



CV186AH HDMI 使用手册

Version: 1.0.0

Release date: 2023/12

©2022 北京晶视智能科技有限公司
本文件所含信息归北京晶视智能科技有限公司所有。
未经授权，严禁全部或部分复制或披露该等信息。

目录

1	声明	2
2	概述	3
2.1	重要概念	3
2.2	规格说明	3
3	API 参考	4
4	数据类型	22
5	错误码	58
6	Proc 调试信息	59
7	Debug 调试	72

修订记录

版本	日期	功能描述
1.0.0	2023/08/30	初稿

1 声明



法律声明

本数据手册包含北京晶视智能科技有限公司（下称“晶视智能”）的保密信息。未经授权，禁止使用或披露本数据手册中包含的信息。如您未经授权披露全部或部分保密信息，导致晶视智能遭受任何损失或损害，您应对因之产生的损失/损害承担责任。

本文件内信息如有更改，恕不另行通知。晶视智能不对使用或依赖本文件所含信息承担任何责任。本数据手册和本文件所含的所有信息均按“原样”提供，无任何明示、暗示、法定或其他形式的保证。晶视智能特别声明未做任何适销性、非侵权性和特定用途适用性的默示保证，亦对本数据手册所使用、包含或提供的任何第三方的软件不提供任何保证；用户同意仅向该第三方寻求与此相关的任何保证索赔。此外，晶视智能亦不对任何其根据用户规格或符合特定标准或公开讨论而制作的可交付成果承担责任。

联系我们

地址 北京市海淀区丰豪东路 9 号院中关村集成电路设计园（ICPARK）1 号楼

深圳市宝安区福海街道展城社区会展湾云岸广场 T10 栋

电话 +86-10-57590723 +86-10-57590724

邮编 100094（北京）518100（深圳）

官方网站 <https://www.sophgo.com/>

技术论坛 <https://developer.sophgo.com/forum/index.html>

2 概述

本章节主要用于指导用户 HDMI 模块开发以便提高开发效率；我们不保证此文档更改后能及时告知用户，因此请使用最新版 released SDK 里面的文档

2.1 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。

2.2 规格说明

文档中未特别注明为 HDMI2.0 规格，默认只支持 HDMI1.4 规格的产品。

须知: CV186AH 处理器及后续计划中的处理器采用的是同一套 API 接口，既支持 HDMI1.4 又支持 HDMI2.0。

处理器的 HDMI 规格说明，如表 1-1 所示。

表 2.1: 处理器的 HDMI 规格说明

产品名称	HDMI 规格
CV186AH	HDMI1.4 & 2.0

3 API 参考

该模块主要为用户提供以下 API

- CVI_HDMI_Init: 初始化 HDMI
- CVI_HDMI_Deinit: 去初始化 HDMI
- CVI_HDMI_Open: 打开 HDMI
- CVI_HDMI_Close: 关闭 HDMI
- CVI_HDMI_GetSinkCapability: 获取 HDMI Sink 端的能力集
- CVI_HDMI_SetAttr: 设置 HDMI 属性
- CVI_HDMI_GetAttr: 获取 HDMI 属性
- CVI_HDMI_Start: 启动 HDMI 输出
- CVI_HDMI_Stop: 停止 HDMI 输出
- CVI_HDMI_ForceGetEdid: 获取 HDMI 的 EDID 原始数据
- CVI_HDMI_RegisterCallback: 注册事件回调函数
- CVI_HDMI_UnRegisterCallback: 撤销事件回调函数
- CVI_HDMI_SetInfoFrame: 设置信息帧
- CVI_HDMI_GetInfoFrame: 获取信息帧
- CVI_HDMI_SetHwSpec: 设置指标参数
- CVI_HDMI_GetHwSpec: 获取指标参数
- CVI_HDMI_SetAvmute: 设置 AVMUTE
- CVI_HDMI_SetAudioMute: 设置 AudioMute

· CVI_HDMI_Init

【描述】

初始化 HDMI

【语法】

```
CVI_S32 CVI_HDMI_Init(CVI_VOID);
```

【参数】

无

【返回值】

表 3.1: 返回值表

返回值	描述
0	成功
非 0	失败，其值为错误码

【需求】

- 头文件: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

无

【举例】

注意：举例时未检查API返回值，建议用户使用时请检查返回值做相应处理，下同。

```
CVI_HDMI_ATTR stAttr;
```

```
/* 初始化HDMI */  
CVI_HDMI_Init();  
/* 打开HDMI */  
CVI_HDMI_Open(CVI_VOID);  
/* 获取HDMI属性 */  
CVI_HDMI_GetAttr(&stAttr);  
/* 设置HDMI属性 */  
stAttr.enable_hdmi = CVI_TRUE;  
stAttr.video_FORMAT = CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p60;  
stAttr.deep_color_mode = CVI_HDMI_DEEP_COLOR_24BIT;  
stAttr.enable_audio = CVI_TRUE;  
stAttr.sample_rate = CVI_HDMI_SAMPLE_RATE_48K;  
stAttr.bit_depth = CVI_HDMI_BIT_DEPTH_16;  
CVI_HDMI_SetAttr(&stAttr);  
/* 启动HDMI */  
CVI_HDMI_Start(CVI_VOID);  
/*以下，为使用结束后退出流程*/  
/* 停止HDMI */  
CVI_HDMI_Stop(CVI_VOID);  
/* 关闭HDMI */  
CVI_HDMI_Close(CVI_VOID);  
/* 去初始化HDMI */  
CVI_HDMI_DeInit(CVI_VOID);
```

【相关主题】

CVI_HDMI_DeInit

CVI_HDMI_DeInit**【描述】**

去初始化 HDMI

【语法】

```
CVI_S32 CVI_HDMI_DeInit(CVI_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

【需求】

- 头文件: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

- 若已初始化成功，程序出现异常需要退出时需要调用此接口。
- 未初始化就去初始化或重复去初始化返回成功。

【举例】

请参见 CVI_HDMI_Init 和 CVI_HDMI_RegisterCallback 的举例。

【相关主题】

CVI_HDMI_Init

· CVI_HDMI_Open**【描述】**

打开 HDMI

【语法】

```
CVI_S32 CVI_HDMI_Open(CVI_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

【需求】

- 头文件: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

- CVI_HDMI_Start 之前须确保 HDMI 已被初始化，否则将返回 CVI_ERR_HDMI_NOT_INIT。
- 重复打开 HDMI 返回成功。

【举例】

请参见 CVI_HDMI_RegisterCallback 的举例。

【相关主题】

CVI_HDMI_Close

· CVI_HDMI_Close**【描述】**

关闭 HDMI

【语法】

```
CVI_S32 CVI_HDMI_DeInit(CVI_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

【需求】

- 头文件：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmitx.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- HDMI Start 成功之后，若应用发生异常，需要退出，需要调用此接口及 CVI_HDMI_DeInit 释放 HDMI 资源

【举例】

请参见 CVI_HDMI_RegisterCallback 的举例。

【相关主题】

CVI_HDMI_Deinit

· CVI_HDMI_SetAttr**【描述】** 设置 HDMI 属性**【语法】**

```
CVI_S32 CVI_HDMI_SetAttr(const CVI_HDMI_ATTR* attr);
```

【参数】

无

【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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmix.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
 - 用户需在启动之前设置 HDMI 属性；若 HDMI 已启动，则应先停止 HDMI，设置属性后再重新启动。不遵循该流程使用的行为是未定义的。
 - 部分属性暂时不支持，见数据类型 CVI_HDMI_ATTR 说明。
 - 若只设置部分属性，设置前应先获取属性，赋值该部分属性后再设置。
 - 部分显示设备的 EDID 可能不够准确，如有的显示设备具备显示 4kP60 的能力，但其 EDID 中申明并不支持 4kP60 和 SCDC，此时调用此接口设置 4kP60 输出时，为了避免引入兼容性问题，HDMI 驱动不会强制设置 4kP60 输出，电视不会正常显示。针对类似情况，若确定显示设备支持 4kP60，可将接口参数中 auth_mode 设置为 CVI_TRUE，HDMI 驱动将强制按照 4kP60 输出。但存在引入各种兼容性问题的风险，如电视花屏、闪屏，甚至死机等。
 - HDMI1.4 中的 video_format 取值最大只能到 CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p30。
 - 此接口为同步接口，在调用过程中，驱动可能会通过 SCDC 与显示设备交互，此过程耗时较长（正常约 3~5s）。
 - 关于 CVI_HDMI_SetAttr，使用时需要先调用 HDMI 的 CVI_HDMI_Stop 接口将 HDMI 的信号关断然后再调用，设置完成后需再调用 HDMI 的 CVI_HDMI_Start 接口打开信号输出。
 - 在调用此接口前一定要先调用 CVI_HDMI_GetSinkCapability 接口获取对端的能力然后合理的配置属性，特别的情况如对端不支持 HDMI 时需要先调用 CVI_HDMI_SetAttr 接口将颜色空间配置成 RGB，然后才能调用此接口来配置 HDMI 的其他属性，否则驱动会有报错。
 - 有开机画面且非平滑过渡的场景下，应用上电后第一次设置属性前一定要先调用一次 CVI_HDMI_Stop，否则可能会导致兼容性问题。
- 备注：当 auth_mode 设置为 CVI_TRUE 时驱动会强制 SCDC 操作，若显示设备不支持或不插 HDMI 线，此接口耗时可能更长。

【举例】

请参见 CVI_HDMI_RegisterCallback 的举例。

【相关主题】

CVI_HDMI_GetAttr

· CVI_HDMI_GetAttr

【描述】

获取 HDMI 属性

【语法】

```
CVI_S32 CVI_HDMI_GetAttr(CVI_HDMI_ATTR* attr);
```

【参数】

参数名称	描述	输入/输出
attr	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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmitx.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 若只设置部分属性，设置前应先获取属性，赋值需要修改的属性后再设置。

【举例】

请参见 CVI_HDMI_RegisterCallback 的举例。

【相关主题】

CVI_HDMI_SetAttr

· CVI_HDMI_Start**【描述】**

启动 HDMI 输出

【语法】

```
CVI_S32 CVI_HDMI_Start(CVI_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

【需求】

- 头文件: `cvi_hdmi.h`、`cvi_comm_hdmi.h`
- 库文件: `libhdmitx.a`

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 `CVI_ERR_HDMI_DEV_NOT_OPEN`。

【举例】

请参见 `CVI_HDMI_Init` 和 `CVI_HDMI_RegisterCallback` 的举例。

【相关主题】

`CVI_HDMI_Stop`

· CVI_HDMI_Stop**【描述】**

停止 HDMI 输出

【语法】

```
CVI_S32 CVI_HDMI_Stop(CVI_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

【需求】

- 头文件: `cvi_hdmi.h`、`cvi_comm_hdmi.h`
- 库文件: `libhdmitx.a`

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 `CVI_ERR_HDMI_DEV_NOT_OPEN`。

【举例】

- 请参见 `CVI_HDMI_RegisterCallback` 的举例。

【相关主题】

- 无

· CVI_HDMI_GetSinkCapability

【描述】

获取 HDMI Sink 端的能力集

【语法】

```
CVI_S32  
CVI_HDMI_GetSinkCapability(CVI_HDMI_SINK_CAPABILITY*  
capability);
```

【参数】

参数名称	描述	输入/输出
capability	HDMI Sink 端能力集结构体指针	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: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。应在 HDMI 启动且插入线缆之后调用。
- 驱动会在应用调用 CVI_HDMI_Open 后读取显示设备的 EDID，读取和解析过程需要一定时间。因此，若应用注册 HDMI 事件回调，则建议在检测到插入事件后调用；若应用不注册回调，则在 open HDMI 后延时 1~2 秒再调用此接口（若不延时，则可能出现获取失败）

【举例】

```
CVI_HDMI_EDID stEdidData;  
CVI_HDMI_SINK_CAPABILITY stSinkCap;  
/*初始化HDMI*/  
CVI_HDMI_Init();  
/*打开HDMI*/  
CVI_HDMI_Open(CVI_VOID);  
/*设置属性*/  
...  
/*启动HDMI*/  
CVI_HDMI_Start(CVI_VOID);  
...  
sleep(2);
```

(下页继续)

(续上页)

```
/*获取EDID*/  
CVI_HDMI_ForceGetEdid(&stEdidData);  
/*获取能力集合*/  
CVI_HDMI_GetSinkCapability(&stSinkCap);。
```

【相关主题】

CVI_HDMI_ForceGetEdid

· CVI_HDMI_ForceGetEdid**【描述】**

获取 HDMI 的 EDID 原始数据

【语法】

CVI_HDMI_ForceGetEdid(CVI_HDMI_EDID* edid_data);

【参数】

参数名称	描述	输入/输出
edid_data	HDMI 的 EDID 信息。	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: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- HDMI 内部在线缆插入后已从 Sink 获取 EDID。该 API 为强制获取 EDID，一般情况下不需使用。

【举例】

- 请参见 CVI_HDMI_GetSinkCapability。

【相关主题】

- CVI_HDMI_GetSinkCapability

· CVI_HDMI_RegisterCallback

【描述】

注册 HDMI 事件回调函数。

【语法】

```
CVI_S32 CVI_HDMI_RegisterCallback(const  
CVI_HDMI_CALLBACK_FUNC* callback_func);
```

【参数】

参数名称	描述	输入/输出
callback_func	HDMI 回调函数结构体指针。	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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmix.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。建议用户注册 HDMI 事件回调函数。例如，当产生热插拔事件时，可以通过注册的回调函数读取热插拔后产生的能力集信息为依据更改 HDMI 的属性，然后重新启动 HDMI，使 HDMI 的属性适配新插入的对端显示器/电视。若用户不注册事件回调函数，则当事件产生时，HDMI 内部会采取默认的处理方式。
- 若用户注册了事件回调函数，则退出 HDMI 前应调用 CVI_HDMI_UnRegisterCallback 注销该回调函数，否则会有内存泄漏的问题。若用户没有注册回调函数，则驱动报的事件会走内核回调处理。
- 内核事件回调函数只在用户 start HDMI 之后才能正常响应拔插事件。
- 内核回调函数中若事件为 HPD 事件驱动会自动 start HDMI，若事件为 UNHPD 事件驱动会自动 stop HDMI。
- 重复注册同一回调函数与参数将返回 CVI_ERR_HDMI_CALLBACK_ALREADY。
- 建议用户最多同时注册 10 个不同的回调函数，注册超过 10 个的回调函数将会影响之前注册的回调函数使用。
- 若同时注册了多个不同的事件回调函数，当一个事件产生时，HDMI 会按事件回调函数注册先后顺序，遍历事件池，响应用户注册的回调函数。

【相关主题】

- CVI_HDMI_UnRegisterCallback

· CVI_HDMI_UnRegisterCallback

【描述】

撤销 HDMI 事件回调函数

【语法】

```
CVI_S32 CVI_HDMI_UnRegisterCallback(const  
CVI_HDMI_CALLBACK_FUNC* callback_func);
```

【参数】

参数名称	描述	输入/输出
callback_func	HDMI 回调函数结构体指针	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: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmitx.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 若用户注册了回调函数，则退出 HDMI 前，应使用撤销 HDMI 事件回调函数。
- 重复撤销或未注册就撤销将返回 CVI_ERR_HDMI_CALLBACK_NOT_REGISTER。

【举例】

- 请参见 CVI_HDMI_RegisterCallback。

【相关主题】

- CVI_HDMI_RegisterCallback

· CVI_HDMI_SetInfoFrame**【描述】**

设置信息帧

【语法】

```
CVI_S32 CVI_HDMI_SetInfoFrame(const CVI_HDMI_INFOFRAME*  
infoframe);
```

【参数】

参数名称	描述	输入/输出
infoframe	HDMI 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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmidx.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 该接口属于高级接口，一般不需要调用。若用户使用，则应根据已设置的音视频相关属性（如 enVideoFORMAT），及遵从《High-Definition Multimedia Interface Specification Version 1.4b》、《High-Definition Multimedia Interface Specification Version 2.0》、《CEA-861-D》与《CEA-861-F》标准基础上设置信息帧，不依据音视频属性遵从标准发送信息帧的行为是未定义的。调用此接口，可能会造成显示异常。

【举例】

```
CVI_HDMI_INFOFRAME stInfoFrame;
/* 初始化HDMI */
CVI_HDMI_Init ();
/* 打开HDMI */
CVI_HDMI_Open (CVI_VOID);
/* 设置属性 */
...
/* 启动HDMI */
CVI_HDMI_Start (CVI_VOID);
...
/*设置AVI的有效宽高比*/
CVI_HDMI_GetInfoFrame(&stInfoFrame);
stInfoFrame.infoframe_type = CVI_INFOFRAME_TYPE_AVI;
stInfoFrame.infoframe_unit.avi_infoframe.active_aspect_ratio = CVI_HDMI_
↪ACTIVE_ASPECT_RATIO_16TO9_TOP;
CVI_HDMI_SetInfoFrame (&stInfoFrame);
```

【相关主题】

- CVI_HDMI_GetInfoFrame

· CVI_HDMI_GetInfoFrame**【描述】**

获取信息帧。

【语法】

```
CVI_S32 CVI_HDMI_GetInfoFrame(CVI_HDMI_INFOFRAME*
    infoframe);
```

【参数】

参数名称	描述	输入/输出
infoframe	HDMI 信息帧结构体指针	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: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

- 该接口属于高级接口，一般不需要调用。
- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 调用前须确保单板上电后已调用过 CVI_HDMI_SetAttr，否则通过此接口获取到的部分参数可能是非法值。

【举例】

- 请参考 CVI_HDMI_SetInfoFrame

【相关主题】

- CVI_HDMI_SetInfoFrame

· CVI_HDMI_SetHwSpec**【描述】**

设置 HDMI 指标参数。

【语法】

```
CVI_S32 CVI_HDMI_SetHwSpec(const CVI_HDMI_HW_SPEC*
    hw_spec);
```

【参数】

参数名称	描述	输入/输出
hw_spec	HDMI 指标参数结构体	输入
infoframe	HDMI 信息帧结构体指针	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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmix.a

【注意】

- 正常情况请勿使用该接口。若必要情况下需要设置 HDMI 硬件指标参数，请务必确保参数设置后 HDMI 电气特性符合协议标准，否则可能出现不可预知的兼容性问题。
- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 若需要设置 HDMI 硬件指标参数，请务必在调用 CVI_HDMI_SetAttr 后调用此接口。
- 调用此接口前请务必先调用 CVI_HDMI_GetHwSpec 获取当前所有频段的指标参数。根据实际需求调整对应频段的具体参数 (参考 CVI_HDMI_HW_SPEC)。
- 根据不同分辨率，色深 (Deep Color) 以及颜色空间 (Color Space) 可划分为四个频段，如表 2-1 所示。
- 调用该接口调试硬件指标时，分辨率，色深以及颜色空间必须按下表设置，其中 [25M, 100M]TMDs 时钟频段需要下表两种配置都要调试。同时必须将 CVI_HDMI_ATTR 成员 auth_mode_en 设置成 CVI_TRUE。若调试结束后，正常使用该接口设置指标参数时，请务必将 CVI_HDMI_ATTR 成员 auth_mode_en 设置成 CVI_FALSE。

TMDs 时钟 频段	分 辨 率 (video_format)	色 深 (deep_color_mode)	颜 色 空 间 (enVidOutMode)
[25M, 100M]	480P_60	24BIT	RGB444
[25M, 100M]	1080P_30	24BIT	RGB444
[100M, 165M]	1080P_60	24BIT	RGB444
[165M, 340M]	3840x2160P_30	24BIT	RGB444
[340M, 600M]	3840x2160P_60	24BIT	RGB444

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。

【注意事项】

- 此接口系高级接口，目前尚未实现。

【相关主题】

- CVI_HDMI_GetHwSpec

· CVI_HDMI_GetHwSpec**【描述】**

获取 HDMI 指标参数。

【语法】

```
CVI_S32 CVI_HDMI_GetHwSpec(CVI_HDMI_HW_SPEC* hw_spec);
```

【参数】

参数名称	描述	输入/输出
hw_spec	HDMI 指标参数结构体	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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmix.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 此接口系高级接口，目前尚未实现。

【举例】

- 请参考 CVI_HDMI_SetHwSpec

【相关主题】

- CVI_HDMI_SetHwSpec

· CVI_HDMI_SetAvmute**【描述】**

设置 AVMUTE。

【语法】

```
CVI_S32 CVI_HDMI_SetAvmute(const CVI_BOOL avmute_en);
```

【参数】

参数名称	描述	输入/输出
avmute_en	AVMUTE 使能标志	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: cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件: libhdmix.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 由于不同设备对 AVMUTE 包有不同的响应，在有些设备上有可能导致黑屏、绿屏等异常现象，建议不要使用此接口。
- AVMUTE 的设置操作在 start、stop 接口中均有涉及，其中在 CVI_HDMI_Start 接口中驱动会 clean avmute，在 CVI_HDMI_Stop 接口中驱动会 set avmute。
- 用户通过此接口设置了 AVMUTE 之后，如果后续有 stop、start 等操作时，AVMUTE 的状态会被改变，最终状态可能会与用户前面设置的值不一样。
- 没有注册回调函数时用户不调用 CVI_HDMI_Start 或 CVI_HDMI_Stop 状态也可能被改变，详细说明请参看 CVI_HDMI_RegisterCallback 接口的注意事项。

【举例】

```
/* 初始化HDMI */
CVI_HDMI_Init();
/* 启动HDMI */
CVI_HDMI_Start(CVI_VOID);
/* avmute set */
CVI_HDMI_SetAvmute(CVI_HDMI_ID_0, CVI_TRUE);
/*以下，为使用结束后退出流程*/
/* avmute clear */
CVI_HDMI_SetAvmute(CVI_VOID, CVI_FALSE);
/* 关闭HDMI */
CVI_HDMI_Close(CVI_VOID);
/* 去初始化HDMI */
CVI_HDMI_DeInit();
```

【相关主题】

- CVI_HDMI_SetHwSpec

· CVI_HDMI_SetAudioMute

【描述】

设置 AudioMUTE。

【语法】

```
CVI_S32 CVI_HDMI_SetAudioMute(const CVI_BOOL* audio_mute_en)
```

【参数】

参数名称	描述	输入/输出
audio_mute_en	AudioMute 使能标志	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：cvi_hdmi.h、cvi_comm_hdmi.h
- 库文件：libhdmity.a

【注意】

- 调用前须确保 HDMI 已被 start，否则将返回 CVI_ERR_HDMI_DEV_NOT_OPEN。
- 由于不同设备对 AudioMUTE 包有不同的响应，在有些设备上有可能导致黑屏、绿屏等异常现象，建议不要使用此接口。
- AudioMute 的设置操作在 start、stop 接口中均有涉及，其中在 CVI_HDMI_Start 接口中驱动会 clean AudioMute，在 CVI_HDMI_Stop 接口中驱动会 set AudioMute。
- 用户通过此接口设置了 AudioMute 之后，如果后续有 stop、start 等操作时，AudioMute 的状态会被改变，最终状态可能会与用户前面设置的值不一样。
- 没有注册回调函数时用户不调用 CVI_HDMI_Start 或 CVI_HDMI_Stop 状态也可能被改变，详细说明请参看 CVI_HDMI_RegisterCallback 接口的注意事项。

【举例】

```
/* 初始化HDMI */  
CVI_HDMI_Init();  
/* 启动HDMI */
```

(下页继续)

(续上页)

```
CVI_HDMI_Start(CVI_VOID);  
/* avmute set */  
CVI_HDMI_SetAudioMute(CVI_TRUE);  
/*以下，为使用结束后退出流程*/  
/* avmute clear */  
CVI_HDMI_SetAudioMute(CVI_FALSE);  
/* 关闭HDMI */  
CVI_HDMI_Close(CVI_VOID);  
/* 去初始化HDMI */  
CVI_HDMI_DeInit();
```

【相关主题】

- CVI_HDMI_SetHwSpec

4 数据类型

该模块主要为用户提供以下数据类型

- CVI_HDMI_CALLBACK: 定义 HDMI 回调函数指针类型
- CVI_HDMI_CALLBACK_FUNC: 定义 HDMI 回调函数结构体。
- CVI_HDMI_EVENT_TYPE: 定义 HDMI 事件通知枚举
- CVI_HDMI_ATTR: 定义 HDMI 属性结构体
- CVI_HDMI_VIDEO_FORMAT: 定义 HDMI 视频制式枚举
- CVI_HDMI_VIDEO_MODE: 定义 HDMI 颜色空间类型枚举
- CVI_HDMI_DEEP_COLOR: 定义 HDMI 深色模式枚举
- CVI_HDMI_SAMPLE_RATE: 定义 HDMI 音频输出采样率枚举
- CVI_HDMI_BIT_DEPTH: 定义 HDMI 音频输出采样位宽枚举
- CVI_HDMI_AUDIO_INFO: 定义 HDMI 音频能力集信息
- CVI_HDMI_TIMING_INFO: 定义 HDMI 详细时序信息
- CVI_HDMI_DETAIL_TIMING: 定义 HDMI 详细时序能力集信息。
- CVI_HDMI_SINK_CAPABILITY: 定义 HDMI Sink 端能力集结构体
- CVI_HDMI_EDID: 定义 HDMI EDID 信息结构体
- CVI_HDMI_INFOFRAME: 定义 HDMI 信息帧结构体
- CVI_HDMI_INFOFRAME_UNIT: 定义 HDMI 信息帧单元结构体
- CVI_HDMI_AVI_INFOFRAME: 定义 AVI 信息帧（版本 2）单元结构体
- CVI_HDMI_AUDIO_INFOFRAME: 定义 AUDIO 信息帧（版本 1）单元结构体
- CVI_HDMI_VENDORSPEC_INFOFRAME: vendor 信息帧结构体
- CVI_HDMI_COLOR_SPACE: 定义颜色空间枚举
- CVI_HDMI_BAR_INFO: 定义 Bar 信息枚举
- CVI_HDMI_SCAN_INFO: 定义 Scan 信息枚举
- CVI_HDMI_COLORIMETRY: 定义 Colorimetry 信息枚举
- CVI_HDMI_EX_COLORIMETRY: 定义扩展 Colorimetry 信息枚举
- CVI_HDMI_ASPECT_RATIO: 定义图像宽高比枚举

- CVI_HDMI_ACTIVE_ASPECT_RATIO: 定义实际图像宽高比枚举
- CVI_HDMI_PIC_SCALINE: 定义图像扫描信息枚举
- CVI_HDMI_RGB_QUANT_RANGE: 定义 YCC 量化范围枚举
- CVI_HDMI_AUDIO_CHN_CNT: 定义像素复制次数枚举
- CVI_HDMI_CODING_TYPE: 定义音频编码类型枚举
- CVI_HDMI_AUDIO_SAMPLE_SIZE: 定义音频采样大小枚举
- CVI_HDMI_AUDIO_SAMPLE_FREQ: 定义音频采样频率枚举
- CVI_HDMI_LEVEL_SHIFT_VAL: 定义音频 Shift 信息枚举
- CVI_HDMI_IFE_PLAYBACK_LEVEL: 定义音频 Playback 信息枚举
- CVI_HDMI_EOTF: 定义支持的 EOTF 能力结构体
- CVI_HDMI_HDR_METADATA_TYPE: 定义支持 metadata 类型结构体
- CVI_HDMI_HDR_CAPABILITY: 定义 HDR 能力结构体
- CVI_HDMI_HW_PARAM: HDMI 指标参数结构体
- CVI_HDMI_HW_SPEC: HDMI 指标参数
- CVI_HDMI_MOD_PARAM: HDMI 模块参数设置
- CVI_HDMI_TRACE_LEN: HDMI 信号线在 PCB 板上的走线长度枚举
- CVI_HDMI_VENDOR_USER_DATA_MAX_LEN: 最大自定义 vendor 数据的长度
- CVI_HDMI_MAX_AUDIO_CAPABILITY_CNT: 能获取的音频能力个数
- CVI_HDMI_MAX_SAMPLE_RATE_NUM: 音频采样率个数
- CVI_HDMI_MAX_BIT_DEPTH_NUM: 音频采样深度个数
- CVI_HDMI_DETAIL_TIMING_MAX: 详细的视频制式信息个数
- CVI_HDMI_EDID_RAW_DATA_LEN: EDID 原始数据的长度
- CVI_HDMI_HW_PARAM_NUM: 硬件参数的个数
- CVI_HDMI_MANUFACTURE_NAME_LEN: 制造商名称的长度
- CVI_HDMI_ID: 定义 HDMI 接口号。
- CVI_HDMI_FORCE_ACTION: 定义 HDMI 强制动作枚举
- CVI_HDMI_INFOFRAME_TYPE: 定义 HDMI 信息帧类型枚举
- CVI_HDMI_VIDEO_INFO: 定义 HDMI 视频信息结构体

· CVI_HDMI_CALLBACK

【说明】

定义 HDMI 回调函数指针类型

【定义】

```
typedef void (* CVI_HDMI_CALLBACK)(CVI_HDMI_EVENT_TYPE event, CVI_
→VOID* private_data)
```

(下页继续)

(续上页)

【成员】

成员名称	描述
event	HDMI 事件通知类型。
private_data	事件私有数据。

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_CALLBACK_FUNC
- CVI_HDMI_RegisterCallback
- CVI_HDMI_UnRegisterCallback

· CVI_HDMI_CALLBACK_FUNC**【说明】**

定义 HDMI 回调函数结构体

【定义】

```
typedef struct _CVI_HDMI_CALLBACK_FUNC{  
    CVI_HDMI_CALLBACK hdmi_event_callback;  
    CVI_VOID* private_data;  
} CVI_HDMI_CALLBACK_FUNC;
```

【成员】

成员名称	描述
hdmi_event_callback	事件处理回调函数。
private_data	回调函数参数私有数据。

【注意事项】

- 建议用户注册事件回调函数，若不注册则 HDMI 内部采取默认行为。见 CVI_HDMI_RegisterCallback。

【相关数据类型及接口】

- CVI_HDMI_CALLBACK
- CVI_HDMI_RegisterCallback
- CVI_HDMI_UnRegisterCallback

· CVI_HDMI_EVENT_TYPE

【说明】

定义 HDMI 事件通知枚举

【定义】

```
typedef enum _CVI_HDMI_EVENT_TYPE{  
    CVI_HDMI_EVENT_HOTPLUG = 1,  
    CVI_HDMI_EVENT_NO_PLUG = 2,  
    CVI_HDMI_EVENT_EDID_FAIL = 3,  
    CVI_HDMI_EVENT_BUTT  
} CVI_HDMI_EVENT_TYPE;
```

【成员】

成员名称	描述
CVI_HDMI_EVENT_HOTPLUG	HDMI Cable 插入事件。
CVI_HDMI_EVENT_NO_PLUG	HDMI Cable 拔出事件。
CVI_HDMI_EVENT_EDID_FAIL	HDMI EDID 读取失败事件。

【注意事项】

- 无

【相关数据类型及接口】

- CVI_HDMI_RegisterCallback
- CVI_HDMI_UnRegisterCallback

· CVI_HDMI_ATTR

【说明】

定义 HDMI 属性结构体

【定义】

```
typedef struct _CVI_HDMI_ATTR{  
    CVI_BOOL hdmi_en;  
    CVI_HDMI_VIDEO_FORMAT video_format;  
    CVI_HDMI_DEEP_COLOR deep_color_mode;  
    CVI_BOOL audio_en;  
    CVI_BOOL hdp14_en;  
    CVI_HDMI_FORCE_ACTION hdmi_force_output;  
    CVI_HDMI_VIDEO_MODE hdmi_video_input;  
    CVI_HDMI_VIDEO_MODE hdmi_video_output;  
    CVI_HDMI_SAMPLE_RATE sample_rate;  
    CVI_HDMI_BIT_DEPTH bit_depth;  
    CVI_BOOL auth_mode_en;  
    CVI_U64 audio_start_paddr;  
    CVI_U64 audio_stop_paddr;
```

(下页继续)

(续上页)

```

    CVI_BOOL deep_color_adapt_en;
    CVI_U32 pix_clk;
} CVI_HDMI_ATTR;

```

【成员】

成员名称	描述
hdmi_en	是否使能 hdmi。
video_format	视频制式，此参数需要与 VO 配置的制式保持一致。建议用户设置为 Sink 能力集支持的制式。
deep_color_mode	DeepColor 输出模式。CVI_HDMI_DEEP_COLOR_24BIT、CVI_HDMI_DEEP_COLOR_30BIT、CVI_HDMI_DEEP_COLOR_36BIT、默认为 CVI_HDMI_DEEP_COLOR_24BIT。目前仅支持 24bit 模式
audio_en	是否使能音频。
hdcp14_en	是否使能 hdcp1.4
hdmi_force_output	是否强制 HDMI 输出。
hdmi_video_input	视频输入的颜色空间类型
hdmi_video_output	视频输出的颜色空间类型
sample_rate	音频采样率，此参数需要与 AO 的配置保持一致。CV186AH 处理器目前不支持采样率小于 32KHz 的采样率，建议用户设置为 Sink 能力集支持的采样率。
bit_depth	音频位宽，默认为 16，此参数需要与 AO 的配置保持一致。建议用户设置为 Sink 能力集支持位宽。
auth_mode_en	使能该模式，HDMI 强制输出，不再去根据显示或认证设备的 EDID 信息来自适应调整，主要针对认证场景。0：不使能 auth_mode；1：使能 auth_mode。说明：默认为 CVI_FALSE，此标志用于调试和 HDMI 认证测试用，建议用户不要修改，其他注意事项请参考 CVI_HDMI_SetAttr。
audio_start_paddr	音频数据在内存中的起始物理地址
audio_stop_paddr	音频数据在内存中的结束物理地址
deep_color_adapt_en	是否打开驱动 Deepcolor 自适应策略。默认为 CVI_FALSE。
pix_clk	用户设置自定义时序的像素时钟（单位：KHz）。说明：用户设置自定义时序时一定设置此参数，此参数仅当 video_format 为 VESA_CUSTOMER_DEFINE 时才会生效。

【注意事项】

- 用户可以根据建议值设置 HDMI 属性。
- 对于暂时不支持的属性，请设置为默认值。

【相关数据类型及接口】

- CVI_HDMI_SetAttr
- CVI_HDMI_GetAttr

CVI_HDMI_VIDEO_FORMAT

【说明】

定义 HDMI 视频制式枚举

【定义】

```
typedef enum CVI_HDMI_VIDEO_FORMAT{
    CVI_HDMI_VIDEO_FORMAT_CEA861_640x480p60    = 1,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x480p60    = 2,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p60    = 4,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080i60    = 5,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x480i60    = 6,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x240p60    = 8,
    CVI_HDMI_VIDEO_FORMAT_CEA861_2880x480i60    = 10,
    CVI_HDMI_VIDEO_FORMAT_CEA861_2880x240p60    = 12,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x480p60    = 14,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p60    = 16,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x576p50    = 17,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p50    = 19,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080i50    = 20,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x576i50    = 21,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x288p50    = 23,
    CVI_HDMI_VIDEO_FORMAT_CEA861_2880x576i50    = 25,
    CVI_HDMI_VIDEO_FORMAT_CEA861_2880x288p50    = 27,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x576p50    = 29,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p50    = 31,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p24    = 32,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p25    = 33,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p30    = 34,
    CVI_HDMI_VIDEO_FORMAT_CEA861_2880x480p60    = 35,
    CVI_HDMI_VIDEO_FORMAT_CEA861_2880x576p50    = 37,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080i100   = 40,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p100   = 41,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x576p100    = 42,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1440x576i100    = 44,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080i120   = 46,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p120   = 47,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x480p120    = 48,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x480i120    = 50,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x576p200    = 52,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x576i200    = 54,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x480p240    = 56,
    CVI_HDMI_VIDEO_FORMAT_CEA861_720x480i240    = 58,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p24    = 60,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p25    = 61,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1280x720p30    = 62,
    CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p120   = 63,
```

(下页继续)

(续上页)

```

CVI_HDMI_VIDEO_FORMAT_CEA861_1920x1080p100 = 64,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p24 = 79,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p25 = 80,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p30 = 81,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p50 = 82,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p60 = 83,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p100 = 84,
CVI_HDMI_VIDEO_FORMAT_CEA861_1680x720p120 = 85,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p24 = 86,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p25 = 87,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p30 = 88,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p50 = 89,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p60 = 90,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p100 = 91,
CVI_HDMI_VIDEO_FORMAT_CEA861_2560x1080p120 = 92,
CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p24 = 93,
CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p25 = 94,
CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p30 = 95,
CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p50 = 96,
CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p60 = 97,
CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p24 = 98,
CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p25 = 99,
CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p30 = 100,
CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p50 = 101,
CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p60 = 102,
CVI_HDMI_VIDEO_FORMAT_CVT_RB_2560X1440p60 = 108,
CVI_HDMI_VIDEO_FORMAT_CUSTOMER_DEFINE,
CVI_HDMI_VIDEO_FORMAT_BUTT,
}CVI_HDMI_VIDEO_FORMAT

```

【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。

CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p50

CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p60

CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p50

CVI_HDMI_VIDEO_FORMAT_CEA861_4096x2160p60 以上四种制式为 HDMI2.0 规格

- 当 CVI_HDMI_ATTR 中的 hdmi_en 设置成 CVI_FALSE(DVI 模式) 时, 不支持 CVI_HDMI_VIDEO_FORMAT_CEA861_3840x2160p24 及以上分辨率, 强行设置时会返回 CVI_ERR_HDMI_FEATURE_NO_SUPPORT

【相关数据类型及接口】

- CVI_HDMI_SetAttr

· CVI_HDMI_VIDEO_MODE**【说明】**

定义 HDMI 颜色空间类型枚举

【定义】

```
typedef enum _CVI_HDMI_VIDEO_MODE{  
    CVI_HDMI_VIDEO_MODE_RGB444,  
    CVI_HDMI_VIDEO_MODE_YCBCR422,  
    CVI_HDMI_VIDEO_MODE_YCBCR444,  
    CVI_HDMI_VIDEO_MODE_YCBCR420,  
    CVI_HDMI_VIDEO_MODE_BUTT  
} CVI_HDMI_VIDEO_MODE;
```

【成员】

成员名称	描述
RGB444	RGB444 输出模式。
YCBCR422	YCBCR422 输出模式。
YCBCR444	YCBCR444 输出模式。
YCBCR420	YCBCR420 输出模式。

【注意事项】

- CVI_HDMI_VIDEO_MODE 不支持 YUV420

【相关数据类型及接口】

- 无

· CVI_HDMI_DEEP_COLOR**【说明】**

定义 HDMI 深色模式枚举

【定义】

```
typedef enum _CVI_HDMI_DEEP_COLOR{  
    CVI_HDMI_DEEP_COLOR_24BIT = 24,  
    CVI_HDMI_DEEP_COLOR_30BIT = 30,  
    CVI_HDMI_DEEP_COLOR_36BIT = 36,  
    CVI_HDMI_DEEP_COLOR_BUTT  
} CVI_HDMI_DEEP_COLOR;
```

【成员】

成员名称	描述
24BIT	HDMI Deep Color 24bit 模式。
30BIT	HDMI Deep Color 30bit 模式。
36BIT	HDMI Deep Color 36bit 模式。

【注意事项】

- 目前仅支持 24bit 模式

【相关数据类型及接口】

- CVI_HDMI_SetAttr

· CVI_HDMI_SAMPLE_RATE**【说明】**

定义 HDMI 音频输出采样率枚举

【定义】

```
typedef enum _CVI_HDMI_SAMPLE_RATE{  
    CVI_HDMI_SAMPLE_RATE_UNKNOWN,  
    CVI_HDMI_SAMPLE_RATE_32K = 32000,  
    CVI_HDMI_SAMPLE_RATE_44K = 44100,  
    CVI_HDMI_SAMPLE_RATE_48K = 48000,  
    CVI_HDMI_SAMPLE_RATE_88K = 88000,  
    CVI_HDMI_SAMPLE_RATE_96K = 96000,  
    CVI_HDMI_SAMPLE_RATE_176K = 176400,  
    CVI_HDMI_SAMPLE_RATE_192K = 192000,  
    CVI_HDMI_SAMPLE_RATE_BUTT,  
}_CVI_HDMI_SAMPLE_RATE;
```

【成员】

无

【注意事项】

- 当前仅支持 CVI_HDMI_SAMPLE_RATE_32K、CVI_HDMI_SAMPLE_RATE_44K、CVI_HDMI_SAMPLE_RATE_48K。
- 设置 CVI_HDMI_SAMPLE_RATE_BUTT 及枚举以外的值将返回 CVI_ERR_HDMI_INVALID_PARA，设置其他参数将返回 CVI_ERR_HDMI_FEATURE_NO_SUPPORT。

【相关数据类型及接口】

- CVI_HDMI_SetAttr

· CVI_HDMI_BIT_DEPTH**【说明】**

定义 HDMI 音频输出采样位宽枚举

【定义】

```
typedef enum _CVI_HDMI_BIT_DEPTH{  
    CVI_HDMI_BIT_DEPTH_UNKNOWN,  
    CVI_HDMI_BIT_DEPTH_16 = 16,  
    CVI_HDMI_BIT_DEPTH_24 = 24,  
    CVI_HDMI_BIT_DEPTH_BUTT,  
}_CVI_HDMI_BIT_DEPTH;
```


【成员】

无

【注意事项】

- 目前只支持设置 CVI_HDMI_BIT_DEPTH_16 和 CVI_HDMI_BIT_DEPTH_24 的音频采样位宽。
- CVI_HDMI_BIT_DEPTH_UNKNOWN、CVI_HDMI_BIT_DEPTH_8、CVI_HDMI_BIT_DEPTH_18、CVI_HDMI_BIT_DEPTH_20、CVI_HDMI_BIT_DEPTH_32 暂不支持，设置这些参数时将返回 CVI_ERR_HDMI_FEATURE_NO_SUPPORT。设置 CVI_HDMI_BIT_DEPTH_BUTT 及枚举以外的值将返回 CVI_ERR_HDMI_INVALID_PARA。

【相关数据类型及接口】

- CVI_HDMI_SetAttr

· CVI_HDMI_AUDIO_FORMAT_CODE**【说明】**

定义 HDMI 音频格式枚举

【定义】

```
typedef enum CVI_HDMI_AUDIO_FORMAT_CODE{
    CVI_HDMI_AUDIO_FORMAT_CODE_RESERVED,
    CVI_HDMI_AUDIO_FORMAT_CODE_PCM,
    CVI_HDMI_AUDIO_FORMAT_CODE_AC3,
    CVI_HDMI_AUDIO_FORMAT_CODE_MPEG1,
    CVI_HDMI_AUDIO_FORMAT_CODE_MP3,
    CVI_HDMI_AUDIO_FORMAT_CODE_MPEG2,
    CVI_HDMI_AUDIO_FORMAT_CODE_AAC,
    CVI_HDMI_AUDIO_FORMAT_CODE_DTS,
    CVI_HDMI_AUDIO_FORMAT_CODE_ATRAC,
    CVI_HDMI_AUDIO_FORMAT_CODE_ONE_BIT,
    CVI_HDMI_AUDIO_FORMAT_CODE_DDP,
    CVI_HDMI_AUDIO_FORMAT_CODE_DTS_HD,
    CVI_HDMI_AUDIO_FORMAT_CODE_MAT,
    CVI_HDMI_AUDIO_FORMAT_CODE_DST,
    CVI_HDMI_AUDIO_FORMAT_CODE_WMA_PRO,
    CVI_HDMI_AUDIO_FORMAT_CODE_BUTT,
}CVI_HDMI_AUDIO_FORMAT_CODE;
```

【成员】

无

【注意事项】

目前仅支持 PCM 格式的音频

【相关数据类型及接口】

- CVI_HDMI_SetAttr

· CVI_HDMI_AUDIO_INFO

【说明】

定义 HDMI 音频能力集信息

【定义】

```
typedef struct _CVI_HDMI_AUDIO_INFO{
    CVI_HDMI_AUDIO_FORMAT_CODE audio_format_code;
    CVI_HDMI_SAMPLE_RATE support_sample_rate[CVI_HDMI_MAX_SAMPLE_
    ↪RATE_NUM];
    CVI_U32 support_sample_rate_num;
    CVI_U8 audio_chn;
    CVI_HDMI_BIT_DEPTH support_bit_depth[CVI_HDMI_MAX_BIT_DEPTH_
    ↪NUM];
    CVI_U32 support_bit_depth_num;
    CVI_U32 max_bit_rate;
} CVI_HDMI_AUDIO_INFO;
```

【成员】

成员名称	描述
audio_format_code	支持的音频格式。
support_sample_rate	支持的音频采样率。
support_sample_rate_num	支持的音频采样率的总数目。
audio_chn	支持的音频通道数。
support_bit_depth	支持的采样 bit 深度。
support_bit_depth_num	支持不同的 bit 深度种类个数。
max_bit_rate	最大比特率。

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_GetSinkCapability
- CVI_HDMI_SINK_CAPABILITY

· CVI_HDMI_TIMING_INFO**【说明】**

定义 HDMI DealingTiming 信息

【定义】

```
typedef struct _CVI_HDMI_TIMING_INFO{
    CVI_U32 vfb;
    CVI_U32 vbb;
    CVI_U32 vact;
    CVI_U32 hfb;
    CVI_U32 hbb;
    CVI_U32 hact;
```

(下页继续)

(续上页)

```

CVI_U32 vpw;
CVI_U32 hpw;
CVI_BOOL idv;
CVI_BOOL ihs;
CVI_BOOL ivs;
CVI_U32 img_width;
CVI_U32 img_height;
CVI_U32 aspect_ratio_w;
CVI_U32 aspect_ratio_h;
CVI_BOOL interlace;
CVI_U32 pixel_clk;
} CVI_HDMI_TIMING_INFO;

```

【成员】

成员名称	描述
vfb	垂直前消隐
vbb	垂直后消隐
vact	垂直活动区域
hfb	水平前消隐
hbb	水平后消隐
hact	水平活动区域
vpw	垂直同步宽度
hpw	水平同步宽度
idv	需要翻转数据有效信号标志
ihs	需要翻转横向同步标志
ivs	需要翻转垂直同步标志
img_width	图像宽度
img_height	图像高度
aspect_ratio_w	宽高比宽
aspect_ratio_h	宽高比高
interlace	隔行标志
pixel_clk	时序的像素时钟

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_GetSinkCapability
- CVI_HDMI_SINK_CAPABILITY

· CVI_HDMI_DETAIL_TIMING**【说明】**

定义 HDMI Detail Timing 能力集信息

【定义】

```
typedef struct _CVI_HDMI_DETAIL_TIMING{
    CVI_U32 detail_timing_num;
    CVI_HDMI_TIMING_INFO detail_timing[CVI_HDMI_DETAIL_TIMING_MAX];
}CVI_HDMI_DETAIL_TIMING;
```

【成员】

成员名称	描述
detail_timing_num	支持的详细时序个数。
detail_timing[CVI_HDMI_DETAIL_TIMING_MAX]	支持的详细时序信息。

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_GetSinkCapability
- CVI_HDMI_SINK_CAPABILITY

· CVI_HDMI_SINK_CAPABILITY

【说明】

定义 HDMI sink 端能力集结构体

【定义】

```
typedef struct _CVI_HDMI_SINK_CAPABILITY{
    CVI_BOOL is_connected;
    CVI_BOOL support_hdmi;
    CVI_BOOL is_sink_power_on;
    CVI_HDMI_VIDEO_FORMAT native_video_format;
    CVI_S32 support_video_format[CVI_HDMI_VIDEO_FORMAT_BUTT];
    CVI_BOOL support_ycbcr;
    CVI_BOOL support_xvycc601;
    CVI_BOOL support_xvycc709;
    CVI_U8 md_bit;
    CVI_U32 audio_info_num;
    CVI_BOOL hdcp14_en;
    CVI_HDMI_VIDEO_MODE hdmi_video_input;
    CVI_HDMI_VIDEO_MODE hdmi_video_output;
    CVI_HDMI_AUDIO_INFO audio_info[CVI_HDMI_MAX_AUDIO_CAPBILITY_
    →CNT];
    CVI_BOOL speaker[CVI_HDMI_AUDIO_SPEAKER_BUTT];
    CVI_U8 manufacture_name[CVI_HDMI_MANUFACTURE_NAME_LEN];
    CVI_U32 pdt_code;
    CVI_U32 serial_num;
    CVI_U32 week_of_manufacture;
    CVI_U32 year_of_manufacture;
```

(下页继续)

(续上页)

```

CVI_U8 version;
CVI_U8 revision;
CVI_U8 edid_ex_blk_num;
CVI_BOOL is_phy_addr_valid;
CVI_U8 phys_addr_a;
CVI_U8 phys_addr_b;
CVI_U8 phys_addr_c;
CVI_U8 phys_addr_d;
CVI_BOOL support_dvi_dual;
CVI_BOOL support_deepcolor_ycbcr444;
CVI_BOOL support_deep_color_30bit;
CVI_BOOL support_deep_color_36bit;
CVI_BOOL support_deep_color_48bit;
CVI_BOOL support_ai;
CVI_U32 max_tmnds_clk;
CVI_BOOL i_latency_fields_present;
CVI_BOOL latency_fields_present;
CVI_BOOL hdmi_video_present;
CVI_U8 video_latency;
CVI_U8 audio_latency;
CVI_U8 interlaced_video_latency;
CVI_U8 interlaced_audio_latency;
CVI_BOOL support_y420_dc_30bit;
CVI_BOOL support_y420_dc_36bit;
CVI_BOOL support_y420_dc_48bit;
CVI_BOOL support_hdmi_2_0;
CVI_BOOL support_y420_format[CVI_HDMI_VIDEO_FORMAT_BUTT];
CVI_BOOL only_support_y420_format[CVI_HDMI_VIDEO_FORMAT_BUTT];
CVI_BOOL ycc_quant_selectable;
CVI_BOOL rgb_quant_selectable;
CVI_HDMI_DETAIL_TIMING detailed_timing;
} CVI_HDMI_SINK_CAPABILITY;

```

【成员】

成员名称	描述
is_connected	设备是否连接。
support_hdmi	设备是否支持 HDMI (默认为 HDMI1.4)，如果不支持，则为 DVI 设备。
is_sink_power_on	Sink 设备是否上电。
native_video_format	显示设备物理分辨率。
support_video_format	视频能力集 CVI_TRUE: 支持这种显示格式; CVI_FALSE: 不支持。

下页继续

表 4.1 – 续上页

support_ycbcr	<p>是否支持 YCBCR 显示。</p> <ul style="list-style-type: none"> – CVI_TRUE: 支持 YCBCR 显示; – CVI_FALSE: 只支持 RGB 显示。
support_xvYcc601	是否支持 xvYCC601 颜色格式。
support_xvYcc709	是否支持 xvYCC709 颜色格式。
md_bit	xvYCC601 支持的传输 profile:1:P0,2:P1,4:P2
audio_info_num	支持的音频信息个数, 取值范围 [1~16]
audio_info	<p>支持的音频信息。最多支持 16 组数据。每组音频信息包括:</p> <ul style="list-style-type: none"> – 音频编码格式 – 采样率 – Channel 个数 – 采样 bit 深度及其个数 – 最大 bit 率。
speaker	扬声器位置。
manufacture_name	设备厂商标识。
pdt_code	设备 ID。
serial_num	设备序列号。
week_of_manufacture	设备生产日期 (周)。
version	设备版本号。
revision	设备子版本号。
edid_ex_blk_num	EDID 扩展块数目。
is_phys_addr_valid	CEC 物理地址是否有效标志。
phys_addr_a	CEC 物理地址 A。
phys_addr_b	CEC 物理地址 B。
phys_addr_c	CEC 物理地址 C。
phys_addr_d	CEC 物理地址 D。
support_dvi_dual	是否支持 DVI dual-link 操作。
support_deepcolor_ycbcr444	是否支持 YCBCR 4:4:4 Deep Color 模式。
support_deepcolor_30bit	是否支持 Deep Color 30bit 模式。
support_deepcolor_36bit	是否支持 Deep Color 36bit 模式。
support_deepcolor_48bit	是否支持 Deep Color 48bit 模式。
support_ai	是否支持 Supports_AI 模式。
max_tmnds_clk	最大 TMDS 时钟。
i_latency_fields_present	延时标志位。

下页继续

表 4.1 – 续上页

latency_fields_present	Video_Latency 和 Audio_Latency fields 是否存在。
hdmi_video_present	特殊的视频格式。
video_latency	视频延时。
audio_latency	音频延时。
interlaced_video_latency	隔行视频模式下的视频延时。
support_y420_dc_30bit	是否支持 YCbCr420 的 deep-color 30bit 模式。
support_y420_dc_36bit	是否支持 YCbCr420 的 deep-color 36bit 模式。
support_y420_dc_48bit	是否支持 YCbCr420 的 deep-color 48bit 模式。
support_hdmi_2_0	是否支持 HDMI2.0。
support_y420_format	支持 YCbCr420 的视频制式。
only_support_y420_format	只支持 YCbCr420 的视频制式。
ycc_quant_selectable	是否支持 YCbCr Quantization 可选。
rgb_quant_selectable	是否支持 RGB Quantization 可选。
support_hdr	Sink 是否支持 HDR。
hdr	Sink 支持 HDR 能力的详细信息。
detailed_timing	支持的详细时序能力集信息。

【注意事项】

能力上报不区分 HDMI1.4 和 HDMI2.0。

【相关数据类型及接口】

- CVI_HDMI_GetSinkCapability

· CVI_HDMI_EDID**【说明】**

定义 HDMI EDID 信息结构体。

【定义】

```
typedef struct _CVI_HDMI_EDID{
    CVI_BOOL edid_valid;
    CVI_U32 edid_len;
    CVI_U8 edid[256];
} CVI_HDMI_EDID;
```

【成员】

成员名称	描述
edid_valid	EDID 信息是否有效。
edid_len	EDID 信息数据长度。
edid[CVI_HDMI_EDID_RAW_DATA_LEN]	EDID 信息数据。

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_ForceGetEdid

· CVI_HDMI_INFOFRAME**【说明】**

定义 HDMI 信息帧结构体。

【定义】

```
typedef struct _CVI_HDMI_INFOFRAME{  
    CVI_HDMI_INFOFRAME_TYPE infoframe_type;  
    CVI_HDMI_INFOFRAME_UNIT infoframe_unit;  
} CVI_HDMI_INFOFRAME;
```

【成员】

成员名称	描述
infoframe_type	信息帧类型
infoframe_unit	信息帧单元 (内容)

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_INFOFRAME_UNIT**【说明】**

定义 HDMI 信息帧单元结构体。

【定义】

```
typedef struct _CVI_HDMI_INFOFRAME_UNIT{  
    CVI_HDMI_AVI_INFOFRAME avi_infoframe;  
    CVI_HDMI_AUDIO_INFOFRAME audio_infoframe;
```

(下页继续)

(续上页)

```
CVI_HDMI_VENDORSPEC_INFOFRAME vendor_spec_infoframe;  
} CVI_HDMI_INFOFRAME_UNIT;
```

【成员】

成员名称	描述
avi_infoframe	AVI 信息帧单元
audio_infoframe	AUDIO 信息帧单元
vendor_spec_infoframe	VENDOR 信息帧单元

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

CVI_HDMI_AVI_INFOFRAME**【说明】**

定义 HDMI AVI 信息帧（版本 2）单元结构体。

【定义】

```
typedef struct CVI_HDMI_AVI_INFOFRAME{  
    CVI_HDMI_VIDEO_FORMAT timing_mode;  
    CVI_HDMI_COLOR_SPACE color_space;  
    CVI_BOOL active_info_present;  
    CVI_HDMI_BAR_INFO bar_info;  
    CVI_HDMI_SCAN_INFO scan_info;  
    CVI_HDMI_COLORIMETRY colorimetry;  
    CVI_HDMI_EX_COLORIMETRY ex_colorimetry;  
    CVI_PIC_ASPECT_RATIO aspect_ratio;  
    CVI_HDMI_ACTIVE_ASPECT_RATIO active_aspect_ratio;  
    CVI_HDMI_PIC_SCALINE pic_scaling;  
    CVI_HDMI_RGB_QUANT_RANGE rgb_quant;  
    CVI_BOOL is_it_content;  
    CVI_HDMI_PIXEL_REPETITION pixel_repetition;  
    CVI_HDMI_CONTENT_TYPE content_type;  
    CVI_HDMI_YCC_QUANT_RANGE ycc_quant;  
    CVI_U16 line_n_end_of_top_bar;  
    CVI_U16 line_n_start_of_bcvl_bar;  
    CVI_U16 pixel_n_end_of_left_bar;  
    CVI_U16 pixel_n_start_of_right_bar;  
} CVI_HDMI_AVI_INFOFRAME;
```

【成员】

成员名称	描述
timing_mode	视频时序
color_space	颜色空间
active_info_present	Info 是否有效
bar_info	Bar 信息
scan_info	Scan 信息
colorimetry	色域
ex_colorimetry	扩展色域
aspect_ratio	图像宽高比
active_aspect_ratio	有效宽高比
pic_scaling	图像均衡
rgb_quant	RGB 量化
is_it_content	ITContent 是否有效
pixel_repetition	像素加倍
content_type	ITContent 类型
ycc_quant	YCC 量化
line_n_end_of_top_bar	TopBar 终止行数
line_n_start_of_bot_bar	BottomBar 开始行数
pixel_n_end_of_left_bar	LeftBar 终止像素数
pixel_n_start_of_right_bar	RightBar 开始像素数

【注意事项】

详细描述请参见《High-Definition Multimedia Interface Specification Version 1.4b》、《High-Definition Multimedia Interface Specification Version 2.0》、《CEA-861-D》与《CEA-861-F》

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_AUDIO_INFOFRAME**【说明】**

定义 HDMI AUDIO 信息帧（版本 1）单元结构体。

【定义】

```
typedef struct _CVI_HDMI_AUDIO_INFOFRAME{
    CVI_HDMI_AUDIO_CHN_CNT chn_cnt;
    CVI_HDMI_CODING_TYPE coding_type;
    CVI_HDMI_AUDIO_SAMPLE_SIZE sample_size;
    CVI_HDMI_AUDIO_SAMPLE_FREQ sampling_freq;
    CVI_U8 chn_alloc;
    CVI_HDMI_LEVEL_SHIFT_VAL level_shift;
    CVI_HDMI_LFE_PLAYBACK_LEVEL lfe_playback_level;
    CVI_BOOL down_mix_inhibit;
} CVI_HDMI_AUDIO_INFOFRAME;
```

【成员】

成员名称	描述
chn_cnt	音频通道个数
coding_type	音频格式
sample_size	音频采样深度（位宽）
sampling_freq	音频采样率
chn_alloc	Channel/Speaker 分配
level_shift	Level Shift Value, 左移值
lfe_playback_level	LFE playback level information, LFE 播放等级信息。
down_mix_inhibit	Down-mix Inhibit 标志位。

【注意事项】

详细描述请参见《High-Definition Multimedia Interface Specification Version 1.4b》、《High-Definition Multimedia Interface Specification Version 2.0》、《CEA-861-D》与《CEA-861-F》

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

CVI_HDMI_VENDORSPEC_INFOFRAME

【说明】

vendor 信息帧结构体。

【定义】

```
typedef struct _CVI_HDMI_VENDORSPEC_INFOFRAME{
    CVI_U8 data_len;
    CVI_U8 user_data[CVI_HDMI_VENDOR_USER_DATA_MAX_LEN];
} CVI_HDMI_VENDORSPEC_INFOFRAME;
```

【成员】

成员名称	描述
data_len	用户数据的长度
user_data	用户数据

【注意事项】

用户可以通过该接口设置自定数据，当 data_len 为 0 时停止发送用户数据。

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame

- CVI_HDMI_GetInfoFrame

· CVI_HDMI_COLOR_SPACE

【说明】

定义颜色空间枚举。

【定义】

```
typedef enum _CVI_HDMI_COLOR_SPACE{  
    CVI_HDMI_COLOR_SPACE_RGB888,  
    CVI_HDMI_COLOR_SPACE_YCBCR422,  
    CVI_HDMI_COLOR_SPACE_YCBCR444,  
    CVI_HDMI_COLOR_SPACE_YCBCR420,  
    CVI_HDMI_COLOR_SPACE_BUTT,  
} CVI_HDMI_COLOR_SPACE;
```

【成员】

无

【注意事项】

CVI_HDMI_COLOR_SPACE_YCBCR420 为 HDMI2.0 规格，仅支持 HDMI1.4 规格的产品不支持此颜色空间。

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_BAR_INFO

【说明】

定义 Bar 信息枚举。

【定义】

```
typedef enum _CVI_HDMI_BAR_INFO{  
    CVI_HDMI_BAR_INFO_NCVI_VALID,  
    CVI_HDMI_BAR_INFO_V,  
    CVI_HDMI_BAR_INFO_H,  
    CVI_HDMI_BAR_INFO_VH,  
    CVI_HDMI_BAR_INFO_BUTT,  
} CVI_HDMI_BAR_INFO;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_SCAN_INFO

【说明】

定义 Scan 信息枚举。

【定义】

```
typedef enum _CVI_HDMI_SCAN_INFO{  
    CVI_HDMI_SCAN_INFO_NO_DATA,  
    CVI_HDMI_SCAN_INFO_OVERSCANNED,  
    CVI_HDMI_SCAN_INFO_UNDERSCANNED,  
    CVI_HDMI_SCAN_INFO_BUTT,  
} CVI_HDMI_SCAN_INFO;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_COLORIMETRY

【说明】

定义 Colorimetry 信息枚举。

【定义】

```
typedef enum _CVI_HDMI_COLORIMETRY{  
    CVI_HDMI_COMMON_COLORIMETRY_NO_DATA,  
    CVI_HDMI_COMMON_COLORIMETRY_ITU601,  
    CVI_HDMI_COMMON_COLORIMETRY_ITU709,  
    CVI_HDMI_COMMON_COLORIMETRY_BUTT,  
} CVI_HDMI_COLORIMETRY;
```

【成员】

无

【注意事项】

CVI_HDMI_COMMON_COLORIMETRY_BUTT 值有效，其含义表示设置为扩展色域，详细描述请参见《CEA-861-D》。

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_EX_COLORIMETRY

【说明】

定义扩展 Colorimetry 信息枚举。

【定义】

```
typedef enum CVI_HDMI_EX_COLORIMETRY{
    CVI_HDMI_COMMON_COLORIMETRY_XVYCC_601,
    CVI_HDMI_COMMON_COLORIMETRY_XVYCC_709,
    CVI_HDMI_COMMON_COLORIMETRY_S_YCC_601,
    CVI_HDMI_COMMON_COLORIMETRY_ADOBE_YCC_601,
    CVI_HDMI_COMMON_COLORIMETRY_ADOBE_RGB,
    CVI_HDMI_COMMON_COLORIMETRY_2020_CONST_LUMINOUS,
    CVI_HDMI_COMMON_COLORIMETRY_2020_NON_CONST_LUMINOUS,
    CVI_HDMI_COMMON_COLORIMETRY_EX_BUTT,
} CVI_HDMI_EX_COLORIMETRY;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

CVI_HDMI_ACTIVE_ASPECT_RATIO

【说明】

定义实际图像宽高比枚举。

【定义】

```
typedef enum CVI_HDMI_ACTIVE_ASPECT_RATIO{
    CVI_HDMI_ACTIVE_ASPECT_RATIO_16TO9_TOP = 2,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_14TO9_TOP,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_16TO9_BOX_CENTER,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_SAME_PIC = 8,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_4TO3_CENTER,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_16TO9_CENTER,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_14TO9_CENTER,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_4TO3_14_9 = 13,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_16TO9_14_9,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_16TO9_4_3,
    CVI_HDMI_ACTIVE_ASPECT_RATIO_BUTT,
} CVI_HDMI_ACTIVE_ASPECT_RATIO;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame

- CVI_HDMI_GetInfoFrame

· CVI_HDMI_PIC_SCALINE

【说明】

定义图像扫描信息枚举。

【定义】

```
typedef enum _CVI_HDMI_PIC_SCALINE{  
    CVI_HDMI_PICTURE_NON_UNIFORM_SCALING,  
    CVI_HDMI_PICTURE_SCALING_H,  
    CVI_HDMI_PICTURE_SCALING_V,  
    CVI_HDMI_PICTURE_SCALING_HV,  
    CVI_HDMI_PICTURE_SCALING_BUTT,  
} CVI_HDMI_PIC_SCALINE;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_RGB_QUANT_RANGE

【说明】

定义 RGB 量化范围枚举。

【定义】

```
typedef enum _CVI_HDMI_RGB_QUANT_RANGE{  
    CVI_HDMI_RGB_QUANT_DEFAULT_RANGE,  
    CVI_HDMI_RGB_QUANT_LIMITED_RANGE,  
    CVI_HDMI_RGB_QUANT_FULL_RANGE,  
    CVI_HDMI_RGB_QUANT_FULL_BUTT,  
} CVI_HDMI_RGB_QUANT_RANGE;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_PIXEL_REPETITION

【说明】

定义像素复制次数枚举。

【定义】

```
typedef enum CVI_HDMI_PIXEL_REPETITION{  
    CVI_HDMI_PIXEL_REPET_NO,  
    CVI_HDMI_PIXEL_REPET_2_TIMES,  
    CVI_HDMI_PIXEL_REPET_3_TIMES,  
    CVI_HDMI_PIXEL_REPET_4_TIMES,  
    CVI_HDMI_PIXEL_REPET_5_TIMES,  
    CVI_HDMI_PIXEL_REPET_6_TIMES,  
    CVI_HDMI_PIXEL_REPET_7_TIMES,  
    CVI_HDMI_PIXEL_REPET_8_TIMES,  
    CVI_HDMI_PIXEL_REPET_9_TIMES,  
    CVI_HDMI_PIXEL_REPET_10_TIMES,  
    CVI_HDMI_PIXEL_REPET_BUTT,  
} CVI_HDMI_PIXEL_REPETITION;
```

【成员】

无

【注意事项】

设置枚举以外的值将返回 CVI_ERR_HDMI_INVALID_PARA。

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_CONTENT_TYPE

【说明】

定义内容信息枚举

【定义】

```
typedef enum _CVI_HDMI_CONTENT_TYPE{  
    CVI_HDMI_CONTNET_GRAPHIC,  
    CVI_HDMI_CONTNET_PHOTO,  
    CVI_HDMI_CONTNET_CINEMA,  
    CVI_HDMI_CONTNET_GAME,  
    CVI_HDMI_CONTNET_BUTT,  
} CVI_HDMI_CONTENT_TYPE;
```

【成员】

无

【注意事项】

设置枚举以外的值将返回 CVI_ERR_HDMI_INVALID_PARA。

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame

- CVI_HDMI_GetInfoFrame

· CVI_HDMI_YCC_QUANT_RANGE

【说明】

定义 YCC 量化范围枚举

【定义】

```
typedef enum _CVI_HDMI_YCC_QUANT_RANGE{  
    CVI_HDMI_YCC_QUANT_LIMITED_RANGE,  
    CVI_HDMI_YCC_QUANT_FULL_RANGE,  
    CVI_HDMI_YCC_QUANT_BUTT,  
} CVI_HDMI_YCC_QUANT_RANGE;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_AUDIO_CHN_CNT

【说明】

定义 YCC 量化范围枚举

【定义】

```
typedef enum _CVI_HDMI_AUDIO_CHN_CNT{  
    CVI_HDMI_AUDIO_CHANEL_CNT_STREAM,  
    CVI_HDMI_AUDIO_CHANEL_CNT_2,  
    CVI_HDMI_AUDIO_CHANEL_CNT_3,  
    CVI_HDMI_AUDIO_CHANEL_CNT_4,  
    CVI_HDMI_AUDIO_CHANEL_CNT_5,  
    CVI_HDMI_AUDIO_CHANEL_CNT_6,  
    CVI_HDMI_AUDIO_CHANEL_CNT_7,  
    CVI_HDMI_AUDIO_CHANEL_CNT_8,  
    CVI_HDMI_AUDIO_CHANEL_BUTT,  
} CVI_HDMI_AUDIO_CHN_CNT;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame

- CVI_HDMI_GetInfoFrame

· CVI_HDMI_CODING_TYPE

【说明】

定义音频编码类型枚举。

【定义】

```
typedef enum CVI_HDMI_CODING_TYPE{  
    CVI_HDMI_AUDIO_CODING_REFER_STREAM_HEAD,  
    CVI_HDMI_AUDIO_CODING_PCM,  
    CVI_HDMI_AUDIO_CODING_AC3,  
    CVI_HDMI_AUDIO_CODING_MPEG1,  
    CVI_HDMI_AUDIO_CODING_MP3,  
    CVI_HDMI_AUDIO_CODING_MPEG2,  
    CVI_HDMI_AUDIO_CODING_AACLC,  
    CVI_HDMI_AUDIO_CODING_DTS,  
    CVI_HDMI_AUDIO_CODING_ATRAC,  
    CVI_HDMI_AUDIO_CODING_ONE_BIT_AUDIO,  
    CVI_HDMI_AUDIO_CODING_ENAHCED_AC3,  
    CVI_HDMI_AUDIO_CODING_DTS_HD,  
    CVI_HDMI_AUDIO_CODING_MAT,  
    CVI_HDMI_AUDIO_CODING_DST,  
    CVI_HDMI_AUDIO_CODING_WMA_PRO,  
    CVI_HDMI_AUDIO_CODING_BUTT,  
} CVI_HDMI_CODING_TYPE;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_AUDIO_SAMPLE_SIZE

【说明】

定义音频采样大小枚举。

【定义】

```
typedef enum CVI_HDMI_AUDIO_SAMPLE_SIZE{  
    CVI_HDMI_AUDIO_SAMPLE_SIZE_STREAM,  
    CVI_HDMI_AUDIO_SAMPLE_SIZE_16 = 16,  
    CVI_HDMI_AUDIO_SAMPLE_SIZE_20 = 20,  
    CVI_HDMI_AUDIO_SAMPLE_SIZE_24 = 24,  
    CVI_HDMI_AUDIO_SAMPLE_SIZE_BUTT,  
} CVI_HDMI_AUDIO_SAMPLE_SIZE;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_AUDIO_SAMPLE_FREQ**【说明】**

定义音频采样频率枚举。

【定义】

```
typedef enum CVI_HDMI_AUDIO_SAMPLE_FREQ{  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_STREAM,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_32000 = 32000,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_44100 = 44100,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_48000 = 48000,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_88200 = 88200,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_96000 = 96000,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_176400 = 176400,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_192000 = 192000,  
    CVI_HDMI_AUDIO_SAMPLE_FREQ_BUTT,  
} CVI_HDMI_AUDIO_SAMPLE_FREQ;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_LEVEL_SHIFT_VAL**【说明】**

定义音频 Shift 信息枚举

【定义】

```
typedef enum CVI_HDMI_LEVEL_SHIFT_VAL{  
    CVI_HDMI_LEVEL_SHIFT_VALUE_0_DB,  
    CVI_HDMI_LEVEL_SHIFT_VALUE_1_DB,  
    CVI_HDMI_LEVEL_SHIFT_VALUE_2_DB,  
    CVI_HDMI_LEVEL_SHIFT_VALUE_3_DB,  
    CVI_HDMI_LEVEL_SHIFT_VALUE_4_DB,  
    CVI_HDMI_LEVEL_SHIFT_VALUE_5_DB,  
}
```

(下页继续)

(续上页)

```

CVI_HDMI_LEVEL_SHIFT_VALUE_6_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_7_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_8_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_9_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_10_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_11_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_12_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_13_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_14_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_15_DB,
CVI_HDMI_LEVEL_SHIFT_VALUE_BUTT,
} CVI_HDMI_LEVEL_SHIFT_VAL;

```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_LFE_PLAYBACK_LEVEL**【说明】**

定义音频 Playback 信息枚举。

【定义】

```

typedef enum CVI_HDMI_LFE_PLAYBACK_LEVEL{
    CVI_HDMI_LFE_PLAYBACK_NO,
    CVI_HDMI_LFE_PLAYBACK_0_DB,
    CVI_HDMI_LFE_PLAYBACK_10_DB,
    CVI_HDMI_LFE_PLAYBACK_BUTT,
} CVI_HDMI_LFE_PLAYBACK_LEVEL;

```

【成员】

无

【注意事项】

设置枚举以外的值将返回 CVI_ERR_HDMI_INVALID_PARA。

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_HW_PARAM**【说明】**

HDMI 指标参数结构体。

【定义】

```
typedef struct _CVI_HDMI_HW_PARAM{
    CVI_U32 i_de_main_clk;
    CVI_U32 i_de_main_data;
    CVI_U32 i_main_clk;
    CVI_U32 i_main_data;
    CVI_U32 ft_cap_clk;
    CVI_U32 ft_cap_data;
} CVI_HDMI_HW_PARAM;
```

【成员】

成员名称	描述
i_de_main_clk	clk 线的辅助区电流。
i_de_main_data	data 线的辅助区电流。
i_main_clk	clk 线的主区电流。
i_main_data	data 线的主区电流。
ft_cap_clk	clk 线的上升沿。
ft_cap_data	data 线的上升沿。

【注意事项】

目前该接口未实现

【相关数据类型及接口】

- CVI_HDMI_SetHwSpec
- CVI_HDMI_GetHwSpec

· CVI_HDMI_HW_SPEC

【说明】

HDMI 指标参数。

【定义】

```
typedef struct _CVI_HDMI_HW_SPEC{
    CVI_HDMI_HW_PARAM hw_param[CVI_HDMI_HW_PARAM_NUM];
} CVI_HDMI_HW_SPEC;
```

【成员】

成员名称	描述
hw_param[CVI_HDMI_HW_PARAM_NUM]	指标参数。

【注意事项】

目前该接口未实现

【相关数据类型及接口】

- CVI_HDMI_SetHwSpec
- CVI_HDMI_HW_PARAM

· CVI_HDMI_MOD_PARAM**【说明】**

HDMI 模块参数设置。

【定义】

```
typedef struct _CVI_HDMI_MOD_PARAM{  
    CVI_HDMI_TRACE_LEN trace_len;  
    CVI_BOOL emi_en;  
} CVI_HDMI_MOD_PARAM;
```

【成员】

成员名称	描述
trace_len	HDMI 信号线在 PCB 板上的走线长度。
emi_en	展频使能开关

【注意事项】

目前该接口未实现

· CVI_HDMI_TRACE_LEN**【说明】**

HDMI 信号线在 PCB 板上的走线长度枚举。

- CVI_HDMI_TRACE_LEN_0 表示 1 英寸，后续每个档位按照 0.5 英寸递增，即 CVI_HDMI_TRACE_LEN_1 表示 1.5 英寸。
- CVI_HDMI_TRACE_DEFAULT 表示默认配置。

【定义】

```
typedef enum _CVI_HDMI_TRACE_LEN{  
    CVI_HDMI_TRACE_LEN_0,  
    CVI_HDMI_TRACE_LEN_1,  
    CVI_HDMI_TRACE_LEN_2,  
    CVI_HDMI_TRACE_LEN_3,  
    CVI_HDMI_TRACE_LEN_4,  
    CVI_HDMI_TRACE_LEN_5,  
    CVI_HDMI_TRACE_LEN_6,  
}
```

(下页继续)

(续上页)

```
CVI_HDMI_TRACE_LEN_7,  
CVI_HDMI_TRACE_LEN_8,  
CVI_HDMI_TRACE_DEFAULT,  
CVI_HDMI_TRACE_BUTT  
} CVI_HDMI_TRACE_LEN;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_MOD_PARAM

· CVI_HDMI_VENDOR_USER_DATA_MAX_LEN**【说明】**

最大自定义 vendor 数据的长度。

【定义】

#define CVI_HDMI_VENDOR_USER_DATA_MAX_LEN 22

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_MAX_AUDIO_CAPBILITY_CNT**【说明】**

能获取的音频能力个数。

【定义】

#define CVI_HDMI_MAX_AUDIO_CAPBILITY_CNT 16

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_MAX_SAMPLE_RATE_NUM**【说明】**

音频采样率个数。

【定义】

#define CVI_HDMI_MAX_SAMPLE_RATE_NUM 8

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_MAX_BIT_DEPTH_NUM**【说明】**

音频采样深度个数。

【定义】

```
#define CVI_HDMI_MAX_SAMPLE_RATE_NUM 6
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_DETAIL_TIMING_MAX**【说明】**

详细的视频制式信息个数。

【定义】

```
#define CVI_HDMI_DETAIL_TIMING_MAX 10
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_EDID_RAW_DATA_LEN**【说明】**

EDID 原始数据的长度。

【定义】

```
#define CVI_HDMI_EDID_RAW_DATA_LEN 512
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_HW_PARAM_NUM**【说明】**

硬件参数的个数。

【定义】

```
#define CVI_HDMI_HW_PARAM_NUM 4
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· **CVI_HDMI_MANUFACTURE_NAME_LEN**

【说明】

制造商名称的长度。

【定义】

```
#define CVI_HDMI_MANUFACTURE_NAME_LEN 4
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· **CVI_HDMI_AUDIO_SPEAKER_BUTT**

【说明】

扬声器最大编号。

【定义】

```
#define CVI_HDMI_AUDIO_SPEAKER_BUTT 1
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· **CVI_HDMI_ID**

【说明】

定义 HDMI 接口号枚举

【定义】

```
typedef enum _CVI_HDMI_ID {
    CVI_HDMI_ID_0 = 0, /* First HDMI device */
    CVI_HDMI_ID_BUTT, /* Invalid HDMI device ID */
} CVI_HDMI_ID;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

无

· CVI_HDMI_FORCE_ACTION

【说明】

定义 HDMI 强制动作枚举

【定义】

```
typedef enum _CVI_HDMI_FORCE_ACTION {
    CVI_HDMI_FORCE_NULL,          /* No force action */
    CVI_HDMI_FORCE_HDMI,         /* Force HDMI mode */
} CVI_HDMI_FORCE_ACTION;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetAttr

· CVI_HDMI_INFOFRAME_TYPE

【说明】

定义 HDMI 信息帧类型枚举

【定义】

```
typedef enum _CVI_HDMI_INFOFRAME_TYPE {
    CVI_INFOFRAME_TYPE_AVI,          /* AVI infoframe type */
    CVI_INFOFRAME_TYPE_AUDIO,        /* Audio infoframe type */
    CVI_INFOFRAME_TYPE_VENDORSPEC,   /* Vendor specific infoframe_
↪type */
    CVI_INFOFRAME_TYPE_BUTT,         /* Invalid infoframe type */
} CVI_HDMI_INFOFRAME_TYPE;
```

【成员】

无

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_SetInfoFrame
- CVI_HDMI_GetInfoFrame

· CVI_HDMI_VIDEO_INFO

【说明】

定义 HDMI 视频信息结构体

【定义】

```
typedef struct CVI_HDMI_VIDEO_INFO {  
    CVI_HDMI_VIDEO_FORMAT mcode; /* Video format code */  
    CVI_HDMI_TIMING_INFO timing_info; /* Detailed timing parameters */  
    CVI_U32 fresh_rate; /* Refresh rate in Hz */  
} CVI_HDMI_VIDEO_INFO;
```

【成员】

成员名称	描述
mcode	视频格式代码
timing_info	详细时序参数
fresh_rate	刷新率 (Hz)

【注意事项】

无

【相关数据类型及接口】

- CVI_HDMI_VIDEO_FORMAT
- CVI_HDMI_TIMING_INFO

5 错误码

HDMI API 错误码如表 4-1 所示。

表 5.1: HDMI API 错误码

错误代码	宏定义	描述
0xa0288001	CVI_ERR_HDMI_NOT_INIT	HDMI 未初始化
0xa0288002	CVI_ERR_HDMI_INVALID_PARA	参数非法
0xa0288003	CVI_ERR_HDMI_NULL_PTR	空指针
0xa0288004	CVI_ERR_HDMI_DEV_NOT_OPEN	HDMI 未打开
0xa0288005	CVI_ERR_HDMI_DEV_NOT_CONNECT	HDMI 设备未连接
0xa0288006	CVI_ERR_HDMI_READ_SINK_FAILED	HDMI 读取 Sink 端失败
0xa0288007	CVI_ERR_HDMI_INIT_ALREADY	HDMI 已经初始化
0xa0288008	CVI_ERR_HDMI_CALLBACK_ALREADY	HDMI 回调已注册
0xa0288009	CVI_ERR_HDMI_INVALID_CALLBACK	回调函数无效
0xa028800a	CVI_ERR_HDMI_FEATURE_NO_SUPPORT	功能不支持
0xa028800b	CVI_ERR_HDMI_FEATURE_NO_SUPPORT	总线忙
0xa028800c	CVI_ERR_HDMI_READ_EVENT_FAILED	事件读失败
0xa028800d	CVI_ERR_HDMI_NOT_START	HDMI 没有启动
0xa028800e	CVI_ERR_HDMI_READ_EDID_FAILED	HDMI 读取 EDID 失败
0xa028800f	CVI_ERR_HDMI_INIT_FAILED	HDMI 初始化失败
0xa0288010	CVI_ERR_HDMI_CREATE_TESK_FAILED	HDMI 内核创建任务失败
0xa0288011	CVI_ERR_HDMI_MALLOC_FAILED	HDMI 分配内存失败
0xa0288012	CVI_ERR_HDMI_FREE_FAILED	HDMI 释放内存失败
0xa0288013	CVI_ERR_HDMI_PTHREAD_CREATE_FAILED	HDMI 创建线程失败
0xa0288014	CVI_ERR_HDMI_PTHREAD_JOIN_FAILED	HDMI 等待线程结束失败
0xa0288015	CVI_ERR_HDMI_STRATEGY_FAILED	HDMI 内核自适应策略失败
0xa0288016	CVI_ERR_HDMI_SET_ATTR_FAILED	HDMI 设置属性失败
0xa0288017	CVI_ERR_HDMI_CALLBACK_NOT_REGISTER	HDMI 回调没有注册
0xa0288018	CVI_ERR_HDMI_UNKNOWN_COMMAND	HDMI 未知命令
0xa0288019	CVI_ERR_HDMI_MUTEX_LOCK_FAILED	HDMI 加锁失败

6 Proc 调试信息

· 【HDMI 软硬件总体状态信息】

须知: 处理器不支持的规格, 该部分 proc 信息无效。

cat /proc/soph/hdmi	
HDMI version: 2.0	
----- app attr -----	
hdmie enable	audio enable
Yes	NO
auth mode enable	deep color mode
Reserved	0
deep color adapt	
NO	
----- sw status -----	
----- HW status -----	
hot plug	RX sense
Yes	Yes
phy output enable	phy power enable
Yes	NO
TMDS mode	HDMI avmute
HDMI1.4	NO
emi enable	src_scramble
--	NO
sink_scramble	tmds_bit_clk_ratio
NO	1/10
----- Detect Timming -----	
SYNC sw enable	progressive
YES	NO
HSYNC polarity	VSYNC polarity
Yes	YES
HSYNC total	VSYNC total
2200	1125

(下页继续)

(续上页)

Hactive 1920	Vactive 1080
-----------------	-----------------

【调试信息分析】记录 HDMI 输出管理模块信息。

【参数说明】

parameter	Description
hdmie enable	是否使能 HDMI 模式。取值：{YES, NO}
audio enable	当前用户是否使能音频。取值：{YES, NO}
deep color enable	当前用户是否设置的图像色彩深度。取值：{24, 30, 36, 48, OFF, UNKNOWN}
deep color adapt	当前用户是否使能 DeepColor 自适应策略。取值：{YES, NO}
auth mode anable	是否使能认证模式。取值：{YES, NO}
hot plug	热插拔检测。取值：{YES, NO}
emi enable	是否使能展频功能。取值：{YES, NO}。
RX sense	硬件 Rsen 状态。取值：{YES, NO}
phy output enable	HDMI Phy 是否已打开。取值：{YES, NO}
phy power enable	HDMI Phy 是否已上电。取值：{YES, NO}
tmds mode	当前硬件 TMDS 工作模式。取值：{NONE, DVI, HDMI1.4, HDMI2.0, AUTO, UNKNOWN}
HDMI avmute	当前是否使能音视频 Mute（显示端）。取值：{YES, NO}
src_scramble	源端加扰状态。
sink_scramble	sink 端加扰状态
tmds_bit_clk_ratio	源端 tmds 时钟速率。
sync sw enable	是否使用软件极性配置。取值：{YES, NO}
hsync polarity	Hsync 极性逻辑检测值。取值：{P, N}
progressive	当前是否逐行输出。取值：{YES, NO}
vsync polarity	Vsync 极性逻辑检测值。取值：{P, N}
hsync total	行总像素点数逻辑检测值。
hactive	行有效像素点逻辑检测值。
vsync total	场总行数逻辑检测值。
vactive	场有效行数逻辑检测值。

【注意事项】

- N/A/NONE/UNKNOWN 为未知/无效；reserved 为保留/未知；ERROR 为错误。
- spec info 的 stage 栏中 first、second、third、fourth 分别表示四个频段。
- 注意 avmute 与下面的 audio mute、video mute 的区别：avmute 是由 TX 发送给

RX 的一个标准的协议包，RX 在接收到该数据包后会自动 mute 自身的音视频输出。proc 里的 avmute 状态表示的是 TX 是否有发送 avmute 包。audio mute、video mute 是 TX 实现的一个功能，主要目的是用于调试的场景，其状态反应的是 TX 音视频数据的输出情况。

【HDMI 音频调试信息】

```
# cat /proc/soph/hdmi_audio
HDMI version: 2.0
----- audio attr ----- audio info ---
      sound intf              audio info enable
      AHB-DMA                  YES

      code type                code type
      PCM                      PCM

      channel cnt              channel cnt
      1                        1

      sample freq              sample freq
      44100                    44100

      sample depth             sample depth
      24                       24

      down sample              sample size
      NO                        STR_HEADER
----- audio path -----
      audio enable              level shift value
      Yes                       0dB

      audio mute                lfe playback
      NO                         UNKNOWN

      sound intf              channel/speakeralloc
      AHB-DMA                  0

      channel cnt              audio info raw data
      1

      sample freq              sample depth
      44100                    24

      down sample              down mix inhibit
      NO                        NO

      cts value                n value
      165000                    6272
```

【注意事项】

- N/A/NONE/UNKNOWN 为未知/无效；reserved 为保留/未知；ERROR 为错误。
- spec info 的 stage 栏中 first、second、third、fourth 分别表示四个频段。
- avmute 是由 TX 发送给 RX 的一个标准的协议包，RX 在接收到该数据包后会自动 mute 自身的音视频输出。proc 里的 avmute 状态表示的是 TX 是否有发送 avmute 包。
- audio mute、video mute 是 TX 实现的一个功能，主要目的是用于调试的场景，其状态反应的是 TX 音视频数据的输出情况。

【调试信息分析】

记录当前 HDMI 音频工作状态。

【参数说明】

parameter	Description
sound intf	用户配置的 HDMI 音频接口类型。取值: {I2S, SPDIF, HBRA, UNKNOWN}
sample depth	采样深度 (位宽)
down sample	是否降采样。取值: {YES, NO}
audio info enable	音频信息帧是否使能。取值: {YES, NO}
code type	音频编码类型。取值: {STREAM, L-PCM, AC3, MPEG1, MP3, MPEG2, AAC_LC, DTS, ATRAC, OneBitAudio, EAC3, DTS-HD, MAT, DST, WMA_PRO, Reserved, UNKNOWN}
channel cnt	音频通道数。取值: {STR_HEADER, 2_CH, 3_CH, 4_CH, 5_CH, 6_CH, 7_CH, 8_CH}
sample freq	音频采样频率。取值: {STR_HEADER, 32KHz, 44.1KHz, 48KHz, 88.2KHz, 96KHz, 176.4KHz, 192KHz}
sample size	采样大小 (位宽)。取值: {STR_HEADER, 16bit, 20bit, 24bit}
down mix inhibit	Down-mix Inhibit 标志位。取值: {YES, NO}
level shift value	Level Shift Value。取值: {0~15 dB}
lfe playback	LFE playback level information, LFE 播放等级信息。
channel/speakeralloc	Channel/Speaker 分配。取值: {0x00~0xff}
audio info raw data	AudioInfoframe 原始数据
audio enable	HDMI 硬件音频是否使能。取值: {YES, NO}
audio mute	音频 Mute 是否使能。取值: {YES, NO}
sound intf	音频接口类型。取值: {I2S, SPDIF, HBRA, UNKNOWN}
cts value	实际输出的 CTS 值
n value	实际输出 N 值。

【注意事项】

- N/A/NONE/UNKNOWN 为未知/无效; reserved 为保留/未知; ERROR 为错误。

【HDMI 软硬件总体状态信息】

```
# cat /proc/soph/hdmi_video
HDMI version: 2.0
----- video attr ----- avi info -----
                video timing                avi infoframe enable
```

(下页继续)

(续上页)

1920*1080p60 16:9	YES
disp fmt 1080P@60	current format 1920*1080p60 16:9(vic=16)
pixel clk 148500	vsif format Reserved
in bit depth 8 bit	bar data present NONE
in color space RGB888	color space RGB888
colorimetry ITU-R BT.709	pixel repeat 0
pic aspect ratio 16: 9	pic aspect ratio Reserved
ycc quantization LIMITED	Vactive DEFAULT
ext colorimetry Reserved	ext colorimetry Reserved
stereo mode NONE	it content valid NO
vsync pol 1	hsync pol 1
----- video path -----	
video mute NO	pic scaling UNKNOWN
out bit depth 8	act fmt present Reserved
out color space RGB888	scan info NONE
it content type GRAPHICS	ycbcr422_444 NO
ycbcr420_422 NO	ycbcr444_422 NO
ycbcr422_420 NO	rgb2ycbcr NO
ycbcr2rgb NO	

(下页继续)

(续上页)

dither Reserved	deep color mode 24 bit(OFF)
avi info raw data Reserved	vsif info raw data Reserved

【调试信息分析】

记录当前 HDMI 视频工作状态。

【参数说明】

parameter	Description
video timing	当前视频时序。
disp fmt	当前视频制式。
pixel clk	像素时钟。
in bit depth	VO 输出给 HDMI 视频色深。取值：{8bit, 10bit, 12bit, 16bit, UNKNOWN}
in color space	VO 输出给 HDMI 的颜色空间。取值：{RGB, YCbCr422, YCbCr444, YCbCr420, UNKNOWN}
colorimetry	VO 输出给 HDMI 的色域。取值：{No Data, SMPTE 170M, ITU-R BT.709, Extended }
pic aspect ratio	输入视频的宽高比。取值：{ NONE, 4:3, 16:9, FUTURE, UNKNOWN}
act aspect ratio	输入视频有效图像的宽高比。取值：{ Reserved, 16:9_TOP, 14:9_TOP, 16:9_CENTER, PICTURE, 4:3, 16:9, 14:9, 4:3_SP_14_9, 16:9_SP_14_9, 16:9_SP_4_3, UNKNOWN}
pixel repeat	像素 Repeat 次数。
ycc quantization	YCC 量化范围。取值：{LIMITED, FULL, UNKNOWN}。
ext colorimetry	输入图像扩展色域。取值：{XV_YCC601, XV_YCC709, S_YCC601, ADOBE_YCC601, ADOBE_RGB, BT2020_cYCC, BT2020_RGB/YCC, UNKNOWN}
stereo mode	StereoMode 取值：{FRAME_PACK, FIELD_ALTER, LINE_ALTERN, SBS_FULL, L_DEPTH, L_DEPTH_GGD, TAndB, Reserved, SByS_HALF, NONE}
vsync pol	HDMI 是否使能垂直反向。取值：{0, 1}

下页继续

表 6.1 – 续上页

video mute	HDMI 硬件是否使能视频 Mute。取值：{YES, NO}
out bit depth	输出视频色深。取值：{8bit, 10bit, 12bit, 16bit, UNKNOWN}
out color space	输出图像颜色空间取值：{RGB, YCbCr422, YCbCr444, YCbCr420, UNKNOWN}
ycbcr420_422	颜色空间是否有进行 YCbCr420 到 YCbCr422 转换取值：{YES, NO}
ycbcr422_444	颜色空间是否有进行 YCbCr422 到 YCbCr444 转换取值：{YES, NO}
ycbcr444_422	颜色空间是否有进行 YCbCr444 到 YCbCr422 转换取值：{YES, NO}
ycbcr422_420	颜色空间是否有进行 YCbCr422 到 YCbCr420 转换取值：{YES, NO}
rgb2ycbcr	颜色空间是否有进行 RGB 到 YCbCr 的转换。取值：{YES, NO}
ycbcr2rgb	颜色空间是否有进行 YCbCr 到 RGB 的转换。取值：{YES, NO}
dither	Dither 工作模式。取值：{12_10, 12_8, 10_8, disable}
deep color mode	逻辑工作的 DeepColor 模式。目前仅支持 8bit 模式
avi infoframe enable	AVI 信息帧是否使能。取值：{YES, NO}
current format	当前视频制式/VIC 码。
vsif format	4k 非 3D 制式/VIC 码。
bar data present	Bar 信息。取值：{NONE, HnVp, HpVn, HpVp}
color space	视频颜色空间。取值：{RGB, YCbCr422, YCbCr444, YCbCr420, UNKNOWN}
pixel repeat	像素 Repeat 次数。
vactive	视频接口使能情况（一般是默认值）
it content valid	IT Content 是否有效。取值：{YES, NO}
hsync pol	是否使能水平反向。取值：{0, 1}
pic scaling	图像均衡。取值：{UNKNOWN, HpVn, HnVp, HpVp}
act fmt present	有效信息是否显示。取值：{YES, NO}
scan info	Scan 标志位取值：{NONE, OVER_SCAN, UNDERS_SCAN, Reserved}
avi info raw data	AVI 信息帧原始数据。
vsif info raw data	VSIF 信息帧原始数据

【EDID 调试信息】

```
# cat /proc/soph/hdmi_sink
HDMI version: 2.0
```

(下页继续)

(续上页)

----- edid raw data -----	
/* 00H: */ 0x00, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x10, 0xac, 0x40, 0xd1, 0x4c, 0x4a, 0x33, ↵ ↵0x30,	
/* 0fH: */ 0x30, 0x20, 0x01, 0x03, 0x80, 0x3c, 0x22, 0x78, 0xea, 0xd7, 0x35, 0xab, 0x53, 0x4f, ↵ ↵0x9e, 0x25,	
/* 1fH: */ 0x0f, 0x50, 0x54, 0xa5, 0x4b, 0x00, 0x81, 0x00, 0xb3, 0x00, 0xd1, 0x00, 0x71, 0x4f, ↵ ↵0xa9, 0x40,	
/* 2fH: */ 0x81, 0x80, 0xd1, 0xc0, 0x01, 0x01, 0x56, 0x5e, 0x00, 0xa0, 0xa0, 0xa0, 0x29, 0x50, ↵ ↵0x30, 0x20,	
/* 3fH: */ 0x35, 0x00, 0x55, 0x50, 0x21, 0x00, 0x00, 0x1a, 0x00, 0x00, 0x00, 0xff, 0x00, 0x34, ↵ ↵0x44, 0x48,	
/* 4fH: */ 0x57, 0x56, 0x50, 0x33, 0x0a, 0x20, 0x20, 0x20, 0x20, 0x20, 0x00, 0x00, 0x00, 0xfc, ↵ ↵0x00, 0x44,	
/* 5fH: */ 0x45, 0x4c, 0x4c, 0x20, 0x50, 0x32, 0x37, 0x32, 0x33, 0x44, 0x0a, 0x20, 0x00, 0x00, ↵ ↵0x00, 0xfd,	
/* 6fH: */ 0x00, 0x31, 0x4b, 0x1d, 0x71, 0x1c, 0x00, 0x0a, 0x20, 0x20, 0x20, 0x20, 0x20, 0x20, ↵ ↵0x01, 0xe9,	
/* 7fH: */ 0x02, 0x03, 0x18, 0xb1, 0x4d, 0x01, 0x02, 0x03, 0x07, 0x11, 0x12, 0x16, 0x13, 0x04, ↵ ↵0x14, 0x1f,	
/* 8fH: */ 0x05, 0x10, 0x65, 0x03, 0x0c, 0x00, 0x10, 0x00, 0x02, 0x3a, 0x80, 0x18, 0x71, 0x38, ↵ ↵0x2d, 0x40,	
/* 9fH: */ 0x58, 0x2c, 0x45, 0x00, 0x55, 0x50, 0x21, 0x00, 0x00, 0x1e, 0x01, 0x1d, 0x80, 0x18, ↵ ↵0x71, 0x1c,	
/* afH: */ 0x16, 0x20, 0x58, 0x2c, 0x25, 0x00, 0x55, 0x50, 0x21, 0x00, 0x00, 0x9e, 0x01, 0x1d, ↵ ↵0x00, 0x72,	
/* bfH: */ 0x51, 0xd0, 0x1e, 0x20, 0x6e, 0x28, 0x55, 0x00, 0x55, 0x50, 0x21, 0x00, 0x00, 0x1e, ↵ ↵0x7e, 0x39,	
/* cfH: */ 0x00, 0xa0, 0x80, 0x38, 0x1f, 0x40, 0x30, 0x20, 0x3a, 0x00, 0x55, 0x50, 0x21, 0x00, ↵ ↵0x00, 0x1a,	
/* dfH: */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, ↵ ↵0x00, 0x00,	
/* efH: */ 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, ↵ ↵0x00, 0x09,	
----- sw status -----	
cap from sink YES	raw update err cnt Reserved
cap is valid YES	parse error type Reserved
raw is valid YES	parse warn type Reserved
raw get err cnt Reserved	raw length 256
----- basic cap -----	
hdmi1.4 support YES	1st block version 1.3
hdmi2.0 support NO	manufacturer name
max tmds clock(MHz)	product code

(下页继续)

(续上页)

0	Reserved
serial number	week of manufacture
Reserved	Reserved
max disp width	maxdisp height
Reserved	Reserved
scdc support	year of manufacture
NO	Reserved
dvi dual support	cec addr is valid
Reserved	Reserved
ai support	cec addr
Reserved	Reserved
ext block cnt	speaker support
Reserved	Reserved
rgb quan selectable	ycc quan selectable
NO	NO
----- video cap -----	
native format	
2560x1440p@59 16:9(vic 108)	
color space	
RGB888 YCbCr444 YCbCr422	
deep color	
ycbcr420 deep color	
ycbcr420[also]	
ycbcr420[olny]	
colorimetry	
----- format cap -----	
2560x1440p@59 16:9(mcode 108)	
640x480p@60 4:3(mcode 1)	
720x480p@60 4:3(mcode 2)	
720x480p@60 16:9(mcode 3)	
1440x480i@60 16:9(mcode 7)	
720x576p@50 4:3(mcode 17)	
720x576p@50 16:9(mcode 18)	
1440x576i@50 16:9(mcode 22)	
1280x720p@50 16:9(mcode 19)	
1280x720p@60 16:9(mcode 4)	
1920x1080i@50 16:9(mcode 20)	
1920x1080p@50 16:9(mcode 31)	
1920x1080i@60 16:9(mcode 5)	

(下页继续)

(续上页)

1920x1080p@60 16:9(mcode 16)

----- audio cap -----

NO.0:

code typemax channel num

L-PCM0

max bit rate(KHz)bit depth

0hz

sample rate(Hz)

----- detail timing -----

[NO.]	:hact	vact	p/i	pclk	aspw	asph	hfb	hpw	hbb	vfb	vpw	vbb	img_w	img_h	ihs	ivs	idv
[NO. 0]:	2560	1440	p	241500	16	4	32	48	80	3	5	33	16	4	YES	NO	NO
[NO. 1]:	640	480	p	25175	4	3	16	96	48	10	2	33	4	3	NO	NO	NO
[NO. 2]:	720	480	p	27027	4	3	16	62	60	9	6	30	4	3	NO	NO	NO
[NO. 3]:	720	480	p	27027	16	9	16	62	60	9	6	30	16	9	NO	NO	NO
[NO. 4]:	1440	480	i	27027	16	9	38	124	114	4	3	15	16	9	NO	NO	NO
[NO. 5]:	720	576	p	27000	4	3	12	64	68	5	5	39	4	3	NO	NO	NO
[NO. 6]:	720	576	p	27000	16	9	12	64	68	5	5	39	16	9	NO	NO	NO
[NO. 7]:	1440	576	i	27000	16	9	24	126	138	2	3	19	16	9	NO	NO	NO
[NO. 8]:	1280	720	p	74250	16	9	440	40	220	5	5	20	16	9	YES	YES	NO
[NO. 9]:	1280	720	p	74250	16	9	110	40	220	5	5	20	16	9	YES	YES	NO
[NO.10]:	1920	1080	i	74250	16	9	528	44	148	2	5	15	16	9	YES	YES	NO
[NO.11]:	1920	1080	p	148500	16	9	528	44	148	4	5	36	16	9	YES	YES	NO
[NO.12]:	1920	1080	i	74250	16	9	88	44	148	2	5	15	16	9	YES	YES	NO
[NO.13]:	1920	1080	p	148500	16	9	88	44	148	4	5	36	16	9	YES	YES	NO

【调试信息分析】

记录当前显示设备的 EDID。

【参数说明】

parameter	Description
edid raw data	EDID 原始数据，长度 256Bytes
cap from sink	能力集是否来自显示设备 (SINK)。取值：{YES, NO}
raw update err cnt	原始数据更新错误次数。
cap is valid	能力集是否有效。取值：{YES, NO}
parse error type	EDID 解析错误标志。取值：0：无错误；1：校验错误；2：数据头错误；3：基础数据块为非 1.3 版本；4：扩展数据块 TAG 值未知；5：CEA 值无效；6 及以上，保留值。
raw is valid	原始数据是否有效。取值：{YES, NO}

下页继续

表 6.2 – 续上页

parse warn type	EDID 解析警告标志。取值: {0x00000000~0xffffffff} 取值中每 bit 表示一类警告, 见下 Bit0: 无警告; Bit1: VENDOR 块无效; Bit2: DTD 块数量溢出; Bit3: DTD 块非法; Bit4: 无扩展块; Bit5: 扩展块数量大于 4; Bit6: AUDIO 块数量溢出; Bit7: VIC 个数溢出; Bit8: VIC 非法; Bit9: VSDB 非法; Bit10: HFVSDB 非法; Bit11: SPEAKER DB 非法; Bit12: YCBCR420 VIC 数量溢出; Bit13: 数据块长度非法。Bit14~Bit31: 保留。
raw get err cnt	原始数据获取失败次数。
raw length	原始数据长度。
HDMI1.4 support	Sink 端是否支持 HDMI1.4。取值: {YES, NO}
1st block version	EDID 第一块版本号。
HDMI 2.0 support	是否支持 HDMI2.0 取值: {YES, NO}
manufacturer name	Sink 端厂商名。
max tmds clock(MHz)	最大 TMDS 时钟。
product code	Sink 端产品号。
serial number	Sink 产品序列号。
week of manufacture	Sink 端产品制造周。
max disp width	Sink 最大显示的水平宽度 (单位: 厘米)。
max disp height	Sink 最大显示的垂直高度 (单位: 厘米)。
SCDC support	Sink 端是否支持 SCDC。取值: {YES, NO}
year of manufacture	Sink 端产品制造年份。
dvi dual support	Sink 端是否支持 DVI Dual。
cec addr is valid	Sink 端 CEC 地址是否有效。取值: {YES, NO}
AI support	Sink 端是否支持 AI。取值: {YES, NO}
cec addr	CEC 物理地址。
ext block cnt	EDID 扩展块数量。
speaker support	Sink 端支持的扬声器。取值: { FL_FR, LFE, FC, RL_RR, RC, FLC_FRC, RLC_RRC, FLW_FRW, FLH_FRH, TC, FCH, UNKNOWN}
rgb quan selectable	Sink 端是否支持 RGB 量化范围可选。取值: {YES, NO}
ycc quan selectable	Sink 端是否支持 YCbCr 量化范围可选。取值: {YES, NO}
native format	显示设备的最佳制式/VIC 码。
color space	显示设备支持的颜色空间。取值: {RGB444,YCbCr444,YCbCr422,YCbCr420}
deep color	支持的深色位数。取值: {RGB_30Bit, RGB_36Bit, RGB_48Bit, YCbCr444_SameRGB}

下页继续

表 6.2 – 续上页

ycbcr420 deep color	支持 YCbCr420 的深色位数取值: { 30Bit, 36Bit, 48Bit}。
ycbcr420[also]	除 RGB、YCbCr444、YCbCr422 外, 同时支持 YCbCr420 的 VIC 码。
ycbcr420[only]	只支持 YCbCr420 的 VIC 码。
colorimetry	Colorimetry 色域。取值: { xvYCC601, xvYCC709, sYCC601, AdobeYCC601, AdobeRGB, BT2020cYCC, BT2020YCC, BT2020RGB }
NO.x:	编号 x
code type	编码类型。取值: {STREAM, L-PCM, AC3, MPEG1, MP3, MPEG2, AAC_LC, DTS, ATRAC, OneBitAudio, EAC3, DTS-HD, MAT, DST, WMA_PRO, Reserved, UNKNOWN}
max channel num	最大通道数。
max bit rate(KHz)	最大 bit 速率。
bit depth	支持的 Bit 深度。取值: {0, 8, 16, 18, 20, 24, 32, N/A}
sample rate(Hz)	采样速率。取值: {0, 8000, 11025, 12000, 16000, 22050, 24000, 32000, 44100, 48000, 88200, 96000, 176400, 192000, 768000}
NO.	编号
hact	水平有效像素
vact	垂直有效像素
P/I	逐/隔行
pclk	像素时钟
aspw	宽高比-宽
asph	宽高比-高
hfb	水平前消隐
hpw	水平脉冲宽度
hbb	水平后消隐
vfb	垂直前消隐
vpw	垂直脉冲宽度
vbb	垂直后消隐
img_w	图像宽
img_h	图像高
ihs	水平同步脉冲是否翻转
ivs	垂直同步脉冲是否翻转
idv	有效信号是否翻转

7 Debug 调试

当出现问题时，可通过 Debug 调试辅助完成一些实验，帮助快速定位问题原因

须知：Debug 命令属于高级调试接口，无特殊原因请不要使用。正常情况下请勿使用，否则可能导致显示异常或无显示。

【HDMI Debug 调试命令】

· help(h)

命令：echo help > /proc/soph/hdmi

功能：查看当前支持的 debug 命令、命令编号以及简化命令。

参数说明：无

说明：每条命令有对应的简化命令以及编号，使用时可选择使用命令全称、简化命令以及命令对应的编号。

· event (e)

命令：echo event argv1 > /proc/soph/hdmi

功能：软件模拟产生事件。

参数说明如表 6-1：

表 7.1: event 命令参数说明

argv1	说明
16	HPD，产生 HotPlug 事件
17	HOTUNPLUG，产生 UN-HotPlug 事件
18	EDID_FAIL，产生 EDID 失败事件
21	RSEN_CONNECT，产生 connect 事件
22	RSEN_DISCONNECT，产生 disconnect 事件
25	SCRAMBLE_FAIL，产生加扰失败注销事件
26	SCRAMBLE_SUCCESS，产生加扰成功注销事件
27	ZERO_DRM_TIMEOUT，产生全 0 信息帧发送超时事件
28	HDR_SWITCH_TIMEOUT，产生 HDR 模式切换事件

· edid (ed)

命令：echo edid argv1 argv2 > /proc/soph/hdmi

功能：EDID 调试。

参数说明如表 6-2 所示：

表 7.2: edid 命令参数说明

argv1	argv2	说明
0 (s)	任意值	从 sink 端读取 EDID
1 (f)	EDID 文件路径	EDID 原始数据文件路径, 软件加载 EDID 数据

- avmute (a)

命令: echo avmute argv1 > /proc/soph/hdmi

功能: AVMUTE 开关。

参数说明: aargv1 取值 [0, 1], 0 表示发 clear AVMUTE 包, 1 表示发 AVMUTE 包。

- cbar (c)

命令: echo cbar argv1 > /proc/soph/hdmi

功能: colorbar 开关。

参数说明: argv1 取值 [0, 1], 0 表示关闭 colorbar, 1 表示打开 colorbar。

- scdc (sc)

命令: eecho scdc argv1 argv2 argv3 > /proc/soph/hdmi

功能: SCDC 调试。

参数说明如表 6-3 所示。

表 7.3: scdc 命令参数说明

argv1	argv2	argv3	说明
0	x	x	关闭 SCDC
1	x	x	打开 SCDC
2	x	x	获取 SCDC 状态

- ddc (dd)

命令: echo ddc argv1 > /proc/soph/hdmi

功能: 设置 DDC 速率。

参数说明: argv1 取值 [1, 300], 表示 DDC 速率为 1-300KHz。

注意事项: 此速率并非精确值, 实际速率可能与此值有偏差。

- hdmi_mode (m)

命令: echo hdmi_mode argv1 > /proc/soph/hdmi

功能: 设置 HDMI 输出模式。

参数说明: argv1 取值 [0, 2], 0 表示 DVI 模式, 1 表示 HDMI1.4 模式, 2 表示 HDMI2.0 模式。

- aut (au)

命令: echo aut argv1 > /proc/soph/hdmi

功能: 认证模式开关。

参数说明: argv1 取值 [0, 1], 0 表示关闭认证模式, 1 表示打开认证模式。

- cmd

命令: echo cmd argv1 argv2(optional) > /proc/soph/hdmi

功能：其他调试命令。
参数说明如表 6-5:

表 7.4: cmd 命令参数说明

argv1	argv2	说明
0	x	关闭 HDMI 输出
1	x	开启 HDMI 输出
2	x	发送黑帧
3	x	关闭黑帧
4	>= 0	0: 选择模拟指标参数 0 1: 选择模拟指标参数 1