_SUCCESS)
{
    printf("Trigger flash failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

- HI_MPI_VI_GetFlashConfig
- HI_MPI_VI_FlashTrigger

HI_MPI_VI_GetFlashConfig

【描述】

获取 VI 闪光灯的配置信息。

【语法】

```
HI_S32 HI_MPI_VI_GetFlashConfig(VI_DEV ViDev, VI_FLASH_CONFIG_S
*pstFlashConfig);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入
pstFlashConfig	闪光灯配置信息指针。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持闪光灯	设备号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_DEV_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

必须先设置闪光灯配置信息后再获取，否则将返回失败。

【举例】

无。

【相关主题】

- HI_MPI_VI_SetFlashConfig
- HI_MPI_VI_FlashTrigger

HI_MPI_VI_FlashTrigger

【描述】

关闭/启用 VI 闪光灯。

【语法】

```
HI_S32 HI_MPI_VI_FlashTrigger(VI_DEV ViDev, HI_BOOL bEnable);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
bEnable	关闭或者启用闪光灯。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持闪光灯	设备号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_DEV_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

必须先配置闪光灯属性后才能启用/关闭闪光灯，否则将返回失败。

【举例】

无。

【相关主题】

- HI_MPI_VI_SetFlashConfig
- HI_MPI_VI_GetFlashConfig

HI_MPI_VI_SetExtChnAttr

【描述】

设置扩展通道属性。

【语法】

```
HI_S32 HI_MPI_VI_SetExtChnAttr(VI_CHN ViChn, const VI_EXT_CHN_ATTR_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
*pstExtChnAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 扩展通道号。	输入
pstExtChnAttr	扩展通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 ExtChn	扩展通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	是	[VIU_EXT_CHN_START, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

扩展通道属性的各项配置限制如下：

- 目标图像大小 stDestSize：
  - 必须配置，且大小不能超出扩展通道的缩放能力，否则扩展通道会无输出。扩展通道支持的缩小倍数为小于 13，且当输入数据格式为 SP422，输出数据格式设置为 SP420 时，扩展通道垂直方向上仅支持 8 倍缩小；扩展通道支持的输出宽高为小于 4096。
  - u32Width 必须 2 对齐； u32Height 逐行采集时必须 2 对齐，隔行且采集两场时必须 4 对齐。
  - 像素格式 enPixFormat：支持 semi-planar4:2:2 和 semi-planar4:2:0。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 原始帧率 s32SrcFrameRate：如果不进行帧率控制，则该值设置为-1。
- 目标帧率 s32FrameRate：如果不进行帧率控制，则该值设置为-1，否则必须先设置原始帧率 s32SrcFrameRate 的值，且目标帧率必须小于原始帧率。
- 扩展通道的输出帧率是其所绑定的父通道和扩展通道本身帧率控制叠加的结果。如 Sensor 的源帧率是 30 帧，扩展通道的父通道是物理通道 0，物理通道 0 帧率控制的源帧率和目标帧率分别是 30 和 10，扩展通道帧率控制的源帧率和目标帧率是 10 和 2，那么最后扩展通道的实际帧率是 2 帧/秒。
- 父通道上绑定的扩展通道不能超过最大数目，否则设置失败。

【举例】

```
HI_S32 s32Ret;
VI_CHN ViChn = VIU_EXT_CHN_START;
VI_EXT_CHN_ATTR_S stExtChnAttr;

/* First enable vi device and vi chn */

/* Init ext-chn attr */
stExtChnAttr.enPixFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_420;
stExtChnAttr.s32BindChn = 0;
stExtChnAttr.s32FrameRate = -1;
stExtChnAttr.s32SrcFrameRate = -1;
stExtChnAttr.stDestSize.u32Width = 720;
stExtChnAttr.stDestSize.u32Height = 576;

/* Set attribute for vi ext-chn */
HI_MPI_VI_SetExtChnAttr(ViExtChn, &stExtChnAttr);
if (HI_SUCCESS != s32ret)
{
    printf("Set vi ext-chn attr err:0x%x\n", s32ret);
    return s32Ret;
}

/* Get attribute for vi ext-chn */
s32Ret = HI_MPI_VI_GetExtChnAttr (ViChn, & stExtChnAttr);
if (HI_SUCCESS != s32ret)
{
    printf("Get vi ext-chn attr err:0x%x\n", s32ret);
    return s32Ret;
}
```

【相关主题】

HI_MPI_VI_GetExtChnAttr

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VI_GetExtChnAttr

【描述】

获取 VI 扩展通道属性。

【语法】

```
HI_S32 HI_MPI_VI_GetExtChnAttr(VI_CHN ViChn, VI_EXT_CHN_ATTR_S
*pstExtChnAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 扩展通道号。	输入
pstExtChnAttr	VI 扩展通道属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 ExtChn	扩展通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	是	[VIU_EXT_CHN_START, VIU_MAX_CHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

必须先设置扩展通道属性后再获取，否则将返回失败。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VI_SetExtChnAttr

HI_MPI_VI_SetLDCAttr

【描述】

设置 VI 镜头畸变校正（LDC）属性。

【语法】

```
HI_S32 HI_MPI_VI_SetLDCAttr(VI_CHN ViChn, const VI_LDC_ATTR_S
*pstLDCAttr);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 物理通道号。	输入
pstLDCAttr	LDC 属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 LDC	物理通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 该接口必须在设置通道属性后调用。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- LDC 属性可动态改变。
- 仅支持非场输出的 sp420 图像。
- 仅支持物理通道的 LDC 设置。
- 启用 LDC 时，VI 的 COVER 可能会变形，此时建议使用 COVEREX 来代替 COVER。

【举例】

```
HI_S32 s32Ret;
VI_LDC_ATTR_S stLDCAttr;
/* first enable vi device and vi chn */
/* Init LDC attr */
stLDCAttr.bEnable = HI_TRUE;
stLDCAttr.stAttr.enViewType = LDC_VIEW_TYPE_ALL;   //LDC_VIEW_TYPE_CROP;
stLDCAttr.stAttr.s32CenterXOffset = 0;
stLDCAttr.stAttr.s32CenterYOffset = 0;
stLDCAttr.stAttr.s32Ratio = 255;

/* Set attribute for vi LDC */
s32Ret = HI_MPI_VI_SetLDCAttr(ViChn, &stLDCAttr);
if (HI_SUCCESS != s32ret)
{
    printf("Set vi LDC attr err:0x%x\n", s32ret);
    return s32Ret;
}
/* Get attribute for vi LDC */
s32Ret = HI_MPI_VI_GetLDCAttr (ViChn, &stLDCAttr);
if (HI_SUCCESS != s32ret)
{
    printf("Get vi LDC attr err:0x%x\n", s32ret);
    return s32Ret;
}
```

【相关主题】

HI_MPI_VI_GetLDCAttr

HI_MPI_VI_GetLDCAttr

【描述】

获取 VI 镜头畸变校正（LDC）属性。

【语法】

```
HI_S32 HI_MPI_VI_GetLDCAttr(VI_CHN ViChn, VI_LDC_ATTR_S *pstLDCAttr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ViChn	VI 物理通道号。	输入
pstLDCAttr	LDC 属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 LDC	物理通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_PHYCHN_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VI_SetLDCAttr

HI_MPI_VI_SetCSCAttr

【描述】

设置 VI 设备的 CSC 属性。

【语法】

```
HI_S32 HI_MPI_VI_SetCSCAttr(VI_DEV ViDev, const VI_CSC_ATTR_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
*pstCSCAttr);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入
pstCSCAttr	CSC 属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 CSC	设备号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_DEV_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 该接口要在设备属性设置后再调用；
- 该接口主要用于图像的亮度、色调、对比度、饱和度的调节。
- 如果是高清摄像头输入，建议 enViCscType 配置为 VI_CSC_TYPE_709。如果是标清摄像头输入，建议 enViCscType 配置为 VI_CSC_TYPE_601。
- 亮度、色调、对比度、饱和度的调节参数取值范围为[0~100]，默认值为 50。
- 关于 CSC 有以下模块参数：

模块参数	意义及取值	备注
ext_csc_en	0：默认，亮度调节范围为[0~100] 1：亮度调节范围为[-128~128]	Hi3518/Hi3516C 有效

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

模块参数	意义及取值	备注
csc_ct_mode	0：默认，亮度随对比度调节而变化 1：亮度不随对比度调节而变化	Hi3518/Hi3516C 有效
csc_tv_en	0：默认，色彩空间转换的常系数矩阵[0，255] 1：色彩空间转换的常系数矩阵[16，235]	

【举例】

```
HI_S32 s32Ret;
VI_CSC_ATTR_S stCscAttr;
/* first enable vi device and vi chn */

/* Init CSC attr */
stCscAttr.enViCscType = VI_CSC_TYPE_709;
stCscAttr.u32ContrVal = 50;
stCscAttr.u32HueVal = 50;
stCscAttr.u32LumaVal = 50;
stCscAttr.u32SatuVal = 50;

/* Set attribute for vi CSC attr */
s32Ret = HI_MPI_VI_SetCSCAttr(ViDev, &stCscAttr);
if (HI_SUCCESS != s32ret)
{
    printf("Set vi CSC attr err:0x%x\n", s32ret);
    return s32Ret;
}

/* Get attribute for vi CSC attr */
s32Ret = HI_MPI_VI_GetCSCAttr(ViDev, &stCscAttr);
if (HI_SUCCESS != s32ret)
{
    printf("Get vi CSC attr err:0x%x\n", s32ret);
    return s32Ret;
}
```

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VI_GetCSCAttr

【描述】

获取 VI 设备的 CSC 属性。

【语法】

```
HI_S32 HI_MPI_VI_GetCSCAttr(VI_DEV ViDev, VI_CSC_ATTR_S *pstCSCAttr);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入
pstCSCAttr	CSC 属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 CSC	设备号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	否	-
Hi3518/Hi3516C	是	[0, VIU_MAX_DEV_NUM)

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无

HI_MPI_VI_SetRotate

【描述】

设置 VI 图像旋转属性。

【语法】

```
HI_S32 HI_MPI_VI_SetRotate(VI_CHN ViChn, const ROTATE_E enRotate);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 物理通道号。	输入
enRotate	旋转属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 Rotate	物理通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	否	-
Hi3518A/Hi3516C	是	[0, VIU_MAX_PHYCHN_NUM)
Hi3518C/Hi3518E	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- Rotate 是静态属性，不能动态修改，且必须在设置通道属性后，在 Enable 通道前设置此属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 仅支持非场输出的 sp420 图像的旋转。
- 仅支持 90、180、270 的旋转，不支持任意角度旋转。
- 仅支持物理通道的 Rotate。
- 180 旋转时，建议使用 Flip+Mirror 方式来实现，这样可以减少带宽占用，提高性能。
- 设置旋转后，物理通道输出的宽高可能会发生变化，但获取的通道属性中的通道宽高仍为用户设置值。比如 720P 输入的图像，在旋转 90、270 时，实际输出的图像大小为（720，1280），但 HI_MPI_VI_GetChnAttr 获取的 DestSize 宽高仍为（1280，720）。
- 设置旋转后，用户图片要根据转后图像大小进行设置。

【举例】

```
HI_S32 s32Ret;
VI_CHN ViChn = 0;
ROTATE_E enRotate;

/* first enable vi device */
/* set vi chn attr */
s32Ret = HI_MPI_VI_SetChnAttr(ViChn,&stChnAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set chn attributes failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}

/* Set Rotate_90 */
s32Ret = HI_MPI_VI_SetRotate(ViChn, ROTATE_90);
if (HI_SUCCESS != s32ret)
{
    printf("Set vi rotate err:0x%x\n", s32ret);
    return s32Ret;
}

s32Ret = HI_MPI_VI_EnableChn(ViChn);
if (s32Ret != HI_SUCCESS)
{
   printf("Enable chn failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}

/* Get Rotate*/
s32Ret = HI_MPI_VI_GetRotate(ViChn, &enRotate);
if (HI_SUCCESS != s32ret)
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    printf("Get vi rotate err:0x%x\n", s32ret);
    return s32Ret;
}
```

【相关主题】

无。

HI_MPI_VI_GetRotate

【描述】

获取 VI 图像旋转属性。

【语法】

```
HI_S32 HI_MPI_VI_GetRotate(VI_CHN ViChn, ROTATE_E *penRotate);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 物理通道号。	输入
penRotate	旋转属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持 Rotate	物理通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	否	-
Hi3518A/Hi3516C	是	[0, VIU_MAX_PHYCHN_NUM)
Hi3518C/Hi3518E	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VI_SetRotate

HI_MPI_VI_GetChnLuma

【描述】

获取通道亮度和统计。

【语法】

```
HI_S32 HI_MPI_VI_GetChnLuma(VI_CHN ViChn, VI_CHN_LUM_S *pstLuma);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 物理通道号。	输入
pstLuma	亮度和统计指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持亮度和统计	物理通道号范围
Hi3531/Hi3532	否	-
Hi3521/Hi3520A	否	-
Hi3518/Hi3516C/ Hi3520D/Hi3515A/Hi3515C	是	[0, VIU_MAX_PHYCHN_NUM)

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_SetPortMode

【描述】

设置 VI 端口模式。

【语法】

```
HI_S32 HI_MPI_VI_SetPortMode(VI_DEV ViDev, const VI_PORTMODE_E
enPortMode);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入
enPortMode	端口模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【芯片差异】

芯片类型	是否支持	VI 设备号范围
Hi3531/Hi3532/ Hi3521/Hi3520A	否	-
Hi3520Dv200	是	[0,1]
Hi3518A/Hi3516C	否	

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片类型	是否支持	VI 设备号范围
Hi3518C/Hi3518E	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

- 系统默认为 D1/960H 模式。
- 只有在 Hi3520Dv200 运行 4 路 720P 业务时，才需要设置 VI 端口模式为 720P 模式，其它业务不需要设置。
- 改接口需要在 HI_MPI_VI_SetDevAttr 之后、HI_MPI_VI_EnableDev 之前调用。

【举例】

无

【相关主题】

无

HI_MPI_VI_GetPortMode

【描述】

获取 VI 端口模式。

【语法】

```
HI_S32 HI_MPI_VI_GetPortMode(VI_DEV ViDev, VI_PORTMODE_E *penPortMode);
```

【参数】

参数名称	描述	输入/输出
ViDev	VI 设备号。	输入
penRotate	端口属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【芯片差异】

芯片类型	是否支持	VI 设备号范围
Hi3531/Hi3532/ Hi3521/Hi3520A	否	-
Hi3520Dv200	是	[0,1]
Hi3518A/Hi3516C	否	
Hi3518C/Hi3518E	否	-

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VI_SetPortMode

HI_MPI_VI_EnableChn422to420

【描述】

开启 VI 通道将采集的 422 图像转为 420 功能。

【语法】

```
HI_S32 HI_MPI_VI_EnableChn422to420(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

典型运用场景：Hi3520Dv200 运行 4 路 720P 业务时，2/6 通道只能采集 422 图像，为节省带宽，可以使能此功能，将其转为 420。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_DisableChn422to420

【描述】

关闭 VI 通道将采集的 422 图像转为 420 功能。

【语法】

```
HI_S32 HI_MPI_VI_DisableChn422to420(VI_CHN ViChn);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

HI_MPI_VI_EnableChn422to420

HI_MPI_VI_SetChnScanMode

【描述】

设置 VI 通道扫描模式。

【语法】

```
HI_S32 HI_MPI_VI_SetChnScanMode(VI_CHN ViChn, VI_SCAN_MODE_E enScanMode);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
enScanMode	扫描模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

典型运用场景：前端 AD 的视频信号为一路标清和一路高清复用，则需要设置其对应的 VI 通道的扫描模式。

【举例】

无。

【相关主题】

无。

HI_MPI_VI_GetChnScanMode

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

获取 VI 通道扫描模式。

【语法】

```
HI_S32 HI_MPI_VI_GetChnScanMode(VI_CHN ViChn, VI_SCAN_MODE_E
*penScanMode);
```

【参数】

参数名称	描述	输入/输出
ViChn	VI 通道号。	输入
penScanMode	扫描模式指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_vi.h、mpi_vi.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

3.5 数据类型

视频输入相关数据类型定义如下：

- VIU_MAX_DEV_NUM：定义视频输入设备的最大个数。
- VIU_MAX_WAY_NUM_PER_DEV：定义一个视频输入设备下的数据通路最大个数。
- VIU_MAX_PHYCHN_NUM：定义 VI 物理通道的最大个数。
- VIU_MAX_CHN_NUM：定义 VI 通道的最大个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- SUBCHN：定义 VI 通道的次通道号。
- VI_CAS_CHN_1：定义 VI 级联扩展通道号 32。
- VI_CAS_CHN_2：定义 VI 级联扩展通道号 33。
- VI_INTF_MODE_E：定义视频输入设备的接口模式。
- VI_INPUT_MODE_E：定义视频设备的输入模式。
- VI_COMBINE_MODE_E：定义视频设备接收的数据是复合还是分离模式。
- VI_COMP_MODE_E：定义视频设备接收的数据的分量类型。
- VI_CLK_EDGE_E：定义视频设备接收的时钟类型。
- VI_SCAN_MODE_E：定义视频设备接收的是隔行或逐行输入图像。
- VI_DATA_YUV_SEQ_E：定义视频设备接收的 YUV 数据的数据排列顺序。
- VI_VSYNC_E：定义视频设备输入数据的垂直同步信号类型。
- VI_VSYNC_NEG_E：定义视频设备输入数据垂直同步信号的极性。
- VI_HSYNC_E：定义视频设备输入数据的水平同步信号类型。
- VI_HSYNC_NEG_E：定义视频设备输入数据水平同步信号的极性。
- VI_VSYNC_VALID_E：定义视频设备输入数据的垂直有效同步信号类型。
- VI_VSYNC_VALID_NEG_E：定义视频设备输入数据垂直有效同步信号的极性。
- VI_SYNC_CFG_S：定义视频设备接收的 BT601 或 DC 时序的同步信息。
- VI_BT656_SYNC_CFG_S：定义视频设备接收的 BT656 时序的同步信息。
- BT656_FIXCODE_E：定义 BT.656 定时基准码最高 bit 配置。
- BT656_FIELD_POLAR_E：BT.656 定时基准码场指示位（F）极性。
- VI_DEV_ATTR_S：定义视频输入设备的属性。
- VI_CAPSEL_E：定义视频输入捕获图像的帧场选择。
- VI_CHN_ATTR_S：定义 VI 通道属性。
- VI_USERPIC_MODE_E：定义用户图片模式。
- VI_USERPIC_BGC_S：纯色背景模式下的用户图片相关信息。
- VI_USERPIC_ATTR_S：用户图片信息。
- VI_CHN_STAT_S：VI 通道信息结构体。
- ROTATE_E：旋转枚举。
- VI_DATA_PATH_E：VI 输入数据通路枚举。
- VI_EXT_CHN_ATTR_S：定义 VI 扩展通道属性。
- LDC_VIEW_TYPE_E：定义镜头畸变校正的类型。
- LDC_ATTR_S：定义镜头畸变校正属性。
- VI_LDC_ATTR_S：定义 VI 镜头畸变校正属性。
- VI_CSC_TYPE_E：定义 YUV 色彩空间标准。
- VI_CSC_ATTR_S：定义 CSC 属性。
- VI_FLASH_MODE_E：定义闪光灯模式。
- VI_FLASH_CONFIG_S：定义闪光灯配置属性。
- VI_CHN_LUM_S：定义通道亮度和结构体。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VI_PORTMODE_E：定义 VI 端口模式

VIU_MAX_DEV_NUM

【说明】

定义视频输入设备的最大个数。

【定义】

For Hi3531/Hi3532:

```
#define VIU_MAX_DEV_NUM        8
```

For Hi3521/Hi3520A:

```
#define VIU_MAX_DEV_NUM       4
```

For Hi3520D:

```
#define VIU_MAX_DEV_NUM       2
```

For Hi3518/Hi3516C/Hi3515A/Hi3515C:

```
#define VIU_MAX_DEV_NUM        1
```

【注意事项】

具体请参见“3.3 功能描述”中各款芯片“视频输入设备”小节的描述。

【相关数据类型及接口】

无。

VIU_MAX_WAY_NUM_PER_DEV

【说明】

定义一个视频输入设备下的数据通路最大个数。

【定义】

For Hi3531/Hi3532/Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C:

```
#define VIU_MAX_WAY_NUM_PER_DEV 4
```

For Hi3518/Hi3516C:

```
#define VIU_MAX_WAY_NUM_PER_DEV        1
```

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VIU_MAX_PHYCHN_NUM

【说明】

定义 VI 物理通道的最大个数。

【定义】

For Hi3531/Hi3532:

```
#define VIU_MAX_PHYCHN_NUM 32
```

For Hi3521/Hi3520A:

```
#define VIU_MAX_PHYCHN_NUM 16
```

For Hi3520D:

```
#define VIU_MAX_PHYCHN_NUM 8
```

For Hi3515A/Hi3515C:

```
#define VIU_MAX_PHYCHN_NUM 4
```

For Hi3518/Hi3516C:

```
#define VIU_MAX_PHYCHN_NUM 1
```

【注意事项】

具体见各款芯片“视频物理通道”小节的描述。

【相关数据类型及接口】

无。

VIU_EXT_CHN_START

【说明】

定义 VI 扩展通道号的起始值。

【定义】

```
#define VIU_EXT_CHN_START    VIU_MAX_PHYCHN_NUM
```

【注意事项】

无。

【相关数据类型及接口】

无。

VIU_MAX_EXT_CHN_NUM

【说明】

定义 VI 扩展通道的最大个数。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

For Hi3531/Hi3532/Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C:

```
#define VIU_MAX_EXT_CHN_NUM         0
```

For Hi3518/Hi3516C

```
#define VIU_MAX_EXT_CHN_NUM         16
```

【注意事项】

无。

【相关数据类型及接口】

无。

VIU_MAX_EXTCHN_BIND_PER_CHN

【说明】

定义 VI 单个通道可绑定的扩展通道最大个数。

【定义】

For Hi3531/Hi3532/Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C:

```
#define VIU_MAX_EXTCHN_BIND_PER_CHN         0
```

For Hi3518/Hi3516C

```
#define VIU_MAX_EXTCHN_BIND_PER_CHN         8
```

【注意事项】

无。

【相关数据类型及接口】

无。

VIU_MAX_CHN_NUM

【说明】

定义 VI 通道的最大个数。

【定义】

For Hi3531/Hi3532:

```
#define VIU_MAX_CHN_NUM         34
```

For Hi3521/Hi3520A:

```
#define VIU_MAX_CHN_NUM         16
```

For Hi3520D:

```
#define VIU_MAX_CHN_NUM          8
```

For Hi3515A/Hi3515C:

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
#define VIU_MAX_CHN_NUM          4
```

For Hi3518/Hi3516C:

```
#define VIU_MAX_CHN_NUM        (VIU_MAX_PHYCHN_NUM + VIU_MAX_EXT_CHN_NUM)
```

【注意事项】

对于 Hi3531/Hi3532：

- 0 至 15 是主通道号；
- 16 至 31 是主通道对应的次通道号，但奇数无效；
- 32 至 33 是级联扩展通道软件上的概念）。

对于 Hi3518/Hi3516C：

- 0 是物理通道
- 1 至 16 是扩展通道

【相关数据类型及接口】

无。

SUBCHN

【说明】

定义 VI 通道的次通道号。

【定义】

```
#define  SUBCHN(ViChn)   (ViChn + 16)
```

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	此宏有意义
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	此宏无意义
Hi3518/Hi3516C	此宏无意义

【注意事项】

ViChn 取值为{0, 15}。

【相关数据类型及接口】

无。

VI_CAS_CHN_1

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义 VI 级联扩展通道号 32。

【定义】

```
#define VI_CAS_CHN_1   32
```

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	此宏有意义
Hi3521/Hi3520A/ Hi3520D/Hi3515A/Hi3515C	此宏无意义
Hi3518/Hi3516C	此宏无意义

【注意事项】

该通道号只适用于 HI_MPI_VI_EnableCascadeChn 和 HI_MPI_VI_DisableCascadeChn 接口。

【相关数据类型及接口】

无。

VI_CAS_CHN_2

【说明】

定义 VI 级联扩展通道号 33。

【定义】

```
#define VI_CAS_CHN_1   33
```

【芯片差异】

芯片类型	差异
Hi3531/Hi3532	此宏有意义
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	此宏无意义
Hi3518/Hi3516C	此宏无意义

【注意事项】

该通道号只适用于 HI_MPI_VI_EnableCascadeChn 和 HI_MPI_VI_DisableCascadeChn 接口。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VI_INTF_MODE_E

【说明】

定义视频设备的接口模式。

【定义】

```
typedef enum hiVI_INTF_MODE_E
{
    VI_MODE_BT656 = 0,
    VI_MODE_BT601,
    VI_MODE_DIGITAL_CAMERA,
    VI_MODE_BT1120_STANDARD,
    VI_MODE_BT1120_INTERLEAVED,
    VI_MODE_BUTT
} VI_INTF_MODE_E;
```

【成员】

成员名称	描述
VI_MODE_BT656	输入数据的协议符合标准 BT656 协议，端口数据输入模式为亮度色度复合模式，分量模式为单分量。
VI_MODE_BT601	输入数据的协议符合标准 BT601 协议，端口数据输入模式为亮度色度复合模式，分量模式为单分量。
VI_MODE_DIGITAL_CAMERA	输入数据的协议为 Digital camera 协议，端口数据输入模式为亮度色度复合模式，分量模式为单分量。
VI_MODE_BT1120_STANDARD	输入数据的协议符合标准 BT1120 协议（BT656+双分量），端口数据输入模式为亮度色度分离模式，分量模式为双分量。
VI_MODE_BT1120_INTERLEAVED	输入数据的协议符合 BT1120 interleave 模式，端口数据输入模式为亮度色度分离模式，分量模式为双分量。

【注意事项】

如果是 Sensor，则为 VI_MODE_DIGITAL_CAMERA。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttr

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VI_INPUT_MODE_E

【说明】

定义视频设备的输入模式。

【定义】

```
typedef enum hiVI_INPUT_MODE_E
{
    VI_MODE_BT656 = 0,
    VI_MODE_BT601,
    VI_MODE_DIGITAL_CAMERA,
    VI_INPUT_MODE_INTERLEAVED,
    VI_MODE_BUTT
} VI_INPUT_MODE_E;
```

【成员】

成员名称	描述
VI_MODE_BT656	输入数据的协议符合标准 BT656 协议。
VI_MODE_BT601	输入数据的协议符合标准 BT601 协议。
VI_MODE_DIGITAL_CAMERA	输入数据的协议为 Digital camera 协议
VI_INPUT_MODE_INTERLEAVED	输入数据的协议为 BT1120 interleave。

【注意事项】

如果是 Sensor，则为 VI_MODE_DIGITAL_CAMERA；如果 VI 设备的输入符合标准 BT1120 协议，在此应设置为 VI_MODE_BT656，且 VI_COMP_MODE_E 应设置为双分量模式，如果设备输入符合 BT1120 interleave 模式，则此处应设置为 VI_INPUT_MODE_INTERLEAVED。

【相关数据类型及接口】

- VI_DEV_ATTR_EX_S
- HI_MPI_VI_SetDevAttrEx

VI_WORK_MODE_E

【说明】

定义视频设备的复合工作模式。

【定义】

```
typedef enum hiVI_WORK_MODE_E
{
    VI_WORK_MODE_1Multiplex = 0,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VI_WORK_MODE_2Multiplex,
    VI_WORK_MODE_4Multiplex,
    VI_WORK_MODE_BUTT
} VI_WORK_MODE_E;
```

【成员】

成员名称	描述
VI_WORK_MODE_1Multiplex	1 路复合工作模式。
VI_WORK_MODE_2Multiplex	2 路复合工作模式，输入数据的协议必须为标准 BT656 协议。
VI_WORK_MODE_4Multiplex	4 路复合工作模式，输入数据的协议必须为标准 BT656 协议。

【注意事项】

当该项设为 2 路或 4 路复合工作模式时，设备输入的协议必须是 BT656 协议。1 路复合工作模式没有限制。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttr

VI_SCAN_MODE_E

【说明】

定义视频设备接收的是隔行或逐行输入图像。

【定义】

```
typedef enum hiVI_SCAN_MODE_E
{
    VI_SCAN_INTERLACED  = 0,    /*隔行*/
    VI_SCAN_PROGRESSIVE,        /*逐行*/
    VI_SCAN_BUTT,
} VI_SCAN_MODE_E;
```

【成员】

成员名称	描述
VI_SCAN_INTERLACED	VI 输入为隔行图像。
VI_SCAN_PROGRESSIVE	VI 输入为逐行图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttrEx

VI_DATA_YUV_SEQ_E

【说明】

定义视频设备接收的 YUV 数据的数据排列顺序。

【定义】

```
typedef enum hiVI_DATA_YUV_SEQ_E
{
    VI_INPUT_DATA_VUVU = 0,
    VI_INPUT_DATA_UVUV,
    VI_INPUT_DATA_UYVY = 0,
    VI_INPUT_DATA_VYUY,
    VI_INPUT_DATA_YUYV,
    VI_INPUT_DATA_YVYU,
     VI_DATA_YUV_BUTT
} VI_DATA_YUV_SEQ_E;
```

【成员】

成员名称	描述
VI_INPUT_DATA_VUVU	YUV 数据通过分离模式输入时，C 分量的输入排列顺序为 VUVU。
VI_INPUT_DATA_UVUV	YUV 数据通过分离模式输入时，C 分量的输入排列顺序为 UVUV。
VI_INPUT_DATA_UYVY	YUV 数据通过复合模式输入时，顺序为 UYVY。
VI_INPUT_DATA_VYUY	YUV 数据通过复合模式输入时，顺序为 VYUY。
VI_INPUT_DATA_YUYV	YUV 数据通过复合模式输入时，顺序为 YUYV。
VI_INPUT_DATA_YVYU	YUV 数据通过复合模式输入时，顺序为 YVYU。

【注意事项】

无。

【相关数据类型及接口】

- VI_DEV_ATTR_S

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VI_SetDevAttr
- VI_DEV_ATTR_EX_S
- HI_MPI_VI_SetDevAttrEx

VI_CLK_EDGE_E

【说明】

定义视频设备接收的时钟类型。

【定义】

```
typedef enum hiVI_CLK_EDGE_E
{
    VI_CLK_EDGE_SINGLE_UP = 0,
    VI_CLK_EDGE_SINGLE_DOWN,
    VI_CLK_EDGE_BUTT
} VI_CLK_EDGE_E;
```

【成员】

成员名称	描述
VI_CLK_EDGE_SINGLE_UP	时钟单沿模式，且 VI 设备在上升沿采样。
VI_CLK_EDGE_SINGLE_DOWN	时钟单沿模式，且 VI 设备在下降沿采样。

【注意事项】

无。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttrEx

VI_COMP_MODE_E

【说明】

定义视频设备接收的数据的分量类型。

【定义】

```
typedef enum hiVI_COMP_MODE_E
{
    VI_COMP_MODE_SINGLE = 0,
    VI_COMP_MODE_DOUBLE = 1,
    VI_COMP_MODE_BUTT,
}VI_COMP_MODE_E;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
VI_COMP_MODE_SINGL	输入数据为单分量。
VI_COMP_MODE_DOUBLE	输入数据为双分量。

【注意事项】

- 此枚举需要与 VI_DEV_ATTR_S 或 VI_DEV_ATTR_EX_S 中的分量掩码 au32CompMask[2]配合设置。
- VI 视频输入设备 Dev0～Dev3 共用 32bit 数据线（对于 Hi3531/Hi3532 的 Dev4～Dev7 也是如此），用于与外部视频源对接，对接时可能仅使用 32bit 数据线中的一部分（由硬件连线决定）。例如 Dev0 输入 8bit 的 YUV 数据，camera 与 VI 对接的是 32bit 数据线的前 8bit 数据线（即 bit0~bit7 对应的数据线），此时配置输入数据为单分量，分量掩码为 0xFF000000。

如果视频数据是双分量输入，则需要将设备属性配为双分量，并根据管脚的实际连接情况指定每个分量的分量掩码。例如 Dev0 设为 BT.1120、双分量输入（亮度分量和色度分量各占用 8bit），则分量掩码配置为 0xFF000000 和 0xFF0000。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- VI_DEV_ATTR_EX_S
- HI_MPI_VI_SetDevAttrEx
- HI_MPI_VI_SetDevAttr

VI_COMBINE_MODE_E

【说明】

定义视频设备接收的数据是复合还是分离模式。

【定义】

```
typedef enum hiVI_COMBINE_MODE_E
{
    VI_COMBINE_COMPOSITE = 0,
    VI_COMBINE_SEPARATE,
    VI_COMBINE_BUTT,
} VI_COMBINE_MODE_E;
```

【成员】

成员名称	描述
VI_COMBINE_COMPOSITE	复合模式。
VI_COMBINE_SEPARATE	分离模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

如果 Y 与 C 分量是通过同一组数据线输入，则为 VI_COMBINE_COMPOSITE，如 BT.656；如果 Y 与 C 分量是通过不同的数据线输入，为 VI_COMBINE_SEPARATE，如 BT.1120。

【相关数据类型及接口】

- VI_DEV_ATTR_EX_S
- HI_MPI_VI_ChnBind

VI_VSYNC_E

【说明】

定义视频设备输入数据的垂直同步信号类型。

【定义】

```
typedef enum hiVI_VSYNC_E
{
    VI_VSYNC_FIELD = 0,
    VI_VSYNC_PULSE,
} VI_VSYNC_E;
```

【成员】

成员名称	描述
VI_VSYNC_FIELD	垂直同步翻转模式，即翻转一次表示一场。BT601 模式下表示场号，DC 模式下表示行有效信号。
VI_VSYNC_PULSE	垂直同步脉冲模式，即一个脉冲到来表示新的一帧或一场。

【注意事项】

无。

【相关数据类型及接口】

- VI_VSYNC_NEG_E
- VI_SYNC_CFG_S

VI_VSYNC_NEG_E

【说明】

定义视频设备输入数据垂直同步信号的极性。

【定义】

```
typedef enum hiVI_VSYNC_NEG_E
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    VI_VSYNC_NEG_HIGH = 0,
    VI_VSYNC_NEG_LOW
} VI_VSYNC_NEG_E;
```

【成员】

成员名称	描述
VI_VSYNC_NEG_HIGH	若 VI_VSYNC_E = VI_VSYNC_FIELD，则表示偶数场的 vsync 信号为高电平； 若 VI_VSYNC_E = VI_VSYNC_PULSE，则表示正脉冲表示 vsync 同步脉冲。
VI_VSYNC_NEG_LOW	若 VI_VSYNC_E = VI_VSYNC_FIELD，则表示偶数场的 vsync 信号为低电平； 若 VI_VSYNC_E = VI_VSYNC_PULSE，则表示负脉冲表示 vsync 同步脉冲。

【注意事项】

无。

【相关数据类型及接口】

- VI_VSYNC_E
- VI_SYNC_CFG_S

VI_HSYNC_E

【说明】

定义视频设备输入数据的水平同步信号类型。

【定义】

```
typedef enum hiVI_HSYNC_E
{
    VI_HSYNC_VALID_SINGNAL = 0,
    VI_HSYNC_PULSE,
} VI_HSYNC_E;
```

【成员】

成员名称	描述
VI_HSYNC_VALID_SINGNAL	水平同步数据有效信号。
VI_HSYNC_PULSE	水平同步脉冲信号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

- VI_HSYNC_NEG_E
- VI_SYNC_CFG_S

VI_HSYNC_NEG_E

【说明】

定义视频设备输入数据水平同步信号的极性。

【定义】

```
typedef enum hiVI_HSYNC_NEG_E
{
    VI_HSYNC_NEG_HIGH = 0,
    VI_HSYNC_NEG_LOW
} VI_HSYNC_NEG_E;
```

【成员】

成员名称	描述
VI_HSYNC_NEG_HIGH	若 VI_HSYNC_E=VI_HSYNC_VALID_SINGNAL，则高电平表示有效数据； 若 VI_HSYNC_E = VI_HSYNC_PULSE，则正脉冲表示同步脉冲。
VI_HSYNC_NEG_LOW	若 VI_HSYNC_E=VI_HSYNC_VALID_SINGNAL，则低电平表示有效数据； 若 VI_HSYNC_E = VI_HSYNC_PULSE，则负脉冲表示同步脉冲。

【注意事项】

无。

【相关数据类型及接口】

- VI_HSYNC_E
- VI_SYNC_CFG_S

VI_VSYNC_VALID_E

【说明】

定义视频设备输入数据的垂直有效同步信号类型。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef enum hiVI_VSYNC_VALID_E
{
    VI_VSYNC_NORM_PULSE = 0,
    VI_VSYNC_VALID_SINGAL,
} VI_VSYNC_VALID_E;
```

【成员】

成员名称	描述
VI_VSYNC_NORM_PULSE	DC 模式下使用，表示垂直有效同步标识。
VI_VSYNC_VALID_SINGAL	DC 模式下使用，表示垂直同步时序行有效信号。

【注意事项】

DC 模式下需要配置该项。

【相关数据类型及接口】

- VI_VSYNC_E
- VI_VSYNC_VALID_NEG_E
- VI_SYNC_CFG_S

VI_VSYNC_VALID_NEG_E

【说明】

定义视频设备输入数据垂直有效同步信号的极性。

【定义】

```
typedef enum hiVI_VSYNC_VALID_NEG_E
{
    VI_VSYNC_VALID_NEG_HIGH = 0,
    VI_VSYNC_VALID_NEG_LOW
} VI_VSYNC_VALID_NEG_E;
```

【成员】

成员名称	描述
VI_VSYNC_VALID_NEG_HIGH	若 VI_VSYNC_VALID_E = VI_VSYNC_NORM_SINGAL，则表示高电平为有效信号。
VI_VSYNC_VALID_NEG_LOW	若 VI_VSYNC_VALID_E = VI_VSYNC_NORM_SINGAL，则表示低电平为有效信号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

只有在 VI_VSYNC_VALID_E = VI_VSYNC_NORM_SINGAL 时，该项才有效。

【相关数据类型及接口】

- VI_VSYNC_VALID_E
- VI_SYNC_CFG_S

VI_TIMING_BLANK_S

【说明】

定义视频设备输入时序的消隐信息。

【定义】

```
typedef struct hiVI_TIMING_BLANK_S
{
    HI_U32 u32HsyncHfb ;
    HI_U32 u32HsyncAct ;
    HI_U32 u32HsyncHbb ;
    HI_U32 u32VsyncVfb ;
    HI_U32 u32VsyncVact ;
    HI_U32 u32VsyncVbb ;
    HI_U32 u32VsyncVbfb ;
    HI_U32 u32VsyncVbact ;
    HI_U32 u32VsyncVbbb ;
}VI_TIMING_BLANK_S;
```

【成员】

成员名称	描述
u32HsyncHfb	水平前消隐区宽度。
u32HsyncAct	水平有效宽度。
u32HsyncHbb	水平后消隐区宽度
u32VsyncVfb	帧图像或隔行输入时奇场图像的垂直前消隐区高度。
u32VsyncVact	帧图像或隔行输入时奇场垂直有效高度。
u32VsyncVbb	帧图像或隔行输入时奇场垂直后消隐区高度。
u32VsyncVbfb	隔行输入时偶场垂直前消隐区高度（帧输入时无效）。
u32VsyncVbact	隔行输入时偶场垂直有效高度（帧输入时无效）。
u32VsyncVbbb	隔行输入时偶场垂直后消隐区高度（帧输入时无效）。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

VI_SYNC_CFG_S

VI_SYNC_CFG_S

【说明】

定义视频设备接收的 BT601 或 DC 时序的同步信息。

【定义】

```
typedef struct hiVI_SYNC_CFG_S
{
    VI_VSYNC_E            enVsync;
    VI_VSYNC_NEG_E        enVsyncNeg;
    VI_HSYNC_E            enHsync;
    VI_HSYNC_NEG_E        enHsyncNeg;
    VI_VSYNC_VALID_E      enVsyncValid;
    VI_VSYNC_VALID_NEG_E  enVsyncValidNeg;
    VI_TIMING_BLANK_S     stTimingBlank;
} VI_SYNC_CFG_S;
```

【成员】

成员名称	描述
enVsync	垂直同步信号类型。
enVsyncNeg	垂直同步信号的极性。
enHsync	水平同步信号类型。
enHsyncNeg	水平同步信号的极性。
enVsyncValid	垂直有效同步信号类型。
enVsyncValidNeg	垂直有效同步信号的极性。
stTimingBlank	输入时序的消隐信息

【注意事项】

- BT601 模式下，垂直同步信号只需配置 enVsync 和 enVsyncNeg 即可，它们分别表示管脚 VI_P_VSYNC_FIELD 的垂直同步信号类型和垂直同步信号的极性，两者决定了采用何种垂直同步时序。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

  - enVsync 配置为 VI_VSYNC_FIELD，BT601 模式下表示场号。此时，如果 enVsyncNeg 配置为 VI_VSYNC_NEG_HIGH，则高电平表示偶场，低电平表示奇场；如果 enVsyncNeg 配置为 VI_VSYNC_NEG_LOW，则低电平表示偶场，高电平表示奇场。
  - enVsync 配置为 VI_VSYNC_PULSE，表示垂直同步脉冲，即一个脉冲到来表示新的一场或一帧。此时，如果 enVsyncNeg 配置为 VI_VSYNC_NEG_HIGH，则正脉冲表示垂直同步脉冲；如果 enVsyncNeg 配置为 VI_VSYNC_NEG_LOW，则负脉冲表示垂直同步脉冲。
- DC 模式下，垂直同步信号需要配置 enVsync、enVsyncValid 和 enVsyncValidNeg 三项，它们分别表示管脚 VI_P_VSYNC_FIELD 的垂直同步信号类型、垂直同步信号标识和垂直有效信号极性。enVsyncValidNeg 只有在 enVsyncValid 配为 VI_VSYNC_VALID_SINGAL 时有效。
  - enVsync 配置为 VI_VSYNC_PULSE，表示垂直时序脉冲方式。如果在垂直消隐区内 hsync 信号仍有效，则 enVsyncValid 需配置为 VI_VSYNC_VALID_SINGAL 选择垂直有效标识，同时 enVsyncValidNeg 需要配置为 VI_VSYNC_VALID_NEG_HIGH（或 VI_VSYNC_VALID_NEG_LOW），以表示高（低）电平有效，其他情况不需关注此属性。
  - 如果输入时序为垂直时序行有效方式，只需将 enVsync 配置为 VI_VSYNC_FIELD 即可。
- enHsync 和 enHsyncNeg 分别表示管脚 VI_P_HSYNC_VD 的水平同步信号类型和水平同步信号的极性，两者决定了采用何种水平同步时序。
  - enHsync 配置为 VI_HSYNC_VALID_SINGNAL，表示数据有效信号。此时，如果 enHsyncNeg 配置为 VI_HSYNC_NEG_HIGH，则高电平表示数据有效；如果 enHsyncNeg 配置为 VI_HSYNC_NEG_LOW，则低电平表示数据有效。
  - enHsync 配置为 VI_HSYNC_PULSE，表示水平同步脉冲信号，即一个脉冲到来表示新一行的开始。此时，如果 enHsyncNeg 配置为 VI_HSYNC_NEG_HIGH，则正脉冲表示水平同步脉冲；如果 enHsyncNeg 配置为 VI_HSYNC_NEG_LOW，则负脉冲表示水平同步脉冲。

对于消隐信息，水平或垂直的有效数据方式不需要配消隐区。

Hi3531 支持的具体时序介绍请参见《Hi3531 H.264 编解码处理用户指南》。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttr
- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttrEx

BT656_FIXCODE_E

【说明】

定义 BT.656 定时基准码最高 bit 配置。

【定义】

```
typedef enum hiBT656_FIXCODE_E
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    BT656_FIXCODE_1 = 0,
    BT656_FIXCODE_0
}BT656_FIXCODE_E;
```

【成员】

成员名称	描述
BT656_FIXCODE_1	BT656 协议的 EAV/SAV 最高 bit 固定为 1。
BT656_FIXCODE_0	BT656 协议的 EAV/SAV 最高 bit 固定为 0。

【注意事项】

无。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttrEx

BT656_FIELD_POLAR_E

【说明】

BT.656 定时基准码场指示位（F）极性。

【定义】

```
typedef enum hiBT656_FIELD_POLAR_E
{
    BT656_FIELD_POLAR_STD = 0,
    BT656_FIELD_POLAR_NSTD
}BT656_FIELD_POLAR_E;
```

【成员】

成员名称	描述
BT656_FIELD_POLAR_STD	标准模式，第一场 F=0，第二场 F=1。
BT656_FIELD_POLAR_NSTD	非标准模式，第一场 F=1，第二场 F=0。

【注意事项】

无。

【相关数据类型及接口】

- VI_DEV_ATTR_S

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VI_SetDevAttrEx

VI_BT656_SYNC_CFG_S

【说明】

定义视频设备接收的 BT.656 时序的同步信息。

【定义】

```
typedef struct hiVI_BT656_SYNC_CFG_S
{
    BT656_FIXCODE_E          enFixCode;
    BT656_FIELD_POLAR_E      enFieldPolar;
}VI_BT656_SYNC_CFG_S;
```

【成员】

成员名称	描述
enFixCode	BT.656 定时基准码最高 bit 配置。
enFieldPolar	BT.656 定时基准码场指示位（F）极性。

【注意事项】

BT.656 相关定义请参见标准协议说明。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- HI_MPI_VI_SetDevAttrEx

VI_DEV_ATTR_S

【说明】

定义视频输入设备的属性。

【定义】

```
typedef struct hiVI_DEV_ATTR_S
{
    VI_INTF_MODE_E       enIntfMode;
    VI_WORK_MODE_E       enWorkMode;
    HI_U32               au32CompMask[2];
    VI_SCAN_MODE_E      enScanMode;
    HI_S32               s32AdChnId[4];
    /*BT601和DC模式时以下必配，其它模式时无效*/
    VI_DATA_YUV_SEQ_E   enDataSeq;
    VI_SYNC_CFG_S       stSynCfg;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VI_DATA_PATH_E      enDataPath;
    VI_DATA_TYPE_E      enInputDataType;
    HI_BOOL             bDataRev;
} VI_DEV_ATTR_S;
```

【成员】

成员名称	描述
enIntfMode	接口模式。
enWorkMode	1、2、4 路复合工作模式。
au32CompMask[2]	分量掩码配置。当 enIntfMode=VI_MODE_BT1120_STANDARD 时，需要配置 Y 和 C 的分量掩码，其他模式时只需配置单分量掩码。
enScanMode	输入扫描模式 (逐行、隔行)。
s32AdChnId[4]	取值范围[-1, 3]，推荐统一设置为默认值-1.
enDataSeq	输入数据顺序 (仅支持 yuv 格式)，BT601 和 DC 模式时必须配置，其它模式时无效。
stSynCfg	同步时序配置，BT601 和 DC 模式时必须配置，其它模式时无效。
enDataPath	该成员只对 Hi3518/Hi3516C 有效。输入数据通路配置，不带 ISP 的 Sensor 输入时，配置为 ISP 通路；带 ISP 的 Sensor 或者 AD 输入时，配置为 BYPASS 通路（VI 内置 ISP 将 Disable）；RAW Data 一般仅在 Debug 时使用，用于捕获 Sensor 输入的原始数据。
enInputDataType	该成员只对 Hi3518/Hi3516C 有效。输入数据类型，Sensor 输入一般为 RGB，AD 输入一般为 YUV。
bDataRev	该成员只对 Hi3520D/Hi3515A/Hi3515C、Hi3518/Hi3516C 有效。因为走线约束等硬件原因,有可能出现 AD/Sensor 的数据线与 VI 数据线连接数据高低位反接，比如，AD_DATA0 与 VIU_DATA7 连接，AD_DATA7 与 VIU_DATA0 连接，以此类推。。当 AD/Sensor 管脚与 VI 管脚正向连接时，取 bDataRev= HI_FALSE；当反向连接时，取 bDataRev= HI_TRUE。

【注意事项】

- 当 s32AdChnId[i]设为-1 值时，MPP 将默认检测第 i 个 AdId，且不允许重复检测相同的 AdId，即 s32AdChnId 数组不允许设置为{-1，0，2，3}，因为 s32AdChnId[0]的值-1 转换后为 0，这与 s32AdChnId[1]的值 0 相冲突。
- 当 enWorkMode 设为 1 路复合模式时，s32AdChnId[0]必须设为-1 或 0；当 enWorkMode 设为 2 路复合模式时，s32AdChnId[0]、s32AdChnId[1]的值只能取{-1，0，1}。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 推荐统一将该数组的值都设为-1（即默认绑定关系下的通道按顺序检测{0，1，2，3}的 AdId）。

【相关数据类型及接口】

HI_MPI_VI_SetDevAttr

VI_DEV_ATTR_EX_S

【说明】

定义视频输入设备的属性。

【定义】

```
typedef struct hiVI_DEV_ATTR_EX_S
{
    VI_INPUT_MODE_E       enInputMode;
    VI_WORK_MODE_E        enWorkMode;
    VI_COMBINE_MODE_E     enCombineMode;
    VI_COMP_MODE_E        enCompMode;
    HI_U32                au32CompMask[2];
    VI_CLK_EDGE_E         enClkEdge;
    VI_SCAN_MODE_E        enScanMode;
    VI_DATA_YUV_SEQ_E     enDataSeq;
    HI_S32                s32AdChnId[4];
    VI_SYNC_CFG_S         stSynCfg;
    VI_BT656_SYNC_CFG_S   stBT656SynCfg;
    VI_DATA_PATH_E        enDataPath;
    VI_DATA_TYPE_E        enInputDataType;
    HI_BOOL               bDataRev;
} VI_DEV_ATTR_EX_S;
```

【成员】

成员名称	描述
enInputMode	接口模式。
enWorkMode	1、2、4 路复合工作模式。
enCombineMode	亮度色度复合/分离模式。
enCompMode	分量模式（单分量、双分量），对应于 enCombineMode。
au32CompMask[2]	分量掩码配置。当 enInputMode = VI_INPUT_MODE_BT656 且 enCompMode =VI_COMP_MODE_DOUBLE，enCombineMode =VI_COMBINE_SEPARATE 时，形成 VI_DEV_ATTR_S 中的时序，此时需要配置 Y 和 C 的分量掩码，其他模式时只需配置单分量掩码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enClkEdge	时钟边沿模式（上升沿采样、下降沿采样）。
enScanMode	输入扫描模式（逐行、隔行）。
enDataSeq	输入数据顺序（仅支持 YUV 格式）。
s32AdChnId[4]	请参见 VI_DEV_ATTR_S。
stSynCfg	同步时序配置，BT.601 和 DC 模式时必须配置，其它模式时无效。
stBT656SynCfg	同步时序配置（BT.656 接口时需要配置）。
enDataPath	该成员只对 Hi3518/Hi3516C 有效。输入数据通路配置，不带 ISP 的 Sensor 输入时，配置为 ISP 通路；带 ISP 的 Sensor 或者 AD 输入时，配置为 BYPASS 通路（VI 内置 ISP 将 Disable）；RAW Data 一般仅在 Debug 时使用，用于捕获 Sensor 输入的原始数据。
enInputDataType	该成员只对 Hi3518/Hi3516C 有效。输入数据类型，Sensor 输入一般为 RGB，AD 输入一般为 YUV。
bDataRev	该成员只对 Hi3520D/Hi3515A/Hi3515C、Hi3518/Hi3516C 有效。因为走线约束等硬件原因，有可能出现 AD/Sensor 的数据线与数据线连接数据高低位反接，比如，AD_DATA0 与 VIU_DATA7 连接，AD_DATA7 与 VIU_DATA0 连接，以此类推。当 AD/Sensor 管脚与 VI 管脚正向连接时，取 bDataRev= HI_FALSE；当反向连接时，取 bDataRev= HI_TRUE。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VI_SetDevAttrEx

VI_CHN_BIND_ATTR_S

【说明】

定义 VI 通道绑定属性。

【定义】

```
typedef struct hiVI_CHN_BIND_ATTR_S
{
    VI_DEV ViDev;
    VI_WAY ViWay;
} VI_CHN_BIND_ATTR_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
ViDev	该 VI 通道绑定的 Dev。
ViWay	该 VI 通道绑定 Dev 的 Way。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VI_ChnBind

VI_CAPSEL_E

【说明】

定义视频输入捕获图像的帧场选择。

【定义】

```
typedef enum hiVI_CAPSEL_E
{
    VI_CAPSEL_TOP=0,
    VI_CAPSEL_BOTTOM,
    VI_CAPSEL_BOTH,
    VI_CAPSEL_BUTT
} VI_CAPSEL_E;
```

【成员】

成员名称	描述
VI_CAPSEL_TOP	选择顶场。
VI_CAPSEL_BOTTOM	选择底场（若采集单场时推荐使用）。
VI_CAPSEL_BOTH	选择顶底两场。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VI_CHN_ATTR_S

【说明】

定义 VI 通道属性。

【定义】

```
typedef struct hiVI_CHN_ATTR_S
{
    RECT_S          stCapRect;
    SIZE_S          stDestSize;
    VI_CAPSEL_E     enCapSel;
    PIXEL_FORMAT_E  enPixFormat;
    HI_BOOL         bMirror;
    HI_BOOL         bFlip;
    HI_BOOL         bChromaResample;
    HI_S32          s32SrcFrameRate;
    HI_S32          s32FrameRate;
} VI_CHN_ATTR_S;
```

【成员】

成员名称	描述
stCapRect	捕获区域起始坐标(相对于设备图像的大小)与宽高。
stDestSize	目标图像大小。
enCapSel	帧场选择，只用于隔行模式，建议捕获单场时选择捕获底场。 逐行模式时，该项必须设置为 VI_CAPSEL_BOTH。
enPixFormat	像素存储格式支持 sp420 and sp422。
bMirror	是否水平翻转。
bFlip	是否垂直翻转。
bChromaResample	是否进行色度重采样。
s32SrcFrameRate	原始帧率，不能小于-1；建议设置为与对接 camera 一致的值。
s32FrameRate	目标帧率，必须小于或等于原始帧率，同时不能小于-1；原始帧率与目标帧率要么同时为-1，要么都不为-1，两者都为-1 时表示不进行帧率控制。

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	Hi3531/Hi3532	Hi3521/Hi3520A/ Hi3520D/Hi3515A/ Hi3515C	Hi3518/Hi3516C
stCapRect	仅对主通道有效，且其宽与高为静态属性。	仅对主属性有效，且其宽与高为静态属性；对于次属性，都为静态属性。	仅对物理通道有效，宽与高为静态属性。
stDestSize	对于主通道来说，由于主通道没有缩放能力，该值必须与 stCapRect 的宽高相同，且是静态属性；对次通道来说该项是动态属性。	对于主、次属性来说都是动态属性。	VI 通道无缩放能力，采双场时，该值必须与 stCapRect 的宽高相同；采单场时，高度必须为 stCapRect 高度的一半，且是静态属性。
enCapSel	对主通道来说该项是静态属性；对次通道来说该项是动态属性。	对于主、次属性来说都是动态属性	动态属性。
bChromaResample	仅对主通道有效。	无效。	无效。
s32SrcFrameRate 与 s32FrameRate	对于主、次通道属性都有效。	只对于主属性都有效，对次属性无效。	只对于主属性有效。

【注意事项】

- 主、次通道共用该属性，必须先设置主通道属性后，才能设置次通道属性。
- 主、次属性共用该属性，必须先设置主属性后，才能设置次属性。
- 对于静态属性项，必须先 disable 主、次通道后才能修改。
- 隔行图像不支持垂直翻转功能（因为开启该功能后顶底场采集顺序与原来的相反，后续模块处理可能会引起图像质量问题），MPP 内部不做此检查，由用户保证。

【相关数据类型及接口】

- HI_MPI_VI_SetChnAttr
- HI_MPI_VI_GetChnAttr

VI_USERPIC_MODE_E

【说明】

定义用户图片模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef enum hi_VI_USERPIC_MODE_E
{
    VI_USERPIC_MODE_PIC = 0,
    VI_USERPIC_MODE_BGC,
    VI_USERPIC_MODE_BUTT,
} VI_USERPIC_MODE_E;
```

【成员】

成员名称	描述
VI_USERPIC_MODE_PIC	YUV 图像模式。
VI_USERPIC_MODE_BGC	纯色背景模式。

【注意事项】

无。

【相关数据类型及接口】

无。

VI_USERPIC_BGC_S

【说明】

纯色背景模式下的用户图片相关信息。

【定义】

```
typedef struct hiVI_USERPIC_BGC_S
{
    HI_U32          u32BgColor;
} VI_USERPIC_BGC_S;
```

【成员】

成员名称	描述
u32BgColor	填充数据。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VI_USERPIC_ATTR_S

【说明】

用户图片信息。

【定义】

```
typedef struct hiVI_USERPIC_ATTR_S
{
    HI_BOOL               bPub;
    VI_USERPIC_MODE_E      enUsrPicMode;
    union
    {
        VIDEO_FRAME_INFO_S  stUsrPicFrm;
        VI_USERPIC_BGC_S    stUsrPicBg;
    }unUsrPic;
} VI_USERPIC_ATTR_S;
```

【成员】

成员名称	描述
bPub	用户图片信息是否为所有 VI 设备和通道公有。
enUsrPicMode	用户图片模式。
stUsrPicFrm	YUV 图像模式下的用户图片信息。
stUsrPicBg	纯色背景模式下的用户图片信息。

【注意事项】

目前用户图片信息仅只支持公有，故 bPub 只能设为 HI_TRUE。否则，内核会返回错误。

【相关数据类型及接口】

无。

VI_CHN_STAT_S

【说明】

VI 通道信息结构体。

【定义】

```
typedef struct hiVI_CHN_STAT_S
{
    HI_BOOL bEnable;   /* Whether this channel is enabled */
    HI_U32 u32IntCnt;  /* The video frame interrupt count */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32FrmRate; /* current frame rate */
    HI_U32 u32LostInt; /* The interrupt is received but nobody care */
    HI_U32 u32VbFail;  /* Video buffer malloc failure */
    HI_U32 u32PicWidth;/* curren pic width */
    HI_U32 u32PicHeight;/* current pic height */
    HI_U32 u32AutoDisInt; * auto disable interrupt count, when VIU
detected too many interrupts */
} VI_CHN_STAT_S;
```

【成员】

成员名称	描述
bEnable	通道是否使能。
u32IntCnt	中断计数。
u32FrmRate	每 10 秒的平均帧率，该值不一定精确。
u32LostInt	中断丢失计数。
u32VbFail	获取 VB 失败计数。
u32PicWidth	图像宽度。
u32PicHeight	图像高度。
u32AutoDisInt	当出现中断过多异常时，内部自动关闭中断的次数。 该参数与于模块参数 stop_int_level 相关，例如，stop_int_level 设为 500us，当相邻两个中断的时间间隔小于该值，则认为中断过多，此时，SDK 会自动关闭中断 10ms，10ms 后再恢复中断响应。

【注意事项】

- 结构体的中断计数，可用于无中断检测。
- 该结构体的帧率是每 10 秒钟的平均帧率，即 VI 会每隔十秒统计一次平均帧率，该值并不精确。
- 如果查询到该结构体的中断丢失计数一直在增加，说明 VI 工作出现异常。

【相关数据类型及接口】

无。

ROTATE_E

【说明】

旋转枚举。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef enum hiROTATE_E
{
    ROTATE_NONE = 0,
    ROTATE_90     = 1,
    ROTATE_180    = 2,
    ROTATE_270    = 3,
    ROTATE_BUTT
} ROTATE_E;;
```

【成员】

成员名称	描述
ROTATE_NONE	不旋转。
ROTATE_90	顺时针旋转 90。
ROTATE_180	顺时针旋转 180。
ROTATE_270	顺时针旋转 270。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VI_SetRotate
- HI_MPI_VI_GetRotate

VI_DATA_PATH_E

【说明】

VI 输入数据通路枚举。

【定义】

```
typedef enum hiVI_DATA_PATH_E
{
    VI_PATH_BYPASS    = 0,      /* ISP bypass */
    VI_PATH_ISP       = 1,      /* ISP enable */
    VI_PATH_RAW       = 2,      /* Capture raw data, for debug */
    VI_PATH_BUTT
}VI_DATA_PATH_E;
```

【成员】

成员名称	描述
VI_PATH_BYPASS	VI 内部 ISP Bypass。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VI_PATH_ISP	VI 内部 ISP Enable。
VI_PATH_RAW	VI 内部 ISP、CSC、LDC 等功能模块关闭，一般在捕获 Sensor 输入的 Raw Data 时使用。

【注意事项】

- YUV、带 ISP 的 Sensor 输入时，必须配置为 VI_PATH_BYPASS。
- 不带 ISP 的 Sensor 输入时，配置为 VI_PATH_ISP。
- VI_PATH_RAW 一般只用于 Debug，用来捕获 Sensor 输入的 Raw Data。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- VI_DEV_ATTR_EX_S

VI_DATA_TYPE_E

【说明】

VI 输入数据类型枚举。

【定义】

```
typedef enum hiVI_DATA_TYPE_E
{
    VI_DATA_TYPE_YUV = 0,
    VI_DATA_TYPE_RGB = 1,
    VI_DATA_TYPE_BUTT
} VI_DATA_TYPE_E;
```

【成员】

成员名称	描述
VI_DATA_TYPE_YUV	输入数据类型为 YUV，VI 前端一般接的是 AD。
VI_DATA_TYPE_RGB	输入数据类型为 RGB，VI 前端一般接的是 Sensor。

【注意事项】

无。

【相关数据类型及接口】

- VI_DEV_ATTR_S
- VI_DEV_ATTR_EX_S

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VI_EXT_CHN_ATTR_S

【说明】

定义 VI 扩展通道属性。

【定义】

```
typedef struct hiVI_EXT_CHN_ATTR_S
{
    VI_CHN   s32BindChn;
    SIZE_S    stDestSize;
    HI_S32    s32SrcFrameRate;
    HI_S32    s32FrameRate;
    PIXEL_FORMAT_E  enPixFormat;
}VI_EXT_CHN_ATTR_S;
```

【成员】

成员名称	描述
s32BindChn	扩展通道所绑定的通道。
stDestSize	目标图像大小。
s32SrcFrameRate	原始帧率，不能小于-1；建议设置为与对接 camera 一致的值。
s32FrameRate	目标帧率，必须小于或等于原始帧率，同时不能小于-1；原始帧率与目标帧率要么同时为-1，要么都不为-1，两者都为-1 时表示不进行帧率控制。
enPixFormat	像素存储格式支持 sp420 和 sp422。

【注意事项】

目标图像的大小即为扩展通道的输出宽高，扩展通道支持的输出宽高为小于 4096，其缩小倍数小于 13。

【相关数据类型及接口】

- HI_MPI_VI_SetChnAttr
- HI_MPI_VI_GetChnAttr

LDC_VIEW_TYPE_E

【说明】

定义镜头畸变校正的类型。

【定义】

```
typedef enum hiLDC_VIEW_TYPE_E
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    LDC_VIEW_TYPE_ALL = 0,
    LDC_VIEW_TYPE_CROP,
    LDC_VIEW_TYPE_BUTT,
} LDC_VIEW_TYPE_E;
```

【成员】

成员名称	描述
LDC_VIEW_TYPE_ALL	全模式，从校正后的图像按边缘获取最大矩形，并缩小到原始图像大小，边缘数据会尽可能保留。
LDC_VIEW_TYPE_CROP	裁剪模式，从校正后的图像中裁剪出原始图像大小，边缘数据可能会丢失。

【注意事项】

无。

【相关数据类型及接口】

- LDC_ATTR_S
- VI_LDC_ATTR_S

LDC_ATTR_S

【说明】

定义镜头畸变校正属性。

【定义】

```
typedef struct hiLDC_ATTR_S
{
    LDC_VIEW_TYPE_E enViewType;

    HI_S32 s32CenterXOffset;
    HI_S32 s32CenterYOffset;
    HI_S32 s32Ratio;
} LDC_ATTR_S;
```

【成员】

成员名称	描述
enViewType	畸变校正类型，裁剪模式、全模式。
s32CenterXOffset	畸变中心点相对图象中心点水平偏移，[-28, 28]。
s32CenterYOffset	畸变中心点相对图象中心点垂直偏移，[-14, 14]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
s32Ratio	畸变程度，[0, 511]。

【注意事项】

VI 采集图像超过 720P 时，s32Ratio 有效范围为[0, 192]。

【相关数据类型及接口】

LDC_VIEW_TYPE_E

VI_LDC_ATTR_S

【说明】

定义 VI 镜头畸变校正属性。

【定义】

```
typedef struct hiVI_LDC_ATTR_S
{
    HI_BOOL bEnable;
    LDC_ATTR_S stAttr;
}VI_LDC_ATTR_S;
```

【成员】

成员名称	描述
bEnable	Enable/Disable LDC 功能。
stAttr	LDC 详细属性。

【注意事项】

无。

【相关数据类型及接口】

- LDC_VIEW_TYPE_E
- LDC_ATTR_S
- HI_MPI_VI_SetLDCAttr
- HI_MPI_VI_GetLDCAttr

VI_CSC_TYPE_E

【说明】

定义 YUV 色彩空间标准。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef enum hiVI_CSC_TYPE_E
{
    VI_CSC_TYPE_601 = 0,
    VI_CSC_TYPE_709,
    VI_CSC_TYPE_BUTT,
} VI_CSC_TYPE_E;
```

【成员】

成员名称	描述
VI_CSC_TYPE_601	BT.601 色彩空间标准。
VI_CSC_TYPE_709	BT.709 色彩空间标准。

【注意事项】

YUV 色彩空间标准有两类：BT.601 和 BT.709。此枚举定义图像从 RGB 转换至 YUV 时所使用的色彩空间标准，通常 BT.601 用于在标清设备上显示，BT.709 用于在高清设备上显示。

【相关数据类型及接口】

无。

VI_CSC_ATTR_S

【说明】

定义 CSC 属性。

【定义】

```
typedef struct hiVI_CSC_ATTR_S
{
    VI_CSC_TYPE_E enViCscType;
    HI_U32 u32LumaVal;                  /* luminance:   [0 ~ 100] */
    HI_U32 u32ContrVal                  /* contrast :   [0 ~ 100] */;
    HI_U32 u32HueVal;                   /* hue      :   [0 ~ 100] */
    HI_U32 u32SatuVal;                  /* satuature:   [0 ~ 100] */
} VI_CSC_ATTR_S;
```

【成员】

成员名称	描述
enViCscType	YUV 色彩空间标准

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32LumaVal	亮度调节参数 默认值：50 取值范围：[0, 100]
u32ContrVal	对比度调节参数 默认值：50 取值范围：[0, 100]
u32HueVal	色调调节参数 默认值：50 取值范围：[0, 100]
u32SatuVal	饱和度调节参数 默认值：50 取值范围：[0, 100]

【注意事项】

无。

【相关数据类型及接口】

- VI_CSC_TYPE_E
- HI_MPI_VI_SetCSCAttr
- HI_MPI_VI_GetCSCAttr

VI_FLASH_MODE_E

【说明】

定义闪光灯模式。

【定义】

```
typedef enum hiVI_FLASH_MODE_E
{
    VI_FLASH_ONCE = 0,          /*Flah one time*/
    VI_FLASH_FREQ = 1,          /*Flash frequently*/
    VI_FLASH_MODE_BUTT
}VI_FLASH_MODE_E;
```

【成员】

成员名称	描述
VI_FLASH_ONCE	单闪模式，只闪一次。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VI_FLASH_FREQ	频闪模式，按配置的周期进行闪光。

【注意事项】

无。

【相关数据类型及接口】

VI_FLASH_CONFIG_S

VI_FLASH_CONFIG_S

【说明】

定义闪光灯配置属性。

【定义】

```
typedef struct hiVI_FlASH_CONFIG_S
{
    VI_FLASH_MODE_E enFlashMode;
    HI_U32 u32StartTime;
    HI_U32 u32Duration;
    HI_U32 u32CapFrmIndex;
    HI_U32 u32Interval;
}VI_FLASH_CONFIG_S;
```

【成员】

成员名称	描述
enFlashMode	闪光灯模式。
u32StartTime	闪光灯启动时间，单位为时钟周期。
u32Duration	闪光灯持续时间，单位为时钟周期。
u32CapFrmIndex	捕获帧，设定闪光后第几帧为捕获帧，0 表示当前帧； 取值范围：[0, 5]。
u32Interval	频闪周期，单位为帧。

【注意事项】

- 当 Trigger 之后的下一帧到来时，VI 设备会在场起始中断的 u32StartTime 时钟周期后开始触发闪光灯信号，并持续 u32Duration 个时钟周期。在根据 Sensor 时序进行配置时，u32StartTime 要包含行消隐区的时钟周期。时钟周期等同于像素数。
- u32StartTime+u32Duration 不能超过一帧图像总时钟周期。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 频闪模式时闪光灯信号会按照 u32Interval 周期多次触发。单闪模式下闪光灯仅触发一次，该值无效。
- 频闪模式下 u32Interval 必须大于 u32CapFrmIndex。

【相关数据类型及接口】

- VI_FLASH_MODE_E
- HI_MPI_VI_SetFlashConfig
- HI_MPI_VI_GetFlashConfig

VI_CHN_LUM_S

【说明】

定义通道亮度和结构体。

【定义】

```
typedef struct hiVI_CHN_LUM_S
{
    HI_U32 u32Lum;
    HI_U64 u64Pts;
} VI_CHN_LUM_S;
```

【成员】

成员名称	描述
u32Lum	亮度和统计值。
u64Pts	获取帧的 pts。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VI_GetChnLuma

VI_PORTMODE_E

【说明】

定义 VI 端口模式。

【定义】

```
typedef enum hiVI_PORTMODE_E
{
    VI_PORTMODE_D1 = 0,              /* D1/960H mode */
    VI_PORTMODE_720P,                /* 720P mode */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VI_PORTMODE_BUTT,
} VI_PORTMODE_E;
```

【成员】

成员名称	描述
VI_PORTMODE_D1	D1/960H 模式。
VI_PORTMODE_720P	720P 模式。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VI_SetPortMode
- HI_MPI_VI_GetPortMode

3.6 错误码

视频输入 API 错误码如表 3-17 所示。

表3-17 视频输入 API 错误码

错误代码	宏定义	描述
0xA0108001	HI_ERR_VI_INVALID_DEVID	视频输入设备号无效
0xA0108002	HI_ERR_VI_INVALID_CHNID	视频输入通道号无效
0xA0108003	HI_ERR_VI_INVALID_PARA	视频输入参数设置无效
0xA0108006	HI_ERR_VI_INVALID_NULL_PTR	输入参数空指针错误
0xA0108007	HI_ERR_VI_FAILED_NOTCONFIG	视频设备或通道属性未配置
0xA0108008	HI_ERR_VI_NOT_SUPPORT	操作不支持
0xA0108009	HI_ERR_VI_NOT_PERM	操作不允许
0xA010800C	HI_ERR_VI_NOMEM	分配内存失败
0xA010800E	HI_ERR_VI_BUF_EMPTY	视频输入缓存为空
0xA010800F	HI_ERR_VI_BUF_FULL	视频输入缓存为满
0xA0108010	HI_ERR_VI_SYS_NOTREADY	视频输入系统未初始化
0xA0108012	HI_ERR_VI_BUSY	视频输入系统忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA0108040	HI_ERR_VI_FAILED_NOTENABLE	视频输入设备或通道未启用
0xA0108041	HI_ERR_VI_FAILED_NOTDISABLE	视频输入设备或通道未禁用
0xA0108042	HI_ERR_VI_FAILED_CHNOTDISABLE	视频输入通道未禁用
0xA0108043	HI_ERR_VI_CFG_TIMEOUT	视频配置属性超时
0xA0108044	HI_ERR_VI_NORM_UNMATCH	不匹配
0xA0108045	HI_ERR_VI_INVALID_WAYID	视频通路号无效
0xA0108046	HI_ERR_VI_INVALID_PHYCHNID	视频物理通道号无效
0xA0108047	HI_ERR_VI_FAILED_NOTBIND	视频通道未绑定
0xA0108048	HI_ERR_VI_FAILED_BINDED	视频通道已绑定

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

4 视频输出.....4-1
4.1 概述.....4-1
4.2 重要概念.....4-2
4.3 API 参考.....4-7
4.4 数据类型.....4-142
4.5 错误码.....4-163

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 4-1 视频层属性的相关概念示意.....4-149
图 4-2 局部放大原理示意.....4-155

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 4-1 各芯片支持的显示设备、图形层和鼠标层情况 4-1
表 4-2 各款芯片 HD 和 SD 设备功能对比 4-2
表 4-3 视频输出 API 错误码 4-163

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

4 视频输出

4.1 概述

VOU（Video Output Unit）模块主动从内存相应位置读取视频和图形数据，并通过相应的显示设备输出。各芯片支持的显示设备、图形层和鼠标层情况如表 4-1 所示。

表4-1 各芯片支持的显示设备、图形层和鼠标层情况

芯片	支持的显示设备		图形层	鼠标层
	HD 设备	SD 设备		
Hi3531	支持 2 个高清设备（DHD0 和 DHD1） 支持 PIP（Picture In Picture）画面叠加	6 个标清设备（DSD0–DSD5） 支持通道之间按优先级叠加	5 个图形层（G0～G4）	2 个鼠标层（HC0 和 HC1）
	每个显示设备最多支持 64 路通道视频图像显示			
Hi3532	支持 1 个高清设备 DHD0 不支持 PIP 画面叠加	-	1 个图形层（G0）	1 个鼠标层（HC0）
Hi3521/Hi3520A	支持 1 个高清设备 DHD0 支持 PIP 画面叠加(PIP 视频层与 DSD0 视频层复用，即 PIP 时 DSD0 无输出)	2 个标清设备（DSD0 和 DSD1） 支持通道之间按优先级叠加	3 个图形层(G0～G2)	1 个鼠标层（HC0）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	支持的显示设备		图形层	鼠标层
	HD 设备	SD 设备		
	每个显示设备最多支持 32 路通道视频图像显示			
Hi3518/Hi3516C	支持 1 个标清设备（DSD1），最多支持 32 路通道视频图像显示		1 个图形层（G3）	不支持鼠标层
Hi3520D/Hi3515A/Hi3515C	支持 1 个高清设备 DHD0 支持 PIP 画面叠加(PIP 视频层与 DSD0 视频层复用，即 PIP 时 DSD0 无输出)	2 个标清设备（DSD0 和 DSD1） 支持通道之间按优先级叠加	3 个图形层 (G0～G2)	1 个鼠标层（HC0）
	每个显示设备，高清设备最多支持 16 路通道视频图像显示，标清设备最多支持 32 路通道视频图像显示			

4.2 重要概念

高清和标清显示设备 SDK 将高清和标清显示设备分别标示为 DHDx 和 DSDx。其中，x 为索引号，从 0 开始取值，表第几路高清/标清显示设备，例如第 0 路高清设备标示为 DHD0，第 0 路标清显示设备标示为 DSD0。所有高清和标清显示设备又可分别简称为 HD 和 SD 设备。各款芯片支持显示设备的情况请参见表 4-1。各款芯片 HD 和 SD 设备功能对比如表 4-2 所示。

表4-2 各款芯片 HD 和 SD 设备功能对比

芯片		最大分辨率	输出接口	缩放能力	支持通道数	PIP 显示	CSC 调节
Hi3531	HD	DHD0: 1920x1080 DHD1: 2560x1600	BT.1120 /HDMI /YPbPr /VGA	仅支持放大	64	支持 8 路（硬件 PIP 层）	支持
	SD	720x576	BT.656 /CVBS	不支持	64	TDE 叠加	支持
Hi3532	HD	1920x1080	BT.1120	不支持	64	不支持	支持

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片		最大分辨率	输出接口	缩放能力	支持通道数	PIP 显示	CSC 调节
	SD	不支持	不支持	不支持	不支持	不支持	不支持
Hi3521/ Hi3520A	HD	1920x1080	BT.1120 /HDMI /LCD /VGA	仅支持放大	32	支持 8 路（硬件 PIP 层）	支持
	SD	720x576	BT.656 /CVBS	不支持	32	TDE 叠加	支持
Hi3518/ Hi3516C	HD	不支持	不支持	不支持	不支持	不支持	不支持
	SD	1920x1080	BT.1120 /CVBS	不支持	32	不支持	支持
Hi3520D/ Hi3515A/ Hi3515C	HD	1920x2048	HDMI /VGA	仅支持放大	16	支持 1 路（硬件 PIP 层）	支持
	SD	720x576	CVBS	不支持	32	TDE 叠加	支持

- 缩放和显示

  每个高清设备总共有 x 个通道（x 的取值请参见表 4-1），通道属性中的优先级决定通道属于设备视频层或 PIP 层。设备视频层内的通道不可重叠，设备 PIP 层内的通道亦不可重叠。高清设备的通道没有缩放能力，需借助外部的 VPSS 进行缩放。VI 或 VDEC 的源图像经由 VPSS 缩放后，输出到 VO 的某一通道显示。如果输出给 VO 通道的图像超过区域大小，将会对图像进行裁剪。高清设备支持对整个视频层进行放大，不支持裁剪。PIP 层例外，不支持对 PIP 层放大，也不支持裁剪。

  标清设备支持 x 个通道（x 的取值请见表 4-1），标清设备的通道通过 TDE 实现缩放，并叠加为一整幅图像，并将这一整幅图像的数据配给硬件显示。标清设备不支持对整个视频层缩放，允许裁剪。各款芯片缩放和显示情况请参见表 4-2。

注意

- **裁剪是指视频层属性中画布大小 stImageSize 中的宽高大于显示设备分辨率大小 stDispRect 中的宽高时，对画布进行裁剪以符合显示分辨率大小。**
- **缩放是指视频层属性中显示设备分辨率大小 stDispRect 中的宽高大于画布大小 stImageSize 中的宽高时，对画布进行放大以符合显示分辨率大小。**

- 通道优先级
  - 高清显示设备支持多个通道同时输出显示，通道优先级只有 0 与 1 两级，优先级描述只在 PIP 开启时有效；

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 标清显示设备支持多个通道同时输出显示，按照优先级顺序对输出图像进行叠加，当各个通道的画面有重叠区域时，优先级高的图像显示在上层，如果各个通道优先级一致，则通道号越大的默认优先级越高。

Hi3532 的通道优先级仅允许设置为 0。

- PIP（Picture In Picture）画面叠加

  PIP 实现的是高清设备上的 PIP 层画面与原视频层画面叠加。高清设备上原视频层与 PIP 层都不支持各个通道的叠加，高清设备上通道优先级只有 0 与 1 两级，通道优先级为 0 的通道显示在原视频层，通道优先级为 1 的通道显示在 PIP 层。PIP 层画面总是叠加在原视频层画面之上。各款芯片支持 PIP 画面叠加情况以及支持的 PIP 通道路数请参见表 4-2。

- 分辨率

  分辨率主要有以下 3 种概念：

  - 设备分辨率指该设备的输出有效像素点数，由设备时序决定。
  - 显示分辨率指画面在显示设备上的有效显示区域。
  - 图像分辨率指图像本身的有效像素点数。

- 局部放大

  - 标清设备支持将显示画面上的一部分图像进行放大显示，放大显示的源区域从源图像上截取，放大显示的目标区域是该显示通道的通道大小；
  - 高清设备对单个通道图像的缩放是通过 VPSS 模块实现，因此高清设备需要配合 VPSS，以实现对源图像的局部放大。

- 图形层绑定

  图形层绑定是指芯片支持将特定的某个图形层或鼠标层绑定到某个设备上。

  - Hi3531 支持 5 个图形层（G0～4），2 个鼠标层（HC0～1）。其中图形层 G0～G3 分别固定绑定到 DHD0、DHD1、DSD0、DSD1 上，G4、HC0、HC1 可以选择绑定到 DHD0、DHD1、DSD0、DSD1 上。

    注意：

    a. G4、HC0、HC1 同时只能绑定到一个设备，即如果 G4 绑定到 DHD0，则其不能再绑定到 DHD1、DSD0、DSD1。

    b. HC0、HC1 只能绑定其中之一到 DHD0，若 HC0 绑定到 DHD0，HC1 必须绑定到 DHD1、DSD0、DSD1；若 HC1 绑定到 DHD0，HC0 必须绑定到 DHD1、DSD0、DSD1。

    c. G4、HC0、HC1 可以选择绑定到 DHD0、DHD1、DSD0、DSD1，如果从绑定当前的设备切换到绑定另外一个设备，不能实时变更，必须先禁用当前需要切换绑定关系的视频层，即如果 G4 要从 DHD0 切换到 DHD1，则必须先禁用 G4，再绑定 DHD1。

  - Hi3532 支持 1 个图形层（G0），1 个鼠标层（HC0），并均固定绑定到 DHD0 上。
  - Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C 支持 3 个图形层（G0~G2），1 个鼠标层（HC0）。其中 G0~G2 图形层分别固定绑到 DHD0、DSD0、DSD1 上，HC0 可以选择绑定到 DHD0、DSD0、DSD1。
  - Hi3518/Hi3516C 支持 1 个图形层（G3），并固定绑定到 DSD1 上。

- PIP 层绑定

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

PIP 层绑定是指将某个特定标清设备的视频层绑定到高清设备上，与高清设备上原有的视频层进行画面叠加。此时该视频层作为高清设备的 PIP 层，不能再作为其他任何设备的视频层使用，即这个视频层可以作为高清设备的 PIP 层，或者标清设备的视频层，但不能同时输出。如果从绑定当前的设备切换到绑定另外一个设备，必须先禁用 PIP 层，再切换绑定的设备，即如果 PIP 层要从 DHD0 切换到 DHD1，必须先禁用 PIP 层，再解绑定 DHD0，最后绑定 DHD1。如果 PIP 层没有绑定到高清设备上，则可以作为它本身默认的标清设备输出。

- Hi3531 VO 支持一个 PIP 层与高清设备上的视频层进行画面叠加。PIP 层可以选择绑定到 DHD0、DHD1 设备，同时只能绑定到其中一个设备上。
- Hi3532/Hi3518/Hi3516C VO 不支持 PIP 层。
- Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C 支持将 DSD0 的视频层作为 PIP 层，与高清设备 DHD0 上视频层进行画面叠加。

- WBC 视频回写。

  视频回写功能是指将高清设备屏幕上显示的画面，包括视频层、图形层、鼠标层等叠加后的图像，经缩放后回写到内存中。用户可以将视频回写理解为一个视频源，通过将这个视频源与标清设备绑定，将回写图像重新送给标清设备显示，以实现同源显示。WBC 回写只支持缩小，不能放大，即高清设备的分辨率要必须大于等于 WBC 分辨率。WBC 支持 semi-plannar422、semi-plannar420 两种数据格式，源的数据格式可以和 WBC 设置的数据格式不一致。

  - Hi3531VO 支持一个高清设备（DHD1）具备视频回写功能。调用系统绑定接口时，WBC 视频回写源的设备号为 DHD1，通道号固定为 0，模块 ID 为 VOU。当 WBC 设置为 semi-plannar422 数据格式时、最大允许 16 倍缩小；当设置为 semi-plannar420 数据格式时，最大允许 8 倍缩小。
  - Hi3532/Hi3518/Hi3516C VO 不支持 WBC 回写。
  - Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C VO 支持高清设备 DHD0 具备视频回写功能。调用系统绑定接口时，WBC 视频回写源的设备号为 DHD0，通道号固定为 0，模块 ID 为 VOU。当 WBC 设置为 semi-plannar422 数据格式时、最大允许 16 倍缩小；当设置为 semi-plannar420 数据格式时，最大允许 8 倍缩小。

- 视频级联

  从片预览图像的输出以 BT.1120 数据格式连接到下级芯片的视频输入端，通过这样的方式连接多个芯片，将芯片上的图像传输到主片同时输出显示。

  - Hi3531/Hi3532 的 HD 设备支持视频级联。视频级联的详细信息请参见《Hi3531/Hi3532 视频级联应用指南》。
  - Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C 不支持视频级联功能。

- VGA 兼容性

  可以通过以下步骤实现 VGA 对接：

  VGA 接口中有 I^2C 连线，可以通过 GPIO 模拟 I^2C，或者直接使用 I^2C。在确认 VGA 显示器已连接的情况下，通过 I^2C 读设备 0xA0 的 0x0～0x70 的内容（地址与硬件无关，属于芯片统一地址），获取 EDID 信息。

  解析返回的 EDID 值，列出设备可能支持的时序。方法如下：

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 通过 I²C 读设备 0xA0 的 0x23 的内容，获取显示器的建立时序 1，7～0bit 分别对应 720x400@70 Hz，720x400@88 Hz，640x480@60 Hz，640x480@67 Hz，640x480@72 Hz，640x480@75 Hz，800x600@56 Hz，800x600@60 Hz
- 通过 I²C 读设备 0xA0 的 0x24 的内容，获取显示器建立时序 2，7～0bit 分别对应 800x600@72 Hz，800x600@75 Hz，832x624@75 Hz，1024x768@87 Hz (交错/隔行)，1024x768@60 Hz，1024x768@70 Hz，1024x768@75 Hz，1280x1024@75 Hz。
- 通过 I²C 读设备 0xA0 的 0x25 的内容，获取制造商保留的时序，00h 表示无，7bit 对应 1152x870 @ 75 Hz (Mac II，苹果)。
- 通过 I²C 读设备 0xA0 的 0x26～0x37 的内容，获取标准时序识别的结果，其中每两个字节为一组，即 0x26 和 0x27，0x28 和 0x29 等。第一个字节（例如 0x26）表示水平结果，读到的值加上 31，再乘上 8，得到水平结果。第二个字节（例如 0x27）5～0bit 表示垂直频率，读到的值加上 60，得到垂直频率。第二个字节 7～6bit 表示外观比例，00=16:10，01=4:3，10=5:4，11=16:9

用户可参考具体协议文档《E-EDID Standard.pdf》

如果分析返回的协议合法，则检查显示器是否支持将要配置的时序。如果分析返回的协议非法，则可能显示器未实现 EDID 协议，用户需配置 VO 模块支持的 VGA 时序，直到 VGA 输出正常。

- 虚拟设备

  虚拟设备是软件上的概念，没有实际对应的物理设备。

  - 内部处理流程：虚拟设备和标清设备相同，标清设备的通道通过 TDE 实现缩放，并叠加为一整幅图像。
  - 发送虚拟设备图像：虚拟设备支持通过用户态线程调用 HI_MPI_VO_GetScreenFrame 接口获取虚拟设备图像，并发送给其他模块；也支持 SYS_Bind 的方式在内核态向其他模块发送虚拟设备图像，但是禁止调用 HI_MPI_SYS_Bind 接口将 VO 虚拟设备绑定到 VO 设备自身。
  - 时序限制：虚拟设备的时序不限制为 P 制或 N 制，并将根据设置的时序，决定 TDE 最终叠加出来的图像大小，例如用户将虚拟设备的时序设置为 1080P60，虚拟设备将发送 1920x1080 大小的图像给其他模块。用户在使用虚拟设备时需充分考虑 TDE 的性能和实际的总业务量大小。
  - 将 VPSS 绑定虚拟设备：虚拟设备和标清设备相同，选择通过 VPSS_BYPASS_CHN 定义，或者通过 HI_MPI_VPSS_SetChnMode 将相应的 VPSS 通道设定为 VPSS_CHN_MODE_USER。

  Hi3531 和 Hi3532 支持 4 个虚拟设备，设备号为 10～13。Hi3521、Hi3520A 和 Hi3520D/Hi3515A/Hi3515C 支持 4 个虚拟设备，设备号为 3～6。

  Hi3518/Hi3516C 不支持虚拟设备。

注意

高清/级联设备不要直接绑定 VDEC，否则解码回放会失控。

标清/虚拟设备不支持绑定 AUTO 模式下的 VPSS 物理通道。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

4.3 API 参考

视频输出（VO）实现启用视频输出设备或通道、发送视频数据到输出通道等功能。

该功能模块提供以下 MPI：

设备操作：

- HI_MPI_VO_Enable：启用视频输出设备。
- HI_MPI_VO_Disable：禁用视频输出设备。
- HI_MPI_VO_SetPubAttr：设置视频输出设备的公共属性。
- HI_MPI_VO_GetPubAttr：获取视频输出设备的公共属性。
- HI_MPI_VO_CloseFd：关闭所有视频输出设备的 Fd。
- HI_MPI_VO_SetDevFramerate：设置设备用户时序下设备帧率。
- HI_MPI_VO_GetDevFramerate：获取设备帧率。
- HI_MPI_VO_EnableSdTdeBypass：使能标清设备的 TDE bypass 功能。
- HI_MPI_VO_DisableSdTdeBypass：关闭标清设备的 TDE bypass 功能。

视频层操作：

- HI_MPI_VO_EnableVideoLayer：使能视频层。
- HI_MPI_VO_DisableVideoLayer：禁止视频层。
- HI_MPI_VO_SetVideoLayerAttr：设置视频层属性。
- HI_MPI_VO_GetVideoLayerAttr：获取视频层属性。

PIP 操作：

- HI_MPI_VO_PipLayerBindDev：绑定 PIP 层到指定设备。
- HI_MPI_VO_PipLayerUnBindDev：把 PIP 层从指定设备解绑定。
- HI_MPI_VO_SetPipLayerAttr：设置 PIP 层属性。
- HI_MPI_VO_GetPipLayerAttr：获取 PIP 层属性。
- HI_MPI_VO_EnablePipLayer：使能 PIP 层。
- HI_MPI_VO_DisablePipLayer：禁止 PIP 层。

通道操作：

- HI_MPI_VO_EnableChn：启用指定的视频输出通道。
- HI_MPI_VO_DisableChn：禁用指定的视频输出通道。
- HI_MPI_VO_SetChnAttr：设置指定视频输出通道的属性。
- HI_MPI_VO_GetChnAttr：获取指定视频输出通道的属性。
- HI_MPI_VO_SetChnDisplayWindow：配置指定视频输出通道的显示窗口。
- HI_MPI_VO_GetChnDisplayWindow：获取指定视频输出通道的显示窗口
- HI_MPI_VO_SetChnDispPos：设置指定视频输出通道的显示位置。
- HI_MPI_VO_GetChnDispPos：获取指定视频输出通道的显示位置。
- HI_MPI_VO_SetChnField：设置指定视频输出通道的帧场显示策略。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VO_GetChnField：获取指定视频输出通道的帧场显示策略。
- HI_MPI_VO_SendFrame：将视频图像送入指定视频输出通道显示。
- HI_MPI_VO_SendFrameTimeOut：将视频图像送入指定输出通道显示。
- HI_MPI_VO_SetChnFrameRate：设置指定视频输出通道的显示帧率。
- HI_MPI_VO_GetChnFrameRate：获取指定视频输出通道的显示帧率。
- HI_MPI_VO_ChnPause：暂停指定的视频输出通道。
- HI_MPI_VO_ChnResume：恢复指定的视频输出通道。
- HI_MPI_VO_ChnStep：单帧播放指定的视频输出通道。
- HI_MPI_VO_ChnRefresh：刷新指定的视频输出通道。
- HI_MPI_VO_SetZoomInWindow：设置视频输出局部放大窗口。
- HI_MPI_VO_GetZoomInWindow：获取视频输出局部放大窗口参数。
- HI_MPI_VO_GetChnPts：获取指定视频输出通道当前显示图像的时间戳。
- HI_MPI_VO_SetAttrBegin：设置属性开始。
- HI_MPI_VO_SetAttrEnd：设置属性结束。
- HI_MPI_VO_ChnShow：设置显示通道。
- HI_MPI_VO_ChnHide：设置隐藏通道。
- HI_MPI_VO_QueryChnStat：查询视频输出通道状态。
- HI_MPI_VO_GetChnFrame：获取输出通道图像数据。
- HI_MPI_VO_ReleaseChnFrame：释放输出通道图像数据。
- HI_MPI_VO_ClearChnBuffer：清空指定输出通道的缓存 buffer 数据。
- HI_MPI_VO_SetChnDispThreshold：设置视频输出通道的显示门限值。
- HI_MPI_VO_GetChnDispThreshold：获取视频输出通道的显示门限值。
- HI_MPI_VO_EnableChnDoubleFrame：使能通道倍帧功能。
- HI_MPI_VO_DisableChnDoubleFrame：禁用通道倍帧功能。

显示操作：

- HI_MPI_VO_GetScreenFrame：获取输出屏幕图像数据。
- HI_MPI_VO_ReleaseScreenFrame：释放输出屏幕图像数据。
- HI_MPI_VO_SetDevCSC：设置设备输出图像效果。
- HI_MPI_VO_GetDevCSC：获取设备输出图像效果。
- HI_MPI_VO_SetVgaParam：设置设备 VGA 输出图像效果。
- HI_MPI_VO_GetVgaParam：获取设备 VGA 输出图像效果。
- HI_MPI_VO_SetDispBufLen：设置显示缓冲的长度。
- HI_MPI_VO_GetDispBufLen：获取显示缓冲的长度。
- HI_MPI_VO_SetPlayToleration：设置播放容忍度。
- HI_MPI_VO_GetPlayToleration：获取播放容忍度。
- HI_MPI_VO_SetVtth：设置设备垂直时序中断门限。
- HI_MPI_VO_GetVtth：设置设备垂直时序中断门限。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

回写操作：

- HI_MPI_VO_EnableWbc：使能视频回写。
- HI_MPI_VO_DisableWbc：禁用视频回写。
- HI_MPI_VO_SetWbcAttr：设置视频回写属性。
- HI_MPI_VO_GetWbcAttr：获取视频回写属性。
- HI_MPI_VO_SetWbcMode：设置视频回写模式。
- HI_MPI_VO_GetWbcMode：获取视频回写模式。
- HI_MPI_VO_SetWbcDepth：设置获取视频回写图像的缓存深度。
- HI_MPI_VO_GetWbcDepth：获取视频回写图像的缓存深度。
- HI_MPI_VO_WbcGetScreenFrame：获取视频回写图像。
- HI_MPI_VO_WbcReleaseScreenFrame：释放视频回写图像。

图形层操作：

- HI_MPI_VO_GfxLayerBindDev：图形层绑定设备。
- HI_MPI_VO_GfxLayerUnBindDev：图形层解绑定设备。
- HI_MPI_VO_SetGfxLayerCSC：设置图形层输出图像效果。
- HI_MPI_VO_GetGfxLayerCSC：获取图形层输出图像效果。

级联操作：

- HI_MPI_VO_SetCascadeAttr：设置级联属性。
- HI_MPI_VO_GetCascadeAttr：获取级联属性。
- HI_MPI_VO_EnableCascadeDev：使能级联扩展设备。
- HI_MPI_VO_DisableCascadeDev：禁止级联扩展设备。
- HI_MPI_VO_SetCascadePattern：设置视频级联画面样式。
- HI_MPI_VO_GetCascadePattern：获取视频级联画面样式。
- HI_MPI_VO_CascadePosBindChn：绑定级联区域与视频输出通道。
- HI_MPI_VO_CascadePosUnBindChn：解绑定级联区域与视频输出通道。
- HI_MPI_VO_EnableCascade：使能视频级联。
- HI_MPI_VO_DisableCascade：禁止视频级联。

HI_MPI_VO_Enable

【描述】

启用视频输出设备。

【语法】

```
HI_S32 HI_MPI_VO_Enable (VO_DEV VoDev);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_DEV_NOT_CONFIG	视频输出设备没有配置。
HI_ERR_VO_DEV_HAS_ENABLED	视频输出设备已经使能。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 由于系统没有初始化设备为使能状态，所以在使用视频输出功能前必须先进行设备使能操作。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 在调用设备使能前，必须对设备公共属性进行配置，否则返回设备未配置错误。
- 为适应开机画面与正常操作界面间顺畅切换，此处需要检查 VO 硬件是否已经使能，如果已使能则返回成功，且沿用已有时序配置。如果希望更改 VO 的时序配置，则需要先调用 HI_MPI_VO_Disable 接口，强制关闭 VO 硬件后再使能。
- 各个 DEV 的使用说明详见 VO_DEV 章节。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 0;
VO_PUB_ATTR_S stPubAttr;

s32Ret = HI_MPI_VO_GetPubAttr(VoDev, &stPubAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Get device attributes failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
}

stPubAttr.u32BgColor = 0xff;
stPubAttr.enIntfType = VO_INTF_VGA;
stPubAttr.enIntfSync = VO_OUTPUT_1280x1024_60;

s32Ret = HI_MPI_VO_SetPubAttr(VoDev, &stPubAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Set device attributes failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_Enable(VoDev);
if (s32Ret != HI_SUCCESS)
{
    printf("Enable vo dev %d failed with errno %#x!\n", VoDev, s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_Disable(VoDev);
if (s32Ret != HI_SUCCESS)
{
    printf("Enable vo dev %d failed with errno %#x!\n", VoDev, s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_CloseFd();
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
if (s32Ret != HI_SUCCESS)
{
    printf("Some device is not disable with errno %#x!\n", VoDev, s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_Disable

HI_MPI_VO_Disable

【描述】

禁用视频输出设备。

【语法】

```
HI_S32 HI_MPI_VO_Disable(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_VIDEO_NOT_DISABLE	视频层未禁止。
HI_ERR_VO_WBC_NOT_DISABLE	WBC 未禁止。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 设备禁止前必须先禁止该设备上的视频层。
- 设备禁止前，如果有使能 WBC，则必须关闭该使能。
- 调用设备使能接口后，如果未调用该接口进行禁止，则 VO 设备将一直保持使能状态，并且下次设置设备属性时不会生效。
- 设备禁止后需要重新设置设备公共属性，才可使能设备。

【举例】

请参见 HI_MPI_VO_Enable 的举例。

【相关主题】

HI_MPI_VO_Enable

HI_MPI_VO_SetPubAttr

【描述】

配置视频输出设备的公共属性。

【语法】

```
HI_S32 HI_MPI_VO_SetPubAttr(VO_DEV VoDev, const VO_PUB_ATTR_S
*pstPubAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstPubAttr	视频输出设备公共属性结构体指针。 静态属性。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	VoDev 取值范围：[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。 CVBS 输出时不支持自动负载检测。
Hi3532	VoDev 取值范围：[0]，[10, 13]。 CVBS 输出时不支持自动负载检测。
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	VoDev 取值范围：[0, VO_MAX_DEV_NUM)。 CVBS 输出时不支持自动负载检测。
Hi3518/Hi3516C	VoDev 取值范围：[0, VO_MAX_DEV_NUM)。 CVBS 输出时支持自动负载检测，详见【注意】栏。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_NULL_PTR	参数指针为空。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_ILLEGAL_PARAM	参数无效。
HI_ERR_VO_NOT_SUPPORT	系统不支持的视频输出属性。
HI_ERR_VO_DEV_HAS_ENABLED	设备已使能。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 视频输出设备属性为静态属性，必须在执行 HI_MPI_VO_Enable 前配置。
- 各个 DEV 的使用说明详见 VO_DEV 章节。
- 视频输出设备属性的使用说明详见 VO_PUB_ATTR_S 章节。
- 当设置视频输出设备属性的输出接口类型 enIntfType 为 VO_INTF_CVBS 时，Hi3518/Hi3516C 支持自动负载检测。
  - 基本原理：周期性检测 CVBS 接口是否有线缆连接，当检测到没有线缆连接时，驱动会自动将 DAC 下电从而达到降低功耗的目的。
  - 配置检测周期：可通过配置 VO 模块参数来配置负载检测周期，检测周期以帧中断为单位，例如 insmod hi3518_vou.ko detectCycle=50（以 Hi3518 为例），当配置视频输出设备属性的输出时序 enIntfSync 为 VO_OUTPUT_PAL 时，检测周期为 2 秒（50/25），当配置输出时序 enIntfSync 为 VO_OUTPUT_NTSC 时，检测周期为 1.67 秒（50/30）。当不带任何模块参数加载 VO 驱动时，负载检测周期默认为 30，当配置 VO 模块参数 detectCycle 为 0 时，将关闭自动负载检测功能。

【举例】

请参见 HI_MPI_VO_Enable 的举例。

【相关主题】

HI_MPI_VO_GetPubAttr

HI_MPI_VO_GetPubAttr

【描述】

获取视频输出设备的公共属性。

【语法】

```
HI_S32 HI_MPI_VO_GetPubAttr(VO_DEV VoDev, VO_PUB_ATTR_S *pstPubAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstPubAttr	视频输出设备公共属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/ Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_NULL_PTR	参数指针为空。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_NOT_PERMIT	不允许的设备，这里指级联设备。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

建议在设置设备公共属性前先获取属性，这样就可以只设置需要改变的配置项。

【举例】

请参见 HI_MPI_VO_Enable 的举例。

【相关主题】

HI_MPI_VO_SetPubAttr

HI_MPI_VO_CloseFd

【描述】

关闭所有视频输出设备的 Fd。

【语法】

```
HI_S32 HI_MPI_VO_CloseFd(HI_VOID);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_BUSY	设备忙，即有设备还未禁止。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

该接口必须在所有设备都禁止的情况下使用。

【举例】

请参见 HI_MPI_VO_Enable 的举例。

【相关主题】

无。

HI_MPI_VO_SetDevFramerate

【描述】

设置设备用户时序下设备帧率。

【语法】

```
HI_S32 HI_MPI_VO_SetDevFramerate(VO_DEV VoDev, HI_U32 u32Framerate);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
u32Framerate	设备帧率。 取值范围：只能配置为 25、30、50、60。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 只能在用户时序下使用。
- 只能在调用 HI_MPI_VO_SetPubAttr 后、HI_MPI_VO_Enable 前调用。

【举例】

无。

【相关主题】

HI_MPI_VO_GetDevFramerate

HI_MPI_VO_GetDevFramerate

【描述】

获取设备帧率。

【语法】

```
HI_S32 HI_MPI_VO_GetDevFramerate(VO_DEV VoDev, HI_U32 *pu32Framerate);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pu32Framerate	u32 类型的指针。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VO_SetDevFramerate

HI_MPI_VO_EnableSdTdeBypass

【描述】

使能标清设备的 TDE bypass 功能。

【语法】

```
HI_S32 HI_MPI_VO_EnableSdTdeBypass (VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 只能在标清设备使用。
- 只有当前设备仅使能一个通道，并且输入图像与显示尺寸一致才能使用此接口。

【举例】

无。

【相关主题】

无。

HI_MPI_VO_DisableSdTdeBypass

【描述】

关闭标清设备的 TDE bypass 功能。

【语法】

```
HI_S32 HI_MPI_VO_DisableSdTdeBypass(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_EnableVideoLayer

【描述】

使能视频层。

【语法】

```
HI_S32 HI_MPI_VO_EnableVideoLayer (VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A /Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_DEV_NOT_ENABLE	视频输出设备未使能。
HI_ERR_VO_VIDEO_NOT_CONFIG	视频层未配置。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_NO_MEM	没有可用的内存。
HI_ERR_VO_NOT_PERMIT	不允许的设备，这里指级联设备。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 视频层使能前必须保证该视频层所在的设备处于使能状态。
- 视频层使能前必须保证该视频层已经配置。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 0;
RECT_S stRect;
VO_VIDEO_LAYER_ATTR_S stLayerAttr;

s32Ret = HI_MPI_VO_GetVideoLayerAttr(VoDev, &stLayerAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Get video layer attributes failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}
stLayerAttr.stDispRect.s32X      = 0;
stLayerAttr.stDispRect.s32Y      = 0;
stLayerAttr.stDispRect.u32Width  = 720;
stLayerAttr.stDispRect.u32Height = 576;

stLayerAttr.stImageSize.u32Width  = 720;
stLayerAttr.stImageSize.u32Height = 576;
stLayerAttr. u32DispFrmRt = 25;
stLayerAttr. enPixFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_422;

s32Ret = HI_MPI_VO_SetVideoLayerAttr(VoDev, &stLayerAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Set video layer attributes failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
s32Ret = HI_MPI_VO_EnableVideoLayer(VoDev);
if (s32Ret != HI_SUCCESS)
{
    printf("Enable video layer failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_DisableVideoLayer(VoDev);
if (s32Ret != HI_SUCCESS)
{
    printf("Disable video layer failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_DisableVideoLayer

HI_MPI_VO_DisableVideoLayer

【描述】

禁止视频层。

【语法】

```
HI_S32 HI_MPI_VO_DisableVideoLayer(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3521/Hi3520A/Hi3520D/ Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_CHN_NOT_DISABLE	该视频层上还有未禁止的通道。
HI_ERR_VB_BUSY	VB 资源没有释放。
HI_ERR_VO_WBC_NOT_DISABLE	WBC 未禁止。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 视频层禁止前必须保证其上的通道全部禁止。
- 在禁止视频层时，如果用户没有释放从 VO 获取的图像 buffer 资源，该接口会返回 HI_ERR_VB_BUSY 的错误码，表示 VO 创建的 VB 资源没有释放。多见于用户调用获取屏幕图像未释放的情况下。
- 在禁止视屏层时，如果用户选择了 WBC 数据源为 VO_WBC_DATASOURCE_VIDEO 模式，则必须保证将该设备的 WBC 功能关闭使能。

【举例】

请参见 HI_MPI_VO_EnableVideoLayer 的举例。

【相关主题】

HI_MPI_VO_EnableVideoLayer

HI_MPI_VO_SetVideoLayerAttr

【描述】

设置视频层属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VO_SetVideoLayerAttr(VO_DEV VoDev, const
VO_VIDEO_LAYER_ATTR_S *pstLayerAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstLayerAttr	视频层属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_NOT_PERMIT	不允许的设备，这里指级联设备。
HI_ERR_VO_ILLEGAL_PARAM	无效参数。
HI_ERR_VO_VIDEO_NOT_DISABLE	视频层未禁用。
HI_ERR_VO_NULL_PTR	参数指针为空。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_DEV_NOT_CONFIG	设备未设置。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 设置视频层属性必须在视频层禁止的情况下进行。
- 视频层属性的使用说明详见 VO_VIDEO_LAYER_ATTR_S 章节。

【举例】

请参见 HI_MPI_VO_EnableVideoLayer 的举例。

【相关主题】

HI_MPI_VO_GetVideoLayerAttr

HI_MPI_VO_GetVideoLayerAttr

【描述】

获取视频层属性。

【语法】

```
HI_S32 HI_MPI_VO_GetVideoLayerAttr(VO_DEV VoDev, VO_VIDEO_LAYER_ATTR_S
*pstLayerAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstLayerAttr	视频层属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。其中 8 和 9 除外。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_NULL_PTR	参数指针为空。
HI_ERR_VO_NOT_PERMIT	不允许的设备，这里指级联设备。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 即使不调用 HI_MPI_VO_EnableVideoLayer 和 HI_MPI_VO_SetVideoLayerAttr 也可获取视频层属性。
- 建议在设置视频层属性前先调用此接口获取视频层属性。

【举例】

请参见 HI_MPI_VO_EnableVideoLayer 的举例。

【相关主题】

HI_MPI_VO_SetVideoLayerAttr

HI_MPI_VO_PipLayerBindDev

【描述】

绑定 PIP 层到指定设备。

【语法】

```
HI_S32 HI_MPI_VO_PipLayerBindDev(VO_DEV VoTargetDev);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VoTargetDev	PIP 要绑定的设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_DEV_HAS_ENABLED	PIP 层已经使能。
HI_ERR_VO_NOT_SUPPORT	芯片不支持。
HI_ERR_VO_DEV_HAS_BINDED	PIP 层已经绑定其他设备。
HI_ERR_VO_NOT_PERMIT	不允许的调用。绑定到标清设备、使能级联模式的高清设备、复用的标清设备已使能等情况下不允许调用本接口。

【差异说明】

芯片	描述
Hi3531	DSD5 的视频层可以作为高清设备的 PIP 层使用。 VoTargetDev 取值范围：[0, 1]
Hi3532	不支持该接口
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	DSD0 的视频层可以作为高清设备的 PIP 层使用。 VoTargetDev 取值范围：0
Hi3518/Hi3516C	不支持该接口。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 如果高清设备的 PIP 层和 DSDx 的视频层复用，且 DSDx 设备已使能时，若需要启动把 PIP 绑定到高清设备上，绑定会返回错误 HI_ERR_VO_DEV_HAS_ENABLED，即绑定前需要先关闭 DSDx 设备。
- 绑定时允许重复绑定到同样的设备，且绑定时不允许 PIP 层使能。
- 切换绑定关系时，需要首先解绑定。
- 如果高清设备的 PIP 层和 DSDx 的视频层复用，设置 PIP 层属性和设置 DSDx 的视频层属性时，针对的是同一视频层设置属性，视频层属性以最后设置的为准，建议先设置视频层属性再使能视频层。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 0;
RECT_S stRect;
VO_VIDEO_LAYER_ATTR_S stLayerAttr;

s32Ret = HI_MPI_VO_PipLayerBindDev(VoDev);
if (s32Ret != HI_SUCCESS)
{
   printf("Pip video layer bind to dev %d failed with errno %#x!\n",
VoDev, s32Ret);
   return HI_FAILURE;
}

s32Ret = HI_MPI_VO_GetPipLayerAttr(&stLayerAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Get pip video layer attributes failed with errno %#x!\n",
s32Ret);
   return HI_FAILURE;
}

stLayerAttr.stDispRect.s32X      = 0;
stLayerAttr.stDispRect.s32Y      = 0;
stLayerAttr.stDispRect.u32Width  = 720;
stLayerAttr.stDispRect.u32Height = 576;

stLayerAttr.stImageSize.u32Width  = 720;
stLayerAttr.stImageSize.u32Height = 576;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
stLayerAttr. u32DispFrmRt = 25;
stLayerAttr. enPixFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_422;

s32Ret = HI_MPI_VO_SetPipLayerAttr(&stLayerAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set pip video layer attributes failed with errno %#x!\n",
s32Ret);
   return HI_FAILURE;
}

s32Ret = HI_MPI_VO_EnablePipLayer();
if (s32Ret != HI_SUCCESS)
{
   printf("Enable pip video layer failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}

s32Ret = HI_MPI_VO_DisablePipLayer();
if (s32Ret != HI_SUCCESS)
{
   printf("Disable pip video layer failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}

s32Ret = HI_MPI_VO_PipLayerUnBindDev(VoDev);
if (s32Ret != HI_SUCCESS)
{
   printf("Pip video layer unbind to dev %d failed with errno %#x!\n",
VoDev, s32Ret);
   return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_PipLayerUnBindDev

HI_MPI_VO_PipLayerUnBindDev

【描述】

把 PIP 层从指定设备解绑定。

【语法】

```
HI_S32 HI_MPI_VO_PipLayerUnBindDev(VO_DEV VoTargetDev);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VoTargetDev	PIP 需要解绑定的设备号。 取值范围：[0, VO_MAX_DEV_NUM]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_VIDEO_NOT_DISABLE	PIP 层未关闭。
HI_ERR_VO_NOT_SUPPORT	芯片不支持

【差异说明】

芯片	描述
Hi3531	DSD5 的视频层可以作为高清设备的 PIP 层使用。
Hi3532	不支持该接口。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DSD0 的视频层可以作为高清设备的 PIP 层使用。
Hi3518/Hi3516C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 把 PIP 从高清设备上解绑定前，需要先调用 HI_MPI_VO_DisablePipLayer 关闭 PIP 层。
- 重复解绑定返回成功，另外不强制要求传入的形参 VoTargetDev 必须是之前 PIP 层绑定过的设备，调用此接口则视为解除 PIP 层的绑定关系。

【举例】

请参见 HI_MPI_VO_PipLayerBindDev 的举例。

【相关主题】

HI_MPI_VO_PipLayerBindDev

HI_MPI_VO_SetPipLayerAttr

【描述】

设置 PIP 视频层属性。

【语法】

```
HI_S32 HI_MPI_VO_SetPipLayerAttr(const VO_VIDEO_LAYER_ATTR_S
*pstLayerAttr);
```

【参数】

参数名称	描述	输入/输出
pstLayerAttr	PIP 视频层属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_ILLEGAL_PARAM	无效参数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_VIDEO_NOT_DISABLE	PIP 视频层未禁止。
HI_ERR_VO_NULL_PTR	参数指针为空。
HI_ERR_VO_DEV_NOT_BINDED	PIP 视频层未绑定
HI_ERR_VO_NOT_SUPPORT	芯片不支持

【差异说明】

芯片	描述
Hi3531	DSD5 的视频层可以作为高清设备的 PIP 层使用。
Hi3532	不支持该接口
Hi3521/Hi3520A/Hi3520D/ Hi3515A/Hi3515C	DSD0 的视频层可以作为高清设备的 PIP 层使用。
Hi3518/Hi3516C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 设置 PIP 层属性之前需要先把 PIP 层绑定到高清设备上。
- 建议在设置 PIP 视频层属性前先获取视频层属性。

【举例】

请参见 HI_MPI_VO_PipLayerBindDev 的举例。

【相关主题】

HI_MPI_VO_GetPipLayerAttr

HI_MPI_VO_GetPipLayerAttr

【描述】

获取视频层属性。

【语法】

```
HI_S32 HI_MPI_VO_GetPipLayerAttr(VO_VIDEO_LAYER_ATTR_S *pstLayerAttr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstLayerAttr	PIP 视频层属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_NULL_PTR	参数指针为空。
HI_ERR_VO_NOT_SUPPORT	芯片不支持

【差异说明】

芯片	描述
Hi3531	DSD5 的视频层可以作为高清设备的 PIP 层使用。
Hi3532	不支持该接口
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DSD0 的视频层可以作为高清设备的 PIP 层使用。
Hi3518/Hi3516C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

获取视频层属性不要求视频层属性已设置或者视频层使能。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

请参见 HI_MPI_VO_PipLayerBindDev 的举例。

【相关主题】

HI_MPI_VO_SetPipLayerAttr

HI_MPI_VO_EnablePipLayer

【描述】

使能 PIP 视频层。

【语法】

```
HI_S32 HI_MPI_VO_EnablePipLayer (HI_VOID);
```

【参数】

参数名称	描述	输入/输出
无	无	无

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_DEV_NOT_ENABLE	视频输出设备未使能。
HI_ERR_VO_DEV_NOT_CONFIG	设备未配置。
HI_ERR_VO_NO_MEM	没有可用的内存。
HI_ERR_VO_NOT_SUPPORT	芯片不支持
HI_ERR_VO_DEV_NOT_BINDED	PIP 视频层未绑定

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3531	DSD5 的视频层可以作为高清设备的 PIP 层使用。
Hi3532	不支持该接口
Hi3521/Hi3520A/ Hi3520D/Hi3515A /Hi3515C	DSD0 的视频层可以作为高清设备的 PIP 层使用。
Hi3518/Hi3516C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- PIP 视频层使能前必须保证该视频层所在的设备处于使能状态。
- PIP 视频层使能前必须保证该视频层已经配置。
- PIP 视频层使能前必须保证该视频层已经绑定。

【举例】

请参见 HI_MPI_VO_PipLayerBindDev 的举例。

【相关主题】

HI_MPI_VO_DisablePipLayer

HI_MPI_VO_DisablePipLayer

【描述】

禁止 PIP 视频层。

【语法】

```
HI_S32 HI_MPI_VO_DisablePipLayer(HI_VOID);
```

【参数】

参数名称	描述	输入/输出
无	无	无

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_CHN_NOT_DISABLE	该视频层上还有未禁止的通道。
HI_ERR_VB_BUSY	VB 资源没有释放。
HI_ERR_VO_NOT_SUPPORT	芯片不支持
HI_ERR_VO_DEV_NOT_BINDED	PIP 视频层未绑定

【差异说明】

芯片	描述
Hi3531	DSD5 的视频层可以作为高清设备的 PIP 层使用。
Hi3532	不支持该接口。
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	DSD0 的视频层可以作为高清设备的 PIP 层使用。
Hi3518/Hi3516C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- PIP 视频层禁止前必须保证其上的通道全部禁止。
- PIP 视频层禁止前必须保证该视频层已经绑定。
- 在禁止 PIP 视频层时，如果用户没有释放从 VO 获取的图像 buffer 资源，该接口会返回 HI_ERR_VB_BUSY 的错误码，表示 VO 创建的 VB 资源没有释放。多见于用户调用获取屏幕图像未释放的情况下。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

请参见 HI_MPI_VO_PipLayerBindDev 的举例。

【相关主题】

HI_MPI_VO_EnablePipLayer

HI_MPI_VO_EnableChn

【描述】

启用指定的视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_EnableChn(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_INVALID_CHNID	视频输出通道号无效。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_SYS_NOTREADY	系统未准备好，这里指设备 Fd 未打开。
HI_ERR_VO_VIDEO_NOT_ENABLE	视频层未使能。
HI_ERR_VO_CHN_NOT_ALLOC	通道资源未分配。
HI_ERR_VO_CHN_NOT_CONFIG	通道未配置。
HI_FAILURE	PIP 层显示通道区域重叠。
HI_ERR_VO_ILLEGAL_PARAM	PIP 层显示的通道总面积超过限制。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前必须先启用视频输出设备，否则将返回失败。
- 调用前必须使能相应设备上的视频层。
- 通道使能前必须进行通道配置，否则返回通道未配置的错误。
- 允许重复使能同一视频输出通道，不返回失败。

【举例】

```
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
VO_CHN_ATTR_S stChnAttr;

s32Ret = HI_MPI_VO_GetChnAttr(VoDev, VoChn, &stChnAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Get channel attr failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}
stChnAttr.u32Priority = 0;
stChnAttr.stRect.s32X = 0;
stChnAttr.stRect.s32Y = 0;
stChnAttr.stRect.u32Width = 720;
stChnAttr.stRect.u32Height = 576;

s32Ret = HI_MPI_VO_SetChnAttr(VoDev, VoChn, &stChnAttr);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
if (s32Ret != HI_SUCCESS)
{
    printf("Set channel attr failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_EnableChn(VoDev, VoChn);
if (s32Ret != HI_SUCCESS)
{
    printf("Enable channel failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_DisableChn(VoDev, VoChn);
if (s32Ret != HI_SUCCESS)
{
    printf("Disable channel failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_DisableChn

HI_MPI_VO_DisableChn

【描述】

禁用指定的视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_DisableChn(VO_DEV VoDev ,VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_VIDEO_NOT_ENABLE	视频层未使能。
HI_ERR_VO_CHN_NOT_ALLOC	通道资源未配置。
HI_ERR_VO_BUSY	VO 等待 VPSS 发送图像超时返回。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 允许重复禁用同一视频输出通道，不返回失败。
- 当高清设备的通道绑定 VPSS 时，建议先调用本接口停止通道后，再解绑定 VO 通道与 VPSS 通道的绑定关系，否则，可能出现 HI_ERR_VO_BUSY 的超时返回错误。

【举例】

请参见 HI_MPI_VO_EnableChn 的举例。

【相关主题】

HI_MPI_VO_EnableChn

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_SetChnAttr

【描述】

配置指定视频输出通道的属性。

【语法】

```
HI_S32 HI_MPI_VO_SetChnAttr(VO_DEV VoDev,VO_CHN VoChn, VO_CHN_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstAttr	视频通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	• VoDev 取值范围：[0, VO_MAX_DEV_NUM)。 • 高清设备优先级有 0 与 1，标清设备优先级为 0～VO_MAX_CHN_NUM-1。 • 标清设备通道宽高必须不小于 16
Hi3532	• VoDev 取值范围：[0]，[10, 13] • Hi3532 优先级必须设置为 0。 • 标清设备通道宽高必须不小于 16。
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	• VoDev 取值范围：[0, VO_MAX_DEV_NUM) • 高清设备优先级有 0 与 1，标清设备优先级为 0～VO_MAX_CHN_NUM-1。 • 标清设备通道宽高必须不小于 16。
Hi3518/Hi3516C	• VoDev 取值范围：[0, VO_MAX_DEV_NUM)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
	• 标清设备通道优先级为 0~ VO_MAX_CHN_NUM-1。 • 标清设备通道宽高必须不小于 32。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_ILLEGAL_PARAM	参数超出合法范围。
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 标清设备中，属性中的优先级，数值越大优先级越高。
  - 高清设备，优先级有 0 与 1，优先级为 0 代表通道在设备原视频层上，优先级为 1 代表通道在 PIP 层上，PIP 层总是叠加在原视频层之上。高清设备视频层的通道不允许出现重叠，PIP 层的通道也不允许出现重叠。
  - 标清设备，优先级有 0～VO_MAX_CHN_NUM-1，当多个通道有重叠的显示区域时，优先级高的通道图像将覆盖优先级低的通道。优先级相同的各通道有重叠时，默认通道号大的图像将覆盖通道号小的通道图像。
- 通道显示区域不能超过视频层属性中设定的画布大小(stImageSize 大小)。
- 该接口为动态设置接口，可在 VO 设备使能且相应视频层已配置的情况下调用。
- HD 设备的通道属性 stRect 需 2 对齐，且宽高必须不小于 32。在隔行时序且像素格式为 SEMIPLANAR_420 时，stRect 的高度必须 4 对齐。**起始点的纵坐标建议按照 4 对齐配置，否则可能会出现不可预料的图像质量问题。**HD 的宽高必须不小于 32。
- 如果 HD 有放大情况，stRect 是放大前视频层上的通道原始的起始位置和宽高，放大后的通道的起始位置和宽高会按比例偏移或放大。
- SD 设备的通道属性 stRect 需 2 对齐。

【举例】

请参见 HI_MPI_VO_EnableChn 的举例。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_GetChnAttr

HI_MPI_VO_GetChnAttr

【描述】

获取指定视频输出通道的属性。

【语法】

```
HI_S32 HI_MPI_VO_GetChnAttr(VO_DEV VoDev,VO_CHN VoChn,
                            VO_CHN_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstAttr	视频通道属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

请参见 HI_MPI_VO_EnableChn 的举例。

【相关主题】

HI_MPI_VO_SetChnAttr

HI_MPI_VO_SetChnDisplayWindow

【描述】

配置指定视频输出通道的显示窗口。

【语法】

```
HI_S32 HI_MPI_VO_SetChnDisplayWindow(VO_DEV VoDev, VO_CHN VoChn, const
RECT_S *pstWindow);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstWindow	视频通道显示窗口指针。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，参见错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 此接口主要应用于 VO 高清多画面显示时，在保持画面格局不变的情况下仅改变某个通道的显示窗口大小。
- 该接口用于设置通道的显示窗口位置，仅适用于高清视频层 VHD 和 PIP 视频层。
- PIP 层如果调用了接口 HI_MPI_VO_SetChnDispPos 设置显示位置，则实际显示位置以 HI_MPI_VO_SetChnDispPos 接口设置的位置为准。
- 该接口仅在 VPSS 以 AUTO 模式绑定 VO 下设置有效，VPSS 以 USER 模式绑定 VO 设置无效。
- 调用该接口前必须先调用接口 HI_MPI_VO_SetChnAttr 设置通道属性。重新设置通道属性后，显示窗口区域位置复位至与通道属性中的区域位置一致。
- 显示窗口属性中的 X、Y 是以当前通道属性的左上角坐标为原点的相对坐标。显示窗口设置的区域不得超出通道属性中设置的区域大小，对齐要求与通道属性的对齐要求一致。
- 该接口可以动态设置，并支持批处理。

【举例】

无。

【相关主题】

HI_MPI_VO_GetChnDisplayWindow

HI_MPI_VO_GetChnDisplayWindow

【描述】

获取指定视频输出通道的显示窗口。

【语法】

```
HI_S32 HI_MPI_VO_GetChnDisplayWindow(VO_DEV VoDev, VO_CHN VoChn, RECT_S
*pstWindow);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstWindow	视频通道显示窗口指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VO_SetChnDisplayWindow

HI_MPI_VO_SetChnDispPos

【描述】

设置指定视频输出通道的显示位置。

【语法】

```
HI_S32 HI_MPI_VO_SetChnDispPos(VO_DEV VoDev, VO_CHN VoChn, const POINT_S
*pstDispPos);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstDispPos	通道显示位置。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	VoDev 取值范围：[0, 1]。
Hi3532	不支持该接口。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	VoDev 取值范围：[0]。
Hi3518/Hi3516C	不支持该接口。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_ILLEGAL_PARAM	参数超出合法范围。
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_CHN_NOT_CONFIG	通道属性未设置。
HI_ERR_VO_NOT_SUPPORT	不支持当前操作。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 该接口用于设置通道的实际显示位置，仅适用于高清设备 PIP 层上的通道。
  - 默认情况下，VO 设备从视频层的左上角开始显示视频层画布上的通道，设置通道属性时设定了通道在画布上的位置，通道在画布上的位置即为默认的显示位置。
  - 希望节省 PIP 层内存时，可以将 PIP 层的画布大小（stImageSize）设置为实际显示图像需要的内存大小，显示区域大小（stDispRect）设置为 VO 设备输出分辨率大小。这样设置对 PIP 层来说，不会将画布放大到整个显示区域（高清设备的普通视频层这样设置时会将画布放大到整个显示区域。标清和虚拟设备的画布大小和显示区域大小必须一致）。
  - 当调用该接口设置通道的显示位置时，新的显示位置将取代默认值，成为 VO 设备上通道的实际显示位置。
- 通道显示位置不能超过视频层属性中设定的显示区域 stDispRect 的范围。
- 调用该接口前必须先设置通道属性。
- 通道显示位置的（x，y）必须 2 对齐。
- 该接口可以动态设置，并支持批处理。建议用户使用批处理方式设置通道的实际显示位置。
- 通道显示位置的（x，y）设置为（-1，-1）时表示恢复通道原有的默认显示位置，用户需注意恢复通道原有的默认显示位置时不要与其他通道的区域产出重叠。建议用户使用批处理方式恢复通道的默认显示位置。
- 该接口设置的显示位置（x,y）在通道 disable 之后仍然有效，用户再使能通道，该通道的显示位置仍然为（x,y）；若用户需要恢复默认的显示位置，需要手动的去设置显示位置为（-1，-1）。

【举例】

```
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
VO_CHN_ATTR_S stChnAttr;
VO_VIDEO_LAYER_ATTR_S stLayerAttr;
POINT_S stDispPos;
……
stLayerAttr->stDispRect.s32X       = 0;
stLayerAttr->stDispRect.s32Y       = 0;
stLayerAttr->stDispRect.u32Width   = 1920;
stLayerAttr->stDispRect.u32Height  = 1080;
stLayerAttr->stImageSize.u32Width  = 720;
stLayerAttr->stImageSize.u32Height = 576;
stLayerAttr->u32DispFrmRt          = 25;
stLayerAttr->enPixFormat           = PIXEL_FORMAT_YUV_SEMIPLANAR_422;

s32Ret = HI_MPI_VO_PipLayerBindDev(VoDev);
if (s32Ret != HI_SUCCESS)
{
    printf("Pip Layer Bind Dev failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}
s32Ret = HI_MPI_VO_SetPipLayerAttr(&stLayerAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set Pip Layer Attr failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}
s32Ret = HI_MPI_VO_EnablePipLayer();
if (s32Ret != HI_SUCCESS)
{
   printf("Enable Pip Layer failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}

stChnAttr.u32Priority = 1;
stChnAttr.stRect.s32X = 0;
stChnAttr.stRect.s32Y = 0;
stChnAttr.stRect.u32Width = 720;
stChnAttr.stRect.u32Height = 576;
stChnAttr.bDeflicker = HI_FALSE;

s32Ret = HI_MPI_VO_SetChnAttr(VoDev, VoChn, &stChnAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set channel attr failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}

s32Ret = HI_MPI_VO_EnableChn(VoDev, VoChn);
if (s32Ret != HI_SUCCESS)
{
   printf("Enable channel failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}
stDispPos.s32X = 500;
stDispPos.s32Y = 500;
s32Ret = HI_MPI_VO_SetChnDispPos (VoDev, VoChn, & stDispPos);
if (s32Ret != HI_SUCCESS)
{
   printf("Set channel display position failed with errno %#x!\n",
s32Ret);
   return HI_FAILURE;
}
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
s32Ret = HI_MPI_VO_DisableChn(VoDev, VoChn);
if (s32Ret != HI_SUCCESS)
{
    printf("Disable channel failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_GetChnDispPos

HI_MPI_VO_GetChnDispPos

【描述】

获取指定视频输出通道的显示位置。

【语法】

```
HI_S32 HI_MPI_VO_GetChnDispPos(VO_DEV VoDev, VO_CHN VoChn, POINT_S
*pstDispPos);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。 取值范围：[0, VO_MAX_DEV_NUM)	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstDispPos	视频通道显示位置属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VO_SetChnDispPos

HI_MPI_VO_SetChnField

【描述】

设置指定视频输出通道的帧场显示策略。

主要的应用场合有三个：

- 在预览 CIF 图像时，将视频输入设置为 2 场 CIF，对应 VO 通道设置成显示 2 场，比只显示一场的流畅性好；
- 在低帧率 2 场图像预览的情况下，如在低帧率的 D1 预览时，应该将对应 VO 通道设置成只显示一场，否则会出现画面上下抖动的现象。
- 视频输入为 D1/CIF 混合采集图像时，需要调用本接口对 D1 图像抽场显示，否则会出现图像上下抖动的现象。

【语法】

```
HI_S32 HI_MPI_VO_SetChnField(VO_DEV VoDev, VO_CHN VoChn, const
VO_DISPLAY_FIELD_E enField);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。 取值范围：[0, VO_MAX_DEV_NUM）	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
enField	视频通道显示帧场策略。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_ILLEGAL_PARAM	参数超出合法范围。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_NOT_SUPPORT	不支持当前操作。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 如果不显式调用此接口，VO 默认为两场显示（即 VO_FIELD_BOTH）。
- 如果送 VO 显示的图像为帧图像（例如单场 CIF 图像），则调用此接口对图像显示无影响。
- 该接口仅对标清、虚拟设备有效，高清设备调用该接口无效。

【举例】

```
/* 这是一个预览低帧率的例子，先将视频输入设置成8帧每秒，然后对应VO通道用底场显示 */
VO_DISPLAY_FIELD_E enField;
VI_CHN_ATTR_S stViChnAttr;
…
stViChnAttr.s32SrcFrameRate = 25;
stViChnAttr.s32FrameRate = 8;
…
if (HI_SUCCESS!= HI_MPI_VI_SetChnAttr(0, & stViChnAttr))
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    printf("HI_MPI_VI_SetChnAttr failed !\n");
    return HI_FAILURE;
}
if (HI_SUCCESS!=HI_MPI_VO_SetChnField(2, 0, VO_FIELD_BOTTOM))
{
    printf("HI_MPI_VO_SetChnField failed !\n");
    return HI_FAILURE;
}
if (HI_SUCCESS!=HI_MPI_VO_GetChnField(2, 0, &enField))
{
    printf("HI_MPI_VO_GetChnField failed !\n");
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_GetChnField

HI_MPI_VO_GetChnField

【描述】

获取指定视频输出通道的帧场显示策略。

【语法】

```
HI_S32 HI_MPI_VO_GetChnField(VO_DEV VoDev, VO_CHN VoChn,
VO_DISPLAY_FIELD_E *pField);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。 取值范围：[0, VO_MAX_DEV_NUM)	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pField	视频通道显示帧场属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

如未设置显示策略则返回系统默认策略 VO_FIELD_BOTH。

【举例】

请参见 HI_MPI_VO_SetChnField 的举例。

【相关主题】

HI_MPI_VO_SetChnField

HI_MPI_VO_SendFrame

【描述】

将视频图像送入指定输出通道显示。

【语法】

```
HI_S32 HI_MPI_VO_SendFrame(VO_DEV VoDev, VO_CHN VoChn,
                           VIDEO_FRAME_INFO_S *pstVFrame);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstVFrame	视频数据信息指针。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

调用该接口前必须保证通道已经使能。

【举例】

无。

【相关主题】

无。

HI_MPI_VO_SendFrameTimeOut

【描述】

将视频图像送入指定输出通道显示。

【语法】

```
HI_S32 HI_MPI_VO_SendFrame(VO_DEV VoDev, VO_CHN VoChn,
                           VIDEO_FRAME_INFO_S *pstVFrame, HI_U32 u32MilliSec);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstVFrame	视频数据信息指针。	输入
u32MilliSec	超时时间。单位：ms	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用该接口前必须保证通道已经使能。
- 该接口支持阻塞式发送，u32MilliSec 用于设定超时时间。超时时间为 0 表示一直阻塞，直到发送成功为止。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_SetChnFrameRate

【描述】

设置指定视频输出通道的显示帧率。

【语法】

```
HI_S32 HI_MPI_VO_SetChnFrameRate (VO_DEV VoDev,VO_CHN VoChn, HI_S32
s32VoFramerate);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
s32VoFramerate	视频通道显示帧率。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_ILLEGAL_PARAM	参数非法。
HI_ERR_VO_CHN_NOT_CONFIG	通道未配置。
HI_ERR_VO_BUSY	系统忙，一般表明此前未调用系统初始化函数。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 需先设置 VO 相应视频输出通道属性，才能调用此接口，否则返回失败。
- 帧率可设置为 Nx（其中 N 为[-64，+64]的任意整数，x 为 P 制或 N 制下的帧率）。负数的倍数用于倒放操作，此时需要用户来倒序送图像到 VO，即送时间戳递减的图像。
- 调用 HI_MPI_VO_EnableChn 接口会使通道重置为正常速度播放，因此，建议在通道使能后再调用本接口进行播放控制。
- 在通道禁用的情况下，调用 HI_MPI_VO_SetChnAttr 接口会将通道帧率重置为显示帧率。

【举例】

```
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
HI_S32 s32ChnFrmRate = 30;
s32Ret = HI_MPI_VO_SetChnFrameRate(VoDev, VoChn, s32ChnFrmRate);
if (s32Ret != HI_SUCCESS)
{
   printf("Set channel %d frame rate failed with errno %#x!\n", VoChn,
s32Ret);
   return HI_FAILURE;
}
```

【相关主题】

无。

HI_MPI_VO_GetChnFrameRate

【描述】

获取指定视频输出通道的显示帧率。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VO_GetChnFrameRate(VO_DEV VoDev, VO_CHN VoChn, HI_S32
*ps32VoFramerate);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
ps32VoFramerate	视频通道显示帧率。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_BUSY	系统忙，一般表明此前未调用系统初始化函数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

获取的为用户配置的帧率，如未设置则返回满帧率。

【举例】

```
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
HI_S32 s32ChnFrmRate;

s32Ret = HI_MPI_VO_GetChnFrameRate(VoDev, VoChn, &s32ChnFrmRate);
if (s32Ret != HI_SUCCESS)
{
   printf("Set channel %d frame rate failed with errno %#x!\n", VoChn,
s32Ret);
   return HI_FAILURE;
}
printf("Channel %d frame rate is %d.\n", VoChn, s32ChnFrmRate);
```

【相关主题】

无。

HI_MPI_VO_ChnPause

【描述】

暂停指定的视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_ChnPause(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	视频输出通道号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- VI-VO 预览时不推荐使用此接口。
- 允许重复暂停同一通道，不返回失败。

【举例】

```
HI_S32 s32ret;
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
...
/* enable vo chn */
s32ret = HI_MPI_VO_EnableChn(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    printf("enable vo chn failed! \n");
    return s32ret;
}

/* pause current vo channel */
s32ret = HI_MPI_VO_ChnPause(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("pause vo chn failed! \n");
    return s32ret;
}

/* resume current vo channel */
s32ret = HI_MPI_VO_ChnResume(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("resume vo chn failed! \n");
    return s32ret;
}

while (getchar() != 'q')
{
    /* step forward current vo channel */
    s32ret = HI_MPI_VO_ChnStep(VoDev, VoChn);
    if (HI_SUCCESS != s32ret)
    {
        printf("step play vo chn failed! \n");
        return s32ret;
    }
}

/* resume current vo channel */
s32ret = HI_MPI_VO_ChnResume(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("resume vo chn failed! \n");
    return s32ret;
}

(void)HI_MPI_VO_DisableChn(VoDev, VoChn);
```

【相关主题】

HI_MPI_VO_ChnResume

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_ChnResume

【描述】

恢复指定的视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_ChnResume(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	视频输出通道号无效。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

允许重复恢复同一通道，不返回失败。

【举例】

请参见 HI_MPI_VO_ChnPause 的举例。

【相关主题】

HI_MPI_VO_ChnPause

HI_MPI_VO_ChnStep

【描述】

单帧播放指定的视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_ChnStep(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	视频输出通道号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 恢复正常播放使用 HI_MPI_VO_ChnResume 接口。
- VI-VO 预览时不推荐使用此接口。

【举例】

请参见 HI_MPI_VO_ChnPause 的举例。

【相关主题】

HI_MPI_VO_ChnResume

HI_MPI_VO_ChnRefresh

【描述】

刷新指定的视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_ChnRefresh(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	视频输出通道号无效。
HI_ERR_VO_NOT_SUPPORT	不支持。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 主要应用场景是解码回放中，HD 设备暂停时的电子放大场景。HD 设备的电子放大场景是通过对 VPSS Group 调用 HI_MPI_VPSS_SetCropCfg 接口来实现，当 HD 设备暂停时，VPSS 裁剪后的图像不能送给 VO，此时调用 HI_MPI_VO_ChnRefresh 接口刷新视频输出通道即可。
- 该接口还适用于解码低帧率下的电子放大场景。
- 该接口调用后立即生效，因此应该在调用 HI_MPI_VPSS_SetCropCfg 接口之后调用，否则视频输出通道刷新的是 VPSS 裁剪之前的图像。
- 该接口在正常播放时看不到任何效果，包括快进慢放。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 该接口不适用于标清设备，也不适用于 VI-VO 预览场景。

【举例】

```
HI_S32 s32ret;
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
...
/* enable vo chn */
s32ret = HI_MPI_VO_EnableChn(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("enable vo chn failed! \n");
    return s32ret;
}
/* pause current vo channel */
s32ret = HI_MPI_VO_ChnPause(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("pause vo chn failed! \n");
    return s32ret;
}
/* set vpss clip config */
...
/* refresh current vo channel */
s32ret = HI_MPI_VO_ChnRefresh(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("refresh vo chn failed! \n");
    return s32ret;
}
(void)HI_MPI_VO_DisableChn(VoDev, VoChn);
```

【相关主题】

无。

HI_MPI_VO_EnableRecvFrameRateMatch

【描述】

启用指定通道的帧率匹配功能。

【语法】

```
HI_S32 HI_MPI_VO_EnableRecvFrameRateMatch (VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	视频输出通道号无效。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 此接口仅支持高清通道。
- 典型应用场景：VIVO 级联时，主片开启倍帧，为了保证图像流畅性，显示帧率需设成 30。从片同样开启倍帧，那么从片传到主片的帧率为 60，这样就会导致丢帧。调用此接口可以解决丢帧问题。

【举例】

无。

【相关主题】

HI_MPI_VO_DisableRecvFrameRateMatch

HI_MPI_VO_DisableRecvFrameRateMatch

【描述】

禁用指定通道的帧率匹配功能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VO_DisableRecvFrameRateMatch (VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

此接口仅支持高清通道。

【举例】

无。

【相关主题】

HI_MPI_VO_EnableRecvFrameRateMatch

HI_MPI_VO_SetZoomInWindow

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

设置视频输出局部放大窗口。

【语法】

```
HI_S32 HI_MPI_VO_SetZoomInWindow(VO_DEV VoDev, VO_CHN VoChn, const
VO_ZOOM_ATTR_S *pstZoomAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstZoomAttr	局部放大属性结构体。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	通道号无效。
HI_ERR_VO_INVALID_RECT_PARA	矩形结构体参数无效。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_VIDEO_NOT_CONFIG	视频层未配置。
HI_ERR_VO_NOT_SUPPORT	芯片不支持高清通道优先级设为 1。
HI_ERR_VO_ILLEGAL_PARAM	输入参数超出有效范围。
HI_ERR_VO_NULL_PTR	输入参数空指针。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前必须先启用视频输出设备，并且配置视频输出公共属性。
- 调用前必须保证视频层已经配置。
- 该接口对 HD 和 SD 具有不同的意义。

  HD 只有针对整个视频层的放大能力，对单个通道的缩放，譬如 VI 源图像是 D1，对应的 VO 通道大小是 CIF，这个缩放是通过 VPSS 实现的。因此对于 HD，该接口的作用在于截取一部分 VO 通道的图像数据，不经放大，直接在区域上显示，不足的部分显示背景色。

  SD 通过 TDE 实现对单个通道的缩放，故而对于 SD，该接口的作用在于截取一部分 VO 通道的图像数据，并利用 TDE 缩放到通道大小，实现局部放大的功能。
- 支持两种截取 VO 通道源数据的方式：按位置和按比例。详见 VO_ZOOM_ATTR_S 章节。
- 对于标清设备，如果输入的视频图像是 D1/2CIF（两场）混合采集序列，则 pstZoomAttr 属性以小图像为参考来配置，SDK 会自动对大图像进行适配处理，以保证视域相同。特别地，当 pstZoomAttr 属性按比例方式配置时，用户需保证大、小图像按比例换算成位置后的视域要一致，否则图像会左右抖动。
- 在视频级联的主片上，利用局部放大功能在级联的图像上选取不同区域进行多画面拼接。

【举例】

```
/* define input parameter */
HI_S32 s32Ret;
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
VO_ZOOM_ATTR_S stZoomWindow, stZoomWindowGet;
/*
 * we should enable VO and VO channel first!
 * TODO: enable operation.
 */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
stZoomAttr.enZoomType = VOU_ZOOM_IN_RECT;
stZoomWindow.stZoomRect.s32X = 128;
stZoomWindow.stZoomRect.s32Y = 128;
stZoomWindow.stZoomRect.u32Width = 200;
stZoomWindow.stZoomRect.u32Height = 200;

/* set zoom window */
s32Ret = HI_MPI_VO_SetZoomInWindow(VoDev, VoChn, &stZoomWindow);
if (s32Ret != HI_SUCCESS)
{
    printf("Set zoom attribute failed, ret = %#x.\n", s32Ret);
    return HI_FAILURE;
}
/* get current zoom window parameter */
s32Ret = HI_MPI_VO_GetZoomInWindow(VoDev, VoChn, &stZoomWindowGet);
if (s32Ret != HI_SUCCESS)
{
    printf("Get zoom attribute failed, ret = %#x.\n", s32Ret);
    return HI_FAILURE;
}
else
{
    printf("Current zoom window is (%d,%d,%d,%d)!\n",
           stZoomWindowGet.stZoomRect.s32X,
           stZoomWindowGet.stZoomRect.s32Y,
           stZoomWindowGet.stZoomRect.u32Width,
           stZoomWindowGet.stZoomRect.u32Height);
}
/*
 * TODO: something else ...
 */
stZoomWindow.stZoomRect.s32X = 0;
stZoomWindow.stZoomRect.s32Y = 0;
stZoomWindow.stZoomRect.u32Width = 0;
stZoomWindow.stZoomRect.u32Height = 0;
/* cancel zoom window, we use (0,0,0,0) to recover */
s32Ret = HI_MPI_VO_SetZoomInWindow(VoDev, VoChn, &stZoomWindow);
if (s32Ret != HI_SUCCESS)
{
    printf("Recover zoom attribute failed, ret = %#x.\n", s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_GetZoomInWindow

HI_MPI_VO_GetZoomInWindow

【描述】

获取视频输出局部放大窗口。

【语法】

```
HI_S32 HI_MPI_VO_GetZoomInWindow(VO_DEV VoDev, VO_CHN VoChn,
VO_ZOOM_ATTR_S *pstZoomAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstZoomAttr	局部放大属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	通道号无效。
HI_ERR_VO_NULL_PTR	参数中有空指针。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

如果没有设置局部放大窗口，默认获取到（0, 0, 0, 0）。

【举例】

请参见 HI_MPI_VO_SetZoomInWindow 的举例。

【相关主题】

HI_MPI_VO_SetZoomInWindow

HI_MPI_VO_GetChnPts

【描述】

获取指定视频输出通道正在显示图像的时间戳。

【语法】

```
HI_S32 HI_MPI_VO_GetChnPts(VO_DEV VoDev, VO_CHN VoChn,
                           HI_U64 *pu64VoChnPts);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pu64VoChnPts	通道时间戳指针。	输出

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_NULL_PTR	参数中有空指针。
HI_ERR_VO_CHN_NOT_ENABLE	通道未使能。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证视频设备使能，视频通道使能。
- 解码通过系统绑定的方式向 VO 发送视频帧时，如果解码帧的 pts 为 0，VO 会从 0 开始，按照每帧的播放时间间隔打上时间戳，以确保每帧播放一次。其他情况下不会改动源视频的时间戳。

【举例】

```
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
HI_U64 u64ChnPts;
s32Ret = HI_MPI_VO_GetChnPts(VoDev, VoChn, &u64ChnPts);
if (s32Ret != HI_SUCCESS)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    printf("Get channel %d pts failed with errno %#x!\n", VoChn, s32Ret);
    return HI_FAILURE;
}
printf("Channel %d pts is %llu.\n", VoChn, u64ChnPts);
```

【相关主题】

无。

HI_MPI_VO_SetAttrBegin

【描述】

设置属性开始。

【语法】

```
HI_S32 HI_MPI_VO_SetAttrBegin(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]、[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围。
HI_ERR_VO_SETBEGIN_ALREADY	视频输出设置属性已经开始。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- BEGIN 和 END 接口要配对使用，否则 BEGIN 后设置的通道属性不会生效。
- 此接口可对通道的动态操作进行批处理，例如设置通道属性、通道显示、通道隐藏、使能通道、禁止通道、级联区域绑定通道、级联区域解绑定通道等。当 VO 切换画面大小或显示隐藏时，需要重新准备内存并保证切换的及时性，因此建议的调用顺序为：在 BEGIN 和 END 接口之间调用设置通道属性、通道显示、通道隐藏等接口，对设备的多个通道进行批处理操作。
- 批处理中的通道操作都是在 HI_MPI_VO_SetAttrEnd 接口调用之后生效，建议合理控制 BEGIN 和 END 接口的范围。例如希望批处理禁止通道，并禁止视频层，要按照如下顺序调用 HI_MPI_VO_SetAttrBegin、HI_MPI_VO_DisableChn、HI_MPI_VO_SetAttrEnd 和 HI_MPI_VO_DisableVideoLayer。由于禁止视频层的调用必须在禁止通道成功后生效，禁止通道必须在 HI_MPI_VO_SetAttrEnd 接口调用后生效，如果是按照顺序调用 HI_MPI_VO_SetAttrBegin、HI_MPI_VO_DisableChn、HI_MPI_VO_DisableVideoLayer 和 HI_MPI_VO_SetAttrEnd，将导致禁止视频层失败。

【举例】

```
HI_S32 s32ret;
HI_U32 i;
VO_DEV VoDev = 0;
VO_PUB_ATTR_S VoAttr;
VO_CHN_ATTR_S VoChnAttr[4];

memset(&VoAttr, 0, sizeof(VO_PUB_ATTR_S));
VoAttr.enIntfType = VO_INTF_CVBS;  /*CVBS */
VoAttr.enIntfSync = VO_OUTPUT_PAL ; /*PAL Mode*/
VoAttr.u32BgColor = 0x0;

/* set public attribute of vo device */
s32ret = HI_MPI_VO_SetPubAttr(VoDev, &VoAttr);
if (HI_SUCCESS != s32ret)
{
    printf("vo set pub attr failed! \n");
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    return s32ret;
}

/* enable vo device */
s32ret = HI_MPI_VO_Enable(VoDev);
if (HI_SUCCESS != s32ret)
{
    printf("enable vo device failed! \n");
    return s32ret;
}
/* configure and enable vo channel          */
for (i = 0; i < 4; i++)
{
    VoChnAttr[i].bDeflicker = HI_FALSE;
    VoChnAttr[i].u32Priority = 1;
    VoChnAttr[i].stRect.s32X = 360*(i%2);
    VoChnAttr[i].stRect.s32Y = 288*(i/2);
    VoChnAttr[i].stRect.u32Width = 360;
    VoChnAttr[i].stRect.u32Height = 288;
    /* set attribute of vo chn*/
    s32ret = HI_MPI_VO_SetChnAttr(VoDev, i, &VoChnAttr[i]);
    if (HI_SUCCESS != s32ret)
    {
    printf("vo set dev %d chn %d attr failed! \n", VoDev , i);
    return s32ret;
    }
    /* enable vo chn */
    s32ret = HI_MPI_VO_EnableChn(VoDev, i);
    if (HI_SUCCESS != s32ret)
    {
    printf("enable vo dev %d chn %d failed! \n", VoDev,i);
    return s32ret;
    }
}

/* do somethng else                          */
sleep(20);

/* set channel attributes begin              */
s32ret = HI_MPI_VO_SetAttrBegin(VoDev);
if (HI_SUCCESS != s32ret)
{
    printf("set attributes begin failed!\n");
    return s32ret;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}
/* reset channel attributes
 * we just show channel 0 and 1 on screen, and scale them as vertical
 * half D1
 */
for (i = 0; i < 2; i++)
{
    VoChnAttr[i].bDeflicker = HI_FALSE;
    VoChnAttr[i].u32Priority = 1;
    VoChnAttr[i].stRect.s32X = 352*(i%2);
    VoChnAttr[i].stRect.s32Y = 288*(i/2);
    VoChnAttr[i].stRect.u32Width = 360;
    VoChnAttr[i].stRect.u32Height = 576;

    /* set attribute of vo chn*/
    s32ret = HI_MPI_VO_SetChnAttr(VoDev, i, &VoChnAttr[i]);
    if (HI_SUCCESS != s32ret)
    {
    printf("vo set chn %d attr failed! \n", i);
    return s32ret;
    }
}
/* hide channel 2 and 3
 * we do this just for saving bandwidth
 */
for (i = 2; i < 4; i++)
{
    s32ret = HI_MPI_VO_ChnHide(VoDev,i);
    if (s32ret != HI_SUCCESS)
    {
        printf("vo hide channel %d failed!\n", i);
        return HI_FAILURE;
    }
}
/* set channel attributes end                                          */
s32ret = HI_MPI_VO_SetAttrEnd(VoDev);
if (HI_SUCCESS != s32ret)
{
    printf("set attributes end failed!\n");
    return s32ret;
}

/* do somethng else                                                    */
sleep(20);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
/* set channel attributes begin                                    */
s32ret = HI_MPI_VO_SetAttrBegin(VoDev);
if (HI_SUCCESS != s32ret)
{
    printf("set attributes begin failed!\n");
    return s32ret;
}
/* reset channel attributes
 * now we set them back as 4 cif on screen
 */
for (i = 0; i < 4; i++)
{
    VoChnAttr[i].bDeflicker = HI_FALSE;
    VoChnAttr[i].u32Priority = 1;
    VoChnAttr[i].stRect.s32X = 360*(i%2);
    VoChnAttr[i].stRect.s32Y = 288*(i/2);
    VoChnAttr[i].stRect.u32Width = 360;
    VoChnAttr[i].stRect.u32Height = 288;

    /* set attribute of vo chn*/
    s32ret = HI_MPI_VO_SetChnAttr(VoDev,i, &VoChnAttr[i]);
    if (HI_SUCCESS != s32ret)
    {
    printf("vo set chn %d attr failed! \n", i);
    return s32ret;
    }
}
/* show channel 2 and 3                                            */
for (i = 2; i < 4; i++)
{
    s32ret = HI_MPI_VO_ChnShow(VoDev,i);
    if (s32ret != HI_SUCCESS)
    {
        printf("vo show channel %d failed!\n", i);
        return HI_FAILURE;
    }
}
/* set channel attributes end                                      */
s32ret = HI_MPI_VO_SetAttrEnd(VoDev);
if (HI_SUCCESS != s32ret)
{
    printf("set attributes end failed!\n");
    return s32ret;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}
/* do somethng else                                                    */
sleep(20);

/* disable all channels                                                */
for (i = 0; i < 4; i++)
{
    s32ret = HI_MPI_VO_DisableChn(VoDev , i);
    if (HI_SUCCESS != s32ret)
    {
        printf("vo disable chn %d failed!\n", i);
        return s32ret;
    }
}
/* disable vou                                          */
(void)HI_MPI_VO_Disable(VoDev);
```

【相关主题】

HI_MPI_VO_SetAttrEnd

HI_MPI_VO_SetAttrEnd

【描述】

设置属性结束。

【语法】

```
HI_S32 HI_MPI_VO_SetAttrEnd(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	视频输出设备 ID 超出合法范围。
HI_ERR_VO_SETBEGIN_NOTYET	视频输出设置属性没有开始。
HI_ERR_VO_SETEND_ALREADY	视频输出设置属性未完成。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证 HI_MPI_VO_SetAttrBegin 已经调用。
- BEGIN 和 END 接口要配对使用，否则 BEGIN 后设置的通道属性不会生效。

【举例】

请参见 HI_MPI_VO_SetAttrBegin 的举例。

【相关主题】

HI_MPI_VO_SetAttrBegin

HI_MPI_VO_ChnShow

【描述】

显示指定通道。

【语法】

```
HI_S32 HI_MPI_VO_ChnShow(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_VIDEO_NOT_ENABLE	视频输出视频层未使能。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证视频输出通道所在的视频层已经使能。
- 默认情况下通道处于显示状态。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

请参考 HI_MPI_VO_SetAttrBegin 的举例。

【相关主题】

HI_MPI_VO_ChnHide

HI_MPI_VO_ChnHide

【描述】

隐藏指定通道。

【语法】

```
HI_S32 HI_MPI_VO_ChnHide(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

当高清设备的通道绑定 VPSS 时，建议先调用本接口隐藏通道后，再解绑定 VO 通道与 VPSS 通道的绑定关系，否则，可能出现 HI_ERR_VO_BUSY 的超时返回错误。

【举例】

请参考 HI_MPI_VO_SetAttrBegin 的举例。

【相关主题】

HI_MPI_VO_ChnShow

HI_MPI_VO_QueryChnStat

【描述】

查询视频输出通道状态。

【语法】

```
HI_S32 HI_MPI_VO_QueryChnStat (VO_DEV VoDev, VO_CHN VoChn,
                               VO_QUERY_STATUS_S *pstStatus);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstStatus	通道状态结构体指针。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_CHN_NOT_ENABLE	视频输出通道未使能。
HI_ERR_VO_INVALID_DEVID	视频输出设备号无效。
HI_ERR_VO_CHN_NOT_ALLOC	通道资源未配置。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证视频设备使能。
- 调用前需要保证所要查询的通道已经使能。
- 可以多次调用获取通道状态接口。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
VO_QUERY_STATUS_S stVoQueryStatus;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
/* enable vo device and vo channel */
…

s32Ret = HI_MPI_VO_QueryChnStat(VoDev, VoChn, &stVoQueryStatus);
if (HI_SUCCESS != s32Ret)
{
    printf("Query channel status failed with errorcode %#x!\n", s32Ret);
    return HI_FAILURE;
}
printf("Current vo dev %d channel %d has %d VB occupied!\n",
            VoDev, VoChn, stVoQueryStatus .u32ChnBufUsed );
```

【相关主题】

无。

HI_MPI_VO_GetChnFrame

【描述】

获取输出通道图像数据。

【语法】

```
HI_S32 HI_MPI_VO_GetChnFrame(VO_DEV VoDev, VO_CHN VoChn,
                             VIDEO_FRAME_INFO_S *pstFrame);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstFrame	获取的输出通道图像数据信息结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_CHN_NOT_ENABLE	通道未使能。
HI_ERR_VO_NULL_PTR	视频输出参数空指针。
HI_ERR_VO_SYS_NOTREADY	通道无图像或通道隐藏。
HI_ERR_VO_NOT_SUPPORT	不支持重复获取。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证视频设备、通道使能。
- 调用前需要保证视频层使能。
- 不允许重复获取。
- 获取后应保证及时的释放。

【举例】

```
...
VIDEO_FRAME_INFO_S *pstFrame =
(VIDEO_FRAME_INFO_S*)malloc(sizeof(VIDEO_FRAME_INFO_S));

if (HI_SUCCESS != HI_MPI_VO_GetScreenFrame(VoDev, pstFrame))
{
    printf("Get screen frame failed!\n");
    return HI_FAILURE;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}

/* do some process, i.e. store file or do JPEG encodeing ... */

if (HI_SUCCESS != HI_MPI_VO_ReleaseScreenFrame(VoDev, pstFrame))
{
   printf("Release screen frame failed!\n");
   return HI_FAILURE;
}

/* enable vo chn */
s32ret = HI_MPI_VO_EnableChn(VoDev, VoChn);
if (HI_SUCCESS != s32ret)
{
    printf("enable vo chn failed! \n");
    return s32ret;
}

if (HI_SUCCESS != HI_MPI_VO_GetChnFrame(VoDev, VoChn, pstFrame))
{
   printf("Get channel frame failed!\n");
   return HI_FAILURE;
}

/* do some process, i.e. store file or do JPEG encodeing ... */

if (HI_SUCCESS != HI_MPI_VO_ReleaseChnFrame(VoDev, VoChn, pstFrame))
{
   printf("Release screen frame failed!\n");
   return HI_FAILURE;
}
(void)HI_MPI_VO_DisableChn(VoDev,VoChn);
```

【相关主题】

HI_MPI_VO_ReleaseChnFrame

HI_MPI_VO_ReleaseChnFrame

【描述】

释放输出通道图像数据。

【语法】

```
HI_S32 HI_MPI_VO_ReleaseChnFrame(VO_DEV VoDev, VO_CHN
                                 VoChn, VIDEO_FRAME_INFO_S *pstFrame);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
pstFrame	释放的输出通道图像数据信息结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围。
HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VO_NOT_PERMIT	操作不允许。
HI_ERR_VO_NULL_PTR	输入参数空指针。
HI_ERR_VO_ILLEGAL_PARAM	视频输出参数无效。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 获取操作应保证与释放操作配对。
- 不允许重复释放。

【举例】

请参见 HI_MPI_VO_GetChnFrame 的举例。

【相关主题】

HI_MPI_VO_GetChnFrame

HI_MPI_VO_ClearChnBuffer

【描述】

清空指定输出通道的缓存 buffer 数据。

【语法】

```
HI_S32 HI_MPI_VO_ClearChnBuffer(VO_DEV VoDev, VO_CHN VoChn,
                                HI_BOOL bClrAll)
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
bClrAll	是否将通道 buffer 中的数据全部清空。取值范围： • HI_TRUE：将通道 buffer 中的数据全部清空，此时屏幕上该通道的区域将无图像显示，直到有新的图像数据到来。 • HI_FALSE：在通道 buffer 中保留一幅图像数据，而将其他数据清空。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

用户可以调用此接口实现清除 VO 通道 buffer 的功能。

【举例】

```
VO_DEV VoDev = 0;
VO_CHN VoChn = 0;
HI_BOOL bClearAll = HI_TRUE;

s32Ret = HI_MPI_VO_ClearChnBuffer (VoDev, VoChn, & bClearAll);
if (s32Ret != HI_SUCCESS)
{
    printf("Clear channel %d buf failed with errno %#x!\n", VoChn, s32Ret);
    return HI_FAILURE;
}
```

【相关主题】

无。

HI_MPI_VO_SetChnDispThreshold

【描述】

设置视频输出通道的显示门限值。主要应用场景：PCIV 级联，从片传输图像到主片进行预览显示时，由于传输到主片的图像的时间间隔不均匀，可能造成通道待显示图像超过默认的门限值，引起丢帧卡顿现象，需要调大门限值。

【语法】

```
HI_S32 HI_MPI_VO_SetChnDispThreshold(VO_DEV VoDev, VO_CHN VoChn, HI_U32
u32Threshold);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
u32Threshold	视频输出通道的显示门限值	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 高清设备的显示门限取值范围为[3, 8]，默认门限为 3，表示该通道图像总的待显示帧数为 3；标清设备的显示门限取值范围为[2, 8]，默认门限为 2，表示该通道缓冲队列最多可接收 2 帧图像，再加上正在（或即将）显示的一帧图像，即总的待显示帧数也为 3。
- 门限值越小，预览延迟越小，反之越大，因此，正常情况下，不建议用户调大门限。
- 可动态调整门限值，但建议在通道使能前设置好。如需恢复默认值，需由用户调用本接口将门限设回默认值。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VO_GetChnDispThreshold

HI_MPI_VO_GetChnDispThreshold

【描述】

获取视频输出通道的显示门限值。

【语法】

```
HI_S32 HI_MPI_VO_GetChnDispThreshold(VO_DEV VoDev, VO_CHN VoChn, HI_U32
*pu32Threshold);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入
u32Threshold	视频输出通道的显示门限值。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)。
Hi3532	[0]，[10, 13]。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)。
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

调用 HI_MPI_VO_SetChnDispThreshold 设置前，调用本接口将获取到默认的门限值，反之，将获取到 HI_MPI_VO_SetChnDispThreshold 设置的门限值。

【举例】

无。

【相关主题】

HI_MPI_VO_SetChnDispThreshold

HI_MPI_VO_EnableChnDoubleFrame

【描述】

使能通道倍帧功能。

【语法】

```
HI_S32 HI_MPI_VO_EnableChnDoubleFrame(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 该接口为高级接口、非必设接口，用于动态使能/禁用 VO 通道的倍帧功能。如果 VO_PUB_ATTR_S 中的 bDoubleFrame 没打开，即使调用本接口使能倍帧功能，也不会产生倍帧效果。
- 只有高清和级联设备支持倍帧功能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 调用 HI_MPI_VO_EnableChn 接口后，程序会根据 VO_PUB_ATTR_S 中的 bDoubleFrame 的值以及当前的设备号重置通道的倍帧功能是否打开，例如 bDoubleFrame 设为 TRUE，且为高清（级联）设备，则倍帧功能默认打开，否则关闭。

【举例】

无。

【相关主题】

HI_MPI_VO_DisableChnDoubleFrame

HI_MPI_VO_DisableChnDoubleFrame

【描述】

禁用通道倍帧功能。

【语法】

```
HI_S32 HI_MPI_VO_DisableChnDoubleFrame(VO_DEV VoDev, VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
VoChn	视频输出通道的通道号。 取值范围：[0, VO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

请参考 HI_MPI_VO_EnableChnDoubleFrame。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VO_EnableChnDoubleFrame

HI_MPI_VO_GetScreenFrame

【描述】

获取输出屏幕图像数据。

【语法】

```
HI_S32 HI_MPI_VO_GetScreenFrame(VO_DEV VoDev,
                                VIDEO_FRAME_INFO_S *pstFrame);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstFrame	获取的输出屏幕图像数据信息结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[2, 7]和[10, VO_MAX_DEV_NUM)
Hi3532	[10, 13]
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[1, VO_MAX_DEV_NUM)
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证视频层已经使能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 可以多次调用，获取后应保证及时的释放。
- 如果用户获取了屏幕图像没有进行释放，则在禁止视频层的时候会返回 busy 的错误，此时用户应该释放获取的 buffer，再次调用禁止视频层操作，这样才能保证禁止成功。
- 该接口获取的图像含有时间戳，该时间戳表示图像拼接时的时间信息，如果用户不希望采用该时间戳进行编码，可以自行修改时间戳信息。

【举例】

请参考 HI_MPI_VO_GetChnFrame 的举例。

【相关主题】

HI_MPI_VO_ReleaseScreenFrame

HI_MPI_VO_ReleaseScreenFrame

【描述】

释放输出屏幕图像数据。

【语法】

```
HI_S32 HI_MPI_VO_ReleaseScreenFrame(VO_DEV VoDev,
                                    VIDEO_FRAME_INFO_S *pstFrame);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstFrame	释放的输出屏幕图像数据信息结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[2, 7]和[10, VO_MAX_DEV_NUM)
Hi3532	[10, 13]
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[1, VO_MAX_DEV_NUM)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	VoDev 取值范围
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 禁止视频层前要释放已经获取到的屏幕图像。
- 可以多次调用，获取操作应保证与释放操作配对。

【举例】

请参考 HI_MPI_VO_GetChnFrame 的举例。

【相关主题】

HI_MPI_VO_GetScreenFrame

HI_MPI_VO_SetDevCSC

【描述】

设置设备输出图像效果。

【语法】

```
HI_S32 HI_MPI_VO_SetDevCSC(VO_DEV VoDev, VO_CSC_S *pstDevCSC);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstDevCSC	图像输出效果结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3531	• VoDev 取值范围：[0, 7]。 • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]，默认值：50。 • enCscMatrix 默认值：单位矩阵。
Hi3532	• VoDev 取值范围：0。 • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]，默认值：50。 • enCscMatrix 默认值：单位矩阵。
Hi3521/Hi3520A	• VoDev 取值范围：[0, 2]。 • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]，默认值：50。 • enCscMatrix 默认值：单位矩阵。
Hi3520D/Hi3515A/Hi3515C	• VoDev 取值范围：[0, 2]。 • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]。标清设备的 u32Luma、u32Contrast、u32Hue 和 u32Satuature 的默认值均为 50；高清设备的 u32Luma、u32Contrast、u32Hue 和 u32Satuature 默认值分别为 50、50、50 和 59。 • enCscMatrix 默认值：标清设备为单位矩阵，高清设备为 VO_CSC_MATRIX_BT601_TO_BT709。
Hi3518/Hi3516C	• VoDev 取值范围：[0, VO_MAX_DEV_NUM) • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]，默认值：50。 • enCscMatrix 默认值：单位矩阵。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 该接口主要用于调节图像的输出效果，包括亮度、对比度、色调、饱和度，其取值范围为[0, 100]。
- CSC 转换矩阵只能选择 VO_CSC_MATRIX_IDENTITY、VO_CSC_MATRIX_BT601_TO_BT709 或 VO_CSC_MATRIX_BT709_TO_BT601。
- 当选择 VO_CSC_MATRIX_BT709_TO_BT601 矩阵时，VGA 的图像存在图像质量问题，所以，当使用 VGA 输出时，建议不要选择 VO_CSC_MATRIX_BT709_TO_BT601 这个矩阵。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 禁止视频层时会将亮度、对比度、色调、饱和度的值恢复为默认值，CSC 转换矩阵恢复为默认值。

【举例】

无。

【相关主题】

HI_MPI_VO_GetDevCSC

HI_MPI_VO_GetDevCSC

【描述】

获取设备输出图像效果。

【语法】

```
HI_S32 HI_MPI_VO_GetDevCSC(VO_DEV VoDev, VO_CSC_S *pstDevCSC);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstDevCSC	图像输出效果结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, 7]
Hi3532	0
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, 2]
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

该接口主要用于获取图像的输出效果，包括亮度、对比度、色调、饱和度，其取值范围在为[0, 100]。

【举例】

无。

【相关主题】

HI_MPI_VO_SetDevCSC

HI_MPI_VO_SetVgaParam

【描述】

设置设备 VGA 输出图像效果。

【语法】

```
HI_S32 HI_MPI_VO_SetVgaParam (VO_DEV VoDev, VO_VGA_PARAM_S *pstVgaParam);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstVgaParam	VGA 图像输出效果结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	• VoDev 取值范围：[0, 7]。 • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]，默认值：50。
Hi3532	• VoDev 取值范围：0。 • u32Luma、u32Contrast、u32Hue、

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
	u32Satuature 取值范围：[0, 100]，默认值：50。
Hi3521/Hi3520A	• VoDev 取值范围：[0, 2]。 • u32Luma、u32Contrast、u32Hue、u32Satuature 取值范围：[0, 100]，默认值：u32Satuature 为 59，其余为 50。
Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C	暂不支持

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 该接口主要用于调节 VGA 图像的输出效果，包括亮度、对比度、色调、饱和度和 DAC 增益，亮度、对比度、色调、饱和度的取值范围在 0 到 100 之间，DAC 增益的取值范围在 0 到 63 之间。
- CSC 转换矩阵只能选择 VO_CSC_MATRIX_BT601_TO_RGB_PC 或 VO_CSC_MATRIX_BT709_TO_RGB_PC。
- 禁止设备时会将亮度、对比度、色调、饱和度的值恢复为默认值，DAC 增益的值恢复为默认 10，CSC 转换矩阵的值恢复为默认 VO_CSC_MATRIX_BT601_TO_RGB_PC。
- HI_MPI_VO_SetDevCSC 接口中对亮度、对比度、色调、饱和度的调节会影响到该设备的所有接口的图像输出效果。HI_MPI_VO_SetVgaParam 接口中对亮度、对比度、色调、饱和度的调节仅影响该设备的 VGA 接口的图像输出效果。如果同时调用这两个接口，VGA 的图像输出效果是叠加的效果。

【举例】

无。

【相关主题】

HI_MPI_VO_GetDevCSC

HI_MPI_VO_GetVgaParam

【描述】

获取设备 VGA 输出图像效果。

【语法】

```
HI_S32 HI_MPI_VO_GetVgaParam(VO_DEV VoDev, VO_VGA_PARAM_S *pstVgaParam);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstVgaParam	VGA 图像输出效果结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, 7]
Hi3532	0
Hi3521/Hi3520A	[0, 2]
Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无

【举例】

无。

【相关主题】

HI_MPI_VO_SetDevCSC

HI_MPI_VO_SetDispBufLen

【描述】

设置显示缓冲的长度。

【语法】

```
HI_S32 HI_MPI_VO_SetDispBufLen(VO_DEV VoDev, HI_U32 u32BufLen);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
u32BufLen	显示缓冲的长度。 取值范围：[3, 15]	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)
Hi3532	[0]，[10, 13]
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证视频输出设备未使能。
- 缓冲长度具有默认值，标清设备和虚拟设备的默认值是 3，对于 Hi3531 和 Hi3532 高清设备和和级联显示设备的默认值是 6，对于 Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C/Hi3518/Hi3516C 的高清设备的默认值是 4。
- 标清设备和虚拟设备的缓冲长度最小值是 3，高清设备和级联显示设备的最小值是 3。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 10;
VO_PUB_ATTR_S stPubAttr;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_U32 u32DispBufLen = 10;

s32Ret = HI_MPI_VO_SetDispBufLen(VoDev, u32DispBufLen);
if (s32Ret != HI_SUCCESS)
{
    printf("Set display buf len failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_GetPubAttr(VoDev, &stPubAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Get device attributes failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
}

stPubAttr.u32BgColor = 0xff;
stPubAttr.enIntfType = VO_INTF_VGA;
stPubAttr.enIntfSync = VO_OUTPUT_1280x1024_60;

s32Ret = HI_MPI_VO_SetPubAttr(VoDev, &stPubAttr);
if (s32Ret != HI_SUCCESS)
{
    printf("Set device attributes failed with errno %#x!\n", s32Ret);
    return HI_FAILURE;
}

s32Ret = HI_MPI_VO_Enable(VoDev);
if (s32Ret != HI_SUCCESS)
{
    printf("Enable vo dev %d failed with errno %#x!\n", VoDev, s32Ret);
    return HI_FAILURE;

}
```

【相关主题】

无。

HI_MPI_VO_GetDispBufLen

【描述】

获取显示缓冲的长度。

【语法】

```
HI_S32 HI_MPI_VO_GetDispBufLen(VO_DEV VoDev, HI_U32 *pu32BufLen);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pu32BufLen	u32 类型的指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)
Hi3532	[0]，[10, 13]
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 10;
VO_PUB_ATTR_S stPubAttr;
HI_U32 u32DispBufLen;

s32Ret = HI_MPI_VO_GetDispBufLen(VoDev, &u32DispBufLen);
if (s32Ret != HI_SUCCESS)
{
    printf("Get display buf len failed with error code %#x!\n", s32Ret);
    return HI_FAILURE;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}

printf("Get dev %d disp buf len is %d.\n", VoDev, u32DispBufLen);
```

【相关主题】

无。

HI_MPI_VO_SetPlayToleration

【描述】

设置播放容忍度。

【语法】

```
HI_S32 HI_MPI_VO_SetPlayToleration(VO_DEV VoDev, HI_U32 u32Toleration);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
u32Toleration	播放容忍度。 取值范围：[1, 100000]	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)
Hi3532	[0]，[10, 13]
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- 调用前需保证设备已使能。
- 播放容忍度是以毫秒为单位，默认值为 10000。
- 当两帧解码数据的时间戳之差的绝对值超过容忍度时，系统将重置时间戳，以当前帧的时间戳为起点，对解码数据进行帧率控制。播放容忍度仅对解码图像的播放控制生效，对预览的播放控制无效。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 1;
VO_PUB_ATTR_S stPubAttr;
HI_U32 u32Toleration = 5000;

s32Ret = HI_MPI_VO_GetPubAttr(VoDev, &stPubAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Get device attributes failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}

stPubAttr.u32BgColor = 0xff;
stPubAttr.enIntfType = VO_INTF_VGA;
stPubAttr.enIntfSync = VO_OUTPUT_1280x1024_60;

s32Ret = HI_MPI_VO_SetPubAttr(VoDev, &stPubAttr);
if (s32Ret != HI_SUCCESS)
{
   printf("Set device attributes failed with errno %#x!\n", s32Ret);
   return HI_FAILURE;
}

s32Ret = HI_MPI_VO_Enable(VoDev);
if (s32Ret != HI_SUCCESS)
{
   printf("Enable vo dev %d failed with errno %#x!\n", VoDev, s32Ret);
   return HI_FAILURE;

}
s32Ret = HI_MPI_VO_ SetPlayToleration (VoDev, u32Toleration);
if (s32Ret != HI_SUCCESS)
{
   printf("Set play toleration failed with error code %#x!\n", s32Ret);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    return HI_FAILURE;
}
```

【相关主题】

无。

HI_MPI_VO_GetPlayToleration

【描述】

获取播放容忍度。

【语法】

```
HI_S32 HI_MPI_VO_GetPlayToleration(VO_DEV VoDev, HI_U32 *pu32Toleration);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pu32Toleration	u32 类型的指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	VoDev 取值范围
Hi3531	[0, VO_MAX_DEV_NUM)
Hi3532	[0]，[10, 13]
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	[0, VO_MAX_DEV_NUM)
Hi3518/Hi3516C	[0, VO_MAX_DEV_NUM)

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

调用前需保证设备已使能。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 0;
HI_U32 u32Toleration;

s32Ret = HI_MPI_VO_ GetPlayToleration (VoDev, & u32Toleration);
if (s32Ret != HI_SUCCESS)
{
   printf("Get play toleration failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}
printf("Get dev %d play toleration is %d.\n", VoDev, u32Toleration);
```

【相关主题】

无。

HI_MPI_VO_SetVtth

【描述】

设置设备垂直时序中断门限。

【语法】

```
HI_S32 HI_MPI_VO_SetVtth(VO_DEV VoDev, HI_U32 u32Vtth);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
u32Vtth	门限值，表示在倒数第 u32Vtth 行有效像素行处上报垂直时序中断。 取值范围：高清设备为[240, 8191]，标清设备为[100, 8191]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 一般在多片视频级联场景下，因业务量大，中断处理被延迟响应时，可通过本接口调大门限值，使垂直时序中断提前上报。其他情况下不建议调用本接口。
- 调用前需保证设备未使能。
- 只有物理设备支持设置门限值。
- 高清和标清设备的默认垂直时序中断门限分别为 240 和 100。用户必须保证设置的门限值不超过当前时序的有效像素行行数。

【举例】

无。

【相关主题】

HI_MPI_VO_GetVtth

HI_MPI_VO_GetVtth

【描述】

获取设备垂直时序中断门限值。

【语法】

```
HI_S32 HI_MPI_VO_GetVtth(VO_DEV VoDev, HI_U32 *pu32Vtth);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pu32Vtth	u32 类型的指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

只有物理设备支持获取门限值。

【举例】

无。

【相关主题】

HI_MPI_VO_SetVtth

HI_MPI_VO_EnableWbc

【描述】

使能视频回写。

【语法】

```
HI_S32 HI_MPI_VO_EnableWbc(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 支持回写功能的高清设备存在芯片差异，下文统一用 DHDx 设备称呼。
- DHDx 设备必须先使能。
- 必须先设置视频回写属性，再使能视频回写功能。
- 重复使能视频回写返回成功。
- 如果使能 DHDx 设备后立即使能 WBC，由于在短时间内 DHDx 可能还没接收到 VI 或其他模块发送的图像数据，此时 DHDx 会显示背景色，WBC 也将回写背景色。且如果 WBC 回写模式为 VO_WBC_MODE_PROG_TO_INTL（将两帧图像分别回写成顶底场图像，最终形成一帧图像），此时两帧图像可能刚好一帧是背景色图像，另一帧是数据图像，此时回写合成的一帧图像将一场是背景色另一场是图像数据。因此，建议使能 DHDx 后，等待该设备显示图像数据后再使能 WBC 回写功能。

【举例】

```
……
VO_WBC_ATTR_S stWbcAttr;
VO_WBC_MODE_E enWbcMode;
stWbcAttr.stTargetSize.u32Width = 720;
stWbcAttr.stTargetSize.u32Height = 576;
stWbcAttr.enPixelFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_422;
stWbcAttr.u32FrameRate = 25;
stWbcAttr.enDataSource = VO_WBC_DATASOURCE_MIXER;
}

if (HI_SUCCESS != HI_MPI_VO_SetWbcAttr(VoDev,&stWbcAttr))
{
    printf("Set wbc attr failed!\n");
    return HI_FAILURE;
}
if (HI_SUCCESS != HI_MPI_VO_GetWbcMode(VoDev, VO_WBC_MODE_NOMAL))
{
    printf("Get wbc mode failed!\n");
    return HI_FAILURE;
}
printf("Get wbc mode %d\n", enWbcMode);
enWbcMode = VO_WBC_MODE_NOMAL;

if (HI_SUCCESS != HI_MPI_VO_SetWbcMode(VoDev, enWbcMode))
{
    printf("Set wbc mode failed!\n");
    return HI_FAILURE;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}

/* enable wbc */
if (HI_SUCCESS != HI_MPI_VO_EnableWbc(VoDev))
{
    printf("Enable wbc failed!\n");
    return HI_FAILURE;
}
if (HI_SUCCESS != HI_MPI_VO_GetWbcAttr(VoDev,&stWbcAttr))
{
    printf("Get wbc attr failed!\n");
    return HI_FAILURE;
}

/* do some process, i.e. */

/* disable wbc */
if (HI_SUCCESS != HI_MPI_VO_DisableWbc (VoDev))
{
    printf("Disable wbc failed!\n");
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_DisableWbc

HI_MPI_VO_DisableWbc

【描述】

禁用视频回写。

【语法】

```
HI_S32 HI_MPI_VO_DisableWbc(VO_DEV VoDev);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

重复禁用视频回写返回成功。

【举例】

请参考 HI_MPI_VO_EnableWbc。

【相关主题】

HI_MPI_VO_EnableWbc

HI_MPI_VO_SetWbcAttr

【描述】

设置视频回写属性。

【语法】

```
HI_S32 HI_MPI_VO_SetWbcAttr(VO_DEV VoDev, const VO_WBC_ATTR_S
*pstWbcAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstWbcAttr	视频回写属性指针。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 属性中的目标大小不能超过回写设备设定的显示大小，且确保需要缩小的倍数在 WBC 允许的缩小倍数范围内。SPYCbCr422 格式图像最大支持 16 倍（不含 16 倍）缩小，SPYCbCr420 格式最大支持 8 倍（不含 8 倍）缩小。
- WBC 回写模式设为 VO_WBC_MODE_PROG_TO_INTL 时，SPYCbCr422 格式的图像最大支持缩放 8 倍（不含 8 倍）缩小，SPYCbCr420 格式最大支持 4 倍（不含 4 倍）缩小。
- 该接口为动态设置接口，可在 WBC 视频回写使能的情况下调用。
- WBC 回写数据源可以选择回写视频层 VO_WBC_DATASOURCE_VIDEO，或者，视屏层和图形层的叠加 VO_WBC_DATASOURCE_MIXER。当选择只回写视频层时，不会对绑定在该设备上面的 PIP 层进行回写。对于这项的配置只是针对于 Hi3520D/Hi3515A/Hi3515C 有效。对于其他的芯片，这个值配置任何值都是无效的，默认处理为 VO_WBC_DATASOURCE_MIXER 类型。
- 视频回写属性中，宽高必须大于 32，且 2 对齐。

【举例】

请参见 HI_MPI_VO_EnableWbc 的举例。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_GetWbcAttr

HI_MPI_VO_GetWbcAttr

【描述】

获取视频回写属性。

【语法】

```
HI_S32 HI_MPI_VO_GetWbcAttr(VO_DEV VoDev, VO_WBC_ATTR_S *pstWbcAttr);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstWbcAttr	视频回写属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

请参见 HI_MPI_VO_EnableWbc 的举例。

【相关主题】

HI_MPI_VO_SetWbcAttr

HI_MPI_VO_SetWbcMode

【描述】

设置视频回写模式。

【语法】

```
HI_S32 HI_MPI_VO_SetWbcMode(VO_DEV VoDev, VO_WBC_MODE_E enWbcMode);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
enWbcMode	视频回写模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 可调用该接口设置回写模式，达到最佳的回写显示效果。
- 该接口为静态接口，必须先禁用 WBC 后才能改变模式。
- 该接口使用前必须对 WBC 属性进行设置（包括在禁用 WBC 或调用该接口修改模式后，再重新使能的情况下，也需要重新设置 WBC 的属性。）
- WBC 回写模式设为 VO_WBC_MODE_PROG_TO_INTL 时，SPYCbCr422 格式的图像最大支持缩放 8 倍（不含 8 倍）缩小，SPYCbCr420 格式最大支持 4 倍（不含 4 倍）缩小。

【举例】

请参见 HI_MPI_VO_EnableWbc 的举例。

【相关主题】

无。

HI_MPI_VO_GetWbcMode

【描述】

获取视频回写模式。

【语法】

```
HI_S32 HI_MPI_VO_GetWbcMode(VO_DEV VoDev, VO_WBC_MODE_E *penWbcMode);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
penWbcMode	视频回写模式指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

请参考 HI_MPI_VO_EnableWbc。

【相关主题】

无。

HI_MPI_VO_SetWbcDepth

【描述】

设置获取视频回写图像的缓存深度。

【语法】

```
HI_S32 HI_MPI_VO_SetWbcDepth(VO_DEV VoDev, HI_U32 u32Depth);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
u32Depth	视频回写图像的缓存深度。 取值范围：[0, VO_DEF_WBC_DEPTH_LEN]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

用户可以获取回写图像用于编码、或标清显示、或其他目的，设置视频回写图像的缓存深度可以让用户获取回写图像时不丢失视频帧。

【举例】

```
HI_S32 s32Ret;
VO_DEV VoDev = 1;
HI_U32 u32Depth;
VIDEO_FRAME_INFO_S *pstFrame =
(VIDEO_FRAME_INFO_S*)malloc(sizeof(VIDEO_FRAME_INFO_S));

s32Ret = HI_MPI_VO_GetWbcDepth (VoDev, &u32Depth);
if (s32Ret != HI_SUCCESS)
{
   printf("Get wbc depth failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}

u32Depth = 10;

s32Ret = HI_MPI_VO_SetWbcDepth (VoDev, u32Depth);
if (s32Ret != HI_SUCCESS)
{
   printf("Set wbc depth failed with error code %#x!\n", s32Ret);
   return HI_FAILURE;
}

if (HI_SUCCESS != HI_MPI_VO_WbcGetScreenFrame(VoDev, pstFrame))
{
   printf("Get wbc screen frame failed!\n");
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    return HI_FAILURE;
}

/* do some process, i.e. store file or do JPEG encodeing ... */

if (HI_SUCCESS != HI_MPI_VO_WbcReleaseScreenFrame(VoDev, pstFrame))
{
    printf("Release wbc screen frame failed!\n");
    return HI_FAILURE;
}
```

【相关主题】

HI_MPI_VO_WbcGetScreenFrame

HI_MPI_VO_GetWbcDepth

【描述】

获取视频回写图像的缓存深度。

【语法】

```
HI_S32 HI_MPI_VO_GetWbcDepth(VO_DEV VoDev, HI_U32 *pu32Depth);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pu32Depth	视频回写图像的缓存深度指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

请参见 HI_MPI_VO_SetWbcDepth 的举例。

【相关主题】

HI_MPI_VO_WbcGetScreenFrame

HI_MPI_VO_WbcGetScreenFrame

【描述】

获取视频回写图像。

【语法】

```
HI_S32 HI_MPI_VO_WbcGetScreenFrame(VO_DEV VoDev, VIDEO_FRAME_INFO_S
*pstVFrame);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstFrame	获取的视频回写图像数据信息结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前应先设置视频回写缓存深度为大于 0 的值，否则获取不到图像。
- 调用前应该视频回写使能，否则获取不到图像。
- 获取后应保证及时的释放。

【举例】

请参考 HI_MPI_VO_SetWbcDepth 的举例。

【相关主题】

HI_MPI_VO_WbcReleaseScreenFrame

HI_MPI_VO_WbcReleaseScreenFrame

【描述】

释放视频回写图像。

【语法】

```
HI_S32 HI_MPI_VO_WbcReleaseScreenFrame(VO_DEV VoDev, VIDEO_FRAME_INFO_S
*pstVFrame);
```

【参数】

参数名称	描述	输入/输出
VoDev	视频输出设备号。	输入
pstFrame	释放的输出通道图像数据信息结构体指针。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531	DHD1 设备支持视频回写功能。 VoDev 取值范围：1
Hi3532/Hi3518/Hi3516C	不支持视频回写功能。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	DHD0 设备支持视频回写功能。 VoDev 取值范围：0

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

获取操作应保证与释放操作配对。

【举例】

请参见 HI_MPI_VO_SetWbcDepth 的举例。

【相关主题】

HI_MPI_VO_WbcGetScreenFrame

HI_MPI_VO_GfxLayerBindDev

【描述】

绑定图形层到 VO 设备。

【语法】

```
HI_S32 HI_MPI_VO_GfxLayerBindDev(VOU_GFX_BIND_LAYER_E enGfxLayer, VO_DEV
                                 VoTargetDev);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
enGfxLayer	图形层号。 取值范围：[0, VOU_GRAPHICS_LAYER_BUTT)。	输入
VoTargetDev	视频设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VO_GFX_NOT_DISABLE	图形层未关闭。
HI_ERR_VO_GFX_NOT_UNBIND	图形层未解绑定。

【差异说明】

芯片	描述
Hi3531	支持 G4/HC0/HC1 三个图形层动态绑定。 VoTargetDev 取值范围：[0, 3]
Hi3532/Hi3518/Hi3516C	不支持该接口。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	支持 HC0 动态绑定。 VoTargetDev 取值范围：[0, 2]

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证待绑定的图形层处于关闭状态。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 不允许重复绑定，需要先解绑定才能重新绑定。
- 不允许两个鼠标层同时绑定到同一个设备。
- 在 Hi3520D/Hi3515A/Hi3515C 中，当鼠标层动态切换绑定关系时，鼠标层的 CSC 模式不做动态修改，建议用户将鼠标层的 CSC 模式做如下设置：
  - 鼠标层绑定高清时，将鼠标层的 CSC 模式设置为 VO_CSC_MATRIX_RGB_TO_BT709_PC；
  - 鼠标层绑定标清时，将模式设置为 VO_CSC_MATRIX_RGB_TO_BT601_PC。

【举例】

无。

【相关主题】

HI_MPI_VO_GfxLayerUnBindDev

HI_MPI_VO_GfxLayerUnBindDev

【描述】

解除图形层与 VO 设备的解绑定关系。

【语法】

```
HI_S32 HI_MPI_VO_GfxLayerUnBindDev(VOU_GFX_BIND_LAYER_E enGfxLayer,
                                   VO_DEV VoTargetDev);
```

【参数】

参数名称	描述	输入/输出
enGfxLayer	图形层号。 取值范围：[0, VOU_GRAPHICS_LAYER_BUTT)。	输入
VoTargetDev	视频设备号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VO_GFX_NOT_DISABLE	图形层未关闭。

【差异说明】

芯片	描述
Hi3531	支持 G4/HC0/HC1 三个图形层动态绑定。 VoTargetDev 取值范围：[0, 3]
Hi3532/Hi3518/Hi3516C	不支持该接口。
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	支持 HC0 动态绑定。 VoTargetDev 取值范围：[0, 2]

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 调用前需保证待绑定的图形层处于关闭状态。
- VoTargetDev 目前没有什么意义，将其赋值为 0 即可。
- 对未绑定的图形层进行解绑定会返回成功，即允许对同一个图形层解绑定多次。

【举例】

无。

【相关主题】

HI_MPI_VO_GfxLayerBindDev

HI_MPI_VO_SetGfxLayerCSC

【描述】

设置图形层输出图像效果。

【语法】

```
HI_S32 HI_MPI_VO_SetGfxLayerCSC(HI_U32 u32Layer, const VO_CSC_S *pstCSC);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
u32Layer	图形层号。 取值范围：[0, VOU_GRAPHICS_LAYER_NUM)	输入
pstCSC	图形层输出效果结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532/ Hi3521/Hi3520A	支持包括亮度、对比度、色调、饱和度的调节及 CSC 转换矩阵的选择。
Hi3518/Hi3516C	仅支持 CSC 转换矩阵选择。
Hi3520D/Hi3515A/Hi3515C	G0/G1/G2 支持包括亮度、对比度、色调、饱和度的调节及 CSC 转换矩阵的选择。 HC0 鼠标层仅支持转换矩阵的选择，其他 4 个属性值用户可以随意设置，但是不会对图像产生效果，同时在获取这 4 个属性值时，如果用户没有设置，则返回默认值，反之则返回用户自己设置的值。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 该接口主要用于调节图像的输出效果，包括亮度、对比度、色调、饱和度，其取值范围为[0, 100]。
- CSC 转换矩阵只能选择 VO_CSC_MATRIX_RGB_TO_BT601_PC 或 VO_CSC_MATRIX_RGB_TO_BT709_PC。
- 禁止图形层时会将亮度、对比度、色调、饱和度的值恢复为默认值。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VO_GetGfxLayerCSC

HI_MPI_VO_GetGfxLayerCSC

【描述】

获取图形层输出图像效果。

【语法】

```
HI_S32 HI_MPI_VO_GetGfxLayerCSC(HI_U32 u32Layer, VO_CSC_S *pstCSC);
```

【参数】

参数名称	描述	输入/输出
u32Layer	图形层号。 取值范围：[0, VOU_GRAPHICS_LAYER_NUM)	输入
pstCSC	图形层输出效果结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

该接口主要用于获取图形层的输出效果，包括亮度、对比度、色调、饱和度，其取值范围为[0, 100]。

【举例】

无。

【相关主题】

HI_MPI_VO_SetGfxLayerCSC

HI_MPI_VO_SetCascadeAttr

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

设置视频级联属性。

【语法】

```
HI_S32 HI_MPI_VO_SetCascadeAttr(const VO_CAS_ATTR_S *pstCasAttr);
```

【参数】

参数名称	描述	输入/输出
pstCasAttr	视频输出级联属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	Hi3531/Hi3532 级联支持 64/32 两种区域类型，如果与 Hi3520 级联需配置为 32 区域。 Hi3531/Hi3532 级联支持 1080P60 奇数帧和偶数帧分别传输两路不同的数据，如果与 Hi3520 级联 enCasMode 需配置为单路模式，且必须为 1080P30 的级联数据。
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 设置视频级联属性时必须保证设备已禁止，即要先设置视频级联属性再使能设备。
- 属性中的 bSlave 标识主从模式。级联的第一芯片设置为主模式，其它级联芯片设置为从模式。
- 属性中的 enCasMode 标识单路双路模式。
  - 级联双路模式支持 1080P60 奇数帧和偶数帧分别传输两路不同的数据，即相当于双路 1080P30。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 级联单路模式支持 1080P60 和 1080P30。

【举例】

无。

【相关主题】

HI_MPI_VO_GetCascadeAttr

HI_MPI_VO_GetCascadeAttr

【描述】

获取视频级联属性。

【语法】

```
HI_S32 HI_MPI_VO_GetCascadeAttr(VO_CAS_ATTR_S *pstCasAttr);
```

【参数】

参数名称	描述	输入/输出
pstCasAttr	视频级联属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

无。

【相关主题】

HI_MPI_VO_SetCascadeAttr

HI_MPI_VO_EnableCascadeDev

【描述】

使能级联扩展设备。

【语法】

```
HI_S32 HI_MPI_VO_EnableCascadeDev (VO_DEV VoCasDev);
```

【参数】

参数名称	描述	输入/输出
VoCasDev	级联扩展设备。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 级联扩展设备是一个软件概念，不是实际的硬件设备。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 级联属性设置为双路模式时，需使能两个级联扩展设备（VO_CAS_DEV_1 和 VO_CAS_DEV_2）传输级联数据。
- 级联属性设置为单路模式时，只能使能级联扩展设备 VO_CAS_DEV_1。
- 需要在级联设备 DEV0 的视频层使能后才能调用此接口，使能级联扩展设备。
- 级联扩展设备不需要设置硬件相关的设备公共属性、视频层属性，只需对级联设备 DEV0 设置以上属性即可。
- 级联情况下，级联设备 DEV0 的通道不可用，其他模块需绑定级联扩展设备的通道发送数据。

【举例】

无。

【相关主题】

HI_MPI_VO_DisableCascadeDev

HI_MPI_VO_DisableCascadeDev

【描述】

禁止级联扩展设备。

【语法】

```
HI_S32 HI_MPI_VO_DisableCascadeDev (VO_DEV VoCasDev);
```

【参数】

参数名称	描述	输入/输出
VoCasDev	级联扩展设备。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 重复关闭视为成功。
- 必须先禁止级联扩展设备上的所有通道。

【举例】

无。

【相关主题】

HI_MPI_VO_EnableCascadeDev

HI_MPI_VO_SetCascadePattern

【描述】

设置视频级联画面样式。

【语法】

```
HI_S32 HI_MPI_VO_SetCascadePattern(VO_DEV VoCasDev, HI_U32 u32Pattern);
```

【参数】

参数名称	描述	输入/输出
VoCasDev	级联扩展设备。	输入
u32Pattern	视频级联画面布局样式。 取值范围：[0, 127]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C	不支持该接口。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

画面布局样式是一个小于 128 的正整数，它的主要作用是检测各片之间传输图像的一致性。即如果要保证图像传输的正确性，就必须保证各片的 pattern 一致。

- 用 pattern 来标识当前画面布局样式，如 4 画面、9 画面、16 画面的 pattern 对应设置为 4、9、16。设置的 pattetrn 值不必和画面布局样式中的画面个数一致，只要保证各片设置的 pattern 值一致即可，如也可以用 1 来表示每个片子上都是 16 分屏的画面布局。
- 使用 pattern 的另一个作用是确保多画面切换时的同步性。当级联的芯片正在进行 4 画面显示时，其中一片先切换到 9 画面，而其它片子陆续切换为 9 画面，如果 pattern 始终不变，画面就会在切换过程中出现错误的几帧。但是如果在 4 画面时 pattern 设置为 4，在 9 画面时 pattern 设置为 9，那么切换过程中出现的错误图像就会被丢弃，保证画面切换的正确性。

【举例】

无。

【相关主题】

HI_MPI_VO_GetCascadePattern

HI_MPI_VO_GetCascadePattern

【描述】

获取视频级联画面样式。

【语法】

```
HI_S32 HI_MPI_VO_GetCascadePattern(VO_DEV VoCasDev, HI_U32 *pu32Pattern);
```

【参数】

参数名称	描述	输入/输出
VoCasDev	级联扩展设备。	输入
pu32Pattern	视频级联画面布局样式指针。	输出

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

HI_MPI_VO_SetCascadePattern

HI_MPI_VO_CascadePosBindChn

【描述】

绑定级联区域与视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_CascadePosBindChn (HI_U32 u32Pos, VO_DEV VoCasDev,
VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
u32Pos	视频级联位置编号。 取值范围：[0, 31]或[0, 63]。	输入
VoCasDev	级联扩展设备。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VoChn	视频层通道号。 取值范围：[0, 63]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 视频级联位置编号必须在所有的级联芯片上是唯一标识的。
- 级联属性设置为 64 区域时，Pos 取值范围为[0, 63]；级联属性设置为 32 区域时，Pos 取值范围为[0, 31]。
- Pos 的值不能有重复，如果存在两个或者两个以上通道的 Pos 值相同，传输图像就会错误。Pos 相当于给 Chn 取了个别名，所有需要级联的通道都必须有一个名字区别于其它通道，这样才能保证传输图像的正确性。
- 此接口支持批处理操作，并建议在 HI_MPI_VO_SetAttrBegin 和 HI_MPI_VO_SetAttrEnd 之间调用。

【举例】

无。

【相关主题】

HI_MPI_VO_CascadePosUnBindChn

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_CascadePosUnBindChn

【描述】

解绑定级联区域与视频输出通道。

【语法】

```
HI_S32 HI_MPI_VO_CascadePosUnBindChn(HI_U32 u32Pos, VO_DEV VoCasDev,
VO_CHN VoChn);
```

【参数】

参数名称	描述	输入/输出
u32Pos	视频级联位置编号。 取值范围：[0, 31] 或[0, 63]。	输入
VoCasDev	级联扩展设备。	输入
VoChn	视频层通道号。 取值范围：[0, 63]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 视频级联位置编号必须在所有的级联芯片上是唯一标识的。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 级联属性设置为 64 区域时，Pos 取值范围为[0, 63]；级联属性设置为 32 区域时，Pos 取值范围为[0, 31]。
- 此接口支持批处理操作，并建议在 HI_MPI_VO_SetAttrBegin 和 HI_MPI_VO_SetAttrEnd 之间调用。

【举例】

无。

【相关主题】

HI_MPI_VO_CascadePosBindChn

HI_MPI_VO_EnableCascade

【描述】

使能视频级联。

【语法】

```
HI_S32 HI_MPI_VO_EnableCascade (HI_VOID);
```

【参数】

无。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

- 级联只能在设备 DEV0 上进行。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 级联使能前必须保证设置级联属性、设备 DEV0 使能和该设备上的视频层使能。

【举例】

无。

【相关主题】

HI_MPI_VO_DisableCascade

HI_MPI_VO_DisableCascade

【描述】

禁止视频级联。

【语法】

```
HI_S32 HI_MPI_VO_DisableCascade(HI_VOID);
```

【参数】

无。

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【差异说明】

芯片	描述
Hi3531/Hi3532	支持该接口
Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/ Hi3515A/Hi3515C	不支持该接口。

【需求】

- 头文件：mpi_vo.h、hi_comm_vo.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_VO_EnableCascade

4.4 数据类型

视频输出相关数据类型定义如下：

- VO_DEV：定义设备号。
- VO_PUB_ATTR_S：定义视频输出设备属性结构体。
- VO_VIDEO_LAYER_ATTR_S：定义视频层属性结构体。
- VO_CHN_ATTR_S：定义视频输出通道属性结构体。
- VO_DISPLAY_FIELD_E：定义视频输出时的顶底场模式。
- VO_QUERY_STATUS_S：定义视频输出通道状态结构体。
- VO_ZOOM_RATIO_S：定义按比例局部放大结构体。
- VO_ZOOM_ATTR_S：定义局部放大属性结构体。
- VO_CSC_S：定义图像输出效果结构体。
- VO_VGA_PARAM_S：定义 VGA 图像输出效果结构体。
- VO_WBC_ATTR_S：定义视频回写属性结构体。
- VO_CAS_MODE_E：定义级联的单路双路模式。
- VO_CAS_RGN_E：定义级联的区域类型。
- VO_CAS_ATTR_S：定义级联属性。

VO_DEV

【说明】

定义设备号。

【定义】

```
typedef HI_S32 VO_DEV;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3531	视频输出模块有 14 个视频输出设备，分别进行如下定义： 0：DHD0 设备，即高清显示设备 0。 1：DHD1 设备，即高清显示设备 1。 2：DSD0 设备，即标清显示设备 0。 3：DSD1 设备，即标清显示设备 1。 4：DSD2 设备，即标清显示设备 2。 5：DSD3 设备，即标清显示设备 3。 6：DSD4 设备，即标清显示设备 4。 7：DSD5 设备，即标清显示设备 5。 8：VO_CAS_DEV_1 设备，即级联扩展设备 1。 9：VO_CAS_DEV_2 设备，即级联扩展设备 2。 10：VO_VIRT_DEV_0 设备，即虚拟设备 0 11：VO_VIRT_DEV_1 设备，即虚拟设备 1 12：VO_VIRT_DEV_2 设备，即虚拟设备 2 13：VO_VIRT_DEV_3 设备，即虚拟设备 3
Hi3532	视频输出模块有 14 个视频输出设备，分别进行如下定义： 0：DHD0 设备，即高清显示设备 0。 1~7：保留设备号，目前无效。 8：VO_CAS_DEV_1 设备，即级联扩展设备 1。 9：VO_CAS_DEV_2 设备，即级联扩展设备 2。 10：VO_VIRT_DEV_0 设备，即虚拟设备 0 11：VO_VIRT_DEV_1 设备，即虚拟设备 1 12：VO_VIRT_DEV_2 设备，即虚拟设备 2 13：VO_VIRT_DEV_3 设备，即虚拟设备 3
Hi3521/Hi3520A/Hi3520D /Hi3515A/Hi3515C	视频输出模块有 7 个视频输出设备，分别进行如下定义： 0：DHD0 设备，即高清显示设备 0。 1：DSD0 设备，即标清显示设备 0。 2：DSD1 设备，即标清显示设备 1。 3：VO_VIRT_DEV_0 设备，即虚拟设备 0 4：VO_VIRT_DEV_1 设备，即虚拟设备 1 5：VO_VIRT_DEV_2 设备，即虚拟设备 2 6：VO_VIRT_DEV_3 设备，即虚拟设备 3

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3518/Hi3516C	视频输出模块仅有 1 个视频输出设备，进行如下定义： 0：DSD1 设备，即标清显示设备 1

【芯片差异】

芯片	描述
Hi3531	• 高清设备支持同时输出 BT.1120、HDMI、VGA(YPBPR)高清数字、模拟信号。 • 两个高清设备同时只能有一路 BT.1120、HDMI、VGA(YPBPR)输出。 • VGA 与 YPBPR 不能同时输出。 • 高清设备支持 BT656 输出，当 DHD0 输出 BT656 时，DHD0 的其他输出接口不能用，同样的，当 DHD1 输出 BT656 时，DHD1 的其他输出接口也不能用。 • 标清设备 DSD0～DSD1，支持同时输出 BT.656、CVBS（PAL 制和 NTSC 制）信号，标清设备 DSD2～DSD5 只支持 BT656 输出。DSD5 和高清设备的 PIP 视频层复用，二者只能使用一个。 • 每个设备最多容纳 64 个通道。 • 级联扩展设备不是实际的硬件设备，不能设置设备属性和视频层属性。
Hi3532	• Hi3532 仅有 DHD0 一个高清设备，且只能输出 BT.1120。 • 级联扩展设备不是实际的硬件设备，不能设置设备属性和视频层属性。
Hi3521/ Hi3520A	• 高清设备支持同时输出 BT.1120（LCD）、HDMI、VGA 高清数字、模拟信号。 • LCD 不能与 BT1120、VGA 同时输出。且如果 LCD 使能，标清设备的 BT656 输出接口不能使用。 • 标清设备支持同时输出 BT.656、CVBS 信号，BT656 接口可以选择输出 DSD0 或 DSD1 的视频数据，只能二选一。 • BT1120 其中的高 8bit 或低 8bit 可作为标清设备的 BT656 输出接口，选择输出 DSD0 或 DSD1 设备的视频数据（需硬件支持此连线方案）。此时高清设备的 BT1120 接口将不能使用。此 BT656 接口不能和标清设备自有的 BT656 接口同时输出同一设备的视频数据。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3518/Hi3516C	• 不支持高清设备。 • 标清设备支持输出 BT1120 或 CVBS 信号，但不能同时输出。
Hi3520D/Hi3515A/Hi3515C	• 高清设备支持同时输出 HDMI、VGA 等高清数字、模拟信号。 • 标清设备只支持输出 CVBS 信号。

【注意事项】

无。

【相关数据类型及接口】

VO_PUB_ATTR_S

VO_PUB_ATTR_S

【说明】

定义视频输出公共属性结构体。

【定义】

```
typedef struct hiVO_PUB_ATTR_S
{
    HI_U32                u32BgColor;         /* 设备背景色 RGB表示 */
    VO_INTF_TYPE_E        enIntfType;         /* Vo 接口类型 */
    VO_INTF_SYNC_E        enIntfSync;         /* Vo接口时序类型 */
    VO_SYNC_INFO_S        stSyncInfo;         /* Vo接口时序信息 */
    HI_BOOL               bDoubleFrame;       /* 是否需要倍帧 */
} VO_PUB_ATTR_S;
```

【成员】

成员名称	描述
u32BgColor	设备背景色，表示方法 RGB888。
enIntfType	接口类型典型配置，原型定义： Typedef HI_S32 VO_INTF_TYPE_E; #define VO_INTF_CVBS (0x01L<<0) #define VO_INTF_YPBPR (0x01L<<1) #define VO_INTF_VGA (0x01L<<2) #define VO_INTF_BT656 (0x01L<<3) #define VO_INTF_BT1120 (0x01L<<4)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
	#define VO_INTF_HDMI (0x01L<<5) #define VO_INTF_LCD (0x01L<<6) #define VO_INTF_BT656_H (0x01L<<7) #define VO_INTF_BT656_L (0x01L<<8)
enIntfSync	接口时序典型配置，原型定义： typedef enum hiVO_INTF_SYNC_E { VO_OUTPUT_PAL = 0, VO_OUTPUT_NTSC, VO_OUTPUT_1080P24, VO_OUTPUT_1080P25, VO_OUTPUT_1080P30, VO_OUTPUT_720P50, VO_OUTPUT_720P60, VO_OUTPUT_1080I50, VO_OUTPUT_1080I60, VO_OUTPUT_1080P50, VO_OUTPUT_1080P60, VO_OUTPUT_576P50, VO_OUTPUT_480P60, VO_OUTPUT_800x600_60, VO_OUTPUT_1024x768_60, VO_OUTPUT_1280x1024_60, VO_OUTPUT_1366x768_60, VO_OUTPUT_1440x900_60, VO_OUTPUT_1280x800_60, VO_OUTPUT_1600x1200_60, VO_OUTPUT_1680x1050_60, VO_OUTPUT_1920x1200_60, VO_OUTPUT_640x480_60, VO_OUTPUT_1920x2160_30, VO_OUTPUT_2560x1440_30, VO_OUTPUT_USER, VO_OUTPUT_BUTT } VO_INTF_SYNC_E;

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stSyncInfo	接口时序结构体，原型定义： typedef struct tagVO_SYNC_INFO_S { HI_BOOL bSynm; HI_BOOL bIop; HI_U8 u8Intfb; HI_U16 u16Vact ; HI_U16 u16Vbb; HI_U16 u16Vfb; HI_U16 u16Hact; HI_U16 u16Hbb; HI_U16 u16Hfb; HI_U16 u16Hmid; HI_U16 u16Bvact; HI_U16 u16Bvbb; HI_U16 u16Bvfb; HI_U16 u16Hpw; HI_U16 u16Vpw; HI_BOOL bIdv; HI_BOOL bIhs; HI_BOOL bIvs; } VO_SYNC_INFO_S;

【芯片差异】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3531	支持 VO_INTF_CVBS、VO_INTF_YPBPR、VO_INTF_VGA、VO_INTF_BT656、VO_INTF_BT1120、VO_INTF_HDMI 时序 接口时序配置为 VO_OUTPUT_1920x2160_30 时，视频层、PIP、WBC 的时序要设置成 60 帧每秒，否则会有丢帧；HDMI 视频制式设成 HI_HDMI_VIDEO_FMT_1080P_60。 接口时序配置为 VO_OUTPUT_2560x1440_30 时，视频层、PIP、WBC 的时序要设置成 60 帧每秒，否则会有丢帧；HDMI 视频制式设成 HI_HDMI_VIDEO_FMT_VESA_2560X1440_30。
Hi3532	仅支持 VO_INTF_BT1120 时序
Hi3521/ Hi3520A	支持 VO_INTF_CVBS、VO_INTF_VGA、VO_INTF_BT656、VO_INTF_BT1120、VO_INTF_HDMI、VO_INTF_LCD、VO_INTF_BT656_H、VO_INTF_BT656_L 时序 VO_INTF_LCD 与 BT1120、VGA、BT656、BT656_H、BT656_L 不能同时使用。 BT1120 与 BT656_H、BT656_L 不能同时使用.
Hi3518/Hi3516C	支持 VO_INTF_BT1120、VO_INTF_CVBS 时序。VO_INTF_BT1120 与 VO_INTF_CVBS 不能同时使用。
Hi3520D/Hi3515A/Hi3515C	支持 VO_INTF_CVBS、VO_INTF_VGA、VO_INTF_HDMI 时序。

【注意事项】

- 当接口时序配置为 VO_OUTPUT_USER 时，stSyncInfo 定义的时序结构才会生效，表示用户自定义的时序结构。
- 同时使用多个接口类型示意：enIntfSync = VO_INTF_BT1120 | VO_INTF_HDMI。
- 接口类型配置为 VO_INTF_CVBS 或 VO_INTF_BT656 时，enIntfSync 的取值范围为[0, 1]。
- 接口类型配置为 VO_INTF_BT1120、VO_INTF_HDMI、VO_INTF_VGA 时，enIntfSync 的取值范围为[2, 22]。接口类型配置为 VO_INTF_YPBPR 时，enIntfSync 的取值范围为[5, 10]。
- 配置设备属性在设备下一次打开时生效。

【相关数据类型及接口】

HI_MPI_VO_SetPubAttr

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VO_VIDEO_LAYER_ATTR_S

【说明】

定义视频层属性。

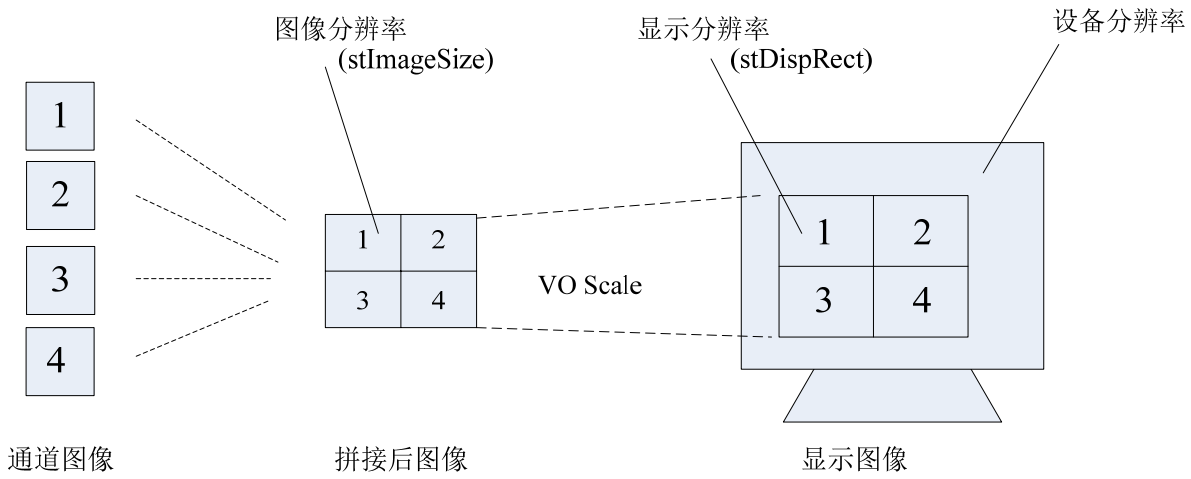
在视频层属性中存在三个概念，即设备分辨率、显示分辨率和图像分辨率。每种分辨率的概念可以从图 4-1 中看出。

图像分辨率指放置各个通道图像的画布大小。

显示分辨率是把图像分辨率中描述的画布经过 VO 放大后的显示区域。

设备分辨率与设备时序一致，即如果时序为 1920 x 1080，那设备分辨率就为 1920 x 1080。

图4-1 视频层属性的相关概念示意



【定义】

```
typedef struct hiVO_VIDEO_LAYER_ATTR_S
{
    RECT_S stDispRect;                  /* 显示分辨率大小 */
    SIZE_S stImageSize;                 /* 视频层画布大小 */
    HI_U32 u32DispFrmRt;                /* 显示帧率 */
    PIXEL_FORMAT_E enPixFormat;         /* 视频层使用的像素格式 */
} VO_VIDEO_LAYER_ATTR_S;
```

【成员】

成员名称	描述
stDispRect	视频显示区域矩形结构体。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stImageSize	图像分辨率结构体，即合成画面尺寸。
u32DispFrmRt	视频显示帧率。
enPixFormat	视频层输入像素格式，SPYCbCr420 或者 SPYCbCr422。

【差异说明】

芯片	描述
Hi3531	DHD0 视频层的最大分辨率为 2560 x 1440，即 stDispRect 不能超过这个值。DHD1 视频层的最大分辨率为 2560 x 1600，DSD0~DSD5 视频层的最大分辨率为 720 x 576. HD 的缩放存在限制，DHD0 不支持宽度为 1280(不含 1280)至 1920 的分辨率放大到 1920，当宽度超过 1920 时，只支持 YUV422 格式显示。DHD1 不支持宽度为 2048(不含 2048)至 2560 的分辨率放大到 2560。
Hi3532	DHD0 视频层的最大分辨率为 1920 x 1080，即 stDispRect 不能超过这个值。不支持视频层缩放。
Hi3521/ Hi3520A	DHD0 视频层的最大分辨率为 1920 x 1080，即 stDispRect 不能超过这个值。DSD0~DSD1 视频层的最大分辨率为 720 x 576. HD 的缩放存在限制，DHD0 不支持宽度为 1280(不含 1280)至 1920 的分辨率放大到 1920。
Hi3518/Hi3516C	DSD1 视频层的最大分辨率为 1920x1080，即 stDispRect 不能超过这个值。
Hi3520D/Hi3515A/ Hi3515C	DHD0 视频层的最大分辨率为 1920 x 2048，即 stDispRect 不能超过这个值。DSD0~DSD1 视频层的最大分辨率为 720 x 576. HD 的缩放存在限制，DHD0 不支持宽度为 1440(不含 1440)至 1920 的分辨率放大到 1920。

【注意事项】

- stDispRect 范围不能超出设备分辨率。stImageSize 和 stDispRect 的范围都要大于或等于显示的最小分辨率 32x32。
- HD 设备支持缩放，不支持裁剪。stImageSize 如果小于 stDispRect，视频层的图像将从 stImageSize 大小放大到 stDispRect 大小；stImageSize 不能大于 stDispRect。
- SD 设备不支持缩放，可以裁剪。stImageSize 如果大于 stDispRect，合成图像将被剪裁。
- SD 要求 stImageSize 和 stDispRect 的 Y 和 H 必须是 4 对齐，这是因为 SD 设备是将 TDE 叠加后的图像配置硬件显示，硬件要求 Y 和 H 是 4 对齐，TDE 叠加的图像的分辨率由 stImageSize 和 stDispRect 决定。X 和 W 只需 2 对齐即可。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HD 的 stImageSize 和 stDispRect 只需 2 对齐即可。
- u32DispFrmRt 是视频显示帧率。该值可设范围为(0, VO_DISP_MAX_FRMRATE]，该值一般设置为通道满帧率，即：
  - 如果源是 PAL 制式，设置为 25；
  - 如果源是 NTSC 制式，设置为 30。

  如果设置的帧率比通道的源帧率高，则会造成不必要的性能浪费。
- SD 设备上不支持 VO 的缩放，所以对于 SD 上的输出，通常保持图像分辨率和显示分辨率一致。
- HD 设备上的 VO 缩放只支持放大。即图像分辨率不能大于显示分辨率。
- SD 和 HD 的 stDispRect 中的起始位置 s32X 和 s32Y 无效，视频层始终是以(0，0)为起始坐标。用户通过配置设备下各通道的起始位置来实现任意位置显示。

【相关数据类型及接口】

HI_MPI_VO_SetPubAttr

VO_CHN_ATTR_S

【说明】

定义视频输出通道属性。

【定义】

```
typedef struct hiVO_CHN_ATTR_S
{
    HI_U32  u32Priority;                /* 通道优先级 */
    RECT_S  stRect;                     /* 通道显示区域 */
    HI_BOOL bDeflicker;                 /* 是否开启抗闪烁 */
}VO_CHN_ATTR_S;
```

【成员】

成员名称	描述
u32Priority	视频通道叠加优先级，优先级高的在上层。 高清设备取值范围：[0, 1]。 标清设备取值范围：[0, VO_MAX_CHN_NUM-1]。 动态属性。
stRect	通道矩形显示区域。以屏幕的左上角为原点。其取值必须是 2 对齐，且该矩形区域必须在屏幕范围之内。**注意：当高清设备是隔行时序且 SPYCbCr420 显示时，高度必须 4 对齐。** 动态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bDeflicker	是否开启抗闪烁。 HI_TRUE：开启； HI_FALSE：关闭。 抗闪烁效果仅针对使用 TDE 缩放的设备的通道有效，即该参数仅对标清设备和虚拟设备有效。 动态属性。

【差异说明】

芯片	描述
Hi3531	• 高清设备的通道优先级有 0 和 1，标清设备的优先级为 0～VO_MAX_CHN_NUM-1 • 标清设备的通道宽高必须不小于 16
Hi3532	• 高清设备的通道优先级只有 0 • 标清设备的通道宽高必须不小于 16
Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C	• 高清设备的通道优先级有 0 和 1，标清设备的优先级为 0～VO_MAX_CHN_NUM-1 • 标清设备的通道宽高必须不小于 16
Hi3518/Hi3516C	• 标清设备的通道优先级为 0~VO_MAX_CHN_NUM-1 • 标清设备的通道宽高必须不小于 32

【注意事项】

- 属性中的优先级，数值越大优先级越高。
  - 高清设备：优先级有 0 与 1，优先级为 0 代表通道在设备原视频层上，优先级为 1 代表通道在 PIP 层上，PIP 层总是叠加在原视频层之上。高清设备视频层的通道不允许出现重叠，PIP 层的通道也不允许出现重叠。
  - 标清设备：优先级有 0～ VO_MAX_CHN_NUM-1，当多个通道有重叠的显示区域时，优先级高的通道图像将覆盖优先级低的通道。优先级相同的各通道有重叠时，默认通道号大的图像将覆盖通道号小的通道图像。
- 通道显示区域不能超过视频层属性中设定的画布大小(stImageSize 大小)。
- HD 设备的通道属性 stRect 需 2 对齐，在隔行时序且像素格式为 SEMIPLANAR_420 时，stRect 的高度必须 4 对齐。HD 的宽高必须大于等于 32。
- 如果 HD 有放大情况，stRect 是放大前视频层上的起始位置和宽高，放大后起始位置和宽高会按比例偏移或放大。
- SD 设备的通道属性 stRect 需 2 对齐。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VO_SetChnAttr

VO_DISPLAY_FIELD_E

【说明】

定义视频输出时的顶底场模式。

【定义】

```
typedef enum hiVO_DISPLAY_FIELD_E
{
  VO_FIELD_TOP,     /* top field*/
  VO_FIELD_BOTTOM,  /* bottom field*/
  VO_FIELD_BOTH,    /* top and bottom field*/
  VO_FIELD_BUTT
} VO_DISPLAY_FIELD_E;
```

【成员】

成员名称	描述
VO_FIELD_TOP	通道视频输出时只处理顶场。
VO_FIELD_BOTTOM	通道视频输出时只处理底场。
VO_FIELD_BOTH	通道视频输出缩放两场都处理。

【注意事项】

无。

【相关数据类型及接口】

无。

VO_QUERY_STATUS_S

【说明】

视频输出通道状态结构体。

【定义】

```
typedef struct hiVO_QUERY_STATUS_S
{
    HI_U32 u32ChnBufUsed;      /* channel buffer that been occupied */
} VO_QUERY_STATUS_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32ChnBufUsed	视频输出通道当前占用的视频 buffer 数目。

【注意事项】

无。

【相关数据类型及接口】

无。

VO_ZOOM_RATIO_S

【说明】

定义按比例局部放大结构体。

【定义】

```
typedef struct hiVO_ZOOM_RATIO_S
{
   HI_U32 u32XRatio;
   HI_U32 u32YRatio;
   HI_U32 u32WRatio;
   HI_U32 u32HRatio;
 } VO_ZOOM_RATIO_S;
```

【成员】

成员名称	描述
u32XRatio	屏幕坐标上，待缩放区域起始点横坐标与显示通道宽的比例。
u32YRatio	屏幕坐标上，待缩放区域起始点纵坐标与显示通道高的比例。
u32WRatio	屏幕坐标上，待缩放区域宽与显示通道宽的比例。
u32HRatio	屏幕坐标上，待缩放区域高与显示通道高的比例。

【注意事项】

- 结构体成员变量的取值范围[0, 1000]。
- 比例值是放大了 1000 倍的，譬如待缩放区域宽与显示通道宽的比例是 0.6，对应的参数 u32WRatio 则是 600。
- 取消局部放大功能时，只要将该结构体中的成员都置 0 即可。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

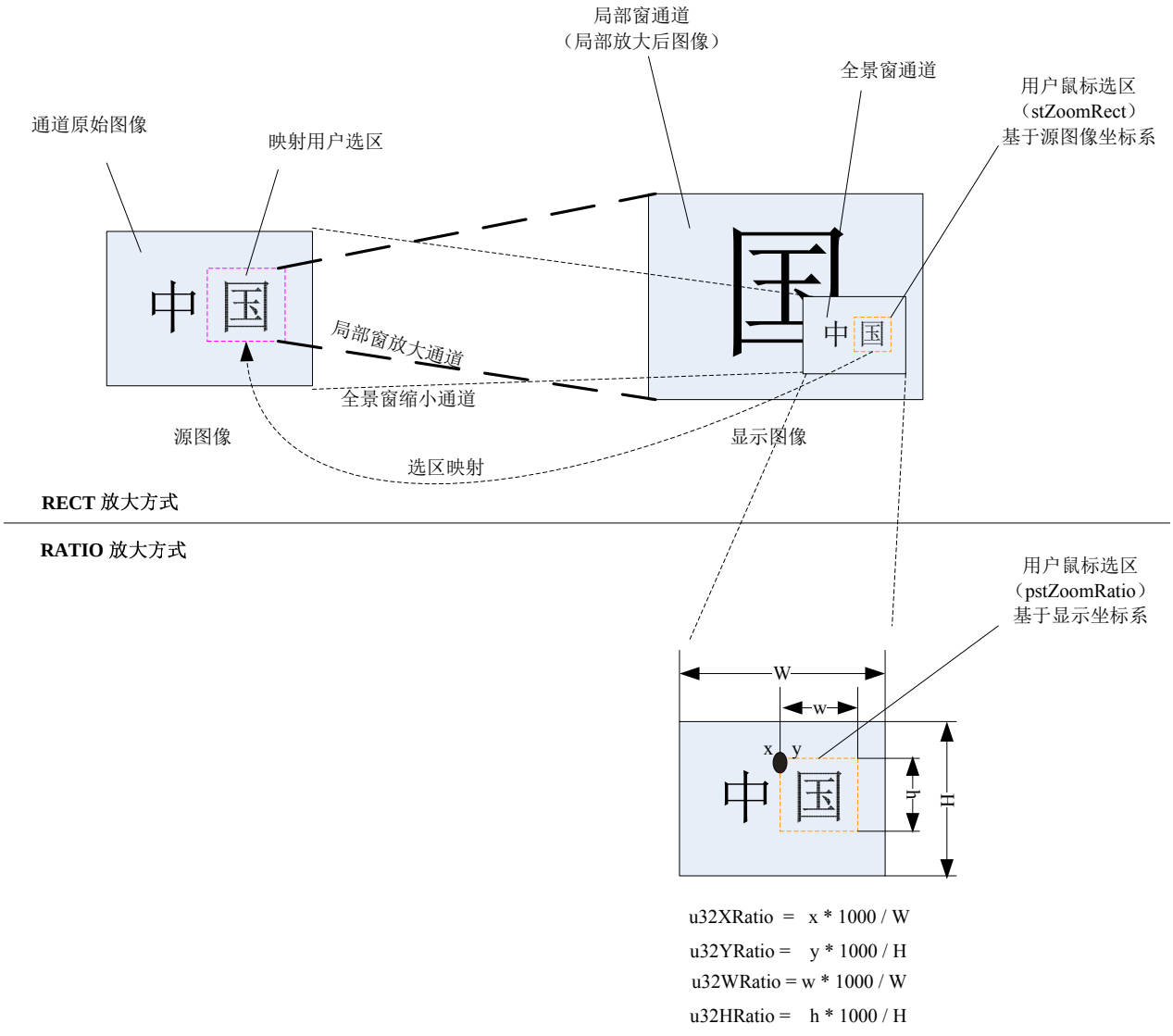
HI_MPI_VO_SetZoomInWindow

VO_ZOOM_ATTR_S

【说明】

定义局部放大属性结构体。局部放大原理示意如图 4-2 所示。

图4-2 局部放大原理示意



【定义】

```
typedef struct hiVO_ZOOM_ATTR_S
{
    VOU_ZOOM_IN_E   enZoomType;         /* choose the type of zoom in */
    union
    {
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
        RECT_S          stZoomRect;     /* zoom in by rect */
        VO_ZOOM_RATIO_S stZoomRatio;    /* zoom in by ratio */
    };
} VO_ZOOM_ATTR_S;
```

【成员】

成员名称	描述
enZoomType	局部放大类型
stZoomRect	局部放大窗口矩形坐标
stZoomRatio	局部放大窗口比例结构体

【差异说明】

芯片	描述
Hi3531/Hi3532 /Hi3521/Hi3520A	标清设备的局部放大窗口不小于 16x16（即宽高不小于 16）。
Hi3518/Hi3516C	标清设备的局部放大窗口不小于 32x32（即宽高不小于 32）。

【注意事项】

- 局部放大支持两种截取 VO 通道源数据的方式：按位置和按比例
- 按位置时，配置联合体中的 stZoomRect，要求参数为非负数，按 2 像素对齐。且 HD 设备局部放大窗口不小于 32x32（即宽高不小于 32）。
- 按比例时，配置联合体中的 stZoomRatio。参数范围要求[0, 1000]，1000 表示 1:1。将用户配置的比例与输入图像分辨率换算得到的局部放大窗口，可能出现不对齐的值，SDK 内部会按减小方向自动适配到 2 对齐。对于标清设备的输入图像是混合采集序列时，用户需保证大、小图像按比例换算成窗口后的视域要一致，否则图像会左右抖动。
- 按位置和按比例两种类型都是相对于输入图像数据而言，而不是屏幕坐标。
- 取消局部放大时，可以按位置方式设置 stZoomRect 为（0, 0, 0, 0），或按比例方式设置 stZoomRatio 为（0, 0, 0, 0）。

【相关数据类型及接口】

HI_MPI_VO_SetZoomInWindow

VO_CSC_S

【说明】

定义图像输出效果结构体。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef struct hiVO_CSC_S
{
    VO_CSC_MATRIX_E enCscMatrix;
    HI_U32 u32Luma;                 /* luminance:   0 ~ 100 */
    HI_U32 u32Contrast;             /* contrast :   0 ~ 100*/
    HI_U32 u32Hue;                  /* hue      :   0 ~ 100 */
    HI_U32 u32Satuature;            /* satuature:   0 ~ 100 */
} VO_CSC_S;
```

【成员】

成员名称	描述
enCscMatrix	CSC 矩阵选择。
u32Luma	亮度值。 取值范围：[0, 100]。
u32Contrast	对比度值。 取值范围：[0, 100]。
u32Hue	色调值。 取值范围：[0, 100]。
u32Satuature	饱和度值。 取值范围：[0, 100]。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VO_SetDevCSC

VO_VGA_PARAM_S

【说明】

定义 VGA 图像输出效果结构体。

【定义】

```
typedef struct hiVO_VGA_PARAM_S
{
    VO_CSC_MATRIX_E enCscMatrix;
    HI_U32 u32Luma;                 /* luminance:   0 ~ 100 */
    HI_U32 u32Contrast;             /* contrast :   0 ~ 100 */
    HI_U32 u32Hue;                  /* hue      :   0 ~ 100 */
    HI_U32 u32Satuature;            /* satuature:   0 ~ 100 */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32Gain;
} VO_VGA_PARAM_S;
```

【成员】

成员名称	描述
enCscMatrix	CSC 矩阵选择。
u32Luma	亮度值。 取值范围：[0, 100]。
u32Contrast	对比度值。 取值范围：[0, 100]。
u32Hue	色调值。 取值范围：[0, 100]。
u32Satuature	饱和度值。 取值范围：[0, 100]。
u32Gain	DAC 增益值。 取值范围：[0, 64)。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VO_SetVgaParam

VO_CSC_MATRIX_E

【说明】

定义 CSC 转换矩阵。

【定义】

```
typedef enum hiVO_CSC_MATRIX_E
{
    VO_CSC_MATRIX_IDENTITY = 0,
    VO_CSC_MATRIX_BT601_TO_BT709,
    VO_CSC_MATRIX_BT709_TO_BT601,
    VO_CSC_MATRIX_BT601_TO_RGB_PC,
    VO_CSC_MATRIX_BT709_TO_RGB_PC,
    VO_CSC_MATRIX_RGB_TO_BT601_PC,
    VO_CSC_MATRIX_RGB_TO_BT709_PC,
    VO_CSC_MATRIX_BUTT
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
} VO_CSC_MATRIX_E;
```

【成员】

成员名称	描述
VO_CSC_MATRIX_IDENTITY	单位转换矩阵。
VO_CSC_MATRIX_BT601_TO_BT709	BT.601 到 BT.709 色彩空间的 CSC 矩阵。
VO_CSC_MATRIX_BT709_TO_BT601	BT.709 到 BT.601 色彩空间的 CSC 矩阵。
VO_CSC_MATRIX_BT601_TO_RGB_PC	BT.601 到 RGB 色彩空间的 CSC 矩阵。
VO_CSC_MATRIX_BT709_TO_RGB_PC	BT.709 到 RGB 色彩空间的 CSC 矩阵。
VO_CSC_MATRIX_RGB_TO_BT601_PC	RGB 到 BT.601 色彩空间的 CSC 矩阵
VO_CSC_MATRIX_RGB_TO_BT709_PC	RGB 到 BT.709 色彩空间的 CSC 矩阵

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VO_SetDevCSC
- HI_MPI_VO_SetVgaParam

VO_WBC_ATTR_S

【说明】

定义视频回写属性。

【定义】

```
typedef struct hiVO_WBC_ATTR_S
{
    SIZE_S                stTargetSize;
    PIXEL_FORMAT_E        enPixelFormat;
    HI_U32                u32FrameRate;
    VO_WBC_DATASOURCE_E   enDataSource;
} VO_WBC_ATTR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stTargetSize	视频回写图像大小。最大为 720x576，最小为 32x32。 动态属性。
enPixelFormat	数据格式，SPYCbCr420 或者 SPYCbCr422。 动态属性。
u32FrameRate	帧率控制。 动态属性。
enDataSource	回写数据源，VO_WBC_DATASOURCE_MIXER 或者 VO_WBC_DATASOURCE_VIDEO。 暂不支持。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VO_SetWbcAttr

VO_WBC_MODE_E

【说明】

定义视频回写模式。

【定义】

```
typedef enum hiVO_WBC_MODE_E
{
    VO_WBC_MODE_NOMAL = 0,
    VO_WBC_MODE_DROP_REPEAT,
    VO_WBC_MODE_PROG_TO_INTL,
    VO_WBC_MODE_BUTT,
} VO_WBC_MODE_E;
```

【成员】

成员名称	描述
VO_WBC_MODE_NOMAL	视频回写普通模式。该模式下，仅根据设备帧率和回写帧率来控制采集图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VO_WBC_MODE_DROP_REPEAT	丢弃重复显示帧模式。该模式下，WBC 对高清设备 HD 重复显示的帧只回写一次，丢弃重复显示 2 次以上的帧中断，并根据视频层设置的显示帧率和 WBC 回写帧率进行帧率控制。 该模式仅在高清设备 HD 是逐行显示时才有效。
VO_WBC_MODE_PROG_TO_INTL	逐行显示图像转隔行图像模式。该模式下，WBC 会将 HD 逐行显示的相邻两帧处理成一帧隔行图像。丢弃重复显示 3 次及以上的帧中断。 该模式仅在高清设备 HD 是逐行显示时才有效。

【注意事项】

- WBC 回写模式设为 VO_WBC_MODE_PROG_TO_INTL 时，如果高清设备 HD 是 1080P@60fps 显示，VI 实际采集帧率为 30fps，WBC 会将 60 帧逐行显示的图像转为 30 帧隔行图像，使回写后的图像在监视器上达到较好的显示效果。因此，当 HD 是 1080P@30fps 显示时，该模式下 WBC 的回写帧率最大只能达到 15fps，且该模式下设置的帧率控制无效。在该模式下，WBC 实际回写帧率分以下几种情况：
  - 当 DispFrmRt <= 1/2 x DevFrmRt 时，WBC 实际回写帧率为 DispFrmRt。
  - 当 DispFrmRt == DevFrmRt 时，WBC 实际回写帧率为 1/2 x DevFrmRt。
  - 当 1/2 x DevFrmRt < DispFrmRt < DevFrmRt 时，WBC 实际回写帧率不固定，随 dispFrmRt 不同而变。
- 建议用户设置 DispFrmRt == 1/2 x DevFrmRt。
- WBC 回写模式设为 VO_WBC_MODE_PROG_TO_INTL 时，SPYCbCr422 格式的图像最大支持缩放 8 倍（不含 8 倍）缩小，SPYCbCr420 格式最大支持 4 倍（不含 4 倍）缩小。
- WBC 回写模式设为 VO_WBC_MODE_DROP_REPEAT 时，如果高清设备 HD 是 1080P@60fps 显示，VI 实际采集帧率为 30fps，WBC 只回写 VI 实际采集的 30 帧图像，丢弃 HD 设备上重复显示的帧图像，即该模式下回写帧率最大不会超过 VI 的实际采集帧率。例如，HD 是 1080P@60fps 显示，VI 实际采集帧率为 10fps，WBC 回写帧率最大只能达到 10fps。

【相关数据类型及接口】

HI_MPI_VO_SetWbcMode

VO_CAS_MODE_E

【说明】

定义级联的单路双路模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef enum hiVO_CAS_MODE_E
{
    VO_CAS_MODE_SINGLE = 0,
    VO_CAS_MODE_DUAL,
    VO_CAS_MODE_BUTT,
} VO_CAS_MODE_E;
```

【成员】

成员名称	描述
VO_CAS_MODE_SINGLE	级联单路模式。
VO_CAS_MODE_DUAL	级联双路模式。

【注意事项】

无。

【相关数据类型及接口】

无。

VO_CAS_RGN_E

【说明】

定义级联的区域类型。

【定义】

```
typedef enum hiVO_CAS_RGN_E
{
    VO_CAS_64_RGN = 0,
    VO_CAS_32_RGN,
    VO_CAS_RGN_BUTT,
} VO_CAS_RGN_E;
```

【成员】

成员名称	描述
VO_CAS_64_RGN	级联 64 区域类型。
VO_CAS_32_RGN	级联 32 区域类型。

【注意事项】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关数据类型及接口】

无。

VO_CAS_ATTR_S

【说明】

定义级联属性。

【定义】

```
typedef struct hiVO_CAS_ATTR_S
{
    HI_BOOL        bSlave;
    VO_CAS_RGN_E    enCasRgn;
    VO_CAS_MODE_E   enCasMode;
} VO_CAS_ATTR_S;
```

【成员】

成员名称	描述
bSlave	视频级联主从模式，静态属性。
enCasRgn	视频级联区域类型，静态属性。
enCasMode	视频级联单路双路模式，静态属性。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VO_SetCascadeAttr

4.5 错误码

视频输出 API 错误码如表 4-3 所示。

表4-3 视频输出 API 错误码

错误代码	宏定义	描述
0xA00F8001	HI_ERR_VO_INVALID_DEVID	设备 ID 超出合法范围
0xA00F8002	HI_ERR_VO_INVALID_CHNID	通道 ID 超出合法范围

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA00F8003	HI_ERR_VO_ILLEGAL_PARAM	参数超出合法范围
0xA00F8006	HI_ERR_VO_NULL_PTR	函数参数中有空指针
0xA00F8008	HI_ERR_VO_NOT_SUPPORT	不支持的操作
0xA00F8009	HI_ERR_VO_NOT_PERMIT	操作不允许
0xA00F800C	HI_ERR_VO_NO_MEM	内存不足
0xA00F8010	HI_ERR_VO_SYS_NOTREADY	系统未初始化
0xA00F8012	HI_ERR_VO_BUSY	资源忙
0xA00F8040	HI_ERR_VO_DEV_NOT_CONFIG	设备未配置
0xA00F8041	HI_ERR_VO_DEV_NOT_ENABLE	设备未使能
0xA00F8042	HI_ERR_VO_DEV_HAS_ENABLED	设备已使能
0xA00F8043	HI_ERR_VO_DEV_HAS_BINDED	设备已被绑定
0xA00F8044	HI_ERR_VO_DEV_NOT_BINDED	设备未被绑定
0xA00F8045	HI_ERR_VO_VIDEO_NOT_ENABLE	视频层未使能
0xA00F8046	HI_ERR_VO_VIDEO_NOT_DISABLE	视频层未禁止
0xA00F8047	HI_ERR_VO_VIDEO_NOT_CONFIG	视频层未配置
0xA00F8048	HI_ERR_VO_CHN_NOT_DISABLE	通道未禁止
0xA00F8049	HI_ERR_VO_CHN_NOT_ENABLE	通道未使能
0xA00F804A	HI_ERR_VO_CHN_NOT_CONFIG	通道未配置
0xA00F804B	HI_ERR_VO_CHN_NOT_ALLOC	通道未分配资源
0xA00F804C	HI_ERR_VO_INVALID_PATTERN	无效样式
0xA00F804D	HI_ERR_VO_INVALID_POSITION	无效级联位置
0xA00F804E	HI_ERR_VO_WAIT_TIMEOUT	等待超时
0xA00F804F	HI_ERR_VO_INVALID_VFRAME	无效视频帧
0xA00F8050	HI_ERR_VO_INVALID_RECT_PARA	无效矩形参数
0xA00F8051	HI_ERR_VO_SETBEGIN_ALREADY	BEGIN 已设置
0xA00F8052	HI_ERR_VO_SETBEGIN_NOTYET	BEGIN 没有设置
0xA00F8053	HI_ERR_VO_SETEND_ALREADY	END 已设置
0xA00F8054	HI_ERR_VO_SETEND_NOTYET	END 没有设置
0xA00F8055	HI_ERR_VO_GRP_INVALID_ID	同步组 ID 无效

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA00F8056	HI_ERR_VO_GRP_NOT_CREATE	同步组未创建
0xA00F8057	HI_ERR_VO_GRP_HAS_CREATED	同步组已经创建
0xA00F8058	HI_ERR_VO_GRP_NOT_DESTROY	同步组未销毁
0xA00F8059	HI_ERR_VO_GRP_CHN_FULL	同步组通道注册满
0xA00F805A	HI_ERR_VO_GRP_CHN_EMPTY	同步组无注册通道
0xA00F805B	HI_ERR_VO_GRP_CHN_NOT_EMPTY	同步组通道不为空
0xA00F805C	HI_ERR_VO_GRP_INVALID_SYN_MODE	无效同步模式
0xA00F805D	HI_ERR_VO_GRP_INVALID_BASE_PTS	无效基准时间戳
0xA00F805E	HI_ERR_VO_GRP_NOT_START	同步组未启动
0xA00F805F	HI_ERR_VO_GRP_NOT_STOP	同步组未停止
0xA00F8060	HI_ERR_VO_GRP_INVALID_FRMRATE	同步组无效帧率
0xA00F8061	HI_ERR_VO_GRP_CHN_HAS_REG	通道已经注册
0xA00F8062	HI_ERR_VO_GRP_CHN_NOT_REG	通道未注册
0xA00F8063	HI_ERR_VO_GRP_CHN_NOT_UNREG	通道未注销
0xA00F8064	HI_ERR_VO_GRP_BASE_NOT_CFG	基准时间戳未配置
0xA00F8065	HI_ERR_VO_GFX_NOT_DISABLE	图形层未关闭
0xA00F8066	HI_ERR_VO_GFX_NOT_BIND	图形层未绑定
0xA00F8067	HI_ERR_VO_GFX_NOT_UNBIND	图形层未解绑定
0xA00F8068	HI_ERR_VO_GFX_INVALID_ID	图形层 ID 超出范围
0xA00F8069	HI_ERR_VO_WBC_NOT_DISABLE	WBC 未禁止
0xA00F806A	HI_ERR_VO_WBC_NOT_CONFIG	WBC 未设置属性
0xA00F806B	HI_ERR_VO_CHN_AREA_OVERLAP	VO 通道区域重叠

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

5 视频处理子系统.....5-1
5.1 概述.....5-1
5.2 功能描述.....5-1
5.2.1 基本概念.....5-1
5.2.2 功能描述.....5-2
5.2.3 芯片差异.....5-3
5.3 API 参考.....5-8
5.4 数据类型.....5-60
5.5 错误码.....5-82

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 5-1 VPSS 上下文关系 5-3
图 5-2 Hi3531/Hi3532 VPSS 的数据流图 5-4
图 5-3 Hi3521/Hi3520A VPSS 的数据流图 5-4
图 5-4 Hi3518/Hi3516C VPSS 的数据流图 5-5
图 5-5 Hi3520D/Hi3515A/Hi3515C VPSS 的数据流图 5-5

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 5-1 Hi3531/Hi3532 VPSS 通道规格 5-5
表 5-2 Hi3521/Hi3520A VPSS 通道规格 5-6
表 5-3 Hi3518/Hi3516C VPSS 通道规格 5-7
表 5-4 Hi3520D/Hi3515A/Hi3515C VPSS 通道规格 5-7
表 5-5 VPSS API 错误码 5-82

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

5 视频处理子系统

5.1 概述

VPSS 是视频前处理单元，全称为 Video Process Sub-System。支持对一幅输入图像进行统一预处理，如去噪、去隔行等，然后再对各通道分别进行缩放、锐化等处理，最后输出多种不同分辨率的图像。

VPSS 单元支持的具体图像处理功能包括 PreScale、De-ring/De-block、NR、IE、DIE、Sharpen 等。

5.2 功能描述

5.2.1 基本概念

- GROUP

  VPSS 对用户提供组（GROUP）的概念。最大可用数为 128 个，各 GROUP 分时复用 VPSS 硬件。每个 VPSS GROUP 包含多个通道，通道数目视方案实现有所不同，具体描述参见 CHANNEL。

- CHANNEL

  VPSS 组的通道。通道分为 3 种：物理通道、直通通道和扩展通道。VPSS 硬件提供多个物理通道，每个通道有不同的功能（请参考图 5-1）与限制，此外提供 1 个直通通道（bypass），此通道只具备 CROP 功能，对于无需前处理的图像，可以经此通道透传给接收者。扩展通道具备缩放功能，它通过绑定物理通道，将物理通道输出作为自己的输入，把图像缩放成用户设置的目标分辨率输出，目前仅 Hi3518/Hi3516C 支持。不同芯片间通道规格略有差异，具体描述参见“5.2.3 芯片差异”。

- PreScale

  预缩放。针对输入图像大于预设值的情景，先缩放至预设值。

- Sizer

  筛选。VPSS 只筛选出特定尺寸的图像进行进一步处理。

- De-ring/De-block

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

去环效应/去块效应。去环效应也称为去振铃效应，补偿由于视频压缩所造成的低频分量损失，去除图像边缘的振铃；去块效应补偿由于视频按块压缩时的直流分量量化损失。

- DIE

  De-interlace，去隔行。将交错的隔行视频源还原成逐行视频源。

- NR

  Noise Reduce，去噪。通过参数配置，把图像中的高斯噪声去除，使得图像变得平滑，有助于降低编码码率。

- IE

  Image Enhance，图像增强。将辨别图像中的细节区域，把图像细节凸显，使得图像清晰，增加图像的对比度。

- Sharpen

  图像锐化。锐化图像的边缘和凸显图像细节，对经过缩放（scaler）后的图像进行频率补偿或增强，使得图像边缘锐利，轮廓清晰。

- FRC

  Frame Rate Control，帧率控制。

- 备份节点

  原始图像的备份节点。每个 GROUP 都有一个备份节点，用于备份即将提交硬件处理的那帧原始图像。VPSS 在以下情况会将缓存队列队头节点的图像放入备份节点：

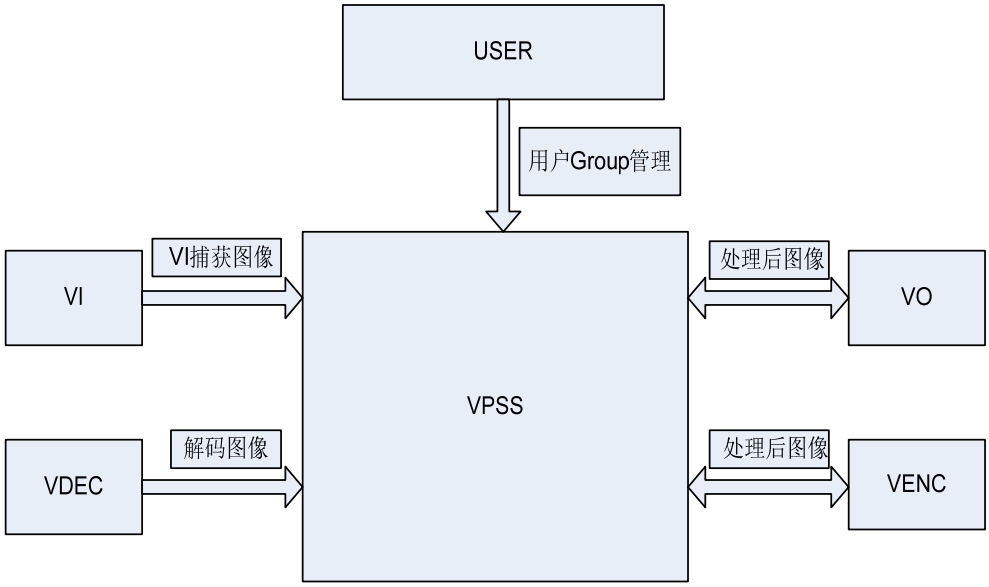
  - 当队头节点的图像要经过 VPSS 硬件处理时，VPSS 会将其放入备份节点，并替换掉原有图像。
  - 当后端绑定的接收模块要求 VPSS 将队头图像放入备份节点时，VPSS 也会替换备份节点中的图像，即使该图像不经过硬件处理。

5.2.2 功能描述

VPSS 在系统中位置如图 5-1 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图5-1 VPSS 上下文关系



通过调用 SYS 模块的绑定接口，可与 VI/VDEC 和 VO/VENC 等模块进行绑定，其中前者为 VPSS 的输入源，后者为 VPSS 的接收者。用户可通过提供的 MPI 接口对 GROUP 进行管理。每个 GROUP 仅可与一个输入源绑定。GROUP 的物理通道有两种工作模式：AUTO 和 USER，两种模式间可动态切换。默认的工作模式为 AUTO，此模式下各通道仅可与一个接收者绑定。若想使用 USER 模式，则需调用 MPI 接口进行设置，同时指定所需图像的大小和格式，此模式下各通道可与多个接收者绑定。需要**特别注意**的是，**USER 模式主要用于对同一通道图像进行多路编码的场景，此模式下播放控制不生效，因此预览和回放场景下不建议使用 USER 模式**。bypass 通道只能工作在 AUTO 模式，可与多个接收者绑定。

注意

当 VPSS 通道用 Auto 模式绑定编码模块时，如果 VPSS 接收到的源图像分辨率比编码图像分辨率小，则：编码模块将不对图像进行编码，即使 VPSS 通道具有图像放大功能。此时，如果用户想对小图像进行放大编码，则需将 VPSS 通道设为 User 模式。

5.2.3 芯片差异

Hi3531/Hi3532 和 Hi3521/Hi3520A 在数据处理流程上略有区别，两者之间的差异如图 5-2 和图 5-3 所示。此外，与 Hi3531/Hi3532 相比，Hi3521/Hi3520A 增加 PreScl 功能，删除 Dr/Db，同时 Hi3520D/Hi3515A/Hi3515C 和 Hi3521 在 Prescale 前增加 Sizer 和 FRC 功能，如数据流图 5-5 所示。注意，使用 Hi3531/Hi3532 时，Dr/Db 与 NR、DIE 不能同时使能，建议处理解码图像时开启 Dr/Db，处理 VI 采集图像时酌情开启 NR、DIE 功能。通道规格上，Hi3521/Hi3520A 的 chn0/chn1 增加了图像的单双场选择功能，但是 chn0 删除了 sp 功能，另外，4 个物理通道可分别设置 NR 强度。
Hi3520D/Hi3515A 在通道规格上有了进一步的优化，4 路物理通道完全相同，都具有 Scale 功能和 sharpen 功能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

与 Hi3521/Hi3520A/Hi3515C 相比，Hi3518/Hi3516C 的功能进一步精简，删除了 IE 和 Prescale 功能。数据流如图 5-4 所示。输入图像经过 NR 和 DIE，然后分成 2 路物理通道分别进行 Scale、Sharpen 功能模块处理，输出目标图像。**需要注意的是：Hi3518**/Hi3516C **在通道规格上与 Hi3521/Hi3520A 相比，没有单双场选择。另外，物理通道 0 和物理通道 1 的能力不对称，通道 0 没有 Scale 功能。**此外，Hi3518/Hi3516C 提供 5 路扩展通道以提供更好的灵活性，扩展通道具备缩放和帧率控制功能，不经过其他硬件算法单元处理。扩展通道和物理通道的绑定方式不限于图 5-4 的形式，用户可根据需要选择合适的绑定关系。

图5-2 Hi3531/Hi3532 VPSS 的数据流图

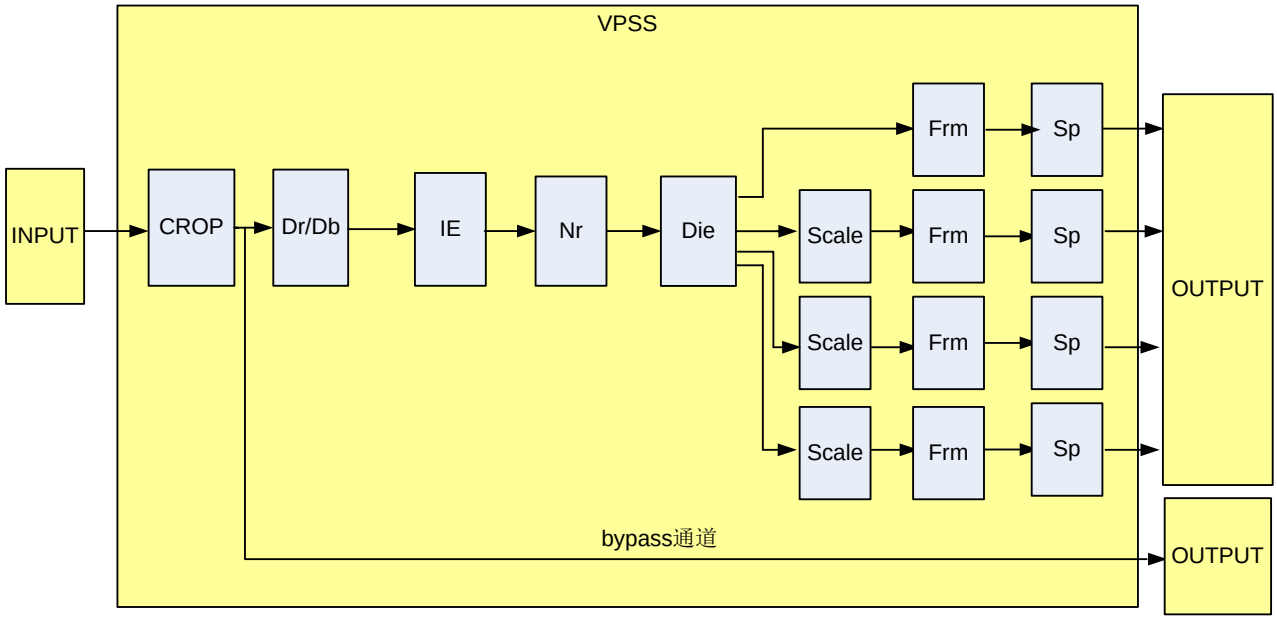
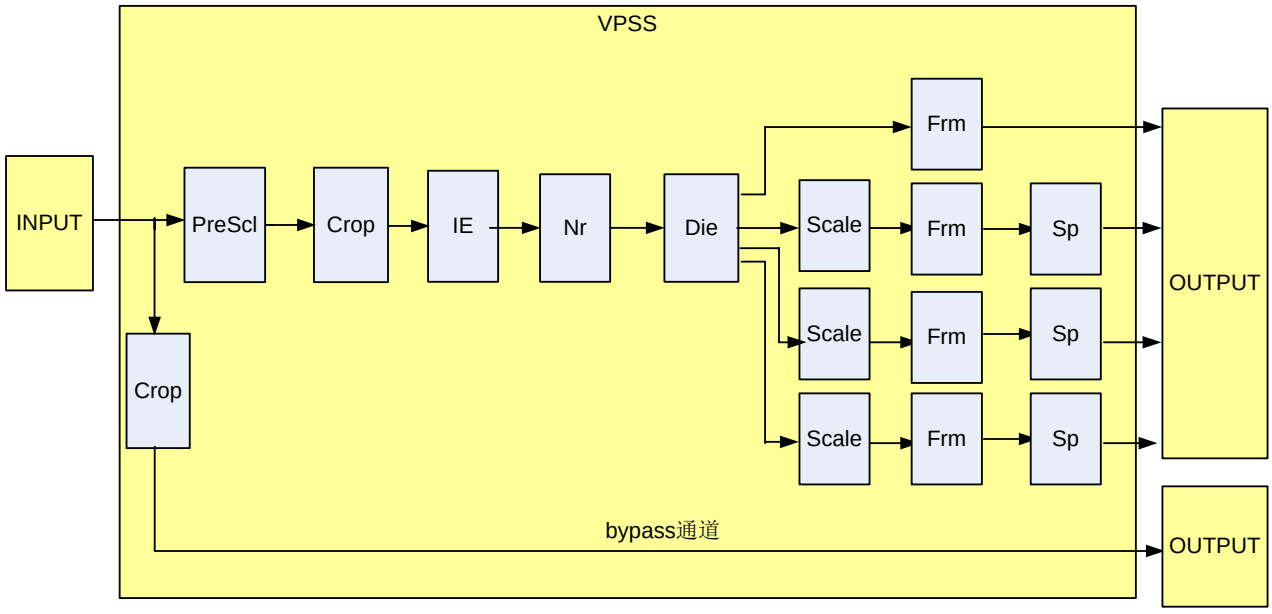


图5-3 Hi3521/Hi3520A VPSS 的数据流图



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图5-4 Hi3518/Hi3516C VPSS 的数据流图

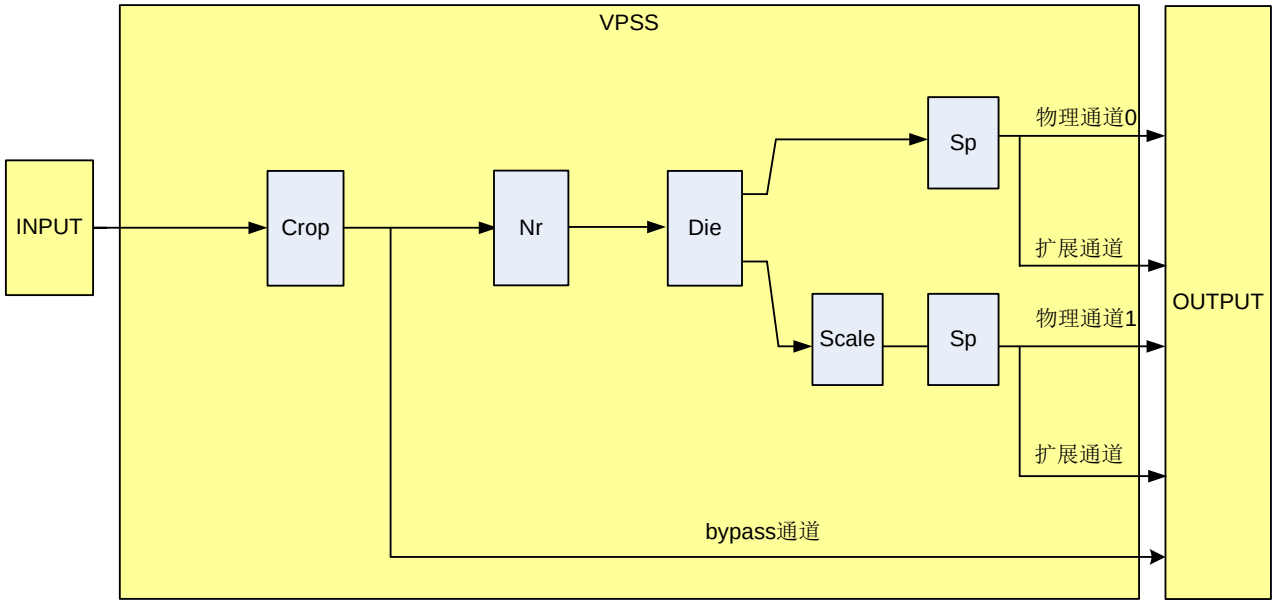


图5-5 Hi3520D/Hi3515A/Hi3515C VPSS 的数据流图

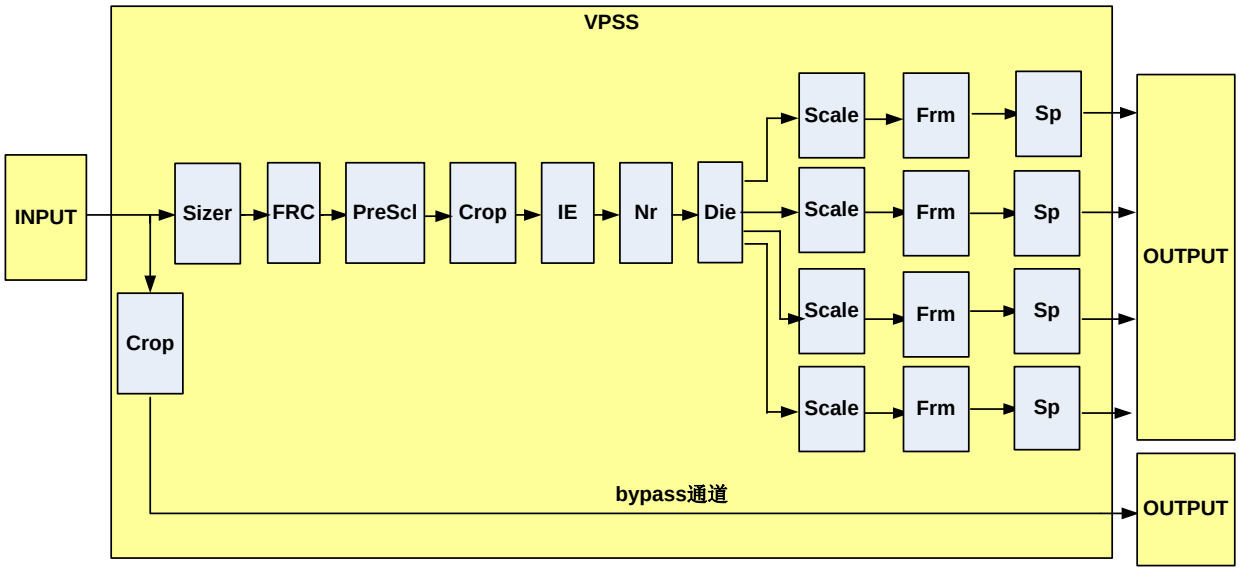


表5-1 Hi3531/Hi3532 VPSS 通道规格

通道号	通道名	缩放能力	锐化	抽场	建议应用场景
0	通道 0	不具备缩放能力	支持	不支持	大码流编码、JPEG 抓拍

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

通道号	通道名	缩放能力	锐化	抽场	建议应用场景
1	通道 1	水平、垂直均最大支持 16 倍缩小 水平、垂直均最大支持 16 倍放大	支持	不支持	小码流编码
2	通道 2	水平、垂直均最大支持 16 倍缩小 水平、垂直均最大支持 16 倍放大 支持帧场转换	支持	不支持	高清预览
3	通道 3	水平、垂直均最大支持 16 倍缩小 水平、垂直均最大支持 16 倍放大	支持	不支持	高清预览
4	Bypass 通道	不具备缩放能力	不支持	不支持	用于输出 CVBS、MD、IVE 所需的 D1 图像

表5-2 Hi3521/Hi3520A VPSS 通道规格

通道号	通道名	缩放能力	锐化	抽场	建议应用场景
0	通道 0	无缩放能力。	不支持	支持	大码流编码、JPEG 抓拍
1	通道 1	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大	支持	支持	小码流编码
2	通道 2	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大 支持帧场转换	支持	不支持	高清预览
3	通道 3	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大	支持	不支持	高清预览
4	Bypass 通道	不具备缩放能力	不支持	不支持	用于输出 CVBS、MD、IVE 所需的 D1 图像

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表5-3 Hi3518/Hi3516C VPSS 通道规格

通道号	通道名	缩放能力	锐化	抽场	建议应用场景
0	通道 0	无缩放能力	支持	不支持	大码流编码、JPEG 抓拍
1	通道 1	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 8 倍放大	支持	不支持	小码流编码，CVBS 预览
2	Bypass 通道	不具备缩放能力	不支持	不支持	用于输出 VI 同源图像
3	扩展通道 0	水平、垂直均最大支持 15 倍缩小 放大无限制	不支持	不支持	小码流编码，或者输出 MD、IVE 所需的目标图像
4	扩展通道 1	水平、垂直均最大支持 15 倍缩小 放大无限制	不支持	不支持	小码流编码，或者输出 MD、IVE 所需的目标图像
5	扩展通道 2	水平、垂直均最大支持 15 倍缩小 放大无限制	不支持	不支持	小码流编码，或者输出 MD，IVE 所需的目标图像
6	扩展通道 3	水平、垂直均最大支持 15 倍缩放 放大无限制	不支持	不支持	小码流编码，或者输出 MD，IVE 所需的目标图像
7	扩展通道 4	水平、垂直均最大支持 15 倍缩放 放大无限制	不支持	不支持	小码流编码，或者输出 MD，IVE 所需的目标图像

对于 Hi3518/Hi3516C 的扩展通道（包括 0、1、2、3 和 4），当输入数据格式设置为 YUV422SP，即其绑定的物理通道输出为 YUV422SP，同时输出数据格式设置为 YUV420SP 时，扩展通道垂直方向上仅支持 8 倍缩小。

表5-4 Hi3520D/Hi3515A/Hi3515C VPSS 通道规格

通道号	通道名	缩放能力	锐化	抽场	建议应用场景
0	通道 0	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大	支持	不支持	大码流编码、JPEG 抓拍

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

通道号	通道名	缩放能力	锐化	抽场	建议应用场景
1	通道 1	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大	支持	不支持	小码流编码
2	通道 2	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大 支持帧场转换	支持	不支持	高清预览
3	通道 3	水平、垂直均最大支持 8 倍缩小 水平、垂直均最大支持 16 倍放大	支持	不支持	高清预览
4	Bypass 通道	不具备缩放能力	不支持	不支持	用于输出 CVBS、MD、IVE 所需的 D1 图像

【注意】

对于通道 0、1、2、3，“应用场景”一栏为**建议**的使用方案，实际上用户可以根据自身需要和通道的功能来自由分配各个通道的用途。**但应注意只有通道 2 具有帧场转换能力，并且使用此功能后，通道 3 将不会有输出。**

5.3 API 参考

该功能模块为用户提供以下 MPI：

- HI_MPI_VPSS_CreateGrp：创建一个 VPSS GROUP。
- HI_MPI_VPSS_DestroyGrp：销毁一个 VPSS GROUP。
- HI_MPI_VPSS_GetGrpAttr：获取 VPSS GROUP 属性。
- HI_MPI_VPSS_SetGrpAttr：设置 VPSS GROUP 属性。
- HI_MPI_VPSS_StartGrp：启用 VPSS GROUP。
- HI_MPI_VPSS_StopGrp：禁用 VPSS GROUP。
- HI_MPI_VPSS_ResetGrp：重置一个 VPSS。
- HI_MPI_VPSS_EnableChn：启用 VPSS 通道。
- HI_MPI_VPSS_DisableChn：禁用 VPSS 通道。
- HI_MPI_VPSS_EnableBackupFrame：启用 backup 帧。
- HI_MPI_VPSS_DisableBackupFrame：禁用 backup 帧。
- HI_MPI_VPSS_GetChnAttr：获取 VPSS 通道属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VPSS_SetChnAttr：设置 VPSS 通道属性。
- HI_MPI_VPSS_SetGrpParam：设置 VPSS GRUOP 高级属性。
- HI_MPI_VPSS_GetGrpParam：获取 VPSS GROUP 高级属性。
- HI_MPI_VPSS_SetCropCfg：设置 VPSS CROP 功能属性。
- HI_MPI_VPSS_GetCropCfg：获取 VPSS CROP 功能属性。
- HI_MPI_VPSS_UserSendFrame：用户向 VPSS GROUP 发送数据。
- HI_MPI_VPSS_UserSendFrameTimeout：用户向 VPSS 阻塞发送数据。
- HI_MPI_VPSS_SetChnMode：设置 VPSS 通道工作模式。
- HI_MPI_VPSS_GetChnMode：获取 VPSS 通道工作模式。
- HI_MPI_VPSS_SetDepth：设置用户图像队列深度。
- HI_MPI_VPSS_GetDepth：获取用户图像队列深度。
- HI_MPI_VPSS_UserGetFrame：用户获取一帧图像。
- HI_MPI_VPSS_UserReleaseFrame：用户释放一帧图像。
- HI_MPI_VPSS_UserGetGrpFrame：用户从 Group 获取一帧原始图像。
- HI_MPI_VPSS_UserReleaseGrpFrame：用户释放一帧图像。
- HI_MPI_VPSS_SetChnNrParam：设置通道 NR 高级参数。
- HI_MPI_VPSS_GetChnNrParam：获取通道 NR 高级参数。
- HI_MPI_VPSS_SetChnSpParam：设置通道 SP 高级参数。
- HI_MPI_VPSS_GetChnSpParam：获取通道 SP 高级参数。
- HI_MPI_VPSS_SetPreScale：设置 VPSS 预缩放属性。
- HI_MPI_VPSS_GetPreScale：获取 VPSS 预缩放属性。
- HI_MPI_VPSS_SetChnField：设置通道抽场属性。
- HI_MPI_VPSS_GetChnField：获取通道抽场属性。
- HI_MPI_VPSS_SetGrpField：设置组抽场属性。
- HI_MPI_VPSS_GetGrpField：获取组抽场属性。
- HI_MPI_VPSS_SetGrpSizer：设置 VPSS 尺寸筛选信息。
- HI_MPI_VPSS_GetGrpSizer：获取 VPSS 尺寸筛选信息。
- HI_MPI_VPSS_SetGrpFrameRate：设置 VPSS 帧率控制信息。
- HI_MPI_VPSS_GetGrpFrameRate：获取 VPSS 帧率控制信息。
- HI_MPI_VPSS_SetDelay：设置延迟启动 VPSS 处理的延时队列长度。
- HI_MPI_VPSS_GetDelay：获取延迟启动 VPSS 处理的延时队列长度。
- HI_MPI_VPSS_SetExtChnAttr：设置 VPSS Grp 扩展通道属性。
- HI_MPI_VPSS_GetExtChnAttr：获取 VPSS Grp 扩展通道属性。

HI_MPI_VPSS_CreateGrp

【描述】

创建一个 VPSS GROUP。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VPSS_CreateGrp(VPSS_GRP VpssGrp, VPSS_GRP_ATTR_S
*pstVpssGrpAttr)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssGrpAttr	VPSS GROUP 属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- 可创建多个 GROUP，最大 GROUP 数为 VPSS_MAX_GRP_NUM。
- 不支持重复创建。
- GROUP 属性必须合法，具体参见 VPSS_GRP_ATTR_S。
- 假设[w, 64]是 w 按照 64 对齐之后的长度，则对于宽为 W，高为 H 的 VPSS GROUP 来说，只要输入的图像[w, 64] x h 不超过[W, 64] x H，VPSS 都能处理这样的图像。
- VPSS 如果开启 IE 或者缩放功能，输出图像的像素值会被钳位在[16, 235]之间。

【举例】

```
VPSS_GRP_ATTR_S stGrpVpssAttr;
VPSS_CHN_ATTR_S stChnAttr;
VPSS_CROP_INFO_S stCropInfo;
HI_S32 s32Ret = HI_SUCCESS;
VPSS_GRP VpssGrp;
VPSS_CHN VpssChn

VpssGrp = 0;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VpssChn = 0;

    stGrpVpssAttr.u32MaxW = 720;
    stGrpVpssAttr.u32MaxH = 576;
    stGrpVpssAttr.bDrEn = HI_FALSE;
    stGrpVpssAttr.bDbEn = HI_FALSE;
    stGrpVpssAttr.bIeEn = HI_FALSE;
    stGrpVpssAttr.bNrEn = HI_FALSE;
    stGrpVpssAttr.bHistEn = HI_FALSE;
    stGrpVpssAttr.enDieMode = VPSS_DIE_MODE_NODIE;
    stGrpVpssAttr.enPixFmt = PIXEL_FORMAT_YUV_SEMIPLANAR_422;

    s32Ret = HI_MPI_VPSS_CreateGrp(VpssGrp, &stGrpVpssAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_GetGrpAttr(VpssGrp, &stGrpVpssAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    stGrpVpssAttr.bIeEn = HI_TRUE;
    stGrpVpssAttr.bNrEn = HI_TRUE;
    s32Ret = HI_MPI_VPSS_SetGrpAttr(VpssGrp, &stGrpVpssAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_GetCropCfg(VpssGrp, &stCropInfo);
    if(s32Ret != HI_SUCCESS)
{
return s32Ret;
}
stCropInfo.bEnable = 1;
stCropInfo.enCropCoordinate = VPSS_CROP_ABS_COOR;
stCropInfo.stCropRect.s32X = 180;
stCropInfo.stCropRect.s32Y = 252;
stCropInfo.stCropRect.u32Width = 1920;
stCropInfo.stCropRect.u32Height = 1080;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
s32Ret = HI_MPI_VPSS_SetCropCfg(VpssGrp, &stCropInfo);
if(s32Ret != HI_SUCCESS)
    {
            return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_GetChnAttr(VpssGrp, VpssChn,&stChnAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    stChnAttr.bFrameEn = 0;
    stChnAttr.bSpEn = 1;
    s32Ret = HI_MPI_VPSS_SetChnAttr(VpssGrp, VpssChn,&stChnAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_EnableChn(VpssGrp, VpssChn);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_StartGrp (VpssGrp);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    /*****************************/
    /*     call sys bind interface    */
    /*****************************/

    s32Ret = HI_MPI_VPSS_StopGrp (VpssGrp);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_DisableChn(VpssGrp, VpssChn);
    if(s32Ret != HI_SUCCESS)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    return s32Ret;
}

/*****************************/
/*     call sys unbind interface  */
/*****************************/

s32Ret = HI_MPI_VPSS_DestroyGrp(VpssGrp);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
```

【相关主题】

HI_MPI_VPSS_DestroyGrp

HI_MPI_VPSS_DestroyGrp

【描述】

销毁一个 VPSS GROUP。

【语法】

```
HI_S32 HI_MPI_VPSS_DestroyGrp(VPSS_GRP VpssGrp)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 调用此接口之前，必须先调用 HI_MPI_VPSS_StopGrp 禁用此 GROUP。
- 调用此接口时，会一直等待此 GROUP 当前任务处理结束才会真正销毁。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_CreateGrp

HI_MPI_VPSS_GetGrpAttr

【描述】

获取 VPSS GROUP 属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetGrpAttr (VPSS_GRP VpssGrp, VPSS_GRP_ATTR_S
*pstVpssGrpAttr)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssGrpAttr	VPSS GROUP 属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- GROUP 必须已创建。
- GROUP 属性必须合法，其中部分静态属性不可动态设置，具体参见 VPSS_GRP_ATTR_S。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_SetGrpAttr

HI_MPI_VPSS_SetGrpAttr

【描述】

设置 VPSS GROUP 属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetGrpAttr (VPSS_GRP VpssGrp, VPSS_GRP_ATTR_S
*pstVpssGrpAttr)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssGrpAttr	VPSS GROUP 属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- GROUP 属性必须合法，其中部分静态属性不可动态设置，具体参见 VPSS_GRP_ATTR_S。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_GetGrpAttr

HI_MPI_VPSS_StartGrp

【描述】

启用 VPSS GROUP。

【语法】

```
HI_S32 HI_MPI_VPSS_StartGrp(VPSS_GRP VpssGrp)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_StopGrp

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VPSS_StopGrp

【描述】

禁用 VPSS GROUP。

【语法】

```
HI_S32 HI_MPI_VPSS_StopGrp(VPSS_GRP VpssGrp)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_StartGrp

HI_MPI_VPSS_ResetGrp

【描述】

复位 VPSS GROUP。

【语法】

```
HI_S32 HI_MPI_VPSS_ResetGrp (VPSS_GRP VpssGrp)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

无。

HI_MPI_VPSS_GetChnAttr

【描述】

获取 VPSS 通道属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetChnAttr(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_ATTR_S *pstChnAttr)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstChnAttr	VPSS 通道属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法[0, VPSS_MAX_CHN_NUM)。
- 直通通道、扩展通道不支持此接口，对其调用无效，但不会报错。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_SetChnAttr

HI_MPI_VPSS_SetChnAttr

【描述】

设置 VPSS 通道属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetChnAttr(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_ATTR_S *pstChnAttr)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstChnAttr	VPSS 通道属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法[0, VPSS_MAX_CHN_NUM)。
- 直通通道、扩展通道不支持此接口，对其调用无效，但不会报错。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_GetChnAttr

HI_MPI_VPSS_EnableChn

【描述】

启用 VPSS 通道。

【语法】

```
HI_S32 HI_MPI_VPSS_EnableChn(VPSS_GRP VpssGrp, VPSS_CHN VpssChn)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法[0, VPSS_MAX_CHN_NUM)。
- 扩展通道必须保证此通道绑定的源物理通道已经使能，否则返回失败错误码。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_DisableChn

HI_MPI_VPSS_DisableChn

【描述】

禁用 VPSS 通道。

【语法】

```
HI_S32 HI_MPI_VPSS_DisableChn(VPSS_GRP VpssGrp, VPSS_CHN VpssChn)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法[0, VPSS_MAX_CHN_NUM)。
- 若支持扩展通道，必须保证此通道绑定的目的扩展通道已经全部禁用，否则将会返回失败的错误码。

【举例】

参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

HI_MPI_VPSS_EnableChn

HI_MPI_VPSS_EnableBackupFrame

【描述】

启用 backup 帧。

【语法】

```
HI_S32 HI_MPI_VPSS_EnableBackupFrame(VPSS_GRP VpssGrp)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 必须已创建。
- VPSS 默认启用此功能。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_DisableBackupFrame

【描述】

禁用 backup 帧。

【语法】

```
HI_S32 HI_MPI_VPSS_DisableBackupFrame(VPSS_GRP VpssGrp)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 必须已创建。
- 禁用 backup 帧后，每个 VPSS 组可以节省一帧的内存，但会影响 VO 的画面切换、播放控制等功能，应慎用此功能。
- 如果可用内存比较紧张，各模块按最小需求分配，例如其前端的解码通道的显示帧（DisplayFrameNum）设置为 0 的情况下，此接口应在 HI_MPI_VPSS_StartGrp 之前调用。其余情况可动态调用。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetGrpParam

【描述】

设置 VPSS GRUOP 高级属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetGrpParam(VPSS_GRP VpssGrp, VPSS_GRP_PARAM_S
*pstVpssParam)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssParam	高级属性设置	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- pstVpssParam 各成员的限制参见 VPSS_GRP_PARAM_S 结构说明。
- 对于 Hi3521/Hi3520A，调用此接口不会改变 Nr 参数。Hi3521/Hi3520A 修改 Nr 参数应该调用 HI_MPI_VPSS_SetChnNrParam( )。
- 对于 Hi3520Dv200，对接 AHD 前端时，噪声会比较严重，并且有条状的噪声上下跳动。此时仅仅增大 sf 是不够的，sf 可以去掉底噪，但对于跳动的“条状”噪声是不起作用的，因此 tf 也要相应提高。应当注意的是，tf 过高会对运动场景有副作用，造成“拖尾”。推荐值 sf 为 32，tf 为 10。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_GetGrpParam

【描述】

获取 VPSS GRUOP 高级属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetGrpParam(VPSS_GRP VpssGrp, VPSS_GRP_PARAM_S
*pstVpssParam)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssParam	高级属性。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法[0, VPSS_MAX_CHN_NUM)。

【举例】

无。

【相关主题】

HI_MPI_VPSS_SetCropCfg

HI_MPI_VPSS_SetCropCfg

【描述】

设置 VPSS CROP 功能属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetCropCfg(VPSS_GRP VpssGrp, VPSS_CROP_INFO_S
*pstCropInfo)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstCropInfo	CROP 功能参数	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 当起始点坐标类型为 VPSS_CROP_RITIO_COOR（相对坐标）时，s32X、s32Y 的合法取值范围为[0,999]。
- 当起始点坐标类型为 VPSS_CROP_ABS_COOR（绝对坐标）时，s32X、s32Y 的合法取值范围为[0，VPSS_MAX_IMAGE_WIDTH]，裁剪宽、高的合法范围为[VPSS_MIN_IMAGE_WIDTH, VPSS_MAX_IMAGE_WIDTH]。如果设置的起始坐标超出实际图像的有效范围，则不进行裁剪处理；如果设置的裁剪宽高超出实际图像的有效范围，则自动裁剪到图像有效范围之内。
- CROP 区域的最大、最小尺寸请参见 VPSS_GRP_ATTR_S 结构体成员 u32MaxW 和 u32MaxH 的取值范围。
- CROP 区域的起始点坐标会自动向下圆整到 4 的整数倍，宽高会自动向下圆整到 16 的整数倍。
- CROP 区域的抽场功能仅对隔行图像有效，对于帧图像无效。CROP 区域是相对于原始输入的整帧图像，如果开启预缩放，CROP 区域是相对于预缩放后的图像大小；对于 bypass 通道，无论是否开启预缩放，CROP 区域都是相对于原始输入的图像大小。

【举例】

请参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

无。

HI_MPI_VPSS_GetCropCfg

【描述】

获取 VPSS CROP 功能属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetCropCfg(VPSS_GRP VpssGrp, VPSS_CROP_INFO_S
*pstCropInfo)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstCropInfo	CROP 功能参数	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

无。

【举例】

请参见 HI_MPI_VPSS_CreateGrp。

【相关主题】

无。

HI_MPI_VPSS_UserSendFrame

【描述】

用户向 VPSS 发送数据。

【语法】

```
HI_S32 HI_MPI_VPSS_UserSendFrame(VPSS_GRP VpssGrp, VIDEO_FRAME_INFO_S
*pstVideoFrame)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstVideoFrame	待发送的图像信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。
- GROUP 必须已创建。
- 用户使用此接口时，应自行进行帧率控制。
- 使用此接口前，VPSS 应正常启动，并与后端接收者绑定。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_UserSendFrameTimeout

【描述】

用户向 VPSS 阻塞发送数据。

【语法】

```
HI_S32 HI_MPI_VPSS_UserSendFrameTimeout(VPSS_GRP VpssGrp,
VIDEO_FRAME_INFO_S *pstVideoFrame, HI_U32 u32MilliSec)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstVideoFrame	待发送的图像信息。	输入
u32MilliSec	超时时间，0 表示不超时。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。
- GROUP 必须已创建。
- 用户使用此接口时，应自行进行帧率控制。
- 使用此接口前，VPSS 应正常启动，并与后端接收者绑定。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetChnMode

【描述】

设置 VPSS 通道工作模式。

【语法】

```
HI_S32 HI_MPI_VPSS_SetChnMode(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_MODE_S *pstVpssMode)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssChn	VPSS 通道号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstVpssMode	通道工作模式信息，具体限制参见 VPSS_CHN_MODE_S 结构。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。
- GROUP 必须已创建。
- CHN 号必须在取值范围内。
- 默认为 AUTO 模式，工作时，可在两种模式间动态切换。
- 直通通道、扩展通道不支持工作模式设置，对其调用接口无效，但不会报错。
- VPSS 如果开启缩放功能，输出图像的像素值会被钳位在[16, 235]之间。

【举例】

```
VPSS_GRP_ATTR_S stGrpVpssAttr;
  VPSS_CHN_ATTR_S stChnAttr;
  VIDEO_FRAME_INFO_S stFrame;
  HI_S32 s32Ret = HI_SUCCESS;
  VPSS_CHN_MODE_S stVpssMode;
  HI_U32 u32Depth = 8;
  VPSS_GRP VpssGrp;
  VPSS_CHN VpssChn

  VpssGrp = 0;
  VpssChn = 0;

  stGrpVpssAttr.u32MaxW = 720;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    stGrpVpssAttr.u32MaxH = 576;
    stGrpVpssAttr.bDrEn = HI_FALSE;
    stGrpVpssAttr.bDbEn = HI_FALSE;
    stGrpVpssAttr.bIeEn = HI_FALSE;
    stGrpVpssAttr.bNrEn = HI_FALSE;
    stGrpVpssAttr.bHistEn = HI_FALSE;
    stGrpVpssAttr.enDieMode = VPSS_DIE_MODE_NODIE;
    stGrpVpssAttr.enPixFmt = PIXEL_FORMAT_YUV_SEMIPLANAR_422;

    s32Ret = HI_MPI_VPSS_CreateGrp(VpssGrp, &stGrpVpssAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    s32Ret = HI_MPI_VPSS_GetGrpAttr(VpssGrp, &stGrpVpssAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }

    stGrpVpssAttr.bIeEn = HI_TRUE;
    stGrpVpssAttr.bNrEn = HI_TRUE;
    s32Ret = HI_MPI_VPSS_SetGrpAttr(VpssGrp, &stGrpVpssAttr);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }
     s32Ret = HI_MPI_VPSS_GetChnMode(VpssGrp,VpssChn,&stVpssMode);
if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }
     stVpssMode.enChnMode = VPSS_CHN_MODE_USER;
     stVpssMode.enPixelFormat = PIXEL_FORMAT_YUV_SEMIPLANAR_420;
     stVpssMode.u32Width = 720;
     stVpssMode.u32Height = 576;
     s32Ret = HI_MPI_VPSS_SetChnMode(VpssGrp,VpssChn,&stVpssMode);
if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }
HI_MPI_VPSS_SetDepth(VpssGrp,VpssChn,&u32Depth);
HI_MPI_VPSS_EnableChn(VpssGrp, VpssChn);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_MPI_VPSS_StartGrp(VpssGrp);
s32Ret = HI_MPI_VPSS_UserGetFrame(VpssGrp, VpssChn, &stFrame);
if(s32Ret == HI_SUCCESS)
{
HI_MPI_VPSS_UserRlsFrame(VpssGrp, VpssChn, &stFrame);
}
```

【相关主题】

无。

HI_MPI_VPSS_GetChnMode

【描述】

获取 VPSS 通道工作模式。

【语法】

```
HI_S32 HI_MPI_VPSS_GetChnMode(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_MODE_S *pstVpssMode)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 通道号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstVpssMode	通道工作模式信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- GROUP 必须已创建。
- CHN 号必须在取值范围内。
- 直通通道、扩展通道不支持工作模式设置，对其调用接口无效，但不会报错。

【举例】

请参见 HI_MPI_VPSS_SetChnMode。

【相关主题】

HI_MPI_VPSS_SetChnMode

HI_MPI_VPSS_SetDepth

【描述】

设置用户图像队列深度。

【语法】

```
HI_S32 HI_MPI_VPSS_SetDepth(VPSS_GRP VpssGrp, VPSS_CHN VpssChn, HI_U32 u32Depth)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 通道号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
u32Depth	队列深度。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- GROUP 必须已创建。
- CHN 号必须在取值范围内。
- 默认的用户队列深度为 0，最大深度为 8。

【举例】

请参见 HI_MPI_VPSS_SetChnMode。

【相关主题】

无。

HI_MPI_VPSS_GetDepth

【描述】

获取用户图像队列深度。

【语法】

```
HI_S32 HI_MPI_VPSS_GetDepth(VPSS_GRP VpssGrp, VPSS_CHN VpssChn, HI_U32 *pu32Depth)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 通道号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pu32Depth	队列深度。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- GROUP 必须已创建。
- CHN 号必须在取值范围内。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_UserGetFrame

【描述】

用户从通道获取一帧处理完成的图像。

【语法】

```
HI_S32 HI_MPI_VPSS_UserGetFrame(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VIDEO_FRAME_INFO_S *pstVideoFrame)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 通道号。 取值范围：[0, VPSS_MAX_PHY_CHN_NUM)。	输入
pstVideoFrame	图像信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。
- GROUP 必须已创建。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 该接口仅适用于物理通道。
- 只有在 USER 模式下，并且队列深度不为 0，才能获取到图像。

【举例】

请参见 HI_MPI_VPSS_SetChnMode。

【相关主题】

无。

HI_MPI_VPSS_UserReleaseFrame

【描述】

用户释放一帧图像。

【语法】

```
HI_S32 HI_MPI_VPSS_UserReleaseFrame (VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VIDEO_FRAME_INFO_S *pstVideoFrame)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 通道号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstVideoFrame	图像信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

实际上，此接口的前两个参数并无实际用途，可在取值范围内任意设置

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

请参见 HI_MPI_VPSS_SetChnMode。

【相关主题】

无。

HI_MPI_VPSS_UserGetGrpFrame

【描述】

用户从 Group 获取一帧原始图像。主要应用场景：高清设备解码回放，要求暂停、步进时，PIP 层和普通视频层上的两个通道显示同一帧图像。通过本接口和 HI_MPI_VPSS_UserSendFrame 等接口的配合使用，可实现该功能。

【语法】

```
HI_S32 HI_MPI_VPSS_UserGetGrpFrame(VPSS_GRP VpssGrp, VIDEO_FRAME_INFO_S
*pstVideoFrame, HI_U32 u32FrameIndex)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVideoFrame	图像信息。	输出
u32FrameIndex	图像索引号。目前无效，赋 0	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须在取值范围内。
- GROUP 必须已创建。
- 调用该函数可获取到 GROUP 中备份节点的图像，该图像为原始图像。VPSS 在以下情况会将缓存队列队头节点的图像放入备份节点：

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

  - 当队头节点的图像要经过 VPSS 硬件处理时，VPSS 会将其放入备份节点，并替换原有图像。
  - 当后端绑定的接收模块要求 VPSS 将队头图像放入备份节点时，VPSS 也会替换备份节点中的图像，即使该图像不经过硬件处理。
- 当备份节点为空时，调用该函数将返回 HI_ERR_VPSS_NOTREADY 错误码。
- 获取的图像要及时释放，否则将造成 VB 不够或解码回放时停止，建议与 5.3 HI_MPI_VPSS_UserReleaseGrpFrame 接口配对使用。

【举例】

参考发布包中 vdec 的 sample。

【相关主题】

无。

HI_MPI_VPSS_UserReleaseGrpFrame

【描述】

用户释放一帧图像。

【语法】

```
HI_S32 HI_MPI_VPSS_UserReleaseGrpFrame(VPSS_GRP VpssGrp,
VIDEO_FRAME_INFO_S *pstVideoFrame)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVideoFrame	图像信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

实际上，此接口的 VpssGrp 参数并无实际用途，可在取值范围内任意设置。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetChnNrParam

【描述】

设置通道 Nr 高级属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetChnNrParam(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_NR_PARAM_S *pstChnNrParam)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstChnNrParam	Nr 高级属性设置	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	0/1 通道共用一套 Nr 系数，因此调用此接口改变 0 通道 Nr 配置，1 通道也会随之改变；2/3 通道亦是如此。
Hi3518/Hi3516C	不支持

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3520D/Hi3515A/Hi3515C	不支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法，[0, VPSS_MAX_CHN_NUM)。
- pstChnNrParam 各成员的限制参见 VPSS_CHN_NR_PARAM_S 结构说明。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_GetChnNrParam

【描述】

获取通道 Nr 高级属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetChnNrParam(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_NR_PARAM_S *pstChnNrParam)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstChnNrParam	Nr 高级属性设置	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	支持
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A/Hi3515C	不支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法，[0, VPSS_MAX_CHN_NUM)。
- pstChnNrParam 各成员的限制参见 VPSS_CHN_NR_PARAM_S 结构说明。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetChnSpParam

【描述】

设置通道 SP 高级属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetChnSpParam(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_SP_PARAM_S *pstChnSpParam)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstChnSpParam	SP 高级属性设置。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	0 通道不支持 SP 功能，对其调用此接口无效，但不会报错。
Hi3518/Hi3516C	支持
Hi3520D/Hi3515A/Hi3515C	支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法，[0, VPSS_MAX_CHN_NUM)。
- pstChnSpParam 各成员的限制参见 VPSS_CHN_SP_PARAM_S 结构说明。

【举例】

无。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

HI_MPI_VPSS_GetChnSpParam

【描述】

获取通道 Sp 高级属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetChnSpParam(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CHN_SP_PARAM_S *pstChnSpParam)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
pstChnSpParam	Sp 高级属性设置	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	0 通道不支持 SP 功能，对其调用此接口无效，但不会报错。
Hi3518/Hi3516C	支持
Hi3520D/Hi3515A/ Hi3515C	支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法，[0, VPSS_MAX_CHN_NUM)。
- pstChnSpParam 各成员的限制参见 VPSS_CHN_SP_PARAM_S 结构说明。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetPreScale

【描述】

设置 VPSS 预缩放属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetPreScale(VPSS_GRP VpssGrp,VPSS_PRESCALE_INFO_S
*pstPreScaleInfo)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstPreScaleInfo	预缩放属性设置	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3521/Hi3520A	支持，但应注意目标图像的大小不要超过 GROUP 最大处理能力，参见 VPSS_PRESCALE_INFO_S。
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A /Hi3515C	支持，但应注意目标图像的大小不要超过 GROUP 最大处理能力，请参见 VPSS_PRESCALE_INFO_S。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 预缩放只针对 VI 采集图像有效。
- Bypass 通道不支持此功能。
- pstPreScaleInfo 各成员的限制参见 VPSS_PRESCALE_INFO_S 结构说明。
- Hi3520D/Hi3515A/Hi3515C 只支持水平 2 倍的预缩放，并且对于宽度大于 960 的图像，禁止使用预缩放。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_GetPreScale

【描述】

获取 VPSS 预缩放属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetPreScale(VPSS_GRP VpssGrp, VPSS_PRESCALE_INFO_S
*pstPreScaleInfo)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstPreScaleInfo	预缩放属性设置	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	支持
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A/Hi3515C	支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- pstPreScaleInfo 各成员的限制参见 VPSS_PRESCALE_INFO_S 结构说明。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetChnField

【描述】

设置通道抽场属性。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_VPSS_SetChnField(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CAPSEL_E *enCapSel)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。	输入
enCapSel	通道抽场信息	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	只有通道 0、1 支持抽场功能，并且必须是相同配置。因此调用此接口会同时改变 0、1 通道的抽场选择。
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A/Hi3515C	不支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法，只有 0、1 通道支持此功能。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

无。

HI_MPI_VPSS_GetChnField

【描述】

获取通道抽场属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetChnField(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_CAPSEL_E *enCapSel)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS Chn 号。 取值范围：[0, VPSS_MAX_CHN_NUM)。	输入
*enCapSel	通道抽场信息	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持
Hi3521/Hi3520A	支持。
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A/Hi3515C	不支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- CHN 号必须合法，[0, VPSS_MAX_CHN_NUM)。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetGrpField

【描述】

设置组抽场属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetGrpField(VPSS_GRP VpssGrp, VPSS_CAPSEL_E enCapSel)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
enCapSel	组抽场信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	支持
Hi3521/Hi3520A	支持
Hi3518/Hi3516C	支持

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3520D/Hi3515A/Hi3515C	支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 此接口对帧图像也会强制抽场。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_GetGrpField

【描述】

获取组抽场属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetGrpField(VPSS_GRP VpssGrp, VPSS_CAPSEL_E *penCapSel)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
*enCapSel	组抽场信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【芯片差异】

芯片	描述
Hi3531/Hi3532	支持
Hi3521/Hi3520A	支持
Hi3518/Hi3516C	支持
Hi3520D/Hi3515A/Hi3515C	支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetGrpSizer

【描述】

设置 VPSS 尺寸筛选信息。

【语法】

```
HI_S32 HI_MPI_VPSS_SetGrpSizer(VPSS_GRP VpssGrp, VPSS_SIZER_INFO_S
*pstVpssSizerInfo)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssSizerInfo	尺寸筛选信息。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	支持，但一般情况下无需调用此接口
Hi3521/Hi3520A	支持。
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A/Hi3515C	支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 尺寸筛选只针对 VI 采集图像有效。
- Bypass 通道不支持此功能。
- 尺寸筛选位于 VPSS 处理的最前端，影响后续所有处理

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_GetGrpSizer

【描述】

获取 VPSS 尺寸筛选信息。

【语法】

```
HI_S32 HI_MPI_VPSS_GetGrpSizer(VPSS_GRP VpssGrp, VPSS_SIZER_INFO_S
*pstVpssSizerInfo)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssSizerInfo	尺寸筛选信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	支持。
Hi3521/Hi3520A	支持。
Hi3518/Hi3516C	不支持
Hi3520D/Hi3515A/Hi3515C	支持

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetGrpFrameRate

【描述】

设置 VPSS 帧率控制信息。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VPSS_SetGrpFrameRate(VPSS_GRP VpssGrp, VPSS_FRAME_RATE_S
*pstVpssFrameRate)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssFrameRate	帧率控制信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	支持，但一般情况下无需调用此接口
Hi3521/Hi3520A	支持。
Hi3518/Hi3516C	支持。
Hi3520D/Hi3515A/Hi3515C	支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 帧率控制只针对 VI 采集图像有效。
- Bypass 通道不支持此功能。
- 帧率控制位于尺寸筛选之后，其它所有功能之前，影响后续所有处理。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

无。

HI_MPI_VPSS_GetGrpFrameRate

【描述】

获取 VPSS 帧率控制信息。

【语法】

```
HI_S32 HI_MPI_VPSS_GetGrpFrameRate(VPSS_GRP VpssGrp, VPSS_FRAME_RATE_S
*pstVpssFrameRate)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pstVpssFrameRate	帧率控制信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	支持。
Hi3521/Hi3520A	支持。
Hi3518/Hi3516C	支持。
Hi3520D/Hi3515A/Hi3515C	支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_SetDelay

【描述】

设置延迟启动 VPSS 处理的延时队列长度。

主要应用场景：

- 智能应用时，VI 采集的图像提交给 VPSS 处理前，需要对图像进行 TDE 方面的一些处理。由于 VPSS 处理非常快，需保证在做 VPSS 处理前，TDE 已经完成相应处理，因此提供延迟启动 VPSS 处理的功能。
- VI-VO 视频级联时，通过该接口可调节主片视频图像与从片之间的延时，减少两者的时间差。

【语法】

```
HI_S32 HI_MPI_VPSS_SetDelay(VPSS_GRP VpssGrp, HI_U32 u32Delay)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
u32Delay	延时队列长度。 取值范围：[0, 5]	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。
- 延时功能仅对 VI 采集的图像有效。u32Delay 为延时队列的长度，延迟的时间大约为：VI 采集一帧图像所需时间 x u32Delay。
- Bypass 通道不支持此功能。

【举例】

无。

【相关主题】

HI_MPI_VPSS_GetDelay

HI_MPI_VPSS_GetDelay

【描述】

获取延迟启动 VPSS 处理的延时队列长度。

【语法】

```
HI_S32 HI_MPI_VPSS_GetDelay(VPSS_GRP VpssGrp, HI_U32 *pu32Delay)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
pu32Delay	获取延时队列长度指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- GROUP 号必须合法，[0, VPSS_MAX_GRP_NUM)。
- GROUP 必须已创建。

【举例】

无。

【相关主题】

HI_MPI_VPSS_SetDelay

HI_MPI_VPSS_SetExtChnAttr

【描述】

设置 VPSS 扩展通道属性。

【语法】

```
HI_S32 HI_MPI_VPSS_SetExtChnAttr(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_EXT_CHN_ATTR_S *pstExtChnAttr)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 扩展通道号 取值范围： (VPSS_MAX_PHY_CHN_NUM,VPSS_MAX_CHN_NUM)	输入
pExtChnAttr	VPSS 扩展通道属性	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

芯片	描述
Hi3521/Hi3520A	不支持。
Hi3518/Hi3516C	支持。
Hi3520D/Hi3515A/Hi3515C	不支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

- 扩展通道属性必须正确设置，其中源绑定通道必须为物理通道，宽、高需要满足 2 对齐，SrcFrameRate 必须大于等于 DstFrameRate，若都设置成-1，表示不进行帧率控制。
- 多个扩展通道可以绑定到同一个物理通道，若源物理通道绑定的扩展通道达到最大数目，设置将会失败。
- GROUP 必须已创建。

【举例】

无。

【相关主题】

无。

HI_MPI_VPSS_GetExtChnAttr

【描述】

获取 VPSS 扩展通道属性。

【语法】

```
HI_S32 HI_MPI_VPSS_GetExtChnAttr(VPSS_GRP VpssGrp, VPSS_CHN VpssChn,
VPSS_EXT_CHN_ATTR_S *pstExtChnAttr)
```

【参数】

参数名称	描述	输入/输出
VpssGrp	VPSS GROUP 号。 取值范围：[0, VPSS_MAX_GRP_NUM)。	输入
VpssChn	VPSS 扩展通道号 取值范围： (VPSS_MAX_PHY_CHN_NUM,VPSS_MAX_CHN_NUM)	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pExtChnAttr	VPSS 扩展通道属性	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【芯片差异】

芯片	描述
Hi3531/Hi3532	不支持。
Hi3521/Hi3520A	不支持。
Hi3518/Hi3516C	支持
Hi3520D/Hi3515A/Hi3515C	不支持。

【需求】

- 头文件：hi_comm_vpss.h、mpi_vpss.h
- 库文件：libmpi.a

【注意】

GROUP 必须已创建。

【举例】

无。

【相关主题】

无。

5.4 数据类型

VPSS 模块相关数据类型定义如下：

- VPSS_MAX_GRP_NUM：定义 VPSS GROUP 的最大个数。
- VPSS_MAX_CHN_NUM：定义 VPSS 通道的最大个数。
- VPSS_MAX_PHY_CHN_NUM：定义 VPSS 物理通道的最大个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VPSS_MAX_EXT_CHN_NUM：定义 VPSS 扩展通道的最大个数。
- VPSS_BSTR_CHN：定义大码流通道的 ID 号。
- VPSS_LSTR_CHN：定义小码流通道的 ID 号。
- VPSS_PRE0_CHN：定义预览通道 0 的 ID 号。
- VPSS_PRE1_CHN：定义预览通道 1 的 ID 号。
- VPSS_BYPASS_CHN：定义直通通道的 ID 号。
- VPSS_GRP：定义 VPSS_GRP 类型。
- VPSS_CHN：定义 VPSS_CHN 类型。
- VPSS_FRAME_WORK_E：定义视频边框。
- VPSS_DIE_MODE_E：定义 DIE 模式。
- VPSS_SF_WINDOW_E：定义空域滤波控制窗大小。
- VPSS_DISPLAY_MODE_E：定义显示模式。
- VPSS_CROP_COORDINATE_E：定义 CROP 起点坐标的模式。
- RECT_S：定义一个矩形区域。
- VPSS_CROP_INFO_S：定义 CROP 功能所需信息。
- VPSS_GRP_ATTR_S：定义 VPSS GROUP 静态属性。
- VPSS_FRAME_S：定义视频边框属性。
- VPSS_CHN_ATTR_S：定义 VPSS 物理通道静态属性。
- VPSS_GRP_PARAM_S：定义 VPSS 高级属性。
- VPSS_CHN_MODE_E：定义 VPSS CHN 工作类型。
- VPSS_CHN_MODE_S：定义 VPSS CHN 工作模式结构。
- VPSS_CHN_NR_PARAM_S：定义通道 Nr 高级属性。
- VPSS_CHN_SP_PARAM_S：定义通道 Sp 高级属性。
- VPSS_CHN_PARAM_S：定义通道高级属性。
- VPSS_CAPSEL_E：定义 VPSS 抽场类型。
- VPSS_PRESCALE_INFO_S：定义 VPSS 预缩放属性。
- VPSS_SIZER_INFO_S：定义 VPSS 尺寸筛选属性。
- VPSS_FRAME_RATE_S：定义 VPSS 帧率控制属性。
- VPSS_EXT_CHN_ATTR_S：定义 VPSS 扩展通道的属性。

VPSS_MAX_GRP_NUM

【说明】

定义 VPSS GROUP 的最大个数。

【定义】

```
#define VPSS_MAX_GRP_NUM   128
```

【注意事项】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关数据类型及接口】

无。

VPSS_MAX_CHN_NUM

【说明】

定义 VPSS 通道的最大个数。

【定义】

```
#define VPSS_MAX_CHN_NUM   (VPSS_MAX_PHY_CHN_NUM + VPSS_MAX_EXT_CHN_NUM +
1)
```

【注意事项】

通道包括物理通道，扩展通道，直通(BYPASS)通道之和。

【相关数据类型及接口】

无。

VPSS_MAX_PHY_CHN_NUM

【说明】

定义 VPSS 物理通道的最大个数。

【定义】

```
#define  VPSS_MAX_PHY_CHN_NUM 2
```

【注意事项】

和硬件资源相关，Hi3521/Hi3520A 为 4，Hi3518/Hi3516C 为 2

【相关数据类型及接口】

无。

VPSS_MAX_EXT_CHN_NUM

【说明】

定义 VPSS 扩展通道的最大个数。

【定义】

```
#define  VPSS_MAX_EXT_CHN_NUM 2
```

【注意事项】

数目根据方案设计和系统能力而定。Hi3518/Hi3516C 为 5

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VPSS_BSTR_CHN

【说明】

定义大码流通道的 ID 号。

【定义】

```
#define VPSS_BSTR_CHN   0
```

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_LSTR_CHN

【说明】

定义小码流通道的 ID 号。

【定义】

```
#define VPSS_LSTR_CHN   1
```

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_PRE0_CHN

【说明】

定义预览通道 0 的 ID 号。

【定义】

```
#define VPSS_PRE0_CHN   2
```

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_PRE1_CHN

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义预览通道 1 的 ID 号。

【定义】

```
#define VPSS_PRE1_CHN   3
```

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_BYPASS_CHN

【说明】

定义直通通道的 ID 号。

【定义】

```
#define VPSS_BYPASS_CHN   4
```

【注意事项】

直通通道索引排布在物理通道后，因此 Hi3531/Hi3521 的 ID 号为 4，Hi3518/Hi3516C 的 ID 号为 2。

【相关数据类型及接口】

无。

VPSS_GRP

【说明】

定义 VPSS_GRP 类型。

【定义】

```
typedef   HI_S32    VPSS_GRP;
```

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_CHN

【说明】

定义 VPSS_CHN 类型。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef   HI_S32    VPSS_CHN;
```

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_DIE_MODE_E

【说明】

定义 DIE 模式。

【定义】

```
typedef enum  HI_VPSS_DIE_MODE_E
{
   VPSS_DIE_MODE_AUTO      = 0,
   VPSS_DIE_MODE_NODIE     = 1,
   VPSS_DIE_MODE_DIE       = 2,
   VPSS_DIE_MODE_BUTT
}VPSS_DIE_MODE_E;
```

【成员】

成员名称	描述
VPSS_DIE_MODE_AUTO	根据当前图像格式，自适应是否做 De-interlace。
VPSS_DIE_MODE_NODIE	强制不做 De-interlace。
VPSS_DIE_MODE_DIE	强制做 De-interlace。

【注意事项】

建议首选自适应模式 VPSS_DIE_MODE_AUTO。

【相关数据类型及接口】

VPSS_GRP_ATTR_S

VPSS_DISPLAY_MODE_E

【说明】

定义显示模式。

【定义】

```
typedef enum  HI_VPSS_DISPLAY_MODE_E
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VPSS_DISPLAY_MODE_TV    = 0,
    VPSS_DISPLAY_MODE_PC    = 1,
    VPSS_DISPLAY_MODE
}VPSS_DISPLAY_MODE_E;
```

【成员】

成员名称	描述
VPSS_DISPLAY_MODE_TV	TV 模式。
VPSS_DISPLAY_MODE_PC	PC 模式。

【注意事项】

此结构体影响亮度输出范围，TV 模式下，亮度范围为 16～235；PC 模式下，亮度范围为 0～255。

【相关数据类型及接口】

VPSS_GRP_PARAM_S

VPSS_SF_WINDOW_E

【说明】

定义空域滤波控制窗大小。窗越大，滤波强度越大，得到的图像越模糊。

【定义】

```
typedef enum  HI_VPSS_SF_WINDOW_E
{
    VPSS_SF_WINDOW_3X3      = 0,
    VPSS_SF_WINDOW_5X5      = 1,
    VPSS_SF_WINDOW_7X7      = 2,
    VPSS_SF_WINDOW_9X9     = 3,
    VPSS_SF_WINDOW_BUTT
}VPSS_SF_WINDOW_E;
```

【成员】

成员名称	描述
VPSS_SF_WINDOW_3X3	控制窗大小为 3x3。
VPSS_SF_WINDOW_5X5	控制窗大小为 5x5。
VPSS_SF_WINDOW_7X7	控制窗大小为 7x7。
VPSS_SF_WINDOW_9X9	控制窗大小为 9x9。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

VPSS_GRP_PARAM_S

VPSS_FRAME_WORK_E

【说明】

定义四条视频边框。

【定义】

```
typedef enum HI_VPSS_FRAME_WORK_E
{
    VPSS_FRAME_WORK_LEFT     =  0,
    VPSS_FRAME_WORK_RIGHT    =  1,
    VPSS_FRAME_WORK_BOTTOM   =  2,
    VPSS_FRAME_WORK_TOP      =  3,
    VPSS_FRAME_WORK_BUTT
}VPSS_FRAME_WORK_E;;
```

【成员】

成员名称	描述
VPSS_FRAME_WORK_LEFT	左边框。
VPSS_FRAME_WORK_RIGHT	右边框。
VPSS_FRAME_WORK_BOTTOM	下边框。
VPSS_FRAME_WORK_TOP	上边框。

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_CROP_COORDINATE_E

【说明】

定义 CROP 起点坐标的模式。

【定义】

```
typedef enum HI_VPSS_CROP_COORDINATE_E
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    VPSS_CROP_RITIO_COOR = 0,
    VPSS_CROP_ABS_COOR
}VPSS_CROP_COORDINATE_E;
```

【成员】

成员名称	描述
VPSS_CROP_RITIO_COOR	相对坐标。
VPSS_CROP_ABS_COOR	绝对坐标。

【注意事项】

相对坐标，即起始点的坐标值是以与当前图像宽高的比率来表示，使用时需做转换，具体参见 VPSS_CROP_INFO_S。

【相关数据类型及接口】

VPSS_CROP_INFO_S

RECT_S

【说明】

定义一个矩形区域。

【定义】

```
typedef struct hiRECT_S
{
    HI_S32 s32X;
    HI_S32 s32Y;
    HI_U32 u32Width;
    HI_U32 u32Height;
}RECT_S;
```

【成员】

成员名称	描述
s32X	起始点横坐标
s32Y	起始点纵坐标
u32Width	矩形的宽
u32Height	矩形的高

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

VPSS_CROP_INFO_S

VPSS_CROP_INFO_S

【说明】

定义 CROP 功能所需信息。

【定义】

```
typedef struct
{
    HI_BOOL bEnable;
    VPSS_CROP_COORDINATE_E  enCropCoordinate;
    RECT_S  stCropRect;
    VPSS_CAPSEL_E   enCapSel;
}VPSS_CROP_INFO_S;
```

【成员】

成员名称	描述
bEnable	CROP 使能开关。
enCropCoordinate	CROP 起始点坐标模式。
stCropRect	CROP 的矩形区域
enCapSel	抽场选择。

【注意事项】

若 enCropCoordinate 为 VPSS_CROP_RITIO_COOR（相对坐标模式），使用 stCropRect 的成员时应做转换，计算公式为 s32X = 起始点坐标/实际图像宽 x1000 $\frac{\text{起始点坐标}}{\text{实际图像宽}} * 1000$，合法取值范围[0, 999]。公式同样适用于纵坐标计算。

【相关数据类型及接口】

VPSS_CROP_COORDINATE_E

VPSS_GRP_ATTR_S

【说明】

定义 VPSS GROUP 静态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct HI_VPSS_GRP_ATTR_S
{
    HI_U32  u32MaxW;
    HI_U32  u32MaxH;
    PIXEL_FORMAT_E enPixFmt;
    VPSS_DIE_MODE_E enDieMode;
    HI_BOOL bDrEn;
    HI_BOOL bDbEn;
    HI_BOOL bIeEn;
    HI_BOOL bNrEn;
    HI_BOOL bHistEn;
}VPSS_GRP_ATTR_S;
```

【成员】

成员名称	描述			
	Hi3518/Hi3516C	Hi3521/Hi3520A	Hi3531/ Hi3532	Hi3520D/Hi3515A/Hi3515C
u32MaxW	最大图像宽度 取值范围：[32, 2048] 静态属性，创建 Group 时设定，不可更改。	最大图像宽度 取值范围：[32, 4096] 静态属性，创建 Group 时设定，不可更改。	最大图像宽度 取值范围：[32, 4096] 静态属性，创建 Group 时设定，不可更改。	最大图像宽度 取值范围：[48, 4080] 静态属性，创建 Group 时设定，不可更改。
u32MaxH	最大图像高度 取值范围：[32, 2048] 静态属性，创建 Group 时设定，不可更改。	最大图像高度 取值范围：[32, 4080] 静态属性，创建 Group 时设定，不可更改。	最大图像高度 取值范围：[32, 4080] 静态属性，创建 Group 时设定，不可更改。	最大图像高度 取值范围：[32, 4080] 静态属性，创建 Group 时设定，不可更改。
enPixFmt	像素格式。仅支持 SemiPlannar422 和 SemiPlannar420。 静态属性，创建 Group 时设定，不可更改。			
enDieMode	Die 模式。			
bDrEn	保留	保留	Dr 使能开关	保留
bDbEn	保留	保留	Db 使能开关	保留
bIeEn	保留	IE 使能开关	IE 使能开关	IE 使能开关
bNrEn	Nr 使能开关			
bHistEn	直方图使能开关			

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

- Dr/Db 与 Nr、Die 的硬件在工作时是互斥的，设置通道属性时不能同时使能。
- 直方图和 Die 使能时，必须同时使能 Nr。
- Hi3520D/Hi3515A/Hi3515C 不支持对超过 960 的图像做 DIE。
- u32MaxW、u32MaxH、enPixFmt 定义了当前 GRP 最大可接收的输入图像大小，创建 GRP 时设定，不可动态更改。
- 若后端是编码，则必须打开直方图开关，否则码率控制不准确。

【相关数据类型及接口】

- PIXEL_FORMAT_E
- VPSS_DIE_MODE_E

VPSS_FRAME_S

【说明】

定义视频边框属性。

【定义】

```
typedef struct HI_VPSS_FRAME_S
{
   HI_U32  u32Width[4];
   HI_U32  u32Color[4];
}VPSS_FRAME_S;
```

【成员】

成员名称	描述
u32Width[4]	边框宽度。
u32Color[4]	边框颜色。

【注意事项】

- u32Width[4]下标与上下左右四条边框对应关系
  - 0：左
  - 1：右
  - 2：下
  - 3：上
- 边框宽度合法取值范围为[0, 14]，并且只能取偶数值。
- 边框颜色为 RGB 模式，R、G、B 分别对应位域如下：

```
 31          23          15          7           0
  |-------------|--------R------|--------G------|--------B------|
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- Hi3520D/Hi3515A/Hi3515C 四条边框在设置时必须为同一颜色。

【相关数据类型及接口】

VPSS_CHN_ATTR_S

VPSS_CHN_ATTR_S

【说明】

定义 VPSS 物理通道的属性。

【定义】

```
typedef struct HI_VPSS_CHN_ATTR_S
{
    HI_BOOL bSpEn;
    HI_BOOL bFrameEn;
    VPSS_FRAME_S  stFrame;
}VPSS_CHN_ATTR_S;
```

【成员】

成员名称	描述
bSpEn	Sp 使能开关。
bFrameEn	边框使能开关。
stFrame	边框属性。

【注意事项】

无。

【相关数据类型及接口】

VPSS_FRAME_S

VPSS_GRP_PARAM_S

【说明】

定义 VPSS 高级属性。

【定义】

```
typedef struct HI_VPSS_GRP_PARAM_S
{
    HI_U32 u32Luminance;
    HI_U32 u32Contrast;
    HI_U32 u32DarkEnhance;
    HI_U32 u32BrightEnhance;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32IeStrength;
    HI_U32 u32IeSharp;
    HI_U32 u32SfStrength;
    HI_U32 u32TfStrength;
    HI_U32 u32MotionThresh;
    HI_U32 u32DiStrength;
    HI_U32 u32ChromaRange;
    HI_U32 u32NrWforTsr;
    VPSS_SF_WINDOW_E enSfWindow;
    VPSS_DISPLAY_MODE_E enDisMode;
}VPSS_GRP_PARAM_S;
```

【成员】

成员名称	描述			
	Hi3518/Hi3516C	Hi3521/Hi3520A	Hi3531/ Hi3532	Hi3520D/Hi3515A/Hi3515C
u32Luminance	保留。	保留。	亮度。 取值范围：0～48，默认值为 32。 影响暗区定义范围，值越大，暗区范围越小。	保留。
u32Contrast	保留。	保留。	对比度。 取值范围：0～48，默认值为 8。 影响暗区和亮区调整幅度，值越大，调整越明显。	对比度。 取值范围：0～255，默认值为 64。 影响暗区和亮区调整幅度，值越大，调整越明显。
u32DarkEnhance	保留。	保留。	暗区增强。 取值范围：0～48，默认值为 16。 影响暗区细节增强强度，值越大，增强越明显。	保留。
u32BrightEnhance	保留。	保留。	亮区增强。 取值范围：0～48，默认值为 16。 影响亮区细节增强强度，值越大，增强越明显。	保留。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述			
	Hi3518/Hi3516C	Hi3521/Hi3520A	Hi3531/ Hi3532	Hi3520D/Hi3515A/Hi3515C
u32IeStrength	保留。	IE 强度。 取值范围：0～8，默认值为 2。 影响对比度，值越大，对比越明显。	IE 强度。 取值范围：0～255，默认值为 32。 影响对比度，值越大，对比越明显。	IE 强度。 取值范围：0～255，默认值为 6。 影响对比度，值越大，对比越明显。
u32IeSharp	保留。	IE 锐度。 取值范围：0～100，默认值为 10。 影响对比度，值越小，对比越明显。	IE 锐度。 取值范围：0～7，默认值为 7。 影响对比度，值越大，对比越明显。	保留。
u32SfStrength	空域去噪强度。 取值范围：0～255，默认为 32。 影响空域滤波强度，值越大，滤波强度越大。	保留。	空域去噪强度。 取值范围：0～7，默认值为 3。 影响空域滤波强度，值越大，滤波强度越大。	空域去噪强度。 取值范围：0～255，默认值为 16。 影响空域滤波强度，值越大，滤波强度越大。
u32TfStrength	时域去噪强度。 取值范围：0～63，默认为 8。 影响时域滤波强度，值越大，滤波强度越大。	保留	时域去噪强度。 取值范围：0～15，默认值为 1。 影响时域滤波强度，值越大，滤波强度越大。	时域去噪强度。 取值范围：0～31，默认值为 4。 影响时域滤波强度，值越大，滤波强度越大。
u32MotionThresh	保留。	保留。	移动判决阈值。 取值范围：0～7，默认值为 1。 影响参与滤波的运动区域，值越大，运动区域越容易参与滤波。	0、1 通道的滤波强度。 取值范围：0～8，默认值为 2。 影响编码通道的滤波强度，值越大，噪声抑制越强。
u32DiStrength	保留。	De-interlace 强度。 取值范围：0～15，默认值为 0。 影响去隔行的效果，值越大，拉丝	De-interlace 强度。 取值范围：0～7，默认值为 7。 影响去隔行的效果，值越大，拉丝越少，	保留。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述			
	Hi3518/Hi3516C	Hi3521/Hi3520A	Hi3531/ Hi3532	Hi3520D/Hi3515A/Hi3515C
		越少，但闪烁现象会增加。	但闪烁现象会增加。	
u32ChromaRange	色度幅度。 取值范围：0～255。默认值为8。 影响色度滤波强度，值越大，滤波强度越大。	保留。	色度幅度。 取值范围：0～31。默认值为4。 影响色度滤波强度，值越大，滤波强度越大。	保留。
u32NrWforTsr	保留。	保留。	时域滤波比重。 取值范围：0～8。默认值为8。 影响滤波强度，值越小，时域滤波比重越大。	保留。
enSfWindow	保留。	保留。	空域滤波控制窗大小。 取值范围：0～4。默认值为0。 影响滤波强度，窗越大，滤波强度越大。	保留。
enDisMode	保留。	保留。	显示模式。 取值范围：0～1。默认值为0。 影响显示亮度输出，TV模式下，亮度范围为16～235；PC模式下，亮度范围为0～255。	保留。

【注意事项】

此结构各成员取值在不同芯片上有差异，参见成员描述。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VPSS_CHN_MODE_E

【说明】

定义 VPSS CHN 工作类型。

【定义】

```
typedef enum HI_VPSS_CHN_MODE_E
{
    VPSS_CHN_MODE_AUTO,
    VPSS_CHN_MODE_USER
}VPSS_CHN_MODE_E;
```

【成员】

成员名称	描述
VPSS_CHN_MODE_AUTO	自动模式，通道的默认工作模式。
VPSS_CHN_MODE_USER	用户设置模式。

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_CHN_MODE_S

【说明】

定义 VPSS CHN 工作模式结构。

【定义】

```
typedef struct HI_VPSS_CHN_MODE_S
{
    VPSS_CHN_MODE_E  enChnMode;
    HI_U32 u32Width;
    HI_U32 u32Height;
    HI_BOOL bDouble;
    PIXEL_FORMAT_E  enPixelFormat;
}VPSS_CHN_MODE_S;
```

【成员】

成员名称	描述
enChnMode	VPSS CHN 工作模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32Width	目标图像宽。
u32Height	目标图像高。
bDouble	是否做倍帧，只对 VPSS_PRE0_CHN 有效。
enPixelFormat	目标图像像素格式。

【注意事项】

u32Width 和 u32Height 取值范围为应符合 VPSS 处理能力，请参见 VPSS_GRP_ATTR_S，并且必须为偶数。

【相关数据类型及接口】

无。

VPSS_CHN_NR_PARAM_S

【说明】

定义通道 Nr 高级参数。

【定义】

```
typedef struct
{
    HI_U32 u32SfStrength;
    HI_U32 u32TfStrength;
    HI_U32 u32MotionThresh;
    HI_U32 u32ChromaRange;
    HI_U32 u32NrWforTsr;
}VPSS_CHN_NR_PARAM_S;
```

【成员】

成员名称	描述
u32SfStrength	空域滤波强度。 取值范围：0～16，默认值为 16。 影响空域滤波强度，值越大，滤波强度越大
u32TfStrength	时域滤波强度。 取值范围：0～16，默认值为 8。 影响时域滤波强度，值越大，滤波强度越大
u32MotionThresh	保留。
u32ChromaRange	保留。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32NrWforTsr	保留。

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_CHN_SP_PARAM_S

【说明】

定义通道 SP 高级属性。

【定义】

```
typedef struct
{
  HI_U32 u32LumaGain;
}VPSS_CHN_SP_PARAM_S;
```

【成员】

成员名称	描述	
	Hi3521	Hi3520D/Hi3515A/Hi3515C
u32LumaGain	SP 强度。 取值范围：0～255，默认值为 40。 影响锐化强度，值越大，锐化强度越大	SP 强度。 取值范围：0～255，默认值为 64。 影响锐化强度，值越大，锐化强度越大

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_CHN_PARAM_S

【说明】

定义 VPSS CHN 高级属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct
{
  VPSS_CHN_SP_PARAM_S  stVpssChnSpParam;
  VPSS_CHN_NR_PARAM_S  stVpssChnNrParam;
} VPSS_CHN_PARAM_S;
```

【成员】

成员名称	描述
stVpssChnSpParam	通道 Sp 高级属性。
stVpssChnNrParam	通道 Nr 高级属性。

【注意事项】

无。

【相关数据类型及接口】

- VPSS_CHN_SP_PARAM_S
- VPSS_CHN_NR_PARAM_S

VPSS_CAPSEL_E

【说明】

定义 VPSS 抽场类型。

【定义】

```
typedef enum hiVPSS_CAPSEL_E
{
    VPSS_CAPSEL_BOTH = 0,
    VPSS_CAPSEL_TOP,
    VPSS_CAPSEL_BOTTOM,
    VPSS_CAPSEL_BUTT
} VPSS_CAPSEL_E;
```

【成员】

成员名称	描述
VPSS_CAPSEL_BOTH	两场。
VPSS_CAPSEL_TOP	顶场。
VPSS_CAPSEL_BOTTOM	底场。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_PRESCALE_INFO_S

【说明】

定义 VPSS 预缩放属性。

【定义】

```
typedef struct
{
    HI_BOOL bPreScale;
    VPSS_CAPSEL_E     enCapSel;
    SIZE_S         stDestSize;
}VPSS_PRESCALE_INFO_S;
```

【成员】

成员名称	描述
bPreScale	预缩放开关。
enCapSel	抽场选择。
stDestSize	目标图像大小，特别注意此大小不得超过 GROUP 的最大处理能力。

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_SIZER_INFO_S

【说明】

定义 VPSS 尺寸筛选属性。

【定义】

```
typedef struct
{
    HI_BOOL bSizer;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    SIZE_S  stSize;
}VPSS_SIZER_INFO_S;
```

【成员】

成员名称	描述
bSizer	尺寸筛选功能开关。
SIZE_S	筛选的尺寸。

【注意事项】

无。

【相关数据类型及接口】

无。

VPSS_FRAME_RATE_S

【说明】

定义 VPSS 帧率控制属性。

【定义】

```
typedef struct
{
    HI_S32  s32SrcFrmRate;
    HI_S32  s32DstFrmRate;
} VPSS_FRAME_RATE_S;
```

【成员】

成员名称	描述
s32SrcFrmRate	源帧率。
s32DstFrmRate	目标帧率。

【注意事项】

- 源帧率与目标帧率都为-1，则不进行帧率控制。
- 目标帧率不能大于源帧率。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VPSS_EXT_CHN_ATTR_S

【说明】

定义 VPSS 扩展通道的属性。

【定义】

```
typedef struct HI_VPSS_EXT_CHN_ATTR_S
{
    VPSS_CHN  s32BindChn;
    HI_U32 u32Width;
    HI_U32 u32Height;
    HI_S32 s32SrcFrameRate;
    HI_S32 s32DstFrameRate;
    PIXEL_FORMAT_E enPixelFormat;
} VPSS_EXT_CHN_ATTR_S;
```

【成员】

成员名称	描述
s32BindChn	绑定源物理通道
u32Width	扩展通道输出目标宽度
u32Height	扩展通道输出目标高度
s32SrcFrameRate	扩展通道输入源帧率
s32DstFrameRate	扩展通道输出目标帧率
enPixelFormat	扩展通道输出图像数据格式

【注意事项】

- 绑定源物理通道必须合法，范围为[0, VPSS_MAX_PHY_CHN_NUM)。
- 宽、高必须满足 2 对齐。
- 目标帧率不能大于源帧率，源帧率与目标帧率都为-1，则不进行帧率控制。
- 输出数据格式支持 YUV420 semiplanar 及 YUV422 semiplanar

【相关数据类型及接口】

无。

5.5 错误码

视频处理子系统 API 错误码如表 5-5 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表5-5 VPSS API 错误码

错误代码	宏定义	描述
0xA0088001	HI_ERR_VPSS_INVALID_DEVID	VPSS GROUP 号无效
0xA0088002	HI_ERR_VPSS_INVALID_CHNID	VPSS 通道号无效
0xA0088003	HI_ERR_VPSS_ILLEGAL_PARAM	VPSS 参数设置无效
0xA0088004	HI_ERR_VPSS_EXIST	VPSS GROUP 已创建
0xA0088005	HI_ERR_VPSS_UNEXIST	VPSS GROUP 未创建
0xA0088006	HI_ERR_VPSS_NULL_PTR	输入参数空指针错误
0xA0086008	HI_ERR_VPSS_NOT_SUPPORT	操作不支持
0xA0088009	HI_ERR_VPSS_NOT_PERM	操作不允许
0xA008800C	HI_ERR_VPSS_NOMEM	分配内存失败
0xA008800D	HI_ERR_VPSS_NOBUF	分配 BUF 池失败
0xA0088010	HI_ERR_VPSS_NOTREADY	VPSS 系统未初始化
0xA0088 012	HI_ERR_VPSS_BUSY	VPSS 系统忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

6 视频编码.....6-1
6.1 概述.....6-1
6.2 功能描述.....6-2
6.2.1 编码数据流程图.....6-2
6.2.2 编码通道与通道组.....6-3
6.2.3 码率控制.....6-3
6.2.4 跳帧参考.....6-4
6.2.5 高级跳帧参考模式.....6-5
6.2.6 彩转灰.....6-6
6.2.7 裁剪编码.....6-6
6.2.8 ROI.....6-7
6.2.9 JPEG 抓拍模式.....6-8
6.2.10 编码码流 buffer 配置模式.....6-8
6.2.11 编码帧存分配方式.....6-9
6.2.12 编码重构帧复用参考帧亮度内存模式.....6-10
6.3 API 参考.....6-11
6.4 数据类型.....6-105
6.5 错误码.....6-172

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 6-1 VENC 的数据流程图 6-2
图 6-2 编码通道功能模块划分 6-3
图 6-3 跳帧参考示意图 6-5
图 6-4 高级跳帧参考示意图 6-6
图 6-5 裁剪编码示意图 6-7
图 6-6 ROI 示意图 6-8
图 6-7 编码码流 buffer 配置模式 6-9

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 6-1 芯片编码规格 6-1
表 6-2 n 个通道编码使用 UserVB 方式时节省帧存数 6-10
表 6-3 H264eVBSource 和 H264eRcnEqualRef 组合使用说明 6-10
表 6-4 芯片相关编码通道宽高差异 6-18
表 6-5 编码器属性的约束 6-19
表 6-6 码率控制器属性的约束 6-19
表 6-7 典型平均比特率配置 6-20
表 6-8 搜索窗范围 6-50
表 6-9 Hi3531/Hi3532 H.264 编码器在不同图像分辨率下搜索窗默认配置 6-51
表 6-10 Hi3521/Hi3520A H.264 编码器在不同图像分辨率下搜索窗默认配置 6-51
表 6-11 Hi3518/Hi3516C H.264 编码器不同图像分辨率下搜索窗默认配置 6-51
表 6-12 Hi3520D/Hi3515A/Hi3515C H.264 编码器不同图像分辨率下搜索窗默认配置 6-51
表 6-13 不同 profile 下系统默认 trans 参数 6-59
表 6-14 不同 profile 下系统默认熵编码参数 6-62
表 6-15 视频编码 API 错误码 6-172

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

6 视频编码

6.1 概述

VENC 模块，即视频编码模块。本模块支持多路实时编码，且每路编码彼此独立，编码协议和编码 profile 可以不同。本模块支持视频编码同时，调度 Region 模块对编码图像内容进行叠加和遮挡。本模块的输入源包括三类：第一类是用户态读取图像文件向编码模块发送数据；第二类是视频输入（VIU）模块采集的图像经视频处理子系统（VPSS）发送到编码模块；第三类是视频输入（VIU）模块采集的图像直接发送到编码模块。不同型号的芯片支持不同的编码规格，芯片支持的编码规格如表 6-1 所示。

表6-1 芯片编码规格

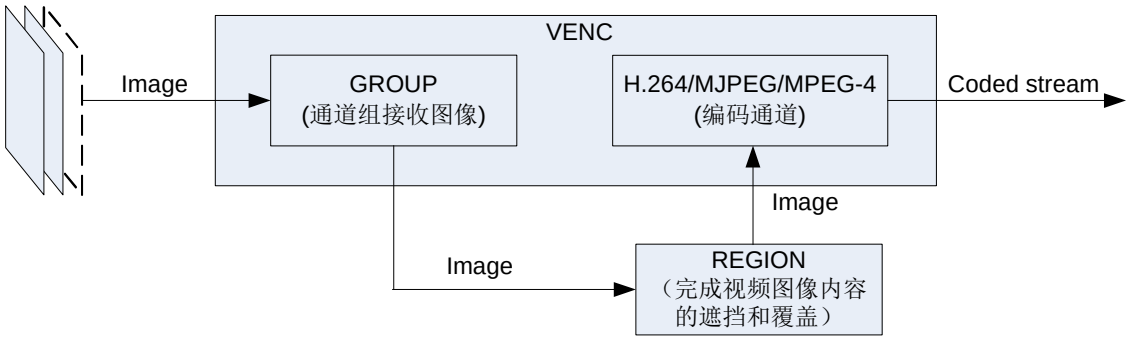
芯片	H.264				JPEG	MOTION JPEG	MPEG-4
	BP	MP	HP	SVC-T			
Hi3531/Hi3532	支持	支持	支持	不支持	支持	支持	支持
Hi3521/Hi3520A	支持	支持	不支持	不支持	支持	支持	不支持
Hi3518/Hi3516C	支持	支持	支持	支持	支持	支持	不支持
Hi3520D/Hi3515A/ Hi3515C	支持	支持	支持	不支持	支持	支持	不支持

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

6.2 功能描述

6.2.1 编码数据流程图

图6-1 VENC 的数据流程图



典型的编码流程包括了输入图像的接收、图像内容的遮挡和覆盖、图像的编码、以及码流的输出等过程。

VENC 模块由编码通道组子模块（GROUP）和编码协议子模块（H.264/JPEG/MJPEG/MPEG-4）组成。

通道组支持接收 YUV 格式图像输入，支持格式为 Semi-planar YUV 4:2:0 或 Semi-planar YUV 4:2:2，其中 H264E/MPEG-4 只支持 Semi-planar YUV 4:2:0，JPEG/MJPEG 支持 Semi-planar YUV 4:2:0 或 Semi-planar YUV 4:2:2。通道组模块接收外部原始图像数据，而不关心图像数据是来自哪个外部模块。

说明
通道组现在仅支持 VPSS 模块、VIU 模块和用户态发送图像。Hi3518/Hi3516C 支持 JPEG 闪光灯抓拍模式。当 Hi3518/Hi3516C 的 JPEG 设置为闪光灯抓拍模式时，通道组只能接收来自 VIU 模块（不经过 VPSS 模块）的图像，否则 JPEG 抓拍不到任何图像。

通道组接收到图像之后，比较图像尺寸和编码通道尺寸：

- 如果输入图像比编码通道尺寸大，VENC 将按照编码通道尺寸大小，调用 DSU 对源图像进行缩小，然后对缩小之后的图像进行编码。
- 如果输入图像比编码通道尺寸小，VENC 丢弃源图像。VENC 不支持放大输入图像编码。
- 如果输入图像与编码通道尺寸相当，VENC 直接接受源图像，进行编码。

说明
通道组的帧率控制默认不打开，需要用户调用接口设置。RC 中也具有帧率控制功能。推荐使用 RC 的帧率控制，这样不会对码率控制造成过大的冲击。

REGION 模块支持对图像内容的遮挡和叠加。

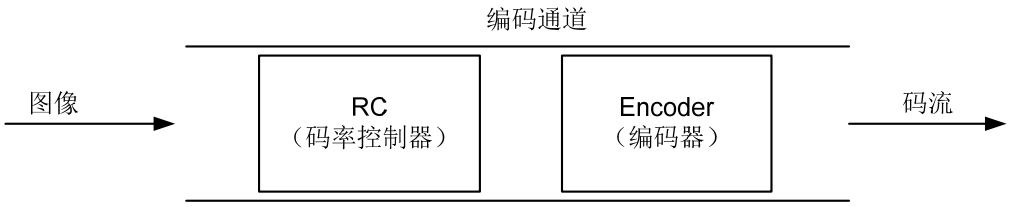
完成视频区域管理之后，图像被送入具体协议类型编码通道，完成视频编码，输出码流。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

6.2.2 编码通道与通道组

编码通道做为基本容器，保存不同编码通道的用户设置，管理不同编码通道的内部资源。编码通道完成图像转化为码流的功能，具体由码率控制器和编码器协同完成。这里的编码器指的是狭义上的编码器，只完成编码功能。码率控制器提供了对编码参数的控制和调整，从而对输出码率进行控制。

图6-2 编码通道功能模块划分



编码通道可分为主编码通道和次编码通道。对于一帧输入图像，VENC 仅支持一路主编码通道，不支持次编码通道。

编码通道组是指芯片启动一次，能够同时处理的编码通道的集合。VENC 编码通道组支持启动一次，仅编码一路主码流通道（H.264/JPEG/MJPEG/MPEG-4）。所以在创建编码通道时，只允许注册一个主码流通道。

6.2.3 码率控制

码率控制器实现对编码码率进行控制。

从信息学的角度分析，图像的压缩比越低，压缩图像的质量越高；图像压缩比例越高，压缩图像的质量越低。对于场景变化的真实场景，图像质量稳定，编码码率会波动；编码码率稳定，图像质量会波动。以 H.264 编码为例，通常图像 Qp 越低，图像的质量越好，码率越高；图像 Qp 越高，图像质量越差，码率越低。

码率控制必须是针对连续的编码码流而言，所以，JPEG 协议编码通道不包括码率控制功能。

码率控制器分别提供了对 H.264\MPEG-4\MJPEG 协议编码通道 CBR、VBR、FIXQP 等三种码率控制模式，对图像质量和码率进行调节。

Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C 不支持 MPEG-4 编码，所以，也不支持 MPEG-4 类型的码率控制。

CBR

CBR（Constant Bit Rate）固定比特率。即在码率统计时间内保证编码码率平稳。码率稳定主要由两个量来评估，这两个量都可以由用户在创建编码通道时指定。

- 码率统计时间

  单位为秒(s)，码率统计时间越长，每帧图像的码率波动对于码率调节的影响越弱，码率的调节会更缓慢，图像质量的波动会更轻微；码率统计时间越短，每帧图像的码率波动对于码率调节的影响越强，图像码率的调节会更灵敏，图像质量的波动会更剧烈。

- 码率的波动等级

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

系统提供了 6 阶的图像质量波动等级，波动等级越大，系统允许码率的波动范围更大。如果波动等级设置高，对于一些图像复杂，变化剧烈的场景，图像质量可能会更平稳，适用于网络带宽富裕的场景；如果波动等级设置低，编码的码率会比较平稳，对于一些图像复杂，变化剧烈的场景，图像质量可能不如高波动等级，适用于带宽不富裕的场景。

VBR

VBR（Variable Bit Rate）可变比特率，即在码率统计时间内编码码率波动，从而保证编码图像质量平稳。以 H.264 编码为例，VENC 模块提供用户可设置 MaxQp，MinQp，MaxBitrate。MaxQp，MinQp 用于控制图像的质量范围，MaxBitrate 用于钳位码率统计时间内的最大编码码率。当编码码率接近最大码率时，图像 QP 会逐步向 MaxQp 调整，当编码码率远小于最大码率时，图像 QP 会逐步向 MinQp 调整。当图像 QP 达到 MaxQp 时，QP 被钳位到最大值，MaxBirate 的钳位效果失效，编码码率有可能会超出 MaxBirate。当图像 QP 达到 MinQp 时，QP 被钳位到最小值，此时编码的码率已经达到最大值，而且图像质量最好。

FIXQP

Fix Qp 固定 Qp 值。在码率统计时间内，编码图像所有宏块 Qp 值相同，采用用户设定的图像 Qp 值，I 帧和 P 帧的 QP 值可以分别设置。

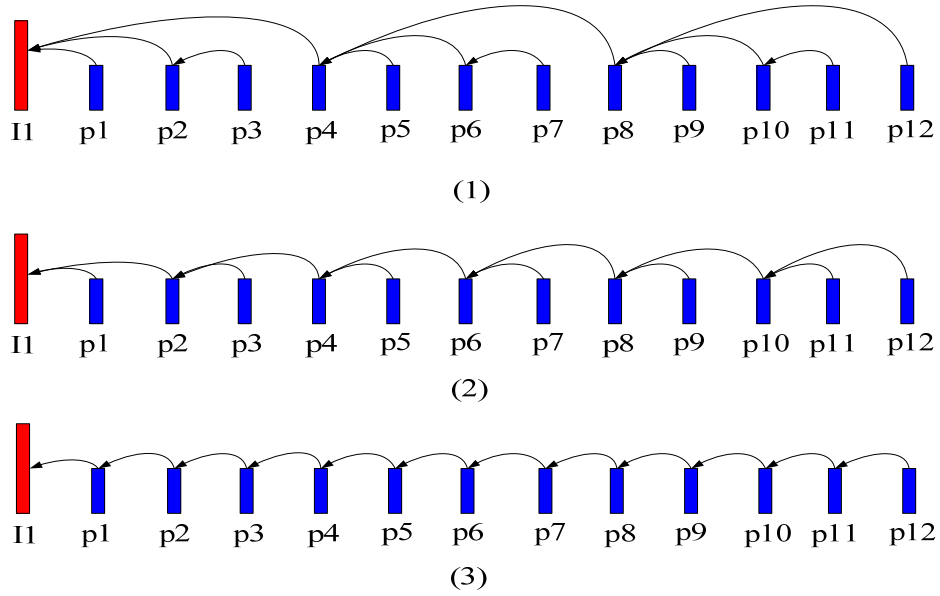
6.2.4 跳帧参考

H.264 编码提供 3 种跳帧参考模式：1 倍跳帧参考、2 倍跳帧参考、4 倍跳帧参考。3 种跳帧参考模式的示意如图 6-3 所示。

- 图中的（1）小图是 4 倍跳帧参考示意图。I1 是 I 帧，p1,p2…p12 都是 P 帧。图中有箭头指入的帧，如 I1,p2,p4 等为参考帧，没有箭头指入的帧，如 p1,p3，是非参考帧。用户在保存码流时，可以只保存 4 倍跳帧参考的参考帧，如 I1,p4,p8；也可以保存所有参考帧，甚至所有的帧。
- 图中的（2）小图是 2 倍跳帧参考示意图。图中有箭头指入的帧，如 I1,p2,p4 等为参考帧，没有箭头指入的帧，如 p1,p3，是非参考帧。用户在保存码流的时候可以保存所有参考帧，也可以保存所有的帧。
- 图中的（3）小图是 1 倍跳帧参考示意图。所有的帧都是参考帧，用户在保存码流的时候需要保存所有参考帧。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图6-3 跳帧参考示意图

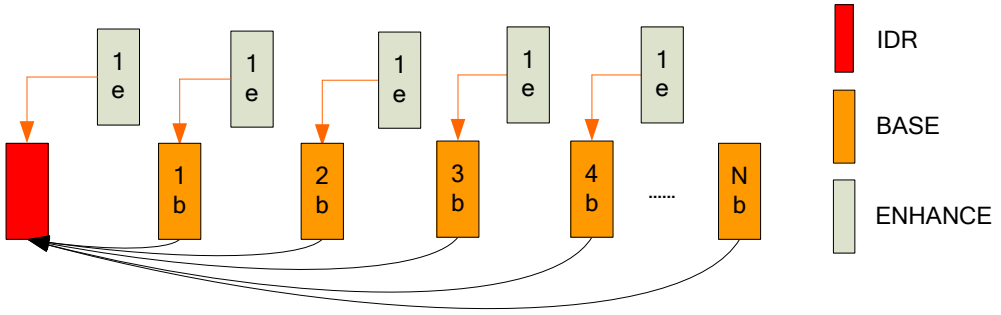


6.2.5 高级跳帧参考模式

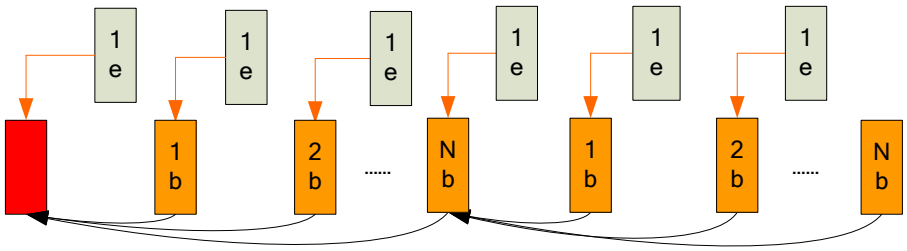
高级跳帧参考模式涉及 3 个参数：u32Base、u32Enhance 和 bEnablePred，其含义请参考 VENC_ATTR_H264_REF_PARAM_S。高级跳帧参考模式可以兼容 6.2.4 跳帧参考的三种参考模式。高级跳帧参考模式示意如图 6-4 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

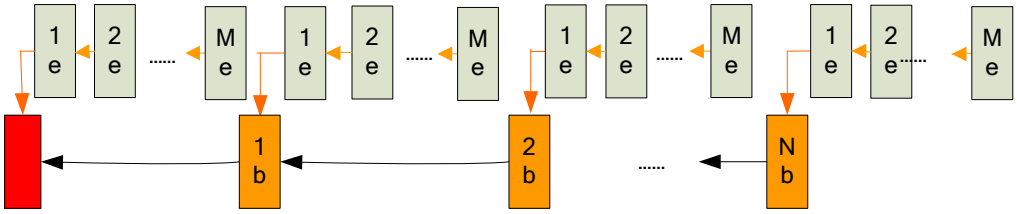
图6-4 高级跳帧参考示意图



(N,1,HI_FALSE)



(N,1,HI_TRUE)



(1,M,HI_TRUE)

6.2.6 彩转灰

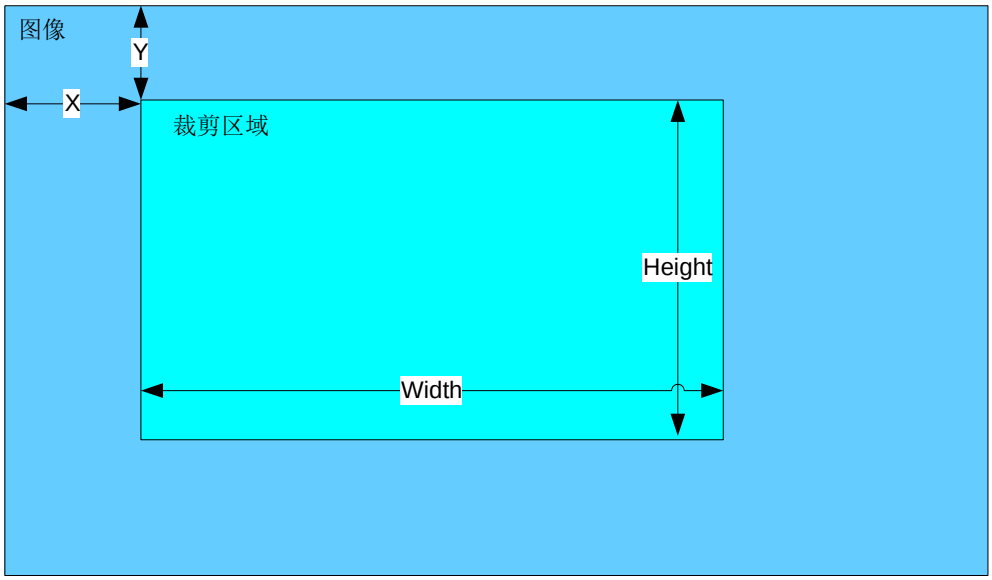
彩转灰，即 VENC 支持把彩色图像转换成灰度图像进行编码。具体功能请参考相关 API：HI_MPI_VENC_SetColor2GreyConf，HI_MPI_VENC_SetGrpColor2Grey 等。

6.2.7 裁剪编码

裁剪编码，即 VENC 从图像中裁剪出一部分进行编码，用户可以设置裁剪的起始点 X、Y 和裁剪的宽度 width 和高度 height，具体功能请参考相关 API：HI_MPI_VENC_GetGrpCrop，HI_MPI_VENC_SetGrpCrop 等。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图6-5 裁剪编码示意图



6.2.8 ROI

ROI（Region Of Interest）编码，感兴趣区域编码。

用户可以通过配置 ROI 区域，对该区域的图像 Qp 进行限制，从而实现图像中该区域的 Qp 与其他图像区域的差异化。系统现仅支持对 H.264 通道进行 ROI 设置。系统提供了 8 个感兴趣区域，可供用户同时使用。

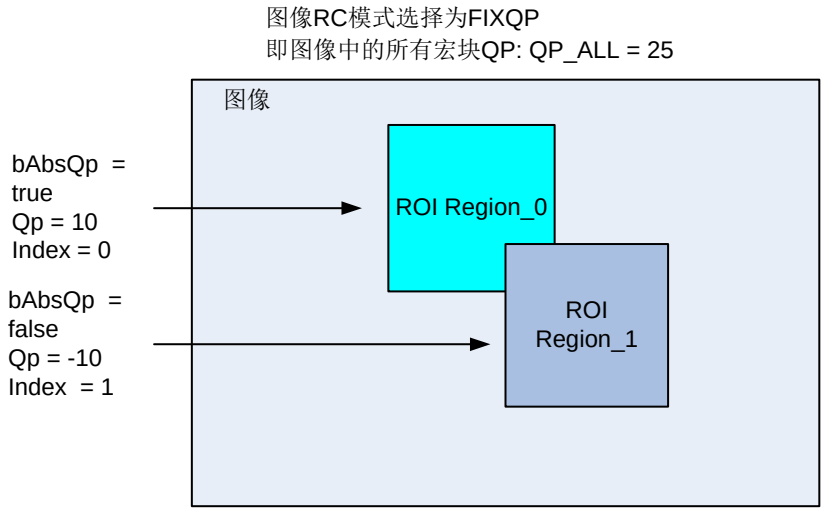
8 个区域可以互相叠加，且叠加时的优先级按照 0～7 的索引号依次提高，这里，叠加优先级是指发生叠加时，图像区域的最终 Qp 值的判定，最终的区域 Qp 值按照优先级最高的区域设定。ROI 区域可配置绝对 Qp 和相对 Qp 两种模式。

- 绝对 Qp：ROI 区域的 Qp 为用户设定的 Qp 值。
- 相对 Qp：ROI 区域的 Qp 为码率控制产生的 Qp 与用户设定的 Qp 偏移值的和。

以下示例编码图像采用 FixQp 模式，设置图像 Qp 为 25，即图像中所有宏块 Qp 值为 25。Roi 区域 0 设置为绝对 Qp 模式，Qp 值为 10，索引为 0；Roi 区域 1 设置为相对 Qp 模式，Qp 为-10，索引为 1。区域 0 的 index 小于区域 1 的 index，所以在发生互相重叠的图像区域按高优先级的区域（区域 1）Qp 设置。区域 0 除了发生重叠的部分的 Qp 值等于 10。区域 1 的 Qp 值为 25-10=15。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图6-6 ROI 示意图



6.2.9 JPEG 抓拍模式

Hi3518/Hi3516C 的 JPEG 编码支持两种工作模式：全部抓拍模式和闪光灯抓拍模式。Hi3531/Hi3532/Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C 等芯片的 JPEG 编码只支持全部抓拍模式。

- 全部抓拍模式：通道启动接收图像后，编码所有接收的图像。
- 闪光灯抓拍模式：通道启动接收图像后，只编码在闪光灯亮时采集到的图像。

说明
当 Hi3518/Hi3516C 的 JPEG 设置为闪光灯抓拍模式时，通道组只能接收来自 VIU 模块（不经过 VPSS 模块）的图像，否则 JPEG 抓拍不到任何图像。

6.2.10 编码码流 buffer 配置模式

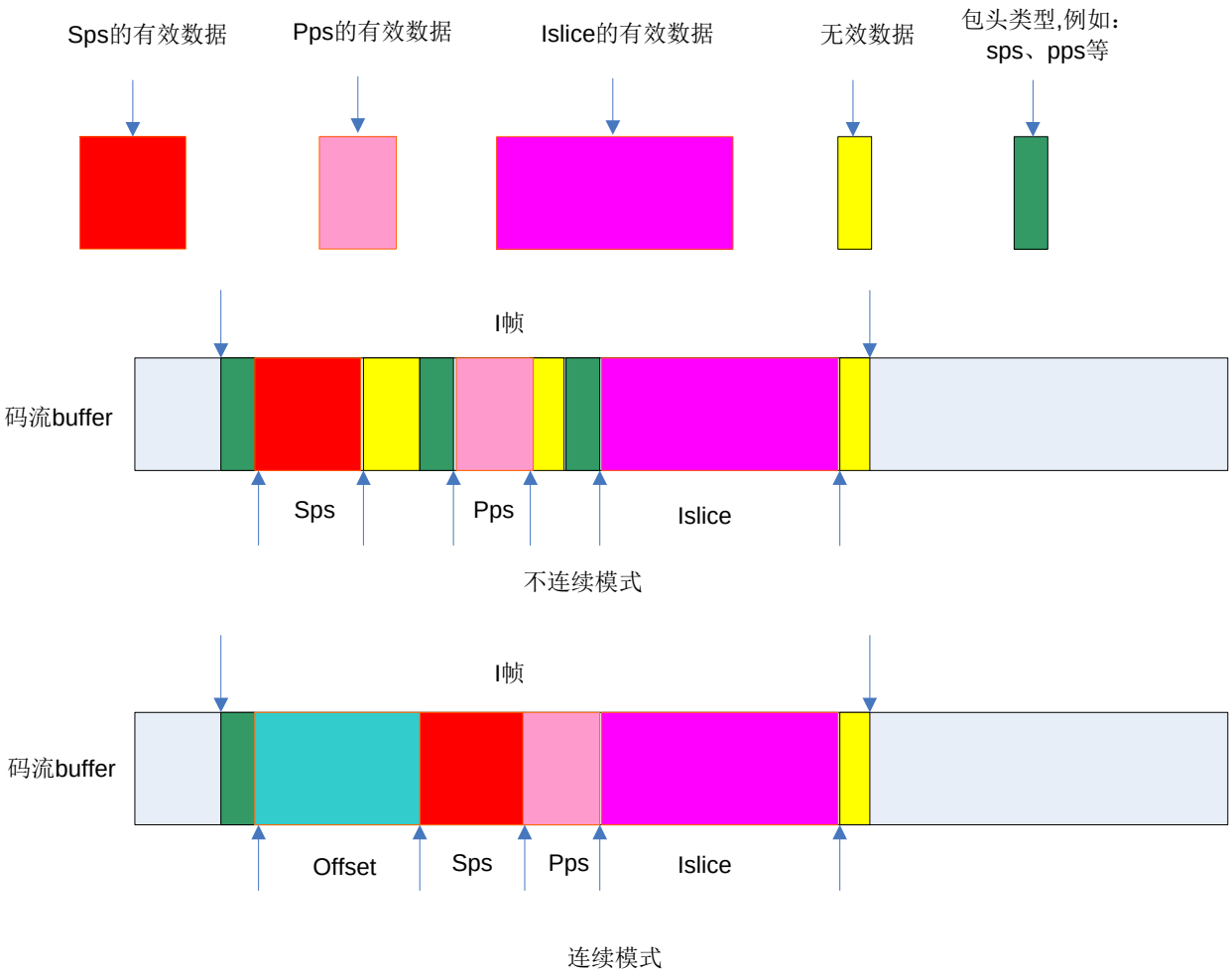
编码码流 buffer 配置支持两种模式：非连续模式和连续模式，如图 6-7 所示。

- 非连续模式：当为 I 帧时，调用 HI_MPI_GetStream（）接口，一帧 I 帧包含 3 个包（3 个包为 sps 包、pps 包、Islice 包，这里假设 pps 包只有一个），且包与包之间的地址并不连续，在码流 buffer 回绕时，一帧会有两个地址且不连续，即 pstPack[0].pu8Addr[0] + pstPack[0].u32Len[0] ≠ pstPack[0].pu8Addr[1]。
- 连续模式：当为 I 帧时，调用 HI_MPI_GetStream（）接口，一帧 I 帧包含 1 个包（1 个包为 Islice 包），且一帧地址连续，即在任何时候，pstPack[0].pu8Addr[0] + pstPack[0]. u32Len[0] = pstPack[0].pu8Addr[1]。

两种模式可通过 ko 加载时设置模块参数 ContiguousBuffer 来选择。ContiguousBuffer=1 表示连续模式；ContiguousBuffer=0 表示非连续模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图6-7 编码码流 buffer 配置模式



6.2.11 编码帧存分配方式

编码帧存（参考帧和重构帧）分配支持两种方式：PrivateVB 池方式和 UserVB 池方式。

- 编码 PrivateVB 池方式：创建编码通道时由 VENC 创建私有 VB 池作为该通道的参考帧和重构帧 buffer。
- 编码 UserVB 池方式（仅 H264 编码支持）：创建编码通道时不分配参考帧和重构帧 buffer，而是由用户调用接口 HI_MPI_VB_CreatePool 创建一个视频缓存 VB 池，再通过调用接口 HI_MPI_VENC_AttachVbPool 把某个编码通道绑定到固定的视频缓存 VB 池中。

两种方式可通过加载 h264e.ko 时设置模块参数 H264eVBSource 来选择，默认使用 PrivateVB 池方式。H264eVBSource=1 表示使用编码 PrivateVB 池方式；H264eVBSource=2 表示使用编码 UserVB 池方式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

使用 PrivateVB 方式时每个编码通道之间没有关联，单独销毁某个通道对其它通道无影响，使用灵活。

表6-2 n 个通道编码使用 UserVB 方式时节省帧存数

芯片	VEDU 个数	节省帧存数
Hi3531、Hi3532	2	n – 4 （n > 4）
Hi3521、Hi3520A、Hi3520D、Hi3515A、Hi3515C	1	n – 2 （n > 2）

如果使用 UserVB 方式，需要先创建一个 VB 池，再把 n 个通道绑定到这个 VB 池中。

VB 块个数的计算方法是：

BlkCnt =n + 2 * VEDU；（如果有通道设置 4 倍跳帧参考、高级跳帧参考或者 svc-t 编码，则该通道需要多增加一个帧存）

每个 VB 块大小的计算方法是：

BlkSize = align(PicWidth,16)*align(PicHeight,16)*3/2

6.2.12 编码重构帧复用参考帧亮度内存模式

编码重构帧可以复用参考帧亮度内存，色度内存重构帧和参考帧都需要单独分配。可在加载 h264e.ko 时设置模块参数 H264eRcnEqualRef 来选择，H264eRcnEqualRef=0 为默认的重构帧、参考帧单独分配内存模式；H264eRcnEqualRef=1 为重构帧复用参考帧内存模式。

重构帧复用参考帧亮度内存带来的问题有：

a. 因为在编码过程中参考帧已被重构帧覆盖，因此在异常情况下（比如超大帧、码率过冲、码流 buffer 满等）只能插入 I 帧。

b. 2 倍跳帧参考编码比使用一般模式多占用一个亮度大小的内存。

如果同时配置 H264eVBSource=2 和 H264eRcnEqualRef=1，调用方法与 6.2.11 节描述的方法基本相同，区别是每个 VB 块的大小计算方法修改为：

Size = align(PicWidth,16)*align(PicHeight,16) / 2

表6-3 H264eVBSource 和 H264eRcnEqualRef 组合使用说明

H264eVBSource	H264eRcnEqualRef	用法说明
1	0	加载 ko 时直接设置模块参数即可
1	1	加载 ko 时直接设置模块参数即可
2	0	（1）加载 ko 时设置模块参数； （2）创建 VB 池，VB 个数和大小参照前面描述内容；

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

H264eVBSource	H264eRcnEqualRef	用法说明
		（3）将编码通道绑定到 VB 池中。
2	1	（1）加载 ko 时设置模块参数； （2）创建 VB 池，VB 个数和大小参照前面描述内容； （3）将编码通道绑定到 VB 池中。

6.2.13 编码码流 buffer 配置模式

编码码流 buffer 配置支持两种模式：一般模式和省内存模式。

- 一般模式：考虑到超大帧的情况，码流 Buffer 大小配置的下限为：H264 和 MPEG4 是通道宽 x 通道高 x3/4，JPEG 和 MJPEG 是通道宽 x 通道高。
- 省内存模式：码流 Buffer 大小配置的下限是 32*1024 bytes，此模式需要用户保证码流 buffer 大小设置合理，否则会出现因码流 buffer 不足而不断重编或者丢帧的情况。

两种模式可通过 ko 加载时设置相应的模块参数来选择。模块参数值为 1 表示省内存模式，模块参数值 0 表示一般模式。

hi35xx_h264e.ko 模块参数：H264eMiniBufMode；

hi35xx_mpeg4e.ko 模块参数：Mpeg4eMiniBufMode；

hi35xx_jpege.ko 模块参数：JpegeMiniBufMode；

6.3 API 参考

视频编码模块包含 VENC（视频编码）和 GROUP（通道组管理）两个重要的部分，主要提供视频编码通道组的创建和销毁、视频编码通道的创建和销毁、视频编码通道的注册和注销、开启和停止接收图像、设置和获取编码通道属性、获取和释放码流、视频编码通道属性的设置和查询等功能。

该功能模块提供以下 MPI：

- HI_MPI_VENC_CreateGroup：创建编码通道组。
- HI_MPI_VENC_DestroyGroup：销毁编码通道组。
- HI_MPI_VENC_CreateChn：创建编码通道。
- HI_MPI_VENC_DestroyChn：销毁编码通道。
- HI_MPI_VENC_RegisterChn：注册编码通道到通道组。
- HI_MPI_VENC_UnRegisterChn：反注册编码通道到通道组。
- HI_MPI_VENC_StartRecvPic：开启编码通道接收输入图像。
- HI_MPI_VENC_StartRecvPicEx：开启编码通道接收输入图像，超出指定的帧数后自动停止接收图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VENC_StopRecvPic：停止编码通道接收输入图像。
- HI_MPI_VENC_Query：查询编码通道状态。
- HI_MPI_VENC_SetChnAttr 设置编码通道的编码属性。
- HI_MPI_VENC_GetChnAttr：获取编码通道的编码属性。
- HI_MPI_VENC_GetStream：获取编码码流。
- HI_MPI_VENC_ReleaseStream：释放码流缓存。
- HI_MPI_VENC_InsertUserData：插入用户数据。
- HI_MPI_VENC_SendFrame：支持用户发送原始图像进行编码。
- HI_MPI_VENC_SetMaxStreamCnt：设置最大码流缓存帧数。
- HI_MPI_VENC_GetMaxStreamCnt：获取最大码流缓存帧数。
- HI_MPI_VENC_RequestIDR：请求 I 帧。
- HI_MPI_VENC_RequestIDRInst：请求 I 帧并立即生效。
- HI_MPI_VENC_GetFd：获取编码通道对应的设备文件句柄。
- HI_MPI_VENC_SetRoiCfg：设置编码通道的感兴趣区域编码配置。
- HI_MPI_VENC_GetRoiCfg：获取编码通道的感兴趣区域编码配置。
- HI_MPI_VENC_SetH264SliceSplit：设置 H.264 编码的 slice 分割配置。
- HI_MPI_VENC_GetH264SliceSplit：获取 H.264 编码的 slice 分割配置。
- HI_MPI_VENC_SetH264InterPred：设置 H.264 编码的帧间预测配置。
- HI_MPI_VENC_GetH264InterPred：获取 H.264 编码的帧间预测配置。
- HI_MPI_VENC_SetH264IntraPred：设置 H.264 编码的帧内预测配置。
- HI_MPI_VENC_GetH264IntraPred：获取 H.264 编码的帧内预测配置。
- HI_MPI_VENC_SetH264Trans：设置 H.264 编码的变换、量化配置。
- HI_MPI_VENC_GetH264Trans：获取 H.264 编码的变换、量化配置。
- HI_MPI_VENC_SetH264Entropy：设置 H.264 编码的熵编码配置。
- HI_MPI_VENC_GetH264Entropy：获取 H.264 编码的熵编码配置。
- HI_MPI_VENC_SetH264Poc：设置 H.264 编码的 POC 类型。
- HI_MPI_VENC_GetH264Poc：获取 H.264 编码的 POC 类型。
- HI_MPI_VENC_SetH264Dblk：设置 H.264 编码的 deblocking 配置。
- HI_MPI_VENC_GetH264Dblk：获取 H.264 编码的 deblocking 配置。
- HI_MPI_VENC_SetH264Vui：设置 H.264 编码的 VUI 配置。
- HI_MPI_VENC_GetH264Vui：获取 H.264 编码的 VUI 配置。
- HI_MPI_VENC_SetJpegParam：设置 JPEG 编码的参数集合。
- HI_MPI_VENC_GetJpegParam：获取 JPEG 编码的参数集合。
- HI_MPI_VENC_SetMpeg4Param：设置 MPEG-4 编码的参数集合。
- HI_MPI_VENC_GetMpeg4Param：获取 MPEG-4 编码的参数集合。
- HI_MPI_VENC_SetMjpegParam：设置 JPEG 协议编码通道的高级参数。
- HI_MPI_VENC_GetMjpegParam：获取 JPEG 协议编码通道的高级参数。
- HI_MPI_VENC_SetGrpFrmRate：设置 GROUP 的帧率控制参数集合。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VENC_GetGrpFrmRate：获取 GROUP 的帧率控制参数集合。
- HI_MPI_VENC_SetRcPara：设置通道码率控制高级参数。
- HI_MPI_VENC_GetRcPara：获取通道码率控制高级参数。
- HI_MPI_VENC_SetH264eRefMode：设置 H.264 编码通道跳帧参考模式。
- HI_MPI_VENC_GetH264eRefMode：获取 H.264 编码通道跳帧参考模式。
- HI_MPI_VENC_SetH264eRefParam：设置 H.264 编码通道高级跳帧参考参数。
- HI_MPI_VENC_GetH264eRefParam：获取 H.264 编码通道高级跳帧参考参数。
- HI_MPI_VENC_SetColor2GreyConf：设置彩转灰功能的全局配置。
- HI_MPI_VENC_GetColor2GreyConf：获取彩转灰功能的全局配置。
- HI_MPI_VENC_SetGrpColor2Grey：开启或关闭一个 GROUP 的彩转灰功能。
- HI_MPI_VENC_GetGrpColor2Grey：获取一个 GROUP 是否开启彩转灰功能。
- HI_MPI_VENC_SetGrpCrop：设置 GROUP 的裁剪属性。
- HI_MPI_VENC_GetGrpCrop：获取 GROUP 的裁剪属性。
- HI_MPI_VENC_SetJpegSnapMode：设置 JPEG 抓拍通道的抓拍模式。
- HI_MPI_VENC_GetJpegSnapMode：获取 JPEG 抓拍通道的抓拍模式。
- HI_MPI_VENC_SetRcPriority：设置码率控制的优先级类型。
- HI_MPI_VENC_GetRcPriority：获取码流控制优先级类型。
- HI_MPI_VENC_SetLostFrameStrategy：设置编码丢帧策略。
- HI_MPI_VENC_GetLostFrameStrategy：获取编码丢帧策略。
- HI_MPI_VENC_AttachVbPool：将编码通道绑定到某个视频缓存 VB 池中。
- HI_MPI_VENC_DetachVbPool：将编码通道从某个视频缓存 VB 池中解绑定。
- HI_MPI_VENC_SetH264VideoSignal：设置 H.264 编码的 VideoSignal 配置。
- HI_MPI_VENC_GetH264VideoSignal：获取 H.264 编码的 VideoSignal 配置。

HI_MPI_VENC_CreateGroup

【描述】

创建编码通道组。

【语法】

```
HI_S32 HI_MPI_VENC_CreateGroup(VENC_GRP VeGroup);
```

【参数】

参数名称	描述	输入/输出
VeGroup	编码通道组号。 取值范围：[0, VENC_MAX_GRP_NUM)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本文档中含有通道组号的接口的通道组号的取值范围为[0, VENC_MAX_GRP_NUM)。
- 编码通道组是指芯片能够同时处理的编码通道的集合，一个通道组最多可同时包含 1 路主码流（H.264/MJPEG/MPEG-4），或者包含 1 路 JPEG 抓拍。
- 如果指定的通道组已经存在，返回失败。

【举例】

```
HI_S32 StartVenc(HI_VOID)
{
    HI_S32 s32Ret;
    VI_CHN ViChn = 0;
    VENC_GRP VeGroup = 0;
    VENC_CHN VeChn = 0;
    VENC_CHN_ATTR_S stAttr;
    MPP_CHN_S stSrcChn, stDestChn;
    /* set h264 chnnel video encode attribute */
    stAttr.stVeAttr.enType   = PT_H264;
    stAttr.stVeAttr.stAttrH264e.u32PicWidth = u32PicWidth;
    stAttr.stVeAttr.stAttrH264e.u32PicHeight = u32PicHeigh;
    stAttr.stVeAttr.stAttrH264e.u32MaxPicWidth = u32MaxPicWidth;
    stAttr.stVeAttr.stAttrH264e.u32MaxPicHeight = u32MaxPicHeigh;

    stAttr.stVeAttr.stAttrH264e.u32Profile = 2;
    …… // omit other video encode assignments here.
    /* set h264 chnnel rate control attribute */
    stAttr.stRcAttr.enRcMode          =  VENC_RC_MODE_H264CBR ;
    stAttr.stRcAttr.stAttrH264Cbr.u32BitRate            = 10*1024;
    stAttr.stRcAttr.stAttrH264Cbr.fr32TargetFrmRate     = 30;
    stAttr.stRcAttr.stAttrH264Cbr.u32ViFrmRate          = 30;
    stAttr.stRcAttr.stAttrH264Cbr.u32Gop                = 30;
    stAttr.stRcAttr.stAttrH264Cbr.u32FluctuateLevel       = 1;
    stAttr.stRcAttr.stAttrH264Cbr.u32StatTime             = 1;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    …… // omit other rate control assignments here.

    s32Ret = HI_MPI_VENC_CreateGroup(VeGroup);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_CreateGroup err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VENC_CreateChn(VeChn, &stAttr);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_CreateChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VENC_RegisterChn(VeGroup, VeChn);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_RegisterChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    s32Ret = HI_MPI_VENC_StartRecvPic(VeChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_StartRecvPic err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    …… // omit other code here.
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_DestroyGroup

【描述】

销毁编码通道组。

【语法】

```
HI_S32 HI_MPI_VENC_DestroyGroup(VENC_GRP VeGroup);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeGroup	编码通道组号。 取值范围：[0, VENC_MAX_GRP_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 销毁通道组时，必须保证通道组为空，即没有任何通道在通道组中注册，或注册到通道组中的通道已经反注册，否则会返回失败。
- 销毁并不存在的通道组，返回失败。

【举例】

```
HI_S32 StopVenc(HI_VOID)
{
    HI_S32 s32Ret;
    VENC_CHN VeChn = 0;
    VENC_GRP VeGroup = 0;

    s32Ret = HI_MPI_VENC_StopRecvPic(VeChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_StopRecvPic err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    s32Ret = HI_MPI_VENC_UnRegisterChn(VeChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_UnRegisterChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VENC_DestroyChn(VeChn);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_DestroyChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    s32Ret =HI_MPI_VENC_DestroyGroup(VeGroup);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_DestroyGroup err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_CreateChn

【描述】

创建编码通道。

【语法】

```
HI_S32 HI_MPI_VENC_CreateChn(VENC_CHN VeChn, const VENC_CHN_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstAttr	编码通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_venc.h、hi_comm_rc.h、mpi_venc.h
- 库文件：libmpi.a

【芯片差异】

- Hi3531/Hi3532 支持 H.264 high profile(HP)编码，支持 MPEG-4 Simple profile 编码。
- Hi3521/Hi3520A 不支持 H.264 high profile(HP)编码，不支持 MPEG-4 编码。
- Hi3518/Hi3516C 支持 H.264 high profile(HP)编码，但是在 HP 编码时不支持 8x8 变换及量化表配置。
- Hi3518/Hi3516C 支持 SVC-T profile 编码，在 SVC profile 下不支持跳帧参考接口设置。
- Hi3520D/Hi3515A/Hi3515C 支持 H.264 HighProfile 编码，不支持 MPEG-4 编码。
- Hi3520D 最多支持 64 个 H.264 编码通道，Hi3515A/Hi3515C 最多支持 8 个 H.264 编码通道，且最多支持 4 个分辨率大于等于 D1 分辨率的通道，其他通道分辨率小于等于 CIF。
- RC 属性参数的详细使用说明请参见《码率控制使用说明》。
- Hi35xx 支持编码通道宽高差异如表 6-4 所示。

表6-4 芯片相关编码通道宽高差异

芯片	W/H（pixel）	H.264	JPEG	MOTION JPEG	MPEG-4
Hi3531/Hi3532	MIN_WIDTH	160	32	32	160
	MAX_WIDTH	7040	8192	8192	1920
	MIN_HEIGHT	64	32	32	64
	MAX_HEIGHT	8176	8192	8192	1088
	MIN_ALIGN	2	4	4	2
Hi3521/Hi3520A/Hi3518/Hi3516C	MIN_WIDTH	160	64	64	不支持
	MAX_WIDTH	1920	8192	8192	不支持
	MIN_HEIGHT	64	64	64	不支持
	MAX_HEIGHT	2048	8192	8192	不支持
	MIN_ALIGN	2	4	4	不支持
Hi3520D/Hi3515A/Hi3515C	MIN_WIDTH	160	64	64	不支持
	MAX_WIDTH	1920	8192	8192	不支持
	MIN_HEIGHT	64	64	64	不支持
	MAX_HEIGHT	2048	8192	8192	不支持
	MIN_ALIGN	2	4	4	不支持

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 编码通道属性由两部分组成，编码器属性和码率控制器属性。
- 编码器属性首先需要选择编码协议，然后分别对各种协议对应的属性进行赋值。
- 编码器属性不支持次码流，必须将编码器码流类型设置为主码流类型。
- 编码器属性不支持场编码模式，必须将通道编码类型设置为帧编码模式。
- 编码器属性最大宽高，通道宽高必须满足如下约束：
  - MaxPicWidth∈[MIN_WIDTH, MAX_WIDTH]
  - MaxPicHeight∈[MIN_HEIGHT, MAX_HEIGHT]
  - PicWidth∈[MIN_WIDTH, MaxPicwidth]
  - PicHeight∈[MIN_HEIGHT, MaxPicHeight]
  - 最大宽高，通道宽高必须是 MIN_ALIGN 的整数倍。

  其中 MIN_WIDTH，MAX_WIDTH，MIN_HEIGHT，MAX_HEIGHT，MIN_ALIGN 分别表示编码通道支持的最小宽度，最大宽度，最小高度，最大高度，最小对齐单元（像素）。编码器属性中的 u32MaxPicWidth 和 u32MaxPicHeight 用于分配内存，实际编码码流分辨率大小由 u32PicWidth 和 u32PicHeight 指定。
- 编码器属性必须设置编码码流 buffer 深度，获取码流方式等，表 6-5 详细描述了各种协议的各项属性的特性。

表6-5 编码器属性的约束

编码协议	主次码流	编码方式	码流 buffer 深度	获取码流模式	编码 profile
H.264	主码流	Frame	Buffer≥ MaxPicwidth x MaxPicheight x 3/4	Frame/ Slice	Baseline Mainprofile Highprofile SVC profile
MJPEG	主码流	Frame	Buffer≥ MaxPicwidth x MaxPicheight x2	Frame/Ecs	-
JPEG	主码流	Frame	Buffer≥ MaxPicwidth x MaxPicheight x 2	Frame/Ecs	Baseline
MPEG-4	主码流	Frame	Buffer≥MaxPicwidth x MaxPicheight x 3/4	Frame/ Packet	Simple profile

- 当输入图像大小不大于通道编码图像最大宽高时，才能启动编码通道进行编码。
- 推荐的编码宽高为：2592x1944（5M 像素）、1920x1080（1080P）、1280x720（720P）、960x540、640x360、704x576、704x480、352x288、352x240。
- 编码器属性中除了 u32PicWidth 和 u32PicHeight 外，其它所有项都是静态属性，一旦创建编码通道成功，静态属性不支持被修改，除非该通道被销毁，重新创建。设置时需要注意的事项请参考 HI_MPI_VENC_SetChnAttr 接口说明。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 码率控制器属性首先需要配置 RC 模式，JPEG 抓拍通道不需要配置码率控制器属性，其他协议类型通道（H.264/MJPEG/MPEG-4）都必须配置。码率控制器属性 RC 模式必须与编码器属性协议类型匹配。
- 三种编码协议（H.264/MJPEG/MPEG-4），码率控制均支持三种模式：CBR，VBR，FIXQP。并且对于不同协议，相同 RC 模式的属性变量基本一致。表 6-6 介绍了三种 RC 模式的公共的属性。

表6-6 码率控制器属性的约束

RC 模式	GOP	StatTime(s)	FrmRate(ViFrmRate/TargetFrmRate)
CBR	≥1	≥1	ViFrmRate ≥ TargetFrmRate
VBR	≥1	≥1	ViFrmRate ≥ TargetFrmRate
FIXQP	≥1	≥1	ViFrmRate ≥ TargetFrmRate

- ViFrmRate 应该设置为输入编码器的实际帧率，RC 需要根据 ViFrmRate 统计实际的帧率以及进行码率控制。假设 VI 的输出帧率是 30，如果 GROUP 不进行帧率控制，ViFrmRate 应该设置为 30，如果 GROUP 进行帧率控制，GROUP 帧率控制中的源帧率和 ViFrmRate 都应该设置为 30。设置目标帧率 TargetFrmRate 时，目标帧率类型定义为分数类型 HI_FR32 类型。HI_FR32 实际为 HI_U32 型，高 16 位用于表示分母，低 16 位表示分子。如果用户设置目标帧率为整数，设置分母高 16 位为 0 即可表示整数。例如设置 ViFrmRate 等于 25，TargetFrmRate 等于 12，则表示将从 25 帧输入图像中取出 12 帧进行编码，其余 13 帧将丢掉。如果设置 ViFrmRate 等于 25，TargetFrmRate 等于 15/2，则表示要求编码器 2 秒钟编码 15 帧。
- TargetFrmRate 的设置方式如下：如果要设置为整数帧率 25，则可采用以下方式 TargetFrmRate = 25；如果要设置为分数帧率 15/2，则采用以下方式：TargetFrmRate = 15+（2<<16）。
- CBR 除了上述的属性之外，还需要设置平均比特率和波动等级。平均比特率的单位是 kbps。平均比特率的设置与编码通道宽高以及图像帧率都有关系。典型的平均比特率的设置如下表所示。注意，下表中的平均比特率的设置是在通道编码帧率为满帧率（30fps）时的设置。当用户设置编码输出帧率不为满帧率时，可以对下表中的码率按用户设置帧率与满帧率（30fps）的比例进行换算。

表6-7 典型平均比特率配置

图像宽高/码率等级	D1(720x576)	720p(1280x720)	1080p(1920x1080)
低码率	<400kbps	<800kbps	<2000kbps
中码率	400～1000Kbps	800kbps～4000Kpbs	2000～8000Kbps
高码率	>1000Kbps	>4000Kbps	>8000Kbps

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 波动等级设置分为 6 档，波动等级越大，系统允许码率的波动范围更大。如果波动等级设置高，对于一些图像复杂，变化剧烈的场景，图像质量可能会更平稳，适用于网络带宽富裕的场景；如果波动等级设置低，编码的码率会比较平稳，对于一些图像复杂，变化剧烈的场景，图像质量可能不如高波动等级，适用于带宽不富裕的场景。
- VBR 除了上述属性之外，还需要设置 MaxBitRate，MaxQp，MinQp。MaxBitRate：编码通道在码率统计时间内允许的最大码率。MaxQp：图像允许的最大 QP。MinQp：图像允许的最小 QP。
- FIXQP 除了上述属性之外，还需要设置 IQp，PQp。IQp：I 帧时，图像固定使用的 QP 值。PQp：P 帧时，图像固定使用的 QP 值。在设置 I 帧 QP，P 帧 QP 时，可以根据当前的带宽限制对 I 帧 QP 和 P 帧 QP 同时进行向上或是向下调整。为了减少呼吸效应，推荐 I 帧 QP 始终比 P 帧的 QP 小 2～3。

【举例】

请参见 HI_MPI_VENC_CreateGroup 的举例。

【相关主题】

无。

HI_MPI_VENC_DestroyChn

【描述】

销毁编码通道。

【语法】

```
HI_S32 HI_MPI_VENC_DestroyChn(VENC_CHN VeChn);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 销毁并不存在的通道，返回失败。
- 销毁前必须保证通道已经从通道组反注册，否则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_RegisterChn

【描述】

注册编码通道到通道组。

【语法】

```
HI_S32 HI_MPI_VENC_RegisterChn(VENC_GRP VeGroup,VENC_CHN VeChn);
```

【参数】

参数名称	描述	输入/输出
VeGroup	通道组号。 取值范围：[0, VENC_MAX_GRP_NUM)。	输入
VeChn	通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 注册并不存在的通道，返回失败。
- 注册通道到不存在的通道组，返回失败。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 同一个编码通道只能注册到一个通道组，如果该通道已经注册到某个通道组，则返回失败。
- 如果一个通道组已经被注册，那么这个通道组就不能再被其他的通道注册，除非之前的注册关系被解除。

【举例】

请参见 HI_MPI_VENC_CreateGroup 的举例。

【相关主题】

无。

HI_MPI_VENC_UnRegisterChn

【描述】

反注册编码通道到通道组。

【语法】

```
HI_S32 HI_MPI_VENC_UnRegisterChn(VENC_CHN VeChn);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 注销未创建的通道，则返回失败。
- 注销未注册的通道，则返回失败。
- 如果编码通道未停止接收图像编码（HI_MPI_VENC_StopRecvPic 可停止接收），则返回失败。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 通道注销之后，编码通道会被复位，编码通道里的码流 buffer 都会被清空，如果用户还在使用未及时释放的码流 buffer，将不能保证此 buffer 数据的正确性。用户可以使用 HI_MPI_VENC_Query 接口来查询编码通道码流 buffer 状态，确认码流 buffer 里的码流都取完之后再反注册通道。

【举例】

请参见 HI_MPI_VENC_DestroyGroup 的举例。

【相关主题】

无。

HI_MPI_VENC_StartRecvPic

【描述】

开启编码通道接收输入图像。

【语法】

```
HI_S32 HI_MPI_VENC_StartRecvPic(VENC_CHN VeChn);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回失败 HI_ERR_VENC_UNEXIST。
- 如果通道没有注册到通道组，则返回失败。
- 此接口不判断当前是否已经开启接收，即允许重复开启不返回错误。
- 开始接收输入是针对通道的，只有开启接收之后编码器才开始接收图像编码。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

请参见 HI_MPI_VENC_CreateGroup 的举例。

【相关主题】

无。

HI_MPI_VENC_StartRecvPicEx

【描述】

开启编码通道接收输入图像，超出指定的帧数后自动停止接收图像。

【语法】

```
HI_S32 HI_MPI_VENC_StartRecvPicEx(VENC_CHN VeChn, VENC_RECV_PIC_PARAM_S
*pstRecvParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstRecvParam	接收图像参数结构体指针，用于指定需要接收的图像帧数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回失败 HI_ERR_VENC_UNEXIST。
- 如果通道没有注册到通道组，则返回失败。
- 如果通道已经调用了 HI_MPI_VENC_StartRecvPic 开始接收图像而没有停止接收图像，或者上次调用 HI_MPI_VENC_StartRecvPicEx 后还没有接收到足够的图像，再一次调用此接口返回操作不允许。

【举例】

```
HI_S32 JpegSnapProcess(HI_VOID)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    HI_S32 s32Ret;
    VENC_GRP VeGroup = 0;
    VENC_CHN VeChn = 0;
    VENC_CHN_ATTR_S stAttr;
    VENC_RECV_PIC_PARAM_S stRecvParam;
    /* set jpeg chnnel video encode attribute */
    stAttr.stVeAttr.enType   = PT_JPEG;
    stAttr.stVeAttr.stAttrJpeg.u32PicWidth = u32PicWidth;
    stAttr.stVeAttr.stAttrJpeg.u32PicHeight = u32PicHeigh;
    stAttr.stVeAttr.stAttrJpeg.u32MaxPicWidth = u32MaxPicWidth;
    stAttr.stVeAttr.stAttrJpeg.u32MaxPicHeight = u32MaxPicHeigh;
    …… // omit other video encode assignments here.
    // create jpeg channel
    s32Ret = HI_MPI_VENC_CreateGroup(VeGroup);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_CreateGroup err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    s32Ret = HI_MPI_VENC_CreateChn(VeChn, &stAttr, HI_NULL);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_CreateChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    s32Ret = HI_MPI_VENC_RegisterChn(VeGroup, VeChn);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_RegisterChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    // start snapping
    stRecvParam.s32RecvPicNum = 2;
    s32Ret = HI_MPI_VENC_StartRecvPicEx(VeChn, &stRecvParam);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_StartRecvPicEx err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    …… // wait until all pictures have been encoded.
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    s32Ret = HI_MPI_VENC_StopRecvPic(VeChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_StopRecvPic err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    // destroy jpeg channel
    s32Ret = HI_MPI_VENC_UnRegisterChn(VeChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_UnRegisterChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    s32Ret = HI_MPI_VENC_DestroyChn(VeChn);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_DestroyChn err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    s32Ret = HI_MPI_VENC_DestroyGroup(VeGroup);
    if (s32Ret != HI_SUCCESS)
    {
        printf("HI_MPI_VENC_DestroyGroup err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_StopRecvPic

【描述】

停止编码通道接收输入图像。

【语法】

```
HI_S32 HI_MPI_VENC_StopRecvPic(VENC_CHN VeChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回失败。
- 如果通道没有注册到通道组，则返回失败。
- 此接口并不判断当前是否停止接收，即允许重复停止接收不返回错误。
- 此接口用于编码通道停止接收图像来编码，在编码通道反注册前必须停止接收图像。
- 调用此接口仅停止接收原始数据编码，码流 buffer 并不会被清除。

【举例】

请参见 HI_MPI_VENC_DestroyGroup 的举例。

【相关主题】

无。

HI_MPI_VENC_Query

【描述】

查询编码通道状态。

【语法】

```
HI_S32 HI_MPI_VENC_Query(VENC_CHN VeChn, VENC_CHN_STAT_S *pstStat);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstStat	编码通道的状态指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回失败。
- 此接口用于查询此函数调用时刻的编码器状态，pstStat 包含五个主要的信息：
  - 在编码通道状态结构体中，bRegistered 表示当前编码通道的注册状态。
  - 在编码通道状态结构体中，u32LeftPics 表示待编码的帧个数。

    在反注册通道前，可以通过查询是否还有图像待编码来决定反注册时机，防止反注册时将可能需要编码的帧清理出去。
  - 在编码通道状态结构体中，u32LeftStreamBytes 表示码流 buffer 中剩余的 byte 数目。

    在反注册通道前，可以通过查询是否还有码流没有被处理来决定反注册时机，防止反注册时将可能需要的码流清理出去。
  - 在编码通道状态结构体中，u32LeftStreamFrames 表示码流 buffer 中剩余的帧数目。

    在反注册通道前，可以通过查询是否还有图像的码流没有被取走来决定反注册时机，防止反注册时将可能需要的码流清理出去。
  - 在编码通道状态结构体中，u32CurPacks 表示当前帧的码流包个数。在调用 HI_MPI_VENC_GetStream 之前应确保 u32CurPacks 大于 0。

    在按包获取时当前帧可能不是一个完整帧（被取走一部分），按帧获取时表示当前一个完整帧的包个数（如果没有一帧数据则为 0）。用户在需要按帧获取码流时，需要查询一个完整帧的包个数，在这种情况下，通常可以在 select 成功后执行 query 操作，此时 u32CurPacks 是当前完整帧中包的个数。
  - 在编码通道状态结构体中，u32LeftRecvPics 表示调用 HI_MPI_VENC_StartRecvPicEx 接口后剩余等待接收的帧数目。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 在编码通道状态结构体中，u32LeftEncPics 表示调用 HI_MPI_VENC_StartRecvPicEx 接口后剩余等待编码的帧数目。

  如果没有调用 HI_MPI_VENC_StartRecvPicEx，u32LeftRecvPics 和 u32LeftEncPics 数目始终为 0。

【举例】

请参见 HI_MPI_VENC_GetStream 的举例。

【相关主题】

无。

HI_MPI_VENC_SetChnAttr

【描述】

设置编码通道属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetChnAttr(VENC_CHN VeChn, const VENC_CHN_ATTR_S*
pstAttr);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstChnAttr	编码通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 此接口必须在通道创建之后调用。
- 编码通道属性包括了编码器属性和码率控制器属性两部分。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 编码器属性只能修改设置编码通道 u32PicWidth 和 u32PicHeight，其它属性不能修改设置。
- 调用此接口修改 u32PicWidth 或者 u32PicHeight 前，必须先调用 HI_MPI_VENC_StopRecvPic 接口停止接收图像。
- 当调用此接口修改 u32PicWidth 和 u32PicHeight 后，除了通道组帧率控制属性外，编码组和通道的其它所有参数配置恢复默认值，并且清空码流 buffer 和缓存图像队列。
- 如果 pstAttr 为空，则返回失败。
- 编码通道属性分为动态属性和静态属性两种。其中，动态属性的属性值在通道创建时配置，在通道销毁之前可以被修改；静态属性的属性值在通道创建时配置，在通道创建之后不能被修改。
- 此接口只能设置编码通道属性中的动态属性，如果设置静态属性，则返回失败。编码通道的最大宽和最大高、主次码流、编码协议、编码的帧场模式、输入图像的帧场模式、获取码流的方式（按帧还是按包获取码流）属性属于静态属性。另外，各个编码协议的静态属性由各个协议模块指定，具体请参见 VENC_CHN_ATTR_S。
- RC 属性参数的详细使用说明请参见《码率控制使用说明》。
- 当码率控制模式改变时，所有的码率控制高级参数自动恢复默认值，用户需要重新设置 RC 高级参数。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetChnAttr

【描述】

获取编码通道属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetChnAttr(VENC_CHN VeChn,VENC_CHN_ATTR_S*pstAttr);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstChnAttr	编码通道属性指针。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 获取未创建的通道的属性，返回失败 HI_ERR_VENC_UNEXIST。
- 如果 pstAttr 为空，则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetStream

【描述】

获取编码的码流。

【语法】

```
HI_S32 HI_MPI_VENC_GetStream(VENC_CHN VeChn, VENC_STREAM_S *pstStream,
HI_BOOL bBlockFlag);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstStream	码流结构体指针。	输出
bBlockFlag	阻塞方式。 取值范围： • HI_TRUE：阻塞。 • HI_FALSE：非阻塞。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，返回失败。
- 如果 pstStream 为空，返回失败
- 支持阻塞或非阻塞两种方式获取。支持 select/poll 系统调用。
  - 非阻塞获取时，如果缓冲无数据，则返回失败 HI_ERR_VENC_BUF_EMPTY。
  - 阻塞时，如果缓冲无数据，则会等待有数据时才返回获取成功。
- 支持按包或按帧方式获取码流。如果按包获取，则：
  - 对于 H.264 编码协议，每次获取的是一个 NAL 单元。
  - 对于 JPEG 编码协议（包括 JPEG 抓拍和 MJPEG），每次获取的是一个 ECS 或图像参数码流包。
  - 对于 MPEG-4 协议编码通道，每次获取的是一个 Packet。
- 码流结构体 VENC_STREAM_S 包含 4 个部分：
  - 码流包信息指针 pstPack

    指向一组 VENC_PACK_S 的内存空间，该空间由调用者分配。如果是按包获取，则此空间不小于 sizeof（VENC_PACK_S）的大小；如果按帧获取，则此空间不小于 N × sizeof（VENC_PACK_S）的大小，其中 N 代表当前帧之中的包的个数，可以在 select 之后通过查询接口获得。
  - 码流包个数 u32PackCount

    在输入时，此值指定 pstPack 中 VENC_PACK_S 的个数。按包获取时，u32PackCount 必须不小于 1；按帧获取时，u32PackCount 必须不小于当前帧的包个数。在函数调用成功后，u32PackCount 返回实际填充 pstPack 的包的个数。
  - 序列号 u32Seq

    按帧获取时是帧序列号；按包获取时为包序列号。
  - 码流特征信息，数据类型为联合体，包含了不同编码协议对应的码流特征信息 stH264Info/ stJpegInfo/stMpeg4Info，码流特征信息的输出用于支持用户的上层应用。
- 如果用户长时间不获取码流，码流缓冲区就会满。一个编码通道如果发生码流缓冲区满，就会把后面接收的图像丢掉，直到用户获取码流，从而有足够的码流缓冲可以用于编码时，才开始继续编码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 建议用户获取码流接口调用与释放码流的接口调用成对出现，且尽快释放码流，防止出现由于用户态获取码流，释放不及时而导致的码流 buffer 满，停止编码。

【举例】

```
HI_S32 VencGetH264Stream(HI_VOID)
{
    HI_S32 i;
    HI_S32 s32Ret;
    HI_S32 s32VencFd;
    HI_U32 u32FrameIdx = 0;
    VENC_CHN VeChn = 0;
    VENC_CHN_STAT_S stStat;
    VENC_STREAM_S stStream;
    fd_set read_fds;
    FILE *pFile  = NULL;

    pFile = fopen("stream.h264","wb");
    if(pFile == NULL)
    {
        return HI_FAILURE;
    }

    s32VencFd = HI_MPI_VENC_GetFd(VeChn);
    do{
        FD_ZERO(&read_fds);
        FD_SET(s32VencFd,&read_fds);

        s32Ret = select(s32VencFd+1, &read_fds, NULL, NULL, NULL);
        if (s32Ret < 0)
        {
            printf("select err\n");
            return HI_FAILURE;
        }
        else if (0 == s32Ret)
        {
            printf("time out\n");
            return HI_FAILURE;
        }
        else
        {
            if (FD_ISSET(s32VencFd, &read_fds))
            {
                s32Ret = HI_MPI_VENC_Query(VeChn, &stStat);
                if (s32Ret != HI_SUCCESS)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
            {
                return HI_FAILURE;
            }

            stStream.pstPack = (VENC_PACK_S*)
                malloc(sizeof(VENC_PACK_S)*stStat.u32CurPacks);
            if (NULL == stStream.pstPack)
            {
                return HI_FAILURE;
            }

            stStream.u32PackCount = stStat.u32CurPacks;
            s32Ret = HI_MPI_VENC_GetStream(VeChn, &stStream, HI_TRUE);
            if (HI_SUCCESS != s32Ret)
            {
                free(stStream.pstPack);
                stStream.pstPack = NULL;
                return HI_FAILURE;
            }

            for (i=0; i< stStream.u32PackCount; i++)
            {
                fwrite(stStream.pstPack[i].pu8Addr[0],
                        stStream.pstPack[i].u32Len[0], 1, pFile);
                if (stStream.pstPack[i].u32Len[1] > 0)
                {
                    fwrite(stStream.pstPack[i].pu8Addr[1],
                            stStream.pstPack[i].u32Len[1], 1, pFile);
                }
            }

            s32Ret = HI_MPI_VENC_ReleaseStream(VeChn,&stStream);
            if (HI_SUCCESS != s32Ret)
            {
                free(stStream.pstPack);
                stStream.pstPack = NULL;
                return HI_FAILURE;
            }

            free(stStream.pstPack);
            stStream.pstPack = NULL;
        }
    }
    u32FrameIdx++;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    }while (u32FrameIdx < 0xff);

    fclose(pFile);
    return HI_SUCCESS;
}
```

更详细的内容，请参考sample代码。

【相关主题】

无。

HI_MPI_VENC_ReleaseStream

【描述】

释放码流缓存。

【语法】

```
HI_S32 HI_MPI_VENC_ReleaseStream(VENC_CHN VeChn, VENC_STREAM_S
*pstStream);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstStream	码流结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回错误码 HI_ERR_VENC_UNEXIST。
- 如果 pstStream 为空，则返回错误码 HI_ERR_VENC_NULL_PTR。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 此接口应当和 HI_MPI_VENC_GetStream 配对起来使用，用户获取码流后必须及时释放已经获取的码流缓存，否则可能会导致码流 buffer 满，影响编码器编码，并且用户必须按先获取先释放的顺序释放已经获取的码流缓存。
- 在编码通道反注册以后，所有未释放的码流包均无效，不能再使用或者释放这部分无效的码流缓存。
- 释放无效的码流会返回失败 HI_ERR_VENC_ILLEGAL_PARAM。

【举例】

请参见 HI_MPI_VENC_GetStream 的举例。

【相关主题】

无。

HI_MPI_VENC_InsertUserData

【描述】

插入用户数据。

【语法】

```
HI_S32 HI_MPI_VENC_InsertUserData(VENC_CHN VeChn, HI_U8 *pu8Data, HI_U32 u32Len);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pu8Data	用户数据指针。	输入
u32Len	用户数据长度。 取值范围：(0, 1024]，以 byte 为单位。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 如果通道未创建，则返回失败。
- 如果 pu8Data 为空，则返回失败。
- 插入用户数据，只支持 H.264 和 MJPEG/JPEG 编码协议，不支持 MPEG-4 编码协议。
- 向 H.264 通道插入用户数据时，建议用户在启动编码器后再操作。因为如果用户在编码器启动之前插入用户数据，编码器编出的码流就会出现以下状况：在码流的第一个 SPS 包之前会出现若干个 SEI 包。很多解码器（包括 HI3531 的解码器）遇到这种码流会把第一个 SPS 包之前的 SEI 包丢掉，造成用户丢失存储在这些 SEI 包里的用户数据。
- H.264 协议通道最多同时分配 4 块内存空间用于缓存用户数据，且每段用户数据大小不超过 1k byte。如果用户插入的数据多余 4 块，或插入的一段用户数据大于 1k byte 时，此接口会返回错误。每段用户数据以 SEI 包的形式被插入到最新的图像码流包之前。在某段用户数据包被编码发送之后，H.264 通道内缓存这段用户数据的内存空间被清零，用于存放新的用户数据。
- JPEG/MJPEG 协议通道最多分配 1 块内存空间，用于缓存 1Kbyte 的用户数据。用户数据以 APP segment（0xFFEF）形式添加到图像码流中。在用户数据被编码发送之后，JPEG/MJPEG 通道内缓存这段用户数据的内存空间被清零，用于存放新的用户数据。

【举例】

```
HI_U8 au8UserData[] = "hisilicon2011";
s32Ret = HI_MPI_VENC_InsertUserData(VeChn, au8UserData,
            sizeof(au8UserData));
    if(HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_InsertUserData err 0x%xn",s32Ret);
    }
```

【相关主题】

无。

HI_MPI_VENC_SendFrame

【描述】

支持用户发送原始图像进行编码。

【语法】

```
HI_S32 HI_MPI_VENC_SendFrame(VENC_GRP VeGroup, VIDEO_FRAME_INFO_S
*pstFrame)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeGroup	编码通道组号。 取值范围：[0, VENC_MAX_GRP_NUM)。	输入
pstFrame	原始图像信息结构指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 此接口支持用户发送图像至编码通道组。
- 用户发送原始图像必须为 Semi-planar YUV 4:2:0 或 Semi-planar YUV 4:2:2 格式。H.264/MPEG-4 编码通道支持接收 Semi-plannar420 图像，JPEG/MJPEG 编码通道支持接收 Semi-plannar420 或 Semi-plannar422 图像。
- 视频输入的原始图像大小必须不小于编码通道的大小。
- 此接口不支持用户发送图像直方图至编码通道组。

【举例】

详细过程请参照 sample。

【相关主题】

无。

HI_MPI_VENC_SetMaxStreamCnt

【描述】

设置码流最大缓存帧数。

【语法】

```
HI_S32 HI_MPI_VENC_SetMaxStreamCnt(VENC_CHN VeChn,HI_U32 u32MaxStrmCnt);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
u32MaxStrmCnt	最大码流缓存帧数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 此接口用于设置编码通道的码流 buffer 中能够缓存的最大码流帧数。
- 若缓存码流帧数已达到最大码流帧数，当前待编码图像会被丢弃。
- 最大码流帧数在创建通道时由系统内部指定默认值，默认值为 200。
- 此接口在创建通道之后，销毁编码通道之前均可以被调用。在下一帧开始编码之前生效。此接口允许被多次调用。建议在创建编码通道成功之后，启动编码前进行设置，不建议在编码过程中动态调整。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetMaxStreamCnt

【描述】

获取码流最大缓存帧数。

【语法】

```
HI_S32 HI_MPI_VENC_GetMaxStreamCnt(VENC_CHN VeChn,HI_U32 *pu32MaxStrmCnt);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pu32MaxStrmCnt	最大码流缓存帧数的指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_RequestIDR

【描述】

请求 I 帧。

【语法】

```
HI_S32 HI_MPI_VENC_RequestIDR( VENC_CHN VeChn );
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回失败。
- 接受 IDR 帧或 I 帧请求后，在帧率控制的下一帧编出 IDR 帧或 I 帧。
- I 帧请求，只支持 H.264 和 MPEG-4 编码协议。

【举例】

```
HI_S32 s32Ret;
VENC_CHN VeChn = 0;
    s32Ret = HI_MPI_VENC_RequestIDR(VeChn);
    if(HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_RequestIDR err 0x%xn",s32Ret);
    }
```

【相关主题】

无。

HI_MPI_VENC_RequestIDRInst

【描述】

请求 I 帧并立即生效。

【语法】

```
HI_S32 HI_MPI_VENC_RequestIDRInst(VENC_CHN VeChn);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果通道未创建，则返回失败。
- 接受 IDR 帧或 I 帧请求后，立即编出 IDR 帧或 I 帧，不受帧率控制约束。
- I 帧请求，只支持 H.264 和 MPEG-4 编码协议。
- 由于此接口不受帧率控制影响，每调用一次即编出一个 IDR 或 I 帧，调用频繁会影响码流帧率和码率的稳定，使用时需要注意。

【举例】

```
HI_S32 s32Ret;
VENC_CHN VeChn = 0;
    s32Ret = HI_MPI_VENC_RequestIDRInst(VeChn);
    if(HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_RequestIDRInst err 0x%xn",s32Ret);
    }
```

【相关主题】

无。

HI_MPI_VENC_GetFd

【描述】

获取编码通道对应的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_VENC_GetFd(VENC_CHN VeChn);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetRoiCfg

【描述】

设置 H.264 通道的 ROI 属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetRoiCfg(VENC_CHN VeChn, VENC_ROI_CFG_S
*pstVencRoiCfg);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstVencRoiCfg	ROI 区域参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道 ROI 区域的参数
- ROI 参数主要由 5 个参数决定
  - u32Index：系统支持每个通道可设置 8 个 ROI 区域，系统内部按照 0～7 的索引号对 ROI 区域进行管理，u32Index 表示的用户设置 ROI 的索引号。ROI 区域之间可以互相叠加，且当发生叠加时，ROI 区域之间的优先级按照索引号 0～7 依次提高。
  - bEnable：指定当前的 ROI 区域是否使能。
  - bAbsQp：指定当前的 ROI 区域采用绝对 QP 方式或是相对 QP。
  - s32Qp：当 bAbsQp 为 true 时，s32Qp 表示 ROI 区域内部的所有宏块采用的 QP 值，当 bAbsQp 为 false 时，s32Qp 表示 ROI 区域内部的所有宏块采用的相对 QP 值。
  - stRect：指定当前的 ROI 区域的位置坐标和区域的大小。ROI 区域的起始点坐标必须在图像范围内，且必须 16 对齐；ROI 区域的长宽必须是 16 对齐；ROI 区域必须在图像范围内。
- 本接口属于高级接口，系统默认没有 ROI 区域使能，用户必须调用此接口启动 ROI。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。此接口在编码过程中被调用时，等到下一个帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetRoiCfg 接口，获取当前通道的 ROI 配置，然后再进行设置。

【举例】

```
HI_S32 SetRoi(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_ROI_CFG_S    stRoiCfg;
    VENC_CHN_ATTR_S   stChnAttr;
    HI_S32 index = 0;
    VENC_CHN VeChnId = 0;

    //...omit other thing
    s32Ret = HI_MPI_VENC_GetChnAttr(VeChnId,&stChnAttr);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetChnAttr err 0x%x\n",s32Ret);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
        return HI_FAILURE;
    }

    s32Ret = HI_MPI_VENC_GetRoiCfg(VeChnId, index, &stRoiCfg);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetRoiCfg err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stSrcVencRoiCfg.bEnable         = HI_TRUE;
    stSrcVencRoiCfg.bAbsQp          = HI_TRUE;
    stSrcVencRoiCfg.s32Qp           = 10;
    stSrcVencRoiCfg.stRect.s32X     = 16;
    stSrcVencRoiCfg.stRect.s32Y     = 16;
    stSrcVencRoiCfg.stRect.u32Width  = 16;
    stSrcVencRoiCfg.stRect.u32Height = 16;
    stSrcVencRoiCfg.u32Index        = 0;
    s32Ret = HI_MPI_VENC_SetRoiCfg(VeChnId,  &stSrcVencRoiCfg);
    if (HI_SUCCESS != s32Ret)
    {
            printf("HI_MPI_VENC_SetRoiCfg err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetRoiCfg

【描述】

获取 H.264 通道的 Roi 配置属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetRoiCfg(VENC_CHN VeChn, HI_U32 u32Index,
VENC_ROI_CFG_S *pstVencRoiCfg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
u32Index	H.264 协议编码通道 ROI 区域索引。	输入
pstVencRoiCfg	对应 ROI 区域的配置。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的 index 的 ROI 配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_GetRoiCfg 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264SliceSplit

【描述】

设置 H.264 通道的 slice 分割属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264SliceSplit(VENC_CHN VeChn, const
VENC_PARAM_H264_SLICE_SPLIT_S * pstSliceSplit);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstSliceSplit	H.264 码流 slice 分割参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道码流的分割方式
- Slice 分割属性主要由三个参数决定
  - bSplitEnable：当前帧是否进行 slice 分割。
  - u32SplitMode：slice 分割模式，u32SplitMode = 0 表示按照 byte 数来进行分割，u32SplitMode = 1 表示按照宏块行进行分割。u32SplitMode>=2 不被支持。

    u32SplitMode 必须在 bSplitEnable 为真的条件下有效，否则，无效。
  - u32SliceSize：slice 的大小，当 u32SplitMode 不同时，u32SliceSize 表示不同的意义。当 u32SplitMode = 0 时，u32SliceSize 表示平均每个 slice 的 byte 数的多少，编码器从图像的左上角开始，从左至右，从上至下，按光栅顺序按宏块进行编码。每个 slice 大小按照 u32SliceSize 进行划分。最终编码 slice 大小不一定与用户设置的 u32SliceSize 严格相等，存在少许的误差量，且实际的偏移量总是正数。当编码至图像的最后一个 slice 时，剩余的宏块 byte 数不足 u32SliceSize，剩余的所有宏块被编码成一个 slice。当 u32SplitMode = 1 时，u32SliceSize 表示每个 slice 占图像宏块行数，u32SliceSize 的单位为宏块行。且当编码至图像的最后几行，不足 u32SliceSize 时，剩余的宏块行被划分为一个 slice。
- 本接口属于高级接口，用户可以选择性调用，建议不调用，系统默认 bSplitEnable 为 false。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。此接口在编码过程中被调用时，等到下一帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264SliceSplit 接口，获取当前通道的 slicesplit 配置，然后再进行设置。

【举例】

```
HI_S32 SetSliceSplit(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_SLICE_SPLIT_S stSlice;
    VENC_CHN VeChnId = 0;

    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264SliceSplit(VeChnId,  &stSlice);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_ GetH264SliceSplit err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stSlice.bSplitEnable = HI_TRUE;
    stSlice.u32SplitMode = 0;
    stSlice.u32SliceSize = 2048;
    s32Ret = HI_MPI_VENC_SetH264SliceSplit(VeChnId,  &stSlice);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_ SetH264SliceSplit err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264SliceSplit

【描述】

获取 H.264 通道的 slice 分割属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264SliceSplit(VENC_CHN VeChn,
VENC_PARAM_H264_SLICE_SPLIT_S *pstSliceSplit);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstSliceSplit	H.264 码流 slice 分割参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的 slice 分割属性。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetH264SliceSplit 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264InterPred

【描述】

设置 H.264 协议编码通道的帧间预测属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264InterPred(VENC_CHN VeChn, const
VENC_PARAM_H264_INTER_PRED_S *pstH264InterPred);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264InterPred	H.264 协议编码通道的帧间预测配置。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【芯片差异】

- Hi35xx H.264 编码器搜索窗范围如表 6-8 所示。

表6-8 搜索窗范围

芯片	MIN_HW	MAX_HW	MIN_VW	MAX_VW
Hi3531/Hi3532	0	11	0	4
Hi3521/Hi3520A/Hi3518/Hi3516C	0	5	0	2
Hi3520D/Hi3515A/Hi3515C	0	5	0	2

其中，MIN_HW，MAX_HW，MIN_VW，MAX_VW 分别表示编码通道支持最小水平搜索索引，最大水平搜索索引，最小垂直搜索索引，最大垂直搜索索引。

- Hi35xx H.264 编码器在在不同分辨率下的搜索窗典型配置如表 6-9、表 6-10 和表 6-11 所示。

表6-9 Hi3531/Hi3532 H.264 编码器在不同图像分辨率下搜索窗默认配置

PicWidth(pixel)	u32HWSize	u32VWSize
≤1000	3	2
(1000,1600]	7	4
(1600,1920]	9	4
(1920,2133]	9	3
(2133,2742]	9	2

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

PicWidth(pixel)	u32HWSize	u32VWSize
(2742,3840]	9	1
≥3840	9	0

表6-10 Hi3521/Hi3520A H.264 编码器在不同图像分辨率下搜索窗默认配置

PicWidth(pixel)	u32HWSize	u32VWSize
≤960	5	2
(960,1280]	5	1
≥1280	5	0

表6-11 Hi3518/Hi3516C H.264 编码器不同图像分辨率下搜索窗默认配置

PicWidth(pixel)	u32HWSize	u32VWSize
≤1280	5	2
>1280	5	1

表6-12 Hi3520D/Hi3515A/Hi3515C H.264 编码器不同图像分辨率下搜索窗默认配置

PicWidth(pixel)	u32HWSize	u32VWSize
≤720	5	2
(720，1280]	5	1
(1280，1920]	5	0

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的帧间预测的配置
- 帧间预测的属性主要由七个参数决定
  - u32HWSize：帧间预测时的水平搜索窗的大小。
  - u32VWSize：帧间预测时的垂直搜索窗的大小。
  - bInter16x16PredEn：16x16 块帧间预测使能标识。
  - bInter16x8PredEn：16x8 块帧间预测使能标识。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- bInter8x16PredEn：8x16 块帧间预测使能标识。
- bInter8x8PredEn：8x8 块帧间预测使能标识。
- bExtedgeEn：帧间预测扩边搜索使能。对于图像边界上的宏块进行帧间预测时，搜索窗的范围会超出图像的边界，此时，扩边搜索使能会对图像进行扩边，进行帧间预测。如果扩边搜索使能被关闭，那么对于超出图像范围的搜索窗，不会进行预测。

- 此接口属于高级接口，用户可以选择性调用，建议不调用，系统会有默认值。默认搜索窗的大小会根据图像的分辨率不同，其他 5 个预测使能开关全部默认使能。
- bInter16x16PredEn、bInter16x8PredEn 、bInter8x16PredEn 、bInter8x8PredEn 的四个帧间预测使能开关必须有一个是开启的，系统不支持四个开关全部关闭。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264InterPred 接口，获取当前编码通道的 InterPred 配置，然后再进行设置。
- 水平搜索窗的大小存在以下约束：

  水平搜索窗必须大于等于 MIN_HW，且小于等于 MAX_HW。

  图像宽度（以宏块为单位）必须比搜索窗实际宽度至少大 2 个宏块。（以宏块为单位，实际搜索窗宽度= (1+ u32HWSize) x 2+1。如搜索窗的宽度设置为 10，则实际的图像宽度至少大于：((1+ 10) x 2+1+1）x 16 = 384，如 386。
- 垂直搜索窗的大小存在以下约束：

  垂直搜索窗必须大于等于 MIN_VW，且小于等于 MAX_VW。

  图像高度（以宏块为单位）必须比搜索窗实际高度至少大 1 个宏块。（以宏块为单位，实际搜索窗高度= (1+ u32VWSize) x 2+1。如搜索窗的高度设置为 4，则实际的图像高度至少大于：(1+ 4) x 2+1） x 16 = 176，如 178。

  对于 Hi3521，当图像宽度小于等于 960 时，垂直搜索窗必须小于等于 2；当图像宽度大于 960，且小于等于 1280 时，垂直搜索窗必须小于等于 1；当图像宽度大于 1280，且小于等于 1920 时，垂直搜索窗只能等于 0。

  对于 Hi3518/Hi3516C，当图像宽度小于等于 1280 时，垂直搜索窗必须小于等于 2；当图像宽度大于 1280，且小于等于 1920 时，垂直搜索窗必须小于等于 1。

  对于 Hi3520D/Hi3515A/Hi3515C，当图像宽度小于等于 720 时，垂直搜索窗必须小于等于 2；当图像宽度大于 720，且小于等于 1280 时，垂直搜索窗必须小于等于 1；当图像宽度大于 1280，且小于等于 1920 时，垂直搜索窗只能等于 0。

【举例】

```
HI_S32 SetInterPred(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_INTER_PRED_S  stInterPred;
    VENC_CHN VeChnId = 0;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264InterPred(VeChnId,  &stInterPred);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetH264InterPred err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stInterPred.bExtedgeEn         = HI_TRUE;
    stInterPred.bInter16x16PredEn = HI_TRUE;
    stInterPred.bInter16x8PredEn  = HI_FALSE;
    stInterPred.bInter8x16PredEn  = HI_FALSE;
    stInterPred.bInter8x8PredEn   = HI_FALSE;
    stInterPred.u32HWSize        = 4;
    stInterPred.u32VWSize        = 2;
    s32Ret = HI_MPI_VENC_SetH264InterPred(VeChnId,  &stInterPred);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_ SetH264InterPred err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264InterPred

【描述】

获取 H.264 协议编码通道的帧间预测属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264InterPred(VENC_CHN VeChn,
VENC_PARAM_H264_INTER_PRED_S *pstH264InterPred);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264InterPred	H.264 协议编码通道的帧间预测参数。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的帧间预测方式。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- Hi3518/Hi3516C 不支持 8x8 变换及量化表配置，在 High profile 及 SVC profile 时不能配置 8x8 变换及量化表。

【举例】

请参见 HI_MPI_VENC_ SetH264InterPred 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264IntraPred

【描述】

设置 H.264 协议编码通道的帧内预测属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264IntraPred(VENC_CHN VeChn, const
VENC_PARAM_H264_INTRA_PRED_S *pstH264IntraPred);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264IntraPred	H.264 协议编码通道的帧内预测配置。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【芯片差异】

- Hi3531/Hi3532 不支持 IPCM 预测功能，bIpcmEn 只能设置为 HI_FALSE。
- Hi3521/Hi3520A 支持 IPCM 预测功能，用户可根据需求，设定 IPCM 的开关，VENC 默认使能 IPCM 功能。
- Hi3518/Hi3516C 支持 IPCM 预测功能，用户可根据需求，设定 IPCM 的开关，VENC 默认使能 IPCM 功能。
- Hi3520D/Hi3515A/Hi3515C 支持 IPCM 预测功能，用户可根据需求，设定 IPCM 的开关，VENC 默认使能 IPCM 功能。

【注意】

- 本接口用于设置 H.264 协议编码通道的帧内预测的配置
- 帧间预测的属性主要由四个参数决定
  - bIntra16x16PredEn：16x16 块帧内预测使能标识。
  - bIntraNxNPredEn：NxN 块帧内预测使能标识。其中，NxN 表示 4x4 和 8x8。
  - bIpcmEn：IPCM 块帧内预测使能标识。
  - constrained_intra_pred_flag：具体的含义，请参见 H.264 协议。
- 此接口属于高级接口，用户可以选择性调用，不建议调用，系统会有默认值。系统默认使能 bIntra16x16PredEn 和 bIntraNxNPredEn，bIpcmEn 对于不同芯片有不同默认值，constrained_intra_pred_flag 默认为 0。
- bIntra16x16PredEn、bIntraNxNPredEn 两种帧内预测使能开关必须有一个是开启的，系统不支持两个开关全部关闭。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264IntraPred 接口，获取当前编码通道的 IntraPred 配置，然后再进行设置。

【举例】

```
HI_S32 SetIntraPred(HI_VOID)
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_INTRA_PRED_S  stIntraPred;
    VENC_CHN VeChnId = 0;
    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264IntraPred(VeChnId,  &stIntraPred);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetH264IntraPred err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stIntraPred.bIntra16x16PredEn = HI_TRUE;
    stIntraPred.bIntraNxNPredEn    = HI_FALSE;
    stIntraPred.bIpcmEn        = HI_FALSE;
    stIntraPred.constrained_intra_pred_flag = 0;
    s32Ret = HI_MPI_VENC_SetH264IntraPred(VeChnId,  &stIntraPred);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetH264IntraPred err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264IntraPred

【描述】

获取 H.264 协议编码通道的帧内预测属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264IntraPred(VENC_CHN VeChn,
VENC_PARAM_H264_INTRA_PRED_S *pstH264IntraPred);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264IntraPred	H.264 协议编码通道的帧内预测参数。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的帧内预测方式。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_ SetH264IntraPred 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264Trans

【描述】

设置 H.264 协议编码通道的变换、量化的属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264Trans(VENC_CHN VeChn, const
VENC_PARAM_H264_TRANS_S *pstH264Trans);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Trans	H.264 协议编码通道的变换、量化属性。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的变换、量化的配置。
- 变换、量化属性主要由六个参数组成：
  - u32IntraTransMode：帧内预测宏块的变换属性。u32IntraTransMode = 0 表示支持对帧内预测宏块支持 4x4 变换和 8x8 变换；u32IntraTransMode = 1 表示只支持对帧内预测宏块支持 4x4 变换；u32IntraTransMode = 2 表示只支持对帧内预测宏块支持 8x8 变换。只有在编码通道协议为 high profile 时，才能选择 8x8 变换，即在 baseline profile 和 main profile 下，系统只支持 u32IntraTransMode = 1 的配置。
  - u32InterTransMode：帧间预测宏块的变换属性。u32InterTransMode = 0 表示支持对帧间预测宏块支持 4x4 变换和 8x8 变换；u32InterTransMode = 1 表示只支持对帧间预测宏块支持 4x4 变换；u32InterTransMode = 2 表示只支持对帧间预测宏块支持 8x8 变换。只有在编码通道协议为 high profile 时，才能选择 8x8 变换，即在 baseline profile 和 main profile 下，系统只支持 u32InterTransMode = 1 的配置。
  - bScalingListValid：表示 InterScalingList8x8、IntraScalingList8x8 量化表是否有效。Baseline、main profile 时，协议不支持 8x8 变换，此标志位没有意义。high profile 编码通道支持用户传递量化表。如果此标志位置为 false，那么系统将使用 H.264 协议推荐的量化表，否则，将使用用户传递的量化表。
  - InterScalingList8x8：对帧间预测宏块进行 8x8 变换时，可由用户通过此数组提供量化表。
  - IntraScalingList8x8：对帧内预测宏块进行 8x8 变换时，可由用户通过此数组提供量化表。
  - chroma_qp_index_offset：具体含义请参见 H.264 协议。
- 此接口属于高级接口，用户可以选择性调用，不建议调用，系统会有默认值。系统会根据不同的 profile 设置默认参数。

表6-13 不同 profile 下系统默认 trans 参数

Profile	u32IntraTransMode	u32InterTransMode	bScalingListValid
Baseline /main profile	1	1	false
High profile	0	0	false

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264Trans 接口，获取当前编码通道的 trans 配置，然后再进行设置。

【举例】

```
HI_S32 SetTrans(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_TRANS_S stTrans;
    VENC_CHN VeChnId = 0;

    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264Trans(VeChnId,  &stTrans);
    if (HI_SUCCESS != s32Ret)
    {
        printf("stTrans err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stTrans.u32IntraTransMode     = 2;
    stTrans.u32InterTransMode     = 2;
    stTrans.bScalingListValid     = HI_FALSE;
    stTrans.chroma_qp_index_offset = 2;
    s32Ret = HI_MPI_VENC_SetH264Trans(VeChnId,  &stTrans);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetH264Trans err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264Trans

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

获取 H.264 协议编码通道的变换、量化属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264Trans(VENC_CHN VeChn, VENC_PARAM_H264_TRANS_S
*pstH264Trans);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Trans	H.264 协议编码通道的变换、量化参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的变换、量化配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_ SetH264Trans 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264Entropy

【描述】

设置 H.264 协议编码通道的熵编码模式。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_VENC_SetH264Entropy(VENC_CHN VeChn, const
VENC_PARAM_H264_ENTROPY_S *pstH264EntropyEnc);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264EntropyEnc	H.264 协议编码通道的熵编码模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的熵编码的配置。
- 变换、量化属性主要由四个参数组成：
  - u32EntropyEncModeI：I 帧的熵编码方式，u32EntropyEncModeI = 0 表示 I 帧使用 cavlc 编码，u32EntropyEncModeI = 1 表示 I 帧使用 cabac 编码方式。
  - u32EntropyEncModeP：p 帧的熵编码方式，u32EntropyEncModeP = 0 表示 P 帧使用 cavlc 编码，u32EntropyEncModeP = 1 表示 P 帧使用 cabac 编码方式。
  - cabac_stuff_en：cabac 编码填充使能，系统默认为 0。具体含义请参见 H.264 协议。
  - Cabac_init_idc：cabac 初始化索引，系统默认为 0。具体含义请参见 H.264 协议。
- I 帧的熵编码方式与 P 帧的熵编码方式可以分别设置。
- Baseline profile 不支持 cabac 编码方式，支持 cavlc 编码方式，main profile 和 high profile 支持 cabac 编码方式和 cavlc 编码方式。
- Cabac 编码方式相对于 cavlc 编码方式，需要更大的计算量，可是，会产生更少的码流。如果系统性能不够富裕，建议用户可以采取 I 帧 cavlc 编码，P 帧 cabac 编码。
- 此接口属于高级接口，用户可以选择性调用，不建议调用，系统会有默认值。系统会根据不同的 profile 设置默认参数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表6-14 不同 profile 下系统默认熵编码参数

Profile	u32EntropyEncModeI	u32EntropyEncModeP
Baseline	0	0
Main profile/High profile	1	1

- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264Entropy 接口，获取当前编码通道的 Entropy 配置，然后再进行设置。

【举例】

```
HI_S32 SetEntropy(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_ENTROPY_S stEntropy;
    VENC_CHN VeChnId = 0;
    //...omit other thing
    s32Ret = HI_MPI_VENC_GetH264Entropy(VeChnId,  &stTrans);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetH264Entropy err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    stEntropy.u32EntropyEncModeI  = 0;
    stEntropy.u32EntropyEncModeP  = 0;
    stEntropy.cabac_stuff_en      = 0;
    stEntropy.Cabac_init_idc      = 0;
    s32Ret = HI_MPI_VENC_SetH264Entropy(VeChnId,  &stEntropy);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetH264Entropy err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VENC_GetH264Entropy

【描述】

获取 H.264 协议编码通道的熵编码属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264Entropy(VENC_CHN VeChn,
VENC_PARAM_H264_ENTROPY_S *pstH264EntropyEnc);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264EntropyEnc	H.264 协议编码通道的熵编码属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的熵编码配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetH264Entropy 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264Poc

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

设置 H.264 协议编码通道的 POC 类型。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264Poc(VENC_CHN VeChn, const VENC_PARAM_H264_POC_S
*pstH264Poc);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Poc	H.264 协议编码通道的 POC 类型。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的 POC 的配置。
- POC 属性主要指 H.264 码流的 POC 类型，具体含义请见 H.264 协议。
- POC 类型现提供了 pic_order_cnt_type = 0/1/2 三种协议类型，系统默认 pic_order_cnt_type = 2。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264Poc 接口，获取当前编码通道的 Poc 配置，然后再进行设置。

【举例】

```
HI_S32 SetPoc(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_POC_S        stPoc;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VENC_CHN VeChnId = 0;
    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264Poc(VeChnId,  &stPoc);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetH264Poc err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stPoc.pic_order_cnt_type      = 0;
    s32Ret = HI_MPI_VENC_SetH264Poc(VeChnId,  &stPoc);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetH264Poc err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264Poc

【描述】

获取 H.264 协议编码通道的 Poc 类型。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264Poc(VENC_CHN VeChn, VENC_PARAM_H264_POC_S
*pstH264Poc);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Poc	H.264 协议编码通道的 Poc 属性。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的 poc 配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetH264Poc 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264Dblk

【描述】

设置 H.264 协议编码通道的 Deblocking 类型。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264Dblk(VENC_CHN VeChn, const
VENC_PARAM_H264_DBLK_S *pstH264Dblk);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Dblk	H.264 协议编码通道的 Deblocking 参数。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的 Deblocking 的配置
- 变换、量化属性主要由三个参数组成：
  - disable_deblocking_filter_idc：具体含义请参见 H.264 协议。
  - slice_alpha_c0_offset_div2：具体含义请参见 H.264 协议。
  - slice_beta_offset_div2：具体含义请参见 H.264 协议。
- 系统默认打开 deblocking 功能，默认 disable_deblocking_filter_idc = 0，slice_alpha_c0_offset_div2 = 0，slice_beta_offset_div2 = 0。
- 如果用户想关闭 deblocking 功能，可以将 disable_deblocking_filter_idc 置为 1。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264Dblk 接口，获取当前编码通道的 Dblk 配置，然后再进行设置。

【举例】

```
HI_S32 SetDblk(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_DBLK_S       stDblk;
    VENC_CHN VeChnId = 0;

    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264Dblk(VeChnId,  &stDblk);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetH264Dblk err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    stDblk.disable_deblocking_filter_idc = 0;
    stDblk.slice_alpha_c0_offset_div2    = 6;
    stDblk.slice_beta_offset_div2        = 5;
    s32Ret = HI_MPI_VENC_SetH264Dblk(VeChnId,  &stDblk);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetH264Dblk err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264Dblk

【描述】

获取 H.264 协议编码通道的 dblk 类型。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264Dblk(VENC_CHN VeChn, VENC_PARAM_H264_DBLK_S
*pstH264Dblk);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Dblk	H.264 协议编码通道的 dblk 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 本接口用于获取 H.264 协议编码通道的 dblk 配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetH264Dblk 的举例。

【相关主题】

无。

HI_MPI_VENC_SetH264Vui

【描述】

设置 H.264 协议编码通道的 vui 参数。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264Vui(VENC_CHN VeChn, const
VENC_PARAM_H264_VUI_S*pstH264Vui);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Vui	H.264 协议编码通道的 Vui 参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的 Vui 的配置

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- Vui 主要由四个参数组成：
  - timing_info_present_flag：具体含义请参见 H.264 协议。
  - num_units_in_tick：具体含义请参见 H.264 协议。
  - time_scale：具体含义请参见 H.264 协议。
  - fixed_frame_rate_flag：具体含义请参见 H.264 协议。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264Vui 接口，获取当前编码通道的 Vui 配置，然后再进行设置。

【举例】

```
HI_S32 SetVui(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_H264_VUI_S       stVui;
    VENC_CHN VeChnId = 0;
    //...omit other thing

    s32Ret = HI_MPI_VENC_GetH264Vui(VeChnId,  &stVui);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetH264Vui err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stVui.timing_info_present_flag = 1;
    s32Ret = HI_MPI_VENC_SetH264Vui(VeChnId,  &stVui);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetH264Vui err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetH264Vui

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

获取 H.264 协议编码通道的 Vui 配置。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264Vui(VENC_CHN VeChn, VENC_PARAM_H264_VUI_S
*pstH264Vui);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstH264Vui	H.264 协议编码通道的 Vui 属性。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 协议编码通道的 Vui 配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetH264Vui 的举例

【相关主题】

无。

HI_MPI_VENC_SetJpegParam

【描述】

设置 JPEG 协议编码通道的高级参数。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_VENC_SetJpegParam(VENC_CHN VeChn, const VENC_PARAM_JPEG_S
*pstJpegParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstJpegParam	JPEG 协议编码通道的高级参数集合。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 JPEG 协议编码通道的高级参数
- 高级参数主要由 5 个参数组成
  - u32Qfactor：量化表因子范围为[1, 99]，u32Qfactor 越大，量化表中的量化系数越小，得到的图像质量会更好，同时，编码压缩率更低。同理 u32Qfactor 越小，量化表中的量化系数越大，得到的图像质量会更差，同时，编码压缩率更高。具体的 u32Qfactor 与量化表的关系请见 RFC2435 标准。
  - u8YQt[64]，u8CbQt[64]，u8CrQt[64]：对应三个量化表空间，用户可以通过这三个参数设置用户的量化表。
  - u32MCUPerECS：每个 Ecs 中包含多少 Mcu。系统模式 u32MCUPerECS = 0，表示当前帧的所有的 MCU 被编码为一个 ECS。u32MCUPerECS 的最小值为 0，最大值不超过(picwidth+15)>>4 x (picheight+15)>>4 x 2。
- 如果用户想使用自己的量化表，在设置量化表的同时，请将 Qfactor 设置为 50。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个帧编码时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetJpegParam 接口，获取当前编码通道的 JpegParam 配置，然后再进行设置。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

```
HI_S32 SetJpegParam(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_JPEG_S stParamJpeg;
    VENC_CHN VeChnId = 0;

    //...omit other thing

    s32Ret = HI_MPI_VENC_GetJpegParam(VeChnId,  &stParamJpeg);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetJpegParam err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stParamJpeg.u32MCUPerECS = 100;
    for (i = 0; i < 64; i++)
    {
        stParamJpeg.u8YQt[i] = 16;
        stParamJpeg.u8CbQt[i] = 17;
        stParamJpeg.u8CrQt[i] = 18;
    }
    s32Ret = HI_MPI_VENC_SetJpegParam(VeChnId,  &stParamJpeg);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetJpegParam err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetJpegParam

【描述】

获取 JPEG 协议编码通道的高级参数配置。

【语法】

```
HI_S32 HI_MPI_VENC_GetJpegParam(VENC_CHN VeChn, VENC_PARAM_JPEG_S
*pstJpegParam);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstJpegParam	Jpeg 协议编码通道的高级参数配置。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 JPEG 协议编码通道的高级参数配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetJpegParam 的举例。

【相关主题】

无。

HI_MPI_VENC_SetMpeg4Param

【描述】

设置 MPEG-4 协议编码通道的高级参数。

【语法】

```
HI_S32 HI_MPI_VENC_SetMpeg4Param(VENC_CHN VeChn, const VENC_PARAM_MPEG4_S
*pstMpeg4Param);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstMpeg4Param	MPEG-4 协议编码通道的高级参数集合。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 MPEG-4 协议编码通道的高级参数。
- 高级参数主要由 5 个参数组成：
  - bSplitEnable：当前帧是否进行 packet 分割。
  - u32SplitMode：packet 分割模式，u32SplitMode = 0 表示按照 bit 数来进行分割，u32SplitMode = 1 表示按照宏块行进行分割。u32SplitMode>=2 不被支持。

    u32SplitMode 必须在 bSplitEnable 为真的条件下有效，否则，无效。
  - u32PacketSize：packet 的大小，当 u32SplitMode 不同时，u32PacketSize 表示不同的意义。当 u32SplitMode = 0 时，u32PacketSize 表示平均每个 packet 的 bit 数的多少，编码器从图像的左上角开始，从左至右，从上至下，按光栅顺序按宏块进行编码。每个 packet 大小按照 u32PacketSize 进行划分。最终编码 packet 大小不一定与用户设置的 u32PacketSize 严格相等，存在少许的误差量，且实际的偏移量总是正数。当编码至图像的最后一个 packet 时，剩余的宏块 bit 数不足 u32PacketSize，剩余的所有宏块被编码成一个 packet。当 u32SplitMode = 1 时，u32PacketSize 表示每个 packet 占图像宏块行数，u32PacketSize 的单位为宏块行。且当编码至图像的最后几行，不足 u32PacketSize 时，剩余的宏块行被划分为一个 packet。
  - u32HWSize：帧间预测时的水平搜索窗的大小，只能设置为 0 或 1。
  - u32VWSize：帧间预测时的垂直搜索窗的大小，只能设置为 0。
- MPEG-4 编码通道的 packet 分割与 h.264 编码通道的 slice 分割类似
- MPEG-4 编码通道的搜索窗大小与 h.264 搜索窗大小类似。但是可以设置的参数范围不一样。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 本接口是 MPEG-4 编码通道的高级接口，系统内部有默认的 mpeg4param 参数，用户不是必须调用此接口。系统默认 bSplitEnable=HI_FALSE，u32HWSize=1，u32VWSize=0。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个帧编码时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetMpeg4Param 接口，获取当前编码通道的 Mpeg4Param 配置，然后再进行设置。

【举例】

```
HI_S32 SetMpeg4Param(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_MPEG4_S stParamMpeg4;
    VENC_CHN VeChnId = 0;
    //...omit other thing
    s32Ret = HI_MPI_VENC_GetMpeg4Param(VeChnId,  &stParamMpeg4);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetMpeg4Param err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    stParamMpeg4. bSplitEnable = HI_TRUE;
    stParamMpeg4. u32SplitMode = 1;
    stParamMpeg4. u32PacketSize = 10;

    s32Ret = HI_MPI_VENC_SetMpeg4Param(VeChnId,  &stParamMpeg4);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetMpeg4Param err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetMpeg4Param

【描述】

获取 MPEG-4 协议编码通道的高级参数配置。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VENC_GetMpeg4Param(VENC_CHN VeChn, VENC_PARAM_MPEG4_S
*pstMpeg4Param);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstMpeg4Param	MPEG-4 协议编码通道的高级参数配置。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 MPEG-4 协议编码通道的高级参数配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetJpegParam 的举例。

【相关主题】

无。

HI_MPI_VENC_SetMjpegParam

【描述】

设置 JPEG 协议编码通道的高级参数。

【语法】

```
HI_S32 HI_MPI_VENC_SetMjpegParam(VENC_CHN VeChn, const VENC_PARAM_MJPEG_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
*pstMjpegParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstMjpegParam	JPEG 协议编码通道的高级参数集合	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 JPEG 协议编码通道的高级参数
- 高级参数主要由 4 个参数组成
  - U8YQt[64]，u8CbQt[64]，u8CrQt[64]：对应三个量化表空间，用户可以通过这三个参数设置用户的量化表。
  - u32MCUPerECS：每个 Ecs 中包含多少 Mcu。系统模式 u32MCUPerECS = 0，表示当前帧的所有的 MCU 被编码为一个 ECS。u32MCUPerECS 的最小值为 0，最大值不超过(picwidth+15)>>4 x (picheight+15)>>4 x 2。
- 如果用户想使用自己的量化表，在设置量化表的同时，请将 Qfactor 设置为 50。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个帧编码时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetMjpegParam 接口，获取当前编码通道的 MjpegParam 配置，然后再进行设置。

【举例】

```
HI_S32 SetMjpegParam(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_PARAM_MJPEG_S stParamMjpeg;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    VENC_CHN VeChnId = 0;
    //...omit other thing
    s32Ret = HI_MPI_VENC_GetMjpegParam(VeChnId,  &stParamMjpeg);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_GetMjpegParam err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }

    stParamMjpeg.u32MCUPerECS = 100;
    for (i = 0; i < 64; i++)
    {
        stParamMjpeg.u8YQt[i] = 16;
        stParamMjpeg.u8CbQt[i] = 17;
        stParamMjpeg.u8CrQt[i] = 18;
    }
    s32Ret = HI_MPI_VENC_SetMjpegParam(VeChnId,  &stParamMjpeg);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetMjpegParam err 0x%x\n",s32Ret);
        return HI_FAILURE;
    }
    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetMjpegParam

【描述】

获取 JPEG 协议编码通道的高级参数配置。

【语法】

```
HI_S32 HI_MPI_VENC_GetMjpegParam(VENC_CHN VeChn, VENC_PARAM_MJPEG_S
*pstMjpegParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstMjpegParam	Jpeg 协议编码通道的高级参数配置	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 JPEG 协议编码通道的高级参数配置。
- 本接口可在编码通道创建之后，编码通道销毁之前调用。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。

【举例】

请参见 HI_MPI_VENC_SetMjpegParam 的举例

【相关主题】

无。

HI_MPI_VENC_SetGrpFrmRate

【描述】

设置通道组帧率控制属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetGrpFrmRate(VENC_GRP VeGroup, const
GROUP_FRAME_RATE_S *pstGrpFrmRate);
```

【参数】

参数名称	描述	输入/输出
VeGroup	编码通道组号。 取值范围：[0, VENC_MAX_GRP_NUM)。	输入
pstGrpFrmRate	通道组帧率控制属性指针。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 通道组帧率控制属性包括了输入帧率 ViFrmRate 和输出帧率 VpssFrmRate 两部分。
- 设置未创建的通道组的属性，则返回失败。
- 如果 pstGrpFrmRate 为空，则返回失败。
- 设置通道组帧率控制属性时，输入帧率 ViFrmRate 和输出帧率 VpssFrmRate 必须同时大于 0 或同时等于-1。
- 当输入帧率 ViFrmRate 和输出帧率 VpssFrmRate 均大于 0 时，输入帧率 ViFrmRate 必须大于或等于输出帧率 VpssFrmRate。例如，ViFrmRate = 30，VpssFrmRate = 15 时，表示每输入 30 帧图像，只取其中的 15 帧进行编码。
- 当输入帧率 ViFrmRate 和输出帧率 VpssFrmRate 均等于-1 时，此时表示不进行帧率控制。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetGrpFrmRate

【描述】

获取通道组帧率控制属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetGrpFrmRate(VENC_GRP VeGroup, GROUP_FRAME_RATE_S
*pstGrpFrmRate);
```

【参数】

参数名称	描述	输入/输出
VeGroup	编码通道组号。 取值范围：[0, VENC_MAX_GRP_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

pstGrpFrmRate	通道组帧率控制属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 获取未创建的通道组的属性，返回失败 HI_ERR_VENC_UNEXIST。
- 如果 pstGrpFrmRate 为空，则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetRcPara

【描述】

设置编码通道码率控制器的高级参数。

【语法】

```
HI_S32 HI_MPI_VENC_SetRcPara(VENC_CHN VeChn, VENC_RC_PARAM_S *pstRcPara);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstRcPara	编码通道码率控制器的高级参数。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 编码通道码率控制器的高级参数都有默认值，而不是必须调用这个接口才能启动编码通道。
- 建议用户先调用 HI_MPI_VENC_GetRcPara 接口，获取 RC 高级参数，然后修改相应参数，再调用本接口对高级参数进行设置。
- RC 高级参数现仅支持 H.264 CBR、H.264 VBR 码率控制模式。
- 当码率控制模式改变时，所有的码率控制高级参数自动恢复默认值，用户需要重新设置 RC 高级参数。
- 码率控制器的高级参数由以下参数组成：
  - u32ThrdI[12]，u32ThrdP[12]：基于纹理的宏块级码率控制，分别衡量 I 帧，P 帧的宏块复杂度的一组阈值。这 12 个阈值按照从小到大的顺序依次排列，每个阈值的取值范围为[0, 255]。这组阈值用于在进行宏块级码率控制时，根据图像复杂度对每个宏块的 Qp 进行适当的调整。如果当前宏块的图像复杂度处于某两阈值之间时，当前宏块的 Qp 值就在宏块行起始 Qp 值的基础上加上较小阈值的索引号，譬如：u32Thrd[2]<C<u32Thr[3]，C 表示图像复杂度，当前宏块的 Qp 值在宏块行起始 Qp 值的基础上加 2。当前通道设置的码率是高码率时，为了保存图像细节，可以将组阈值设置为 255，图像的细节可以得到最好的保存；当前通道设置的码率为低码率时，对于过于复杂的图像细节，可以适当的减小阈值，忽略部分图像细节，在不影响图像整体质量的情况下，保证码率满足低码率条件。所有的数值都设置为 255 表示关闭基于纹理的宏块级码率控制，u32ThrdI[12]默认值为 [5, 5, 5, 10, 10, 10, 255, 255, 255, 255, 255, 255]，u32ThrdP[12]默认值为[5, 5, 5, 255, 255, 255, 255, 255, 255, 255, 255, 255]。
  - u32QpDelta：基于行的宏块级码率控制，在宏块级码率控制时，每一行宏块的起始 Qp 相对于帧起始 Qp 的波动幅度值。对于码率波动较严格的场景下，可以尝试将此参数调大，实现更加精确的码率控制，但可能会到导致某些帧图像内部的图像质量有差异。在高码率时，该值推荐为 0；中码率时推荐该值为 0 或 1；低码率时推荐该值为 2～5。默认值为 2。
  - u32MinIprop&u32MaxIprop：CBR 高级参数，分别表示的是最小 IP 比例和最大 IP 比例。IP 比例表示 I 帧和 P 帧的 Bits 数的比例。这两个值用于钳位 IP 比例的范围。当 u32MinIprop 被调整较大时，会导致 I 帧清晰，P 帧模糊。当 u32MaxIprop 被调整较小时，会导致 I 帧模糊，P 帧清晰。在正常情况下不建议对 IP 大小比进行约束，避免带来呼吸效应和码率波动，默认 u32MinIprop 设为 1，u32MaxIprop 设为 20。在对 I 帧大小有约束的场景时，可以根据对 I 帧大小波动的需求来设置 u32MinIprop 和 u32MaxIprop 的值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- u32MaxQp&u32MinQp&u32MaxStartQp：CBR 高级参数，u32MaxQp、u32MinQp 表示的是当前帧的最大 Qp 和最小 Qp。这个钳位效果最强烈，所有其他对图像 Qp 的调整，如宏块级码率控制，最终都会被约束到这个最大 Qp 和最小 Qp。u32MaxStartQp 表示的是帧级码率控制输出的最大 QP，取值范围是[u32MinQp, u32MaxQp]。宏块级码率控制可能会使某些宏块的 QP 大于或小于 u32MaxStartQp，但都会被钳位到[u32MinQp, u32MaxQp]。默认值 u32MinQp 为 10，u32MaxQp 为 51，u32MaxStartQp 为 51。在对质量无特殊需求下，建议不更改此组参数。
- u32MaxPPDeltaQp&u32MaxIPDeltaQp：CBR 高级参数，表示的是连续两个 P 帧起始 Qp 之间的最大差值和连续 IP 帧起始 Qp 之间的最大差值。当出现大运动，场景切换等变化时，保证码率平稳条件下，可能会导致连续 P 帧之间 Qp 差异过大，导致图像马赛克等，可以通过调整这两个参数对 Qp 变化进行钳位，避免出现图像质量波动。u32MaxPPDelta 默认值为 3；u32MaxIPdeltaQP 默认值为 5，可以根据场景对质量与码率要求的差异修改。
- bLostFrmOpen：CBR 高级参数，表示的是在瞬时码率超出的情况下，是否启动丢帧，保证码率稳定。系统默认为启动丢帧。
- u32LostFrmBpsThr：CBR 高级参数，表示启动丢帧的瞬时码率的阈值，单位为 bit/s。系统默认为 80x1024x1024，应根据实际需求决定是否允许丢帧和丢帧时的码率。
- enSuperFrmMode：CBR、VBR 高级参数，表示的是对于超大帧的处理模式，CBR 提供了无处理、丢帧、重编三种处理模式。无处理模式：对于超大帧图像，没有特殊处理方式；丢帧模式：丢弃当前超大帧；重编：对当前超大帧进行重编。系统默认超大帧处理方式为无处理模式。
- u32SuperIFrmBitsThr&u32SuperPFrmBitsThr：CBR、VBR 高级参数，表示的启动超大帧模式的超大 I 帧和超大 P 帧的 Bit 数阈值。如果允许丢弃某些帧的情况下，不建议选择重编，可避免消耗过多硬件资源和导致图像质量突变，同时建议设置阈值应当合理，以避免出现连续丢失多帧或频繁重编的情况。
- s32IPQPDelta：CBR 高级参数，表示的是平均 Qp 值与当前 I 帧 Qp 的差值，此参数可为负值。可用于调整 I 帧过大和呼吸效应。系统默认值为 2，增大此值，I 帧变清晰。
- u32RQRatio[8]：CBR 高级参数，表示的是码率稳定和质量稳定在进行码率控制中的权重。u32RQRatio[i]/100 表示质量稳定权重，1- u32RQRatio[i]/100 表示码率稳定权重，譬如：u32RQRatio[i] =75，表示质量稳定占 75%的权重，码率稳定占 25%的权重。CBR 提供了 8 种场景模式，分别是：Normal(正常场景), Move(运动场景), Still(静止场景), StillToMove(静止转为运动场景), MoveToStill(运动转为静止场景), SceneSwitch(场景切换), SharpMove(剧烈运动场景), Init(程序初始化)，分别对应了 u32RQRatio 中 0～7 的索引号。现在系统默认值为 u32RQRatio []= {75, 75, 75, 50, 50, 20, 30, 0}。用户可以设置不同场景下，码率控制的权重，以满足特定的需求，譬如：在大运动场景下，质量稳定的比例为 30%，码率稳定的比例为 70%，用户可以将质量比例继续调小，码率的波动变的更小。
- s32DeltaQP：VBR 高级参数，表示的是在 VBR 质量出现波动的过程中，帧与帧之间的最大的 Qp 变化。这个参数可用于防止出现马赛克。系统默认为 2。
- s32ChangePos：VBR 高级参数，表示的是 VBR 开始调整 Qp 时码率与最大码率的比值。系统默认为 90，如果在内容变化剧烈且要求不超出最大码率的场

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

景，建议减小此值，同时增大 s32DeltaQP，但码率控制稳定时的码率偏小和质量偏差。

- pRcParam：由用户制定的 RC 高级参数，保留，暂时没有使用。

- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- RC 高级参数的详细使用说明请参见《码率控制使用说明》。

【举例】

```
HI_S32 SetRcParam(HI_VOID)
{
    HI_S32 s32Ret = HI_FAILURE;
    VENC_RC_PARAM_S stVencRcPara;
    VENC_CHN VeChnId = 0;
    //...omit other thing
    s32Ret = HI_MPI_VENC_GetRcPara(VeChnId, &stVencRcPara);
    if (HI_SUCCESS != s32Ret)
    {
    printf("HI_MPI_VENC_GetRcPara err 0x%x\n", s32Ret);
       return HI_FAILURE;
    }

    stVencRcPara.stParamH264Cbr.enSuperFrmMode = SUPERFRM_DISCARD;

    s32Ret = HI_MPI_VENC_SetRcPara(VeChnId, &stVencRcPara);
    if (HI_SUCCESS != s32Ret)
    {
        printf("HI_MPI_VENC_SetRcPara err 0x%x\n", s32Ret);
    return HI_FAILURE;
    }

    //...omit other thing

    return HI_SUCCESS;
}
```

【相关主题】

无。

HI_MPI_VENC_GetRcPara

【描述】

获取通道码率控制高级参数。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_VENC_GetRcPara(VENC_CHN VeChn, VENC_RC_PARAM_S *pstRcPara);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstRcPara	通道码率控制参数指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 H.264 编码通道码率控制的高级参数。各参数的含义请具体请参见 VENC_RC_PARAM_S。
- 如果 pstRcPara 为空，则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetH264eRefMode

【描述】

设置 H.264 编码通道跳帧参考模式。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264eRefMode(VENC_CHN VeChn, VENC_ATTR_H264_REF
_MODE_E enRefMode);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
enRefMode	H.264 编码通道跳帧参考模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 创建 H.264 协议编码通道时，默认的跳帧参考模式是 1 倍跳帧参考模式（即没有开启跳帧参考）。如果用户需要修改编码通道的跳帧参考模式，建议在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 在 SVC profile 下不支持跳帧参考设置。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetH264eRefMode

【描述】

获取 H.264 编码通道跳帧参考模式。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264eRefMode(VENC_CHN VeChn,
VENC_ATTR_H264_REF_MODE_E* penRefMode);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
penRefMode	H.264 编码通道跳帧参考模式。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

如果 penRefMode 为空，则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetH264eRefParam

【描述】

设置 H.264 编码通道高级跳帧参考参数。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264eRefParam(VENC_CHN VeChn,
VENC_ATTR_H264_REF_PARAM_S * pstRefParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstRefParam	H.264 编码通道高级跳帧参考参数。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 如果 pstRefParam 为空，则返回失败。
- 创建 H.264 协议编码通道时，默认跳帧参考模式是 1 倍跳帧参考模式。如果用户需要修改编码通道的跳帧参考，建议在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 在 SVC profile 下不支持跳帧参考设置。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetH264eRefParam

【描述】

获取 H.264 编码通道高级跳帧参考参数。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264eRefParam(VENC_CHN VeChn,
VENC_ATTR_H264_REF_PARAM_S * pstRefParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	编码通道号。 取值范围：[0, VENC_MAX_CHN_NUM)。	输入
pstRefParam	H.264 编码通道高级跳帧参考参数。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

如果 pstRefParam 为空，则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetColor2GreyConf

【描述】

设置彩转灰功能的全局配置，即调用此接口会影响所有 GROUP。

【语法】

```
HI_S32 HI_MPI_VENC_SetColor2GreyConf (const GROUP_COLOR2GREY_CONF_S*
pstGrpColor2GreyConf );
```

【参数】

参数名称	描述	输入/输出
pstGrpColor2GreyConf	彩转灰功能的全局配置参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- GROUP_COLOR2GREY_CONF_S 由 3 个参数组成：
  - bEnable。所有 GROUP 是否开启彩转灰功能的全局开关。
  - u32MaxWidth。要开启彩转灰功能的 GROUP 的最大宽度。一个 GROUP 的宽度就是注册到其中的通道的宽度。
  - u32MaxHeight。要开启彩转灰功能的 GROUP 的最大高度。一个 GROUP 的高度就是注册到其中的通道的高度。
- 设置彩转灰功能的全局配置，即调用此接口会影响所有 GROUP。一个 GROUP 如果要调用 HI_MPI_VENC_SetGrpColor2Grey 开启彩转灰功能，必须先调用此接口，并置 bEnable 为 HI_TRUE。
- 要改变上述 3 个参数中的任何一个，必须满足：所有 GROUP 均未开启彩转灰功能。

【举例】

参考 VENC 的 sample。

【相关主题】

无。

HI_MPI_VENC_GetColor2GreyConf

【描述】

获取彩转灰功能的全局配置。

【语法】

```
HI_S32 HI_MPI_VENC_GetColor2GreyConf (const GROUP_COLOR2GREY_CONF_S*
pstGrpColor2GreyConf )
```

【参数】

参数名称	描述	输入/输出
pstGrpColor2GreyConf	彩转灰功能的全局配置。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

如果 pstGrpColor2GreyConf 为空，则返回失败。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetGrpColor2Grey

【描述】

开启或关闭一个 GROUP 的彩转灰功能。

【语法】

```
HI_S32 HI_MPI_VENC_SetGrpColor2Grey(VENC_GRP VeGroup,const
GROUP_COLOR2GREY_S* pstGrpColor2Grey)
```

【参数】

参数名称	描述	输入/输出
VeGroup	通道组号	输入
pstGrpColor2Grey	开启或关闭彩转灰功能的参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

一个 GROUP 如果要调用此 API 开启彩转灰功能，必须满足以下条件：

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 先调用 HI_MPI_VENC_SetColor2GreyConf 设置彩转灰功能的全局配置，并置 bEnable 为 HI_TRUE。
- 此 GROUP 中必须已经有通道注册。
- 注册到 GROUP 中的通道的宽高必须小于彩转灰功能全局配置中的最大宽高。

【举例】

参考 VENC 的 sample。

【相关主题】

无。

HI_MPI_VENC_GetGrpColor2Grey

【描述】

获取一个 GROUP 是否开启彩转灰功能。

【语法】

```
HI_S32 HI_MPI_VENC_GetGrpColor2Grey(VENC_GRP VeGroup,GROUP_COLOR2GREY_S*
pstGrpColor2Grey)
```

【参数】

参数名称	描述	输入/输出
VeGroup	通道组号	输入
pstGrpColor2Grey	开启或关闭彩转灰功能的参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

如果 pstGrpColor2Grey 为空，则返回失败。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

无。

HI_MPI_VENC_SetGrpCrop

【描述】

设置 GROUP 的裁剪属性。

【语法】

```
HI_S32 HI_MPI_VENC_SetGrpCrop(VENC_GRP VeGroup, const GROUP_CROP_CFG_S
*pstGrpCropCfg);
```

【参数】

参数名称	描述	输入/输出
VeGroup	通道组号	输入
pstGrpCropCfg	通道组裁剪属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置通道组的裁剪属性。
- 通道组会优先进行图像裁剪，然后基于裁剪之后的图像尺寸，与编码通道的尺寸进行比较，决定是否进行缩小。
- 本接口必须在通道向通道组注册之后，通道从通道组注销之前调用。
- 通道组内的通道注销之前，用户必须保证通道组的裁剪功能关闭。
- 裁剪属性由两部分组成：
  - bEnable：是否使能通道组裁剪功能。
  - stRect：裁剪区域属性，包括裁剪区域起始点坐标，以及裁剪区域的尺寸。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

无。

HI_MPI_VENC_GetGrpCrop

【描述】

获取 GROUP 的裁剪属性。

【语法】

```
HI_S32 HI_MPI_VENC_GetGrpCrop(VENC_GRP VeGroup, GROUP_CROP_CFG_S
*pstGrpCropCfg);
```

【参数】

参数名称	描述	输入/输出
VeGroup	通道组号	输入
pstGrpCropCfg	通道组裁剪属性	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取通道组的裁剪属性。
- 本接口必须在通道组创建之后，通道组销毁之前调用。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetJpegSnapMode

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

设置 JPEG 抓拍通道的抓拍模式，只适用于 Hi3518/Hi3516C。

【语法】

```
HI_S32 HI_MPI_VENC_SetJpegSnapMode(VENC_CHN VeChn, VENC_JPEG_SNAP_MODE_E
enJpegSnapMode);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
enJpegSnapMode	通道抓拍模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 JPEG 抓拍通道的抓拍模式。
- 本接口必须在编码通道创建之后，编码通道销毁之前调用。
- 本接口只对 JPEG 类型的编码通道有效。
- 通道抓拍模式有两种
  - JPEG_SNAP_ALL 模式，通道启动接收图像之后，编码所有接收的图像；
  - JPEG_SNAP_FLASH 模式，通道启动接收图像之后，只编码被闪光的图像。
- 创建 JPEG 通道之后，JPEG 通道默认处于 JPEG_SNAP_ALL 模式。当用户需要只抓拍被闪光的图像时，可以调用本接口，设置 JPEG 通道为 JPEG_SNAP_FLASH 模式。
- 当用户需要使用抓拍通道的 JPEG_SNAP_FLASH 模式时，必须配合 VI 的闪光灯功能一起使用。
- 当 JPEG 设置为闪光灯抓拍模式时，通道组只能接收来自 VIU 模块（不经过 VPSS 模块）的图像，否则 JPEG 抓拍不到任何图像。

【举例】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

无。

HI_MPI_VENC_GetJpegSnapMode

【描述】

获取 JPEG 编码通道的抓拍模式，只适用于 Hi3518/Hi3516C。

【语法】

```
HI_S32 HI_MPI_VENC_GetJpegSnapMode(VENC_CHN VeChn, VENC_JPEG_SNAP_MODE_E
*penJpegSnapMode);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
penJpegSnapMode	抓拍模式	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于获取 JPEG 通道的抓拍模式。
- 本接口只对 JPEG 类型的编码通道有效。
- 本接口必须在通道创建之后，通道销毁之前调用。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VENC_SetRcPriority

【描述】

设置码率控制的优先级类型。

【语法】

```
HI_S32 HI_MPI_VENC_SetRcPriority(VENC_CHN VeChn, VENC_RC_PRIORITY_E
enRcPriority);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
enRcPriority	优先级类型枚举。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置码率控制以目标码率还是以超大帧阈值高优先级。当码率控制以目标码率为高优先级时，在码率不足时可能编码出超大帧以补偿码率，这时超大帧不会重编。当码率控制以超大帧阈值为高优先级时，超大帧则会重编降比特数，结果可能导致码率不足。
- 本接口必须在编码通道创建之后，编码通道销毁之前调用。
- 本接口仅对 CBRv2 和 VBRv2 有效。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VENC_GetRcPriority

【描述】

设置码率控制的优先级类型。

【语法】

```
HI_S32 HI_MPI_VENC_GetRcPriority(VENC_CHN VeChn, VENC_RC_PRIORITY_E
*penRcPriority);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
penRcPriority	优先级类型指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口必须在编码通道创建之后，编码通道销毁之前调用。
- 本接口仅对 CBRv2 和 VBRv2 有效。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_SetLostFrameStrategy

【描述】

设置码率过高时编码的丢帧策略。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_VENC_SetLostFrameStrategy(VENC_CHN VeChn, const
VENC_RC_PARAM_LOSTFRM_S *pstLostFrmParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
pstLostFrmParam	丢帧策略结构体指针	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口必须在通道创建之后，通道销毁之前调用。
- 本接口只对 H264 的 CBRV2 和 VBRV2 编码有效。
- 码率过高时的丢帧模式有两种：一是正常丢帧，即连续丢掉不超过 u32EncFrmGaps 帧码流直到码率降到阈值以下；二是编码为 Pskip 帧，即连续编码不超过 u32EncFrmGaps 个 Pskip 帧码流直到码率降到阈值以下。
- 正常丢帧模式下如果 u32EncFrmGaps 不为 0，则限制丢帧个数，即最多只能丢弃 u32EncFrmGaps 帧码流；如果 u32EncFrmGaps 为 0，则不限制丢帧个数，即一直丢帧到码率将到阈值以下。
- Pskip 帧模式下如果 u32EncFrmGaps 不为 0，则限制编码 Pskip 帧个数，即最多只能编码 u32EncFrmGaps 个 Pskip 帧；如果 u32EncFrmGaps 为 0，则不限制编码 Pskip 帧个数，即编码 Pskip 帧直到码率降到阈值以下。但是如果当前帧是高级跳帧参考的 base 层，则不能编为 pskip 帧，这种情况下当前帧正常输出。
- 丢帧模式在打开丢帧开关 bLostFrmOpen 和设置阈值 u32LostFrmBpsThr 之后才会生效。
- RC 高级参数接口（HI_MPI_VENC_SetRcPara）和本接口都可以设置丢帧开关 bLostFrmOpen 和阈值 u32LostFrmBpsThr，以最后设置的为准。
- 当码率控制模式改变时，所有的码率控制高级参数自动恢复默认值，用户需要重新设置 RC 高级参数。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

无。

HI_MPI_VENC_GetLostFrameStrategy

【描述】

获取码率过高时编码的丢帧策略。

【语法】

```
HI_S32 HI_MPI_VENC_GetLostFrameStrategy(VENC_CHN VeChn, const
VENC_RC_PARAM_LOSTFRM_S *pstLostFrmParam);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
pstLostFrmParam	丢帧策略结构体指针	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

本接口必须在通道创建之后，通道销毁之前调用。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_AttachVbPool

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

将编码通道绑定到某个视频缓存 VB 池中。

【语法】

```
HI_S32 HI_MPI_VENC_AttachVbPool(VENC_CHN VeChn, VB_POOL hVbPool);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号。	输入
hVbPool	视频缓存 VB 池的 Id 号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h、hi_comm_vb.h
- 库文件：libmpi.a

【注意】

- 必须保证通道已创建，否则会返回通道未创建的错误码。
- 用户必须调用接口 HI_MPI_VB_CreatePool 创建一个视频缓存 VB 池，再通过调用接口 HI_MPI_VENC_AttachVbPool 把当前编码通道绑定到固定 PoolId 的 VB 池中。可以把多个编码通道绑定到同一个 VB 池中，但是不能把同一个编码通道绑定到多个 VB 池中。
- 当要切换当前编码通道绑定的 VB 池时，只需再调一次接口 HI_MPI_VENC_AttachVbPool 正确配置需要绑定到的 VB 池即可。
- 只有 H264 编码支持 UserVB 池方式，如果当前编码帧存分配方式使用的不是编码 UserVB 池，则返回通道不支持的错误码。
- hVbPool 必须是已创建 VB 池的有效 PoolId。

HI_MPI_VENC_DetachVbPool

【描述】

将编码通道从某个视频缓存 VB 池中解绑定。

【语法】

```
HI_S32 HI_MPI_VENC_DetachVbPool(VENC_CHN VeChn);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
VeChn	通道号。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h、hi_comm_vb.h
- 库文件：libmpi.a

【注意】

- 必须保证通道已创建，否则会返回通道未创建的错误码。
- 如果当前编码帧存分配方式使用的不是编码 UserVB 池，则返回通道不支持的错误码。

HI_MPI_VENC_SetH264VideoSignal

【描述】

设置 H.264 协议编码通道的 VideoSignal 参数。

【语法】

```
HI_S32 HI_MPI_VENC_SetH264VideoSignal(VENC_CHN VeChn, const
VENC_PARAM_H264_VIDEO_SIGNAL_S *pstH264VideoSignal);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
pstH264VideoSignal	H.264 协议编码通道的 VideoSignal 参数。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

- 本接口用于设置 H.264 协议编码通道的 VideoSignal 参数配置，每个参数的具体含义请参见 H.264。
- 本接口可在编码通道创建之后，编码通道销毁之前设置。本接口在编码过程中被调用时，等到下一个 I 帧时生效。
- 建议用户在创建通道之后，启动编码之前调用此接口，减少在编码过程中调用的次数。
- 建议用户在调用此接口之前，先调用 HI_MPI_VENC_GetH264VideoSignal 接口，获取当前编码通道的 VideoSignal 配置，然后再进行设置。

【举例】

无。

【相关主题】

无。

HI_MPI_VENC_GetH264VideoSignal

【描述】

获取 H.264 协议编码通道的 VideoSignal 参数。

【语法】

```
HI_S32 HI_MPI_VENC_GetH264VideoSignal(VENC_CHN VeChn,
VENC_PARAM_H264_VIDEO_SIGNAL_S *pstH264VideoSignal);
```

【参数】

参数名称	描述	输入/输出
VeChn	通道号	输入
pstH264VideoSignal	H.264 协议编码通道的 VideoSignal 参数。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，返回错误码。

【需求】

- 头文件：hi_comm_venc.h、mpi_venc.h
- 库文件：libmpi.a

【注意】

本接口可在编码通道创建之后，编码通道销毁之前设置。

【举例】

无。

【相关主题】

无。

6.4 数据类型

相关数据类型、数据结构定义如下：

- H264E_NALU_TYPE_E：定义 H.264 码流 NALU 类型。
- H264E_REFSLICE_TYPE_E：定义获取的 H.264 码流属于何种跳帧参考模式下的参考帧。
- H264E_REF_TYPE_E：定义 H.264 跳帧参考码流的帧类型以及参考属性。
- JPEGE_PACK_TYPE_E：定义 JPEG 码流的 PACK 类型。
- MPEG4E_PACK_TYPE_E：定义 MPEG4 码流的 PACK 类型。
- VENC_DATA_TYPE_U：定义码流结果联合体。
- VENC_PACK_S：定义帧码流包结构体。
- VENC_STREAM_INFO_H264_S：定义 H.264 协议码流特征信息。
- VENC_STREAM_INFO_JPEG_S：定义 JPEG 协议码流特征信息。
- VENC_STREAM_INFO_MPEG4_S：定义 MPEG-4 定义协议码流特征信息。
- VENC_STREAM_S：定义帧码流类型结构体。
- VENC_ATTR_H264_S：定义 H.264 编码器属性结构体。
- VENC_ATTR_MJPEG_S：定义 MJPEG 编码器属性结构体。
- VENC_ATTR_JPEG_S：定义 JPEG 抓拍编码器属性结构体。
- VENC_ATTR_MPEG4_S：定义 MPEG-4 编码器属性结构体。
- VENC_ATTR_S：定义编码器属性结构体。
- VENC_CHN_ATTR_S：定义编码通道属性结构体。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VENC_CHN_STAT_S：定义编码通道的状态结构体。
- VENC_PARAM_H264_SLICE_SPLIT_S：定义 H.264 编码通道 slice 分割属性。
- VENC_PARAM_H264_INTER_PRED_S：定义 H.264 编码通道帧间预测属性。
- VENC_PARAM_H264_INTRA_PRED_S：定义 H.264 编码通道帧内预测属性。
- VENC_PARAM_H264_TRANS_S：定义 H.264 编码通道变换、量化属性。
- VENC_PARAM_H264_ENTROPY_S：定义 H.264 编码通道熵编码属性。
- VENC_PARAM_H264_POC_S：定义 H.264 编码通道 POC 属性。
- VENC_PARAM_H264_DBLK_S：定义 H.264 编码通道 Deblocking 属性。
- VENC_PARAM_H264_VUI_S：定义 H.264 编码通道 VUI 属性。
- VENC_PARAM_JPEG_S：定义 JPEG 编码参数集合。
- VENC_PARAM_MJPEG_S：定义 MJPEG 协议编码通道高级参数结构体。
- VENC_PARAM_MPEG4_S：定义 MPEG-4 编码参数集合。
- VENC_ROI_CFG_S：定义编码通道感兴趣区域编码属性。
- VENC_ATTR_H264_REF_MODE_E：定义 H.264 编码的跳帧参考模式。
- VENC_ATTR_H264_REF_PARAM_S：定义 H.264 编码的高级跳帧参考参数。
- VENC_RC_ATTR_S：定义编码通道码率控制器属性。
- VENC_RC_MODE_E：定义编码通道码率控制器模式。
- VENC_ATTR_H264_CBR_S：定义 H.264 编码通道 CBR 属性结构。
- VENC_ATTR_H264_VBR_S：定义 H.264 编码通道 VBR 属性结构。
- VENC_ATTR_H264_FIXQP_S：定义 H.264 编码通道 Fixqp 属性结构。
- VENC_ATTR_H264_ABR_S：定义 H.264 编码通道 ABR 属性结构。
- VENC_ATTR_MPEG4_CBR_S：定义 MPEG-4 编码通道 CBR 属性结构。
- VENC_ATTR_MPEG4_VBR_S：定义 MPEG-4 编码通道 VBR 属性结构。
- VENC_ATTR_MPEG4_FIXQP_S：定义 MPEG-4 编码通道 Fixqp 属性结构。
- VENC_ATTR_MJPEG_FIXQP_S：定义 MJPEG 编码通道 Fixqp 属性结构。
- VENC_ATTR_MJPEG_CBR_S：定义 MJPEG 编码通道 CBR 属性结构。
- VENC_ATTR_MJPEG_VBR_S：定义 MJPEG 编码通道 VBR 属性结构。
- VENC_SUPERFRM_MODE_EN：定义码率控制中超大帧处理模式。
- VENC_PARAM_H264_CBR_S：定义 H264 协议编码通道 CBR 码率控制模式高级参数配置。
- VENC_PARAM_H264_VBR_S：定义 H264 协议编码通道 VBR 码率控制模式高级参数配置。
- VENC_PARAM_MJPEG_CBR_S：定义 MJPEG 协议编码通道 CBR 码率控制模式高级参数配置。
- VENC_PARAM_MJPEG_VBR_S：定义 MJPEG 协议编码通道 VBR 码率控制模式高级参数配置。
- VENC_PARAM_MPEG4_CBR_S：定义 MPEG4 协议编码通道 CBR 码率控制模式高级参数配置。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VENC_PARAM_MPEG4_VBR_S：定义 MPEG4 协议编码通道 VBR 码率控制模式高级参数配置。
- VENC_PARAM_H264_CBRV2_S：新码率控制算法 H264 CBR 模式高级参数配置。
- VENC_RC_PARAM_S：定义编码通道的码率控制高级参数。
- GROUP_CROP_CFG_S：定义通道组截取（Clip）参数。
- GROUP_FRAME_RATE_S：定义通道组帧率控制参数。
- GROUP_COLOR2GREY_S：定义彩转灰全局设置参数。
  GROUP_COLOR2GREY_CONF_S：定义通道组彩转灰参数。
- VENC_JPEG_SNAP_MODE_E：定义 JPEG 编码通道的抓拍模式。
- VENC_RECV_PIC_PARAM_S：接收图像参数。
- VENC_RC_PRIORITY_E：码流控制优先级类型。
- VENC_RC_LOSTFRM_MODE_E：丢帧模式枚举。
- VENC_RC_PARAM_LOSTFRM_S：丢帧策略结构体。
- VENC_PARAM_H264_VIDEO_SIGNAL_S：定义 H.264 编码通道 VideoSignal 属性

H264E_NALU_TYPE_E

【说明】

定义 H.264 码流 NALU 类型。

【定义】

```
typedef enum hiH264E_NALU_TYPE_E
{
    H264E_NALU_PSLICE   = 1,
    H264E_NALU_ISLICE   = 5,
    H264E_NALU_SEI      = 6,
    H264E_NALU_SPS      = 7,
    H264E_NALU_PPS      = 8,
    H264E_NALU_BUTT
} H264E_NALU_TYPE_E;
```

【成员】

成员名称	描述
H264E_NALU_PSLICE	PSLICE 类型。
H264E_NALU_ISLICE	ISLICE 类型。
H264E_NALU_SEI	SEI 类型。
H264E_NALU_SPS	SPS 类型。
H264E_NALU_PPS	PPS 类型。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

无。

H264E_REFSLICE_TYPE_E

【说明】

定义获取的 H.264 码流属于何种跳帧参考模式下的参考帧。

【定义】

```
typedef enum hiH264E_REFSLICE_TYPE_E
{
    H264E_REFSLICE_FOR_1X       = 1,
    H264E_REFSLICE_FOR_2X       = 2,
    H264E_REFSLICE_FOR_4X       = 5,
    H264E_REFSLICE_FOR_BUTT
} H264E_REFSLICE_TYPE_E;
```

【成员】

成员名称	描述
H264E_REFSLICE_FOR_1X	1 倍跳帧参考时的参考帧。
H264E_REFSLICE_FOR_2X	2 倍跳帧参考时的参考帧或 4 倍跳帧参考时用于 2 倍跳帧参考的参考帧。
H264E_REFSLICE_FOR_4X	4 倍跳帧参考时的参考帧。
H264E_REFSLICE_BUTT	非参考帧。

【注意事项】

无。

【相关数据类型及接口】

无。

H264E_REF_TYPE_E

【说明】

定义 H.264 跳帧参考码流的帧类型以及参考属性。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef enum hiH264E_REF_TYPE_E
{
    BASE_IDRSLICE = 0,                  //base层中的IDR帧
    BASE_PSLICE_REFBYBASE,              //base层中的P帧，用于base层中其他帧的参考
    BASE_PSLICE_REFBYENHANCE,           //base层中的P帧，用于enhance层中的帧的参考
    ENHANCE_PSLICE_REFBYENHANCE,        //enhance层中的P帧，用于enhance层中其他帧的参考
    ENHANCE_PSLICE_NOTFORREF,           //enhance层中的P帧，不用于参考
    ENHANCE_PSLICE_BUTT
} H264E_REF_TYPE_E;
```

【成员】

成员名称	描述
BASE_IDRSLICE	base 层中的 IDR 帧。
BASE_PSLICE_REFBYBASE	base 层中的 P 帧，用于 base 层中其他帧的参考。
BASE_PSLICE_REFBYENHANCE	base 层中的 P 帧，用于 enhance 层中的帧的参考。
ENHANCE_PSLICE_REFBYENHANCE	enhance 层中的 P 帧，用于 enhance 层中其他帧的参考。
ENHANCE_PSLICE_NOTFORREF	enhance 层中的 P 帧，不用于参考

【注意事项】

无。

【相关数据类型及接口】

无。

JPEGE_PACK_TYPE_E

【说明】

定义 JPEG 码流的 PACK 类型。

【定义】

```
typedef enum hiJPEGE_PACK_TYPE_E
{
    JPEGE_PACK_ECS = 5,
    JPEGE_PACK_APP = 6,
    JPEGE_PACK_VDO = 7,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    JPEGE_PACK_PIC = 8,
    JPEGE_PACK_BUTT
} JPEGE_PACK_TYPE_E;
```

【成员】

成员名称	描述
JPEGE_PACK_ECS	ECS 类型。
JPEGE_PACK_APP	APP 类型。
JPEGE_PACK_VDO	VDO 类型。
JPEGE_PACK_PIC	PIC 类型。

【注意事项】

无。

【相关数据类型及接口】

无。

MPEG4E_PACK_TYPE_E

【说明】

定义 MPEG-4 码流的 PACK 类型。

【定义】

```
typedef enum hiMPEG4E_PACK_TYPE_E
{
    MPEG4E_PACK_VOP_P = 1,
    MPEG4E_PACK_VOP_I = 5,
    MPEG4E_PACK_VOS   = 6,
    MPEG4E_PACK_VO    = 7,
    MPEG4E_PACK_VOL   = 8,
    MPEG4E_PACK_GVOP  = 9

} MPEG4E_PACK_TYPE_E;
```

【成员】

成员名称	描述
MPEG4E_PACK_VOS	Video object sequence types。
MPEG4E_PACK_VO	Video object types。
MPEG4E_PACK_VOL	Video object layer types。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

MPEG4E_PACK_GVOP	Group of video object plane types。
MPEG4E_PACK_VOP_P	video object plane types for p frame。
MPEG4E_PACK_VOP_I	video packet types for I frame。

【注意事项】

无。

【相关数据类型及接口】

无。

VENC_DATA_TYPE_U

【说明】

定义码流结果类型。

【定义】

```
typedef union hiVENC_DATA_TYPE_U
{
    H264E_NALU_TYPE_E   enH264EType;
    JPEGE_PACK_TYPE_E   enJPEGEType;
    MPEG4E_PACK_TYPE_E  enMPEG4EType;
}VENC_DATA_TYPE_U;
```

【成员】

成员名称	描述
enH264EType	H.264 码流包类型。
enJPEGEType	JPEG 码流包类型。
enMPEG4EType	MPEG-4 码流包类型。

【注意事项】

无。

【相关数据类型及接口】

- H264E_NALU_TYPE_E
- JPEGE_PACK_TYPE_E
- MPEG4E_PACK_TYPE_E

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_PACK_S

【说明】

定义帧码流包结构体。

【定义】

```
typedef struct hiVENC_PACK_S
{
    HI_U32              u32PhyAddr[2];
    HI_U8               *pu8Addr[2];
    HI_U32              u32Len[2];
    VENC_DATA_TYPE_U    DataType;
    HI_U64              u64PTS;
    HI_BOOL             bFieldEnd;
    HI_BOOL             bFrameEnd;
}VENC_PACK_S;
```

【成员】

成员名称	描述
pu8Addr[2]	码流包首地址。
u32PhyAddr[2]	码流包物理地址。
u32Len[2]	码流包长度。
DataType	码流类型，支持 H.264/JPEG/MPEG-4 协议类型的数据包。
u64PTS	时间戳。单位：us。
bFieldEnd	场结束标识。 Hi3531 不支持场编码，该字段无效。
bFrameEnd	帧结束标识。 取值范围： HI_TRUE：该码流包是该帧的最后一个包。 HI_FALSE：该码流包不是该帧的最后一个包。

【注意事项】

无。

【相关数据类型及接口】

VENC_DATA_TYPE_U

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_STREAM_INFO_H264_S

【说明】

定义 H.264 协议码流特征信息。

【定义】

```
typedef struct hiVENC_STREAM_INFO_H264_S
{
    HI_U32 u32PicBytesNum;
    HI_U32 u32PSkipMbNum;
    HI_U32 u32IpcmMbNum;
    HI_U32 u32Inter16x8MbNum;
    HI_U32 u32Inter16x16MbNum;
    HI_U32 u32Inter8x16MbNum;
    HI_U32 u32Inter8x8MbNum;
    HI_U32 u32Intra16MbNum;
    HI_U32 u32Intra8MbNum;
    HI_U32 u32Intra4MbNum;
    H264E_REFSLICE_TYPE_E enRefSliceType;
    H264E_REF_TYPE_E      enRefType;
    HI_U32 u32UpdateAttrCnt;
}VENC_STREAM_INFO_H264_S;
```

【成员】

成员名称	描述
u32PicBytesNum	编码当前帧的字节（BYTE）数
u32PSkipMbNum	编码当前帧中采用跳跃（SKIP）编码模式的宏块数
u32IpcmMbNum	编码当前帧中采用 IPCM 编码模式的宏块数
u32Inter16x8MbNum	编码当前帧中采用 Inter16x8 预测模式的宏块数
u32Inter16x16MbNum	编码当前帧中采用 Inter16x16 预测模式的宏块数
u32Inter8x16MbNum	编码当前帧中采用 Inter8x16 预测模式的宏块数
u32Inter8x8MbNum	编码当前帧中采用 Inter8x8 预测模式的宏块数
u32Intra16MbNum	编码当前帧中采用 Intra16 预测模式的宏块数
u32Intra8MbNum	编码当前帧中采用 Intra8 预测模式的宏块数
u32Intra4MbNum	编码当前帧中采用 Intra4 预测模式的宏块数
enRefSliceType	编码当前帧属于何种跳帧参考模式下的参考帧
enRefType	高级跳帧参考下的编码帧类型

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32UpdateAttrCnt	通道属性或参数(包含 RC 参数)被设置的次数

【注意事项】

保存 H.264 码流时，可只选择相应跳帧参考模式下的参考帧：

- 当跳帧参考模式是 1 倍跳帧参考模式时，可只保存 enRefSliceType 等于 H264E_REFSLICE_FOR_1X 的码流。
- 当跳帧参考模式是 2 倍跳帧参考模式时，可只保存 enRefSliceType 等于 H264E_REFSLICE_FOR_2X 的码流。
- 当跳帧参考模式是 4 倍跳帧参考模式时，可只保存 enRefSliceType 等于 H264E_REFSLICE_FOR_4X 的码流，或同时保存 enRefSliceType 等于 H264E_REFSLICE_FOR_2X 和 H264E_REFSLICE_FOR_4X 的码流。

【相关数据类型及接口】

- VENC_DATA_TYPE_U
- H264E_REFSLICE_TYPE_E

VENC_STREAM_INFO_JPEG_S

【说明】

定义 JPEG 协议码流特征信息。

【定义】

```
typedef struct hiVENC_STREAM_INFO_JPEG_S
{
    HI_U32 u32PicBytesNum;
    HI_U32 u32UpdateAttrCnt;
    HI_U32 u32Qfactor;
}VENC_STREAM_INFO_JPEG_S
```

【成员】

成员名称	描述
u32PicBytesNum	一帧 jpeg 码流大小，以字节（byte）为单位。
u32UpdateAttrCnt	通道属性或参数(包含 RC 参数)被设置的次数。
u32Qfactor	编码当前帧的 Qfactor。

【注意事项】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关数据类型及接口】

VENC_DATA_TYPE_U

VENC_STREAM_INFO_MPEG4_S

【说明】

定义 MPEG-4 协议码流特征信息。

【定义】

```
typedef struct hiVENC_STREAM_INFO_MPEG4_S
{
   HI_U32 u32PicBytesNum;
   HI_U32 u32UpdateAttrCnt;
}VENC_STREAM_INFO_MPEG4_S;
```

【成员】

成员名称	描述
u32PicBytesNum	一帧 mpeg-4 码流大小，以字节（byte）为单位。
u32UpdateAttrCnt	通道属性或参数(包含 RC 参数)被设置的次数。

【注意事项】

无。

【相关数据类型及接口】

VENC_DATA_TYPE_U

VENC_STREAM_S

【说明】

定义帧码流类型结构体。

【定义】

```
typedef struct hiVENC_STREAM_S
{
    VENC_PACK_S  *pstPack;
    HI_U32        u32PackCount;
    HI_U32        u32Seq;
    union
    {
        VENC_STREAM_INFO_H264_S stH264Info;
        VENC_STREAM_INFO_JPEG_S stJpegInfo;
        VENC_STREAM_INFO_MPEG4_S stMpeg4Info;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    };
}VENC_STREAM_S
```

【成员】

成员名称	描述
pstPack	帧码流包结构。
u32PackCount	一帧码流的所有包的个数。
u32Seq	码流序列号。 按帧获取帧序号；按包获取包序号。
stH624Info/ stJpegInfo/stMpeg4Info	码流特征信息。

【注意事项】

无。

【相关数据类型及接口】

- VENC_PACK_S
- HI_MPI_VENC_GetStream

VENC_ATTR_H264_S

【说明】

定义 H.264 编码属性结构体。

【定义】

```
typedef struct hiVENC_ATTR_H264_S
{
    HI_U32  u32MaxPicWidth;
    HI_U32  u32MaxPicHeight;
    HI_U32  u32BufSize;
    HI_U32  u32Profile;
    HI_BOOL bByFrame;
    HI_S32  bField;
    HI_S32  bMainStream;
    HI_U32  u32Priority;
    HI_BOOL bVIField;
    HI_U32  u32PicWidth;
    HI_U32  u32PicHeight;
}VENC_ATTR_H264_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32MaxPicWidth	编码图像最大宽度。 取值范围：[MIN_WIDTH,MAX_WIDTH]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 静态属性。
u32PicWidth	编码图像宽度。 取值范围：[MIN_WIDTH, u32MaxPicWidth]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 动态属性。
u32MaxPicHeight	编码图像最大高度。 取值范围：[MIN_HEIGHT,MAX_HEIGHT]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 静态属性。
u32PicHeight	编码图像高度。 取值范围：[MIN_HEIGHT, u32MaxPicHeight]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 动态属性。
u32BufSize	码流 buffer 大小。 取值范围：[Min, Max]，以 byte 为单位。 必须是 64 的整数倍。 推荐值：一幅最大编码图像大小。推荐值为 u32MaxPicWidth × u32MaxPicHeight ×1.5 byte。 Min：码流 buffer 一般模式为一幅最大编码图像大小的 1/2，省内存模式为 32*1024 bytes。 Max：无限制，但是会消耗更多的内存。 静态属性。
bByFrame	帧/包模式获取码流。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：按帧获取。 • HI_FALSE：按包获取。 静态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bField	帧场编码模式。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：场编码。 • HI_FALSE：帧编码。 只支持帧编码。 静态属性。
bMainStream	主次码流标识。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：主码流。 • HI_FALSE：次码流。 只支持主码流。 静态属性。
u32Priority	通道优先级。 目前未使用，取值不限。
bVIField	输入图像的帧场标志。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：场。 • HI_FALSE：帧。 目前未使用，取值不限。 建议用户取值为 HI_FALSE。 静态属性。
u32Profile	编码的等级。 取值范围：[0, 3]。 0：Baseline。 1：Main Profile。 2：High Profile。 3：SVC-T Profile。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_ATTR_MJPEG_S

【说明】

定义 MJPEG 编码属性结构体。

【定义】

```
typedef struct hiVENC_ATTR_MJPEG_S
{
   HI_U32  u32MaxPicWidth;
   HI_U32  u32MaxPicHeight;
   HI_U32   u32BufSize;
   HI_BOOL  bByFrame;
   HI_BOOL  bMainStream;
   HI_BOOL  bVIField;
   HI_U32   u32Priority;
   HI_U32   u32PicWidth;
   HI_U32   u32PicHeight;
}VENC_ATTR_MJPEG_S;
```

【成员】

成员名称	描述
u32MaxPicWidth	编码图像最大宽度。 取值范围：[MIN_WIDTH,MAX_WIDTH]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 静态属性。
u32PicWidth	编码图像宽度。 取值范围：[MIN_WIDTH, u32MaxPicWidth]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 动态属性。
u32MaxPicHeight	编码图像最大高度。 取值范围：[MIN_HEIGHT,MAX_HEIGHT]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 静态属性。
u32PicHeight	编码图像高度。 取值范围：[MIN_HEIGHT, u32MaxPicHeight]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 动态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32BufSize	配置 buffer 大小。 取值范围：[Min, Max]，以 byte 为单位。 必须是 64 的整数倍。 Min：码流 buffer 一般模式为 u32MaxPicWidth × u32MaxPicHeight，省内存模式为 32*1024 bytes。 Max：无限制，但是会消耗更多的内存。 静态属性。
bByFrame	获取码流模式。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：按帧获取。 • HI_FALSE：按包获取。 静态属性。
bMainStream	主次码流标志。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：主码流。 • HI_FALSE：次码流。 系统只支持主码流。 静态属性。
bVIField	输入图像的帧场标志。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：场。 • HI_FALSE：帧。 目前未使用，取值不限。 建议用户取值为 HI_FALSE。 静态属性。
u32Priority	通道优先级。 目前未使用，取值不限。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_ATTR_JPEG_S

【说明】

定义 JPEG 抓拍属性结构体。

【定义】

```
typedef struct hiVENC_ATTR_JPEG_S
{
   HI_U32  u32MaxPicWidth;
   HI_U32  u32MaxPicHeight;
   HI_U32   u32BufSize;
   HI_BOOL  bByFrame;
   HI_BOOL  bVIField;
   HI_U32   u32Priority;
   HI_U32   u32PicWidth;
   HI_U32   u32PicHeight;
}VENC_ATTR_JPEG_S;
```

【成员】

成员名称	描述
u32MaxPicWidth	编码图像最大宽度。 取值范围：[MIN_WIDTH,MAX_WIDTH]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 静态属性。
u32PicWidth	编码图像宽度。 取值范围：[MIN_WIDTH, u32MaxPicWidth]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 动态属性。
u32MaxPicHeight	编码图像最大高度。 取值范围：[MIN_HEIGHT,MAX_HEIGHT]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 静态属性。
u32PicHeight	编码图像高度。 取值范围：[MIN_HEIGHT, u32MaxPicHeight]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 动态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32BufSize	配置 buffer 大小。 取值范围：[Min, Max]，以 byte 为单位。 必须是 64 的整数倍。 Min：码流 buffer 一般模式为 u32MaxPicWidth × u32MaxPicHeight，省内存模式为 32*1024 bytes。 Max：无限制，但是会消耗更多的内存 静态属性。
bByFrame	获取码流模式,帧或包。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：按帧获取。 • HI_FALSE：按包获取。 静态属性。
u32Priority	通道优先级。 目前未使用，取值不限
bVIField	输入图像的帧场标志。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：场。 • HI_FALSE：帧。 目前未使用，取值不限。 建议用户取值为 HI_FALSE。 静态属性。

【注意事项】

无。

【相关数据类型及接口】

无。

VENC_ATTR_MPEG4_S

【说明】

定义 MPEG-4 协议编码通道属性结构体。

【定义】

```
typedef struct  hiVENC_ATTR_MPEG4_S
{
    HI_U32  u32MaxPicWidth;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32  u32MaxPicHeight;
    HI_U32  u32BufSize;
    HI_BOOL bByFrame;
    HI_S32  bField;
    HI_S32  bMainStream;
    HI_U32  u32Priority;
    HI_BOOL bVIField;
    HI_U32  u32PicWidth;
    HI_U32  u32PicHeight;
}VENC_ATTR_MPEG4_S;
```

【成员】

成员名称	描述
u32MaxPicWidth	编码图像最大宽度。 取值范围：[MIN_WIDTH,MAX_WIDTH]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 静态属性。
u32PicWidth	编码图像宽度。 取值范围：[MIN_WIDTH, u32MaxPicWidth]，以像素为单位。 必须是 MIN_ALIGN 的整数倍。 动态属性。
u32MaxPicHeight	编码图像最大高度。 取值范围：[MIN_HEIGHT,MAX_HEIGHT]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 静态属性。
u32PicHeight	编码图像高度。 取值范围：[MIN_HEIGHT, u32MaxPicHeight]，以像素为单位。 必须是 MIN_ALIGN 的整数倍 动态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32BufSize	码流 buffer 大小。 取值范围：[Min, Max]，以 byte 为单位。 必须是 64 的整数倍。 推荐值：一幅最大编码图像大小。推荐值为 u32MaxPicWidth × u32MaxPicHeight ×1.5 byte。 Min：码流 buffer 一般模式为一幅最大编码图像大小的 1/2，省内存模式为 32*1024 bytes。 Max：无限制，但是会消耗更多的内存。 静态属性。
bByFrame	帧/包模式获取码流。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：按帧获取。 • HI_FALSE：按包获取。 静态属性。
bField	帧场编码模式。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：场编码。 • HI_FALSE：帧编码。 只支持帧编码。 静态属性。
bMainStream	主次码流标识。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：主码流。 • HI_FALSE：次码流。 只支持主码流。 静态属性。
u32Priority	通道优先级。 目前未使用，取值不限。
bVIField	输入图像的帧场标志。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：场。 • HI_FALSE：帧。 只支持帧图像。 静态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

无。

VENC_ATTR_S

【说明】

定义编码器属性结构体。

【定义】

```
typedef struct hiVENC_ATTR_S
{
    PAYLOAD_TYPE_E  enType;
    union
    {
        VENC_ATTR_H264_S  stAttrH264e;
        VENC_ATTR_MJPEG_S stAttrMjpeg;
        VENC_ATTR_JPEG_S  stAttrJpeg;
        VENC_ATTR_MPEG4_S stAttrMpeg4;
    };
}VENC_ATTR_S;
```

【成员】

成员名称	描述
enType	编码协议类型。
stAttrH264e/stAttrMjpeg/stAttrJpeg/stAttrMpeg4	某种协议的编码器属性。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_CHN_ATTR_S

【说明】

定义编码通道属性结构体。

【定义】

```
typedef struct hiVENC_CHN_ATTR_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    VENC_ATTR_S     stVeAttr;
    VENC_RC_ATTR_S stRcAttr;
}VENC_CHN_ATTR_S;
```

【成员】

成员名称	描述
stVeAttr	编码器属性。
stRcAttr	码率控制器属性。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_CHN_STAT_S

【说明】

定义编码通道的状态结构体。

【定义】

```
typedef struct hiVENC_CHN_STAT_S
{
    HI_BOOL bRegistered;
    HI_U32  u32LeftPics;
    HI_U32  u32LeftStreamBytes;
    HI_U32 u32LeftStreamFrames;
    HI_U32  u32CurPacks;
    HI_U32 u32LeftRecvPics;
    HI_U32 u32LeftEncPics;
}VENC_CHN_STAT_S;
```

【成员】

成员名称	描述
bRegistered	注册到通道组标志。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：注册。 • HI_FALSE：未注册。
u32LeftPics	待编码的图像数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32LeftStreamBytes	码流 buffer 剩余的 byte 数。
u32LeftStreamFrames	码流 buffer 剩余的帧数。
u32CurPacks	当前帧的码流包个数。
u32LeftRecvPics	剩余待接收的帧数，在用户设置 HI_MPI_VENC_StartRecvPicEx 后有效。
u32LeftEncPics	剩余待编码的帧数，在用户设置 HI_MPI_VENC_StartRecvPicEx 后有效。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VENC_Query

VENC_PARAM_H264_SLICE_SPLIT_S

【说明】

定义 H.264 协议编码通道 SLICE 分割结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264_SLICE_SPLIT_S
{
    HI_BOOL bSplitEnable;
    HI_U32  u32SplitMode;
    HI_U32  u32SliceSize;
} VENC_PARAM_H264_SLICE_SPLIT_S;
```

【成员】

成员名称	描述
bSplitEnable	Slice 分割是否使能。
u32SplitMode	Slice 分割模式。 • 0：按 byte 数分割。 • 1：按宏块行分割。 不支持 u32SplitMode≥2。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32SliceSize	u32SplitMode=0，表示每个 slice 的 byte 数。 最小值为：128，最大值为 min((0xFFFF), u32PicSize/2) 其中 u32PicSize=picwidth x picheight x 3/2 u32SplitMode=1，表示每个 slice 占的宏块行数。 最小值为：1，最大值为：(图像高+15)/16。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264SliceSplit
- HI_MPI_VENC_GetH264SliceSplit

VENC_PARAM_H264_INTER_PRED_S

【说明】

定义 H.264 协议编码通道帧间预测结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264_INTER_PRED_S
{
    /* search window */
    HI_U32 u32HWSize;
    HI_U32 u32VWSize;
    HI_BOOL bInter16x16PredEn;
    HI_BOOL bInter16x8PredEn;
    HI_BOOL bInter8x16PredEn;
    HI_BOOL bInter8x8PredEn;
    HI_BOOL bExtedgeEn;
} VENC_PARAM_H264_INTER_PRED_S;
```

【成员】

成员名称	描述
u32HWSize	水平搜索窗大小。
u32VWSize	垂直搜索窗大小。
bInter16x16PredEn	16x16 帧间预测使能开关，默认使能。
bInter16x8PredEn	16x8 帧间预测使能开关，默认使能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bInter8x16PredEn	8x16 帧间预测使能开关，默认使能。
bInter8x8PredEn	8x8 帧间预测使能开关，默认使能。
bExtedgeEn	当搜索遇见图像边界时，超出了图像范围，是否进行补搜的使能开关，默认使能。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264InterPred
- HI_MPI_VENC_GetH264InterPred

VENC_PARAM_H264_INTRA_PRED_S

【说明】

定义 H.264 协议编码通道帧内预测结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264_INTRA_PRED_S
{
    HI_BOOL bIntra16x16PredEn;
    HI_BOOL bIntraNxNPredEn;
    HI_U32 constrained_intra_pred_flag;
    HI_BOOL bIpcmEn;
}VENC_PARAM_H264_INTRA_PRED_S;
```

【成员】

成员名称	描述
bIntra16x16PredEn	16x16 帧内预测使能，默认使能。 取值范围：0 或 1。 0：不使能； 1：使能。
bIntraNxNPredEn	NxN 帧内预测使能，默认使能。 取值范围：0 或 1。 0：不使能； 1：使能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
constrained_intra_pred_flag	默认为 0。 取值范围：0 或 1。
bIpcmEn	IPCM 预测使能，默认值根据不同的芯片有差异。 取值范围：0 或 1。

【注意事项】

以上参数具体含义请参见 H.264 协议。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264IntraPred
- HI_MPI_VENC_GetH264IntraPred

VENC_PARAM_H264_TRANS_S

【说明】

定义 H.264 协议编码通道变换、量化结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264E_TRANS_S
{
    HI_U32 u32IntraTransMode;
    HI_U32 u32InterTransMode;
    HI_BOOL bScalingListValid;
    HI_U8  InterScalingList8x8[64];
    HI_U8  IntraScalingList8x8[64];
    HI_S32 chroma_qp_index_offset;
}VENC_PARAM_H264_TRANS_S;
```

【成员】

成员名称	描述
u32IntraTransMode	帧内预测的变换模式： • 0：支持 4x4，8x8 变换，high profile 支持。 • 1：4x4 变换，baseline, main, high profile 均支持。 • 2：8x8 变换，high profile 支持。 系统根据通道协议类型默认 TransMode 的选择。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32InterTransMode	帧间预测的变换模式： • 0：支持 4x4，8x8 变换，high profile 支持。 • 1：4x4 变换，baseline, main, high profile 均支持。 • 2：8x8 变换，high profile 支持。 系统根据通道协议类型默认 TransMode 的选择。
bScalingListValid	InterScalingList8x8、IntraScalingList8x8 是否有效标识，只在 high profile 下才有意义。 取值范围：0 或 1。 0：无效； 1：有效。
InterScalingList8x8	帧间预测 8x8 的量化表，在 high profile 下，用户可以使用自己的量化表。 取值范围：[1, 255]。
IntraScalingList8x8	帧内预测 8x8 的量化表，在 high profile 下，用户可以使用自己的量化表。 取值范围：[1, 255]。
chroma_qp_index_offset	具体含义请参见 H.264 协议。 系统默认值为 0。 取值范围：[-12, 12]。

【注意事项】

以上参数具体含义请参见 H.264 协议。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264Trans
- HI_MPI_VENC_GetH264Trans

VENC_PARAM_H264_ENTROPY_S

【说明】

定义 H.264 协议编码通道熵编码结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264E_ENTROPY_S
{
    HI_U32 u32EntropyEncModeI;
    HI_U32 u32EntropyEncModeP;
    HI_U32 cabac_stuff_en;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 Cabac_init_idc;
}VENC_PARAM_H264_ENTROPY_S;
```

【成员】

成员名称	描述
u32EntropyEncModeI	I 帧熵编码模式。 • 0：cavlc • 1：cabac ≥2 没有意义。 Baseline 不支持 cabac。
u32EntropyEncModeP	P 帧熵编码模式。 • 0：cavlc • 1：cabac ≥2 没有意义。 Baseline 不支持 cabac。
cabac_stuff_en	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：0 或 1。
Cabac_init_idc	取值范围[0, 2]，默认值 0，具体含义请参见 H.264 协议。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264Entropy
- HI_MPI_VENC_GetH264Entropy

VENC_PARAM_H264_POC_S

【说明】

定义 H.264 协议编码通道 Poc 结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264_POC_S
{
    HI_U32 pic_order_cnt_type;
}VENC_PARAM_H264_POC_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
pic_order_cnt_type	取值范围[0, 2]，默认值 2，具体含义请参见 H.264 协议

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264Poc
- HI_MPI_VENC_GetH264Poc

VENC_PARAM_H264_DBLK_S

【说明】

定义 H.264 协议编码通道 Dblk 结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264E_DBLK_S
{
    HI_U32 disable_deblocking_filter_idc;
    HI_S32 slice_alpha_c0_offset_div2;
    HI_S32 slice_beta_offset_div2;
}VENC_PARAM_H264_DBLK_S;
```

【成员】

成员名称	描述
disable_deblocking_filter_idc	取值范围[0, 2]，默认值 0，具体含义请参见 H.264 协议。
slice_alpha_c0_offset_div2	取值范围[-6, 6]，默认值 0，具体含义请参见 H.264 协议。
slice_beta_offset_div2	取值范围[-6, 6]，默认值 0，具体含义请参见 H.264 协议。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264Dblk
- HI_MPI_VENC_GetH264Dblk

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_PARAM_H264_VUI_S

【说明】

定义 H.264 协议编码通道 Vui 结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264E_VUI_S
{
    HI_S32 timing_info_present_flag;
    HI_S32 num_units_in_tick;
    HI_S32 time_scale;
    HI_S32 fixed_frame_rate_flag;
}VENC_PARAM_H264_VUI_S;
```

【成员】

成员名称	描述
timing_info_present_flag	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：0 或 1。
num_units_in_tick	具体含义请参见 H.264 协议，系统默认为 1。 取值范围：大于 0。
time_scale	具体含义请参见 H.264 协议，系统默认为 60。 取值范围：大于 0。
fixed_frame_rate_flag;	具体含义请参见 H.264 协议，系统默认为 1。 取值范围：0 或 1。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetH264Vui
- HI_MPI_VENC_GetH264Vui

VENC_PARAM_JPEG_S

【说明】

定义 JPEG 协议编码通道高级参数结构体。

【定义】

```
typedef struct hiVENC_PARAM_JPEG_S
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32Qfactor;
    HI_U8  u8YQt[64];
    HI_U8  u8CbQt[64];
    HI_U8  u8CrQt[64];
    HI_U32 u32MCUPerECS;
} VENC_PARAM_JPEG_S;
```

【成员】

成员名称	描述
u32Qfactor	具体含义请参见 RFC2435 协议，系统默认为 90。 取值范围：[1, 99]。
u8YQt	Y 量化表。 取值范围：[1, 255]。
u8CbQt	Cb 量化表。 取值范围：[1, 255]。
u8CrQt	Cr 量化表。 取值范围：[1, 255]。
u32MCUPerECS	每个 ECS 中包含多少个 MCU，系统默认为 0，表示不划分 Ecs。 u32MCUPerECS：[0, (picwidth+15)>>4 x (picheight+15)>>4 x 2]

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetJpegParam
- HI_MPI_VENC_GetJpegParam

VENC_PARAM_MJPEG_S

【说明】

定义 MJPEG 协议编码通道高级参数结构体。

【定义】

```
typedef struct hiVENC_PARAM_MJPEG_S
{
    HI_U8  u8YQt[64];
    HI_U8  u8CbQt[64];
    HI_U8  u8CrQt[64];
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32MCUPerECS;
} VENC_PARAM_MJPEG_S;
```

【成员】

成员名称	描述
u8YQt	Y 量化表。 取值范围：[1, 255]。
u8CbQt	Cb 量化表。 取值范围：[1, 255]。
u8CrQt	Cr 量化表。 取值范围：[1, 255]。
u32MCUPerECS	每个 ECS 中包含多少个 MCU，系统默认为 0，表示不划分 Ecs。 u32MCUPerECS：[0,（picwidth+15）>>4 x (picheight+15)>>4 x 2]

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetMjpegParam
- HI_MPI_VENC_GetMjpegParam

VENC_PARAM_MPEG4_S

【说明】

定义 MPEG-4 协议编码通道高级参数结构体。

【定义】

```
typedef struct hiVENC_PARAM_MPEG4_S
{
    HI_BOOL bSplitEnable;
    HI_U32  u32SplitMode;
    HI_U32  u32PacketSize;
    HI_U32 u32HWSize;
    HI_U32 u32VWSize;
} VENC_PARAM_MPEG4_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bSplitEnable	packet 分割是否使能，系统默认不使能分割。 取值范围：0 或 1。 0：不使能； 1：使能。
u32SplitMode	packet 分割模式。 • 0：按 bit 数分割。 • 1：按宏块行分割。 不支持 u32SplitMode>=2
u32PacketSize	u32SplitMode=0，表示每个 packet 的 bit 数。 最小值为：128，最大值为 min((0xFFFF), u32PicSize/2) 其中 u32PicSize=picwidth x picheight x 3/2 u32SplitMode=1，表示每个 slice 占的宏块行数。 最小值为：1，最大值为：（图像高+15）/16。
u32HWSize	水平搜索窗大小。 系统默认为 1。其中每个值对应了实际的搜索窗的范围如下： 0000：[-16, +15]； 0001：[-32, +31]； 其他：no meaning。
u32VWSize	垂直搜索窗大小。 系统默认 0。其中每个对应了实际的搜索窗的范围如下： 000：[-16, +15]； 其他：no meaning。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetMpeg4Param
- HI_MPI_VENC_GetMpeg4Param

VENC_ROI_CFG_S

【说明】

定义编码感兴趣区域信息。

【定义】

```
typedef struct hiVENC_ROI_CFG_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    HI_U32  u32Index;
    HI_BOOL bEnable;
    HI_BOOL bAbsQp;
    HI_S32  s32Qp;
    RECT_S  stRect;
}VENC_ROI_CFG_S;
```

【成员】

成员名称	描述
u32Index	ROI 区域的索引，系统支持的索引范围为[0,7]，不支持超出这个范围的索引。
bEnable	是否使能这个 ROI 区域
bAbsQp	ROI 区域 QP 模式 • HI_FALSE：相对 QP • HI_TURE：绝对 QP
s32Qp	QP 值，当 QP 模式为 HI_FALSE 时，s32Qp 为 QP 偏移，s32Qp 范围[-51,51]，当 QP 模式为 HI_TRUE 时，s32Qp 为宏块 QP 值，s32Qp 范围[0,51]。
stRect	ROI 区域。 s32X、s32Y、u32Width、u32Height 必须是 16 对齐。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetRoiCfg
- HI_MPI_VENC_GetRoiCfg

VENC_ATTR_H264_REF_MODE_E

【说明】

定义 H.264 编码的跳帧参考模式。

【定义】

```
typedef enum hiVENC_ATTR_H264_REF_MODE_E
{
    H264E_REF_MODE_1X = 1,
    H264E_REF_MODE_2X = 2,
    H264E_REF_MODE_4X = 5,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    H264E_REF_MODE_BUTT,
}VENC_ATTR_H264_REF_MODE_E;
```

【成员】

成员名称	描述
H264E_REF_MODE_1X	正常参考模式。
H264E_REF_MODE_2X	间隔 2 的跳帧参考模式。
H264E_REF_MODE_4X	间隔 4 的跳帧参考模式。

【注意事项】

无。

【相关数据类型及接口】

无。

VENC_ATTR_H264_REF_PARAM_S

【说明】

定义 H.264 编码的高级跳帧参考参数。

【定义】

```
typedef enum hiVENC_ATTR_H264_REF_PARAM_S
{
    HI_U32   u32Base;
    HI_U32   u32Enhance;
    HI_BOOL  bEnablePred;
}VENC_ATTR_H264_REF_PARAM_S;
```

【成员】

成员名称	描述
u32Base	base 层的周期。
u32Enhance	enhance 层的周期。
bEnablePred	代表 base 层的帧是否被 base 层其他帧用作参考。当为 HI_FALSE 时，base 层的所有帧都参考 IDR 帧。

【注意事项】

无。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

VENC_RC_ATTR_S

【说明】

定义编码通道码率控制器属性。

【定义】

```
typedef struct hiVENC_RC_ATTR_S
{
    VENC_RC_MODE_E      enRcMode;
    union
    {
        VENC_ATTR_H264_CBR_S    stAttrH264Cbr;
        VENC_ATTR_H264_VBR_S    stAttrH264Vbr;
        VENC_ATTR_H264_FIXQP_S  stAttrH264FixQp;
        VENC_ATTR_H264_ABR_S    stAttrH264Abr;
        VENC_ATTR_MPEG4_VBR_S   stAttrMpeg4Vbr;
        VENC_ATTR_MPEG4_FIXQP_S stAttrMpeg4FixQp;
        VENC_ATTR_MPEG4_CBR_S   stAttrMpeg4Cbr;
        VENC_ATTR_MJPEG_FIXQP_S stAttrMjpegeFixQp;
        VENC_ATTR_MJPEG_CBR_S   stAttrMjpegeCbr;
        VENC_ATTR_MJPEG_VBR_S    stAttrMjpegeVbr;
    };
    HI_VOID*      pRcAttr ;

}VENC_RC_ATTR_S;
```

【成员】

成员名称	描述
enRcMode	RC 模式。
stAttrH264Cbr	H.264 协议编码通道 Cbr 模式属性。
stAttrH264Vbr	H.264 协议编码通道 Vbr 模式属性。
stAttrH264FixQp	H.264 协议编码通道 Fixqp 模式属性。
stAttrH264Abr	H.264 协议编码通道 Abr 模式属性（暂不支持）。
stAttrMpeg4Vbr	Mpeg4 协议编码通道 Vbr 模式属性。
stAttrMpeg4FixQp	Mpeg4 协议编码通道 Fixqp 模式属性。
stAttrMpeg4Cbr	Mpeg4 协议编码通道 Cbr 模式属性。
stAttrMjpegeFixQp	Mjpeg 协议编码通道 Fixqp 模式属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stAttrMjpegeCbr	Mjpeg 协议编码通道 cbr 模式属性。
stAttrMjpegeVbr	Mjpeg 协议编码通道 vbr 模式属性

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_RC_MODE_E

【说明】

定义编码通道码率控制器模式。

【定义】

```
typedef enum hiVENC_RC_MODE_E
{
    VENC_RC_MODE_H264CBR = 1,
    VENC_RC_MODE_H264VBR,
    VENC_RC_MODE_H264ABR,
    VENC_RC_MODE_H264FIXQP,
    VENC_RC_MODE_MJPEGCBR,
    VENC_RC_MODE_MJPEGVBR,
    VENC_RC_MODE_MJPEGABR,
    VENC_RC_MODE_MJPEGFIXQP,
    VENC_RC_MODE_MPEG4CBR,
    VENC_RC_MODE_MPEG4VBR,
    VENC_RC_MODE_MPEG4ABR,
    VENC_RC_MODE_MPEG4FIXQP,
    VENC_RC_MODE_H264CBRv2,
    VENC_RC_MODE_H264VBRv2,
    VENC_RC_MODE_MJPEGCBRv2,
    VENC_RC_MODE_MJPEGVBRv2,
    VENC_RC_MODE_BUTT,
}VENC_RC_MODE_E;
```

【成员】

成员名称	描述
VENC_RC_MODE_H264CBR	H.264 CBR 模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VENC_RC_MODE_H264VBR	H.264 VBR 模式。
VENC_RC_MODE_H264ABR	H.264 ABR 模式（暂时不提供）。
VENC_RC_MODE_H264FIXQP	H.264 Fixqp 模式。
VENC_RC_MODE_MJPEGCBR	MJPEG CBR 模式。
VENC_RC_MODE_MJPEGVBR	MJPEG VBR 模式。
VENC_RC_MODE_MJPEGABR	MJPEG ABR 模式（暂时不提供）。
VENC_RC_MODE_MJPEGFIXQP	MJPEG Fixqp 模式。
VENC_RC_MODE_MPEG4CBR	MPEG4 CBR 模式（暂时不提供）。
VENC_RC_MODE_MPEG4VBR	MPEG4 VBR 模式（暂时不提供）。
VENC_RC_MODE_MPEG4ABR	MPEG4 ABR 模式（暂时不提供）。
VENC_RC_MODE_MPEG4FIXQP	MPEG4 Fixqp 模式。
VENC_RC_MODE_H264CBRv2	新增码率控制算法 H264 CBR 模式。
VENC_RC_MODE_H264VBRv2	新增码率控制算法 H264 VBR 模式。
VENC_RC_MODE_MJPEGCBRv2	新增码率控制算法 MJPEG CBR 模式。
VENC_RC_MODE_MJPEGVBRv2	新增码率控制算法 MJPEG VBR 模式

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_H264_CBR_S

【说明】

定义 H.264 编码通道 CBR 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_H264_CBR_S
{
    HI_U32      u32Gop;
    HI_U32      u32StatTime;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32BitRate;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32      u32FluctuateLevel;
} VENC_ATTR_H264_CBR_S;
```

【成员】

成员名称	描述
u32Gop	H.264 gop 值。 取值范围：[1, 65536]。
u32StatTime	CBR 码率统计时间，以秒为单位。 取值范围：[1, 60]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32BitRate	平均 bitrate，以 kbps 为单位。 取值范围：[2, 40960]。
u32FluctuateLevel	最大码率相对平均码率的波动等级。 取值范围：[0, 5]。 推荐使用波动等级 0。

【注意事项】

- ViFrmRate 应该设置为输入编码器的实际帧率，RC 需要根据 ViFrmRate 统计实际的帧率以及进行码率控制。假设 VI 的输出帧率是 30，如果 GROUP 不进行帧率控制，ViFrmRate 应该设置为 30，如果 GROUP 进行帧率控制，GROUP 帧率控制中的源帧率和 ViFrmRate 都应该设置为 30。如果 GROUP 和 RC 都进行帧率控制，那么最终的帧率将以两者的较小值为准。比如 GROUP 设置为源帧率 30 帧，目标帧率 6 帧，而 RC 设置为源帧率 30，目标帧率为 10 帧，那么最终的目标帧率为 6 帧。设置目标帧率 TargetFrmRate 时，目标帧率类型定义为分数类型 HI_FR32 类型。HI_FR32 实际为 HI_U32 型，高 16 位用于表示分母，低 16 位表示分子。如果用户设置目标帧率为整数，设置分母高 16 位为 0 即可表示整数。例如设置 ViFrmRate 等于 25，TargetFrmRate 等于 12，则表示将从 25 帧输入图像中取出 12 帧进行编码，其余 13 帧将丢掉。如果设置 ViFrmRate 等于 25，TargetFrmRate 等于 15/2，则表示要求编码器 2 秒钟编码 15 帧。
- TargetFrmRate 的设置方式如下：如果要设置为整数帧率 25，则可采用以下方式 TargetFrmRate = 25；如果要设置为分数帧率 15/2，则采用以下方式：TargetFrmRate = 15+（2<<16）。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_ATTR_H264_VBR_S

【说明】

定义 H.264 编码通道 VBR 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_H264_VBR_S
{
    HI_U32      u32Gop;
    HI_U32      u32StatTime;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32MaxBitRate;
    HI_U32      u32MaxQp;
    HI_U32      u32MinQp;
}VENC_ATTR_H264_VBR_S;
```

【成员】

成员名称	描述
u32Gop	H.264 gop 值。 取值范围：[1, 65536]。
u32StatTime	VBR 码率统计时间，以秒为单位。 取值范围：[1, 60]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32MaxBitRate	编码器输出最大码率，以 kbps 为单位。 取值范围：[2, 40960]。
u32MaxQp	编码器支持图像最大 QP。 取值范围：(u32MinQp, 51]。
u32MinQp	编码器支持图像最小 QP。 取值范围：[0, 51]。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VENC_CreateChn

VENC_ATTR_H264_FIXQP_S

【说明】

定义 H.264 编码通道 Fixqp 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_H264_FIXQP_S
{
    HI_U32      u32Gop;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate ;
    HI_U32      u32IQp;
    HI_U32      u32PQp;
} VENC_ATTR_H264_FIXQP_S;
```

【成员】

成员名称	描述
u32Gop	H.264 gop 值。 取值范围：[1, 65536]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32IQp	I 帧所有宏块 Qp 值。 取值范围：[0, 51]。
u32PQp	P 帧所有宏块 Qp 值。 取值范围：[0, 51]。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_H264_ABR_S

【说明】

定义 H.264 编码通道 ABR 属性结构。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct hiVENC_ATTR_H264_ABR_S
{
    HI_U32   u32Gop;               /*the interval of ISLICE. */
    HI_U32   u32StatTime;          /* the rate statistic time, the unit is
senconds(s) */
    HI_U32   u32ViFrmRate;         /* the input frame rate of the venc
chnnel */
    HI_FR32  fr32TargetFrmRate ;    /* the target frame rate of the venc
chnnel */
    HI_U32   u32AvgBitRate;        /* average bitrate */
    HI_U32   u32MaxBitRate;        /* the max bitrate */
}VENC_ATTR_H264_ABR_S;
```

【成员】

成员名称	描述
u32Gop	H.264 gop 值。 取值范围：[1, 65536]。
u32StatTime	ABR 码率统计时间，以秒为单位。 取值范围：[1, 60]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32AvgBitRate	平均码率。 取值范围[2000, 40960000]
u32MaxBitRate	最大码率。 取值范围[2000, 40960000]

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_MPEG4_CBR_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义 MPEG-4 编码通道 CBR 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_MPEG4_CBR_S
{
    HI_U32      u32Gop;
    HI_U32      u32StatTime;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32BitRate;
    HI_U32      u32FluctuateLevel;
}VENC_ATTR_MPEG4_CBR_S;
```

【成员】

成员名称	描述
u32Gop	MPEG4 gop 值。 取值范围：[1, 65536]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32StatTime	CBR 码率统计时间，以秒为单位。 取值范围：[1, 60]。
u32BitRate	平均 bitrate，以 kbps 为单位。 取值范围：[2, 40960]。
u32FluctuateLevel	最大码率相对平均码率的波动等级。 取值范围：[0, 5]。 推荐使用波动等级 0。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_MPEG4_VBR_S

【说明】

定义 MPEG-4 编码通道 VBR 属性结构。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct hiVENC_ATTR_MPEG4_VBR_S
{
    HI_U32      u32Gop;
    HI_U32      u32StatTime;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32MaxBitRate;
    HI_U32      u32MaxQp;
    HI_U32      u32MinQp;
}VENC_ATTR_MPEG4_VBR_S;
```

【成员】

成员名称	描述
u32Gop	Mpeg-4 gop 值。 取值范围：[1, 65536]。
u32StatTime	VBR 码率统计时间，以秒为单位。 取值范围：[1, 60]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32MaxBitRate	编码器输出最大码率，以 kbps 为单位。 取值范围：[2, 40960]。
u32MaxQp	编码器支持图像最大 QP。 取值范围：(u32MinQp, 31]。
u32MinQp	编码器支持图像最小 QP。 取值范围：[1, 31]。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_MPEG4_FIXQP_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义 MPEG-4 编码通道 Fixqp 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_MPEG4_FIXQP_S
{
    HI_U32      u32Gop;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32IQp;
    HI_U32      u32PQp;

}VENC_ATTR_MPEG4_FIXQP_S;
```

【成员】

成员名称	描述
u32Gop	MPEG4 gop 值。 取值范围：[1, 65536]。
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32IQp	I 帧所有宏块 Qp 值。 取值范围：[1, 31]。
u32PQp	P 帧所有宏块 Qp 值。 取值范围：[1, 31]。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_MJPEG_FIXQP_S

【说明】

定义 MJPEG 编码通道 Fixqp 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_MJPEG_FIXQP_S
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32Qfactor;
}VENC_ATTR_MJPEG_FIXQP_S;
```

【成员】

成员名称	描述
u32ViFrmRate	VI 输入帧率，以 fps 为单位。 取值范围：[1, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32Qfactor	MJPEG 编码的 Qfactor。 取值范围：[1, 99]。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_MJPEG_CBR_S

【说明】

定义 MJPEG 编码通道 CBR 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_MJPEG_CBR_S
{
    HI_U32      u32StatTime;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32BitRate;
    HI_U32      u32FluctuateLevel;
} VENC_ATTR_MJPEG_CBR_S;
```

【成员】

成员名称	描述
u32StatTime	CBR 码率统计时间，以秒为单位。 取值范围：[1, 16]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32ViFrmRate	编码器输入帧率，以 fps 为单位。 取值范围：（0, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32BitRate	平均 bitrate，以 kbps 为单位。 取值范围：[2, 40960]。
u32FluctuateLevel	最大码率相对平均码率的波动等级。 取值范围：[0, 5]。 推荐使用波动等级为 0。

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_ATTR_MJPEG_VBR_S

【说明】

定义 MJPEG 编码通道 VBR 属性结构。

【定义】

```
typedef struct hiVENC_ATTR_MJPEG_VBR_S
{
    HI_U32      u32StatTime;
    HI_U32      u32ViFrmRate;
    HI_FR32     fr32TargetFrmRate;
    HI_U32      u32MaxBitRate;
    HI_U32     u32MaxQfactor;
    HI_U32     u32MinQfactor;
}VENC_ATTR_MJPEG_VBR_S;
```

【成员】

成员名称	描述
u32StatTime	VBR 码率统计时间，以秒为单位。 取值范围：[1, 60]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32ViFrmRate	编码器输入帧率，以 fps 为单位。 取值范围：（0, 60]。
fr32TargetFrmRate	编码器输出帧率，以 fps 为单位。 取值范围：(0, u32ViFrmRate]。
u32MaxBitRate	最大 bitrate，以 kbps 为单位。 取值范围：[2, 40960]。
u32MaxQfactor	最大量化因子，取值范围：[1, 99]。
u32MinQfactor	最小量化因子，取值范围：[1, u32MaxQfactor)

【注意事项】

请参见 VENC_ATTR_H264_CBR_S 关于 u32ViFrmRate 和 fr32TargetFrmRate 的说明。

【相关数据类型及接口】

HI_MPI_VENC_CreateChn

VENC_SUPERFRM_MODE_EN

【说明】

定义码率控制中超大帧处理模式。

【定义】

```
typedef enum hiRC_SUPERFRM_MODE_EN
{
    SUPERFRM_NONE,
    SUPERFRM_DISCARD,
    SUPERFRM_REENCODE,
    SUPERFRM_BUTT
}VENC_SUPERFRM_MODE_EN;
```

【成员】

成员名称	描述
SUPERFRM_NONE	无特殊策略。
SUPERFRM_DISCARD	丢弃超大帧。
SUPERFRM_REENCODE	重编超大帧。

【注意事项】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关数据类型及接口】

HI_MPI_VENC_GetRcPara

VENC_PARAM_H264_CBR_S

【说明】

定义 H264 协议编码通道 CBR 码率控制模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_H264_CBR_S
{
    HI_U32  u32MinIprop;
    HI_U32  u32MaxIprop;
    HI_U32  u32MaxQp;
    HI_U32   u32MaxStartQp;
    HI_U32  u32MinQp;
    HI_U32  u32MaxPPDeltaQp;
    HI_U32  u32MaxIPDeltaQp;
    HI_BOOL bLostFrmOpen;
    HI_U32  u32LostFrmBpsThr;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperIFrmBitsThr;
    HI_U32  u32SuperPFrmBitsThr;
    HI_S32 s32IPQPDelta;
    HI_U32  u32RQRatio[8];
    HI_U32  u32MinIQp;
}VENC_PARAM_H264_CBR_S;
```

【成员】

成员名称	描述
u32MinIprop	最小 IP 帧比例。 取值范围：(0, 100]。
u32MaxIprop	最大 IP 帧比例。 取值范围：[u32MinIprop, 100]。
u32MaxQp	帧最大 QP，用于钳位质量。 取值范围：(u32MinQp, 51]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32MaxStartQp	帧级码率控制输出的最大 QP。 取值范围：[u32MinQp, u32MaxQp]。 宏块级码率控制可能会使某些宏块的 QP 大于或小于 u32MaxStartQp，但都会被钳位到[u32MinQp, u32MaxQp]。
u32MinQp	帧最小 QP，用于钳位码率波动。 取值范围：[0, 51]。
u32MaxPPDeltaQp	P 帧间的最大 QP。 取值范围：[0, 10]。
u32MaxIPDeltaQp	I/P 帧之间的最大 QP。 取值范围：[0, 10]。
bLostFrmOpen	瞬时码率超出，是否允许丢帧。 取值范围： HI_TRUE：允许丢帧。 HI_FALSE：不允许丢帧。
u32LostFrmBpsThr	瞬时码率阈值，超过此阈值，允许丢帧，单位 bit。 取值范围：[64x1024, 80x1024x1024]。
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。
u32SuperIFrmBitsThr	超大 I 帧，单位 bit。 取值范围：大于等于 0。
u32SuperPFrmBitsThr	超大 P 帧，单位 bit。 取值范围：大于等于 0。
s32IPQPDelta	IP QP 变化值。 取值范围：[-10, 10]。
u32RQRatio	R-Q 的侧重比例。 取值范围：[0, 100]。
u32MinIQp	I 帧的最小 QP。用于控制 I 帧的最大 bits 数。 取值范围：[0, 51]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_PARAM_H264_VBR_S

【说明】

定义 H264 协议编码通道 VBR 码率控制模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_H264_VBR_S
{
    HI_S32  s32DeltaQP;
    HI_S32  s32ChangePos;
   HI_U32  u32MinIprop;
   HI_U32  u32MaxIprop;
   HI_BOOL bLostFrmOpen;
   HI_U32  u32LostFrmBpsThr;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperIFrmBitsThr;
    HI_U32  u32SuperPFrmBitsThr;
    HI_U32  u32MinIQP;
}VENC_PARAM_H264_VBR_S;
```

【成员】

成员名称	描述
s32DeltaQP	VBR 调整 Qp 时的帧与帧之间 Qp 变化的最大值。 取值范围：[0, 10]。
s32ChangePos	VBR 开始调整 Qp 时的码率相对于最大码率的比例。 取值范围：[50, 100]。
u32MinIprop	最小 IP 帧码率的比值，最小值为 1。
u32MaxIprop	最大 IP 帧码率的比值，最大值为 100。
bLostFrmOpen	是否丢帧开关。
u32LostFrmBpsThr	丢帧的码率阈值，丢帧开关打开时有效。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。
u32SuperIFrmBitsThr	超大 I 帧，以 bit 为单位。 取值范围：大于等于 0。
u32SuperPFrmBitsThr	超大 P 帧，以 bit 为单位。 取值范围：大于等于 0。
u32MinIQP	I 帧的最小 QP。用于控制 I 帧的最大 bits 数。 取值范围：[0, 51]。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_PARAM_MJPEG_CBR_S

【说明】

定义 MJPEG 协议编码通道 CBR 码率控制模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_MJPEG_CBR_S
{
    HI_U32  u32MaxQfactor;;
    HI_U32  u32MinQfactor;
    HI_BOOL bLostFrmOpen;
    HI_U32  u32LostFrmBpsThr;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperFrmBitsThr;
    HI_U32  u32RQRatio[8];
}VENC_PARAM_MJPEG_CBR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32MaxQfactor	帧最大 Qfactor，用于钳位质量。 取值范围：[u32MinQfactor, 99]。
u32MinQfactor	帧最小 Qfactor，用于钳位质量。 取值范围：[1, 99]。
bLostFrmOpen	是否丢帧开关。
u32LostFrmBpsThr	丢帧的码率阈值，丢帧开关打开时有效。
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。
u32SuperFrmBitsThr	超大帧阈值，单位为 bit。 取值范围：大于等于 0。
u32RQRatio	R-Q 的侧重比例。 取值范围：[0, 100]。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_PARAM_MJPEG_VBR_S

【说明】

定义 MJPEG 协议编码通道 VBR 码率控制模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_MJPEG_VBR_S
{
    HI_S32  s32DeltaQfactor;
    HI_S32  s32ChangePos;
    HI_BOOL bLostFrmOpen;
    HI_U32  u32LostFrmBpsThr;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperFrmBitsThr;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}VENC_PARAM_MJPEG_VBR_S;
```

【成员】

成员名称	描述
s32DeltaQfactor	VBR 调整 Qfactor 时的帧与帧之间 Qfactor 变化的最大值。 取值范围：[0, 10]。
s32ChangePos	VBR 开始调整 Qfactor 时的码率相对于最大码率的比例。 取值范围：[50, 100]。
bLostFrmOpen	是否丢帧开关。
u32LostFrmBpsThr	丢帧的码率阈值，丢帧开关打开时有效。
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。
u32SuperFrmBitsThr	超大帧阈值，以 bit 为单位。 取值范围：大于等于 0。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_PARAM_MPEG4_CBR_S

【说明】

定义 MPEG-4 协议编码通道 CBR 码率控制模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_MPEG4_CBR_S
{

    HI_U32  u32MaxQp;
    HI_U32  u32MaxStartQp;
    HI_U32  u32MinQp;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32  u32MaxPPDeltaQp;
    HI_U32  u32MaxIPDeltaQp;
    HI_U32  u32MinIprop;
    HI_U32  u32MaxIprop;
    HI_BOOL bLostFrmOpen;
    HI_U32  u32LostFrmBpsThr;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperIFrmBitsThr;
    HI_U32  u32SuperPFrmBitsThr;
    HI_S32 s32IPQPDelta;
    HI_U32  u32RQRatio[8];
}VENC_PARAM_MPEG4_CBR_S;
```

【成员】

成员名称	描述
u32MaxQp	帧最大 QP，用于钳位质量。 取值范围：(u32MinQp, 31]。
u32MaxStartQp	帧级码率控制输出的最大 QP。 取值范围：[u32MinQp, u32MaxQp]。 宏块级码率控制可能会使某些宏块的 QP 大于或小于 u32MaxStartQp，但都会被钳位到[u32MinQp, u32MaxQp]。
u32MinQp	帧最小 QP，用于钳位码率波动。 取值范围：[1, 31]。
u32MaxPPDeltaQp	P 帧间的最大 QP。 取值范围：[0, 10]。
u32MaxIPDeltaQp	I/P 帧之间的最大 QP。 取值范围：[0, 10]。
u32MinIprop	最小 IP 帧码率的比值，最小值为 1。
u32MaxIprop	最大 IP 帧码率的比值，最大值为 100。
bLostFrmOpen	是否丢帧开关。
u32LostFrmBpsThr	丢帧的码率阈值，丢帧开关打开时有效。
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32SuperIFrmBitsThr	超大 I 帧，以 bit 为单位。 取值范围：大于等于 0。
u32SuperPFrmBitsThr	超大 P 帧，以 bit 为单位。 取值范围：大于等于 0。
s32IPQPDelta	IP QP 变化值。 取值范围：[-10, 10]。
u32RQRatio	R-Q 的侧重比例。 取值范围：[0, 100]。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_PARAM_MPEG4_VBR_S

【说明】

定义 MPEG-4 协议编码通道 VBR 码率控制模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_MPEG4_VBR_S
{
    HI_S32  s32DeltaQP;
    HI_S32  s32ChangePos;
    HI_U32  u32MinIprop;
    HI_U32  u32MaxIprop;
    HI_BOOL bLostFrmOpen;
    HI_U32  u32LostFrmBpsTh;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperIFrmBitsThr;
    HI_U32  u32SuperPFrmBitsThr;
}VENC_PARAM_MPEG4_VBR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
s32DeltaQP	VBR 调整 Qp 时的帧与帧之间 Qp 变化的最大值。 取值范围：[0, 10]。
s32ChangePos	VBR 开始调整 Qp 时的码率相对于最大码率的比例。 取值范围：[50, 100]。
u32MinIprop	最小 IP 帧比例。 取值范围：(0, 100]。
u32MaxIprop	最大 IP 帧比例。 取值范围：(u32MinIprop, 100]。
bLostFrmOpen	是否丢帧开关。
u32LostFrmBpsThr	丢帧的码率阈值，丢帧开关打开时有效。
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。
u32SuperIFrmBitsThr	超大 I 帧，以 bit 为单位。 取值范围：大于等于 0。
u32SuperPFrmBitsThr	超大 P 帧，以 bit 为单位。 取值范围：大于等于 0。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_PARAM_H264_CBRV2_S

【说明】

新码率控制算法 H264 CBR 模式高级参数配置。

【定义】

```
typedef struct hiVENC_PARAM_H264_CBRV2_S
{
    HI_U32  u32MinIprop;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32  u32MaxIprop;
    HI_U32  u32MaxQp;
    HI_U32  u32MaxStartQp;
    HI_U32  u32MinQp;
    HI_BOOL bLostFrmOpen;
    HI_U32  u32LostFrmBpsThr;
    VENC_SUPERFRM_MODE_EN  enSuperFrmMode;
    HI_U32  u32SuperIFrmBitsThr;
    HI_U32  u32SuperPFrmBitsThr;
    HI_S32  s32IPQPDelta;
    HI_S32  s32QualityLevel;
    HI_S32  s32MaxReEncodeTimes;
    HI_U32  u32MinIQp;
}VENC_PARAM_H264_CBRV2_S;
```

【成员】

成员名称	描述
u32MinIprop	最小 IP 帧比例。 取值范围：(0, 100]。
u32MaxIprop	最大 IP 帧比例。 取值范围：(u32MinIprop, 100]。
u32MaxQp	帧最大 QP，用于钳位质量。 取值范围：(u32MinQp, 51]。
u32MaxStartQp	帧级码率控制输出的最大 QP。 取值范围：[u32MinQp, u32MaxQp]。 宏块级码率控制可能会使某些宏块的 QP 大于或小于 u32MaxStartQp，但都会被钳位到[u32MinQp, u32MaxQp]。
u32MinQp	帧最小 QP，用于钳位码率波动。 取值范围：[0, 51]。
bLostFrmOpen	瞬时码率超出，是否允许丢帧。 取值范围： HI_TRUE：允许丢帧。 HI_FALSE：不允许丢帧。
u32LostFrmBpsThr	瞬时码率阈值，超过此阈值，允许丢帧，单位 bit。 取值范围：[64x1024, 80x1024x1024]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enSuperFrmMode	超大帧模式。 取值范围： SUPERFRM_NONE：无特殊策略。 SUPERFRM_DISCARD：丢帧模式。 SUPERFRM_REENCODE：重编码模式。
u32SuperIFrmBitsThr	超大 I 帧，单位 bit。 取值范围：大于等于 0。
u32SuperPFrmBitsThr	超大 P 帧，单位 bit。 取值范围：大于等于 0。
s32IPQPDelta	IP QP 变化值。 取值范围：[-10, 10]。
s32QualityLevel	质量等级。数值越小表示质量越好。 取值范围：[1, 5]。
s32MaxReEncodeTimes	每帧重编码次数。0 表示不进行重编码。 取值范围：[0, 3]。
u32MinIQp	I 帧的最小 QP。用于控制 I 帧的最大 bits 数。 取值范围：[0, 51]。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_GetRcPara
- HI_MPI_VENC_SetRcPara

VENC_RC_PARAM_S

【说明】

定义编码通道的码率控制高级参数。

【定义】

```
typedef struct hiVENC_RC_PARAM_S
{
    HI_U32 u32ThrdI[12];
    HI_U32 u32ThrdP[12];
    HI_U32 u32QpDelta;
    union
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    {
        VENC_PARAM_H264_CBR_S stParamH264Cbr;
        VENC_PARAM_H264_VBR_S stParamH264VBR;
        VENC_PARAM_H264_ABR_S stParamH264ABR;
        VENC_PARAM_MJPEG_CBR_S stParamMjpegCbr;
        VENC_PARAM_MJPEG_VBR_S stParamMjpegVbr;
        VENC_PARAM_MPEG4_CBR_S stParamMpeg4Cbr;
        VENC_PARAM_MPEG4_VBR_S stParamMpeg4Vbr;
        VENC_PARAM_H264_CBRV2_S   stParamH264CbrV2;
    };
    HI_VOID* pRcParam;
}VENC_RC_PARAM_S;
```

【成员】

成员名称	描述
u32ThrdI	I 帧宏块级码率控制的 mad 门限。 取值范围：[0, 255]。
u32ThrdP	P 帧宏块级码率控制的 mad 门限。 取值范围：[0, 255]。
u32QpDelta	行级码率控制。 取值范围：[0, 10]。
stParamH264Cbr	H.264 通道 CBR（Constant Bit Rate）码率控制模式高级参数。
stParamH264Vbr	H.264 通道 VBR（Variable Bit Rate）码率控制模式高级参数。 新码率控制算法 H264 VBR 高级参数。
stParamH264Abr	H.264 通道 ABR（Average Bit Rate）码率控制模式高级参数。
stParamMjpegCbr	MJPEG 通道 CBR（Constant Bit Rate）码率控制模式高级参数。
stParamMjpegVbr	MJPEG 通道 VBR（Variable Bit Rate）码率控制模式高级参数。
stParamMpeg4Cbr	MPEG_4 通道 CBR（Constant Bit Rate）码率控制模式高级参数。
stParamMpeg4Vbr	MPEG_4 通道 VBR（Variable Bit Rate）码率控制模式高级参数。
stParamH264CbrV2	新码率控制算法 H264 CBR 高级参数。
pRcParam	保留参数，目前没有使用到。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetRcPara
- HI_MPI_VENC_GetRcPara

GROUP_CROP_CFG_S

【说明】

定义通道组截取（Crop）参数。

【定义】

```
typedef struct hiGROUP_CROP_CFG_S
{
    HI_BOOL bEnable;    /* Crop region enable */
    RECT_S  stRect;     /* Crop region, note: s32X must be multi of 16 */
}GROUP_CROP_CFG_S;
```

【成员】

成员名称	描述
bEnable	是否进行裁剪。 取值范围：[HI_FALSE, HI_TRUE] HI_TRUE：使能裁剪。 HI_FALSE：不使能裁剪。
stRect	裁剪的区域。 stRect.s32X：必须 16 像素对齐。 stRect.s32Y：无约束。 stRect.u32Width，s32Rect.u32Height，满足对应编码通道的宽高约束。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetGrpCrop
- HI_MPI_VENC_GetGrpCrop

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

GROUP_FRAME_RATE_S

【说明】

定义通道组帧率控制参数。

【定义】

```
typedef struct hiGROUP_FRAME_RATE_S
{
    HI_S32  s32ViFrmRate;
    HI_S32  s32VpssFrmRate;
} GROUP_FRAME_RATE_S;
```

【成员】

成员名称	描述
s32ViFrmRate	进入通道组的帧率。
s32VpssFrmRate	通道组输出的帧率。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetGrpFrmRate
- HI_MPI_VENC_GetGrpFrmRate

GROUP_COLOR2GREY_S

【说明】

开启或关闭一个 GROUP 的彩转灰功能。

【定义】

```
typedef struct hiGROUP_COLOR2GREY_S
{
    HI_BOOL  bColor2Grey;       /* Whether to enable Color2Grey.*/
}GROUP_COLOR2GREY_S;
```

【成员】

成员名称	描述
bColor2Grey	开启或关闭一个 GROUP 的彩转灰功能。

【注意事项】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetGrpColor2Grey
- HI_MPI_VENC_GetGrpColor2Grey

GROUP_COLOR2GREY_CONF_S

【说明】

彩转灰功能的全局配置参数。

【定义】

```
typedef struct hiGROUP_COLOR2GREY_CONF_S
{
    HI_BOOL     bEnable;
    HI_U32      u32MaxWidth;
    HI_U32      u32MaxHeight;
}GROUP_COLOR2GREY_CONF_S;
```

【成员】

成员名称	描述
bEnable	所有 GROUP 是否开启彩转灰功能的全局开关。
u32MaxWidth	要开启彩转灰功能的 GROUP 的最大宽度。一个 GROUP 的宽度就是注册到其中的通道的宽度
u32MaxHeight	要开启彩转灰功能的 GROUP 的最大高度。一个 GROUP 的高度就是注册到其中的通道的高度

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetColor2GreyConf
- HI_MPI_VENC_GetColor2GreyConf

VENC_JPEG_SNAP_MODE_E

【说明】

JPEG 编码通道的抓拍模式。

【定义】

```
typedef enum hiVENC_JPEG_SNAP_MODE_E
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    JPEG_SNAP_ALL    = 0,
    JPEG_SNAP_FLASH  = 1,
    JPEG_SNAP_BUTT,
}VENC_JPEG_SNAP_MODE_E;
```

【成员】

成员名称	描述
JPEG_SNAP_ALL	全部抓拍模式，JPEG 通道的默认抓拍模式。
JPEG_SNAP_FLASH	闪光灯抓拍模式，JPEG 通道配合前端闪光灯时使用的抓拍模式。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetJpegSnapMode
- HI_MPI_VENC_GetJpegSnapMode

VENC_RECV_PIC_PARAM_S

【说明】

接收图像参数。

【定义】

```
typedef enum hiVENC_RECV_PIC_PARAM_S
{
    HI_S32 s32RecvPicNum;
} VENC_RECV_PIC_PARAM_S;
```

【成员】

成员名称	描述
s32RecvPicNum	编码通道连续接收并编码的帧数。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_VENC_SetJpegSnapMode

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_RC_PRIORITY_E

【说明】

码率控制优先级枚举。

【定义】

```
typedef enum hiVENC_RC_PRIORITY_E
{
    VENC_RC_PRIORITY_BITRATE_FIRST = 1,
    VENC_RC_PRIORITY_FRAMEBITS_FIRST,
    VENC_RC_PRIORITY_BUTT,
} VENC_RC_PRIORITY_E;
```

【成员】

成员名称	描述
VENC_RC_PRIORITY_BITRATE_FIRST	目标码率优先。
VENC_RC_PRIORITY_FRAMEBITS_FIRST	超大帧阈值优先。

【注意事项】

无。

【相关数据类型及接口】

无

VENC_RC_LOSTFRM_MODE_E

【说明】

丢帧模式枚举。

【定义】

```
typedef enum hiVENC_RC_LOSTFRM_MODE_E
{
    LOSTFRM_NORMAL,                          /*normal mode*/
    LOSTFRM_PSKIP,                           /*pskip*/
    LOSTFRM_BUTT,
}VENC_RC_LOSTFRM_MODE_E;
```

【成员】

成员名称	描述
LOSTFRM_NORMAL	正常丢帧。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
LOSTFRM_PSKIP	编码为 Pskip 帧。

【注意事项】

无。

【相关数据类型及接口】

VENC_RC_PARAM_LOSTFRM_S

VENC_RC_PARAM_LOSTFRM_S

【说明】

丢帧策略结构体。

【定义】

```
typedef struct hiVENC_RC_PARAM_LOSTFRM_S
{
    HI_BOOL bLostFrmOpen;             /* Indicates whether to discard
frames to ensure stable bit rate when the instant bit rate is exceeded */
    HI_U32  u32LostFrmBpsThr;         /* the instant bit rate threshold */
    VENC_LOSTFRM_MODE_E enLostFrmMode;  /* lost frame strategy */
    HI_U32  u32LostFrmGaps; }VENC_RC_PARAM_LOSTFRM_S;
```

【成员】

成员名称	描述
bLostFrmOpen	瞬时码率超出时是否允许丢帧
u32LostFrmBpsThr	瞬时码率阈值。 取值范围：[64x1024, 80x1024x1024]
enLostFrmMode	丢帧策略枚举。
u32EncFrmGaps	连续丢帧的最大个数。 取值范围：[0, 65535]

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_VENC_SetLostFrameStrategy
- HI_MPI_VENC_GetLostFrameStrategy

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VENC_PARAM_H264_VIDEO_SIGNAL_S

【说明】

定义 H.264 协议编码通道 VideoSignali 结构体。

【定义】

```
typedef struct hiVENC_PARAM_H264E_VIDEO_SIGNAL_S
{
   HI_U8  video_signal_type_present_flag ;
   HI_U8  video_format ;
   HI_U8  video_full_range_flag;
   HI_U8  colour_description_present_flag ;
   HI_U8  colour_primaries ;
   HI_U8  transfer_characteristics;
   HI_U8  matrix_coefficients;
}VENC_PARAM_H264_VIDEO_SIGNAL_S;
```

【成员】

成员名称	描述
video_signal_type_present_flag	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[0, 1]。
video_format	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[0, 5]。
video_full_range_flag	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[0, 1]。
colour_description_present_flag	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[0, 1]。
colour_primaries	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[0, 9]。
transfer_characteristics	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[1, 15]。
matrix_coefficients	具体含义请参见 H.264 协议，系统默认为 0。 取值范围：[0, 10]。

【注意事项】

无。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VENC_SetH264VideoSignal
- HI_MPI_VENC_GetH264VideoSignal

6.5 错误码

视频编码 API 错误码如表 6-15 所示。

表6-15 视频编码 API 错误码

错误代码	宏定义	描述
0xA0078001	HI_ERR_VENC_INVALID_DEVID	通道组超出合法范围
0xA0078002	HI_ERR_VENC_INVALID_CHNID	通道 ID 超出合法范围
0xA0078003	HI_ERR_VENC_ILLEGAL_PARAM	参数超出合法范围
0xA0078004	HI_ERR_VENC_EXIST	试图申请或者创建已经存在的设备、通道或者资源
0xA0078005	HI_ERR_VENC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源
0xA0078006	HI_ERR_VENC_NULL_PTR	函数参数中有空指针
0xA0078007	HI_ERR_VENC_NOT_CONFIG	使用前未配置
0xA0078008	HI_ERR_VENC_NOT_SUPPORT	不支持的参数或者功能
0xA0078009	HI_ERR_VENC_NOT_PERM	该操作不允许，如试图修改静态配置参数
0xA007800C	HI_ERR_VENC_NOMEM	分配内存失败，如系统内存不足
0xA007800D	HI_ERR_VENC_NOBUF	分配缓存失败，如申请的数据缓冲区太大
0xA007800E	HI_ERR_VENC_BUF_EMPTY	缓冲区中无数据
0xA007800F	HI_ERR_VENC_BUF_FULL	缓冲区中数据满
0xA0078010	HI_ERR_VENC_SYS_NOTREADY	系统没有初始化或没有加载相应模块

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

7 视频侦测分析.....7-1
7.1 概述.....7-1
7.2 功能描述.....7-1
7.2.1 基本概念.....7-1
7.2.2 创建通道.....7-3
7.2.3 输入源.....7-5
7.2.4 处理 VDA 结果.....7-5
7.3 API 参考.....7-6
7.4 数据类型.....7-21
7.5 错误码.....7-38

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 7-1 宏块 SAD 值结果内存分布示意图.....7-6

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 7-1 视频侦测分析 API 错误码 7-39

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

7 视频侦测分析

7.1 概述

视频侦测分析（VDA）通过检测视频的亮度变化，得出视频侦测分析结果。VDA 包含运动侦测（MD）和遮挡检测（OD）两种工作模式，视频侦测分析结果也由当前工作模式区分为运动侦测结果和遮挡检测结果：

- 运动侦测（MD）侦测 VDA 通道接收的视频的运动状态，并输出运动侦测结果。包含每帧图像的宏块 SAD 值信息、运动区域（OBJ）信息和整帧报警像素个数。
- 遮挡检测（OD）检测 VDA 通道接收的视频是否出现被遮挡现象，并输出遮挡检测结果。

用户通过创建 VDA 通道，绑定视频输入源，启动 VDA。每个通道只能选择一种工作模式，并根据工作模式配置相关通道属性，具体配置事项参见 HI_MPI_VDA_CreateChn。

7.2 功能描述

7.2.1 基本概念

- VDA 间隔

  间隔地接收输入源图像，以达到帧率控制的目的。根据输入源实际帧率进行调节。以 VI 帧率 25fps 为例，推荐每秒接收 5 帧图像，间隔设置为 4。如果没有特别说明，本章节出现的帧，皆指从输入源接收之后，进入 VDA 通道队列的图像。

- 宏块

  将图像划分为 8x8 或 16x16 或其它大小(以像素为单位)的块，每一块称为一个宏块。

- 宏块大小

  目前只支持 8x8 和 16x16 两种宏块。

- 宏块 SAD(Sum of Absolute Difference)

  两帧图像相应宏块之间的亮度差的绝对值之和。宏块 SAD 值越大，说明两帧对应宏块间的亮度变化越大。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 宏块 SAD 输出位数

  根据具体的需求，Hi35xx 提供了两种规格的 SAD 值输出 bit 位数：16bit 和 8bit。检测出的宏块 SAD 值最多占 16bit。选择 16bit 时，将输出实际的宏块 SAD 值，需要较高带宽；选择 8bit 时，将输出实际宏块 SAD 值的高 8 位，需要较低带宽。用户根据实际情况进行取舍。

- SAD 阈值

  芯片内部处理采用 4x4 宏块，该阈值用于芯片内部处理，用于判断宏块是否运动（宏块实际 SAD 值大或等于阈值，则为运动）。然后再根据这些运动的宏块分析和统计出 MD 的运动区域信息、整帧报警像素个数，或者 OD 的遮挡信息。以上的宏块皆指芯片内部处理的 4x4 宏块。SAD 阈值以 4x4 宏块为统计单位，与用户配置的宏块大小无关。

- 参考帧模式

  参考帧模式有三种：静态参考帧模式、动态参考帧模式和用户参考帧模式。参考帧模式决定 VDA 处理过程中参考帧的产生方式。

  暂不支持用户参考帧模式。推荐使用动态参考帧模式。

- VDA 算法

  VDA 算法包含帧差法（VDA_ALG_REF）和背景法（VDA_ALG_BG）两种。在创建通道时指定，推荐使用背景法。

- 帧差法（VDA_ALG_REF）

  直接以图像为参考帧，得出视频侦测分析结果的算法，称为帧差法。

  参考帧有两种产生方式：

  - 静态参考帧模式下，直接以某一固定图像为参考帧。
  - 动态参考帧模式下，以上一帧为参考帧。

- 背景法（VDA_ALG_BG）

  在 VDA 处理的过程中，将产生当前视频的背景图像。然后以背景图像为参考帧，得出视频侦测分析结果的算法，称为背景法。

  背景产生的方式有两种：

  - 静态参考帧模式下，直接以某一固定帧为背景。
  - 动态参考帧模式下，通过不断检测出视频的静止部分，内部组合得到背景。

- 背景更新权重

  当 VDA 算法选择为背景法，且参考模式为动态参考模式，设置的背景更新权重才会生效，但需要配置，以作参数检查。每次 VDA 处理都会产生静止部分图像，这部分图像和背景会作一次像素值叠加，新背景 = （静止部分图像的叠加权重×静止部分图像 + （256 -静止部分图像的叠加权重）x 旧背景）/256。静止部分图像的叠加权重就是用户需要设置的背景更新权重。值越大，背景更新速度越快。推荐值为 128。

- VDA 结果

  即视频侦测分析结果，根据通道工作模式输出的运动侦测结果或遮挡分析结果。具体内容参见 VDA_DATA_S。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

7.2.2 创建通道

用户通过创建一路 VDA 通道，绑定输入源，调用开始接收图像接口，来启动 VDA 功能。每一路通道只能指定一种工作模式，同时配置通道宽高和当前工作模式下的具体通道属性。两种工作模式的通道属性有一定差别，称工作模式为 MD 的通道为 MD 通道，工作模式为 OD 的为 OD 通道，本节简要描述两种类型通道的异同点和注意事项。

相同配置项

- VDA 算法(enVdaAlg)

  支持帧差法和背景法。
- 参考帧模式(enRefMode)

  支持静态参考帧模式和动态参考帧模式。
- VDA 间隔(u32VdaIntvl)

  调整适合间隔，以在 VDA 性能范围之内。VDA 支持最大性能：D1 为 15fps，CIF 为 30fps。
- 宏块大小(enMbSize)

  支持 8x8 和 16x16。
- SAD 输出 bit 位(enMbSadBits)

  支持 8bit 和 16bit。
- 背景更新权重(u32BgUpSrcWgt)

  仅当 VDA 算法选择为背景法时，该项有效。

不同配置项(only for MD)

- MD 结果缓存个数（u32MdBufNum）

  MD 通道会输出每一帧图像的每个宏块 SAD 值等运动侦测结果，每帧图像的结果都会占据一块较大的内存，此时会产生一个结果队列，用户可以指定这个结果队列的深度，即 MD 结果缓存个数。
- SAD 阈值(u32SadTh)

  MD 对整帧图像进行处理，输出整帧的运动侦测结果，所有运动相关的信息都根据该值进行判定。
- 运动区域输出最大个数(u32ObjNumMax)

  芯片会根据宏块的实际 SAD 值和用户设定的 SAD 阈值去判断宏块是否发生运动，通过内部边缘搜索算法将运动的区域检测出来，并输出该区域的左上角和右下角坐标。一帧图像检测出的区域个数不确定，可以不全部输出，用户通过设置该参数来控制输出的最大个数。推荐设置为 128。

不同配置项(only for OD)

- OD 区域个数(u32RgnNum)

  OD 支持将一帧图像分割成多个区域分别进行遮挡检测。最大支持个数为 4，用户可以自行设置区域的个数，并设置每个区域的属性 astOdRgnAttr。配置区域属性

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

时，其数组下标必须从 0 开始，直到（u32RgnNum -1）。OD 每次都只处理一帧的一个区域。

- 区域范围(stRect)

  描述该区域在整帧图像的位置和大小。区域之间可以重叠，其它约束参见 VDA_OD_RGN_ATTR_S。

- 区域 SAD 阈值(u32SadTh)

  用途如同 MD 中的 SAD 阈值，区别在于：该值只用于当前区域，用户设置的每个区域都必须设置 SAD 阈值，可以不同。

- 区域面积报警阈值(u32AreaTh)

  根据用户设定区域 SAD 阈值，每次 VDA 处理一帧图像之后，会检测出图像当前区域的最大运动区域，当该运动区域面积占当前区域面积的百分比大于或等于用户设定的区域面积报警阈值，则当前区域发生了遮挡。面积报警阈值是相对于区域面积的一个百分比值，因此它的取值范围是[0, 100]，推荐在 70%以上。

- 区域遮挡次数报警阈值(u32OccCntTh)

  通常，某一个区域被检测出一次遮挡，并不立即报警，而是该区域持续检测出若干次遮挡之后，才被认定为发生了遮挡。这个次数限制即是区域遮挡次数报警阈值，当区域持续检测到的遮挡计数大于或等于该区域遮挡次数报警阈值，则将报警信息输出，并停止该区域的遮挡检测。

- 区域遮挡计数时，允许不发生遮挡的最大次数(u32UnOccCntTh)

  也称误差值。在区域遮挡计数期间，只要检测到的不遮挡次数小于用户设置的误差值，依然进行遮挡计数，而不是只要被检测出不遮挡，就中止遮挡次数计数。在很多次检测到遮挡中间出现少数几次没有检测到遮挡的情况，依然会被判定为发生区域遮挡，并将其报警信息提供给用户。

配置注意事项

- 如何配置 SAD 阈值

  在画面噪声较大时，该值相对较大，相反，该值相对较小。

  在初次进行 SAD 阈值时，可以如下操作：

  - 先创建 MD 通道，指定宏块大小为 8x8，输出 bit 位为 16bit，输出宏块的 SAD 值。
  - 进行一些遮挡操作，观察输出的 SAD 值，选取一个中间值（大概即可）。
  - 将该中间值除于 4，得到一个粗 SAD 域值
  - 创建一个 OD 通道，在这个粗 SAD 阈值上下范围进行调试。以找到合适的 SAD 阈值。建议以 16 为步长，在粗 SAD 阈值上下范围各尝试 5 次左右。

- 如何配置 VDA 间隔

  以经典场景为例，VI 输入 D1@30fps，1 路 MD，1 路 OD（1 个区域）。

  推荐设置：MD，OD 间隔都为 5

  此时 MD 和 OD 的工作性能都为 30/(5+1)=5fps，总和为 10fps，在 VDA 性能之内。

- 如何配置 MD 结果缓存个数(u32MdBufNum)

  缓存个数设置的越大，占用内存越大，表示 VDA 的水线越大。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

缓存个数设置的越小，占用内存越小，但对结果获取的频率要求越高，否则会阻塞 VDA 的处理。

推荐值为 8。

- 如何配置宏块大小(enMbSize)和输出 bit 位数(enMbSadBits)

  宏块 8x8 时，精度高，带宽大；宏块 16x16 时，精度低，带宽小。

  输出 8bit 时，精度低，带宽小；输出 16bit 时，精度高，带宽大。

  用户可以在精度和带宽之间选取一个平衡点。

- 如何配置 u32OccCntTh、u32UnOccCntTh

  u32OccCntTh 和输入源帧率、VDA 间隔共同决定：输出遮挡报警信息需要持续出现区域遮挡的时长。

  一般地，VI 输入 30fps，VDA 间隔为 5，u32OccCntTh 为 20（暂不考虑 u32UnOccCntTh 的影响）。

  需要持续出现区域遮挡的时长为：20/(30/(5+1)) = 4s。即是，当用一个物体遮挡住镜头，大致 4s 就会输出遮挡报警信息。用户根据上述关系调整 u32OccCntTh（配合输入源帧率和 VDA 间隔）以达到在用户预期时长内输出报警信息。

  u32UnOccCntTh 是一个辅助参数，推荐和 u32OccCntTh 基本成 1:10 比例关系，甚至更小。例如，在上述场景下，u32UnOccCntTh 可以设置为 2，甚至为 1 或 0。

7.2.3 输入源

用户可选择的绑定输入源有 VI 输出，解码输出(Hi3532/Hi3518/Hi3516C 不支持)，VPSS 的 bypass 通道输出，以及 VPSS 其它通道的 user 模式输出。用户也可以通过接口函数 HI_MPI_VDA_UserSendPic 发送图像给 VDA 通道，不需要绑定。

7.2.4 处理 VDA 结果

启动通道之后（创建通道，绑定输入源，开始接收图像），将会启动硬件进行 VDA 处理。本节将分别介绍 MD 通道和 OD 通道产生的结果，以及相关事项。

MD 通道

- MD 结果

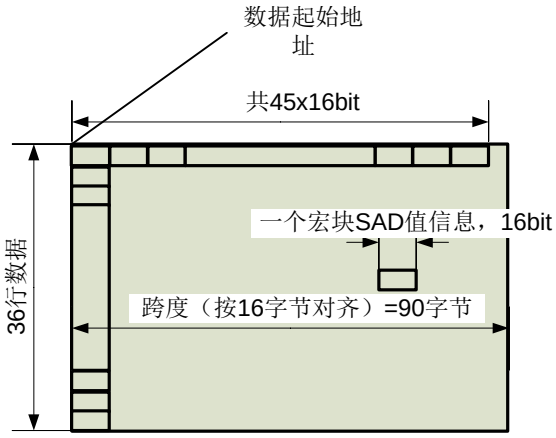
  参见 VDA_MD_DATA_S。

- 宏块 SAD 值结果

  输出整帧图像每个宏块的 SAD 阈值，具体结构参见 VDA_MB_SAD_DATA_S。它包含三个变量，数据起始地址，跨度，以及输出 bit 位。例如，用户配置宏块大小为 16x16，输出 bit 位为 16bit，MD 通道大小为 720x576。则它以宏块为单位的宽高是 45x36，即包含 45x36 个宏块。其宏块 SAD 值结果的内存分布信息如图 7-1 所示。具体读写操作参考 sample。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图7-1 宏块 SAD 值结果内存分布示意图



- 运动区域（OBJ）信息结果

  运动区域指，硬件通过用户配置的 SAD 阈值检测出当前帧的运动的部分，并以矩形框的形式划出的区域。具体结构参见 VDA_OBJ_DATA_S。具体读写操作参考 sample。

- 整帧报警像素个数

  输出整帧图像运动的像素总数。当进行喷雾实验时，芯片检测到的运动区域会很多而小。此时，可通过整帧报警像素个数来作为辅助参数。

OD 通道

OD 通道中只有一个报警信息，就是用户设定的区域是否发生了遮挡。

- 区域报警信息

  具体结构参见 VDA_OD_DATA_S。首先输出当前用户设定的区域个数，然后将输出一个布尔数组，用于依次标注该区域是否报警，HI_TRUE 表示已经遮挡，HI_FALSE 表示不遮挡。例如，用户设定了 3 个区域，依次为 A，B，C。当 B 区域发生遮挡之后，用户获取结果，将得到如下信息：

  `u32RgnNum = 3; abRgnAlarm[0] = 0, abRgnAlarm[1] = 1, abRgnAlarm[2] = 0, abRgnAlarm[3]无效。`

7.3 API 参考

该功能模块提供以下 MPI：

- HI_MPI_VDA_CreateChn：创建视频侦测通道。
- HI_MPI_VDA_DestroyChn：销毁视频侦测通道。
- HI_MPI_VDA_SetChnAttr：设置视频侦测通道属性。
- HI_MPI_VDA_GetChnAttr：获取视频侦测通道属性。
- HI_MPI_VDA_StartRecvPic：开始接收图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VDA_StopRecvPic：停止接收图像。
- HI_MPI_VDA_GetData：获取视频侦测分析结果。
- HI_MPI_VDA_ReleaseData：释放视频侦测分析结果。
- HI_MPI_VDA_ResetOdRegion：重新进行区域遮挡侦测
- HI_MPI_VDA_Query：查询视频侦测通道状态
- HI_MPI_VDA_UpdateRef：更新参考图像
- HI_MPI_VDA_GetFd：获取视频侦测分析通道对应的设备文件句柄。
- HI_MPI_VDA_UserSendPic：向视频侦测通道发送图像。

HI_MPI_VDA_CreateChn

【描述】

创建视频侦测通道。

【语法】

```
HI_S32 HI_MPI_VDA_CreateChn(VDA_CHN VdaChn, const VDA_CHN_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测通道号。 取值范围：[0,VDA_CHN_NUM_MAX)。	输入
pstAttr	视频侦测通道属性	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 一个通道只能指定一种工作模式。
- 可创建多个通道，最大通道数为 VDA_CHN_NUM_MAX。
- 当选择 VDA 算法为帧差法时，同样需要设置背景权重，以作参数检查。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 绑定输入源，调用 HI_MPI_VDA_StartRecvPic 接口之后，才开始启动 VDA 功能。

【举例】

```
HI_S32 s32Ret = HI_SUCCESS;
VDA_CHN VdaChn;
VDA_CHN_ATTR_S stVdaChnAttr;
MPP_CHN_S stSrcChn;
MPP_CHN_S stDestChn;
VdaChn = 0;
stVdaChnAttr.enWorkMode = VDA_WORK_MODE_MD;
stVdaChnAttr.u32Width   = 720;
stVdaChnAttr.u32Height  = 576;
stVdaChnAttr.unAttr.stMdAttr.enVdaAlg      = VDA_ALG_BG;
stVdaChnAttr.unAttr.stMdAttr.enMbSize      = VDA_MB_16PIXEL;
stVdaChnAttr.unAttr.stMdAttr.enMbSadBits   = VDA_MB_SAD_8BIT;
stVdaChnAttr.unAttr.stMdAttr.enRefMode     = VDA_REF_MODE_DYNAMIC;
stVdaChnAttr.unAttr.stMdAttr.u32VdaIntvl   = 12;
stVdaChnAttr.unAttr.stMdAttr.u32BgUpSrcWgt = 128;
stVdaChnAttr.unAttr.stMdAttr.u32MdBufNum   = 8;
stVdaChnAttr.unAttr.stMdAttr.u32ObjNumMax  = 128;
stVdaChnAttr.unAttr.stMdAttr.u32SadTh      = 40;
 /*创建通道*/
s32Ret = HI_MPI_VDA_CreateChn(VdaChn, &stVdaChnAttr);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
/*绑定VI*/
stSrcChn.enModId = HI_ID_VIU;
stSrcChn.s32ChnId = 0;
stDestChn.enModId = HI_ID_VDA;
stDestChn.s32ChnId = VdaChn;

s32Ret = HI_MPI_SYS_Bind(&stSrcChn, &stDestChn);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
/*启动接收图像*/
s32Ret = HI_MPI_VDA_StartRecvPic(VdaChn);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    }
     /*停止接收图像*/
    s32Ret = HI_MPI_VDA_StopRecvPic(VdaChn);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
    }
    /*解绑定VI*/
    s32Ret = HI_MPI_SYS_UnBind(&stSrcChn, &stDestChn);
    if(s32Ret != HI_SUCCESS)
    {
        return s32Ret;
}
s32Ret = HI_MPI_VDA_DestroyChn(VdaChn);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
```

【相关主题】

HI_MPI_VDA_DestroyChn

HI_MPI_VDA_DestroyChn

【描述】

销毁视频侦测通道。

【语法】

```
HI_S32 HI_MPI_VDA_DestroyChn(VDA_CHN VdaChn);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 通道必须已创建。
- 调用该接口之前，必须先调用 HI_MPI_VDA_StopRecvPic 接口停止接收图像。

【举例】

请参见 HI_MPI_VDA_CreateChn 举例。

【相关主题】

无。

HI_MPI_VDA_SetChnAttr

【描述】

设置视频侦测通道属性。

【语法】

```
HI_S32 HI_MPI_VDA_SetChnAttr(VDA_CHN VdaChn, const VDA_CHN_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入
pstAttr	视频侦测通道属性	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 只可以修改动态属性，不能修改静态属性，具体约束参见 VDA_CHN_ATTR_S。
- 通道必须已创建。
- 建议用户先获取属性，再进行设置。
- 输入通道属性指针不能为空。

【举例】

```
HI_S32 s32Ret = HI_SUCCESS;
VDA_CHN VdaChn;
VDA_CHN_ATTR_S stVdaChnAttrIn;
VdaChn = 0;
s32Ret = HI_MPI_VDA_GetChnAttr(VdaChn, &stVdaChnAttrIn);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
 /*更改间隔*/
stVdaChnAttrIn.unAttr.stMdAttr.u32VdaIntvl   = 7;
s32Ret = HI_MPI_VDA_SetChnAttr(VdaChn, &stVdaChnAttrIn);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
```

【相关主题】

HI_MPI_VDA_GetChnAttr

HI_MPI_VDA_GetChnAttr

【描述】

获取视频侦测通道属性。

【语法】

```
HI_S32 HI_MPI_VDA_GetChnAttr(VDA_CHN VdaChn, VDA_CHN_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入
pstAttr	视频侦测通道属性	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 通道必须已创建。
- 输出通道属性指针不能为空。

【举例】

请参见 HI_MPI_VDA_SetChnAttr 举例。

【相关主题】

无。

HI_MPI_VDA_StartRecvPic

【描述】

开始接收图像。

【语法】

```
HI_S32 HI_MPI_VDA_StartRecvPic(VDA_CHN VdaChn);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 通道必须已创建。
- 与绑定输入源操作无先后关系。
- 支持多次调用。

【举例】

请参见 HI_MPI_VDA_CreateChn 举例。

【相关主题】

无。

HI_MPI_VDA_StopRecvPic

【描述】

停止接收图像。

【语法】

```
HI_S32 HI_MPI_VDA_StopRecvPic(VDA_CHN VdaChn);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 通道必须已创建。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 与解绑定输入源操作无先后关系。
- 支持多次调用该接口。

【举例】

请参见 HI_MPI_VDA_CreateChn 举例。

【相关主题】

无。

HI_MPI_VDA_GetData

【描述】

获取视频侦测分析结果，如果通道工作模式是 MD，它将获取到的是 MD 结果；如果工作模式是 OD，它将获取到的是 OD 结果。

【语法】

```
HI_S32 HI_MPI_VDA_GetData(VDA_CHN VdaChn, VDA_DATA_S *pstVdaData, HI_BOOL bBlock);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入
pstVdaData	视频侦测分析结果。	输出
bBlock	阻塞标志。 取值范围： • HI_TRUE：阻塞。 • HI_FALSE：非阻塞。	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 通道必须已创建。
- 支持阻塞模式获取结果。
- 输出结果指针不能为空。
- MD 通道，阻塞模式下，MD 结果缓存为空时，等待 MD 处理完成。
- MD 通道，非阻塞模式下，MD 结果缓存为空时，返回错误。
- MD 通道，获取结果之后，不能修改结果中的时间戳信息。
- OD 通道，不区分阻塞和非阻塞模式，可一直获取结果。

【举例】

请参见 sample。

【相关主题】

无。

HI_MPI_VDA_ReleaseData

【描述】

释放视频侦测分析结果。

【语法】

```
HI_S32 HI_MPI_VDA_ReleaseData(VDA_CHN VdaChn, const VDA_DATA_S*
pstVdaData);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0,VDA_CHN_NUM_MAX)。	输入
pstVdaData	视频侦测分析结果。	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- 通道必须已创建。
- 输入结果指针不能为空。
- MD 通道，视频侦测分析结果需要在使用完之后立即释放，如果不及时释放会导致视频侦测分析结果缓存占满，阻塞通道启动。
- MD 通道，获取结果之后，不支持多次释放结果。
- MD 通道，不支持释放不存在的结果。
- OD 通道，获取结果之后，支持不释放、一次释放和多次释放。
- 不支持调用该接口未返回时，删除该通道。可能会出现，通道删除成功，但该接口会返回通道不存在错误。

【举例】

请参见 sample。

【相关主题】

HI_MPI_VDA_GetData

HI_MPI_VDA_ResetOdRegion

【描述】

重新进行遮挡侦测，仅 OD 通道使用。假如 OD 通道设定了四个区域，依次是 A、B、C、D。若 OD 的区域 A 发生了遮挡，A 区域将自动停止检测，并输出区域遮挡报警信息，但 B、C、D 区域将继续检测。

用户获取到区域报警信息之后，为了让该通道的 A 区域继续工作，需调用该接口，输入通道号和 A 区域所在的数组下标 0。A 区域将可以继续检测，不会对其它区域产生影响。

【语法】

```
HI_S32 HI_MPI_VDA_ResetOdRegion(VDA_CHN VdaChn, HI_S32 s32RgnIndex);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0,VDA_CHN_NUM_MAX)。	输入
s32RgnIndex	区域数组下标值[0, VDA_OD_RGN_NUM_MAX)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 通道必须已创建。
- 支持多次调用。
- 获取到区域报警信息之后，需要调用此接口重新让该区域进行遮挡检测。

【举例】

请参见 sample。

【相关主题】

无

HI_MPI_VDA_Query

【描述】

查询视频侦测分析通道状态，查看当前通道图像队列中有多少帧图像、当前是否处于接收图像状态，当前结果缓存中的结果个数。

【语法】

```
HI_S32 HI_MPI_VDA_Query(VDA_CHN VdaChn, VDA_CHN_STAT_S *pstChnStat);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入
pstChnStat	通道状态。	输出

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	参见错误码

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

- 通道必须已创建。
- 输出通道状态指针不为空。

【举例】

请参见 sample。

【相关主题】

无。

HI_MPI_VDA_UpdateRef

【描述】

更新参考帧，此接口只适用于参考帧模式为用户模式的情形。

【语法】

```
HI_S32 HI_MPI_VDA_UpdateRef(VDA_CHN VdaChn, const VIDEO_FRAME_INFO_S
*pstRefFrame);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0,VDA_CHN_NUM_MAX)。	输入
pstRefFrame	参考图像信息。	输入

【返回值】

返回值	描述
0	成功。
非 0	参见错误码

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

当参考帧模式为用户模式时，必须先调用此接口设置参考帧，否则调用 HI_MPI_VDA_StartRecvPic 接口会出错，VDA 不能正常接收图像。

【举例】

无。

【相关主题】

无。

HI_MPI_VDA_GetFd

【描述】

获取视频侦测分析通道对应的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_VDA_GetFd(VDA_CHN VdaChn);
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【举例】

无。

【相关主题】

无。

HI_MPI_VDA_UserSendPic

【描述】

向视频侦测通道发送图像

【语法】

```
HI_S32 HI_MPI_VDA_UserSendPic(VDA_CHN VdaChn, const VIDEO_FRAME_INFO_S
*pstUserFrame, HI_BOOL bBlock, HI_U32 u32MilliSec)
```

【参数】

参数名称	描述	输入/输出
VdaChn	视频侦测分析通道号。 取值范围：[0, VDA_CHN_NUM_MAX)。	输入
pstUserFrame	用户要发送的图像信息指针	输入
bBlock	阻塞标志。 取值范围： • HI_TRUE：阻塞。 • HI_FALSE：非阻塞	输入
u32MilliSec	使用阻塞方式时的等待超时时间(单位：ms)	输入

【返回值】

返回值	描述
0	成功。
非 0	发生超时或出错，参见错误码。

【需求】

- 头文件：hi_comm_vda.h、mpi_vda.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 通道必须已创建。
- 通道必须处于接收数据状态。
- 发送的图像信息指针不为空。
- 使用阻塞方式时,当 u32MilliSec 为 0 时表示函数将会一直阻塞,直到 VDA 能够接收图像。当 u32MilliSec 为非 0 时,表示在等待 u32MilliSec ms 内若能接收图像则返回成功,若等待时间超过 u32MilliSec ms 则发生超时,返回错误。
- 超时的有效时间单位为 10ms，因此实际最长等待时间为 10ms 倍数。

【举例】

无。

【相关主题】

无。

7.4 数据类型

相关数据类型、数据结构定义如下：

- VDA_CHN：定义视频侦测分析通道。
- VDA_CHN_NUM_MAX：定义视频侦测分析最大通道的个数。
- VDA_MAX_WIDTH：定义视频侦测分析通道的最大宽度。
- VDA_MAX_HEIGHT：定义视频侦测分析通道的最大高度。
- VDA_OD_RGN_NUM_MAX：定义遮挡区域的最大个数。
- VDA_OBJ_S：定义运动区域信息结构体。
- VDA_REF_MODE_E：定义参考帧模式枚举。
- VDA_ALG_E：定义 VDA 算法模式枚举。
- VDA_MB_SIZE_E：定义宏块大小枚举。
- VDA_MB_SADBITS_E：定义运动侦测的 SAD 的精度。
- VDA_OD_RGN_ATTR_S：定义遮挡区域属性结构体。
- VDA_MD_ATTR_S：定义移动侦测属性结构体。
- VDA_OD_ATTR_S：定义遮挡侦测属性结构体。
- VDA_WORK_MODE_E：定义视频侦测分析工作模式枚举。
- VDA_WORK_MODE_ATTR_U：定义模式属性联合体。
- VDA_CHN_ATTR_S：定义通道属性结构体。
- VDA_MB_SAD_DATA_S：定义宏块的 SAD 值信息结构体。
- VDA_OBJ_DATA_S：定义运动区域信息结构体。
- VDA_MD_DATA_S：定义移动侦测结果结构体。
- VDA_OD_DATA_S：定义遮挡侦测结果结构体。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VDA_DATA_U：定义视频侦测结果联合体。
- VDA_DATA_S：定义视频侦测分析结果结构体。
- VDA_CHN_STAT_S：定义视频侦测分析通道状态结构体。

VDA_CHN

【说明】

定义视频侦测分析通道。

【定义】

```
typedef HI_S32 VDA_CHN;
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_CHN_NUM_MAX

【说明】

定义视频侦测分析最大通道的个数。

【定义】

For Hi3521/Hi3520A/Hi3518/Hi3516C/Hi3520D/Hi3515A/Hi3515C

```
#define VDA_CHN_NUM_MAX   32
```

For Hi3531/Hi3532

```
#define VDA_CHN_NUM_MAX   48
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_MAX_WIDTH

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义视频侦测分析通道的最大宽度。

【定义】

```
#define VDA_MAX_WIDTH   960
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_MAX_HEIGHT

【说明】

定义视频侦测分析通道的最大高度。

【定义】

```
#define VDA_MAX_HEIGHT   576
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_OD_RGN_NUM_MAX

【说明】

定义遮挡区域的最大个数。

【定义】

```
#define VDA_OD_RGN_NUM_MAX 4
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

VDA_OBJ_S

【说明】

定义运动区域信息结构体。

【定义】

```
typedef struct hiVDA_OBJ_S
{
    HI_U16 u16Left;
    HI_U16 u16Top;
    HI_U16 u16Right;
    HI_U16 u16Bottom;
}VDA_OBJ_S;
```

【成员】

成员名称	描述
u16Left	区域左上角 X 坐标的值。
u16Top	区域左上角 Y 坐标的值。
u16Right	区域右下角 X 坐标的值。
u16Bottom	区域右下角 Y 坐标的值。

【注意事项】

无

【相关数据类型及接口】

无。

VDA_REF_MODE_E

【说明】

定义参考帧模式枚举。

【定义】

```
typedef enum hiVDA_REF_MODE_E
{
    VDA_REF_MODE_DYNAMIC = 0,
    VDA_REF_MODE_STATIC,
    VDA_REF_MODE_USER,
    VDA_REF_MODE_BUTT
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
}VDA_REF_MODE_E;
```

【成员】

成员名称	描述
VDA_REF_MODE_DYNAMIC	动态参考帧模式。
VDA_REF_MODE_STATIC	静态参考帧模式。
VDA_REF_MODE_USER	用户参考帧模式（未实现）。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_ALG_E

【说明】

定义 VDA 算法模式枚举。

【定义】

```
typedef enum hiVDA_ALG_E
{
    VDA_ALG_BG = 0,
    VDA_ALG_REF,
    VDA_ALG_BUTT
}VDA_ALG_E;
```

【成员】

成员名称	描述
VDA_ALG_BG	背景法。
VDA_ALG_REF	帧差法。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

VDA_MB_SIZE_E

【说明】

定义宏块大小枚举。

【定义】

```
typedef enum hiVDA_MB_SIZE_E
{
    VDA_MB_8PIXEL,      /* 8*8 */
    VDA_MB_16PIXEL,     /* 16*16 */
    VDA_MB_BUTT
}VDA_MB_SIZE_E;
```

【成员】

成员名称	描述
VDA_MB_8PIXEL	宏块大小 8x8。
VDA_MB_16PIXEL	宏块大小 16x16。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_MB_SADBITS_E

【说明】

定义运动侦测的 SAD 的输出位数。

【定义】

```
typedef enum hiVDA_MB_SADBITS_E
{
    VDA_MB_SAD_8BIT = 0,  /*SAD precision 8bits*/
    VDA_MB_SAD_16BIT,     /*SAD precision 16bits*/
    VDA_MB_SAD_BUTT      /*reserve*/
} VDA_MB_SADBITS_E;
```

【成员】

成员名称	描述
VDA_MB_SAD_8BIT	宏块 SAD 输出位数为 8bit。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VDA_MB_SAD_16BIT	宏块 SAD 输出位数为 16bit。

【注意事项】

- 输出值以用户设定的宏块大小为统计单位。
- 8bit 输出的值是 16bit 输出值的高 8 位，使用时请注意。

【相关数据类型及接口】

无。

VDA_OD_RGN_ATTR_S

【说明】

定义遮挡区域属性结构体。

【定义】

```
typedef struct hiVDA_OD_RGN_ATTR_S
{
    RECT_S stRect;
    HI_U32 u32SadTh;
    HI_U32 u32AreaTh;
    HI_U32 u32OccCntTh;
    HI_U32 u32UnOccCntTh;
}VDA_OD_RGN_ATTR_S;
```

【成员】

成员名称	描述
stRect	区域范围。 X 取值范围：[0, VDA_MAX_WIDTH)；对齐：16 Y 取值范围：[0, VDA_MAX_HEIGHT) W 取值范围：[16, VDA_MAX_WIDTH]；对齐：16 H 取值范围：[16, VDA_MAX_HEIGHT]；对齐：16 X+W≤通道宽度。 Y+H ≤通道高度。 静态属性。
u32SadTh	区域宏块报警阈值。 取值范围：[0, 4080]。 动态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32AreaTh	区域面积报警阈值。 取值范围：[0, 100]。 动态属性。
u32OccCntTh	区域遮挡次数报警阈值。 取值范围：[1, 256]。 动态属性。
u32UnOccCntTh	区域遮挡计数时，允许不发生遮挡的最大次数。 取值范围：[0, 256]。 动态属性。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_MD_ATTR_S

【说明】

定义移动侦测属性结构体。

【定义】

```
typedef struct hiVDA_MD_ATTR_S
{
    VDA_ALG_E       enVdaAlg;
    VDA_MB_SIZE_E    enMbSize;
    VDA_MB_SADBITS_E enMbSadBits;
    VDA_REF_MODE_E  enRefMode;
    HI_U32          u32MdBufNum;
    HI_U32          u32VdaIntvl;
    HI_U32          u32BgUpSrcWgt;
    HI_U32            u32SadTh;
    HI_U32          u32ObjNumMax;
}VDA_MD_ATTR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enVdaAlg	VDA 算法。 静态属性。
enMbSize	宏块大小。 静态属性。
enMbSadBits	SAD 输出精度。 静态属性。
enRefMode	参考图像模式。 静态属性。
u32MdBufNum	MD 结果缓存个数。 取值范围：[1, 16]。 静态属性。
u32VdaIntvl	侦测间隔。 取值范围：[0, 256]，以帧为单位。 动态属性。
u32BgUpSrcWgt	背景图像更新时源图像所占的权重，总权重为 256。 范围：[1, 255]，推荐值为：128。 动态属性。
u32SadTh	SAD 报警阈值。 动态属性。
u32ObjNumMax	输出的运动区域的最大个数。 范围：[1, 128]。 动态属性。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_OD_ATTR_S

【说明】

定义遮挡侦测属性结构体。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef struct hiVDA_OD_ATTR_S
{
    HI_U32          u32RgnNum;
    VDA_OD_RGN_ATTR_S astOdRgnAttr[VDA_OD_RGN_NUM_MAX];
    VDA_ALG_E        enVdaAlg;
    VDA_MB_SIZE_E    enMbSize;
    VDA_MB_SADBITS_E enMbSadBits;
    VDA_REF_MODE_E  enRefMode;
    HI_U32          u32VdaIntvl;
    HI_U32          u32BgUpSrcWgt;
}VDA_OD_ATTR_S;
```

【成员】

成员名称	描述
u32RgnNum	区域个数。 范围：[1, VDA_OD_RGN_NUM_MAX]。 静态属性。
astOdRgnAttr[VDA_OD_RGN_NUM_MAX]	区域属性，参见内部结构体。
enVdaAlg	VDA 算法。 静态属性。
enMbSize	宏块大小。 静态属性。
enMbSadBits	SAD 输出精度。 静态属性。
enRefMode	参考图像模式。 静态属性。
u32VdaIntvl	侦测间隔。 取值范围：[0, 256]，以帧为单位。 动态属性。
u32BgUpSrcWgt	背景图像更新时源图像所占的权重，总权重为 256。 范围：[1, 255]，推荐值为：128。 动态属性。

【注意事项】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关数据类型及接口】

无。

VDA_WORK_MODE_E

【说明】

定义视频侦测分析工作模式枚举。

【定义】

```
typedef enum hiVDA_WORK_MODE_E
{
    VDA_WORK_MODE_MD = 0,
    VDA_WORK_MODE_OD,
    VDA_WORK_MODE_BUTT
}VDA_WORK_MODE_E;
```

【成员】

成员名称	描述
VDA_WORK_MODE_MD	移动侦测
VDA_WORK_MODE_OD	遮挡侦测

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_WORK_MODE_ATTR_U

【说明】

定义模式属性联合体。

【定义】

```
typedef union hiVDA_WORK_MODE_ATTR_U
{
    VDA_MD_ATTR_S stMdAttr;
    VDA_OD_ATTR_S stOdAttr;
}VDA_WORK_MODE_ATTR_U;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stMdAttr	移动侦测属性。
stOdAttr	遮挡侦测属性。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_CHN_ATTR_S

【说明】

定义通道属性结构体。

【定义】

```
typedef struct hiVDA_CHN_ATTR_S
{
    VDA_WORK_MODE_E enWorkMode;
    VDA_WORK_MODE_ATTR_U unAttr;
    HI_U32 u32Width;
    HI_U32 u32Height;
}VDA_CHN_ATTR_S;
```

【成员】

成员名称	描述
enWorkMode	工作模式。 静态属性。
unAttr	工作模式属性。
u32Width	通道宽度。 范围：[16, VDA_MAX_WIDTH]。16 对齐。 静态属性。
u32Height	通道高度。 范围：[16, VDA_MAX_HEIGHT]。16 对齐。 静态属性。

【注意事项】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

通道大小可以和输入源大小不一致，输入源图像的宽和高都必须大于等于通道属性的宽和高。

【相关数据类型及接口】

无。

VDA_MB_SAD_DATA_S

【说明】

定义宏块的 SAD 值信息结构体。

【定义】

```
typedef struct hiVDA_MB_SAD_DATA_S
{
    HI_VOID    *pAddr;           /*address*/
    HI_U32      u32Stride;        /*stride*/
    VDA_MB_SADBITS_E enMbSadBits;  /*the MB SAD size*/
}VDA_MB_SAD_DATA_S;
```

【成员】

成员名称	描述
pAddr	宏块 SAD 值得内存首地址。
u32Stride	宏块数据以字节为单位的内存行宽度。
enMbSadBits	SAD 输出 bit 位数。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_OBJ_DATA_S

【说明】

定义运动区域信息结构体。

【定义】

```
typedef struct hiVDA_OBJ_DATA_S
{
    HI_U32 u32ObjNum;
    VDA_OBJ_S    *pstAddr;
    HI_U32 u32IndexOfMaxObj;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32SizeOfMaxObj;
    HI_U32 u32SizeOfTotalObj;
}VDA_OBJ_DATA_S;
```

【成员】

成员名称	描述
u32ObjNum	运动区域的个数。
pstAddr	运动区域信息内存首地址。
u32IndexOfMaxObj	最大面积区域索引值。
u32SizeOfMaxObj	最大面积区域的面积。
u32SizeOfTotalObj	运动区域的总面积。

【注意事项】

- 最大运动区域的索引值从 0 开始。
- 若运动区域上报的坐标为(0,4),(16,36)。则区域面积为（16-0+4）x(36-4+4)=720.。实际宽=上报区域坐标计算出的宽+4；高度也是如此。
- 从 pstAddr 地址开始，依次存放检测出的运动区域 VDA_OBJ_S 信息。存放个数为 u32ObjNum。

【相关数据类型及接口】

无。

VDA_MD_DATA_S

【说明】

定义移动侦测结果结构体。

【定义】

```
typedef struct hiVDA_MD_DATA_S
{
    HI_BOOL          bMbSadValid;
    VDA_MB_SAD_DATA_S stMbSadData;
    HI_BOOL          bObjValid;
    VDA_OBJ_DATA_S    stObjData;
    HI_BOOL          bPelsNumValid;
    HI_U32            u32AlarmPixCnt;
}VDA_MD_DATA_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bMbSadValid	宏块 SAD 值的有效性。 HI_TRUE：有效； HI_FALSE：无效。
stMbSadData	宏块 SAD 值结果。
bObjValid	运动区域值的有效性。 HI_TRUE：有效； HI_FALSE：无效。
stObjData	运动区域结果。
bPelsNumValid	报警像素的个数值的有效性。 HI_TRUE：有效； HI_FALSE：无效。
u32AlarmPixCnt	报警像素的总个数。

【注意事项】

bMbSadValid，bObjValid，bPelsNumValid 默认有效，暂不提供接口进行设置。

【相关数据类型及接口】

无。

VDA_OD_DATA_S

【说明】

定义遮挡侦测结果结构体。

【定义】

```
typedef struct hiVDA_OD_DATA_S
{
    HI_U32  u32RgnNum;
    HI_BOOL abRgnAlarm[VDA_OD_RGN_NUM_MAX];
}VDA_OD_DATA_S;
```

【成员】

成员名称	描述
u32RgnNum	区域的个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
abRgnAlarm[VDA_OD_RGN_NUM_MAX]	区域是否告警。 HI_TRUE：告警； HI_FALSE：不告警。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_DATA_U

【说明】

定义视频侦测结果联合体。

【定义】

```
typedef union hiVDA_DATA_U
{
    VDA_MD_DATA_S stMdData;
    VDA_OD_DATA_S stOdData;
}VDA_DATA_U;
```

【成员】

成员名称	描述
stMdData	移动侦测结果。
stOdData	遮挡侦测结果。

【注意事项】

无

【相关数据类型及接口】

无。

VDA_DATA_S

【说明】

定义视频侦测分析结果结构体。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef struct hiVDA_DATA_S
{
    VDA_WORK_MODE_E enWorkMode;
    VDA_DATA_U      unData;
    VDA_MB_SIZE_E   enMbSize;
    HI_U32          u32MbWidth;
    HI_U32          u32MbHeight;
    HI_U64          u64Pts;
}VDA_DATA_S;
```

【成员】

成员名称	描述
enWorkMode	工作模式。
unData	结果联合体。
enMbSize	宏块大小。
u32MbWidth	通道以宏块为单位的宽度。
u32MbHeight	通道以宏块为单位的高度。
u64Pts	时间戳。

【注意事项】

无。

【相关数据类型及接口】

无。

VDA_CHN_STAT_S

【说明】

定义视频侦测分析通道状态结构体。

【定义】

```
typedef struct hiVDA_CHN_STAT_S
{
    HI_BOOL bStartRecvPic;
    HI_U32 u32LeftPic;
    HI_U32 u32LeftRst;
}VDA_CHN_STAT_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bStartRecvPic	是否接受图像。
u32LeftPic	剩余的通道图像队列的图像个数。
u32LeftRst	剩余的结果缓存个数。

【注意事项】

无。

【相关数据类型及接口】

无。

7.5 错误码

视频侦测分析 API 错误码如表 7-1 所示。

表7-1 视频侦测分析 API 错误码

错误代码	宏定义	描述
0xA0098001	HI_ERR_VDA_INVALID_DEVID	设备 ID 超出合法范围
0xA0098002	HI_ERR_VDA_INVALID_CHNID	通道 ID 超出合法范围
0xA0098003	HI_ERR_VDA_ILLEGAL_PARAM	参数超出合法范围
0xA0098004	HI_ERR_VDA_EXIST	试图申请或者创建已经存在的设备、通道或者资源
0xA0098005	HI_ERR_VDA_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源
0xA0098006	HI_ERR_VDA_NULL_PTR	参数中有空指针
0xA0098007	HI_ERR_VDA_NOT_CONFIG	使用前未配置
0xA0098008	HI_ERR_VDA_NOT_SUPPORT	不支持的参数或者功能
0xA0098009	HI_ERR_VDA_NOT_PERM	该操作不允许，如试图修改静态配置参数
0xA009800C	HI_ERR_VDA_NOMEM	分配内存失败，如系统内存不足
0xA009800D	HI_ERR_VDA_NOBUF	分配缓存失败，如申请的数据缓冲区太大
0xA009800E	HI_ERR_VDA_BUF_EMPTY	缓冲区中无数据

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA009800F	HI_ERR_VDA_BUF_FULL	缓冲区中数据满
0xA0098010	HI_ERR_VDA_SYS_NOTREADY	系统没有初始化或没有加载相应模块
0xA0098012	HI_ERR_VDA_BUSY	系统忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

8 区域管理.....8-1
8.1 概述.....8-1
8.2 功能描述.....8-1
8.2.1 重要概念.....8-1
8.2.2 使用示意.....8-4
8.3 API 参考.....8-4
8.4 数据类型.....8-17
8.5 错误码.....8-34

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 8-1 层次叠加示意图 8-2
图 8-2 Overlay 的宽高非 16 对齐示意图 8-27

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 8-1 region 支持的模块信息 8-3
表 8-2 区域管理 API 错误码 8-34

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

8 区域管理

8.1 概述

用户一般都需要在视频中叠加 OSD 用于显示一些特定的信息（如：通道号、时间戳等），必要时还会填充色块。这些叠加在视频上的 OSD 和遮挡在视频上的色块统称为区域。REGION 模块，用于统一管理这些区域资源。

区域管理可以实现区域的创建，并叠加到视频中或对视频进行遮挡。例如，实际应用中，用户通过创建一个区域，通过 HI_MPI_RGN_AttachToChn，将该区域叠加到某个通道（如 GROUP 通道）中。在通道进行调度时，则会将 OSD 叠加在视频中。一个区域支持通过设置通道显示属性接口指定到多个通道中（如：多个 GROUP 通道，多个 VI 通道，甚至多个 GROUP 和 VI 通道），且支持在每个通道的显示属性（如位置、层次、透明度等）都不同。

目前，Hi35xx 支持在 GROUP 通道叠加 Overlay、在 VI 通道填充 Cover 和 CoverEx 并叠加 OverlayEx，支持区域重叠，重叠时会根据区域的层次进行选择哪个区域显示在上层，而下层将会被覆盖。由于芯片等其它限制，不同模块的通道可以创建的区域种类和区域数也不一样，如每个 GROUP 通道最多可以创建 8 个 Overlay 区域；每个 VI 通道最多可创建 4 个 Cover，16 个 CoverEx，16 个 OverlayEx 区域。

8.2 功能描述

8.2.1 重要概念

- Overlay

  视频叠加区域，实现在 GROUP 通道进行位图的加载和背景色更新。
- Cover

  视频遮挡区域，实现在 VI 通道填充纯色块。由 VI 硬件自身完成。
- CoverEx

  扩展视频遮挡区域，实现在绑定通道填充纯色块，当 Cover 个数不能满足用户需求时，可以使用此扩展 cover。
- OverlayEx

  扩展视频叠加区域，实现在绑定通道进行位图的加载和背景色更新。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

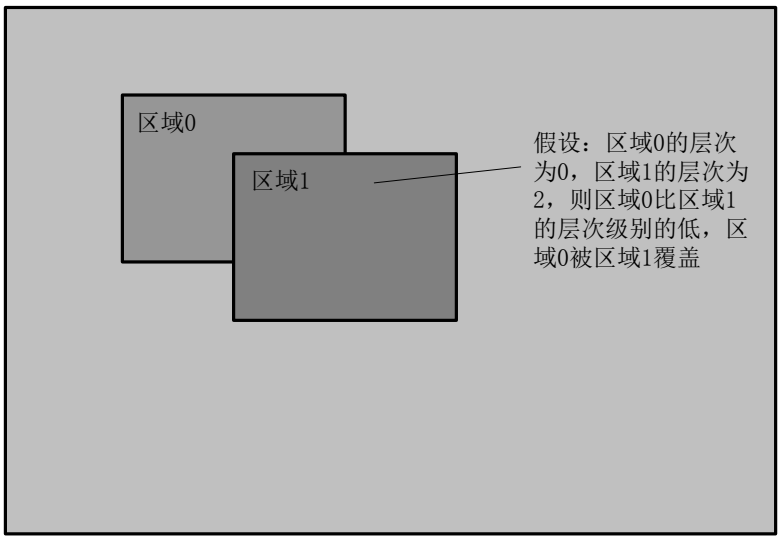
注意

对于 Hi3520D/Hi3515A/Hi3515C，如果需要用到这项功能，则应注意 TDE 模块 g_u32TdeTmpBufW 与 g_u32TdeTmpBufH 参数的设置，详细内容请参见《TDE API 参考》。

- 区域层次

  区域层次表示区域的叠加级别，层次值越大，表示区域的显示级别越高。当发生重叠时，层次值大的将会覆盖层次值小的。如果是同一级别，则根据区域叠加的先后顺序，后叠加的将显示在上层。

图8-1 层次叠加示意图



- 位图填充(针对 Overlay 和 OverlayEx 有效)

  位图填充是指将位图的内存值填充到区域内存空间中位图将会从区域的左上角开始填充。当位图小于区域时，将只能填充一部分内存，剩余部分保持原有值；位图大小等于区域时，将刚好全部填充；当位图大于区域时，位图只能将自身和区域一样大小的内存信息填充到区域中。

- 区域属性

  用户创建一个区域时，需要设置该属性信息，它包含公共的资源信息。例如，Overlay 包含像素格式，大小和背景色。

- 通道显示属性(RGN_CHN_ATTR_S)

  通道显示属性表明区域在某通道的显示特征。例如，Overlay 的通道显示属性包含显示位置，层次，前景 Alpha，背景 Alpha，还有编码用到的 QP 信息。当通道显示属性中的区域是否显示(bShow)为 TRUE 时，表示显示在该通道中；反之，表示在该通道中存在，但处于隐藏状态。

- OSD 反色(仅针对 Overlay)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

叠加在视频上的 OSD 可能会随着视频背景的变化而变得不清晰， OSD 反色功能会自适应背景变化，自动对 OSD 的颜色作出调整，达到 OSD 清晰可见的目的。具体原理为：当视频背景亮度发生变化，大于或小于(具体跟反色模式有关，大于对应反色触发模式为 MORETHAN_LUM_THRESH，小于对应反色触发模式为 LESSTHAN_LUM_THRESH)用户设置的亮度阈值，此时会对 OSD 做反色处理。

- 支持的模块

  目前 region 支持的模块信息如表 8-1 所示，在绑定 region 区域到通道时须参考此表。

表8-1 region 支持的模块信息

模块号	区域号	支持区域与否	设备号 取值范围	通道号 取值范围
HI_ID_VIU	OVERLAY_RGN	N	\	\
	COVER_RGN	Y	0	[0, VIU_MAX_CHN_NUM-1]
	COVEREX_RGN	Y	0	[0, VIU_MAX_CHN_NUM-1]
	OVERLAYEX_RGN	Y	0	[0, VIU_MAX_CHN_NUM-1]
HI_ID_GROUP	OVERLAY_RGN	Y	[0,VENC_MAX_GRP_NUM-1]	0
	COVER_RGN	N	\	\
	COVEREX_RGN	N	\	\
	OVERLAYEX_RGN	N	\	\
HI_ID_PCIV	OVERLAY_RGN	N	\	\
	COVER_RGN	N	\	\
	COVEREX_RGN	N	\	\
	OVERLAYEX_RGN	Y	0	[0,PCIV_MAX_CHN_NUM-1]

注：Hi3518/Hi3516C 不支持 OVERLAYEX_RGN。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

8.2.2 使用示意

使用过程包含以下步骤：

- 用户填充区域属性并创建区域。
- 将该区域指定到具体通道中(如 GROUP)。在指定到具体通道时，需要输入通道的显示属性。

以上步骤完成区域的创建和使用。用户还可以通过以下操作来控制区域属性以及在某通道的通道显示属性。

- 通过 HI_MPI_RGN_GetAttr、HI_MPI_RGN_SetAttr 获取和设置区域属性。
- 通过 HI_MPI_RGN_SetBitMap(仅针对 Overlay)设置区域的位图信息。
- 通过 HI_MPI_RGN_GetDisplayAttr，HI_MPI_RGN_SetDisplayAttr 获取和设置区域在某通道（如 GROUP 通道）的通道显示属性。
- 最后用户可以将该区域从通道中撤出（非必须操作），再销毁区域。

其它具体细节参考 sample。

8.3 API 参考

区域管理模块主要提供区域资源的控制管理功能，包括区域的创建、销毁，获取与设置区域属性，获取与设置区域的通道显示属性等。

该功能模块提供以下 MPI：

- HI_MPI_RGN_Create：创建区域。
- HI_MPI_RGN_Destroy：销毁区域。
- HI_MPI_RGN_GetAttr：获取区域属性
- HI_MPI_RGN_SetAttr：设置区域属性。
- HI_MPI_RGN_SetBitMap：设置区域位图。
- HI_MPI_RGN_SetAttachField：设置区域要叠加到的帧/场标识。
- HI_MPI_RGN_GetAttachField：获取区域要叠加到的帧/场标识。
- HI_MPI_RGN_AttachToChn：将区域叠加到通道上。
- HI_MPI_RGN_DetachFrmChn：将区域从通道中撤出。
- HI_MPI_RGN_SetDisplayAttr：设置区域的通道显示属性。
- HI_MPI_RGN_GetDisplayAttr：获取区域的通道显示属性。

HI_MPI_RGN_Create

【描述】

创建区域。

【语法】

```
HI_S32 HI_MPI_RGN_Create(RGN_HANDLE Handle,const RGN_ATTR_S *pstRegion);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 必须是未使用的 Handle 号 取值范围：[0, RGN_HANDLE_MAX)。	输入
pstRegion	区域属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 该句柄由用户指定，意义等同于 ID 号。
- 不支持重复创建。
- 区域属性必须合法，具体约束参见 RGN_ATTR_S。
- 区域属性指针不能为空。
- 创建 Cover、CoverEx 时，只需指定区域类型即可。其它的属性，如区域位置，层次等信息在调用 HI_MPI_RGN_AttachToChn 接口时指定。

【举例】

```
RGN_HANDLE Handle;
RGN_ATTR_S stRgnAttr;
HI_S32 s32Ret = HI_SUCCESS;
Handle = 0;
stRgnAttr.enType = OVERLAY_RGN;
stRgnAttr.unAttr.stOverlay.enPixelFmt = PIXEL_FORMAT_RGB_1555;
stRgnAttr.unAttr.stOverlay.stSize.u32Width  = 16;
stRgnAttr.unAttr.stOverlay.stSize.u32Height = 16;
stRgnAttr.unAttr.stOverlay.u32BgColor = 0x0000ffff;
s32Ret = HI_MPI_RGN_Create(Handle,&stRgnAttr);
if(s32Ret != HI_SUCCESS)
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    return s32Ret;
}
s32Ret = HI_MPI_RGN_GetAttr(Handle,&stRgnAttr);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
stRgnAttr.unAttr.stOverlay.u32BgColor = 0x0000cccc;
s32Ret = HI_MPI_RGN_SetAttr(Handle,&stRgnAttr);
if(s32Ret != HI_SUCCESS)
{
return s32Ret;
}
s32Ret = HI_MPI_RGN_Destroy(Handle);
if(s32Ret != HI_SUCCESS)
{
return s32Ret;
}
```

【相关主题】

- HI_MPI_RGN_Destroy
- HI_MPI_RGN_GetAttr
- HI_MPI_RGN_SetAttr

HI_MPI_RGN_Destroy

【描述】

销毁区域。

【语法】

```
HI_S32 HI_MPI_RGN_Destroy(RGN_HANDLE Handle);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

区域必须已创建。

【举例】

参见 HI_MPI_RGN_Create 举例。

【相关主题】

无。

HI_MPI_RGN_GetAttr

【描述】

获取区域属性。

【语法】

```
HI_S32 HI_MPI_RGN_GetAttr(RGN_HANDLE Handle,RGN_ATTR_S *pstRegion);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入
pstRegion	区域属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 区域属性指针不能为空。

【举例】

参见 HI_MPI_RGN_Create 举例。

【相关主题】

无。

HI_MPI_RGN_SetAttr

【描述】

设置区域属性。

【语法】

```
HI_S32 HI_MPI_RGN_SetAttr(RGN_HANDLE Handle,const RGN_ATTR_S *pstRegion);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0,RGN_HANDLE_MAX)。	输入
pstRegion	区域属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 只能更改区域的动态属性，不同类型区域的动态属性可参见各区域属性结构体描述

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 区域属性指针不能为空。

【举例】

参见 HI_MPI_RGN_Create 举例。

【相关主题】

无。

HI_MPI_RGN_SetBitMap

【描述】

设置区域位图，即对区域进行位图填充。

【语法】

```
HI_S32 HI_MPI_RGN_SetBitMap(RGN_HANDLE Handle,const BITMAP_S *pstBitmap);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入
pstBitmap	位图属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 支持位图的大小和区域的大小不一致。
- 位图从区域的(0,0)点开始加载。当位图比区域大时，将会自动将图像剪裁成区域大小。
- 位图的像素格式必须和区域的像素格式一致。
- 位图属性指针不能为空。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 支持多次调用。
- 此接口仅对于 Overlay 有效。

【举例】

```
RGN_HANDLE Handle;
BITMAP_S stBitmap;
HI_S32 s32Ret = HI_SUCCESS;
stBitmap.enPixelFormat = PIXEL_FORMAT_RGB_1555;
stBitmap.u32Width  = 4;
stBitmap.u32Height = 4;
stBitmap.pData = malloc(2 * stBitmap.u32Width * stBitmap.u32Height);
if(HI_NULL == stBitmap.pData)
{
    return HI_FAILURE;
}
memset(stBitmap.pData,0xcc,u32Size);

s32Ret = HI_MPI_RGN_SetBitMap(Handle,&stBitmap);
if(s32Ret != HI_SUCCESS)
{
    free(stBitmap.pData);
    return s32Ret;
}

free(stBitmap.pData);
```

【相关主题】

无。

HI_MPI_RGN_SetAttachField

【描述】

设置区域要叠加到的帧/场标识。

【语法】

```
HI_S32 HI_MPI_RGN_SetAttachField(RGN_HANDLE Handle, RGN_ATTACH_FIELD_E
enAttachField);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
enAttachField	区域要叠加到的帧、场标识。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 该接口只支持 COVEREX_RGN 和 OVERLAYEX_RGN 类型的区域。
- 按场叠加时，仍以整帧图像作为参考系来配置区域的坐标及宽高。
- 区域按场叠加功能只针对视频输入模块的隔行两场图像有效，帧图像或者叠加到其他模块的区域调用本接口无效。
- 必须在 HI_MPI_RGN_AttachToChn 之前调用本接口。
- 当区域叠加到顶场或底场时，要求区域的 Y 坐标和高度要按 4 对齐配置，叠加到帧时，2 对齐即可。

【举例】

无。

【相关主题】

无。

HI_MPI_RGN_GetAttachField

【描述】

获取区域要叠加到的帧/场标识。

【语法】

```
HI_S32 HI_MPI_RGN_GetAttachField(RGN_HANDLE Handle, RGN_ATTACH_FIELD_E
*penAttachField);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入
penAttachField	区域要叠加到的帧、场标识指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 该接口只支持 COVEREX_RGN 和 OVERLAYEX_RGN 类型的区域。

【举例】

无。

【相关主题】

无。

HI_MPI_RGN_AttachToChn

【描述】

将区域叠加到通道上。

【语法】

```
HI_MPI_RGN_AttachToChn(RGN_HANDLE Handle,const MPP_CHN_S *pstChn,const
RGN_CHN_ATTR_S *pstChnAttr);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstChn	通道结构体指针。	输入
pstChnAttr	区域通道显示属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 通道结构体指针不能为空。
- 区域通道显示属性指针不能为空。
- 支持多次叠加，但此接口不用于改变属性。
- 如果是 Overlay 区域，通道显示属性中的 QP 信息需要注意，参见结构体 OVERLAY_QP_INFO_S 描述。

【举例】

无。

【相关主题】

无。

HI_MPI_RGN_DetachFrmChn

【描述】

将区域从通道中撤出。

【语法】

```
HI_MPI_RGN_DetachFrmChn(RGN_HANDLE Handle,const MPP_CHN_S *pstChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入
pstChn	通道结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 通道结构体指针不能为空。
- 支持多次调用。

【举例】

无。

【相关主题】

无。

HI_MPI_RGN_SetDisplayAttr

【描述】

设置区域的通道显示属性。

【语法】

```
HI_S32 HI_MPI_RGN_SetDisplayAttr(RGN_HANDLE Handle,const MPP_CHN_S
*pstChn,const RGN_CHN_ATTR_S *pstChnAttr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0, RGN_HANDLE_MAX)。	输入
pstChn	通道结构体指针。	输入
pstChnAttr	区域通道显示属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 建议先获取属性，再设置。
- 通道结构体指针不能为空。
- 区域通道显示属性指针不能为空。
- 区域必须先叠加到通道上。
- 静态属性不能修改，动态属性可以被修改，详细情况参见结构体RGN_CHN_ATTR_S。

【举例】

```
HI_S32 s32Ret = HI_SUCCESS;
RGN_HANDLE Handle;
MPP_CHN_S stChn;
 RGN_ATTR_S stRgnAttr;
RGN_CHN_ATTR_S stChnAttr;

Handle = 0;

stChn.enModId = HI_ID_GROUP;
stChn.s32DevId = 0;
stChn.s32ChnId = 0;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
s32Ret =  HI_MPI_RGN_GetDisplayAttr(Handle,&stChn,&stChnAttr);
if(s32Ret != HI_SUCCESS)
{
    NOT_PASS(s32Ret);
    goto END;
}

stChnAttr.bShow = HI_TRUE;
stChnAttr.enType = OVERLAY_RGN;
stChnAttr.unChnAttr.stOverlayChn.stPoint.s32X = 4;
stChnAttr.unChnAttr.stOverlayChn.stPoint.s32Y = 4;
stChnAttr.unChnAttr.stOverlayChn.u32BgAlpha = 128;
stChnAttr.unChnAttr.stOverlayChn.u32FgAlpha = 128;
stChnAttr.unChnAttr.stOverlayChn.u32Layer = 0;

stChnAttr.unChnAttr.stOverlayChn.stQpInfo.bAbsQp = HI_FALSE;
stChnAttr.unChnAttr.stOverlayChn.stQpInfo.s32Qp  = 0;

s32Ret = HI_MPI_RGN_SetDisplayAttr(Handle,&stChn,&stChnAttr);
if(s32Ret != HI_SUCCESS)
{
    return s32Ret;
}
```

【相关主题】

无。

HI_MPI_RGN_GetDisplayAttr

【描述】

获取区域的通道显示属性。

【语法】

```
HI_S32 HI_MPI_RGN_GetDisplayAttr(RGN_HANDLE Handle,const MPP_CHN_S
*pstChn,RGN_CHN_ATTR_S *pstChnAttr);
```

【参数】

参数名称	描述	输入/输出
Handle	区域句柄号。 取值范围：[0,RGN_HANDLE_MAX)。	输入
pstChn	通道结构体指针	输入
pstChnAttr	区域通道显示属性指针	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_region.h、mpi_region.h
- 库文件：libmpi.a

【注意】

- 区域必须已创建。
- 通道结构体指针不能为空。
- 区域通道显示属性指针不能为空。

【举例】

参见 HI_MPI_RGN_SetDisplayAttr 举例。

【相关主题】

无。

8.4 数据类型

视频前处理相关数据类型、数据结构定义如下：

- RGN_HANDLE_MAX：定义区域的最大句柄数。
- OVERLAY_MAX_NUM：定义一个 GROUP 通道最多可显示的叠加区域的个数
- COVER_MAX_NUM：定义一个通道最大遮挡区域的个数。
- COVEREX_MAX_NUM：。定义一个通道最大扩展遮挡区域的个数。
- OVERLAYEX_MAX_NUM：定义一个通道的最大扩展叠加区域的个数。。
- RGN_HANDLE：定义区域句柄。
- RGN_TYPE_E：定义区域类型。
- INVERT_COLOR_MODE_E：定义 OSD 反色触发模式。
- OVERLAY_QP_INFO_S：定义叠加区域 QP 属性结构体。
- OVERLAY_INVERT_COLOR_S：定义 OSD 反色相关属性。
- RGN_ATTACH_FIELD_E：定义区域要叠加到的帧场类型。
- OVERLAY_ATTR_S：定义叠加区域属性结构体。
- OVERLAY_CHN_ATTR_S：定义叠加区域通道显示属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- COVER_CHN_ATTR_S：定义遮挡区域的通道显示属性。
- COVEREX_CHN_ATTR_S：定义扩展遮挡区域的通道显示属性。
- OVERLAYEX_ATTR_S：定义扩展叠加区域属性结构体。
- OVERLAYEX_CHN_ATTR_S：定义扩展叠加区域的通道显示属性。
- RGN_ATTR_U：定义区域属性联合体。
- RGN_CHN_ATTR_U：定义区域通道显示属性联合体。
- RGN_ATTR_S：定义区域类型结构体。
- RGN_CHN_ATTR_S：定义区域通道显示属性结构体。

RGN_HANDLE_MAX

【说明】

定义区域的最大句柄。

【定义】

```
#define RGN_HANDLE_MAX  1024
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

OVERLAY_MAX_NUM

【说明】

定义一个 GROUP 通道的最大叠加区域的个数。

【定义】

```
#define OVERLAY_MAX_NUM        8
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

COVER_MAX_NUM

【说明】

定义一个通道最大遮挡区域的个数。

【定义】

```
#define COVER_MAX_NUM        4
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

COVEREX_MAX_NUM

【说明】

定义一个通道最大扩展遮挡区域的个数。

【定义】

```
#define COVEREX_MAX_NUM        16
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

OVERLAYEX_MAX_NUM

【说明】

定义一个通道的最大扩展叠加区域的个数。

【定义】

For Hi3531/Hi3532/Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C:

```
#define OVERLAYEX_MAX_NUM        16
```

For Hi3518/Hi3516C:

```
#define OVERLAYEX_MAX_NUM        0
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

RGN_HANDLE

【说明】

定义区域句柄。

【定义】

```
typedef HI_U32 RGN_HANDLE;
```

【成员】

成员名称	描述
RGN_HANDLE	区域句柄。

【注意事项】

无。

【相关数据类型及接口】

无

RGN_TYPE_E

【说明】

定义区域类型。

【定义】

```
typedef enum hiRGN_TYPE_E
{
    OVERLAY_RGN = 0,
    COVER_RGN,
    COVEREX_RGN,
    OVERLAYEX_RGN,
    RGN_BUTT
} RGN_TYPE_E;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
OVERLAY_RGN	GROUP 通道视频叠加区域。
COVER_RGN	VI 通道视频遮挡区域。
COVEREX_RGN	扩展视频遮挡区域。
OVERLAYEX_RGN	扩展视频叠加区域。

【注意事项】

无。

【相关数据类型及接口】

无

INVERT_COLOR_MODE_E

【说明】

定义 OSD 反色触发模式。

【定义】

```
typedef enum hiINVERT_COLOR_MODE_E
{
    LESSTHAN_LUM_THRESH = 0,
    MORETHAN_LUM_THRESH,
    INVERT_COLOR_BUTT
}INVERT_COLOR_MODE_E;
```

【成员】

成员名称	描述
LESSTHAN_LUM_THRESH	当视频背景亮度均值小于亮度阈值时触发反色。
MORETHAN_LUM_THRESH	当视频背景亮度均值大于亮度阈值时触发反色。

【注意事项】

当反色开关打开时，此设置有效。

【相关数据类型及接口】

无

OVERLAY_QP_INFO_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义叠加区域 QP 属性结构体。

【定义】

```
typedef struct hiOVERLAY_QP_INFO_S
{
    HI_BOOL bAbsQp;
    HI_S32 s32Qp;
}OVERLAY_QP_INFO_S;
```

【成员】

成员名称	描述
bAbsQp	是否绝对 QP
s32Qp	QP 值。 当 bAbsQp 为 HI_TRUE 时 取值范围：[0, 51]。 当 bAbsQp 为 HI_FALSE 时 取值范围：[-51, 51]。

【注意事项】

仅用于编码中的码率控制，如果对此选项不了解，建议全部设置为 0。

【相关数据类型及接口】

无。

OVERLAY_INVERT_COLOR_S

【说明】

定义 OSD 反色相关属性。

【定义】

```
typedef struct hiOVERLAY_INVERT_COLOR_S
{
    SIZE_S stInvColArea;
    HI_U32 u32LumThresh;
    INVERT_COLOR_MODE_E enChgMod;
    HI_BOOL bInvColEn;
}OVERLAY_INVERT_COLOR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stInvColArea	单元反色区域，反色处理的基本单元。 取值范围： 高：[16, 64]，需 16 对齐。 宽：[16, 64]，需 16 对齐。
u32LumThresh	亮度阈值。 取值范围：[0, 255]。
enChgMod	OSD 反色触发模式。
bInvColEn	OSD 反色开关。 HI_TRUE：使能 OSD 反色； HI_FALSE：不使能 OSD 反色。

【注意事项】

- 单元反色区域是反色处理的基本单元，即会被作为一个整体进行反色处理，所以用户应根据自己实际的 OSD 的大小，设置相应的单元反色区域。
- 反色区域，亮度阈值，反色触发模式属于公共属性，对使能反色功能的 OSD 区域均有效，若用户每次配置得不一样，以最后一次配置为准。
- 反色开关控制是否使能 OSD 区域反色，每个 OSD 区域各有一个反色开关，用于控制是否使能相应区域的反色功能。
- 当某个 OSD 区域使能反色功能时，OSD 区域起始位置和宽高需 16 像素对齐，且建议配置为反色基本单元的整数倍，否则可能会出现一个反色单元里，只有部分反色的现象。

【相关数据类型及接口】

无。

RGN_ATTACH_FIELD_E

【说明】

定义区域要叠加到的帧场类型。

【定义】

```
typedef enum hiRGN_ATTACH_FIELD_E
{
    RGN_ATTACH_FIELD_FRAME = 0,           /* whole frame */
    RGN_ATTACH_FIELD_TOP,                 /* top field */
    RGN_ATTACH_FIELD_BOTTOM,              /* bottom field */

    RGN_ATTACH_FIELD_BUTT
} RGN_ATTACH_FIELD_E;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
RGN_ATTACH_FIELD_FRAME	将区域叠加到整帧图像上。
RGN_ATTACH_FIELD_TOP	将区域叠加到顶场图像上。
RGN_ATTACH_FIELD_BOTTOM	将区域叠加到底场图像上。

【注意事项】

当图像是隔行两场图像时，RGN_ATTACH_FIELD_TOP 和 RGN_ATTACH_FIELD_BOTTOM 才有效。

【相关数据类型及接口】

无。

OVERLAY_ATTR_S

【说明】

定义 GROUP 通道叠加区域属性结构体。

【定义】

```
typedef struct hiOVERLAY_COMM_ATTR_S
{
    PIXEL_FORMAT_E enPixelFmt;
    HI_U32 u32BgColor;
    SIZE_S stSize;
}OVERLAY_ATTR_S;
```

【成员】

成员名称	描述
enPixelFmt	像素格式。 取值范围：PIXEL_FORMAT_RGB_1555 或 PIXEL_FORMAT_RGB_4444。 静态属性。
u32BgColor	区域背景色。 取值范围：无 动态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stSize	区域的高宽。 取值范围： 当未使能 OSD 反色时： 宽度：[4,4096]，要求以 2 对齐。 高度：[4,4096]，要求以 2 对齐。 当使能 OSD 反色时： 宽度：[16,4096]，要求以 16 对齐。 高度：[16,4096]，要求以 16 对齐。 静态属性。

【注意事项】

- 在设置区域宽高时，注意是否使能 OSD 反色功能。
- 当某个 OSD 区域使能反色功能时，OSD 区域起始位置和宽高需 16 像素对齐，且建议配置为反色基本单元的整数倍，否则可能会出现一个反色单元里，只有部分反色的现象。

【相关数据类型及接口】

无。

OVERLAY_CHN_ATTR_S

【说明】

定义 GROUP 通道叠加区域的通道显示属性。

【定义】

```
typedef struct hiOVERLAY_CHN_ATTR_S
{
    POINT_S stPoint;
    HI_U32 u32FgAlpha;
    HI_U32 u32BgAlpha;
    HI_U32 u32Layer;
    OVERLAY_QP_INFO_S stQpInfo;
    OVERLAY_INVERT_COLOR_S stInvertColor;
}OVERLAY_CHN_ATTR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

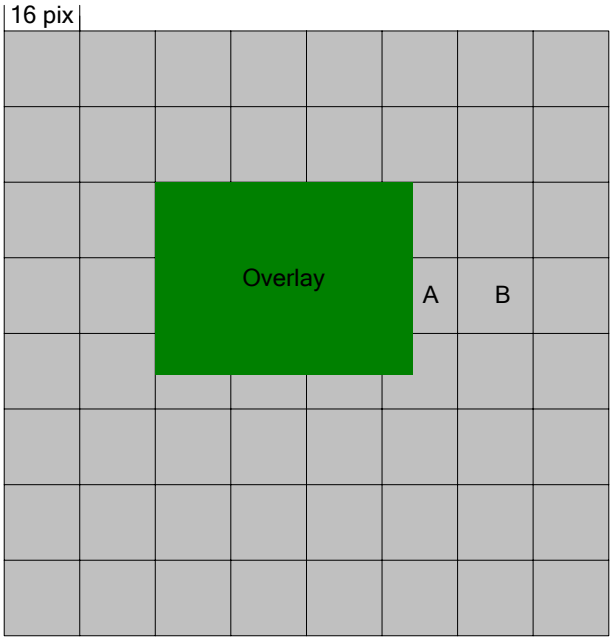
成员名称	描述
stPoint	区域位置。 取值范围： 当未使能 OSD 反色时： 水平位置 X：[0,4096]，要求以 4 对齐。 垂直位置 Y：[0,4096]，要求以 4 对齐。 当使能 OSD 反色时： 水平位置 X：[0,4096]，要求以 16 对齐。 垂直位置 Y：[0,4096]，要求以 16 对齐。 动态属性。
u32FgAlpha	Alpha 位为 1 的像素点的透明度。也称前景 Alpha。 取值范围：[0, 128]。取值越小，越透明。 动态属性。
u32BgAlpha	Alpha 位为 0 的像素点的透明度。也称背景 Alpha。 取值范围：[0, 128]。取值越小，越透明。 动态属性。
u32Layer	区域层次。 取值范围：[0, 7]。值越大，层次越高。 动态属性。
stQpInfo	此区域编码时使用的 QP 值。 取值范围：参见结构体 OVERLAY_QP_INFO_S。 动态属性。
stInvertColor	区域反色配置信息。 取值范围：参加结构体 OVERLAY_INVERT_COLOR_S。 动态属性。

【注意事项】

建议用户把 Overlay 的起始位置和宽高限定为 16 对齐。如图 8-2 所示，由于该 Overlay 的宽高不是 16 对齐，就会出现一个宏块有两个 QP 的情况，比如宏块 A 会有 Region 块的 Qp 以及编码计算出来的 QP。但是一个宏块只能有一个 QP，编码器会选择 Region 的 QP，如果宏块 A 的 QP（即 Region 的 QP）和它的相邻宏块 B 的 QP 相差较大，会导致 Region 区域周边出现图像质量不一致的现象。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图8-2 Overlay 的宽高非 16 对齐示意图



- 区域内存信息为 PIXEL_FORMAT_RGB_1555 格式时，芯片将会扩展 Alpha 值。当 Alpha 位为 1 时，芯片使用 u32FgAlpha 进行透明度叠加；当 Alpha 位为 0 时，芯片使用 u32BgAlpha 进行透明度叠加。
- 0 表示全透明；128 表示不透明。
- 在设置区域起始位置时，注意是否使能了 OSD 反色功能。
- 当某个 OSD 区域使能反色功能时，OSD 区域起始位置和宽高需 16 像素对齐，且建议配置为反色基本单元的整数倍，否则可能会出现一个反色单元里，只有部分反色的现象。

【相关数据类型及接口】

无。

COVER_CHN_ATTR_S

【说明】

定义遮挡区域的通道显示属性。

【定义】

```
typedef struct hiCOVER_CHN_ATTR_S
{
    RECT_S stRect;
    HI_U32 u32Color;
    HI_U32 u32Layer;
}COVER_CHN_ATTR_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
stRect	区域位置，宽高。 位置取值范围： 水平位置 X：[0, 4094)，要求以 2 对齐。 垂直位置 Y：[0, 4636]，要求以 4 对齐。 宽高取值范围： 宽度：[2, 4094)，要求以 2 对齐。 高度：[4, 4094)，要求以 4 对齐。 动态属性。
u32Color	区域颜色。 取值范围：无。 动态属性。
u32Layer	区域层次。 取值范围：[0, 3]。值越大，层次越高。 动态属性。

【注意事项】

区域颜色数据格式为 PIXEL_FORMAT_RGB_888。

【相关数据类型及接口】

无。

COVEREX_CHN_ATTR_S

【说明】

定义扩展遮挡区域的通道显示属性。

【定义】

```
typedef struct hiCOVEREX_CHN_ATTR_S
{
    RECT_S stRect;
    HI_U32 u32Color;
    HI_U32 u32Layer;
}COVEREX_CHN_ATTR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stRect	区域位置，宽高。 位置取值范围： 水平位置 X：[0, 4096]，要求以 2 对齐。 垂直位置 Y：[0, 4096]，要求以 2 对齐。 宽高取值范围： 宽度：[2, 4096]，要求以 2 对齐。 高度：[2, 4096]，要求以 2 对齐。 动态属性。
u32Color	区域颜色。 取值范围：无。 动态属性。
u32Layer	区域层次。 取值范围：[0, 15]。值越大，层次越高。 动态属性。

【注意事项】

- 区域颜色数据格式为 PIXEL_FORMAT_RGB_888。
- 当 VI 通道是混合采集状态时，以 VI 通道的 Crop 大小为参考系配置 stRect，SDK 内部会自动根据当前的图像大小进行遮挡块适配，使最终看到的遮挡块在同一个视域，此时，建议用户按照 4 对齐配置参数，以保证适配后的参数满足 2 对齐的要求；当 VI 通道不是混合采集状态时，则以 VI 当前图像的大小为参考系配置 stRect。
- CVBS 单画面预览时，如果要打 CoverEx，此时建议关闭 VI 当前通道的混合采集功能，改为 D1 实时采集，否则 CoverEx 在预览时会出现因缩放引起的微微抖动现象，多画面预览时则基本看不出抖动现象。
- VI 混合采集时打 coverEx，图像经过 VPSS 处理后送 VENC 进行 cif 编码，按原始码流图像大小查看码流，肉眼看不出明显的抖动现象，但将码流放大来看，可看出微微的抖动现象，请用户谨慎使用混合采集下打 coverEx 功能。

【相关数据类型及接口】

无。

OVERLAYEX_ATTR_S

【说明】

定义扩展叠加区域属性结构体。

【定义】

```
typedef struct hiOVERLAYEX_COMM_ATTR_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    PIXEL_FORMAT_E enPixelFmt;
    HI_U32 u32BgColor;
    SIZE_S stSize;
}OVERLAYEX_ATTR_S;
```

【成员】

成员名称	描述
enPixelFmt	像素格式。 取值范围：PIXEL_FORMAT_RGB_1555。 静态属性。
u32BgColor	区域背景色。 取值范围：无。 动态属性。
stSize	区域的高宽。 取值范围： 宽度：[16, 1920]，要求以 2 对齐。 高度：[16, 1080]，要求以 2 对齐。 宽度 x 高度 ≤ 480000。 静态属性。

【注意事项】

无。

【相关数据类型及接口】

无。

OVERLAYEX_CHN_ATTR_S

【说明】

定义扩展叠加区域的通道显示属性。

【定义】

```
typedef struct hiOVERLAYEX_CHN_ATTR_S
{
    POINT_S stPoint;
    HI_U32 u32FgAlpha;
    HI_U32 u32BgAlpha;
    HI_U32 u32Layer;
}OVERLAYEX_CHN_ATTR_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
stPoint	区域位置。 取值范围： 水平位置 X：[0,4096]，要求以 2 对齐。 垂直位置 Y：[0,4636]，要求以 2 对齐。
u32FgAlpha	Alpha 位为 1 的像素点的透明度。也称前景 Alpha。 取值范围：[0, 255]。取值越小，越透明。 动态属性。
u32BgAlpha	Alpha 位为 0 的像素点的透明度。也称背景 Alpha。 取值范围：[0, 255]。取值越小，越透明。 动态属性。
u32Layer	区域层次。 取值范围：[0, 15]。值越大，层次越高。 动态属性。

【注意事项】

- 区域内存信息为 PIXEL_FORMAT_RGB_1555 格式时，芯片将会扩展 Alpha 值。当 Alpha 位为 1 时，芯片使用 u32FgAlpha 进行透明度叠加；当 Alpha 位为 0 时，芯片使用 u32BgAlpha 进行透明度叠加。
- 0 表示全透明；255 表示不透明。

【相关数据类型及接口】

无。

RGN_ATTR_U

【说明】

定义区域属性联合体。

【定义】

```
typedef union hiRGN_ATTR_U
{
    OVERLAY_ATTR_S      stOverlay;
    OVERLAYEX_ATTR_S    stOverlayEx;
} RGN_ATTR_U;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
stOverlay	GROUP 通道叠加区域属性。
stOverlayEx	扩展叠加区域属性。

【注意事项】

由于 Cover 和 CoverEx 没有区域属性，所以此联合体不包含相关成员。

【相关数据类型及接口】

无。

RGN_CHN_ATTR_U

【说明】

定义区域通道显示属性联合体。

【定义】

```
typedef union hiRGN_CHN_ATTR_U
{
    OVERLAY_CHN_ATTR_S      stOverlayChn;
    COVER_CHN_ATTR_S      stcoverChn;
    COVEREX_CHN_ATTR_S    stcoverExChn;
    OVERLAYEX_CHN_ATTR_S    stOverlayEXChn;
} RGN_CHN_ATTR_U;
```

【成员】

成员名称	描述
stOverlayChn	GROUP 通道叠加区域通道显示属性。
stcoverChn	遮挡区域通道显示属性。
stcoverExChn	扩展遮挡区域通道显示属性。
stOverlayExChn	扩展叠加区域通道显示属性。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

RGN_ATTR_S

【说明】

定义区域属性结构体。

【定义】

```
typedef struct hiRGN_ATTR_S
{
    RGN_TYPE_E enType;
    RGN_ATTR_U unAttr;

} RGN_ATTR_S;
```

【成员】

成员名称	描述
enType	区域类型。
unAttr	区域属性。

【注意事项】

无。

【相关数据类型及接口】

无。

RGN_CHN_ATTR_S

【说明】

定义区域通道显示属性结构体。

【定义】

```
typedef struct hiRGN_CHN_ATTR_S
{
    HI_BOOL        bShow;
    RGN_TYPE_E     enType;
    RGN_CHN_ATTR_U unChnAttr;
} RGN_CHN_ATTR_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bShow	区域是否显示。 取值范围：HI_TRUE 或者 HI_FALSE。 动态属性。
enType	区域类型。 静态属性。
unChnAttr	区域通道显示属性。

【注意事项】

无。

【相关数据类型及接口】

无。

8.5 错误码

区域管理 API 错误码如表 8-2 所示。

表8-2 区域管理 API 错误码

错误代码	宏定义	描述
0xA0128001	HI_ERR_RGN_INVALID_DEVID	设备 ID 超出合法范围
0xA0128002	HI_ERR_RGN_INVALID_CHNID	通道组号错误或无效区域句柄
0xA0128003	HI_ERR_RGN_ILLEGAL_PARAM	参数超出合法范围
0xA0128004	HI_ERR_RGN_EXIST	重复创建已存在的设备、通道或资源
0xA0128005	HI_ERR_RGN_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源
0xA0128006	HI_ERR_RGN_NULL_PTR	函数参数中有空指针
0xA0128007	HI_ERR_RGN_NOT_CONFIG	模块没有配置
0xA0128008	HI_ERR_RGN_NOT_SUPPORT	不支持的参数或者功能
0xA0128009	HI_ERR_RGN_NOT_PERM	该操作不允许，如试图修改静态配置参数

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA012800C	HI_ERR_RGN_NOMEM	分配内存失败，如系统内存不足
0xA012800D	HI_ERR_RGN_NOBUF	分配缓存失败，如申请的数据缓冲区太大
0xA012800E	HI_ERR_RGN_BUF_EMPTY	缓冲区中无数据
0xA012800F	HI_ERR_RGN_BUF_FULL	缓冲区中数据满
0xA0128010	HI_ERR_RGN_NOTREADY	系统没有初始化或没有加载相应模块
0xA0128011	HI_ERR_RGN_BADADDR	地址非法
0xA0128012	HI_ERR_RGN_BUSY	系统忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

9 音频.....9-1
9.1 概述.....9-1
9.2 功能描述.....9-1
9.2.1 音频输入和音频输出.....9-1
9.2.2 音频编码和解码.....9-8
9.2.3 芯片差异介绍.....9-10
9.2.4 内置 Audio Codec.....9-13
9.3 API 参考.....9-16
9.3.1 音频输入.....9-16
9.3.2 音频输出.....9-43
9.3.3 音频编码.....9-63
9.3.4 音频解码.....9-72
9.3.5 内置 Audio Codec.....9-82
9.4 数据类型.....9-131
9.4.1 音频输入输出.....9-131
9.4.2 音频编码.....9-149
9.4.3 音频解码.....9-153
9.4.4 内置 Audio Codec.....9-159
9.5 错误码.....9-161

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 9-1 SIO 最大支持 AI、AO 通道数 9-3
表 9-2 AIO 最大支持 AI、AO 通道数 9-4
表 9-3 重采样类型支持的采样率 9-6
表 9-4 音频编解码协议说明 9-8
表 9-5 海思语音帧结构 9-10
表 9-6 芯片音频接口差异 9-10
表 9-7 音频输入输出设备号 9-13
表 9-8 ioctl 函数的 3 个参数 9-14
表 9-9 音频输入 API 错误码 9-161
表 9-10 音频输出 API 错误码 9-162
表 9-11 音频编码 API 错误码 9-162
表 9-12 音频解码 API 错误码 9-163

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 9-1 录音和播放示意图.....9-2
图 9-2 I²S 2/4/8/16 路接收.....9-3
图 9-3 PCM 时序发送示意图.....9-4
图 9-4 回声抵消示意图.....9-5
图 9-5 Hi3531/Hi3532 SIO 与 AI、AO 关系示意图.....9-11
图 9-6 Hi3521 /Hi3520ASIO 与 AI、AO 关系示意图.....9-12
图 9-7 Hi3518/Hi3516C SIO 与 AI、AO 关系示意图.....9-12
图 9-8 Hi3520D/Hi3515A/Hi3515C AIO 中 AIP/AOP 与 AI、AO 关系示意图.....9-13

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

9 音频

9.1 概述

AUDIO 模块包括音频输入、音频输出、音频编码、音频解码四个子模块。音频输入和输出模块通过对 Hi35xx 芯片音频接口的控制实现音频输入输出功能。音频编码和解码模块提供对 G711、G726、ADPCM 格式的音频编解码功能，并支持录制和播放 LPCM 格式的原始音频文件。

9.2 功能描述

9.2.1 音频输入和音频输出

9.2.1.1 音频接口和 AI、AO 设备

音频输入输出接口分为 SIO（Sonic Input/Output）接口和 AIO（Audio Input/Output）接口两种类型，均用于和 Audio Codec 对接，完成声音的录制和播放。其中 SIO 接口既可以支持音频输入，也可以支持音频输出；AIO 接口又分为两种类型：只支持输入或只支持输出，当为输入类型时，又称为 AIP，当为输出类型时，又称为 AOP。

软件中负责抽象音频接口输入功能的单元，称之为 AI 设备；负责抽象音频接口输出功能的单元，称之为 AO 设备。

对每个输入输出接口，软件根据该接口支持的功能，分别与 AI 设备和 AO 设备建立映射关系。例如：SIO0 支持音频输入输出，则 SIO0 的输入单元映射为 AiDev0，输出单元映射为 AoDev0；AIP1 只支持音频输入，则 AIP1 映射为 AiDev1；AOP2 只支持音频输出，则 AOP2 映射为 AoDev2。

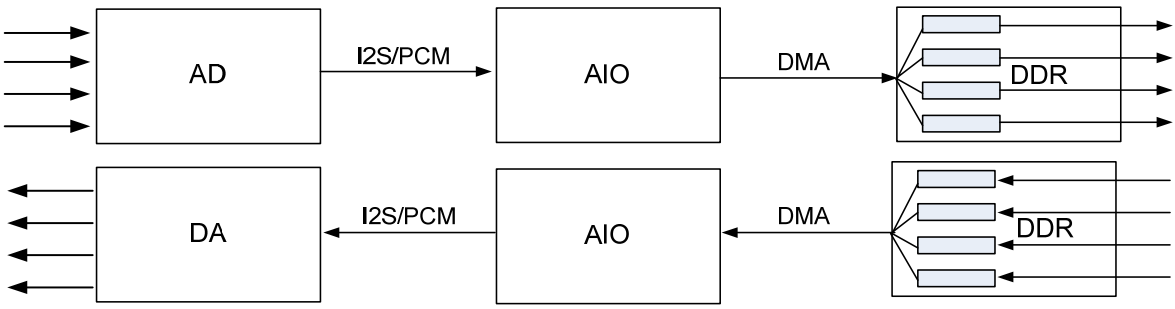
9.2.1.2 录音和播放原理

原始音频信号以模拟信号的形式给出后，通过 Audio Codec，按一定采样率和采样精度转换为数字信号。Audio Codec 以 I2S 时序或 PCM 时序的方式，将数字信号传输给 AI 设备。芯片利用 DMA 将 AI 设备中的音频数据搬移到内存中，完成录音操作。

播放和录音是基于同样的原理。芯片利用 DMA 将内存中的数据传输到 AO 设备。AO 设备通过 I2S 时序或 PCM 时序向 Audio CODEC 发送数据。Audio Codec 完成数字信号到模拟信号的转换过程，并输出模拟信号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图9-1 录音和播放示意图



9.2.1.3 音频接口时序和 AI、AO 通道排列

音频接口时序

音频接口支持标准的 I²S 接口时序模式和 PCM 接口时序模式，并提供灵活的配置以支持与多种 Audio Codec 对接。详细的时序支持情况请参考《Hi35xx 编码处理器用户指南》。

为实现成功对接，需要对 I²S 或者 PCM 协议以及对接的 Audio Codec 时序支持情况有足够了解，这里只简单介绍下 I²S 及 PCM 接口时序的几个特性：

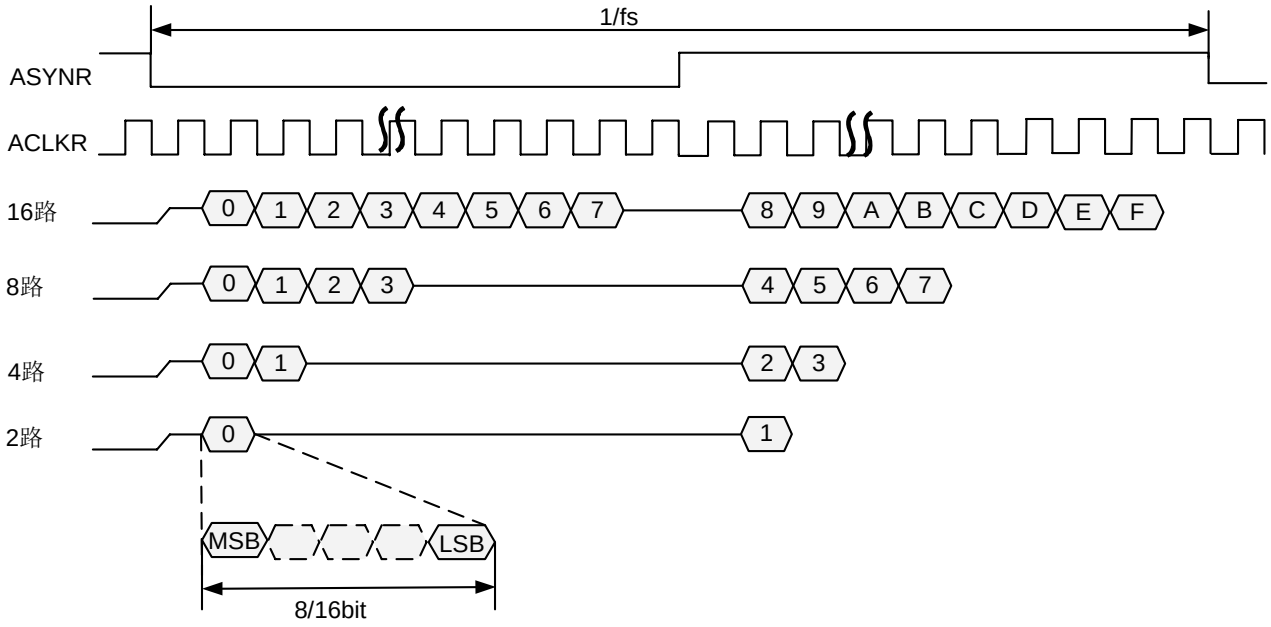
- 按照标准 I²S 或 PCM 协议，总是先传送最高有效位 MSB，后传送最低有效位 LSB，即按照从高位到低位的顺序传输串行数据。
- AI 设备支持扩展的多路接收的 I²S 及 PCM 接口时序，对接 CODEC 的时序模式选择、同步时钟、采样位宽等配置必须与 AI 设备的配置保持一致，否则可能采集不到正确的数据。
- AI/AO 设备支持主模式和从模式，主模式即 AI/AO 设备提供时钟，从模式即 Audio Codec 提供时钟。主模式时对于使用 SIO 接口的芯片，时钟供输入和输出共同使用；对于使用 AIO 接口的芯片，如果 AI 设备与 AO 设备没有同时对接同一个 CODEC，则没有此限制。而从模式时的输入输出可以分别由外围 Audio Codec 提供时钟。
- 由于时序的问题，在 AI/AO 设备从模式下，建议用户先配置好对接的 CODEC，再配置 AI 或 AO 设备；而在 AI/AO 设备主模式下，建议用户先配置好 AI 或 AO 设备，再配置对接的 CODEC。
- AI/AO 设备选择主模式时，有些 AI/AO 设备只提供用于时序同步的帧同步时钟和位流时钟，不提供 MCLK，这时如果 Audio Codec 使用外接的晶振作为工作时钟，这样可能导致声音失真，因此推荐使用从模式或者使用位流时钟产生 Codec 内部工作主时钟。
- 当 AI/AO 设备为主模式时，对于向外提供了 MCLK 的 AI/AO 设备，MCLK 的设定为：
  - 采样率为 48k/24k/12k 时提供 12.288MHz 的主时钟
  - 采样率为 32k/16k/8k（32k 采样位宽不是 256bits，8k 要求采样位宽不是 16bits）时，提供 12.288MHz 的主时钟
  - 采样率为 32k/16k/8k（32k 采样位宽为 256bits 或 8k 采样位宽为 16bits）时，提供 8.192MHz 的主时钟

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 采样率为 44.1k/22.05k/11.025 时提供 11.2896MHz 的主时钟

- AI/AO 设备支持标准 PCM 模式和自定义 PCM 模式。

图9-2 I²S 2/4/8/16 路接收



AI、AO 通道

AI 和 AO 通道是软件层次的概念，下面通过举例的方式来说明通道的概念。例如当 AI 设备使用多路复用的 I²S 接收模式时，标准的 I²S 协议只有左右声道这个概念，AI 设备最大支持左右声道各接收 128bit 音频数据。此时假设 Codec 具有复用功能，可以将 16 路 16bit 采样精度的音频数据复用为 2 路 128bit 的 I²S 左右声道数据，那么 AI 设备可以解析这 16 路音频数据，并称这 16 路音频数据为 16 个音频通道。

SIO 在不同协议时，支持的 AI、AO 多路复用模式有差异，如表 9-1 所示。

AIO 在不同协议时，支持的 AI、AO 多路复用模式有差异，如表 9-2 所示。

表9-1 SIO 最大支持 AI、AO 通道数

SIO 多路复用	最大支持	通道举例
I²S 时序接收	左右声道各 128bit	16bit16chn、8bit16chn 等
PCM 时序接收	单声道 256bit	16bit16chn、8bit16chn 等
I²S 时序发送	左右声道各 32bit	16bit4chn、8bit8chn 等
PCM 时序发送	单声道 8bit~128bit	16bit8chn 等

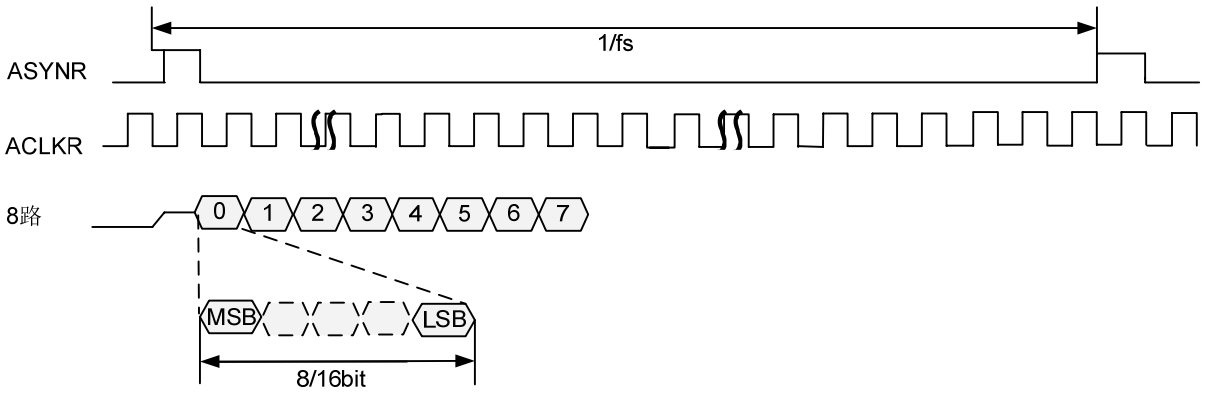
海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表9-2 AIO 最大支持 AI、AO 通道数

AIO 多路复用	最大支持	通道举例
I^2S 时序接收	左右声道各 128bit	16bit16chn、8bit16chn 等
PCM 时序接收	单声道 256bit	16bit16chn、8bit16chn 等
I^2S 时序发送	左右声道各 128bit	16bit16chn、8bit16chn 等
PCM 时序发送	单声道 256bit	16bit16chn、8bit16chn 等

AI、AO 可以在 AI/AO 设备最大支持的比特范围内，按采样精度拆分 AI 和 AO 通道，并按照时序上的顺序，依次视为 AiChn0、AiChn1 等或 AoChn0、AoChn1 等。只有 AI/AO 设备配置的 I^2S 或 PCM 时序与外部 Codec 配置的时序一致时，才能接收或传送正确的音频数据，将 AI/AO 设备的时序配置和 Codec 的时序配置调为一致的过程称作对接。例如配置 AO 为标准 PCM 从模式，8k 采样率，16bit 采样精度，4 通道；配置外部 Codec 提供的帧同步时钟为 8k，位流时钟为 8kx64bit，每个采样点为 64bit（Codec 支持将四个 16bit 的采样点复用为一个采样点）；PCM 时序上的通道排列如图 9-3 所示。

图9-3 PCM 时序发送示意图



音频 AI 和 AO 支持的最大通道数为 16 通道(其中 Hi3518/Hi3516C 芯片受内置 codec 限制，只支持 2 通道)，且配置 AI 和 AO 设备时需要将通道配置为偶数。多通道情况下，AI、AO 视通道排列中相对应的两通道为立体声输入输出，例如图 9-3 中，通道 0 和 4、1 和 5、2 和 6、3 和 7 视为立体声的左右声道，即共有四路 16bit 采样精度的立体声输出。这时只应对左声道 0~3 进行操作。

对于使用 SIO 接口的芯片，同一 SIO 对应的 AI 和 AO 配置的工作模式必须一致，例如 AI 和 AO 均配置为 I2S 时序的从模式。当 SIO 工作在主模式时，AI 和 AO 的通道数与采样精度的乘积也必须相等，这样才能保证由 SIO 发送给 Audio Codec 的同步时钟，对 AI 和 AO 是一致的，例如 AI 是 16bit 4 通道，AO 可以是 8bit 8 通道。当 SIO 工作在从模式时，AI 和 AO 的通道数与采样精度的乘积可以不一致，此时可以由 Audio Codec 发送不同的同步时钟给 AI 和 AO。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

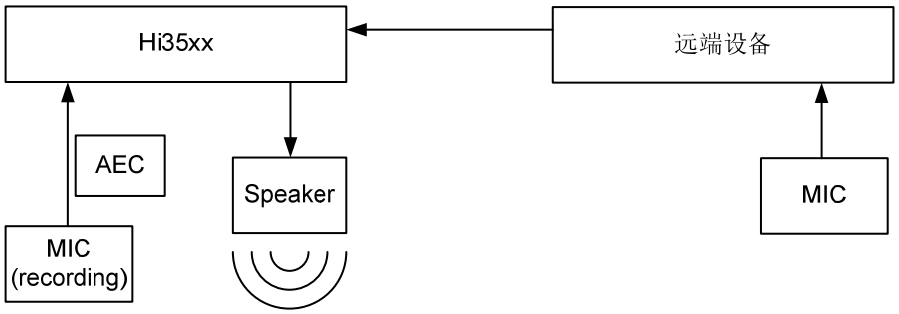
对于使用 AIO 接口的芯片，AIO 接口中的 AI 设备与 AO 设备是相互独立的，如果它们没有对接同一个的 CODEC，则没有此限制。Hi35xx SDK 支持通过 SYS 模块绑定接口，建立两个 AI、AO 通道间的绑定关系，实现音频数据的实时播放。

9.2.1.4 回声抵消、重采样和噪声抑制

回声抵消

Hi35xx SDK 支持对采集的语音数据进行回声抵消处理。远端语音数据在 AO 设备上播放，此时在本地通过 MIC 采集语音数据，它支持消除录制的语音数据中的 AO 设备播放的声音（回声）。

图9-4 回声抵消示意图



成功启用回声抵消需要具备一定条件：单声道模式，采样率为 8kHz/11kHz/16kHZ，采样精度为 8bit/16bit，当采样率为 8kHZ 时，支持 80 和 160 两种帧长，采样率为 11kHZ 和 16kHZ 时，只支持 160 帧长，且 MIC 采集语音数据的 AI 和远程语音播放的 AO 配置帧长必须相同。以上条件 AI 和 AO 都必须满足。

注意

- 当 AI-AENC 的数据传输方式为系统绑定方式时，使能 AI 的回声抵消功能后，在 AENC 中先进行回声抵消处理，再进行编码。
- 当 AI-AENC/AO 的数据传输方式为非系统绑定方式时，使能 AI 的回声抵消功能后，用户可以通过 MPI 接口获取 AI 音频帧，在 AI 中进行回声抵消的处理，然后发送给 AENC/AO，以建立 AI-AENC 或 AI-AO 的数据传输，此时回声抵消功能在 AI 中生效。

重采样

Hi35xx 的音频输入和音频输出模块支持对音频数据实施重采样。如果启用 AI 重采样功能，则在 HI_MPI_AI_GetFrame 获取数据返回前，内部将会先执行重采样处理，再返回处理后的数据。如果启用了 AO 重采样功能，则音频数据在发送给 AO 之前，内部先执行重采样处理，处理完成后再发送给 AO 通道进行播放。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

音频重采样支持 2 到 1 倍、4 到 1 倍、6 到 1 倍、1 到 2 倍、1 到 4 倍、1 到 6 倍的重采样类型。在选择不同的重采样类型时，支持的采样率有差异，如表 9-3 所示。

表9-3 重采样类型支持的采样率

重采样倍数	采样率
1x2	8k、11.025k、16k、22.05k、24k
1x4	8k、11.025k、12k
1x6	8k
2x1	48k、44.1k、32k、16k
4x1	48k、44.1k、32k
6x1	48k

重采样仅支持处理单声道，且不支持同时处理回声抵消。

注意

- **当 AI-AENC 或 AI-AO 的数据传输方式为系统绑定方式时，AI 或 AO 的重采样无效。用户可以通过 MPI 接口获取 AI 音频帧，并发送给 AENC/AO，以建立 AI-AENC 或 AI-AO 的数据传输，此时 AI 或 AO 的重采样有效。**
- **ADEC-AO 的数据传输方式无上述限制，当为系统绑定方式时，AO 的重采样仍有效。**

音频重采样需要配置重采样属性：

- u32InPointNum：重采样之前的音频帧的每帧采样点个数。
- enInSampleRate：重采样之前的音频帧的采样率。
- enReSampleType：重采样类型。

重采样之后音频帧的采样率和采样点个数会发生变化。例如 AI 采样率配置为 32K，采样点个数配置为 1280，那么 AI 的重采样参数中 enInSampleRate 应配置为 32K，u32InPointNum 应配置为 1280。当 enReSampleType 配置为 4 到 1 倍重采样时，重采样之后的音频帧的采样率和采样点个数将是 8K 和 320。

噪声抑制

Hi35xx 的音频输入模块支持对音频数据实施噪声抑制。如果启用 AI 抑制功能，与重采样一样，它也是在 HI_MPI_AI_GetFrame 内部进行处理。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

注意

- 如果 AI-AO 或 AI-AENC 之间采取的是 SYS_BIND 方式传输数据，噪声抑制功能不起作用，用户可以选择用户线程方式传输数据。
- 回声抵消包含了噪声抑制功能，如果启用回声抵消，就不需要做噪声抑制。
- 噪声抑制仅支持 8/16kHz、80/160 采样点的音频数据，不支持立体声。

9.2.1.5 高级重采样和声音质量增强

高级重采样

Hi35xx 的音频高级重采样支持任意两种不同采样率之间的重采样。重采样支持的输入输出采样率为：8kHz，11.025kHz，12kHz，16kHz，22.05kHz，24kHz，32kHz，44.1kHz，48kHz。

声音质量增强

Hi35xx 的音频输入模块支持对音频数据进行声音质量增强处理。声音质量增强处理包含回声抵消，噪声抑制，自动增益，高通滤波四个处理模块。声音质量增强中的四个处理模块可以同时使用，也可以部分使用或单独使用。如果启用 AI 声音质量增强功能，非绑定模式时，它是在 HI_MPI_AI_GetFrame 接口内部进行处理；绑定模式时，它是在 HI_MPI_AENC_SendFrame 接口内部进行处理。

回声抵消：此处的回声抵消功能与前一章节的回声抵消处理方式相同。对音频帧的要求为：单声道模式，采样率为 8kHz，11.025kHz，12kHz，16kHz，采样精度为 16bit，且 MIC 采集语音数据的 AI 帧长和远程语音播放的 AO 配置帧长必须相同。以上条件 AI 和 AO 都必须满足。

噪声抑制：此处的噪声抑制功能与前一章节的噪声抑制处理方式相同。对音频帧的要求为：单声道模式，支持 8kHz，11.025kHz，12kHz，16kHz。

自动增益：Hi35xx 的音频输入模块支持对音频数据进行自动增益处理（使用声音质量增强功能中的自动增益处理模块），在声音输入音量有大小变化时，能将输出音量控制在比较一致的范围内。支持 8kHz，11.025kHz，12kHz，16kHz。

高通滤波：滤去音频帧中混杂的低频噪声。支持 8kHz，11.025kHz，12kHz，16kHz。建议在使用声音质量增强功能时，始终开启高通滤波功能。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

注意

- 如果 AI-AO 或 AI-AENC 之间采取的是系统绑定方式传输数据，只有声音质量增强功能中的回声抵消在 AENC 中起作用，噪声抑制和自动增益不起作用。用户可以采用非系统绑定方式，通过 HI_MPI_AI_GetFrame 接口获取声音质量增强处理后的 AI 音频帧，并发送给 AENC/AO，以建立 AI-AENC 或 AI-AO 的数据传输，此时 AI 声音质量增强功能中的所有处理模块都能生效。
- 声音质量增强功能包含了噪声抑制功能，自动增益控制功能，回声抵消功能。用户可以通过 HI_MPI_AI_SetVqeAttr 接口配置声音增强功能的某些功能或全部功能。
- 声音质量增强功能不支持立体声，对采样率，帧长，自动电平范围，降噪功能，回声抵消相关参数的范围要求，详见 9.4 数据类型中，AI_ALC_CONFIG_S, AI_AEC_CONFIG_S，AI_VQE_CONFIG_S 中各成员的描述。

9.2.2 音频编码和解码

9.2.2.1 音频编解码流程

Hi35xx SDK 音频部分的编码类型 G711、G726、ADPCM_DVI4 与 ADPCM_ORG_DVI4 是使用硬编码，ADPCM_IMA 是使用 CPU 软件解码，其中 Hi3518/Hi3516C 芯片没有硬件编码模块所有的编码方式都使用软件编码；而所有的解码功能基于独立封装的海思音频编解码库，核心解码器工作在用户态，使用 CPU 软件解码。SDK 支持通过 SYS 模块的绑定接口，将一个 AI 通道绑定到 AENC 通道，实现录音编码功能；也可以将一个 ADEC 通道绑定到 AO 通道，实现解码播放功能。

9.2.2.2 音频编解码协议

Hi35xx 支持的音频编解码协议说明如表 9-4 所示。

表9-4 音频编解码协议说明

协议	采样率	帧长（采样点）	码率（kbps）	压缩率	CPU 消耗	描述
G711	8kHz	80/160/240/320/480	64	2	1 MHz	优点：语音质量最好；CPU 消耗小；支持广泛，协议免费。 缺点：压缩效率低。 G.711 提供 A 律与 μ 律压缩编码，适用于综合业务网和大多数数字电话链路。北美与日本通常采用 μ 律编码，欧洲和其他地区大都采用 A 律编码。
G726	8kHz	80/160/240/320/480	16、24、32、40（注：G726 编码是一种有损压缩方法，码率比	8～3.2	5 MHz	优点：算法简单；语音质量高，多次转换后语音质量有保证，能够在低码率上达到网络等级的话音质量。 缺点：压缩效率较低。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

协议	采样率	帧长（采样点）	码率（kbps）	压缩率	CPU 消耗	描述
			较小时，压缩比和量化误差较大，可能会影响声音质量）			G726_16KBPS 与 MEDIA_G726_16KBPS 两种编码器区别在于编码输出的打包格式。G726_16KBPS 适用于网络传输；MEDIA_G726_16KBPS 适用于 ASF 存储。请参考 RFC3551.pdf。
ADPCM	8kHz	80/160/240/320/480 或 81/161/241/321/481	32	4	2 MHz	优点：算法简单；语音质量高，多次转换后语音质量有保证，能够在低码率上达到网络等级的话音质量。 缺点：压缩效率较低。 ADPCM_IMA4、ADPCM_ORG_DVI4 和 ADPCM_DVI 是 ADPCM 编码算法的三种封包格式。IMA 封包格式以第一个采样点作为预测值，DVI 封包格式以上一帧作为预测值。即 IMA 编码每帧需要多输入一个采样点，输入采样点个数为 81/161/241/321/481。DVI 编码每帧输入采样点个数为 80/160/240/320/480。

注：“cpu 消耗”的结果值基于 ARM9 288MHz 环境，2 MHz 表示解码分别占有 2M CPU。

注：G726 编码是一种有损压缩方法，码率比较小时，压缩比和量化误差较大，可能会影响声音质量。

注：由于 ADPCM_ORG_DVI4 没有传输预测值，这样在网络丢包时，可能会引起异常。因此在音频码流网络传输应用中，不推荐使用该协议编码。

当编码或解码协议选择 LPCM 时，音频编码和解码模块可以录制和播放 LPCM 格式的原始音频文件。

9.2.2.3 海思语音帧结构

使用海思语音编解码库进行 G711、G726、ADPCM 格式的编码，编码后的码流遵循以下表格中描述的帧结构，即在每帧码流数据的净荷数据之前填充有 4 个字节的帧头；使用语音编解码库进行以上格式的解码时，需要读取相应的帧头信息。帧头中的数据净荷长度字段计算公式：对于 G726 编码，数据净荷长度 = 码率*采样点/1280000；对于 G711a 编码，数据净荷长度 = 采样点/2；对于 ADPCM-DVI4 编码，数据净荷长度 = 采样点/4 +2。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表9-5 海思语音帧结构

参数位置(单位：HI_S16)	参数比特位说明	参数含义
0	[15:8]	数据帧类型标志位。 01：语音帧； 其他：保留。
	[7:0]	保留。
1	[15:8]	帧循环计数器：0～255。
	[7:0]	数据净荷长度(单位：HI_S16)。
2	[15:0]	净荷数据。
3	[15:0]	净荷数据。
……	[15:0]	净荷数据。
2+n-1	[15:0]	净荷数据。
2+n	[15:0]	净荷数据。

9.2.3 芯片差异介绍

音频接口差异

各芯片使用的音频接口如表 9-6 所示。

表9-6 芯片音频接口差异

芯片类型	差异
Hi3531/Hi3532 Hi3521/Hi3520A Hi3518/Hi3516C	使用 SIO 接口做为音频接口
Hi3520D/Hi3515A/Hi3515C	使用 AIO 接口做为音频接口

- 对于使用 SIO 接口的芯片，芯片集成的 SIO 分三类：只支持音频输入的 SIO；支持语音对讲的 SIO；用于与 HDMI 在芯片内部对接的 SIO，相应的 SIO 规格详见 Hi35xx 视频编解码处理器用户指南。其中 Hi3531/Hi3532 的 SIO 与 AI、AO 关系如图 9-5 所示；Hi3521/Hi3520A 的 SIO 与 AI、AO 关系如图 9-6 所示；Hi3518/Hi3516C 的 SIO 与 AI、AO 关系如图 9-7 所示。对于使用 SIO 接口的芯片。
  - 当只支持音频输入的 SIO 或支持语音对讲的 SIO 选择主模式时，它只提供用于时序同步的帧同步时钟和位流时钟，不向外提供 MCLK，这时如果 Audio Codec 使用外接的晶振作为工作时钟，这样可能导致声音失真，所以推荐使用

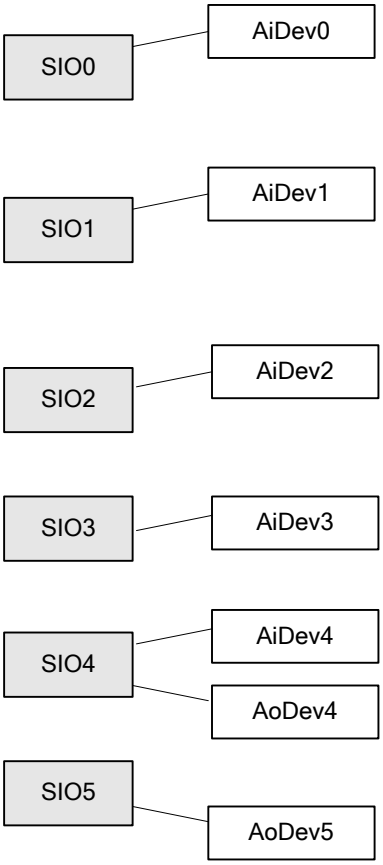
海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

从模式或者使用位流时钟产生 Codec 内部工作主时钟。而用于 HDMI 对接的 SIO 必须使用 I2S 主模式。

- 当两个或两个以上只支持音频输入的 SIO 选择主模式时，它们共用一个 MCLK，此 MCLK 的值在第一个选择主模式的 SIO 成功设置 AI 或 AO 设备属性时确定，在这之后对这几个选择主模式的 SIO 设置 AI 或 AO 设备属性时必须确保它们所要求的时钟与此 MCLK 一致，否则出错。

- 对于使用 AIO 接口的芯片
  - 芯片集成的 AIO 内部分两类：只支持音频输入的 AIP；只支持音频输出的 AOP，相应的 AIO 规格详见 Hi35xx H.264 编解码处理器用户指南。Hi3520D/Hi3515A/Hi3515C 的 AIO 中 AIP/AOP 与 AI、AO 关系如图 9-8 所示。
  - 当 AIO 接口中只支持音频输出的 AOP0 选择主模式时，可以配置是否共用 AOP0 的帧同步时钟配置和位流时钟配置，当只支持音频输出的 AOP0 选择从模式时，不能共用 AOP0 的帧同步时钟配置和位流时钟配置。

图9-5 Hi3531/Hi3532 SIO 与 AI、AO 关系示意图

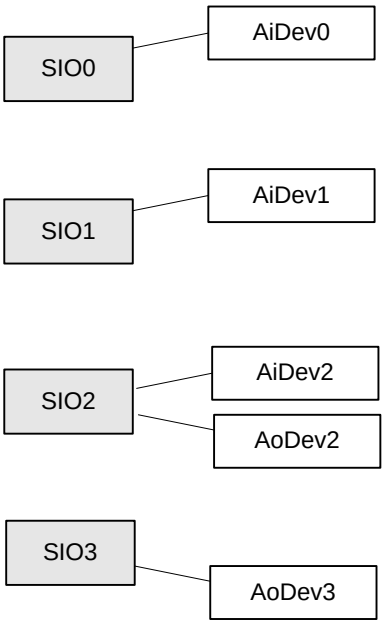


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

说明

- 对于只存在输入或输出功能的 SIO，SDK 还是为其编号了 AI 设备和 AO 设备，只是操作这些不支持的 AI 设备或 AO 设备时会出错，如操作 AoDev0～ AoDev3、AiDev5 会出错。
- SIO4 支持语音对讲。
- SIO5 与 HDMI 对接，只支持 I^2S 主模式。

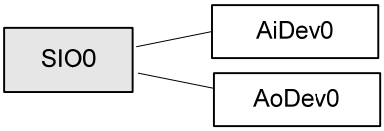
图9-6 Hi3521 /Hi3520ASIO 与 AI、AO 关系示意图



说明

- 对于只存在输入或输出功能的 SIO，SDK 还是为其编号了 AI 设备和 AO 设备，只是操作这些不支持的 AI 设备或 AO 设备时会出错，如操作 AoDev0～AoDev1、AiDev3 会出错。
- SIO2 支持语音对讲。
- SIO3 与 HDMI 对接，只支持 I^2S 主模式。

图9-7 Hi3518/Hi3516C SIO 与 AI、AO 关系示意图

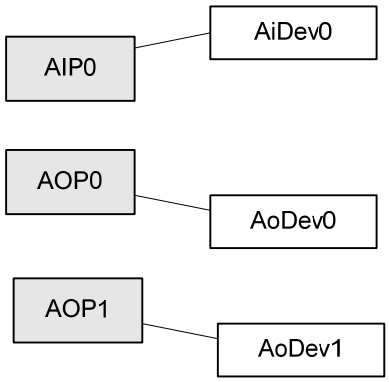


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

说明

- Hi3518/Hi3516C 使用内部 audio codec。Hi3518A/Hi3516C 支持双声道，左右声道输入，左右声道输出。Hi3518C/Hi3518E 只支持单声道，左声道输入、左声道输出。
- SIO0 只支持 I^2S 主模式，可用于对讲。

图9-8 Hi3520D/Hi3515A/Hi3515C AIO 中 AIP/AOP 与 AI、AO 关系示意图



说明

- AIP0 与 AOP0 同时对接同一个 CODEC，可用于语音对讲。
- AOP1 与 HDMI 对接，只支持 I^2S 主模式。

音频的输入输出设备号范围

各芯片的音频输入输出设备号范围如表 9-7 所示。

表9-7 音频输入输出设备号

芯片类型	音频输入设备 ID	音频输出设备 ID
Hi3531	[0, 4]	[4, 5]
Hi3532	[0, 4]	[4]
Hi3521/Hi3520A	[0, 2]	[2, 3]
Hi3518/Hi3516C	[0]	[0]
Hi3520D/Hi3515A/Hi3515C	[0]	[0, 1]

9.2.4 内置 Audio Codec

9.2.4.1 概述

Hi3518A/Hi3518C/Hi3518E/Hi3516C 提供一个内置的 Audio Codec，并在芯片内部对接到 SIO0 接口，即 SIO0 接口只能通过内置的 Audio Codec 完成声音的播放及录制。因为 Audio Codec 不能发送同步时钟，所以 SIO0 接口只能配置为 I^2S 时序的主模式，用户仍需要正确配置 SIO0 和 Audio Codec 对接时序才可接收或发送音频数据。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

9.2.4.2 重要概念

Audio Codec 分为模拟部分和数字部分。模拟部分可以通过模拟混音（MICPGA）选择由麦克风输入或 LINEIN 输入，模拟混音支持增益调节。数字部分有 ADC 和 DAC，完成模拟信号和数字信号之间的转换，并且可分别调节音量。用户在进行音量调节时，可综合模拟部分和数字部分的音量调节，建议优先调节模拟部分音量，如果模拟增益达不到要求，可开启 BOOST 功能。数字增益默认已调为最优。

Audio Codec 的 DAC 支持软静音和软撤销静音。软静音控制数字音频数据被逐渐衰减到 0，软撤销静音控制数字音频数据逐渐增大到配置的增益值，改变的速率由 mute_rate 控制。当软静音使能时，数字音频数据将会逐渐静音为 0；当软静音禁用时，数字音频数据恢复为配置的增益值。

Audio Codec 的工作时钟由 SIO0 提供，并由工作时钟分别产生模拟部分和数字部分的时钟，再由数字部分的时钟产生 ADC 的时钟和 DAC 的时钟。如果默认配置下声音不正常，可以配置数字部分时钟和模拟部分时钟反向，或 DAC 时钟和 ADC 时钟反向。Audio Codec 配置默认采样精度为 16bit。

另外 Audio Codec 支持去加重滤波、pop 音抑制和高通滤波，并默认开启这些功能。

9.2.4.3 ioctl 函数

Audio Codec 的用户态接口以 ioctl 形式体现，其形式如下：

```
int ioctl (int fd,
           unsigned long cmd,
          ……
          );
```

该函数是 Linux 标准接口，具备可变参数特性。但在 Audio Codec 中，实际只需要 3 个参数。因此，其语法形式等同于：

```
int ioctl (int fd,
           unsigned long cmd,
           CMD_DATA_TYPE *cmddata);
```

其中，CMD_DATA_TYPE 随参数 cmd 的变化而变化。这 3 个参数的详细描述如表 9-8 所示。

表9-8 ioctl 函数的 3 个参数

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符，是调用 open 函数打开 Audio Codec 设备文件之后的返回值。	输入
cmd	主要的 cmd（命令控制字）如下： • ACODEC_SOFT_RESET_CTRL：将 codec 恢复为默认设置 • ACODEC_SET_I2S1_FS：设置 I²S1 接口采样率 • ACODEC_SET_MIXER_MIC：输入声道选择 MICIN 或	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
	LINEIN • ACODEC_SET_GAIN_MICL：左声道输入的模拟增益控制 • ACODEC_SET_GAIN_MICR：右声道输入的模拟增益控制 • ACODEC_SET_DACL_VOL：左声道输出音量控制 • ACODEC_SET_DACR_VOL：右声道输出音量控制 • ACODEC_SET_ADCL_VOL：左声道输入音量控制 • ACODEC_SET_ADCR_VOL：右声道输入音量控制 • ACODEC_SET_MICL_MUTE：左声道输入静音控制 • ACODEC_SET_MICR_MUTE：右声道输入静音控制 • ACODEC_SET_DACL_MUTE：左声道输出静音控制 • ACODEC_SET_DACR_MUTE：右声道输出静音控制 • ACODEC_DAC_SOFT_MUTE：输出软静音控制 • ACODEC_DAC_SOFT_UNMUTE：输出软撤销静音控制 • ACODEC_GET_GAIN_MICL：获取模拟左声道输入的增益 • ACODEC_GET_GAIN_MICR：获取模拟右声道输入的增益 • ACODEC_GET_DACL_VOL：获取左声道输出的音量控制 • ACODEC_GET_DACR_VOL：获取右声道输出的音量控制 • ACODEC_GET_ADCL_VOL：获取左声道输入的音量控制 • ACODEC_GET_ADCR_VOL：获取右声道输入的音量控制 • ACODEC_SET_PD_DACL：左声道输出的下电控制 • ACODEC_SET_PD_DACR：右声道输出的下电控制 • ACODEC_SET_PD_ADCL：左声道输入的下电控制 • ACODEC_SET_PD_ADCR：右声道输入的下电控制 • ACODEC_SEL_DAC_CLK：设置 DAC 和 ADC 的时钟沿同沿或反沿 • ACODEC_SEL_ANA_MCLK：设置模拟部分和数字部分的时钟沿同沿或反沿 • ACODEC_DACL_SEL_TRACK：设置 DACL 选择左声道或右声道 • ACODEC_DACR_SEL_TRACK：设置 DACR 选择左声道或右声道 • ACODEC_ADCL_SEL_TRACK：设置 ADCL 选择左声道或右声道 • ACODEC_ADCR_SEL_TRACK：设置 ADCR 选择左声道或右声道 • ACODEC_SET_DAC_DE_EMPHASIS：DAC 的去加重滤波控制	

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
	• ACODEC_SET_ADC_HP_FILTER：ADC 的高通滤波控制 • ACODEC_DAC_POP_FREE：DAC 的去 POP 音控制 • ACODEC_DAC_SOFT_MUTE_RATE：DAC 软静音速率控制 • ACODEC_DAC_SEL_I2S：DAC I^2S 接口选择 • ACODEC_ADC_SEL_I2S：ADC I^2S 接口选择 • ACODEC_SET_I2S1_DATAWIDTH：I^2S1 数据接口位宽 • ACODEC_SET_I2S2_DATAWIDTH：I^2S2 数据接口位宽 • ACODEC_SET_I2S2_FS：设置 I^2S2 接口采样率 • ACODEC_SET_DACR2DACL_VOL：DACR 到 DACL 音量控制 • ACODEC_SET_DACL2DACR_VOL：DACL 到 DACR 音量控制 • ACODEC_SET_ADCL2DACL_VOL：ADCL 到 DACL 音量控制 • ACODEC_SET_ADCR2DACL_VOL：ADCR 到 DACL 音量控制 • ACODEC_SET_ADCL2DACR_VOL：DACR 到 DACL 音量控制 • ACODEC_SET_ADCR2DACR_VOL：ADCR 到 DACR 音量控制	
cmddata	各 cmd 对应的数据指针	输入/输出

9.3 API 参考

9.3.1 音频输入

音频输入（AI）主要实现配置及启用音频输入设备、获取音频帧数据等功能。

该功能模块提供以下 MPI：

- HI_MPI_AI_SetPubAttr：设置 AI 设备属性。
- HI_MPI_AI_GetPubAttr：获取 AI 设备属性。
- HI_MPI_AI_Enable：启用 AI 设备。
- HI_MPI_AI_Disable：禁用 AI 设备。
- HI_MPI_AI_EnableChn：启用 AI 通道。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_AI_DisableChn：禁用 AI 通道。
- HI_MPI_AI_GetFrame：获取音频帧。
- HI_MPI_AI_ReleaseFrame：释放音频帧。
- HI_MPI_AI_SetChnParam：设置 AI 通道参数。
- HI_MPI_AI_GetChnParam：获取 AI 通道参数。
- HI_MPI_AI_EnableAec：启用回声抵消功能。
- HI_MPI_AI_DisableAec：禁用回声抵消功能。
- HI_MPI_AI_EnableReSmp：启用 AI 重采样。
- HI_MPI_AI_DisableReSmp：禁用 AI 重采样。
- HI_MPI_AI_EnableAnr：启用 AI 噪声抑制。
- HI_MPI_AI_DisableAnr：禁用 AI 噪声抑制。
- HI_MPI_AI_GetFd：获取 AI 通道对应设备文件句柄。
- HI_MPI_AI_SetHpfAttr：设置 AI 的声音高通滤波功能相关属性。
- HI_MPI_AI_GetHpfAttr：获取 AI 的声音高通滤波功能相关属性。
- HI_MPI_AI_SetVqeAttr：设置 Vqe 属性。
- HI_MPI_AI_ GetVqeAttr：获取 Vqe 属性。
- HI_MPI_AI_ EnableVqe：使能 Vqe 功能。
- HI_MPI_AI_ DisableVqe：禁用 Vqe 功能。
- HI_MPI_AI_EnableReSmpEx：启用 AI 高级重采样功能。
- HI_MPI_AI_DisableReSmpEx：禁用 AI 高级重采样功能。
- HI_MPI_AI_ClrPubAttr：清空 Pub 属性。
- HI_MPI_AI_Save_File：开启音频输入保存文件功能。

HI_MPI_AI_SetPubAttr

【描述】

设置 AI 设备属性。

【语法】

```
HI_S32 HI_MPI_AI_SetPubAttr(AUDIO_DEV AudioDevId,const AIO_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
pstAttr	AI 设备属性指针。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

音频输入设备的属性决定了输入数据的格式，输入设备属性包括工作模式、采样率、采样精度、buffer 大小、每帧的采样点数、扩展标志、时钟选择和通道数目。这些属性应与对接 Codec 配置的时序一致，即能成功对接。

- 工作模式

  音频输入输出目前支持 I^2S 主模式、I^2S 从模式、标准 PCM 主模式、标准 PCM 从模式、自定义 PCM 主模式以及自定义 PCM 从模式。

  由于时序的问题，AI 设备在从模式下，建议用户先配置好对接的 CODEC，再配置 AI 设备；而在主模式下，建议用户先配置好 AI 设备，再配置对接的 CODEC。

- 采样率

  采样率指一秒中内的采样点数，采样率越高表明失真度越小，处理的数据量也就随之增加。一般来说语音使用 8k 采样率，音频使用 32k 或以上的采样率；设置时请确认对接的 Audio Codec 是否支持所要设定的采样率

- 采样精度

  采样精度指某个通道的采样点数据宽度，同时决定整个设备的通道分布。采样精度可以设置为 8bit 或 16bit，实际应用中采样精度还受 Audio Codec 限制。

- buffer 大小

  AIO_ATTR_S 中的 u32FrmNum 项用于配置 AI 中用于接收音频数据的缓存的音频帧帧数，建议配置为 5 以上，否则可能出现采集丢帧等异常。

- 每帧的采样点数

  当音频采样率较高时，建议相应地增加每帧的采样点数目。如要将这些采集到的音频数据送编码，则应保证每帧的持续时长不少于 10ms（例如 16K 的采样频率下每帧的采样点数至少应设置为 160），否则解码后声音可能有异常。

- 扩展标志

  扩展标志表明在 8bit 采样精度的条件下是否需要对 8bit 数据进行 8bit 到 16bit 带符号扩展，扩展后获取的数据就为 16bit，以满足编码器需求。在设置非 8bit 采样精度的情况下，此标志是无效的。

- 通道数目

  通道数目指当前输入设备的 AI 功能的通道数目，需与对接的 AUDIO CODEC 的配置保持一致；支持 2 路、4 路、8 路和 16 路。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 时钟选择

  AI 必须和 AD 配合起来才能正常工作，用户必须清楚 AD 采集的数据分布和通道的关系才能从正确的通道取得数据。AI 设备为主模式时，决定 AI 设备输出时钟的关键配置项是采样率、采样精度以及通道数目，采样精度乘以通道数目即为 AI 设备时序一次采样的位宽。对于使用 SIO 接口的芯片，能同时支持接收与发送的 SIO 中，它有分别用于 Audio Codec 对接的接收、发送同步时钟线；该项用于配置 SIO 的发送同步时钟和接收同步时钟是否复用。主模式时 AI 和 AO 的同步时钟均由 SIO 提供且总是一致，此时 u32ClkSel 的配置无效。从模式时，当 u32ClkSel 配置为 1 时，表示 AI 和 AO 设备的同步时钟是复用的，此时要保证 AI 和 AO 的同步时钟是一致的；当 u32ClkSel 配置为 0 时，表示 AI 和 AO 设备的同步时钟是由不同的源提供。在设置属性之前需要保证 AI 处于禁用状态，如果处于启用状态则需要首先禁用 AI 设备。

  同一 SIO 下的 AI 和 AO 设备的主从模式、时序选择应该一致，否则设置属性时会返回错误。例如先设置 AI 为 I2S 从模式，此时能正常设置；再设置 AO 为 PCM 主模式，此时将返回错误。

  SIO 为主模式或 u32ClkSel 为 1 时，同一 SIO 下的 AI 和 AO 设备的时钟配置必须相同，即采样精度乘以通道数目必须一致，并且采样频率也必须一致。

  对于使用 AIO 接口的芯片，AIO 接口中 AI 设备与 AO 设备是各自独立的，AO 设备 0 为主模式时，AI 设备 0 和 AO 设备 0 的帧同步时钟配置与位流时钟配置是否共用可以通过 u32ClkSel 来配置，当 u32ClkSel 配置为 1 时，表示 AI 设备 0 和 AO 设备 0 的帧同步时钟配置与位流时钟配置是共用的，此时要保证 AI 设备 0 和 AO 设备 0 的帧同步时钟配置与位流时钟配置是一致的，当 u32ClkSel 配置为 0 时，表示 AI 设备 0 和 AO 设备 0 的帧同步时钟配置与位流时钟配置是由不同的源提供的；从模式时 AI 设备 0 与 AO 设备 0 的帧同步时钟配置与位流时钟配置不共用，此时 u32ClkSel 的配置无效。AI 设备 0 与 AO 设备 0 均为主模式时，当 u32ClkSel 为 1 时，AI 设备 0 和 AO 设备 0 的帧同步时钟配置与位流时钟配置必须相同，即采样精度乘以通道数目必须一致，并且采样频率也必须一致。

【举例】

下面的代码实现设置 AI 设备属性及启用 AI 设备。

```
HI_S32 s32ret;
AIO_ATTR_S stAttr;
AUDIO_DEV AiDevId = 1;

stAttr.enBitwidth = AUDIO_BIT_WIDTH_16;
stAttr.enSamplerate = AUDIO_SAMPLE_RATE_8000;
stAttr.enSoundmode = AUDIO_SOUND_MODE_MOMO;
stAttr.enWorkmode = AIO_MODE_I2S_SLAVE;
stAttr.u32EXFlag = 0;
stAttr.u32FrmNum = 5;
stAttr.u32PtNumPerFrm = 160;
stAttr.u32ChnCnt = 16;
stAttr.u32ClkSel = 1;

/* set public attribute of AI device*/
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
s32ret = HI_MPI_AI_SetPubAttr(AiDevId, &stAttr);
if(HI_SUCCESS != s32ret)
{
    printf("set ai %d attr err:0x%x\n", AiDevId,s32ret);
    return s32ret;
}
/* enable AI device */
s32ret = HI_MPI_AI_Enable(AiDevId);
if(HI_SUCCESS != s32ret)
{
    printf("enable ai dev %d err:0x%x\n", AiDevId, s32ret);
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AI_GetPubAttr

【描述】

获取 AI 设备属性。

【语法】

```
HI_S32 HI_MPI_AI_GetPubAttr(AUDIO_DEV AudioDevId, AIO_ATTR_S*pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
pstAttr	AI 设备公共属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- 获取的属性为前一次配置的属性。
- 如果从来没有配置过属性，则返回失败。

【举例】

```
HI_S32 s32ret;
AUDIO_DEV AiDevId = 1;
AIO_ATTR_S stAttr;
s32ret = HI_MPI_AI_GetPubAttr(AiDevId, &stAttr);
if(HI_SUCCESS != s32ret)
{
    printf("get ai %d attr err:0x%x\n", AiDevId,s32ret);
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AI_Enable

【描述】

启用 AI 设备。

【语法】

```
HI_S32 HI_MPI_AI_Enable(AUDIO_DEV AudioDevId);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 必须在启用前配置 AI 设备属性，否则返回属性未配置错误。
- 如果 AI 设备已经处于启用状态，则直接返回成功。

【举例】

请参见 HI_MPI_AI_SetPubAttr 的举例。

【相关主题】

无。

HI_MPI_AI_Disable

【描述】

禁用 AI 设备。

【语法】

```
HI_S32 HI_MPI_AI_Disable(AUDIO_DEV AudioDevId);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 如果 AI 设备已经处于禁用状态，则直接返回成功。
- 禁用 AI 设备前必须先禁用该设备下已启用的所有 AI 通道。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 要求在禁用 AI 设备之前，先禁用与之关联、使用 AI 的音频数据的 AENC 通道和 AO 设备，否则可能导致该接口调用失败。

【举例】

```
HI_S32 s32ret;
AUDIO_DEV AiDevId = 1;

s32ret = HI_MPI_AI_Disable(AiDevId);
if(HI_SUCCESS != s32ret)
{
    printf("disable ai %d err:0x%x\n", AiDevId);
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AI_EnableChn

【描述】

启用 AI 通道。

【语法】

```
HI_S32 HI_MPI_AI_EnableChn(AUDIO_DEV AudioDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定，详见 AUDIO_SOUND_MODE_E 定义的描述。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

启用 AI 通道前，必须先启用其所属的 AI 设备，否则返回设备未启动的错误码。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_DisableChn

【描述】

禁用 AI 通道。

【语法】

```
HI_S32 HI_MPI_AI_DisableChn(AUDIO_DEV AudioDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【举例】

无。

HI_MPI_AI_GetFrame

【描述】

获取音频帧。

【语法】

```
HI_S32 HI_MPI_AI_GetFrame(AUDIO_DEV AudioDevId, AI_CHN AiChn,
AUDIO_FRAME_S *pstFrm, AEC_FRAME_S *pstAecFrm, HI_BOOL bBlock);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入
pstFrm	音频帧结构体指针。	输出
pstAecFrm	回声抵消参考帧结构体指针。	输出
bBlock	阻塞/非阻塞标识。 取值范围：HI_TRUE；HI_FALSE。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 调用此接口前必须保证 AI 通道是使能或者暂停的，否则返回通道未使能的错误码。
- 如果 AI 的回声抵消功能已使能，pstAecFrm 不能是空指针；如果 AI 的回声抵消功能没有使能，pstAecFrm 可以置为空。
- AI 模块会缓存音频帧数据，用于用户态获取。缓存的深度通过 HI_MPI_AI_SetChnParam 接口设定，默认为 0。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_ReleaseFrame

【描述】

释放音频帧。

【语法】

```
HI_S32 HI_MPI_AI_ReleaseFrame(AUDIO_DEV AudioDevId, AI_CHN AiChn,
AUDIO_FRAME_S *pstFrm, AEC_FRAME_S *pstAecFrm);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入
pstFrm	音频帧结构体指针。	输入
pstAecFrm	回声抵消参考帧结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

参数 pstAecFrm 应与 HI_MPI_AI_GetFrame 接口中 pstAecFrm 参数对应。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_SetChnParam

【描述】

设置 AI 通道参数。

【语法】

```
HI_S32 HI_MPI_AI_SetChnParam(AUDIO_DEV AudioDevId, AI_CHN AiChn,
AI_CHN_PARAM_S *pstChnParam);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入
pstChnParam	音频通道参数。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 调用此接口前必须保证 AI 设备是使能的，否则返回 AI 设备未使能的错误码。
- 通道参数目前只有一个成员变量，用于设置用户获取音频帧的缓存深度，默认深度为 0。该成员变量的值不能大于 30。
- 建议先调用 HI_MPI_AI_GetChnParam 接口获取默认配置，再调用本接口修改配置，以便于后续扩展。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_GetChnParam

【描述】

获取 AI 通道参数。

【语法】

```
HI_S32 HI_MPI_AI_GetChnParam(AUDIO_DEV AudioDevId, AI_CHN AiChn,
AI_CHN_PARAM_S *pstChnParam);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入
pstChnParam	音频通道参数。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

调用此接口前必须保证 AI 设备是使能的，否则返回 AI 设备未使能的错误码。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_EnableAec

【描述】

启用指定 AI 及 AO 的回声抵消功能。

【语法】

```
HI_S32 HI_MPI_AI_EnableAec(AUDIO_DEV AiDevId, AI_CHN AiChn, AUDIO_DEV AoDevId, AO_CHN AoChn);
```

【参数】

参数名称	描述	输入/输出
AiDevId	需要进行回声抵消的 AI 设备号。 取值范围：请参见表 9-7。	输入
AiChn	需要进行回声抵消的 AI 通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 决定。	输入
AoDevId	用于回声抵消的 AO 设备号。 取值范围：请参见表 9-7。	输入
AoChn	用于回声抵消的 AO 通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 启用回声抵消前必须先启用相对应的 AI 通道。
- 多次启用相同 AI、AO 的回声抵消，则返回成功。
- 禁用 AI 通道后，如果重新启用 AI 通道，并使用回声抵消功能，需调用此接口重新启用回声抵消。

【举例】

```
HI_S32 s32ret;
AUDIO_DEV AiDevId = 1;
AI_CHN AiChn = 0;
AUDIO_DEV AoDevId = 0;
AO_CHN AoChn = 0;

s32ret = HI_MPI_AI_EnableAec(AiDevId, AiChn,AUDIO_DEV AoDevId,AO_CHN
AoChn);
if(HI_SUCCESS != s32ret)
{
    printf("enable aec err:0x%x\n", s32ret);
    return s32ret;
}

s32ret = HI_MPI_AI_DisableAec(AiDevId, AiChn)
if(HI_SUCCESS != s32ret)
{
    printf("disable aec err:0x%x\n", s32ret);
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AI_DisableAec

【描述】

禁用回声抵消功能。

【语法】

```
HI_S32 HI_MPI_AI_DisableAec(AUDIO_DEV AiDevId, AI_CHN AiChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AiDevId	AI 设备号。 取值范围：请参见表 9-7。	输入
AiChn	AI 通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

重复调用本接口返回成功。

【举例】

参考 HI_MPI_AI_EnableAec 的举例。

【相关主题】

无。

HI_MPI_AI_EnableReSmp

【描述】

启用 AI 重采样。

【语法】

```
HI_S32 HI_MPI_AI_EnableReSmp(AUDIO_DEV AudioDevId, AI_CHN AiChn,
AUDIO_RESAMPLE_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 决定。	输入
pstAttr	音频重采样属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 在启用 AI 通道之后，调用此接口启用重采样功能。
- 允许重复启用重采样功能，但必须保证与后来配置的属性与之前配置的属性一样。
- 在禁用 AI 通道之后，如果重新启用 AI 通道，并使用重采样功能，需调用此接口重新启用重采样。

【举例】

以 AI 从 32K 到 8K 的重采样为例，配置如下：

```
/* dev attr of ai */
stAioAttr.u32ChnCnt = 2;
stAioAttr.enBitwidth = AUDIO_BIT_WIDTH_16;
stAioAttr.enSamplerate = AUDIO_SAMPLE_RATE_32000;
stAioAttr.enSoundmode = AUDIO_SOUND_MODE_MOMO;
stAioAttr.u32EXFlag = 1;
stAioAttr.u32FrmNum = 30;
stAioAttr.u32PtNumPerFrm = 320*4;

/* attr of resample */
stReSampleAttr.u32InPointNum = 320 * 4;
stReSampleAttr.enInSampleRate = AUDIO_SAMPLE_RATE_32000;
stReSampleAttr.enReSampleType = AUDIO_RESAMPLE_4X1;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_AI_DisableReSmp

【描述】

禁用 AI 重采样。

【语法】

```
HI_S32 HI_MPI_AI_DisableReSmp(AUDIO_DEV AudioDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 不再使用 AI 重采样功能的话，应该调用此接口将其禁用。
- 要求在调用此接口之前，先禁用使用该 AI 设备相应通道音频数据的 AENC 通道和 AO 通道，否则可能导致该接口调用失败。

HI_MPI_AI_EnableReSmpEx

【描述】

启用 AI 高级重采样。

【语法】

```
HI_S32 HI_MPI_AI_EnableReSmpEx(AUDIO_DEV AudioDevId, AI_CHN AiChn,
AUDIO_RESAMPLE_ATTR_EX_S *pstAttr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstAttr	音频高级重采样属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 在启用 AI 通道之后，调用此接口启用重采样功能。
- 允许重复启用重采样功能，但必须保证与后来配置的属性与之前配置的属性一样。
- 在禁用 AI 通道之后，如果重新启用 AI 通道，并使用重采样功能，需调用此接口重新启用重采样。
- 高级重采样支持任意两种不同采样率之间的重采样。支持的输入输出采样率为：8kHz，11.025kHz，12kHz，16kHz，22.05kHz，24kHz，32kHz，44.1kHz，48kHz。
- AI 重采样与高级重采样功能不能同时启用。

【举例】

无。

HI_MPI_AI_DisableReSmpEx

【描述】

禁用 AI 高级重采样。

【语法】

```
HI_S32 HI_MPI_AI_DisableReSmpEx(AUDIO_DEV AudioDevId, AI_CHN AiChn);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 不再使用 AI 高级重采样功能的话，应该调用此接口将其禁用。
- 要求在调用此接口之前，先禁用使用该 AI 设备相应通道音频数据的 AENC 通道和 AO 通道，否则可能导致该接口调用失败。

【举例】

无。

HI_MPI_AI_EnableAnr

【描述】

启用 AI 噪声抑制。

【语法】

```
HI_S32 HI_MPI_AI_EnableAnr(AUDIO_DEV AudioDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AiChn	音频输入通道号。 支持的通道范围由 AI 设备属性中的最大通道个数 u32ChnCnt 决定。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 在 AI 通道使能之后，调用此接口启用噪声抑制功能。
- 重复启用视为成功。
- 禁用 AI 通道后，如果重新启用 AI 通道，并使用噪声抑制功能，需调用此接口重新启用噪声抑制。

【举例】

无。

HI_MPI_AI_DisableAnr

【描述】

禁用 AI 回声抵消。

【语法】

```
HI_S32 HI_MPI_AI_DisableAnr (AUDIO_DEV AudioDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

不再使用 AI 噪声抑制功能时，应该调用此接口将其禁用。

【举例】

无。

HI_MPI_AI_GetFd

【描述】

获取音频输入通道号对应的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_AI_GetFd(AUDIO_DEV AudioDevId ,AI_CHN AiChn)
```

【参数】

参数名称	描述	输入/输出
AiDevId	AI 设备号。 取值范围：请参见表 9-7。	输入
AiChn	AI 通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h
- 库文件：libmpi.a

【注意】

无。

【举例】

```
HI_S32 s32AiFd;
HI_S32 s32ret;
AUDIO_DEV AiDevId = 1;
AI_CHN AiChn = 0;
s32AiFd = HI_MPI_AI_GetFd(AiDevId, AiChn);
if(s32AiFd <= 0)
{
    return HI_FAILURE;
}
```

【相关主题】

无。

HI_MPI_AI_SetHpfAttr

【描述】

设置 AI 的声音高通滤波功能相关属性。

【语法】

```
HI_S32 HI_MPI_AI_SetHpfAttr(AUDIO_DEV AiDevId, AI_CHN AiChn,
AI_HPF_ATTR_S *pstHpfAttr);
```

【参数】

参数名称	描述	输入/输出
AiDevId	AI 设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstHpfAttr	音频输入高通滤波配置结构体指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 必须在使能 vqe 之前调用此接口，否则会报错；vqe 使能后不支持动态调用此接口。
- 如果没有调用此接口就使能 vqe，则 hpf 功能默认是关闭的。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_GetHpfAttr

【描述】

获取 AI 的声音高通滤波功能相关属性。

【语法】

```
HI_S32 HI_MPI_AI_GetHpfAttr(AUDIO_DEV AiDevId, AI_CHN AiChn,
AI_HPF_ATTR_S *pstHpfAttr);
```

【参数】

参数名称	描述	输入/输出
AiDevId	AI 设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstHpfAttr	音频输入高通滤波配置结构体指针。	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_SetVqeAttr

【描述】

设置 AI 的声音质量增强功能相关属性。

【语法】

```
HI_S32 HI_MPI_AI_SetVqeAttr(AUDIO_DEV AiDevId, AI_CHN AiChn, AUDIO_DEV AoDevId, AO_CHN AoChn, AI_VQE_CONFIG_S *pstVqeConfig);
```

【参数】

参数名称	描述	输入/输出
AiDevId	AI 设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
AoDevId	用于回声抵消的 AO 设备号。 取值范围：请参见表 9-7。	输入
AoChn	用于回声抵消的 AO 通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstVqeConfig	音频输入声音质量增强配置结构体指针	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 启用声音质量增强功能前必须先设置相对应 AI 通道的声音质量增强功能相关属性。
- 设置 AI 的声音质量增强功能相关属性前，必须先使能对应的 AI 通道。
- 相同 AI 通道的声音质量增强功能不支持动态设置属性，重新设置 AI 通道的声音质量增强功能相关属性时，需要先关闭 AI 通道的声音质量功能，再设置 AI 通道的声音质量增强功能相关属性。
- 声音质量增强功能包括了回声抵消功能，降噪功能，自动增益控制功能。在设置声音质量增强功能属性时，可通过配置相应的声音质量增强功能属性来选择使能其中的某些功能或全部功能。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_ GetVqeAttr

【描述】

获取 AI 的声音质量增强功能相关属性。

【语法】

```
HI_S32 HI_MPI_AI_GetVqeAttr(AUDIO_DEV AiDevId, AI_CHN AiChn,
AI_VQE_CONFIG_S *pstVqeConfig);
```

【参数】

参数名称	描述	输入/输出
AiDevId	AI 设备号。 取值范围：请参见表 9-7。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstVqeConfig	音频输入声音质量增强配置结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

调用此接口前必须保证 Vqe 属性是设置过的。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_ EnableVqe

【描述】

使能 AI 的声音质量增强功能。

【语法】

```
HI_S32 HI_MPI_AI_EnableVqe(AUDIO_DEV AiDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AiDevId	音频输入设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 启用声音质量增强功能前必须先启用相对应的 AI 通道。
- 多次使能相同 AI 通道的声音质量增强功能时，返回成功。
- 禁用 AI 通道后，如果重新启用 AI 通道，并使用声音质量增强功能，需调用此接口重新启用声音质量增强功能。

【相关主题】

无。

HI_MPI_AI_ DisableVqe

【描述】

禁用 AI 的声音质量增强功能。

【语法】

```
HI_S32 HI_MPI_AI_DisableVqe(AUDIO_DEV AiDevId, AI_CHN AiChn);
```

【参数】

参数名称	描述	输入/输出
AiDevId	音频输入设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

重复调用本接口返回成功。

【举例】

无。

【相关主题】

无。

HI_MPI_AI_ClrPubAttr

【描述】

清空 Pub 属性。

【语法】

```
HI_S32 HI_MPI_AI_ClrPubAttr(AUDIO_DEV AudioDevId);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	AI 设备号。 取值范围：请参见表 9-7。	输入

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

无。

【相关主题】

无。

HI_MPI_AI_Save_File

【描述】

开启音频输入保存文件功能。

【语法】

```
HI_S32 HI_MPI_AI_Save_File(AUDIO_DEV AudioDevId, AI_CHN
AiChn,AUDIO_SAVE_FILE_INFO_S *pstSaveFileInfo);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AiChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstSaveFileInfo	音频保存文件属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

此接口仅用于非绑定模式下 dump AI 文件。调用后会在指定目录下写出三个指定大小文件。Sin.pcm 为输入帧，Rin.pcm 为参考帧，Sou.pcm 为输出帧。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

9.3.2 音频输出

音频输出（AO）主要实现启用音频输出设备、发送音频帧到输出通道等功能。

该功能模块提供以下 MPI：

- HI_MPI_AO_SetPubAttr：设置 AO 设备属性。
- HI_MPI_AO_GetPubAttr：获取 AO 设备属性。
- HI_MPI_AO_Enable：启用 AO 设备。
- HI_MPI_AO_Disable：禁用 AO 设备。
- HI_MPI_AO_EnableChn：启用 AO 通道。
- HI_MPI_AO_DisableChn：禁用 AO 通道。
- HI_MPI_AO_SendFrame：发送 AO 音频帧
- HI_MPI_AO_EnableReSmp：启用 AO 重采样。
- HI_MPI_AO_DisableReSmp：禁用 AO 重采样。
- HI_MPI_AO_EnableReSmpEx：启用 AO 高级重采样功能。
- HI_MPI_AO_DisableReSmpEx：禁用 AO 高级重采样功能。
- HI_MPI_AO_PauseChn：暂停 AO 通道。
- HI_MPI_AO_ResumeChn：恢复 AO 通道。
- HI_MPI_AO_ClearChnBuf：清除 AO 通道中当前的音频数据缓存。
- HI_MPI_AO_QueryChnStat：查询 AO 通道中当前的音频数据缓存状态。
- HI_MPI_AO_SetVolume：设置 AO 设备的音量大小。
- HI_MPI_AO_GetVolume：获取 AO 设备的音量大小。
- HI_MPI_AO_GetFd：获取 AO 通道对应的设备文件句柄。
- HI_MPI_AO_ ClrPubAttr：清空 Pub 属性。

HI_MPI_AO_SetPubAttr

【描述】

设置 AO 设备属性。

【语法】

```
HI_S32 HI_MPI_AO_SetPubAttr(AUDIO_DEV AudioDevId ,const AIO_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstAttr	音频输出设备属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

- 在设置属性之前需要保证 AO 处于禁用状态，如果处于启用状态则需要首先禁用 AO 设备。
- AO 必须和 DA 配合起来才能正常工作，用户必须清楚 DA 发送的数据分布和通道的关系才能从正确的通道发送数据。
- 对于使用 SIO 接口的芯片，同一 SIO 下的 AI 和 AO 设备的主从模式、时钟选择应该一致，否则设置属性时会返回错误。
- SIO 主模式时，同一 SIO 下的 AI 和 AO 设备的时钟选择配置（即 AIO_ATTR_S 中的 u32ClkSel 项）必须相同。
- 由于时序的问题，在 AO 设备从模式下，建议用户先配置好对接的 CODEC，再配置 AO 设备；而在 AO 设备主模式下，建议用户先配置好 AO 设备，再配置对接的 CODEC。
- AO 设备主模式时，决定 AO 设备输出时钟的关键配置项是采样率、采样精度以及通道数目，采样精度乘以通道数目即为 AO 设备时序一次采样的位宽。
- 扩展标志对 AO 设备无效。
- AO 设备属性结构体中其他项请参见 AI 模块中相关接口的描述。

【举例】

```
HI_S32 s32ret;
AIO_ATTR_S stAttr;
AUDIO_DEV AoDevId = 0;

stAttr.enBitwidth = AUDIO_BIT_WIDTH_16;
stAttr.enSamplerate = AUDIO_SAMPLE_RATE_8000;
stAttr.enSoundmode = AUDIO_SOUND_MODE_MOMO;
stAttr.enWorkmode = AIO_MODE_I2S_SLAVE;
stAttr.u32EXFlag = 0;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
stAttr.u32FrmNum = 5;
stAttr.u32PtNumPerFrm = 160;
stAttr.u32ChnCnt = 2;
stAttr.u32ClkSel = 0;

/* set ao public attr*/
s32ret = HI_MPI_AO_SetPubAttr(AoDevId, &stAttr);
if(HI_SUCCESS != s32ret)
{
    printf("set ao %d attr err:0x%x\n", AoDevId,s32ret);
    return s32ret;
}
/* enable ao device*/
s32ret = HI_MPI_AO_Enable(AoDevId);
if(HI_SUCCESS != s32ret)
{
    printf("enable ao dev %d err:0x%x\n", AoDevId, s32ret);
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AO_GetPubAttr

【描述】

获取 AO 设备属性。

【语法】

```
HI_S32 HI_MPI_AO_GetPubAttr(AUDIO_DEV AudioDevId, AIO_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
pstAttr	音频输出设备属性指针。	输出

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

- 获取的属性为前一次配置的属性。
- 如果从未配置过属性，则返回属性未配置的错误。

【举例】

```
HI_S32 s32ret;
AUDIO_DEV AoDevId = 0;
AIO_ATTR_S stAttr;

/* first enable ao device*/

s32ret = HI_MPI_AO_GetPubAttr(AoDevId, &stAttr);
if(HI_SUCCESS != s32ret)
{
    printf("get ao %d attr err:0x%x\n", AoDevId,s32ret);
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AO_Enable

【描述】

启用 AO 设备。

【语法】

```
HI_S32 HI_MPI_AO_Enable(AUDIO_DEV AudioDevId);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

- 要求在启用前配置 AO 设备属性，否则会返回属性未配置的错误。
- 如果 AO 设备已经启用，则直接返回成功。

【举例】

请参见 HI_MPI_AO_SetPubAttr 的举例。

【相关主题】

无。

HI_MPI_AO_Disable

【描述】

禁用 AO 设备。

【语法】

```
HI_S32 HI_MPI_AO_Disable(AUDIO_DEV AudioDevId);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

- 如果 AO 设备已经禁用，则直接返回成功。
- 禁用 AO 设备前必须先禁用设备下所有 AO 通道。

【举例】

无。

【相关主题】

无。

HI_MPI_AO_EnableChn

【描述】

启用 AO 通道。

【语法】

```
HI_S32 HI_MPI_AO_EnableChn(AUDIO_DEV AudioDevId, AI_CHN AoChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

启用 AO 通道前，必须先启用其所属的 AO 设备，否则返回设备未启动的错误码。

【举例】

请参见 HI_MPI_AI_SetPubAttr 的举例。

【相关主题】

无。

HI_MPI_AO_DisableChn

【描述】

禁用 AO 通道。

【语法】

```
HI_S32 HI_MPI_AO_DisableChn(AUDIO_DEV AudioDevId, AI_CHN AoChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

无。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

HI_MPI_AO_SendFrame

【描述】

发送 AO 音频帧。

【语法】

```
HI_S32 HI_MPI_AO_SendFrame(AUDIO_DEV AudioDevId, AO_CHN AoChn, const
AUDIO_FRAME_S *pstData, HI_BOOL bBlock);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入
pstData	音频帧结构体指针。	输入
bBlock	阻塞/非阻塞标识。 取值范围：HI_TRUE；HI_FALSE。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 调用此接口前必须保证 AO 通道是使能或者暂停的，否则返回通道未使能的错误码。
- 该接口用于用户主动发送音频帧至 AO 输出，如果 AO 通道已经通过 HI_MPI_SYS_Bind 接口与 AI 或 ADEC 绑定，不需要也不建议调此接口。

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

HI_MPI_AO_EnableReSmp

【描述】

启用 AO 重采样。

【语法】

```
HI_S32 HI_MPI_AO_EnableReSmp(AUDIO_DEV AudioDevId, AO_CHN AoChn,
AUDIO_RESAMPLE_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 决定。	输入
pstAttr	音频重采样属性结构体指针	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 应该在启用 AO 通道之后，绑定 AO 通道之前，调用此接口启用重采样功能。
- 允许重复启用重采样功能，但必须保证与后来配置的属性与之前配置的属性一样。
- 在禁用 AO 通道后，如果重新启用 AO 通道，并使用重采样功能，需调用此接口重新启用重采样。

【举例】

以 ADEC 到 AO 的解码回放 8K 到 32K 重采样为例，配置如下：

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
/* dev attr of ao */
stAioAttr.u32ChnCnt = 2;
stAioAttr.enBitwidth = AUDIO_BIT_WIDTH_16;
stAioAttr.enSamplerate = AUDIO_SAMPLE_RATE_32000;
stAioAttr.enSoundmode = AUDIO_SOUND_MODE_MOMO;
stAioAttr.u32EXFlag = 1;
stAioAttr.u32FrmNum = 30;
stAioAttr.u32PtNumPerFrm = 320*4;

/* attr of resample */
stReSampleAttr.u32InPointNum = 320;
stReSampleAttr.enInSampleRate = AUDIO_SAMPLE_RATE_8000;
stReSampleAttr.enReSampleType = AUDIO_RESAMPLE_1X4;
```

HI_MPI_AO_DisableReSmp

【描述】

禁用 AO 重采样。

【语法】

```
HI_S32 HI_MPI_AO_DisableReSmp(AUDIO_DEV AudioDevId, AO_CHN AoChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

不再使用 AO 重采样功能的话，应该调用此接口将其禁用。

HI_MPI_AO_EnableReSmpEx

【描述】

启用 AO 高级重采样。

【语法】

```
HI_S32 HI_MPI_AO_EnableReSmpEx(AUDIO_DEV AudioDevId, AO_CHN AoChn,
AUDIO_RESAMPLE_ATTR_EX_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstAttr	音频高级重采样属性结构体指针	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

- 应该在启用 AO 通道之后，绑定 AO 通道之前，调用此接口启用重采样功能。
- 允许重复启用重采样功能，但必须保证与后来配置的属性与之前配置的属性一样。
- 在禁用 AO 通道后，如果重新启用 AO 通道，并使用重采样功能，需调用此接口重新启用重采样。

【举例】

无

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_AO_DisableReSmpEx

【描述】

禁用 AO 高级重采样。

【语法】

```
HI_S32 HI_MPI_AO_DisableReSmpEx(AUDIO_DEV AudioDevId, AO_CHN AoChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a libresampler.a libaec.a libanr.a

【注意】

不再使用 AO 高级重采样功能的话，应该调用此接口将其禁用。

【相关主题】

无。

HI_MPI_AO_PauseChn

【描述】

暂停 AO 通道。

【语法】

```
HI_S32 HI_MPI_AO_PauseChn(AUDIO_DEV AudioDevId, AI_CHN AoChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- AO 通道暂停后，如果绑定的 ADEC 通道继续向此通道发送音频帧数据，发送的音频帧数据将会被阻塞；而如果绑定的 AI 通道继续向此通道发送音频帧数据，在通道缓冲未满的情况下则将音频帧放入缓冲区，在满的情况下则将音频帧丢弃。
- AO 通道为禁用状态时，不允许调用此接口暂停 AO 通道。

【举例】

无。

【相关主题】

无。

HI_MPI_AO_ResumeChn

【描述】

恢复 AO 通道。

【语法】

```
HI_S32 HI_MPI_AO_ResumeChn(AUDIO_DEV AudioDevId, AI_CHN AoChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	音频输出通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- AO 通道暂停后可以通过调用此接口重新恢复。
- AO 通道为暂停状态或使能状态下，调用此接口返回成功；否则调用将返回错误。

【举例】

无。

HI_MPI_AO_ClearChnBuf

【描述】

清除 AO 通道中当前的音频数据缓存。

【语法】

```
HI_S32 HI_MPI_AO_ClearChnBuf(AUDIO_DEV AudioDevId ,AO_CHN AoChn);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AoChn	AO 通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 在 AO 通道成功启用后再调用此接口。
- 为完全清除解码回放通路上所有缓存数据，此接口还应该与 HI_MPI_ADEC_ClearChnBuf 接口配合使用。

【举例】

无。

HI_MPI_AO_QueryChnStat

【描述】

查询 AO 通道中当前的音频数据缓存状态。

【语法】

```
HI_S32 HI_MPI_AO_QueryChnStat(AUDIO_DEV AudioDevId, AO_CHN AoChn,
AO_CHN_STATE_S *pstStatus);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	AO 通道号。 支持的通道范围由 AO 设备属性中的最大通道个数	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
	u32ChnCnt 与声道模式 enSoundmode 决定。	
pstStatus	缓存状态结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

在 AO 通道成功启用后再调用此接口。

【举例】

无。

HI_MPI_AO_SetVolume

【描述】

设置 AO 设备的音量大小。

【语法】

```
HI_S32 HI_MPI_AO_SetVolume(AUDIO_DEV AudioDevId, AO_CHN AoChn, HI_S32
s32VolumeDb);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	AO 通道号。 支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	输入
s32VolumeDb	音量。 取值范围：[-81, 6]，单位：dB	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 在 AO 设备成功启用后再调用此接口。
- 仅支持对 16bit 位宽的数据进行音量控制。
- 本接口由软件算法实现音量的调节，默认输出音量大小为 0dB，即原始数据的音量大小，0 至-81dB 表示进行音量减小控制，0 至 6dB 表示进行音量放大控制，-81dB 的音量最小，达到静音状态。某些高音音频数据进行音量放大时，可能会超出波形幅值，产生“嗤嗤”的杂音。
- 禁用设备后，再使能设备，音量会恢复原始值 0dB。
- 音量控制是针对整个设备，通道号仅作为保留参数，目前无效，输入的通道号只需在合法的取值范围内即可。
- Hi3518/Hi3516C 不支持本功能。

【举例】

无。

HI_MPI_AO_GetVolume

【描述】

获取 AO 设备的音量大小。

【语法】

```
HI_S32 HI_MPI_AO_GetVolume(AUDIO_DEV AudioDevId, AO_CHN AoChn, HI_S32
*ps32VolumeDb);
```

【参数】

参数名称	描述	输入/输出
AudioDevId	音频设备号。 取值范围：请参见表 9-7。	输入
AoChn	AO 通道号。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
	支持的通道范围由 AO 设备属性中的最大通道个数 u32ChnCnt 与声道模式 enSoundmode 决定。	
ps32VolumeDb	音量的指针变量。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 在 AO 设备成功启用后再调用此接口。
- 音量控制是针对整个设备的，通道号仅作为保留参数，目前无效，输入的通道号只需在合法的取值范围内即可。
- Hi3518/Hi3516C 不支持本功能。

【举例】

无。

HI_MPI_AO_GetFd

【描述】

获取音频输出通道号对应的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_AO_GetFd(AUDIO_DEV AudioDevId, AO_CHN AoChn)
```

【参数】

参数名称	描述	输入/输出
AudioDevId	AO 设备号。 取值范围：请参见表 9-7。	输入
AoChn	AO 通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

无。

【举例】

```
HI_S32 s32AoFd;
HI_S32 s32ret;
AUDIO_DEV AoDevId = 0;
AO_CHN AoChn = 0;

/* first enable ao device */

s32AoFd = HI_MPI_AO_GetFd(AoDevId, AoChn);
if(s32AoFd <= 0)
{
    return HI_FAILURE;
}
```

【相关主题】

无。

HI_MPI_AO_ ClrPubAttr

【描述】

清空 Pub 属性。

【语法】

```
HI_S32 HI_MPI_AO_ClrPubAttr(AUDIO_DEV AudioDevId);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
AudioDevId	AO 设备号。 取值范围：请参见表 9-7。	输入

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

无。

【相关主题】

无。

9.3.3 音频编码

音频编码主要实现创建编码通道、发送音频帧编码及获取编码码流等功能。

该功能模块提供以下 MPI：

- HI_MPI_AENC_CreateChn：创建音频编码通道。
- HI_MPI_AENC_DestroyChn：销毁音频编码通道。
- HI_MPI_AENC_SendFrame：发送音频编码音频帧
- HI_MPI_AENC_GetStream：获取音频编码码流。
- HI_MPI_AENC_ReleaseStream：释放音频编码码流。
- HI_MPI_AENC_GetFd:获取音频编码通道号对应的设备文件句柄.
- HI_MPI_AENC_RegeisterEncoder：注册编码器。
- HI_MPI_AENC_UnRegisterEncoder：注销编码器。
- HI_MPI_AENC_Save_File：开启音频编码保存文件功能。

HI_MPI_AENC_CreateChn

【描述】

创建音频编码通道。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_AENC_CreateChn(AENC_CHN AeChn, const AENC_CHN_ATTR_S
*pstAttr);
```

【参数】

参数名称	描述	输入/输出
AeChn	通道号。 取值范围：[0, AENC_MAX_CHN_NUM)。	输入
pstAttr	音频编码通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 协议类型指定该通道的编码协议，目前支持 G711、G726、ADPCM，具体内容如表 9-4 所示。
- 表 9-4 中列举的编解码协议只支持 16bit 线性 PCM 音频数据处理，如果输入的是 8bit 采样精度的数据，AENC 内部会将其扩展为 16bit；建议配置 AI 设备公共属性时将扩展标志置为 1，使得 AI 数据由 8bit 自动扩展到 16bit。
- 海思语音帧结构如表 9-5 所示。
- 音频编码的部分属性需要与输入的音频数据属性相匹配，例如采样率、帧长（每帧采样点数目）等。
- buffer 大小以帧为单位，取值范围是[2, MAX_AUDIO_FRAME_NUM]，建议配置为 10 以上，过小的 buffer 配置可能导致丢帧等异常。
- 在通道闲置时才能使用此接口，如果通道已经被创建，则返回通道已经创建的错误。

【举例】

```
HI_S32 s32ret;
AENC_CHN_ATTR_S stAencAttr;
ADEC_ATTR_ADPCM_S stAdpcmAenc;
AENC_CHN AencChn = 0;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
AUDIO_FRAME_S stAudioFrm;
AUDIO_STREAM_S stAudioStream;

stAencAttr.enType = PT_ADPCMA;/* ADPCM */
stAencAttr.u32BufSize = 8;
stAencAttr.pValue = &stAdpcmAenc;
stAdpcmAenc.enADPCMType = ADPCM_TYPE_DVI4;

/* create aenc chn*/
s32ret = HI_MPI_AENC_CreateChn(AencChn, &stAencAttr);
if (HI_SUCCESS != s32ret)
{
    printf("create aenc chn %d err:0x%x\n", AencChn,s32ret);
    return s32ret;
}

/* bind AENC to AI channel  */
s32Ret = HI_MPI_AENC_BindAi(AencChn, AiDev, AiChn);
if (s32Ret != HI_SUCCESS)
{
     return s32Ret;
}

/* get stream from aenc chn */
s32ret = HI_MPI_AENC_GetStream(AencChn, &stAudioStream);
if (HI_SUCCESS != s32ret )
{
    printf("get stream from aenc chn %d fail \n", AencChn);
    return s32ret;
}

/* deal with audio stream */

/* release audio stream */
s32ret = HI_MPI_AENC_ReleaseStream(AencChn, &stAudioStream);
if (HI_SUCCESS != s32ret )
{
    return s32ret;
}

/* destroy aenc chn */
s32ret = HI_MPI_AENC_DestroyChn(AencChn);
if (HI_SUCCESS != s32ret )
{
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    return s32ret;
}
```

【相关主题】

无。

HI_MPI_AENC_DestroyChn

【描述】

销毁音频编码通道。

【语法】

```
HI_S32 HI_MPI_AENC_DestroyChn(AENC_CHN AeChn);
```

【参数】

参数名称	描述	输入/输出
AeChn	通道号。 取值范围：[0, AENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 通道未创建的情况下调用此接口会返回成功。
- 如果正在获取/释放码流或者发送帧时销毁该通道，则会返回失败，用户同步处理时需要注意。

【举例】

请参见 HI_MPI_AENC_CreateChn 的举例。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_AENC_SendFrame

【描述】

发送音频编码音频帧。

【语法】

```
HI_S32 HI_MPI_AENC_SendFrame(AENC_CHN AeChn, const AUDIO_FRAME_S *pstFrm,
const AEC_FRAME_S *pstAecFrm);
```

【参数】

参数名称	描述	输入/输出
AeChn	通道号。 取值范围：[0, AENC_MAX_CHN_NUM)。	输入
pstFrm	音频帧结构体指针。	输入
pstAecFrm	回声抵消参考帧结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ai.h、hi_comm_aio.h
- 库文件：libmpi.a

【注意】

- 调用此接口前必须保证 AENC 通道已经创建。
- 如果不需要回声抵消，pstAecFrm 可置为 NULL。
- 音频编码发送码流是非阻塞接口，如果音频码流缓存满，则直接返回失败。
- 该接口用于用户主动发送音频帧进行编码，如果 AENC 通道已经通过 HI_MPI_SYS_Bind 接口与 AI 绑定，不需要也不建议调此接口。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_AENC_GetStream

【描述】

获取编码后码流。

【语法】

```
HI_S32 HI_MPI_AENC_GetStream(AENC_CHN AeChn,
AUDIO_STREAM_S*pstStream ,HI_BOOL bBlock);
```

【参数】

参数名称	描述	输入/输出
AeChn	通道号。 取值范围：[0, AENC_MAX_CHN_NUM)。	输入
pstStream	获取的音频码流。	输出
bBlock	阻塞标识。 取值范围： HI_TRUE：阻塞。 HI_FALSE：非阻塞。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 必须创建通道后才可能获取码流，否则直接返回失败，如果在获取码流过程中销毁通道则会立刻返回失败。
- 支持阻塞或非阻塞方式获取码流，并且支持标准的 select 系统调用。
- 当阻塞方式获取码流时，如果音频数据 Buffer 空则此接口调用会被阻塞，直至 Buffer 中有新的数据、销毁 AENC 通道，或者阻塞 1 秒后 Buffer 仍为空。
- 直接获取 AI 原始音频数据的方法

  创建一路 AENC 通道，编码协议类型设置为 PT_LPCM，绑定 AI 通道后，从此 AENC 通道获取的音频数据即 AI 原始数据。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

请参见 HI_MPI_AENC_CreateChn 的举例。

【相关主题】

无。

HI_MPI_AENC_ReleaseStream

【描述】

释放从音频编码通道获取的码流。

【语法】

```
HI_S32 HI_MPI_AENC_ReleaseStream(AENC_CHN AeChn, const AUDIO_STREAM_S
*pstStream);
```

【参数】

参数名称	描述	输入/输出
AeChn	通道号。 取值范围：[0, AENC_MAX_CHN_NUM)。	输入
pstStream	获取的码流指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 码流最好能够在使用完之后立即释放，如果不及时释放，会导致编码过程阻塞。
- 释放的码流必须是从该通道获取的码流，不得对码流信息结构体进行任何修改，否则会导致码流不能释放，使此码流 buffer 丢失，甚至导致程序异常。
- 释放码流时必须保证通道已经被创建，否则直接返回失败，如果在释放码流过程中销毁通道则会立刻返回失败。

【举例】

请参见 HI_MPI_AENC_CreateChn 的举例。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

无。

HI_MPI_AENC_GetFd

【描述】

获取音频编码通道号对应的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_AENC_GetFd(AENC_CHN AeChn)
```

【参数】

参数名称	描述	输入/输出
AeChn	AENC 通道号。 取值范围：[0, AENC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a

【注意】

无。

HI_MPI_AENC_RegeisterEncoder

【描述】

注册编码器。

【语法】

```
HI_S32 HI_MPI_AENC_RegeisterEncoder(HI_S32 *ps32Handle, AENC_ENCODER_S
*pstEncoder)
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
ps32Handle	注册句柄。	输出
pstEncoder	编码器属性结构体。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a

【注意】

- 用户通过传入编码器属性结构体，向 AENC 模块注册一个编码器，并返回注册句柄，用户可以最后通过注册句柄来注销该编码器。
- AENC 模块最大可注册 20 个编码器，且自身已注册 LPCM、G711.a、G711.u、G726、ADPCM 五个编码器。
- 同一种编码协议不允许重复注册编码器，例如假如已注册 G726 编码器，不允许另外再注册一个 G726 编码器。

【举例】

无。

【相关主题】

无。

HI_MPI_AENC_UnRegisterEncoder

【描述】

注销编码器。

【语法】

```
HI_S32 HI_MPI_AENC_UnRegisterEncoder(HI_S32 s32Handle)
```

【参数】

参数名称	描述	输入/输出
s32Handle	注册句柄（注册编码器时获得的句柄）。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aenc.h、mpi_aenc.h
- 库文件：libmpi.a

【注意】

通常不需要注销编码器。

【举例】

无。

【相关主题】

无。

HI_MPI_AENC_Save_File

【描述】

开启音频编码保存文件功能。

【语法】

```
HI_S32 HI_MPI_AENC_Save_File(AENC_CHN AeChn, AUDIO_SAVE_FILE_INFO_S
*pstSaveFileInfo);
```

【参数】

参数名称	描述	输入/输出
AeChn	音频输入通道号。 取值范围：[0, AIO_MAX_CHN_NUM)。	输入
pstSaveFileInfo	音频保存文件属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、mpi_ao.h
- 库文件：libmpi.a

【注意】

此接口仅用于绑定模式下 dump AI 文件。调用后会在指定目录下写出三个指定大小文件。Sin.pcm 为输入帧，Rin.pcm 为参考帧，Sou.pcm 为输出帧。

【相关主题】

无。

9.3.4 音频解码

音频解码主要实现创建解码通道、发送音频码流解码及获取解码后音频帧等功能。

该功能模块提供以下 MPI：

- HI_MPI_ADEC_CreateChn：创建音频解码通道。
- HI_MPI_ADEC_DestroyChn：销毁音频解码通道。
- HI_MPI_ADEC_SendStream：发送音频码流到音频解码通道。
- HI_MPI_ADEC_ClearChnBuf：清除 ADEC 通道中当前的音频数据缓存。
- HI_MPI_ADEC_RegeisterDecoder：注册解码器。
- HI_MPI_ADEC_UnRegisterDecoder：注销解码器。
- HI_MPI_ADEC_GetFrame：获取音频解码帧数据。
- HI_MPI_ADEC_ReleaseFrame：释放音频解码帧数据。

HI_MPI_ADEC_CreateChn

【描述】

创建音频解码通道。

【语法】

```
HI_S32 HI_MPI_ADEC_CreateChn(ADEC_CHN AdChn, ADEC_CHN_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstAttr	通道属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 协议类型指定了该通道的解码协议，目前支持 G711、G726、ADPCM。
- 音频解码的部分属性需要与输出设备属性相匹配，例如采样率、帧长（每帧采样点数目）等。
- buffer 大小以帧为单位，取值范围是[2, MAX_AUDIO_FRAME_NUM]，建议配置为 10 以上，过小的 buffer 配置可能导致丢帧等异常。
- 在通道未创建前（或销毁后）才能使用此接口，如果通道已经被创建，则返回通道已经创建。

【举例】

```
HI_S32 s32ret;
ADEC_CHN_ATTR_S stAdecAttr;
ADEC_ATTR_ADPCM_S stAdpcm;
ADEC_CHN AdChn = 0;
AUDIO_STREAM_S stAudioStream;
AUDIO_DEV AoDev = 0;
AO_CHN AoChn = 0;
MPP_CHN_S stSrcChn;
MPP_CHN_S stDestChn;

stAdecAttr.enType = PT_ADPCMA;
stAdecAttr.u32BufSize = 8;
stAdecAttr.enMode = ADEC_MODE_STREAM;
stAdecAttr.pValue = &stAdpcm;
stAdpcm.enADPCMType = ADPCM_TYPE_DVI4;

/* create adec chn*/
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
s32ret = HI_MPI_ADEC_CreateChn(AdChn, &stAdecAttr);
if (s32ret)
{
    printf("create adec chn %d err:0x%x\n", AdChn,s32ret);
    return s32ret;
}
/* bind ADEC to AO channel*/
stSrcChn.enModId = HI_ID_ADEC;
stSrcChn.s32DevId = 0;
stSrcChn.s32ChnId = AdChn;
stDestChn.enModId = HI_ID_AO;
stDestChn.s32DevId = AoDev;
stDestChn.s32ChnId = AoChn;

s32ret = HI_MPI_SYS_Bind(&stSrcChn, &stDestChn);
if (s32ret)
{
    printf("bind adec chn %d to ao(%d, %d) error:0x%x\n", AdChn, AoDev,
AoChn, s32ret);
    return s32ret;
}

/* get audio stream from network or file*/

/* send audio stream to adec chn */
s32ret = HI_MPI_ADEC_SendStream(AdChn, &stAudioStream);
if (s32ret)
{
    printf("send stream to adec fail\n");
    return s32ret;
}

/* destroy adec chn */
s32ret = HI_MPI_ADEC_DestroyChn(AdChn);
if (HI_SUCCESS != s32ret )
{
    return s32ret;
}
```

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_ADEC_DestroyChn

【描述】

销毁音频解码通道。

【语法】

```
HI_S32 HI_MPI_ADEC_DestroyChn(ADEC_CHN AdChn);
```

【参数】

参数名称	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 通道未创建的情况下调用此接口会返回成功。
- 如果正在获取/释放码流或者发送帧，销毁该通道则这些操作都会立即返回失败。

【举例】

请参见 HI_MPI_ADEC_CreateChn 的举例。

【相关主题】

无。

HI_MPI_ADEC_SendStream

【描述】

向音频解码通道发送码流。

【语法】

```
HI_S32 HI_MPI_ADEC_SendStream(ADEC_CHN AdChn,
       const AUDIO_STREAM_S *pstStream, HI_BOOL bBlock);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入
pstStream	音频码流。	输入
bBlock	阻塞标识。 HI_TRUE：阻塞。 HI_FALSE：非阻塞。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 创建解码通道时可以指定解码方式为 pack 方式或 stream 方式。
  - pack 方式用于确定码流包为一帧的情况下，比如从 AENC 直接获取的码流，从文件读取确切知道一帧边界（语音编码码流长度固定，很容易确定边界）的码流，效率较高。
  - stream 方式用于不确定码流包为一帧的情况下，效率较低，且可能会有延迟。
- LPCM 解码只支持 pack 方式；其他音频格式的解码两种方式都支持。
- 发送数据时必须保证通道已经被创建，否则直接返回失败，如果在送数据过程中销毁通道则会立刻返回失败。
- 支持阻塞或非阻塞方式发送码流。
- 当阻塞方式发送码流时，如果用于缓存解码后的音频帧的 Buffer 满，则此接口调用会被阻塞，直至解码后的音频帧数据被取走，或 ADEC 通道被销毁。
- 确保发送给 ADEC 通道的码流数据的正确性，否则可能引起解码器异常退出。

【举例】

请参见 HI_MPI_ADEC_CreateChn 的举例。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_ADEC_ClearChnBuf

【描述】

清除 ADEC 通道中当前的音频数据缓存。

【语法】

```
HI_S32 HI_MPI_ADEC_ClearChnBuf(ADEC_CHN AdChn);
```

【参数】

参数名称	描述	输入/输出
AdChn	通道号。 取值范围：[0, ADEC_MAX_CHN_NUM)。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_aio.h、hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a lib_VoiceEngine.a lib_aec.a

【注意】

- 要求解码通道已经被创建，如果通道未被创建则返回通道不存在错误码。
- 使用本接口时，不建议使用 stream 方式解码。使用 stream 方式解码进行清除缓存操作时，用户需要确保清除完缓存后，发送给解码器的数据必须是完整的一帧码流，否则可能导致解码器不能正常工作。
- 无论是否使用 stream 方式解码，都要确保送数据解码的操作和清除缓存的操作之间的同步。

【举例】

无。

【相关主题】

无。

HI_MPI_ADEC_RegeisterDecoder

【描述】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

注册解码器。

【语法】

```
HI_S32 HI_MPI_ADEC_RegeisterDecoder(HI_S32 *ps32Handle, ADEC_DECODER_S
*pstDecoder);
```

【参数】

参数名称	描述	输入/输出
ps32Handle	注册句柄。	输出
pstDecoder	解码器属性结构体。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a

【注意】

- 用户通过传入解码器属性结构体，向 ADEC 模块注册一个解码器，并返回注册句柄，用户可以最后通过注册句柄来注销该解码器。
- ADEC 模块最大可注册 20 个解码器，且自身已注册 LPCM、G711a、G711u、G726、ADPCM 五个解码器。
- 同一种解码协议不允许重复注册解码器，例如假如已注册 G726 解码器，不允许另外再注册一个 G726 解码器。

【举例】

无。

【相关主题】

无。

HI_MPI_ADEC_UnRegisterDecoder

【描述】

注销解码器。

【语法】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_S32 HI_MPI_ADEC_UnRegisterDecoder(HI_S32 s32Handle);
```

【参数】

参数名称	描述	输入/输出
s32Handle	注册句柄（注册解码器时获得的句柄）。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a

【注意】

通常不需要注销解码器。

【举例】

无。

【相关主题】

无。

HI_MPI_ADEC_GetFrame

【描述】

获取音频解码帧数据。

【语法】

```
HI_S32 HI_MPI_ADEC_GetFrame(ADEC_CHN AdChn, AUDIO_FRAME_INFO_S
*pstFrmInfo, HI_BOOL bBlock);
```

【参数】

参数名称	描述	输入/输出
AdChn	音频解码通道。	输入
pstFrmInfo	音频帧数据结构体	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
bBlock	是否以阻塞方式获取	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a

【注意】

- 必须在 ADEC 通道创建之后调用。
- 使用本接口获取解码帧数据时，建议发送码流时按帧发送。
- 使用本接口获取音频帧数据时，如果发送码流按流发送，请务必保证获取解码帧数据的及时性，否则会有异常。
- 使用本接口获取音频数据时，请解除 ADEC 与 AO 的绑定关系，否则获取到的帧是不连续的。

【举例】

无。

【相关主题】

无。

HI_MPI_ADEC_ReleaseFrame

【描述】

释放获取到的音频解码帧数据。

【语法】

```
HI_S32 HI_MPI_ADEC_ReleaseFrame(ADEC_CHN AdChn, AUDIO_FRAME_INFO_S
*pstFrmInfo);
```

【参数】

参数名称	描述	输入/输出
AdChn	音频解码通道。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstFrmInfo	音频帧数据结构体	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：hi_comm_adec.h、mpi_adec.h
- 库文件：libmpi.a

【注意】

- 必须在 ADEC 通道创建之后调用。
- 本接口必须与接口 HI_MPI_ADEC_GetFrame 配合使用。

【举例】

无。

【相关主题】

无。

9.3.5 内置 Audio Codec

内置 Audio Codec 主要通过 ioctl 提供对硬件设备的操作。在提供的 ioctl 的 cmd 中，有些 cmd 用户可以不需要调用，直接使用模块加载时的默认配置即可。ioctl 调用实现的是对 Audio Codec 寄存器的读写。

9.3.5.1 Audio Codec 标准功能 cmd

- ACODEC_SOFT_RESET_CTRL：将 codec 恢复为默认设置。
- ACODEC_SET_I2S1_FS：设置 I^2S1 接口采样率。
- ACODEC_SET_MIXER_MIC：输入声道选择 MICIN 或 LINEIN。
- ACODEC_SET_GAIN_MICL：左声道输入的模拟增益控制。
- ACODEC_SET_GAIN_MICR：右声道输入的模拟增益控制。
- ACODEC_SET_DACL_VOL：左声道输出音量控制。
- ACODEC_SET_DACR_VOL：右声道输出音量控制。
- ACODEC_SET_ADCL_VOL：左声道输入音量控制。
- ACODEC_SET_ADCR_VOL：右声道输入音量控制。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- ACODEC_SET_MICL_MUTE：左声道输入静音控制。
- ACODEC_SET_MICR_MUTE：右声道输入静音控制。
- ACODEC_SET_DACL_MUTE：左声道输出静音控制。
- ACODEC_SET_DACR_MUTE：右声道输出静音控制。
- ACODEC_DAC_SOFT_MUTE：输出软静音控制。
- ACODEC_DAC_SOFT_UNMUTE：输出软撤销静音控制。
- ACODEC_GET_GAIN_MICL：获取模拟左声道输入的增益。
- ACODEC_GET_GAIN_MICR：获取模拟右声道输入的增益。
- ACODEC_GET_DACL_VOL：获取左声道输出的音量控制。
- ACODEC_GET_DACR_VOL：获取右声道输出的音量控制。
- ACODEC_GET_ADCL_VOL：获取左声道输入的音量控制。
- ACODEC_GET_ADCR_VOL：获取右声道输入的音量控制。
- ACODEC_SET_PD_DACL：左声道输出的下电控制。
- ACODEC_SET_PD_DACR：右声道输出的下电控制。
- ACODEC_SET_PD_ADCL：左声道输入的下电控制。
- ACODEC_SET_PD_ADCR：右声道输入的下电控制。
- ACODEC_ENABLE_BOOSTL：使能左声道模拟 BOOST 增益。
- ACODEC_ENABLE_BOOSTR：使能右声道模拟 BOOST 增益。
- ACODEC_SET_INPUT_VOL：设置 codec 输入音量。
- ACODEC_GET_INPUT_VOL：获取 codec 输入音量。
- ACODEC_SET_OUTPUT_VOL：设置 codec 输出音量。
- ACODEC_GET_OUTPUT_VOL：获取 codec 输出音量。

ACODEC_SOFT_RESET_CTRL

【描述】

将 codec 恢复为默认设置。

【语法】

```
int ioctl (int fd,
           ACODEC_SOFT_RESET_CTRL);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SOFT_RESET_CTRL	ioctl 号	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

在 AI/AO 设备都禁用时再调用该接口。

【举例】

```
if (ioctl(s32Acodec_Fd, ACODEC_SOFT_RESET_CTRL))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_I2S1_FS

【描述】

设置 I^2S1 接口采样率。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_I2S1_FS,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_I2S1_FS	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败

【需求】

acodec.h

【注意】

无

【举例】

```
ACODEC_FS_E i2s_fs_sel;
i2s_fs_sel = ACODEC_FS_48000;
if (ioctl(s32Acodec_Fd, ACODEC_SET_I2S1_FS, &i2s_fs_sel))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_MIXER_MIC

【描述】

输入声道选择 MICIN 或 LINEIN。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_MIXER_MIC,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_MIXER_MIC	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

默认选择 LINEIN。只能选择 LINEIN。如果选择 MICIN，将不会有信号输入。在使用无源音频信号输入时，需适当的增大模拟增益。

【举例】

```
ACODEC_MIXER_E mixer_mic_ctrl;
mixer_mic_ctrl = ACODEC_MIXER_LINEIN;
if (ioctl(s32Acodec_Fd, ACODEC_SET_MIXER_MIC, &mixer_mic_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_GAIN_MICL

【描述】

左声道输入的模拟增益控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_GAIN_MICL,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_GAIN_MICL	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

在使用无源音频信号输入时，需适当的增大模拟增益。

【举例】

```
unsigned int gain_mac;
gain_mic = 0x18;
if (ioctl(s32Acodec_Fd,  ACODEC_SET_GAIN_MICL, &gain_mic))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_GAIN_MICR

【描述】

右声道输入的模拟增益控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_GAIN_MICR,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_GAIN_MICR	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

在使用无源音频信号输入时，需适当的增大模拟增益。

【举例】

```
unsigned int gain_mic;
gain_mic = 0x18;
if (ioctl(s32Acodec_Fd, ACODEC_SET_GAIN_MICR, &gain_mic))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DACL_VOL

【描述】

左声道输出音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DACL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DACL_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DACR_VOL

【描述】

右声道输出音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DACR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DACR_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADCL_VOL

【描述】

左声道输入音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADCL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADCL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADCL_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADCR_VOL

【描述】

右声道输入音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADCR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADCR_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADCR_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_MICL_MUTE

【描述】

左声道输入静音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_MICL_MUTE,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_MICL_MUTE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int mute_ctrl;
mute_ctrl = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SET_MICL_MUTE, &mute_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_MICR_MUTE

【描述】

右声道输入静音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_MICR_MUTE,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_MICR_MUTE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int mute_ctrl;
ute_ctrl = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SET_MICR_MUTE, &mute_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DACL_MUTE

【描述】

左声道输出静音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DACL_MUTE,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DACL_MUTE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int mute_ctrl;
mute_ctrl = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DACL_MUTE, &mute_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DACR_MUTE

【描述】

右声道输出静音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DACR_MUTE,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DACR_MUTE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int  mute_ctrl;
mute_ctrl = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DACR_MUTE, &mute_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DAC_SOFT_MUTE

【描述】

输出软静音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_DAC_SOFT_MUTE,
           unsigned int  *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DAC_SOFT_MUTE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int  soft_mute_ctrl;
soft_mute_ctrl = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_DAC_SOFT_MUTE, &soft_mute_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DAC_SOFT_UNMUTE

【描述】

输出软撤销静音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_DAC_SOFT_UNMUTE,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DAC_SOFT_UNMUTE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int  soft_mute_ctrl;
soft_nute_ctrl = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_DAC_SOFT_UNMUTE, &soft_mute_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_GET_GAIN_MICL

【描述】

获取模拟左声道输入的增益。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_GAIN_MICL,
           unsigned int  *arg);
```

【参数】

参数名称	描述	输入/输出
Fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_GAIN_MICL	ioctl 号	输入
arg	无符号整型指针	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int gain_mic;
if (ioctl(s32Acodec_Fd,  ACODEC_GET_GAIN_MICL, &gain_mic))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_GET_GAIN_MICR

【描述】

获取模拟右声道输入的增益。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_GAIN_MICR,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_GAIN_MICR	ioctl 号	输入
arg	无符号整型指针	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int gain_mic;
if (ioctl(s32Acodec_Fd, ACODEC_GET_GAIN_MICR, &gain_mic))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_GET_DACL_VOL

【描述】

获取左声道输出的音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_DACL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
if (ioctl(s32Acodec_Fd, ACODEC_GET_DACL_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_GET_DACR_VOL

【描述】

获取右声道输出的音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_DACR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOD_CTRL vol_ctrl;
if (ioctl(s32Acodec_Fd, ACODEC_GET_DACR_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_GET_ADCL_VOL

【描述】

获取左声道输入的音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_ADCL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_ADCL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 结构体指针	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
if (ioctl(s32Acodec_Fd, ACODEC_GET_ADCL_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_GET_ADCR_VOL

【描述】

获取右声道输入的音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_ADCR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_ADCR_VOL	ioctl 号	输入
arg	ACODEC_CTRL 结构体指针	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
if (ioctl(s32Acodec_Fd, ACODEC_GET_ADCR_VOL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_PD_DACL

【描述】

左声道输出的下电控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_PD_DACL,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_PD_DACL	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int pd_ctrl;
pd_ctrl = 0x01;
if (ioctl(s32Acodec_Fd, ACODEC_SET_PD_DACL, &pd_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_PD_DACR

【描述】

右声道输出的下电控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_PD_DACR,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_PD_DACR	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int pd_ctrl;
pd_ctrl = 0x01
if (ioctl(s32Acodec_Fd, ACODEC_SET_PD_DACR, &pd_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_PD_ADCL

【描述】

左声道输入的下电控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_PD_ADCL,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_PD_ADCL	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int pd_ctrl;
pd_ctrl = 0x01;
if (ioctl(s32Acodec_Fd, ACODEC_SET_PD_ADCL, &pd_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_PD_ADCR

【描述】

右声道输入的下电控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_PD_ADCR,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_PD_ADCR	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int pd_ctrl;
pd_ctl = 0x01;
if (ioctl(s32Acodec_Fd, ACODEC_SET_PD_ADCR, &pd_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_ENABLE_BOOSTL

【描述】

使能左声道模拟 BOOST 增益。

【语法】

```
int ioctl (int fd,
           ACODEC_ENABLE_BOOST,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_ENABLE_BOOST	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

使能 boost 功能后，模拟增益会增加 26db。可以联合 ACODEC_SET_GAIN_MICL 和 ACODEC_SET_GAIN_MICR 达到增益细调的目的。

【举例】

无。

【相关接口】

无。

ACODEC_ENABLE_BOOSTR

【描述】

使能右声道模拟 BOOST 增益。

【语法】

```
int ioctl (int fd,
           ACODEC_ENABLE_BOOST,
           unsigned int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_ENABLE_BOOST	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

acodec.h

【注意】

使能 boost 功能后，模拟增益会增加 26db。可以联合 ACODEC_SET_GAIN_MICL 和 ACODEC_SET_GAIN_MICR 达到增益细调的目的。

【举例】

无。

【相关接口】

无。

ACODEC_SET_INPUT_VOL

【描述】

设置 codec 输入音量。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_INPUT_VOL,
           int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_INPUT_VOL	ioctl 号	输入
arg	音量分贝值指针。	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【举例】

```
int db;
db = 0x30;
if (ioctl(s32Acodec_Fd, ACODEC_SET_INPUT_VOL, &db))
{
    printf("ioctl err!\n");
}
```

【相关接口】

ACODEC_GET_INPUT_VOL

ACODEC_GET_INPUT_VOL

【描述】

获取 codec 输入音量。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_INPUT_VOL,
           int *db);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_INPUT_VOL	ioctl 号	输入
arg	音量指针	输出

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

```
int *db;
if (ioctl(s32Acodec_Fd, ACODEC_GET_INPUT_VOL, db))
{
    printf("ioctl err!\n");
}
```

【相关接口】

ACODEC_SET_INPUT_VOL

ACODEC_SET_OUTPUT_VOL

【描述】

设置 codec 输出音量。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_OUTPUT_VOL,
           int *arg);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_OUTPUT_VOL	ioctl 号	输入
arg	音量分贝值指针。	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
int db;
db = 0x30;
if (ioctl(s32Acodec_Fd, ACODEC_SET_OUTPUT_VOL, &db))
{
    printf("ioctl err!\n");
}
```

【相关接口】

ACODEC_GET_OUTPUT_VOL

ACODEC_GET_OUTPUT_VOL

【描述】

获取 codec 输出音量。

【语法】

```
int ioctl (int fd,
           ACODEC_GET_OUTPUT_VOL,
           int *db);
```

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_GET_OUTPUT_VOL	ioctl 号	输入
arg	音量指针	输出

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
int *db;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
if (ioctl(s32Acodec_Fd, ACODEC_GET_OUTPUT_VOL, db))
{
    printf("ioctl err!\n");
}
```

【相关接口】

ACODEC_SET_OUTPUT_VOL

9.3.5.2 扩展功能 cmd

- ACODEC_SEL_DAC_CLK：设置 DAC 和 ADC 的时钟沿同沿或反沿。
- ACODEC_SEL_ANA_MCLK：设置模拟部分和数字部分的时钟沿同沿或反沿。
- ACODEC_DACL_SEL_TRACK：设置 DACL 选择左声道或右声道。
- ACODEC_DACR_SEL_TRACK：设置 DACR 选择左声道或右声道。
- ACODEC_ADCL_SEL_TRACK：设置 ADCL 选择左声道或右声道。
- ACODEC_ADCR_SEL_TRACK：设置 ADCR 选择左声道或右声道。
- ACODEC_SET_DAC_DE_EMPHASIS：DAC 的去加重滤波控制。
- ACODEC_SET_ADC_HP_FILTER：ADC 的高通滤波控制。
- ACODEC_DAC_POP_FREE：DAC 的去 POP 音控制。
- ACODEC_DAC_SOFT_MUTE_RATE：DAC 软静音速率控制。
- ACODEC_DAC_SEL_I2S：DAC I2S 接口选择。
- ACODEC_ADC_SEL_I2S：ADC I2S 接口选择。
- ACODEC_SET_I2S1_DATAWIDTH：I2S1 数据接口位宽。
- ACODEC_SET_I2S2_DATAWIDTH：I2S2 数据接口位宽。
- ACODEC_SET_I2S2_FS：设置 I2S2 接口采样率。
- ACODEC_SET_DACR2DACL_VOL：DACR 到 DACL 音量控制。
- ACODEC_SET_DACL2DACR_VOL：DACL 到 DACR 音量控制。
- ACODEC_SET_ADCL2DACL_VOL：ADCL 到 DACL 音量控制。
- ACODEC_SET_ADCR2DACL_VOL：ADCR 到 DACL 音量控制。
- ACODEC_SET_ADCL2DACR_VOL：DACR 到 DACL 音量控制。
- ACODEC_SET_ADCR2DACR_VOL：ADCR 到 DACR 音量控制。

ACODEC_SEL_DAC_CLK

【描述】

设置 DAC 和 ADC 的时钟沿同沿或反沿。

【语法】

```
int ioctl (int fd,
           ACODEC_SEL_DAC_CLK,
           unsigned int *arg);
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【参数】

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SEL_DAC_CLK	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int clk_sel;
clk_sel = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SEL_DAC_CLK, &clk_sel))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SEL_ANA_MCLK

【描述】

设置模拟部分和数字部分的时钟沿同沿或反沿。

【语法】

```
int ioctl (int fd,
           ACODEC_SEL_ANA_MCLK,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SEL_ANA_MCLK	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int clk_sel;
clk_sel = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SEL_ANA_MCLK, &clk_sel))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DACL_SEL_TRACK

【描述】

设置 DACL 选择左声道或右声道。

【语法】

```
int ioctl (int fd,
           ACODEC_DACL_SEL_TRACK,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DACL_SEL_TRACK	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int track_select;
track_select = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_DACL_SEL_TRACK, &track_select))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DACR_SEL_TRACK

【描述】

设置 DACR 选择左声道或右声道。

【语法】

```
int ioctl (int fd,
           ACODEC_DACR_SEL_TRACK,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DACR_SEL_TRACK	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int track_select;
track_select = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_DACR_SEL_TRACK, &track_select)
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_ADCL_SEL_TRACK

【描述】

设置 ADCL 选择左声道或右声道。

【语法】

```
int ioctl (int fd,
           ACODEC_ADCL_SEL_TRACK,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_ADCL_SEL_TRACK	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int track_select;
track_select = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_ADCL_SEL_TRACK, &track_select))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_ADCR_SEL_TRACK

【描述】

设置 ADCR 选择左声道或右声道。

【语法】

```
int ioctl (int fd,
           ACODEC_ADCR_SEL_TRACK,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_ADCR_SEL_TRACK	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int track_select;
track_select = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_ADCR_SEL_TRACK, &track_select))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DAC_DE_EMPHASIS

【描述】

DAC 的去加重滤波控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DAC_DE_EMPHASIS,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DAC_DE_EMPHASIS	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int dac_deemphasis;
dac_deemphasis = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DAC_DE_EMPHASIS, &dac_deemphasis))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADC_HP_FILTER

【描述】

ADC 的高通滤波控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADC_HP_FILTER,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADC_HP_FILTER	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int adc_hpf;
adc_hpf = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADC_HP_FILTER, &adc_hpf))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DAC_POP_FREE

【描述】

DAC 的去 POP 音控制。

【语法】

```
int ioctl (int fd,
           ACODEC_DAC_POP_FREE,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DAC_POP_FREE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int dac_pop_free;
dac_pop_free = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_DAC_POP_FREE, &dac_pop_free))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DAC_SOFT_MUTE_RATE

【描述】

DAC 软静音速率控制。

【语法】

```
int ioctl (int fd,
           ACODEC_DAC_POP_FREE,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DAC_SOFT_MUTE_RATE	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int soft_mute_rate;
soft_mute_rate = 0x1;
if (ioctl(s32Acodec_Fd, ACODEC_DAC_SOFT_MUTE_RATE, &soft_mute_rate))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_DAC_SEL_I2S

【描述】

DAC I^2S 接口选择。

【语法】

```
int ioctl (int fd,
           ACODEC_DAC_SEL_I2S,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_DAC_SEL_I2S	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int i2s_sel;
i2s_sel = 0x0;
if (ioctl(s32Acodec_Fd, ACODEC_DAC_SOFT_MUTE_RATE, &i2s_sel))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_ADC_SEL_I2S

【描述】

ADC I^2S 接口选择。

【语法】

```
int ioctl (int fd,
           ACODEC_ADC_SEL_I2S,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_ADC_SEL_I2S	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int i2s_sel;
i2s_sel = 0x0;
if (ioctl(s32Acodec_Fd, ACODEC_ADC_SEL_I2S, &i2s_sel))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_I2S1_DATAWIDTH

【描述】

I^2S1 数据接口位宽。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_I2S1_DATAWIDTH,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_I2S1_DATAWIDTH	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int i2s_datawidth;
i2s_datawidth = 0x0;
if (ioctl(s32Acodec_Fd, ACODEC_SET_I2S1_DATAWIDTH, &i2s_datawidth))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_I2S2_DATAWIDTH

【描述】

I^2S2 数据接口位宽。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_I2S2_DATAWIDTH,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_I2S2_DATAWIDTH	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
unsigned int i2s_datawidth;
i2s_datawidth = 0x0;
if (ioctl(s32Acodec_Fd, ACODEC_SET_I2S2_DATAWIDTH, &i2s_datawidth))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_I2S2_FS

【描述】

设置 I^2S2 接口采样率。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_I2S2_FS,
           unsigned int *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_I2S2_FS	ioctl 号	输入
arg	无符号整型指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_FS_E i2s_fs_sel;
i2s_fs_sel = ACODEC_FS_48000;
if (ioctl(s32Acodec_Fd, ACODEC_SET_I2S2_FS, &i2s_fs_sel))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DACR2DACL_VOL

【描述】

DACR 到 DACL 音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DACR2DACL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DACR2DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DACR2DACL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_DACL2DACR_VOL

【描述】

DACL 到 DACR 音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_DACL2DACR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_DACL2DACR_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_DACL2DACR, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADCL2DACL_VOL

【描述】

ADCL 到 DACL 音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADCL2DACL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADCL2DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADCL2DACL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADCR2DACL_VOL

【描述】

ADCR 到 DACL 音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADCR2DACL_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADCR2DACL_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADCR2DACL, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADCL2DACR_VOL

【描述】

DACR 到 DACL 音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADCL2DACR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADCL2DACR_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADCL2DACR, &vol_ctrl))
{
    printf("ioctl err!\n");
}
```

【相关接口】

无。

ACODEC_SET_ADCR2DACR_VOL

【描述】

ADCR 到 DACR 音量控制。

【语法】

```
int ioctl (int fd,
           ACODEC_SET_ADCR2DACR_VOL,
           ACODEC_VOL_CTRL *arg);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
fd	Audio Codec 设备文件描述符	输入
ACODEC_SET_ADCR2DACR_VOL	ioctl 号	输入
arg	ACODEC_VOL_CTRL 指针	输入

【返回值】

返回值	描述
0	成功
非 0	失败

【需求】

acodec.h

【注意】

无。

【举例】

```
ACODEC_VOL_CTRL vol_ctrl;
vol_ctrl. vol_ctrl_mute = 0x0;
vol_ctrl. vol_ctrl = 0x06;
if (ioctl(s32Acodec_Fd, ACODEC_SET_ADCR2DACR, &vol_ctrl))
{
printf("ioctl err!\n");
}
```

【相关接口】

无。

9.4 数据类型

9.4.1 音频输入输出

音频输入输出相关数据类型、数据结构定义如下：

- SIO_MAX_NUM：定义音频输入/输出设备的最大个数。
- AIO_MAX_CHN_NUM：定义音频输入/输出设备的最大通道数。
- AUDIO_SAMPLE_RATE_E：定义音频采样率。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- AUDIO_BIT_WIDTH_E：定义音频采样精度。
- AIO_MODE_E：定义音频输入输出工作模式。
- AUDIO_SOUND_MODE_E：定义音频声道模式。
- AIO_ATTR_S：定义音频输入输出设备属性结构体。
- AUDIO_FRAME_S：定义音频帧数据结构体。
- AEC_FRAME_S：定义回声抵消参考帧信息结构体。
- AI_CHN_PARAM_S：定义通道参数结构体。
- AUDIO_STREAM_S：定义音频码流结构体。
- AUDIO_RESAMPLE_TYPE_E：定义音频重采样类型。
- AUDIO_RESAMPLE_ATTR_S：定义音频重采样属性配置结构体。
- AUDIO_RESAMPLE_ATTR_EX_S：定义音频高级重采样属性配置结构体。
- AO_CHN_STATE_S：音频输出通道的数据缓存状态结构体。
- G726_BPS_E：定义 G.726 编解码协议速率。
- ADPCM_TYPE_E：定义 ADPCM 编解码协议类型。
- AUDIO_AEC_MODE_E：定义音频回声抵消模式。
- AI_AEC_CONFIG_S：定义音频回声抵消配置信息结构体。
- AI_ANR_CONFIG_S：定义音频降噪功能配置信息结构体。
- AI_ALC_CONFIG_S：定义音频自动电平控制配置信息结构体。
- AUDIO_HPF_FREQ_E：定义高通滤波截止频率。
- AI_HPF_CONFIG_S：定义高通滤波参数结构体。
- AI_HPF_ATTR_S：定义高通滤波配置信息结构体。
- AI_VQE_CONFIG_S：Vqe 属性结构体。

SIO_MAX_NUM

【说明】

定义音频输入/输出设备的最大个数。

【定义】

针对 Hi3531：

```
#define SIO_MAX_NUM        6
```

针对 Hi3532：

```
#define SIO_MAX_NUM        5
```

针对 Hi3521/Hi3520A：

```
#define SIO_MAX_NUM        4
```

针对 Hi3518/Hi3516C:

```
#define SIO_MAX_NUM        1
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

针对 Hi3520D/Hi3515A/Hi3515C:

```
#define SIO_MAX_NUM        2
```

【注意事项】

无。

【相关数据类型及接口】

无。

AIO_MAX_CHN_NUM

【说明】

定义音频输入/输出设备的最大通道数。

【定义】

针对 Hi3531/Hi3532：

```
#define AIO_MAX_CHN_NUM     16
```

针对 Hi3521/Hi3520A：

```
#define AIO_MAX_CHN_NUM     16
```

针对 Hi3518/Hi3516C:

```
#define AIO_MAX_CHN_NUM       2
```

针对 Hi3520D/Hi3515A/Hi3515C：

```
#define AIO_MAX_CHN_NUM     16
```

【注意事项】

无。

【相关数据类型及接口】

无。

AUDIO_SAMPLE_RATE_E

【说明】

定义音频采样率。

【定义】

```
typedef enum hiAUDIO_SAMPLE_RATE_E
{
    AUDIO_SAMPLE_RATE_8000  =8000,      /* 8kHz sampling rate */
    AUDIO_SAMPLE_RATE_12000 =12000,     /* 12kHz sampling rate */
    AUDIO_SAMPLE_RATE_11025 =11025,     /* 11.025kHz sampling rate */
    AUDIO_SAMPLE_RATE_16000 =16000,     /* 16kHz sampling rate */
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    AUDIO_SAMPLE_RATE_22050 =22050,     /* 22.050kHz sampling rate */
    AUDIO_SAMPLE_RATE_24000 =24000,     /* 24kHz sampling rate */
    AUDIO_SAMPLE_RATE_32000 =32000,     /* 32kHz sampling rate */
    AUDIO_SAMPLE_RATE_44100 =44100,     /* 44.1kHz sampling rate */
    AUDIO_SAMPLE_RATE_48000 =48000,     /* 48kHz sampling rate */
}AUDIO_SAMPLE_RATE_E;
```

【成员】

成员名称	描述
AUDIO_SAMPLE_RATE_8000	8kHz 采样率。
AUDIO_SAMPLE_RATE_12000	12kHz 采样率。
AUDIO_SAMPLE_RATE_11025	11.025kHz 采样率。
AUDIO_SAMPLE_RATE_16000	16kHz 采样率。
AUDIO_SAMPLE_RATE_22050	22.050kHz 采样率。
AUDIO_SAMPLE_RATE_24000	24kHz 采样率。
AUDIO_SAMPLE_RATE_32000	32kHz 采样率。
AUDIO_SAMPLE_RATE_44100	44.1kHz 采样率。
AUDIO_SAMPLE_RATE_48000	48kHz 采样率。

【注意事项】

这里枚举值不是从 0 开始，而是与实际的采样率值相同。

【相关数据类型及接口】

AIO_ATTR_S

AUDIO_BIT_WIDTH_E

【说明】

定义音频采样精度。

【定义】

```
typedef enum hiAUDIO_BIT_WIDTH_E
{
    AUDIO_BIT_WIDTH_8   =0,     /* 8bit/sample */
    AUDIO_BIT_WIDTH_16  =1,     /* 16bit/sample */
    AUDIO_BIT_WIDTH_32  =2,     /* 32bit/sample */
    AUDIO_BIT_WIDTH_BUTT,
}AUDIO_BIT_WIDTH_E;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
AUDIO_BIT_WIDTH_8	采样精度为 8bit 位宽。
AUDIO_BIT_WIDTH_16	采样精度为 16bit 位宽。
AUDIO_BIT_WIDTH_32	采样精度为 32bit 位宽。

【注意事项】

- 目前不支持 32bit 位宽。
- Hi3518/Hi3516C 只支持 16bit 位宽。

【相关数据类型及接口】

AIO_ATTR_S

AIO_MODE_E

【说明】

定义音频输入输出设备工作模式。

【定义】

```
typedef enum hiAIO_MODE_E
{
    AIO_MODE_I2S_MASTER = 0,        /* I2S master mode */
    AIO_MODE_I2S_SLAVE  = 1,        /* I2S slave mode */
    AIO_MODE_PCM_SLAVE_STD,         /* SIO PCM slave standard mode */
    AIO_MODE_PCM_SLAVE_NSTD,        /* SIO PCM slave non-standard mode */
    AIO_MODE_PCM_MASTER_STD,       /* SIO PCM master standard mode */
    AIO_MODE_PCM_MASTER_NSTD,      /* SIO PCM master non-standard mode */
    AIO_MODE_BUTT
}AIO_MODE_E;
```

【成员】

成员名称	描述
AIO_MODE_I2S_MASTER	I^2S 主模式。
AIO_MODE_I2S_SLAVE	I^2S 从模式。
AIO_MODE_PCM_SLAVE_STD	PCM 从模式（标准协议）
AIO_MODE_PCM_SLAVE_NSTD	PCM 从模式（自定义协议）
AIO_MODE_PCM_MASTER_STD	PCM 主模式（标准协议）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
AIO_MODE_PCM_MASTER_NSTD	PCM 主模式（自定义协议）

【注意事项】

- 针对 Hi3531/Hi3532

  SIO0～SIO4 选择主模式时，它只提供用于时序同步的帧同步时钟和位流时钟，不提供 MCLK，此时如果 Audio Codec 使用外接的晶振作为工作时钟，这样可能导致声音失真，所以推荐使用从模式。而 SIO5 必须使用 I^2S 主模式。

- 针对 Hi3521/Hi3520A

  SIO0～SIO2 选择主模式时，它只提供用于时序同步的帧同步时钟和位流时钟，不提供 MCLK，此时如果 Audio Codec 使用外接的晶振作为工作时钟，这样可能导致声音失真，所以推荐使用从模式。而 SIO3 必须使用 I^2S 主模式。

- 针对 Hi3518//Hi3516C

  Hi3518/Hi3516C 使用内部 Audio Codec，使用 I^2S 主模式。

- 针对 Hi3520D/Hi3515A/Hi3515C

  Hi3520D/Hi3515A/Hi3515C 使用 AIO 接口做为音频输入输出接口，AIO 接口中的 AI 设备 0 或 AO 设备 0 为从模式时，AI 设备 0 不复用 AO 设备 0 的帧同步时钟配置及位流时钟配置。AO 设备 1 用于对接 HDMI，必须使用 I^2S 主模式。

【相关数据类型及接口】

AIO_ATTR_S

AUDIO_SOUND_MODE_E

【说明】

定义音频声道模式。

【定义】

```
typedef enum hiAIO_SOUND_MODE_E
{
    AUDIO_SOUND_MODE_MOMO   =0,     /*momo*/
    AUDIO_SOUND_MODE_STEREO =1,     /*stereo*/
    AUDIO_SOUND_MODE_BUTT
}AUDIO_SOUND_MODE_E;
```

【成员】

成员名称	描述
AUDIO_SOUND_MODE_MOMO	单声道。
AUDIO_SOUND_MODE_STEREO	双声道。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

对于双声道模式，只应对左声道（即编号小于设备属性中通道数 u32ChnCnt 一半的通道）进行操作，SDK 内部会自动对右声道也进行相应的操作。

【相关数据类型及接口】

AIO_ATTR_S

AIO_ATTR_S

【说明】

定义音频输入输出设备属性结构体。

【定义】

```
typedef struct hiAIO_ATTR_S
{
    AUDIO_SAMPLE_RATE_E enSamplerate;       /*sample rate*/
    AUDIO_BIT_WIDTH_E   enBitwidth;         /*bitwidth*/
    AIO_MODE_E          enWorkmode;         /*master or slave mode*/
    AUDIO_SOUND_MODE_E  enSoundmode;        /*mono or stereo*/
    HI_U32              u32EXFlag;          /*expand 8bit to 16bit */
    HI_U32              u32FrmNum;          /*frame num in buffer*/
    HI_U32              u32PtNumPerFrm;     /*number of samples*/
    HI_U32              u32ChnCnt;
    HI_U32              u32ClkSel;
}AIO_ATTR_S;
```

【成员】

成员名称	描述
enSamplerate	音频采样率（从模式下，此参数不起作用）。 静态属性。
enBitwidth	音频采样精度（从模式下，此参数必须和音频 AD/DA 的采样精度匹配）。 静态属性。
enWorkmode	音频输入输出工作模式。 静态属性。
enSoundmode	音频声道模式。 静态属性。
u32EXFlag	8bit 到 16bit 扩展标志（只对 AI 采样精度为 8bit 时有效）。 取值范围：{0, 1}。 • 0：不扩展。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
	• 1：扩展。 静态属性。
u32FrmNum	缓存帧数目。 取值范围：[2, MAX_AUDIO_FRAME_NUM]。 静态属性。
u32PtNumPerFrm	每帧的采样点个数。 取值范围：G711、G726、ADPCM_DVI4 编码时取值为 80、160、240、320、480；ADPCM_IMA 编码时取值为 81、161、241、321、481。 静态属性。
u32ChnCnt	支持的通道数目。 取值：2、4、8、16。
u32ClkSel	对于使用 SIO 接口的芯片，配置用于对讲 SIO 的发送时钟和接收时钟是否复用，其他 SIO 不用关注；对于使用 AIO 接口的芯片，配置 AI 设备 0 是否复用 AO 设备 0 的帧同步时钟配置及位流时钟配置，AO 设备 0 为主模式时有效。 取值：0、1。 0：不复用； 1：复用。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_AI_SetPubAttr

AUDIO_FRAME_S

【说明】

定义音频帧结构体。

【定义】

```
typedef struct hiAUDIO_FRAME_S
{
    AUDIO_BIT_WIDTH_E   enBitwidth;  /*audio frame bitwidth*/
    AUDIO_SOUND_MODE_E  enSoundmode; /*audio frame momo or stereo mode*/
    HI_VOID             *pVirAddr[2];
    HI_U32              u32PhyAddr[2];
    HI_U64              u64TimeStamp;/*audio frame timestamp*/
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32              u32Seq;      /*audio frame seq*/
    HI_U32              u32Len;      /*data lenth per channel in frame*/
    HI_U32              u32PoolId[2];
}AUDIO_FRAME_S;
```

【成员】

成员名称	描述
enBitwidth	音频采样精度。
enSoundmode	音频声道模式。
pVirAddr[2]	音频帧数据虚拟地址。
u32PhyAddr[2]	音频帧数据物理地址。
u64TimeStamp	音频帧时间戳。 以 μs 为单位。
u32Seq	音频帧序号。
u32Len	音频帧长度。 以 byte 为单位。
u32PoolId[2]	音频帧缓存池 ID。

【注意事项】

- u32Len（音频帧长度）指单个声道的数据长度。
- 单声道数据直接存放，采样点数为 ptnum，长度为 len；立体声数据按左右声道分开存放，先存放采样点为 ptnum、长度为 len 的左声道数据，然后存放采样点为 ptnum，长度为 len 的右声道数据。

【相关数据类型及接口】

无。

AEC_FRAME_S

【说明】

定义音频回声抵消参考帧信息结构体。

【定义】

```
typedef struct hiAEC_FRAME_S
{
    AUDIO_FRAME_S   stRefFrame;     /* aec reference audio frame */
    HI_BOOL         bValid;         /* whether frame is valid */
}AEC_FRAME_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
stRefFrame	回声抵消参考帧结构体。
bValid	参考帧有效的标志。 取值范围： HI_TRUE：参考帧有效。 HI_FALSE：参考帧无效，无效时不能使用此参考帧进行回声抵消。

【注意事项】

无。

【相关数据类型及接口】

无。

AI_CHN_PARAM_S

【说明】

定义通道参数结构体。

【定义】

```
typedef struct hiAI_CHN_PARAM_S
{
    HI_U32 u32UsrFrmDepth;
    HI_S32 s32Rev;
} AI_CHN_PARAM_S;
```

【成员】

成员名称	描述
u32UsrFrmDepth	音频帧缓存深度。
s32Rev	保留。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

AUDIO_STREAM_S

【说明】

定义音频码流结构体。

【定义】

```
typedef struct hiAUDIO_STREAM_S
{
    HI_U8 *pStream;         /* the virtual address of stream */
    HI_U32 u32PhyAddr;      /* the physics address of stream */
    HI_U32 u32Len;          /* stream lenth, by bytes */
    HI_U64 u64TimeStamp;    /* frame time stamp*/
    HI_U32 u32Seq;          /* frame seq,if stream is not a valid
frame,u32Seq is 0*/

} AUDIO_STREAM_S;
```

【成员】

成员名称	描述
pStream	音频码流数据指针。
u32PhyAddr	音频码流的物理地址。
u32Len	音频码流长度。以 byte 为单位。
u64TimeStamp	音频码流时间戳。
u32Seq	音频码流序号。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_AENC_GetStream

AUDIO_RESAMPLE_TYPE_E

【说明】

定义音频重采样类型。

【定义】

```
typedef enum hiAUDIO_RESAMPLE_TYPE_E
{
    AUDIO_RESAMPLE_1X2 = 0x1,
    AUDIO_RESAMPLE_2X1 = 0x2,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    AUDIO_RESAMPLE_1X4 = 0x3,
    AUDIO_RESAMPLE_4X1 = 0x4,
    AUDIO_RESAMPLE_1X6 = 0x5,
    AUDIO_RESAMPLE_6X1 = 0x6,
    AUDIO_RESAMPLE_BUTT
} AUDIO_RESAMPLE_TYPE_E;
```

【成员】

成员名称	描述
AUDIO_RESAMPLE_1X2	2 倍重采样。
AUDIO_RESAMPLE_2X1	1/2 重采样。
AUDIO_RESAMPLE_1X4	4 倍重采样。
AUDIO_RESAMPLE_4X1	1/4 重采样。
AUDIO_RESAMPLE_1X6	6 倍重采样。
AUDIO_RESAMPLE_6X1	1/6 重采样。

【注意事项】

无。

【相关数据类型及接口】

- AUDIO_RESAMPLE_ATTR_S
- HI_MPI_AO_EnableReSmp

AUDIO_RESAMPLE_ATTR_S

【说明】

定义音频重采样属性配置结构体。

【定义】

```
typedef struct hiAUDIO_RESAMPLE_ATTR_S
{
    HI_U32                u32InPointNum;
    AUDIO_SAMPLE_RATE_E     enInSampleRate;
    AUDIO_RESAMPLE_TYPE_E   enReSampleType;
} AUDIO_RESAMPLE_ATTR_S;
```

【成员】

成员名称	描述
u32InPointNum	输入的每帧采样点个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enInSampleRate	输入的采样率。
enReSampleType	重采样类型。

【注意事项】

无。

【相关数据类型及接口】

- AUDIO_SAMPLE_RATE_E
- AUDIO_RESAMPLE_TYPE_E

AUDIO_RESAMPLE_ATTR_EX_S

【说明】

定义音频高级重采样属性配置结构体。

【定义】

```
typedef struct hiAUDIO_RESAMPLE_ATTR_EX_S
{
    HI_U32                u32InPointNum;
    AUDIO_SAMPLE_RATE_E     enInSampleRate;
    AUDIO_SAMPLE_RATE_E     enOutSampleRate;
} AUDIO_RESAMPLE_ATTR_EX_S;
```

【成员】

成员名称	描述
u32InPointNum	输入的每帧采样点个数。
enInSampleRate	输入的采样率。
enOutSampleRate	输出的采样率。

【注意事项】

无。

【相关数据类型及接口】

AUDIO_SAMPLE_RATE_E

AO_CHN_STATE_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

音频输出通道的数据缓存状态结构体。

【定义】

```
typedef struct hiAO_CHN_STATE_S
{
    HI_U32              u32ChnTotalNum;
    HI_U32              u32ChnFreeNum;
    HI_U32              u32ChnBusyNum;
} AO_CHN_STATE_S;
```

【成员】

成员名称	描述
u32ChnTotalNum	输出通道总的缓存块数。
u32ChnFreeNum	空闲缓存块数。
u32ChnBusyNum	被占用缓存块数。

【注意事项】

无。

【相关数据类型及接口】

HI_MPI_AO_QueryChnStat

G726_BPS_E

【说明】

定义 G.726 编解码协议速率。

【定义】

```
typedef enum hiG726_BPS_E
{
    G726_16K = 0,
    G726_24K,
    G726_32K,
    G726_40K,
    MEDIA_G726_16K,     /* G726 16kbit/s for ASF */
    MEDIA_G726_24K,     /* G726 24kbit/s for ASF */
    MEDIA_G726_32K,     /* G726 32kbit/s for ASF */
    MEDIA_G726_40K,     /* G726 40kbit/s for ASF */
    G726_BUTT,
}G726_BPS_E;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
G726_16K	16kbit/s G.726，请参见“RFC3551 文档 4.5.4 G72616”。
G726_24K	24kbit/s G.726，请参见“RFC3551 文档 4.5.4 G72624”。
G726_32K	32kbit/s G.726，请参见“RFC3551 文档 4.5.4 G72632”。
G726_40K	40kbit/s G.726，请参见“RFC3551 文档 4.5.4 G72640”。
MEDIA_G726_16K	G726 16kbit/s for ASF。
MEDIA_G726_24K	G726 24kbit/s for ASF。
MEDIA_G726_32K	G726 32kbit/s for ASF。
MEDIA_G726_40K	G726 40kbit/s for ASF。

【注意事项】

无。

【相关数据类型及接口】

无。

ADPCM_TYPE_E

【说明】

定义 ADPCM 编解码协议类型。

【定义】

```
typedef enum hiADPCM_TYPE_E
{
    ADPCM_TYPE_DVI4 = 0,
    ADPCM_TYPE_IMA,
    ADPCM_TYPE_BUTT,
}ADPCM_TYPE_E;
```

【成员】

成员名称	描述
ADPCM_TYPE_DVI4	32kbit/s ADPCM(DVI4)。
ADPCM_TYPE_IMA	32kbit/s ADPCM(IMA)。

【注意事项】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关数据类型及接口】

无。

AUDIO_AEC_MODE_E

【说明】

定义音频回声抵消模式。

【定义】

```
typedef enum hiAUDIO_AEC_MODE_E
{
    AUDIO_AEC_MODE_RECEIVER  = 0,
    AUDIO_AEC_MODE_SPEAKER   = 1,
    AUDIO_AEC_MODE_HEADPHONE = 2,
    AUDIO_AEC_MODE_BUTT
} AUDIO_AEC_MODE_E;
```

【成员】

成员名称	描述
AUDIO_AEC_MODE_RECEIVER	听筒模式
AUDIO_AEC_MODE_SPEAKER	扬声器模式
AUDIO_AEC_MODE_HEADPHONE	耳机模式

【注意事项】

默认配置为扬声器模式。

【相关数据类型及接口】

无。

AI_AEC_CONFIG_S

【说明】

定义音频回声抵消配置信息结构体。

【定义】

```
typedef struct hiAI_AEC_CONFIG_S
{
    AUDIO_AEC_MODE_E    enAecMode;      /* AEC mode, default is speaker,
0:receiver  1:speaker  2:headphone */
    HI_S32              s32Reserved;
} AI_AEC_CONFIG_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
enAecMode	回声抵消处理模式选择： 0：听筒模式 1：扬声器模式 2：耳机模式
s32Reserved	保留，未使用。

【注意事项】

无。

【相关数据类型及接口】

无。

AI_ALC_CONFIG_S

【说明】

定义音频自动电平控制配置信息结构体。

【定义】

```
typedef struct hiAI_ALC_CONFIG_S
{
    HI_S32 s32MaxLev;        /*s32MaxLev: [-23dBm0, -4dBm0]。default: -
4dBm0*/
    HI_S32 s32MinLev;        /*s32MinLev:  [-23dBm0, -4dBm0]。default: -
16dBm0*/
    HI_U32 u32MaxGain;       /*u32MaxGain: [3dB,12dB]。default: 12dB*/
} AI_ALC_CONFIG_S;
```

【成员】

成员名称	描述
s32MaxLev	最高目标电平。 取值范围：[-23dBm0, -4dBm0]。默认值为-4dBm0。
s32MinLev	最低目标电平。 取值范围：[-23dBm0, -4dBm0]。默认值为-16dBm0。
u32MaxGain	最大增益值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
	取值范围：[3dB,12dB]。默认值为 12dB。

【注意事项】

最低目标电平不能大于最高目标电平。

【相关数据类型及接口】

无。

AUDIO_HPF_FREQ_E

【说明】

定义高通滤波截止频率。

【定义】

```
typedef enum hiAUDIO_HPF_FREQ_E
{
    AUDIO_HPF_FREQ_80   = 80,    /* 80Hz */
    AUDIO_HPF_FREQ_120  = 120,   /* 120Hz */
    AUDIO_HPF_FREQ_150  = 150,   /* 150Hz */
    AUDIO_HPF_FREQ_BUTT,
} AUDIO_HPF_FREQ_E;
```

【成员】

成员名称	描述
AUDIO_HPF_FREQ_80	截止频率为 80HZ。
AUDIO_HPF_FREQ_120	截止频率为 120HZ。
AUDIO_HPF_FREQ_150	截止频率为 150HZ。

【注意事项】

建议使用 AUDIO_HPF_FREQ_150 参数，如果对低频要求比较高，可以设置 80 和 120。

【相关数据类型及接口】

无。

AI_HPF_CONFIG_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义高通滤波参数。

【定义】

```
typedef struct hiAI_HPF_CONFIG_S
{
    AUDIO_HPF_FREQ_E emHpfFreq; /*freq to be processed*/
} AI_HPF_CONFIG_S;;
```

【成员】

成员名称	描述
AUDIO_HPF_FREQ_E	高通滤波截止频率。

【注意事项】

无。

【相关数据类型及接口】

AUDIO_HPF_FREQ_E

AI_HPF_ATTR_S

【说明】

定义音频输入高通滤波配置信息结构体。

【定义】

```
typedef struct hiAI_HPF_ATTR_S
{
    AI_HPF_CONFIG_S stHpfCfg;
    HI_S32          bHpfOpen;
} AI_HPF_ATTR_S;
```

【成员】

成员名称	描述
stHpfCfg	高通滤波参数。
bHpfOpen	高通滤波功能是否使能标志。

【注意事项】

无。

【相关数据类型及接口】

AI_HPF_CONFIG_S

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

AI_ANR_CONFIG_S

【说明】

定义音频降噪功能配置信息结构体。

【定义】

```
typedef struct hiAI_ANR_CONFIG_S
{
    HI_S32 s32Reserved;
} AI_ANR_CONFIG_S;
```

【成员】

成员名称	描述
s32Reserved	保留，未使用。

【注意事项】

无。

【相关数据类型及接口】

无。

AI_VQE_CONFIG_S

【说明】

定义音频输入声音质量增强配置信息结构体。

【定义】

```
typedef struct hiAI_VQE_CONFIG_S
{
    HI_S32            bAecOpen;
    HI_S32            bAnrOpen;
    HI_S32            bAlcOpen;
    HI_S32            s32SampleRate;
    HI_S32            s32FrameSample;
    AI_AEC_CONFIG_S stAecCfg;
    AI_ANR_CONFIG_S stAnrCfg;
    AI_ALC_CONFIG_S stAlcCfg;
} AI_VQE_CONFIG_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bAecOpen	回声抵消功能是否使能标志。
bAnrOpen	降噪功能是否使能标志。
bAlcOpen	自动电平控制功能是否使能标志。
s32SampleRate	采样频率。 取值范围：8KHz/11KHz/16KHz。默认值为 8KHz。
s32FrameSample	VQE 的帧长，即采样点数目。 采样频率为 8KHz，VQE 帧长范围为 80/160/240/320/400/480/960，默认值为 160。 采样频率为 16KHz，VQE 帧长范围为 160/320/480/960，默认值为 160。
stAecCfg	回声抵消相关配置信息。
stAnrCfg	降噪功能相关配置信息。
stAlcCfg	自动电平控制相关配置信息。

【注意事项】

无。

【相关数据类型及接口】

- AI_AEC_CONFIG_S
- AI_ANR_CONFIG_S
- AI_ALC_CONFIG_S

AUDIO_SAVE_FILE_INFO_S

【说明】

定义音频保存文件功能配置信息结构体。

【定义】

```
typedef struct hiAUDIO_SAVE_FILE_INFO_S
{
    HI_BOOL  bCfg;
    HI_CHAR  aFilePath[256];
    HI_U32 u32FileSize;  /*in KB*/
} AUDIO_SAVE_FILE_INFO_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
bCfg	配置使能开关。
aFilePath[256]	音频文件保存路径
u32FileSize	文件大小，取值范围[1,8192]KB。

【注意事项】

无。

【相关数据类型及接口】

无。

9.4.2 音频编码

音频编码相关数据类型、数据结构定义如下：

- AENC_MAX_CHN_NUM：定义音频编码通道的最大个数。
- AENC_ATTR_G711_S：定义 G.711 编码协议属性结构体。
- AENC_ATTR_G726_S：定义 G.726 编码协议属性结构体。
- AENC_ATTR_ADPCM_S：定义 ADPCM 编码协议属性结构体。
- AENC_CHN_ATTR_S：定义音频编码通道属性结构体。
- AENC_ENCODER_S：定义编码器属性结构体。

AENC_MAX_CHN_NUM

【说明】

定义音频编码通道的最大个数。

【定义】

```
#define AENC_MAX_CHN_NUM        32
```

【注意事项】

无

【相关数据类型及接口】

无

AENC_ATTR_G711_S

【说明】

定义 G.711 编码协议属性结构体。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef struct hiAENC_ATTR_G711_S
{
    HI_U32 resv;
}AENC_ATTR_G711_S;
```

【成员】

成员名称	描述
resv	待扩展用（目前暂未使用）。

【注意事项】

无

【相关数据类型及接口】

无

AENC_ATTR_G726_S

【说明】

定义 G.726 编码协议属性结构体。

【定义】

```
typedef struct hiAENC_ATTR_G726_S
{
    G726_BPS_E  enG726bps;
}AENC_ATTR_G726_S;
```

【成员】

成员名称	描述
enG726bps	G.726 协议码率。

【注意事项】

无。

【相关数据类型及接口】

G726_BPS_E

AENC_ATTR_ADPCM_S

【说明】

定义 ADPCM 编码协议属性结构体。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct hiAENC_ATTR_ADPCM_S
{
   ADPCM_TYPE_E enADPCMType;
}AENC_ATTR_ADPCM_S;
```

【成员】

成员名称	描述
enADPCMType	ADPCM 类型。

【注意事项】

无。

【相关数据类型及接口】

ADPCM_TYPE_E

AENC_CHN_ATTR_S

【说明】

定义音频编码通道属性结构体。

【定义】

```
typedef struct hiAENC_CHN_ATTR_S
{
    PAYLOAD_TYPE_E  enType;
    HI_U32  u32BufSize; /*buffer size，以帧为单位，[2～MAX_AUDIO_FRAME_NUM]*/
    HI_VOID *pValue;
}AENC_CHN_ATTR_S;
```

【成员】

成员名称	描述
enType	音频编码协议类型。 静态属性。
u32BufSize	音频编码缓存大小。 取值范围：[2, MAX_AUDIO_FRAME_NUM]，以帧为单位。 静态属性。
pValue	具体协议属性指针。 静态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

无。

AENC_ENCODER_S

【说明】

定义编码器属性结构体。

【定义】

```
typedef struct hiAENC_ENCODER_S
{
    PAYLOAD_TYPE_E  enType;
    HI_U32     u32MaxFrmLen;
    HI_CHAR    aszName[16];
    HI_S32     (*pfnOpenEncoder)(HI_VOID *pEncoderAttr, HI_VOID
**ppEncoder);
    HI_S32     (*pfnEncodeFrm)(HI_VOID *pEncoder, const AUDIO_FRAME_S
*pstData, HI_U8 *pu8Outbuf,HI_U32 *pu32OutLen);
    HI_S32     (*pfnCloseEncoder)(HI_VOID *pEncoder);
} AENC_ENCODER_S;
```

【成员】

成员名称	描述
enType	编码协议类型。
u32MaxFrmLen	最大码流长度。
aszName	编码器名称。
pfnOpenEncoder	打开编码器的函数指针。
pfnEncodeFrm	进行编码的函数指针。
pfnCloseEncoder	关闭编码器的函数指针。

【注意事项】

请参见《音频组件 API 参考》。

【相关数据类型及接口】

HI_MPI_AENC_RegeisterEncoder

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

9.4.3 音频解码

音频解码相关数据类型、数据结构定义如下：

- MAX_AUDIO_FRAME_NUM：定义最大音频解码缓存帧数。
- ADEC_MAX_CHN_NUM：定义音频解码通道的最大个数。
- ADEC_ATTR_G711_S：定义 G.711 解码协议属性结构体。
- ADEC_ATTR_G726_S：定义 G.726 解码协议属性结构体。
- ADEC_ATTR_ADPCM_S：定义 ADPCM 编码协议属性结构体。
- ADEC_MODE_E：定义解码方式。
- ADEC_CHN_ATTR_S：定义解码通道属性结构体。
- ADEC_DECODER_S：定义解码器属性结构体。
- AUDIO_FRAME_INFO_S：定义解码帧信息结构体指针。

MAX_AUDIO_FRAME_NUM

【说明】

定义最大音频解码缓存帧数。

【定义】

```
#define MAX_AUDIO_FRAME_NUM       50
```

【注意事项】

无

【相关数据类型及接口】

无

ADEC_MAX_CHN_NUM

【说明】

定义音频解码通道的最大个数。

【定义】

```
#define ADEC_MAX_CHN_NUM        32
```

【注意事项】

无

【相关数据类型及接口】

无

ADEC_ATTR_G711_S

【说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

定义 G.711 解码协议属性结构体。

【定义】

```
typedef struct hiADEC_ATTR_G711_S
{
    HI_U32  resv;
}ADEC_ATTR_G711_S;
```

【成员】

成员名称	描述
resv	待扩展用（目前暂未使用）。

【注意事项】

无。

【相关数据类型及接口】

无。

ADEC_ATTR_G726_S

【说明】

定义 G.726 解码协议属性结构体。

【定义】

```
typedef struct hiADEC_ATTR_G726_S
{
    G726_BPS_E  enG726bps;
}ADEC_ATTR_G726_S;
```

【成员】

成员名称	描述
enG726bps	G.726 协议码率。

【注意事项】

无。

【相关数据类型及接口】

G726_BPS_E

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

ADEC_ATTR_ADPCM_S

【说明】

定义 ADPCM 编码协议属性结构体。

【定义】

```
typedef struct hiADEC_ATTR_ADPCM_S
{
    ADPCM_TYPE_E    enADPCMType;
}ADEC_ATTR_ADPCM_S;
```

【成员】

成员名称	描述
enADPCMType	ADPCM 类型。

【注意事项】

无。

【相关数据类型及接口】

ADPCM_TYPE_E

ADEC_MODE_E

【说明】

定义解码方式。

【定义】

```
typedef enum hiADEC_MODE_E
{
    ADEC_MODE_PACK = 0,/*require input is valid dec pack(a
                        complete frame encode result),
                        e.g.the stream get from AENC is a
                        valid dec pack, the stream know actually
                        pack len from file is also a dec pack.
                        this mode is high-performative*/
    ADEC_MODE_STREAM ,/*input is stream,low-performative,
                        if you couldn't find out whether a stream is
                        vaild dec pack,you could use
                        this mode*/
    ADEC_MODE_BUTT
}ADEC_MODE_E;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
ADEC_MODE_PACK	Pack 方式解码。
ADEC_MODE_STREAM	stream 方式解码。

【注意事项】

- pack 方式用于用户确认当前码流包为一帧数据编码结果的情况下，解码器会直接进行对其解码，如果不是一帧，解码器会出错。这种模式的效率比较高，在使用 AENC 模块编码的码流包如果没有破坏，均可以使用此方式解码。
- stream 方式用于用户不能确认当前码流包是不是一帧数据的情况下，解码器需要对码流进行判断并缓存，此工作方式的效率低下，一般用于读文件码流送解码或者不确定码流包边界的情况。当然由于语音编码码流长度固定，很容易确定在码流中的帧边界，推荐使用 pack 方式解码。

【相关数据类型及接口】

无

ADEC_CHN_ATTR_S

【说明】

定义解码通道属性结构体。

【定义】

```
typedef struct hiADEC_CH_ATTR_S
{
    PAYLOAD_TYPE_E  enType;
    HI_U32          u32BufSize;
    ADEC_MODE_E     enMode;
    HI_VOID         *pValue;
}ADEC_CHN_ATTR_S;
```

【成员】

成员名称	描述
enType	音频解码协议类型。 静态属性。
u32BufSize	音频解码缓存大小。 取值范围：[2, MAX_AUDIO_FRAME_NUM]，以帧为单位。 静态属性。
enMode	解码方式。 静态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
pValue	具体协议属性指针。

【注意事项】

无。

【相关数据类型及接口】

无。

ADEC_DECODER_S

【说明】

定义解码器属性结构体。

【定义】

```
typedef struct hiADEC_DECODER_S
{
    PAYLOAD_TYPE_E  enType;
    HI_CHAR         aszName[16];
    HI_S32          (*pfnOpenDecoder)(HI_VOID *pDecoderAttr, HI_VOID
**ppDecoder);
    HI_S32          (*pfnDecodeFrm)(HI_VOID *pDecoder, HI_U8
**pu8Inbuf,HI_S32 *ps32LeftByte, HI_U16 *pu16Outbuf,HI_U32
*pu32OutLen,HI_U32 *pu32Chns);
    HI_S32          (*pfnGetFrmInfo)(HI_VOID *pDecoder, HI_VOID *pInfo);
    HI_S32          (*pfnCloseDecoder)(HI_VOID *pDecoder);
} ADEC_DECODER_S;
```

【成员】

成员名称	描述
enType	解码协议类型。
aszName	解码器名称。
pfnOpenDecoder	打开解码器的函数指针。
pfnDecodeFrm	进行解码的函数指针。
pfnGetFrmInfo	获取音频帧信息的函数指针。
pfnCloseDecoder	关闭解码器的函数指针。

【注意事项】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

请参见《音频组件 API 参考》。

【相关数据类型及接口】

HI_MPI_ADEC_RegeisterDecoder

AUDIO_FRAME_INFO_S

【说明】

定义解码帧信息结构体指针。

【定义】

```
typedef struct hiAUDIO_FRAME_INFO_S
{
    AUDIO_FRAME_S *pstFrame;/*frame ptr*/
    HI_U32        u32Id;   /*frame id*/
} AUDIO_FRAME_INFO_S;
```

【成员】

成员名称	描述
pstFrame	解码帧数据指针。
u32Id	保留。

【注意事项】

无。

【相关数据类型及接口】

- HI_MPI_ADEC_GetFrame
- HI_MPI_ADEC_ReleaseFrame

9.4.4 内置 Audio Codec

ACODEC_FS_E

【说明】

定义 I^2S 接口的采样率选择。

【定义】

```
typedef enum hiACODEC_FS_E {
    ACODEC_FS_48000 =   0x1a,
    ACODEC_FS_24000 =   0x19,
    ACODEC_FS_12000 =   0x18,
    ACODEC_FS_44100 =   0x1a,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    ACODEC_FS_22050 =   0x19,
    ACODEC_FS_11025 =   0x18,
    ACODEC_FS_32000 =   0x1a,
    ACODEC_FS_16000 =   0x19,
    ACODEC_FS_8000  =   0x18,
    ACODEC_FS_BUTT = 0x1b,
} ACODEC_FS_E;
```

【成员】

成员名称	描述
ACODEC_FS_48000	48K 采样率。
ACODEC_FS_24000	24K 采样率。
ACODEC_FS_12000	12K 采样率。
ACODEC_FS_44100	44.1K 采样率。
ACODEC_FS_22050	22.05K 采样率。
ACODEC_FS_11025	11.025K 采样率。
ACODEC_FS_32000	32K 采样率。
ACODEC_FS_16000	16K 采样率。
ACODEC_FS_8000	8K 采样率。

【注意事项】

无。

【相关数据类型及接口】

无。

ACODEC_MIXER_E

【说明】

定义 Audio Codec 输入选择。

【定义】

```
typedef enum hiACODEC_MIXER_E {
    ACODEC_MIXER_LINEIN = 0x0,
    ACODEC_MIXER_MICIN  = 0x1,
    ACODEC_MIXER_BUTT,
} ACODEC_MIXER_E
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
ACODEC_MIXER_LINEIN	MICPGA 选择 LINEIN 输入。
ACODEC_MIXER_MICIN	MICPGA 选择 MICIN 输入。

【注意事项】

无。

【相关数据类型及接口】

无。

ACODEC_VOL_CTRL

【说明】

定义 Audio Codec 音量控制结构体

【定义】

```
typedef struct {
    /*volume control, 0x00~0x7e, 0x7F:mute*/
    unsigned int vol_ctrl;
    /*adc/dac mute control, 1:mute, 0:unmute*/
    unsigned int vol_ctrl_mute;
} ACODEC_VOL_CTRL;
```

【成员】

成员名称	描述
vol_ctrl	音量大小。
vol_ctrl_mute	静音控制。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

9.5 错误码

音频输入错误码

音频输入 API 错误码如表 9-9 所示。

表9-9 音频输入 API 错误码

错误代码	宏定义	描述
0xA0158001	HI_ERR_AI_INVALID_DEVID	音频输入设备号无效
0xA0158002	HI_ERR_AI_INVALID_CHNID	音频输入通道号无效
0xA0158003	HI_ERR_AI_ILLEGAL_PARAM	音频输入参数设置无效
0xA0158005	HI_ERR_AI_NOT_ENABLED	音频输入设备或通道没有使能
0xA0158006	HI_ERR_AI_NULL_PTR	输入参数空指针错误
0xA0158007	HI_ERR_AI_NOT_CONFIG	音频输入设备属性未设置
0xA0158008	HI_ERR_AI_NOT_SUPPORT	操作不支持
0xA0158009	HI_ERR_AI_NOT_PERM	操作不允许
0xA015800C	HI_ERR_AI_NOMEM	分配内存失败
0xA015800D	HI_ERR_AI_NOBUF	音频输入缓存不足
0xA015800E	HI_ERR_AI_BUF_EMPTY	音频输入缓存为空
0xA015800F	HI_ERR_AI_BUF_FULL	音频输入缓存为满
0xA0158010	HI_ERR_AI_SYS_NOTREADY	音频输入系统未初始化
0xA0158012	HI_ERR_AI_BUSY	音频输入系统忙

音频输出错误码

音频输出 API 错误码如表 9-10 所示。

表9-10 音频输出 API 错误码

错误代码	宏定义	描述
0xA0168001	HI_ERR_AO_INVALID_DEVID	音频输出设备号无效
0xA0168002	HI_ERR_AO_INVALID_CHNID	音频输出通道号无效
0xA0168003	HI_ERR_AO_ILLEGAL_PARAM	音频输出参数设置无效
0xA0168005	HI_ERR_AO_NOT_ENABLED	音频输出设备或通道没使能

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA0168006	HI_ERR_AO_NULL_PTR	输出空指针错误
0xA0168007	HI_ERR_AO_NOT_CONFIG	音频输出设备属性未设置
0xA0168008	HI_ERR_AO_NOT_SUPPORT	操作不被支持
0xA0168009	HI_ERR_AO_NOT_PERM	操作不允许
0xA016800C	HI_ERR_AO_NOMEM	系统内存不足
0xA016800D	HI_ERR_AO_NOBUF	音频输出缓存不足
0xA016800E	HI_ERR_AO_BUF_EMPTY	音频输出缓存为空
0xA016800F	HI_ERR_AO_BUF_FULL	音频输出缓存为满
0xA0168010	HI_ERR_AO_SYS_NOTREADY	音频输出系统未初始化
0xA0168012	HI_ERR_AO_BUSY	音频输出系统忙

音频编码错误码

音频编码 API 错误码如表 9-11 所示。

表9-11 音频编码 API 错误码

错误代码	宏定义	描述
0xA0178001	HI_ERR_AENC_INVALID_DEVID	音频设备号无效
0xA0178002	HI_ERR_AENC_INVALID_CHNID	音频编码通道号无效
0xA0178003	HI_ERR_AENC_ILLEGAL_PARAM	音频编码参数设置无效
0xA0178004	HI_ERR_AENC_EXIST	音频编码通道已经创建
0xA0178005	HI_ERR_AENC_UNEXIST	音频编码通道未创建
0xA0178006	HI_ERR_AENC_NULL_PTR	输入参数空指针错误
0xA0178007	HI_ERR_AENC_NOT_CONFIG	编码通道未配置
0xA0178008	HI_ERR_AENC_NOT_SUPPORT	操作不被支持
0xA0178009	HI_ERR_AENC_NOT_PERM	操作不允许
0xA017800C	HI_ERR_AENC_NOMEM	系统内存不足
0xA017800D	HI_ERR_AENC_NOBUF	编码通道缓存分配失败
0xA017800E	HI_ERR_AENC_BUF_EMPTY	编码通道缓存空
0xA017800F	HI_ERR_AENC_BUF_FULL	编码通道缓存满

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA0178010	HI_ERR_AENC_SYS_NOTREADY	系统没有初始化
0xA0178040	HI_ERR_AENC_ENCODER_ERR	音频编码数据错误

音频解码错误码

音频解码 API 错误码如表 9-12 所示。

表9-12 音频解码 API 错误码

错误代码	宏定义	描述
0xA0188001	HI_ERR_ADEC_INVALID_DEVID	音频解码设备号无效
0xA0188002	HI_ERR_ADEC_INVALID_CHNID	音频解码通道号无效
0xA0188003	HI_ERR_ADEC_ILLEGAL_PARAM	音频解码参数设置无效
0xA0188004	HI_ERR_ADEC_EXIST	音频解码通道已经创建
0xA0188005	HI_ERR_ADEC_UNEXIST	音频解码通道未创建
0xA0188006	HI_ERR_ADEC_NULL_PTR	输入参数空指针错误
0xA0188007	HI_ERR_ADEC_NOT_CONFIG	解码通道属性未配置
0xA0188008	HI_ERR_ADEC_NOT_SUPPORT	操作不被支持
0xA0188009	HI_ERR_ADEC_NOT_PERM	操作不允许
0xA018800C	HI_ERR_ADEC_NOMEM	系统内存不足
0xA018800D	HI_ERR_ADEC_NOBUF	解码通道缓存分配失败
0xA018800E	HI_ERR_ADEC_BUF_EMPTY	解码通道缓存空
0xA018800F	HI_ERR_ADEC_BUF_FULL	解码通道缓存满
0xA0188010	HI_ERR_ADEC_SYS_NOTREADY	系统没有初始化
0xA0188040	HI_ERR_ADEC_DECODER_ERR	音频解码数据错误

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

10 视频解码.....10-1
10.1 概述.....10-1
10.2 重要概念.....10-2
10.3 API 参考.....10-4
10.4 数据类型.....10-38
10.5 错误码.....10-49

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

10 视频解码

10.1 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。

不同芯片的 VDEC 特性如表 10-1 所示。

表10-1 不同芯片的 VDEC 特性

特性	硬件解码模块	支持最大通道数	支持协议	最大、最小分辨率支持	每秒可解码的像素数 此参数为解码器解码性能的估计，仅供参考	需要加载的模块
Hi3531	VDH JPEGD	64	**VDH:** H.264/VC1/MPEG4/ MPEG2/AVS **JPEGD:** JPEG/MJPEG	**VDH:** H.264: max 5632x4224, min 64x64 MPEG4/MPEG2/VC1/AVS: max1920x1080, min 64x64 **JPEGD：** JPEG/MJPEG: max 4096x4080, min 64x64	3Mpixel x 60fps x2= 360Mpixel	hi3531_vdec.ko hi3531_vfmw.ko jpeg.ko
Hi3521	VDH VEDU JPEGD	64（0~31 为 VDH 专用通道； 32~63 为 VEDU 专用通道）	**VDH:** H.264/MPEG4 **VEDU:** H.264（仅支持本芯片自编自解） **JPEGD:** JPEG/MJPEG	**VDH:** H.264: max 5632x4224, min 64x64 MPEG4: max 1920x1080, min 64x64 **VEDU：**	2Mpixelx60fps= 120Mpixel	hi3521_vdec.ko hi3521_vfmw.ko hi3521_chnl.ko (使用 VEDU 解码)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

特性	硬件解码模块	支持最大通道数	支持协议	最大、最小分辨率支持	每秒可解码的像素数 此参数为解码器解码性能的估计，仅供参考	需要加载的模块
				H.264: max 1920x2048, min 160x64 **JPEGD：** JPEG/MJPEG: max 4096x4080, min 64x64		jpeg.ko
Hi3520A	VDH VEDU JPEGD	64（0~31 为 VDH 专用通道； 32~63 为 VEDU 专用通道）	**VDH:** H.264(不支持场解码，不支持 B 帧) **VEDU:** H.264（仅支持本芯片自编自解） **JPEGD:** JPEG/MJPEG	**VDH:** H.264: max 1920x2048, min 64x64 **VEDU：** H.264: max 1920x2048, min 160x64 **JPEGD：** JPEG/MJPEG: max 4096x4080, min 64x64	2Mpixelx60fps=120Mpixel	hi3520A_vdec.ko hi3520A_vfmw.ko jpeg.ko hi3520A_chnl.ko jpeg.ko
Hi3520D /Hi3515A/Hi3515C	VEDU JPEGD	64	**VEDU:** H.264(不支持场解码，不支持 B 帧) **JPEGD:** JPEG/MJPEG	**VEDU：** H.264: max 1920x2048, min 80x64 **JPEGD：** JPEG/MJPEG: max 4096x4080, min 64x64	2Mpixelx30fps=60Mpixel	hi3520D_vdec.ko hi3520D_vfmw.ko hi3520D_chnl.ko (使用 VEDU 解码) jpeg.ko

10.2 重要概念

- 码流发送方式

  VDEC 解码器提供两种码流发送方式：

  - 流式发送：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 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 码流，不能希望马上开始解码图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 按帧发送：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，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。

发送方式在通道属性中设置，具体使用方法可见后面的 API 参考。两种发送方式都须保证每次调用发送接口发送的码流大小不超过通道属性中设置的 buffer 大小。

通过设置发送方式和通道参数中的解码器输出顺序，可以实现不同的图像输出方式。

- 图象输出方式

根据 H.264 协议，解码图象可能不会在解码后立即输出。VDEC 解码器可以通过设置不同的图象输出方式达到尽快输出的目的。图象输出方式包括以下三种：

- 普通输出：完全按照 H.264 协议输出图像。
- 直接输出：收到下一帧码流，输出当前帧图象。
- 按帧输出：收到当前帧码流，输出当前帧图象。

以上三种输出方式，自上而下，输出速度越来越快。其中直接输出和按帧输出方式都不符合 H.264 协议，因此需用户正确配置和使用解码器，才可实现，配置方法如表 10-2 所示。

- 通道解码能力

通道解码能力是指当前解码通道所能解码的总分辨率。例如，用户创建了一个宽为 720，高为 576 的解码通道，那么当前解码通道的通道解码能力就为 720x576，在内存足够的前提下，当前通道能解码任何 WxH <=720x576 分辨率的码流。

这里要注意的是，即使码流 WxH 的总分辨率一样，但是不同的宽高组合解码所需要的内存大小还是不一样的。帧存大小计算方法请参见 HI_MPI_VDEC_CreateChn 的相关描述。

表10-2 不同解码输出方式的参数配置

输出方式	通道属性	通道参数	注意事项
	enMode	s32DecOrderOutput	
普通输出	VIDEO_MODE_STREAM（流式发送）	0（按显示序输出）	-
	VIDEO_MODE_FRAME（按帧发送）	0（按显示序输出）	用户保证按帧发送
直接输出	VIDEO_MODE_STREAM（流式发送）	1（按解码序输出）	用户需保证码流解码序与显示序相同
按帧输出	VIDEO_MODE_FRAME（按帧发送）	1（按解码序输出）	用户保证按帧发送 用户需保证码流解码序与显示序相同

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

解码序和显示序：根据 H.264 协议，视频的解码顺序未必是视频的输出顺序（即显示序）。例如 B 帧解码时需要前后的 P 帧作为参考，所以 B 帧后的 P 帧先于 B 帧解码，但 B 帧先于 P 帧输出。按解码序输出是保证快速输出的一个必要条件，用户选择按解码序输出，需保证码流的解码序与显示序相同。

码流发送方式 enMode 和图象输出方式 s32DecOrderOutput 分别可在创建通道接口（HI_MPI_VDEC_CreateChn）和设置通道参数接口（HI_MPI_VDEC_SetChnParam）中设置，具体使用方法可见后面的 API 参考。

- 时间戳（PTS）处理

  解码输出的图像时间戳 PTS 在 VIDEO_MODE_FRAME 下为发送码流(HI_MPI_VDEC_SendStream)中送入对应的码流的 PTS，解码器不会更改此值；在 VIDEO_MODE_STREAM 下发送码流时，解码输出图像的时间戳统一设为 0。

- 码流 Buffer 配置模式

  解码码流 buffer 配置支持两种模式：一般模式和省内存模式。

  - 一般模式：码流 Buffer 总大小为用户配置的 u32BufSize+通道宽 x 通道高/2，u32BufSize 可配置的最小值为通道宽 x 通道高 x3/4，此模式下码流 Buffer 能容纳每一帧码流，包括超大帧。
  - 省内存模式：码流 Buffer 的总大小为用户创建解码通道时配置的 u32BufSize 的大小，u32BufSize 可配置的最小值为 32KB，但是用户必须保证送入码流每帧大小不能超过 u32BufSize 的一半。码流 buffer 省内存模式只在帧模式解码下设置有效。

  两种模式可通过 ko 加载时设置模块参数 MiniBufMode 来选择。MiniBufMode=1 表示省内存模式；MiniBufMode=0 表示一般模式。

10.3 API 参考

视频解码模块实现创建解码通道、发送视频码流、获取解码后图像等功能。

该功能模块提供以下 MPI：

- HI_MPI_VDEC_CreateChn：创建视频解码通道。
- HI_MPI_VDEC_DestroyChn：销毁视频解码通道。
- HI_MPI_VDEC_StartRecvStream：解码器开始接收用户发送的码流。
- HI_MPI_VDEC_StopRecvStream：解码器停止接收用户发送的码流。
- HI_MPI_VDEC_Query：查询解码通道状态。
- HI_MPI_VDEC_QueryData：查询解码通道是否有图像或用户数据输出。
- HI_MPI_VDEC_GetChnAttr：获取视频解码通道属性。
- HI_MPI_VDEC_SendStream：向视频解码通道发送码流数据。
- HI_MPI_VDEC_SendStream_TimeOut：向视频解码通道发送码流数据。
- HI_MPI_VDEC_GetUserData：获取视频解码通道的用户数据。
- HI_MPI_VDEC_GetUserData_TimeOut：获取视频解码通道的用户数据。
- HI_MPI_VDEC_ReleaseUserData：释放视频解码通道的用户数据。
- HI_MPI_VDEC_GetImage：获取视频解码通道的解码图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_VDEC_GetImage_TimeOut：获取视频解码通道的解码图像。
- HI_MPI_VDEC_ReleaseImage：释放视频解码通道的解码图像
- HI_MPI_VDEC_GetFd：获取视频解码通道的设备文件句柄。
- HI_MPI_VDEC_ResetChn：复位解码通道。
- HI_MPI_VDEC_SetChnParam：设置解码通道参数。
- HI_MPI_VDEC_GetChnParam：获取解码通道参数。
- HI_MPI_VDEC_SetPrtclParam：设置协议相关的内存分配通道参数。
- HI_MPI_VDEC_GetPrtclParam：获取协议相关的内存分配通道参数。
- HI_MPI_VDEC_SetDisplayFrameNum：设置显示帧存个数。
- HI_MPI_VDEC_GetDisplayFrameNum：获取显示帧存个数。
- HI_MPI_VDEC_SetChnPrevMode：设置通道预览模式。
- HI_MPI_VDEC_GetChnPrevMode：获取通道预览模式。

HI_MPI_VDEC_CreateChn

【描述】

创建视频解码通道。

【语法】

```
HI_S32 HI_MPI_VDEC_CreateChn(VDEC_CHN VdChn, const VDEC_CHN_ATTR_S
*pstAttr)
```

【参数】

参数名称	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	解码属性指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者相关依赖的模块没有加载
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_ILLEGAL_PARAM	参数错或超出合法范围。
HI_ERR_VDEC_NOT_SUPPORT	超过解码能力集（解码能力集如表 10-1 所示）。
HI_ERR_VDEC_EXIST	试图申请或者创建已经存在的设备、通道或者资源。
HI_ERR_VDEC_NOMEM	分配内存失败（如系统内存不足）。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 通道号不能超出最大的通道号范围。
- 执行相关协议的解码时，必须加载相关协议的 ko。请参见表 10-1 关于”需要加载的模块”的描述。
- 如果参数 pstAttr 错误或者为空会返回错误码 HI_ERR_VDEC_ILLEGAL_PARAM 或 HI_ERR_VDEC_NULL_PTR。参数 pstAtt 和取值范围见数据类型 VDEC_CHN_ATTR_S。
- 在创建视频解码通道之前必须保证通道未创建（或者已经销毁），否则会直接返回通道已存在错误。
- 系统内存不足时会返回 HI_ERR_VDEC_NOMEM 的错误码，可考虑扩展 MMZ 内存或 OS 内存。
- 当通道属性中的值超过解码能力集时，会返回 HI_ERR_VDEC_NOT_SURPPORT 的错误码。目前解码能力集如表 10-1 所示。
- 解码器不能解码超过通道的最大高宽的码流。如果图像宽高超出，解码器没有图像输出，可以在 proc 信息中可以查询 PicSizeErr 项是否有增加，log 中也有相应提示。
- 通道属性全为静态属性，只有创建时可以设置。
- Hi3520D 最多支持创建 64 个 H.264 通道，Hi3515A/Hi3515C 最多支持创建 4 个 H.264 通道。
- 如果用户使用同一个解码通道同时解码 720x576 和 576x720 两种分辨率的码流，则用户需要按照以下帧存计算方法分别计算这两种码流所需内存的大小，720x576 需要的帧存为 663552 字节，576x720 需要的帧存为 622080 字节，然后取大值 720x576 创建解码通道即可同时解码 720x576 和 576x720 两种分辨率的码流。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

注意

- 目前 JPEG/MJPEG 解码仅支持按帧发送方式。
- 如果需要解码 YUV444 格式的 JPEG 图片，则需要把解码通道的宽（或高）设置为 JPEG 图片的宽（或高）的 2 倍，否则会由于解码 buffer 不够导致无法解出 YUV444 格式的图片。

【举例】

无。

【相关主题】

解码帧存计算公式：

假定码流的宽为 Width，高为 Height，解码需要的帧存大小为 FrameSize。

SlotWidth = (Width + 63) & (~63);

SlotHeight = (Height+ 16) & (~16);

对于 H264 和 MPEG4，FrameSize = (SlotWidth * SlotHeight * 3) / 2；

对于 JPEG，FrameSize = SlotWidth * SlotHeight *2；

HI_MPI_VDEC_DestroyChn

【描述】

销毁视频解码通道。

【语法】

```
HI_S32 HI_MPI_VDEC_DestroyChn(VDEC_CHN VdChn);
```

【参数】

参数名称	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，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_NOT_PERM	操作不允许，在进行此操作前必须停止接收码流。
HI_ERR_VDEC_UNEXIST	通道未创建或已销毁。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者没有加载

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 销毁前必须保证通道已创建，否则会返回通道未创建的错误。
- 销毁前必须停止接收码流（或者尚未开始接收码流），否则返回错误码 HI_ERR_VDEC_NOT_PERM。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_StartRecvStream

【描述】

解码器开始接收用户发送的码流。

【语法】

```
HI_S32 HI_MPI_VDEC_StartRecvStream(VDEC_CHN VdChn);
```

【参数】

参数名称	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】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道尚未创建或者已销毁。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者没有加载

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 启动接收码流前必须保证通道已创建，否则会返回通道未创建的错误。
- 启动接收码流之后，才能调用 HI_MPI_VDEC_SendStream 发送码流成功。
- 重复调用启动接收码流接口时，返回成功。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_StopRecvStream

【描述】

解码器停止接收用户发送的码流。

【语法】

```
HI_S32 HI_MPI_VDEC_StopRecvStream(VDEC_CHN VdChn);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	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，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道尚未创建或者已销毁。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者没有加载

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 停止接收码流前必须保证通道已创建，否则会返回通道未创建的错误。
- 调用此接口后，调用发送码流接口 HI_MPI_VDEC_SendStream 会返回失败。
- 重复调用停止接收码流接口时，返回成功

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_Query

【描述】

查询解码通道状态。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【语法】

```
HI_S32 HI_MPI_VDEC_Query(VDEC_CHN VdChn,VDEC_CHN_STAT_S *pstStat);
```

【参数】

参数名称	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
pstStat	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，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道尚未创建或者不存在。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者没有加载
HI_ERR_VDEC_NULL_PTR	视频解码通道状态 结构体指针为空。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

查询通道状态前必须保证通道已创建，否则会返回通道未创建的错误。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VDEC_QueryData

【描述】

查询解码通道是否有图像或用户数据输出。

【语法】

```
HI_S32 HI_MPI_VDEC_QueryData(VDEC_CHN VdChn,HI_BOOL *pbIsData);
```

【参数】

参数名称	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
pbIsData	视频解码通道是否有图像或用户数据输出。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道尚未创建或者不存在。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者没有加载
HI_ERR_VDEC_NULL_PTR	视频解码通道状态 结构体指针为空。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

查询通道数据前必须保证通道已创建，否则会返回通道未创建的错误。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

无。

HI_MPI_VDEC_GetChnAttr

【描述】

获取视频解码通道属性。

【语法】

```
HI_S32 HI_MPI_VDEC_GetChnAttr(VDEC_CHN VdChn, VDEC_CHN_ATTR_S *pstAttr);
```

【参数】

参数名称	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	解码属性指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_UNEXIST	试图使用不存在的通道。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

获取属性前必须保证通道已创建，否则会返回通道未创建的错误。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_SendStream

【描述】

向视频解码通道发送码流数据。

【语法】

```
HI_S32 HI_MPI_VDEC_SendStream(VDEC_CHN VdChn, const VDEC_STREAM_S
*pstStream, HI_U32 u32BlockFlag);
```

【参数】

参数名称	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
u32BlockFlag	阻塞非阻塞标志。 取值范围： • HI_IO_BLOCK：阻塞。 • HI_IO_NOBLOCK：非阻塞。 动态属性。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_ILLEGAL_PARAM	参数超出合法范围。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_NOT_PERM	操作不允许，在通道未准备好接收码流的情况下向此通道发送码流。
HI_ERR_VDEC_BUF_FULL	缓冲区满，非阻塞接口如果发送失败则返回此错误。
HI_ERR_VDEC_BADADDR	码流地址错误。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 此接口通过改变 u32BlockFlag 值支持阻塞方式和非阻塞方式工作。
- 发送数据前必须保证已经调用 HI_MPI_VDEC_StartRecvStream 接口启动接收码流，否则直接返回 HI_ERR_VDEC_NOT_PERM。如果在发送数据过程中停止接收码流，就会立刻返回 HI_ERR_VDEC_NOT_PERM。
- 发送数据前必须保证通道已经被创建，否则直接返回 HI_ERR_VDEC_UNEXIST。如果在发送数据过程中销毁通道，就会立刻返回 HI_ERR_VDEC_UNEXIST。
- 发送码流时需要按照创建解码通道时设置的发送方式（按帧发送或流式发送）进行发送。按帧发送时，调用此接口一次，必须发送完整的一帧码流，否则，解码会出现错误。流式发送则无此限制。
- 不能发送长度为 0 的码流，否则返回 HI_ERR_VDEC_ILLEGAL_PARAM 的错误码。
- 按帧发送时，解码图像的时间戳等于传入参数 pstStream 结构体中的时间戳，按流发送时，解码图像的时间戳等于 0

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VDEC_SendStream_TimeOut

【描述】

向视频解码通道发送码流数据。

【语法】

```
HI_S32 HI_MPI_VDEC_SendStream_TimeOut(VDEC_CHN VdChn, const VDEC_STREAM_S
*pstStream, HI_U32 u32MilliSec);
```

【参数】

参数名称	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
u32MilliSec	超时时间，以 ms 为单位 取值范围：[0, 10000]	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_ILLEGAL_PARAM	参数超出合法范围。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_NOT_PERM	操作不允许，在通道未准备好接收码流的情况下向此通道发送码流。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VDEC_BUSY	超时，等待时间超出设定值仍然没有足够的码流缓冲区。
HI_ERR_VDEC_BADADDR	码流地址错误。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 发送数据前必须保证已经调用 HI_MPI_VDEC_StartRecvStream 接口启动接收码流，否则直接返回 HI_ERR_VDEC_NOT_PERM。如果在发送数据过程中停止接收码流，就会立刻返回 HI_ERR_VDEC_NOT_PERM。
- 发送数据前必须保证通道已经被创建，否则直接返回 HI_ERR_VDEC_UNEXIST。如果在发送数据过程中销毁通道，就会立刻返回 HI_ERR_VDEC_UNEXIST。
- 发送码流时需要按照创建解码通道时设置的发送方式（按帧发送或流式发送）进行发送。按帧发送时，调用此接口一次，必须发送完整的一帧码流，否则，解码会出现错误。流式发送则无此限制。
- 不能发送长度为 0 的码流，否则返回 HI_ERR_VDEC_ILLEGAL_PARAM 的错误码。
- 按帧发送时，解码图像的时间戳等于传入参数 pstStream 结构体中的时间戳，按流发送时，解码图像的时间戳等于 0。
- 超时的有效时间单位为 10ms，因此实际最长等待时间为 10ms 倍数
- 超时时间为 0 时，为 HI_IO_BLOCK 操作，直到码流存入到缓冲区或解码通道停止接收码流时才返回。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetUserData

【描述】

获取视频解码通道的用户数据。

【语法】

```
HI_S32 HI_MPI_VDEC_GetUserData(VDEC_CHN VdChn, VDEC_USERDATA_S *
pstUserData, HI_U32 u32BlockFlag);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	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
u32BlockFlag	阻塞非阻塞标志。 取值范围： • HI_IO_BLOCK：阻塞。 • HI_IO_NOBLOCK：非阻塞。 动态属性。	输入
pstUserData	获取的解码用户数据	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_BUF_EMPTY	非阻塞获取时，缓冲区中数据不足。
HI_ERR_VDEC_NOT_PERM	获取用户数据过程中，有 reset 操作时，直接返回 HI_ERR_VDEC_NOT_PERM 的错误码。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 用户数据是 h264 编码时，插入码流 SEI segment 的一段用户私有数据。（JPEG/MJPEG 不支持用户数据）
- 获取用户数据时必须保证通道已经被创建，否则直接返回失败，如果在获取用户数据的过程中销毁通道，则会立刻返回 HI_ERR_VDEC_UNEXIST 的错误码。
- 此接口支持阻塞方式和非阻塞方式工作，如果使用非阻塞接口获取用户数据，当解码器中已解用户数据小于 1byte 时，接口会返回 HI_ERR_VDEC_BUF_EMPTY 的错误码。
- 通过 HI_MPI_VDEC_GetUserData 获取解码用户数据后，需要通过 HI_MPI_VDEC_ReleaseUserData 来释放，且必须和获取时一致。
- 在获取用户数据过程中，如果遇到 reset 操作，可能会直接返回 HI_ERR_VDEC_NOT_PERM 的错误码。
- 获取用户数据如果不够及时，造成用户数据 buffer 满时，会出现丢用户数据的情况。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetUserData_TimeOut

【描述】

获取视频解码通道的用户数据。

【语法】

```
HI_S32 HI_MPI_VDEC_GetUserData_TimeOut(VDEC_CHN VdChn, VDEC_USERDATA_S *
pstUserData, HI_U32 u32MilliSec);
```

【参数】

参数名称	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
u32MilliSec	超时时间，以 ms 为单位 取值范围：[0, 10000]	输入
pstUserData	获取的解码用户数据	输出

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_BUSY	超时，缓冲区中没有用户数据。
HI_ERR_VDEC_NOT_PERM	获取用户数据过程中，有 reset 操作时，直接返回 HI_ERR_VDEC_NOT_PERM 的错误码。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 用户数据是 h264 编码时，插入码流 SEI segment 的一段用户私有数据。（JPEG/MJPEG 不支持用户数据）
- 获取用户数据时必须保证通道已经被创建，否则直接返回失败，如果在获取用户数据的过程中销毁通道，则会立刻返回 HI_ERR_VDEC_UNEXIST 的错误码。
- 此接口支持阻塞方式和非阻塞方式工作，如果使用非阻塞接口获取用户数据，当解码器中已解用户数据小于 1byte 时，接口会返回 HI_ERR_VDEC_BUF_EMPTY 的错误码。
- 通过 HI_MPI_VDEC_GetUserData 获取解码用户数据后，需要通过 HI_MPI_VDEC_ReleaseUserData 来释放，且必须和获取时一致。
- 在获取用户数据过程中，如果遇到 reset 操作，可能会直接返回 HI_ERR_VDEC_NOT_PERM 的错误码。
- 获取用户数据如果不够及时，造成用户数据 buffer 满时，会出现丢用户数据的情况。
- 超时的有效时间单位为 10ms，因此实际最长等待时间为 10ms 倍数

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 超时时间为 0 时，为 HI_IO_BLOCK 操作，直到有用户数据或解码通道被销毁时才返回。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_ReleaseUserData

【描述】

释放用户数据。

【语法】

```
HI_S32 HI_MPI_VDEC_ReleaseUserData(VDEC_CHN VdChn, VDEC_USERDATA_S *
pstUserData);
```

【参数】

参数名称	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
pstUserData	解码后的用户数据指针，由 HI_MPI_VDEC_GetUserData 接口获取。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko
HI_ERR_VDEC_NULL_PTR	参数中有空指针。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_ILLEGAL_PARAM	参数超出合法范围。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 此接口需与 HI_MPI_VDEC_GetUserData 配对使用，获取的数据应当在使用完之后立即释放。
- 释放的数据必须是 HI_MPI_VDEC_GetUserData 从该通道获取的数据，不得对数据信息结构体进行任何修改，否则会返回 HI_ERR_VDEC_ILLEGAL_PARAM 的错误码 。
- 释放数据时必须保证通道已经被创建，否则直接返回 HI_ERR_VDEC_UNEXIST。如果在释放数据过程中销毁通道，则会立即返回 HI_ERR_VDEC_UNEXIST。
- 释放数据过程中如果有 reset 操作，会返回 HI_ERR_VDEC_NOT_PERM 的错误码。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetImage

【描述】

获取视频解码通道的解码图像。

【语法】

```
HI_S32 HI_MPI_VDEC_GetImage(VDEC_CHN VdChn, VIDEO_FRAME_INFO_S
*pstFrameInfo, HI_U32 u32BlockFlag);
```

【参数】

参数名称	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

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
u32BlockFlag	阻塞非阻塞标志。 取值范围： • HI_IO_BLOCK：阻塞。 • HI_IO_NOBLOCK：非阻塞。 动态属性。	输入
pstFrameInfo	获取的解码图像信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_BUF_EMPTY	非阻塞获取时，缓冲区中无数据。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 此接口支持阻塞方式和非阻塞方式工作。
- 通过 HI_MPI_VDEC_GetImage 获取解码图像数据后，需要通过 HI_MPI_VDEC_ReleaseImage 来释放。
- 获取解码图像时必须保证通道已经被创建，否则直接返回失败，如果在获取解码图像的过程中销毁通道，则会立刻返回失败。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 如果在获取解码的过程中复位通道，则会立刻返回 HI_ERR_VDEC_NOT_PERM 的错误码。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetImage_TimeOut

【描述】

获取视频解码通道的解码图像。

【语法】

```
HI_MPI_VDEC_GetImage_TimeOut(VDEC_CHN VdChn, VIDEO_FRAME_INFO_S
*pstFrameInfo,HI_U32 u32MilliSec)
```

【参数】

参数名称	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
u32MilliSec	超时时间，以毫秒为单位。 取值范围：[0, 10000]	输入
pstFrameInfo	获取的解码图像信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_BUSY	在规定时间内无新图像,超时返回
HI_ERR_VDEC_NOT_PERM	在操作中该通道为销毁或复位

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 此接口工作在阻塞模式下,在规定时间内不能获取图像时会返回超时错误。
- 通过 HI_MPI_VDEC_GetImage_TimeOut 获取解码图像数据后，需要通过 HI_MPI_VDEC_ReleaseImage 来释放。
- 获取解码图像时必须保证通道已经被创建，否则直接返回失败，如果在获取解码图像的过程中销毁通道，则会立刻返回失败。
- 如果在获取解码的过程中复位通道，则会立刻返回 HI_ERR_VDEC_NOT_PERM 的错误码。
- 超时的有效时间单位为 10ms，因此实际最长等待时间为 10ms 倍数。
- 超时时间为 0 时，为 HI_IO_BLOCK 操作，直到有图像或解码通道被销毁时才返回。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_ReleaseImage

【描述】

释放视频解码通道的图像。

【语法】

```
HI_S32 HI_MPI_VDEC_ReleaseImage(VDEC_CHN VdChn, VIDEO_FRAME_INFO_S
*pstFrameInfo);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	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 HI_MPI_VDEC_GetImage 接口获取。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或 ko 未加载
HI_ERR_VDEC_NULL_PTR	参数中有空指针。
HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源。
HI_ERR_VDEC_ILLEGAL_PARAM	参数超出合法范围。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 此接口需与 HI_MPI_VDEC_GetImage 配对使用，获取的数据应当在使用完之后立即释放，如果不及时释放，会导致解码过程阻塞等待资源。
- 释放的数据必须是 HI_MPI_VDEC_GetImage 从该通道获取的数据，不得对数据信息结构体进行任何修改，也不允许释放从其他的通道获取的数据，否则会导致数据不能释放，使此数据 buffer 丢失，甚至导致程序异常。
- 释放数据时必须保证通道已经被创建，否则直接返回 HI_ERR_VDEC_UNEXIST。如果在释放数据过程中销毁通道，则会立即返回 HI_ERR_VDEC_UNEXIST。
- 允许用户非顺序释放，即不按照获取图像的顺序释放图像。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 如果在获取图像的过程中，复位通道，会立刻返回错误。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetFd

【描述】

获取视频解码通道的设备文件句柄。

【语法】

```
HI_S32 HI_MPI_VDEC_GetFd(VDEC_CHN VdChn);
```

【参数】

参数名称	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
正数值	有效返回值。
非正数值	无效返回值。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

无。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VDEC_ResetChn

【描述】

复位视频解码通道。

【语法】

```
HI_S32 HI_MPI_VDEC_ResetChn(VDEC_CHN VdChn);
```

【参数】

参数名称	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，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_SYS_NOTREADY	系统未初始化或未加载 ko。
HI_ERR_VDEC_UNEXIST	通道不存在。
HI_ERR_VDEC_NOT_PERM	操作不允许。在进行此操作前必须停止接收码流。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 复位前必须保证通道已创建，否则会返回错误码 HI_ERR_VDEC_UNEXIST。
- 复位前必须停止接收码流，否则返回错误码 HI_ERR_VDEC_NOT_PERM。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_SetChnParam

【描述】

设置解码通道参数。

【语法】

```
HI_S32 HI_MPI_VDEC_SetChnParam(VDEC_CHN VdChn, VDEC_CHN_PARAM_S*
pstParam);
```

【参数】

参数名称	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，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道不存在。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者未加载 ko。
HI_ERR_VDEC_ILLEGAL_PARAM	参数超出合法范围
HI_ERR_VDEC_NOT_SURPPORT	通道类型为 JPEG/MJPEG 通道

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 设置通道参数前必须保证通道已创建，否则会返回通道未创建的错误。
- 该接口可设置通道的一些高级属性，创建通道后有默认值，默认值见数据类型 VDEC_CHN_PARAM_S 的说明。
- 各项参数如超过合法范围，会返回 HI_ERR_VDEC_ILLEGAL_PARAM 的错误码，各参数范围见数据类型 VDEC_CHN_PARAM_S 的说明。
- JPEG/MJPEG 不支持该接口。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetChnParam

【描述】

获取解码通道参数。

【语法】

```
HI_S32 HI_MPI_VDEC_GetChnParam(VDEC_CHN VdChn, VDEC_CHN_PARAM_S*
pstParam);
```

【参数】

参数名称	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，其值为错误码。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道不存在。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者未加载 ko。
HI_ERR_VDEC_NOT_SURPPORT	通道类型为 JPEG/MJPEG 通道

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- JPEG/MJPEG 不支持该接口。
- 获取通道参数前必须保证通道已创建，否则会返回通道未创建的错误。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_SetPrtclParam

【描述】

设置协议相关的内存分配通道参数。

【语法】

```
HI_S32 HI_MPI_VDEC SetPrtclParam (VDEC_CHN VdChn, VDEC_PRTCL_PARAM_S*
pstParam);
```

【参数】

参数名称	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	通道协议参数。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道不存在。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者未加载 ko。
HI_ERR_VDEC_ILLEGAL_PARAM	参数错或超出合法范围。
HI_ERR_VDEC_NOT_PERM	未停止接受码流或操作中该通道被同时创建、销毁

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

- 设置协议参数前必须保证通道已创建，否则会返回通道未创建的错误。
- 在调用该函数时不能同时进行重复创建或销毁该通道。
- 设置协议参数前必须先停止接收码流，否则返回错误码 HI_ERR_VDEC_NOT_PERM。
- 设置协议参数之前建议先调用接口 HI_MPI_VDEC_GetPrtclParam 获取默认的协议参数再进行设置，且建议参数 s32SCDBufSize 保持默认值，默认值是按照最小值配置的。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_VDEC_GetPrtclParam

【描述】

获取协议相关的内存分配通道参数。

【语法】

```
HI_S32 HI_MPI_VDEC_GetPrtclParam(VDEC_CHN VdChn, VDEC_PRTCL_PARAM_S *
pstParam);
```

【参数】

参数名称	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，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道不存在。
HI_ERR_VDEC_SYS_NOTREADY	系统没有初始化或者未加载 ko。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

获取协议参数前必须保证通道已创建，否则会返回通道未创建的错误。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

【相关主题】

无。

HI_MPI_VDEC_SetDisplayFrameNum

【描述】

设置显示帧存个数。

【语法】

```
HI_S32 HI_MPI_VDEC_SetDisplayFrameNum(VDEC_CHN VdChn, HI_S32
s32DisplayFrameNum);
```

【参数】

参数名称	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
s32DisplayFrameNum	显示帧个数。 取值范围：[0, 16]。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_NOT_PERM	该操作不允许
HI_ERR_VDEC_ILLEGAL_PARAM	参数错或超出合法范围。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- 该接口只能在创建解码通道前调用，否则返回该操作不允许的错误码HI_ERR_VDEC_NOT_PERM。
- 显示帧个数的取值范围是[0, 16]，默认为 2 个。
- 只有在用户态获取解码图像或者 VDEC 绑定的后级模块不会缓存任何解码图像的情况下才能把 s32 DisplayFrameNum 设置为 0，否则解码会由于帧存不够而被反压，停止解码。
- H264 和 MPEG4 解码通道的帧存数量为参考帧+显示帧+1，JPEG 解码通道的帧存数量为显示帧+1。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetDisplayFrameNum

【描述】

获取显示帧存个数。

【语法】

```
HI_S32 HI_MPI_VDEC_GetDisplayFrameNum(VDEC_CHN VdChn, HI_S32
*ps32DisplayFrameNum);
```

【参数】

参数名称	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
ps32DisplayFrameNum	显示帧个数指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

该接口在创建解码通道前、后都可以调用。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_SetChnPrevMode

【描述】

设置通道预览模式。

【语法】

```
HI_S32 HI_MPI_VDEC_SetChnPrevMode(VDEC_CHN VdChn, const HI_BOOL bPrevMode);
```

【参数】

参数名称	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
bPrevMode	是否为预览模式。 0：否； 1：是。	输入

【返回值】

返回值	描述
0	成功。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道未创建。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

此接口只能在通道创建之后调用，否则返回通道未创建的错误码 HI_ERR_VDEC_UNEXIST。

【举例】

无。

【相关主题】

无。

HI_MPI_VDEC_GetChnPrevMode

【描述】

获取通道预览模式。

【语法】

```
HI_S32 HI_MPI_VDEC_GetChnPrevMode(VDEC_CHN VdChn, HI_BOOL *pbPrevMode);
```

【参数】

参数名称	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
pbPrevMode	预览模式指针。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【错误码】

接口返回值	含义
HI_SUCCESS	成功。
HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围。
HI_ERR_VDEC_UNEXIST	通道未创建。

【需求】

- 头文件：mpi_vdec.h、hi_comm_vdec.h
- 库文件：libmpi.a

【注意】

此接口只能在通道创建之后调用，否则返回通道未创建的错误码 HI_ERR_VDEC_UNEXIST。

【举例】

无。

【相关主题】

无。

10.4 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：定义视频解码通道属性。
- VDEC_ATTR_JPEG_S：定义使用 JPEG 解码协议的视频解码通道属性结构体。
- VDEC_ATTR_VIDEO_S：定义使用除 JPEG 解码协议以外所支持的其它视频解码通道属性。
- VDEC_STREAM_S：定义解码码流结构体。
- VDEC_USERDATA_S：定义用户私有数据结构体。
- VDEC_CHN_STAT_S：定义通道状态结构体。
- VDEC_DECODE_ERROR_S：定义解码错误信息结构体。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- VIDEO_MODE_E：定义码流发送方式。
- VDEC_CHN_PARAM_S：定义解码通道参数。
- VDEC_PRTCL_PARAM_S：定义与协议相关的内存分配参数。
- VDEC_STD_EXTENSION_U：定义 VC1 相关的码流是否为 AP 和版本信息。

VDEC_CHN_ATTR_S

【说明】

定义视频解码通道属性结构体。

【定义】

```
typedef struct hiVDEC_CHN_ATTR_S
{
    PAYLOAD_TYPE_E enType;
    HI_U32      u32BufSize      ;       /*stream buf size(Byte)*/
    HI_U32      u32Priority     ;       /*priority*/
    HI_U32      u32PicWidth     ;       /*max pic width*/
    HI_U32      u32PicHeight    ;       /*max pic height*/
    union
    {
        VDEC_ATTR_JPEG_S stVdecJpegAttr;
        VDEC_ATTR_VIDEO_S stVdecVideoAttr;
    };
}VDEC_CHN_ATTR_S;
```

【成员】

成员名称	描述
enType	解码协议类型枚举值。 静态属性。
u32BufSize	码流缓存的大小。 取值范围：帧模式解码且使用码流 Buffer 省内存模式时，大于等于 32KB；其它情况大于等于解码通道大小（宽 x 高）的 3/4 倍，即 420 图象大小的一半（宽 x 高 x3/2x1/2），以 byte 为单位。 推荐值：一幅 YUV420 解码图像大小。即：宽 x 高 x1.5。 静态属性。
u32Priority	通道优先级，取值范围为 1～255，值越大优先级越高。 注意：0 表明此通道不被调度，Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C 的 VEDU 优先级无效但必须设置为非 0 才能解码
u32PicWidth	通道支持的解码图像最大宽（以像素为单位） 静态属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32PicHeight	通道支持的解码图像最大高（以像素为单位） 静态属性。
stVdecJpegAttr	JPEG 通道的相关属性
stVdecVideoAttr	除 JPEG 以外其它所支持类型的通道的相关属性

【描述】

无。

【相关数据类型及接口】

无。

VDEC_ATTR_JPEG_S

【说明】

定义使用 JPEG 解码协议的视频解码通道属性。

【定义】

```
typedef struct hiVDEC_ATTR_JPEG_S
{
    VIDEO_MODE_E     enMode;
}VDEC_ATTR_JPEG_S,*PTR_VDEC_ATTR_JPEG_S;
```

【成员】

成员名称	描述
enMode	码流发送方式。 JPEG 支持按帧。 静态属性。

【注意事项】

无。

【相关数据类型及接口】

无。

VDEC_ATTR_VIDEO_S

【说明】

定义使用除 JPEG 解码协议以外所支持的其它视频解码通道属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct hiVDEC_ATTR_VIDEO_S
{
    HI_U32          u32RefFrameNum  ;
    VIDEO_MODE_E    enMode;
    HI_S32          s32SupportBFrame;
}VDEC_ATTR_VIDEO_S;
```

【成员】

成员名称	描述
u32RefFrameNum	参考帧的数目。 取值范围：[0, 16]，以帧为单位。 参考帧的数目决定解码时需要的参考帧个数，会较大的影响内存占用，根据实际情况设置合适的值。 • 海思自编码流：推荐设为 2。 • 其他监控码流：推荐设为 5。 • 测试码流：推荐设为 16。 静态属性。
enMode	码流发送方式。 支持按帧或者按流。 静态属性。
s32SupportBFrame	是否支持 B 帧解码 取值范围：[0, 1]

【注意事项】

- 帧码流解码只解 I 帧时可以把参考帧设置为 0 以节省帧存。
- 场码流解码只解 I 场时必须把参考帧设置为 1 才能解码，把参考帧设置为 0 解不出图像。

【相关数据类型及接口】

无。

VDEC_STREAM_S

【说明】

定义视频解码的码流结构体。

【定义】

```
typedef struct hiVDEC_STREAM_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    HI_U8*  pu8Addr;
    HI_U32  u32Len;
    HI_U64  u64PTS;
}VDEC_STREAM_S;
```

【成员】

成员名称	描述
pu8Addr	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 μs 为单位。

【注意事项】

- 按帧发送时，解码图像的时间戳等于码流包中的时间戳，
- 按流发送时，解码图像的时间戳等于 0。

【相关数据类型及接口】

无。

VDEC_USERDATA_S

【说明】

定义用户私有数据结构体。

【定义】

```
typedef struct hi VDEC_USERDATA_S
{
    HI_U8*      pu8Addr;        /*privite data address*/
    HI_U32      u32PhyAddr;
    HI_U32      u32Len;         /*privite data len*/
    HI_BOOL     bValid;         /*is valid?*/}VDEC_USERDATA_S;
```

【成员】

成员名称	描述
pu8Addr	私有数据的虚拟地址。
u32PhyAddr	私有数据的物理地址

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32Len	私有数据的长度。 以 byte 为单位。
bValid	当前私有数据的有效标识。 取值范围：{HI_TRUE, HI_FALSE}。 • HI_TRUE：有效。 • HI_FALSE：无效。

【注意事项】

- 当使用非阻塞接口获取用户数据失败时，bvalid 等于 HI_FALSE。
- 获取成功时，bvalid 等于 HI_TRUE。

【相关数据类型及接口】

无。

VDEC_CHN_STAT_S

【说明】

定义通道状态结构体。

【定义】

```
typedef struct hiVDEC_CHN_STAT_S
{
    PAYLOAD_TYPE_E enType;       /* 解码协议 */
    HI_U32  u32LeftStreamBytes; /*待解码byte数*/
    HI_U32 u32LeftStreamFrames;/*待解码的frame数 */
    HI_U32  u32LeftPics;        /*待获取的pic数*/
    HI_BOOL bStartRecvStream;   /*是否允许接收码流*/
    HI_U32 u32RecvStreamFrames;  /*已接收的frame数*/
    HI_U32 u32DecodeStreamFrames; /*已解码的frame数*/
    VDEC_DECODE_ERROE_S stVdecDecErr;  /* 解码错误信息 */
}VDEC_CHN_STAT_S;
```

【成员】

成员名称	描述
enType	解码协议。
u32LeftStreamBytes	码流 buffer 中待解码的 byte 数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u32LeftStreamFrames	码流 buffer 中待解码的帧数。 –1 表示无效。 仅按帧发送时有效。
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。
u32RecvStreamFrames	码流 buffer 中已接收帧数。 -1 表示无效。 仅按帧发送时有效。
u32DecodeStreamFrames	码流 buffer 中已解码帧数。 -1 表示无效。 仅按帧发送时有效。 u32RecvStreamFrames = u32DecodeStreamFrames + u32LeftStreamFrames。
stVdecDecErr	解码错误信息。

【注意事项】

无。

【相关数据类型及接口】

无。

VDEC_DECODE_ERROR_S

【说明】

定义解码错误信息结构体。

【定义】

```
typedef struct hi_VDEC_DECODE_ERROR_S
{
    HI_S32 s32FormatErr;          /* 不支持的格式 */
    HI_S32 s32PicSizeErrSet;      /* 图像的宽（或高）比通道的宽（或高）大 */
    HI_S32 s32StreamUnsprt;       /* 不支持的规格 */
    HI_S32 s32PackErr;            /* 码流错误 */
    HI_S32 s32PrtclNumErrSet;     /* 设置的协议参数个数不够 */
    HI_S32 s32RefErrSet;          /* 设置的参考帧个数不够 */
    HI_S32 s32PicBufSizeErrSet;   /* 图像Buffer大小不够 */
}VDEC_DECODE_ERROR_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
s32FormatErr	不支持的格式。比如 H264 场码流，JPEG 的 Progressive 码流。
s32PicSizeErrSet	图像的宽（或高）比通道的宽（或高）大。
s32StreamUnsprt	不支持的规格（码流规格与芯片宣称支持的规格不一致）。
s32PackErr	码流有错。
s32PrtclNumErrSet	设置的协议参数个数不够。比如：Slice、Pps、Sps 个数。（此参数仅对 H264 解码有效）
s32RefErrSet	设置的参考帧个数不够。（JPEG 解码此参数无效）
s32PicBufSizeErrSet	解码 JPEGYUV444 格式的图片时，图像 Buffer 大小不够。可以通过扩大解码通道的宽或高来扩大图像 Buffer 大小。

【注意事项】

- 所有错误信息均以累加计数的形式表现。比如解码每发现一次码流有错，s32PackErr 数值就加 1。
- 复位通道后所有错误信息计数清零。

【相关数据类型及接口】

无。

VIDEO_MODE_E

【说明】

定义码流发送方式。

【定义】

```
typedef enum hiVIDEO_MODE_E
{
    VIDEO_MODE_STREAM = 0,
    VIDEO_MODE_FRAME,
    VIDEO_MODE_BUTT
}VIDEO_MODE_E;
```

【成员】

成员名称	描述
VIDEO_MODE_STREAM	按流方式发送码流。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
VIDEO_MODE_FRAME	发送码流。 以帧为单位。

【注意事项】

无。

【相关数据类型及接口】

无。

VDEC_CHN_PARAM_S

【说明】

定义解码通道参数。

【定义】

```
typedef struct hiVDEC_CHN_PARAM_S
{
    VDEC_STD_EXTENSION_U    StdExt;
    HI_S32  s32ChanErrThr;
    HI_S32  s32ChanStrmOFThr;
    HI_S32  s32DecMode;
    HI_S32  s32DecOrderOutput;
    HI_S32  s32DnrTfEnable;
    HI_S32  s32DnrDispOutEnable;
}VDEC_CHN_PARAM_S;
```

【成员】

成员名称	描述
StdExt	扩展信息，VC1 相关的码流是否为 AP 和版本信息。
s32ChanErrThr	错误阈值。 取值范围：[0, 100]。 0 代表有错即丢，100 代表无论错误多少均继续解码显示。 解码器计算每帧的错误率（错误宏块数 / 总宏块数 x 100%），如果错误率大于 s32ChanErrThr 则丢弃该帧不显示，而且如果此帧为参考帧，则后续所有帧都丢弃，直到下一个 I 帧。 Default: 30 (仅对除 JPEG 之外的解码通道有效)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
s32ChanStrmOFThr	解码前丢帧阈值。 取值范围：≥0 。 解码器运行过程中会监视前端码流 buffer 是否“即将溢出”，如果是，则放弃解码，快速消耗码流，直到码流降低到安全值以内。码流总量超过 s32ChanStrmOFThr 被视为“即将溢出”。 如果不希望解码器主动丢弃码流，将 s32ChanStrmOFThr 配置为 0。 Default: 0 (仅对除 JPEG 之外的解码通道，且只在 IPB 解码模式下有效)
s32DecMode	解码模式： 0：正常模式（I,P,B 帧都解码）； 1：IP 模式（不作参考的 B 帧丢弃不解）； 2：I 模式（只解 I 帧，P,B 帧均丢弃）； Default：1。 (仅对除 JPEG 之外的解码通道有效)
s32DecOrderOutput	解码图像输出顺序。 0：显示序输出； 1：解码序输出。 Default: 1 (仅对除 JPEG 之外的解码通道有效)
s32DnrTfEnable	<保留，暂无关>
s32DnrDispOutEnable	<保留，暂无关>。

【注意事项】

在使用 Hi3521/Hi3520A/Hi3520D/Hi3515A/Hi3515C VEDU 解码时，仅能检测图像有错/无错，s32ChanErrThr 在[0, 10)时丢弃存在错误的解码图像，[10~100]时则不丢弃任何解码图像。

【相关数据类型及接口】

无。

VDEC_PRTCL_PARAM_S

【说明】

定义与协议相关的内存分配参数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct hiVDEC_PRTCL_PARAM_S
{
    HI_S32  s32MaxSliceNum;
    HI_S32  s32MaxSpsNum;
    HI_S32  s32MaxPpsNum;
    HI_S32  s32SCDBufSize;
    HI_S32  s32DisplayFrameNum;
}VDEC_PRTCL_PARAM_S;
```

【成员】

成员名称	描述
s32MaxSliceNum	该通道解码支持的最大 Slice 个数，,默认为 16，仅对 H.264 有效，取值范围为[1, 136]。
s32MaxSpsNum	该通道解码支持的最大 SPS 个数，默认为 2，仅对 H.264 有效，取值范围为[1, 32]。
s32MaxPpsNum	该通道解码支持的最大 PPS 个数，默认为 2，仅对 H.264 有效，取值范围为[1, 256]。
s32SCDBufSize	码流切分模块的 buffer 大小，对 JPEG/MJPEG 无效，取值范围： 码流 buffer 一般模式下为大于等于 MAX（256 x 1024，通道宽 x 通道高 x 3 / 2）； 码流 buffer 省内存模式且帧解码模型下为大于等于 MAX（32 x 1024，u32BufSize），其中 u32BufSize 为用户设置的码流 buffer 大小。
s32DisplayFrameNum	显示及解码缓存的图像的最小帧数，默认为 2 个。 取值范围：[0, 16]

【注意事项】

只有在用户态获取解码图像或者 VDEC 绑定的后级模块不会缓存任何解码图像的情况下才能把 s32 DisplayFrameNum 设置为 0，否则解码会由于帧存不够而被反压，停止解码。

【相关数据类型及接口】

无。

VDEC_STD_EXTENSION_U

【说明】

扩展信息，VC1 相关的码流是否为 AP 和版本信息。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef union
{
    struct
    {
        HI_S32 IsAdvProfile;
       HI_S32 CodecVersion;
    } Vc1Ext;

    struct
    {
       HI_S32 bReversed;
    } Vp6Ext;
} VDEC_STD_EXTENSION_U;
```

【成员】

成员名称	描述
IsAdvProfile	Vc1 扩展信息，VC1 相关的码流是否为 AP。 默认值为 0。
CodecVersion	Vc1 扩展信息，版本信息。 默认值为 0。
bReversed	Vp6Ext 扩展信息，图像需要倒转时置 1，否则置 0。 默认值为 0。

【注意事项】

无。

【相关数据类型及接口】

无。

10.5 错误码

错误代码	宏定义	描述
0xA0058001	HI_ERR_VDEC_INVALID_DEVID	设备 ID 超出合法范围
0xA0058002	HI_ERR_VDEC_INVALID_CHNID	通道 ID 超出合法范围
0xA0058003	HI_ERR_VDEC_ILLEGAL_PARAM	参数超出合法范围

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA0058004	HI_ERR_VDEC_EXIST	试图申请或者创建已经存在的设备、通道或者资源
0xA0058005	HI_ERR_VDEC_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源
0xA0058006	HI_ERR_VDEC_NULL_PTR	函数参数中有空指针
0xA0058007	HI_ERR_VDEC_NOT_CONFIG	使用前未配置
0xA0058008	HI_ERR_VDEC_NOT_SUPPORT	不支持的参数或者功能
0xA0058009	HI_ERR_VDEC_NOT_PERM	该操作不允许，如试图修改静态配置参数
0xA005800C	HI_ERR_VDEC_NOMEM	分配内存失败，如系统内存不足
0xA005800D	HI_ERR_VDEC_NOBUF	分配缓存失败，如申请的数据缓冲区太大
0xA005800E	HI_ERR_VDEC_BUF_EMPTY	缓冲区中无数据
0xA005800F	HI_ERR_VDEC_BUF_FULL	缓冲区中数据满
0xA0058010	HI_ERR_VDEC_SYS_NOTREADY	系统未初始化
0xA0058011	HI_ERR_VDEC_BADADDR	地址错误
0xA0058012	HI_ERR_VDEC_BUSY	系统忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

11 智能加速引擎.....11-1
11.1 概述.....11-1
11.2 功能描述.....11-1
11.2.1 重要概念.....11-1
11.2.2 使用示意.....11-6
11.3 API 参考.....11-7
11.4 数据类型.....11-29
11.5 错误码.....11-41

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

插图目录

图 11-1 IVE_SRC_FMT_SINGLE（单分量）数据格式示意 11-2
图 11-2 IVE_SRC_FMT_SP420 数据格式示意 11-3
图 11-3 IVE_SRC_FMT_SP422 数据格式示意 11-3
图 11-4 IVE_CSC_OUT_FMT_PLANAR 数据格式示意 11-4
图 11-5 IVE_CSC_OUT_FMT_PACKAGE 数据格式示意 11-5
图 11-6 Sobel 输出 H/V 数据格式示意 11-5
图 11-7 CANNY 输出格式为(MAG 和 ANG) 11-6
图 11-8 积分图输出格式示意 11-6
图 11-9 直方图输出格式示意 11-6

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 11-1 智能加速引擎 API 错误码 11-41

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

11 智能加速引擎

11.1 概述

IVE（Intelligent Video Engineering）模块是智能分析系统中的硬件加速模块。ARM 在做智能分析时，可以借助 IVE 完成一些基本操作，来加速智能分析。IVE 共提供了 14 个基本算子，包括：快速拷贝、模板滤波、色彩空间转换、模板滤波加色彩空间转换、sobel 边缘提取、canny 边缘提取、膨胀、腐蚀、图像二值化、图像与、图像减、图像或、积分图和直方图统计。

11.2 功能描述

11.2.1 重要概念

- 跨度（stride）：图像一行以像素计算的单元个数，像素位宽可以是 8bit，16bit 等。
- 数据源地址必须 8byte 对齐的跨度计算

  模板滤波、色彩空间转换、模板滤波加色彩空间转换、sobel 边缘提取、canny 边缘提取、膨胀、腐蚀、积分图和直方图统计，共 9 个算子要求输入输出地址必须 8byte 对齐。在配置它们的跨度时只需满足两个条件：跨度值为 8 对齐；跨度值大于或等于源数据宽度。具体，请参见如下公式：

  $$\begin{cases} stride \geq width \\ stride\%8 = 0 \end{cases}$$

- 数据源地址只需 byte 对齐的跨度计算

  快速拷贝、图像二值化、图像与、图像减和图像或，共 5 个算子。在配置它们的跨度时需要考虑 3 个因素：源数据地址、数据宽度和 8 字节对齐。具体，请参见如下公式：

  $$\begin{cases} stride \geq \{(8-((width+(src\%8))\%8))\%8+(src\%8)+width\} \\ stride\%8 = 0 \end{cases}$$

- 及时返回结果标志 bInstant

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

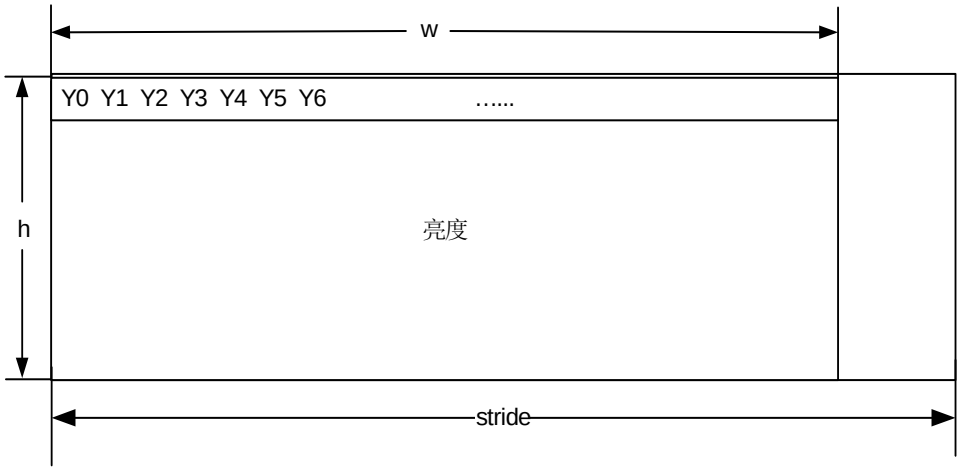
用户在创建某个任务后，希望及时得到该任务完成的信息，则需要在创建该任务时，将 bInstant 设置为 HI_TRUE。否则，如果用户不关心该任务是否完成，建议将 bInstant 设置为 HI_FALSE。

- 及时刷 cache

  IVE 硬件只能从 DDR 中获取数据，如果用户在调用 IVE 任务时，访问空间可 cache 而且 CPU 曾经访问，为了保证 IVE 输入输出数据不被 CPU cahce 干扰，此时用户需要调用 HI_MPI_SYS_MmzFlushCache 接口刷 cache，将数据从 cache 刷到 DDR，以供 IVE 使用。

- 输入数据类型
  - IVE_SRC_FMT_SINGLE：单分量格式，以 Byte 为单位的无符号数据。可以为任意数据，如 Y/Cb/Cr/R/G/B。取值范围为[0, 255]。如所图 11-1 示。
  - IVE_SRC_FMT_SP420：YCbCr420_SemiPlannar 数据格式，如图 11-2 所示。
  - IVE_SRC_FMT_SP422：YcbCr422 SemiPlannar 数据格式，如图 11-3 所示。

图11-1 IVE_SRC_FMT_SINGLE（单分量）数据格式示意



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图11-2 IVE_SRC_FMT_SP420 数据格式示意

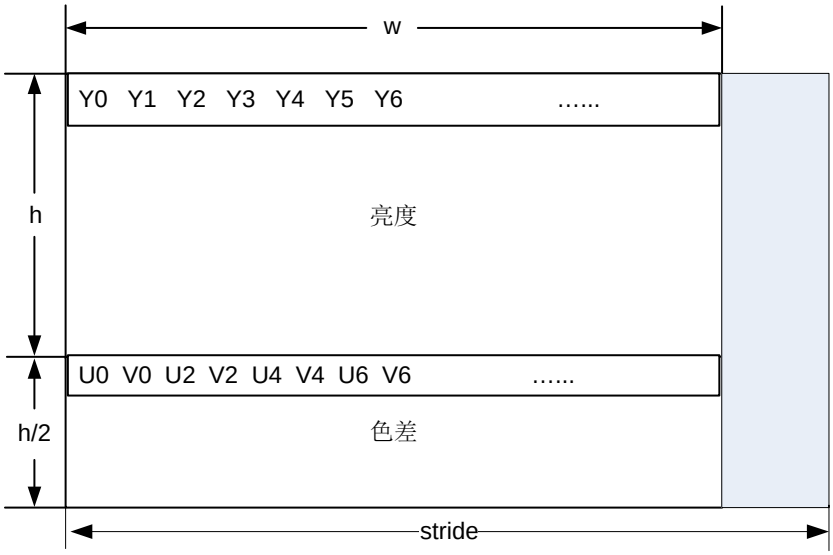
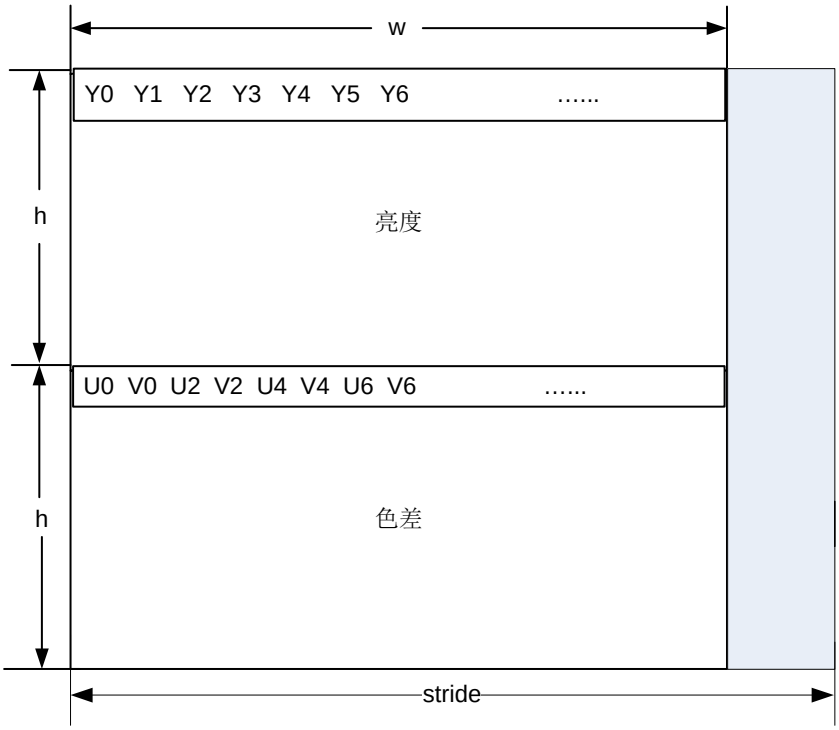


图11-3 IVE_SRC_FMT_SP422 数据格式示意

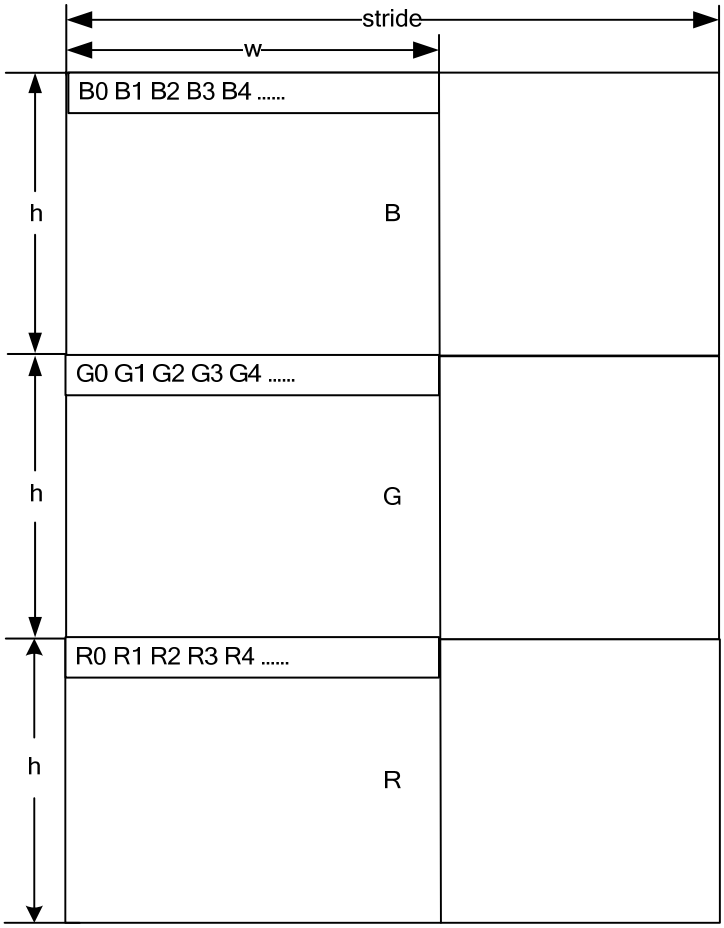


- 输出数据类型
  - 单分量：与输入类型中的单分量类型定义一致
  - IVE_CSC_OUT_FMT_PLANAR：格式如图 11-4 所示。
  - IVE_CSC_OUT_FMT_PACKAGE：格式如图 11-5 所示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

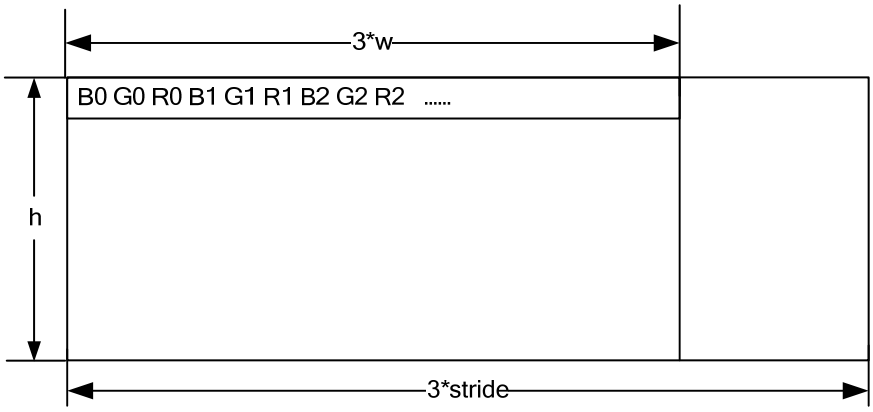
- Sobel 输出 H/V 数据格式：格式如图 11-6 所示。
- CANNY 输出格式为(MAG 和 ANG)：格式如图 11-7 所示。
- 积分图输出格式：在双字中，S（图像和）占低 28bit，SQ(图像平方和)占高 36bit。格式如图 11-8 所示。
- 直方图输出格式如图 11-9 所示。

图11-4 IVE_CSC_OUT_FMT_PLANAR 数据格式示意



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

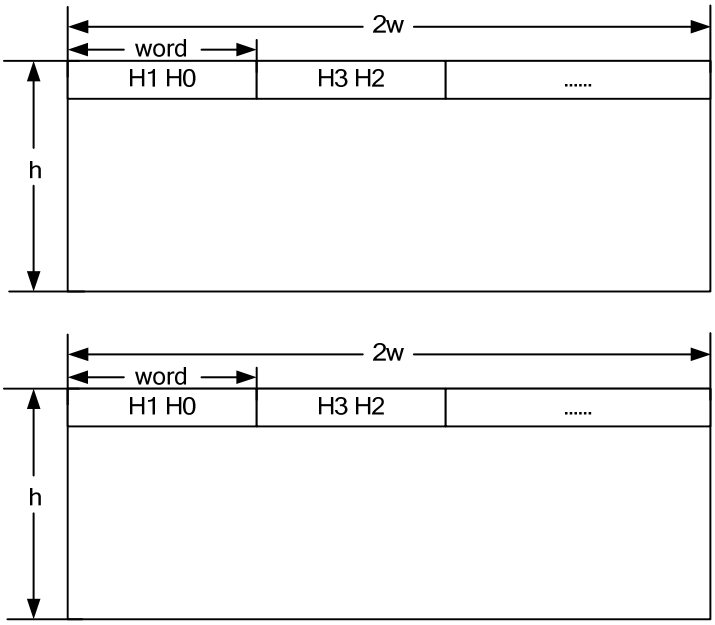
图11-5 IVE_CSC_OUT_FMT_PACKAGE 数据格式示意



说明

数据存放内存方式为 little endian 方式，为了方便描述，统一使用 Word、Double Word 作为存储单位进行描述。

图11-6 Sobel 输出 H/V 数据格式示意



海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

图11-7 CANNY 输出格式为(MAG 和 ANG)

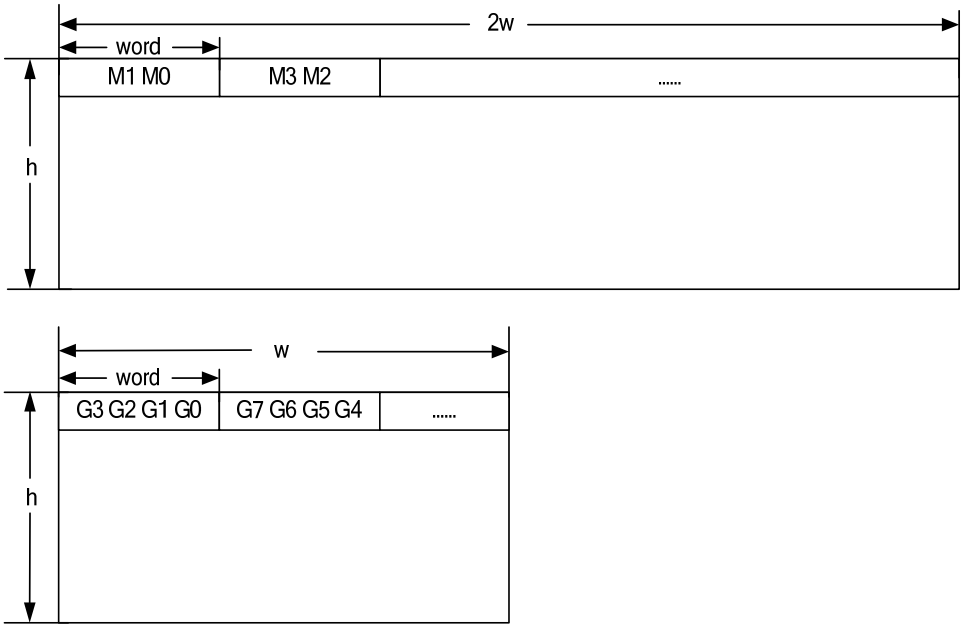


图11-8 积分图输出格式示意

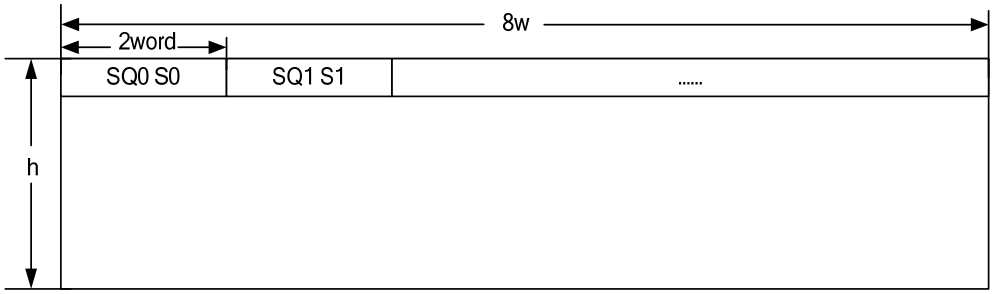
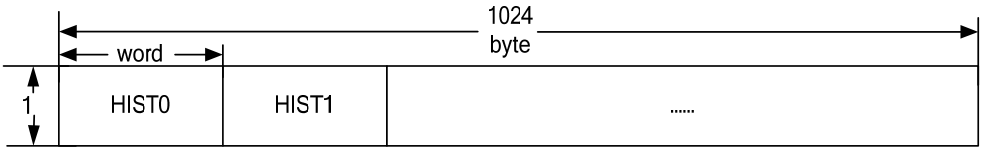


图11-9 直方图输出格式示意



11.2.2 使用示意

- 用户根据需求调用相应的算子接口，指定 bInstant 类型，创建任务，并根据需求记录该任务返回的 handle 号。
- 根据返回的 handle 号，指定阻塞方式，可以查询到该任务的完成状态。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

11.3 API 参考

IVE 模块提供了创建任务和查询任务的基本接口。

该功能模块提供以下 MPI：

- HI_MPI_IVE_DMA：创建快速拷贝任务。
- HI_MPI_IVE_FILTER：创建模板滤波任务。
- HI_MPI_IVE_CSC：创建色彩空间转换任务。
- HI_MPI_IVE_FILTER_AND_CSC：创建模板滤波加色彩空间转换复合任务
- HI_MPI_IVE_SOBEL：创建 sobel 边缘提取任务。
- HI_MPI_IVE_CANNY：创建 canny 边缘提取任务。
- HI_MPI_IVE_DILATE：创建膨胀任务。
- HI_MPI_IVE_ERODE：创建腐蚀任务。
- HI_MPI_IVE_THRESH：创建图像二值化任务。
- HI_MPI_IVE_AND：创建两图像相与任务。
- HI_MPI_IVE_SUB：创建两图像相减任务。
- HI_MPI_IVE_OR：创建两图像相或任务。
- HI_MPI_IVE_INTEG：创建积分图统计任务。
- HI_MPI_IVE_HIST：创建直方图统计任务。
- HI_MPI_IVE_Query：查询已创建任务完成情况。

HI_MPI_IVE_DMA

【描述】

创建快速拷贝任务，实现数据从一块内存搬运到另一块内存。

【语法】

```
HI_S32 HI_MPI_IVE_DMA(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针和输出数据指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址只要求 byte 对齐。
- 源数据宽高范围为：32x1～1920x1080。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。
- 计算公示：

  $I_{out}(x,y) = I(x,y) \qquad (0 \le x \le width, 0 \le y \le height)$

  其中 $I(x,y)$ 对应 pstSrc，$I_{out}(x,y)$ 对应 pstDst。

【举例】

```
HI_S32 s32Ret = HI_SUCCESS;
IVE_HANDLE IveHandle;
IVE_SRC_INFO_S stSrc;
IVE_MEM_INFO_S stDst;
HI_VOID *pVirtSrc;
HI_VOID *pVirtDst;
stSrc.stSrcMem.u32Stride  = 352;
stSrc.u32Height           = 288;
stSrc.u32Width            = 352;
stSrc.enSrcFmt            = IVE_SRC_FMT_SINGLE;
stDst.u32Stride           = 352;
bInstant                  = HI_TRUE;
s32Ret = HI_MPI_SYS_MmzAlloc_Cached(&stSrc.stSrcMem.u32PhyAddr,
&pVirtSrc, "User",HI_NULL, stSrc.u32Height * stSrc.stSrcMem.u32Stride);
memset(pVirtSrc,1,stSrc.u32Height * stSrc.stSrcMem.u32Stride);
s32Ret = HI_MPI_SYS_MmzAlloc_Cached(&stDst.u32PhyAddr, &pVirtDst,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
        "User", HI_NULL, stSrc.u32Height * stDst.u32Stride);

    s32Ret = HI_MPI_IVE_DMA(&IveHandle, &stSrc, &stDst, bInstant);

    HI_MPI_SYS_MmzFree(stSrc.stSrcMem.u32PhyAddr, pVirtSrc);
    HI_MPI_SYS_MmzFree(stDst.u32PhyAddr, pVirtDst);

    return s32Ret;
```

【相关主题】

无。

HI_MPI_IVE_FILTER

【描述】

创建模板滤波任务，通过配置不同的模板系数，可以实现不同的滤波。

【语法】

```
HI_S32 HI_MPI_IVE_FILTER(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, IVE_FILTER_CTRL_S *pstFilterCtrl,HI_BOOL
bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
pstFilterCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

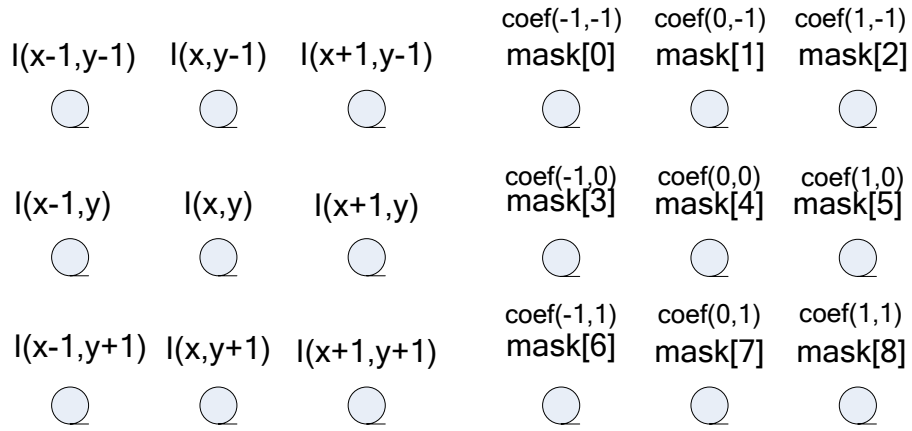
【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- handle 号由系统返回。
- 源数据指针、输出数据指针和控制信息指针不能为空。
- 源数据类型可以为单分量、SP420 或 SP422 格式。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1024。
- 源数据格式为 SP420 和 SP422 时，宽度必须为偶数。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。
- 计算公式



$$Iout(x,y)=\left\{\sum_{-2<j<2}\sum_{-2<i<2}I(x+i,y+j)\bullet coef(i,j)\right\}>>\text{norm}$$

其中，$I(x,y)$对应 pstSrc，$I_{out}(x,y)$对应 pstDst，$coef$ (mask)对应 pstCscCtrl 中的 as8Mask[9]，norm 对应 pstCscCtrl 中的 u8Norm。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_CSC

【描述】

创建色彩空间转换任务，实现从 YUV 空间到 RGB 空间的转换。

【语法】

```
HI_S32 HI_MPI_IVE_CSC(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, IVE_CSC_CTRL_S *pstCscCtrl,HI_BOOL bInstant);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
pstCscCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针、输出数据指针控制信息指针不能为空。
- 源数据类型可以为 SP420 或 SP422。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 源数据宽度必须为偶数。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。输出数据所需内存大小须配置为：源图像高 x 输出数据跨度 x3。
- 可以配置 2 种输出格式，具体参见 IVE_CSC_OUT_FMT_E。
- 支持 4 种工作模式，具体参见 IVE_CSC_MODE_E。
- 计算公式

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = A_{CSC} \begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} + B_{CSC}$$

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

其中，输入图像$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix}$对应 pstSrc，输出图像$\begin{bmatrix} R \\ G \\ B \end{bmatrix}$对应 pstDst 和 pstCscCtrl 的 enOutFmt，A_{CSC}、B_{CSC}由 pstCscCtrl 的 enCscMode 决定。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_FILTER_AND_CSC

【描述】

创建模板滤波加色彩空间转换复合任务，通过一次创建完成两种功能。

【语法】

```
HI_S32 HI_MPI_IVE_FILTER_AND_CSC(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S
*pstSrc, IVE_MEM_INFO_S *pstDst, IVE_FILTER_AND_CSC_CTRL_S
*pstFltCscCtrl,HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
pstFltCscCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针、输出数据指针和控制信息指针不能为空。
- 源数据类型可以为 SP420 或 SP422。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1024。
- 源数据宽度必须为偶数。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。输出数据所需内存大小须配置为：源图像高 x 输出数据跨度 x3。
- 可以配置 2 种输出格式，具体参见 IVE_CSC_OUT_FMT_E。
- 支持 4 种工作模式，具体参见 IVE_CSC_MODE_E。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_SOBEL

【描述】

创建 sobel 边缘提取任务。

【语法】

```
HI_S32 HI_MPI_IVE_SOBEL(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDstH, IVE_MEM_INFO_S *pstDstV, IVE_SOBEL_CTRL_S
*pstSobelCtrl,HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDstH	水平方向输出数据指针。	输入
pstDstV	竖直方向输出数据指针。	输入
pstSobelCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【返回值】

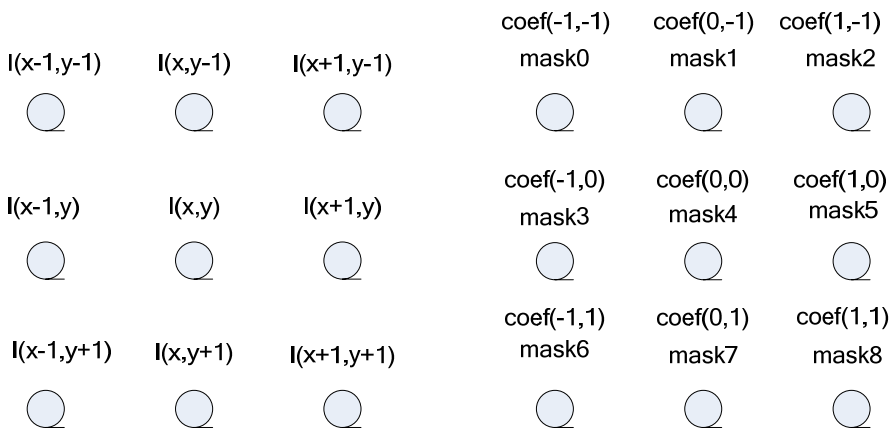
返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针、水平方向输出数据指针、竖直方向输出数据和控制信息指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1024。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。水平方向和竖直方向输出数据所需内存大小均必须配置为：源图像高 x 输出数据跨度 x2。
- 计算公式



$$Hout(x,y)=\sum_{-2<j<2}\sum_{-2<i<2}I(x+i,y+j)\bullet coef(i,j)$$

$$Vout(x,y)=\sum_{-2<j<2}\sum_{-2<i<2}I(x+i,y+j)\bullet coef(j,i)$$

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

其中，$I(x,y)$对应 pstSrc，$Hout(x,y)$对应 pstDstH，$Vout(x,y)$对应 pstDstV，$coef$ (mask)为 pstSobelCtrl 中的 as8Mask[9]

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_CANNY

【描述】

创建 canny 边缘提取任务。

【语法】

```
HI_S32 HI_MPI_IVE_CANNY(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDstMag, IVE_MEM_INFO_S *pstDstAng, IVE_CANNY_CTRL_S
*pstCannyCtrl,HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDstMag	输出幅值指针	输入
pstDstAng	输出角度指针。	输入
pstCannyCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

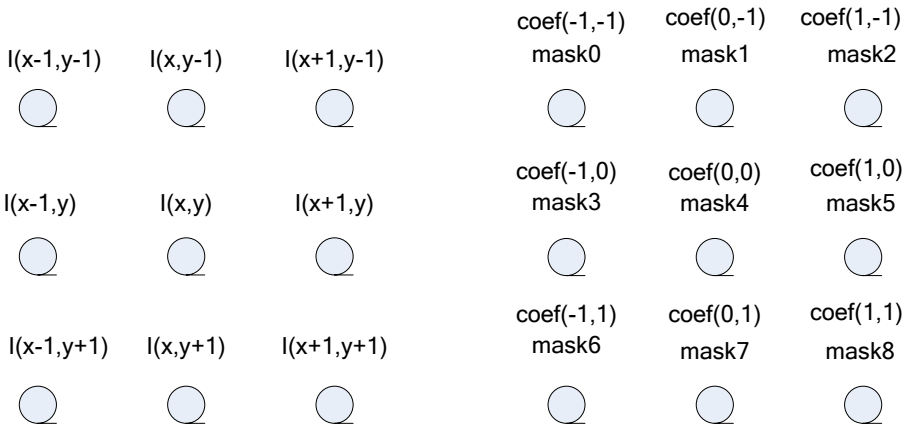
【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- handle 号由系统返回。
- 可以配置 2 种输出格式，具体参见 IVE_CANNY_OUT_FMT_E。
- 当配置输出结果仅仅输出幅值时，源数据指针、输出幅值指针和控制信息指针不能为空。
- 当配置输出结果同时输出幅值和角度值时，源数据指针、输出幅值指针、输出角度值指针和控制信息指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1024。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。输出幅值数据所需内存大小须配置为：源图像高 x 输出数据跨度 x2。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。输出角度数据所需内存大小须配置为：源图像高 x 输出数据跨度。
- 计算公式



$$Hout(x,y)=\sum_{-2<j<2}\sum_{-2<i<2}I(x+i,y+j)\bullet coef(i,j)$$

$$Vout(x,y)=\sum_{-2<j<2}\sum_{-2<i<2}I(x+i,y+j)\bullet coef(j,i)$$

$$Mag(x,y)=abs(Hout(x,y))+abs(Vout(x,y))$$

$$\theta=\left[\frac{\arctan(\frac{V}{H})*12}{\pi}\right]$$

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

其中，$I(x,y)$对应 pstSrc，$Mag(x,y)$对应 pstDstMag，$\theta(x,y)$对应 pstDstAng，$coef$ (mask)为 pstCannyCtrl 中的 as8Mask[9]。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_DILATE

【描述】

创建膨胀任务。

【语法】

```
HI_S32 HI_MPI_IVE_DILATE(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, IVE_DILATE_CTRL_S *pstDilateCtrl,HI_BOOL
bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
pstDilateCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

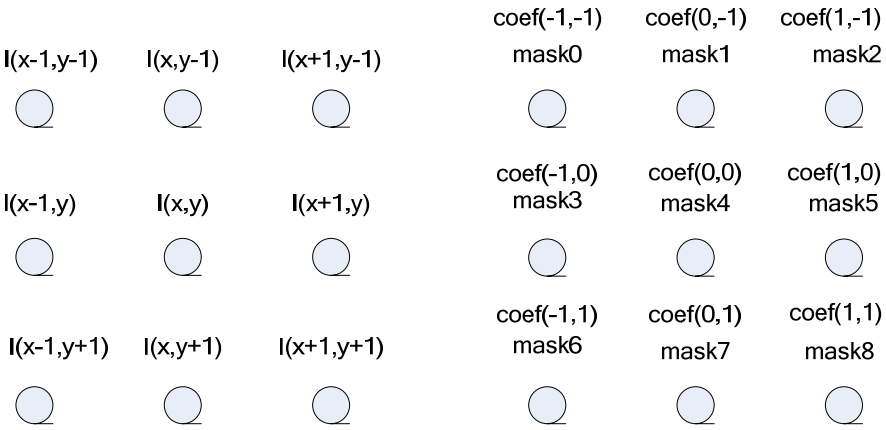
【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- handle 号由系统返回。
- 源数据指针、输出数据指针和控制信息指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1024。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。模板系数只能为 0 或 255。
- 计算公式



$$Iout(x,y)=f(I(x+(k\&3)-1,y+(k\%3)-1)\&coef((k\&3)-1,(k\%3)-1),|,0,8)$$

$$f(A_k,\Theta,c_{\min},c_{\max})=A_{c_{\min}}\Theta A_{c_{\min+1}}\cdots\Theta A_{c_{\max}}$$

其中，公式中|为位或运算，&为位与运算，%为除余运算。$I(x,y)$对应 pstSrc，$I_{out}(x,y)$对应 pstDst，$coef$ (mask)对应 pstDilateCtrl 中的 au8Mask[9]。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_ERODE

【描述】

创建腐蚀任务。

【语法】

```
HI_S32 HI_MPI_IVE_ERODE(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, IVE_ERODE_CTRL_S *pstErodeCtrl,HI_BOOL bInstant);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
pstErodeCtrl	控制信息指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

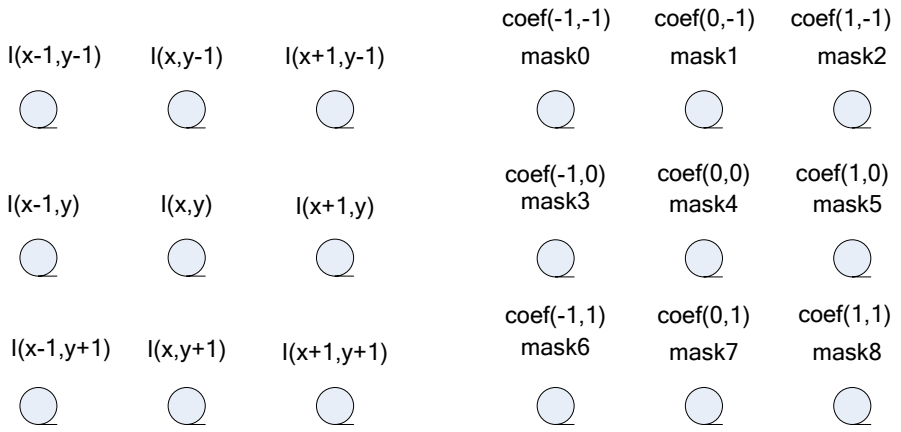
【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针、输出数据指针和控制信息指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址只要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1024。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。模板系数只能为 0 或 255。
- 计算公式

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司



$$Iout(x,y) = f(I(x+(k\&3)-1, y+(k\%3)-1) \mid coef((k\&3)-1,(k\%3)-1),\&,0,8)$$

$$f(A_k, \Theta, c_{\min}, c_{\max}) = A_{c_{\min}} \Theta A_{c_{\min+1}} \cdots \Theta A_{c_{\max}}$$

其中，公式中|为位或运算，&为位与运算，%为除余运算。$I(x,y)$对应 pstSrc，$I_{out}(x,y)$对应 pstDst，$coef$ (mask)对应 pstErodeCtrl 中的 au8Mask[9]。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_THRESH

【描述】

创建图像二值化任务。

【语法】

```
HI_S32 HI_MPI_IVE_THRESH(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, IVE_THRESH_CTRL_S *pstThreshCtrl, HI_BOOL
bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
pstThreshCtrl	控制信息指针。	输入

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针、输出数据指针和控制信息指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址只要求 byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。可以配置 3 种输出格式，具体参见 IVE_THRESH_OUT_FMT_E。
- 计算公式
  - mode=2：$I_{out}(x,y)=\begin{cases}\min Value & (I(x,y)\le threshold)\\ I(x,y) & (I(x,y)>threshold)\end{cases}$
  - mode=1：$I_{out}(x,y)=\begin{cases}I(x,y) & (I(x,y)\le threshold)\\ \max Value & (I(x,y)>threshold)\end{cases}$
  - mode=0：$I_{out}(x,y)=\begin{cases}\min Value & (I(x,y)\le threshold)\\ \max Value & (I(x,y)>threshold)\end{cases}$

  其中，$I(x,y)$对应 pstSrc，$I_{out}(x,y)$对应 pstDst，mode 对应 pstThreshCtrl 中 enOutFmt，threshold、minValue 和 maxValue 分别对应 pstThreshCtrl 的 u32Thresh、u32MinVal 和 u32MaxVal。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_IVE_AND

【描述】

创建两图像相与任务。

【语法】

```
HI_S32 HI_MPI_IVE_AND(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc1,
IVE_SRC_INFO_S *pstSrc2, IVE_MEM_INFO_S *pstDst, HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc1	源数据 1 指针。	输入
pstSrc2	源数据 2 指针	输入
pstDst	输出数据指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据 1 指针、源数据 2 指针和输出数据指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址只要求 byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。
- 计算公式

$$I_{out}(x,y) = I_{src1}(x,y) \& I_{src2}(x,y)$$

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

其中，$I_{src1}(x,y)$ 对应 pstSrc1，$I_{src2}(x,y)$ 对应 pstSrc2，$I_{out}(x,y)$ 对应 pstDst

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_SUB

【描述】

创建两图像相减任务。

【语法】

```
HI_S32 HI_MPI_IVE_SUB(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc1,
IVE_SRC_INFO_S *pstSrc2, IVE_MEM_INFO_S *pstDst, IVE_SUB_OUT_FMT_E
enOutFmt,HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc1	源数据 1 指针。	输入
pstSrc2	源数据 2 指针。	输入
pstDst	输出数据指针。	输入
enOutFmt	输出数据格式。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- handle 号由系统返回。
- 源数据 1 指针、源数据 2 指针和输出数据指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址只要求 byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。可以配置 2 种输出格式，具体参见 IVE_SUB_OUT_FMT_E。
- 计算公式

$$I_{out}(x,y) = abs(I_{src1}(x,y) - I_{src2}(x,y)) \qquad \mathrm{mod}\, e = 0$$
$$I_{out}(x,y) = (I_{src1}(x,y) - I_{src2}(x,y)) >> 1 \qquad \mathrm{mod}\, e = 1$$

其中，$I_{src1}(x,y)$ 对应 pstSrc1，$I_{src2}(x,y)$ 对应 pstSrc2，$I_{out}(x,y)$ 对应 pstDst ，$\mathrm{mod}\, e$ 对应 enOutFmt。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_OR

【描述】

创建两图像相或任务。

【语法】

```
HI_S32 HI_MPI_IVE_OR(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc1,
IVE_SRC_INFO_S *pstSrc2, IVE_MEM_INFO_S *pstDst, HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc1	源数据 1 指针。	输入
pstSrc2	源数据 2 指针。	输入
pstDst	输出数据指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据 1 指针、源数据 2 指针和输出数据指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址只要求 byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。也可以对数据使用此算子功能。
- 计算公式

  $$I_{out}(x,y) = I_{src1}(x,y) \mid I_{src2}(x,y)$$

  其中，$I_{src1}(x,y)$ 对应 pstSrc1，$I_{src2}(x,y)$ 对应 pstSrc2，$I_{out}(x,y)$ 对应 pstDst。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_INTEG

【描述】

创建积分图任务。

【语法】

```
HI_S32 HI_MPI_IVE_INTEG(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针和输出数据指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 输出数据跨度请按 11.2.1 “重要概念”中的跨度要求配置。输出数据所需内存大小须配置为：源图像高 x 输出数据跨度 x8。
- 计算公式

$$I_{sum}(x,y)=\sum_{i\geq 0}^{i\leq x}\sum_{j\geq 0}^{j\leq y}I(i,j)$$

$$I_{sq}(x,y)=\sum_{i\geq 0}^{i\leq x}\sum_{j\geq 0}^{j\leq y}(I(i,j)\bullet I(i,j))$$

$$I_{out}(x,y)=(I_{sum}(x,y)\,\&\,0\text{x}FFFFFFFF)\,|\,(I_{sq}(x,y)<<28)$$

其中，$I(x,y)$对应 pstSrc，$I_{out}(x,y)$对应 pstDst。

【举例】

无。

【相关主题】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_IVE_HIST

【描述】

创建直方图统计任务。

【语法】

```
HI_S32 HI_MPI_IVE_HIST(IVE_HANDLE *pIveHandle, IVE_SRC_INFO_S *pstSrc,
IVE_MEM_INFO_S *pstDst, HI_BOOL bInstant);
```

【参数】

参数名称	描述	输入/输出
pIveHandle	handle 号指针。	输出
pstSrc	源数据指针。	输入
pstDst	输出数据指针。	输入
bInstant	及时返回结果标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- handle 号由系统返回。
- 源数据指针和输出数据指针不能为空。
- 源数据类型只能为单分量。
- 源数据和输出数据地址要求 8byte 对齐。
- 源数据宽高范围为：64x64～1920x1080。
- 输出跨度设置为源数据的跨度，输出直方图数据所需内存大小配置为 1024Byte 即可。
- 计算公式

$$I_{out}(x) = \sum_i \sum_j ((I(i,j) == x)?1:0) \qquad x = 0 \ldots 255$$

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

其中，$I(x,y)$对应 pstSrc，$I_{out}(x,y)$对应 pstDst。

【举例】

无。

【相关主题】

无。

HI_MPI_IVE_Query

【描述】

查询已创建任务完成情况。

【语法】

```
HI_S32 HI_MPI_IVE_Query(IVE_HANDLE IveHandle, HI_BOOL *pbFinish, HI_BOOL bBlock);
```

【参数】

参数名称	描述	输入/输出
IveHandle	任务的 handle 号。	输入
pbFinish	任务完成状态指针。	输出
bBlock	是否阻塞查询标志。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，参见错误码。

【需求】

- 头文件：hi_comm_ive.h、mpi_ive.h
- 库文件：libmpi.a

【注意】

- pbFinish 指针不能为空。
- 在用户使用 IVE 任务结果前，为确保 IVE 任务已完成，用户可以使用阻塞方式调用此接口查询。另外，IVE 内部是按任务创建顺序依次执行任务的，所以用户不必每次都使用查询接口，如用户依次创建了 A，B 两个任务，那么如果 B 任务完

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成了，这个时候 A 任务肯定也完成了，此时使用 A 任务的结果时不必再次调用查询接口。

【举例】

```
HI_S32 s32Ret = HI_SUCCESS;
IVE_HANDLE IveHandle;
IVE_SRC_INFO_S stSrc;
IVE_MEM_INFO_S stDst;
HI_BOOL bInstant;
HI_VOID *pVirtSrc;
HI_VOID *pVirtDst;
HI_BOOL bFinish, bBlock;
stSrc.stSrcMem.u32Stride  = 352;
stSrc.u32Height           = 288;
stSrc.u32Width            = 352;
stSrc.enSrcFmt            = IVE_SRC_FMT_SINGLE;
stDst.u32Stride           = 352;
bInstant                  = HI_TRUE;

s32Ret = HI_MPI_SYS_MmzAlloc_Cached(&stSrc.stSrcMem.u32PhyAddr,
&pVirtSrc, "User", HI_NULL, stSrc.u32Height*stSrc.stSrcMem.u32Stride);

memset(pVirtSrc,1,stSrc.u32Height * stSrc.stSrcMem.u32Stride);

s32Ret = HI_MPI_SYS_MmzAlloc_Cached(&stDst.u32PhyAddr, &pVirtDst,
    "User", HI_NULL, stSrc.u32Height * stDst.u32Stride);

memset(pVirtDst,0,stSrc.u32Height * stDst.u32Stride);

s32Ret = HI_MPI_IVE_DMA(&IveHandle, &stSrc, &stDst, bInstant);

bBlock = HI_FALSE;
s32Ret = HI_MPI_IVE_Query(IveHandle, &bFinish, bBlock);
if (SUCCESS == s32Ret)
{
    printf("bFinish=%d\n", bFinish);
}

HI_MPI_SYS_MmzFree(stSrc.stSrcMem.u32PhyAddr, pVirtSrc);
HI_MPI_SYS_MmzFree(stDst.u32PhyAddr, pVirtDst);

return s32Ret;
```

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无。

11.4 数据类型

IVE 相关数据类型、数据结构定义如下：

- IVE_MAX_HANDLE_NR：定义任务的最大句柄数。
- IVE_MEM_INFO_S：定义输入或输出数据内存信息。
- IVE_SRC_FMT_E：定义源数据格式。
- IVE_SRC_INFO_S：定义源数据信息。
- IVE_CSC_OUT_FMT_E：定义色彩空间转换输出格式。
- IVE_CSC_MODE_E：定义色彩空间转换模式。
- IVE_CSC_CTRL_S：定义色彩空间转换控制信息。
- IVE_FILTER_CTRL_S：定义模板滤波控制信息。
- IVE_FILTER_AND_CSC_CTRL_S：定义模板滤波加色彩空间转换复合功能控制信息。
- IVE_SOBEL_CTRL_S：定义 sobel 边缘提取控制信息。
- IVE_CANNY_OUT_FMT_E：定义 canny 输出格式。
- IVE_CANNY_CTRL_S 定义 canny 边缘提取控制信息。
- IVE_DILATE_CTRL_S：定义膨胀控制信息。
- IVE_ERODE_CTRL_S：定义腐蚀控制信息。
- IVE_THRESH_OUT_FMT_E：定义图像二值化输出格式。
- IVE_THRESH_CTRL_S：定义图像二值化控制信息。
- IVE_SUB_OUT_FMT_E：定义两图像相减输出格式。

IVE_MAX_HANDLE_NR

【说明】

定义任务的最大句柄。

【定义】

```
#define IVE_MAX_HANDLE_NR 0x0FFFFFFF
```

【成员】

无。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

IVE_MEM_INFO_S

【说明】

定义源数据或输出数据内存信息。

【定义】

```
typedef struct hiIVE_DATA_ATTR_S
{
   HI_U32 u32PhyAddr;
   HI_U32 u32Stride;
}IVE_MEM_INFO_S;
```

【成员】

成员名称	描述
u32PhyAddr	数据物理地址。
u32Stride	数据跨度。

【注意事项】

不同的算子对数据输入输出地址是否对齐有不同的要求，同理，跨度也会出现差异。

【相关数据类型及接口】

无。

IVE_SRC_FMT_E

【说明】

定义源数据格式。

【定义】

```
typedef enum hiIVE_SRC_FMT_E
{
    IVE_SRC_FMT_SINGLE = 0, /*单分量*/
    IVE_SRC_FMT_SP420,      /*YUV SP420*/
    IVE_SRC_FMT_SP422,      /*YUV SP422*/
    IVE_SRC_FMT_BUTT
}IVE_SRC_FMT_E;
```

【成员】

成员名称	描述
IVE_SRC_FMT_SINGLE	源数据为单分量格式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
IVE_SRC_FMT_SP420	源数据为 SP420 格式。
IVE_SRC_FMT_SP422	源数据为 SP422 格式。

【注意事项】

- 模板滤波算子、色彩空间转换算子和模板滤波加色彩空间转换复合算子允许输入数据为 SP420 格式，或 SP422 格式。
- 其它算子输入只能为单分量。

【相关数据类型及接口】

无。

IVE_SRC_INFO_S

【说明】

定义源数据信息。

【定义】

```
typedef struct hiIVE_SRC_INFO_S
{
    IVE_SRC_FMT_E  enSrcFmt;
    IVE_MEM_INFO_S stSrcMem;
    HI_U32 u32Height;
    HI_U32 u32Width;
}IVE_SRC_INFO_S;
```

【成员】

成员名称	描述
enSrcFmt	源数据格式。
stSrcMem	源数据内存信息
u32Height	源数据的高度。
u32Width	源数据的宽度。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

IVE_CSC_OUT_FMT_E

【说明】

定义色彩空间转换输出格式。

【定义】

```
typedef enum hiIVE_CSC_OUT_FMT_E
{
    IVE_CSC_OUT_FMT_PACKAGE = 0,
    IVE_CSC_OUT_FMT_PLANAR,
    IVE_CSC_OUT_FMT_BUTT
}IVE_CSC_OUT_FMT_E;
```

【成员】

成员名称	描述
IVE_CSC_OUT_FMT_PACKAGE	package 格式。
IVE_CSC_OUT_FMT_PLANAR	planar 格式。

【注意事项】

无。

【相关数据类型及接口】

无。

IVE_CSC_MODE_E

【说明】

定义色彩空间转换模式。

【定义】

```
typedef enum hiIVE_CSC_MODE_E
{
    IVE_CSC_MODE_VIDEO_BT601_AND_BT656 = 0,
    IVE_CSC_MODE_VIDEO_BT709,
    IVE_CSC_MODE_PIC_BT601_AND_BT656,
    IVE_CSC_MODE_PIC_BT709,
    IVE_CSC_MODE_BUTT
}IVE_CSC_MODE_E;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
IVE_CSC_MODE_VIDEO_BT601_AND_BT656	BT601&BT656 的视频变换。
IVE_CSC_MODE_VIDEO_BT709	BT709 的视频变换。
IVE_CSC_MODE_PIC_BT601_AND_BT656	BT601&BT656 的图像变换。
IVE_CSC_MODE_PIC_BT709	BT709 的图像变换。

【注意事项】

- IVE_CSC_MODE_VIDEO_BT601_AND_BT656 和 IVE_CSC_MODE_VIDEO_BT709 模式，输出满足 16≤R，G，B≤235。
- IVE_CSC_MODE_PIC_BT601_AND_BT656 和 IVE_CSC_MODE_PIC_BT709 模式，输出满足 0≤R，G，B≤255。

【相关数据类型及接口】

无。

IVE_CSC_CTRL_S

【说明】

定义色彩空间转换控制信息。

【定义】

```
typedef struct hiIVE_CSC_CTRL_S
{
    IVE_CSC_OUT_FMT_E enOutFmt;
    IVE_CSC_MODE_E    enCscMode;
}IVE_CSC_CTRL_S;
```

【成员】

成员名称	描述
enOutFmt	输出格式。
enCscMode	工作模式。

【注意事项】

无。

【相关数据类型及接口】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

IVE_FILTER_CTRL_S

【说明】

定义模板滤波控制信息。

【定义】

```
typedef struct hiIVE_FILTER_CTRL_S
{
    HI_S8 as8Mask[9];
    HI_S8 u8Norm;
}IVE_FILTER_CTRL_S;
```

【成员】

成员名称	描述
as8Mask[9]	3x3 模板系数。
u8Norm	归一化参数。 取值范围：[0, 11]。

【注意事项】

通过配置不同的模板系数可以达到不同的滤波效果。

【相关数据类型及接口】

无。

IVE_FILTER_AND_CSC_CTRL_S

【说明】

定义模板滤波加色彩空间转换复合功能控制信息。

【定义】

```
typedef struct hiIVE_FILTER_AND_CSC_CTRL_S
{
    IVE_CSC_OUT_FMT_E enOutFmt;
    IVE_CSC_MODE_E enCscMode;
    HI_S8 as8Mask[9];
    HI_S8 u8Norm;
}IVE_FILTER_AND_CSC_CTRL_S;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enOutFmt	输出格式。
enCscMode	工作模式。
as8Mask[9]	3x3 模板系数。
u8Norm	归一化参数。 取值范围：[0, 11]。

【注意事项】

无。

【相关数据类型及接口】

无

IVE_SOBEL_CTRL_S

【说明】

定义 sobel 边缘提取控制信息。

【定义】

```
typedef struct hiIVE_SOBEL_CTRL_S
{
   HI_S8 as8Mask[9];
}IVE_SOBEL_CTRL_S;
```

【成员】

成员名称	描述
as8Mask[9]	3x3 模板系数。

【注意事项】

无。

【相关数据类型及接口】

无

IVE_CANNY_OUT_FMT_E

【说明】

定义 canny 输出格式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef enum hiIVE_CANNY_OUT_FMT_E
{
    IVE_CANNY_OUT_FMT_ONLY_MAG = 0,
    IVE_CANNY_OUT_FMT_MAG_AND_ANG,
    IVE_CANNY_OUT_FMT_BUTT
}IVE_CANNY_OUT_FMT_E;
```

【成员】

成员名称	描述
IVE_CANNY_OUT_FMT_ONLY_MAG	仅输出幅值。
IVE_CANNY_OUT_FMT_MAG_AND_ANG	同时输出幅值和角度值。

【注意事项】

无。

【相关数据类型及接口】

无。

IVE_CANNY_CTRL_S

【说明】

定义 canny 边缘提取控制信息。

【定义】

```
typedef struct hiIVE_CANNY_CTRL_S
{
    IVE_CANNY_OUT_FMT_E enOutFmt;
    HI_S8 as8Mask[9];
}IVE_CANNY_CTRL_S;
```

【成员】

成员名称	描述
enOutFmt	输出格式。
as8Mask[9]	3x3 模板系数。

【注意事项】

无。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关数据类型及接口】

无。

IVE_DILATE_CTRL_S

【说明】

定义膨胀控制信息。

【定义】

```
typedef struct hiIVE_DILATE_CTRL_S
{
    HI_U8 au8Mask[9];
}IVE_DILATE_CTRL_S;
```

【成员】

成员名称	描述
au8Mask[9]	3x3 模板系数。 取值范围：0 或 255。

【注意事项】

无。

【相关数据类型及接口】

无。

IVE_ERODE_CTRL_S

【说明】

定义腐蚀控制信息。

【定义】

```
typedef struct hiIVE_ERODE_CTRL_S
{
    HI_U8 au8Mask[9];
}IVE_ERODE_CTRL_S;
```

【成员】

成员名称	描述
au8Mask[9]	3x3 模板系数。 取值范围：0 或 255。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无。

【相关数据类型及接口】

无。

IVE_THRESH_OUT_FMT_E

【说明】

定义图像二值化输出格式。

【定义】

```
typedef enum hiIVE_THRESH_OUT_FMT_E
{
    IVE_THRESH_OUT_FMT_BINARY = 0,
    IVE_THRESH_OUT_FMT_TRUNC,
    IVE_THRESH_OUT_FMT_TOZERO,
    IVE_THRESH_OUT_FMT_BUTT
}IVE_THRESH_OUT_FMT_E;
```

【成员】

成员名称	描述
IVE_THRESH_OUT_FMT_BINARY	源值比 thresh 值大的，取最大值，比 thresh 小或等的，取最小值。
IVE_THRESH_OUT_FMT_TRUNC	源值比 thresh 值大的，取最大值，比 thresh 小或等的不变。
IVE_THRESH_OUT_FMT_TOZERO	源值比 thresh 值大的不变，比 thresh 小或等的取最小值。

【注意事项】

无。

【相关数据类型及接口】

无。

IVE_THRESH_CTRL_S

【说明】

定义图像二值化控制信息。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【定义】

```
typedef struct hiIVE_THRESH_CTRL_S
{
    IVE_THRESH_OUT_FMT_E enOutFmt;
    HI_U32 u32Thresh;
    HI_U32 u32MinVal;
    HI_U32 u32MaxVal;
}IVE_THRESH_CTRL_S;
```

【成员】

成员名称	描述
enOutFmt	输出格式。
u32Thresh	阈值。 取值范围：[0,255]。
u32MinVal	最小值。 取值范围：[0,255]。
u32MaxVal	最大值。 取值范围：[0,255]。

【注意事项】

无。

【相关数据类型及接口】

无。

IVE_SUB_OUT_FMT_E

【说明】

定义两图像相减输出格式。

【定义】

```
typedef enum hiIVE_SUB_OUT_FMT_E
{
    IVE_SUB_OUT_FMT_ABS = 0,
    IVE_SUB_OUT_FMT_SFR,
    IVE_SUB_OUT_FMT_BUTT
}IVE_SUB_OUT_FMT_E;
```

【成员】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
IVE_SUB_OUT_FMT_ABS	取差的绝对值。
IVE_SUB_OUT_FMT_SFR	将结果右移一位输出，保留符号位。

【注意事项】

无。

【相关数据类型及接口】

无。

11.5 错误码

智能加速引擎 API 错误码如表 11-1 所示。

表11-1 智能加速引擎 API 错误码

错误代码	宏定义	描述
0xA01D8001	HI_ERR_IVE_INVALID_DEVID	设备 ID 超出合法范围
0xA01D8002	HI_ERR_IVE_INVALID_CHNID	通道组号错误或无效区域句柄
0xA01D8003	HI_ERR_IVE_ILLEGAL_PARAM	参数超出合法范围
0xA01D8004	HI_ERR_IVE_EXIST	重复创建已存在的设备、通道或资源
0xA01D8005	HI_ERR_IVE_UNEXIST	试图使用或者销毁不存在的设备、通道或者资源
0xA01D8006	HI_ERR_IVE_NULL_PTR	函数参数中有空指针
0xA01D8007	HI_ERR_IVE_NOT_CONFIG	模块没有配置
0xA01D8008	HI_ERR_IVE_NOT_SUPPORT	不支持的参数或者功能
0xA01D8009	HI_ERR_IVE_NOT_PERM	该操作不允许，如试图修改静态配置参数
0xA01D800C	HI_ERR_IVE_NOMEM	分配内存失败，如系统内存不足
0xA01D800D	HI_ERR_IVE_NOBUF	分配缓存失败，如申请的数据缓冲区太大
0xA01D800E	HI_ERR_IVE_BUF_EMPTY	缓冲区中无数据

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

错误代码	宏定义	描述
0xA01D800F	HI_ERR_IVE_BUF_FULL	缓冲区中数据满
0xA01D8010	HI_ERR_IVE_NOTREADY	系统没有初始化或没有加载相应模块
0xA01D8011	HI_ERR_IVE_BADADDR	地址非法
0xA01D8012	HI_ERR_IVE_BUSY	系统忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

12 HDMI 12-1
12.1 概述.....12-1
12.2 重要概念.....12-1
12.3 API 参考.....12-1
12.4 数据类型.....12-19
12.5 错误码.....12-41

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

表格目录

表 12-1 HDMI API 错误码.....12-41

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

12 HDMI

12.1 概述

Hi35xx 芯片的内置 HDMI（High-Definition MultiMedia Interface）模块支持视频的 HDMI 输出。

12.2 重要概念

HDMI 的音频不能单独输出，必须依赖于视频输出，且 HDMI 的时钟来源于 VO 的时钟，因此接口调用顺序上需要先使能 VO，再调用 HDMI 接口，然后配置音视频输出。此外 Hi35xx 芯片内置的 HDMI 不支持 HDCP 功能。

12.3 API 参考

该功能模块提供以下 MPI：

- HI_MPI_HDMI_Init：初始化 HDMI。
- HI_MPI_HDMI_DeInit：去初始化 HDMI。
- HI_MPI_HDMI_Open：打开 HDMI。
- HI_MPI_HDMI_Close：关闭 HDMI。
- HI_MPI_HDMI_SetAttr：设置 HDMI 属性。
- HI_MPI_HDMI_GetAttr：获取 HDMI 属性。
- HI_MPI_HDMI_Start：启动 HDMI。
- HI_MPI_HDMI_Stop：停止 HDMI。
- HI_MPI_HDMI_GetSinkCapability：获取 HDMI Sink 端的能力级。
- HI_MPI_HDMI_SetAVMute：设置 HDMI 音视频静音功能。
- HI_MPI_HDMI_Force_GetEDID：获取 HDMI 的 EDID 信息。
- HI_MPI_HDMI_SetDeepColor：设置 HDMI 的 Deep Color 模式。
- HI_MPI_HDMI_GetDeepColor：获取 HDMI 的 Deep Color 模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_HDMI_SetCsc：设置设备 HDMI 的图像输出效果。
- HI_MPI_HDMI_GetCsc：获取设备 HDMI 的图像输出效果。
- HI_MPI_HDMI_SetInfoFrame：设置 HDMI 的信息帧。
- HI_MPI_HDMI_GetInfoFrame：获取 HDMI 的信息帧。

HI_MPI_HDMI_Init

【描述】

初始化 HDMI。

【语法】

```
HI_S32 HI_MPI_HDMI_Init(HI_HDMI_INIT_PARA_S *pstHdmiPara);
```

【参数】

参数名称	描述	输入/输出
pstHdmiPara	初始化参数结构体指针	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

HDMI 在内核态有默认的事件响应处理函数，如果用户不需要对事件做特殊响应，可以将初始化参数结构体中的函数指针置为 NULL。

【举例】

```
HI_HDMI_INIT_PARA_S stHdmiPara;
HI_HDMI_ATTR_S      stAttr;
HI_HDMI_VIDEO_FMT_E enVideoFmt;

/* 启动VO */

enVideoFmt = HI_HDMI_VIDEO_FMT_1080P_60;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
/* 启动HDMI */
stHdmiPara.enForceMode = HI_HDMI_FORCE_HDMI;
stHdmiPara.pCallBackArgs = NULL;
stHdmiPara.pfnHdmiEventCallback = NULL;
HI_MPI_HDMI_Init(&stHdmiPara);

HI_MPI_HDMI_Open(0);

HI_MPI_HDMI_GetAttr(0, &stAttr);
stAttr.bEnableHdmi = HI_TRUE;
stAttr.bEnableVideo = HI_TRUE;
stAttr.enVideoFmt = enVideoFmt;

stAttr.enVidOutMode = HI_HDMI_VIDEO_MODE_YCBCR444;
stAttr.enDeepColorMode = HI_HDMI_DEEP_COLOR_24BIT;
stAttr.bxvYCCMode = HI_FALSE;

stAttr.bEnableAudio = HI_FALSE;
stAttr.enSoundIntf = HI_HDMI_SND_INTERFACE_I2S;
stAttr.bIsMultiChannel = HI_FALSE;

stAttr.enBitDepth = HI_HDMI_BIT_DEPTH_16;

stAttr.bEnableAviInfoFrame = HI_TRUE;
stAttr.bEnableAudInfoFrame = HI_TRUE;
stAttr.bEnableSpdInfoFrame = HI_FALSE;
stAttr.bEnableMpegInfoFrame = HI_FALSE;

stAttr.bDebugFlag = HI_FALSE;
stAttr.bHDCPEnable = HI_FALSE;

stAttr.b3DEnable = HI_FALSE;
HI_MPI_HDMI_SetAttr(0, &stAttr);

HI_MPI_HDMI_Start(0);

/* 停止HDMI */
HI_MPI_HDMI_Stop(0);

HI_MPI_HDMI_Close(0);

HI_MPI_HDMI_DeInit();
```

【相关主题】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_HDMI_DeInit

HI_MPI_HDMI_DeInit

【描述】

去初始化 HDMI。

【语法】

```
HI_S32 HI_MPI_HDMI_DeInit(HI_VOID);
```

【参数】

参数名称	描述	输入/输出
-	-	-

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

无

【举例】

请参见 HI_MPI_HDMI_Init 的举例。

【相关主题】

HI_MPI_HDMI_Init

HI_MPI_HDMI_Open

【描述】

打开 HDMI。

【语法】

```
HI_S32 HI_MPI_HDMI_Open(HI_HDMI_ID_E enHdmi);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 打开 HDMI 之前必须先调用 HI_MPI_HDMI_Init。
- 重复打开 HDMI 返回成功。

【举例】

请参见 HI_MPI_HDMI_Init 的举例。

【相关主题】

HI_MPI_HDMI_Close

HI_MPI_HDMI_Close

【描述】

关闭 HDMI。

【语法】

```
HI_S32 HI_MPI_HDMI_Close(HI_HDMI_ID_E enHdmi);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

重复关闭返回成功。

【举例】

请参见 HI_MPI_HDMI_Init 的举例。

【相关主题】

HI_MPI_HDMI_Open

HI_MPI_HDMI_SetAttr

【描述】

设置 HDMI 属性。

【语法】

```
HI_S32 HI_MPI_HDMI_SetAttr(HI_HDMI_ID_E enHdmi, HI_HDMI_ATTR_S *pstAttr);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstAttr	HDMI 属性结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 设置 HDMI 属性之前需要打开 HDMI。
- 建议用户调用 HI_MPI_HDMI_Start 之前设置 HDMI 属性。
- 如果在调用 HI_MPI_HDMI_Start 之后设置 HDMI 属性，建议先调用 HI_MPI_HDMI_SetAVMute 设置音视频静音，然后再设置新的 HDMI 属性，最后关闭音视频静音。

【举例】

```
HI_HDMI_INIT_PARA_S stHdmiPara;
HI_HDMI_ATTR_S      stAttr;
HI_HDMI_VIDEO_FMT_E enVideoFmt;

/* 启动VO */

/* 启动HDMI */

/* HDMI start后设置HDMI属性 */
HI_MPI_HDMI_SetAVMute(0, HI_TRUE);
HI_MPI_HDMI_GetAttr(0, &stAttr);

/* 用户更改HDMI属性 */
stAttr.enVideoFmt = HI_HDMI_VIDEO_FMT_720P_60;

HI_MPI_HDMI_SetAttr(0, &stAttr);
HI_MPI_HDMI_SetAVMute(0, HI_FALSE);
```

【相关主题】

HI_MPI_HDMI_SetAttr

HI_MPI_HDMI_GetAttr

【描述】

获取 HDMI 属性。

【语法】

```
HI_S32 HI_MPI_HDMI_GetAttr(HI_HDMI_ID_E enHdmi, HI_HDMI_ATTR_S *pstAttr);
```

【参数】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstAttr	HDMI 属性结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

无

【举例】

请参见 HI_MPI_HDMI_Init 的举例。

【相关主题】

HI_MPI_HDMI_SetAttr

HI_MPI_HDMI_Start

【描述】

启动 HDMI。

【语法】

```
HI_S32 HI_MPI_HDMI_Start(HI_HDMI_ID_E enHdmi);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入

【返回值】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

启动 HDMI 前必须先调用 HI_MPI_HDMI_Open。

【举例】

请参见 HI_MPI_HDMI_Init 的举例。

【相关主题】

HI_MPI_HDMI_Stop

HI_MPI_HDMI_Stop

【描述】

停止 HDMI。

【语法】

```
HI_S32 HI_MPI_HDMI_Stop(HI_HDMI_ID_E enHdmi);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libhdmi.a

【注意】

无

【举例】

请参见 HI_MPI_HDMI_Init 的举例。

【相关主题】

HI_MPI_HDMI_Start

HI_MPI_HDMI_GetSinkCapability

【描述】

获取 HDMI Sink 端的能力级。

【语法】

```
HI_S32 HI_MPI_HDMI_GetSinkCapability(HI_HDMI_ID_E enHdmi,
    HI_HDMI_SINK_CAPABILITY_S *pstSinkCap);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstSinkCap	HDMI Sink 端能力级结构体指针。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

调用前必须先打开 HDMI。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_HDMI_INIT_PARA_S stHdmiPara;
HI_HDMI_ATTR_S      stAttr;
HI_HDMI_VIDEO_FMT_E enVideoFmt;
HI_HDMI_SINK_CAPABILITY_S stSinkCap;

/* 启动VO */

/* 启动HDMI */

/* 获取HDMI Sink端能力级 */
HI_MPI_HDMI_GetSinkCapability(0, &stSinkCap);
```

【相关主题】

无

HI_MPI_HDMI_SetAVMute

【描述】

设置 HDMI 音视频静音功能。

【语法】

```
HI_S32 HI_MPI_HDMI_SetAVMute(HI_HDMI_ID_E enHdmi, HI_BOOL bAvMute);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
bAvMute	是否设置 HDMI 音视频静音。取值范围： • HI_TRUE：设置音视频静音。 • HI_FALSE：关闭音视频静音。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 库文件：libhdmi.a

【注意】

无

【举例】

请参见 HI_MPI_HDMI_SetAttr 的举例。

【相关主题】

无

HI_MPI_HDMI_Force_GetEDID

【描述】

获取 HDMI 的 EDID 信息。

【语法】

```
HI_S32 HI_MPI_HDMI_Force_GetEDID(HI_HDMI_ID_E enHdmi, HI_HDMI_EDID_S
*pstEdidData);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstEdidData	HDMI 的 EDID 信息。	输出

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

调用前必须先打开 HDMI。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
HI_HDMI_INIT_PARA_S stHdmiPara;
HI_HDMI_ATTR_S      stAttr;
HI_HDMI_EDID_S stEdidData;

/* 启动VO */

/* 启动HDMI */

/* 获取HDMI EDID信息 */

HI_MPI_HDMI_Force_GetEDID(0, &stEdidData);
```

【相关主题】

无

HI_MPI_HDMI_SetDeepColor

【描述】

设置 Deep Color 模式。

【语法】

```
HI_S32 HI_MPI_HDMI_SetDeepColor(HI_HDMI_ID_E enHdmi, HI_HDMI_DEEP_COLOR_E enDeepColor);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
enDeepColor	HDMI 的 Deep Color 模式。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意】

- 该接口属于高级接口，一般不需要调用。
- 调用前必须先打开 HDMI。
- 该接口可用于设置 HI_MPI_HDMI_SetAttr 不支持的某些 Deep Color 模式。

【举例】

```
HI_HDMI_INIT_PARA_S stHdmiPara;
HI_HDMI_ATTR_S      stAttr;
/* 启动 VO */
/* 启动 HDMI */
/* 设置 Deep color 模式 */
HI_MPI_HDMI_SetDeepColor (0, HI_HDMI_DEEP_COLOR_24BIT);
```

【相关主题】

无

HI_MPI_HDMI_GetDeepColor

【描述】

获取 Deep Color 模式。

【语法】

```
HI_S32 HI_MPI_HDMI_GetDeepColor(HI_HDMI_ID_E enHdmi, HI_HDMI_DEEP_COLOR_E
*penDeepColor);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
penDeepColor	HDMI 的 Deep Color 模式指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 该接口属于高级接口，一般不需要调用。
- 调用前必须先打开 HDMI。

【举例】

无

【相关主题】

HI_MPI_HDMI_SetDeepColor

HI_MPI_HDMI_SetCsc

【描述】

设置设备 HDMI 的图像输出效果。

【语法】

```
HI_S32 HI_MPI_HDMI_SetCsc(HI_HDMI_ID_E enHdmi, HI_HDMI_CSC_S *pstCsc);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstCsc	图像输出效果结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 该接口只适用于 Hi3520D。
- 调用前必须先打开 HDMI。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- 该接口主要用于调节图像的输出效果，包括亮度、对比度、色调、饱和度，其取值范围为[0, 100]。
- CSC 转换矩阵只能选择 HI_HDMI_CSC_MATRIX_RGB_TO_BT601_PC 和 HI_HDMI_CSC_MATRIX_RGB_TO_BT709_PC。

【举例】

无

【相关主题】

HI_MPI_HDMI_GetCsc

HI_MPI_HDMI_GetCsc

【描述】

获取设备 HDMI 输出图像效果。

【语法】

```
HI_S32 HI_MPI_HDMI_GetCsc(HI_HDMI_ID_E enHdmi, HI_HDMI_CSC_S *pstCsc)
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstCsc	HDMI 图像输出效果结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 该接口只适用于 Hi3520D。
- 调用前必须先打开 HDMI。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无

【相关主题】

HI_MPI_HDMI_SetCsc

HI_MPI_HDMI_SetInfoFrame

【描述】

设置 HDMI 的信息帧。

【语法】

```
HI_S32 HI_MPI_HDMI_SetInfoFrame(HI_HDMI_ID_E enHdmi, HI_HDMI_INFOFRAME_S
*pstInfoFrame);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
pstInfoFrame	图像信息帧结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 该接口一般用于用户态回调函数中配置信息帧属性。其他情况，一般不需调用本接口。
- 调用前必须先打开 HDMI。
- 本接口只支持设置 AVI 和 AUDIO 信息帧，其他类型的信息帧设置无效，且 AVI 信息不支持 HI_HDMI_VIDEO_MODE_YCBCR422 输出类型。
- 建议设置前先用。

【举例】

无

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【相关主题】

HI_MPI_HDMI_GetInfoFrame

HI_MPI_HDMI_GetInfoFrame

【描述】

获取 HDMI 的信息帧。

【语法】

```
HI_S32 HI_MPI_HDMI_GetInfoFrame(HI_HDMI_ID_E enHdmi,
HI_HDMI_INFOFRAME_TYPE_E enInfoFrameType, HI_HDMI_INFOFRAME_S
*pstInfoFrame);
```

【参数】

参数名称	描述	输入/输出
enHdmi	HDMI 接口号。 取值范围：0	输入
enInfoFrameType	信息帧类型。	输入
pstInfoFrame	图像信息帧结构体指针。	输入

【返回值】

返回值	描述
0	成功。
非 0	失败，其值为错误码。

【需求】

- 头文件：mpi_hdmi.h、hi_comm_hdmi.h
- 库文件：libhdmi.a

【注意】

- 该接口一般用于用户态回调函数中配置信息帧属性。其他情况，一般不需调用本接口。
- 调用前必须先打开 HDMI。
- 本接口只支持设置 AVI 和 AUDIO 信息帧，其他类型的信息帧设置无效，且 AVI 信息不支持 HI_HDMI_VIDEO_MODE_YCBCR422 输出类型。
- 建议设置前先用。

【举例】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

无

【相关主题】

HI_MPI_HDMI_SetInfoFrame

12.4 数据类型

HDMI 相关数据类型定义如下：

- HI_HDMI_ID_E：定义 HDMI 接口号。
- HI_HDMI_INIT_PARA_S：定义 HDMI 初始化属性结构体。
- HI_HDMI_EVENT_TYPE_E：定义 HDMI 事件通知枚举。
- HI_HDMI_FORCE_ACTION_E：定义 HDMI 强制输出模式枚举。
- HI_HDMI_ATTR_S：定义 HDMI 属性结构体。
- HI_HDMI_VIDEO_FMT_E：定义 HDMI 视频制式枚举。
- HI_HDMI_VIDEO_MODE_E：定义 HDMI 颜色空间类型枚举。
- HI_HDMI_DEEP_COLOR_E：定义 HDMI 深色模式枚举。
- HI_HDMI_SND_INTERFACE_E：定义 HDMI 音频输出接口类型枚举。
- HI_HDMI_SAMPLE_RATE_E：定义 HDMI 音频输出采样率枚举。
- HI_HDMI_BIT_DEPTH_E：定义 HDMI 音频输出采样位宽枚举。
- HI_HDMI_SINK_CAPABILITY_S：定义 HDMI Sink 端能力级结构体。
- HI_HDMI_CSC_S：定义 HDMI 图像输出效果结构体。
- HI_HDMI_INFOFRAME_TYPE_E：定义 HDMI 信息帧类型枚举。
- HI_HDMI_AVI_INFOFRAME_VER2_S：定义 AVI 信息帧结构体。
- HI_HDMI_AUD_INFOFRAME_VER1_S：定义 AUDIO 信息帧结构体。
- HI_HDMI_SPD_INFOFRAME_S：定义 SPD 信息帧结构体。
- HI_HDMI_MPEGSOURCE_INFOFRAME_S：定义 MPEGSource 信息帧结构体。
- HI_HDMI_VENDORSPEC_INFOFRAME_S：定义 VS 信息帧结构体。
- HI_HMDI_INFORFRAME_UNIT_U：定义 HDMI 信息帧联合体。
- HI_HDMI_INFOFRAME_S：定义 HDMI 信息帧结构体。

HI_HDMI_ID_E

【说明】

定义 HDMI 接口号。

【定义】

```
typedef enum hiHDMI_ID_E
{
    HI_HDMI_ID_0        = 0,
    HI_HDMI_ID_BUTT
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
} HI_HDMI_ID_E;
```

【成员】

成员名称	描述
HI_HDMI_ID_0	HDMI 接口 0

【注意事项】

无

【相关数据类型及接口】

无

HI_HDMI_INIT_PARA_S

【说明】

定义 HDMI 初始化属性结构体。

【定义】

```
typedef struct hiHDMI_INIT_PARA_S
{
    HI_HDMI_CallBack   pfnHdmiEventCallback;
    HI_VOID            *pCallBackArgs;
    HI_HDMI_FORCE_ACTION_E enForceMode;
}HI_HDMI_INIT_PARA_S;
```

【成员】

成员名称	描述
pfnHdmiEventCallback	事件处理回调函数。
pCallBackArgs	回调函数参数私有数据。
enForceMode	HDMI 强制输出模式。

【注意事项】

- HDMI 在内核态有默认的事件响应处理函数，如果用户不需要对事件做特殊响应，可以将初始化参数结构体中的函数指针置为 NULL。
- Hi3531 的 HDMI 不支持 HDCP，HDCP 相关的事件为无效事件。
- 如果显示器不支持 HDMI，enForceMode 可以设置为 HI_HDMI_FORCE_DVI，强制 DVI 输出，否则建议设置为 HI_HDMI_FORCE_HDMI，强制 HDMI 输出。

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_MPI_HDMI_Init

HI_HDMI_EVENT_TYPE_E

【说明】

定义 HDMI 事件通知枚举。

【定义】

```
typedef enum hiHDMI_EVENT_TYPE_E
{
    HI_HDMI_EVENT_HOTPLUG = 0x10,
    HI_HDMI_EVENT_NO_PLUG,
    HI_HDMI_EVENT_EDID_FAIL,
    HI_HDMI_EVENT_HDCP_FAIL,
    HI_HDMI_EVENT_HDCP_SUCCESS,
    HI_HDMI_EVENT_HDCP_USERSETTING,
    HI_HDMI_EVENT_BUTT
}HI_HDMI_EVENT_TYPE_E;
```

【成员】

成员名称	描述
HI_HDMI_EVENT_HOTPLUG	HDMI 热插拔事件。
HI_HDMI_EVENT_NO_PLUG	HDMI Cable 没有连接事件。
HI_HDMI_EVENT_EDID_FAIL	HDMI EDID 读取失败事件。
HI_HDMI_EVENT_HDCP_FAIL	HDCP 验证失败事件。
HI_HDMI_EVENT_HDCP_SUCCESS	HDCP 验证成功。
HI_HDMI_EVENT_HDCP_USERSETTING	HDCP Reset。

【注意事项】

Hi35xx 的 HDMI 不支持 HDCP，HDCP 相关的事件为无效事件。

【相关数据类型及接口】

HI_MPI_HDMI_Init

HI_HDMI_FORCE_ACTION_E

【说明】

定义 HDMI 强制输出模式枚举。

【定义】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
typedef enum hiHDMI_FORCE_ACTION_E
{
    HI_HDMI_FORCE_NULL,
    HI_HDMI_FORCE_HDMI,
    HI_HDMI_FORCE_DVI,
    HI_HDMI_INIT_BOOT_CONFIG
}HI_HDMI_FORCE_ACTION_E;
```

【成员】

成员名称	描述
HI_HDMI_FORCE_NULL	标准模式
HI_HDMI_FORCE_HDMI	强制按 HDMI 方式输出
HI_HDMI_FORCE_DVI	强制按 DVI 方式输出
HI_HDMI_INIT_BOOT_CONFIG	Test Only

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_Init

HI_HDMI_ATTR_S

【说明】

定义 HDMI 属性结构体。

【定义】

```
typedef struct hiHDMI_ATTR_S
{
    HI_BOOL               bEnableHdmi;
    HI_BOOL               bEnableVideo;
    HI_HDMI_VIDEO_FMT_E   enVideoFmt;
    HI_HDMI_VIDEO_MODE_E  enVidOutMode;
    HI_HDMI_DEEP_COLOR_E  enDeepColorMode;
    HI_BOOL               bxvYCCMode;
    HI_BOOL               bEnableAudio;
    HI_HDMI_SND_INTERFACE_E enSoundIntf;
    HI_BOOL               bIsMultiChannel;
    HI_HDMI_SAMPLE_RATE_E enSampleRate;
    HI_U8                 u8DownSampleParm;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_HDMI_BIT_DEPTH_E   enBitDepth;
    HI_U8                 u8I2SCtlVbit;
    HI_BOOL               bEnableAviInfoFrame;
    HI_BOOL               bEnableAudInfoFrame;
    HI_BOOL               bEnableSpdInfoFrame;
    HI_BOOL               bEnableMpegInfoFrame;
    HI_U32                bDebugFlag;
    HI_BOOL               bHDCPEnable;
    HI_BOOL               b3DEnable;
    HI_U32                u83DParam;
} HI_HDMI_ATTR_S;
```

【成员】

成员名称	描述
bEnableHdmi	是否强制 HDMI 输出。 HI_TRUE：强制 HDMI 输出； HI_FALSE：DVI 输出。 **该值必须在 HI_HDMI_Start 之前或者 HI_HDMI_Stop 之后设置。**
bEnableVideo	是否输出视频。 **必须是 HI_TRUE。**
enVideoFmt	视频制式，此参数需要与视频输出配置的制式保持一致。
enVidOutMode	HDMI 输出视频模式。 VIDEO_MODE_YCBCR444、VIDEO_MODE_YCBCR422、VIDEO_MODE_RGB444
enDeepColorMode	DeepColor 输出模式。 默认为 HI_HDMI_DEEP_COLOR_24BIT 不支持 HI_HDMI_DEEP_COLOR_30BIT 和 HI_HDMI_DEEP_COLOR_36BIT。
bxvYCCMode	是否使能 xvYCC 输出模式。 默认为 HI_FALSE
bEnableAudio	是否使能音频。
enSoundIntf	HDMI 音频来源。 建议 HI_HDMI_SND_INTERFACE_I2S，此参数需要与 AO 保持一致。
bIsMultiChannel	多声道还是立体声。 0：立体声； 1：多声道，固定为 8 声道。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
enSampleRate	音频采样率，此参数需要与 AO 的配置保持一致。
u8DownSampleParm	音频向下 downsample 采样率的参数. 默认为 0，该参数设置无效
enBitDepth	音频位宽，默认为 16，此参数需要与 AO 的配置保持一致。
u8I2SCtlVbit	保留，请配置为 0，影响 HDMI 中 I2S control 寄存器的内容(0x7A：0x1D) 该参数设置无效。
bEnableAviInfoFrame	是否使能 AVI InfoFrame。 建议使能
bEnableAudInfoFrame	是否使能 AUDIO InfoFrame。 建议使能 DVI 模式不能使能 AUDIO InfoFrame。
bEnableSpdInfoFrame	是否使能 SPD InfoFrame。 默认关闭，该参数设置无效。
bEnableMpegInfoFrame	是否使能 MPEG InfoFrame。 默认关闭，该参数设置无效。
bDebugFlag	是否使能打开 hdmi 内部 debug 信息。 默认为 0，该参数设置无效。
bHDCPEnable	是否激活 HDCP。 0：HDCP 不激活； 1：HDCP 模式打开。 默认为 0，该参数设置无效。
b3DEnable	是否激活 3D 模式。 0：3D 不激活； 1：3D 模式打开。 默认为 0，该参数设置无效。
u83DParam	3D Parameter。 默认为 0，该参数设置无效。

【注意事项】

- 用户可以根据建议值设置 HDMI 属性。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HDMI 的以下属性的值已在内部固定：bDebugFlag、bHDCPEnable、b3Denable、u83Dparam、u8DownSampleParm、u8I2SCtlVbit、bEnableMpegInfoFrame 和 bEnableSpdInfoFrame，Hi35xx 的 HDMI 不支持 HDCP、3D。

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

HI_HDMI_VIDEO_FMT_E

【说明】

定义 HDMI 视频制式枚举。

【定义】

```
typedef enum hiHDMI_VIDEO_FMT_E
{
    HI_HDMI_VIDEO_FMT_1080P_60 = 0,
    HI_HDMI_VIDEO_FMT_1080P_50,
    HI_HDMI_VIDEO_FMT_1080P_30,
    HI_HDMI_VIDEO_FMT_1080P_25,
    HI_HDMI_VIDEO_FMT_1080P_24,

    HI_HDMI_VIDEO_FMT_1080i_60,
    HI_HDMI_VIDEO_FMT_1080i_50,

    HI_HDMI_VIDEO_FMT_720P_60,
    HI_HDMI_VIDEO_FMT_720P_50,

    HI_HDMI_VIDEO_FMT_576P_50,
    HI_HDMI_VIDEO_FMT_480P_60,

    HI_HDMI_VIDEO_FMT_PAL,
    HI_HDMI_VIDEO_FMT_PAL_N,
    HI_HDMI_VIDEO_FMT_PAL_Nc,

    HI_HDMI_VIDEO_FMT_NTSC,
    HI_HDMI_VIDEO_FMT_NTSC_J,
    HI_HDMI_VIDEO_FMT_NTSC_PAL_M,

    HI_HDMI_VIDEO_FMT_SECAM_SIN,
    HI_HDMI_VIDEO_FMT_SECAM_COS,

    HI_HDMI_VIDEO_FMT_861D_640X480_60,
    HI_HDMI_VIDEO_FMT_VESA_800X600_60,
    HI_HDMI_VIDEO_FMT_VESA_1024X768_60,
    HI_HDMI_VIDEO_FMT_VESA_1280X720_60,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_HDMI_VIDEO_FMT_VESA_1280X800_60,
    HI_HDMI_VIDEO_FMT_VESA_1280X1024_60,
    HI_HDMI_VIDEO_FMT_VESA_1366X768_60,
    HI_HDMI_VIDEO_FMT_VESA_1440X900_60,
    HI_HDMI_VIDEO_FMT_VESA_1440X900_60_RB,
    HI_HDMI_VIDEO_FMT_VESA_1600X900_60_RB,
    HI_HDMI_VIDEO_FMT_VESA_1600X1200_60,
    HI_HDMI_VIDEO_FMT_VESA_1920X1080_60,
    HI_HDMI_VIDEO_FMT_VESA_1920X1200_60,
    HI_HDMI_VIDEO_FMT_VESA_2560X1440_30,
    HI_HDMI_VIDEO_FMT_VESA_2048X1152_60,
    HI_HDMI_VIDEO_FMT_VESA_CUSTOMER_DEFINE,

    HI_HDMI_VIDEO_FMT_BUTT
}HI_HDMI_VIDEO_FMT_E;
```

【成员】

无

【注意事项】

需要根据视频输出的制式设置相应的 HDMI 的制式。

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

HI_HDMI_VIDEO_MODE_E

【说明】

定义 HDMI 颜色空间类型枚举。

【定义】

```
typedef enum hiHDMI_VIDEO_MODE
{
    HI_HDMI_VIDEO_MODE_RGB444,
    HI_HDMI_VIDEO_MODE_YCBCR422,
    HI_HDMI_VIDEO_MODE_YCBCR444,
    HI_HDMI_VIDEO_MODE_BUTT
}HI_HDMI_VIDEO_MODE_E;
```

【成员】

成员名称	描述
HI_HDMI_VIDEO_MODE_RGB444	RGB444 输出模式
HI_HDMI_VIDEO_MODE_YCBCR422	YCBCR422 输出模式

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
HI_HDMI_VIDEO_MODE_YCBCR444	YCBCR444 输出模式

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

HI_HDMI_DEEP_COLOR_E

【说明】

定义 HDMI 深色模式枚举。

【定义】

```
typedef enum hiHDMI_DEEP_COLOR_E
{
    HI_HDMI_DEEP_COLOR_24BIT = 0x00,
    HI_HDMI_DEEP_COLOR_30BIT,
    HI_HDMI_DEEP_COLOR_36BIT,
    HI_HDMI_DEEP_COLOR_OFF   = 0xff,
    HI_HDMI_DEEP_COLOR_BUTT
}HI_HDMI_DEEP_COLOR_E;
```

【成员】

成员名称	描述
HI_HDMI_DEEP_COLOR_24BIT	HDMI Deep Color 24bit 模式
HI_HDMI_DEEP_COLOR_30BIT	HDMI Deep Color 30bit 模式
HI_HDMI_DEEP_COLOR_36BIT	HDMI Deep Color 36bit 模式
HI_HDMI_DEEP_COLOR_OFF	HDMI Deep Color 关闭模式

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_HDMI_SND_INTERFACE_E

【说明】

定义 HDMI 音频输出接口类型枚举。

【定义】

```
typedef enum hiHDMI_SND_INTERFACE_E
{
    HI_HDMI_SND_INTERFACE_I2S,
    HI_HDMI_SND_INTERFACE_SPDIF,
    HI_HDMI_SND_INTERFACE_HBR,
    HI_HDMI_SND_INTERFACE_BUTT
}HI_HDMI_SND_INTERFACE_E;
```

【成员】

成员名称	描述
HI_HDMI_SND_INTERFACE_I2S	I2S 接口类型
HI_HDMI_SND_INTERFACE_SPDIF	SPDIF 接口类型
HI_HDMI_SND_INTERFACE_HBR	HBR 接口类型

【注意事项】

Hi35xx 芯片的 HDMI 的音频输出是 I^2S 接口类型。

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

HI_HDMI_SAMPLE_RATE_E

【说明】

定义 HDMI 音频输出采样率枚举。

【定义】

```
typedef enum hiHDMI_SAMPLE_RATE_E
{
    HI_HDMI_SAMPLE_RATE_UNKNOWN=0,
    HI_HDMI_SAMPLE_RATE_8K    = 8000,
    HI_HDMI_SAMPLE_RATE_11K   = 11025,
    HI_HDMI_SAMPLE_RATE_12K   = 12000,
    HI_HDMI_SAMPLE_RATE_16K   = 16000,
    HI_HDMI_SAMPLE_RATE_22K   = 22050,
    HI_HDMI_SAMPLE_RATE_24K   = 24000,
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_HDMI_SAMPLE_RATE_32K   = 32000,
    HI_HDMI_SAMPLE_RATE_44K   = 44100,
    HI_HDMI_SAMPLE_RATE_48K   = 48000,
    HI_HDMI_SAMPLE_RATE_88K   = 88200,
    HI_HDMI_SAMPLE_RATE_96K   = 96000,
    HI_HDMI_SAMPLE_RATE_176K  = 176400,
    HI_HDMI_SAMPLE_RATE_192K  = 192000,
    HI_HDMI_SAMPLE_RATE_BUTT
}HI_HDMI_SAMPLE_RATE_E;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

HI_HDMI_BIT_DEPTH_E

【说明】

定义 HDMI 音频输出采样位宽枚举。

【定义】

```
typedef enum hiHDMI_BIT_DEPTH_E
{
    HI_HDMI_BIT_DEPTH_UNKNOWN =0,
    HI_HDMI_BIT_DEPTH_8  = 8,
    HI_HDMI_BIT_DEPTH_16 = 16,
    HI_HDMI_BIT_DEPTH_18 = 18,
    HI_HDMI_BIT_DEPTH_20 = 20,
    HI_HDMI_BIT_DEPTH_24 = 24,
    HI_HDMI_BIT_DEPTH_32 = 32,
    HI_HDMI_BIT_DEPTH_BUTT
}HI_HDMI_BIT_DEPTH_E;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_SetAttr

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_HDMI_SINK_CAPABILITY_S

【说明】

定义 HDMI Sink 端能力级结构体。

【定义】

```
typedef struct hiHDMI_SINK_CAPABILITY_S
{
    HI_BOOL             bConnected;
    HI_BOOL             bSupportHdmi;
    HI_BOOL             bIsSinkPowerOn;
    HI_BOOL             bIsRealEDID;
    HI_HDMI_VIDEO_FMT_E enNativeVideoFormat;
    HI_BOOL             bVideoFmtSupported[HI_HDMI_VIDEO_FMT_BUTT];
    HI_BOOL             bSupportYCbCr;
    HI_BOOL             bSupportxvYCC601;
    HI_BOOL             bSupportxvYCC709;
    HI_U8               u8MDBit;
    HI_BOOL             bAudioFmtSupported[HI_HDMI_MAX_AUDIO_CAP_COUNT];
    HI_U32
u32AudioSampleRateSupported[HI_HDMI_MAX_AUDIO_SMPRATE_COUNT];
    HI_U32              u32MaxPcmChannels;
    HI_U8               u8Speaker;
    HI_U8               u8IDManufactureName[4];
    HI_U32              u32IDProductCode;
    HI_U32              u32IDSerialNumber;
    HI_U32              u32WeekOfManufacture;
    HI_U32              u32YearOfManufacture;
    HI_U8               u8Version;
    HI_U8               u8Revision;
    HI_U8               u8EDIDExternBlockNum;
    HI_U8               u8IEERegId[3];
    HI_BOOL             bIsPhyAddrValid;
    HI_U8               u8PhyAddr_A;
    HI_U8               u8PhyAddr_B;
    HI_U8               u8PhyAddr_C;
    HI_U8               u8PhyAddr_D;
    HI_BOOL             bSupportDVIDual;
    HI_BOOL             bSupportDeepColorYCBCR444
    HI_BOOL             bSupportDeepColor30Bit;
    HI_BOOL             bSupportDeepColor36Bit;
    HI_BOOL             bSupportDeepColor48Bit;
    HI_BOOL             bSupportAI;
    HI_U32              u8MaxTMDSClock;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_BOOL           bI_Latency_Fields_Present
    HI_BOOL           bLatency_Fields_Present;
    HI_BOOL           bHDMI_Video_Present;
    HI_U8             u8Video_Latency;
    HI_U8             u8Audio_Latency;
    HI_U8             u8Interlaced_Video_Latency;
    HI_U8             u8Interlaced_Audio_Latency;
} HI_HDMI_SINK_CAPABILITY_S;
```

【成员】

成员名称	描述
bConnected	设备是否连接
bSupportHdmi	设备是否支持 HDMI，如果不支持，则为 DVI 设备。
bIsSinkPowerOn	Sink 设备是否上电。
bIsRealEDID	EDID 是否是真正从 sink 设备获取标志。 HI_TRUE：正确读取； HI_FASE：强制设置结果。
enNativeVideoFormat	显示设备物理分辨率。
bVideoFmtSupported	视频能力集。 HI_TRUE：支持这种显示格式； HI_FALSE：不支持。
bSupportYCbCr	是否支持 YCBCR 显示。 HI_TRUE：支持 YCBCR 显示； HI_FALSE：只支持 RGB 显示。
bSupportxvYCC601	是否支持 xvYCC601 颜色格式。
bSupportxvYCC709	是否支持 xvYCC709 颜色格式。
u8MDBit	xvYCC601 支持的传输 profile:1:P0,2:P1,4:P2
bAudioFmtSupported	音频能力集，请参考 EIA-CEA-861-D 表 37 HI_TRUE：支持这种显示格式； HI_FALSE：不支持。
u32AudioSampleRateSupported	音频采样率能力集，0 为非法值，其他为支持的音频采样率。
u32MaxPcmChannels	音频最大的 PCM 通道数。
u8Speaker	扬声器位置，请参考 EIA-CEA-861-D 中 SpeakerDATABlock 的定义。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
u8IDManufactureName	设备厂商标识。
u32IDProductCode	设备 ID。
u32IDSerialNumber	设备序列号。
u32WeekOfManufacture	设备生产日期(周)。
u32YearOfManufacture	设备生产日期(年)。
u8Version	设备版本号。
u8Revision	设备子版本号。
u8EDIDExternBlockNum	EDID 扩展块数目。
u8IEERegId	24-bit IEEE Registration Identifier (0x000C03)。
bIsPhyAddrValid	CEC 物理地址是否有效标志。
u8PhyAddr_A	CEC 物理地址 A。
u8PhyAddr_B	CEC 物理地址 B。
u8PhyAddr_C	CEC 物理地址 C。
u8PhyAddr_D	CEC 物理地址 D。
bSupportDVIDual	是否支持 DVI dual-link 操作。
bSupportDeepColorYCBCR444	是否支持 YCBCR 4:4:4 Deep Color 模式。
bSupportDeepColor30Bit	是否支持 Deep Color 30bit 模式。
bSupportDeepColor36Bit	是否支持 Deep Color 36bit 模式。
bSupportDeepColor48Bit	是否支持 Deep Color 48bit 模式。
bSupportAI	是否支持 Supports_AI 模式。
u8MaxTMDSClock	最大 TMDS 时钟。
bI_Latency_Fields_Present	延时标志位。
bLatency_Fields_Present	Video_Latency 和 Audio_Latency fields 是否存在。
bHDMI_Video_Present	特殊的视频格式。
u8Video_Latency	视频延时。
u8Audio_Latency	音频延时。
u8Interlaced_Video_Latency	隔行视频模式下的视频延时。
u8Interlaced_Audio_Latency	隔行视频模式下的音频延时。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_GetSinkCapability

HI_HDMI_EDID_S

【说明】

定义 HDMI EDID 信息结构体。

【定义】

```
typedef struct hiHI_HDMI_EDID_S
{
    HI_BOOL              bEdidValid;
    HI_U32               u32Edidlength;
    HI_U8                u8Edid[512];
}HI_HDMI_EDID_S;
```

【成员】

成员名称	描述
bEdidValid	EDID 信息是否有效。
u32Edidlength	EDID 信息数据长度。
u8Edid	EDID 信息数据。

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_Force_GetEDID

HI_HDMI_CSC_S

【说明】

定义 HDMI 图像输出效果结构体。

【定义】

```
typedef struct hiHI_HDMI_CSC_S
{
    VO_CSC_MATRIX_E enCscMatrix;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_U32 u32Luma;                 /* luminance:   0 ~ 100 default: 50 */
    HI_U32 u32Contrast;             /* contrast :   0 ~ 100 default: 50 */
    HI_U32 u32Hue;                  /* hue      :   0 ~ 100 default: 50 */
    HI_U32 u32Satuature;            /* satuature:   0 ~ 100 default: 50 */
} HI_HDMI_CSC_S;
```

【成员】

成员名称	描述
enCscMatrix	CSC 矩阵选择。
u32Luma	亮度值。
u32Contrast	对比度值。
u32Hue	色调值
u32Satuature	饱和度值

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_SetCsc

HI_HDMI_INFOFRAME_TYPE_E

【说明】

定义 HDMI 信息帧类型枚举。

【定义】

```
typedef enum tagHI_HDMI_INFOFRAME_TYPE_E
{
    HI_INFOFRAME_TYPE_AVI,         /* HDMI AVI InfoFrame */
    HI_INFOFRAME_TYPE_SPD,         /* HDMI SPD InfoFrame */
    HI_INFOFRAME_TYPE_AUDIO,       /* HDMI AUDIO InfoFrame */
    HI_INFOFRAME_TYPE_MPEG,        /* HDMI MPEG InfoFrame */
    HI_INFOFRAME_TYPE_VENDORSPEC,  /* HDMI Vendor Specific InfoFrame */
    HI_INFOFRAME_TYPE_BUTT
}HI_HDMI_INFOFRAME_TYPE_E;
```

【成员】

成员名称	描述
HI_INFOFRAME_TYPE_AVI	AVI 信息帧类型。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

成员名称	描述
HI_INFOFRAME_TYPE_SPD	SPD 信息帧类型。
HI_INFOFRAME_TYPE_MPEG	MPEG 信息帧类型。
HI_INFOFRAME_TYPE_VENDORSPEC	厂商特定的信息帧类型。

【注意事项】

无

【相关数据类型及接口】

HI_MPI_HDMI_GetInfoFrame

HI_HDMI_AVI_INFOFRAME_VER2_S

【说明】

定义 AVI 信息帧结构体。

【定义】

```
typedef struct hiUNF_HDMI_AVI_INFOFRAME_VER2_S
{
    HI_HDMI_VIDEO_FMT_E       enTimingMode;
    HI_HDMI_VIDEO_MODE_E      enOutputType;
    HI_BOOL                 bActive_Infor_Present;
    HI_HDMI_BARINFO_E         enBarInfo;
    HI_HDMI_SCANINFO_E        enScanInfo;
    HI_HDMI_COLORSPACE_E      enColorimetry;
    HI_HDMI_ASPECT_RATIO_E    enAspectRatio;
    HI_HDMI_ASPECT_RATIO_E    enActiveAspectRatio;
    HI_HDMI_PICTURE_SCALING_E  enPictureScaling;
    HI_HDMI_RGB_QUAN_RAGE_E   enRGBQuantization;
    HI_BOOL                 bIsITContent;
    HI_U32                  u32PixelRepetition;

    HI_HDMI_CONTENT_TYPE_E    enContentType;
    HI_HDMI_YCC_QUAN_RAGE_E   enYCCQuantization;

    HI_U32                  u32LineNEndofTopBar;
    HI_U32                  u32LineNStartofBotBar;
    HI_U32                  u32PixelNEndofLeftBar;
    HI_U32                  u32PixelNStartofRightBar;
}HI_HDMI_AVI_INFOFRAME_VER2_S;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【成员】

成员名称	描述
enTimingMode	时序模式枚举变量。
enOutputType	输出像素格式。
bActive_Infor_Present	是否使当前信息帧生效。
enBarInfo	BAR 信息枚举变量。
enScanInfo	扫描信息枚举变量。
enColorimetry	色彩空间矩阵。
enAspectRatio	幅形比枚举变量。
enActiveAspectRatio	使能幅形比枚举变量。
enPictureScaling	图像缩放枚举。
enRGBQuantization	RGB 色彩空间范围。
bIsITContent	是否含有 IT 内容。
u32PixelRepetition	像素复制模式。
enContentType	内容类别。
enYCCQuantization	YCC 色彩空间范围。
u32LineNEndofTopBar	顶部边界的终止行。
u32LineNStartofBotBar	底部边界的起始行。
u32PixelNEndofLeftBar	左侧边界终止像素数。
u32PixelNStartofRightBar	右侧边界起始像素数。

【注意事项】

各成员变量的枚举成员详见 Hi_comm_hdmi.h 文件。

【相关数据类型及接口】

无

HI_HDMI_AUD_INFOFRAME_VER1_S

【说明】

定义 AUDIO 信息帧结构体。

【定义】

```
typedef struct hiUNF_HDMI_AUD_INFOFRAME_VER1_S
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
{
    HI_U32                      u32ChannelCount;
    HI_HDMI_CODING_TYPE_E         enCodingType;
    HI_U32                      u32SampleSize;
    HI_U32                      u32SamplingFrequency;
    HI_U32                      u32ChannelAlloc;
    HI_U32                      u32LevelShift;
    HI_BOOL                     u32DownmixInhibit;
}HI_HDMI_AUD_INFOFRAME_VER1_S;
```

【成员】

成员名称	描述
u32ChannelCount	通道个数。
enCodingType	音频数据编码类型。
u32SampleSize	采样位宽大小。一般设 0。 0：表由码流头决定。 1：表 16bit。 2：表 20bit。 3：表 24bit。
u32SamplingFrequency	采样频率。推荐设 0. 0：表由码流头决定。 1：32kHz 2：44.1 kHz 3：48 kHz 4：88.2 kHz 5：96 kHz 6：176.4 kHz 7：192 kHz
u32ChannelAlloc	通道分配，设 0 即可。
u32LevelShift	设 0。
u32DownmixInhibit	设 0。

【注意事项】

无

【相关数据类型及接口】

无

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

HI_HDMI_SPD_INFOFRAME_S

【说明】

定义 SPD 信息帧结构体。

【定义】

```
typedef struct hiUNF_HDMI_SPD_INFOFRAME_S
{
    HI_U8                       u8VendorName[8];
    HI_U8                       u8ProductDescription[16];
}HI_HDMI_SPD_INFOFRAME_S;
```

【成员】

成员名称	描述
u8VendorName[8]	卖方名称。
u8ProductDescription[16]	产品描述符。

【注意事项】

无

【相关数据类型及接口】

- HI_MPI_HDMI_SetInfoFrame
- HI_MPI_HDMI_GetInfoFrame

HI_HDMI_MPEGSOURCE_INFOFRAME_S

【说明】

定义 MPEGSource 信息帧结构体。

【定义】

```
typedef struct hiUNF_HDMI_MPEGSOURCE_INFOFRAME_S
{
    HI_U32                      u32MPEGBitRate;
    HI_BOOL                     bIsFieldRepeated;
}HI_HDMI_MPEGSOURCE_INFOFRAME_S;
```

【成员】

成员名称	描述
u32MPEGBitRate	MPEG 位率。
bIsFieldRepeated	FieldRepeater 标志位。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【注意事项】

无

【相关数据类型及接口】

- HI_MPI_HDMI_SetInfoFrame
- HI_MPI_HDMI_GetInfoFrame

HI_HDMI_VENDORSPEC_INFOFRAME_S

【说明】

定义 VS 信息帧结构体。

【定义】

```
typedef struct hiUNF_HDMI_VENDORSPEC_INFOFRAME_S
{
    HI_U32                      u32RegistrationId;
}HI_HDMI_VENDORSPEC_INFOFRAME_S;
```

【成员】

成员名称	描述
u32RegistrationId	注册 ID 号。

【注意事项】

无

【相关数据类型及接口】

- HI_MPI_HDMI_SetInfoFrame
- HI_MPI_HDMI_GetInfoFrame

HI_HMDI_INFORFRAME_UNIT_U

【说明】

定义 HDMI 信息帧联合体。

【定义】

```
typedef union hiUNF_HDMI_INFOFRAME_UNIT_U
{
    HI_HDMI_AVI_INFOFRAME_VER2_S   stAVIInfoFrame;
    HI_HDMI_AUD_INFOFRAME_VER1_S   stAUDInfoFrame;
    HI_HDMI_SPD_INFOFRAME_S        stSPDInfoFrame;
    HI_HDMI_MPEGSOURCE_INFOFRAME_S stMPEGSourceInfoFrame;
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    HI_HDMI_VENDORSPEC_INFOFRAME_S stVendorSpecInfoFrame;
}HI_HMDI_INFORFRAME_UNIT_U;
```

【成员】

成员名称	描述
stAVIInfoFrame	AVI 信息帧结构体变量。
stAUDInfoFrame	AUDIO 信息帧结构体变量。
stSPDInfoFrame	SPD 信息帧结构体变量。
stMPEGSourceInfoFrame	MPEG 信息帧结构体变量。
stVendorSpecInfoFrame	厂商特定信息帧结构体变量。

【注意事项】

无

【相关数据类型及接口】

无

HI_HDMI_INFOFRAME_S

【说明】

定义 HDMI 信息帧结构体。

【定义】

```
typedef struct hiUNF_HDMI_INFOFRAME_S
{
    HI_HDMI_INFOFRAME_TYPE_E    enInfoFrameType;  /* InfoFrame type */
    HI_HMDI_INFORFRAME_UNIT_U   unInforUnit;      /* InfoFrame date */
}HI_HDMI_INFOFRAME_S;
```

【成员】

成员名称	描述
enInfoFrameType	信息帧类型枚举变量。
unInforUnit	信息帧联合体变量。

【注意事项】

无

【相关数据类型及接口】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

- HI_MPI_HDMI_SetInfoFrame
- HI_MPI_HDMI_GetInfoFrame

12.5 错误码

HDMI API 错误码如表 12-1 所示。

表12-1 HDMI API 错误码

错误代码	宏定义	描述
0xA0288001	HI_ERR_HDMI_NOT_INIT	HDMI 未初始化
0xA0288002	HI_ERR_HDMI_INVALID_PARA	参数非法
0xA0288003	HI_ERR_HDMI_NUL_PTR	空指针
0xA0288004	HI_ERR_HDMI_DEV_NOT_OPEN	HDMI 未打开
0xA0288005	HI_ERR_HDMI_DEV_NOT_CONNECT	HDMI 设备未连接
0xA0288006	HI_ERR_HDMI_READ_SINK_FAILED	HDMI 读取 Sink 端失败
0xA0288007	HI_ERR_HDMI_INIT_ALREADY	HDMI 已经初始化
0xA0288008	HI_ERR_HDMI_CALLBACK_ALREADY	HDMI 回调已注册
0xA0288009	HI_ERR_HDMI_INVALID_CALLBACK	回调函数无效
0xA028800A	HI_ERR_HDMI_FEATURE_NO_USPPORT	功能不支持
0xA028800B	HI_ERR_HDMI_BUS_BUSY	总线忙

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

目 录

目 录 i
13 Proc 调试信息说明 13-1
13.1 概述 13-1
13.2 SYS 13-2
13.3 VB 13-4
13.4 LOG 13-6
13.5 CHNL 13-8
13.6 DSU 13-10
13.7 GROUP 13-14
13.8 H264E 13-18
13.9 JPEGE 13-24
13.10 RC 13-26
13.11 REGION 13-31
13.12 VENC 13-36
13.13 VDEC 13-37
13.14 VI 13-42
13.15 VO 13-50
13.16 VPSS 13-61
13.17 IVE 13-70
13.18 VDA 13-70
13.19 AI 13-74
13.20 AO 13-81
13.21 AENC 13-87
13.22 ADEC 13-88
13.23 MPEG4E 13-88
13.24 HDMI 13-91
13.25 ACODEC 13-95

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

13 Proc 调试信息说明

13.1 概述

调试信息采用了 Linux 下的 proc 文件系统，可实时反映当前系统的运行状态，所记录的信息可供问题定位及分析时使用。

【文件目录】

/proc/umap

【文件清单】

文件名称	描述
sys	记录当前 SYS 模块的使用情况。
vb	记录当前 VB 模块的 buffer 使用情况。
logmpp	记录当前各个模块的调试级别。内部调试用。
chnl	CHNL 模块状态。
dsu	视频缩放处理单元状态信息。
grp	当前编码通道组的属性配置以及当前编码通道统计状态。
h264e	H.264 编码过程中，各通道的编码属性、状态以及历史信息统计。
jpege	JPEG 编码过程中，各通道的编码属性、状态以及历史信息统计。
rc	编码通道的码流控制属性、状态以及历史信息统计。
rgn	视频叠加 OSD 的区域管理信息。
venc	视频编码器信息。
vdec	视频解码器信息。
vi	视频输入模块信息。
vo	视频输出模块信息。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

文件名称	描述
vpss	视频预处理模块信息。
ive	视频智能处理算子状态信息。
vda	当前开启 VDA 的通道的使用状况及其属性配置。
ai	音频输入通道信息。
ao	音频输出通道信息。
aenc	音频编码信息。
adec	音频解码信息。
mpeg4e	mpeg4 编码过程中，各通道的编码属性、状态以及历史信息统计。
hdmi	HDMI 信息。

【信息查看方法】

- 在控制台上可以使用 cat 命令查看信息，例如 cat /proc/umap/venc；也可以使用其他常用的文件操作命令，例如 cp /proc/umap/ ./ -rf，将所有 umap 下的 proc 文件拷贝到当前目录。
- 在应用程序中可以将上述文件当作普通只读文件进行读操作，例如 fopen、fread 等。

说明

参数在描述时有以下 2 种情况需要注意：

- 取值为{0, 1}的参数，如未列出具体取值和含义的对应关系，则参数为 1 时表示肯定，为 0 时表示否定。
- 取值为{aaa, bbb, ccc}的参数，未列出具体取值和含义的对应关系，但可直接根据取值 aaa、bbb 或 ccc 判断参数含义

13.2 SYS

【调试信息】

```
# cat /proc/umap/sys
[SYS] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Dec 20 2011, 10:40:10]

System State: 0 (0: initialized; 1: exiting; 2: exited)

-----MEM TABLE-------------------------------------------------------------
MOD          MODNAME DEV CHN          MMZNAME
5            grp   1   0          ddr1
    5            grp   3   0          ddr1
    5            grp   5   0          ddr1
    5            grp   7   0          ddr1
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
-----BIND RELATION TABLE-------------------------------------------------
FirMod FirDev FirChn  SecMod  SecDev  SecChn  TirMod  TirDev  TirChn SendCnt  rstCnt
vpss       0      2       vo       0     0     null       0      0      0     0
vi       0      0     vpss       0       2    vo     0       0     448     0
```

【调试信息分析】

记录当前 SYS 模块的使用情况。

【参数说明】

参数		描述
System State	initialized	初始化状态。
	exiting	正在退出状态。
	exited	退出状态。
MEMTABLE	MOD	模块 ID 号
	MODNAME	模块名
	DEV	模块的设备号
	CHN	模块的通道号
	MMZNAME	MMZ 名字（其值为 DDR 名，如果 DDR 命名为 NULL，此表将不显示该信息。）
BIND RELATION TABLE	FirMod	绑定关系中第一级的模块名，数据由第一级发送给第二级
	FirDev	绑定关系中第一级的设备号，数据由第一级发送给第二级
	FirChn	绑定关系中第一级的通道号，数据由第一级发送给第二级
	SecMod	绑定关系中第二级的模块名，数据由第一级发送给第二级
	SecDev	绑定关系中第二级的设备号，数据由第一级发送给第二级
	SecChn	绑定关系中第二级的通道号，数据由第一级发送给第二级
	TirMod	绑定关系中第三级的模块名，如果有第三级绑定关系，则数据由第二级发送给第三级，否则显示为 null
	TirDev	绑定关系中第三级的设备号。
	TirChn	绑定关系中第三级的通道号。
	SendCnt	第一级向第二级的发送数据计数（一般以帧为单位。VPSS 到 VO 的发送计数始终为 0，因其不在 SYS 中计数，需到各个模块 proc 信息里查看计数）。
	rstCnt	第一级 Reset 第二级的次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

13.3 VB

【调试信息】

```
# cat /proc/umap/vb
[VB] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Nov 22 2012, 16:56:02]
-----VB PUB CONFIG-------------------------------------------------------
Max Count of Pools:       256
-----COMMON POOL CONFIG-------------------------------------------------
PoolId      0      1      2      3      4      5      6      7      8      9  …  15
Size    884736 1107968 4149248   3616      0      0      0      0      0      0 … 0
Count       0    128      32      48      0      0      0      0      0      0 …  0
-----     ---------------------------------------------------------------
PoolId    PhysAddr    VirtAddr    IsComm    BlkSz    BlkCnt     Free      MinFree
0  0x85de0000  0xcf000000        1   1107968      128      124(124)      123
BLK   VIU   VOU   DSU  VENC  VDEC VDA H264E JPEGE MPEGE H264D JPEGD MPEGD  VPSS …
85     0     1     0     0     0     0     0    0     0     0     0   0     0  0   …
86     0     1     0     0     0     0     0   0     0     0     0   0     0  0   …
87     1     0     0     0     0     0     0     0     0     0     0     0     0     0   …
88     1     0     0     0   0     0     0     0     0     0     0     0    0  0
Sum      2     2     0     0     0     0     0     0     0     0     0     0    0   0
-----     ---------------------------------------------------------------
PoolId    PhysAddr    VirtAddr    IsComm    BlkSz    BlkCnt     Free      MinFree
1  0x8e520000  0xd8000000        1   4149248       32       32(32)       32
-----     ---------------------------------------------------------------
PoolId    PhysAddr    VirtAddr    IsComm    BlkSz    BlkCnt     Free      MinFree
2  0x963c0000  0xc49c0000        1     3616       48       48(48)       48
-----  ddr1-------------------------------------------------------------
PoolId    PhysAddr    VirtAddr    IsComm    BlkSz    BlkCnt     Free      MinFree
3  0xc0000000  0xe0000000        0   4147200        6       2(2)        2
BLK   VIU   VOU   DSU  VENC  VDEC   VDA H264E JPEGE MPEGE H264D JPEGD MPEGD  VPSS …
4   0     2     0     0     0     0     0     0     0   0     0     0     0    0  0
5     0     2     0     0     0     0   0     0     0     0     0     0   0   0  0 …
0     0     2     0     0     0     0     0     0     0     0   0     0   0   0   0
1    0     2     0     0     0     0     0     0     0     0     0     0    0   0  0
Sum      0     4     0     0     0     0     0     0     0   0   0   0   0   0   0
```

【调试信息分析】

记录当前 VB 模块的 buffer 使用情况。

【参数说明】

参数		描述
VB PUB CONFIG	Max Count of Pools	最大的缓存池的个数。
COMMON POOL CONFIG	PoolId	公共缓存池的句柄。
	Size	缓存池内块的大小。
	Count	缓存池内块的个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
NULL(表空，即无名的DDR)	PoolId	公共/私有缓存池的句柄。
	PhysAddr	公共/私有缓存池的开始物理地址。
	VirtAddr	公共/私有缓存池的开始虚拟地址。
	IsComm	是否公共缓存池。 取值：{0, 1}。
	BlkSz	缓存池内缓存块的大小。
	BlkCnt	缓存池内缓存块的个数。
	Free	缓存池空闲缓存块的个数。
	MinFree	程序运行以来，空闲缓存块的最小剩余个数。若该计数为 0，则说明可能存在因缓存块不够而丢帧的情况。
	BLK	缓存池内缓存块的句柄。
	VIU/VOU/DSU/VENC/VDEC/VDA/H264E/JPEGE/MPEGE/H264D/JPEGD/MPEGD/VPSS/GRP/MPI/PCIV/AI/AENC/RC	模块名。 下面对应的数字表示当前模块有多少个地方占用缓存池内的该缓存块。 0：没占用。 非 0：占用次数。
ddr1(即名字为 ddr1 的内存条)	PoolId	公共/私有缓存池的句柄。
	PhysAddr	公共/私有缓存池的开始物理地址。
	VirtAddr	公共/私有缓存池的开始逻辑地址。
	IsComm	是否公共缓存池。 取值：{0, 1}。
	BlkSz	公共/私有缓存池内缓存块的大小。
	BlkCnt	公共/私有缓存池内缓存块的个数。
	Free	公共/私有缓存池空闲缓存块的个数。
	MinFree	程序运行以来，空闲缓存块的最小剩余个数。若该计数为 0，则说明可能存在因缓存块不够而丢帧的情况。
	BLK	公共/私有缓存池内缓存块的句柄。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	VIU/VOU/DSU/VENC/VDEC/VDA/H264E/JPEGE/MPEGE/H264D/JPEGD/MPEGD/VPSS/GRP/MPI/PCIV/AI/AENC/RC	模块名。 下面对应的数字表示当前模块有多少个地方占用了缓存池内的该缓存块。 0：没占用。 非 0：占用次数。

13.4 LOG

【调试信息】

```
# cat /proc/umap/logmpp

-----LOG BUFFER STATE----------------------------------------------------------
MaxLen  ReadPos WritePos ButtPos
 64(KB)      0    1378   65536

-----CURRENT LOG LEVEL---------------------------------------------------------
cmpi    :  3
vb      :  3
sys     :  3
chnl    :  3
grp     :  3
venc    :  3
vpss    :  3
vda     :  3
h264e   :  3
jpege   :  3
mpeg4e  :  3
vdec    :  3
h264d   :  3
jpegd   :  3
vo      :  3
vi      :  3
dsu     :  3
rgn     :  3
rc      :  3
sio     :  3
ai      :  3
ao      :  3
aenc    :  3
adec    :  3
pciv    :  3
pcivfmw :  3
ive     :  3
vcmp    :  3
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
fb      :  3
hdmi    :  3
voie    :  3
```

【调试信息分析】

记录当前各个模块的调试级别。

cat /proc/umap/logmpp 用于获取 Hi35xx 的 log 级别信息。

【参数说明】

参数		描述
LOG BUFFER STATE	MaxLen	LOG 缓存大小（KB）
	ReadPos	读指针位置。
	WritePos	写指针位置。
	ButtPos	缓存底位置。
CURRENT LOG LEVEL	cmpi	cmpi 模块名。
	vb	vb 模块名。
	sys	sys 模块名。
	chnl	chnl 模块名。
	grp	group 模块名。
	venc	venc 模块名。
	vpss	vpss 模块名。
	vda	vda 模块名。
	h264e	h264e 模块名。
	jpege	JPEGe 模块名。
	mpeg4e	mpeg4e 模块名。
	vdec	vdec 模块名。
	h264d	h264d 模块名。
	jpegd	JPEGd 模块名。
	vo	vo 模块名。
	vi	vi 模块名。
	dsu	dsu 模块名。
	rgn	rgn 模块名
	rc	rc 模块名

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	sio	sio 模块名。
	ai	ai 模块名。
	ao	ao 模块名。
	aenc	aenc 模块名
	adec	adec 模块名
	pciv	pciv 模块名
	pcivfmw	pcivfmw 模块名
	ive	ive 模块名
	vcmp	vcmp 模块名
	fb	fb 模块名
	hdmi	hdmi 模块名
	voie	voie 模块名

13.5 CHNL

【调试信息】

```
# cat /proc/umap/chnl

[CHNL] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Sep  2 2011, 15:36:02]

Total Chnl Scheduler Count: 1

-----SCHEDULER 0 INFO--------------------------------------------------
SchdlId VpuNum
  0       1
-----VPU INFO----------------------------------------------------------------
VpuId   Name    State   IntCnt  TimeCnt   VpuCnt  ErrCnt  InqCnt  StartOk StartNo Config
Reset
0       VEUD_0  RUN     5252    312    4940      0    7735    4941     0
233       0
-----CHNL STATE-------------------------------------------------------------
ChnlId  Type    TskNum  State   InqCnt  StartOk StartNo  IntPro
0       H264E     6     IDLE    304     155      0        4940
-----CHNL PERF--------------------------------------------------------------
ChnlId  Type    StartCost IntCost IntCostL  HWCost  HWCycle HWCostL HWCostAcc
0        H264E   2829      4354       6352    22234     0     25635     3
-----CHNL CURRENT RUN STATE-------------------------------------------------
VpuId   VpuName ChnlId  Type
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
0      VEUD_0    27    H264E
```

【调试信息分析】

记录当前通道调度器模块的状态调试信息。

【参数说明】

参数		描述
SCHEDULER 0 INFO 调度器信息	SchdlId	调度器编号。
	VpuNum	调度器管理的 vpu 的个数。
VPU INFO 视频处理单元信息	VpuId	视频处理单元编号。
	Name	视频处理单元名称。
	State	视频处理单元的运行状态。
	IntCnt	视频处理单元的中断次数，包括了软中断次数和硬件中断次数。
	TmCnt	视频处理单元软中断次数。
	VpuCnt	视频处理单元硬中断次数。
	ErrCnt	视频处理单元上报中断信息错误次数。
	InqCnt	视频处理单元查询任务片次数。
	StartOk	视频处理单元启动成功次数。
	StartNo	视频处理单元启动失败次数。
	Config	视频处理单元配置成功的次数。
	Reset	视频处理单元死机次数。
CHNL STATE 通道状态	ChnlId	注册到当前调度器的通道编号。
	Type	通道类型。
	TskNum	当前通道的任务片个数。
	State	当前任务片的状态。
	InqCnt	当前任务片被查询的次数。
	StartOk	当前任务片启动的次数。
	StartNo	当前任务片启动失败的次数。
	IntPro	当前任务片中断次数。
CHNL PERF 通道性能	ChnlId	注册到当前调度器的通道编号。。
	Type	通道类型。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	StartCost	启动当前任务片消耗时间，单位微秒。
	IntCost	当前任务片中断处理消耗时间，单位微秒。
	IntCostL	当前任务片中断处理消耗的最长时间，单位微秒。
	HWCost	当前任务片消耗的芯片处理时间，单位微秒。
	HWCycle	当前任务片消耗的 cycle 数。
	HWCostL	一个任务片消耗的最长的芯片处理时间，单位微秒。
	HWCostAcc	消耗的芯片处理时间超过 30 毫秒的任务片个数。
CHNL CURRENT RUN STATE 通道当前运行状态	VpuId	视频处理单元编号。
	VpuName	视频处理单元名称。
	ChnlId	注册到当前调度器的通道编号。
	Type	通道类型。

13.6 DSU

【调试信息】

```
# cat /proc/umap/dsu

[DSU] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Sep 15 2011, 19:41:34]

-----MODULE PARAM------------------------------------------------------ max_job_num
        40
-----RECENT JOB INFO--------------------------------------------------
   SeqNo   ModNameJobHdl TaskNum   State  InSize    OutSize CostTime
       0     VOU      13       4      ProcOk 1658880  414720   10533
       1     VOU      12       4      ProcOk 1658880  414720   10599
       2     VOU      11       4      ProcOk 1658880  414720   10583
       3     VOU      10       4      ProcOk 1658880  414720   10542
       4     VOU       9       4      ProcOk 1658880  414720   10573
       5     VOU       8       4      ProcOk 1658880  414720   10538
       6     VOU       7       4      ProcOk 1658880  414720   10605
       7     VOU       6       4      ProcOk 1658880  414720   10564

-----MAX WASTE TIME JOB INFO-------------------------------------------
   SeqNo   ModNameJobHdl TaskNum   State  InSize OutSize CostTime
       8     VOU      23       4   ProcOk    1658880  414720   11005
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
-----DSU JOB STATUS-------------------------------------------------------
   Success      Fail    Cancel  AllJobNd    FreeNd   BeginNd  EndJobNd    ProcNd
      1294         0         0        40        39         0         0         1

-----DSU TASK STATUS-------------------------------------------------------
   Success      Fail    Cancel AllTaskNd    FreeNd    BusyNd
     10349         0         0       320       312         8

-----DSU INT STATUS-------------------------------------------------------
    IntNum     IntTm      FifoDepth   HalProcTm
      1294        52              1         314

-----DSU MEM REQ STATUS ---------------------------------------------------
     ReqOk    FreeOk   ReqFail  FreeFail
        0         0         0         0
-----DSU TASKLET STATUS ---------------------------------------------------    TslNum
MaxTslNum     TslTm  MaxTslTm
       13        13     23265     24034

-----DSU JOB STATISTIC FOR EACH MOD-----------------------------------------    ModId
ModName    JobNum    MaxJob     psNum   MaxPNum
       15        vo        13        13         0         0
```

【调试信息分析】

记录 DSU 模块最近完成的若干任务、最近耗时最大的任务、历史累计信息及中断相关信息。

【参数说明】

参数		描述
MODULE PARAM	max_job_num	最大 job 数。默认为 40。 仅当需要调整 DSU 支持的 job 数，才需要在插入模块时指定该值。
RECENT JOB INFO 最近完成的 job 的信息	SeqNo	打印序号。 取值范围：[0, 7]
	ModName	提交该 job 的模块名。
	JobHdl	该 job 的 handle 号，调试使用。 取值范围：[0, 39]
	TaskNum	该 job 包含的 task 数目。 取值范围：[0, 320）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	State	该 job 的处理状态。 取值范围：{ProcNok、Procok、Procing} ProcNok：表示 job 执行失败 Procok ：表示 job 执行成功 Procing ：表示 job 正在硬件处理
	InSize	该 job 下各 task 的输入图像面积之和 单位：像素 每向该 job 添加一个 task，此项就加上该 task 的输入面积。 DSU 缩放性能主要受图像输入、输出面积的影响。一般，Job 的总输入/输出图像面积（InSize/OutSize）越大，处理该任务的耗时（CostTime）就越长。
	OutSize	该 job 下各 task 的输出图像面积之和 单位：像素 每向该 job 添加一个 task，此项就加上该 task 的输出面积。
	CostTime	该 job 从提交到完成的耗时时长。 单位：us。 该时间包括针对该任务的软件、硬件及中断服务程序处理时间。 如果 DSU 性能不足（如提交缩放的任务太多，超出性能限制），可能该值会超过预期。如 PAL 制预览业务，一个 Job 的完成时间超过 40ms，则会导致预览图像卡顿。此情况下，可检查业务是否超过 DSU 性能。
MAX WASTE TIME JOB INFO 最近耗时最大的 job 信息	各项同 RECENT JOB INFO 的成员	最近 500 个任务中耗时最长的 job 的信息。 其各项同 RECENT JOB INFO 的成员，具体意义请参见前述。 当出现耗时更长的任务或任务总数已超过 500 时，就更新该组值。 通过该组值可知最近的 dsu 运行性能，以及是否出现过 DSU 处理不及时的情况。
DSU JOB STATUS DSU 任务状态	Success	累计成功处理的 job 数。 当硬件处理成功时加 1。
	Fail	累计处理失败的 job 数。 当 DSU 提交任务给驱动层并失败时加 1。该值增加时可通过察看日志了解失败原因。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Cancel	累计的用户主动取消的 job 数。 当用户调用 cancleJob 接口时加 1，如向 job 添加 task 时失败，用户会 cancleJob。该值增加时可通过察看日志了解失败原因。
	AllJobNd	DSU 任务节点总数，同 max_job_num 值，一般为 40。
	FreeNd	空闲的 job 节点数。
	BeginNd	用户已创建任务但还未提交（EndJob 接口）的 job 数。
	EndJobNd	用户已提交（EndJob 接口）但还未提交给硬件处理的任务数。
	ProcNd	正在进行硬件处理的任务数。
DSU TASK STATUS DSU Task 状态	Success	累计成功处理的 Task 数。 一个 job 包含 1 到多个缩放 task，所以 1 个 job 成功表明其下的多个 task 都成功，故该值累加的比 job 的 Success 项更快。 当硬件处理一个 Job 成功时，该值累加 job 下的 task 数。
	Fail	累计处理失败的 Task 数。 一个 job 失败将导致其下的所有 task 失败。当 Job 执行失败时，该项累加其下失败的 task 数。 该值增加时可通过察看日志了解失败原因。
	Cancel	累计的用户主动 Cancel 的 Task 数。 用户调用 cancleJob 接口时，即取消了 job 下的所有 task 的执行，该项累加。 该值增加时可通过察看日志了解失败原因。
	AllJobNd	DSU Task 节点总数，一般为 320。
	FreeNd	空闲的 Task 节点数。
	BusyNd	已添加到 Job 下的 task 数。
DSU INT STATUS DSU 中断状态	IntNum	DSU 中断个数。 硬件完成一个 job，该值加 1。
	IntTm	DSU 中断处理时间。 单位：us 包含 DSU 中断处理和回调用户的时间。
	FifoDepth	硬件处理深度，内部调试使用。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	HalProcTm	将 Job 提交给驱动层的时间，内部调试使用。 单位：us
DSU MEM REQ STATUS DSU 内存申请状态	ReqOk	申请 VB 成功次数。 当申请成功时加 1。 注意，用户使用 DSU 的 OSD 叠加功能或将视频转换到 PACKAGE 格式时，DSU 才申请/释放内存。
	FreeOk	释放 VB 成功次数。 当释放成功时加 1。一般应与 ReqOk 项相同，表明没有内存泄露。
	ReqFail	申请 VB 失败次数。 当申请失败时加 1，一般应为 0。
	FreeFail	释放 VB 失败次数。 当释放失败时加 1，一般应为 0。
DSU TASKLET STATUS DSU 内部 tasklet 的状态	TslNum	最近一秒 tasklet 执行的次数。
	MaxTslNum	历史上一秒钟内 tasklet 执行的最大次数。
	TslTm	最近一秒 tasklet 执行占用 CPU 的时间，单位为 us。
	MaxTslTm	历史上一秒钟内 tasklet 执行占用 CPU 的最大时间，单位为 us。
DSU JOB STATISTIC FOR EACH MOD 各个模块调用 DSU 的统计信息（没有调用的模块则不显示）	ModId	模块 ID。
	ModName	模块名。
	JobNum	最近一秒该模块调用 DSU JOB（含非 bypass 与 bypass JOB）的次数。
	MaxJob	历史上一秒钟内该模块调用 DSU JOB 的最大次数。
	psNum	最近一秒该模块调用 DSU bypass JOB 的次数。
	MaxPNum	历史上一秒钟内该模块调用 DSU bypass JOB 的次数。

13.7 GROUP

【调试信息】

```
cat /proc/umap/grp
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
[GRP] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Apr  9 2012, 15:52:20]

-----GROUP COLOR2GREY CONF----------------------------------------------Enabled
MaxWidth MaxHeight
0          0        0
-----GROUP CHN ATTR------------------------------------------------------ NO.   ChnId
EnType   Width  Height   C2GEn VeStr  OsdStr SrcFr  TarFr   Timeref
 0      0      96        704     576       0     YES     YES      -1      -1    20065876
PixFmt   PicAddr
YUV420  0x8773f000
-----GROUP CHN RECEIVE STAT---------------------------------------------- NO.      Start
StartEx    RecvLeft     EncLeft
 0          1          0          0          0
-----GROUP VPSS QUERY--------------------------------------------------- NO.   Query
QueryOk    QueryFR   Invld    Full  VbFail    Fail InfoErr    Stop
0     60074037 60074037   0        50041107    0    31600     0       0          0

-----GROUP SEND1-------------------------------------------------------------
NO.  VpssSnd VInfErr OthrSnd OInfErr     Send     Stop     Full CropErr DrectSnd  SizeErr
0    10001330  1          0       0      10001329   0         0     0      10001329    0

-----GROUP SEND2-------------------------------------------------------------
NO.  SendDsu  StartOk StartFail IntOk  IntFail SrcAdd  SrcSub   HistAdd   HistSub
 0       0          0        0       0       0        0       0        0          0
DestAdd   DestSub
 0          0

-----GROUP PIC QUEUE STATE---------------------------------------------
NO.    Free    busy     dsu
 0       5        1        0

-----GROUP CHNL INFO---------------------------------------------------------
NO.     Inq         InqOk       Start        StartOk    Config     GrpInt
0    69949490    10001328    10001328     9455744     9455744     9455744

-----GROUP CROP INFO-------------------------------------------------------------
 NO.  CropEn  StartX  StartY   Width  Height
   0      NO       0       0       0       0
```

【调试信息分析】

记录当前编码通道组属性配置以及状态信息。

【参数说明】

参数		描述
GROUP COLOR2	Enabled	所有通道组是否支持把彩色图像转换成灰度图像进行编码的全局开关。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
GREY CONF	MaxWidth	支持把彩色图像转换成灰度图像进行编码的 group 的最大宽度，即一个 group 如果支持把彩色图像转换成灰度图像进行编码（C2Gen 为 1），它的宽度（Width）必须小于或等于 MaxWidth。
	MaxHeight	支持把彩色图像转换成灰度图像进行编码的 group 的最大高度，即一个 group 如果支持把彩色图像转换成灰度图像进行编码（C2Gen 为 1），它的高度（Height）必须小于或等于 MaxHeight。
GROUP CHN ATTR	NO.	GROUP 通道号。
	ChnId	注册到 GROUP 内的编码通道的通道号。
	EnType	GROUP 内编码通道的类型，如 h264，jpege 等。
	Width	GROUP 通道内编码通道宽度。
	Height	GROUP 通道内编码通道高度。
	C2GEn	GROUP 是否支持把彩色图像转换成灰度图像进行编码。
	VeStart	是否启动编码。
	OsdStart	是否启动 Osd。
	SrcFr	GROUP 帧率控制的源帧率，即输入帧率。
	TarFr	GROUP 帧率控制的目标帧率
	Timeref	图像 busy 队列最新帧的 timeref。
	PixFmt	编码当前帧的格式。
	PicAdr	编码当前帧的地址。
GROUP CHN RECEIVE STAT	NO.	GROUP 通道号。
	Start	通道组是否开始接收图像。
	StartEx	通道组是否开始按帧数接收图像。
	RecvLeft	GROUP 剩余未接收图像的帧数；配合 HI_MPI_VENC_StartRecvPicEx（）来使用的。
	EncLeft	GROUP 剩余未完成编码的帧数；配合 HI_MPI_VENC_StartRecvPicEx（）来使用的。
GROUP VPSS QUERY	NO.	GROUP 通道号。
	Query	GROUP 被 VPSS 查询的总的次数。
	QueryOk	GROUP 被 VPSS 查询成功的次数。
	QueryFR	GROUP 被 VPSS 查询，帧率控制丢帧的次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Invld	GROUP 被 VPSS 查询，源图像为空图像的次数。
	Full	GROUP 被 VPSS 查询，图像队列满的次数。
	VbFail	GROUP 被 VPSS 查询，申请 VB 失败的次数。
	Fail	GROUP 被 VPSS 查询失败的次数。
	InfoErr	GROUP 被 VPSS 查询，VPSS 信息错误导致查询失败的次数。
	Stop	GROUP 被 VPSS 查询，GROUP 停止接受图像导致查询失败的次数。
GROUP SEND1	NO.	GROUP 通道号。
	VpssSnd	VPSS 发送图像次数。
	VInfErr	VPSS 发送图像，图像信息错误导致发送失败的次数。
	OthrSnd	其他模块发送图像次数。
	OInfErr	其他模块发送图像，图像信息错误导致发送失败的次数。
	Send	外部模块发送正确图像次数。
	Stop	外部模块发送图像，GROUP 停止接受图像次数。
	Full	外部模块发送图像，图像队列满的次数。
	CropErr	裁剪图像失败的次数。
	DrectSnd	外部模块发动图像，直接送入图像 busy 队列的次数（不过 DSU）。
	SizeErr	GROUP 拒绝接受图像次数。当前端发送的图像的宽度（或高度）大于 GROUP 的宽度（或高度）时，此项会递增。
GROUP SEND2	NO	GROUP 通道号
	SendDsu	图像发送给 DSU 次数。
	StrtOk	DSU 启动成功的次数。
	StartFail	DSU 启动失败的次数。
	IntOk	DSU 中断成功的次数。
	IntFail	DSU 中断失败的次数。
	SrcAdd	源图像的 VB 引用计数增加次数。
	SrcSub	源图像的 VB 引用计数减少次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	HistAdd	直方图的 VB 引用计数增加次数。
	HistSub	直方图的 VB 引用计数减少次数。
	DestAdd	目标图像的 VB 引用计数增加次数。
	DestSub	目标图像的 VB 引用计数减少次数。
GROUP PIC QUEUE STATE	NO	GROUP 通道号。
	Free	GROUP free 队列节点数（可接受图像帧数）。
	Busy	GROUP Busy 队列节点数（接受被占用的图像帧数）。
	Dsu	DSU 正在处理的图像帧数。
GROUP CHNL INFO	NO.	GROUP 通道组号。
	Inq	Chnl 查询的次数。
	InqOk	Chnl 查询成功的次数。
	Start	Chnl 启动编码次数。
	StartOk	Chnl 启动编码成功次数。
	Config	Chnl 配置编码硬件的次数。
	GrpInt	Chnl 接收中断的次数。
GROUP CROP INFO	NO.	GROUP 通道组号。
	CropEn	是否开启编码的 GROUP 裁剪
	StartX	裁剪区域起始横坐标
	StartY	裁剪区域起始纵坐标
	Width	裁剪区域宽
	Height	裁剪区域高

13.8 H264E

【调试信息】

```
[H264E] Version: [Hi3531_MPP_V1.0.0.0 ], Build Time[Dec  3 2014, 11:34:44]
-----MODULE PARAM-----------------------------------------------------------
   ContiBuffer  H264eVBSource    H264eRcnEqualRef     H264eMiniBufMode
            0            1            0                     0
-----CHN ATTR--------------------------------------------------------------
    ID  MaxWidth MaxHeight  PicWidth PicHeight   profile   MainStr   RefMode   BufSize
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
ByFrame   MaxStrCnt  PoolId
     0     1920     1080      1920      1080        hp       Yes       1X     4147200
No          200       4

-----PICTURE INFO-----------------------------------------------------------------
ID     EncdStart   EncdSucceed    Lost    Disc    Skip  BufLeak   PSkip  Recode      RlsStr
UnrdStr   GetVBFail
0          161          161         0       0       0       0       0      0         152
9          0

-----STREAM BUFFER----------------------------------------------------------------
    ID     Base       RdTail     RdHead     WrTail     WrHead     DataLen     BufFree
    0      0xe7400000  0xbb500     0xea800     0xea800     0xea800     0          3955904

-----RefParam INFO----------------------------------------------------------------
    ID    EnPred     Base   Enhance     EnIDR
     0      Yes        1        0      Yes

-----ROI INFO---------------------------------------------------------------------
    ID     index    bAbsQp    Qp     width    height    startx    starty

-----Syntax INFO------------------------------------------------------------------
    ID   slcsplt   slcmode   slcsize
     0      No       N/A       N/A

-----Inter & Intra prediction INFO------------------------------------------------
    ID   profile  HWsize  VWsize  P16x16   P16x8   P8x16    P8x8   MvExt  I16x16    Inxn
Ipcm
     0       hp       9       4     Yes     Yes     Yes     Yes     Yes     Yes     Yes     No

-----Syntax INFO------------------------------------------------------------------
    ID   profile   EntrpyI   EntrpyP  Itrans  ptrans QMatrix   POC   DblkIdc   alpha    beta
     0       hp     cabac     cabac     all     all     No      2        0       0       0
```

【调试信息分析】

在进行多路 h264 编码时，最常见的问题是用户的上层软件获取码流不及时，导致码流 buffer 不足，进而导致编码丢帧。确认这种情况最好的办法是查看 Disc、 BufLeak 和 UnrdStr。BufLeak 表示由于软件发现码流 buffer 不足而丢弃的帧数，Disc 往往表示由于硬件发现码流 buffer 不足而丢弃的帧数。BufLeak+Disc 通常就是由于码流 buffer 不足而丢弃的帧数。UnrdStr 表示码流 buffer 中剩余的码流帧数，这个数字越大，则表示用户获取码流越不及时。

【参数说明】

参数		描述
MODULE PARAM	ContiBuffer	码流用户态虚拟地址连续。 0：不连续； 1：连续。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	H264eVBSource	编码帧存分配方式： 1：编码 PrivateVB 池方式； 2：编码 UserVB 池方式。
	H264eRcnEqualRef	重构帧亮度是否复用参考值亮度内存： 0：否； 1：是。
	H264eMiniBufMode	码流 buffer 配置模式。 0：一般模式； 1：省内存模式。
CHN ATTR	ID	通道号。
	MaxWidth	编码通道最大宽度，以像素为单位。
	MaxHeight	编码通道最大高度，以像素为单位。
	PicWidth	编码码流宽度，以像素为单位。
	PicHeight	编码码流高度，以像素为单位。
	profile	编码通道 profile。 Base：baseline； Mp：main profile； Hp：high profile。
	MainStr	是否主码流。 取值：{0, 1}。
	RefMode	编码通道的跳帧参考模式。 1X：1 倍跳帧参考； 2X：2 倍跳帧参考； 4X：4 倍跳帧参考。
	BufSize	码流 buffer 大小，以字节为单位。
	ByFrame	是否按帧获取码流。 取值：{0, 1}。
	MaxStrCnt	允许码流 Buffer 缓存的最大帧数。 缺省值：0x1FFFFFFF。
	PoolId	当前编码通道帧存来自哪个 VB 池。
PICTURE	ID	通道号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
INFO	EncdStart	接收图像数。 接收到待编码图像后，此项值加 1。
	EncdSucceed	编码成功的帧数。
	Lost	编码时丢弃的帧数，包括以下几项： 1、 由于编码器硬件异常或出现超大帧而丢弃的帧数，即 Disc 表示的帧数。 2、 RC 进行帧率控制而丢弃的帧数，即 Skip 表示的帧数。 3、 码流 buffer 不够而丢弃的帧数，即 BufLeak 表示的帧数
	Disc	由于编码器硬件异常或出现超大帧而丢弃的帧数,有以下几种可能: 1、 编码器硬件在编码的过程中发现码流 buffer 不够。 2、 出现超大帧。 3、 编码器硬件出现异常，如死机。 第 1 种情况最为常见。如果 Disc 持续增加，很有可能是上层软件获取码流不及时导致码流 buffer 不足。
	Skip	编码进行帧率控制而丢弃的帧数。
	BufLeak	由于码流 buffer 不足导致的丢帧个数。 编码器启动编码前，发现码流 Buffer 可用空间非常小而丢掉一帧图像时，此项值加 1。 当用户未能及时取走码流时，易出现此项值不为 0。
	Recode	编码器重编的帧数。
	RlsStr	用户已经取走并且释放的帧数。
	UnrdStr	用户未获取的帧数。
	GetVBFail	获取 VB 块失败的次数。
STREAM BUFFER	ID	通道号。
	base	码流 buffer 基地址。
	RdTail	读尾指针。
	RdHead	读头指针。
	WrTail	写尾指针。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	WrHead	写头指针。
	datalen	数据长度。
	buffree	空闲长度。
RefParam INFO	ID	通道编号。
	EnPred	base 层的帧是否被 base 层其他帧用作参考。
	Base	base 层的周期。
	Enhance	enhance 的周期。
	EnIDR	是否使能 IDR 帧。
ROI INFO	ID	通道号。
	index	Roi 的索引号。
	bAbsQp	Roi 是否采用绝对 qp 模式。
	Qp	Roi 配置的 qp 值。
	width	Roi 区域的宽度，以像素为单位。
	height	Roi 区域的高度，以像素为单位。
	startx	Roi 区域的起始水平坐标，以像素为单位。
	starty	Roi 区域的起始垂直坐标，以像素为单位。
Syntax INFO	ID	通道号。
	slcsplt	是否使用 slice 划分。
	slcmode	slice 划分的模式。 • ByteNum：按字节数进行划分。 • MbLine：按宏块行进行划分。
	slcsize	每个 slice 的大小。 当 slice 划分的模式是按字节数进行划分时，表示每个 slice 的字节数； 当 slice 划分的模式是按宏块行进行划分时，表示每个 slice 的宏块行数；
Inter & Intra prediction INFO	ID	通道号。
	profile	编码通道 profile 类型。 • Base：baseline。 • Mp：Main profile。 • hp ：High profile。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	HWsize	水平搜索窗的大小。
	VWsize	垂直搜索窗的大小。
	P16x16	Inter16x16 预测模式是否使能。
	P16x8	Inter16x8 预测模式是否使能。
	P8x16	Inter8x16 预测模式是否使能。
	P8x8	Inter8x8 预测模式是否使能。
	MvExt	运动搜索到图像边界时是否作扩边处理
	I16x16	Intra16x16 预测模式是否使能。
	Inxn	Intra8x8 和 Intra4x4 预测模式是否使能。
	Ipcm	Ipcm 预测模式是否使能。
Syntax INFO	ID	通道号。
	profile	编码通道 profile 类型。 • base：baseline。 • mp：Main profile。 • hp：High profile。
	EntrpyI	I 帧熵编码模式，cavlc 或 cabac。
	EntrpyP	P 帧熵编码模式，cavlc 或 cabac。
	Itrans	I 帧使用的变换模式： • all：同时使能 trans8x8 模式和 trans4x4 模式。 • 4x4：使能 trans4x4 模式。 • 8x8：使能 trans8x8 模式。
	ptrans	P 帧使用的变换模式： • all：同时使能 trans8x8 模式和 trans4x4 模式。 • 4x4：使能 trans4x4 模式。 • 8x8：使能 trans8x8 模式。
	QMatrix	是否使用自定义的量化表。
	POC	语法元素 pic_order_cnt_type 的值。
	DblkIdc	语法元素 disable_deblocking_filter_idc 的值。
	alpha	语法元素 slice_alpha_c0_offset_div2 的值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	beta	语法元素 slice_beta_offset_div2 的值。

13.9 JPEGE

【调试信息】

```
[JPEGE] Version: [Hi3531_MPP_V1.0.0.0 ], Build Time[Dec  3 2014, 11:34:45]
-----MODULE PARAM-----------------------------------------------------------
  ContiBuffer       JpegeMiniBufMode
           0              0

-----ATTRIBUTE--------------------------------------------------------------
ID  bMjpeg   PicType  MaxWidth MaxHeight  PicWidth PicHeight   ViFld   bMain   ByFrm    MCU
Qfactor
0     Yes     YUV420      1920      1080      704      576        No      Yes     Yes      0
48

-----STATUS1----------------------------------------------------------------
ID            BufLen         FreeLen        StrmCnt        MaxStrm
0            4149248        775744         59             200

-----STATUS2----------------------------------------------------------------
ID    PicRec    PicCoded    PicDroped    PicDisc    NoStmCnt    NoBuf    RcFail    FullInt
PicRecode
0     812       643         168          0          0           168      0         0         0

-----STREAM BUFFER----------------------------------------------------------
ID      Base           RdTail  RdHead  WrTail  WrHead  BufLen  DataLen BufFree
0      0xcb400000      788480  11904   11904   12672   4149248    0    775744
```

【调试信息分析】

记录 JPEG 编码过程中，各通道的编码属性、状态以及历史信息统计，最多有 64 路编码通道。可配合用于定位系统阻塞以及丢帧等问题。

【参数说明】

参数		描述
MODULE PARAM	ContiBuffer	码流用户态虚拟地址连续。 0：不连续； 1：连续。
	JpegeMiniBufMode	码流 buffer 配置模式。 0：一般模式； 1：省内存模式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
ATTRIBUTE	ID	通道号。
	bMjpeg	是否 MJPEG 编码。 NO：JPEG 抓拍。 YES：MJPEG。
	PicType	图像类型，yuv422 或 yuv420。
	MaxWidth	编码通道最大宽度，以像素为单位。
	MaxHeight	编码通道最大高度，以像素为单位。
	PicWidth	图像宽度。
	PicHeight	图像高度。
	ViFld	VI 输入图像属性。 0：帧； 1：场。
	MCU	每个 ECS 的 MCU 个数。
	ByFrm	是否按帧获取码流。 取值：{0, 1}。
	bMain	是否主码流。 取值：{0, 1}。
	Qfactor	通道采用的 Qfactor。
	ContiBuffer	码流用户态虚拟地址连续。 0：不连续； 1：连续。
Status1	ID	通道号
	BufLen	码流 Buffer 总长度。
	FreeLen	空闲码流 Buffer 长度。
	StrmCnt	当前码流 buffer 缓存的帧数。
	MaxStrm	允许码流 buffer 缓存的最大帧数。 缺省值：100000。
Status2	ID	通道号。
	PicRec	发送来编码的图像总数。
	PicCoded	编码成功的帧数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	PicDroped	编码前总共丢弃的图像，包括 Nostmcnt、NoBuf，RcFail 表示的图像的总和。
	PicDisc	由于编码器硬件异常或出现超大帧而丢弃的帧数,有以下几种可能: 1、 编码器硬件在编码的过程中发现码流 buffer 不够。 2、 出现超大帧。 3、 编码器硬件出现异常，如死机。 第 1 种情况最为常见。如果 Disc 持续增加，很有可能是上层软件获取码流不及时导致码流 buffer 不足。
	Nostmcnt	码流 Buffer 中缓存的码流帧数超出最大码流帧数导致的丢帧次数。
	NoBuf	发现码流 Buffer 不足的次数（会导致丢帧）。编码器启动编码前，发现码流 Buffer 可用空间非常小而丢掉一帧图像时，此项值加 1。 当用户未能及时取走码流时，易出现此项值不为 0。
	RcFail	码率/帧率控制失败的次数（会导致丢帧）。
	FullInt	硬件上报满中断次数。编码器在编码过程中如果发现码流 buffer 不足，会上报码流 buffer 满中断。 当用户未能及时取走码流时，易出现此项值不为 0。
	PicRecode	超大帧重编码的帧数。
STREAM BUFFER	ID	通道号。
	base	码流 buffer 基地址。
	RdTail	读尾指针。
	RdHead	读头指针。
	WrTail	写尾指针。
	WrHead	写头指针。
	Buflen	码流 buffer 长度
	datalen	数据长度。
	buffree	空闲长度。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

13.10 RC

【调试信息】

```
# cat /proc/umap/rc
```

以下是 CBR 模式的 PORC 信息：

```
[RC] Version: [Hi3531_MPP_V1.0.D.0 ], Build Time[Mar 27 2015, 17:01:18]

----- BASE PARAMS -----------------------------------------------------------
    ID    Gop  StatTm  ViFr      TrgFr ProType  RcMode  Br(kbps)  FluLev  IQp  PQp   MinQp
MinIQp   MaxQp
     0     30       1    30       30/0     96   CBRV2      4096       0  N/A  N/A      10
10      51

-----RUN COMM PARAM 1-------------------------------------------------------
    ID   QpDelta                   ThrdI(12)
     0        2  [  5,  5,  5, 10, 10, 10,255,255,255,255,255,255]

-----RUN COMM PARAM 2-------------------------------------------------------
    ID     PARAM                   ThrdP(12)
     0   DEFAULT  [  5,  5,  5,255,255,255,255,255,255,255,255,255]

-----RUN COMM PARAM 3-------------------------------------------------------
    ID   bLost     LostThr  LostMode   LostGap      Priority
     0       0    83886080    NORMAl         0       BitRate

-----RUN CBR PARAM1 --------------------------------------------------------
ID  MinIprop  MaxIprop   MaxQp   MaxStQp   MinQp  MaxPPDltQp  MaxIPDltQp IPDltQp  QLevel
MaxReEncTimes
0         1       20      51       51      10          3          5       6       3         2

-----RUN CBR PARAM2 --------------------------------------------------------
    ID   SprFrmMod   SprIFrm   SprPFrm   RQRtio0   RQRtio1   RQRtio2   RQRtio3   RQRtio4
RQRtio5   RQRtio6   RQRtio7
     0        None    500000    500000        50        75        75        80        80        80
80         0

-----RUN VBR PARAM ---------------------------------------------------------
    ID   DltQp   ChgPs   SprFrmMod   SprIFrm   SprPFrm  MinIprop  MaxIprop

-----RUN INFO---------------------------------------------------------------
    ID   InsBr(kbps)   InsFr    WatL   CfgBt(kb)  RealBt(kb)   MaxRlBt(kb)   StartQp   MinQp
MaxQp
     0         3567      29    4600         107         146          936        24      10
51

-----RUN INFO2--------------------------------------------------------------
    ID    IpRt      Tr   Delta   TbTar      Tb

-----IMG INFO---------------------------------------------------------------
    ID   InterRl   IntraRl  InterSum  IntraSum    MvCnt   InterMb    Status
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

以下是 VBR 模式的 PROC 信息：

```
# cat /proc/umap/rc

[RC] Version: [Hi3531_MPP_V1.0.D.0 ], Build Time[Mar 31 2015, 16:30:08]

----- BASE PARAMS -----------------------------------------------------------
    ID    Gop  StatTm  ViFr      TrgFr ProType  RcMode  Br(kbps)  FluLev  IQp  PQp   MinQp
MinIQp   MaxQp
     0     25       1    25       25/0     96     VBR      6144     N/A  N/A  N/A      10
10      40

-----RUN COMM PARAM 1-------------------------------------------------------
    ID   QpDelta                    ThrdI(12)
     0        0  [255,255,255,255,255,255,255,255,255,255,255,255]

-----RUN COMM PARAM 2-------------------------------------------------------
    ID     PARAM                    ThrdP(12)
     0   DEFAULT  [255,255,255,255,255,255,255,255,255,255,255,255]

-----RUN COMM PARAM 3-------------------------------------------------------
    ID   bLost     LostThr  LostMode   LostGap      Priority
     0       0    83886080    NORMAl         0     SuperFrame

-----RUN CBR PARAM1 --------------------------------------------------------
    ID  MinIprop  MaxIprop   MaxQp   MaxStQp   MinQp  MaxPPDltQp  MaxIPDltQp IPDltQp
QLevel  MaxReEncTimes

-----RUN CBR PARAM2 --------------------------------------------------------
    ID   SprFrmMod   SprIFrm   SprPFrm   RQRtio0   RQRtio1   RQRtio2   RQRtio3   RQRtio4
RQRtio5   RQRtio6   RQRtio7

-----RUN VBR PARAM ---------------------------------------------------------
    ID   DltQp   ChgPs   SprFrmMod   SprIFrm   SprPFrm  MinIprop  MaxIprop
     0       2      90        None    500000    500000         1        20

-----RUN INFO---------------------------------------------------------------
    ID   InsBr(kbps)   InsFr    WatL   CfgBt(kb)  RealBt(kb)   MaxRlBt(kb)   StartQp   MinQp
MaxQp
     0            0       0     384          0          0            0         0       0       0

-----RUN INFO2--------------------------------------------------------------
    ID    IpRt      Tr   Delta   TbTar      Tb
     0       0       0       0       0       0

-----IMG INFO---------------------------------------------------------------
    ID   InterRl   IntraRl  InterSum  IntraSum     MvCnt   InterMb    Status
     0         0         0         0         0         0         0         7
```

以下是 FIXQP 模式的 PROC 信息：

```
[RC] Version: [Hi3531_MPP_V1.0.D.0 ], Build Time[Mar 27 2015, 17:01:18]

----- BASE PARAMS -----------------------------------------------------------
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
    ID    Gop  StatTm  ViFr      TrgFr ProType  RcMode  Br(kbps)  FluLev  IQp  PQp   MinQp
MinIQp   MaxQp
     0     25       1    25       25/0     96   FIXQP       N/A     N/A   20   23     N/A
N/A     N/A

-----RUN COMM PARAM 1-----------------------------------------------------------
    ID   QpDelta                    ThrdI(12)
     0        0  [255,255,255,255,255,255,255,255,255,255,255,255]

-----RUN COMM PARAM 2-----------------------------------------------------------
    ID    PARAM                    ThrdP(12)
     0   DEFAULT  [255,255,255,255,255,255,255,255,255,255,255,255]


-----RUN COMM PARAM 3-----------------------------------------------------------
    ID   bLost    LostThr  LostMode   LostGap      Priority
     0      0   83886080    NORMAl        0     SuperFrame

-----RUN CBR PARAM1 ------------------------------------------------------------
    ID  MinIprop  MaxIprop   MaxQp   MaxStQp   MinQp  MaxPPDltQp  MaxIPDltQp IPDltQp
QLevel  MaxReEncTimes

-----RUN CBR PARAM2 ------------------------------------------------------------
    ID   SprFrmMod   SprIFrm   SprPFrm   RQRtio0   RQRtio1   RQRtio2   RQRtio3   RQRtio4
RQRtio5   RQRtio6   RQRtio7

-----RUN VBR PARAM -------------------------------------------------------------
    ID   DltQp   ChgPs   SprFrmMod   SprIFrm   SprPFrm  MinIprop  MaxIprop

-----RUN INFO------------------------------------------------------------------
    ID   InsBr(kbps)   InsFr    WatL   CfgBt(kb)  RealBt(kb)   MaxRlBt(kb)   StartQp   MinQp
MaxQp
     0           0      0      0          0          0           0        0      0      0

-----RUN INFO2-----------------------------------------------------------------
    ID    IpRt      Tr   Delta   TbTar     Tb

-----IMG INFO------------------------------------------------------------------
    ID   InterRl   IntraRl  InterSum  IntraSum    MvCnt   InterMb    Status
```

【调试信息分析】

记录当前编码码率控制的信息。

【参数说明】

参数		描述
BASE PARAMS	ID	编码通道号。
	Gop	编码 Gop。（Group Of Pictures）
	StatTm	码率统计时间，单位秒。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	ViFr	VI 发送图像帧率。
	TrgFr	编码目标帧率。
	ProType	编码类型。
	RcMode	码率控制模式。（CBR/VBR/FixQp）
	Br(kbps)	码率，单位 kbps。
	FluLev	波动等级，CBR 有效。
	IQp	I 帧 Qp，FixQp 有效。
	PQp	P 帧 Qp，FixQp 有效。
	MinQp	最小 Qp，VBR 时有效
	MinIQp	I 帧最小 QP，CBR 和 VBR 时有效。
	MaxQp	最大 Qp，VBR 时有效
RUN COMM PARAM1		请参见视频编码章节中关于 VENC_RC_PARAM_S 结构体的描述。
RUN COMM PARAM 2		请参见视频编码章节中关于 VENC_RC_PARAM_S 结构体的描述。
RUN COMM PARAM 3	ChnId	通道号
	bLost	瞬时码率超出时的丢帧开关
	LostThr	瞬时码率阈值
	LostMode	丢帧策略
	LostGap	连续丢帧最大个数
	Priority	码率控制的优先级。 BitRate：目标码率优先； SuperFrame：超大帧优先。
RUN CBR PARAM		请参见视频编码章节中关于 VENC_PARAM_H264_CBR_S 和 VENC_PARAM_H264_CBRV2_S 结构体的描述。
RUN VBR PARAM		请参见视频编码章节中关于 VENC_PARAM_H264_VBR_S 结构体的描述。
RUN INFO	ChnId	编码通道号。
	InsBr(kbps)	瞬时码率，单位 kbps。
	InsFr	瞬时帧率。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	WatL	码率水线，内部调试。
	CfgBt(kb)	当前帧的目标大小，单位 kb。
	RealBt(kb)	上一帧码流的实际大小，单位 kb。
	MaxRlBt(kb)	单帧码流的最大大小，单位 kb。
	StartQp	起始 Qp。
	MinQp	最小 Qp。
	MaxQp	最大 Qp。
RUN INFO2	ChnId	编码通道号。
	IpRt	统计的 I 帧和 P 帧的大小比例。（需要除以 64）
	Tr	根据瞬时码率调整的当前帧目标码率。
	Delta	水线降低因子，内部调试。
	TbTar	目标水线码率。
	Tb	根据水线码率调整的当前帧目标码率。
IMG INFO	ChnId	编码通道号。
	InterRl	帧间信息相关系数。
	IntraRl	帧内信息相关系数。
	InterSum	VPP 的 MAD 值。
	IntraSum	VPP 的 MSE 值。
	MvCnt	MV 向量和。
	InterMb	帧间模式的宏块比例。
	Status	场景模式。

13.11 REGION

【调试信息】

```
# cat /proc/umap/rgn

[RGN] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Sep 16 2011, 11:38:03]

-----REGION STATUS OF OVERLAY-----------------------------------------------------
 Hdl type Used PiFmt   W   H BgColor     Phy     Virt  Stride
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
   1    0    0     8 180 144    fc00 87c7e800 c90ed800     384

-----REGION CHN STATUS OF OVERLAY-----------------------------------------------
 Hdl typ mod dev chn bSh   X   Y AphF AphB lay bAQp QP bInv InvW InvH Luma ChnM
0    0   5    0   0    1  64  64  128     0    0    0   0     1   16    16   50     0

-----REGION STATUS OF COVER-----------------------------------------------------
 Hdl type Used
   0    1    0

-----REGION CHN STATUS OF COVER-------------------------------------------------
 Hdl type  mod  dev  chn bShow   X   Y   W   H   Field Color   lay
   0    1   16    0    0     0 100  82 260 260    frame f800     1
-----REGION STATUS OF COVEREX---------------------------------------------------
 Hdl type Used
0     2    0

-----REGION CHN STATUS OF COVEREX-----------------------------------------------
 Hdl type  mod  dev  chn bShow   X   Y   W   H   Field  Color   lay
0     2    16    0    0     1    96  96  64  64  frame   ff     3

-----REGION STATUS OF OVERLAYEX-------------------------------------------------
 Hdl type Used PiFmt   W   H   BgColor     Phy     Virt      Stride
0     3      0     8    64  64    fc      8f15d000  c4cc0000    128

-----REGION CHN STATUS OF OVERLAYEX---------------------------------------------
 Hdl typ mod dev chn Field bSh   X   Y  AphF AphB lay
0    3   16    0    0 frame  1   96  12  255  255   3
```

【调试信息分析】

记录当前区域资源信息。

【参数说明】

参数		描述
REGION STATUS OF OVERLAY OVERLAY 状态	Hdl	OVERLAY 的 Handle 号。
	type	OVERLAY 类型，值为 0。
	Used	资源正被使用的次数。
	PiFmt	OVERLAY 像素格式，参看 PIXEL_FORMAT_E。
	W	OVERLAY 宽度（单位：像素）。
	H	OVERLAY 高度（单位：像素）。
	BgColor	OVERLAY 背景色。
	Phy	OVERLAY 占用内存的物理地址。
	Virt	OVERLAY 占用内存的虚拟地址。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Stride	OVERLAY 内存跨度（单位：byte）。
REGION CHN STATUS OF OVERLAY OVERLAY 在通道中的显示状态	Hdl	OVERLAY 的 Handle 号。
	type	OVERLAY 类型，值为 0。
	mod	模块 ID，GRP 模块 ID 为 5。
	dev	设备号。
	bSh	是否在该通道显示。 0：隐藏； 1：显示。
	chn	通道号。
	X	在该通道显示的起始 X 坐标。
	Y	在该通道显示的起始 Y 坐标。
	AphF	在该通道显示的前景 alpha。
	AphB	在该通道显示的背景 alpha。
	lay	在该通道显示的层次。
	bAQp	是否为绝对 QP 值。 0：相对 QP 值； 1：绝对 QP 值。
	QP	供编码模块使用的 QP 参数。
	bInv	是否开启反色功能。
	InvW	反色处理基本单元宽度。
	InvH	反色处理基本单元宽度。
	Luma	反色处理时的亮度阈值。
	ChnM	反色模式。 0: 视频背景亮度小于亮度阈值时触发反色。 1: 视频背景亮度大于亮度阈值时触发反色。
REGION CHN STATUS OF COVER COVER 状态	Hdl	COVER 的 Handle 号。
	type	COVER 类型，值为 1。
	Used	资源正被使用的次数。
REGION CHN	Hdl	COVER 的 Handle 号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
STATUS OF COVER COVER 在通道中的显示状态	type	COVER 类型，值为 1。
	mod	模块 ID，VI 模块 ID 为 16。
	dev	设备号。
	chn	通道号。
	bShow	是否在该通道显示。 0：隐藏； 1：显示。
	X	在该通道显示的起始 X 坐标。
	Y	在该通道显示的起始 Y 坐标。
	W	COVER 宽度（单位：像素）。
	H	COVER 高度（单位：像素）。
	Field	区域叠加到的帧/场属性。 frame：叠加到帧。 top：叠加到顶场。 botm：叠加到底场。
	Color	COVER 颜色。
	lay	在该通道显示的层次。
REGION CHN STATUS OF COVEREX COVEREX 状态	Hdl	COVEREX 的 Handle 号。
	type	COVEREX 类型，值为 2。
	Used	资源正被使用的次数。
REGION CHN STATUS OF COVEREX COVEREX 在通道中的显示状态	Hdl	COVEREX 的 Handle 号。
	type	COVEREX 类型，值为 2。
	mod	模块 ID，VI 模块 ID 为 16。
	dev	设备号。
	chn	通道号。
	bShow	是否在该通道显示。 0：隐藏； 1：显示。
	X	在该通道显示的起始 X 坐标。
	Y	在该通道显示的起始 Y 坐标。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	W	COVEREX 宽度（单位：像素）。
	H	COVEREX 高度（单位：像素）。
	Field	区域叠加到的帧/场属性。 frame：叠加到帧； top：叠加到顶场； botm：叠加到底场。
	Color	COVEREX 颜色。
	lay	在该通道显示的层次。
REGION STATUS OF OVERLAYEX OVERLAYEX 状态	Hdl	OVERLAYEX 的 Handle 号。
	type	OVERLAYEX 类型，值为 0。
	Used	资源正被使用的次数。
	PiFmt	OVERLAYEX 像素格式，参看 PIXEL_FORMAT_E。
	W	OVERLAYEX 宽度（单位：像素）。
	H	OVERLAYEX 高度（单位：像素）。
	BgColor	OVERLAYEX 背景色。
	Phy	OVERLAYEX 占用内存的物理地址。
	Virt	OVERLAYEX 占用内存的虚拟地址。
	Stride	OVERLAYEX 内存跨度（单位：byte）。
REGION CHN STATUS OF OVERLAYEX OVERLAYEX 在通道中的显示状态	Hdl	OVERLAYEX 的 Handle 号。
	type	OVERLAYEX 类型，值为 0。
	mod	模块 ID，VI 模块 ID 为 16。
	dev	设备号。
	chn	通道号。
	Field	区域叠加到的帧/场属性。 frame：叠加到帧； top：叠加到顶场； botm：叠加到底场。
	bSh	是否在该通道显示。 0：隐藏； 1：显示。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	X	在该通道显示的起始 X 坐标。
	Y	在该通道显示的起始 Y 坐标。
	AphF	在该通道显示的前景 alpha。
	AphB	在该通道显示的背景 alpha。
	lay	在该通道显示的层次。

13.12 VENC

【调试信息】

```
# cat /proc/umap/venc
[VENC] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Sep  6 2011, 09:51:16]
-----MODULE PARAM-----------------------------------------------------------
   BuffCache
          0
-----VENC CHN ATTR----------------------------------------------------------
NO.   Width  Height  Type Field VIField StreamType  ByFrame  BlockFlag Sequence
0     720    576     96    0      0         1        1         1         2ef
Registered  LeftBytes  LeftFrm  CurPacks
1                 0       0         0

-----VENC STREAM STATE------------------------------------------------------
NO.   FreeCnt   BusyCnt   UserCnt   UserGet   UserRls  GetTimes  Interval   FrameRate
0        4        0          0      875          875    850      15         25
```

【调试信息分析】

记录当前视频编码属性配置以及状态信息。

【参数说明】

参数		描述
MODULE PARAM	BuffCache	码流 buffer 物理地址映射方式。 0：非 cache 映射方式； 1：cache 映射方式。
VENC CHN ATTR 视频编码通道属性	NO	通道号。
	Width	编码通道宽度。
	Height	编码通道高度。
	Type	编码通道类型

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Field	帧场编码标识。 0：帧编码； 1：场编码。
	VIField	输入图像的帧场标识。 0：帧模式； 1：场模式。
	StreamType	主次码流标识。 0：次码流； 1：主码流。
	ByFrame	按帧获取码流标识。 0：按包获取； 1：按帧获取。
	BlockFlag	阻塞或非阻塞获取码流标识。 0：阻塞； 1：非阻塞。
	Sequence	序列号。 按帧获取时为帧序列号，按包获取时为包序列号。
	Registered	当前通道是否注册到通道组中。 取值：{0, 1}。
	LeftBytes	码流 buff 剩余的 byte 数。
	LeftFrm	码流 buffer 剩余的码流帧数。
	CurPacks	当前帧的码流包个数。此标记暂无效。
VENC CHN STATE 视频编码通道状态	NO	通道号。
	FreeCnt	码流 buffer 中 FREE 节点的个数。
	BusyCnt	码流 buffer 中 BUSY 节点的个数。
	UserCnt	码流 buffer 中 USER 节点的个数。 用户成功获取一帧码流后，该值加 1。 用户成功释放一帧码流后，该值减 1。
	UserGet	用户成功获取码流的码流包计数。
	UserRls	用户成功释放码流的码流包计数。
	GetTimes	用户获取码流的次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Interval	Venc 从编码器得到两帧码流的间隔时间，单位是微秒（us）。
	FrameRate	Venc 在 1 秒内从编码器得到的码流帧数，即编码器的编码帧率。

13.13 VDEC

【调试信息】

```
[VDEC] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Jan  7 2013, 11:30:57]

-----MODULE PARAM-----------------------------------------------------------
obey_minCR   MiniBufMode
0             0
-----CHN ATTR & PARAMS-----------------------------------------------------
  ID  TYPE   Prior    MaxW    MaxH   Width  Height   StrmInputMode   STATE   PREVMODE
   0  H264       5     720     576     640     480   FRAME/BLOCK     START       0
  ID  RefNum  SupportB  DispNum  BufSize  SCDBufSize  MaxSlice  MaxSPS  MaxPPS
   0    4        No        1     1658880    622080      100      15      12
ID  ErrThr   StrmThr   DecMode   OutMode   DnrDisp     DnrTf
0       0         0        IP       Dec       N/A       N/A

-----CHN STATE--------------------------------------------------------------
ID  PrtclErr  StrmUnSP StrmError RefNumErr  PicSizeErr  fmterror  PicBufSizeErr   RlsFail
Notify
0      120       20494        583          0          0           0         0             0
31393  ID  fps  TimerCnt   BufFLen   DataLen   UsrFLen    UsrLen   ptsBufF   ptsBufU
0    0     2868    1856885         0      5088         0        40         0

----- Detail Stream STATE ------------------------------------------------------
ID MpiSndNum   MpiSndLen  VdecNum  VdecLen  FmGetNum   FmGetLen  FmRlsNum  FmRlsLen
FmLstGet  FmRlsFail
   0    9852       133272073   9852     147780   9923       133419853   9922     133410498
9355         0

----- Detail FrameStore STATE --------------------------------------------------
ID  FmNewPic GetFromFm  RlsToFm Discard UsrSnd KerSnd KerRls FreeNode BusyNode UserNode
MeetEnd FrmInVdec
0       0       0         0        0        0       0       0        40       0      0      9852      0

----- Detail UserData STATE ----------------------------------------------------
  ID    MpiGet      MpiGetLen    MpiRls      MpiRlsLen   Discard     DiscardLen GetFromFm
GetFromFmLen   UsrFLen    UsrLen
0       0         0          0        0         0          0         0           0       5088      0
```

【调试信息分析】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

记录当前解码通道的使用状况及其属性配置，最多有 64 路解码通道。可用于检查属性配置以及当前解码通道统计状态。

【参数说明】

参数		描述
MODULE PARAM	obey_minCR	H.264 码流是否遵守协议中 minCR 的约定。 0：不遵守。默认值； 1：遵守。此时 minCR=2； 创建通道时，会根据 obey_minCR 取值来确定帧最大值。 0：帧最大值为 u32BufSize / 2。 1：帧最大值为 u32PicWidth x u32PicHeight x 1.5 / 2。 当码流中有超大帧（帧大小超过 u32PicWidth x u32PicHeight x 1.5 / 2）时，将 obey_minCR 选为 0。
	MiniBufMode	码流 buffer 配置模式 0：一般模式 1：省内存模式（只在帧模式解码下配置有效）
CHN ATTR PARAMS	ID	通道号。
	Type	解码通道类型。 PT_H264； PT_MJPEG； PT_JPEG。
	Prior	解码通道优先级。
	MaxW	配置的解码图像最大宽度。
	MaxH	配置的解码图像最大高度。
	Width	解码图像宽度。
	Height	解码图像高度。
	StrmInputMode	解码通道码流发送模式。包含两方面的内容： FRAME、STREAM：按帧还是按流式发送； BLOCK、NOBLOCK：阻塞还是非阻塞发送。
	STATE	解码通道是否开始接收码流。 START：开始接收； STOP：停止接收

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	PREVMODE	是否为解码预览模式。 0：回放模式； 1：预览模式。
	RefNum	最大参考帧个数，除 JPEG 解码，其余解码通道有效。
	SupportB	是否支持 B 帧解码。
	DispNum	显示端占用的帧数。
	BufSize	Vdec 内部码流 buffer 大小，以 byte 为单位。
	SCDBufSize	Vfmw 内部经过 SCD 切割之后码流 buffer 大小，以 byte 为单位。
	MaxSlice	每帧最大 slice 数，只有 H264 协议有效
	MaxSPS	每帧最大 SPS 数，只有 H264 协议有效
	MaxPPS	每帧最大 PPS 数，只有 H264 协议有效
	ErrThr	码流错误阈值。
	StrmThr	解码前丢帧阈值。
	DecMode	解码模式。
	OutMode	解码图像输出顺序。
	DnrDisp	保留参数。
	DnrTf	保留参数。
CHN STATE	ID	通道号。
	PrtclErr	解码器上报协议错误次数。
	StrmUnSP	解码器上报不支持规格的次数。
	StrmError	解码器上报码流语法错误的次数。
	RefNumErr	解码器上报码流参考帧帧数超过配置帧数的次数。
	PicSizeErr	解码器上报解码图像超过通道大小的次数。
	fmterror	不支持的格式。
	PicBufSizeErr	JPEG 解码 YUV422 或 YUV444 的图像时帧存 Buffer 不够大。
	RlsFail	释放图像错误。
	fmterror	不支持的格式。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Notify	解码器内部事件上报的次数。
	ID	通道号。
	fps	解码实际帧率。
	TimerCnt	解码器内部定时器的启动次数。
	BufFLen	VDEC 码流 buffer 的剩余空间，以 byte 为单位。
	DataLen	VDEC 码流 buffer 的数据长度，以 byte 为单位。
	UsrFLen	VDEC 用户数据 buffer 的剩余空间，以 byte 为单位。
	UsrLen	VDEC 用户数据 buffer 的数据长度，以 byte 为单位。
	ptsBufF	按帧发送时，VDEC 帧队列的空闲个数。
	ptsBufU	按帧发送时，VDEC 帧队列的使用个数。
Detail Stream STATE	ID	通道号。
	MpiSndNum	用户发送码流次数。
	MpiSndLen	用户发送码流累计长度，字节为单位。
	VdecNum	Vdec 内部增加结束码次数，仅在按帧发送时有效。
	VdecLen	Vdec 内部增加码流累计长度，字节为单位。
	FmGetNum	Vfmw 获取码流次数。
	FmGetLen	Vfmw 获取码流累计长度，字节为单位。
	FmRlsNum	Vfmw 释放码流次数。
	FmRlsLen	Vfmw 释放码流累计长度，字节为单位。
	FmLstGet	Vfmw 最后一次获取的数据长度，字节为单位。
	FmRlsFail	Vfmw 释放码流失败的次数。
Detail FrameStore STATE	ID	通道号。
	FmNewPic	Vfmw 解码出图像帧数
	GetFromFm	从 Vfmw 成功获取的图像帧数。
	RlsToFm	释放给 Vfmw 的图像帧数。
	Discard	Vdec 丢弃图像帧数。
	UsrSnd	Vdec 从用户态发送图像帧数。
	KerSnd	Vdec 从内核态发送图像给其他模块的帧数。
	KerRls	用户或其他模块释放图像给 Vdec 的帧数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	FreeNode	空闲帧存节点数。
	BusyNode	未发送的帧存节点数。
	UserNode	已发送的帧存节点数。
	MeetEnd	Vfmw 已经解析的结束码次数。
	FrmInVdec	解码器中缓存的帧数。包含未解码的码流及已经解码但未发送的图像。
Detail UserData STATE	ID	通道号。
	MpiGet	用户获取用户数据次数。
	MpiGetLen	用户获取用户数据长度。
	MpiRls	用户释放用户数据次数。
	MpiRlsLen	用户释放用户数据长度。
	Discard	Vdec 丢弃用户数据的次数。
	DiscardLen	Vdec 丢弃用户数据长度。
	GetFromFm	Vdec 得到用户数据的次数。
	GetFromFmLen	Vdec 得到用户数据的长度。
	UsrFLen	用户数据 buffer 空闲长度。
	UsrLen	用户数据 buffer 已占用长度。

13.14 VI

【调试信息】

```
# cat /proc/umap/vi

[VIU] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time: [Apr 12 2012, 14:54:51]

-----MODULE PARAM-----------------------------------------------------------
detect_err_frame drop_err_frame stop_int_level  max_cas_gap
          0          0                0         28000

-----VI DEV ATTR------------------------------------------------------------
 Dev   IntfM  WkM  ComMsk0  ComMsk1 ScanM AD0 AD1 AD2 AD3   Seq  DPath DType DRev
   0   BT656 4Mux ff000000       0     I   -1   -1  -1  -1   UYVY ByPass   YUV    N

-----VI HIGH DEV ATTR-------------------------------------------------------
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
 Dev  InputM  WkM  ComMsk0  ComMsk1 ScanM AD0 AD1 AD2 AD3   Seq CombM CompM ClkM  Fix FldP
DPath DType DRev

-----VI PHYCHN ATTR-----------------------------------------------------------
 PhyChn CapX CapY CapW CapH DstW DstH CapSel Mirror  Flip IntEn PixFom SrcRat DstRat
     0    0    0     720   576 720   576   both     N      N     Y    sp422     25      6

-----VI PHYCHN MINOR ATTR-----------------------------------------------------------
 PhyChn CapX CapY  CapW  CapH  DstW  DstH CapSel Mirror   Flip PixFom MixCap DwScal
     0    0    0     360    288   360   288   both     N      N      sp422      Y      N

-----VI PHYCHN STATUS 1-------------------------------------------------------------
PhyChn  Dev  IntCnt  VbFail  LosInt  TopLos  BotLos BufCnt  IntT  SendT  Field  Stride
     0    0     277       0       1       0       1       2      105     25    intl     368

-----VI PHYCHN STATUS 2-------------------------------------------------------------
PhyChn  MaxIntT IntGapT MaxGapT  OverCnt LIntCnt  ThrCnt AutoDis CasAutD TmgErr        0
208    20069   20074        1       1        25      0        0       0
-----VI PHYCHN STATUS 2-------------------------------------------------------------
ccErrN    IntRat
2         24
-----VI PHYCHN VBI ATTR-------------------------------------------------------------
 PhyChn   VbiId      X      Y   Local    Len   CasErrN

-----VI CASCADE CHN ATTR-------------------------------------------------------------
 CasChn    Dev  PhyChn  SrcRat  DstRat

-----USER PIC INFO-----------------------------------------------------------------
UpicID  Width  Height  Stride  Field  PixForm  PoolID  PhyAddr  bUpdate

-----VI CHN STATUS-----------------------------------------------------------------
ViChn   bEnUsrP   FrmTime   FrmRate   SendCnt    SwLost   Rotate     422to420
   0       N       39990       25       275        0        NONE       N

-----VI CHN CALL DSU STATUS 1------------------------------------------------
 ViChn UsrBgnNOk  UsrCancel  UsrEndOk UsrCbOk  CvrBgnNOk  CvrCancel  CvrEndOk CvrCbOk

-----VI CHN CALL DSU STATUS 2------------------------------------------------
 ViChn  OsdBgnNOk  OsdCancel OsdEndOk  OsdCbOk  ScaleNOk  SclCancel  SclEndOk  SclCbOk

-----VI CHN CALL DSU STATUS 3------------------------------------------------
 ViChn  RotateNOk  RotCancel  RotEndOk  RotCbOk  LDCNOk  LDCCancel  LDCEndOk  LDCCbOk
```

【调试信息分析】

记录当前视频输入设备及通道的属性配置以及状态信息。

【参数说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
MODULE PARAM	detect_err_frame	信号不稳定时，实时丢弃侦测到的错误帧图像策略。 >0：当检测到连续错误帧数大于该值时，则认为是时序配错，后面的帧不再丢弃； 0：默认值，表示实时丢弃检测到的错误帧图像。 <0：关闭检测错误帧功能。
	drop_err_frame	当检测到当前帧是错误帧时，认为接下来的几帧也可能是错误帧，应该丢弃。 0：默认值，表示不开启连续丢帧功能，只丢弃当前的错误帧； >0：该参数表示当检测到图像错误时，连续丢 drop_err_frame 帧(含当前帧)，不管后面的图像是否正确。
	stop_int_level	当检测到两个中断之间的时间间隔小于该值时，自动关闭响应中断，防止因中断过多导致应用程序得不到响应。 默认值：0，单位为 us。
	max_cas_gap	VI-VO 双路级联 1080P60 模式下，当业务量大时会导致 vi 中断被延迟，需要丢帧，否则会串屏。因此，当两个中断之间的时间间隔大于该值时，就丢弃该帧。 默认值：28000，单位为 us。
VI DEV ATTR 视频输入设备属性（请参考数据结构 VI_DEV_ATTR_S）	Dev	设备号。
	IntfM	输入模式。
	WkM	工作模式。
	ComMsk0	分量 0 掩码。
	ComMsk1	分量 1 掩码。
	ScanM	隔行或逐行输入。 取值：{I，P}。
	AD0	AD 号。
	AD1	AD 号。
	AD2	AD 号。
	AD3	AD 号。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Seq	数据顺序。 取值：{ VUVU, UVUV, UYVY ,VYUY, YUYV, YVYU}。
	DPath	ISP 是否使能。默认为“ByPass”，不开启（DVR 不支持 ISP）。
	DType	输入数据类型，默认为 YUV。
	DRev	AD/Sensor 与 VI 连接是否反向，默认为“N”，不反向。
VI HIGH DEV ATTR 视频输入高级设备属性（请参考数据结构 VI_DEV_ATTR_S）	Dev	设备号。
	InputM	输入模式。
	WkM	工作模式。
	ComMsk0	分量 0 掩码。
	ComMsk1	分量 1 掩码。
	ScanM	隔行或逐行输入。 取值：{I，P}。
	AD0	AD 号。
	AD1	AD 号。
	AD2	AD 号。
	AD3	AD 号。
	Seq	数据顺序。 取值：{ VUVU, UVUV, UYVY ,VYUY, YUYV, YVYU}。
	CombM	分量复合/分离模式。 取值：{ COMP, SEPAR}
	CompM	分量单/双模式。 取值：{SING, DOUB}
	ClkM	时钟模式。 取值：{UP, DOWN}
	Fix	定时基准码最高位配置。 取值：{0, 1}
	FldP	定时基准码场指示模式。 取值：{STD, NSTD}

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	DPath	ISP 是否使能。默认为“ByPass”，不开启（DVR 不支持 ISP）。
	DType	输入数据类型，默认为 YUV。
	DRev	AD/Sensor 与 VI 连接是否反向，默认为“N”，不反向。
VI PHYCHN ATTR VI 物理通道主属性（请参考数据结构 VI_CHN_ATTR_S）	PhyChn	VI 物理通道号。
	CapX	捕获区域起始位置 X 坐标。
	CapY	捕获区域起始位置 Y 坐标。
	CapW	捕获区域宽度。
	CapH	捕获区域高度。
	DstW	图像目标宽度（由捕获图像缩放得到）。
	DstH	图像目标高度（由捕获图像缩放得到）。
	CapSel	帧场选择。 取值：{top, bottom, both}。
	Mirror	是否水平翻转。 取值：{Y, N}。
	Flip	是否垂直翻转。 取值：{Y, N}。
	IntEn	是否响应中断。 取值：{Y，N}。
	PixFom	像素格式。 取值：{sp420, sp422}。
	SrcRat	VI 物理通道采集帧率控制的源帧率。 取值直接根据设置通道属性的 VI_CHN_ATTR_S 数据结构的值来配置。
	DstRa	VI 物理通道采集帧率控制的目标帧率。 取值直接根据设置通道属性的 VI_CHN_ATTR_S 数据结构的值来配置。
VI PHYCHN MINOR ATTR	MixCap	是否混合采集状态。 取值：{Y, N}。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
VI 物理通道次属性（相对主属性，少了 SrcRat、IntEn 与 DstRa 项，但多了 MixCap 与 DwScal 成员）	DwScal	VI 是否已经缩放。 取值：{Y, N}。
VI PHYCHN STATUS VI 物理通道基本状态 1	Dev	VI 物理通道绑定的 VI 设备号。
	PhyChn	VI 物理通道号。
	IntCnt	中断次数。 每次帧中断中取值增加 1。 如果 VI 工作不正常，此值可能不会增加。
	VBFail	获取 VideoBuffer 失败次数。 VI 采集每帧图像前先获取通道图像大小的 VideoBuffer，如果失败，此值加 1；应该使此项取值为 0，否则说明相应的公共 VideoBuffer 配置不当。
	LosInt	丢失中断次数。 VI 需要在最初的 1 或 2 个中断中做图像采集的准备工作，此项值为 1 或 2 时是正常情况。
	TopLos	丢失顶场中断数
	BotLos	丢失底场中断数
	BufCnt	VI 占用的图像 VB 数目。
	IntT	中断处理时间（us）。
	SendT	中断处理中发送图像的时间（us）。
	Field	图像帧场标志。 frm：逐行图像。 intl：隔行图像。
	Stride	图片 Stride。
VI PHYCHN STATUS 2 VI 物理通道基本状态 2	PhyChn	VI 物理通道号。
	MaxIntT	最大中断处理时间。
	IntGapT	相邻两中断之间的时间间隔。
	MaxGapT	相邻两中断之间的最大时间间隔。
	OverCnt	相邻两个中断时间间隔超过 40ms 的次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	LIntCnt	中断处理时间超过 10ms 的次数。
	ThrCnt	帧/场丢失次数。程序刚起来时，多数时候其值会为 1，只要后面不再增加，则为正常现象。
	AutoDis	因中断过多而自动关闭中断 mask 的次数。
	CasAutD	VI-VO 级联时，因中断被延迟而丢弃的图像帧数。
	TmgErr	统计因 BUF 溢出等异常的中断数。
	ccErrN	中断异常次数（记录未采集完成的中断次数，如果非满帧采集或 VBFail 项有增加，该计数会增加；如果检查上述两项没有问题，该计数还是持续增加，请检查输入时序是否正确或者输入线接触是否良好）。
	IntRat	由 VI 中断决定的采集帧率。
VI PHYCHN VBI ATTR VI 物理通道 VBI 属性	PhyChn	VI 物理通道号。
	VbiId	VBI 区域号
	X	VBI 起始位置 X 坐标。
	Y	VBI 起始位置 Y 坐标。
	Local	VBI 位置。
	Len	VBI 数据长度，以 word 即 4 个字节为单位。
	CasErrN	VI-VO 双路级联时，检测出图像 id 错误的计数。
VI CASCADE CHN ATTR VI 级联通道属性	CasChn	VI 级联通道号。
	Dev	开启级联功能的 VI 设备号。
	PhyChn	开启级联功能的 VI 物理通道号。
	SrcRat	VI 级联通道采集帧率控制的源帧率。
	DstRat	VI 级联通道采集帧率控制的目标帧率。
USER PIC INFO VI 插入用户图片信息	UPicID	通道使用的用户图片号
	Width	开启用户图片功能的通道宽度。
	Height	开启用户图片功能的通道高度。
	Stride	开启用户图片功能的通道跨度。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Field	帧场选择。 固定为“frm”。
	PixForm	像素格式。 取值：{sp420, sp422}。
	PoolID	用户图片对应的 pool 池号。
	PhyAddr	用户图片对应的 pool 池内存物理地址。
	bUpdate	用户图片是否更新。
VI CHN STATUS VI 通道基本状态	ViChn	VI 通道号。
	bEnUsrP	是否开启插入用户图片。
	FrmTime	视频帧帧间隔时间。 以微秒为单位。
	FrmRate	VI 通道采集帧率。
	SendCnt	VI 通道将采集到的图像发送出去的次数。
	SwLost	VI 通道发送图像时的丢帧次数。
	Rotate	旋转类型，可以旋转 90、180、270 度。
	422to420	是否开启 422 图像转为 420 功能。 取值：{Y，N}。
VI CHN CALL DSU STATUS VI 通道调用 DSU 的信息：Usr 对应用户图片；Cvr 对应 COVEREX_RGN 。	ViChn	VI 通道号。
	UsrBgnNOk	调用 DSU BeginJob 失败的次数。
	UsrCancel	调用 Cancel Job 的次数。
	UsrEndOk	提交 DSU Job 成功的次数。
	UsrCbOk	DSU 回调成功的次数。
	CvrBgnNOk	调用 DSU BeginJob 失败的次数。
	CvrCancel	调用 Cancel Job 的次数。
	CvrEndOk	提交 DSU Job 成功的次数。
	CvrCbOk	DSU 回调成功的次数。
VI CHN CALL DSU STATUS 2 VI 通道调用 DSU 的信息：	ViChn	VI 通道号。
	OsdBgnNOk	调用 DSU BeginJob 失败的次数。
	OsdCancel	调用 Cancel Job 的次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
Osd 对应 OVERLAYEX_RGN Scl 对应缩放	OsdEndOk	提交 DSU Job 成功的次数。
	OsdCbOk	DSU 回调成功的次数。
	ScaleNOk	调用 DSU BeginJob 失败的次数。
	SclCancel	调用 Cancel Job 的次数。
	SclEndOk	提交 DSU Job 成功的次数。
	SclCbOk	DSU 回调成功的次数。
VI CHN CALL DSU STATUS 3 VI 通道调用 DSU 的信息： Rotate 对应旋转； LDC 对应畸变校正。	ViChn	VI 通道号。
	RotateNOk	调用 DSU BeginJob 失败的次数。
	RotCancel	调用 Cancel Job 的次数。
	RotEndOk	提交 DSU Job 成功的次数。
	RotCbOk	DSU 回调成功的次数。
	LDCNOk	调用 DSU BeginJob 失败的次数。
	LDCCancel	调用 Cancel Job 的次数。
	LDCEndOk	提交 DSU Job 成功的次数。
	LDCCbOk	DSU 回调成功的次数。

13.15 VO

【调试信息】

```
# cat /proc/umap/vo

[VOU] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Sep 13 2011, 19:44:04]

-----MODULE PARAM-----------------------------------------------------------

-----DEV CONFIG-------------------------------------------------------------------
 DevId  DevEn   Mux1  Mux2  Mux3     InfSync   BkClr  DevFrt DispFrt DoubFrm  Toleration
Vtth
    0       y  BT1120               720P@50     ff     50       25        n   10000000
240

-----DEV VDAC STATUS--------------------------------------------------------
 DevId    VDAC
    0       -
-----DEV VIDEO STATUS-------------------------------------------------------
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
 DevId VideoEn     PiP  PixFmt    ImgW    ImgH   DispW   DispH
    0      y       n      420    1280     720    1280     720

-----DEV VIDEO STATUS 2-----------------------------------------------------
 DevId VideoEn     PiP EnChNum   Matrix    Luma    Cont     Hue    Satu
    0      y       n      1          0      50      50      50      50

-----DEV VIDEO STATUS 3-----------------------------------------------------
 DevId  SetBeg  SetEnd   WBCEn bCasCfg  bCasEn bCasSlv  CasRgn CasMode CasPatn
     0       n       n       n       n       n       n                       0

-----GRP STATUS-------------------------------------------------------------
 GrpId OnDev ChNum FulFr TgtFr Start Resrv   GrpGap            BasePts
-----CHN BASE INFO----------------------------------------------------------
DevId ChnId  ChnEn Prio DeFlk  ChnX  ChnY  ChnW  ChnH WinX  WinY  WinW  WinH DispX DispY
bSnap Field DoubFrm bCas CcPos  bRcvMatch
0     0      y     1      n      0      0     640    360   0     0     640    320    -1   -1   n
both   n  n     0

-----CHN PLAY INFO----------------------------------------------------------
 DevId ChnId Batch  Show Pause  Step Revrs Refsh Thrshd ChnFrt ChnGap
    0     0      n     y     n      n      n      n       3     25     40000

-----CHN PLAY INFO----------------------------------------------------------
 DevId ChnId   DisplayPts       PrePts      CurrPts     ScalePts    SetPts
    0     0    49320948834   49320948834   49321028288           0         0

-----HD CHN STATUS1---------------------------------------------------------
 DevId ChnId  QCnt NewDo OldDo  LCnt  SCnt ChRpt BusyN ShouD Dsped  Area AreaT
    0     0     17    12     0      0     11     7     0          1      1      0      0

-----HD CHN STATUS2---------------------------------------------------------
 DevId ChnId  Buf1  Buf2  Buf3  DBuf Stat1 Stat2 QNodeAddr SNodeAddr  DispAddr
    0     0    UseF  UseF    F     2    end    end   97291000   973e2800  973e2800

-----SD CHN STATUS----------------------------------------------------------
 DevId ChnId   Job  Task  LCnt  SCnt  ChRpt  DRpt CBusy DBusy ShouD Dsped b2Scl  ChnAddr
DispAddr
-----CHN OTHER INFO---------------------------------------------------------
 DevId ChnId bZoom ZmTyp ZoomX ZoomY ZoomW ZoomH  SrcW  SrcH FltTp  HFlt VlFlt VcFlt
    0     0     n      0      0      0      0      0      0      0      0      0      0      0

-----Wbc INFO---------------------------------------------------------------
 DevId  WbcW  WbcH PixFmt FrmRat RealRat  Mode  DataSrc  Depth  NotFin
-----GRAPHICS LAYER---------------------------------------------------------
 Layer BindDev
   HC0       0

-----LAYER CSC PARAM--------------------------------------------------------
 LAYERID  Matrix    Luma    Cont     Hue    Satu
       0        5      50      50       50      50

-----VGA PARAM--------------------------------------------------------------
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
DevId    Matrix  Luma    Cont    Hue    Satu    Gain
```

【调试信息分析】

记录当前 VO 的使用状况及其属性配置，包含设备状态，视频层状态，同步组状态以及通道状态。可用于动态获取当前 VO 的使用状态以便于调试或测试。

【参数说明】

参数		描述
DEV CONFIG	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	DevEn	设备是否使能。 n：禁止； y：使能。
	Mux1/Mux2 /Mux3	接口类型。 取值范围：VO_INTF_CVBS, VO_INTF_YPBPR, VO_INTF_VGA, VO_INTF_BT656, VO_INTF_BT1120, VO_INTF_HDMI
	InfSync	接口时序。 取值范围：[0, VO_OUTPUT_BUTT）
	BkClr	设备背景色。RGB888 格式。
	DevFrt	设备帧率，即刷新率，与时序相关。
	DispFrt	目标显示帧率，即拼接帧率。 取值范围：（0, DevFrt]
	DoubFrm	设备是否开启倍帧。 n：未开启； y：开启。
	Toleration	设备的容忍度，与帧率控制相关。
	Vtth	设备垂直时序中断门限。
DEV VDAC STATUS	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	VDAC	设备是否打开负载检测 Y：打开检测； N：未打开检测。 该芯片不支持负载检测功能。
DEV VIDEO	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
STATUS	VideoEn	视频层是否使能。 n：禁止；y：使能。
	PiP	PIP 层是否使能。 n：禁止；y：使能。
	PixFmt	输入图像像素格式。 只支持 PIXEL_FORMAT_YUV_SEMIPLANAR_422 和 PIXEL_FORMAT_YUV_SEMIPLANAR_420 两种。
	ImgW	视频层画布宽度
	ImgH	视频层画布高度
	DispW	显示区域宽度。
	DispH	显示区域高度。
DEV VIDEO STATUS 2	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	VideoEn	视频层是否使能。 n：禁止； y：使能。
	PiP	PIP 层是否使能。 n：禁止； y：使能。
	EnChNum	通道使能计数。即该设备有多少个通道处于使能状态。 取值范围：[0, VO_MAX_CHN_NUM）
	Matrix	CSC 矩阵选择。 取值范围：[0, 2]
	Luma	亮度 取值范围：[0, 100]，默认值 50
	Cont	对比度 取值范围：[0, 100]，默认值 50
	Hue	色调。 取值范围：[0, 100]，默认值 50
	Satu	饱和度 取值范围：[0, 100]，默认值 50

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
DEV VIDEO STATUS 3	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	SetBeg	设备是否设置 SetAttrBegin。 n：未设置；y：设置。
	SetEnd	设备是否设置 SetAttrEnd。 n：未设置；y：设置。
	WBCEn	设备是否使能视频回写。 n：禁止；y：使能。
	bCasCfg	设备是否设置级联属性。 n：未设置；y：设置。
	bCasEn	设备是否使能级联。 n：禁止；y：使能。
	bCasSlv	设备级联模式。 n：主模式；y：从模式。
	CasRgn	设备级联区域类型。 32Rgn：32 区域类型；64Rgn：64 区域类型。
	CasMode	设备级联单双路模式。 Single：单路模式；Dual：双路模式。
	CasPatn	设备级联 Pattern。 取值范围：[0, 128）
GRP STATUS	-	暂不支持
CHN BASE INFO	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）
	ChnEn	通道是否使能。 Y：是 N：否
	Prio	通道优先级。 取值范围：[0, VO_MAX_CHN_NUM）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	DeFlk	通道是否抗闪。隔行设备上，单场图像或者帧图像缩小时一般需要使能抗闪，否则都应设置为关闭。对于逐行输出设备该值都应该为禁止状态。 n：禁止； y：使能。
	ChnX	通道起始横坐标。
	ChnY	通道起始纵坐标。
	ChnW	通道宽度。
	ChnH	通道高度
	WinX	通道显示窗口起始横坐标（以通道属性中的起始点为原点的横坐标）
	WinY	通道显示窗口起始纵坐标（以通道属性中的起始点为原点的纵坐标）
	WinW	通道显示窗口宽度
	WinH	通道显示窗口高度
	DispX	高清设备 PIP 层上通道显示位置的水平坐标。该值对于非高清 PIP 层上的通道没有意义。
	DispY	高清设备 PIP 层上通道的显示位置的垂直坐标。该值对于非高清 PIP 层上的通道没有意义。
	bSnap	是否通道抓拍。 n：否； y：是。
	Field	显示帧场信息。设置图像的帧场标识。 取值范围：[0, VO_FIELD_BUTT)
	DoubFrm	通道的倍帧功能使能标志。 n：否； y：是。
	bCas	是否级联通道。 n：否； y：是。
	CcPos	级联位置。暂不支持。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	bRcvMatch	是否使能通道帧率匹配功能。 0：否； 1：是。
CHN PLAY INFO1	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）
	Batch	通道是否正在进行批处理。 n：否；y：是。
	Show	通道是否显示。 n：隐藏； y：显示。
	Pause	通道是否暂停。 n：禁止； y：使能。
	Step	通道是否步进。 n：禁止； y：使能。
	Revrs	是否反向播放。 n：禁止； y：使能。
	Refsh	通道是否刷新。 n：禁止； y：使能。
	Thrshd	通道显示门限值。对于高清设备表示通道中所有图像总的待显示次数最大值；对于标清设备表示通道缓冲队列最大可接收的图像帧数。
	ChnFrt	通道帧率。通道播放控制可以通过该数值反映。
	ChnGap	通道帧间隔。与通道帧率成反比。单位微秒。
CHN PLAY INFO2	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	DisplayPts	当前正在显示帧的时间戳。单位微秒。
	PrePts	通道 busy 队列中队尾节点图像的时间戳。单位微秒。
	CurrPts	当前 VPSS 拿来查询的图像时间戳。单位微秒。
	ScalePts	目标时间戳。下一帧要显示的时间戳。单位微秒。
	SetPts	HD 画面切换大小，最后一帧旧图像的时间戳。单位微秒。
HD CHN STATUS1	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）
	QCnt	VPSS 来查询的次数
	NewDo	返回 New+Do 的次数
	OldDo	返回 Old+Do 的次数
	LCnt	丢帧次数
	SCnt	通道接收其他模块发送帧计数。如果该数值为 0，表示 vo 还没有收到图像数据；如果该数值一直不增加，表示 vo 接收不到图像。
	BusyN	Busy 队列里的节点数
	ChRpt	通道重复显示总次数。如果通道暂停，该值会不断增加。
	ShouD	当前显示的图像应该显示的次数。
	Dsped	当前显示的图像已经显示的次数。如果通道暂停，该值会不断增加。
	Area	通道对应的视频层显示区域
	AreaT	通道切换大小时重新排序的视频层显示区域
HD CHN STATUS2	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）
	Buf1/Buf2/ Buf3	通道盒子 1/2/3 的使用状态。 取值：{UseF，F，E}。UseF 表示该盒子填充了内存，且被使用；F 表示该盒子填充了内存，但没有使用；E 表示盒子为空，没有填充内存。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	DBuf	通道当前显示帧用的盒子号。
	Stat1	通道切换大小的状态。 end：切换大小完成 ing：VPSS 正在处理切换后新分辨率的图像 seted：已收到新分辨率的图像，通道处于可切换状态。 begin：切换开始
	Stat2	通道由隐藏转为显示的状态。 end：隐藏转显示完成 ing：通道队列中已有图像待显示 begin：开始处理显示标志。
	QNodeAddr	VPSS 来查询时，VO 返回给 VPSS 的图像物理地址
	SNodeAddr	VO 接收到其他模块发来图像的物理地址
	DispAddr	当前显示图像的物理地址
SD CHN STATUS	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）
	Job	DSU 中任务计数。该数值表明 DSU 目前正在处理的拼接任务计数。
	Task	DSU 中 task 计数。
	LCnt	通道丢弃其他模块的发送的帧计数。
	SCnt	通道接收其他模块发送帧计数。如果该数值为 0，表示 vo 还没有收到图像数据；如果该数值一直不增加，表示 vo 接收不到图像了。
	ChRpt	通道重复显示总次数。如果通道暂停，该值会不断增加。
	DRpt	拼接图像重复显示计数。
	CBusy	通道 busy 队列节点数。
	DBusy	显示 busy 队列节点数。
	ShouD	当前显示的图像应该显示的次数。
	Dsped	当前显示的图像已经显示的次数。如果通道暂停，该值会不断增加。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	b2Scl	通道当前显示图像是否经过 2 级缩放。 Y：是 N：否
	ChnAddr	通道当前显示图像的物理地址。
	DispAddr	视频层显示画面的物理地址。
CHN OTHER INFO	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	ChnId	通道 ID。 取值范围：[0, VO_MAX_CHN_NUM）
	bZoom	是否局部放大。 n：否； y：是。
	ZmTyp	局部放大类型。选择通过区域还是比例放大。 取值范围：[0, VOU_ZOOM_IN_BUTT)
	ZoomX	局部放大区域起始横坐标。如果是通过比例方式设置局部放大区域，那么该值为按比例处理、对齐处理后的源图像坐标。
	ZoomY	局部放大区域起始纵坐标。如果是通过比例方式设置局部放大区域，那么该值为按比例处理、对齐处理后的源图像坐标。
	ZoomW	局部放大区域宽度。如果是通过比例方式设置局部放大区域，那么该值为按比例处理、对齐处理后的源图像坐标。
	ZoomH	局部放大区域高度。如果是通过比例方式设置局部放大区域，那么该值为按比例处理、对齐处理后的源图像坐标。
	SrcW	通道源图像宽度。
	SrcH	通道源图像高度。
	FltTp	通道滤波器类型。即 DSU 滤波器类型。 取值范围：[0, FILTER_PARAM_TYPE_BUTT)
	HFlt	通道水平滤波系数。即 DSU 水平系数。 取值范围：[0, DSU_HSCALE_FILTER_BUTT)
	VlFlt	通道垂直亮度滤波系数。即 DSU 垂直亮度系数。 取值范围：[0, DSU_VSCALE_FILTER_BUTT)

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	VcFlt	通道垂直色度滤波系数。即 DSU 垂直色度系数。 取值范围：[0, DSU_VSCALE_FILTER_BUTT)
Wbc INFO	DevId	设备 ID。 取值范围：仅支持 DHD1 设备视频回写。
	WbcW	视频回写目标图像宽度，最大 720。
	WbcH	视频回写目标图像高度，最大 576
	PixFmt	视频回写图像像素格式。 只支持 PIXEL_FORMAT_YUV_SEMIPLANAR_422 和 PIXEL_FORMAT_YUV_SEMIPLANAR_420 两种。
	FrmRat	视频回写目标图像帧率。
	RealRat	视频回写的平均帧率，每 10 秒统计一次。
	Mode	视频回写模式。
	DataSrc	视频回写源。 Mixer：回写视频层和图像层 Mixer 之后的图像。 Video：只回写视频层。
	Depth	视频回写深度。
	NotFin	视频回写未完成帧计数。
GRAPHICS LAYER	Layer	图形层。 取值：G4，HC0，HC1
	BindDev	图形层绑定的设备号。
LAYER CSC PARAM	LAYERID	与设备绑定的图形层号。 取值范围：[0, VOU_GRAPHICS_LAYER_NUM）
	Matrix	VGA CSC 转换矩阵选择。 取值范围：[5, 6]
	Luma	亮度。 取值范围：[0, 100]，默认值 50。
	Cont	对比度。 取值范围：[0, 100]，默认值 50。
	Hue	色调。 取值范围：[0, 100]，默认值 50。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Satu	饱和度。 取值范围：[0, 100]，默认值 50。
VGA PARAM	DevId	设备 ID。 取值范围：[0, VO_MAX_DEV_NUM）
	Matrix	VGA CSC 转换矩阵选择。 取值范围：[3, 4]
	Luma	亮度。 取值范围：[0, 100]，默认值 50。
	Cont	对比度。 取值范围：[0, 100]，默认值 50。
	Hue	色调。 取值范围：[0, 100]，默认值 50。
	Satu	饱和度。 取值范围：[0, 100]，默认值 50。
	Gain	DAC 增益。 取值范围：[0, 63]，默认值 48。

13.16 VPSS

【调试信息】

```
# cat /proc/umap/vpss

[VPSS] Version: [Hi3520D_MPP_V0.0.0.0 Debug], Build Time[Feb 18 2013, 16:46:55]
-------------------------VPSS GRP ATTR-----------------------------------
   GrpID    MaxW    MaxH  PixFmt DieMode    DrEn    DbEn    IeEn    NrEn  HistEn     Feild
      0     720     576 semi422       0       0       0       1       1       0    BOTH

-----------------------VPSS GRP FRAME CONTROL INFO-------------------------
   GrpID bFilter   Width  Height    SrcFRate    DstFRate
      0       0       0       0          -1          -1

-------------------------VPSS GRP PARAM-----------------------------------
   GrpID     Lum    Cont    Dark  Bright   IeStr IeSharp   SfStr   TfStr  Motion   DiStr
ChrmRg     NrW   SfWin DisMode
      0       0      64       0       0       4       0      24       8       0       0       0
0       0       0

-------------------------VPSS CHN PARAM-----------------------------------
   GrpID   ChnID   ChnSp   ChnSf   ChnTf
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
      0      0     64      0      0
      0      1     64      0      0
      0      2     64      0      0
      0      3     64      0      0

----------------------VPSS GRP PRESCALE INFO-------------------------
  GrpID bPreScl  CapSel   Width  Height   totde
     0    OFF    BOTH       0       0       0

-------------------------VPSS CHN ATTR-----------------------------------
  GrpId  PhyChnId  Enable    SpEn FrmWkEn     LW      RW      TW      BW
     0       0       1       1       1       0       2       6      14
     0       1       1       1       1       0       2       6      14
     0       2       1       1       1       0       2       6      14
     0       3       1       1       1       0       2       6      14
     0       4       1       -       -       -       -       -       -

-------------------------VPSS CROP INFO-----------------------------------
  GrpId  CropEn  CoorTp   CoorX   CoorY   Width  Height  CapSel    OriW    OriH TrimWid
TrimHgt
     0       1     ABS       0       0     640     480    BOTH     704     576     640     480

-------------------------VPSS GRP PIC QUEUE-------------------------
  GrpID FreeLen BusyLen   Delay   Backup
     0       6       0       0         1

----------------------VPSS GRP WORK STATUS-------------------------
  GrpID        RecvPic  ViLost    VdecLost          NewDo   OldDo NewUnDo        OldUnDo
NoHist StartFl  bStart  CostTm MaxCostTm
     0           463       0          0           463       0       0          1386       0
0       1   15482    15807

----------------------VPSS CHN WORK STATUS-------------------------
  GrpID   ChnID    WorkMode   Depth         SendOk  bConfident    bDouble     CapSel
     0       0       AUTO       0           461          1         --       BOTH
     0       1       AUTO       0           461          1         --       BOTH
     0       2       AUTO       0           461          1          0         --
     0       3       AUTO       0           461          1         --         --

----------------------VPSS CHN OUTPUT RESOLUTION-----------------------
  GrpID   ChnID  Enable   Width  Height  pixfmt
     0       0       1     640     480      19
     0       1       1     352     288      19
     0       2       1     640     360      19
     0       3       1     640     360      19


----------------------TIMER WORK STATUS-------------------------
    CntPerSec  MaxCntPerSec       CostTm    MostCostTm  CostTmPerSec MCostTmPerSec
           75           76           75         1916        20821        43525

-------------------------DRV WORK STATUS-----------------------------------
      StartSuc0   StartSuc1        LinkInt     NodeInt   StartErr0  NodeIdErr0   StartErr1
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
NodeIdErr1      BusErr
           463          0           463          0          0          0          0
0          0

-------------------------DRV NODE QUEUE-----------------------------------
 FreeNum WaitNum  Busy00  Busy01    Sel0  Busy10  Busy11    Sel1
     128       0       0       0       1       0       0       0

-----------------------INT WORK STATUS-------------------------
     CntPerSec  MaxCntPerSec       CostTm    MostCostTm  CostTmPerSec MCostTmPerSec
           25           26          482          552         12596        13028
```

【调试信息分析】

记录当前 VPSS 属性配置以及状态信息。

【参数说明】

参数		描述
VPSS GRP ATTR VPSS GRP 属性，主要包含用户配置的最大处理能力及各功能模块使能信息。	GrpID	GRP ID 号。 有效范围[0,127]。
	MaxW	最大处理宽度。 有效范围[64, 4096]。
	MaxH	最大处理高度。 有效范围[64, 4080]。
	PixFmt	像素格式：支持 semi420 和 semi422 两种格式。
	DieMode	De-interlace 模式。 0：自适应； 1：强制不做 DIE； 2：强制做 DIE。
	DrEn	是否使能去环（De-ring）功能。 0：关闭； 1：打开。
	DbEn	是否使能去块(De-block)功能。 0：关闭； 1：打开。
	IeEn	是否使能图像增强功能。 0：关闭； 1：打开。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	NrEn	是否使能去噪功能。 0：关闭； 1：打开。
	HistEn	是否使能直方图输出。 0：关闭； 1：打开。
	Feild	GRP 抽场信息。 BOTH：两场； BOTTOM：抽取底场； TOP：抽取顶场。
VPSS GRP FRAME CONTROL INFO VPSS 帧控制信息	GrpID	GRP ID 号。 有效范围[0,127]。
	bSizer	是否开启尺寸筛选。
	Width	筛选尺寸宽。
	Height	筛选尺寸高。
	SrcFRate	源帧率。
	DstFRate	目标帧率
VPSS GRP PARAM VPSS 组高级属性	GrpID	GRP ID 号。 有效范围：[0, 127]。
	Lum	亮度。
	Cont	对比度。
	Dark	暗区增强。
	Bright	亮区增强。
	IeStr	IE 强度。
	IeSharp	IE 锐度。
	SfStr	空域去噪强度。
	TfStr	时域去噪强度。
	Motion	移动判决阈值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	DiStr	De-interlace 强度。
	ChrmRg	色度幅度。
	NrW	时域去噪比重。
	SfWin	空域滤波控制窗大小。
	DisMode	显示模式。
VPSS CHN PARAM VPSS 通道高级属性	GrpID	GRP ID 号。 有效范围[0, 127]。
	ChnID	通道 ID 号。 有效范围[0,4]。
	ChnSp	通道 SP 强度。
	ChnSf	通道空域去噪强度。
	ChnTf	通道时域去噪强度。
VPSS GRP PRESCALE INFO VPSS 预缩放信息	GrpID	GRP ID 号。 有效范围：[0, 127]
	bPreScl	是否使能预缩放。 0：关闭； 1：打开。
	CapSel	抽场类型。 TOP：抽顶场； BOTTOM：抽底场 BOTH:不抽场
	Width	预缩放目标图像宽
	Height	预缩放目标图像高
	totde	TDE 正在处理 VPSS 提交的图像数目。
VPSS CHN ATTR VPSS 各物理通道属性	GrpId	GRP ID 号。 有效范围：[0, 127]
	PhyChnID	物理通道 ID 号。 有效范围：[0, 4]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Enable	是否使能该通道。 0：关闭； 1：打开。
	SpEn	是否使能锐化功能。 0：关闭； 1：打开。
	FrmWkEn	是否使能边框功能。 0：关闭； 1：打开。
	LW	左边框宽度。 合法取值范围为[0, 14]，并且只能取偶数值。
	RW	右边框宽度。 合法取值范围为[0, 14]，并且只能取偶数值。
	TW	上边框宽度。 合法取值范围为[0, 14]，并且只能取偶数值。
	BW	下边框宽度。 合法取值范围为[0, 14]，并且只能取偶数值。
VPSS CROP INFO VPSS CROP 操作信息	GrpId	GRP ID 号。 有效范围：[0, 127]。
	CropEn	是否使能 CROP 功能。 0：关闭； 1：打开。
	CoorTp	坐标类型。 RIT：相对坐标； ABS：绝对坐标
	CoorX	水平方向起始坐标。 坐标类型为相对坐标时，合法取值范围为[0，999]； 坐标类型为绝对坐标时，合法取值范围为[0, MAX_WIDTH]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	CoorY	垂直方向起始坐标。 坐标类型为相对坐标时，合法取值范围为[0, 999]； 坐标类型为绝对坐标时，合法取值范围为[0, MAX_HEIGHT]。
	Width	CROP RECT 的宽，不能超出最大图像宽度。
	Height	CROP RECT 的高，不能超出最大图像高度。
	CapSel	抽场类型。 TOP：抽顶场； BOTTOM：抽底场； BOTH：不抽场。
	OriWth	原始图像宽。
	OriHgt	原始图像高。
	TrimWid	实际图像宽。
	TrimHgt	实际图像高。
VPSS GRP PIC QUEUE VPSS GRP 缓存图像队列状态信息。	GrpID	GRP ID 号。 有效范围：[0, 127]。
	FreeLen	可用节点数。
	BusyLen	已用节点数。
	Delay	Delay 队列长度。
	Backup	是否开启 Backup 帧。 0：关闭； 1：开启。
VPSS GRP WORK STATUS VPSS GRP 工作状态信息	GrpID	GRP ID 号。 有效范围：[0,127]。
	RecvPic	VPSS 接收到的图像数
	ViLost	因队列满而丢弃的 VI 图像数。
	VdecLost	因队列满而丢弃的 VDEC 图像数。
	NewDo	NewDo 次数。
	OldDo	OldDo 次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	NewUnDo	NewUnDo 次数。
	OldUnDo	OldUnDo 次数。
	NoHist	申请不到直方图 buffer 次数。
	StartFl	Start 任务失败次数。
	bStart	是否开始接收图像。
	CostTm	当前完成任务的耗时。
	MaxCostTm	历史上耗时最长任务的执行时间。
VPSS CHN WORK STATUS VPSS 通道运行状态信息	GrpID	GRP ID 号。 有效范围：[0, 127]。
	ChnID	通道 ID 号。 有效范围：[0, 4]。
	SendOk	VPSS 成功发送图像次数。
	WorkMode	通道工作模式。
	Depth	用户队列深度。
	bConfident	后端图像处理要求是否跟通道能力相符。
	bDouble	通道是否正在使用帧场转换功能。
	CapSel	抽场类型。 TOP：抽顶场； BOTTOM：抽底场 BOTH:不抽场
VPSS CHN OUTPUT RESOLUTION VPSS 通道输出分辨率	GrpID	GRP ID 号。 有效范围：[0, 127]。
	ChnID	通道 ID 号。 有效范围：[0, 4]。
	Enable	通道使能。 0：关闭； 1：打开。
	Width	目标图像的宽度，以像素为单位。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Height	目标图像的高度，以像素为单位。
	pixfmt	目标图像的像素格式。
TIMER WORK STATUS 定时器工作状态信息	CntPerSec	最近一次的 1 秒内定时器执行次数。
	MaxCntPerSec	历史上的 1 秒内最大的定时器执行次数。
	CostTm	最近一次执行定时器的执行耗时。
	MostCostTm	执行一次定时器的最大耗时。
	CostTmPerSec	最近一秒执行定时器的执行耗时。
	MCostTmPerSec	历史上一秒执行定时器的最大执行耗时。
DRV WORK STATUS VPSS 硬件工作状态信息	StartSuc0	启动 VPSS0 成功次数。
	StartSuc1	启动 VPSS1 成功次数。
	LinkInt	链中断次数。
	NodeInt	节点中断次数。
	StartErr0	启动 VPSS0 失败次数。
	NodeIdErr0	读取 VPSS0 NodeId 错误次数。
	StartErr1	启动 VPSS1 失败次数。
	NodeIdErr1	读取 VPSS1 NodeId 错误次数。
	BusErr	Bus Error 中断次数。
DRV NODE QUEUE VPSS 硬件节点队列状态信息	FreeNum	Free 队列上可用节点个数。
	WaitNum	Wait 队列上待处理节点个数。
	Busy00	VPSS0 的 busy0 队列上的节点个数。
	Busy01	VPSS0 的 busy1 队列上的节点个数。
	Sel0	VPSS0 用于工作的 busy 队列 id 号。
	Busy10	VPSS1 的 busy0 队列上的节点个数。
	Busy11	VPSS1 的 busy1 队列上的节点个数。
	Sel1	VPSS1 用于工作的 busy 队列 id 号。
INT WORK STATUS	CntPerSec	最近一次的 1 秒内中断执行次数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
VPSS 中断工作状态信息	MaxCntPerSec	历史上的 1 秒内最大的中断执行次数。
	CostTm	最近一次执行中断的执行耗时。
	MostCostTm	执行一次中断的最大耗时。
	CostTmPerSec	最近一秒执行中断的执行耗时。
	MCostTmPerSec	历史上一秒执行中断的最大执行耗时。

13.17 IVE

【调试信息】

```
# cat /proc/umap/ive

[IVE] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Sep 16 2011, 11:38:03]

-----IVE CONTEXT ATTR-----------------------------------------------------------
 W  B WCI WEI BCI BEI     HndU     TskF LstId TskId bInt HWrap FWrap    IntCnt
 1  0    0   0   0  309     4060     3751     0    197    0     0     0       29
```

【调试信息分析】

记录当前 IVE 工作状态资源信息。

【参数说明】

参数		描述
IVE CONTEXT ATTR IVE 工作状态	W	等待队列编号（0 或 1）。
	B	正在调度队列编号（0，1 或-1），-1 表示 IVE 硬件空闲。
	WCI	等待队列的首个有效任务 ID。
	WEI	等待队列的末尾有效任务 ID + 1。
	BCI	调度队列的首个有效任务 ID。
	BEI	调度队列的末尾有效任务 ID + 1。
	HndU	当前可分配的任务 handle 号。
	TskF	当前已完成任务的个数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	LstId	上一次中断完成的任务 ID。
	TskId	本次中断完成的任务 ID。
	bInt	用户最后一次提交任务时传入的 bInstant 值。
	HWrap	用户 handle 号分配发生回写的次数。
	FWrap	完成任务数发生回写的次数。
	IntCnt	IVE 产生中断的次数。

13.18 VDA

【调试信息】

```
# cat /proc/umap/vda

[VDA] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[May 29 2012, 09:29:27]

-----VDA CHN ATTR-----------------------------------------------------------
NO.    W    H  Mode   Alg  MbSz MbBit RfM BufN  Itl BgWt RgnN
  0  720  576     0     1     1     0   0    8    6  128    1
  1  720  576     1     1     0     0   0    0    6  128    1

-----VDA CHN ATTR 2---------------------------------------------------------
 NO. Rgn    X    Y    W    H SadTh  ArTh  OcTh  UnOcTh  ObjN  Sad  Obj  Plx
   0   0    0    0  720  576   100     0     0       0   128    1    1    1
   1   0    0    0  720  576   100    60     6       2   128    0    0    0

-----VDA STATE--------------------------------------------------------------
 NO. Rgn    BgPhy BgSrd    RfPhy   RfStd bScdu bFst  OcCt  UnOc      VdaC    bRefUp
   0   0        0     0        0        0      1     0     0     0        36       0
   1   0        0     0        0        0      1     0     0     0        36       0

-----VDA STATE2-------------------------------------------------------------
 NO.   MemPhy   MemVir MemSz  CRfPhy RgnIx  ViSd VdecSd  VpssSd  UserSd   Recv CalDsuC
LstPicC
0 868ec000 c4918000 22944 862f4000    0    250      0       0        0     36       0
0
1 868f2000 c48fa000   352  862f4000    0    250      0       0        0     36       0
0
```

【调试信息分析】

记录当前 VDA 属性配置以及状态信息。

【参数说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
VDA CHN ATTR VDA 通道属性	NO.	通道号。
	W	通道宽度（单位：像素）。
	H	通道高度（单位：像素）。
	Mode	通道模式。 0：MD； 1：OD。
	Alg	工作算法。 0：背景法； 1：帧差法。
	MbSz	宏块大小。 0：8x8； 1：16x16。
	MbBit	SAD 输出 bit 位。 0：8bits； 1：16bits。
	RfM	参考帧模式。 0：动态； 1：静态； 2：用户。
	BufN	MD 结果缓存个数。
	Itl	VDA 间隔，单位帧。
	BgWt	背景法更新权重。
	RgnN	该通道包含的区域个数。OD 可设置多个区域。
VDA CHN ATTR 2 VDA 通道属性之区域属性 （MD 只有一个区域，OD 包含多个区域）	NO.	通道号。
	Rgn	区域编号。
	X	区域相对通道的起始 X 坐标。
	Y	区域相对通道的起始 Y 坐标。
	W	区域宽度（单位：像素）。
	H	区域高度（单位：像素）。
	SadTh	区域 SAD 报警阈值。
	ArTh	区域报警面积阈值。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	OcTh	区域遮挡次数报警阈值。
	UnOcTh	区域遮挡计数时，允许不发生遮挡的次数阈值。
	ObjN	运动区域（运动局部）输出的最大个数。
	Sad	是否输出宏块 SAD 值。 0：不输出； 1：输出。
	Obj	是否输出运动区域。 0：不输出； 1：输出。
	Plx	是否输出报警像素个数。 0：不输出； 1：输出。
VDA STATE VDA 通道基本状态	NO.	通道号。
	Rgn	区域编号。
	BgPhy	背景图像地址。
	BgSrd	背景图像跨度。
	RfPhy	参考帧图像地址。
	RfStd	参考帧跨度。
	bScdu	是否在调度。
	bFst	是否是第一次调度。
	OcCt	区域遮挡计数。
	UnOc	区域遮挡计数时，允许不发生遮挡的次数计数。
	VdaC	VDA 完成计数。
	bRefUp	用户参考帧模式下是否手动更新过参考帧。
VDA STATE2 VDA 通道的基本状态（续）	NO.	通道号。
	MemPhy	通道占用内存的起始物理地址。
	MemVir	通道占用内存的起始虚拟地址。
	MemSz	通道占用的内存大小。
	CRfPhy	公共参考帧的物理地址。
	RgnIx	当前调度的区域。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	ViSd	VI 发送的图像计数。
	VdecSd	Vdec 发送图像计数。
	VpssSd	VPSS 发送图像计数。
	UserSd	USER 发送图像计数。
	Recv	VDA 通道接收的图像计数。
	CalDsuC	调 dsu 的次数。
	LstPicC	VDA 乱序导致丢掉的图像计数。

13.19 AI

AI 的调试信息与芯片使用的音频接口有关，根据芯片使用的音频接口，AI 的调试信息分为以下两类：

【使用 SIO 接口的调试信息】

```
# cat /proc/umap/ai

[AI] Version: [Hi3518_MPP_V1.0.5.0 Debug], Build Time: [Mar 15 2013, 13:07:47]

-----AI DEV ATTR-----------------------------------------------------------------------
---------------------------------
AiDev WorkMod   SampR  BitWid  ChnCnt  ClkSel SondMod  PoiNum  ExFlag  FrmNum
    0 i2s_mas    8kHZ   16bit       2       1    mono     320       1      30

-----AI DEV STATUS---------------------------------------------------------------------
---------------------------------
AiDev      IntCnt FrmTime  MaxFrmTime  DMAChn  DMAReq TranLen IsrTime  MaxIsrTime   DMAPhy0
DMAPhy1 CurDMAPhy PinPon    ErrCnt
    0        146   39999       40021       0       0    2560     264        415  84ff9000
84ff9a00  84ff9000      1         0

-----AI CHN STATUS---------------------------------------------------------------------
---------------------------------
AiDev   AiChn   State    Read   Write     BufFul u32Data0 u32Data1   AecAo     AecFail
bAnr  bResmp  PoiNum   SampR ResmpType
    0       0  enable       0       0          0   df0187  1cd01f7 (-1,-1)          0 disable
disable       0  (null)    (null)
0       1  enable       0       0          0   6300c7   4c0051 (-1,-1)          0 disable
disable       0  (null)    (null)
-----VQE & ResampleEx-----------------------------------------------------------------
AiDev   AiChn    bVqe    bAlc  MinLev  MaxLev MaxGain    bAec AecMode    bAnr   bHpf  bReSp
SamIn  SamOut  WrFile
0       0  enable  enable     -16      -4      12  enable   speaker  enable disable  disable
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
(null)  (null) disable
```

【调试信息分析】

记录当前音频输入设备及通道的属性配置以及状态信息。

【参数说明】

参数		描述
AI DEV ATTR（音频输入设备属性）（参考数据结构 AIO_ATTR_S）	AiDev	AI 设备号。
	WorkMod	SIO 工作模式。 i2s_mas：I2S 主模式。 i2s_sla：I2S 从模式。 pcm0_mt：标准 PCM 主模式。 pcm0_sl：标准 PCM 从模式。 pcm1_mt：非标准 PCM 主模式。 pcm1_sl：非标准 PCM 从模式。
	SampR	采样率。 取值：{8k~48k}。
	BitWid	采样精度。 取值：{8bit, 16bit}。
	ChnCnt	最大通道个数。 取值：{2, 4, 8, 16}。
	ClkSel	时钟选择。
	SondMod	声音模式。 mono：单声道； stereo：立体声。
	PoiNum	每帧的采样点个数。
	ExFlag	8bit 扩展标志。
	FrmNum	帧缓存数目。
AI DEV STATUS（音频输入设备信息）	AiDev	AI 设备号。
	IntCnt	中断计数。 完成 1 帧音频数据采集后，此值加 1。 如果此值未递增，说明与外围 codec 对接有问题。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	FrmTime	帧间隔时间。以 us 为单位。 由此项可以计算出实际的音频采样率。例如当此项为 40000us 左右、每帧采样点个数配置为 320 时，可以计算出每秒采样 8000 点，即采样率为 8kHZ。
	MaxFrmTime	帧间隔最长时间。 此项统计系统从开始到当前为止最长的帧间隔时间。
	DMAChn	DMA 通道。
	DMAReq	DMA 请求线。
	TranLen	一次 DMA 传输的长度（字节）
	IsrTime	DMA 中断处理时间
	MaxIsrTime	DMA 中断处理最长时间。 此项统计系统从开始到当前为止最长的 DMA 中断处理时间。
	DMAPhy0	DMA buffer0 的物理地址。
	DMAPhy1	DMA buffer1 的物理地址。
	CurDMAPhy	当前 DMA 正在使用的物理地址。
	PinPon	当前软件正在使用的 PinPon 内存。
	ErrCnt	PingPong 内存出错的次数。
AI CHN STATUS（音频输入通道信息）	AiDev	AI 设备号。
	AiChn	AI 通道号。
	State	通道状态。 orig：初始状态； enable：启用； disable：禁用。
	Read	通道 buffer 的读指针。
	Write	通道 buffer 的写指针。
	BufFul	帧 buffer 满的次数。
	u32Data0	通道 Buffer 中第一个 32 位数据。
	u32Data1	通道 Buffer 中第二个 32 位数据。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	AecAo	回声抵消 AEC 的 AO 设备号和通道号。（-1 表示未启用 AEC）
	AecFail	回声抵消失败次数。
	bAnr	该通道是否启用了降噪功能 enable：启用降噪功能； disable：关闭降噪功能。
	bHpf	该通道是否启用了高通滤波功能。 enable：启用高通滤波功能； disable：关闭高通滤波功能。
	bResmp	该通道是否启用了重采样功能 enable：启用重采样功能； disable：关闭重采样功能。
	PoiNum	该通道重采样时，输入给重采样的数据帧中包含的采样点数目。
	SampR	该通道重采样时，输入给重采样的数据帧的采样率。
	ResmpType	该通道的重采样类型。
VQE & ResampleEx（VQE 和高级重采样信息）	AiDev	AI 设备号。
	AiChn	AI 通道号。
	bVqe	Vqe 是否启用。 enable：启用； disable：禁用。
	bAlc	Alc 是否启用。 enable：启用； disable：禁用。
	MinLev	Alc 最低目标电平。
	MaxLev	Alc 最高目标电平。
	MaxGain	Alc 最大增益值。
	bAec	Aec 是否启用。 enable：启用； disable：禁用。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	AecMode	Aec 模式。 receiver：听筒模式； speaker：扬声器模式； headphone：耳机模式
	bAnr	Anr 是否启用。 enable：启用； disable：禁用。
	bResmp	是否启用了高级重采样功能 enable：启用重采样功能； disable：关闭重采样功能。
	SamIn	输入采样率
	SamOut	经过重采样后，输出的采样率
	WrFile	是否启用保存文件功能。 enable：启用； disable：禁用。

【使用 AIO 接口的调试信息】

```
# cat /proc/umap/ai

[AI] Version: [Hi3520D_MPP_V0.0.0.0 Debug], Build Time: [Feb 27 2013, 14:01:00]

-----AI DEV ATTR-----------------------------------------------------------------------------
---------------------------------
AiDev WorkMod   SampR  BitWid  ChnCnt  ClkSel SondMod  PoiNum  ExFlag  FrmNum
    0 i2s_sla    8kHZ   16bit       2       0    mono     320       1      30

-----AI DEV STATUS---------------------------------------------------------------------------
---------------------------------
AiDev      IntCnt     fifoCnt     buffInt FrmTime  MaxFrmTime TranLen IsrTime  MaxIsrTime
CBPhy  CBSize   ROffSet   WOffSet
    0       1124          0          0   40026      40081    1280    1628       1733
84e72000    2560        0        0

-----AI CHN STATUS---------------------------------------------------------------------------
---------------------------------
AiDev   AiChn   State    Read   Write     BufFul u32Data0 u32Data1   AecAo     AecFail
bAnr  bResmp  PoiNum   SampR RespType
    0       0  enable       0       0          0 f376f37d f363f368 (-1,-1)          0 disable
disable       0  (null)    (null)
0       1  enable       0       0          0 f46af464 f469f471 (-1,-1)          0 disable
disable       0  (null)    (null)
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

【调试信息分析】

记录当前音频输入设备及通道的属性配置以及状态信息。

【参数说明】

参数		描述
AI DEV ATTR（音频输入设备属性）（参考数据结构 AIO_ATTR_S）	AiDev	AI 设备号。
	WorkMod	AI 工作模式。 i2s_mas：I2S 主模式。 i2s_sla：I2S 从模式。 pcm0_mt：标准 PCM 主模式。 pcm0_sl：标准 PCM 从模式。 pcm1_mt：非标准 PCM 主模式。 pcm1_sl：非标准 PCM 从模式。
	SampR	采样率。 取值：{8k~48k}。
	BitWid	采样精度。 取值：{8bit, 16bit}。
	ChnCnt	最大通道个数。 取值：{2, 4, 8, 16}。
	ClkSel	时钟选择。
	SondMod	声音模式。 mono：单声道； stereo：立体声。
	PoiNum	每帧的采样点个数。
	ExFlag	8bit 扩展标志。
	FrmNum	帧缓存数目。
AI DEV STATUS（音频输入设备信息）	AiDev	AI 设备号。
	IntCnt	中断计数。 完成 1 帧音频数据采集后，此值加 1。 如果此值未递增，说明与外围 codec 对接有问题。
	fifoCnt	Fifo 上溢中断计数。 AI 接收 fifo 满，发生溢出时，此值加 1。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	buffInt	DMA buffer 满中断计数。 AI 接收数据的 DMA buffer 满时，此值加 1。
	FrmTime	帧间隔时间。以 us 为单位。 由此项可以计算出实际的音频采样率。例如当此项为 40000us 左右、每帧采样点个数配置为 320 时，可以计算出每秒采样 8000 点，即采样率为 8kHZ。
	MaxFrmTime	帧间隔最长时间。 此项统计系统从开始到当前为止最长的帧间隔时间。
	TranLen	一次 DMA 传输的长度（字节）
	IsrTime	DMA 中断处理时间
	MaxIsrTime	DMA 中断处理最长时间。 此项统计系统从开始到当前为止最长的 DMA 中断处理时间。
	CBPhy	DMA buffer 的物理地址。
	CBSize	DMA buffer 的大小。
	ROffSet	DMA buffer 的读指针相对于起始地址的偏移量。
	WOffSet	DMA buffer 的写指针相对于起始地址的偏移量。
AI CHN STATUS（音频输入通道信息）	AiDev	AI 设备号。
	AiChn	AI 通道号。
	State	通道状态。 orig：初始状态； enable：启用； disable：禁用。
	Read	通道 buffer 的读指针。
	Write	通道 buffer 的写指针。
	BufFul	帧 buffer 满的次数。
	u32Data0	通道 Buffer 中第一个 32 位数据。
	u32Data1	通道 Buffer 中第二个 32 位数据。
	AecAo	回声抵消 AEC 的 AO 设备号和通道号。（-1 表示未启用 AEC）

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	AecFail	回声抵消失败次数。
	bAnr	该通道是否启用了降噪功能 enable：启用降噪功能； disable：关闭降噪功能。
	bResmp	该通道是否启用了重采样功能 enable：启用重采样功能； disable：关闭重采样功能。
	PoiNum	该通道重采样时，输入给重采样的数据帧中包含的采样点数目。
	SampR	该通道重采样时，输入给重采样的数据帧的采样率。
	ResmpType	该通道的重采样类型。

13.20 AO

AO 的调试信息与芯片使用的音频接口有关，根据芯片使用的音频接口，AO 的调试信息分为以下两类：

【使用 SIO 接口的调试信息】

```
# cat /proc/umap/ao

[AO] Version: [Hi3518_MPP_V1.0.5.0 Debug], Build Time: [Mar 15 2013, 13:07:47]

-----AO DEV ATTR-----------------------------------------------------------------------
AoDev WorkMod   SampR  BitWid  ChnCnt  ClkSel SondMod  PoiNum  ExFlag  FrmNum
    0 i2s_mas    8kHZ   16bit       2       1    mono     320       1     30

-----AO DEV STATUS--------------------------------------------------------------------
AoDev      IntCnt FrmTime  MaxFrmTime  DMAChn  DMAReq TranLen IsrTime  MaxIsrTime   DMAPhy0
DMAPhy1 CurDMAPhy PinPon    ErrCnt
    0         89   40015       40019       1       1    2560    161         161  85021000
85021a00  85021a10      0         0
-----AO CHN STATUS--------------------------------------------------------------------
AoDev   AoChn   State    Read   Write     BufEmp  u32Data0  u32Data1  bResmp  PoiNum   SampR
ResmpType   Volume
0       0  enable      29      29          0    f00148     60120 disable       0  (null)
(null)        0
0       1  enable       0       0         89         0         0 disable       0  (null)    (null)

-----AO ResampleEx STATUS-------------------------------------------------------------
AoDev   AoChn   bReSp   SamIn  SamOut
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
0       0 disable  (null)  (null)
```

【调试信息分析】

记录当前音频输出设备属性配置以及状态信息。

【参数说明】

参数		描述
AO DEV ATTR（音频输出设备属性）（参考数据结构 AIO_ATTR_S）	AoDev	AO 设备号。
	WorkMod	SIO 工作模式。 i2s_mas：I2S 主模式。 i2s_sla：I2S 从模式。 pcm0_mt：标准 PCM 主模式。 pcm0_sl：标准 PCM 从模式。 pcm1_mt：非标准 PCM 主模式。 pcm1_sl：非标准 PCM 从模式。
	SampR	采样率。 取值：{8k~48k}。
	BitWid	采样精度。 取值：{8bit, 16bit}。
	ChnCnt	最大通道个数。 取值：{2, 4, 8, 16}。
	ClkSel	时钟选择。
	SondMod	声音模式。 mono：单声道； stereo：立体声。
	PoiNum	每帧的采样点个数。
	ExFlag	8bit 扩展标志。
	FrmNum	帧缓存数目。
AO DEV STATUS（音频输出设备信息）	AoDev	AO 设备号。
	IntCnt	中断计数。 发送 1 帧音频数据后，此值加 1。 如果此值未递增，说明与外围 codec 对接有问题。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	FrmTime	帧间隔时间。以 us 为单位。 由此项可以计算出实际的音频采样率。例如当此项为 40000us 左右、每帧采样点个数配置为 320 时，可以计算出每秒采样 8000 点，即采样率为 8kHZ。
	MaxFrmTime	帧间隔最长时间 此项统计系统从开始到当前为止最长的帧间隔时间。
	DMAChn	DMA 通道。
	DMAReq	DMA 请求线。
	TranLen	一次 DMA 传输的长度（字节）
	IsrTime	DMA 中断处理时间。
	MaxIsrTime	DMA 中断处理最长时间。 此项统计系统从开始到当前为止最长的 DMA 中断处理时间。
	DMAPhy0	DMA Buffer0 的物理地址。
	DMAPhy1	DMA Buffer1 的物理地址。
	CurDMAPhy	当前 DMA 正在使用的物理地址。
	PinPon	当前软件正在使用的 PinPon 内存。
	ErrCnt	PingPong 内存出错的次数。
AO CHN STATUS（音频输出通道信息）	AoDev	AO 设备号。
	AoChn	AO 通道号。
	State	通道状态。 orig：初始状态； enable：启用； disable：禁用。
	Read	通道 buffer 的读指针。
	Write	通道 buffer 的写指针。
	BufEmp	帧 buffer 空的次数。
	u32Data0	通道 Buffer 中第一个 32 位数据。
	u32Data1	通道 Buffer 中第二个 32 位数据。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	bResmp	该通道是否启用了重采样功能 enable：启用重采样功能； disable：关闭重采样功能。
	PoiNum	该通道重采样时，输入给重采样的数据帧中包含的采样点数目。
	SampR	该通道重采样时，输入给重采样的数据帧的采样率。
	ResmpType	该通道的重采样类型。
	Volume	AO 设备的音量大小。
AO ResampleEx STATUS（AO 高级重采样信息）	AoDev	AO 设备号。
	AoChn	AO 通道号。
	bReSp	该通道是否启用了高级重采样功能 enable：启用重采样功能； disable：关闭重采样功能。
	SamIn	输入采样率
	SamOut	经过重采样后的输出采样率

【使用 AIO 接口的调试信息】

```
# cat /proc/umap/ao

[AO] Version: [Hi3520D_MPP_V0.0.0.0 Debug], Build Time: [Feb 27 2013, 14:01:00]

-----AO DEV ATTR-----------------------------------------------------------------------
AoDev WorkMod   SampR  BitWid  ChnCnt  ClkSel SondMod  PoiNum  ExFlag  FrmNum
    0 i2s_sla    8kHZ   16bit       2       0    mono     320       1      30

-----AO DEV STATUS---------------------------------------------------------------------
AoDev      IntCnt     fifoCnt     buffInt FrmTime  MaxFrmTime TranLen IsrTime  MaxIsrTime
CBPhy  CBSize   ROffSet   WOffSet
    0       1192          0          0   40053      40053    1280     738        740
84e99000    3840        0        0

-----AO CHN STATUS---------------------------------------------------------------------
AoDev   AoChn   State    Read   Write     BufEmp  u32Data0  u32Data1  bResmp  PoiNum   SampR
ResmpType
0       0  enable     22     22          0  f380f387  f388f37c disable       0  (null)
(null)
```

【调试信息分析】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

记录当前音频输出设备属性配置以及状态信息。

【参数说明】

参数		描述
AO DEV ATTR（音频输出设备属性）（参考数据结构 AIO_ATTR_S）	AoDev	AO 设备号。
	WorkMod	AO 工作模式。 i2s_mas：I2S 主模式。 i2s_sla：I2S 从模式。 pcm0_mt：标准 PCM 主模式。 pcm0_sl：标准 PCM 从模式。 pcm1_mt：非标准 PCM 主模式。 pcm1_sl：非标准 PCM 从模式。
	SampR	采样率。 取值：{8k~48k}。
	BitWid	采样精度。 取值：{8bit, 16bit}。
	ChnCnt	最大通道个数。 取值：{2, 4, 8, 16}。
	ClkSel	时钟选择。
	SondMod	声音模式。 mono：单声道； stereo：立体声。
	PoiNum	每帧的采样点个数。
	ExFlag	8bit 扩展标志。
	FrmNum	帧缓存数目。
AO DEV STATUS（音频输出设备信息）	AoDev	AO 设备号。
	IntCnt	中断计数。 发送 1 帧音频数据后，此值加 1。 如果此值未递增，说明与外围 codec 对接有问题。
	fifoCnt	Fifo 下溢中断计数。 AO 发送 fifo 空，发生溢出时，此值加 1。
	buffInt	DMA buffer 空中断计数。 AO 发送数据的 DMA buffer 空时，此值加 1。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	FrmTime	帧间隔时间。以 us 为单位。 由此项可以计算出实际的音频采样率。例如当此项为 40000us 左右、每帧采样点个数配置为 320 时，可以计算出每秒采样 8000 点，即采样率为 8kHZ。
	MaxFrmTime	帧间隔最长时间 此项统计系统从开始到当前为止最长的帧间隔时间。
	TranLen	一次 DMA 传输的长度（字节）
	IsrTime	DMA 中断处理时间。
	MaxIsrTime	DMA 中断处理最长时间。 此项统计系统从开始到当前为止最长的 DMA 中断处理时间。
	CBPhy	DMA buffer 的物理地址。
	CBSize	DMA buffer 的大小。
	ROffSet	DMA buffer 的读指针相对于起始地址的偏移量。
	WOffSet	DMA buffer 的写指针相对于起始地址的偏移量。
AO CHN STATUS（音频输出通道信息）	AoDev	AO 设备号。
	AoChn	AO 通道号。
	State	通道状态。 orig：初始状态； enable：启用； disable：禁用。
	Read	通道 buffer 的读指针。
	Write	通道 buffer 的写指针。
	BufEmp	帧 buffer 空的次数。
	u32Data0	通道 Buffer 中第一个 32 位数据。
	u32Data1	通道 Buffer 中第二个 32 位数据。
	bResmp	该通道是否启用了重采样功能 enable：启用重采样功能； disable：关闭重采样功能。
	PoiNum	该通道重采样时，输入给重采样的数据帧中包含的采样点数目。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	SampR	该通道重采样时，输入给重采样的数据帧的采样率。
	ResmpType	该通道的重采样类型。

13.21 AENC

【调试信息】

```
# cat /proc/umap/aenc

[AENC] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Nov 30 2011, 21:17:11]

-----AENC CHN ATTR-----------------------------------------------------------
ChnId  PlType BufSize   Attr1   Attr2   Attr3   Attr4   Attr5
    0   adpcm      30

-----AENC CHN STATUS---------------------------------------------------------
ChnId  RcvFrm   EncOk  FrmErr BufFull GetStrm RlsStrm  WtFile
    0     316     315       0       0       315     315     disable
```

【调试信息分析】

记录当前音频编码属性配置以及状态信息。

【参数说明】

参数		描述
Attribution of AENC Channel（音频编码通道属性）	ChnId	AENC 通道号。
	PlType	编码协议类型。
	BufSize	帧缓存数目。
	Attr1～Attr5	协议相关属性。（具体意义由协议类型决定）
Status of AENC Channel（音频编码通道状态）	ChnId	AENC 通道号。
	RcvFrm	接收的音频帧数目。
	EncOk	成功编码的音频帧数目。
	EncErr	编码失败的音频帧数目。
	BufFull	音频码流缓冲区满的次数（用户未及时获取音频码流数据会导致缓冲区满，进而会导致码流数据丢失）。
	GetStrm	用户获取音频码流的次数

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	RlsStrm	用户释放音频码流的次数
	WtFile	是否启用保存文件功能。 enable：启用； disable：禁用。

13.22 ADEC

【调试信息】

```
[ADEC] Version: [Hi3531_MPP_V1.0.D.0 ], Build Time[Mar 27 2015, 17:01:17]

-----ADEC CHN ATTR-----------------------------------------------------------
ChnId  PlType BufSize     SendCnt     GetCnt     PutCnt    AdecMode    SendFail
```

【调试信息分析】

记录当前音频解码属性配置以及状态信息。

【参数说明】

参数		描述
Attribution of ADEC Channel 音频解码通道属性及状态	ChnId	通道号。
	PlType	解码码协议类型。
	BufSize	帧缓存数目。
	SendCnt	成功发送解码器进行解码的音频帧数目。
	GetCnt	用户获取的码流帧数目。
	PutCnt	用户释放的码流帧数目。
	AdecMode	解码方式： PACK：pack 方式解码 STREAM：stream 方式解码
	SendFail	用户发送码流失败的次数

13.23 MPEG4E

【调试信息】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
# cat /proc/umap/mpeg4e

[MPEG4E] Version: [Hi3531_MPP_V1.0.0.0 ], Build Time[Dec  3 2014, 11:34:45]
-----MODULE PARAM-----------------------------------------------------------
 Mpeg4eMiniBufMode
              0
-----CHN ATTR---------------------------------------------------------------
  ID  Priority  MaxWidth MaxHeight  PicWidth PicHeight   MainStr   VIField   BufSize
ByFrame MaxStrCnt
   0        0       1920      1080       704       576       Yes        No      4147200
Yes       200

-----PICTURE INFO-----------------------------------------------------------
   ID   EncdStart      EncdSucceed     Lost   Disc    Skip    BufLeak Recode  RlsStr
UnrdStr
   0     1683           1683          0      0      0      0      0    1683       0

-----Syntax INFO------------------------------------------------------------
   ID   slcsplt slcmode slcsize HWSize  VWSize
   0    Yes     MbLine    12       0       0

-----STREAM BUFFER----------------------------------------------------------
   ID   Base       RdTail     RdHead     WrTail     WrHead     DataLen     BufFree
   0    0xcb400000  0x9ce80     0x9ce80     0x9ce80     0x9ce80        0        4149184
```

【调试信息分析】

在进行多路 mpeg4 编码时，最常见的问题是用户的上层软件获取码流不及时，导致码流 buffer 不足，进而导致编码丢帧。确认这种情况最好的办法是查看 Disc、BufLeak 和 UnrdStr。BufLeak 表示由于软件发现码流 buffer 不足而丢弃的帧数，Disc 往往表示由于硬件发现码流 buffer 不足而丢弃的帧数。BufLeak+Disc 通常就是由于码流 buffer 不足而丢弃的帧数。UnrdStr 表示码流 buffer 中剩余的码流帧数，这个数字越大，则表示用户获取码流越不及时。

【参数说明】

参数		描述
MODULE PARAM	Mpeg4eMiniBufMode	码流 buffer 配置模式。 0：一般模式； 1：省内存模式。
CHN ATTR	ID	通道号。
	Pri	通道优先级。目前未使用。
	MaxWidth	编码通道最大宽度，以像素为单位。
	MaxHeight	编码通道最大高度，以像素为单位。
	PicWidth	宽度，以像素为单位。
	PicHeight	高度，以像素为单位。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	MainStr	是否主码流。 取值：{0, 1}。
	VIField	目前未使用。都为 No。
	BufSize	码流 buffer 大小，以字节为单位。
	ByFrame	是否按帧获取码流。 取值：{0, 1}。
	MaxStrCnt	允许码流 Buffer 缓存的最大帧数。 缺省值：0x1FFFFFFF。
PICTURE INFO	ID	通道号。
	EncdStart	接收图像数。 接收到待编码图像后，此项值加 1。
	EncdSucceed	编码成功的帧数。
	Lost	编码时丢弃的帧数，包括以下几项： • 由于编码器硬件异常或出现超大帧而丢弃的帧数，即 Disc 表示的帧数。 • RC 进行帧率控制而丢弃的帧数，即 Skip 表示的帧数。 • 码流 buffer 不够而丢弃的帧数，即 BufLeak 表示的帧数
	Disc	由于编码器硬件异常或出现超大帧而丢弃的帧数,有以下几种可能: • 编码器硬件在编码的过程中发现码流 buffer 不够。 • 出现超大帧。 • 编码器硬件出现异常，如死机。 第 1 种情况最为常见。如果 Disc 持续增加，很有可能是上层软件获取码流不及时导致码流 buffer 不足。
	Skip	编码进行帧率控制而丢弃的帧数。
	BufLeak	由于码流 buffer 不足导致的丢帧个数。 编码器启动编码前，发现码流 Buffer 可用空间非常小而丢掉一帧图像时，此项值加 1。 当用户未能及时取走码流时，易出现此项值不为 0。
	Recode	编码器重编的帧数。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	RlsStr	用户已经取走并且释放的帧数。
	UnrdStr	用户未获取的帧数
STREAM BUFFER	ID	通道号。
	Base	码流 buffer 基地址。
	RdTail	读尾指针。
	RdHead	读头指针。
	WrTail	写尾指针。
	WrHead	写头指针。
	DataLen	数据长度。
	BufFree	空闲长度。

13.24 HDMI

【调试信息】

```
# cat /proc/umap/hdmi
[HDMI] Version: [Hi3531_MPP_V1.0.0.0 Debug], Build Time[Nov 26 2012, 17:07:39]

-----MODULE PARAM-------------------------------------------------------------

-----HDMI STATE---------------------------------------------------------------
 DevId  Open Start    Event
     0     y     y     HOT PLUG

############ Hisi HDMI  Dev Stat ############
HPD Status:              IN
TV:                     Power On
EDID Parse Status:        Ok
TV ManufactureName:       DEL
HDMI Sink video capability :
TV support:              1080P_60
TV support:              1080P_50
TV support:              1080P_24
TV support:              1080i_60
TV support:              1080i_50
TV support:              720P_60
TV support:              720P_50
TV support:              576P_50
TV support:              480P_60
TV support:              576i_50
```


海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

```
TV support:              480i_60
TV support:              640X480_60
TV support:              800X600_60
TV support:              1024X768_60
CEC Status:              OFF
HDMI Output Attribute:
HDCP Encryption:          OFF
HDMI PHY Output:          Enable
Video Output:            Enable
Video Format:            1080P_60
Video Output mode:        YCbCr444
Video DeepColor mode:     off
Audio Output:            Enable
Audio input format:       I2S
Audio sample rate:        48000Hz
HDMI Mode:               HDMI
AVI Infoframe:           Enable
AUD Infoframe:           Enable
MPG/VendorSpec Infoframe:  Disable
Gamut Metadata Packet:    Disable
Generic Packet Packet:    Disable
AVMUTE:                 Disable
AVI Inforframe:
0x82,0x02,0x0d,0x67,0x50,0xa8,0x00,0x10,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,
AUD Inforframe:
0x84,0x01,0x0a,0x70,0x01,0x00,0x00,0x00,0x00,
EDID Raw Data:
00 ff ff ff ff ff ff 00 10 ac 16 f0 4c 39 44 41
1c 15 01 03 80 34 20 78 ea 1e c5 ae 4f 34 b1 26
0e 50 54 a5 4b 00 81 80 a9 40 d1 00 71 4f 01 01
01 01 01 01 01 01 28 3c 80 a0 70 b0 23 40 30 20
36 00 06 44 21 00 00 1a 00 00 00 ff 00 38 32 57
58 44 31 37 34 41 44 39 4c 0a 00 00 00 fc 00 44
45 4c 4c 20 55 32 34 31 30 0a 20 20 00 00 00 fd
00 38 4c 1e 51 11 00 0a 20 20 20 20 20 20 01 32
02 03 29 f1 50 90 05 04 03 02 07 16 01 1f 12 13
14 20 15 11 06 23 09 07 07 67 03 0c 00 10 00 38
2d 83 01 00 00 e3 05 03 01 02 3a 80 18 71 38 2d
40 58 2c 45 00 06 44 21 00 00 1e 01 1d 80 18 71
1c 16 20 58 2c 25 00 06 44 21 00 00 9e 01 1d 00
72 51 d0 1e 20 6e 28 55 00 06 44 21 00 00 1e 8c
0a d0 8a 20 e0 2d 10 10 3e 96 00 06 44 21 00 00
18 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3e
#################### END ###################
```

【调试信息分析】

记录 HDMI 当前的运行参数、热插拔状态、显示器 EDID 信息等。

【参数说明】

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
HDMI STATE	DevId	HDMI 设备号。
	Open	HDMI 是否打开。
	Start	HDMI 是否启动。 取值：{y，n} 当热插拔检测程序检测到未连接 HDMI 线时，其状态会自动变为 n，重新连上 HDMI 线后，会自动变为 y。
	Event	热插拔事件。 HOT PLUG：表 HDMI 端口与显示器已连接。 NO PLUG：表端口与显示器未连接 HDMI 线 EDID FAL：表读取显示器 EDID 失败 HDCP FAL：表 HDCP 认证失败 HDCP SUC：表 HDCP 认证成功 HDCP RST：表 HDCP 复位，需重新进行 HDCP 认证。
Hisi HDMI Dev Stat	HPD Status	热插拔状态。 IN：表 HDMI 端口与显示器已建立连接。 OUT：表 HDMI 端口或显示器未连接 HDMI 线。
	TV	显示器上电状态。 Power off：表显示器未上电。 Power on：表显示器已上电。
	EDID Parse Status	EDID 信息解析状态。 Ok：表 EDID 信息解析成功。 Fail：表 EDID 信息解析失败
	TV ManufactureName	显示器厂家名称。
	HDMI Sink video capability	显示器 Sink 端支持的分辨率能力集
	TV support	显示器 Sink 端支持的分辨率。
	CEC Status	CEC 状态。 On：打开。如果状态为 On，将显示以下两个子项： - CEC Physical Add：CEC 物理地址 - CEC Logical Add：CEC 逻辑地址 Off：关闭。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	HDMI Output Attribute	HDMI 输出属性配置。
	HDCP Encryption	HDCP 加密状态。 Enable：使能 Disable：关闭
	HDMI PHY Output	HDMI 传输物理 IP 状态。 Enable：使能。 Disable：关闭。
	Video Output	视频输出状态： Enable：使能。 Disable：关闭。
	Video Format	视频输出时序。
	Video Output mode	视频输出模式。 取值：{RGB444，YCbCr422，YCbCr444}
	Video DeepColor mode	视频 DeepColor 模式。
	Audio Output	音频输出状态. Enable：使能 Disable：关闭
	Audio input format	音频输入模式。 取值：{I2S，SPDIF，HighBitRate}
	Audio sample rate	音频采样率。
	HDMI Mode	HDMI/DVI 模式。
	AVI Infoframe	AVI 信息帧状态 Enable：使能。 Disable：关闭。
	AUD Infoframe	音频信息帧状态。 Enable：使能 Disable：关闭
	MPG/VendorSpec Infoframe	MPEG 信息帧状态。 Enable：使能 Disable：关闭

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	Gamut Metadata Packet	Gamut Metadata 传输使能状态。 Enable：使能 Disable：关闭
	Generic Packet Packet	Generic Packet 传输使能状态。 Enable：使能。 Disable：关闭。
	AVMUTE	静音状态。 Enable：使能 Disable：关闭
	AVI Inforframe	AVI 信息帧数据。
	AUD Inforframe	AUD 信息帧数据。
	EDID Raw Data	EDID 原始数据。

13.25 ACODEC

【调试信息】

```
# cat /proc/umap/acodec
[ACODEC] Version: [Hi3518_MPP_V1.0.0.0 Release], Build Time[Aug 14 2014, 14:11:21]

-----ACODEC PARAM-----------------------------------------------------
LGain RGain DacLVol DacRVol AdcLVol AdcRVol MicLMut MicRMut DacLMut DacRMut Boost
11     11      9      9      30      30       0       0       0
0      1
```

参数		描述
ACODEC PARAM	LGain	输入左声道模拟增益。 取值范围[0,0x1f]。
	RGain	输入右声道模拟增益。 取值范围[0,0x1f]。
	DacLVol	输入左声道数字增益。 取值范围[0,0x7f]。
	DacRVol	输入右声道数字增益。 取值范围[0,0x7f]。

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司

参数		描述
	AdcLVol	输出左声道数字增益。 取值范围[0,0x7f]。
	AdcRVol	输出右声道数字增益。 取值范围[0,0x7f]。
	MicLMut	输入左声道静音状态 0：关闭 1：使能
	MicRMut	输入右声道静音状态 0：关闭 1：使能
	DacLMut	输出左声道静音状态 0：关闭 1：使能
	DacRMut	输出右声道静音状态 0：关闭 1：使能
	Boost	Boost 功能状态。 0：关闭 1：使能

海思专有和保密信息
版权所有 © 深圳市海思半导体有限公司