

HMS

调试指南

文档版本 04

发布日期 2015-12-11

版权所有 © 深圳市海思半导体有限公司2016。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总部办公楼 邮编: 518129

网址: http://www.hisilicon.com/cn/

客户服务电话: 4008302118

客户服务邮箱: support@hisilicon.com

前言

概述

本文档主要介绍高清SDK的proc调试信息。

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
Hi3798C	V1XX
Hi3798M	V1XX
Hi3796C	V1XX
Hi3796M	V1XX
Hi3798C	V2XX

读者对象

本文档(本指南)主要适用于以下工程师:

- 技术支持工程师
- 软件开发工程师

作者信息

章节号	章节名称	作者信息
1	概述	H00183514
2.1	SYS	H00183514

章节号	章节名称	作者信息
2.2	DEMUX	L 64565
2.3	AVPLAY	H00183514
2.4	VDEC	L00225186
2.5	SYNC	H00183514
2.6	WINDOW	Z56248
2.7	DISPLAY	Z56248
2.8	ADEC	D00241353
2.9	AO	W00196555/Z00186919
2.10	HiGo	Z00141204
2.11	HiFB	Z00141204
2.12	TDE	Z00141204
2.13	HDMI	Y00229039
2.14	VOIP	W00196555/Z00186919
2.15	TUNER	W00136565
2.16	IR	Z00149549
2.17	I2C	100211254
2.18	SCI	100211254
2.19	LOG	L64565
2.20	Interrupts	W58735
2.21	Media-mem	L64565/ L00227819
2.22	STAT	L64565
2.23	开机画面及瞬播	100211254
2.24	AENC	D00241353
2.25	PVR	Z00111416
2.26	VI	L00214567
2.27	VP	L00214567
2.28	PDM	100211254
2.29	VENC	L00228308
2.30	AI	W00196555/Z00186919
2.31	VPSS	F00228503/G45208

章节号	章节名称	作者信息
2.32	KEYLED	Z00149549
2.33	PMOC	Z00149549
2.34	Hipalyer	L00194915
3.1	DEMUX问题定位方法	L00188263
3.2	VDEC问题定位方法	L00225186
3.3	显示通道问题定位方法	Z56248
3.4	HDM问题定位方法	Y00229039
3.5	视频显示问题定位方法	G00246722
3.6	音频问题定位方法	W00196555/Z00186919
3.7	同步问题的定位方法	H00183514
3.8	信道前端问题定位方法	W00136565
3.9	开机画面和瞬播问题定位方 法	L00211254
3.10	VP问题定位方法	L00214567
3.11	智能卡问题定位方法	L00211254
3.12	KEYLED问题定位方法	Z00149549
3.13	LOADER问题定位方法	L64565
3.14	PMOC调试定位方法	Z00149549
3.15	gprof使用方法	L00227819

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
2014-04-30	00B01	第一次临时版本发布。
2014-10-14	00B02	更新2.10、2.11、2.12、2.21章节。
2014-10-30	01	新增支持Hi3796MV100芯片。
2015-04-30	02	新增支持Hi3798CV200的芯片。

修订日期	版本	修订说明
2015-10-22	03	修改2.33章节。
2015-12-11	04	修改VDEC、ADEC、AO、HDMI、Media_mem、PMOC、Hiplayer模块的调试信息。

目录

前言	ii
1 概述	1
2 Proc 信息	2
2.1 SYS	2
2.2 DEMUX	3
2.2.1 demux_main.	
2.2.2 demux_port	4
2.2.3 demux_chan.	8
2.2.4 demux_chanbuf	8
2.2.5 demux_filter	9
2.2.6 demux_key	10
2.2.7 demux_pcr.	11
2.2.8 demux_rec.	11
2.2.9 demux_rec_index	12
2.3 AVPLAY	13
2.4 VDEC	16
2.5 SYNC	21
2.6 WINDOW	26
2.7 DISPLAY	31
2.8 ADEC	36
2.9 AO	39
2.10 HiGo	43
2.11 HiFB	45
2.12 TDE	46
2.13 HDMI	49
2.14 TUNER	69
2.14.1 Tuner	69
2.14.2 Tuner_Reg	70
2.15 IR	70
2.16 I2C	72
2.17 SCI	74
2.18 LOG	77

2.19 Interrupts.	80
2.20 Media-mem.	81
2.21 SATA	83
2.22 开机画面及瞬播	85
2.23 AENC	86
2.24 PVR	88
2.24.1 pvr_play	88
2.24.2 pvr_rec	93
2.25 VI	94
2.26 VP	97
2.27 PDM	99
2.28 VENC	101
2.29 AI	105
2.30 VPSS	106
2.30.1 实例信息	106
2.30.2 PORT 信息	108
2.31 KEYLED	110
2.32 PMOC	112
2.33 HiPlayer	114
2.33.1 查看 HiPlayer proc 功能命令	114
2.33.2 查看文件信息命令	115
2.33.3 查看播放器控制信息	117
2.33.4 查看解析器信息	121
2.33.5 查看 SDK 适配信息	123
2.33.6 Hiplayer 静态 proc 命令	125
2.33.7 查看切台性能统计信息	126
2.33.8 查看 SEEK 性能统计信息	129
3 常见问题定位方法	134
3.1 DEMUX 问题定位方法	134
3.1.1 调试手段介绍	134
3.1.2 收不到数据相关问题定位	136
3.1.3 播放视频马赛克或者声音卡顿问题定位(包括加扰流)	138
3.1.4 RAM 端口相关问题定位	139
3.2 VDEC 问题定位方法	
3.2.1 VDEC 模块介绍	139
3.2.2 用 echo 动态调整 vfmw 的行为	141
3.2.3 播放停滞定位	
3.3 显示通道问题定位方法	
3.3.1 显示通道 PROC 信息分析	
3.3.2 常见显示问题	
3.4 HDMI 问题定位方法	
3.4.1 proc 信息分析	

3.4.2 常见问题分析和定位	152
3.5 视频显示问题定位方法	153
3.5.1 常见视频显示问题	153
3.6 音频问题定位方法	154
3.6.1 通过分析 PROC 信息定位音频问题	154
3.6.2 分析音频无输出问题	156
3.6.3 音频透传格式相关设置	156
3.6.4 音频输出不正常	157
3.7 同步问题的定位方法	157
3.7.1 关闭同步	157
3.7.2 查看同步的 proc 信息	158
3.7.3 使用 msp_debug 工具定位同步问题	160
3.7.4 查看可能导致不同步的原因	161
3.8 信道前端问题定位方法	163
3.8.1 调试手段	163
3.8.2 不能锁频的问题定位方法	164
3.9 开机画面和瞬播问题定位方法	166
3.9.1 查看参数设置是否正确	166
3.9.2 查看开机画面及瞬播 Log	166
3.9.3 如何加快瞬播启动速度	166
3.10 VP 问题定位方法	167
3.10.1 本地视频无法显示的问题	167
3.10.2 远端视频无法显示的问题	167
3.10.3 画面卡顿、延时问题	167
3.11 智能卡问题定位方法	167
3.11.1 插卡没任何反应如何定位?	168
3.11.2 卡复位失败收不到 ATR 如何定位?	168
3.11.3 能收到 ATR, 发送完命令之后,接收数据没响应,提示接收超时如何定位?	168
3.11.4 如何增加冷复位后等待接收 ATR 的时间	169
3.12 KEYLED 问题定位方法	170
3.12.1 KEYLED 面板没显示如何定位?	170
3.12.2 CT1642 如何支持系统启动显示?	170
3.12.3 前面板 GPIO 使用修改说明	171
3.13 LOADER 问题定位方法	171
3.13.1 BootLoader 输出调试日志的方法	172
3.13.2 AppLoader 输出调试日志的方法	172
3.13.3 烧写 1080 制式的开机 logo 后,检测升级不成功问题	172
3.13.4 BOOT 对 kernel 校验失败问题	172
3.14 PMOC 调试定位方法	172
3.14.1 串口打印	172
3.14.2 LED 数码管显示	
3.14.3 将 Debug 信息保存到内存中	

HMS

调试指南	目录
3 14 4 待机时内核相关驱动调试方法	174

插图目录

图 2-1 HDMI 正常输出时的打印信息一	50
图 2-2 HDMI 正常输出时的打印信息二	51
图 2-3 HDMI 正常输出时的打印信息三	52
图 2-4 HDMI 正常输出时的打印信息四	53
图 2-5 HiDebugger 界面	80
图 2-6 media-mem 调试信息	82
图 3-1 音频数据流	155
图 3-2 同步属性设置	158
图 3-3 预同步状态及 PCR 状态	159
图 3-4 视频同步调整状态	159
图 3-5 HiDebugger 配置界面	160
图 3-6 执行 cat /proc/msp/win0100 后的打印信息	
图 3-7 执行 cat /proc/msp/ sound0 后的打印信息	163

】 概述

调试信息采用了Linux下的文件系统,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,也可以通过proc系统控制驱动实现基于调试目的驱动设置。

【文件目录】

除了linux系统自带的proc文件系统外,海思SDK提供的proc信息主要位于/proc/msp目录下。

【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/msp/vdec00。
- 如需查看帮助信息,请向模块输入help命令,例如echo help > /proc/msp/avplay00。
- 不同版本中,或者同一版本在不同的运行状态下,这些调试信息可能稍有不同,现以HiSTBLinuxV100R003版本为例。



1 注意

下文的参数在描述时有以下情况需要注意:

- 属于"是否"、"开关"或"使能"类型的参数,涉及到未列出取值和含义的关系时,默认取值为1表示肯定,为0表示否定。
- 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。

2 Proc 信息

2.1 SYS

【调试信息】

cat /proc/msp/sys

SDK VERSION: [HiSTBLinuxV100R003] Build Time: [Jun5 2014, 12:00:40]

CHIP VERSION: HI3798C(0x21) v200

DOLBY: YES DTS: YES ADVCA: NO

ROVI (Macrovision): NO

【调试信息分析】

记录当前系统的基本信息。

【参数说明】

参数	描述	
SDK_VERSION	SDK的版本号和编译时间。	
CHIP_VERSION	芯片版本号。	
DOLBY	是否支持杜比,YES表示支持,NO表示不支持。	
DTS	是否支持DTS,YES表示支持,NO表示不支持。	
ROVI	是否支持Macrovision, YES表示支持, NO表示不支持。	

更换了ko文件后,查看SDK的版本号和编译时间可以确认是否使用了正确的ko文件,避免使用了错误的ko文件,这一点在版本升级的时候尤其重要。

芯片版本号是指当前正在运行程序的芯片版本号,如果芯片表面被散热片遮挡导致无法查看其标识,可以用查看此处获取芯片版本号。

【提示】

对于SDK发布的库、ko文件,在Linux下,可以采用以下方法查看SDK版本号,命令格式如下:

strings 库文件或者ko文件 | grep SDK_VERSION

执行此命令后,会打印出库文件或者ko文件对应的SDK版本号,举例如下:

\$ strings ./pub/lib/share/libhi_common.so | grep SDK_VERSION
SDK_VERSION:[HiSTBLinuxV100R003C00SPC010] Build Time:[Jun 5 2014,
10:31:53]
\$ strings ./pub /kmod/hi common.ko | grep SDK VERSION
SDK_VERSION:[HiSTBLinuxV100R003C00SPC010] Build Time:[Jun 5 2014,
10:31:12]

2.2 DEMUX

2.2.1 demux_main

【调试信息】

cat /proc/msp/demux_main

DmxId PortId

- 0 --
- 1 ---
- 2 ---
- 3 ---
- 4 --
- 5 ---
- 6 ---

type "echo help > /proc/msp/demux_main" to get help information

【调试信息分析】

记录Demux和端口的绑定关系。

【参数说明】

参数	描述
DmxId	Demux号。
PortId	端口号,IF端口从0开始编号,TSI端口从32开始编号,RAM端口从128开始编号。表示Demux绑定的端口号,""表示Demux没有绑定端口,此时Demux收不到任何数据。

【调试命令】

● 获取帮助:

echo help > /proc/msp/demux main

- 录制端口码流:
 - 开始录制: echo save allts start x[portid] > /proc/msp/demux_main
 - 停止录制: echo save allts stop > /proc/msp/demux_main 例如,根据以上示例proc内容,要选择录制TSI1端口的码流,则执行命令: echo save allts start 32 > /proc/msp/demux main
- 录制送入IP端口的码流
 - 开始录制: echo save ipts start x[ram portid]>/proc/msp/demux main
 - 停止录制: echo save ipts stop > /proc/msp/demux_main 例如,根据以上示例proc内容,要选择录制RAM0端口的码流,则执行命令: echo save allts start 128 > /proc/msp/demux_main
- 录制demux 输出的TS流
 - 开始录制: echo save dmxts start x[dmxid] > /proc/msp/demux_main
 - 停止录制: echo save dmxts stop > /proc/msp/demux main
- 录制demux 输出的ES流
 - 开始录制: echo save es start > /proc/msp/demux_main
 - 停止录制: echo save es stop > /proc/msp/demux_main

2.2.2 demux_port

【调试信息】

```
Type
PARALLEL_NOSYNC_188
                                 ClkReverse
                                                            PARALLEL NOSYNC 188
                                                  D7
D7
                                                            PARALLEL NOSYNC 188
0x0
            0x0
                                                  D7
D7
                                                            PARALLEL NOSYNO
0x0
            0x0
                                                            PARALLEL NOSYN
                                   ClkMode
                                                 VldMode
                                                                                                     ClkDiv
                                   NORMAL
                                                                                              150M
             none
```

【调试信息分析】

记录Demux端口的信息。

IF Port是来自Tuner的信号输入端口,使用了芯片内置QAM的端口,编号(ID)从0开始,一般不多于1个。

TSI Port 也是来自Tuner的信号输入端口,与IF Port的区别是,TSI Port 使用外置QAM。

RAM Port是DDR端口。

TSO Port是TS输出端口。

参数	描述
Id	端口号,TUNER端口从0开始编号, RAM端口从128开始编号。
AllTsCnt	总的TS包计数,32位。
ErrTsCnt	错误的TS包计数,16位。
Lock/lost	同步锁定门限和同步丢失门限。
ClkReverse	TS输入时钟是否反相(只针对tuner port 有效)。 0: 不反相
	1: 表示反相
BitSel	端口线序选择(只针对tuner port有 效)。
	D7: 若端口类型为并行,表示cdata[7]为 最高位;
	若端口类型为串行,表示cdata[7]为数据 线;
	D0: 若端口类型为并行,表示cdata[0]为 最高位;
	若端口类型为串行,表示cdata[0]为数据 线;

参数	描述
Туре	端口类型。
	PARALLEL_BURST: 并行BURST模式;
	PARALLEL_VALID: 并行VALID模式;
	PARALLEL_NOSYNC_188: 自同步188 模式;
	PARALLEL_NOSYNC_204: 自同步204 模式;
	PARALLEL_NOSYNC_188_204: 188/204自动识别模式;
	AUTO: 自同步;
	USER_DEFINED: 用户自定义模式;
	SERIAL: 串行模式;
	SER_SERIAL2BIT_NOSYNC: 2 bit串行 模式;
	SER_NOSYNC: 串行自同步模式;
	SER_2BIT_NOSYNC: 2 bit串行自同步模式;
	ERR: 设置错误。
Enable	TSO端口是否使能。
	0: 没有使能;
	1: 使能。
TSPortID	TSO 端口码流源来自哪一个TS Port。对应IF/TSI/RAM 端口ID。
ClkMode	TSO 端口时钟模式。
	NORMAL: 均匀时钟;
	JITTER: 抖动时钟。
VldMode	TSO端口Valid信号工作模式。
	0: Valid信号只在输出数据的时候为高
	(对应
	HI_UNF_DMX_TSO_VALID_ACTIVE_O UTPUT);
	1: Valid信号始终为高,
	(对应
	HI_UNF_DMX_TSO_VALID_ACTIVE_H IGH)

参数	描述
Sync	TSO端口输出Sync信号持续时间。
	Bit: 只在输出同步字节的第一个Bit的时候给出Sync信号;
	Byte: 在输出整个同步字节的时候给出 Sync信号。
Serial	TSO 端口是否为串行输出。
	0: 并行。
	1: 串行;
LSB	TSO端口大小端模式。
	0: 首先输出最高位(大端)
	1: 首先输出最低位(小端)。
Clk	TSO 模块时钟频率。
ClkDiv	TSO端口分频因子。
	TSO端口输出频率 = TSO模块时钟频率 (Clk)/ TSO 端口分频因子(ClkDiv)
TsChkRange	Ts包长度检测范围(只针对ram port有效)
	none: Ram端口为AUTO类型,硬件自动 检测包长(范围: 188~255);
	x~y:包长检测范围。(当Ram端口为 USER_DEFINED类型时候,支持用户设 定包场检测范围:x~y)。
BufAddr	TS Buffer的起始地址。
BufSize	TS Buffer的大小。
BufUsed	TS Buffer已使用的大小和百分比。
Read	TS Buffer的读指针偏移位置。
Write	TS Buffer的写指针偏移位置。
Get(Try/Ok)	Try表示调用HI_UNF_DMX_GetTSBuffer 的次数;
	Ok表示调用HI_UNF_DMX_GetTSBuffer 的次数。
Put	调用HI_UNF_DMX_PutTSBufferr成功的 次数。

正常情况下AllTsCnt会逐渐增加,否则表示端口没有数据进入,请检查:

● IF/TSI端口:

- Tuner是否锁定;
- 端口类型是否设置正确。

■ RAM端口

- 端口类型是否设置正确;
- 是否有外部数据的输入,即应用是否调用HI_UNF_DMX_GetTSBuffer和 HI UNF DMX PutTSBuffer进行了TS数据的注入。

ErrTsCnt记录错误的TS包个数,如果其值不断增加,说明Demux不停地收到错误的TS包,在这期间将可能导致视频播放马赛克现象和音频卡顿现象。

2.2.3 demux chan

【调试信息】

cat /proc/msp/demux_chan
DmxIdChnIdPIDType Mod StatKeyIdAcquire(Try/Ok)Release
800x1022VIDPLY OPEN--1936/1481475
810x1023AUDPLY OPEN--593/569568

【调试信息分析】

记录Demux通道的信息。

【参数说明】

参数	描述
DmxId	Demux号,表示通道所属的Demux。
ChnId	通道号。
PID	通道的PID。
Туре	通道类型: SEC、PES、AUD、VID、ECM、PST。 在创建通道时指定通道的类型。
Mod	通道的输出模式: PLY、REC、P&R。 在创建通道时指定通道的输出模式。
Stat	通道状态: OPEN、CLOSE。
KeyId	通道绑定的密钥区号,""表示没有绑定密钥区。
Acquire(T ry/Ok)	Try表示尝试获取通道Buffer的次数; Ok表示成功获取通道Buffer的次数; 通道的输出模式为REC时,Try和Ok始终为0。
Release	调用HI_UNF_DMX_ReleaseBuf成功的次数。

2.2.4 demux_chanbuf

【调试信息】

cat /proc/msp/demux chanbuf

DmxIdChnIdFqIdOqIdSizeBlkCntBlkSizeReadWriteUsedOverflow
801016383K10081664447547399%0
8121255K204128016015899%0

【调试信息分析】

记录Demux通道缓存的信息。

【参数说明】

参数	描述
DmxId	Demux号,表示通道所属的Demux。
ChnId	通道号。
FqId	通道绑定的FQ号。
OqId	通道绑定的OQ号。
Size	通道缓存的大小,Demux将通道缓存分成若干个块使用。
BlkCnt	通道缓存块的个数。
BlkSize	通道缓存块的大小。
Read	通道缓存块的读指针,数据被取走时Read的值会不断变化。
Write	通道缓存块的写指针,有数据进入通道时Write的值会不断变化。
Used	通道缓存已使用的百分比。
Overflow	通道缓存溢出的次数,溢出时会丢失数据。

2.2.5 demux_filter

【调试信息】

【调试信息分析】

记录Demux过滤器的信息。

参数	描述
DmxId	Demux号,表示通道所属的Demux。

参数	描述
ChnId	过滤器绑定的通道号。""表示没有绑定通道。
FltId	过滤器号。
Depth	过滤器的深度。
Param	Match表示过滤器的匹配字节,十六进制显示;
	Mask表示过滤器的屏蔽字节,十六进制 显示;
	Negate表示过滤器的取反字节,十六进制 显示。

2.2.6 demux_key

【调试信息】

【调试信息分析】

记录Demux密钥区的信息。

参数	描述
DmxId	Demux号,表示通道所属的Demux。
KeyId	密钥区号。
ChanCnt	密钥区绑定的通道个数,0表示没有绑定 通道。
AttachCnt	密钥key绑定次数统计。
DetachCnt	密钥key去绑定次数统计。
SetEvenKeyCnt	设置偶密钥次数统计。
SetOddKeyCnt	设置奇密钥次数统计。

参数	描述
EvenKey	偶密钥,十六进制显示,第一个值对应 CW1~CW4,第二个值对应CW5~ CW8,第三个值对应CW9~CW12,第四 个值对应CW13~CW16。
OddKey	奇密钥,十六进制显示,第一个值对应 CW1~CW4,第二个值对应CW5~ CW8,第三个值对应CW9~CW12,第四 个值对应CW13~CW16。

2.2.7 demux_pcr

【调试信息】

```
cat /proc/msp/demux pcr
DmxId PcrId PID CurrPcr CurrScr
0 0 0x1fff 0xffffffff 0xffffffff
```

【调试信息分析】

记录Demux PCR通道的信息。

【参数说明】

参数	描述
DmxId	Demux号,表示PCR通道所属的Demux。
PcrId	PCR通道号。
PID	PCR通道的PID。
CurrPcr	当前PCR值,单位:毫秒。
CurrScr	当前本地时钟的值,单位:毫秒。

2.2.8 demux_rec

【调试信息】

【调试信息分析】

记录Demux录制的信息。

参数	描述
DmxId	Demux号。
RecId	录制实例号。
FqId	录制绑定的FQ号。
OqId	录制绑定的OQ号。
Туре	录制类型,pid表示录制选定的PID,all 表示录制所有的PID。
Descramed	是否录制解扰后的数据,0表示录制解扰 前的数据;1表示录制解扰后数据。
Status	录制状态,start表示正在录制,stop表示 录制停止。
Size	录制缓存的大小,Demux将录制缓存分 成若干个块使用。。
BlkCnt	录制缓存块的个数。
BlkSize	录制缓存块的大小。
Read	录制缓存块的读指针。
Write	录制缓存块的写指针。
Overflow	录制缓存溢出的次数。

2.2.9 demux_rec_index

【调试信息】

cat /proc/msp/demux_rec_index
DmxId RecId FqId OqId ScdId Type Pid Size BlkCnt BlkSize Read Write
Overflow
 1 0 5 4 0 video 0x200 56K 64 896
18 18 0

【调试信息分析】

记录Demux录制索引的信息。

参数	描述
DmxId	Demux号。
RecId	录制实例号。
FqId	录制索引绑定的FQ号。

参数	描述
OqId	录制索引绑定的OQ号。
ScdId	录制索引绑定的SCD号。
Туре	索引类型,audio表示音频索引,video表示视频索引,none表示无索引。
Pid	索引信息的PID。
Size	索引缓存的大小,Demux将索引缓存分成若干个块使用。
BlkCnt	索引缓存块的个数。
BlkSize	索引缓存块的大小。
Read	索引缓存块的读指针。
Write	索引缓存块的写指针。
Overflow	索引缓存溢出的次数。

2.3 AVPLAY

【调试信息】

#cat /proc/msp/avplay00

	-Hisilicon AVE	PLAYO Out Info	
	:TS		:0
4.1		OverflowProc	
		ThreadID	
ThreadScheTimeOutCnt	:0	ThreadExeTimeOutCnt	:0
		CHANNEL	
Vid Enable	:TRUE	Vdec Type	:MPEG2
		Vdec Mode	:NORMAL
		FrcEnable	:TRUE
		FrcOutRate	:50.0
TplaySpeed	:1.0	LowDelayEnable	:FALSE
Vdec ID	:vdec00	-	
FrameChanID	:port0000->win0100(master)		
FrameChanID	:port0001->win0000(slave00)		
AcquireFrame(Try/OK)			
SendFrame (Try/OK)			
SendFrame (Try/OK)			
	AUD C	CHANNEL	
Aud Enable	:TRUE	Adec Type	:mp3
AudOverflowNum	:0	AdecDelayMs	:0
DmxAudChnNum	:1		
DmxAudPid	:0x28a		
Adec ID	:adec00		
Track ID	:track00(mast	er)	
AcquireStream(Try/OK)	:1864/279		
SendStream(Try/OK)	:279/279		
AcquireFrame(Try/OK)	:1864/426		

SendFrame(Try/OK) :426/426(master)

【调试信息分析】

记录某个AVPLAY的状态,高清芯片支持16个播放器,最大支持16路节目同时播放(受限于芯片的解码能力),每一个播放器都会在/proc/msp生成一个节点,调试的时候需要注意选择正确的avplay。

参数	描述
Stream Type	输入流类型,高清芯片支持TS、ES两种 输入方式。
DmxId	Demux ID
CurStatus	avplay播放器的状态,STOP、PLAY、 TPLAY、PAUSE、EOS等状态。
OverflowProc	缓存溢出处理类型。 RESET:溢出时,复位; DISCARD:溢出时,丢弃。
Sync ID	播放器对应的SYNC实例。
Thread ID	AVPLAY数据处理线程ID。
ThreadScheTimeOutCnt	AVPLAY数据处理线程调度超时计数,超时门限缺省为30ms。
ThreadExeTimeOutCnt	AVPLAY数据处理线程执行超时计数,超时门限缺省为30ms。
Vid Enable	视频解码器工作状态。 HI_TRUE: 运行; HI_FALSE: 停止。
Vdec Type	视频解码类型: MPEG2、MPEG4、 AVS、H263、H264、VC1、UNKNOW。
VidOverflowNum	视频ES Buffer上溢次数。
Vdec Mode	视频解码模式。 NORMAL:解所有帧; IP: 只解I帧和p帧; I: 只解I帧。
VidPid	视频PID。
FrcEnable	帧率转换是否使能。
FrcInRate	帧率转换输入帧率。
FrcOutRate	帧率转换输出帧率。
TplaySpeed	Tplay播放速度。

参数	描述
LowDelayEnable	低延时是否使能。
Vdec ID	播放器对应的Vdec实例ID。
FrameChanID	播放器对应的视频帧通道ID,如 port0000->win0100(master),表示VPSS PORT0000的视频帧经AVPLAY送往 WIN0100,master/slave/virtual分别主通 道/从通道/虚拟通道。
AcquireFrame(Try/OK)	AVPLAY尝试从VDEC获取视频帧次数和成功次数。
SendFrame(Try/OK)	AVPLAY尝试送视频帧给VO的次数和成功次数,master/slave/virtual分别主通道/从通道/虚拟通道。
Aud Enable	音频解码器工作状态。 HI_TRUE:运行; HI_FALSE:停止。
Adec Type	音频解码器类型。
AudOverflowNum	音频ES Buffer上溢次数。
AdecDelayMs	ADEC Buffer数据量(ms)。
DmxAudChnNum	音频Demux通道数。
AudPid	音频PID。
Track ID	播放器对应的音频Track ID,master/slave/virtual分别主通道/从通道/虚拟通道。
AcquireStream(Try/OK)	AVPLAY尝试从DEMUX获取ES buffer的次数和成功次数。
SendStream(Try/OK)	AVPLAY释放ES buffer给DEMUX的次数和成功次数。
AcquireFrame(Try/OK)	AVPLAY尝试从ADEC获取解码后音频帧 送AO播放的次数和成功次数。
SendFrame(Try/OK)	AVPLAY送音频帧给AO的次数和成功次数。

【调试命令】

- 获取帮助 echo help > /proc/msp/avplayxx
- 打开/关闭帧率转换

echo FrcEnable=true|false > /proc/msp/avplayxx

如果音视频不能播放,可以从下面几个方面定位:

- 按照前面描述的proc调试说明,保证tuner锁定、该avplay对应的Demux有数据(根据DmxId找到该demux对应的TS 端口,然后查看demux port信息);
- AVPLAY使用的音视频通道是否有数据进入(demux_chan调试信息中类型为音视频通道, Pid和Avplay调试信息中Pid一致的通道是该AVPLAY使用的通道);
- 音视频解码器是否使能(查看Vid Enable与Aud Enable值);
- 工作状态是否是Play状态(查看CurrStatus);
- 音视频PID是否设置正确(查看VidPid与AudPid);
- 音视频的解码类型是否正确(查看Vdec Type与Aud Type);
- 如果上面都没有异常,进一步查看该AVPLAY使用的音视频解码器的调试信息。

如果音视频同步不正常,可以通过hSync句柄,查看同步相关信息。

2.4 VDEC

【调试信息】

```
# cat /proc/msp/vdec00
           Work State: RUN
VpssID: vpss00
VfmwID: vfmw00
Codec ID: MPEG2(0x0)
Mode: NORMAL
Priority: 3
ErrCover: 100
OrderOutput: 0
CtrlOption: 0x0
Capbility: NORMAL/FULLHD/H264
         ----- Information-----
Source: User0
StreamSize(Total/Current): 0x4f2d3dc/0x4fc000
BitRate(bps): 14145256
StreamBuffer(Total/Used/Persent): 0x500000/0x4fec24/99%
   -----Picture Information-----
Width*Height: 1280*720
Stride (Y/C): 0x500/0x500
FrameRate(fps): Real(25.19) FrameInfo(23999)
PlayFormat: OTHER(43)
FrmPackingType: Normal
Aspect (User/Decode): 0:0/16:9
FieldMode: Frame
Type: Progressive
VideoFormat: UNKNOWN
TopFirst: 1
ErrFrame: 2
TypeNum(I/P): 0/2403
DMX/USER->VDEC
GetStreamBuffer(Try/OK): 9665/5068
PutStreamBuffer(Try/OK): 5068/5068
VDEC->VFMW
AcquireStream(Try/OK): 52793/5068
ReleaseStream (Try/OK): 4749/4749
VFMW->VPSS
```

AcquireFrame(Try/OK): 1186/1184 ReleaseFrame(Try/OK): 1183/1183

VPSS->AVPLAY

AcquireFrame(Try/OK): 1186/1184 ReleaseFrame(Try/OK): 1183/1183

cat /proc/vfmw00

----- vfmw00 info -----

VersionNum:2013091102

VfmwID:0

----- scd stream info ------

RawStream(Size/Num):5230258/320
SCDSegStream(Size/Num):1753506/65

SCDSegBuffer(Total/Use/Percent):1965056/1753506/89%

----- vdh frame info ------

VDH 0 load:15.9%

Decode (Width*Height):2560*544 Display(Width*Height):2560*544

FrameRate:24.2 fps

VDHFrameBuffer(Total/Use/Percent):18/17/94%

【调试信息分析】

记录视频解码器的工作状态,如果多个视频解码器同时工作,将列出多个视频解码器的调试信息,VDEC00,VFMW00表示第0路vdec解码。

参数	描述
Work State	工作状态: RUN、STOP。
Codec ID	解码类型: MPEG2、MPEG4、AVS、H263、H264、VC1、VP6、 VP6F、VP6A、SORENSON、RAW、JPEG、VP8、 VP9、OTHER。
Mode	解码模式。 NORMAL:解码所有帧; IP: 只解码I帧和P帧; I: 只解码I帧; DROP_INVALID_B:解码除了紧跟着I帧后面的B帧之外的所有帧。
Priority	解码优先级
ErrCover	视频解码器的输出帧错误隐藏门限(百分比)。 0:出现错误即不输出; 100:不管错误比例全部输出。 其他值表示按出现错误的百分比控制是否输出。
OrderOutput	输出顺序。 0: 按显示序输出; 1: 按解码序输出。

参数	描述
CtrlOption	解码器解码的特殊控制选项。
	其值请参考HI_UNF_VCODEC_ATTR_S中s32CtrlOptions的定义。
Capbility(MPEG/H264)	视频解码器当前配置的解码能力,分三个参数:
	解码器类型:
	NORMAL: 常规解码器;
	IFRAME: 只用于解I帧的解码器。
	支持分辨率级别:
	QCIF:解码的图像大小不超过176*144;
	CIF:解码的图像大小不超过352*288;
	D1: 解码的图像大小不超过720*576;
	720P:解码的图像大小不超过1280*720;
	FULLHD:解码的图像大小不超过1920*1080;
	UNKNOWN: 未知。
	支持协议族:
	NOT_H264: 不支持H264, 支持H264之外的(本芯片支持的)其他协议;
	H264: 支持包括H264的(本芯片支持的)所有协议。
Stream	解码器送流状态:
	Source: 有User和Demux两种,代表数据来源是用户注入还是Demux。后面跟的值代表ES buffer或Demux通道的handle;
	StreamSize(Total/Current): Total表示当前解码器的总输入码流字节数, Current表示VDEC_Frimware缓冲区码流字节数;
	BitRate(bps): 码流输入平均码率。
	StreamBuffer(Total/Used/Percent):Total代表解码码流缓冲区的大小,Used代码缓冲区已使用的大小,Percent代表已使用缓冲区的比例。

参数	描述
Picture	解码出的图像参数:
	Width*Height: 原始图象宽高;
	Stride(Y/C): Y/C分量数据的跨幅;
	FrameRate(fps): 前一数值为统计平均帧率,后一数值为流中描述的帧率或默认帧率;
	PlayFormat: 视频制式,见HI_UNF_ENC_FMT_E定义;
	Aspect: 图象宽高比,User表示用户设置的宽高比0: 0的 意思的用户没有设置,Decode表示解码信息中的宽高比;
	FieldMode: 帧或场编码模式,有 Frame/Top/Bottom/ UNKNOWN,分别表示帧模式/顶场模式/底场模式/未知;
	Type: 视频采样类型,Progressive表示逐行采样,Interlace 表示隔行采样;
	VideoFormat: 视频格式,见HI_UNF_VIDEO_FORMAT_E 定义;
	TopFirst: 顶场优先标记;
	ErrFrame: 错误的视频帧数目;
	TypeNum(P/I): 图像逐隔行信息统计计数,含义分别为: 逐行/隔行。用户一般不需要关注。
DMX/USER->VDEC	ES模式下,AVPLAY向VDEC写入数据次数。
	Get(Try/OK):尝试获取buffer次数/成功次数;
	Put(Try/OK): 尝试写入buffer次数/成功次数。
	仅在用户注入数据模式下出现。
VFMW->VPSS	记录解码完的视频帧,输出到VPSS的情况。
	AcquireFrame(Try/OK):
	VPSS获取解码完视频帧的次数/获取成功的次数;
	ReleaseFrame(Try/OK):
	VPSS释放视频帧buffer的次数/释放成功的次数;
VDEC->VFMW	AcquireStream(Try/OK):
	Firmware获取Video ES数据的次数/获取成功的次数;
	ReleaseStream(Try/OK):
	Firmware释放Video ES buffer的次数/释放成功的次数;
VPSS->AVPLAY	AcquireFrame(Try/OK):
	AVPLAY获取VPSS解码完的帧数据的次数/获取成功的次数;
	ReleaseFrame(Try/OK):
	VO释放解码帧buffer的次数/释放成功的次数。
VersionNum	表示VFMW模块的版本号。
VfmwID	当前解码的通道号。
<u> </u>	

参数	描述
RawStream(Size/Num)	解码收到的但是还没有切割的原始码流的大小和包数。
SCDSegStream	解码已经切割出来但是还没有解码的码流的大小和包数。
SCDSegBuffer(Total/Use/ Percent)	Scd模块的码流缓存的总大小,当前占用的缓存的大小和 比例。
VDH 0 load	解码VDH硬件的占用率。
Decode(Width*Height)	解码宽高。
Display(Width*Height)	显示宽高。
FrameRate	解码帧率。
VDHFrameBuffer(Total/ Use/Percent)	解码帧存的总数,当前被占用的个数和比例。

查看视频解码是否工作正常的主要看以上几个buffer倒换的是否正常,是否有数据丢失。如果获取和释放不一致,解码和送到VO的数据不一致很可能会导致图象卡顿等现象:

视频解码的简单流程是:

步骤1 VDEC从Demux或ES buffer获取ES数据送Firmware进行解码(Firmware: Stream Input(VDEC->Firmware))。

步骤2 Firmware将解码完的1D或者2D帧存放在帧buffer中。

步骤3 VDEC从帧buffer中获取(Firmware: Frame Output(Firmware->VDEC))视频帧放入其帧队列中。

步骤4 VPSS在线程中从VDEC的帧队列中取帧做解码后处理(Frame Output(VDEC->VPSS))。

步骤5 AVPLAY向VDEC获取视频帧,VDEC从VPSS获取视频帧返回给AVPLAY。

----结束

ErrCover由VDEC属性中的u32ErrCover指定。

- 0: 错误帧不输出。
- 100: 全部输出。

如果ErrFrame不断增加,将导致马赛克或者卡顿。

- ErrCover为100:将导致马赛克。
- ErrCover为0: 将导致卡顿。
- ErrFrame的增加是由于码流有错误导致的,请检查输入码流的正确性。

正常情况下,各项值的Try/OK数目应该是不断增加的,如果哪项值不变化了,说明在此环境的处理出现了问题。

2.5 SYNC

【调试信息】

```
#cat /proc/msp/sync00
------Hisilicon SYNCO Out Info-----
         Hisilicon SYNC ATTR
SyncPrint
SyncRef
                             :AUDIO
SyncStart.VidPlusTime :60
SyncStart.VidNegativeTime :-20
SyncStart.bSmoothPlay :1
SyncNovel.VidPlusTime :3000
SyncNovel.VidNegativeTime :-3000
SyncNovel.bSmoothPlay :0
VidPtsAdjust :0
AudPtsAdjust :0
PreSyncTimeoutMs :10
bQuickOutput :0
                            :1000
          __Hisilicon PCR
PcrFirst
                  :87037404
:87037417
:0
PcrLast
PcrLocalTime
PcrAdjustMode
PcrAudSyncOK :1
PcrVidSyncOK :1
         Hisilicon VID
VidFirstCome
VidFirstSysTime
VidFirstPts
                             :1
                            :2622232
                            :87017132
VidLastPts
                            :87036412
VidPreSyncTargetInit :1
VidPreSyncTargetTime :87017132
VidFirstPlay :1
VidFirstPlayTime :2622712
VidBlockFlag
                             :0
                             :0
:87036394
VidBufPercent
VidLocalTime
VidPcrDiff
                             :-1022
                             :14
VidAudDiff
VidDiscard
                             : 0
VidSyndAdjust
                             :0
VidDiscardCnt
                             :0
VidRepeatCnt
         Hisilicon AUD
AudFirstCome :1
AudFirstSysTime :2621955
AudFirstPts :-1
AudLastPts
                       :87037352
AudPreSyncTargetInit :1
```

AudFreSyncTargetTime :87016640
AudFirstPlay :1
AudFirstPlayTime :2623481
AudBlockFlag :0
AudBufPercent :0
AudLocalTime :87036380
AudPcrDiff :-1036
AudBufTime :1047
AudLastBufTime :1047
AudDiscardCnt :0
AudRepeatCnt :0

【调试信息分析】

记录同步信息,高清芯片支持16个播放器,支持16路视频节目同时播放,每个播放器使用单独的sync。每一个sync都会在/proc/msp生成一个节点,调试的时候需要注意选择正确的sync。

参数	描述
Hisilicon SYNC ATTR	记录同步设置的同步属性。
	SyncPrint: 打印是否打开: 1打开、0关闭;
	SyncRef: 同步控制模式,见 HI_UNF_SYNC_REF_E定义;
	SyncStart.VidPlusTime: 同步起调区间视 频同步超前的时间范围;
	SyncStart.VidNegativeTime: 同步起调区 间视频同步落后的时间范围;
	SyncStart.bSmoothPlay: 同步起调区间慢 放使能标识: 1使能, 0未使能;
	SyncNovel.VidPlusTime: 同步异常区间 视频同步超前的时间范围;
	SyncNovel.VidNegativeTime: 同步异常区 间视频同步落后的时间范围;
	SyncNovel.bSmoothPlay: 同步异常区间 慢放使能标识: 1使能, 0未使能;
	VidPtsAdjust:视频pts调整;
	AudPtsAdjust:音频pts调整;
	PreSyncTimeoutMs: 预同步超时时间;
	bQuickOutput:快速输出第一帧使能标识:1使能,0未使能。

参数	描述
Hisilicon PCR	记录PCR的相关信息:
	CrtStatus: 当前工作状态: STOP/PLAY/ TPLAY/PAUSE;
	PreSyncStartSysTime: 预同步起始时间;
	PreSyncEndSysTime: 预同步结束时间;
	PreSyncFinish: 预同步完成标识: 1完成, 0未完成;
	BufFundEndSysTime: 音视频数据积攒结束时间;
	BufFundFinish: 音视频数据积攒完成标识: 1完成, 0未完成;
	PreSyncTarget: 预同步目标: AUD/VID/PCR;
	PreSyncTargetTime: 预同步目标时间;
	PcrFirstCome: PCR是否到来, 1到来, 0 未带来;
	PcrFirstSysTime: PCR第一次到来的系统时间;
	PcrFirst: PCR第一次的时间值;
	PcrLast: PCR上一次的时间值;
	PcrLocalTime: PCR设置的本地时间;
	PcrAdjustMode: PCR同步模式。只有当 SyncRef为PCR时有效。0表示pcr修正本 地时钟,1表示APTS修正本地时钟;
	PcrAudSyncOK: PCR和音频是否已经同步完成: 1完成, 0未完成;
	PcrVidSyncOK: PCR和视频是否已经同步完成: 1完成, 0未完成。

参数	描述
Hisilicon VID	记录视频相关信息:
	VidFirstCome:视频第一帧是否到来:1 到来,0未到来;
	VidFirstSysTime: 视频第一帧到来的系统 时间;
	VidFirstPts: 视频第一帧的pts;
	VidLastPts: 视频上一帧的pts;
	VidPreSyncTargetInit: 预同步过程视频是 否已经到来,1到来,0未到来;
	VidPreSyncTargetTime: 预同步过程视频 到来的时间;
	VidFirstPlay: 视频第一帧是否播放: 1播 放,0未播放;
	VidFirstPlayTime:视频第一帧播放的系统时间;
	VidBlockFlag: 视频缓冲阻塞标记: 1阻 塞,0未阻塞;
	VidBufPercent: 视频缓冲百分比;
	VidDiscard: 视频进入丢帧模式标记: 1 进入,0未进入;
	VidSyndAdjust: 视频进入同步调节标记: 1进入,0未进入;
	VidLocalTime: 视频Pts设置的本地时间;
	VidPcrDiff: 同步过程中视频与PCR的时间差;
	VidAudDiff: 同步过程中视频与音频的时间差。
	VidDiscardCnt: 同步调节过程中的视频 丢帧数;
	VidRepeatCnt: 同步调节过程中的视频重 复帧数;

参数	描述
Hisilicon AUD	记录音频相关信息:
	AudFirstCome: 音频第一帧是否到来标记: 1到来,0未到来;
	AudFirstSysTime: 音频第一帧到来的系统时间;
	AudFirstPts: 音频第一帧的pts;
	AudLastPts:音频上一帧的pts;
	AudPreSyncTargetInit: 预同步过程音频 是否已经到来,1到来,0未到来;
	AudPreSyncTargetTime: 预同步过程音频 到来的时间;
	AudFirstPlay: 音频第一帧是否播放: 1播 放,0未播放;
	AudFirstPlayTime: 音频第一帧播放的系 统时间;
	AudBlockFlag: 音频缓冲阻塞标记: 1阻塞, 0未阻塞;
	AudBufPercent: 音频缓冲百分比;
	AudLocalTime: 音频pts设置的本地时间;
	AudPcrDiff: 同步过程中音频与PCR的时间差;
	AudBufTime: AO缓存数据播放时间;
	AudLastBufTime: AO上一时刻缓存数据播放时间
	AudDiscardCnt: 同步调节过程中的音频 丢帧数;
	AudRepeatCnt: 同步调节过程中的音频 重复帧数。

【调试命令】

● 获取帮助

echo help > /proc/msp/sync00

● 修改同步参考

echo SyncRef = audio|pcr|scr|none > proc/msp/syncxx

● 修改同步起调区间

echo SyncStart.VidPlusTime = xxx > /proc/msp/syncxx
echo SyncStart.VidNegativeTime = xxx > /proc/msp/syncxx
echo SyncStart.bSmoothPlay = true|false > /proc/msp/syncxx

● 修改同步止调区间

echo SyncNovel.VidPlusTime = xxx > /proc/msp/syncxx
echo SyncNovel.VidNegativeTime = xxx > /proc/msp/syncxx
echo SyncNovel.bSmoothPlay = true|false > /proc/msp/syncxx

- 修改预同步超时时间 echo PreSyncTimeoutMs = xxx > /proc/msp/syncxx
- 修改第一帧是否快速输出 echo bQuickOutput = true|false > /proc/msp/syncxx

【调试说明】

VidAudDiff描述了视频和音频pts的差值,如果该值在起调区间 [SyncStart.VidNegativeTime, SyncStart.VidPlusTime]外,则说明当前处于不同步状态,需要进行同步调整。

VidDiscardCnt和VidRepeatCnt反映当前的同步视频调整过程,VidDiscardCnt有增加表明同步有丢帧,VidRepeatCnt有增加表明同步在重复帧,都会引起视频播放的异常。

2.6 WINDOW

【调试信息】

cat /proc/msp/windowXXYY ------Win0100[Z=0]-----------Frame Info-----Frame Info-----Enable:True|Type/PixFmt:2D/NV21 State:Run|Rotation:Rotation 00 Type:Main|W/H(Aspect W:H):1920/1080(16:9) *LayerID:0|Disp(X/Y/W/H):0/0/1920/1080 AspectRatioConvert:Full|FrameRate:500.50000 CustAspectRatio:0:0|ColorSpace:BT601 YUV LIMITED Crop:False|Fieldmode(Origin):Frame(Top) Crop(L/T/R/B):0/0/0/0 | OriRect(X/Y/W/H):0/0/720/576 In(X/Y/W/H):0/0/0/0 | FrameIndex:0x63 Out(X/Y/W/H):0/0/0/0 | SrcPTS/PTS:0xffffffff/0x4a41d52 DispMode/RightFirst:2D/False|PlayTime:1 *Masked:False|FieldMode:All AttachSource:False|Fidelity:0 *CallBack(Acquire) :N|YAddr/YStride:0x24e41000/0x900 *CallBack(Release):Y|CAddr/CStride:0x250a0800/0x900 *CallBack(SetAttr) :Y| SlaveWinID:0000| bOuickoutMode:0| bVirtualCoordinate:1|

bDcmp:0

bMute:0|

-----Buffer State-----

Queue(Try/OK):213/213

Release(Try/OK):194/194

Config:155

Underload:81631

Discard:58

UndispFrame(Q/DQ):4/4

*FieldUnmatchCnt:40880

BufferQueue[state, FrameID]

(State: 1,Empty[15]; 2,Write[0]; 3,ToDisp[0]; 4,Disp[1]; 5,Disped[0])

[1,0x62] [1,0x5c] [1,0x61] [1,0x62]

[1,0x61] [1,0x5c] [1,0x5e] [4,0x63]

[1,0x5d] [1,0x5b] [1,0x5d] [1,0x5f]

[1,0x60] [1,0x5f] [1,0x5e] [1,0x60]

【调试信息分析】

记录某个Window的状态,每一个window都会在/proc/msp生成一个节点,调试的时候需要注意选择正确的window。

msp目录下的"windowXXYY"节点的中的XX为显示通道的编号,YY为window序号。比如window0100,bit[15:8]的01表示该window基于DISPLAY1创建,bit[7:0]的00表示window序号为0。

【参数说明】

window属性配置信息。

参数	描述
Enable	TRUE: 使能状态 FALSE: 未使能状态
State	当前工作状态
	Run
	Pause
	FreezeLast
	FreezeBlack

参数	描述
Туре	窗口类型
	Display: 独立显示窗口
	Virtual: 虚拟窗口
	Main: 同源显示主窗口
	Slave: 同源显示从窗口
LayerID	窗口使用的视频层序号
AspectRatioConvert	宽高比转换模式
	Full: 填满
	LetterBox: 加黑边
	PanAndScan: 裁剪
	Combined: 混合
	FullHori: 水平填满
	FullVert: 垂直填满
	Customer: 用户自定义,此时通过CustAspectRatio强制指定的宽高比例
CustAspectRatio	用户自定义的视频显示比例
Crop	TRUE: 使用Crop
	FALSE: 不使用Crop
In (X/Y/W/H)	视频图像显示区域坐标,Crop为FALSE时有效
Crop(L/T/R/B)	视频图像上、下、左、右裁剪宽度,Crop为TRUE时有效
Out (X/Y/W/H)	视频图像在虚拟屏幕上显示位置座标
DispMode	显示模式
	2D
	FPK
	SBS_HALF
	TAB
	FILED_ALTE
	LINE_ALTE
	SBS_FULL
	L_DEPTH
	LDEP_GDEP
RightFirst	TRUE: 右眼优先,显示顺序为右眼->左眼
	FALSE: 左眼优先,显示顺序为左眼->右眼
Masked	TRUE: 被屏蔽状态
	FALSE: 未被屏蔽状态

参数	描述
AttachSource	TRUE: 窗口与数据源绑定
	FALSE: 窗口未与数据源绑定
CallBack(Acquire)	TRUE: 绑定请求视频帧函数
	FALSE: 未绑定请求视频帧函数
CallBack(Release)	Y: 绑定释放视频帧函数
	N: 未绑释放求视频帧函数
CallBack(SetAttr)	Y: 绑定设置视频帧属性函数
	N: 未绑定设置视频帧属性函数
SlaveWinID	FFFF: 未绑定从窗口; 其他值,从窗口ID
bQuickoutMode	TRUE:窗口处于快速输出状态;
	FALSE: 窗口处于正常状态;
bVirtualCoordinate	TRUE:窗口采用了DISPLAY中设置的虚拟屏幕坐标系
	FALSE: 窗口坐标系采用的是物理坐标系, 和当前
	DISPLAY输出的制式分辨率相关。
bDcmp	TRUE: 窗口解压缩功能打开
	FALSE: 窗口解压缩功能关闭
bMute	TRUE: 窗口黑帧功能打开
	FALSE: 窗口黑帧功能关闭

当前显示的视频帧信息

参数	描述
Type	帧类型
	NotStereo
	SideBySide
	TopAndBottom
	MVC
PixFmt	像素格式
Rotation	旋转度数:
	"Rotation_00": 旋转0度
	"Rotation_90 ": 旋转90度
	"Rotation_180 ": 旋转180度
	"Rotation_270 ": 旋转270度

参数	描述
W/H(WvsH)	W: 图像水平像素数目
	H: 图像垂直像素数目
	WvsH: 图像显示宽高比
Disp(X/Y/W/H)	图像可显示区域
FrameRate	视频帧率
ColorSpace	视频色彩空间
Fieldmode(Origin)	图像的原始场标识和当前场标识
	Тор
	Bottom
	All
OriRect(X/Y/W/H)	图像的原始显示区域(分辨率)
FrameIndex	图像帧序号
SrcPTS/PTS	图像的原始时间戳和当前时间戳
PlayTime	图像需要显示的次数
FieldMode	当前视频帧包含的数据:顶场,底场,顶场+底场
Fidelity	保真处理标识
Y/CAddr	图像的亮度、色度分量物理地址
Y/CStride	图像的亮度、色度分量行间距

window buffer状态信息

参数	描述
Queue(Try/OK)	应用推送视频帧的计数和推送成功计数
Release(Try/OK)	window释放视频帧计数和释放成功计数
Config	显示视频帧计数
Underload	欠载计数
Discard	忽略的视频帧计数
UndispFrame(Q/DQ)	不需要显示的视频帧I/O计数
FieldUnmatchCnt	视频场序与显示接口场序不匹配计数
Empty	空buffer数目
Write	执行写入操作的buffer数目

参数	描述
ToDisp	准备显示的buffer数目
Disp	正在显示的buffer数目
Disped	显示完毕等待释放的buffer数目
Empty(Read/Write)	空闲队列读写指针
Full(Read/Write)	视频帧队列读写指针
BQState	buffer队列状态
[state, FrameID]	State: 1,Empty; 2,Write; 3,ToDisp; 4,Disp; 5,Disped
	FrameID: 当前容纳的视频帧ID

如果在视频播放期间,视频有一种很剧烈的抖动或者回退现象,请查看Underload次数是否有增加。如果Underload次数有增加,说明Window有欠载,会有重复显示。一般情况下,Underload次数不应该增加(除了刚刚开始播放的时候可能有若干帧欠载)。

【调试命令】

● 获取帮助:

echo help> /proc/msp/winXXXX

● 暂停/恢复window:

echo pauseon/off> /proc/msp/winXXXX

● 采用黑帧或者静帧方式复位window:

echo resetblack/still> /proc/msp/winXXXX

- 采用黑帧或者静帧方式冻结window: echo freezeblack/still/off > /proc/msp/winXXXX
- 向上/下移动window:

echo orderup/down> /proc/msp/winXXXX

● 使能/停止快速输出模式:

echo quickon/off> /proc/msp/winXXXX

- 设置window景深(仅在3D显示模式下生效): echo depthX> /proc/msp/winXXXX
- 捕获当前显示的视频图像,并存储到指定路径(使用绝对路径): echo capture path> /proc/msp/winXXXX
- 设置window旋转:

echo rota0/90/180/270> /proc/msp/winXXXX

● 设置window翻转:

echo flip hori/vert> /proc/msp/winXXXX

2.7 DISPLAY

【调试信息】

cat /proc/msp/disp1

------Hisilicon DISP 1 State-----

State:Open

Formt/DispMode:1080i50/2D

```
RightEyeFirst:Disable
VirtualScreen:1280/720
Offset (L/T/R/B):0/0/0/0
AspectRatioMode:Custmer Setting
AspectRatio:16:9
ColorSpace:BT709_YUV_LIMITED->BT709_YUV_LIMITED
Bright:50
Contrast:50
Saturation:50
Hue:50
Background (R/G/B):0x0/0x0/0x0
Zorder(Bot->Top):VIDEO->GFX
AttachRole:source
AttachDisp:display0
Interface:YPbPr0 (0/1/3) HDMI0
InitCount:2
OpenCnt[User/Kernel]:1/1
LowbandCount:0
-----CAST Info-----
State:Enable
Crop:False
CropRect (L/T/R/B):0/0/0/0
Resolution:1280/720
PixelFormat:NV21
FrameRate:2500
LowDelay: Enable
MemoryType:DispAllocate
BufferNumber :5
BufferWidth/Height:1280/720
BufferSize:0x0
BufferStride: 0
bAttached:0
 -----Buffer-----
Acquire(Try/OK):0/0
Release(Try/OK):0/0
BufferQueue: [state, FrameID]
(State: 1, Empty[0]; 2, Write[0]; 3, Full[5]; 4, Use[0]
[3,0x1] [3,0x2] [3,0x3] [3,0x4]
[3,0x5]
```

【调试信息分析】

调试信息记录了DISPLAY模块的配置和运行状态。高清芯片支持2个显示通道同时工作,disp0为DISPLAY0的 proc,disp1为DISPLAY1的proc。

【参数说明】

display调试信息

参数	描述
Open	显示通道打开状态
	Open: 关闭
	Close: 打开

参数	描述
Formt	显示通道输出制式
	1080P60
	1080P50
	1080P30
	1080P25
	1080P24
	1080i60
	1080i50
	720P60
	720P50
	576P50
	480P60
	PAL
	NTSC
DispMode	显示模式
	2D
	FPK
	SBS_HALF
	TAB
	FILED_ALTE
	LINE_ALTE
	SBS_FULL
	L_DEPTH
	LDEP_GDEP
RightEyeFirst	是否设置右眼优先
	TRUE: 右眼优先显示
	FALSE: 左眼优先显示
VirtualScreen	虚拟屏幕分辨率
Offset (L/T/R/B)	虚拟屏幕在物理屏幕上的偏移,依次为左、上、右、下四个方向
AspectRatioMode	显示设备的屏幕比例模式
	Auto: 采用输出分辨率的比例
	User: 用户设置
AspectRatio	显示设备的屏幕比例

参数	描述
ColorSpace	叠加色彩空间与输出色彩空间
	BT601_YUV_LIMITED
	BT601_YUV_FULL
	BT601_RGB_LIMITED
	BT601_RGB_FULL
	NTSC1953
	BT470_M
	BT470_BG
	BT709_YUV_LIMITED
	BT709_YUV_FULL
	BT709_RGB_LIMITED
	BT709_RGB_FULL REC709
	SMPT170M
	SMPT240M
	BT878
	XVYCC
	JPEG
Bright	亮度0~100
Contrast	对比度0~100
Saturation	饱和度0~100
Hue	色调0~100
Background (R/G/B)	背景色R/G/B三分量设置,各分量取值0~255
Zorder(Bot->Top)	视频层、图形层顺序
AttachRole	绑定关系
	source: 作为源显示通道
	destination: 作为目标显示通道
AttachDisp	被绑定的显示通道ID
Interface	视频输出接口
	CVBS0: VDAC编号,255表示未使用
	YPbPr0:后面依次为Y/Pb/Pr信号的VDAC编号
	HDMI0: HDMI编号
InitCount	初始化次数计数
OpenCnt[User/ Kerne]	显示通道打开次数计数(用户态计数/内核态计数)

参数	描述
LowbandCount	低带宽中断计数

CAST调试信息

参数	描述
State	CAST状态
	Enable: 使能
	Disable: 停止
Crop	是否对屏幕进行裁剪
	TRUE: 裁剪
	FALSE: 不裁剪
CropRect(L/T/R/B)	裁剪区域设置,依次为左/上/右/下四个方向裁剪像素数目, Crop为TRUE时有效
Resolution	图像分辨率
PixelFormat	图像像素格式
FrameRate	图像帧率
LowDelay	是否使能低延迟
	TRUE: 使能
	FALSE: 比使能
MemoryType	图像帧存类型
	DispAlloc: CAST模块提供
	UserAlloc: 用户提供
BufferNumber	帧存数目
BufferWidth/Height	帧存宽度和高度,以像素为单位,图像分辨率不能大于帧存 宽度和高度
BufferSize	每个图像帧存大小
BufferStride	图像相邻行间距
bAttached	是否采用绑定模式
	TRUE: cast和消费者绑定在一起,收帧和还帧在驱动内部通过回调来实现。
	FALSE: 分离模式,帧的收发通过在UNF接口层调用实现。
Acquire(Try/OK)	用户请求图像的次数和请求成功的次数
Release(Try/OK)	用户释放图像的次数和释放成功的次数

参数	描述
Empty	空buffer数目
Full	满buffer数目
Use	用户正在使用的buffer数目
[state, FrameID]	buffer队列状态 state: 1,Empty; 2,Write; 3,Full; 4,Use FrameID: 图像ID

【调试命令】

● 调试帮助:

echo help> /proc/msp/dispX

● 切换输出制式:

echo fmt 1080i50/720p50/pal/ntsc/... > /proc/msp/dispX

● 切换输出模式:

echo 3d fp/sbs_hf/tab> /proc/msp/dispX

● 设置3D模式下右眼优先:

echo rf on/off> /proc/msp/dispX

设置亮度:

echo brightX> /proc/msp/dispX

● 设置对比度:

echo contrast X> /proc/msp/dispX

● 设置色调:

echo hueX> /proc/msp/dispX

● 设置饱和度:

echo satuX> /proc/msp/dispX

- 设置画面左方偏移像素数目: echo leftX> /proc/msp/dispX
- 设置画面上方偏移像素数目: echo topX> /proc/msp/dispX
- 设置画面右方偏移像素数目:
- echo rightX> /proc/msp/dispX

 设置画面下方偏移像素数目:
- echo bottomX> /proc/msp/dispX
- 向上/下移动视频层叠加顺序: echo video up/down> /proc/msp/dispX
- 复位参数为默认值:

echo reset> /proc/msp/dispX

● 将屏幕图像捕获存为文件,存储到指定路径下(使用绝对路径): echo cat AbsolutePath> /proc/msp/dispX

2.8 ADEC

【调试信息】

cat /proc/msp/adec00

----- ADEC[00] State -----

WorkState:start CodecID:0x202f1011

DecoderName: Dolby TrueHD Decoder

Description:hisi_truehd DecodeThreadID:2537

Volume:100 SampleRate:96000 BitWidth:24 Channels:2

PcmSamplesPerFrame:1840 BitsBytePerFrame:0x0 StreamFormat:non-packet TryDecodetimes:1088

FrameNum(Total/Error):767/0

StreamBuf(Total/Use/Percent)(Bytes):262144/205883/78%

StreamBuf(readPos/writePos):0xdbcc/0x9000
OutFrameBuf(Total/Use/Percent):8/7/87%

GetBuffer(Try/OK):4967/3529 PutBuffer(Try/OK):3529/3529 SendStream(Try/OK):0/0 ReceiveFrame(Try/OK):767/760

PtsLostNum:0

DecodeThreadExecTimeOutCnt:1
DecodeThreadScheTimeOutCnt:0

【调试信息分析】

显示ADEC的状态。

参数	描述
WorkState	ADEC工作状态。start/stop分别标识工作与非工作状态
Codec ID	音频解码类型。括号内是HA_AUDIO_ID的值
DecoderName	音频解码器类型的名称
Description	解码器描述
DecodeThreadID	解码线程ID,客户无需关注此信息
Volume	音量大小。这个音量大小是指解码器解码出来的音量大小,与音频输出的音量大小没有关系。这个值是只读的,没有接口可以配置
SampleRate	码流解码出的PCM数据的采样率,由码流本身属性决定
BitWidth	码流解码出的PCM数据的位宽,由码流本身属性决定
Channels	码流解码出的PCM数据的声道数,由码流本身属性决定
PcmSamplesPerFrame	码流解码出的每帧PCM数据采样点数,客户无需关注此信息
BitsBytePerFrame	码流解码出的每帧透传数据大小,客户无需关注此信息
StreamFormat	是否为packet模式

参数	描述	
TryDecodetimes	尝试解码缓冲区中没有解码过的数据的次数,用户一般不需要关注	
FrameNum(Total/Error)	音频解码器共解码的帧数以及错误帧数	
StreamBuf(Total/Use/ Percent)(Bytes)	音频解码器输入缓存的总大小,已经使用大小,已经使用 的百分比	
StreamBuf(readPos/ writePos)	音频解码器输入缓存读写指针位置	
OutFrameBuf(Total/Use/ Percent)	音频解码器输出帧存的总大小,已经使用大小,已经使用 的百分比	
GetBuffer(Try/OK)	Try: 用户从ADEC获取Es Buffer的次数 OK: 成功获取次数 在ES播放模式下有效	
PutBuffer(Try/OK)	Try: 用户送Es buffer给ADEC的次数OK: 成功次数在ES播放模式下有效	
SendStream(Try/OK)	Try: Avplay送音频Es数据给ADEC的次数 OK: 成功次数 Avplay从demux获取音频ES数据,送ADEC解码,TS模式下有效	
ReceiveFrame(Try/OK)	Try: Avplay从ADEC获取解码完的音频帧的次数 OK: 成功获取次数 Avplay从ADEC获取音频帧数据,送AO进行播放	
PtsLostNum	Pts丢失次数	
DecodeThreadExecTime OutCnt	Adec解码线程执行超时(30ms)的次数	
DecodeThreadScheTime OutCnt	Adec解码线程调度超时(30ms)的次数	

在ES播放模式下,用户通过调用HI_UNF_AVPLAY_GetBuf向ADEC送音频Es数据进行解码播放。

- 如果GetBuffer的Try值保持不变,说明用户没有调用HI_UNF_AVPLAY_GetBuf进 行数据注入:
- 如果PutBuffer的OK值不变,说明注入数据失败,可能是ADEC的buffer已满。

【调试信息】

```
#echo save es start > /proc/msp/adec00
#echo save es stop > /proc/msp/adec00
#echo save pcm start > /proc/msp/adec00
```

#echo save_pcm stop > /proc/msp/adec00
#echo help > /proc/msp/adec00

【调试信息分析】

通过Proc录制解码前的ES数据:

echo save es start|stop > /proc/msp/adecXX(XX为adec通道数)

● 启动录制

echo save_es start > /proc/msp/adec00 向adec00发送存ES数据的命令(00是adec通道的编号)。

● 停止录制

echo save es stop > /proc/msp/adec00

- 成功录制之后,可以在设置的目录下找到如下类型文件: adec0 00.es, adec0 01.es。
- adec0 00.es: 第0次存adec00的ES数据;
- adec0_01.pcm: 第1次存adec00的ES数据;

通过Proc录制解码后的PCM数据:

echo save pcm start|stop > /proc/msp/adecXX

- 启动录制
 - echo save_pcm start > /proc/msp/adec00 向adec00发送存解码后 PCM数据的命令。
- 停止录制
 - echo save pcm stop > /proc/msp/adec00
- 成功录制之后,可以在设置的目录下找到如下类型文件: adec0_00.pcm, adec0_01.pcm。
- adec0_00.pcm: 第0次存adec00的PCM数据;
- adec0 01.pcm: 第1次存adec00的PCM数据;

显示Proc帮助信息:

echo help > /proc/msp/adecXX

2.9 AO

【调试信息】

```
SampleRate(048000), Channel(02), BitWidth(16), Engine(SPDIF RAW),
AOP(0x1), PortID(0x21)
DmaCnt(000828), BufEmptyCnt(000000), FiFoEmptyCnt(000001)
HDMI0: Status(start), Mute(off), Volume(00dB), TrackMode(STEREO)
SampleRate(192000), Channel(02), BitWidth(16), Engine(SPDIF RAW),
AOP(0x3), PortID(0x20)
DmaCnt(000756), BufEmptyCnt(000000), FiFoEmptyCnt(000000)
12S0: Status(start), Mute(off), Volume(00dB), TrackMode(STEREO)
SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x4),
PortID(0x10)
DmaCnt(000850), BufEmptyCnt(000000), FiFoEmptyCnt(000000)
----- Cast Status-----
Cast(0): *Aop(0x5), Status(start), UserEnable(On), Weight(100/100),
Mute (off)
SampleRate(048000), Channel(02), BitWidth(16)
MaxFrameNum(08), SamplePerFrame(01024), AcquireFrame(On)
----- Track Status-----
Track(0): Type(master), Status(start), Weight(100/100),
ChannelMode (STEREO), Mute (off), StartThresHold (050ms)
SpeedRate(00), AddMuteFrames(0000), SendCnt(Try/OK)(001112/000322)
AIP(0): Engine(PCM), SampleRate(048000), Channel(02), BitWidth(16),
DataFormat(PCM) , Priority(low), FifoBypass(off)
EmptyCnt(000000), EmptyWarningCnt(000000), Latency/Threshold(378ms/400ms)
AIP(1): Engine(SPDIF RAW), SampleRate(048000), Channel(02),
BitWidth(16), DataFormat(DD), Priority(low), FifoBypass(off)
EmptyCnt(000000), EmptyWarningCnt(000000), Latency/Threshold(388ms/400ms)
AIP(2): Engine(HDMI RAW), SampleRate(192000), Channel(02), BitWidth(16),
DataFormat(DDP) , Priority(low), FifoBypass(off)
EmptyCnt(000000), EmptyWarningCnt(000000), Latency/Threshold(348ms/400ms)
```

【调试信息分析】

cat /proc/msp/soundN显示第N个音频输出设备的状态。

参数	描述
SampleRate	输出设备的采样率。
SPDIF Status	SPDIF输出状态: UserSetMode(RAW): 用户设置的SPDIF输出模式。 DataFormat(DD): SPDIF实际输出的数据格式(如PCM、DD、DTS等)。
HDMI Status	HDMI输出状态 UserSetMode(RAW):用户设置的HDMI输出模式。 DataFormat(DDP): HDMI实际输出的数据格式(如 PCM、DD、DDP、DTS、DTSHD、TRUEHD等)。

参数	描述
ADAC0 (SPDIF0,	Sound设备绑定的各个端口的状态:
HDMI0)	Status(start): 端口Start/Stop状态。
	Mute(off/0x0):端口静音状态,Off:端口静音开启与否,取值范围{On,Off}。
	Volume(00dB): 端口音量值。
	TrackMode(STEREO): 端口声道模式。
	SampleRate(048000): 端口的输出采样率。
	Channel(02): 端口的输出声道数。
	BitWidth(16): 端口的输出采样位宽。
	Engine(PCM):端口绑定哪一个AOE引擎,用户不需要关注。
	AOP(0x0):端口绑定哪一个AOP,用户不需要关注。
	PortID(0x12):端口绑定哪一条输出通道,用户不需要关注。
	DmaCnt(002245): Dma搬运次数,查看端口有无输出,先 看此项计数是否增加。
	BufEmptyCnt(000000): 端口Buffer下溢中断次数。
	FiFoEmptyCnt(000000): 端口Fifo中无数据中断次数。
Cast(n)	Cast状态(n是Cast编号):
	Aop(n): Cast使用的Aop编号,用户不需要关注。
	Status(start): Cast状态。
	UserEnable(On): Cast是否使能。
	Weight(100/100): Cast权重(即Cast左右声道音量)。
	Mute(off):端口静音状态,Off:端口静音关闭,取值范围{On,Off}。
	SampleRate(048000): 端口的输出采样率。
	Channel(02): 端口的输出声道数。
	BitWidth(16): 端口的输出采样位宽。
	MaxFrameNum(08): 最大帧数。
	SamplePerFrame(01024): 每次读取的PCM采样点数量。
	AcquireFrame(On): 当前读取状态。取值范围{On,Off}。

参数	描述
Track(n)	Track状态(n是Track编号):
	Type(master): Track类型。
	Status(start): Track Start/Stop/Pause状态。
	Weight(100/100): 左声道和右声道Track权重(即Track音量)。
	ChannelMode(STEREO): Track的声道模式
	Mute(off): Track静音状态, off: 关闭Track静音, 即有声音输出。取值范围{on,off}。
	SpeedRate(00): Track调速速率。
	AddMuteFrames(0000): 增加的透传静音帧帧数。
	SendCnt(Try/OK): 往Track送数据的尝试次数/成功次数。
	AIP(n): Track使用AIP状态(n是AIP编号),用户不需要 关注。Engine(PCM): AIP绑定哪个AOE引擎; SampleRate(048000): AIP输入采样率; Channel(02): AIP 输入通道数; BitWidth(16): AIP输入采样位宽; DataFormat(PCM): AIP数据格式; Priority(low): AIP优先 级; FifoBypass(off): AIP数据是否不经过FIFO。
	EmptyCnt(000000): AIP Buffer无数据次数; EmptyWarningCnt(000000): AIP Buffer有数据时, AIP Fifo中无数据的次数。Latency/Threshold(328ms/400ms): AIP Buffer缓冲数据量及缓冲数据量的限制值(单位: ms)
	StartThresHold: track启动水线,仅master类型track拥有该属性(单位: ms)。

【调试信息】

```
#echo save_track 0 start > /proc/msp/sound0
#echo save track 0 stop > /proc/msp/sound0
#echo save_sound start > /proc/msp/sound0
#echo save_sound stop > /proc/msp/sound0
#echo help > /proc/msp/sound0
```

【调试信息分析】

通过Proc录制Track的PCM数据:

echo save_track track_id start|stop > /proc/msp/soundN(track_id为Track的编号,N为sound的编号)

- 启动录制
 echo save_track 0 start > /proc/msp/sound0
 向Sound0发送存Track PCM数据的命令。
- 停止录制 echo save track 0 stop > /proc/msp/sound0

● 成功录制之后,可以在设置的目录下找到如下命名格式的文件:

track0_hh_mm_ss.pcm: 系统时间为hh(小时):mm(分钟):ss(秒)时存储的Track 0的PCM数据。

通过Proc录制Sound的PCM数据:

echo save sound start|stop > /proc/msp/soundN (N为sound的编号)

- 启动录制
 - echo save_sound start > /proc/msp/sound0 向Sound0发送存Sound PCM数据的命令。
- 停止录制 echo save_sound stop > /proc/msp/sound0
- 成功录制之后,可以在设置的目录下找到如下命名格式的文件:
- sound0_hh_mm_ss.pcm: 系统时间为hh(小时):mm(分钟):ss(秒)时存储的Sound 0的PCM数据。

显示Proc帮助信息:

echo help > /proc/msp/soundN

【设置HDMI、SPDIF输出模式】:

通过/proc/msp/soundN强制设置HDMI、SPDIF输出模式:

- 将sound0 HDMI输出设置为解码输出 echo hdmi=pcm > /proc/msp/sound0
- 将sound0 HDMI输出设置为AUTO, 自动识别外设能力 echo hdmi=auto > /proc/msp/sound0
- 将sound0 HDMI输出设置为蓝光次时代降级输出 echo hdmi=hbr2lbr > /proc/msp/sound0
- 将sound0 HDMI输出设置为透传输出 echo hdmi=raw > /proc/msp/sound0
- 将sound0 SPDIF输出设置为解码输出 echo spdif=pcm > /proc/msp/sound0
- 将sound0 SPDIF输出设置为透传输出echo spdif=pcm > /proc/msp/sound0

2.10 HiGo

【调试信息】

cat /proc/msp/higo/sys
Time: Oct 25 2011 09:17:57
Version: 3.0.0.1 a12
SurfaceUsed: 1
LayerUsed: 1
MmzMemUsed: 50148B
SysMemUsed: 732B

cat /proc/msp/higo/layer

----- HIGO LAYER HD 0 ------

Visible: TURE Format: 8888 FlushType: TRIPPLE Position: (0,0) Defelicker: AUTO

CanvasSurface: type(Default) resolution(1280,720)

DisplaySurface: resolution (1280, 720)

Alpha:255 BGColor: 0x0 Colorkey: 0x0 StereoDepth: 0x0

cat /proc/msp/higo/surface

number width height pitch phyaddrsize(KB) formatalpha colorkey

0000128072051200x87823000 3600HIGO_PF_8888 255false

【调试信息分析】

记录HIGO模块的所有调试信息。

参数	描述
sys info	系统信息。
	Time: 版本时间;
	Version: 版本号;
	SurfaceUsed: 使用的surface的个数;
	LayerUsed: 使用的图层的个数;
	MmzMemUsed: 使用的MMZ内存的大小;
	SysMemUsed: 使用操作系统内存的大小。
layer Info	Visible:显示状态;
	Format: 图层像素格式;
	FlushType: 刷新类型;
	Position: 起始位置;
	Defelicker: 抗闪烁级别;
	CanvasSurface: canvas surface类别,分辨率;
	DisplaySurface: display surface的分辨率;
	Alpha: alpha信息;
	BGColor: 背景色;
	Colorkey: colorkey值;
	StereoDepth: 3D景深。

调试指南 2 Proc 信息

参数	描述
surface information	number: surface索引号;
	width: surface宽度;
	height: surface高度;
	pitch: surface行间距;
	phyaddr: surface物理地址;
	size(KB): surface内存大小;
	format: surface像素格式;
	alpha: alpha信息;
	colorkey: colorkey信息。

2.11 HiFB

【调试信息】

查看hifb相应的信息(hifb0、hifb1、hifb2为高清,hifb4为标清)。

```
# cat /proc/msp/hifb0
LayerId:layer_hd_0
ShowState:ON
ColorFormat::ARGB8888
Stride:5120
Offset: (0,0)
Resolution(real/virtual/max): (1280,720)/(1280,720)/(1920,1080)
MemSize:3600 KB
StartPosition: (114, 69)
BufferMode:single
PixelAlpha:enable(TRUE) alpha0(0x0) alpha1(0xff)
GlobalAlpha: 0xff
Colorkey :enable(FALSE) value(0)
Deflicker :enable(FALSE) mode(NONE) level(NONE)
3DMode:input(Mono) output(Mono)
DisplayResolution: (1280, 720)
CanavasAddr:0x0
CanavasUpdateRect: (0,0,1280,720)
CanvasResolution: (0,0)
CanvasPitch:0
CanvasFormat: RGB565
```

【调试信息分析】

该proc是查看HiFB信息,有两个常用的hifb:

- hifb0对应高清图层0
- hifb4对应标清图层0

参数	描述
HIFB Info	LayerId: 图层ID;
	ShowState: 图层状态标志, on表示打开;
	ColorFormat: 像素格式;
	Stride: 行距;
	Offset: 虚拟分辨率中的偏移;
	Resolution(real/virtual/max): 分辨率(实际分辨率/虚拟分辨率/最大分辨率);
	MemSize: 内存大小;
	StartPosition:显示起点位置;
	BufferMode: 刷新模式;
	PixelAlpha: 像素alpha(使能/alpha0/alpha1);
	GlobalAlpha: 全局alpha;
	Colorkey: colorkey(使能/colorkey值);
	Deflicker: 抗闪(使能/模式/级别);
	3DMode: 3D模式(源3D模式/显示3D模式);
	CanavasAddr: canvas buffer地址;
	CanavasUpdateRect: canvas 刷新区域;
	CanvasResolution: canvas分辨率;
	CanvasPitch: canvas行距
	CanvasFormat: canvas像素格式;
	DisplayResolution: display分辨率。

【其他】

```
echo help > /proc/msp/hifb0查看帮助信息
echo show > /proc/msp/hifb0显示图层
echo hide > /proc/msp/hifb0隐藏图层
echo alpha=255 > /proc/msp/hifb0设置图层alpha
echo capture > /proc/msp/hifb0抓取显存图片
```

2.12 TDE

【调试信息】

```
# cat /proc/msp/tde
----- Hisilicon TDE Node params Info -----
--- HWNode update: 0x373189 ---
(INS):
                     0xbc4f0049
(S1 XY):
                     0x1e6
(S2 XY):
                    0xf3
(S2_SIZE):
                    0x384004c
                    0xf002a1
(TAR XY):
                    0x40f0002f
(TS SIZE):
(2D RSZ):
                     0x3040011
(HF_COEF_ADDR): 0x860ad140
```

 (VF_COEF_ADDR):
 0x860ad520

 (RSZ Y OFST):
 0x0

 (RSZ_X_OFST):
 0x3239

【调试信息分析】

TDE操作最终会以节点的方式提交给硬件,节点中包含该操作对应的寄存器配置信息。TDE调试信息记录了最近提交的8个节点(每个HWNode update:代表一个节点)的配置信息。

该调试信息可以通过命令开启:

/proc/msp \$ echo proc on > tde

也可以通过命令关闭:

/proc/msp \$ echo proc off > tde

默认为proc打开。

参数	描述
HWNode update	表示该节点对应的TDE操作要对哪些寄存器进行配置。用两个32位数表示,每个bit代表1个寄存器,1为配置,0为不配置。第0到31bit分别代表如下寄存器: INS、S1_ADDR、S1_TYPE、S1_XY、S1_FILL、S2_ADDR、S2_TYPE、S2_XY、S2_SIZE、S2_FILL、TAR_ADDR、TAR_TYPE,、TAR_XY、TS_SIZE、COLOR_CONV、CLUT_ADDR、2D_RSZ、HF_COEF_ADDR、VF_COEF_ADDR、RSZ_STEP、RSZ_Y_OFST、RSZ_X_OFST、DFE_COEF0、DFE_COEF1、DFE_COEF2、DFE_COEF3、ALU、CK_MIN、CK_MAX、CLIP_START、CLIP_STOP、Y1_ADDR、Y1_PITCH、Y2_ADDR、Y2_PITCH、RSZ_VSTEP、ARGB_ORDER、CK_MASK、COLORIZE、ALPHA_BLEND、ICSC_ADDR、OCSC_ADDR
INS	该操作配置的INS寄存器的值。仅在该操作需要配置INS寄存器时列出,后面的参数依此类推。
S1_ADDR	该操作配置的S1_ADDR寄存器的值。
S1_TYPE	该操作配置的S1_TYPE寄存器的值。
S1_XY	该操作配置的S1_XY寄存器的值。
S1_FILL	该操作配置的S1_FILL寄存器的值。

参数	描述
S2_ADDR	该操作配置的S2_ADDR寄存器的值。
S2_TYPE	该操作配置的S2_TYPE寄存器的值。
S2_XY	该操作配置的S2_XY寄存器的值。
S2_SIZE	该操作配置的S2_SIZE寄存器的值。
S2_FILL	该操作配置的S2_FILL寄存器的值。
TAR_ADDR	该操作配置的TAR_ADDR寄存器的值。
TAR_TYPE	该操作配置的TAR_TYPE寄存器的值。
TAR_XY	该操作配置的TAR_XY寄存器的值。
TS_SIZE	该操作配置的TS_SIZE寄存器的值。
COLOR_CONV	该操作配置的COLOR_CONV寄存器的值。
CLUT_ADDR	该操作配置的CLUT_ADDR寄存器的值。
2D_RSZ	该操作配置的2D_RSZ寄存器的值。
HF_COEF_ADDR	该操作配置的HF_COEF_ADDR寄存器的值。
VF_COEF_ADDR	该操作配置的VF_COEF_ADDR寄存器的值。
RSZ_STEP	该操作配置的RSZ_STEP寄存器的值。
RSZ_Y_OFST	该操作配置的RSZ_Y_OFST寄存器的值。
RSZ_X_OFST	该操作配置的RSZ_X_OFST寄存器的值。
DFE_COEF0	该操作配置的DFE_COEF0寄存器的值。
DFE_COEF1	该操作配置的DFE_COEF2寄存器的值。
DFE_COEF2	该操作配置的S1_FILL寄存器的值。
DFE_COEF3	该操作配置的DFE_COEF3寄存器的值。
ALU	该操作配置的ALU寄存器的值。
CK_MIN	该操作配置的CK_MIN寄存器的值。
CK_MAX	该操作配置的CK_MAX寄存器的值。
CLIP_START	该操作配置的CLIP_START寄存器的值。
CLIP_STOP	该操作配置的CLIP_STOP寄存器的值。

参数	描述
Y1_ADDR	该操作配置的Y1_ADDR寄存器的值。
Y1_PITCH	该操作配置的Y1_PITCH寄存器的值。
Y2_ADDR	该操作配置的Y2_ADDR寄存器的值。
Y2_PITCH	该操作配置的Y2_PITCH寄存器的值。
RSZ_VSTEP	该操作配置的RSZ_VSTEP寄存器的值。
ARGB_ORDER	该操作配置的ARGB_ORDER寄存器的值。
CK_MASK	该操作配置的CK_MASK寄存器的值。
COLORIZE	该操作配置的COLORIZE寄存器的值。
ALPHA_BLEND	该操作配置的ALPHA_BLEND寄存器的值。
ICSC_ADDR	该操作配置的ICSC_ADDR寄存器的值。
OCSC_ADDR	该操作配置的OCSC_ADDR寄存器的值。

2.13 HDMI

【调试信息】

HDMI正常输出时的打印信息如图2-1、图2-2、图2-3、图2-4所示。

图 2-1 HDMI 正常输出时的打印信息一

```
# cat /proc/msp/hdmi0
SDK VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
 ----- APPAttr -----
HDMIEnable : YES
VideoEnable : YES
                                DefaultAction : NONE
                                AudioEnable : YES
AviInfoEnable : YES
                                AudioInfoEnable: YES
xvYCCMode : NO
                                HDCPEnable : NO
DeepColorMode : 24
                                SpdInfoEnable : NO
OutColorSpace : YES
                                MpeqInfoEnable : NO
                                DebugEnable : NO
ColorSpaceAdapt: YES
CtsAuthEnable : NO
                               DrmInfoEnable : NO
  ----- SWStatus --
                                -----
ThreadRun : YES
                                RunStatus
                                            : OPEN START
            : DVI
TMDSMode
KernelCnt : 0
KCallBack : NO
                                UserCnt
                                            : 1
                               UCallBackCnt : 0
TransitState : NONE
                               HDCPEnable : NO
HDCPMode : NONE
HDCP2.2Support : N/A
HDCPIsOn : NO
CECEnable : NO
KevLoad
                                HdcpAuthStatus : NO
                                CECStatus : OFF
CECLogicAddr : 0
CECPhyAddr : 00.00.00.00
                                CECNetwork
            ----- HWStatus -----
HotPlug : YES
                               Rsen : YES
PhyOutputEnable: YES
                                PhyPowerEnable : YES
TMDSMode : HDMI1.4
                                AvMute : NO
SourceScramble : NO
                                SinkScramble : NO
TMDSBitClkRatio: 10
                               DataSwing : 0x16
CLKSwing : 0x15
                                SrcTermination: 0x0
TxVnbValue : 0x2
TxRegZone : 0x82
                                AdjustClk : 0x1
----- TaskID=1126 Event Pool[0] Status ------
CNT|ErrTotal|HPD|UnHPD|EdidFail|HdcpFail|HdcpSucc|RsenCon|RsenDis|HdcpUsr
Memory[WkFlg=0 |RdAble= 0| RdPtr=1 | WrPtr=1 ]:
```

图 2-2 HDMI 正常输出时的打印信息二

```
# cat /proc/msp/hdmi0 ao
SDK VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
----- AudioAttr ----- AudioIfno -----
SoundIntf : I2S
CodeType : STRE
                                  |AudioInfoEnable: YES
                                  |CodeType
|ChannelCnt : 2_CH
|ChannelCnt : 2_CH : 3_CHEADER
             : STREAM
                                  |CodeType : STREAM
            : 2_CH
ChannelCnt
             : 48000
SampleFreq
SampleDepth : 16
                                  |SampleDepth : 16
|SampleSize : STR_HEADER
             : NO
DownSample
------ AudioPath -----|DownMixInhibit : NO
AudioEnable : YES
                                   |LevelShiftValue: 0
             : YES
                                   |LFEPlayBack : UNKNOW
             : I2S
SoundIntf
                                   |Channel/SpeakerAlloc: 0x00 (0)
ChannelCnt : 2_CH
SampleFreq : 48000
                                   |AudioInfoRawData:
                                  SampleDepth : 16
DownSample
CTS
             : N/A
             : N/A
                                   Т
AudioFIFO : N/A
                                   Τ
#
```

图 2-3 HDMI 正常输出时的打印信息三

```
# cat /proc/msp/hdmi0_vo
SDK_VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
            2.0.0.20151212.0
HDMI Version:
 VideoTiming : 1920*1080p60 16:9 |AVIInfoEnable : YES
DispFmt
           : 1080P@60 |CurrentFormat : 1920*1080p60 16:9(VIC=16)
InBitDepth : 10
                               |VSIFormat : (HDMI_VIC= 0)
Colorimetry: ITU-R BT.709 | Colorimetry: ITU-R BT
                                             : ITU-R BT.709
ActAspectRatio : PICTURE
                                |ActAspectRatio : PICTURE
PixelRepeat : 1
                               |PixelRepeat : No Repetition
                               |YCCQuantization: LIMITED
YCCQuantization: LIMITED
RGBQuantization: DEFAULT
                                |RGBQuantization: DEFAULT
                           |ExtColorimetry : XV_YCC601
ExtColorimetry : XV_YCC601
                              |ItContentValid : NO
StereoMode : NONE
HvSyncPol
            : HPVP
                               |HvSyncPol : HPVP
-----|ITContentType : GRAPHICS
VideoMute : NO
                              |PicScaling
                                             : UNKNOW
OutBitDepth : 08 Bit
                               |ActFmtPresent : YES
OutColorSpace : YCbCr444
                               ScanInfo
            : NO
YCbCr420_422
                                |AVIInfoRawData :
YCbCr422 444 : NO
                               | 82 02 0d 67 50 a8 00 10
YCbCr444_422 : NO
                               | 00 00 00 00 00 00 00
YCbCr422_
RGB2YCbCr : NO
: NO
YCbCr422_420 : NO
                                | 00
                                |VSInfoRawData
YCbCr2RGB : NO
Dither : NO
                               | 00 00 00 00
           ----- HDRAttr -----
                               HdrEotfType : SDR LUMIN
HdrMode : DISABLE
HdrMetaDataID : 0
                                HdrColorimetry : NONE
DispPrimO_X : 0
                                DispPrimO_Y : 0
DispPrim1_X : 0
DispPrim2_X : 0
                                 DispPrim1 Y
                                DispPrim2 Y
                                             : 0
WhitePoint_X : 0
                                WhitePoint_Y : 0
MaxLuminance : 0
MaxLightLevel : 0
                                MinLuminance
                                            : 0
                                 AverLightLevel: 0
                   ----- DRMInfo -----
                                EotfType : SDR_LUMIN
DRMInfoEnable : NO
MetadataID : 0
DRMInfoRawData : 00 00 00 00
```

图 2-4 HDMI 正常输出时的打印信息四

```
# cat /proc/msp/hdmi0 sink
SDK_VERSION:[HiSTBLinuxV100R002] Build Time:[Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
                                   -- EDIDRawData -
/*00H:*/ 0x00,0xff,0xff,0xff, 0xff,0xff,0xff,0x00, 0x4c,0x2d,0xbe,0x0b, 0x01,0x00,0x00,0x00,
/*0fH:*/ 0x02,0x18,0x01,0x03, 0x80,0x59,0x32,0x78, 0x0a,0xee,0x91,0xa3, 0x54,0x4c,0x99,0x26,
/*1fH:*/ 0x0f,0x50,0x54,0xbd, 0xef,0x80,0x71,0x4f, 0x81,0xc0,0x81,0x00, 0x81,0x80,0x95,0x00,
/*2fH:*/ 0xa9,0xc0,0xb3,0x00, 0x01,0x01,0x08,0xe8, 0x00,0x30,0xf2,0x70, 0x5a,0x80,0xb0,0x58,
/*3fH:*/ 0x8a,0x00,0x50,0x1d, 0x74,0x00,0x00,0x1e, 0x02,0x3a,0x80,0x18, 0x71,0x38,0x2d,0x40,
/*6fH:*/ 0x00,0x53,0x41,0x4d, 0x53,0x55,0x4e,0x47, 0x0a,0x20,0x20,0x20, 0x20,0x20,0x20,0x01,0x22,
/*7fH:*/ 0x02,0x03,0x3a,0xf1, 0x57,0x61,0x10,0x1f, 0x04,0x13,0x05,0x14, 0x20,0x21,0x22,0x5d,
/*8fH:*/ 0x5e,0x5f,0x60,0x65, 0x66,0x62,0x63,0x64,
                                                 0x07,0x16,0x03,0x12, 0x23,0x09,0x07,0x07,
0x03,0x0c,0x00,0x10, 0x00,0xb8,0x3c,0x21,
/*9fH:*/ 0x83,0x01,0x00,0x00, 0xe2,0x00,0x0f,0x6e,
CapFromSink
                                        RawUpdateErrCnt
                                       ParseErrorType
ParseWarnType
                                                         : 0
CapIsValid
                 : YES
RawGetErrCnt
RawIsValid
                                                          : 0x00000000
                  : YES
                 : 0
                                                         : 256
                                       RawLength
                ----- BasicCap ----
                                       1stBlockVersion : 1.3
HDMI1.4Support : YES
                                       ManufacturerName
HDMI2.0Support
                  : NO
                                                          : SAM
MaxTMDSClock(MHz) : 300
                                       ProductCode
                                                         : 3006
: 1
HDCP1.4Support
                  : N/A
                                       SerialNumber
WeekOfManufacture : 2
SCDCSupport
DVIDualSupport
                                       YearOfManufacture : 2014
                  : NO
                                       CECAddrIsValid : YES
CECAddr : 01.00.00.00
                  : NO
ExtBlockCnt
                  : YES
                 : 1
----- VidoCap --
                                       SpeakerSupport
                                                         : FL FR
NativeFormat : 3840X2160P60 16:9(VIC 97)

ColorSpace : RGB444 YCbCr444 YCbCr422 YCbCr420

DeepColor : RGB 20Bit PCB 26Bit YCbCr444 County
DeepColor
                  : RGB_30Bit RGB_36Bit YCbCr444_SameRGB
YCbCr420DeepColor :
YCbCr420[Also] :
YCbCr420[Olny] :
                 : 97 96 101 102 98
Colorimetry
3840X2160P60 16:9 1080P_60 16:9 1080P_50 16:9 720P_60 16:9
                    1080i_50 16:9 1080p_24 16:9

1080p_30 16:9 3840x2160p24 16:9 3840x2160p25 16:9

3840x2160p50 16:9 4096x2160p50 256:135 4006x3160p25
720P 50 16:9
720P_50 16:9 1080i_60 16:9 1080P_25 16:9 1080P_30 16:9 3840x2160P30 16:9 3840x2160P50 1
                                         4096x2160P50 256:135 4096x2160P60 256:135
4096X2160P24 256:135 4096X2160P25 256:135 4096X2160P30 256:135 NTSC 16:9
PAL 16:9 480P_60 16:9 576P_50 16:9 3840X2160
3840X2160P25 16:9 3840X2160P24 16:9 4096X2160P24 256:135 1152x864
                                                              3840X2160P30 16:9
                   1280x800
                                                       1440x900
                                        1280x1024
1280x720
1600x900
                    1680x1050
                                         V800X600 60
                                                              V800X600 56
                                        ∀640X480 67
V640X480 75
                    V640X480 72
                                                              V640X480 60

        V720X400_88
        V720X400_70
        V1280X1024_75

        V1024X768_70
        V1024X768_60
        V1024X768_87

                                                             V1024X768 75
                                         V1280X1024 75
       3DSupport : NO
3DDualView : NO
                                       3DOsdDisparity : NO
                   : NO
                                          3DIndepView
3DTypeSupport
                  :
                    ----- AudioCap --
NO.0:
CodeType
                   : L-PCM
                                          MaxChannelNum
MaxBitRate(KHz) : N/A
                                          BitDepth
                                                             : 16 20 24
                   : 32000 44100 48000
SampleRate(Hz)
              ----- DetailTiming -----
[NO.]:HACT|VACT|P/I |PC1k|AspW|AspH|HFB |HPW |HBB |VFB |VPW |VBB |ImgW|ImgH|IHS |IVS |IDV
```

【调试信息分析】

记录HDMI输出管理模块信息。

Parameter		Description
APPAttr	HDMIEnable	HDMI是否使能。
		取值: {YES, NO}
	DefaultAction	默认工作模式。
		取值: {NONE, HDMI, DVI, UNKNOW }
	VideoEnable	当前用户是否使能视频输出。
		取值: {YES, NO}
	AudioEnable	当前用户是否使能音频。
		取值: {YES, NO}
	AviInfoEnable	当前用户是否使能AVI信息帧。
		取值: {YES, NO}
	AudioInfoEnable	当前用户是否使能音频信息帧。
		取值: {YES, NO}
	xvYCCMode	当前用户是否使能xvYCC输出。
		取值: {YES, NO}
	HDCPEnable	当前用户是否使能HDCP。
		取值: {YES, NO}
	DeepColorMode	当前用户设置的图像色彩深度。
		取值: {24, 30, 36, 48, OFF, UNKNOW}
	SpdInfoEnable	当前用户是否使能SPD信息帧。
		取值: {YES, NO}
	OutColorSpace	当前用户设置输出的ColorSpace。
		取值: {RGB, YCbCr422, YCbCr444, YCbCr420, BUTT}
	MpegInfoEnable	当前用户是否使能MPEG信息帧输出。
		取值: {YES, NO}
	DebugEnable	当前用户是否设置开启debug模式。
		取值: {YES, NO}
	CtsAuthEnable	当前用户是否使能认证模式。
		取值: {YES, NO}
	DrmInfoEnable	当前用户是否使能DRM信息帧。
		取值: {YES, NO}

Parameter		Description
SWStatus	ThreadRun	当前HDMI线程是否运行。 取值: {YES, NO}
	RunStatus	HDMI运行状态。 取值: {NONE, OPEN, START, STOP, CLOSE}
	TMDSMode	当前TMDS工作模式。 取值:{NONE, DVI, HDMI1.4, HDMI2.0, AUTO, UNKNOW}
	KernelCnt	Kernel打开HDMI设备的计数。
	UserCnt	用户打开HDMI设备的计数。
	KCallBack	HDMI回调是否被注册。 取值: {YES, NO}
	UCallBackCnt	用户回调次数。
	TransitState	启动过程状态记录。 取值: { BOOT->MCE, MCE->APP, BOOT->APP }
	HDCP2.2Support	当前Sink是否支持HDCP2.2。 取值: {YES, NO}
	HDCPEnable	当前HDCP是否使能。 取值: {YES, NO}
	HDCPIsOn	当前HDCP是否打开。 取值: {YES, NO}
	HDCPMode	当前HDCP模式。 取值:{NONE, HDCP1.4, HDCP2.2, AUTO, UNKNOW}
	KeyLoad	当前是否已经加载key。 取值: {YES, NO}
	HdcpAuthStatus	当前HDCP认证状态。 取值:{YES,NO} 注:YES表示认证成功,NO表示未认证成功
	CECEnable	当前CEC是否使能。 取值: {YES, NO}
	CECStatus	当前CEC状态。 取值: {YES, NO}
	CECLogicAddr	当前CEC设备逻辑地址。

Parameter		Description
	CECNetwork	当前CEC网络。 取值: { TV, RECORDDEV_1, RECORDDEV_2, TUNER_1, PLAYDEV_1, AUDIOSYSTEM, TUNER_2, TUNER_3, PLAYDEV_2, RECORDDEV_3, TUNER_4, PLAYDEV_3, RESERVED_1, RESERVED_2, SPECIALUSE, BROADCAST}
	CECPhyAddr	当前CEC设备物理地址。
HWStatus	HotPlug	当前硬件是否检测到HotPlug。 取值: {YES, NO}
	Rsen	当前硬件是否检测到Rsen。 取值: {YES, NO}
	PhyOutputEnable	当前Phy是否设置输出使能。 取值: {YES, NO}
	PhyPowerEnable	当前Phy Power是否被打开。 取值: {YES, NO}
	SourceScramble	当前Source端是否成功配置Scramble。 取值: {YES, NO}
	SinkScramble	当前Sink端是否Scramble成功。 取值: {YES, NO}
	TMDSBitClkRatio	当前TMDS Clock频率与TMDS Bit Clock频率之比。 取值: {10, 40}
	DataSwing	PHY指标,调整Data幅值,影响眼图大小。
	CLKSwing	PHY指标,调整CLK幅值,影响眼图大小。
	SrcTermination	PHY指标,各通道Data/CLK幅值单独调整。
	TxVnbValue	PHY指标,整体调整Data/CLK,影响单端波形中心点:波峰不变,中心点上下移动
	AdjustClk	PHY指标,微调幅值,影响CLKSwing。
	TxRegZone	PHY指标,HDMI-TX PHY PLL适配。
TaskID	ErrTotal	失败事件总次数
Event Pool Status	HPD	热插拔事件-插入次数
	UnHPD	热插拔事件-拔出次数
	EdidFail	EDID读取失败次数
	HdcpFail	HDCP认证失败次数

Parameter		Description
	HdcpSucc	HDCP认证成功次数
	RsenCon	Rsen连接次数
	RsenDis	Rsen断开次数
	HdcpUsr	用户设置HDCP次数
	WkFlg	唤醒标志
		取值: {HI_TRUE, HI_FALSE}
	RdAble	事件池中可读事件数
	RdPtr	事件池中事件读指针
	WrPtr	事件池中事件写指针
AudioAttr	SoundIntf	HDMI音频接口类型。
		取值: {I2S, SPDIF, HBRA, UNKNOW}
	CodeType	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, UNKNOW}
		注: 详见《EIA-CEA-861-D(F)》。
	ChannelCnt	音频通道数。
		取值: {STR_HEADER, 2_CH, 3_CH, 4_CH, 5_CH, 6_CH, 7_CH, 8_CH}
		注: 详见《EIA-CEA-861-D(F)》。
	SampleFreq	音频采样频率。
		取值: {STR_HEADER, 32KHz, 44.1KHz, 48KHz, 88.2KHz, 96KHz, 176.4KHz, 192KHz} 注: Hi3536目前不支持32000Hz以下的采样率。
	SampleDepth	采样深度(位宽)。 取值: {STR_HEADER, 16bit, 20bit, 24bit}
	DownSample	是否有降采样。 取值: {YES, NO}
AudioIfno	AudioInfoEnable	音频信息帧是否使能。 取值: {YES, NO}

Parameter		Description
	CodeType	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, UNKNOW} 注: 详见《EIA-CEA-861-D(F)》。
	ChannelCnt	音频通道数。 取值: {STR_HEADER, 2_CH, 3_CH, 4_CH, 5_CH, 6_CH, 7_CH, 8_CH} 注: 详见《EIA-CEA-861-D(F)》。
	SampleFreq	音频采样频率。 取值: { STR_HEADER, 32KHz, 44.1KHz, 48KHz, 88.2KHz, 96KHz, 176.4KHz, 192KHz } 注: Hi3536目前不支持32000Hz以下的采样率。
	SampleDepth	采样深度。 取值:{UNKNOW, 8bit, 16bit, 18bit, 20bit, 24bit, 32bit, BUTT}
	SampleSize	采样大小。 取值: {STR_HEADER, 16bit, 20bit, 24bit}
	DownMixInhibit	Down-mix Inhibit 标志位。 取值:{YES, NO} 注:详见《EIA-CEA-861-D(F)》。
	LevelShiftValue	Level Shift Value。 取值: {0~15 dB} 注: 详见《EIA-CEA-861-D(F)》。
	LFEPlayBack	LFE playback level information,LFE播放等级信息。 取值:{UNKNOW,0dB,+10dB,Reserved} 注:详见《EIA-CEA-861-D(F)》。
	Channel/SpeakerAlloc	Channel/Speaker分配。 取值: {0x00~0xff} 注: 以十六进制/十进制显示,详见《EIA- CEA-861-D(F)》。
	AudioInfoRawData	AudioInfoframe原始数据
AudioPath	AudioEnable	音频是否使能。 取值: {YES, NO}

Parameter		Description
	AudioMute	音频Mute(静音)是否使能。 取值: {YES, NO}
	SoundIntf	音频接口类型。 取值: {I2S, SPDIF, HBRA, UNKNOW} 注: Hi3536目前只支持I2S。
	ChannelCnt	当前音频输出通道数。 取值: {STR_HEADER, 2_CH, 3_CH, 4_CH, 5_CH, 6_CH, 7_CH, 8_CH} 注: 详见《EIA-CEA-861-D(F)》。
	SampleFreq	当前音频采样频率。 取值: { STR_HEADER, 32KHz, 44.1KHz, 48KHz, 88.2KHz, 96KHz, 176.4KHz, 192KHz } 注: Hi3536目前不支持32000Hz以下的采样率。
	SampleDepth	当前音频采样深度。 取值: {UNKNOW, 8bit, 16bit, 18bit, 20bit, 24bit, 32bit, BUTT}
	DownSample	当前音频是否降采样。 取值: {YES, NO}
	CTS	CTS值 注:详见《EIA-CEA-861-D(F)》。
	N	N值 注:详见《EIA-CEA-861-D(F)》。
	AudioFIFO	AudioFIFO
VideoAttr	VideoTiming	当前视频时序。 注:详见《EIA-CEA-861-D(F)》、《VESA Display Monitor Timing Standard》。
	DispFmt	当前视频制式 注:详见《EIA-CEA-861-D(F)》、《VESA Display Monitor Timing Standard》。
	PixelClk	像素时钟。 注:详见《EIA-CEA-861-D(F)》、《VESA Display Monitor Timing Standard》。
	InBitDepth	输入视频色深。 取值: {8bit, 10bit, 12bit, 16bit, UNKNOW} 注: UNKNOW为未知。

Parameter		Description
	InColorSpace	输入视频的颜色空间。
		取值: {RGB, YCbCr422, YCbCr444, YCbCr420, UNKNOW}
		注: UNKNOW为未知。
	Colorimetry	输入视频的色域。
		取值: {No Data, SMPTE 170M, ITU-R BT.709, Extended }
	PicAspectRatio	输入视频的宽高比。
		取值:{NONE, 4:3, 16:9, FUTURE, UNKNOW}
		注: UNKNOW为未知。
	ActAspectRatio	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, UNKNOW}
		注: UNKNOW为未知。
	PixelRepeat	像素Repeat次数。
	YCCQuantization	YCC量化范围。 取值: {LIMITED, FULL, UNKNOW}
		文值: {LIMITED, FULL, UNKNOW} 注: UNKNOW为未知。
	RGBQuantization	RGB量化范围。
		取值: {DEFAULT, LIMITED, FULL, UNKNOW}
		注: UNKNOW为未知。
	ExtColorimetry	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, UNKNOW}
	StereoMode	StereoMode 取值: {FRAME_PACK, FIELD_ALTER, LINE_ALTERN, SBS_FULL, L_DEPTH, L_DEPTH_GGD, TAndB, Reserved, SByS_HALF, NONE} 注: NONE为未知。

Parameter		Description
	HvSyncPol	水平/垂直同步极性 取值:{HPVP, HPVN, HNVP, HNVN, UNKNOW} 注:H, horizontal 水平;V, vertical垂直; P, positive 正, N, negative 负, UNKNOW为 未知。
AVIIfno	AVIInfoEnable	AVI信息帧是否使能。 取值: {YES, NO}
	CurrentFormat	当前视频制式/VIC 注:详见《EIA-CEA-861-D(F)》。
	VSIFormat	4k非3D制式/VIC。 注:详见《HDMI Specification 2.0》
	BarDataPresent	Bar信息。 取值: {NONE, HnVp, HpVn, HpVp} 注:: H, horizontal 水平; V, vertical垂直; p, positive 正; n, negative 负; 详见《EIA-CEA-861-D(F)》。
	ColorSpace	视频颜色空间。 取值:{RGB, YCbCr422, YCbCr444, YCbCr420, UNKNOW} 注:UNKNOW为未知。
	Colorimetry	色域。 取值:{ No Data, SMPTE 170M, ITU-R BT.709, Extended}
	PicAspectRatio	视频的宽高比。 取值:{NONE, 4:3, 16:9, FUTURE, UNKNOW} 注:UNKNOW为未知。
	ActAspectRatio	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, UNKNOW} 注: UNKNOW为未知。
	PixelRepeat	像素Repeat次数。
	YCCQuantization	YCC量化范围。 取值: {LIMITED, FULL, UNKNOW} 注: UNKNOW为未知。

Parameter		Description
	RGBQuantization	RGB量化范围。
		取值: {DEFAULT, LIMITED, FULL, UNKNOW}
		注: UNKNOW为未知。
	ExtColorimetry	图像扩展色域。
		取值: {XV_YCC601, XV_YCC709, S_YCC601, ADOBE_YCC601, ADOBE_RGB, BT2020_YCC, BT2020_RGB/YCC, UNKNOW}
		注: UNKNOW为未知。
	ItContentValid	IT Content是否有效。
		取值: {YES, NO}
		注: 详见《EIA-CEA-861-D(F)》。
	HvSyncPol	水平/垂直同步极性
		取值: {HPVP, HPVN, HNVP, HNVN, UNKNOW}
		注: H, horizontal 水平; V, vertical垂直;
		P, positive 正; N, negative 负。
		注: UNKNOW为未知。
	ITContentType	IT Content类型。
		取值: {GRAPHICS, PHOTO, CINEMA, GAME}
	PicScaling	图像均衡。
		取值: { UNKNOW, HpVn, HnVp, HpVp}
		注: H, horizontal 水平; V, vertical垂直; p, positive 正; n, negative 负; 详见《EIA-CEA-861-D(F)》。
	ActFmtPresent	有效信息是否显示。
		取值: {YES, NO}
		注: 详见《EIA-CEA-861-D(F)》。
	ScanInfo	Scan标志位
		取值: {NONE, OVER_SCAN, UNDERS_SCAN, Reserved}
	AVIInfoRawData	AVI 信息帧原始数据。
	VSInfoRawData	VSIF信息帧原始数据。
VedioPath	VideoMute	Video Mute标志 取值: {YES, NO}

Parameter		Description
	OutBitDepth	输出视频色深。 取值: {8bit, 10bit, 12bit, 16bit, UNKNOW}
	OutColorSpace	输出图像颜色空间 取值: {RGB, YCbCr422, YCbCr444, YCbCr420, UNKNOW}
	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。
	Dither	Dither是否打开。 取值: {YES, NO}
HDRAttr	HdrMode	当前HDR模式。 取值:{DISABLE, DOLBY_NORMAL, DOLBY_TUNNELING, HDR, UNKNOW}
	HdrEotfType	HDR模式下EOTF类型。 取值: { SDR_LUMIN, HDR_LUMIN, SMPTE_ST_2048, FUTURE, UNKNOW}
	HdrMetaDataID	HDR模式下MetaData类型。 取值: 0 - Static Metadata Type 1 其他暂不支持
	HdrColorimetry	HDR模式下Colorimetry。 取值: {NONE, ITU_601, ITU_709, EXTENDED, XV_YCC_601, XV_YCC_709, S_YCC_601, ADOBE_YCC_601, ADOBE_RGB, 2020_nconst_luminous, 2020_const_luminous}

Parameter		Description
	DispPrim0_X	display_primaries_x[0]取值。 参考《CEA-861.3》
	DispPrim0_Y	display_primaries_y[0]取值。 参考《CEA-861.3》
	DispPrim1_X	display_primaries_x[1]取值。 参考《CEA-861.3》
	DispPrim1_Y	display_primaries_y[1]取值。 参考《CEA-861.3》
	DispPrim2_X	display_primaries_x[2]取值。 参考《CEA-861.3》
	DispPrim2_Y	display_primaries_y[2]取值。 参考《CEA-861.3》
	WhitePoint_X	white_point_x取值。 参考《CEA-861.3》
	WhitePoint_Y	white_point_y取值。 参考《CEA-861.3》
	MaxLuminance	max_display_mastering_luminance取值。 参考《CEA-861.3》
	MinLuminance	min_display_mastering_luminance取值 参考《CEA-861.3》
	MaxLightLevel	Maximum Content Light Level。 参考《CEA-861.3》
	AverLightLevel	Maximum Frame-average Light Level。 参考《CEA-861.3》
DRMInfo	DRMInfoEnable	当前DRM信息帧是否使能。 取值: {YES, NO}
	EotfType	当前EOTF类型。 取值:{SDR_LUMIN, HDR_LUMIN, SMPTE_ST_2048, FUTURE, UNKNOW}
	MetadataID	当前MetaData类型。
	DRMInfoRawData	DRM信息帧原始数据。
EDIDRaw Data		EDID原始数据,256Bytes

Parameter		Description
SWStatus	CapFromSink	能力集是否来自显示器(SINK)。 取值:{YES, NO} 注:能力集可能来自用于测试的数据,此时指示NO。
	RawUpdateErrCnt	原始数据更新错误次数。
	CapIsValid	能力集是否有效。 取值: {YES, NO}
	ParseErrorType	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。
	RawIsValid	原始数据是否有效。 取值: {YES, NO}
	ParseWarnType	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。
	RawGetErrCnt	原始数据获取失败次数。

Parameter		Description
	RawLength	原始数据长度。
BasicCap	HDMI1.4Support	Sink端是否支持HDMI1.4。 取值: {YES, NO}
	1stBlockVersion	EDID第一块版本号。
	HDMI2.0Support	是否支持HDMI2.0 取值: {YES, NO}
	ManufacturerName	Sink端厂商名。
	MaxTMDSClock(MHz)	最大TMDS时钟。 注:详见《EIA-CEA-861-D(F)》。
	ProductCode	Sink端产品号。
	HDCP1.4Support	Sink端是否只是HDCP1.4。 取值: {YES, NO}
	SerialNumber	Sink产品序列号。
	HDCP2.2Support	Sink端是否支持HDCP2.2。 取值: {YES, NO}
	WeekOfManufacture	Sink端产品制造周。 注:详见《EIA-CEA-861-D(F)》。
	SCDCSupport	Sink端是否支持SCDC。 取值: {YES, NO}
	YearOfManufacture	Sink端产品制造年份。
	DVIDualSupport	Sink端是否支持DVI。
	CECAddrIsValid	Sink端CEC地址是否有效。 取值: {YES, NO}
	AlSupport	Sink端是否支持AI。 取值: {YES, NO}
	CECAddr	CEC地址。
	ExtBlockCnt	EDID扩展块数量。
	SpeakerSupport	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, UNKNOW} 注: 详见《EIA-CEA-861-D(F)》。

Parameter		Description
VidoCap	NativeFormat	支持无需任何变换的制式(最佳制式)。 注:详见《EIA-CEA-861-D(F)》。
	ColorSpace	支持的颜色空间。 取值: { RGB444,YCbCr444,YCbCr422,YCbCr420} 注:详见《EIA-CEA-861-D(F)》。
	DeepColor	支持的深色位数。 取值: { RGB_30Bit, RGB_36Bit, RGB_48Bit, YCbCr444_SameRGB} 注: 详见《EIA-CEA-861-D(F)》。
	YCbCr420DeepColor	支持YCbCr420的深色位数。 取值: { 30Bit, 36Bit, 48Bit} 注: 详见《EIA-CEA-861-D(F)》。
	YCbCr420[Also]	除RGB、YCbCr444、YCbCr422外,同时支持 YCbCr420的VIC码。 注:详见《EIA-CEA-861-D(F)》。
	YCbCr420[Olny]	只支持YCbCr420 的VIC码。 注:详见《EIA-CEA-861-D(F)》。
	Colorimetry	Colorimetry色域。 取值:{xvYCC601, xvYCC709, sYCC601, AdobleYCC601, AdobleRGB, BT2020cYCC, BT2020YCC, BT2020RGB} 注:详见《EIA-CEA-861-D(F)》。
FormatCa p		Sink端支持哪些制式。 注:详见《EIA-CEA-861-D(F)》、《VESA Display Monitor Timing Standard》
3DCap	3DSupport	Sink端是否支持3D。 取值: {YES, NO}
	3DDualView	Sink端是否支持3D Dual View。 取值: {YES, NO}
	3DTypeSupport	Sink端支持的3D显示类型。 取值:{FRAME_PACK, FIELD_ALTER, LINE_ALTERN, SBS_FULL, L_DEPTH, L_DEPTH_GGD, TAndB, Reserved, SByS_HALF, NONE}
	3DOsdDisparity	Sink端是否支持3D OSD Disparity。 取值: {YES, NO}

Parameter		Description
	3DIndepView	Sink是否支持3dIndependentView.
		取值: {YES, NO}
AudioCap	NO.x:	编号x
	CodeType	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, UNKNOW}
		注:详见《EIA-CEA-861-D(F)》。
	MaxChannelNum	最大通道数。
	MaxBitRate(KHz)	最大bit速率。
	BitDepth	bit深度。
		取值: {0, 8, 16, 18, 20, 24, 32, BUTT}
	SampleRate(Hz)	采样速率。 取值: {0, 8kHz, 11025Hz, 12000Hz, 16000Hz, 22050Hz, 24000Hz, 32000Hz, 44100Hz, 48000Hz, 88200Hz, 96000Hz, 176400Hz, 192000Hz, 768000 Hz, BUTT}
DetailTimi	NO.	编号
ng	НАСТ	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
	ImgW	图像宽
	ImgH	图像高
	IHS	水平同步脉冲是否翻转

Parameter		Description
	IVS	垂直同步脉冲是否翻转
	IDV	有效信号是否翻转

2.14 TUNER

2.14.1 Tuner

【调试信息】

【调试信息分析】

记录所有的tuner状态,如是否锁定,锁定的频点等信息。Port的编号即tuner的编号。

参数	描述
Port	tuner的编号。
LockStat	是否锁定标记: Locked/Unlocked。
I2CChannel	对应Tuner使用的I2C通道号
Frequency(KHz)	频率,单位KHz。
SymbRate	符号率, 单位Symb/s。
QamMode	调制方式。QAM_16/QAM_32/QAM_64/ QAM_128/QAM_256/BPSK/QPSK/8PSK/ 16APSK/32APSK/QAM_INVALID (非法)
DemodType	Demod的类型 3130_inside/3130_outside/hi3136i/hi3136e/ avl6211/mxl101等
Demodaddr	Demod的I2C地址
TunerType	Tuner类型。

参数	描述
Tuneraddr	Tuner的I2C地址
BER	误码率。x*(E-7)表示x*10的负7次方 (x是一个变量).
SNR	信噪比(Signal Noises Rate)。
SignalStrength	信号强度,proc信息下所见可能并非真实信号强度值,很可能只是agc的增益值,真实信号强度需通过运算得到,运算过程和具体器件相关。
all_rs_package	统计在阈值设定的时间内输出的TS帧数
corrected_rs_package	统计在阈值设定的时间内由RS纠正的TS 帧数
error_rs_package	统计在阈值设定的时间内RS不能纠正的 TS帧数,关注此项,此项非0时,代表有 误包,需要排查环境和是否为信道问题

Tuner锁定是DVB模式下音视频播放和数据收取的首要条件,在音视频不能播放和 Demux没有数据的情况下,首先看Tuenr是否锁定。

2.14.2 Tuner_Reg

【调试信息】

【调试信息分析】

记录所有qam寄存器的值,仅Hi3130系列芯片和内置qam支持。可以通过寄存器的名称在用户指南找到对应的寄存器说明。比如上面显示的AGC_CTRL_1_ADDR,去掉后面的_ADDR后缀,把字母改为小写,即为其寄存器名称为agc_ctrl_1,在用户指南上搜agc_ctrl_1,找到其对应的寄存器说明即可。

2.15 IR

【调试信息】

```
$ cat /proc/msp/ir
-----Hisilicon IR Info-----
IREnable:Enable
IRCode:IR_S2
IRWorkMode:Chip Report Symbols
IRFetchMode:Drive Report Parsed Symbols
IRKeyUpEnable:Enable
```

IRUpEventDelay:300(ms) IRRepeatkeyEnable:Enable IRRepkeyDelayTime:200(ms) IRReportKeyBlockTime:200(ms) IRModuleFrequence:24(MHz) Registered Protocols info: No.0:Status:Enabled, Name:nec simple 2headers gd No.1:Status:Enabled, Name:extended rc5 14bit No.2:Status:Enabled, Name:rc5 14bit data No.3:Status:Enabled, Name:nec simple 2headers changshu No.4:Status:Enabled, Name:nec full 2headers No.5:Status:Disabled, Name:tc9012 Key getting info: Get(Try/OK):20/2 Key buffer info: Buffer size:100 keys Reader at:2 Writer at:2 Symbol buffer info: Buffer size:100 symbols Reader at:36 Writer at:36

【调试信息分析】

记录遥控模块相关状态。在IR Code取值不同的时候,某些属性可能生效或者无效,所以显示的调试信息和上面的举例可能有一些不同。

参数	描述	
Enable 使能标识:		
	Enable: 使能;	
	Disable: 未使能。	
Code	红外驱动代码标识:	
	IR_STD:标准红外(由逻辑解析,每次只支持一种协议);	
	IR_S2: 普通红外;	
	IR_LIRC: 万能红外。	
WorkMode	红外工作模式:	
	Chip Report Symbols: 驱动解析键值;	
	Chip Report Keys: 芯片解析键值。	
FetchMode	获取遥控码的类型:	
	Drive Report Raw Symbols: 获取原始电平;	
	Drive Report Parsed Symbols: 获取解析过的电平(即键值)。	
KeyUpEnable	e 是否设置了上报按键弹起信息:	
	Enable: 上报;	
	Disable: 不上报。	
UpEventDelay	按键弹起上报延时,单位ms。	

参数	描述
RepeatkeyEnable	是否设置了上报重复按键:
	Enable: 上报;
	Disable: 不上报。
RepkeyDelayTime	重复按键判断门限,单位ms。
ReportKeyBlockTi me	获取键值的最大等待时间,单位ms。
ModuleFrequence	IR模块参考时钟,单位MHz。
Registered Protocols info	当前代码支持的协议名称和使能状态。
Key getting info	Try: 尝试获取键值的次数;
	OK: 获取成功键值的次数。
Key buffer info	按键缓冲区的大小和读写指针的位置。
Symbol buffer info	电平缓冲区的大小和读写指针的位置。

如果遥控器收不到键值,可以按如下步骤排查:

步骤1 检查IR模块是否使能, IR Enable属性是否为1。

步骤2 检查Symbol buffer info内是否有数据,如果有数据,则跳到步骤5检查;如果没有数据,则开始步骤3检查。

步骤3 检查IR寄存器,看IR中断屏蔽寄存器设置是否正确。

步骤4 检查红外接收头硬件连接是否正确。

步骤5 检查使用的遥控器的协议类型是否支持,是否使能,在Registered Protocols info信息中可以看到(如果驱动代码使用的是IR_STD的类型,那么需要检查是否调用HI UNF IR SetCodeType函数设置了对应遥控器的类型)。

步骤6 检查ModuleFrequence的值是否和当前IR模块的参考时钟相符。

----结束

2.16 I2C

【调试信息】

\$ ca	\$ cat /proc/msp/i2c			
	Hisilicon Standard I2C Info			
No.	Rate			
0	40000			
1	40000			
2	40000			
3	40000			
4	40000			

5 400000 ------Hisilicon GPIO simulate I2C Info-----No. SCL IO SDA IO 6 10 11

【调试信息分析】

查看标准I2C的通信速率信息以及用来模拟I2C的GPIO端口信息。

【参数说明】

参数	描述
No.	标准I2C或者GPIO模拟I2C的ID
Rate	标准I2C的通信速率(GPIO模拟的I2C不支持设置速率)
SCL_IO	用来模拟I2C SCL的GPIO口
SDA_IO	用来模拟I2C SDA的GPIO口

【调试命令】

通过I2C通道读写数据

\$ echo id address offset [value] >/proc/msp/i2c

比如:

\$echo 4 a0 5d 80 >/proc/msp/i2c

表示用i2c通道4, 向i2c设备地址a0, 设备偏移地址5d, 写入数据0x80。

\$ echo 4 a0 5d >/proc/msp/i2c 0x80

表示用i2c通道4,向i2c设备地址a0,设备偏移地址5d,读取数据。

【调试命令说明】

该proc命令只适合调试寄存器地址为8位的slave器件,并且读/写数据的长度为1 BYTE,如需调试超过8位寄存器地址的slave器件,或者读/写数据长度超过1 BYTE,请使用i2c sample的i2c read和i2c write功能。

参数	描述
id	设备使用的I2C通道
address	设备在I2C总线上的地址
offset	设备的片内偏移地址
value	写入的数值,这个参数是可选的,不输 入这个参数的话,就表示进行读操作, 输入这个参数,就表示进行写操作

Proc系统会自动根据命令的长度判断是读操作(read)还是写操作(write),没有value参数(即第四个参数)时为read,反之为write,注意当write时,第四个参数后需要有空格。

- 当使用的I2C总线(即第一个参数)I2C不通时,会出现打印: wait write data timeout!。请检查i2c通道是否正确,地址是否正确,i2c硬件连接是否正常。
- 当使用的I2C总线为gpio i2c通道时,如果用没有申请资源的gpio i2c通道操作,例如: echo 8 a0 5d >/proc/msp/i2c, 会提示: gpio_i2c_read failed(I2cNum=8 not valid)!
- 当设备的片内偏移地址(即第三个参数)为0~f(十六进制)时,一定要如此表示: 00~0f,即一定要在前面加0补齐。

注意Address和offset的值前面都没有"0x"字样。

修改标准I2C的通信速率:

\$ echo SetRate id rate > /proc/msp/i2c

比如:

\$ echo SetRate 1 100000 > /proc/msp/i2c

2.17 SCI

【调试信息】

cat /proc/msp/sci0

-----Hisilicon SCI0 Info-----

Sci State :RX

SetFrequency:3570

Protocol:T=0

VccEnLevel:1

DetectLevel:1

ClockMode:OD

ResetMode: CMOS

VccMode:OD

ATR :0x3b 0x9f 0x11 0x40 0x60 0x49 0x52 0x44 0x45 0x54 0x4f 0x20 0x41 0x43 0x53 0x20 0x56 0x35 0x2e 0x33

TS:0x3b

T0:0x9f

TA1 : 0x11 (FI = 372, DI = 1)

TD1:0x40

TC2:0x60 (only for T0 show IC card max char timeout)

HistoryByte :0x49 0x52 0x44 0x45 0x54 0x4f 0x20 0x41 0x43 0x53 0x20 0x56 0x35 0x2e 0x33

ActualSciClk:3600

ExpectBaudRate:9596

CalcBaudFlag:0

b SetExtBaudFlag: 0

ClkRate(F):372

BitRate(D):1

BaudRate:9677

AddCharGuard:0 etu

BlockGuard:22 etu

Value:5

Baud: 2231

CharTimeout:92160

【调试信息分析】

记录SCI运行状态的相关信息。

参数	描述
Sci State	SCI卡当前状态,包括NOCARD、 INACTIVECARD、RX等等。
SetFrequency	要设置的SCI卡频率。对于T0,T1卡,支持频率1MHz~5MHz;对于T14卡,只支持6MHz。单位为kHz。
Protocol	卡类型可选择协议类型T=0、T=1、 T=14。
VccEnLevel	供电电压使能信号高有效还是低有效。 0: 低电平有效; 1: 高电平有效。
DetectLevel	检测卡电平高有效还是低有效。 0: 低电平有效; 1: 高电平有效。
ClockMode	时钟管脚输出模式,CMOS或者OD。 0: CMOS; 1: OD。

参数	描述
ResetMode	Reset管脚输出模式,CMOS或者OD。
	0: CMOS;
	1: OD.
VccMode	Vcc管脚输出模式,CMOS或者OD。
	0: CMOS;
	1: OD.
ATR	复位应答,由一系列字节组成,这些字 节规定了卡和终端之间即将建立的通信 特性。
TS	起始字符是终端接收的第一个字符,用 于约定后续字符的逻辑方向。
	0x3B: 正向约定;
	0x3F: 反向约定。
Т0	T0格式字符。
	高半字节(b5~b8): 后续字符TA1到
	TD1是否存在;
	低半字节(b4~b1): 可选历史字符数 目0~15。
TA1	TA1用于传送FI和DI的值。
	FI (b5~b8) 用于确定F (时钟率转换因子)的值;
	DI(b4~b1)用于确定D(比特速率调节 因子)的值。
TD1	高半字节(b5~b8)表示后续字符TA2到 TD2是否存在,为1表示存在;
	低半字节(b4~b1)后续信息交换所使
	用的协议类型; 0: T0, 1: T1, e: T14。
TC2	表示WI值,用来确定IC卡发送任意一个 字符起始位上升沿与IC卡或者终端发送
	的前一个字符起始位上升沿的最大时间 间隔。
HistoryByte	历史字节。
ActualSciClk	实际SCI时钟通过计算配置寄存器后实际 芯片输出给卡的时钟,ActualSciClk允许 与设置时钟有偏差,但实际值需要在卡 允许的时钟范围内。
ExpectBaudRate	期望的波特率。
	ExpectBaudRate=SetFrequency*BitRate*10 00/ClkRate

参数	描述
CalcBaudFlag	ATR收到TA2并且Bit5=0 为特定模式,该标志位为1,用收到的TA1参数设置etu,没有收到TA2 默认为交互模式,该标志位为0。
bSetExtBaudFlag	为1表示应用调用 HI_UNF_SCI_SetEtuFactor()函数通过 外部设置ETU的F、D因子标志位,0表示 不是外部设置。
ClkRate(F)	实际设置etu的时钟转率因子。
BitRate(D)	实际设置etu的速率调节因子。
BaudRate	实际计算的波特率。 BaudRate = ActualSciClk * BitRate *1000/
	ClkRate
AddCharGuard	额外增加的字符保护时间,单位: etu。
BlockGuard	T1 块保护时间,单位: etu。
Value	用于配置etu 波特率的周期数。
Baud	用于配置etu 波特率的时钟分频值。
CharTimeout	字符超时时间,单位: etu。

2.18 LOG

【调试信息】

```
# cat /proc/msp/log
----- Log Path -----
log path:
----- Store Path -----
store path:/mnt
----- Module Log Level -----
Log moduleLevel
HI SYS1 (ERROR)
HI_MODULE1 (ERROR)
HI_LOG1 (ERROR)
HI PROC1 (ERROR)
HI MEM1 (ERROR)
HI STAT1 (ERROR)
HI PDM1 (ERROR)
HI MEMDEV1 (ERROR)
HI_DEMUX1 (ERROR)
HI_ADEC1(ERROR)
HI AO1 (ERROR)
HI AI1 (ERROR)
HI ADSP1 (ERROR)
```

```
HI VFMW1 (ERROR)
HI SVDEC1 (ERROR)
HI DISP1 (ERROR)
HI HDMI1 (ERROR)
HI VO1 (ERROR)
HI VPSS1 (ERROR)
HI VDEC1 (ERROR)
HI VI1 (ERROR)
HI VENC1 (ERROR)
HI TDE1 (ERROR)
jpeg1 (ERROR)
HI JPGE1 (ERROR)
HI FB1 (ERROR)
HI PNG1 (ERROR)
HI PVR1 (ERROR)
HI AVPLAY1 (ERROR)
HI SYNC1 (ERROR)
VSYNC1 (ERROR)
ASYNC1 (ERROR)
HI MCE1 (ERROR)
HI IR1 (ERROR)
HI I2C1 (ERROR)
HI SCI1 (ERROR)
HI WDG1 (ERROR)
HI GPIO1 (ERROR)
HI GPIO I2C1 (ERROR)
hi tuner1 (ERROR)
HI KEYLED1 (ERROR)
HI CIPHER1 (ERROR)
HI OTP1 (ERROR)
HI PM1 (ERROR)
HI SIL92931 (ERROR)
HI NDPT1 (ERROR)
echo hi avplay=2 > /proc/msp/log
echo log=/mnt > /proc/msp/log
echo storepath=/mnt > /proc/msp/log
# echo help>/proc/msp/log
To modify the level, use command line in shell:
echo module name = level number > /proc/msp/log
level number: 0-fatal, 1-error, 2-warning, 3-info
example: 'echo HI_DEMUX=3 > /proc/msp/log'
will change log levle of module "HI DEMUX" to 3, then,
all message with level higher than "info" will be printed.
Use 'echo "all = x" > /proc/msp/log' to change all modules.
To modify the log path, use command line in shell:
Use 'echo "log = x" > /proc/msp/log' to set log path.
Use 'echo "log = /dev/null" > /proc/msp/log' to close log udisk output.
example: 'echo log=/home > /proc/msp/log'
To modify the debug file store path, use command line in shell:
Use 'echo "storepath = x" > /proc/msp/log' to set debug file path.
example: 'echo storepath=/tmp > /proc/msp/log'
```

【调试信息分析】

记录当前各个模块的调试级别。

参数	描述
Log Module	模块名。

参数	描述
Level	模块的打印级别。
	0: fatal;
	1: error;
	2: warning;
	3: info;
	4: debug.

- 默认所有模块输出ERROR级别的打印信息。
- 修改指定模块打印级别的方法: #echo 模块名=打印级别 > /proc/msp/log

模块名不区分大小写。

- 支持动态设置调试文件(如Demux录制文件)存储路径,命令如下:
- echo storepath=存储路径 > /proc/msp/log支持动态修改日志的输出位置到U盘或者串口,可以通过echo log = 日志输出位置> /proc/msp/log,比如echo log = /home> / proc/msp/log,将日志文件输出到某个目录下,无需输入文件名字,日志的文件名字固定为stb.log,以追加的方式进行日志转储。
- 如果关闭日志的文件方式输出,需要echo log = /dev/null> /proc/msp/log,则日志重新恢复到串口输出。

注意:如果需要修改日志的输出位置,需要软件版本支持,在编译选项中选中"UDisk Log Support",通过menuconfig依次进入如下所列菜单配置:

make menuconfig
Common--->
[*] Log Support
[*]UDisk Log Support

● 如果要配置日志的编译级别,可以配置"Compile Log Level"选项的值,通过 menuconfig依次进入如下所列菜单配置:

make menuconfig
Common--->
[*] Log Support
(1) Compile Log Level (0,4)

修改后需要重新编译SDK。

● 如果要支持网络日志,需在编译选项中选中"Network Log Support",通过 menuconfig依次进入如下所列菜单配置:

make menuconfig
Common--->
[*] Log Support
[*]Network Log Support

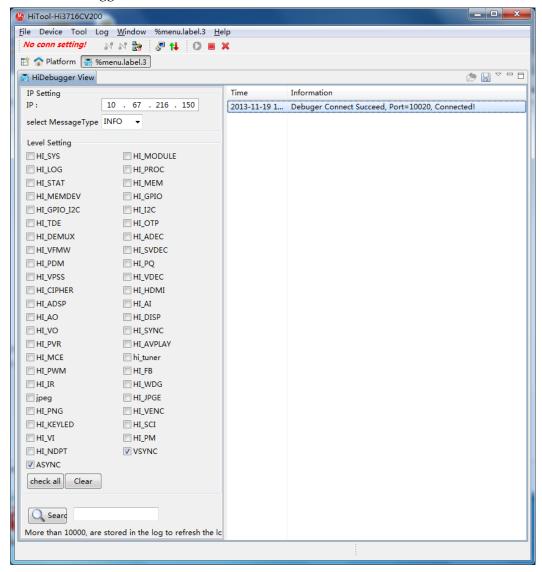
同时需要选中的"Msp Debug Tool Support"选项,通过menuconfig依次进入如下 所列菜单配置:

make menuconfig
Rootfs--->
Board Tools Config--->
[*] Msp Debug Tool Support

重编SDK后文件系统中将有msp_debug工具,在单板上运行msp_debug&(后台运行),再运行需要的用例或某个sample。

然后在PC端运行HiTool工具,选择HiDebugger界面,如图2-5所示。

图 2-5 HiDebugger 界面



配置好IP,点击ToolBar上的 梦 按钮与单板相连,左侧选框中选择调试级别与需要调试信息的模块,即可开始抓取调试信息。

2.19 Interrupts

【调试信息】

```
$ cat /proc/interrupts
CPU0CPU1
57:570GICtimer
58:15913150GICtimer4
59:051281991GICtimer6
66:00GIChi mci
67:00GIChi mci
```

```
79:00GIChi ir
81:11570GICuart-pl011
98:00GICehci hcd:usb1
99:10GICohci hcd:usb2
102:00GICahci
103:00GICgmac0
104:00GICgmac1
107:00GICCI IRQ
114:00GICDemux
122:255069010GICDISP IRQ
123:00GICtde osr isr
125:00GICVPSS IRQ
126:00GICmali_mmu_irq_handlers, mali_gp_irq_handlers,
mali_mmu_irq_handlers, mali_pp_irq_handlers, mali_mmu_irq_handlers,
mali_pp_irq_handlers
128:00GICpng_isr
129:00GICx5 jpeg
134:00GICjpge
140:00GICgpio00
141:00GICgpio01
142:00GICgpio02
143:00GICgpio03
144:00GICgpio04
145:00GICgpio05
146:00GICgpio06
147:00GICgpio07
148:00GICgpio08
149:00GICgpio09
150:00GICgpio10
151:00GICgpio11
152:00GICgpio12
153:00GICgpio13
154:00GICgpio14
155:00GICgpio15
156:00GICgpio16
157:00GIC gpio17
IPIO:00Timer broadcast interrupts
IPI1:93121205Rescheduling interrupts
IPI2:00Function call interrupts
IPI3:1128Single function call interrupts
IPI4:00CPU stop interrupts
IPI5:00CPU backtrace
Err:0
```

● 通过中断响应次数有没有增加可以反映模块是否在运行。

以demux为例可以知道是否有TS流从TSI接口进入demux。

2.20 Media-mem

【调试信息】

cat /proc/media-mem

∭ 说明

下图proc信息中smmu相关的信息仅适用于HI3798CV200,其他芯片不适用

图 2-6 media-mem 调试信息

```
cat /proc/media-mem
                          SMMU
                                                         IOMMU
                                                                     | KVIRT | FLAGS |
                                                                                                      LENGTH (KB)
|ZONE[0]: (0x00100000, 0xffefffff) 2 1 0x00000000 4192256 "iomm
|phys(0xffffffff, 0x0002Afff), smmu=(0x00100000, 0x0012Afff),kvirt=f0ac3000, length=172KB,
|phys(0x7D260000, 0x7D26Afff), smmu=(0x0012C000, 0x00136fff),kvirt=f0f04000, length=44KB,
                                                                                                                      "iommu
                                                                                                                                 name="PQ_FLASH_BIN"|
name="CIPHER_ChnBuf"|
Summary:
   SMMU Total Size |
                                                               Idle
                                                                                                        BLock Number
          4094MB
                                    0MB
                                                               4093MB
                                                       | IOMMU | KVIRT | FLAGS | LENGTH(KB)
                                                                                                                       "ddr"
ZONE[1]: (0x7d000000, 0x7f7fffff)
                                                                                                     40960
                                                                              0x000000000
phys(0x7D000000, 0x7D003FFF), smmu=(0xFFFFFFFF, 0x00003FFF),kvirt=f05ab000, length=16KB,
                                                                                                                                 name="CMN_LogInfo"|
                                                                                                                                 name="CMN_LogTrace"|
name="CMN_LogTrace"|
name="TDE_MemPool"|
name="PQ_IPSEL_ALG"|
name="PQ_VpssZmeCoef"|
phys(0x7D004000, 0x7D103FFF), smmu=(0xFFFFFFFF,
                                                                   0x000FFFFF), kvirt=f06b3000, length=1024KB,
phys(0x7D104000, 0x7D123FFF), smmu=(0xFFFFFFFF, 0x0001FFFFF), kvirt=f080a000, length=128KB, phys(0x7D124000, 0x7D133FFF), smmu=(0xFFFFFFFF, 0x0000FFFF), kvirt=f0b07000, length=64KB,
                                                                                                          length=128KB,
phys(0x7D134000, 0x7D136FFF), smmu=(0xFFFFFFFF,
                                                                   0x00002FFF), kvirt=f0b1e000,
                                                                                                           length=12KB,
phys(0x7D137000, 0x7D138FFF), smmu=(0xFFFFFFFF, 0x00001FFF),kvirt=f0b27000,
phys(0x7D139000, 0x7D13BFFF), smmu=(0xFFFFFFFF, 0x00002FFF),kvirt=f0b30000,
                                                                                                                                name="PQ_VdpZmeCoef"|
name="PQ_GfxZmeCoef"|
name="VPSS_RegBuf"|
                                                                                                           length=8KB,
                                                                                                          length=12KB,
phys(0x7D13C000, 0x7D25FFFF), smmu=(0xFFFFFFFF, 0x00123FFF),kvirt=f0ca8000,
                                                                                                          length=1168KB,
phys(0x7D260000, 0x7D26AFFF), smmu=(0x0012C000, 0x00136FFF), kvirt=f0f04000,
                                                                                                          length=44KB,
                                                                                                                                  name="CIPHER_ChnBuf"|
Summary:
   MMZ Total Size |
                                  CMA Used
                                                              Idle
                                                                           | Zone Number |
                                                                                                       BLock Number
          40MB
                                  2MB
                                                             37MB
```

【调试信息分析】

Media-mem描述了业务对MMZ内存/SMMU内存的使用信息。

参数	描述
SMMU	表示当前的内存zone是SMMU zone,物理内存不一定连续
ID	表示当前内存zone里面的内存块的ID
IOMMU	表示当前内存zone的类型,1表示smmu zone,0表示ddr zone
KVIRT	内存块的起始虚拟地址
FLAGS	内存块的属性标志,保留
LENGTH(KB)	内存块的长度,单位是KB
NAME	zone的名字
ZONE[0]	smmu zone的smmu起始地址和结束地址
phys	内存块的物理地址区间,当起始地址为全F时,地址区间无效,无参考意义
smmu	内存块的smmu虚拟地址区间,当起始地址为全F时,地址 区间无效,无参考意义
kvirt	内存块的内核态虚拟地址的起始地址

参数	描述
length	内存块的长度
name	内存块的名字
SMMU Total Size	smmu zone的总大小
Iommu Used	smmu zone使用的内存,以MB为单位
Idle	空闲内存的总大小
Zone Number	当前zone的序数
Block Number	当前zone里面分配的内存块数
ZONE[1]	DDR zone的物理起始地址和结束地址
DDR	表示当前内存zone是DDR zone,物理内存是连续的
MMZ Total Size	MMZ zone内存块的总大小
CMA Used	MMZ zone中已经使用的内存信息

● 通过手动输入cat/proc/media-mem,查看MMZ内存使用信息。

当MMZ内存不足的时候,申请MMZ内存失败会打印全部MMZ内存的使用信息。

2.21 **SATA**

【调试信息】

```
$ cat /proc/msp/stat
======== host isr stat =======
isr stat is disabled!
======= host thread stat =======
====== host event stat =======
KEYIN= 0(keyvalue 0x0)
KEYOUT= 0(keyvalue 0x0)
ASTOP= 76639
VSTOP= 76639
CONNECT= 39806859
LOCKED= 39807068
ASTART= 39807641
VSTART= 39807639
CWSET= 0
STREAMIN= 39807655
ISTREAMGET= 39807805 (size 5)
FRAMEDECED= 39807819
VPSSGET= 39807819
VPSSOUT= 39807833
AVPLAYGET= 39807837
PRESYNC= 0
BUFREADY= 0(type 0)
FRAMESYNCOK= 39807837
VOGET= 39807837
```

IFRAMEINTER= 480 TOTAL= 39807837

【调试信息分析】

统计换台的时间, 记录换台过程中各个事件的时间点。

参数	描述
KeyIn	按键解析完的时间。
KeyOut	按键被取走的时间。
AudStop	停止音频播放的时间。
VidStop	停止视频播放的时间。
Connect	TUNER锁频开始的时间。
Locked	TUNER锁频成功的时间。
AudStart	开始音频播放的时间。
VidStart	开始视频播放的时间。
CWSET	设置密钥区CW字的时间。
STREAMIN	解码器收到码流的时间。
ISTREAMGET	解码器收到第一个I帧的时间。
FRAMEDECED	解码器解出第一个I帧的时间。
VPSSGET	VPSS收到第一帧的时间。
VPSSOUT	VPSS输出第一帧的时间。
AVPLAYGET	AVPLAY收到第一帧的时间。
PRESYNC	预同步完成的时间。
BUFREADY	帧缓冲结束时间。
FRAMESYNCOK	第一帧同步完成时间。
VOGET	VO收到第一帧的时间。
IFRAMEINTER	码流中I帧间隔的时间。
TOTAL	总的换台时间。



注意

在统计换台时间时,按下换台键松开后,不要再按其他键,直接查看stat,否则会因为 KeyIn和KeyOut的时间错误,导致统计换台时间出错。

2.22 开机画面及瞬播

【调试信息】

cat /proc/msp/mce

-----PlayParam------

enPlayType:DVB,bPlayEnable:1
u32VideoPid:257,u32AudioPid:258
enVideoType:MPEG2,enAudioType:MP2

u32Volume:50,enTrackMode:1

enSigType:1

u32Freq:3750000,u32SymbolRate:10490000

enPolar:0

-----PlayStatus-----

hAvplay:0,hWindow:2359552 BeginTime:1187,EndTime:0

【调试信息分析】

显示瞬播参数信息,以及瞬播当前状态。

参数	描述
enPlayType	瞬播类型:
	0: DVB瞬播;
	1: TS流瞬播;
	2: 动画瞬播。
bPlayEnable	瞬播是否使能:
	0: 不使能;
	1: 使能。
u32VideoPid	视频PID。
u32AudioPid	音频PID。
enVideoType	视频协议类型:
	0: MPEG2;
	1: MPEG4;
	2: H264。
enAudioType	音频协议类型:
	0: MP2;
	1: MP3。

参数	描述
u32Volume	音量。
enTrackMode	音频声道模式,与HI_UNF_TRACK_MODE_E枚举对应。
enSigType	DVB信号类型,与HI_UNF_TUNER_SIG_TYPE_E枚举对应。
u32Freq	DVB频率。
u32SymbolRate	DVB符号率。
enPolar	极化方式。
hAvplay	瞬播AVPLAY句柄。
hWindow	瞬播WINDOW句柄。
BeginTime	瞬播启动时间。
EndTime	瞬播停止时间。

【调试信息】

echo stop > /proc/msp/mce

【调试信息分析】

停止瞬播。

2.23 AENC

【调试信息】

cat /proc/msp/aenc00

----- AENC[00] State -----

WorkStatus:start Codec ID:0x80020001 Description:aac Sample Rate:48000

Channels:2 BitWidth:16

AttachSource:Track08 TryEncodeTimes:6599

EncodeFrameNum(Total/Error):3768/0

FrameBuf(Total/Use/Percent)(Bytes):524288/672/0%

StreamBuf(Total/Use/Percent):32/0/0%

SendFrame(Try/OK):3462/3462
ReceiveStream(Try/OK):11471/3768
ReleaseStream(Try/OK):3768/3768

【调试信息分析】

显示AENC的状态。

参数	描述
WorkStatus	AENC工作状态。start/stop分别标识工作与非工作状态
Codec ID	音频编码类型。括号内是HA_AUDIO_ID的值
Description	编码器描述
Sample Rate	编码后码流的采样率
Channels	编码后码流的声道数
BitWidth	编码后输出数据的采样位宽
AttachSource	编码所绑定的音源输入通道
TryEncodeTimes	尝试编码的次数,用户一般不需要关注
EncodeFrameNum(Total/ Error)	音频编码器共编码的帧数以及错误帧数
FrameBuf(Total/Use/ Percent)(Bytes)	音频编码器输入缓存的总大小,已经使用大小,已经使用 的百分比(单位Bytes)
StreamBuf(Total/Use/ Percent)	音频编码器输出缓存的总大小,已经使用大小,已经使用 的百分比
SendFrame(Try/OK)	Try: 尝试往编码器输入PCM音频帧的次数 OK: 成功次数
ReceiveStream(Try/OK)	Try: 尝试从编码器获取编码完的音频帧的次数 OK: 成功获取次数
ReleaseStream(Try/OK)	Try: 尝试释放编码器输出的音频帧的次数 OK: 成功释放次数

【调试信息】

```
#echo save pcm start > /proc/msp/aenc00
#echo save_pcm stop > /proc/msp/aenc00
#echo save_es start > /proc/msp/aenc00
#echo save_es stop > /proc/msp/aenc00
#echo help > /proc/msp/aenc00
```

【调试信息分析】

通过Proc录制编码前的PCM数据:

echo save_pcm start|stop > /proc/msp/aencXX(XX为aenc通道数)

● 启动录制
echo save_pcm start > /proc/msp/aenc00
向aenc00发送存编码前PCM数据的命令(00是aenc通道的编号)。

● 停止录制 echo save pcm stop > /proc/msp/aenc00

- 成功录制之后,可以在设置的目录下找到如下类型文件:
 - aenc0 00.pcm, aenc0 01.pcm.
- aenc0_00.pcm: 第0次存aenc00的PCM数据;
- aenc0 01.pcm: 第1次存aenc00的PCM数据;

通过Proc录制编码后的ES数据:

echo save_es start|stop > /proc/msp/aencXX

- 启动录制
 - echo save_es start > /proc/msp/aenc00 向aenc00发送存编码后 ES数据的命令。
- 停止录制
 - echo save es stop > /proc/msp/aenc00
- 成功录制之后,可以在设置的目录下找到如下类型文件:
 - aenc0 00.aac, aenc0 01.aac.
- aenc0 00.aac: 第0次存aenc00的ES数据;
- aenc0_01.aac: 第1次存aenc00的ES数据;

显示Proc帮助信息:

echo help > /proc/msp/aencXX

2.24 PVR

2.24.1 pvr_play

【调试信息】

正常播放时的proc信息

```
# cat /proc/hisi/pvr xxxx/pvr play
------Hisilicon PVR Playing channel Info-----
chan 0 infomation
Play filename:/hdd/rec v33 a36.ts
Stram type: H264DemuxID:0
TsBuffer handle :0xa0401
Avplay handle:0x410000
Cipher handle:0
Play State: PLAYPlay Speed: 1024
Stream Read Pos:0x2cef4f4
Index Start:0
Index End: 6258
Index Last: 6258
IDR flag:0
B frame ref flag:1
Continuous flag:0
DispOptimize flag:0
Start Frm&GOP num:1024 25
Total GOP num:148
Max GOP size:53
Average GOP size:42
Index GOP distr:13 9 5 12 12 96 0 0 0 0 0 0
```

Index Read Now:882

快进播放时的proc信息

```
# cat /proc/hisi/pvr xxxx/pvr play
------Hisilicon PVR Playing channel Info-----
chan 0 infomation
Play filename:/hdd/rec v33 a36.ts
Stram type: H264
DemuxID:0
TsBuffer handle:0xa0401
Avplay handle:0x410000
Cipher handle:0
Play State:FF
Play Speed: 4096
Stream Read Pos:0x3ccf94a
Index Start:0
Index End: 6258
Index Last: 6258
Start Frm&GOP num:1024 25
Total GOP num:148
Max GOP size:53
Average GOP size:42
Index GOP distr:13 9 5 12 12 96 0 0 0 0 0 0
Index Read Now:1235
----- FF control -----
Status:FF Till End
Decodec ablity:0
Width:720
Heigth: 576
Disp Frame buffer:6
VO Frame buffer:3
Ori frame rate:25
VO In FrmRate(Int:dec):0:781
Max Dec Frm Num 1s:0
IDR flag:0
B frame ref flag:0
Continuous flag:0
DispOptimize flag:0
Field flag:0
---PVR Extract Frm info---
Extract or not:1
Entry frame num:2316
Try frame num:121
Start frame num:2196
End frame num:2316
frame cnt to send:0
Total frame num:0
Total GOP num:0
Total I frame num:0
Total P frame num:0
Total B frame num:0
```

快退播放启动快退优化时的proc信息

```
# cat /proc/hisi/pvr_xxxx/pvr_play
------Hisilicon PVR Playing channel Info-----
chan 0 infomation
Play filename:/hdd/rec_v33_a36.ts
Stram type:H264
DemuxID:0
TsBuffer handle:0xa0401
```

```
Avplay handle:0x410000
Cipher handle:0
Play State :FB
Play Speed: -4096
Stream Read Pos:0x11a22a40
Index Start:0
Index End: 6258
Index Last: 6258
IDR flag:0
B frame ref flag:1
Continuous flag:0
DispOptimize flag:0
Start Frm&GOP num:1024 25
Total GOP num:148
Max GOP size:53
Average GOP size:42
Index GOP distr:13 9 5 12 12 96 0 0 0 0 0 0
Index Read Now:5538
----- FB control -----
Status:FB Till Start
Decodec ablity:0
Width:720
Heigth: 576
Disp Frame buffer:6
VO Frame buffer:3
Ori frame rate:25
VO In FrmRate(Int:dec):6:250
Max Dec Frm Num 1s:0
IDR flag:0
B frame ref flag:0
Continuous flag:0
DispOptimize flag:0
Field flag:0
Supported max gop size:3
----FB 2X Optimize info--
FBOptimize flag:0
---PVR Extract Frm info---
Extract or not:1
Entry frame num:0
Try frame num:0
Start frame num:0
End frame num:35
frame cnt to send:18
Total frame num:36
Total GOP num: 3
Total I frame num: 3
Total P frame num:9
Total B frame num:24
Need drop I Frm:0
```

快退播放启动挑帧时的proc信息:

```
------Hisilicon PVR Playing channel Info-----
chan 0 infomation
Play filename:/tmp//rec_v518_a710.ts
Stram type:MPEG2
DemuxID:0
TsBuffer handle:0xa0401
Avplay handle:0x410000
Cipher handle:0
Play State:FB
```

```
Play Speed: -2048
Stream Read Pos:0x71af5f
Index Start:0
Index End:1020
Index Last:1020
Start Frm&GOP num:0 0
Total GOP num:84
Max GOP size:12
Average GOP size:12
Index GOP distr:0 83 0 0 0 0 0 0 0 0 0 0
Index Read Now:214
----- FB control -----
Status:Success
Decodec ablity:65
Width:720
Heigth: 576
Disp Frame buffer:6
VO Frame buffer:3
Ori frame rate:26
VO In FrmRate(Int:dec):3:250
Max Dec Frm Num 1s:326
IDR flag:0
B frame ref flag:0
Continuous flag:1
DispOptimize flag:0
Field flag:0
Supported max gop size:3
----FB 2X Optimize info--
FBOptimize flag:1
Display distance:7
FrmCnt not disp From Last :5
Display enable (this gop):1
Vdec FBOptimizeFlag(Should be 1):1
---PVR Extract Frm info---
Extract or not:0
```

【调试信息分析】

显示当前PVR所有播放通道的状态参数,其中proc路径pvr_xxxx中的xxxx为当前PVR播放的进程号。

参数	描述
Play filename	播放的文件路径及文件名
Stram type	视频格式
DemuxID	使用的DEMUX ID
TsBuffer handle	TS Buffer句柄
Avplay handle	AVPLAY句柄
Cipher handle	Cipher 句柄
Play State	播放状态
Play Speed	播放速度

参数	描述
Stream Read Pos	码流读取的位置
Index Start	起始Index编号
Index End	结束Index编号
Index Last	最后一个Index编号
Start Frm&GOP num	码流前n帧中包含的GOP数量
Total GOP num	码流中总共的GOP数量
Max GOP size	码流中最大的GOP的大小
Average GOP size	码流平均GOP大小
Index GOP distr	码流中GOP大小的分布情况
Index Read Now	当前读取的index 位置
Status	平滑控制状态
Decodec ablity	芯片解码能力
Width	当前码流宽度
Heigth	当前码流高度
Disp Frame buffer	视频解码帧存总数量
VO Frame buffer	显示占用帧存数量
Ori frame rate	码流帧率
VO In FrmRate(Int:dec)	输出帧率(Int整数部分 dec小数部分)
Max Dec Frm Num 1s	针对当前码流,解码模块1秒内最大支持解码帧数
IDR flag	码流是否包含IDR帧标志
B frame ref flag	码流B帧是否被参考标志
Continuous flag	码流连续标志
DispOptimize flag	显示优化标志
Field flag	场模式标志
Extract or not	是否进行挑帧
Entry frame num	下一次进入平滑控制的帧号(挑帧时才显示)
Try frame num	本次挑帧范围(挑帧时才显示)
Start frame num	本次挑帧范围的起始帧号(挑帧时才显示)
End frame num	本次挑帧范围的结束帧号(挑帧时才显示)
frame cnt to send	挑帧个数(挑帧时才显示)

参数	描述
Total frame num	码流总帧数(挑帧时才显示)
Total GOP num	码流总gop数(挑帧时才显示)
Total I frame num	总的I帧数(挑帧时才显示)
Total B frame num	总的P帧数(挑帧时才显示)
Need drop I Frm	需要丢弃的I帧数(快退挑帧时才显示)
FBOptimize flag	快退优化标志 (快退优化时才显示)
Display distance	快退显示间距 (快退优化时才显示)
FrmCnt not disp From Last	未被显示的帧数(快退优化时才显示)
Display enable(this gop)	GOP是否显示标志 (快退优化时才显示)
Vdec FBOptimizeFlag(Should be 1)	快退优化使能标志 (快退优化时才显示)

2.24.2 pvr_rec

【调试信息】

```
# cat /proc/hisi/pvr xxxx/pvr rec
-----Hisilicon PVR Recording channel Info-----
chan 0 infomation:
L2Cache support:No
Rec filename:/hdd/rec v513 a660.ts
Stream type:MPEG2
DemuxID:1
Record State:2
Rewind:1
Rewind Type:size
Rewind size:0x27fff1000
Rewind times:0
Max size:0x27fff1000
Max time:0
UserData size:4052
ClearStream:1
IndexType:1
IndexPid:0x201/513
Global offset:0x2a03ba
File offset:0x2a03ba
Index Write:83
CurrentTime(ms) :3204
IndexPos S/E/L:0/236/236
Event history
```

【调试信息分析】

显示当前PVR所有录制通道的状态参数,其中proc路径pvr_xxxx中的xxxx为当前PVR录制的进程号。

【参数说明】

参数	描述
Rec filename	录制文件的路径及文件名
L2Cache support	是否使用L2Cache
Stream type	视频格式
DemuxID	使用的demux ID
Record State	录制状态
Rewind	录制回绕标志
Rewind Type	回绕类型
Rewind size	回绕的大小
Rewind times	回绕的时间
Max size	录制文件最大大小限制
Max time	录制文件最大时间限制
UserData size	用户自定义数据区大小
ClearStream	是否清流标志
IndexType	录制的Index类型
IndexPid	录制的Index PID
Global offset	当前录制的总大小,包含被覆盖的
File offset	当前录制帧在文件中的偏移
Index Write	当前录制的帧号
CurrentTime(ms)	当前录制帧的时间
IndexPos	当前Index的开始帧/结束帧/最后一帧的位置
Event history	最近5次回调注册事件

2.25 VI

【调试信息】

cat /proc/msp/vi0000

------ VI Info ------

Type:Virtual

InputRect:0/0/640/480

BufferNum:6
State:Start

DstID:venc(port100)

:win(port101)
:win(port102)

CAM/USER->VI

CapFrame (Try/OK/Freq):500/500/19

VI->VPSS

Acquire(Try/OK):3160/500 Release(Try/OK):500/500

VPSS->VI

Acquire(Try/OK):3390/1500 Release(Try/OK):1498/1498

VI->MIN

SendFrame(Try/OK):1000/1000

VI->VENC

SendFrame(Try/OK):500/500

VI->USER

Acquire(Try/OK):0/0
Release(Try/OK):0/0
VIBuffer(Total/Used):6/0
VIBufferDetail:[0,0,0,0,0,0]

【调试信息分析】

记录VI模块相关状态,包括VI属性信息,当前处理帧的信息状态,以及缓冲区队列状态。

参数	描述
Туре	VI类型。
	Virtual: 虚拟VI。
InputRect	四个参数分别表示X坐标、Y坐标、宽和高。
BufferNum	帧缓存数目。
State	状态标识。 Start: 启动状态; Stop: 停止状态。
DstID	绑定后级模块信息。 后级可以绑定win和venc模块。 以win(port0)为例,win表示绑定win模块,port0表示此路VPSS端口 句柄为0。
CAM/USER->VI	摄像头或用户送帧到VI信息。 CapFrame(Try/OK/Freq):Try表示总共尝试送帧数目,OK表示总共成功送帧数目,Freq表示送帧频率。
VI->VPSS	VPSS从VI收帧信息。 Acquire(Try/OK): VPSS收帧数目,Try表示尝试收帧数目,OK表示成功收帧数目; Release(Try/OK): VPSS还帧数目,Try表示尝试还帧数目,OK表示成功还帧数目。

参数	描述
VPSS->VI	VI从VPSS收帧信息。
	Acquire (Try/OK): VI收帧数目,Try表示尝试收帧数目,OK表示成功收帧数目;
	Release (Try/OK): VI还帧数目,Try表示尝试还帧数目,OK表示成功还帧数目。
VI->WIN	VI向WIN发送帧信息。
	SendFrame(Try/OK): 发送帧数目,Try表示尝试发送数目,OK表示成功发送数目。
VI->VENC	VI向VENC发送帧信息。
	SendFrame (Try/OK): 发送帧数目,Try表示尝试发送数目,OK表示成功发送数目。
VI->USER	用户从VI获取帧信息。
	Acquire (Try/OK): 用户从VI获取帧数目, Try表示尝试收帧数目, OK表示成功获取帧数目;
	Release (Try/OK): 用户向VI还帧数目,Try表示尝试还帧数目,OK表示成功还帧数目。
VIBuffer	VI内部缓冲区队列状态信息。
	Total: 总共分配缓冲区数目;
	Used: 当前使用到的缓冲区数目。
VIBufferDetail	VI内部缓冲区队列各结点状态信息。
	0: 当前结点未被使用;
	1: 当前结点正在被VPSS使用;
	2: 当前结点使用完毕。

【调试手段】

VI调试手段,支持VI动态运行过程中,用户对VI内部行为或数据处理,用于定位相关问题。VI调试手段支持以下方式:

● 存储内部YUV数据帧

echo save_yuv > /proc/msp/vi0000

保存文件到/mnt/目录下,保存后,使用查看YUV图片相关软件,打开文件,查看定位。

● 时延统计

echo low delay stat start > /proc/msp/vi0000 开启时延统计。

待开启时延统计后,cat /proc/msp/low_delay_statistics可以查看各模块处理帧的耗时。待统计结束后,可以运行下面命令关闭时延统计。

echo low_delay_stat stop > /proc/msp/vi0000关闭时延统计。

2.26 VP

【调试信息】

```
# cat /proc/hisi/vp/vp info
-----Hisilicon VP Out Info-----
chan 0 infomation:
---- VP Source Info ----
Attached: YES
VP SrcMode: USBCAM
Cap Type:MJPEG
Src handle:0x10
Buf count:4
---- VP Preview Info ----
Work State:Run
Window ID:win0100
Avplay ID:avplay01
---- VP Sender Info ----
Work State:Run
Venc Type: H264
Venc ID:venc00
Avplay ID:avplay01
---- VP EmptyPack Info ----
EmptyPackage:Stop
---- VP Receiver Info ----
Work State:Run
Vdec Type :H264
Avplay_ID:avplay00
Window ID:win0101
---- VP Network Info ----
Ipv4:YES
Rtcp Enable: YES
SendInterval: 5000 (ms)
---Sender side Statics:
Send(byte):1315350354
Send (packet): 924580
LostRate:0%
Lost Packets:28
Jitter:3(ms)
Delay:0(ms)
---Receiver side Statics:
Recv(byte):1317308610
Recv (packet): 925951
LostRate:0%
Lost Packets:28
Jitter:3 (ms)
Delay:0(ms)
```

【调试信息分析】

显示VP的状态信息。

参数	描述	
Source Info	输入源信息:	
	Attached: 是否绑定了输入源,YES绑定,NO没有绑定。	
	VP SrcMode:输入源类型,USBCAM、VI。	
	Cap Type: USBCAM捕获数据类型,MJPEG、YUYV。	
	Src handl: 源句柄。	
	Buf count:分配的buf个数,范围4~16。	
Preview Info	本地预览信息:	
	Work State: 预览Run/Stop状态。	
	Window_ID: 预览使用的window实例ID。	
	Avplay_ID: 预览使用的avplay实例ID。	
Sender Info	发送端信息:	
	Work State: 发送Run/Stop状态。	
	Venc Type:编码类型。	
	Venc_ID: 发送本地视频使用的venc实例ID。	
	Avplay_ID: 发送本地视频使用的avplay实例ID。	
EmptyPack Info	空包信息:	
	EmptyPackage: 发送端是否正在发送空包,Run正在发空包,Stop没有发送空包。	
Receiver Info	接收端信息:	
	Work State: 接收Run/Stop状态。	
	Vdec Type:解码类型。	
	Avplay_ID:播放对端视频使用的avplay实例ID。	
	Window_ID: 播放对端视频使用的window实例ID。	

参数	描述
Network Info	网络信息:
	Ipv4: 是否是IPV4, YES是IPV4, NO是IPV6。
	Rtcp Enable: RTCP是否使能,YES使能,NO没有使能。
	SendInterval: RTCP发送间隔,单位毫秒。
	发送端统计:
	Send(byte): 发送字节数。
	Send(packet): 发送包总数。
	LostRate: 丢包率。
	Lost Packets: 丢包数。
	Jitter: 抖动,单位毫秒。
	Delay: 延时,单位毫秒。
	接收端统计:
	Recv(byte):接收字节数。
	Recv(packet):接收包总数。
	LostRate: 丢包率。
	Lost Packets: 丢包数。
	Jitter: 抖动,单位毫秒。
	Delay: 延时,单位毫秒。

2.27 PDM

【调试信息】

cat /proc/msp/pdm

-----Dispaly0-----

format: 11,

source display: 1

background color: 0x000000

HuePlus/Brightness/Contrast/Saturation: 50/50/50/50

virtual screen(Width/Height): 1280/720 offset(Left/Top/Right/Bottom): 0/0/0/0

bGammaEnable: 0 pixelformat: 9

aspectRatio: 4to3

CVBS: 3

-----Dispaly1-----

format: 6,

source display: 1

background color: 0x000000

HuePlus/Brightness/Contrast/Saturation: 50/50/50/50

virtual screen(Width/Height): 1280/720

offset(Left/Top/Right/Bottom): 0/0/0/0

bGammaEnable: 0

pixelformat: 9

aspectRatio: 16to9

HDMI: HDMI_0

YPbPr(Y/Pb/Pr): 1/2/0

-----Sound0-----

DAC: DAC0

SPDIF: SPDIF0

HDMI: HDMI0

I2S0 attr:

Master: TRUE

PcmSampleRiseEdge: TRUE

Master colock: 128*fs

Bit colock: 1 DIV

Channel: 1

Mode: standard

Bit Depth: 16

PCM Delay Cycle: 0

【调试信息分析】

显示baseparam分区的参数信息。

参数	描述
format	显示制式,与制式枚举 HI_UNF_ENC_FMT_E对应
source display	同源display

参数	描述
background color	背景色,如FF00FF,表示R、G、B分别 为0xFF,0x0,0xFF
HuePlus/Brightness/Contrast/Saturation	色度/亮度/对比度/饱和度
virtual screen(Width/Height)	虚拟屏幕宽高值
offset(Left/Top/Right/Bottom)	在虚拟屏幕上的偏移
bGammaEnable	是否使能Gamma
pixelformat	图形像素格式,与HIGO_PF_E枚举对应
aspectRatio	设备宽高比,与 HI_UNF_DISP_ASPECT_RATIO_E枚举 对应
CVBS	CVBS DAC
HDMI	HDMI端口号
YPbPr(Y/Pb/Pr)	YPbPr DAC
DAC	DAC端口
SPDIF	SPDIF端口
HDMI	HDMI端口
Master	主从模式。TRUE:主模式; FALSE:从模式
PcmSampleRiseEdge	Pcm采样边沿。TRUE: 上升沿; FALSE: 下降沿
Master colock	Master 工作时钟
Bit colock	Bit 工作时钟
Channel	I2S接口音频通道
Mode	I2S接口模式。Standard: I2S标准模式; pcm模式
Bit Depth	位宽
PCM Delay Cycle	PCM时隙

2.28 VENC

【调试信息】

cat /proc/msp/venc00
------ VENC[00] ------

```
Version:001.002.2015042101
CodecID:H.264(0x4)
Capability: 720P
Profile (Level): High (31)
Resolution: 1280X720
TargetBitRate: 4194 (kbps)
Gop:100
FrmRate(Input/OutPut):50/25(fps)
priority:0
QuickEncode: FALSE
DriftRateThr:NA
Split:Enable(FALSE)Size(0)
StreamBufSize: 1843 (KB)
MaxQP/MinQP:48/16
QLevel:0
Rotation:0
AutoRequestIFrm:Enable(FALSE)
----- Real-timeStatistics ------
WorkStatus:Start
SourceID:Win00
FrameInfo:SP420 UV
InputFrmRate(Use/Real):60/60(fps)
TargetFrmRate(Use/Real):25/25(fps)
BitRate:3556(kbps)
EncodeNum:268
SkipNum: Total (370) FrmRateCtrl (370) SamePTS (0) QuickEncode (0)
TooFewBuf(0) TooManyBits(0),ErrCfg(0)
FrameBuffer:
VpssQueueBuf(Total/Used) :NA/NA
VpssImgBuf(Total/Used) :NA/NA
VencQueueBuf(Total/Used):6/0
StreamBuffer:
Total/Used/Percent(Bytes):1843200/0/0%
Statistics(Total):
AcquireFrame(Try/OK):950/638
ReleaseFrame(Try/OK):638/638
AcquireStream (Try/OK):1659/274
ReleaseStream(Try/OK):274/274
Statistics (PerSecond):
AcquireFrame(Try/OK):83/60
ReleaseFrame (Try/OK):59/59
```

【调试信息分析】

编码器单元(VENC)调试信息给出了编码器种类,参数等信息,编码器单元将输入帧进行编码,得到裸码流传输给下一个单元处理,调试信息同时给出了帧数据和码流数据信息。

参数	描述	
CodecID	编码器种类:	
	0x1: MPEG-4	
	0x4: H.264	
Capability	编码能力级: QCIF, CIF, D1, 720P, 1080P, UNKNOWN	

参数	描述	
Profile(Level)	编码等级(档次)信息:	
	支持的等级: Baseline,Main,High profile	
	支持的档次: 1.0~4.1不定	
Resolution	编码输出分辨率	
TargetBitRate	用户指定的目标码率,单位kbps;	
Gop	一个画面组至少含有一个I帧,Gop表示一个图像组的长度。	
FrmRate	帧率:	
	Input: 实际输入帧率,单位fps;	
	Encoded: 实际编码帧率,单位fps。	
Priority	通道优先级,取值范围0~7,从0到7优先级依次增加。	
QuickEncode	快速编码模式使能: FALSE不使能; TRUE使能。	
DriftRateThr	编码器码率波动阈值,如设置成20,表示波动阈值为20%	
Split	分slice编码相关配置:	
	Enable: 是否使能。FALSE不使能,TRUE使能。	
	Size: 分slice编码使能情况下的slice分割大小,单位是byte。	
StreamBufSize	用户指定的码流buffer大小,单位: KB。	
MaxQP/MinQP	量化参数:取值范围0~51。.	
	MaxQP: 用户指定的最大量化参数;	
	MinQP: 用户指定的最小量化参数。	
QLevel	JPEG编码量化参数:取值范围1~99.	
	量化参数越大,JPEG图像质量越好。	
Rotation	旋转角度。	
WorkStatus	编码实例的工作状态: Start启动中; Stop停止。	
SourceID	编码源相关信息:	
	字符串表示源的类别,后面的数字代表对应的句柄编号。	
	如: VI01, Win02。	
FrameInfo	输入帧格式信息:	
	英文字符串代表存储格式,紧跟数字代表采样格式,'_'后代表具体像素排列信息。	
	如: SP420_UV, Planer420, Package422_YUYV等。	
InputFrmRate	输入帧率实时信息: 单位: fps。	
	Use: 编码器实际采用的输入帧率数值。	
	Real: 编码器实时计数得出的输入帧率。	

参数	描述		
OutputFrmRate	输出帧率实时信息: 单位: fps。		
	Use: 编码器实际采用的输出帧率数值。		
	Real: 编码器实时计数得出的输出帧率。		
	备注说明:以上InputFrmRate,OutputFrmRate可能会出现实际采用的帧率与用户设置的对应帧率不一致的情况。当绑定模式下,输入帧率以内部帧信息为准,用户指定的输出帧率大于帧信息的帧率时,输出帧率采用帧信息中的帧率,从而避免输出帧率>输入帧率的情况。		
BitRate	实际编码码率,单位kbps。		
EncodedNum	编码的实际帧数目		
SkipNum	跳过编码的帧数目:		
	Total:跳帧总数;		
	FrmRcCtrl: 由于帧率控制导致的跳帧数目;		
	SamePTS: 由于相同PTS导致的跳帧数目;		
	QuickEncode: 由于快速编码导致的跳帧数目;		
	TooFewBuffer:由于内部码流buffer空间不足导致的跳帧数目;		
	ErrCfg: 由于帧信息配置有误导致的跳帧数目。		
FrameBuffer	帧存buffer使用情况:		
	VpssQueueBuf(Total/Used): VENC创建的VPSS输入帧存队列总数/已用帧数;		
	VpssImgBuf(Total/Used): VENC创建的VPSS输出帧存队列总数/已用帧数;		
	VencQueueBuf(Total/Used): VENC帧存总数/已用帧数;		
StreamBuffer	内部码流buffer使用情况:单位Bytes		
	Total/Used/Percent: 码流buffer总大小/已用大小/已用所占百分比(%)。		
Statistics(Total)	总体统计信息:		
	AcquireFrame (Try/OK): 尝试/实际成功获取帧的总数目;		
	ReleaseFrame (Try/OK): 尝试/实际成功释放帧的总数目;		
	AcquireStream(Try/OK): 尝试/实际成功获取码流的总数目;		
	ReleaseStream(Try/OK): 尝试/实际成功释放码流的总数目。		
Statistics	输入帧的信息:		
(PerSecond)	AcquireFrame (Try/OK): 每秒尝试/实际成功获取帧的数目;		
	ReleaseFrame (Try/OK): 每秒尝试/实际成功释放帧的数目。		

【调试手段】

● 在编码器工作时,可以通过写proc文件,读出VENC中正在处理的码流数据,存成文件,文件默认存放在/mnt目录下,方便用户调试。

echo save stream P1 P2 > /proc/msp/vencXX.

● 当P1=second时, P2参数用作时间参数, 以秒为单位, 把编码器生成的码流保存为 文件;

例: echo save stream second 5>/proc/msp/venc表示读取5秒码流数据,生成文件

- 当P1=frame时,P2参数用作帧参数,以帧数为单位,把编码器生成的码流保存为文件;
- 例: echo save_stream frame60>/proc/msp/venc表示读取60帧码流数据,生成文件,用户将此码流文件从单板中读取出来,用于调试。
- 在编码器工作时,可以通过写proc文件,读出VENC编码前的YUV数据,存成文件,方便用户调试。

echo save_yuv start > /proc/msp/vencXX

● 开始存YUV文件

echo save_yuv stop > /proc/msp/vencXX

- 停止存YUV文件
- 在编码器工作时,可以通过写proc文件,控制VENC硬件低功耗相关配置,方便用户调试。

echo ClkGateEn P1 > /proc/msp/vencXX

- 当P1=0时,关闭VEDU时钟门控
- 当P1=1时,只打开VEDU 帧级的时钟门控
- 当P1=2时,同时打开VEDU帧级和宏块级时钟门控

echo LowPowEn P1 > /proc/msp/vencXX

- 当P1=0时,关闭编码算法低功耗
- 当P1=1时,打开编码算法低功耗

2.29 AI

【调试信息】

cat /proc/msp/ai0

----- AI[0] Status -----

Status:start AiPort:I2S0

SampleRate:48000
PcmFrameMaxNum:6

PcmSamplesPerFrame:30720

DmaCnt:1960 BufFullCnt:0 FiFoFullCnt:0

FrameBuf (Total/Use/Percent) (Bytes):184320/82176/44%

AcquireFrame(Try/OK):98/32
ReleaseFrame(Try/OK):97/97

【调试信息分析】

显示AI的状态。

参数	描述	
Status	工作状态。start/stop分别标识工作与非工作状态	
AiPort	音频输入端口	
SampleRate	音频输入采样频率	
PcmFrameMaxNum	缓存最大帧存数量	
PcmSamplesPerFrame	每帧数据采样点数	
DmaCnt	Dma搬运次数	
BufFullCnt	缓存过载次数	
FiFoFullCnt	FiFo过载次数	
FrameBuf(Total/Use/ Percent)(Bytes)	缓存的总大小(单位: Bytes),已经使用大小(单位: Bytes),已经使用的百分比	
AcquireFrame(Try/OK)	Try: 用户获取帧存的次数 OK: 成功次数	
ReleaseFrame(Try/OK)	Try: 用户释放帧存的次数 OK: 成功次数	

2.30 VPSS

【调试信息】

cat /proc/msp/vpss00

Proc信息共分两部分:

```
|------|
|-------|
```

分别表示:

- 实例信息。
- PORT信息。

2.30.1 实例信息

实例信息如下:

```
------VPSS0000-------|
ID:0x0|
State:working|
Priority:0|
QuickOutPut:off|
SourceID:Vdec(00)|
Version:2013082419|
------ Algorithm------|
```

```
P/I Setting:auto|
Deinterlace: 5 field |
Sharpness:auto|
*ProgRevise:on
-----Detect Info-----|
TopFirst(Src):NA(Top) |
InRate(Src):0(25000)|
*Trans:OFF|
Progressive/Interlace(Src):P(P) |
----SourceFrameList Info----|
(source to vpss) |
Mutual Mode: vpss active |
GetSrcImg(Try/OK): 25/25|
GetOutBuf(Try/OK):118/25|
ProcessHZ(Try/OK):118/25 |
Acquire(Try/OK): |
2742/2740|
Release(Try/OK): |
2739/2739|
```

【实例信息分析】

记录VPSS模块实例信息,包含四部分,实例状态、算法配置、检测信息、前级BUFFER交互信息。

参数	描述	
ID	实例ID	
State	实例工作状态。	
	Woring/Stop: VPSS实例是否工作	
Priority	实例优先级。	
	0~32: 默认16。0~32优先级由低到高	
QuickOutPut	快速输出模式。	
	On/off: 打开时,VPSS处理前级的最新一帧	
Source Module ID	前级模块名(模块号。	
	模块名: Vi、Vdec、Venc	
	模块号:00、01、02	
P/I Setting	逐隔行检测模式	
	I/P/auto:分别表示强制指定输入源为隔行、强制指定输入源为逐行、自动矫正逐隔行	
Deinterlace	去隔行算法模式。	
	Off/ auto/3 field/4 field/5 field: 分别表示关闭、根据图像信息自适应、3场模式、4场模式、5场模式	
Sharpness 锐化算法模式。		
	Off/ on/ auto: 分别表示关闭、打开、根据图像信息自适应	
TopFirst(Src)	检测出的场序信息(码流中的场序信息)	

参数	描述	
InRate(Src)	检测出的帧率信息(码流中的帧率信息)	
Progressive/Interlace(Src)	矫正后的逐隔行信息(码流中的逐隔行信息)	
Mutual Mode	与前级交互模式。	
	src active/vpss active: 分别表示前级推送模式、VPSS主 动获取模式	
GetSrcImg(Try/OK)	与前级模块的BUFFER交互实时统计。	
	Try/OK:分别表示1秒内主动向前级请求帧次数、请求成功次数	
GetOutBuf(Try/OK)	获取输出队列BUFFER实时统计。	
	Try/OK:分别表示1秒内查询输出BUFFER的次数、输出BUFFER可写次数	
ProcessHZ(Try/OK)	实例处理次数实时统计。	
	Try/OK:分别表示服务线程一秒内查询实例的次数、一秒内处理当前实例的次数	
Acquire(Try/OK)	与前级模块的BUFFER交互统计。	
	Try/OK:分别表示获取前级帧次数、获取成功次数	
Release(Try/OK)	与前级模块的BUFFER交互统计。	
	Try/OK: 分别表示释放前级帧次数、释放成功次数	

2.30.2 PORT 信息

PORT信息如下:

```
-----PortInfo-----
         :0x0
ID
State
                       :on
State :on
PixelFormat :YCrCb420
Resolution :1280*720
ColorSpace :BT709_YUV
DisplayPixelAR(W/H):16/9
Aspect Mode :Full
Support3DStream :off
MaxFrameRate :2500
LowDelay :off
VerticalFlip :off
Rotation :00
                       :off
*ZME1LResolution: 0*0
-----OutFrameList Info-----
       (vpss to sink)
BufManager :vpss
BufNumber
                         :6+0
                         :5
BufFul
BufEmpty
                         :1
AcquireHZ
                         :26
```

Acquire(Try/OK): 2719/2719
Release(Try/OK): 2714/2714

参数	描述
ID	PORT的ID。0xfffffff表示该PORT未创建
State	PORT状态。
	Off/on: 关闭/打开
PixelFormat	输出像素排布格式
Resolution	输出分辨率
ColorSpace	输出色域空间
DisplayPixelAR(W/H)	输出显示设备画幅比
Aspect Mode	宽高比转换算法模式。
	Full/ LBOX/ PANSCAN/ COMBINED/ FULL_H/ FULL_V/ CUSTOMER/: 分别 表示相应的宽高比转换模式。
Support3DStream	输出3D FramePacking格式帧。 该模式打开时,3D码流经过处理输出 FramePacking数据
MaxFrameRate	输出最大帧率。 输出最大帧存小于输入帧率时,VPSS会 做丢帧处理
LowDelay	低延时模式使能
HorizonFlip	水平翻转开关
VerticalFlip	垂直翻转开关
Rotation	旋转模式。
	Off/90/180/270: 分别表示关闭、90度旋转、180度旋转、270度旋转
BufManager	输出帧存管理模块。
	Vpss/usr: vpss管理/用户管理
BufNumber	输出PORT上的帧存个数。
	M+N: M表示左眼帧存个数,N表示右眼帧存个数
BufFul	当前已经写入的帧存个数
BufEmpty	当前空闲的帧存个数
Acquire HZ	后级每秒获取帧计数

参数	描述
Acquire(Try/OK)	与后级模块的BUFFER交互统计。 Try/OK:分别表示后级获取帧次数、获取成功次数
Release(Try/OK)	与后级模块的BUFFER交互统计。 Try/OK:分别表示后级释放帧次数、释 放成功次数

2.31 KEYLED

【调试信息】

Hisilicon KeyLed Info		
:Open		
:Open		
:CT1642		
:500 (ms)		
:Enable		
:Enable		
:200 (ms)		
:0xbeb666f2		
:12:25(hour:minute)		
:Fourth		
:1		
:8		
:6		
:6		
:6		
:6		

【调试信息分析】

记录了KEYLED模块的状态信息。

参数	描述
KeyStatus	按键状态:
	Open: 打开;
	Close: 关闭。
LedStatus	数码管状态:
	Open: 打开;
	Close: 关闭。

参数	描述
Select KeyLed Type	当前使用的前面板驱动类型:
	PT6961: 使用PT6961芯片的前面板;
	CT1642: 使用CT1642芯片的前面板;
	PT6964: 使用PT6964芯片的前面板;
	FD650: 使用FD650芯片的前面板。
Timeout of Reading Key	获取按键的超时时间,单位为ms。
Key Up Report	是否设置了上报按键弹起信息:
	Enable: 上报;
	Disable: 不上报。
Repeat Key Report	是否设置了上报重复按键:
	Enable: 上报;
	Disable: 不上报。
Repeat Key Report time	重复按键上报间隔时间,单位ms。
Led Display Code	数码管上显示的数字或字符对应的码 值。
Led Display Time	数码管上显示的时间(小时:分钟)。
Led Flash Pin	数码管的闪烁位置(当驱动支持闪烁时 有效):
	Not Set: 没有设置闪烁;
	First: 第一个数码管闪烁;
	Second: 第二个数码管闪烁;
	Third: 第三个数码管闪烁;
	Fourth: 第四个数码管闪烁;
	ALL: 四个数码管都闪烁;
	NONE: 四个数码管都不闪烁。
Led Flash Pin Level	数码管的闪烁级别:最小为1,最大为 5,数字越大代表闪烁频率越高。
Key Buffer Lenth	按键缓冲区的长度。
Key Buffer Head	按键缓冲区的读指针位置。
Key Buffer Tail	按键缓冲区的写指针位置。
Key Come Number	驱动收到的按键的总数。
Key Read Number	驱动被读取的按键的总数。

2.32 PMOC

【调试信息】

cat /proc/msp/pm -----Hisilicon PM Info-----Scene :Standard Ethernet WakeUp :Disable USB WakeUp :Disable Display Mode :Display Time Display Value :0x00093a00 :4294967295(s) WakeUp TimeOut :CT1642 KeyLed Type :0 :NEC Simple KeyLed WakeUp Key IR Type IR WakeUp Key Number :1 IR WakeUp Key :High 32-bit(0x0), Low 32-bit(0x639cff00) Wake Up Reset :Enable :Save debug info to ram Debug Mode Gpio wake up port :255 Power off GPIO Number :0 Power off GPIO Direction :0

【调试信息分析】

记录了PMOC模块的状态信息。

参数	描述
Scene	待机场景:
	Standard: 标准场景;
Ethernet WakeUp	网络唤醒设置:
	Enable: 使能;
	Disable: 未使能。
USB WakeUp	USB鼠标键盘唤醒设置:
	Enable: 使能;
	Disable: 未使能。
Display Mode	待机时前面板显示模式:
	No Display:无显示;
	Display Number:显示数字;
	Display Time:显示时间。

参数	描述
Display Value	待机时前面板数码管显示的数字或时间: 当Display Mode 设置为Display Number时,bit[31:24]表示第一个数码管显示的数字(0-9),bit[23:16]表示第二个数码管显示的数字,bit[15:8]表示第三个数码管显示的数字,bit[7:0]表示第四个数码管显示的数字; 当Display Mode设置为Display Time时,bit[31:24]没有意义,bit[23:16]表示小时,bit[15:8]表示分钟,bit[7:0]表示秒。
WakeUp TimeOut	待机定时唤醒时间间隔设置,单位s。
KeyLed Type	待机唤醒时使用的前面板驱动类型: PT6961:使用PT6961芯片的前面板; CT1642:使用CT1642芯片的前面板; PT6964:使用PT6964芯片的前面板; FD650:使用FD650芯片的前面板。
KeyLed WakeUp Key	· 待机唤醒时使用的前面板按键值。
IR Type	特机唤醒时使用的遥控器类型: NEC Simple: NEC Simple协议类型的遥控器; TC9012: TC9012协议类型的遥控器; NEC Full: NEC Full协议类型的遥控器; SONY: SONY协议类型的遥控器; RAW: 支持多种协议类型的遥控器同时解析。
IR WakeUp Key Number	待机唤醒时使用的遥控器唤醒值的个数,依赖于IR Type; IR Type为RAW时最多可以设置6组,为其他值时只能设置1组。
IR WakeUp Key	待机唤醒时使用的遥控器按键值,分高 32Bit和低32Bit。
Wake Up Reset	待机唤醒时是否重启单板: Enable: 重启; Disable: 不重启。

参数	描述
Debug Mode	调试模式:
	No debug info: 没有调试信息;
	Save debug info to ram: 保存调试信息到内存中;
	Show debug info to LED:显示调试信息 到数码管上;
	Save to ram & Show to LED: 保存调试信息到内存的同时也显示调试信息到数码管上;
	Show debug info in serial port: 打印调试信息到串口。
Gpio wake up port	GPIO唤醒端口,255表示没有使能,目前有效值只支持待机时不下电的GPIO,具体端口范围请参考芯片手册。
Power off GPIO Number	设置待机时单独控制电源的GPIO开关的 管脚号。
Power off GPIO Direction	设置待机时单独控制电源的GPIO开关的 输出是高电平还是低电平。

待机唤醒模式默认原地唤醒,即唤醒到进入待机前的状态;如果希望设置成唤醒重启模式,可以输入如下命令:

#echo reset=1 > /proc/msp/pm

如果希望改回原地唤醒模式,可以再输入如下命令:

#echo reset=0 > /proc/msp/pm

待机默认不打开调试模式,如果想查看进入待机时的唤醒参数,可以输入如下命令#echo debug=4 > /proc/msp/pm

2.33 HiPlayer

2.33.1 查看 HiPlayer proc 功能命令

在串口输入: echo help >/proc/hisi/hiplayer00/control

hiplayer00: 00为hiplayer实例序号,如果创建两个hiplayer,则会有hiplayer00,hiplayer01。

【命令信息】

echo help >/proc/hisi/hiplayer00/control
*********Hisilicon HiPlayer00 Help Info Begin**********
echo SendAud = 1|0 >/proc/hisi/hiplayer00/control

【命令说明】

命令	功能
echo SendAud = 1 0	控制是否输出音频数据。
>/proc/hisi/hiplayer00/ control	0: HiPlayer不送音频数据给AVPlay,当AVPlay缓冲播放 完后,就会无音频输出;
	1: HiPlayer送音频数据给AVPlay。 默认值为1。
echo SendVid = 1 0	控制是否输出视频数据。
>/proc/hisi/hiplayer00/ control	0: HiPlayer不送视频数据给AVPlay,当AVPlay缓冲播放 完后,视频画面静帧;
	1:HiPlayer送视频数据给AVPlay。
	默认值为1。
echo SendSubt = 1 0	控制是否输出字幕数据。
>/proc/hisi/hiplayer00/ control	0: HiPlayer不送字幕数据给SO字幕输出模块,当字幕输出模块缓冲字幕数据输出完后,就会无字幕输出;
	1: HiPlayer送字幕数据给SO字幕输出模块。
	默认值为1。
echo DumpAVEs = 1 0	控制是否启动音视频es数据存储。
>/proc/hisi/hiplayer00/ control	注意:在执行该命令前必须先建立/sdcard/hiplayer目录,音视频es数据文件会默认存储在该目录,并保证目录可写权限。
	1:启动HiPlayer es数据存储功能;
	0: 关闭es数据存储功能。
	生成文件如下:
	# ls /sdcard/hiplayer/ -l
	hiplayer_aud_stream_00.dat
	hiplayer_vid_stream_00.dat
	播放一条码流过程中,如果多次执行开关操作,会生成 多个es文件,以文件尾序号xx来区分。

以上命令设置后,可以查看设置的值信息:

● 通过cat /proc/hisi/hiplayer00/control查看设置后的值信息。

2.33.2 查看文件信息命令

在串口输入: cat /proc/hisi/hiplayer00/fileinfo

【调试信息】

```
# cat /proc/hisi/hiplayer00/fileinfo
************Hisilicon HiPlayer00 Media File Info Begin*********
Stream type: ES
Source type: LOCAL
File size:5273124381 bytes
Start time:0:0:0
Duration:0:41:29
bps:16945523 bits/s
Is Divx File:NO
Program 0:
video 0 info:
stream idx:1
format:H264
w * h:1920 * 1080
fps:23.976
bps:15406664 bits/s
duration:0:41:29
audio 0 info:
stream idx:0
format:PCM
samplerate:48000 Hz
bitpersample: 16
channels:2
bps:16 bits/s
lang:eng
subID:-1
duration:0:41:29
******* Misilicon HiPlayer00 Media File Info
```

【调试信息分析】

通过该命令可以查看当前播放的文件音视频信息,如果出现音频或视频启动播放失败的情况,可以通过该命令查看对应的format是否支持。

参数	功能
Stream type	码流类型,TS或者ES
Source type	媒体类型,LOCAL(本地)或者NETWORK(网络)
File size	文件长度,单位为字节
Start time	起始播放时间
Duration	播放时长
fps	文件帧率,单位是帧/秒
bps	文件码率,单位为比特/秒
Is Divx File	是否为Divx文件,YES(是)或者NO(不是)
Program n	文件中的一路节目,可包含多个视频,多个音频,多个字幕流
Video n info	第n路视频的视频信息

参数	功能
Audio n info	第n路音频的音频信息
stream idx	此路在节目中的序号
format	编码格式类型
w * h	宽和高
samplerate	采样率
bitpersample	采样位宽
channels	声道个数
lang	音频语言
subID	子音频流ID,如果为TrueHD音频,则可能含有一路ac3子流,当前处理策略为将两条流分离开来,并用此字段来记录关系,没有子流则该项值为-1

2.33.3 查看播放器控制信息

在串口输入: cat /proc/hisi/hiplayer00/control

【调试信息】

```
cat /proc/hisi/hiplayer00/control
*****Hisilicon HiPlayer00 Out Info Begin******
        HiPlayer Control Info Begin
CurStatus: PLAY
GotoStatus: INVALID
PlaySpeed: (x)
CurProgramId:0
CurVideoId:0
CurAudioId:0
CurSubtitleId:0
EndOfFile:0
StopCurOpe: 0
EmptyCount:0
PlayProgress:71(0-100)
DownloadProgress:0(0-100)
LastPlayDuration:1773294 (ms)
FirstPts:0(ms)
LastPts: 1784242 (ms)
LastIFramePts:1783783(ms)
LastSeekPts:1773294 (ms)
CallSeekPts:-1(ms)
FirstPtsAfterSeek:0(ms)
CallSeekPos: -1 (byte)
LastVidPos: 3763485855 (byte)
SubtitleCodecType: 0
LogLevel:1 (0|1|2|3|4 =
QUITE | FATAL | ERROR | WARNING | INFO)
         HiPlayer Control Info End
          HiPlayer Tplay Control Info Begin
TotalSendIFrame: 1783
SystimeOfSendLastIFrame:166310397(ms)
```

```
TotalIFrameScale:0(ms)
BaseSystimeOfTPlay:0(ms)
BaseIFramePtsOfTPlay:0(ms)
NumOfDiscardIFrame: 0
NumOfSendPBFrame: 0
        _HiPlayer Tplay Control End
       HiPlayer NetBuffer Info Begin
StatusOfBuffer: INVALID
DurationOfFormatBuffer:0 (ms)
SizeOfFormatBuffer: 0 (bytes)
DurationOfPlayerBuffer:0(ms)
SizeOfPlayerBuffer: 0 (bytes)
DownloadFinish:0
ResetPlayer:0
ReportFirstFrameTime:1
SetBufferUnderrun: 0
UnderrunCount:0
        HiPlayer NetBuffer Control End
        HiPlayer Proc Control Info Begin
CurStreamIdx:0
CurFrameType:AUD
CurFrameSize: 4096 (bytes)
CurFramePts: 1784746 (ms)
SendAudio:0
SendVideo:0
SendSubtitle:0
DumpAVEsData:1
PrintFrameInfo:0
ReadFrameFromDemux(Try/OK) :128478/128478
SendVidToAVPlay(Try/OK):65183/42781
SendAudToAVPlay(Try/OK):206776/85696
SendSubToSo(Try/OK):0/0
       HiPlayer Proc Control End
*****Hisilicon HiPlayer00 Out Info End******
```

【调试信息分析】

查看HiPlayer媒体播放控制信息。

参数	功能
CurStatus:PLAY	播放器当前状态。
GotoStatus:INVALID	播放器要转换的状态。INVALID为无状态转换,比如,如果用户执行了Stop操作,则GotoStatus应该为STOP。
PlaySpeed:(x)	播放器播放倍速,比如: 4倍速、8倍速、16倍速等。
CurProgramId:0	当前播放的节目ID,一条码流里面可能含有多个节目,类似TS流。
CurVideoId:0	当前播放的视频ID,假如播放的节目中有3路视频,则该ID的值应该是0-2。
CurAudioId:0	当前播放的音频ID,假如播放的节目中有5路音频,则该ID的值应该是0-4。

参数	功能
CurSubtitleId:0	当前播放的字幕ID,假如播放的节目中有9路字幕,则该ID的值应该是0-8。
EndOfFile:0	文件是否读完。
	1: 表示文件读完了,但此时AVPlay里面的缓冲可能 还未播放完,播放器还会继续播放缓冲数据; 0: 表示文件还未读完。
StopCurOpe:0	是否强制停止当前播放器的操作,调用 HI_SVR_PLAYER_Invoke(HI_FORMAT_INVOKE_P RE_CLOSE_FILE)接口强制停止当前操作。
	1: 用户执行强制停止当前操作; 0: 用户未执行强制停止操作。
	, , , , , , , , , , , , , , , , , , , ,
EmptyCount:0	文件读完后,播放器判断剩余缓冲是否播放完的次 数。
PlayProgress:71(0-100)	当前播放进度,百分比为单位。
DownloadProgress:0(0-100)	当前下载进度,百分比为单位。
LastPlayDuration:1773294(ms)	此信息无用,请忽略。
FirstPts:0(ms)	第一次送给AVPlay的帧时间戳。
LastPts:1784242(ms)	最后一次送给AVPlay的帧时间戳。
LastIFramePts:1783783(ms)	最后一次送给AVPlay的I帧时间戳。
LastSeekPts:1773294(ms)	保留字段。
CallSeekPts:-1(ms)	用户是否执行Seek操作,如果该属性值不为-1,则 hiplayer会执行Seek操作,>=0的值为有效值。
FirstPtsAfterSeek:0(ms)	保留字段。
CallSeekPos:-1(byte)	保留字段。
LastVidPos:3763485855(byte)	最后一次送给AVPlay的帧在文件中的字节偏移位 置。
SubtitleCodecType:0	保留字段。
LogLevel	应用程序通过调用hiplayer接口 "HI_FORMAT_INVOKE_SET_LOG_LEVEL"设置的log级别,仅当echo SetLogLevel = -1 0 1 2 3 4 5 6 >/proc/hisi/hiplayer/common命令设置值为-1时,应用程序设置的LogLevel值有效
TotalSendIFrame:1783	Tplay总共送的I帧个数,从快进到快退或快退到快进会清0。
SystimeOfSendLastIFrame: 166310397(ms)	最后一次送的I帧时间点,通过系统函数gettimeofday 获取。

参数	功能
TotalIFrameScale:0(ms)	所有送的I帧数据间隔总和,相邻pts相减差值总和。
BaseSystimeOfTPlay:0(ms)	启动Tplay操作的时间点,通过gettimeofday获取。
BaseIFramePtsOfTPlay:0(ms)	Tplay第一次送的I帧时间戳。
NumOfDiscardIFrame:0	Tplay丢弃的I帧个数,显示时间已过(当前系统时间减去Tplay起始时间大于帧pts)则丢弃。
NumOfSendPBFrame:0	送的PB帧总个数。
StatusOfBuffer:INVALID	网络播放当前Buffer状态。
DurationOfFormatBuffer:0(ms)	解析器缓冲Buffer中含有的剩余数据duration。
SizeOfFormatBuffer:0(bytes)	解析器缓冲Buffer中含有的剩余数据字节数。
DurationOfPlayerBuffer:0(ms)	Player缓冲Buffer中含有的剩余数据duration。
SizeOfPlayerBuffer:0(bytes)	Player缓冲Buffer中含有的剩余数据字节数。
DownloadFinish:0	0: 文件未下载完; 1: 文件已下载完。
ResetPlayer:0	0:未发生清楚缓冲数据操作,在直播场景下可能会 发生; 1:发生缓冲数据被清操作。
ReportFirstFrameTime:1	此信息无用,请忽略。
SetBufferUnderrun:0	是否设置了播放器在数据不够的情况下自动暂停的功能。 1:自动暂停并继续缓冲,缓冲足够后自动恢复;
UnderrunCount	0:缓冲不够不会自动暂停。 播放器欠载次数,重新播放后清0
CurStreamIdx:0	当前流ID,值为AUDIO/VIDEO/SUBTITLE中的一个。
CurFrameType:AUD	当前读取的帧的类型。
CurFrameSize:4096(bytes)	当前读取的帧的字节数。
CurFramePts:1784746(ms)	当前读取的帧的时间戳。
SendAudio:0	0: 不送音频数据; 1: 正常送音频数据。
SendVideo:0	0: 不送视频数据; 1: 正常送视频数据。
SendSubtitle:0	0: 不送字幕数据;1: 正常送字幕数据。

参数	功能
DumpAVEsData:1	1: 保存音视频ES数据; 0: 不保存音视频ES数据。
PrintFrameInfo:0	是否设置了打印帧信息标示。 1: 打印音视频、字幕帧信息; 0: 关闭音视频、字幕帧信息打印。
ReadFrameFromDemux(Try/OK):128478/128478	尝试从文件解析器读取数据的次数/从文件解析器读取数据成功次数。如果Try不停增加,但是OK不增加,说明解析异常,卡顿情况下可以先看看是否读数据正常。
SendVidToAVPlay(Try/OK): 65183/42781	HiPlayer发送视频数据给AVPlay的次数/发送成功的次数。 如果视频卡顿,看看是否Try不停增加,OK不增加, 说明一直发送失败。
SendAudToAVPlay(Try/OK): 206776/85696	HiPlayer发送音频数据给AVPlay的次数/发送成功的次数。 如果音频卡顿,看看是否Try不停增加,OK不增加, 说明一直发送失败。
SendSubToSo(Try/OK):0/0	HiPlayer发送字幕数据给AVPlay的次数/发送成功的次数。 如果音视频卡顿,看看是否Try不停增加,OK不增加,说明一直发送字幕失败,导致无法及时发送音视频数据。

2.33.4 查看解析器信息

在串口输入: cat /proc/hisi/hiplayer00/format

【调试信息】

```
cat /proc/hisi/hiplayer00/format
*******Hisilicon HiPlayer00 Format Info Begin********
        _HiPlayer Format Demux Info Begin_
FileUrl:/mnt/[SD] AVC 478Kbps 848x480.mov
DemuxName: libffmpegformat.so
SupportFormatName:
SupportProtocolName:
Merit :0
PreCloseFile:0
GetMsqFin:1
        HiPlayer Format Demux Info End
        HiPlayer Format Parameter Info Begin
BufferConfigType:SIZE
MaxBufferSize:10485760 (bytes)
TimeOutOfNetwork:120000(ms)
BufferStart:524288(size-bytes/time-ms)
BufferEnough:4194304(size-bytes/time-ms)
```

BufferFull:5242880 (size-bytes/time-ms)

HlsStartMode:NORMAL
DstSubCodeType:UTF-8
Cookie:NotSet

Cookie:NotSet
UserAgent:NotSet
Referer :NotSet

HiPlayer Format Parameter Info End

******Hisilicon HiPlayer00 Format Info End********

【调试信息分析】

查看当前使用的解析器信息。

参数	功能
FileUrl: /mnt/ [SD]_AVC_478Kbps_848x480.mov	播放的文件url。
DemuxName: libffmpeg.so	使用的文件解析器名称。
SupportFormatName:	该解析器支持的文件格式。
SupportProtocolName:	该解析器支持的协议格式。
Merit:0	该解析器优先级,默认为0。 值从0x0~0xFFFFFFF。
PreCloseFile:0	保留字段。
GetMsgFin:1	保留字段。
BufferConfigType:SIZE	缓冲配置类型。 SIZE:按字节方式配置; TIME:按时间方式配置。
MaxBufferSize:10485760(bytes)	最大缓冲大小,字节为单位。
TimeOutOfNetwork:120000(ms)	网络超时时间。
BufferStart:524288(size-bytes/time-ms)	缓冲不够水限值,低于该值将上报BufferStart 事件。
BufferEnough:4194304(size-bytes/time-ms)	缓冲足够水限值,高于该值将上报 BufferEnough事件。
BufferFull:5242880(size-bytes/time-ms)	缓冲满水限值,达到该值将上报。BufferFull事件。
HlsStartMode:NORMAL	配置的HLS(Http live streaming)启动模式。 NORMAL: 默认从最高码率开始播放; FAST: 从最低码率开始播放。
DstSubCodeType:UTF-8	配置的字幕目标编码。
Cookie:NotSet	设置的Cookie信息。

参数	功能
UserAgent: NotSet	设置的UserAgent信息。
Referer: NotSet	设置的Referer信息。

2.33.5 查看 SDK 适配信息

在串口输入: cat /proc/hisi/hiplayer00/adapter

【调试信息】

```
cat /proc/hisi/hiplayer00/adapter
*****Hisilicon HiPlayer00 Player Adapter Info Begin*****
AdapterStatus: PLAY
bExtraAVPlay:1
AVPlayHandle:0x410000
AudTrackHandle: 0xffffffff
WindowHandle:0x0
SOHandle: 0x5edfd020
AudMixHeight:100
SoundPort: 0
Display: 0
Window-X:0(pix)
Window-Y:0(pix)
Window-w:0(pix)
Window-H:0 (pix)
VDecErrCover:0
VidPtsOffset:0(ms)
AudPtsOffset:0(ms)
SubPtsOffset:0(ms)
UseFFmpegAudCodec:1
PlaySpeed:0(x)
bVidStart:1
bAudStart:1
CurProgramId:0
CurVideoId:0
CurAudioId:0
CurSubtitleId:0
AudFrameSendLen: 0 (bytes)
LastPgsSubPts:0(ms)
EndOfStream:0
*****Hisilicon HiPlayer00 Player Adapter Info Begin*****
```

【调试信息说明】

查看HiPlayer使用的底层播放模块信息。

参数	功能
AdapterStatus:PLAY	当前AVPlay状态。

参数	功能
bExtraAVPlay:1	是否为外部创建的AVPlay。
	1: 外部创建传递给HiPlayer使用;
	0: HiPlayer自己创建。
AVPlayHandle:0x410000	AVPlay句柄值。
AudTrackHandle:0xffffffff	音频Track句柄值。
WindowHandle:0x0	视频窗口Window句柄值。
SOHandle:0x5edfd020	字幕输出SO句柄值。
AudMixHeight:100	设置的MixHeight值。
SoundPort:0	音频port值。
Display:0	display id值。
Window-X:0(pix)	视频窗口位置X坐标。
Window-Y:0(pix)	视频窗口位置Y坐标。
Window-W:0(pix)	视频窗口宽度。
Window-H:0(pix)	视频窗口高度。
	说明:XYWH四个值都为0表示全屏。
VdecErrCover:0	设置的视频输出错误门限。
VidPtsOffset:0(ms)	设置的视频pts偏移。
AudPtsOffset:0(ms)	设置的音频pts偏移。
SubPtsOffset:0(ms)	设置的字幕pts偏移。
bUseFFmpegAudCodec:1	是否使用ffmpeg软解码。
PlaySpeed:0(x)	播放倍速,(倍速范围: 2/4/8/16/32)。
bVidStart:1	1: 正常启动了视频通道;
	0: 视频通道启动失败,值为0,视频将无法播放。
bAudStart:1	^^
oAudstart.1	1:
	放。
CurProgramId:0	当前节目ID。
CurVideoId:0	当前播放的视频ID。
CurAudioId:0	当前播放的音频ID。
CurSubtitleId:0	当前播放的字幕ID。
AudFrameSendLen:0(bytes)	当前的音频帧已送给AVPlay的字节数。

参数	功能
LastPgsSubPts:0(ms)	此信息无用,请忽略。
EndOfStream:0	AVPlay是否上报了EndOfStream标示。
	1: AVPlay内部缓冲已播放完;
	0: AVPlay内部缓冲还未播放完。

2.33.6 Hiplayer 静态 proc 命令

信息保存在/proc/hisi/hiplayer/common节点下,此节点在以下两种情况下才会存在:

- hiplayer播放过程中。
- hiplayer退出播放后,播放器进程没有退出。播放器进程退出后,common节点也会销毁。

只有hiplayer至少创建一次后才会有common节点。

查看信息命令:

```
cat /proc/hisi/hiplayer/common
Current debug log level: -1
echo SetLogLevel = -1|0|1|2|3|4|5|6 >/proc/hisi/hiplayer/common
Log level (-1|0|1|2|3|4|5|6 = Disable this|QUIET|FATAL|ERROR|WARNING|
INFO|DEBUG|VERBOSE)
echo hiplayerstreampath = dir >/proc/hisi/hiplayer/common
echo enabledumpstream = 1|0 >/proc/hisi/hiplayer/common
```

参数	功能
Current debug log level: -1	显示当前log设置级别
echo SetLogLevel = -1 0 1 2 3 4 5 6 >/proc/hisi/hiplayer/common	设置log级别 -1: 应用程序通过hiplayer接口 "HI_FORMAT_INVOKE_SET_LOG_LE VEL"设置的log级别有效。 0到5分别对应: QUIET, FATAL, ERROR, WARNING, INFO, DEBUG, VERBOSE 如果SetLogLevel设置0到5的值,则应用程序通过 HI_FORMAT_INVOKE_SET_LOG_LEVE L设置的LOG级别无效

参数	功能
echo hiplayerstreampath = /data/ mediadump >/proc/hisi/hiplayer/common	设置码流录制路径,dir为目录,默认目录为/data/mediadump。 此路径对echo DumpAVEs = 1 0 >/proc/hisi/hiplayer00/control命令也有效 果。
echo enabledumpstream = 1 0 >/proc/hisi/ hiplayer/common	开始/停止录制播放文件流,录制的码流 放在hiplayerstreampath命令设置的路径 下。

2.33.7 查看切台性能统计信息

起播后在串口输入: cat/proc/hisi/hiplayer00/switch-pg

【调试信息】

```
cat /proc/hisi/hiplayer00/switch-pg
Begin********
                  SWITCH-PG Info Begin Time (ms)
Demux Name: mov, mp4, m4a, 3gp
Protocol Name: md
Set DataSource: 4249079355
UNF AVPLAY Create: 4249079442
Do Prepare: 4249079371
SVR Format FindStream: 4249079449
Do avformat open input : 4249079449
Do init input: 4249079449
Do avio open h: 4249079449
avio open h Done: 4249079451
av probe input buffer: 4249079451
probe input buffer Done: 4249079593
open input done: 4249080161
estimate duration start: 4249080162
estimate duration done: 4249080162
FindStream Done: 4249080165
SVR FORMAT GetFileInfo: 4249080161
GetFileInfo Done: 4249080163
Prepare async Complete: 4249080167
Do Start Enter: 4249080289
PLAYER STATE change: 4249080316
First VFrame Sent: 4249080319
First VFrame Decoded: 4249080399
FIRST FRAME TIME: 4249080399
-----In Total-----
ExitPlay Consume: 0
Enter Play Consume: 1044
-----Details-----
Prepare Schedule:16
init input:144
iformat->read header:568
Find Stream Info:2
StartSchedule:122
Enter Play UNF Consume :27
```

Send1st VFrame:3
Decode1st VFrame:80
-------Supplement----Exit Play UNF Consume:0
Bytes of 1st VFrame:3081
Send1st AFrame:8
FF Read 1st VFrame:25
FF Read 1st VKFrame:26
FF Read 1st AFrame:27
1st VFramePts:33
1st VKeyFrame Pts:33
Download Bandwidth:89034417
SWITCH-PG Info End

【调试信息分析】

进入播放后,任何其它操作之前输入命令查看切台性能统计信息。如果起播过程中某条操作没有被执行,则对应的参数可能不会显示。

参数	功能
Demux Name: mov,mp4,m4a,3gp	解析器名称(相比2.33.4 节中的解析器概念而言更加具体),可看出文件格式类型。
Protocol Name: md	用于读取数据的协议的名称,比如http, file 等。
Set DataSource: 4249079355	调用HiMediaPlayer::setdatasource()接口的绝对时间,单位毫秒,仅Android版本有效。
UNF_AVPLAY_Create: 4249079442	开始创建AVPLAY时的绝对时间,单位毫秒, 仅Anroid版本有效。
Do Prepare: 4249079371	调用HiMediaPlayer::prepare()接口的绝对时间,单位毫秒,仅Android版本有效。
SVR_Format_FindStream: 4249079449	调用HiPlayer内部接口 SVR_Format_FindStream()的绝对时间,单位 毫秒。
Do avformat_open_input: 4249079449	调用内部接口avformat_open_input()时的绝对时间,单位毫秒。
Do init_input: 4249079449	调用内部接口init_input()时的绝对时间,单位 毫秒。
Do avio_open_h: 4249079449	调用内部接口avio_open_h()时的绝对时间,单位毫秒。
avio_open_h Done: 4249079451	调用内部接口avio_open_h()返回时的绝对时间,单位毫秒。
av_probe_input_buffer: 4249079451	调用内部接口av_probe_input_buffer ()时的绝对时间,单位毫秒。
probe_input_buffer Done:: 4249079593	调用内部接口av_probe_input_buffer ()返回时的绝对时间,单位毫秒。

参数	功能
open_input done : 4249080161	调用内部接口avformat_open_input()返回时的 绝对时间,单位毫秒。
estimate duration start:: 4249080162	开始计算文件duration时的绝对时间,单位毫秒。
estimate duration done: 4249080162	完成计算文件duration时的绝对时间,单位毫秒。
FindStream Done: 4249080165	解析文件信息接口返回时的绝对时间,单位毫秒。
SVR_FORMAT_GetFileInfo: 4249080161	开始获取文件信息时的绝对时间,单位毫秒。
GetFileInfo Done: 4249080163	完成获取文件信息时的绝对时间,单位毫秒。
Prepare async Complete: 4249080167	HiPlayer完成文件解析,可随时进入播放状态 时的绝对时间,单位毫秒
Do Start Enter: 4249080289	Anroid版本:上层应用调用start()接口的绝对时间,单位毫秒。
	Linux版本:上层应用调用HiPlayer HI_SVR_PLAYER_Play()接口的绝对时间,单 位毫秒。
PLAYER_STATE change: 4249080316	HiPlayer上报播放状态事件时的绝对时间,单 位毫秒,仅Android版本有效。
First VFrame Sent: 4249080319	起播后完成发送第1个视频帧给AVPLAY时的 绝对时间,单位毫秒。
First VFrame Decoded: 4249080399	起播后第1个视频帧解码完成时的绝对时间, 单位毫秒
FIRST_FRAME_TIME: 4249080399	HiMediaPlayer收到HiPlayer上报的第1个视频 帧已解码事件时的绝对时间,单位毫秒,仅 Android版本有效。
ExitPlay Consume: 0	上次播放退出时消耗的总时间,单位毫秒。此 项暂时不支持。
Enter Play Consume: 1044	本进入播放消耗的总时间,单位毫秒。
Prepare Schedule: 16	从调用HiMediaPlayer::Setdatasource()到调用HiMediaPlayer::prepare()之间消耗的时间,单位毫秒,仅Android版本有效。
init_input: 144	调用内部接口init_input()消耗时间,单位毫秒。
iformat->read_header: 568	调用内部接口read_header()消耗时间,单位 毫秒。

参数	功能
Find Stream Info: 2	调用内部接口findstream()消耗时间,单位 毫秒。
StartSchedule: 122	从HiPlayer完成文件解析后到上层应用调用 HiPlayer HI_SVR_PLAYER_Play()接口时消耗 的时间,单位毫秒。
Enter Play UNF Consume: 27	进入播放时AVPLAY消耗的总时间(不包括解码消耗时间),单位毫秒。
Send1st VFrame: 3	从HiPlayer进入播放状态后到完成发送第1个视频帧给AVPLAY时消耗的时间,单位毫秒。
Decode1st VFrame: 80	第1个视频帧解码完成消耗的时间,单位毫 秒。
Exit Play UNF Consume: 0	上次播放退出时AVPLAY消耗的总时间,单位 毫秒。此项暂时不支持。
Bytes of 1st VFrame: 3081	第1个视频关键帧的大小,单位字节。
Send1st AFrame: 8	从HiPlayer进入播放状态后到完成发送第1个音频帧给AVPLAY时消耗的时间,单位毫秒。
FF Read 1st VFrame: 25	从内部接口findstream()返回到读取到第1个视 频帧时消耗的时间,单位毫秒。
FF Read 1st VKFrame:26	从内部接口findstream()返回到读取到第1个视 频关键帧时消耗的时间,单位毫秒。
FF Read 1st AFrame:27	从内部接口findstream()返回到读取到第1个音 频帧时消耗的时间,单位毫秒。
1st VFramePts:33	第1个视频帧的PTS,单位毫秒。
1st VKeyFrame Pts:33	第1个视频关键帧的PTS,单位毫秒。如果第1 个视频帧就是关键帧,则"1st VFrame Pts" 和"1st VKeyFrame Pts"相同。
Download Bandwidth:89034417	下载速度,单位bps,仅网络播放有效。

2.33.8 查看 SEEK 性能统计信息

SEEK后在串口输入: cat/proc/hisi/hiplayer00/seek

【调试信息】

cat /proc/hisi/hiplayer00/seek

Begin*************

__Seek Info Begin Time(ms)_

SeekTo: 4251885281

Do async seek: 4251885281 Do seek pts: 4251885284

```
PAdpt Seek Complete: 4251885351
Do AVPLAY Reset: 4251885284
Do AVPLAY ResetDone: 4251885329
Do SeekVidBefore: 4251885351
HI SVR FORMAT SeekPts: 4251885351
Do CMD SEEK: 4251885356
Do Read Seek: 4251885566
Read Seek Done: 4251885567
Thread SeekPts 1 Done: 4251885394
Thread AVSyncDone: 4251885566
Thread SeekPts 2 Done: 4251885568
Do seek complete: 4251885580
First VFrame Sent: 4251885591
First VFrame Decoded: 4251885642
-----In Total-----
Seek Consume: 361
-----Details-----
SeekSchedule 1: 3
PAdptSeek: 67
SeekSchedule 2: 0
FF Seek Schedule: 5
Thread SeekPts 1: 38
Thread AVSync: 172
Thread SeekPts 2: 2
Complete Seek: 12
Send1st VFrame: 11
Decode1st VFrame: 51
-----Supplement-----
Seek UNF Consume: 45
Bytes of 1st VFrame: 71745
Send1st AFrame: 13
FF Read 1st VFrame: 1
FF Read 1st VKFrame: 1
FF Read 1st AFrame: 2
1st VFramePts: 408642
1st VKeyFrame Pts: 408642
FF Read Seek: 1
ff seek frame binary: 0
seek frame generic: 0
Download Bandwidth: 94131622
                          Seek Info End
```

【调试信息分析】

SEEK后,任何其它操作之前输入命令查看SEEK性能统计信息。如果SEEK过程中某条操作没有被执行,则对应的参数可能不会显示。

参数	功能
SeekTo: 4251885281	调用HiMediaPlayer::seekto()时的绝对时间,单位毫秒。仅Android版本有效。
Do async seek: 4251885281	调用HiPlayer HI_SVR_PLAYER_Seek()时的绝对时间,单位毫秒。
Do seek pts: 4251885284	HiPlayer开始执行SEEK时的绝对时间,单位毫秒。

参数	功能
PAdpt Seek Complete: 4251885351	AVPLAY SEEK完成时的绝对时间,单位毫秒。
Do AVPLAY Reset: 4251885284	开始执行AVPLAY RESET时的绝对时间,单位毫秒。
Do AVPLAY ResetDone: 4251885329	完成执行AVPLAY RESET时的绝对时间,单位毫秒。
Do SeekVidBefore: 4251885351	开始执行SEEK视频流时的绝对时间,单位毫秒。
HI_SVR_FORMAT_SeekPts: 4251885351	调用内部接口HI_SVR_FORMAT_SeekPts()时的绝对时间,单位毫秒。
Do CMD_SEEK: 4251885356	FFMPEG收到SEEK命令时的绝对时间,单位 毫秒。仅网络播放有效。
Do Read_Seek: 4251885566	开始执行解析器SEEK接口时的绝对时间,单 位毫秒。
Read_Seek Done: 4251885567	完成执行解析器SEEK接口时的绝对时间,单 位毫秒。
ff_seek_frame_binary: 4251885567	开始调用内部接口ff_seek_frame_binary()时的 绝对时间,单位毫秒。
seek_frame_binary Done: 4251885567	完成调用内部ff_seek_frame_binary()时的绝对时间,单位毫秒。
Do seek_frame_generic: 4251885567	开始调用内部接口seek_frame_generic时的绝对时间,单位毫秒。
seek_frame_generic Done: 4251885567	完成调用内部接口seek_frame_generic时的绝对时间,单位毫秒。
Thread_SeekPts 1 Done: 4251885394	第1次调用内部接口Thread_SeekPts返回时的绝对时间,单位毫秒。仅网络播放有效。
Thread_AVSyncDone: 4251885566	调用内部接口Thread_AVSync()返回时的绝对时间,单位毫秒。仅网络播放有效。
Thread_SeekPts 2 Done: 4251885568	第2次调用内部接口Thread_SeekPts返回时的绝对时间,单位毫秒。仅网络播放有效。
Do seek complete: 4251885580	HiPlayer完成执行SEEK时的绝对时间,单位毫秒(此时第1个视频帧尚未显示)
First VFrame Sent: 4251885591	完成发送第1个视频帧给AVPLAY时的绝对时间,单位毫秒。
First VFrame Decoded: 4251885642	第1个视频帧解码完成时的绝对时间,单位毫秒。
Seek Consume: 361	SEEK时HiPlayer消耗的总时间,单位毫秒。

参数	功能
SeekSchedule 1: 3	Android版本:从调用HiMediaPlayer::seekto() 到HiPlayer开始实际执行SEEK时消耗的时间, 单位毫秒。
	Linux版本:从调用HiPlayer HI_SVR_PLAYER_Seek()接口到HiPlayer开始 实际执行SEEK时消耗的时间,单位毫秒。
PAdptSeek: 67	调用HiPlayer内部接口进行AVPLAY Seek消耗的时间,单位毫秒。
SeekSchedule 2: 0	从AVPLAY SEEK完成到开始调用FFMPEG SEEK接口时消耗的时间,单位毫秒。
FF Seek Schedule: 5	FFMPEG发送和接收SEEK消息所消耗的时间, 单位毫秒,仅网络播放有效。
Thread_SeekPts 1: 38	第1次调用内部接口Thread_SeekPts进行SEEK时消耗的时间,单位毫秒。仅网络播放有效。
Thread_AVSync: 172	调用内部接口Thread_AVSync()进行音视频同步时消耗的时间,单位毫秒。仅网络播放时部分解析器有效。
Thread_SeekPts 2: 2	第2次调用内部接口Thread_SeekPts进行SEEK时消耗的时间,单位毫秒。仅网络播放有效。
Thread_SeekPts: 2	调用内部接口Thread_SeekPts消耗的时间,单位毫秒。仅本地播放有效。
Complete Seek: 12	第2次调用内部接口Thread_SeekPts返回时到 HiPlayer Seek完成时消耗的时间,单位毫秒。仅 网络播放有效。
Send1st VFrame: 11	从HiPlayer完成SEEK时到完成发送第1个视频 帧给AVPLAY时消耗的时间,单位毫秒。
Decode1st VFrame: 51	第1个视频帧解码完成消耗的时间,单位毫秒。
Seek UNF Consume: 45	SEEK流程中AVPLAY消耗的总时间,单位毫秒。
Bytes of 1st VFrame: 71745	Seek后第1个视频帧大小,单位字节。
Send1st AFrame: 13	从HiPlayer 完成SEEK时到完成发送第1个音频 帧给AVPLAY时消耗的时间,单位毫秒。
FF Read 1st VFrame: 1	从HiPlayer 完成SEEK时到FFMPEG读取到第1 个视频帧时消耗的时间,单位毫秒。
FF Read 1st VKFrame: 1	从HiPlayer 完成SEEK时到FFMPEG读取到第1 个视频关键帧时消耗的时间,单位毫秒。
FF Read 1st AFrame: 2	从HiPlayer 完成SEEK时到FFMPEG读取到第1个音频帧时消耗的时间,单位毫秒。

参数	功能
1st VFramePts: 408642	SEEK后第1个视频帧的PTS,单位毫秒。
1st VKeyFrame Pts: 408642	SEEK后第1个视频关键帧的PTS,单位毫秒。 如果第1个视频帧就是关键帧,则"1st VFrame Pts"和"1st VKeyFrame Pts"相同。
FF Read_Seek: 1	FFMPEG解析器SEEK时消耗的时间,单位毫秒。
ff_seek_frame_binary: 0	调用内部接口ff_seek_frame_binary()消耗的时间,单位毫秒。
seek_frame_generic: 0	调用内部接口seek_frame_generic()消耗的时间,单位毫秒。
Download Bandwidth: 94131622	下载速度,单位bps,仅网络播放时有效。

3 常见问题定位方法

3.1 DEMUX 问题定位方法

3.1.1 调试手段介绍

DMX提供了如下调试命令。

修改保存文件默认路径的命令:

echo storepath=路径 > /proc/msp/log

例如:

echo storepath =/home > /proc/msp/log

保存 ES 流的命令

开始:

echo save es start > /proc/msp/demux main

- 音频数据保存到dmx aud x.es中,x从0开始,每保存一次加1。
- 视频数据保存到dmx_vid_x.es中,x从0开始,每保存一次加1。

停止:

echo save es stop > /proc/msp/demux main

通过保存es文件,可以定位播放问题是DMX问题还是前端问题。

如:视频出现马赛克或音频出现卡顿时,使用该命令将es流保存下来,并将保存下来的文件通过sample/sample_esplay或者PC上的es流播放工具(如elecard)播放。

- 如果也存在同样的问题,则说明该问题与播放、VO、音频等模块没有关系;
- 如果播放保存下来的es流正常,或播放现象不一致,则说明播放、VO、音频等模 块处理上有问题,需要进一步定位这些模块。

全码流录制命令

开始.

echo save allts start PortId > /proc/msp/demux main

PortId表示端口号,IF端口从0开始,TSI端口从32开始,RAM端口从128开始。 TS流保存到dmx allts x.ts中,x对应PortId。

停止:

echo save allts stop > /proc/msp/demux_main

全码流录制是指DMX把指定端口的TS包全部录制下来。通过全码流录制可以很好的定位信号源是否有问题。例如当出现section丢包时,通过全码流录制后,分析码流或者通过本地读文件方式收表也出现丢包,则说明TS流到DMX前就已经出现丢包,需要继续定位是QAM/TUNER问题,还是线路问题。



注意

- 在进行端口选择的时候,需要注意。在调用命令之前需要先确认使用的是哪一个端口。如果是通过内置QAM接入的信号则端口为0;
- 全码流录制数据量较大,如果通过网络文件系统进行录制需要注意网络带宽是否满足数据量要求(速度: 5MByte/s以上),如果网络较慢,则建议使用移动硬盘进行保存。

保存 RAM 端口输入数据的命令

开始:

echo save ipts start > /proc/msp/demux main

RAM端口输入的数据保存到dmx ram x.ts中,x从0开始,每保存一次加1。

停止:

echo save ipts stop > /proc/msp/demux_main

在进行网络端口录制时,通过该命令保存的数据就是用户送入的数据,不做任何处理。该方法可以方便的定位是否为网络环境或网络接收(如socket接收)问题。

录制 DMX 上所有通道的 TS 流

开始:

echo save dmxts start DmxId > /proc/msp/demux_main

TS流保存到dmx rects x.ts中, x从0开始, 每保存一次加1。

停止:

echo save dmxts stop > /proc/msp/demux_main

该命令主要用来定位加密码流相关的问题。如果码流是加密的,通过该命令可以将dmx 上挂载的通道数据按清流录制下来。

查看 help 信息

可以通过如下命令查看help信息:

```
# echo help>/proc/msp/demux_main
echo save es start > /proc/msp/demux_main -- begin save es
echo save es stop > /proc/msp/demux_main -- stop save es
echo save allts start x[portid] > /proc/msp/demux_main -- begin save allts
echo save allts stop > /proc/msp/demux_main -- stop save allts
echo save ipts start x[ram portid] > /proc/msp/demux_main -- begin save ram port ts
echo save ipts stop > /proc/msp/demux_main -- stop save ram port ts
echo save dmxts start x[dmxid] > /proc/msp/demux_main -- begin save all channels ts
echo save dmxts stop > /proc/msp/demux_main -- stop save dmx ts
echo help > /proc/msp/demux_main -- stop save dmx ts
```

3.1.2 收不到数据相关问题定位

收不到数据相关问题,可以从如下几个方面进行定位:

- 查看端口是否有数据;
- 查看DMX是否正确设置;
- 通道是否有数据,如果有数据,应用程序是否进行获取;
- 通道数据未正确的释放,造成数据无法获取;
- 查看数据源中有没有对应的数据。

查看端口是否有数据

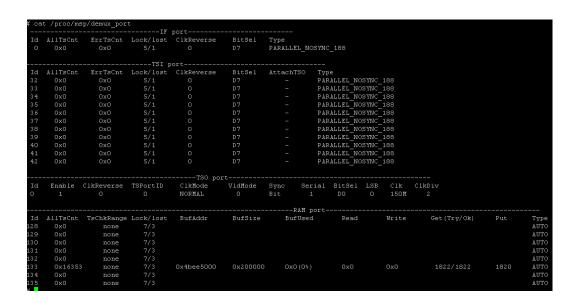
要查看端口是否有数据,首先要找到通道所对应的端口。一般情况下,使用内置QAM的话,端口号是0。其他情况可以通过cat /proc/msp/demux_chan先查看通道对应的DMX号,再通过cat /proc/msp/demux_main来查看DMX所对应的端口号。得到端口号后,可以通过连续运行 cat /proc/msp/demux_port来查看端口ts包有没有增加,以此判断端口有没有数据。如:

```
# cat /proc/msp/demux chan
DmxIdChnIdPIDType Mod StatKeyIdAcquire(Try/Ok)Release
800x1022VIDPLY OPEN--1977/14611461
810x1023AUDPLY OPEN--199/195195
```

通过该proc信息,可以看出通道0和1对应的DMX号为8。

```
# cat /proc/msp/demux_main
DmxIdPortId
0--
1 --
2--
3--
4--
5--
6--
7--
8133
9--
10--
11--
12--
13--
type "echo help > /proc/msp/demux_main" to get help information
```

通过该proc信息可以看出DMX8对应的端口号为133.



通过连续的查看demux_port信息,并通过比较AllTsCnt有没有增加,就可以知道端口有没有数据。

- 如果AllTsCnt没有增加说明端口没有数据 查找端口没有数据的原因:如:端口没有数据,如果是IF/TSI端口需要查看tuner是 否锁定。如果是RAM端口需要检查应用程序是否向对应的端口送数据。
- 有增加则说明有数据。如果端口有数据,则需要继续下面的步骤。

查看 DMX 是否正确设置

通过查看cat /proc/msp/demux chan,可以查看到通道的设置信息,如:

```
# cat /proc/msp/demux_chan
DmxIdChnIdPIDType Mod StatKeyIdAcquire(Try/Ok)Release
800x1022VIDPLY OPEN--1977/14611461
810x1023AUDPLY OPEN--199/195195
```

从上面信息可以看出通道0的类型为视频通道,PID为0x1022,当前状态为OPEN,没有绑定密钥区。如果通道类型、PID或者状态与自己预期不一样,则查找程序查看出错的原因。

查看通道 Buffer, 如果有数据, 检查应用程序是否进行获取

通过cat /proc/msp/demux_chanbuf,可以查看DMX通道有没有数据,如果buffer used为 0,且反复查看demux_chanbuf其读写指针没有变化,则说明该通道没有数据,否则则说明有数据。

```
# cat /proc/msp/demux_chanbuf
DmxIdChnIdFqIdOqIdSizeBlkCntBlkSizeReadWriteUsedOverflow
801016383K10081664445545399%0
8121255K204128019519399%0
```

可以看出通道0和1都有数据。在有数据的情况下,通过连续查看通道的proc可以判断应用程序有没有获取数据。如果Acquire的Try次数没有增加,则说明应用程序没有进行获取操作,需要检查应用程序。如下面proc信息:

cat /proc/msp/demux_chan

DmxIdChnIdPIDType Mod StatKeyIdAcquire(Try/Ok)Release 800x1022VIDPLY OPEN--1977/14611461 810x1023AUDPLY OPEN--199/195195 # cat /proc/msp/demux_chan DmxIdChnIdPIDType Mod StatKeyIdAcquire(Try/Ok)Release 800x1022VIDPLY OPEN--33844/15571557 810x1023AUDPLY OPEN--633/615614

从上面信息可以看出,两次查看proc,通道0的Acquire的Try次数从1977次,增长到33844次,则说明应用程序一直在获取数据。

通道数据未正确的释放。造成数据无法获取

DMX要求接口HI_UNF_DMX_AcquireBuf和HI_UNF_DMX_ReleaseBuf的调用必须成对出现,顺序获取顺序释放。如果查看demux_chanbuf发现通道buffer占用率在80%以上,而应用程序却获取不到数据,就很可能是因为没有正确的释放造成了通道buffer溢出。

通过比较demux_chan信息中获取成功的次数和释放的次数可以判断出有没有出现数据没有释放的情况。如果二者相差不大,如差1或者差2(视频通道可能会相差好几十),则说明获取释放正常。如相差较大,如相差好几百,那就是应用程序获取了数据,但没有释放,出现这种情况就需要检查并修改应用程序了。

cat /proc/msp/demux_chan
DmxIdChnIdPIDType Mod StatKeyIdAcquire(Try/Ok)Release
800x1022VIDPLY OPEN--1977/14611461
810x1023AUDPLY OPEN--199/195195

查看数据源中有没有对应的数据

该方法只能判断从QAM处理后数据有没有问题,不能排除信道的问题。可以通过DMX 全码流录制达到这一目的。使用方法参考上文全码流录制的使用方法。通过码流工具 分析录制后的码流有没有对应PID和过滤条件的数据。

||| 详明

DMX全码流录制,不经过DMX模块处理,其是将端口数据透传到内存中实现录制功能,所以其结果是可信的。

声音或图像播不出的问题,也可以通过该方法进行问题的查找。

3.1.3 播放视频马赛克或者声音卡顿问题定位(包括加扰流)

播放视频马赛克问题需要区分出到底是后端解码出错,还是前端数据源出错。通过 "3.1.1 调试手段介绍3.1.1 调试手段介绍"的"保存es命令"将音视频es流保存下来, 然后将es流放在电脑上或者通过esplay进行播放。如果上述方法播放es流没有问题,则 应该是解码模块或者VO部分出了问题,需要继续查找这些模块进行定位。如果es流本身播放也有问题,则按前面讲的"数据出错或丢数据问题定位"方法继续定位。

如果是加扰流,还可以通过"3.1.1 调试手段介绍3.1.1 调试手段介绍的""*保存某DMX 创建的所以通道对应的 is 流*"将加扰流录制成清流进行进一步的定位。是否为DMX收数据收错了。但这种可能性非常小。加扰流的情况,因为涉及到ca解扰所以问题较复杂,更加稳妥的办法还是通过也其他厂商的盒子进行对比,以确定是否为前端数据源问题。

3.1.4 RAM 端口相关问题定位

RAM 端口获取不到 TS Buffer 的问题

在使用RAM端口时,如果RAM端口对应的DMX上创建了音视频通道,则如果音视频通道不读数据,会造成RAM端口反压。如果出现RAM端口一直获取不到TS Buffer,很有可能是这种情况。

通过cat /proc/msp/demux_chanbuf ,可以查看DMX通道有没有数据,如果某音视频通道的buffer used在80%以上,则是造成了反压,需要分析应用程序查找出现buffer满的原因。而通过连续查看demux chan可以看出该通道是否为音视频通道,软件有没有获取等信息。

另外通过查看demux_port可以查看RAM 端口ts buffer大小、使用的buffer大小、上层应用有没有获取ts buffer等信息。

RAM 端口数据错误的问题

RAM端口相关问题需要判断数据送入RAM端口前是否有问题。通过"3.1.1 调试手段介绍3.1.1 调试手段介绍的"可以将用户送入RAM端口的数据保存下来,通过电脑上播放该录制码流或者使用工具分析该录制码流可以很方便的区分出是否数据到RAM端口前就已经出错了。通过该方法可以很方便的查找IPTV、VOD等应用相关的问题。

3.2 VDEC 问题定位方法

3.2.1 VDEC 模块介绍

VDEC实现对Vfmw的一个封装,Vfmw是操作视频解码器的核心软件模块,解码器上连DMX,下通VPSS或者VO,是视频信息的必经之地,相当多的视频播放问题在此都会留下影子。

Vfmw内部主要包括SCD、VDH各模块的功能如下:

- SCD模块实现对送入Vfmw的ES流的预处理,然后送给VDH模块。
- VDH模块实现对视频解码,解码后的数据是一维的,此时还无法播放。

这两个模块任何一个出问题,都会导致解码器不能正常工作,通过查看/proc/vfmw,/proc/vfmw dec,/proc/vfmw dbg,/proc/vfmw scd能定位大部分解码器问题。

VDEC的总体状态可以通过 cat/proc/vdec00查看。最后两位代表通道号。

cat /proc/msp/vdec00

显示如下:

----- VDEC0 ------

Work State: RUN VpssID: vpss00 VfmwID: vfmw00 Codec ID: H264(0x4)

Mode: NORMAL

```
Priority: 3
ErrCover: 100
OrderOutput: 0
CtrlOption*: 0x0
Capability: NORMAL/2160P(2)/H264
Dynamic Frame Store: Enable
  ------ Information-----
Dynamic Frame Store Mode: Self
DFS use MMZ: 1
DFS config Frame Number: 9
DFS ExtraFrame Number: 6
DFS DelayTime (ms): 0
DFS Max Mem Use(byte): 0xfa00000
DFS Memory PhyAddress: 0x0
DFS Memory Length (byte): 0
 -----Stream Information-----
Source: User0
StreamSize (Total/Current): 0x758959b0/0xe96ac4
BitRate(bps): 7388984
LumaBitdepth: 8
ChromaBitdepth: 8
StreamBufferSize(Total/Used/Persent): 0x1000000/0xe96ac4/91%
       -----Picture Information----
Width*Height: 1920*1080
Stride (Y/C): 0x800/0x800
FrameRate(fps): Real(24.99) FrameInfo(25000)
PlayFormat: OTHER(273)
FrmPackingType: 2D
Aspect (User/Decode): 0:0/0:2
FieldMode: Frame
Type: Interlace
YUVFormat: SP420
TopFirst: TRUE
ErrFrame: 0
TypeNum(I/P): 53362/0
VP6 Picture Reversed: NO
DMX/USER->VDEC
GetStreamBuffer(Try/OK): 331891/120372
PutStreamBuffer(Try/OK): 120372/120372
VDEC->VFMW
AcquireStream(Try/OK): 121607/120372
ReleaseStream(Try/OK): 119438/119438
VFMW->VPSS
AcquireFrame (Try/OK): 53387/53362
ReleaseFrame(Try/OK): 53359/53359
VPSS->AVPLAY
AcquireFrame(Try/OK): 317331/106672
ReleaseFrame (Try/OK): 213342/213342
```

- Work State表示解码通道当前的运行状态; VpssID表示对应的vpss通道; VfmwID表示对应的vfmw通道; Codec ID表示当前通道的解码的视频协议类型; ErrCover代表解码输出的错误阈值,为100所有码流全部输出,为0表示只要码流有错误则不输出; Capability代表通道的能力级;
- Stream Information一栏的信息表示码流一些状态信息。StreamSize(Total/Current)表示当前vdec收到的总码流buffer的大小和当前buffer中未解码的码流的大小; LumaBitdepth和ChromaBitdepth分别表示码流亮度和色度的bit深度,8bit是现实为8,10bit时显示为10; StreamBufferSize(Total/Used/Persent)表示码流buffer的占用情况,Total是vdec码流buffer的大小,Used代表码流buffer已经被占用了多少,Persent是码流buffer的占用率。

- Picture Information一栏的信息表示当前码流的一些解码信息。Width*Height表示分辨率信息;FrameRate(fps)表示帧率信息,Real(24.99)表示的是实际统计值,FrameInfo(25000)代表的是从码流信息中获取到的帧率信息。TopFirst代表是不是顶场优先,True代表顶场优先,False代表底场优先;TypeNum(I/P)代表逐行帧和隔行帧的个数;
- DMX/USER->VDEC表示前级模块从vdec获取码流buffer和释放码流buffer的情况。
- VDEC->VFMW表示VFMW从VDEC获取和释放原始码流数据的情况。
- VFMW->VPSS表示VPSS从VDEC获取和释放解码后帧数据的情况。
- VPSS->AVPLAY表示avplay通过VDEC从VPSS获取和释放帧数据的情况。

3.2.2 用 echo 动态调整 vfmw 的行为

用如下命令可以查看vfmw的全局信息和调试选项: cat /proc/vfmw dbg 显示如下: -----debug options----tracer address: 0x835ee000 path to save debug data:/mnt print enable word:0x3 vfmw state enable word :0xffffffff bitstream control period: 200 frame control period:0 rcv/rls img control period:500 you can perform vfmw debug with such commond: echo [arg1] [arg2] > /proc/vfmw debug action arg1 arg2 set print enable 0x0 print enable word set err thr 0x2 (chan id << 24)|err thr set dec order output 0x4 (chan_id<<24)|dec_order_output_enable set dec mode(0/1/2=IPB/IP/I) 0x5 (chan id<<24)|dec mode set discard before dec thr 0x7 (chan id<<24)|stream size thr set postprocess options 0xa (dc<<8)|(db<<4)|dr, 0000=auto,0001=on,0010=off set frame/adaptive storage 0xb 0:frame only, 1:adaptive pay attention to the channel 0xd channel number channel vcmp config 0xe chanId: arg2>>27,

```
mirror en: (arg2>>26)&1,
vcmp_en: (arg2>>25)&1,
wm_en: (arg2>>24)&1,
wm start: (arg2>>12)&0xfff,
wm end: (arg2)&0xfff
print tracer 0x100 tracer address. do not care if vfmw still running
start/stop raw stream saving 0x200 chan id
start/stop stream seg saving 0x201 chan id
start/stop 2D yuv saving 0x202 chan id
save a single 2D frame 0x203 frame phy addr
save a single 1D frame 0x204 frame phy addr width height=(height+PicStructure)
set dec_task_schedule_delay 0x400 schedual_delay_time(ms)
set dnr_active_interval 0x401 dnr_active_interval(ms)
stop/start syntax dec 0x402 do not care
set trace controller 0x500 vfmw state word in /proc/vfmw prn
set bitstream control period 0x501 period (ms)
set frame control period 0x502 period (ms)
set rcv/rls img control period 0x503 period (ms)
其中:
debug action arg1 arg2
    debug action: 你要进行何种调试操作
    arg1: 调试操作的命令编码
    arg2: 调试操作的参数
cat /proc/vfmw_prn
'print enable word' definition, from bit31 to bit0:
<not used> DEC_MODE PTS DNR
FOD SCD_INFO SCD_STREAM SCD_REGMSG
BLOCK DBG SE SEI
SLICE PIC SEQ MARK MMCO
POC DPB REF QUEUE
IMAGE STR BODY STR TAIL STR HEAD
```

打开有关 PTS 的打印(打印到串口)

举几个例子:

使能print的命令是0,打印内容由一个32bit数字决定,各位定义如下:

'print_enabl	e_word' defi	nition, from	bit31 to bit0:
<not used=""></not>	DEC_MODE	PTS	DNR
FOD	SCD_INFO	SCD_STREAM	SCD_REGMSG
BLOCK	DBG	SE	SEI
SLICE	PIC	SEQ	MARK MMCO
POC	DPB	REF	QUEUE
IMAGE	STR_BODY	STR_TAIL	STR_HEAD
STREAM	RPMSG	UPMSG	DNMSG
VDMREG	CTRL	ERROR	FATAL

例如,使能打印PTS的命令字是: 0x20000000,则使用以下命令则可以打印PTS。

echo 0 0x20000000 > /proc/vfmw

启动解码前存码流和 YUV

如果发现播放的画面有花块,到底是网络到DMX通路有问题,还是解码器有问题?可以将送到解码器的码流存下来进行分析,如果码流没问题则是解码器有问题,否则问题出在解码器之前的模块。目前解码模块有VDEC和VFMW两个模块均有录制码流的功能。当怀疑解码有问题的时候可分别使用VDEC和VFMW的录制码流功能。

启动解码(VDEC)的存储码流的命令编码是savestream, 它对应的参数arg2代表通道编号:

echo savestream 0 > /proc/msp/vdec_ctrl

启动解码(VFMW)前存码流的命令'start/stop raw stream saving'的命令编码是0x200,它对应的参数arg2代表通道编号。如:要存通道0的码流,则可使用如下的命令:

echo 0x200 0 > /proc/vfmw

码流默认存储于/mnt/目录下,文件名为vfmw_raw_save_NUM.dat, NUM会根据存储文件的次数从0递增。您也可以更改存储路径,以下命令将存储文件到/home目录下:

echo $0x200 \ 0$ /home > /proc/vfmw

您可以用sample/esplay或VLC等第三方工具测试存储流。

可以通过直接存储YUV的方式保存VDEC输出的数据,命令编码分别为: saveyuv和 saveoneyuv。 saveyuv命令会连续存储解码数据。saveoneyuv只存储一帧解码数据。默 认存储于/mnt/目录下,您也可以更改存储路径,以下命令将通道0的解码数据存储文件到/home目录下:

echo saveyuv 0 /home > /proc/msp/vdec_ctrl

echo saveoneyuv 0 /home > /proc/msp/vdec ctrl

单帧播放

当需要慢放视频以分析显示问题时,可以调整解码器中VDH的调度时间间隔,如:让VDH每隔1秒钟调度1次,则解码器只能1秒钟解码1帧。

echo 0x401 1000 > /proc/vfmw

如果想回到正常播放模式,取消调度间隔,则命令如下:

echo 0x401 0 > /proc/vfmw

3.2.3 播放停滞定位

通过看vfmw的内部信息,可以定位一些问题。比如,播放停滞不动一般三个原因:

- 码流不足
- 解码器性能(或者调度)有问题
- 显示(包括同步)有问题

看 vfmw 的码流

如果码流供应不足,或者有丢失,则肯定会造成播放停滞。通过以下方法可以判断是 否是这个原因,命令如下:

cat /proc/vfmw scd

显示如下:

==========scd info ==============IsScdDrvOpen:1
SCDState:0 ThisInstID:0 LastProcessTime:169427729 HwMemAddr:0x8729f000
HwMemSize:131072 DownMsgMemAddr:0x8729f000 UpMsgMemAddr:
0x872a2000------inst[0] ------Mode:1 cfg VidStd:0 is wait seg ext:0 cfg BufPhyAddr:0x89d5c7ff cfg BufSize:2129920raw Total size:5224476raw Total num:319seg Total size:1840411 seg Total num:
143actual bitrate:51349 Kbps

其中,着重看raw 包数和seg包数:

- raw Total size: 未经过SCD模块处理的码流,字节数;
- raw Total num: 未经过SCD模块处理的码流包数;
- seg Total size: 经过SCD处理的码流字节数;
- seg Total num: 经过SCD处理的码流包数。

要结合场景和一次送包的大小看这两个包的数值:

本地播放时,两个数值可能都比较大,码流供应充足时字节数可达到100KB级别;

直播时raw包值可能会小一点,但如果seg包比较大,如字节数达到10KB级别,也是正常的;

如果总的包数和字节数都很小,如包数为个位数,字节数只有KB级别时,要分析码流供应问题:

- 如果总包数和总字节数很小, actual bitrate 也很小,则码流供应不足,检查DMX 或者信道是否有问题。
- 如果总包数和总字节数很小,actual bitrate很大,则表示VDH模块在丢帧或者码流在解析的过程中被强制释放了,因为这些视频帧都经过了SCD,因此actual bitrate很大,但是VDH在不停的丢帧或者码流在进入VDH解码前被不断释放,造成SCD的帧存buffer数据量很少。

这个时候就需要判断, VDH为什么会丢帧:

- 通常如果解码格式设置不对,会大量丢帧。 例如码流格式是H264,而解码格式是MPEG2,则会大量丢帧。
- VDH本身出了问题,也会大量丢帧。

看解码器的输出帧存

如果是VO或者VPSS取帧慢的原因造成播放停滞,那么解码器一定反压,表现为解码器的输出帧存占满。

```
cat /proc/vfmw dec
======= DEC INFO Chan [0] =========
Dec by VDH:-1(NA)
IsOpen:1
IsRun:1
Priority:3
eVidStd:H264
ChanCapLevel:18
LastFrameState: BLANK
IsOmxPath:0
DecMode: 0
DecOrderOutput:0
ErrorThred:100
StreamThred: 0
ChanMem VDH Mem Phy: 0x0
ChanMem VDH Mem Size:0
ChanMem SCD Mem Phy:0x4dc91000
ChanMem SCD Mem Size:31539200
ChanMem CTX Mem Phy: 0x4faa5000
ChanMem CTX Mem Size: 6026584
Ref, Read, New: 4, 3, 7
VoQue Detail: (47,50,57)
Actual DecMode: 0
Actual DecOrderOutput:0
Image Format:0x15fa2
Frame Packing Type:0
----- FSP INFO ------
FSP DecFsWidth:1920
FSP DecFsHeight:1088
FSP DispFsWidth:1920
FSP DispFsHeight:1088
FSP ExpectedDecFsNum:5
FSP TotalValidFSNum:15
FSP ActualPmvNum:15
FSP ActualDecFsNum:15
03 03 01 03 03 03 03 03 12 02 03 03 03 12 12
```

```
FSP LogicFsNum:64
03 03 01 03 03 03 03 03 12 02 03 03 03 12 12 00
-----Dynamic Frame Store-----
Frame Size(byte): 3891744
Max Time(ms): 30
Reference Frame Number: 5
Pre-alloced Frame Number/Time(ms): 0/0
Dynamic alloced Frame Number/Time(ms): 15/23
All Frame Number/Time(ms): 15/30
----- VDH INFO -----
VDH 0:17.2%
Frame Rate: 24.9 fps (Chan 0)
Cycle State: 234/mb
VDH 1:0.0%
Frame Rate: 0.0 fps (Chan 1)
Cycle State: 0/mb
```

通过观察: 03 03 01 03 03 03 03 03 13 02 03 03 13 13的变化来确认解码buffer的占用情况。

- 第一位为0表示当前帧不被参考,为1表示当前帧被参考。
- 第二位为3表示buffer已被占用。1表示buffer正在被解码器使用。0表示buffer空闲。

当播放出现卡顿的时候主要关注第二位是不是有很多为3的状态,如果有很多为3的状态说明可能是解码后级没有来及时取帧导致。如果有很多位0的状态说明可能是解码器未及时解码导致。

3.3 显示通道问题定位方法

3.3.1 显示通道 PROC 信息分析

查看显示通道状态的命令为: cat /proc/msp/dispX:

```
# cat /proc/msp/disp1
------Bisilicon DISP 1 State------
State:Open
Formt/DispMode:1080i50/2D
RightEyeFirst:Enable
VirtualScreen:1280/720
Offset:0/0/0/0
AspectRatioMode: Auto
AspectRatio:16:9
ColorSpace:BT709 YUV LIMITED->BT709 YUV LIMITED
Bright:50
Contrast:50
Saturation:50
Background (R/G/B):0x0/0x0/0x0
Zorder(Bot->Top):Video->Gfx
AttachRole:Source
AttachDisp:Disp0
Interface:YPbPr0(1/2/0)HDMI0
InitCount:2
OpenCnt[User/Kerne 1]:1/1
```

LowbandCount: 0

-----CAST Info-----

State:Enable
Crop:FALSE
CropRect:0/0/0/0
Resolution:1280/720
PixelFormat:NV21
FrameRate:30
LowDelay:Disable
MemoryType:DispAlloc
BufferNumber:6

BufferWidth/Height:1280/720

BufferSize:1382400 BufferStride:1280

-----Buffer-----

Acquire(Try/OK):1280/1024 Release(Try/OK):1022/1022

BufferQueue[state, FrameID]

(State: 1, Empty[0]; 2, Write[1]; 3, Full[3]; 4, Use[2])

[3,0x03fe] [3,0x03ff] [3,0x0400] [2,0x0000] [4,0x03fc] [4,0x03fd]

详细参数说明参考2.7 节DISPLAY, disp0表示DISPLAY0, disp1表示DISPLAY1。

Bright、Contrast、Saturation这三个参数很重要,如果**Contrast**为0则看不到图像输出。

3.3.2 常见显示问题

使用分量输出连接电视,电视机无画面

遇到此类问题,首先选择电视机的"显示信息"看一下,确认是电视机没有检测到输入(会显示"无信号输入"等字样),还是已经检测到输入(会显示"1080i50"等输出格式信息),只不过显示的是黑色画面。

步骤1 如果没有检测到输入,需要检查线路是否没有接好;如果检测到输入,先利用proc信息查看当前显示通道状态是否正常。

步骤2 确认电视机是否支持高清的输出制式,比如,较早的电视不能支持1080P50/P60制式。

步骤3 确认分量信号连接是否正确。

只有将发送端的Y/PbPr三个信号准确与接收端的Y/PbPr准确连接,电视机才能够正常显示。其中,Y信号中带有同步信号,如果Y信号没有正确对接,电视将因为没有检测到同步信号而不工作。建议依次将电视机Y分量接收端与发送端的三个接头都连接一遍,如果某个连接关系能够输出画面,说明此时连接的接头是Y分量输出接头,然后分析分量信号设置错误的原因。

此外,如果是VGA输出,则需要检查INTF是否已经绑定RGB接口,VDAC序号是否正确。

----结束

使用分量输出, 电视机有画面显示, 但是颜色不正常

交换输出端的Pb/Pr接头,如果颜色恢复正常,说明两个色度分量信号(Pb/Pr)连接错误,用户需检查DAC的配置顺序与实际是否相符。

使用 VGA 输出、接收端无显示

建议首先采用800*600分辨率,这是绝大部分显示器和电视都能够支持的分辨率。如果发现上述配置无误,显示设备仍然无响应,此时要检查管脚复用设置是否正确。

视频/图形输出过程中, 出现短暂黑屏, 然后恢复

请检查黑屏前后输出制式是否发生变化,比如从1080i50切换到1080P50,如果存在切换,则黑屏为正常现象。

同源显示模式下,相对于高清画面,标清电视显示的画面边缘部分不完整

这是标清电视机的处理引起的,非问题。

3.4 HDMI 问题定位方法

3.4.1 proc 信息分析

【调试信息】

HDMI正常输出时的打印信息。

```
# cat /proc/msp/hdmi0
SDK VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
       ----- APPAttr -----
HDMIEnable : YES VideoEnable : YES
                                   DefaultAction : NONE
                                   AudioEnable : YES
AviInfoEnable : YES
                                    AudioInfoEnable: YES
xvYCCMode
              : NO
                                   HDCPEnable : NO
                                   SpdInfoEnable : NO
DeepColorMode : 24
OutColorSpace : YES
                                   MpegInfoEnable : NO
                                   DebugEnable : NO
ColorSpaceAdapt: YES
                                   DrmInfoEnable : NO
CtsAuthEnable : NO
------ SWStatus ------
ThreadRun : YES
TMDSMode : DVI
                                   RunStatus
                                                : OPEN START
KernelCnt : 0
KCallBack : NO
TransitState : NONE
                                   UserCnt : 1
                                   UCallBackCnt : 0
                                  HDCPEnable : NO
HDCPMode : NONE
HDCP2.2Support : N/A
HDCPIsOn : NO KeyLoad : NO
                                   HdcpAuthStatus : NO
CECEnable : NO CECLogicAddr : 0
                                   CECStatus : OFF
CECNetwork : NON
                                                : NONE
CECPhyAddr : 00.00.00.00
------ HWStatus ------
HotPlug : YES
                                  Rsen : YES
PhyOutputEnable: YES
                                   PhyPowerEnable : YES
TMDSMode : HDMI1.4
                                  AvMute : NO
SourceScramble : NO
                                   SinkScramble : NO
TMDSBitClkRatio: 10
                                   DataSwing : 0x16
CLKSwing : 0x15
TxVnbValue : 0x2
TxRegZone : 0x82
                                  SrcTermination: 0x0
                                   AdjustClk : 0x1
```

```
----- TaskID=1126 Event Pool[0] Status ------
CNT|ErrTotal|HPD|UnHPD|EdidFail|HdcpFail|HdcpSucc|RsenCon|RsenDis|HdcpUsr
10
Memory[WkFlg=0 |RdAble= 0| RdPtr=1 | WrPtr=1 ]:
# cat /proc/msp/hdmi0 vo
SDK VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
----- VideoAttr ------ AVIIfno ------
VideoTiming : 1920*1080p60 16:9 |AVIInfoEnable : YES
              : 1080P@60
DispFmt
                                     |CurrentFormat : 1920*1080p60
16:9 (VIC=16)
PixelClk : 148500 | VSIFormat : (HDMI_VIC= 0)
InBitDepth : 10 Bit | BarDataPresent : NONE
InColorSpace : YCbCr444 | ColorSpace : YCbCr444
Colorimetry : ITU-R BT.709 | Colorimetry : ITU-R BT.709
PicAspectRatio : 16:9 | PicAspectRatio : 16:9
PicAspectRatio : 16:9
ActAspectRatio : PICTURE
                                      |ActAspectRatio : PICTURE
PixelRepeat : 1 | PixelRepeat : No Repetition
YCCQuantization: LIMITED | YCCQuantization: LIMITED
RGBQuantization: DEFAULT | RGBQuantization: DEFAULT
ExtColorimetry : XV_YCC601 | ExtColorimetry : XV_YCC601
StereoMode : NONE | ItContentValid : NO
StereoMode : NONE
              : HPVP
HvSyncPol
                                      |HvSyncPol : HPVP
-----|ITContentType : GRAPHICS
                                     |PicScaling : UNKNOW
VideoMute : NO
OutBitDepth : 08
               : 08 Bit
                                      |ActFmtPresent : YES
OutColorSpace : YCbCr444
                                       |ScanInfo : NONE
YCbCr422_444 : NO
YCbCr421
                                       |AVIInfoRawData :
                                       | 82 02 0d 67 50 a8 00 10
YCbCr444 422 : NO
                                      | 00 00 00 00 00 00 00 00
YCbCr422_420 : NO
                                      1 00
RGB2YCbCr : NO
                                      |VSInfoRawData :
YCbCr2RGB : NO
Dither : NO
                                      | 00 00 00 00
----- HDRAttr -----
HdrMode : DISABLE
                                  HdrEotfType : SDR LUMIN
HdrMetaDataID : 0
                                       HdrColorimetry : NONE
DispPrim0_X : 0
DispPrim1_X : 0
DispPrim2_X : 0
                                       DispPrimO_Y : 0
                                        DispPrim2_Y : 0
WhitePri
WhitePoint_X : 0

MaxLumina
                                       DispPrim2_Y : 0
WhitePoint_Y : 0
MinLuminance : 0
MaxLuminance
               : 0
MaxLightLevel : 0
                                       AverLightLevel : 0
                       ----- DRMInfo -----
                                       EotfType : SDR_LUMIN
DRMInfoEnable : NO
MetadataID : 0
DRMInfoRawData: 00 00 00 00
# cat /proc/msp/hdmi0 ao
SDK VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
----- AudioAttr ------ AudioIfno ------
SoundIntf : I2S
CodeType : STREAM
ChannelCnt : 2_CH
                                       |AudioInfoEnable: YES
               : STREAM
                                  |CodeType : STREAM
|ChannelCnt : 2_CH
SampleFreq : 48000
                                  |SampleFreq : STR_HEADER
```

```
SampleDepth : 16
                                                                                    |SampleDepth : 16
                                                                                      |SampleSize : STR HEADER
DownSample : NO
 -----|DownMixInhibit: NO
                                                                                       |LevelShiftValue: 0
AudioEnable : YES
AudioMute
SoundIntf
                                  : NO
                                                                                        |LFEPlayBack : UNKNOW
                                  : I2S
                                                                                        |Channel/SpeakerAlloc: 0x00 (0)
ChannelCnt : 2_CH
SampleFreq : 48000
                                                                                       |AudioInfoRawData:
                                                                                   | 84 01 0a 70 01 00 00 00
SampleDepth : 16
                                                                                      | 00 00 00 00 00 00
DownSample : NO CTS : N/A
CTS
N
                               : N/A
                            : N/A
AudioFIFO
# cat /proc/msp/hdmi0 sink
SDK VERSION: [HiSTBLinuxV100R002] Build Time: [Dec 12 2015, 11:35:36]
HDMI Version: 2.0.0.20151212.0
 ----- EDIDRawData
0 \times 0 b, 0 \times 0 1, 0 \times 0 0, 0 \times 0 0, 0 \times 0 0,
/*0fH:*/ 0x02,0x18,0x01,0x03, 0x80,0x59,0x32,0x78, 0x0a,0xee,
0x91,0xa3, 0x54,0x4c,0x99,0x26,
/*1fH:*/ 0x0f,0x50,0x54,0xbd, 0xef,0x80,0x71,0x4f,
0x81,0xc0,0x81,0x00, 0x81,0x80,0x95,0x00,
/*2fH:*/ 0xa9,0xc0,0xb3,0x00, 0x01,0x01,0x08,0xe8,
0x00,0x30,0xf2,0x70, 0x5a,0x80,0xb0,0x58,
/*3fH:*/ 0x8a,0x00,0x50,0x1d, 0x74,0x00,0x00,0x1e, 0x02,0x3a,
0x80,0x18, 0x71,0x38,0x2d,0x40,
/*4fH:*/ 0x58,0x2c,0x45,0x00, 0x50,0x1d,0x74,0x00, 0x00,0x1e,
0x00,0x00, 0x00,0xfd,0x00,0x18,
/*5fH:*/ 0x4b,0x0f,0x87,0x3c, 0x00,0x0a,0x20,0x20,
0x20,0x20,0x20,0x20, 0x00,0x00,0x00,0xfc,
/*6fH:*/ 0x00,0x53,0x41,0x4d, 0x53,0x55,0x4e,0x47, 0x0a,
0x20,0x20,0x20, 0x20,0x20,0x01,0x22,
/*7fH:*/ 0x02,0x03,0x3a,0xf1, 0x57,0x61,0x10,0x1f,
0x04,0x13,0x05,0x14, 0x20,0x21,0x22,0x5d,
/*8fH:*/ 0x5e,0x5f,0x60,0x65, 0x66,0x62,0x63,0x64,
0x07,0x16,0x03,0x12, 0x23,0x09,0x07,0x07,
/*9fH:*/ 0x83,0x01,0x00,0x00, 0xe2,0x00,0x0f,0x6e, 0x03,0x0c,
0x00,0x10, 0x00,0xb8,0x3c,0x21,
/*afH:*/ 0x00,0x80,0x01,0x02, 0x03,0x04,0xe3,0x0f,
0x01,0xe0,0x01,0x1d, 0x80,0xd0,0x72,0x1c,
/*bfH:*/ 0x16,0x20,0x10,0x2c, 0x25,0x80,0x50,0x1d,
0x74,0x00,0x00,0x9e, 0x66,0x21,0x56,0xaa,
/*cfH:*/ 0x51,0x00,0x1e,0x30, 0x46,0x8f,0x33,0x00, 0x50,0x1d,
0x74,0x00, 0x00,0x1e,0x00,0x00,
0 \times 00, 0 \times 
 0 \times 00, 0 \times 49,
 ----- SWStatus
CapFromSink : YES
CapIsValid : YES
RawIsValid : YES
                                                                                                  RawUpdateErrCnt : 0
ParseErrorType : 0
                                                                                                   ParseWarnType
0x00000000
                                           : 0
                                                                                                                                                : 256
RawGetErrCnt
                                                                                                RawLength
 ----- BasicCap
```

HDMI1.4Support	: YES	1stBlockVersion	: 1.3
HDMI2.0Support MaxTMDSClock(MHz)	: NO	ManufacturerName ProductCode	: SAM
MaxTMDSClock(MHz)	: 300	ProductCode	: 3006
HDCP1.4Support	: N/A	SerialNumber	: 1
WeekOfManufacture			
SCDCSupport	: NO	YearOfManufacture	
DVIDualSupport AISupport	: NO	CECAddrIsValid	
	: YES	CECAddr	:
01.00.00.00	. 1	Cross les es Courses a ret	. EI ED
ExtBlockCnt		SpeakerSupport	: rL_rk
NativeFormat	: 3840X2160P60 16:9(V	IC 97)	
	: RGB444 YCbCr444 YCbC		
DeepColor	: RGB_30Bit RGB_36Bit	YCbCr444_SameRGB	
YCbCr420DeepColor	:		
YCbCr420[Also]	: 97 96 101 102 98		
YCbCr420[Olny]			
Colorimetry	: 		
		-	
3840X2160P60 16:9 16:9	1080P_60 16:9	1080P_50 16:9	720P_60
	1080i_60 16:9	1080i_50 16:9	1080P_24
16:9			
	1080P_30 16:9	3840X2160P24 16:9	
3840X2160P25 16:9			
3840X2160P30 16:9		4096X2160P50 256:135	
4096X2160P60 256:135			
	4096X2160P25 256:135		NTSC 16:9
PAL 16:9	480P_60 16:9	576P_50 16:9	
3840X2160P30 16:9	3840X2160P24 16:9	400CV21C0D24 2EC-12E	1150064
			1440x900
			1440X900
V800X600 56	1680x1050	V800X600_60	
_	V640X480 72	V640X480 67	
V640X480 60	V 0 1021100_72	V 0 1021100_07	
_	V720X400 70	V1280X1024 75	
V1024X768 75			
V1024X768 70	V1024X768 60	V1024X768 87	
	3DCap	_	
2D0	. NO	3D0-4Diit	. NO
3DSupport 3DDualView		3DOsdDisparity 3DIndepView	
3DTypeSupport		SDINdepview	: NO
NO.0:	. I DOM	MarchannalNi	. 2
CodeType		MaxChannelNum BitDepth	
MaxBitRate(KHz) SampleRate(Hz)		propedit	. 10 20 24
	: 32000 44100 46000 DetailTimir	nα	
[NO.]:HACT VACT P/I	PClk AspW AspH HFB F	HPW HBB VFB VPW V	BB ImgW
ImgH IHS IVS IDV	_		

[0]:3840|2160|P |594M|0 |0 |176 |88 |384 |8 |10 |82 |1872|1053|YES |YES |NO

[1]:480 |1920|P |2 M|0 |0 |28 |278 |3301|2 |0 |718 |2092| 37 |NO |NO |NO 此proc信息具体含义,请参见"2.13 HDMI HDMI"章节中关于HDMI proc信息的分析。

3.4.2 常见问题分析和定位

stbbox 启动后没有声音输出

问题分析和定位如下:

- 检查HDMI是否变为DVI模式导致的。 可以通过分析 /proc/msp/hdmi0 信息,看当前处于HDMI模式还是DVI模式。
- 请确认AO音频设置的采样率,与cat /proc/msp/hdmi0中HDMI设置的采样率一致。
- 如果分析/proc/msp/hdmi0发现被设置成了透传,则cat /proc/msp/hdmi0_sink检查一下sink设备是否支持透传。

fastboot 启动后为什么有杂音

问题分析和定位如下:

这是个兼容性问题,机顶盒处于HDMI模式下,且音频模块无时钟、无音频输出时,电视仍然对HDMI接口输入进行时钟恢复导致的,大部分电视在上述状态下是不会做如下动作的:进行时钟恢复,进而输出声音。

为什么电视会出现红屏现象

问题分析和定位如下:

- 请确认HDMI proc下是否有信息帧的信息,如果没有,说明没有发送信息帧,则会变红。
- HDMI的 HI_UNF_HDMI_Start是否在热插拔函数中,如果没在,热插拔就会处理 异常,例如:红屏等;其次,确认在应用中是否还在调用该函数,如果调用了, 则会出现偶尔红屏的现象。

为什么电视 EDID 获取不到

问题分析和定位如下:

- 请确保HDMI I2C通道的上下拉电阻是否正确,请参考硬件的原理图参考设计。
- HDMI线插拔时,测量I2C的波形,与正常电视对比,出问题多是电视的I2C出现问题。

为什么电视会出现间歇性黑屏

目前碰到的这类问题,都是由于硬件出现问题导致的:

- 请确认HDMI眼图的指标参数是眼图参考范围内的。
- 请确保HDMI的电源设计按照硬件的参考设计来,否则clock jitter会异常,导致出现问题。

为什么电视会出现认证不过的问题

问题分析和定位如下:

● HDCP认证不通过

由于HDCP对于热插拔的影响是比较敏感的,因此,请参考海思的sample/common/hi_adp_hdmi.c 中的热插拔调用流程。

- 和DVI设备对接,发送了信息帧导致不通过 请调用HI_UNF_HDMI_Init()接口,将HDMI输出模式强制设置成 HI_UNF_HDMI_FORCE_DVI。
- 宽高比认证不通过 请确认调用了HI_UNF_HDMI_SetInfoFrame和HI_UNF_HDMI_SetAttr来设置了宽 高比。
- 像素重复认证不通过 先确认VDP的ZME和IFIR的配置正不正确。再确认HDMI的Dither配置。

3.5 视频显示问题定位方法

3.5.1 常见视频显示问题

视频画面为黑屏,或者显示某个画面静止不动

连续多次观察window proc信息,通过proc先确认窗口是否pause和freeze的状态,如果是这两种状态之一,画面静止或者黑屏为正常现象;否则,观察QFrame(Try/OK)是否在增加,如果没有增加,说明应用(一般是AVPLAY)没有推送新的视频图像,需要专业人员定位。

设置 16:9 或者 4:3 输出,并且设置转换模式(加黑边,剪切等),但是画面没有变化

观察window proc信息,确认宽高比转换模式ARConvert是否与预期设置一致。

如果设置无误,请确认视频源的宽高比例信息(WvsH)是否与输出窗口一致,如果一致,则转换模式无效果。

视频画面显示不连续,存在跳帧情况,或者视频画面出现卡顿,尤其是水平滚动字幕

观察window proc信息中的Underload计数:

- 如果有增加,说明出现欠载,可能与系统性能有关,需要针对具体场景分析;
- 如果没有增加,请观察State是为QuickMode,如果是,说明处于"快速输出"模式,则不连续为正常现象。
- 其他情况需要对AVPLAY和同步模块分析。

多窗口场景下, 第 3 个或者后续的窗口在创建或 enable 时, 可能会失败

当窗口个数大于等于3个时,窗口布局有如下约束:

- 多窗口之间,只允许其中一个窗口与其它窗口重叠,剩余窗口彼此之间不能重叠;
- 此窗口zorder值,要么大于所有与之重叠窗口的zorder值,要么小于 所有与之重叠窗口的zorder值,不能处于之间。

需要修改窗口布局,或者将需要放到后台的窗口给disable掉,来满足上述约束。

□说明

- 只有处于enable状态的窗口之间有此约束, disable状态的窗口无此约束。
- 窗口布局是只,多个窗口存在时,彼此之间的位置关系及叠加先后顺序。

多窗口播放码流时,某窗口显示内容可能会发生切换,显示了其它窗口的内容。

此类问题在开发过程中碰到过。可以通过多次抓取出问题窗口上的码流,对比在时间 先后顺序上,内容是否发生切换,来判断出问题的环节,命令如下:

- echo help > /proc/msp/winxxx;
- echo capture path > /proc/msp/winxxx;

如果发生切换,说明前级模块送出的码流有问题。 否则说明VO处理有问题。

3.6 音频问题定位方法

3.6.1 通过分析 PROC 信息定位音频问题

音频数据来源有ES流和TS流,支持的音频格式(mp3/pcm/aac/ac3/dts/dtshd/dd/ddp等) 众多,输出接口有Adac/Spdif/Hdmi,因此音频相关的问题也比较多样。

ADEC和AO的PROC信息中含有很多有用信息可以用来分析音频问题。

ADEC PROC信息:

```
#cat /proc/msp/adec00
       WorkState:start
CodecID: 0x202f1011
DecoderName: Dolby TrueHD Decoder
Description: hisi truehd
DecodeThreadID:2537
Volume:100
SampleRate:96000
BitWidth:24
Channels:2
PcmSamplesPerFrame: 1840
BitsBytePerFrame:0x0
StreamFormat:non-packet
TryDecodetimes:1088
FrameNum (Total/Error):767/0
StreamBuf(Total/Use/Percent)(Bytes):262144/205883/78%
StreamBuf(readPos/writePos):0xdbcc/0x9000
OutFrameBuf(Total/Use/Percent) :8/7/87%
GetBuffer (Try/OK): 4967/3529
PutBuffer (Try/OK): 3529/3529
SendStream(Try/OK):0/0
ReceiveFrame(Try/OK):767/760
PtsLostNum: 0
DecodeThreadExecTimeOutCnt:1
DecodeThreadScheTimeOutCnt:0
AO PROC信息
# cat /proc/msp/sound0
      -----Sound[0]Status------
SampleRate: 48000
SPDIF Status: UserSetMode (RAW) DataFormat (DD)
HDMI Status:UserSetMode(RAW) DataFormat(DDP)
```

```
------SutPort Status------
ADACO: Status(start), Mute(off), Volume(00dB), TrackMode(STEREO)
SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x0),
PortID(0x12)
DmaCnt(000850), BufEmptyCnt(000002), FiFoEmptyCnt(000000)
SPDIF0: Status(start), Mute(off), Volume(00dB), TrackMode(STEREO)
SampleRate(048000), Channel(02), BitWidth(16), Engine(SPDIF RAW),
AOP(0x1), PortID(0x21)
DmaCnt(000828), BufEmptyCnt(000000), FiFoEmptyCnt(000001)
HDMIO: Status(start), Mute(off), Volume(00dB), TrackMode(STEREO)
SampleRate(192000), Channel(02), BitWidth(16), Engine(SPDIF RAW),
AOP(0x3), PortID(0x20)
DmaCnt(000756), BufEmptyCnt(000000), FiFoEmptyCnt(000000)
I2S0: Status(start), Mute(off), Volume(00dB), TrackMode(STEREO)
SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x4),
PortID(0x10)
DmaCnt(000850), BufEmptyCnt(000000), FiFoEmptyCnt(000000)
------ Cast Status-----
Cast(0): *Aop(0x5), Status(start), UserEnable(On), Weight(100/100),
Mute (off)
SampleRate(048000), Channel(02), BitWidth(16)
MaxFrameNum (08), SamplePerFrame (01024), AcquireFrame (On)
----- Track Status-----
Track(0): Type(master), Status(start), Weight (L/R) (100/100),
ChannelMode (STEREO), Mute (off)
SpeedRate(00), AddMuteFrames(0000), SendCnt(Try/OK)(001112/000322)
AIP(0): Engine(PCM), SampleRate(048000), Channel(02), BitWidth(16),
DataFormat(PCM) , Priority(low), FifoBypass(off)
EmptyCnt(000000), EmptyWarningCnt(000000), Latency/Threshold(378ms/400ms)
AIP(1): Engine(SPDIF RAW), SampleRate(048000), Channel(02),
BitWidth(16), DataFormat(DD), Priority(low), FifoBypass(off)
EmptyCnt(000000), EmptyWarningCnt(000000), Latency/Threshold(388ms/400ms)
AIP(2): Engine(HDMI RAW), SampleRate(192000), Channel(02), BitWidth(16),
DataFormat(DDP) , Priority(low), FifoBypass(off)
EmptyCnt(000000), EmptyWarningCnt(000000), Latency/Threshold(348ms/400ms)
```

音频数据流如图3-1所示。

图 3-1 音频数据流

```
ESStream

TSStream

ADECInBuf -Decode -ADECOutBuf -SendData -DPCM -Spdif

MPCM Hdmi
```

当音频出现问题,根据数据流流程分析包括以下步骤:

步骤1 音频数据送ADECInBuf是否正常。

ES流:

```
GetBuffer(Try/OK):a1/a2
PutBuffer(Try/OK):a3/a4
```

如果a1无增加,说明送ES数据线程出问题。

TS流:

SendStream(Try/OK):a5/a6

如果a5无增加,说明DEMUX送音频ES数据出问题。

步骤2 音频解码是否正常。

CodecID:b1

TryDecodetimes:b2

FrameNum (Total/Error):b3/b4

如果a4/a6有增加,b3无增加,说明没有执行音频解码。

如果b2有增加,b3无增加,说明没有成功解码,需查看b1选用的解码器是否正确。

步骤3 AVPLAY从ADECOutBuf获取音频解码后数据是否正常。

ReceiveFrame(Try/OK):c1/c2

- 如果c1没有增加,则没有调用获取音频解码后数据的接口。
- 如果c1增加,c2没有增加,则获取音频解码后数据失败。

步骤4 音频数据送Track是否正常。需要观察Track的状态。

cat /proc/msp/sound0
SendCnt(Try/OK) (d1/d2)

如果d1没有增加,则没有调用向Track送数据的接口。

如果d1增加,d2没有增加,则向Track送数据失败。

步骤5 音频端口输出是否正常。需要观察AO的状态。

cat /proc/msp/sound0
DmaCnt(e1)

若e1没有增加,则端口没有输出数据。

----结束

3.6.2 分析音频无输出问题

如果是ES流不能播放,首先检查是否有对应的解码库,或者该音频格式是否识别错误,对于TS流除可能格式识别错误外,音频PID错误也会导致不能正常播放。

确认音频流格式是否为SDK支持,目前SDK只支持如下采样率(Hz): 8000、11025、12000、16000、22050、24000、32000、44100、48000、88200、96000、176400、192000。

当音频ADEC和AO PROC都正常,却无声音输出,需注意以下:

ADAC

是否加了静音电路, 若加了静音电路, GPIO管脚是否配置正确。

Hdmi

确认Hdmi有连接,Hdmi是否设置了透传,而Hdmi连接的设备却不支持透传输出。

3.6.3 音频透传格式相关设置

当播放ac3/dd/ddp/dts/dtshd等含透传数据音频,选择了HD_DEC_MODE_THRU(仅透传)/ HD_DEC_MODE_SIMUL(PCM+透传)解码模式,同时使能了透传数据输出,AO的PROC有如下信息:

SPDIF Status :UserSetMode(RAW) DataFormat(DD)
HDMI Status :UserSetMode(RAW) DataFormat(DDP)

这时HDMI接电视可能无声音输出,因为HDMI输出的是透传数据(大部分电视不能解码,功放可以),若要电视有声音输出,需将HDMI设置成PCM或AUTO模式。

当音频格式为TrueHD,不能解出PCM数据,只能通过HDMI透传输出。

□□说明

DDP、DTSHD透传输出到功放时,此时采样率为192K,并不是所有的功放都支持。

3.6.4 音频输出不正常

检查 PCM 格式参数配置

当播放PCM格式音频,必须保证如下参数配置正确,否则声音不正常:

- Channels(声道数)
- SamplesPerSec(采样率)
- BitsPerSample(位宽)

声音输出有杂音定位

通过观察AO Proc信息,检查AO是否有欠载。如果欠载:此时可能是"数据源异常或者码流推送不足"。首先建议在PC上使用播放器检查原始码流是否正常,如果数据源是正常的,可能是码流推送不足,则需要继续定位。

本地复合流播放声音有问题

通过观察AO Proc信息,检查AO是否有异常,通过ADEC Proc信息,检查ADEC是否有异常。如果有ADEC有异常,需要分析容器解析是否正确解复用音频ES流,建议:

- 使用mediainfo工具检查复合流音频信息,对照ADECProc信息,检查解码器类型是否一致。
- 使用demux工具,提取音频ES流,使用Sample/Esplay播放,如果有问题,需要分析定位ADEC解码器。如果声音正常,很可能问题出在文件解复用环节。

3.7 同步问题的定位方法

首先需要明确,音视频同步处理解决的是将音视频从不同步调整为同步,同时尽量保证音视频的播放效果。引起不同步产生的原因较多,同步处理没办法去规避不同步的产生;无论快同步或慢同步,同步调整都需要一定时间,同步过程中需要改变音视频的正常播放速度,不可能完全不被用户感知。在这个基础上,如果怀疑是同步处理引起的问题,可以从以下几个方面去排查。

3.7.1 关闭同步

关闭同步的方法:

echo SyncRef = none >/proc/msp/sync00

关闭同步后有以下2种情况:

- 如果关闭同步后问题现象消失,则问题很可能与同步相关,可以跳至"3.7.2 查看同步的proc信息3.7.2 查看同步的proc信息"进行排查。
- 如果关闭同步后问题现象依然存在,则问题不是由同步引起的,需要去定位其他 模块的问题。

3.7.2 查看同步的 proc 信息

查看同步的Proc信息方法如下:

\$cat /proc/msp/sync00

● 查看同步属性:

同步属性设置如图3-2所示。

图 3-2 同步属性设置

Hisilicon SYNC ATTR	
SyncPrint	:1
SyncRef	:AUDIO
SyncStart.VidPlusTime	:60
SyncStart.VidNegativeTime	:-20
SyncStart.bSmoothPlay	:1
SyncNovel.VidPlusTime	:3000
SyncNovel.VidNegativeTime	:-3000
SyncNovel.bSmoothPlay	:0
VidPtsAdjust	:0
AudPtsAdjust	:40
PreSyncTimeoutMs	:0
bQuickOutput	:0

- 查看同步设置的参考基准SyncRef是否设置成功,其中AUDIO为以音频为准,PCR 为以PCR为准,NONE为自由播放。
- 查看设置的同步调整区间。
 - SyncStart.VidPlusTime与SyncStart.VidNegativeTime表示起调区间,如[-20, 60], SyncStart.bSmoothPlay表示起调区间是否设置慢同步。
 - SyncNovel.VidPlusTime与SyncNovel.VidNegativeTime表示异常区间,如 [-3000,3000], SyncNovel.bSmoothPlay表示异常区间是否设置慢同步。
- 查看预同步状态及PCR状态:

图 3-3 预同步状态及 PCR 状态

Hisilicon H	PCR
CrtStatus	: PLAY
PreSyncStartSysTime	:861737453
PreSyncEndSysTime	:861737453
PreSyncFinish	:1
BufFundEndSysTime	:-1
BufFundFinish	:1
PreSyncTarget	:VID
PreSyncTargetTime	:-1
PcrFirstCome	:1
PcrFirstSysTime	:861737474
PcrFirst	:30050398
PcrLast	:30056960
PcrLocalTime	:30056982

如果(PreSyncEndSysTime - PreSyncStartSysTime) >= PreSyncTimeoutMs

表示预同步超时,进入播放状态后,还会继续调整,有可能会出现卡顿。否则,表示在预同步阶段已经调整到同步状态。

预同步目标为VID,表示视频超前音频,视频需要等待音频; 预同步目标为AUD表示音频超前视频,音频需要等待视频。在预同步超时前,当音视频Pts都足够靠近预同步目标时间时,预同步完成。

● 查看视频同步调整状态:

图 3-4 视频同步调整状态

Hisilicon VID			
VidFirstCome	:1		
VidFirstSysTime	:861737924		
VidFirstPts	:30051372		
VidLastPts	:30056932		
VidPreSyncTargetInit	:0		
VidPreSyncTargetTime	:-1		
VidFirstPlay	:1		
VidFirstPlayTime	:861737944		
VidBlockFlag	:0		
VidBufPercent	: 0		
VidDiscard	: 0		
VidSyndAdjust	:0		
VidLocalTime	:30056946		
VidPcrDiff	:-36		
VidAudDiff	:37		
TplaySpeed	:0		

VidAudDiff表示当前视频pts-音频pts的差值,如果在起调区间内,说明同步已调整完毕,如果不在,说明同步仍在调整。

VidDiscardCnt表示同步调整中丢弃的视频帧数, VidRepeatCnt表示同步调整中重复的视频帧数。

3.7.3 使用 msp_debug 工具定位同步问题

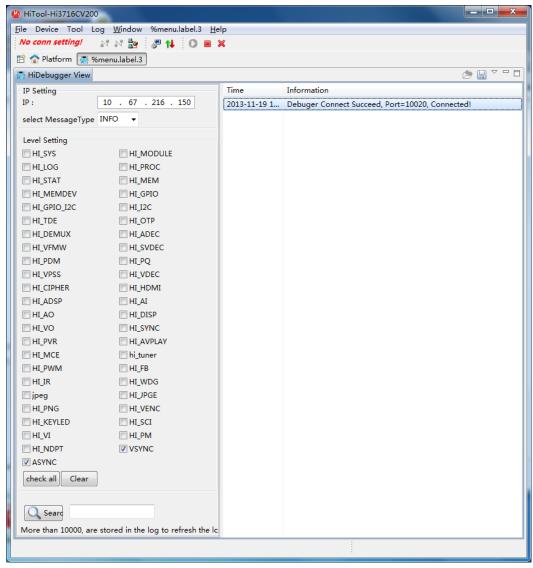
利用msp_debug捕获同步log信息操作步骤如下:

步骤1 单板上运行: msp debug &

步骤2 PC上运行发布包中的HiTool工具: tools/windows/HiTool, 在HiTool平台上选择HiDebugger工具。

步骤3 设置HiDebugger界面,启动日志抓取。如图3-5所示。

图 3-5 HiDebugger 配置界面



设置说明如下

- 设置单板IP: 假设为10.67.212.150。
- 点击 [©] 连接单板,连接成功提示"Debuger Connect Succeed"。
- 设置信息抓取级别:设置为INFO级别。
- 设置需要抓取LOG的模块: 勾选VSYNC与ASYNC。
- 点击 □ 可保存log到文件。

步骤4 一切就绪后,开始播放不同步的节目或码流。Hidebuger将记录下同步的log。

步骤5 可以将该log发给研发人员分析。

----结束

3.7.4 查看可能导致不同步的原因

观察VO和AO的proc信息看是否有音视频欠载。

- 连续多次cat /proc/msp/window0100 观察Unload是否递增。
- 连续多次cat /proc/msp/sound0 观察FifoUnderflowCnt是否递增。
- 如果音视频欠载,肯定会引起音视频不同步,需要查找导致音视频欠载的原因。

步骤1 连续多次: cat/proc/msp/win0100,观察Unload是否递增。

图 3-6 执行 cat /proc/msp/win0100 后的打印信息

```
cat /proc/msp/win0100
                               -Win 100[Z=1]-
                                              -----Frame Info--
       ----Win Info-
                                           |Type/PixFmt
                                                               :NotStereo /NV21
Enable
                    :True
State
                    :Run
                                           |Circurotate
                                                               :False
Туре
                    :Main
                                           |W/H(W:H)
                                                               :1920/1080( 16:
LayerID
                    : 0
                                           |Disp(X/Y/W/H)
                                                              : 0/ 0/1920/1080
                                           FrameRate
                                                               :500.50000
AspectRatioConvert :Full
                                                               :BT601 YUV LIMITED
CustAspectRatio
                                           |ColorSpace
                                           |Fieldmode(Origin) :Frame(Top)
                    :False
Crop
Crop(L/T/R/B)
                        0/
                                   0/
                                        0 |OriRect(X/Y/W/H) :0/0/720/576
                              0/
In (X/Y/W/H)
                        0/
                              0/
                                   0/
                                        0 |FrameIndex
                                                               :0x107
                              0/1280/ 720 |SrcPTS/PTS
                                                               :0x417504d/0x417504d
Out(X/Y/W/H)
                        0/
DispMode/RightFirst:2D
                                           |PlayTime
                                /False
                                                               :1
                                           |FieldMode
Masked
                                                               :All
                    :False
AttachSource
                    :False
                                           |Fidelity
                                                               : 0
                                           |YAddr/YStride
CallBack (Acquire)
                    : N
                                                               :0x236c1020/0x780
CallBack(Release)
                                           |CAddr/CStride
                                                               :0x238bb420/0x780
                    :Y
CallBack(SetAttr)
                    : Y
SlaveWinID
                    :0000
                    ----Buffer State----
Queue (Try/OK)
                     :545/545
                     :522/522
Dequeue (Try/OK)
Config
                     :537
Underload
                     :30
Discard
                     : 2
UndispFrame(Q/DQ)
                     :4/4
FieldUnmatchCnt
                     :16
BufferQueue[state, FrameID]
(State: 1,Empty[8]; 2,Write[0]; 3,ToDisp[6]; 4,Disp[2]; 5,Disped[0])
[3,0x10a] [1,0x103] [1,0x103] [1,0x104] [1,0x104] [1,0x106] [1,0x106] [1,0x106] [1,0x106] [3,0x108]
[3,0x108] [3,0x109] [3,0x109] [3,0x10a]
```

步骤2 连续多次cat /proc/msp/sound0,观察IfFiFoEmptyCnt是否递增。

图 3-7 执行 cat /proc/msp/ sound0 后的打印信息

```
cat /proc/msp/sound0
                                                Sound[0] Status
SampleRate
               :48000
SPDIF Status
               :UserSetMode(PCM) DataFormat(PCM)
HDMI Status
               :UserSetMode(PCM) DataFormat(PCM)

    OutPort Status

ADACO: Status(start), Mute(off), Vol(00dB), TrackMode(STEREO)
      SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x0), PortID(0x12)
      DmaCnt(039693), BufEmptyCnt(000006), FiFoEmptyCnt(000003)
SPDIF0: Status(start), Mute(off), Vol(00dB), TrackMode(STEREO)
      SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x1), PortID(0x21)
      DmaCnt(039693), BufEmptyCnt(000006), FiFoEmptyCnt(000003)
HDMI0: Status(start), Mute(off), Vol(00dB), TrackMode(STEREO)
      SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x2), PortID(0x13)
      DmaCnt(039693), BufEmptyCnt(000008), FiFoEmptyCnt(000003)
I2S0: Status(start), Mute(off), Vol(00dB), TrackMode(STEREO)
SampleRate(048000), Channel(02), BitWidth(16), Engine(PCM), AOP(0x4), PortID(0x10)
      DmaCnt(039693), BufEmptyCnt(000000), FiFoEmptyCnt(000000)
```

----结束

如果音视频欠载,肯定会引起音视频不同步,需要查找导致音视频欠载的原因。

3.8 信道前端问题定位方法

3.8.1 调试手段

/proc/msp目录下提供了Tuner的调试信息:

cat /proc/msp/tuner: 提供了是否锁频成功、tuner类型、demod类型等信息。

通过cat /proc/msp/tuner查看锁频状态LockStat,如果LockStat对应是unlocked,那就是没有锁住,如果LockStat对应是locked,那就是锁了。也可以直接读取demod的锁定指示寄存器进行更加详细的判断,可以判断出锁定到哪个阶段,如agc锁定,cbs锁定,tr锁定,帧同步锁定,fec锁定等等,这样对于机顶盒工作的状态能做出更加准确的判断。举例说明使用方法::例如使用的demod是Hi379XV100的内置demod,锁定指示寄存器地址是0x4a,则可以使用echo命令直接读取demod的寄存器值,如下:读出的寄存器值是0xf8,对照Hi379XV100手册中该寄存器各个bit的含义,可判断出demod已经锁定。需要注意的是:不同demod的锁定指示寄存器地址和寄存器各个bit的含义都可能不一样,具体请参考demod的手册,echo命令详细使用方法后面有详细介绍,这里不再赘述。

echo 5 a0 4a >/proc/msp/i2c
0xf8

3.8.2 不能锁频的问题定位方法

常见的不能锁频的原因有很多,如信号不好,i2c总线不通,器件设置不对等都有可能。遇到这种问题时,首先需要定位清楚导致问题的原因,再予以解决。卫星机和有线机顶盒很大的一个不同就是卫星机的前端LNB,遇到问题需先确认清楚问题出现在哪个环节,是LNB配置不对,还是接收机本身不能正常接收。

判断问题出现环节

首先大致判断问题是信号本身不好,LNB设置不对,tuner还是DEMOD配置不对。可以使用其他正常的机顶盒测试信号是否正常;LNB的设置主要是13/18V和22k信号,都可以使用示波器直接测量,也可以使用正常机顶盒的环出信号测试;部分tuner器件提供PLL是否锁定指示,可通过i2c读取对应寄存器判断tuner是否锁定,也可以直接使用频谱分析仪观察tuner的中频输出信号,判断tuner中频是否正常。

常见问题及解决方法

● LNB输出信号有问题

LNB设置主要指13/18V和22k信号,这两种信号都可以直接使用示波器观察,只要给出的信号符合规范,一般都不会有问题。

● i2c总线不通

常见的tuner和demod器件都是通过i2c总线控制的,i2c总线不通是完全没法正常工作的,这种问题可以从以下几方面定位:

- i2c通道是否设对:
- 从设备设备地址是否正确;
- 从设备是否复位未释放:
- 使用硬件管脚复用是否设对。 可以使用echo命令辅助调试,使用格式:
- 读:

echo <i2c通道号> <从设备地址> <寄存器地址> >/proc/msp/i2c

- 写:

echo <i2c通道号> <从设备地址> <寄存器地址> <写入值> >/proc/msp/i2c

● 判断tuner输出信号电平是否过大或过小

使用上面的echo命令查看bagc_ctrl_1 (0x60) 和bagc_stat_7 (0x79) 寄存器。以 Hi3716CV200为例,假如I2C通道为5,从设备地址为0xa0,则:

echo 5 a0 60 >/proc/msp/i2c echo 5 a0 79 >/proc/msp/i2c

把读出的值作比较,如果差值>4,说明tuner输出信号多大或者多小,需要查看RF信号是否过大或者过小,比如超过110dbuv或者小于30dbuv(取决以tuner的灵敏度)。

● 判断tuner输出信号是否有电平抖动

频点多次读bagc stat 7(0x79)寄存器,命令:

echo 5 a0 79 >/proc/msp/i2c

查看其变化范围,如大于±3,且变化频繁,说明tuner输出信号有电平抖动,使用频谱仪测试RF信号和tuner输出信号进一步定位。

● 判断tuner输出信号是否有频偏

使用echo命令查看cr_stat_4 (0x32)、cr_stat_5 (0x33)、cr_stat_6 (0x34)、cr_stat_7 (0x35) 寄存器,命令:

```
echo 5 a0 25 1 >/proc/msp/i2c
echo 5 a0 32 >/proc/msp/i2c
echo 5 a0 33 >/proc/msp/i2c
echo 5 a0 34 >/proc/msp/i2c
echo 5 a0 35 >/proc/msp/i2c
```

Value1、Value2、Value3、Value4分别代码0x32、0x33、0x34、0x35读出的值, 计算频频偏的方法:

```
FreqOffset = Value1*256*256*256 + Value2*256*256 + Value3*256 + Value4;

If (FreqOffset > 2.^27)
{FreqOffset = FreqOffset - 2.^28;
}
FreqOffset = FreqOffset/(2.^27);
FSymbol = 实际符号率; (例如 6875000)
Frequency = (FSymbol/(2*pi)).*FreqOffset;
```

其中:

- Frequency是以Hz为单位的频率偏差
- pi是3.14159265

如果Frequency和输入的频率值的差值大于25kHz,需要具体定位是RF信号还是TUNER的中频引入的频偏,小于25kHz则无影响。

● 如何配置Hi3130V200的寄存器地址分块以及ts_dat[0..7]、 ts_sync、ts_vld、ts_err的 引脚

Hi3130V200为demod的单芯片,通过寄存器four_reg_sel(0xE0)的第1、0位的值把寄存器分为两块,four reg_sel [1:0]含义如下:

- 00: 配置QAM寄存器;
- 10: 配置系统控制寄存器。

比如,需要查看引脚配置寄存器ioshare_ctrl_7 (0x27) ,这个寄存器属于系统控制寄存器,需要先配置four_reg_sel (0xE0) 的第1、0位为10,然后在进行ioshare_ctrl_8 (0x27) 的操作。例如:

```
echo 3 a0 e0 2 >/proc/msp/i2c
echo 3 a0 27 >/proc/msp/i2c
```

各块寄存器的具体功能请查看Hi3130V200的芯片手册。

通过cat /proc/msp/tuner,看到tuner如果处于锁定状态,且error_rs_package为0,且all_rs_package和 corrected_rs_package有值,说明demod模块已经正常工作。

通过cat /proc/msp/demux_port查看port的信息,参见demux的调试信息,

ErrTsCnt的值一直在大量的增加,请优先查看下Hi3130V200的ts_dat[0..7]、ts_sync、ts_vld、ts_err的引脚是否配置正确,涉及到的寄存器有:

- ioshare_ctrl_4 (0x24)
- ioshare ctrl 5 (0x25)
- ioshare_ctrl_6 (0x26)
- ioshare ctrl 7 (0x27)
- ioshare ctrl 8 (0x28)
- ioshare ctrl 9 (0x29)

各块寄存器的具体功能请查看Hi3130V200的芯片手册。 可以通过HI_UNF_TUNER_SetTSOUT的接口配置引脚功能,请参见 hi unf frontend.c中的接口说明。

● 其余问题

比较常见的问题还有agc电压异常(很多时候是agc管脚复用不对),AD的设置不对,需要对照芯片手册对相关寄存器进行比对。

3.9 开机画面和瞬播问题定位方法

开机画面及瞬播常见问题有:开机画面或瞬播无法显示、花屏、过渡不平滑,遇到类似问题时,可通过以下方法定位。

3.9.1 查看参数设置是否正确

步骤1 单板上执行cat /proc/msp/pdm查看基本参数是否正确。

步骤2 单板上执行cat /proc/msp/mce, 查看瞬播参数是否与设置的相符。

步骤3 如果proc中得到的参数与镜像制作中配置的参数一致,请检查参数是否能够正确运行开机画面或瞬播,如单板DAC配置、瞬播频点、Tuner参数等。

步骤4 如果/proc中得到的参数与镜像制作中配置的参数不一致,请将镜像文件回传给研发分析。

----结束

3.9.2 查看开机画面及瞬播 Log

在boot命令行下执行setenv loglevel 4,将开机画面及瞬播打印级别改为Info级,并将打印Logo回传给研发分析。

3.9.3 如何加快瞬播启动速度

加快瞬播启动速度的配置步骤如下:

步骤1 配置boot下的环境变量bootdelay为0。

方法: 在boot下的命令行中输入: setenv bootdelay 0;saveenv。

说明:为了方便进入boot命令行,boot下默认设置bootdelay为1,配置bootdelay为0可以加快fastplay启动时间约1s(boot中已修改代码配置默认值为0)。

步骤2 配置boot阶段不做内核校验。

方法: 在boot下的命令行中输入: setenv verify n;saveenv。

说明:如果内核出错,在boot阶段做不做校验,系统基本都会挂死,因此设置不做校验理论上不会产生影响,该操作可加快启动时间约1s(boot中已修改代码配置默认值为不做校验)。

步骤3 取消kernel阶段的BogoMIPS计算。

方法: 配置bootargs, 在bootargs中加上lpj=5996544。

说明: BogoMIPS用于衡量cpu运行速度,设置lpj=5996544可以取消该计算过程,此举可加快启动时间约0.2s。

步骤4 设置bootcmd。

方法: 在boot下的命令行中输入: setenv bootcmd 'nand read 0x1FFFFC0 0x0 0x600000;bootm 0x1FFFFC0 '。

说明:设置成上述bootcmd后,boot直接将内核镜像从flash读到0x1FFFFC0,然后从0x1FFFFC0启动。

相反如果按照默认bootcmd配置: nand read 0x82000000 0x0 0x600000;bootm 0x82000000,则boot阶段先将kernel从Flash读到0x82000000地址,再将镜像从0x82000000拷贝至0x1FFFFC0,然后从0x1FFFFC0启动。

□说明

这里的0x1FFFFC0仅是其中的一个例子,不同的SDk版本,这个值可能不同,该值的计算方法: 内核启动时的log "Entry Point: xxxxxx", xxxxxxx-0x40即为该值。该值为0x1FFFFC0时内核启动时的log为: Entry Point: 02000000

----结束

3.10 VP 问题定位方法

Proc调试方法参见VI、VPSS、VENC、VDEC、AVPLAY和VO模块调试说明。

3.10.1 本地视频无法显示的问题

首先请检查采集设备(USB摄像头)是否正确连接。如果使用USB摄像头作为视频源,确认插入摄像头后,可以检测到/dev/video0设备;然后通过查看VENC模块的proc信息,重点关注帧率和码率是否正常。

如果VP配置中,选择USB摄像头采集的是MJPEG数据,请确认摄像头是否支持采集 MJPEG格式。

3.10.2 远端视频无法显示的问题

首先使用ping命令检查网络是否通畅,然后确认本地视频显示正常。如果上述功能正常,通过查看VDEC的proc信息,观看是否从网络侧正常收报送给VDEC模块。如果收报异常,使用网络抓包工具定位未收到视频RTP包原因。如果收报正常,再通过AVPLAY和VO的proc信息观看是否工作正常。

另外,可以使用开发包中的sample_vivenc测试非网络场景下,本地编码解码后,视频是否可以正常播放。

3.10.3 画面卡顿、延时问题

首先请检查系统资源,包括CPU占用率和内存使用情况,关闭在可视电话场景下不需要的其他业务功能。然后,通过观测VENC和VDEC的proc信息,检测编码帧率和解码帧率是否相近,如果编码帧率远大于解码帧率,请检查网络是否出现丢包,延时和抖动现象。如果编解码帧率正常,再通过AVPLAY和VO模块的proc信息,检查是否存在欠载现象。

3.11 智能卡问题定位方法

3.11.1 插卡没任何反应如何定位?

可以从如下几个方面入手定位:

- 通过运行cat /proc/msp/sci0查看卡状态信息。如果是"Sci State:NOCARD"表示卡 没插好或者卡检测电平设置错误。
- 通过运行cat /proc/msp/sci0查看卡DetectLevel信息,卡检测电平设置低有效还是高有效,参考电路和原理图,调用HI_UNF_SCI_ConfigDetect(enSciPort, enSciDetect)函数时,enSciDetect卡检测电平参数与之对应。
- 通过运行cat /proc/msp/sci0查看信息是否是"SCI0 is not open",如果是表示还没打开sci0设备。

3.11.2 卡复位失败收不到 ATR 如何定位?

可从如下几个方面入手定位:

- 不同的智能卡在检测到复位信号后,发送全部ATR的时间不同。查看代码,看在调用HI_UNF_SCI_ResetCard()接口后,是否有足够长的时间等待智能卡的ATR或者查询到卡状态为HI_UNF_SCI_STATUS_READY之后,然后再调用HI_UNF_SCI_GetATR()接口获取ATR。CA认证测试中,这一点尤为重要。
- 通过运行cat /proc/msp/sci0查看信息,"Sci State:INACTIVECARD"可能电压使能是高有效还是低有效设置有误,参考电路和原理图,调用HI UNF SCI ConfigVccEn()函数时,输入enSciVcc参数时与之对应。
- 通过运行cat /proc/msp/sci0查看 "ProtocolType:"项显示的协议类型是否与卡一致,不一致通过修改函数HI_UNF_SCI_Open()的enSciProtocol参数达到一致。
- 通过运行cat /proc/msp/sci0查看信息 "Card State: WAITATR"表示检测到有卡插入 并激活,复位不成功,重复插拔卡重试是否能复位成功。
- 用示波器测量SCI接口电源脚是否有VCC电压信号,如果没有电压,则检查SCI硬件连接是否正常。
- 用示波器测量SCI接口复位脚是否有一段低电平复位电压信号,并且复位之后变成 高电平。
- 前面检测的信号都正常的情况下,在用示波器测量数据I/O线有没有数据响应信号,没有则可能卡有问题。

3.11.3 能收到 ATR,发送完命令之后,接收数据没响应,提示接收超时如何定位?

可从以下几个方面入手定位:

- 通过运行cat /proc/msp/sci0查看 "ProtocolType:"项显示的协议类型是否与卡一致。
- 通过运行cat /proc/msp/sci0查看ClockMode,检查是OD模式还是CMOS模式是否与硬件电路一致,调用HI_UNF_SCI_ConfigClkMode()时,配置正确的enClkMode时钟模式。
- 确认参数都设置正确之后,用cat /proc/msp/sci0查看SCI的proc ATR信息,通过该信息分析卡配置的参数是否正确,例如:
 - ActualSciClk:4000 实际SCI时钟为通过计算配置寄存器后实际芯片输出给卡的时钟, ActualSciClk允许与设置时钟有偏差,但必须在卡允许的时钟范围内。

- ExpectBaudRate:10215

通过设置的的F、D因子和设置的clk计算的ETU值,公式如下: ExpectBaudRate= SetFrequency* BitRate *1000/ ClkRate。

- CalcBaudFlag:0

ATR收到TA2并且Bit5=0 为特定模式,该标志位为1,用收到的TA1参数设置etu,没有收到TA2默认为交互模式,该标志位为0。

bSetExtBaudFlag :1

为1表示应用调用HI_UNF_SCI_SetEtuFactor()函数通过外部设置ETU的F、D因子)。

- ClkRate(F):372

实际设置etu的时钟转率因子

- BitRate(D):1

实际设置etu的速率调节因子

- BaudRate:10752

实际计算的波特率BaudRate = ActualSciClk * BitRate *1000/ ClkRate

- AddCharGuard:0 etu 增加额外的字符保护时间,通过收到ATR的TC2来设置,也可以调用 HI UNF SCI SetGuardTime()函数设置,系统默认为0)

分析完ATR和proc信息之后,核对设置值是否与相应的卡对应正确,系统默认的etu都是没有经过ATR值的协商,采用默认值进行设置的(T0、T1的F=372; T14的F=620)。对etu有特殊要求的卡,必须根据ATR返回的TA1值或者由其他固定波特率来设置。

- 如果卡支持协商模式,则可以调用HI_UNF_SCI_NegotiatePPS()接口实现所需波特率的协商。
- 如果卡不支持协商模式,则可以调用HI_UNF_SCI_SetEtuFactor()函数来设置。 (请慎用HI_UNF_SCI_SetEtuFactor()函数,一般的卡通过该接口配置反而会出错)。

注意事项:

以上两函数执行必须等到卡状态为ready后,调用才有效;调用第二个接口会设置 bSetExtBaudFlag标志位。

- 如果proc信息和卡设置都正确,则需要测量实际的CLK时钟是否在卡的要求范围内(如果CLK时钟是OD模式,还要测量时钟上升沿信号的质量是否良好),在卡厂家规范中说明的时钟范围内,降低时钟频率重新测试,如: T0可设置为3570kHz重新测试。
- 测试智能卡时,不同的卡需要有对应不同的的命令字,例如: SCI发送指令期望从卡接收数据0x19,实际卡返回的数据只有0x09,则驱动会认为还有没收到的数据,一直等待致使接收超时。
- 如果proc信息和卡设置都正确,用示波器测量sci IO脚数据信号,看设置的etu时间(也可用1/BaudRate计算)是否与卡发送的数据最小的时间宽度相符,不符则看etu的设置参数是否符合卡要求,必要时需修正设置etu值后再测试。

3.11.4 如何增加冷复位后等待接收 ATR 的时间

智能卡复位激活后,等待接收ATR时间由SCI_ATRSTIME决定,未收到ATR,将自动启动释放序列,无需软件干预,同时发出ATR等待超时中断(对应SCI_RIS[atrstoutim]为

1)。SCI ATRSTIME最大时间可配为65535个SCI CLK,现在驱动默认设置是40000个 SCI CLK,如果要增加等待接收ATR的时间,需要按下面方法修改宏:

把drv sci.h中SCI DFT ATRS TIME改大。

#define sci dft atrs time 40000 //等待atr clk数可改大点

3.12 KEYLED 问题定位方法

3.12.1 KEYLED 面板没显示如何定位?

可从以下几个方面入手定位:

- 检查选择的KEYLED驱动类型是否与单板上的KEYLED芯片相符,目前海思SDK 驱动支持PT6961驱动、PT6964驱动, CT1642驱动, FD650驱动。还需要确认选择 KEYLED驱动的接口函数HI_UNF_KEYLED_SelectType返回成功(即对应的 KEYLED驱动是否已经编译进了SDK),当前SDK支持同时支持一种或者多种前 面板,需要在menuconfig中的Msp菜单下的KeyledConfig菜单中选择。
- 对于使用芯片逻辑驱动的前面板(CT1642),检查KEYLED接口信号管脚复用是 否配置成LED和KEY模式。
- 对于使用GPIO驱动的前面板,检查面板硬件连接的GPIO管脚是否与驱动中信号定 义的管脚一致,不一致则需修改驱动信号定义为硬件连接的管脚号(也在 menuconfig中的KeyledConfig菜单中设置)。
- 对于使用GPIO驱动的前面板,检查KEYLED接口信号管脚复用是否配置为GPIO模
- 检查运行的应用程序中设置的LED段码是否正确,注意前面板中LED数码管是共 阴极还是共阳极。
- 以上几个方面都确认正确的话,就要用示波器测试KEYLED接口的信号脚是否有 正确的信号输出, 进一步确认问题。

3.12.2 CT1642 如何支持系统启动显示?



SPC091之前的版本SDK不支持CT1642系统启动显示,只支持在boot中阶段性的显示。 如果不是CT1642面板则不要执行该流程。在Hi3798系列芯片中, CT1642前面板由芯片 驱动,只需要在boot中调用对应驱动接口显示指定的字符或数字,LED就会在系统启动 阶段一直显示下去, 并不用做如下处理。

海思SDK SPC091之后的版本才支持该功能。配置步骤如下:

步骤1 将menuconfig中的Msp下的KeyledConfig子菜单中的Ct1642 Keyled Support中的 CFG KEYLED CT1642 KERNEL SUPPORT选项配为YES(注意要先将Kernel菜单中 的Build MSP in Kernel选项选中)。

步骤2 make rebuild 重编SDK。

步骤3 编译完后重新烧录kernel、rootfs。

步骤4 用户传入要显示的字符,如:要显示"boot",在bootargs中添加LedDisplay=boot参数。

----结束

3.12.3 前面板 GPIO 使用修改说明

当客户使用的前面板不是由海思芯片逻辑驱动,而是通过GPIO来扫描KEY按键、输出 LED显示数据时,如果这些GPIO管脚与海思DEMO板所使用的不同时,可以按照以下 办法调整:

步骤1 修改fastboot下表格中的GPIO管脚复用为GPIO功能。

具体修改哪几项,取决于前面所使用的GPIO端口号和个数。

管脚复用配置可以在《Hixxxx 高清网络媒体处理器 硬件 用户指南.pdf》中"复用寄存器"一节中查找。

步骤2 修改相应的KEYLED配置,设定前面板使用的GPIO。

步骤3 在SDK根目录下输入命令make menuconfig,就可以进入图形配置界面,再选择Msp菜单下的KeyledConfig子菜单,就可以根据不同的前面板配置其对应的GPIO值。

【举例】

如果使用的是PT6964前面板,且使用的PT6964的线序是: Data信号接到了LED_DATA 脚,CLK信号接到了LED_CLK脚,STB信号接到了LED_STB脚。

【修改】

fastboot下的表格中配置GPIO管脚复用。将下面这2个都改成2:

- 修改pin_mux页中led_gpio_jtag的value为2(对应于gpio5_2, gpio5_3)。
- 修改pin mux页中csn0 gpio jtag对应的value为2(对应于gpio5 4)。

选中KeyledConfig菜单中的Pt6964KeyledSupport选项,并修改各选项值如下:

ClockGpioNumber 0x2b STBGpioNumber 0x2c DINOUTGpioNumber 0x2a

然后重编SDK。



注意

上述GPIO的设置并不能影响到待机下的KEYLED驱动。

----结束

3.13 LOADER 问题定位方法

3.13.1 BootLoader 输出调试日志的方法

在BootLoader方案中,可在Fastboot的命令模式下更改loglevel(范围: 0~4)的值来调整log的信息,loglevel为0的时候关闭log信息,loglevel为4的时候打开所有的log信息。

fastboot# setenv loglevel 4
fastboot# saveenv

3.13.2 AppLoader 输出调试日志的方法

在AppLoader方案中,可以使用环境变量loglevel(范围: 0~4)的值来控制调整log的信息,loglevel为0的时候关闭log信息,loglevel为4的时候打开所有的log信息。

loglevel=4 ./loader

3.13.3 烧写 1080 制式的开机 logo 后, 检测升级不成功问题

Loader图层创建需要依赖baseparam参数,这样进入升级时不至于调整电视机制式。由于hifb的内存分配默认按照720p的分辨率分配,如果baseparam参数要使用1080P分辨率的图片,需要重新配置hifb的MMZ分配。否则会因内存不足而无法申请到足够的内存导致检测升级不成功。

解决方法:修改loader程序加载ko的脚本,参照脚本中的说明,insmod hi_fb.ko时传入参数指定hifb分配内存的大小。

注意:此问题仅有可能在AppLoader下出现,BootLoader下不需要指定hifb的MMZ内存,不存在此问题。

3.13.4 BOOT 对 kernel 校验失败问题

当kernel的大小(以十六进制表示)的后12个位的值在0x000~0x03F之间时,CRC校验不通过,导致kernel无法被启动。(受影响的版本范围是SPC060~SPC060F,SPC070~SPC091)

解决办法:

使用工具uimage align.sh将kernel按8K字节对齐。使用方法如下:

Usage: ./uimage align.sh uImage hi kernel.img

- uImage: 没有对齐的kernel。
- hi kernel.img: 8K对齐后的kernel。

对齐后的kernel(hi_kernel.img)可以正常启动。

3.14 PMOC 调试定位方法

以下描述的是待机流程中MCU运行部分和内核运行部分的调试方法:

3.14.1 串口打印

在串口模块正常工作的时候,可以在待机代码中使用printf_char打印单个字符,或者使用printf_val打印数字。要保持串口模块的正常工作,需要保证串口的时钟打开,供电正常,具体控制寄存器请参考相关芯片的用户指南文档。

在较早的平台下,待机代码中添加了DEBUG_PRINT_ON宏来控制串口模块的功能,如果定义了这个宏,就可以使用上述打印函数在串口上打印调试信息。

在当前发布的系列芯片平台下,可以通过设置如下debug参数,打开待机模块串口打印功能:

echo debug=4 > /proc/msp/pm

输入上面的命令后进入待机,默认会将进入待机时的唤醒参数打出来,格式如下,可以用来确认想要设置的参数是否能正确的设置下去:

====== [MCU PARAM] ====== Chip Type: 0x00000021 Chip Version:0x00000200 Suspend Type: 0x00000000 Reboot Enable: 0x00000000 Debug Mask: 0x00000004 Power Gpio No:0x00000000 Power Gpio Value:0x00000000 Display Mode: 0x00000002 Display Time Hour: 0x00000009 Display Time Minute: 0x0000003A Display Time Second: 0x00000000 WakeUp Time: 0xFFFFFFFF WakeUp Key: 0x00000000 IR Type: 0x00000000 IR No:0x0000001 IR KeyLow:0x639CFF00 IR KeyHigh: 0x00000000 WakeUpGpioNo:0x000000FF ====== [MCU Init Ready] ======

,如果还有更进一步的调试需求,还可以在待机代码中自行使用printf_char, printf_val,printf_str,printf_hex等函数打印需要的调试信息。值得注意的是,上述几 个串口打印函数都会在内部修改regAddr和regData这两个全局变量,打印之后需要重新 设置或者获取这两个变量的值,防止打印影响代码原来的流程。



注意

如果打开串口打印功能,则待机流程走到MCU退出时就无法继续运行,所以串口打印 仅适用于调试MCU中待机代码。

3.14.2 LED 数码管显示

操作步骤如下:

步骤1 查看debug参数。

cat /proc/msp/pm

步骤2 设置debug参数:取值范围为0(表示不使能调试功能),1(表示将调试信息存到MCU内存中),2(表示将调试信息显示到前面板的LED上),3(表示同时将调试信息保存到MCU内存中和显示到前面板的LED上),4(表示将调试信息输出到串口)。

echo debug=2 > /proc/msp/pm

步骤3 在PMOC代码中,调用函数dbg val()把需要显示的数据显示到LED上。



注意

只有进入待机时使能KEYLED模块之后,退出待机关闭KEYLED模块之前,LED才能显示调试信息。

----结束

3.14.3 将 Debug 信息保存到内存中

在PMOC代码中调试时,将debug信息写入到内存区域,唤醒后再查看debug信息,这种调试方法至少需要1种能够成功唤醒的方式,比如timer、红外唤醒方式等。

操作步骤如下:

步骤1 查看debug参数。

cat /proc/msp/pm

步骤2 设置debug参数。

echo debug=1 > /proc/msp/pm

步骤3 在PMOC代码中,调用dbg val()把调试信息保存到内存中去。

步骤4 待机唤醒后,用himd.l命令直接查看相关内存地址的值。



注意

保存debug信息的内存地址在dbg_val()函数中支持动态修改,用户可以根据实际芯片情况使用MCU的RAM区域或者主芯片的SRAM区域,但需要保证不与待机代码和参数存储区域冲突。

----结束

3.14.4 待机时内核相关驱动调试方法

如果在待机时,没有进入MCU或者从MCU退出后,程序出现了问题,可以打开内核的 待机相关的打印,来进一步判断问题出在哪里。

具体需要修改:

- 修改source/kernel/linux-3.18.y/kernel/printk/printk.c中的变量初始化为false, bool console_suspend_enabled = false;
- 修改source/kernel/linux-3.18.y/include/linux/suspend.h中的宏定义,设置为true, #define pm_print_times_enabled (true)

然后重新编译更新内核就可以了。

3.15 gprof 使用方法

步骤1 gprof工具是linux下一个程序分析工具,能够以"日志"的形式记录程序运行时的统计信息:程序运行中各个函数消耗的时间和函数调用关系,以及每个函数被调用的次数

等。利用gprof工具可以帮助分析最耗时的函数和程序的运行流程等。gprof工具的使用方法如下:

步骤2 编译程序时使用-pg选项(这样可以打印出函数的调用关系),生成可执行程序。

步骤3 运行该可执行程序,生成gmon.out文件,该文件包含了profiling的数据。

步骤4 使用gprof工具分析gmon.out文件以调试程序(以arm-hisiv200-linux-gprof工具为例)。
arm-hisiv200-linux-gprof prog gmon.out > profile.txt //prog为使用-pg编译出来的可执行程序

该命令的含义是将profiling的结果输出到profile.txt文件中,这样就可以查看profiling的结果了。

关于gprof的工具的更多参数的含义,请使用arm-hisiv200-linux-gprof --help查看。

----结束