

Rockchip 安卓摄像头问题排查指南

文档标识: RK-PC-YF-230

发布版本: V1.3.0

日期: 2024-07-12

文件密级: ☐绝密 ☐秘密 ☐内部资料 ☒公开

免责声明

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

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

商标声明

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

本文档可能提及的其他所有注册商标或商标，由其各自拥有者所有。

版权所有 © 2024 瑞芯微电子股份有限公司

超越合理使用范畴，非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd.

地址: 福建省福州市铜盘路软件园A区18号

网址: www.rock-chips.com

客户服务电话: +86-4007-700-590

客户服务传真: +86-591-83951833

客户服务邮箱: fae@rock-chips.com

前言

概述

本文档主要介绍Rockchip Camera模块的常见问题分析。

产品版本

芯片名称	内核版本	Android版本
RK3126c	Linux 4.4	Android9.0
RK3326/PX30	Linux4.4/Linux 4.19/Linux 5.10/Linux6.1	Android9.0/Android10.0/Android11.0/Android12.0/Android13.0/Android14.0
RK3368/PX5	Linux4.4/Linux 4.19	Android9.0/Android11.0/Android12.0
RK3288	Linux4.4/Linux 4.19	Android9.0/Android11.0/Android12.0
RK3399	Linux4.4/Linux 4.19/Linux 5.10/Linux6.1	Android9.0/Android10.0/Android11.0/Android12.0/Android13.0/Android14.0
RK3566/RK3568	Linux 4.19/Linux 5.10/Linux6.1	Android11.0/Android12.0/Android13.0/Android14.0
RK3588	Linux 5.10/Linux6.1	Android12.0/Android13.0/Android14.0
RK3562	Linux 5.10/Linux6.1	Android13.0/Android14.0
RK3576	Linux6.1	Android14.0

读者对象

本文档（本指南）主要适用于以下工程师：

技术支持工程师

软件开发工程师

修订记录

日期	版本	作者	修改说明
2020-02-06	V1.0.0	王潘祯撰	发布初始版本
2020-07-31	V1.0.1	王潘祯撰	1、新增MCLK异常说明 2、新增马达及闪光灯配置注意事项 3、新增调试案例分析
2021-02-22	V1.1.0	王潘祯撰	1、新增rk356x平台参考文档说明 2、新增rk356x平台参考配置说明
2021-04-10	V1.1.1	王潘祯撰	1、新增rk356x平台AHD转MIPI-4路VC芯片NVP6324芯片配置说明 2、新增rk356x平台AHD转BT1120-4路VC芯片NVP6158芯片配置说明
2021-05-06	V1.1.2	王潘祯撰	新增rk356x平台camera CTS测试失败案例分析及说明
2021-06-07	V1.1.3	王潘祯撰	新增rk356x平台单目及双目摄像头dts注册及时钟配置说明
2022-12-25	V1.1.4	王潘祯撰	新增rk356x平台双摄支持HAL修改及双摄开关拷机问题分析方法
2023-04-11	V1.1.5	王潘祯撰	新增rk3588平台dts配置及部分案例分析说明
2024-07-06	V1.2.0	王潘祯撰	1、新增rk3562及rk3576平台dts配置说明 2、更新完善目录标题 3、新增部分案例说明
2024-07-12	V1.3.0	王潘祯撰、 陈潇	1、新增第三方应用打开相机倒转180度排查说明

Rockchip 安卓摄像头问题排查指南

目的

适用范围

SDK已有Camera文档说明

- [camera_hal3_user_manual_vx.x.pdf](#)
- [camera_engine_rkisp_user_manual_vx.x.pdf](#)
- [Rockchip_Development_Guide_ISP2x_CN_v1.x.x.pdf](#)
- [RKCIF_Driver_User_Manual_v1.xx.pdf](#)
- [RKISP_Driver_User_Manual_v1.xx.pdf](#)
- [Rockchip_Driver_Guide_ISP2x_CN_v1.x.x.pdf](#)
- [Rockchip_Developer_Guide_Android_Camera_Of_Multichannel_CN.pdf](#)
- [Rockchip_Driver_Guide_VI_CN_vx.x.x.pdf](#)
- [Rockchip_Tuning_Guide_ISPxx_CN.pdf](#)

本文档位置：

FAQ

各平台芯片ISP版本区分

新增camera驱动的调试步骤说明

调试一个新增camera所需准备资料：

所需基础知识：

调试驱动步骤：

版本查询

[rkci驱动版本查询](#)

[rkisp1驱动版本查询](#)

[rkisp驱动版本查询](#)

[camera Hal3版本查询](#)

[camera_engine_rkisp版本查询](#)

[查看3A库版本号](#)

[Android10.0后查看上述版本号方法](#)

[camera_engine_rkaiq版本查询](#)

[rkisp2x工作模式查询](#)

日志

[camera hal3的LOG抓取](#)

[camera_engine_rkisp的LOG抓取](#)

[camera_engine_rkaiq的LOG抓取](#)

[rkci控制器驱动LOG抓取](#)

[rkisp1控制器驱动的LOG抓取](#)

[rkisp2.x控制器驱动的LOG抓取](#)

[v4l2框架的LOG抓取](#)

[uvc框架的LOG抓取](#)

[aw36518闪光灯驱动LOG抓取](#)

[cn3927v马达驱动LOG抓取](#)

[dw9800v马达驱动LOG抓取](#)

[如何配置camera3_profiles.xml文件?](#)

[新增回读/多摄模式配置](#)

[新增HDR模式配置](#)

[各平台Camera示例dts配置](#)

[rk3126c平台](#)

[rk3288w平台](#)

[rk3326平台](#)

[rk3368平台](#)

[rk3399平台](#)

[RK3399摄像头使用cif_clkoutb配置说明](#)

[修改方法:](#)

[rk356x平台摄像头DTS注册](#)

[单目摄像头使用MIPI-4lane情况](#)

[单目摄像头使用MIPI 1/2lane情况](#)

[双目使用MIPI-RAW+DVP-SOC情况](#)

[双目使用MIPI-RAW+MIPI-RAW情况（即2lane+2lane）](#)

[RK356x-MIPI-CSI-DPHY配置注意事项](#)

[上面dts具体配置参考文档](#)

[RK356x-CIS时钟配置注意事项](#)

[RK356x-CIS各个时钟支持的输出频率](#)

[rk3588平台摄像头DTS注册](#)

[RK3588-目前各个样机使用的摄像头](#)

[dchpy0-单ISP-8M摄像头配置参考](#)

[dhpy0-单ISP-8M摄像头配置参考](#)

[48M-摄像头使用ISP联合模式配置参考](#)

[RK3588-CIS时钟配置注意事项](#)

[RK3588-CIS各个时钟支持的输出频率](#)

[rk3562平台摄像头DTS注册](#)

[RK3562-目前各个样机使用的摄像头](#)

[dhpy0-接8M+5M前后摄像头切换配置参考](#)

[dhpy0-dphy3分别接摄像头配置参考](#)

[双目摄像头配置参考](#)

[13M-摄像头使用配置参考](#)

[RK3562-CIS时钟配置注意事项](#)

[rk3576平台摄像头DTS注册](#)

[RK3576-目前各个样机使用的摄像头](#)

dchpy0-单ISP-8M摄像头配置参考
dphy3-16M 和dchpy0 -5M摄像头前后切换配置参考
rk3576 中dphy/dcphy到mipi_csi 到rkcif_mipi_lvds*的链接关系
RK3576- CIS时钟配置注意事项
RK3576-各个时钟支持的输出频率

Camera 拓扑链路查看

Camera Selinux权限

权限问题确认
权限文件板上位置
权限文件在sdk中位置

马达

如何查看马达驱动是否注册？
如何使用V4l2工具控制马达移动？
camera3_profiles.xml如何配置？
效果文件中马达对焦如何配置？
示例案例：无马达OV2680模组配置错误导致切换到录像卡住问题
 问题描述
 问题分析
 解决办法
注意事项

闪光灯

如何查看闪光灯驱动是否注册？
如何使用V4l2工具控制闪光灯模式？
camera3_profiles.xml如何配置？
示例案例： camera flash如何配置默认关闭？
 问题描述
 问题分析
 解决办法
注意事项

I2C不通，如何确认相关信号？

测量Power情况
 异常分析

PowerDown
 异常分析

Reset
 异常分析

Mclk
 MCLK异常分析
 RK3566 –Android11.0-MCLK测量没有

I2cBus
 异常分析

如何使用v4l2获取底层数据流？

rkisp1配置链路 示例
 MIPI-RAW-Sensor -> MIPI interface->ISP
 MIPI-YUV-Sensor -> MIPI interface->ISP
 Parallel-Sensor ->ISP
 Parallel-Sensor ->CIF

配置各entity格式
 MIPI-RAW-Sensor -> MIPI interface->ISP
 MIPI-YUV-Sensor -> MIPI interface->ISP
 MParallel-Sensor ->ISP
 MParallel-Sensor ->CIF

使用V4l2命令获取数据流
v4l2-ctl及media-ctl工具说明

v4l2命令获取数据流失败问题常见原因及调试方法

可能原因1： DVP接口相关的IOMUX未进行配置
 问题现象
 调试方法

解决办法

可能原因2: 模组供电少了

问题现象

调试方法

解决办法

可能原因3: 驱动中配置的MIPI速率不对

问题现象

调试方法

尝试解决办法

可能原因4: 模组的MIPI CLK P/N或data P/N 和主板的P/N反了;

问题现象

调试方法

解决办法

常见更新SDK后, 摄像头打不开问题分析排查步骤

先将camera3_profiles.xml中 先改成SOC类型

如果还报错, 先重启, 然后使用v4l2-ctl工具底层抓数据流,看下是否正在正常

如何判断驱动中摄像头配置是RAW还是SOC类型?

如何查看驱动注册后各video/v4l-subdev结点名称信息?

查看video结点名称信息:

v4l-subdev结点名称信息

RK356x-接DVP-RAW要求注意事项

如果接isp注意如下限制

RK356x dvp-raw接isp, sensor输出改成高有效

RK356x-BT1120-4路VC芯片NVP6158-DTS配置说明

完整示例配置

各部分配置说明

如下部分是和camera相同的配置

nvp6158驱动私有配置说明

BT656和BT1120配置区别

BT656一路I格式输出配置示例

分辨率: 720x480I

分辨率: 720x576I

分辨率: 960x576I

分辨率: 1920x576I

BT656一路P格式输出配置示例

分辨率: 1280x720P

分辨率: 1920x1080P

BT656-2路无支持格式

BT656-4路无支持格式

BT1120一路P格式输出配置示例

分辨率: 1280x720P

分辨率: 1920x1080P

BT1120-2路P格式输出配置示例

分辨率: 1280x720P只支持只能识别一路, 两路相位差180度

分辨率: 1920x1080P

BT1120-4路P格式输出配置示例

分辨率: 1280x720P

分辨率: 1920x1080P

注意: 以上均是color Bar输出, 接AHD的只验证过1080P;

RK356x-BT1120-4路VC芯片NVP6158-驱动配置注意事项

RK356x-MIPI-4路VC芯片NVP6324-DTS配置说明

RK356x-11.0-AHD-MIPI多路驱动调试问题

示例配置

N4驱动私有配置说明

RK356x-多路VC芯片常见问题

RK356x-11.0-camera如何打开HDR?

确认CMOS sensor是否本身支持HDR模式?

如何配置camera3_profiles.xml支持HDR?

RK356x-11.0-AHD to DVP接口常见转换芯片

录像相关常见问题

跟录像相关的文件？

如何修改录像分辨率？

Android-9.0-13.0版本的USB camera常见问题咨询

Android9.0及以上版本的USB camera代码位置：

sdk中添加usb camera支持的参考文档：

usb camerahal 的log打开支持：

如何配置USB camera为前后置？

RK3566-11.0-如何旋转usb camera角度

如何查看usb camera的格式及分辨率

Android 14.0重启usb camerahal provider命令

Android 14.0如何抓取iep deinterlaced前后数据

PDAF功能调试相关

先确认CMOS sensor本身是否支持PD像素使用mipi virtual channel输出；

支持pdaf的dtsi所需配置

底层确认pd像素输出

pdaf上层配置及调试支持

上层pdaf像素未抓取到问题排查步骤

RK3576支持OV13855

Rockchip OTP常见问题

CMOS sensor添加OTP支持步骤

如何查看模组是否有正确按照ROCKCHIP规范烧录OTP信息

RK3588-12.0-多路帧同步功能调试

拍照问题

如何区分使用的是硬编还是软编

Camera MIPI报错问题排查

SDK自带驱动更新SDK后打不开问题排查步骤

示例IQ文件定义规则

IQ场景参数切换

添加支持场景切换示例修改

Camera第三方应用，方向兼容性问题

相机360/小红书，调用前置/后置摄像头图像180°倒置问题排查

问题现象

排查步骤

示例解决方法

qq/微信等第三方应用 摄像头方向问题

RK356x-11.0-双摄问题汇总

RK356x-11.0-R9/R10/R11版本双摄同时打开异常所需补丁包

补丁包位置：

rk356x-双摄链路组合：

rk356x-11.0-双目摄像头调试底层常见问题

双目摄像头示例链路连接

如何确定链路正确

使用 `grep -H " /sys/class/video4linux/video*/name"`

使用如下两个命令同时抓取两个结点的数据流，如果正常，说明底层通路OK

rk356x-11.0-双目摄像头调试上层常见问题

上层未识别到摄像头，使用APK打不开摄像头

上层识别到摄像头，使用APK打不开摄像头

rk3566-rkisp2x 2lane+2lane双mipi硬件连接注意事项

Android9.0调试案例

SDK自带驱动，使用 `v4l2-ctl`命令获取数据流出现kernel崩溃问题

问题描述

问题分析

解决办法

RK3126C-9.0-GC0312-预览必现分屏显示问题

问题描述

问题分析

解决办法

RK3126C-9.0-GC2145-预览概率性分屏显示问题

问题描述

问题分析

解决办法

RK3126C-9.0-GC2145录像帧率只有20fps左右，不到30fps问题

问题描述

问题分析

解决办法

RK3326-9.0-GC2145烧完机后进camera概率性报错问题

问题描述

问题分析

解决办法

RK3326-9.0-打开camera预览界面概率性出现绿屏

问题描述

问题分析

解决办法

RK3326-9.0-mipi camera预览卡顿问题

问题描述

问题分析

解决办法

RK3326-9.0-支持同时打开两路camera(Defect #258741)

问题描述

问题分析

解决办法

RK3326-9.0-SOC摄像头录像过程拍照，录像视频拍照时间点会闪一下问题

问题描述

问题分析

跟踪CameraHal代码

尝试获取：ANDROID_CONTROL_CAPTURE_INTENT的entry看看

尝试解决办法 Solution

RK3326-9.0-打开补光灯录像模式，重新进入camera app会闪一下灯再常亮补光灯(Defect ## 254851)

问题描述

问题分析

解决办法

RK3288-9.0/10.0-IMX327/214摄像头拍照及录像预览闪绿屏问题

问题描述

问题分析

先定位是不是hwc问题:

使用v4l2-ctl工具抓取原始数据查看

对比客户APP预览和原生APP预览的pipeline配置差异

解决办法

Android9.0 CaptureRequestTest#testEdgeModeControl 失败问题

问题描述

问题分析

解决办法

RK3399-9.0-IMX214调试中binning分辨率2104x1560拍照，一拍照相机APK立马退出问题

问题描述

问题分析

解决办法

RK3399-9.0-IMX214-配置为RAW之后，预览出现后马上卡死问题

问题描述

问题分析

解决办法

RK3399-9.0-IMX214-4096x3120分辨率拍照竖条纹问题

问题描述

问题分析

解决办法

RK3399-9.0-OV8858-4lane录像1080P帧率只有15FPS问题

问题描述

问题分析

解决办法

Android10.0调试案例

RK3126C-10.0-GC0312-预览必现分屏显示问题

问题描述

问题分析

7.1.2 解决办法

RK3326-10.0-OV2680拍照预览帧率一直为30FPS问题

问题描述

问题分析

解决办法

RK3326-10.0-MIPI和DVP摄像头兼容配置问题

问题描述

问题分析

解决办法

RK3326-10.0-camera-GTS错误

问题描述

解决办法

RK3326-10.0-camera-vts错误

问题描述

问题分析

解决办法

Android10.0-CTS和GSI CaptureRequestTest#testFlashTurnOff失败问题

问题描述

问题分析

解决办法

RK3326-10.0接后摄MIPI口GC2145，前摄DVP口GC032A出现切换报错

问题描述

问题分析

解决办法

RK3326-10.0-DVP和MIPI摄像头都兼容问题

问题描述

问题分析

解决办法

Android11.0调试案例

RK3566/68- android11.0 如何配置MIPI YUV camera问题

问题描述及问题解答

使用的MIPI摄像头，在RK3399-7.1上配置的是CamSys_Fmt_Yuv422_8b，在rk3566上应该怎么配置呢？

有没文档或类似的例程？

是否有参考的dts配置？

camera_hal3_user_manual_v2.3.pdf这个文档提到的 RAW sensor 及SOC sensor 其中SOC sensor类型就是YUV输出的sensor吗？

dts中，这些name 要与那些代码或配置保持一致？

不能获取到数据大概是什么原因，模组测量是有数据输出的？

RK356x-android11.0-gc5035-rkisp0: MIPI error: packet: 0x01000000报错问题

问题描述

问题分析

解决办法

RK356x-android11.0-ov8858-rkisp0: MIPI error: overflow: 0x00000001报错问题

问题描述

问题分析

解决办法

RK356x-11.0-Camera配置为RAW之后，预览出现后马上卡住问题

问题描述

问题分析

解决办法

RK356x-11.0- CTS和GSI CaptureRequestTest#testFlashTurnOff[1]失败问题

问题描述

问题分析

解决办法

RK356x-11.0- CTS和GSI CaptureRequestTest#testFlashControl[1]失败问题

问题描述

问题分析

解决办法

RK356x-11.0- CTS FlashlightTest失败问题

问题描述

问题分析

解决办法

RK356x-11.0 -CTS StaticMetadataTest#testCapabilities[1]失败问题

问题描述

问题分析

解决办法

RK3126-11.0-2G-rgo版单前摄2145-CTS的CtsAppTestCases失败问题

问题描述

问题分析

解决办法

RK356x-11.0-AHD to DVP接口转换芯片报错调试建议

问题描述

问题分析

RK356x-11.0-开启双摄支持修改

问题描述

解决办法

RK356x-11.0-R6版本SOC类型的camera点不亮问题

问题描述

问题分析

解决办法

RK356x-11.0-更新到R8后SOC-DVP接口的camera用不了

问题描述

问题分析

解决办法

RK356x-11.0- some iq features are not supported报错

问题描述

问题分析

RK356x-11.0-双摄开关拷机出现卡住黑屏问题

问题描述

问题分析

分析1:

分析2:

分析定位3

解决办法

RK356x-11.0-有关双摄更新代码打不开排查流程

RK356x-11.0-更新到R9后SOC-DVP接口的camera点击拍照录像异常问题

问题描述

问题分析

解决办法

RK356x-11.0用2lane+2lane mipi yuv摄像头报错问题

问题描述

问题分析

解决办法

RK356x-11.0-sensor->csi2_dphy2->mipi_csi2->vicap->isp_vir1链路兼容多个sensor问题

问题描述

问题分析

解决办法

RK3326-11.0-raw-sensor打开闪光灯拍照卡住问题

问题描述

问题分析

问题原因:

解决办法

RK3326-11.0-gc02m2切到gc032a概率报错

问题描述

问题分析

调试修改方法

解决办法

Android12.0调试案例

RK3566-12.0-更新贵司SDK后，后摄预览不停的对焦

解决办法

RK3588-12.0-R10版本拍照报错重启问题

解决办法

RK3588-12.0-R10版本预览手指晃动，会在运动区域出现红黄蓝的条状物

问题描述

解决办法

Android-12.0-版本HDR出现休眠唤醒sensor打开报错问题

解决办法

RK3588S-android 12-R12版本前摄休眠唤醒后预览闪烁

问题描述

问题分析

解决办法

RK3588-12.0-R12版本进入相机预览会闪一下绿屏问题

问题描述

问题分析

解决办法

RK3588-12.0-R12版本摄像头imx415预览分层问题

问题描述

问题分析

解决办法

RK3588-12.0-R12版本修改selfpath裁图的起始位置不生效问题

问题描述

问题分析

解决办法

RK3588-12.0-R12版本多摄打开，关闭其中一个，其他所有都卡住问题

问题描述

问题分析

解决办法

RK3588-12.0-R12版本-GMS-Camera VTS失败问题

问题描述

解决办法

RK3588-12.0-R12-系统相机打开正常，微信前置相机拍照预览很黑

问题描述

问题分析

解决办法

Android13.0调试案例

RK3562-13.0-R5版本-CTSV-R4 ITS-scene4和scene6曝光严重导致fail问题

问题描述

解决办法

RK3562-13.0-R5版本-vicap接两路4lane-mipi，打开一路后再打开一路出现mipi-csi2 ERR2:0x10000000报错

问题描述

解决办法

RK3562-13.0-R5版本-vicap出现Failed to create dummy_buf问题

问题描述

解决办法

目的

本文介绍RK平台上已有的调试文档，总结以往处理Camera中相关问题的步骤和方法，作为后续分析和处理Camera调试过程中遇到问题的参考。

适用范围

目前RK产品的各Android平台中使用的Camera Hal，分为 2 两个大版本：

- Camera Hal1在Android8.1及以下版本中使用
- Camera Hal3在Android9.0及以上版本中使用

本文档仅适用于Android平台上，Camera Hal3及对应新框架的Camera Sensor驱动常见问题分析排查和处理方法。

SDK已有Camera文档说明

文档位置：

- SDK/RKDocs/common/camera/HAL3
- <https://redmine.rock-chips.com/documents/53>

camera_hal3_user_manual_vx.x.pdf

文档介绍RK CameraHal3框架、camera3_profiles.xml文件配置、HAL层数据DUMP及部分调试案例。对应代码目录：SDK/ hardware/rockchip/camera

camera_engine_rkisp_user_manual_vx.x.pdf

文档介绍RK CameraHal3 Engine框架、API简要说明及IQ效果文件相关的规则等。

对应代码目录：SDK/ hardware/rockchip/camera_engine_rkisp

Rockchip_Development_Guide_ISP2x_CN_v1.x.x.pdf

文档介绍RKISP2版本，RKAIQ架构，所处软件框架位置、软件流程、API调用详细说明等。

对应代码目录：SDK/external/camera_engine_rkaiq

RKCIF_Driver_User_Manual_v1.xx.pdf

文档介绍RKCIF驱动框架、DTS配置、驱动中部分数据类型及API简介。

对应代码目录：SDK/kernel/drivers/media/platform/rockchip/cif

RKISP_Driver_User_Manual_v1.xx.pdf

文档介绍RKISP1驱动框架、CIS(CMOS IMAGE SENSOR)驱动、马达驱动、闪光灯驱动、RK1608 AP驱动的DTS配置、部分数据类型及API简要说明和FAQ。

RKISP1驱动对应代码目录：SDK/kernel/drivers/media/platform/rockchip/isp1

Rockchip_Driver_Guide_ISP2x_CN_v1.x.x.pdf

文档介绍RKISP2驱动框架、CIS(CMOS IMAGE SENSOR)驱动、马达驱动、闪光灯驱动的DTS配置、关键数据类型及关键API说明、驱动移植步骤和FAQ。

RKISP2驱动对应代码目录：SDK/kernel/drivers/media/platform/rockchip/isp

Rockchip_Developer_Guide_Android_Camera_Of_Multichannel_CN.pdf

文档介绍RK356X等多路摄像头框架逻辑及常用AHD转多路芯片配置。

多路摄像头文档位置如下：

<https://redmine.rockchip.com.cn/documents/53>中

Rockchip_Developer_Guide_Android_Camera_Of_Multichannel_CN.pdf

Rockchip_Driver_Guide_VI_CN_vx.x.x.pdf

VI 驱动开发文档，本文旨在描述RKISP（Rockchip Image Signal Processing）模块的作用，整体工作流程，及相关的API接口。主要给驱动工程师调试Camera提供帮助。该文档是RKCIF_Driver_User_Manual_v1.0.pdf和RKISP_Driver_User_Manual_v1.3.pdf文档的升级版，同时介绍RK356x、RK3588、RK3562、RK3576等新平台相关dts配置及驱动调试说明。

文档位置：

SDK\RKDocs\common\camera\HAL3\Rockchip_Driver_Guide_VI\CN\
Rockchip_Driver_Guide_VI_CN_vx.x.x.pdf

Rockchip_Tuning_Guide_ISPxx_CN.pdf

IQ调试文档，本文档的第一章主要讲解了ISP调优过程中涉及的文档关系说明，第二章对ISP进行了系统概述，包括ISP的功能框图及各模块简介；第三章主要介绍了整个图像调优过程的操作步骤及注意事项；第四章之后开始分模块介绍各子模块的调试方法。

文档位置：

SDK\RKDocs\common\camera\HAL3\Rockchip_Tuning_Guide_ISP

本文档位置：

<https://redmine.rockchip.com.cn/documents/53>

中Rockchip_Trouble_Shooting_Android_CameraHal3_CN-Vxxx.pdf 文档。

或者SDK\RKDocs\common\camera\HAL3\Rockchip_Trouble_Shooting_CameraHAL3_CN_EN.pdf

FAQ

各平台芯片ISP版本区分

目前使用的soc平台ISP版本区分如下表：

SOC	ISP版本
RK3288	ISP_V1.0
RK3368	ISP_V1.0
RK3399	ISP_V1.0
RK3326	ISP_V1.0
RV1126	ISP_V20
RK3566/68	ISP_V21
RK3588	ISP_V30
RV1106	ISP_V32
RK3562	ISP_V32_lite
RK3576	ISP_V39

新增camera驱动的调试步骤说明

调试一个新增camera所需准备资料：

硬件原理图：

1. 板子原理图及PCB位图

Camera Module:

2. 模组规格书；（查看接口情况）
3. 镜头规格书；（后续调效果使用）
4. Camera Sensor Datasheet；（查相关寄存器说明）
5. Camera Sensor Application note；（查曝光相关gain和exposure time使用说明）
6. Camera Sensor 寄存器配置序列;(寄存器配置)
7. 如有其它平台调试过该sensor，获取其他平台驱动以供参考；

如果支持VCM，需有以下资料

8. 马达驱动IC资料
9. 马达规格书
10. 马达振荡曲线数据

如果支持OTP，需有以下资料

11. OTP读取规范
12. OTP数据

所需基础知识：

1. 读懂原理图和PCB图
2. 能阅读器件的Datasheet文档
3. 了解常用图像格式

调试驱动步骤：

1. 参考相同厂商，相近分辨率的sensor驱动移植

主要修改文件名称及一些寄存器相关的宏

2. 参考已有的dtsi或Rockchip_Driver_Guide_VI_CN_v1.x.x.pdf配置dts

配置dts，需要自行根据本身板子硬件原理图情况，修改sensor的供电、MCLK/PWDN/RESET/I2C等配置；

3. 初步配置好后，开始调通I2C

在I2C不通情况下，在check_id失败的地方加while(1)；

然后硬件测量sensor的供电、MCLK、PWDN、RESET、I2C对比Datasheet排查哪些信号不对；

4. 调通I2C后，先将驱动中xxxx_set_ctrl下面写寄存器值的操作屏蔽

这样保证先使用默认的寄存器值，进行底层数据流的调试；因为可能参考移植的驱动寄存器和现在使用的不一样；

5. 参考章节4.11，使用v4l2-ctl工具抓取数据流看是否正常

如果不正常，回过头排查dts配置是否正确，如果确认正常，就需进一步排查硬件；
可以尝试多换几块板子或者模组尝试；

6. 将4中屏蔽的地方打开，调试线性及HDR模式的曝光和增益控制

曝光和增益控制逻辑可以参考其他平台的驱动或sensor原厂提供的application note

初步修改后再用v4l2-ctl抓取不同增益和曝光值的RAW数据；

使用RAW图查看工具，查看亮度是否跟随增益和曝光值线性变换；

7. 再次核查确认sensor各个功能的寄存器是否正确

8. 底层v4l2-ctl工具抓取数据流正常后，下一步配置camera3_profiles.xml

先将camera类型配置为soc（参考文档说明），点亮，然后将camera类型配置为RAW类型；

logcat出现/vendor/etc/camera/rkisp2/目录出现报错，说明缺少文件，则需要找我司对接业务申请效果调试；

版本查询

rkcif驱动版本查询

rkcif驱动获取版本号命令：

```
cat /sys/module/video_rkcif/parameters/version
```

rkisp1驱动版本查询

适用平台：RK3288、RK3399、RK3326、RK3368

rkisp1驱动获取版本号命令：

```
cat /sys/module/video_rkisp1/parameters/version
```

示例如图：



```
at /sys/module/video_rkisp1/parameters/version
v00.01.01
```

图 1 rkisp1驱动版本号

rkisp驱动版本查询

适用平台：RK3566/68、RK3588、RK3562、RK3576等

rkisp驱动获取版本号命令：`cat /sys/module/video_rkisp/parameters/version`

```
rk3562_ugo:/ # cat /sys/module/video_rkisp/parameters/version
v02.03.00
```

camera Hal3版本查询

获取版本号命令：

```
getprop | grep cam.hal3.ver
```

```
* daemon started successfully *
rk3326_mid:/ # getprop | grep cam
[init.svc.cameraserver]: [running]
[init.svc.vendor.camera-provider-2-4]: [running]
[persist.vendor.camera.debug.logfile]: [0]
[ro.boottime.cameraserver]: [16070770960]
[ro.boottime.vendor.camera-provider-2-4]: [7294072078]
[vendor.cam.hal3.ver]: [v1.9.0]
rk3326_mid:/ #
```

图 2 Camera Hal3版本号

camera_engine_rkisp版本查询

适用平台：RK3288、RK3399、RK3326、RK3368

获取版本号命令：

```
strings /vendor/lib/librkisp.so | grep v1. 或grep v2.
```

```
rk3368:/ #
strings /vendor/lib/librkisp.so | grep v1
_ZTUN10_cxxabi0117__class_type_infoE
_ZTUN10_cxxabi0120__si_class_type_infoE
_ZTUN10_cxxabi0121__vni_class_type_infoE
v1.9.0
```

图 3 librkisp版本号

查看3A库版本号

获取版本号命令：

```
strings /vendor/lib/rkisp/ae/librkisp_aec.so | grep v0
```

```
strings /vendor/lib/rkisp/af/librkisp_af.so | grep v0
```

```
strings /vendor/lib/rkisp/awb/librkisp_awb.so | grep v0
```

```
generic_arm64:/ # strings /vendor/lib/rkisp/ae/librkisp_aec.so | grep v0
v0.0.9
generic_arm64:/ #
generic_arm64:/ # strings /vendor/lib/rkisp/af/librkisp_af.so | grep v0
v0.2.10
generic_arm64:/ # strings /vendor/lib/rkisp/awb/librkisp_awb.so | grep v0
v0.0.9
```

图 4 3A库版本号

Android10.0后查看上述版本号方法

首先确保：SDK/hardware/rockchip/camera_engine_rkisp目录包含如下提交：

```
commit 41670de2aa53f72d00eae837ecd1275e93fd7182
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date: Tue Sep 3 10:32:03 2019 +0800

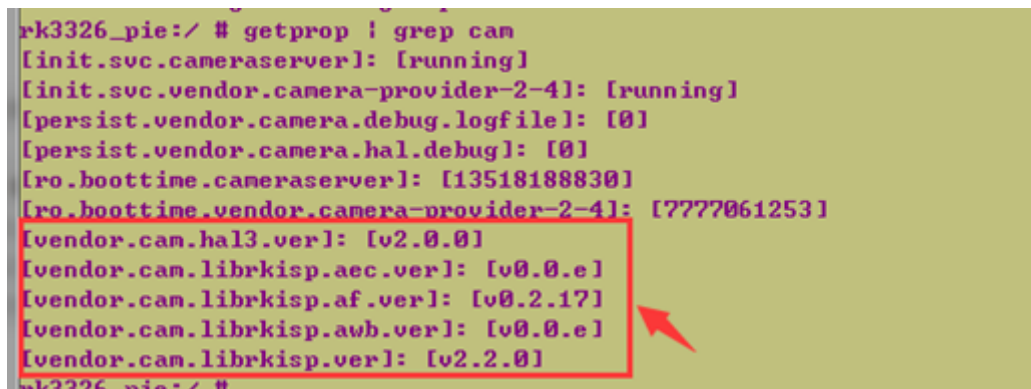
    add properties to show rkisp & 3A lib version

Change-Id: Iccfe8a0fc773489eb9a327d1ba192dc572c4c89f
Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
```

需要使用su权限，获取版本命令：

```
getprop | grep cam
```

结果如图：



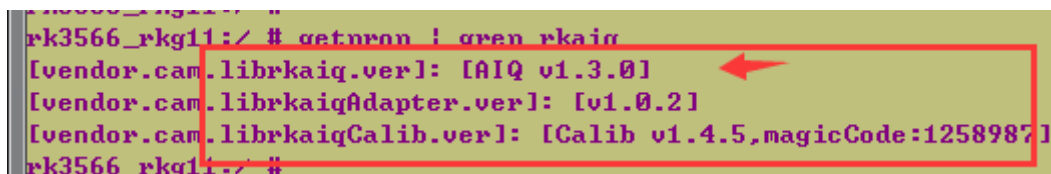
```
rk3326_pie:/ # getprop | grep cam
[init.svc.cameraserver]: [running]
[init.svc.vendor.camera-provider-2-4]: [running]
[persist.vendor.camera.debug.logfile]: [0]
[persist.vendor.camera.hal.debug]: [0]
[ro.boottime.cameraserver]: [13518188830]
[ro.boottime.vendor.camera-provider-2-4]: [7777061253]
[vendor.cam.hal3.ver]: [v2.0.0]
[vendor.cam.librkisp.aec.ver]: [v0.0.e]
[vendor.cam.librkisp.af.ver]: [v0.2.17]
[vendor.cam.librkisp.awb.ver]: [v0.0.e]
[vendor.cam.librkisp.ver]: [v2.2.0]
```

图 5 Hal3、ISP和3A库版本号

camera_engine_rkaiq版本查询

适用平台：RK3566/68、RK3588、RK3562、RK3576等

需要使用su权限，获取版本命令：getprop | grep rkaiq，结果如图：



```
rk3566_rkg11:/ # getprop | grep rkaiq
[vendor.cam.librkaiq.ver]: [AIQ v1.3.0]
[vendor.cam.librkaiqAdapter.ver]: [v1.0.2]
[vendor.cam.librkaiqCalib.ver]: [Calib v1.4.5,magicCode:1258987]
```

图 6 RKAIQ相关版本号

rkisp2x工作模式查询

查询是直通还是回读模式

使用命令：cat /proc/rkisp* | grep frame

Isp read 是回读模式，示例信息如下：

```
rk3588s_s:/ # cat /proc/rkisp* | grep frame
Isp Read    mode:frame1 (frame:1332 rate:33ms idle time:7ms frameloss:0)
cnt(total:1332 X1:1332 X2:-1 X3:-1)
Output      rkisp_mainpath Format:NV12 Size:3200x2400 Dcrop(0,0|3200x2400)
(frame:1328 rate:500ms delay:42ms frameloss:1242)
            module outform vertical counter(L:0 R:0), out frame counter:(L:0 R:0)
```

Isp online 是直通模式，示例信息如下：

```
rk3588s_s:/ # cat /proc/rkisp*
Isp online frame:254 working time:32ms v-blank:709us
Output      rkisp_mainpath Format:NV12 Size:3840x2160 Dcrop(0,0|3840x2160)
(frame:0 rate:0ms delay:0ms frameloss:255)
Output      rkisp_selfpath Format:NV12 Size:1920x1080 Dcrop(0,0|3840x2160)
(frame:254 rate:33ms delay:32ms frameloss:6)
```

日志

camera hal3的LOG抓取

打开camerahal的debug命令：

```
setprop persist.vendor.camera.hal.debug 5
```

关闭命令

```
setprop persist.vendor.camera.hal.debug 1
```

具体参考SDK下文档：

RKDocs\common\camera\HAL3\camera_hal3_user_manual_vx.x.pdf

```
.55: *.module.bit.maps.is.as.follow:
.56: *.
.57: *.bit:.....7-4.....3-0
.58: *.meaning:..[sub-modules].....[level]
.59: *.
.60: *.bit:.....15.....14.....13.....12.....11-8
.61: *.meaning:..[U].....[POOL].....[CAPTURE].....[FLASH].....[sub-modules]
.62: *.
.63: *.bit:.....23.....22.....21.....20.....19.....18.....17.....16
.64: *.meaning:..[U].....[U].....[U].....[U].....[U].....[U].....[U]
.65: *.
.66: *.bit:.....31.....30.....29.....28.....27.....26.....25.....24
.67: *.meaning:..[U].....[U].....[U].....[U].....[U].....[U].....[U]
.68: *.
.69: *.bit:.....39.....38.....37.....36.....35.....34.....33.....32
.70: *.meaning:..[U].....[U].....[U].....[U].....[U].....[U].....[U]
.71: *.
.72: *.bit:.....47.....46.....45.....44.....43.....42.....41.....40
.73: *.meaning:..[U].....[U].....[U].....[U].....[U].....[U].....[U]
.74: *.
.75: *.bit:.....[63-48]
.76: *.meaning:..[U]
.77: *.
.78: *.full maps unused now
```

打开camerahal3中FLASH的debug命令：

```
setprop persist.vendor.camera.hal3.debug 0x1ff5
```

打开CAPTURE的debug命令：

```
setprop persist.vendor.camera.hal3.debug 0x2ff5
```

打开POOL的debug命令:

```
setprop persist.vendor.camera.hal3.debug 0x4ff5
```

打开MESSAGE的debug命令:

```
setprop persist.vendor.camera.hal3.debug 0x8ff5
```

目前所有都打开:

```
setprop persist.vendor.camera.hal3.debug 0xffff5
```

关闭命令

```
setprop persist.vendor.camera.hal3.debug 0
```

camera_engine_rkisp的LOG抓取

打开camera_engine_rkisp的debug命令

```
setprop persist.vendor.rkisp.log 4
```

关闭命令

```
setprop persist.vendor.rkisp.log 0
```

如果SDK/hardware/rockchip/camera_engine_rkisp目录包含如下提交:

```
commit 050834ce2614f8533cdd774fbdf920fbbe910395
Author: ZhongYichong <zyc@rock-chips.com>
Date: Tue Jun 11 15:35:28 2019 +0800

    use new log system, support module logs
...
set debug level example:
    eg. set module afc log level to debug, and others to error:
    Android:
        setprop persist.vendor.rkisp.log 0x4000
    Linux:
        export persist_camera_engine_log=0x4000

Change-Id: I702bc8de878bef021f58de42b5ceb0d71d1a3439
Signed-off-by: ZhongYichong <zyc@rock-chips.com>
```

则debug开关打开命令如下:

```
setprop persist.vendor.rkisp.log 0x444444
```

```

bits:   31-28  27-24  23-20  19-16 15-12  11-8  7-4  3-0
module: [u]    [u]    [xcore] [ISP] [AF]   [AWB] [AEC] [NO]
        *[u] means unused now.
each module log has following ascending levels:
    0: error
    1: warning
    2: info
    3: verbose
    4: debug
    5: low1
    6-7: unused, now the same as debug
usage:
Android:

```

具体参考SDK下文档:

RKDocs\common\camera\HAL3\ camera_engine_rkisp_user_manual_vx.x.pdf

camera_engine_rkaiq的LOG抓取

打开camera_engine_rkaiq中所有模块的debug命令

```
setprop persist.vendor.rkisp.log 0x7fffffff6
```

```

/* use a 64 bits value to represent all modules bug level, and the
 * module bit maps is as follow:
 *
 *
 * \* bit:   7-4           3-0
 * \* meaning: [sub modules]           [level]
 *
 *
 * \* bit:   15    14    13    12    11-8
 * \* meaning: [ABLC]  [AF]  [AWB]  [AEC]  [sub modules]
 *
 *
 * \* bit:   23    22    21    20    19    18    17    16
 * \* meaning: [AGAMMA] [ACCM] [ADEBAYER] [AGIC] [ALSC] [ANR]  [ATMO]
[ADPCC]
 *
 *
 * \* bit:   31    30    29    28    27    26    25    24
 * \* meaning: [ASHARP] [AIE]  [ACP]    [ACSM] [ALDCH] [A3DLUT] [ADEHAZE] [AWDR]
 *
 *
 * \* bit:   39    38    37    36    35    34    33    32
 * \* meaning: [ADEGAMMA] [CAMHW] [ANALYZER] [XCORE] [ASD]  [AFEC]  [ACGC]  [AORB]
 *
 *
 * \* bit:   47    46    45    44    43    42    41    40
 * \* meaning: [U]      [U]      [GROUPAEC] [AWBGROUP] [CAMGROUP] [ACAC]  [AMD]
[AMERGE]
 *
 *
 * \* bit:   [63-48]
 * \* meaning: [U]
 *
 *
 * \* [U] means unused now.
 * \* [level]: use 4 bits to define log levels.
 * \* each module log has following ascending levels:
 * \*    0: error
 * \*    1: warning
 * \*    2: info
 * \*    3: debug
 * \*    4: verbose

```

```

\*      5: low1
\*      6-7: unused, now the same as debug
\* [sub modules]: use bits 4-11 to define the sub modules of each module, the
\*      specific meaning of each bit is decided by the module itself. These bits
\*      is designed to implement the sub module's log switch.
\* [modules]: AEC, AWB, AF ...
\*
\* set debug level example:
\* eg. set module af log level to debug, and enable all sub modules of af:
\* Android:
\*      setprop persist.vendor.rkisp.log 0x4ff4
\* Linux:
\*      export persist_camera_engine_log=0x4ff4
\* And if only want enable the sub module 1 log of af:
\* Android:
\*      setprop persist.vendor.rkisp.log 0x4014
\* Linux:
\*      export persist_camera_engine_log=0x4014
\*
*/

```

关闭命令

```
setprop persist.vendor.rkisp.log 0x00
```

只开librkisp库的log命令:

```
setprop persist.vendor.rkisp.log 0x1000000ff6
```

具体参考文档:

<https://redmine.rock-chips.com/documents/53>

Rockchip_Development_Guide_ISP2x_CN_v1.x.x.pdf

代码目录:

SDK/external/camera_engine_rkaiq

rkcif控制器驱动LOG抓取

通过如下命令打开debug开关:

```

echo 8 > /proc/sys/kernel/printk
echo 3 > /sys/module/video_rkcif/parameters/debug
echo 3 > /sys/module/video_rkcif/parameters/debug_csi2
或者修改SDK/kernel/drivers/media/platform/rockchip/cif/dev.c中int rkCIF_debug = 1;

```

还可以通过如下命令查看当前rkcif的状态:

```
cat /proc/rkcif_mipi_lvds
```

示例输出如下:

```

rk3568_s:/ # cat /proc/rkcif_mipi_lvds
Driver Version:v00.01.0a

```

```
Work Mode:ping pong
Monitor Mode:idle
aclk_cif:396000000
hclk_cif:198000000
dclk_cif:297000000
iclk_cif_g:0
Input Info:
    src subdev:m00_f_gc2093 4-007e
    interface:mipi csi2
    lanes:2
    vc channel: 0 1
    hdr mode: hdr_x2
    format:SRGBB10_1X10/1920x1080@25
    crop.bounds:(0, 0)/1920x1080
Output Info:
    format:RG10/1920x1080(0,0)
    compact:enable
    frame amount:51
    early:0 ms
    single readout:29 ms
    total readout:29 ms
    rate:40 ms
    fps:25
    irq statistics:
        total:102
        csi over flow:0
        csi bandwidth lack:0
        all err count:0
        frame dma end:102
```

RK3588为:

```
cat /proc/rkcif-mipi-lvds*
```

示例如下:

```
1|rk3588s_s:/ # cat /proc/rkcif-mipi-lvds*

Driver Version:v00.01.0a

Work Mode:ping pong

Monitor Mode:idle

aclk_cif:750000000

hclk_cif:198000000

dclk_cif:594000000

Driver Version:v00.01.0a

Work Mode:ping pong

Monitor Mode:idle

aclk_cif:750000000
```

```
hclk_cif:198000000

dclk_cif:594000000

Driver Version:v00.01.0a

Work Mode:ping pong

Monitor Mode:idle

aclk_cif:750000000

hclk_cif:198000000

dclk_cif:594000000

rk3588s_s:/ #
```

具体参考SKD下文档：

SDK\RKDocs\common\camera\HAL3\RKCIF_Driver_User_Manual_v1.xx.pdf

rkisp1控制器驱动的LOG抓取

通过如下命令打开debug开关：

```
echo 1 > /sys/module/video_rkisp1/parameters/debug
```

或者修改SDK/kernel/drivers/media/platform/rockchip/isp1/dev.c中int rkisp1_debug = 1;

具体参考SKD下文档：

SDK\RKDocs\common\camera\HAL3\RKISP_Driver_User_Manual_v1.xx.pdf

rkisp2.x控制器驱动的LOG抓取

通过如下命令打开debug开关：

```
echo 8 > /proc/sys/kernel/printk
echo 3 > /sys/module/video_rkisp/parameters/debug
```

关闭命令：

```
echo 0 > /sys/module/video_rkisp/parameters/debug
```

或者修改SDK/kernel/drivers/media/platform/rockchip/isp/dev.c中int rkisp_debug = 1;

还可以通过如下命令查看当前rkisp的状态：原理是添加了proc节点；

```
cat /proc/rkisp*
```

示例输出如下：

```
rkisp-vir0 Version:v01.06.00
clk_isp  500000000
aclk_isp  396000000
```

```
hclk_isp 198000000
Interrupt Cnt:1719 ErrCnt:0
Input rockchip-csi2-dphy0 Format:SBGGR10_1X10 Size:3264x2448@30fps Offset(0,0)
Isp online frame:610 working time:32ms
Output rkisp_mainpath Format:NV12 Size:3264x2448 (frame:331 rate:2166ms
delay:33ms)
Output rkisp_selfpath Format:NV12 Size:1280x960 (frame:610 rate:33ms
delay:33ms)
DPCC0 ON(0x40000005)
DPCC1 ON(0x40000005)
BLS ON(0x1)
SDG OFF(0x6197)
LSC ON(0x1)
AWBGAIN ON(0x6197) (gain: 0x01040104, 0x018c01af)
DEBAYER ON(0xf000111)
CCM ON(0xc0000003)
GAMMA_OUT ON(0xc0000001)
CPROC ON(0xf)
IE OFF(0x0) (effect: BLACKWHITE)
HDMTO OFF(0x0)
HDMGE OFF(0x0)
BAYNR ON(0xc0000101)
BAY3D ON(0x78010011)
YNR ON(0x801001)
CNR ON(0xc0000003)
SHARP ON(0x60000001)
GIC OFF(0x0)
DHAZ ON(0xc0111011)
3DLUT OFF(0x0)
LDCH OFF(0x0)
CSM FULL(0x6197)
SIAF OFF(0x0)
SIAWB OFF(0x0)
YUVAE OFF(0x0)
SIHST OFF(0x0)
RAWAF ON(0x7)
RAWAWB ON(0x7740d1f)
RAWAE0 ON(0x3)
RAWAE1 ON(0xf5)
RAWAE2 OFF(0x0)
RAWAE3 ON(0xf5)
RAWHIST0 ON(0x501)
RAWHIST1 ON(0x20000501)
RAWHIST2 OFF(0x0)
RAWHIST3 ON(0x20000501)
Monitor OFF Cnt:0
```

v4l2框架的LOG抓取

通过如下命令打开debug开关:

```
echo 7 > /sys/module/videobuf2_core/parameters/debug

kernel4.19改成:

echo 7 > /sys/module/videobuf2_v4l2/parameters/debug

echo 7 > /sys/module/videobuf2_common/parameters/debug
```

关闭debug

```
echo 0 > /sys/module/videobuf2_core/parameters/debug
echo 0 > /sys/module/videobuf2_v4l2/parameters/debug
echo 0 > /sys/module/videobuf2_common/parameters/debug
```

对应代码位置: SDK/kernel/drivers/media/ v4l2-core/videobuf2-core.c

及

```
echo 0x1f > /sys/class/video4linux/video0/dev_debug
```

对应代码位置: SDK/kernel/drivers/media/v4l2-core/v4l2-dev.c

将对应video的v4l2-ctl调用开关打开;

```
echo 0x3f > /sys/class/video4linux/videox/dev_debug
echo 0 > /sys/class/video4linux/video0/dev_debug
dmesg -n 8
```

uvc框架的LOG抓取

// 开启调试日志

```
echo 0xffff > /sys/module/uvcvideo/parameters/trace
echo 8 > /proc/sys/kernel/printk或dmesg -n 8
```

// 关闭调试日志

```
echo 0 > /sys/module/uvcvideo/parameters/trace
```

aw36518闪光灯驱动LOG抓取

通过如下命令打开debug开关:

```
echo 3 > /sys/module/aw36518/parameters/debug
echo 8 4 1 8 > /proc/sys/kernel/printk
```

关闭LOG开关:

```
echo 7 4 1 7 > /proc/sys/kernel/printk
echo 0 > /sys/module/aw36518/parameters/debug
```

cn3927v马达驱动LOG抓取

通过如下命令打开debug开关:

```
echo 3 > /sys/module/cn3927v/parameters/debug
echo 8 4 1 8 > /proc/sys/kernel/printk
```

关闭LOG开关:

```
echo 7 4 1 7 > /proc/sys/kernel/printk
echo 0 > /sys/module/cn3927v/parameters/debug
```

dw9800v马达驱动LOG抓取

通过如下命令打开debug开关:

```
echo 3 > /sys/module/dw9800v/parameters/debug
echo 8 4 1 8 > /proc/sys/kernel/printk
```

关闭LOG开关:

```
echo 7 4 1 7 > /proc/sys/kernel/printk
echo 0 > /sys/module/dw9800v/parameters/debug
```

如何配置camera3_profiles.xml文件?

一般是在各平台默认的camera3_profiles.xml基础上进行修改, 修改参考CameraHal文档:

SDK\RKDocs\common\camera\HAL3\camera_hal3_user_manual_vx.x.pdf

或<https://redmine.rockchip.com.cn/documents/53>上的camera_hal3_user_manual_vx.x.pdf文档

camera3_profiles.xml里面<Android_metadata>中的参数详细含义及配置说明,

可参考SDK中Google官方文档: SDK/system/media/camera/docs/docs.html

新增回读/多摄模式配置

lsdk/hardware/rockchip/camera\$ 目录查看是否有如下提交

如果有如下提交, 配置多摄支持时候仅需在camera3_profiles.xml中添加:

```
<aiq.multicamera value="true"/> <!-- true or false -->
```

```

commit b4bed1c0cf31d06865a339bb3814a9d8511a68f6
Author: Wang Panzhenhuan <randy.wang@rock-chips.com>
Date: Sat Aug 6 16:29:17 2022 +0800

    rkisp2: support xml config multi camera mode & fix rk356x double camera link
    issue

    usage:

    add following key value in Sensor_info_RKISP1 section
    <aiq.multicamera value="true"/> <!-- true or false -->

    Signed-off-by: Wang Panzhenhuan <randy.wang@rock-chips.com>
    Change-Id: I179ce6d5cd3662fd788f7a0226e4590e2b279c90

```

示例参考: camera3_profiles_rk356x.xml中gc2093

在: Sensor_info_RKISP1字段内添加

```

<aiq.multicamera value="true"/> <!-- true or false -->

...<Sensor_info_RKISP1><!-- Information that parametrizes the behavior or qualities of the ph...
...<sensorType value="SENSOR_TYPE_RAW"/><!-- SENSOR_TYPE_SOC or SENSOR_TYPE_RAW -->
...<exposure.sync value="true"/><!-- compensate exposure sync -->
...<sensor.digitalGain value="true"/><!-- digital gain support on sensor -->
...<gain.lag value="2"/><!-- camera3 HAL CPF parameters moved here start -->
...<exposure.lag value="2"/>
...<fov value="54.8" value_v="42.5"/>
...<statistics.initialSkip value="1"/><!-- camera3 HAL CPF parameters moved here end -->
...<frame.initialSkip value="3"/><!-- should equal actual skipFrames -- 2 for driver dropp...
...<isoAnalogGain1 value="75"/><!-- Pseudo ISO corresponding analog gain value 1.0 -->
...<ciTMaxMargin value="10"/><!-- coarse integration time max margin -->
...<aiq.workingMode value="HDR2"/><!-- NORMAL or HDR2 or HDR3 -->
...<aiq.multicamera value="true"/><!-- true or false -->
...</Sensor_info_RKISP1>

```

如果没有上述提交, 只需要SDK/hardware/rockchip/camera\$目录如下修改即可;

```

diff --git a/common/platformdata/PlatformData.cpp
b/common/platformdata/PlatformData.cpp
index 4d9257f..91cf277 100755
--- a/common/platformdata/PlatformData.cpp
+++ b/common/platformdata/PlatformData.cpp
@@ -1047,7 +1047,7 @@ CameraHWInfo::CameraHWInfo() :
    mProductName = "<not_set>";
    mManufacturerName = "<not_set>";
    mCameraDeviceAPIVersion = CAMERA_DEVICE_API_VERSION_3_3;
-   mSupportDualVideo = false;
+   mSupportDualVideo = true;
    mSupportExtendedMakernote = false;
    mSupportFullColorRange = true;
    mSupportIPUAcceleration = false;

```

新增HDR模式配置

camera3_profiles.xml 中在 Sensor_info_RKISP1 关键字段内部添加如下字段:

```

<aiq.workingMode value="NORMAL"/> <!-- NORMAL or HDR2 or HDR3 -->

```

NORMAL表示线性模式, HDR2和HDR3别对应2帧、3帧的HDR模式;

各平台Camera示例dts配置

RK Android9.0中各平台样机对应dts及所配置Sensor情况如下：

rk3126c平台

可参考rk3126-bnd-d708.dtsi，默认配置前置GC0329+后置GC2145

rk3288w平台

可参考rk3288-th804-avb.dts，默认配置前置GC2145+后置OV8858

rk3326平台

可参考rk3326-863-lp3-v10-avb.dts，默认配置前置GC0312+后置GC2145

rk3368平台

可参考rk3368-xikp-avb.dts，默认配置前置GC2145+后置OV8858

rk3399平台

可参考rk3399-tve1030g-avb.dts，默认只配置前置GC2355

RK3399摄像头使用cif_clkoutb配置说明

dts默认配置都是GPIO2_B3作为cif_clkout，cif_clkoutb没有用过

```
4860
4861     &pinctrl {
4862         .isp {
4863             cif_clkout: cif-clkout {
4864                 .rockchip,pins =
4865                 .<2 11 3 &pcfg_pull_none>;
4866             };
4867         };
4868     };
```

图 7 默认cif_clkout引脚查看

而cif_clkoutb为GPIO3_b7

			software;
			gpio3b7_sel
			GPIO3B[7] iomux select
15:14	RW	0x0	2'b00: gpio
			2'b01: mac_crs
			2'b10: uart3gps_sout
			2'b11: cif_clkoutb

图 8 cif_clkoutb 对应引脚可复用功能查看

修改方法：

使用GPIO3_B7作为cif_clkoutb修改如下：

```

--- a/arch/arm64/boot/dts/rockchip/rk3399-android.dtsi
+++ b/arch/arm64/boot/dts/rockchip/rk3399-android.dtsi
@@ -375,7 +375,7 @@
        cif_clkout: cif-clkout {
            rockchip,pins =
                /*cif_clkout*/
-               <2 11 RK_FUNC_3 &pcfg_pull_none>;
+               <3 15 RK_FUNC_3 &pcfg_pull_none>;
        };

        isp_dvp_d0d7: isp-dvp-d0d7 {

```

使用命令： `cat /sys/kernel/debug/pinctrl/pinctrl/pinmux-pins`

查看gpio3_b7是否已经是cif_clkout功能了？结果如下：说明已经为cif_clkout了

```

pin 110 (gpio3-14): <MUX UNCLAIMED> <GPIO UNCLAIMED>
pin 111 (gpio3-15): ff920000.rkisp1 <GPIO UNCLAIMED> function isp group cif-clkout
pin 112 (gpio3-16): <MUX UNCLAIMED> <GPIO UNCLAIMED>

```

图9 GPIO3_B7设置为cif_clkoutb功能查看

经测量可以正常输出24MHZ、12MHX的MCLK，OK；

rk356x平台摄像头DTS注册

单目摄像头使用MIPI-4lane情况

链接关系必须为： `sensor->csi2_dphy0->isp*`；* 向下兼容1/2lane*；*

示例参考：arch/arm64/boot/dts/rockchip/rk3566-rk817-tablet.dts中OV8858链路配置

```

224 &csi2_dphy0 {
225     status = "okay";
226
227     ports {
228         #address-cells = <1>;
229         #size-cells = <0>;
230         port@0 {
231             reg = <0>;
232             #address-cells = <1>;
233             #size-cells = <0>;
234
235             mipi_in_ucam0: endpoint@0 {
236                 reg = <0>;
237                 remote-endpoint = <&gc2385_out>;
238                 data-lanes = <1>;
239             };
240             mipi_in_ucam1: endpoint@1 {
241                 reg = <1>;
242                 remote-endpoint = <&ov8858_out>;
243                 data-lanes = <1 2 3 4>;
244             };
245         };

```

单目摄像头使用MIPI 1/2lane情况

链接关系1： `*sensor->csi2_dphy1->isp*`

链接关系2： `*sensor->csi2_dphy2->isp*`

示例参考：arch/arm64/boot/dts/rk3566-evb1-ddr4-v10.dtsi

```

&csi2_dphy1.{
    >status = "okay";

    > /*
    > *. dphy1 only used for split mode,
    > *. can be used concurrently with dphy2
    > *. full mode and split mode are mutually exclusive
    > */
    > ports {
    >     > #address-cells = <1>;
    >     > #size-cells = <0>;

    >     > port@0 {
    >         > reg = <0>;
    >         > #address-cells = <1>;
    >         > #size-cells = <0>;

    >         > dphy1_in: endpoint@1 {
    >             > reg = <1>;
    >             > remote-endpoint = <&ov5695_out>;
    >             > data-lanes = <1.2>;
    >         };

    >     > port {
    >         > ov5695_out: endpoint {
    >             > remote-endpoint = <&dphy1_in>;
    >             > data-lanes = <1.2>;
    >         };
    >     };
}

&csi2_dphy2.{
    >status = "okay";

    > /*
    > *. dphy2 only used for split mode,
    > *. can be used concurrently with dphy1
    > *. full mode and split mode are mutually exclusive
    > */
    > ports {
    >     > #address-cells = <1>;
    >     > #size-cells = <0>;

    >     > port@0 {
    >         > reg = <0>;
    >         > #address-cells = <1>;
    >         > #size-cells = <0>;

    >         > dphy2_in: endpoint@1 {
    >             > reg = <1>;
    >             > remote-endpoint = <&ov02k10_out>;
    >             > data-lanes = <1.2>;
    >         };
    >     };

    >     > port {
    >         > ov02k10_out: endpoint {
    >             > remote-endpoint = <&dphy2_in>;
    >             > data-lanes = <1.2>;
    >         };
    >     };
}

```

双目使用MIPI-RAW+DVP-SOC情况

链接关系1: *sensor1->csi2_dphy0->isp_vir0*

链接关系2: *sensor2->vicap*

示例参考: arch/arm64/boot/dts/rockchip/rk3566-evb3-ddr3-v10.dtsi

里面包含ov5695-mipi camera+ gc2145-dvp camera配置

```
&rkCIF.{
    >status = "okay";
};

&rkCIF_dvp.{
    >status = "okay";

    >port.{
        >/* Parallel bus endpoint */
        >dvp_in_bcam: endpoint.{
            >remote-endpoint = <&gc2145_out>;
            >bus-width = <8>;
            >vsync-active = <0>;
            >hsync-active = <1>;
        };
    };
};

&rkisp.{
    >status = "okay";
};

&rkisp_mmu.{
    >status = "okay";
};

&rkisp_vir0.{
    >status = "okay";

    >port.{
        >#address-cells = <1>;
        >#size-cells = <0>;

        >isp0_in: endpoint@0.{
            >reg = <0>;
            >remote-endpoint = <&csi2dphy_out>;
        };
    };
};
```

双目使用MIPI-RAW+MIPI-RAW情况（即2lane+2lane）

这时单个硬件isp通过虚拟多个设备，分时复用处理多路sensor数据。

链接关系1: *sensor1->csi2_dphy1->isp_vir0*

链接关系2: *sensor2->csi2_dphy2->mipi_csi2->vicap->isp_vir1*

示例参考: arch/arm64/boot/dts/rockchip/rk3566-evb2-lp4x-v10.dtsi

里面包含ov5695-mipi camera+ gc5025-mipi camera配置，但vicap->isp_vir1还未连接，

需要添加如下配置：

```

521 &rkCIF_mipi_lvds_sditf {
522     → status = "okay";
523
524     → port {
525         → mipi_lvds_sditf: endpoint {
526             → remote-endpoint = <&isp1_in>;
527             → data-lanes = <1 2>;
528         };
529     };
530 };
531
&rkisp_vir1 {
    → status = "okay";
    → /*.gc2093->dphy2->csi2->vicap.*/
    → /*.vicap.sditf->isp_vir1.*/
    → port {
    →     → #address-cells = <1>;
    →     → #size-cells = <0>;
    →
    →     → isp1_in: endpoint@0 {
    →         → reg = <0>;
    →         → remote-endpoint = <&mipi_lvds_sditf>;
    →     };
    → };
};

```

RK356x-MIPI-CSI-DPHY配置注意事项

MIPI-CSI-DPHY支持两种配置模式，即full mode和split mode

Option1	Sensor1 x4Lane	MIPI_CSI_RX_D0-3 MIPI_CSI_RX_CLK0
Option2	Sensor1 x2Lane	MIPI_CSI_RX_D0-1 MIPI_CSI_RX_CLK0
	+ Sensor2 x2Lane	MIPI_CSI_RX_D2-3 MIPI_CSI_RX_CLK1

full mode和split mode不能同时使用, 反应在dts配置上即

csi2_dphy0 不能和 csi2_dphy1、csi2_dphy2同时使能;

csi2_dphy0 使能的时候, csi2_dphy1和csi2_dphy2需要disabled;

同理: csi2_dphy1和csi2_dphy2使能的时候, csi2_dphy0需要disabled

上面dts具体配置参考文档

参考SDK\RKDocs\common\camera\HAL3\Rockchip_Driver_Guide_VI\CN\

里面的Rockchip_Driver_Guide_VI_CN_vx.x.x.pdf文档排查

RK356x-CIS时钟配置注意事项

RK356x有四个输出时钟，可作为给CIS模组的MCLK：

1、REFCLK_OUT（GPIO0_A0）

示例dts：arch/arm64/boot/dts/rockchip/rk3566-evb2-lp4x-v10.dtsi中的gc5025时钟配置

```
camera: camera @xx {
    ....
    clocks = <&pmucru CLK_WIFI>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&refclk_pins>;
};
```

2、CAM_CLKOUT0（GPIO4_A7）

示例dts：arch/arm64/boot/dts/rockchip/rk3566-rk817-tablet.dts中的ov8858时钟配置

```
camera: camera @xx {
    ....
    clocks = <&cru CLK_CAM0_OUT>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clkout0 >;
};
```

3、CAM_CLKOUT1（GPIO4_A8）

示例dts：arch/arm64/boot/dts/rockchip/rk3566-evb1-ddr4-v10.dtsi中的ov02k10时钟配置

```
camera: camera @xx {
    ....
    clocks = <&cru CLK_CAM1_OUT>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clkout1>;
};
```

4、CLK_CIF_OUT（GPIO4_C0）

示例dts：arch/arm64/boot/dts/rockchip/rk3566-rk817-tablet.dts中的gc2385时钟配置

```
camera: camera @xx {
    ....
    clocks = <&cru CLK_CIF_OUT >;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cif_clk >;
};
```

RK356x-CIS各个时钟支持的输出频率

RK356X	CLK_WIFI	GPIO0_A0	24M
RK356X	CLK_MAC0_OUT	GPIO2_C1	24M、25M、50M、125M
RK356X	CLK_MAC1_OUT	GPIO3_B0	24M、25M、50M、125M
RK356X	CLK_MAC1_OUT	GPIO4_B3	24M、25M、50M、125M
RK356X	CLK_CIF_OUT	GPIO4_C0	24M、27M、37.125M
RK356X	CLK_CAM0_OUT	GPIO4_A7	24M、27M、37.125M
RK356X	CLK_CAM1_OUT	GPIO4_B0	24M、27M、37.125M

图 11 RK356x-CIS-CLOCKS

rk3588平台摄像头DTS注册

RK3588-目前各个样机使用的摄像头

RK3588s样机平板搭配摄像头型号：ov50c40(4800w) + ov13855(1300w)前后摄切换;

RK 3588-EVB1搭配摄像头型号：IMX415（800w）

RK3588s-EVB1搭配摄像头型号：ov50c40(4800w) + ov50c40 (4800w)前后摄切换;

dcphy0-单ISP-8M摄像头配置参考

可参考rk3588-evb1-imx415.dtsi，默认配置8M摄像头imx415连接RK3588-dcphy0;

dhpy0-单ISP-8M摄像头配置参考

可参考rk3588-evb2-imx415.dtsi，默认配置8M摄像头imx415连接RK3588-dphy0;

48M-摄像头使用ISP联合模式配置参考

可参考：rk3588s-tablet.dtsi：默认配置后置48M摄像头OV50C40，前置13M摄像头ov13855;

也可参考：rk3588s-evb1-lp4x-v10-camera.dtsi，默认dcphy0/1均连接OV50c40，切换使用;

RK3588-CIS时钟配置注意事项

RK3588有四个输出时钟，可作为给CIS模组的MCLK:

1、CLK_MIPI_CAMARAOUT_M0（GPIO4_B1/GPIO3_A5）

示例时钟配置

```
gc8034: gc8034@37 {
    camera: camera @xx {
        ....
        clocks = <&cru CLK_MIPI_CAMARAOUT_M0>;
        clock-names = "xvclk";
        pinctrl-names = "default";
        pinctrl-0 = <&mipim0_camera0_clk>; //<&mipim0_camera0_clk> (gpio3_a5)
    };
};
```

2、CLK_MIPI_CAMARAOUT_M1（GPIO1_B6/GPIO3_A6）

示例dts: ./arch/arm64/boot/dts/rockchip/rk3588s-evb1-lp4x-v10-camera.dtsi中ov50c40时钟配置

```

camera: camera @xx {
    .....
    clocks = <&cru CLK_MIPI_CAMARAOUT_M1>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&mipim0_camera1_clk>; //<&mipim1_camera1_clk> (gpio3_a6)
};

```

3、CLK_MIPI_CAMARAOUT_M2 (GPIO1_B7/GPIO3_A7)

示例dts: arch/arm64/boot/dts/rockchip/rk3588s-evb1-lp4x-v10-camera.dtsi中的ov50c40b时钟配置

```

camera: camera @xx {
    .....
    clocks = <&cru CLK_MIPI_CAMARAOUT_M2>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&mipim0_camera2_clk>; //<&mipim1_camera2_clk> (gpio3_a6)
};

```

4、CLK_MIPI_CAMARAOUT_M3 (GPIO1_D6/GPIO3_B0)

示例dts: ./arch/arm64/boot/dts/rockchip/rk3588s-evb2-imx415.dtsi 中的imx415时钟配置

```

camera: camera @xx {
    .....
    clocks = <&cru CLK_MIPI_CAMARAOUT_M3>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&mipim0_camera3_clk>; //<&mipim1_camera3_clk> (gpio3_b0)
};

```

5、CLK_MIPI_CAMARAOUT_M4 (GPIO1_D7/GPIO3_B1)

示例dts: ./arch/arm64/boot/dts/rockchip/rk3588s-tablet-single.dtsi中的ov13855时钟配置

```

camera: camera @xx {
    .....
    clocks = <&cru CLK_MIPI_CAMARAOUT_M4>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&mipim0_camera4_clk>; //<&mipim1_camera4_clk> (gpio3_b0)
};

```

6、CLK_CIF_OUT (GPIO4_B4)

示例dts配置

```
camera: camera @xx {
    .....
    clocks = <&cru CLK_CIF_OUT >;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <& cif_clk>;
};
```

RK3588-CIS各个时钟支持的输出频率

RK3588	CLK_MIPI_CAMARAOUT_M0	GPIO4_B1	24M、27M、37.125M
RK3588	CLK_MIPI_CAMARAOUT_M1	GPIO1_B6	24M、27M、37.125M
RK3588	CLK_MIPI_CAMARAOUT_M2	GPIO1_B7	24M、27M、37.125M
RK3588	CLK_MIPI_CAMARAOUT_M3	GPIO1_D6	24M、27M、37.125M
RK3588	CLK_MIPI_CAMARAOUT_M4	GPIO1_D7	24M、27M、37.125M
RK3588	CLK_CIF_OUT	GPIO4_B4	24M、27M、37.125M

rk3562平台摄像头DTS注册

RK3562-目前各个样机使用的摄像头

RK3562样机平板搭配摄像头型号：ov13855(1300w) + gc8034(800w)前后摄切换;

RK3562-evb1搭配摄像头型号：gc8034(800w) + ov5695(500w)前后摄切换;

RK3562-evb2搭配摄像头型号：gc8034(800w) + ov5695(500w)双摄同时工作;

dhpy0-接8M+5M前后摄像头切换配置参考

可参考rk3562-evb1-cam.dtsi，默认配置8M+5M摄像头连接RK3562-dphy0;

dhpy0-dphy3分别接摄像头配置参考

可参考rk3562-evb2-cam.dtsi，配置

8M摄像头gc8034-0连接RK3562-dphy0;

8M摄像头gc8034-1连接RK3562-dphy3;

双目摄像头配置参考

可参考rk3562-evb2-ddr4-v10-dual-camera.dts，配置

8M摄像头gc8034连接RK3562-dphy1;

5M摄像头ov5695连接RK3562-dphy2;

13M-摄像头使用配置参考

可参考：rk3562-rk817-tablet-camera.dtsi：默认配置后置13M摄像头OV13855，

前置8M摄像头GC8034

RK3562-CIS时钟配置注意事项

RK3562有四个输出时钟，可作为给CIS模组的MCLK：

1、CLK_CAM0_OUT2IO（GPIO3_B2/GPIO4_B1）

示例dts：arch/arm64/boot/dts/rockchip/rk3562-evb2-cam.dtsi中的gc8034时钟配置

```
gc8034: gc8034@37 {
    camera: camera @xx {
        .....
        clocks = <&cru CLK_CAM0_OUT2IO>;
        clock-names = "xvclk";
        pinctrl-names = "default";
        pinctrl-0 = <&cam0_clk0_out>; //<&cam1_clk0_out> (gpio4_b1)
    };
};
```

2、CLK_CAM1_OUT2IO（GPIO3_B3/GPIO4_B7）

示例dts：./arch/arm64/boot/dts/rockchip/rk3562-evb2-ddr4-v10-dual-camera.dts中ov5695时钟配置

```
camera: camera @xx {
    .....
    clocks = <&cru CLK_CAM1_OUT2IO>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam0_clk1_out>; //<&cam1_clk1_out> (gpio4_b7)
};
```

3、CLK_CAM2_OUT2IO（GPIO3_B4）

示例dts：arch/arm64/boot/dts/rockchip/rk3562-evb2-cam.dtsi中的gc8034_1时钟配置

```
camera: camera @xx {
    .....
    clocks = <&cru CLK_CAM2_OUT2IO>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clk2_out>;
};
```

4、CLK_CAM3_OUT2IO（GPIO3_B5）

暂无示例dts配置；

```
camera: camera @xx {
    .....
    clocks = <&cru CLK_CAM3_OUT2IO>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clk3_out>;
};
```

rk3576平台摄像头DTS注册

RK3576-目前各个样机使用的摄像头

RK3576样机平板搭配摄像头型号：ov16880(1600w) + g05a2(500w)前后摄切换;

RK3576 EVB1搭配摄像头型号：IMX415（800w）

dchpy0-单ISP-8M摄像头配置参考

可参考rk3576-evb1-cam-dcphy0.dtsi，默认配置8M摄像头imx415连接RK3576-dcphy0;

dphy3-16M 和dchpy0 -5M摄像头前后切换配置参考

可参考rk3576-rk806-tablet-camera.dtsi，配置

5M摄像头gc05a2连接RK3576-dcphy0;

16M摄像头ov16880接RK3576-dphy3

rk3576 中dphy/dcphy到mipi_csi 到rkcif_mipi_lvds*的链接关系

rk3576 中dphy/dcphy到mipi_csi 到rkcif_mipi_lvds*的链接关系如下:

dcphy:

csi2_dcphy0->mipi0_csi2->rkcif_mipi_lvds

dphy0:

full mode: csi2_dphy0->mipi1_csi2->rkcif_mipi_lvds1

split mode:

csi2_dphy1->mipi1_csi2->rkcif_mipi_lvds1

csi2_dphy2->mipi2_csi2->rkcif_mipi_lvds2,

dphy1:

full mode: csi2_dphy3->mipi3_csi2->rkcif_mipi_lvds3

split mode:

csi2_dphy4->mipi3_csi2->rkcif_mipi_lvds3

csi2_dphy5->mipi4_csi2->rkcif_mipi_lvds4

full mode和split mode说明见

SDK\RKDocs\common\camera\HAL3\Rockchip_Driver_Guide_VI_CN\
Rockchip_Driver_Guide_VI_CN_v1.1.5.pdf文档

RK3576- CIS时钟配置注意事项

RK3576有四个输出时钟，可作为给CIS模组的MCLK：

1、CLK_MIPI_CAMERAOUT_M0（GPIO3_D7/GPIO2_D2）

```
amera: camera @xx {
    ....
    clocks = <&cru CLK_MIPI_CAMERAOUT_M0>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clk0m0_clk0>; //<&cam_clk0m1_clk0> (gpio2_d2)
};
```

2、CLK_MIPI_CAMERAOUT_M1（GPIO4_A0/GPIO2_D6）

```
camera: camera @xx {
    ....
    clocks = <&cru CLK_MIPI_CAMERAOUT_M1>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clk1m0_clk1>; //<&cam_clk1m1_clk1> (gpio2_d6)
};
```

3、CLK_MIPI_CAMERAOUT_M2（GPIO4_A1/GPIO2_D7）

```
camera: camera @xx {
    ....
    clocks = <&cru CLK_MIPI_CAMERAOUT_M2>;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&cam_clk2m0_clk2>; //<&cam_clk2m1_clk2> (gpio2_d7)
};
```

4、CLK_CIFOUT_OUT（GPIO3_A2）

```
camera: camera @xx {
    ....
    clocks = <&cru CLK_CIFOUT_OUT >;
    clock-names = "xvclk";
    pinctrl-names = "default";
    pinctrl-0 = <&vi_cif_pins >; //bt1120所有引脚一起，后续需单独拿出来
};
```

RK3576-各个时钟支持的输出频率

RK3576	VI_CIF_CLKOUT	GPIO3_A2	24M、27M、37.125M
RK3576	CAM_CLK0_OUT_M0\M1	GPIO3_D7\GPIO2_D2	24M、25M、26M、27M、37.125M、50M
RK3576	CAM_CLK1_OUT_M0\M1	GPIO4_A0\GPIO2_D6	24M、25M、26M、27M、37.125M、50M
RK3576	CAM_CLK2_OUT_M0\M1	GPIO4_A1\GPIO2_D7	24M、25M、26M、27M、37.125M、50M

Camera 拓扑链路查看

使用media-ctl -d /dev/mediax -p 查看：

其中mediax为：media0/media1/media2...，根据ls /dev/media*查看注册了多少个media结点确定；

示例链路如图：

```
rk3588_s:/ # media-ctl -d /dev/media0 -p
Opening media device /dev/media0
Enumerating entities
Found 14 entities
Enumerating pads and links
Media controller API version 0.0.66

Media device information
/-----
driver          rkCIF
model           rkCIF-mipi-lvds
serial
bus info
hw revision     0x0
driver version  0.0.66

Device topology
- entity 1: stream_cif_mipi_id0 (1 pad, 11 links)
    type Node subtype V4L
    device node name /dev/video0
    pad0: Sink
        <- "rockchip-mipi-csi2":1 []
        <- "rockchip-mipi-csi2":2 []
        <- "rockchip-mipi-csi2":3 []
        <- "rockchip-mipi-csi2":4 []
        <- "rockchip-mipi-csi2":5 []
        <- "rockchip-mipi-csi2":6 []
        <- "rockchip-mipi-csi2":7 []
        <- "rockchip-mipi-csi2":8 []
        <- "rockchip-mipi-csi2":9 []
        <- "rockchip-mipi-csi2":10 []
        <- "rockchip-mipi-csi2":11 []

- entity 5: stream_cif_mipi_id1 (1 pad, 11 links)
```

```

type Node subtype V4L
device node name /dev/video1
pad0: Sink
    <- "rockchip-mipi-csi2":1 []
    <- "rockchip-mipi-csi2":2 []
    <- "rockchip-mipi-csi2":3 []
    <- "rockchip-mipi-csi2":4 []
    <- "rockchip-mipi-csi2":5 []
    <- "rockchip-mipi-csi2":6 []
    <- "rockchip-mipi-csi2":7 []
    <- "rockchip-mipi-csi2":8 []
    <- "rockchip-mipi-csi2":9 []
    <- "rockchip-mipi-csi2":10 []
    <- "rockchip-mipi-csi2":11 []

- entity 9: stream_cif_mipi_id2 (1 pad, 11 links)
  type Node subtype V4L
  device node name /dev/video2
  pad0: Sink
      <- "rockchip-mipi-csi2":1 []
      <- "rockchip-mipi-csi2":2 []
      <- "rockchip-mipi-csi2":3 []
      <- "rockchip-mipi-csi2":4 []
      <- "rockchip-mipi-csi2":5 []
      <- "rockchip-mipi-csi2":6 []
      <- "rockchip-mipi-csi2":7 []
      <- "rockchip-mipi-csi2":8 []
      <- "rockchip-mipi-csi2":9 []
      <- "rockchip-mipi-csi2":10 []
      <- "rockchip-mipi-csi2":11 []

- entity 13: stream_cif_mipi_id3 (1 pad, 11 links)
  type Node subtype V4L
  device node name /dev/video3
  pad0: Sink
      <- "rockchip-mipi-csi2":1 []
      <- "rockchip-mipi-csi2":2 []
      <- "rockchip-mipi-csi2":3 []
      <- "rockchip-mipi-csi2":4 []
      <- "rockchip-mipi-csi2":5 []
      <- "rockchip-mipi-csi2":6 []
      <- "rockchip-mipi-csi2":7 []
      <- "rockchip-mipi-csi2":8 []
      <- "rockchip-mipi-csi2":9 []
      <- "rockchip-mipi-csi2":10 []
      <- "rockchip-mipi-csi2":11 []

- entity 17: rkCIF_scale_ch0 (1 pad, 11 links)
  type Node subtype V4L
  device node name /dev/video4
  pad0: Sink
      <- "rockchip-mipi-csi2":1 []
      <- "rockchip-mipi-csi2":2 []
      <- "rockchip-mipi-csi2":3 []
      <- "rockchip-mipi-csi2":4 []
      <- "rockchip-mipi-csi2":5 []
      <- "rockchip-mipi-csi2":6 []
      <- "rockchip-mipi-csi2":7 []

```

```

        <- "rockchip-mipi-csi2":8 []
        <- "rockchip-mipi-csi2":9 []
        <- "rockchip-mipi-csi2":10 []
        <- "rockchip-mipi-csi2":11 []

- entity 21: rkCIF_scale_ch1 (1 pad, 11 links)
    type Node subtype V4L
    device node name /dev/video5
    pad0: Sink
        <- "rockchip-mipi-csi2":1 []
        <- "rockchip-mipi-csi2":2 []
        <- "rockchip-mipi-csi2":3 []
        <- "rockchip-mipi-csi2":4 []
        <- "rockchip-mipi-csi2":5 []
        <- "rockchip-mipi-csi2":6 []
        <- "rockchip-mipi-csi2":7 []
        <- "rockchip-mipi-csi2":8 []
        <- "rockchip-mipi-csi2":9 []
        <- "rockchip-mipi-csi2":10 []
        <- "rockchip-mipi-csi2":11 []

- entity 25: rkCIF_scale_ch2 (1 pad, 11 links)
    type Node subtype V4L
    device node name /dev/video6
    pad0: Sink
        <- "rockchip-mipi-csi2":1 []
        <- "rockchip-mipi-csi2":2 []
        <- "rockchip-mipi-csi2":3 []
        <- "rockchip-mipi-csi2":4 []
        <- "rockchip-mipi-csi2":5 []
        <- "rockchip-mipi-csi2":6 []
        <- "rockchip-mipi-csi2":7 []
        <- "rockchip-mipi-csi2":8 []
        <- "rockchip-mipi-csi2":9 []
        <- "rockchip-mipi-csi2":10 []
        <- "rockchip-mipi-csi2":11 []

- entity 29: rkCIF_scale_ch3 (1 pad, 11 links)
    type Node subtype V4L
    device node name /dev/video7
    pad0: Sink
        <- "rockchip-mipi-csi2":1 []
        <- "rockchip-mipi-csi2":2 []
        <- "rockchip-mipi-csi2":3 []
        <- "rockchip-mipi-csi2":4 []
        <- "rockchip-mipi-csi2":5 []
        <- "rockchip-mipi-csi2":6 []
        <- "rockchip-mipi-csi2":7 []
        <- "rockchip-mipi-csi2":8 []
        <- "rockchip-mipi-csi2":9 []
        <- "rockchip-mipi-csi2":10 []
        <- "rockchip-mipi-csi2":11 []

- entity 33: rkCIF_tools_id0 (1 pad, 11 links)
    type Node subtype V4L
    device node name /dev/video8
    pad0: Sink
        <- "rockchip-mipi-csi2":1 []

```

```

        <- "rockchip-mipi-csi2":2 []
        <- "rockchip-mipi-csi2":3 []
        <- "rockchip-mipi-csi2":4 []
        <- "rockchip-mipi-csi2":5 []
        <- "rockchip-mipi-csi2":6 []
        <- "rockchip-mipi-csi2":7 []
        <- "rockchip-mipi-csi2":8 []
        <- "rockchip-mipi-csi2":9 []
        <- "rockchip-mipi-csi2":10 []
        <- "rockchip-mipi-csi2":11 []

- entity 37: rkCIF_tools_id1 (1 pad, 11 links)
  type Node subtype V4L
  device node name /dev/video9
  pad0: Sink
    <- "rockchip-mipi-csi2":1 []
    <- "rockchip-mipi-csi2":2 []
    <- "rockchip-mipi-csi2":3 []
    <- "rockchip-mipi-csi2":4 []
    <- "rockchip-mipi-csi2":5 []
    <- "rockchip-mipi-csi2":6 []
    <- "rockchip-mipi-csi2":7 []
    <- "rockchip-mipi-csi2":8 []
    <- "rockchip-mipi-csi2":9 []
    <- "rockchip-mipi-csi2":10 []
    <- "rockchip-mipi-csi2":11 []

- entity 41: rkCIF_tools_id2 (1 pad, 11 links)
  type Node subtype V4L
  device node name /dev/video10
  pad0: Sink
    <- "rockchip-mipi-csi2":1 []
    <- "rockchip-mipi-csi2":2 []
    <- "rockchip-mipi-csi2":3 []
    <- "rockchip-mipi-csi2":4 []
    <- "rockchip-mipi-csi2":5 []
    <- "rockchip-mipi-csi2":6 []
    <- "rockchip-mipi-csi2":7 []
    <- "rockchip-mipi-csi2":8 []
    <- "rockchip-mipi-csi2":9 []
    <- "rockchip-mipi-csi2":10 []
    <- "rockchip-mipi-csi2":11 []

- entity 45: rockchip-mipi-csi2 (12 pads, 122 links)
  type V4L2 subdev subtype Unknown
  device node name /dev/v4l-subdev0
  pad0: Sink
    <- "rockchip-csi2-dphy0":1 []
  pad1: Source
    -> "stream_cif_mipi_id0":0 []
    -> "stream_cif_mipi_id1":0 []
    -> "stream_cif_mipi_id2":0 []
    -> "stream_cif_mipi_id3":0 []
    -> "rkCIF_scale_ch0":0 []
    -> "rkCIF_scale_ch1":0 []
    -> "rkCIF_scale_ch2":0 []
    -> "rkCIF_scale_ch3":0 []
    -> "rkCIF_tools_id0":0 []

```

```
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []
pad2: Source
-> "stream_cif_mipi_id0":0 []
-> "stream_cif_mipi_id1":0 []
-> "stream_cif_mipi_id2":0 []
-> "stream_cif_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []
pad3: Source
-> "stream_cif_mipi_id0":0 []
-> "stream_cif_mipi_id1":0 []
-> "stream_cif_mipi_id2":0 []
-> "stream_cif_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []
pad4: Source
-> "stream_cif_mipi_id0":0 []
-> "stream_cif_mipi_id1":0 []
-> "stream_cif_mipi_id2":0 []
-> "stream_cif_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []
pad5: Source
-> "stream_cif_mipi_id0":0 []
-> "stream_cif_mipi_id1":0 []
-> "stream_cif_mipi_id2":0 []
-> "stream_cif_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []
pad6: Source
-> "stream_cif_mipi_id0":0 []
-> "stream_cif_mipi_id1":0 []
-> "stream_cif_mipi_id2":0 []
-> "stream_cif_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
```

```
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []

pad7: Source
-> "stream_CIF_mipi_id0":0 []
-> "stream_CIF_mipi_id1":0 []
-> "stream_CIF_mipi_id2":0 []
-> "stream_CIF_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []

pad8: Source
-> "stream_CIF_mipi_id0":0 []
-> "stream_CIF_mipi_id1":0 []
-> "stream_CIF_mipi_id2":0 []
-> "stream_CIF_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []

pad9: Source
-> "stream_CIF_mipi_id0":0 []
-> "stream_CIF_mipi_id1":0 []
-> "stream_CIF_mipi_id2":0 []
-> "stream_CIF_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []

pad10: Source
-> "stream_CIF_mipi_id0":0 []
-> "stream_CIF_mipi_id1":0 []
-> "stream_CIF_mipi_id2":0 []
-> "stream_CIF_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []

pad11: Source
-> "stream_CIF_mipi_id0":0 []
-> "stream_CIF_mipi_id1":0 []
-> "stream_CIF_mipi_id2":0 []
-> "stream_CIF_mipi_id3":0 []
-> "rkCIF_scale_ch0":0 []
```

```

-> "rkCIF_scale_ch1":0 []
-> "rkCIF_scale_ch2":0 []
-> "rkCIF_scale_ch3":0 []
-> "rkCIF_tools_id0":0 []
-> "rkCIF_tools_id1":0 []
-> "rkCIF_tools_id2":0 []

- entity 58: rockchip-csi2-dphy0 (2 pads, 2 links)
    type V4L2 subdev subtype Unknown
    device node name /dev/v4l-subdev1
    pad0: Sink
        <- "m00_b_imx415 5-001a":0 []
    pad1: Source
        -> "rockchip-mipi-csi2":0 []

- entity 63: m00_b_imx415 5-001a (1 pad, 1 link)
    type V4L2 subdev subtype Sensor
    device node name /dev/v4l-subdev2
    pad0: Source
        [fmt:SGBRG10/3864x2192
        crop.bounds:(12,16)/3840x2160]
        -> "rockchip-csi2-dphy0":0 []

```

Camera Selinux权限

权限问题确认

使用userdebug版本

setenforce 0正常打开camera,

setenforce 1打开camera失败;

说明是权限问题;

权限文件板上位置

手动修改权限文件试验:

```

adb pull /vendor/etc/selinux/vendor_file_contexts
将
/dev/media[0-9]          u:object_r:video_device:s0
/dev/v4l-subdev[0-9] u:object_r:camera_device:s0
改成:
/dev/media.*             u:object_r:video_device:s0
/dev/v4l-subdev.* u:object_r:video_device:s0
/dev/video.* u:object_r:video_device:s0

```

权限文件在sdk中位置

sdk/device/rockchip/common\$ sepolicy/vendor/file_contexts

按照试验后正常修改该文件:

马达

可以参考驱动调试文档：

“SDK\RKDocs\common\camera\HAL3\RKISP_Driver_User_Manual_v1.xx.pdf”，

文档中对马达驱动做了一些说明；调试方法/步骤总结如下三点：

如何查看马达驱动是否注册？

以马达vm149c为例，使用命令：`media-ctl -p`，查看是否包含如图的entity

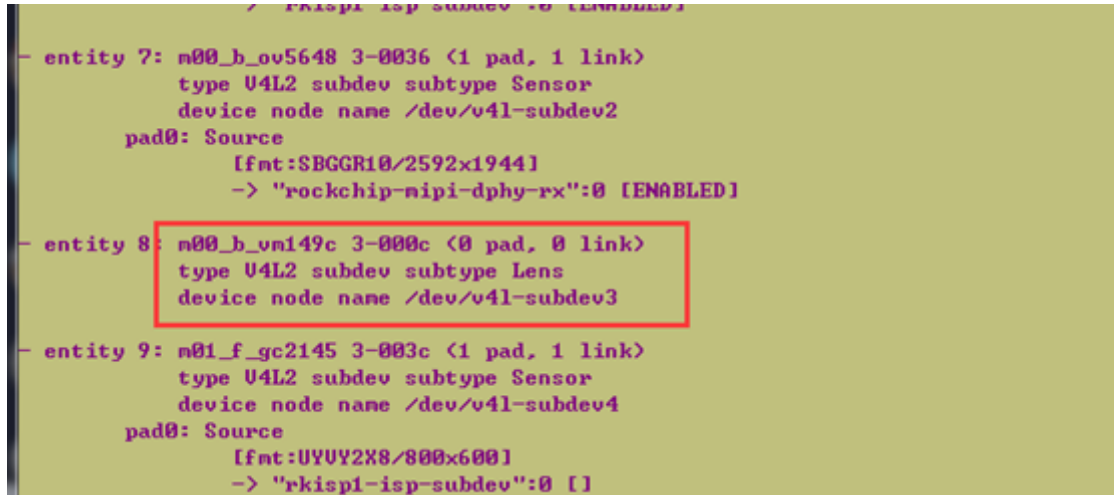


图 8 vm149c马达注册上示例图

如果有，说明调用到马达驱动注册成功；

如果没有请参考SDK\RKDocs\common\camera\HAL3\RKISP_Driver_User_Manual_v1.xx.pdf文档正确配置dts，及移植对应的马达驱动。

如何使用V4L2工具控制马达移动？

打开包含马达的摄像头，使用如下命令：控制马达移动

```
v4l2-ctl -d dev/videox --set-ctrl 'focus_absolute=32' (0~64)
```

或者：

```
v4l2-ctl -d dev/v4l-subdevx --set-ctrl 'focus_absolute=32' (0~64)
```

v4l-subdevx是对应马达注册的结点名称，自己使用`media-ctl -d /dev/mediax -p`查看；

mediax为：media0/media1/media2

如果会移动，说明底层马达控制OK；

如果马达不移动说明马达底层驱动调用有问题；需要在马达驱动的static int xxxx_set_ctrl()和static int xxxx_set_pos()这两个函数中加打印信息调试；

以vm149c为例：控制命令肯定会调到如下函数中

```

222: static int vm149c_set_ctrl(struct v4l2_ctrl *ctrl)
223: {
224:     struct vm149c_device *dev_vcm = to_vm149c_vcm(ctrl);
225:     struct i2c_client *client = v4l2_get_subdevdata(&dev_vcm->sd);
226:     unsigned int dest_pos = ctrl->val;
227:     int move_pos;
228:     long int mv_us;
229:     int ret = 0;
230:
231:     if (ctrl->id == V4L2_CID_FOCUS_ABSOLUTE) {
232:         if (dest_pos > VCMDRV_MAX_LOG) {
233:             dev_info(&client->dev,
234:                 "%s dest_pos is error. %d > %d\n",
235:                 __func__, dest_pos, VCMDRV_MAX_LOG);
236:             return -EINVAL;
237:         } else {
238:             /* calculate move time */
239:             move_pos = dev_vcm->current_related_pos - dest_pos;
240:             if (move_pos < 0)
241:                 move_pos = -move_pos;
242:
243:             ret = vm149c_set_pos(dev_vcm, dest_pos);

```

图9 马达移动调用函数

camera3_profiles.xml如何配置？

只有RAW Sensor才支持马达控制，如果支持马达，camera3_profiles.xml中需要如下配置：

```

<control.afAvailableModes
value="OFF,AUTO,MACRO,CONTINUOUS_VIDEO,CONTINUOUS_PICTURE,EDOF"/>

<control.maxRegions value="1,0,1"/>

<lens.info.minimumFocusDistance value="0.1"/> <!-- HAL may override this value
from CMC for RAW sensors -->

```

如果录像中不需要马达对焦，可以尝试将CONTINUOUS_VIDEO关键字去掉；

<request.availableRequestKeys中需要包含： control.afRegions 字段

<request.availableResultKeys中需要包含： control.afRegions 字段

// SENSOR_TYPE需要是RAW的而不是SOC；

不然调用不到3A库，从而不会调到底层马达进行自动对焦；

如果不支持马达，camera3_profiles.xml中马达相关配置如下：

```

<control.afAvailableModes value="OFF "/>
<control.maxRegions value="1,0,0"/>
<lens.info.minimumFocusDistance value="0.0"/> <!-- HAL may override this value
from CMC for RAW sensors -->

```

<request.availableRequestKeys 中删除： control.afRegions 字段

<request.availableResultKeys 中删除： control.afRegions 字段

效果文件中马达对焦如何配置？

如果需要支持马达对焦，效果文件中 contrast_af 中enable设置为： 1；

```
<contrast_af index="1" type="struct" size="[1 1]">
  <enable index="1" type="double" size="[1 1]">
    [1]
  </enable>
```

如果不需要支持马达对焦，则将enable设置为：0

```
<contrast_af index="1" type="struct" size="[1 1]">
  <enable index="1" type="double" size="[1 1]">
    [0]
  </enable>
```

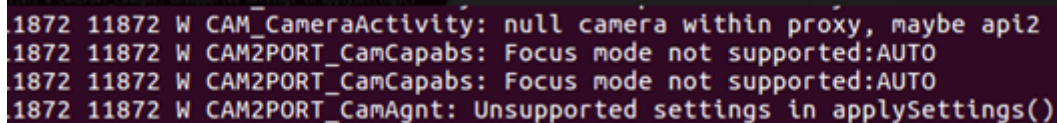
示例案例：无马达OV2680模组配置错误导致切换到录像卡住问题

问题描述

RK3326-9.0机器，使用不带马达的OV2680模组，切换到录像界面出现卡住问题。

卡住打印的关键log如下：

```
W CAM2PORT_CamCapabs: Focus mode not supported:AUTO
W CAM2PORT_CamCapabs: Focus mode not supported:AUTO
W CAM2PORT_CamAgnt: Unsupported settings in applySettings()
```



```
1872 11872 W CAM_CameraActivity: null camera within proxy, maybe api2
1872 11872 W CAM2PORT_CamCapabs: Focus mode not supported:AUTO
1872 11872 W CAM2PORT_CamCapabs: Focus mode not supported:AUTO
1872 11872 W CAM2PORT_CamAgnt: Unsupported settings in applySettings()
```

图 10 关键LOG截图

问题分析

因为客户机器不带VCM马达，所以修改camera3_profiles.xml配置

```
<control.afAvailableModes
value="OFF,AUTO,MACRO,CONTINUOUS_VIDEO,CONTINUOUS_PICTURE,EDOF"/>
```

改成了：

```
<control.afAvailableModes value="OFF"/>
```

但还漏了其他地方没有修改：

解决办法

camera3_profiles.xml配置还需要如下修改：

```
<control.afAvailableModes value="OFF"/>

<lens.info.minimumFocusDistance value="0.0"/>
```

不支持af时，afAvailableModes 必须为OFF，minimumFocusDistance必须为0.0

```
<control.maxRegions value="1,0,1"/>
```

改成

```
<control.maxRegions value="1,0,0"/>
```

注意事项

dts中马达节点中配置的module index需要和对应camera sensor的module index一样;

例如: dts中OV13850节点配置module index如下

```
ov13850: ov13850@10 {
    compatible = "ovti,ov13850";
    .....
    rockchip,camera-module-index = <0>;
    ...
};
```

则 dts中对应马达节点配置module index如下:

```
vm149c: vm149c@0c {
    compatible = "silicon touch,vm149c";
    .....
    rockchip,camera-module-index = <0>;
    .....
};
```

闪光灯

首先参考驱动调试文档:

SDK\RKDocs\common\camera\HAL3\RKISP_Driver_User_Manual_v1.xx.pdf 文档中对闪光灯驱动做了一些说明。根据实际使用的led_flash类型定义flash节点;

1 如果是i2c控制的led_flash设备, 定义在i2c节点中;

1 如果是简单gpio设备, 定义在根节点中;

目前支持的2种led_flash驱动

1、sgm3784 (i2c控制类型, 双色温);

I2C控制的LED FLASH示例参考文档为:

SDK/kernel/Documentation/devicetree/bindings/media/i2c/sgm3784.txt

2、flash_rgb13h (gpio控制类型, 简单led);

GPIO控制的LED FLASH示例参考文档为:

SDK/kernel/Documentation/devicetree/bindings/leds/leds-rgb13h.txt

调试方法/步骤总结如下三点:

如何查看闪光灯驱动是否注册？

使用命令：`media-ctl -p` 查看是否包含如图的entity：

I2C控制的sgm3784注册后如下：

```
- entity 10: m00_b_sgm3784_led1 1-0030 (0 pad, 0 link)
    type V4L2 subdev subtype Flash
    device node name /dev/v4l-subdev4

- entity 11: m00_b_sgm3784_led0 1-0030 (0 pad, 0 link)
    type V4L2 subdev subtype Flash
    device node name /dev/v4l-subdev5
```

图 11 sgm3784注册上示例图

GPIO控制的flash_rgb13h注册后如下：

```
- entity 10: m00_b_ov8858 2-0036 <1 pad, 1 link>
    type U4L2 subdev subtype Sensor
    device node name /dev/v4l-subdev3
    pad0: Source
        [fmt:SBGGR10/3264x2448]
        -> "rockchip-mipi-dphy-rx":0 [ENABLED]

- entity 11: m00_b_gpio-flash flash-rgb13h <0 pad, 0 link>
    type U4L2 subdev subtype Flash
    device node name /dev/v4l-subdev4
```

图 12 rgb13h注册上示例图

如何使用V4L2工具控制闪光灯模式？

使用V4L2工具控制闪光灯模式

闪光灯驱动添加后，可以使用如下命令查看闪光灯是否变化：

- Flash off模式：

```
v4l2-ctl -d dev/video0 --set-ctrl 'led_mode=0'
```

- Flash模式：会闪一下

```
v4l2-ctl -d dev/video0 --set-ctrl 'led_mode=1'
```

```
v4l2-ctl -d dev/video0 --set-ctrl 'strobe=1'
```

- Torch模式：常亮

```
v4l2-ctl -d dev/video0 --set-ctrl 'led_mode=2'
```

或者

```
v4l2-ctl -d /dev/v4l-subdev6 --set-ctrl 'led_mode=2'
```

如果没有，自己先添加打印信息，排查调试闪光灯驱动：

camera3_profiles.xml如何配置？

如果需要支持flash， camera3_profiles.xml需要修改如下配置项：

```
<control.aeAvailableModes value="ON,OFF,ON_AUTO_FLASH,ON_ALWAYS_FLASH"/>

<flash.info.available value="TRUE"/>
```

然后清理camera APK， 重启：

配置好后： 抓取关键log： 例如： 闪光灯挂载在OV8858上时候的log如下：

开启Camera Hal3的log开关：

抓取LOG：

```
logcat -c;pkill provider;logcat | grep flash
```

或者：

```
logcat -c;pkill provider;logcat | grep -i attach
```

抓取的关键LOG如下：

```
D RkCamera: <HAL> PlatformData: findAttachedSubdevs:827,found flashlight
m00_b_gpio-flash flash-rgb13h attatched to sensor ov8858
```

示例案例： camera flash如何配置默认关闭？

问题描述

相机APK默认闪光灯是开启的，要求默认关闭。

问题分析

查看camera3_profiles.xml中支持闪光灯的配置如下：

```
<control.aeAvailableModes value="ON,ON_AUTO_FLASH,ON_ALWAYS_FLASH "/> <!-- remove
ON_AUTO_FLASH,ON_ALWAYS_FLASH if flash is not available-->
```

按照SDK/system/media/camera/docs/docs.html配置正确

android.control.aeAvailableModes	byte x n List of enums	public as enumList [legacy]	List of auto-exposure modes for android.control.aeMode that are supported by this camera device.	Any value listed in android.control.aeMode	3.2	BC
Details						
Not all the auto-exposure modes may be supported by a given camera device, especially if no flash unit is available. This entry lists the valid modes for android.control.aeMode for this camera device.						
All camera devices support ON, and all camera devices with flash units support ON_AUTO_FLASH and ON_ALWAYS_FLASH.						
FULL mode camera devices always support OFF mode, which enables application control of camera exposure time, sensitivity, and frame duration.						
LEGACY mode camera devices never support OFF mode. LIMITED mode devices support OFF if they support the MANUAL_SENSOR capability.						

图 13 aeAvailableModes配置说明

需要清除APK缓存再确认下，其他带闪光灯的客户应该都是这么配置的； 需要排查看下APK默认的flash选项是在哪里配置的。

解决办法

SDK/packages/apps/Camera2/res/values/strings.xml需要默认配置成false;



```
1024:
1025: static void __ov5648_power_off(struct ov5648 *ov5648)
1026: {
1027:     #if 0
1028:         if (!IS_ERR(ov5648->pwdn_gpio))
1029:             gpiod_set_value_cansleep(ov5648->pwdn_gpio, 0);
1030:         clk_disable_unprepare(ov5648->xvclk);
1031:         if (!IS_ERR(ov5648->reset_gpio))
1032:             gpiod_set_value_cansleep(ov5648->reset_gpio, 1);
1033:         regulator_bulk_disable(OV5648_NUM_SUPPLIES, ov5648->supplies);
1034:     #endif
1035: }
1036:
```

图 14 默认配置位置

注意事项

dts中闪光灯节点中配置的module index需要和对应camera sensor的module index一样;

例如: dts中OV13850节点配置module index如下

```
ov13850: ov13850@10 {
    compatible = "ovti,ov13850";
    .....
    rockchip,camera-module-index = <0>;
    .....
};
```

则 dts中对应闪光灯节点配置module index如下:

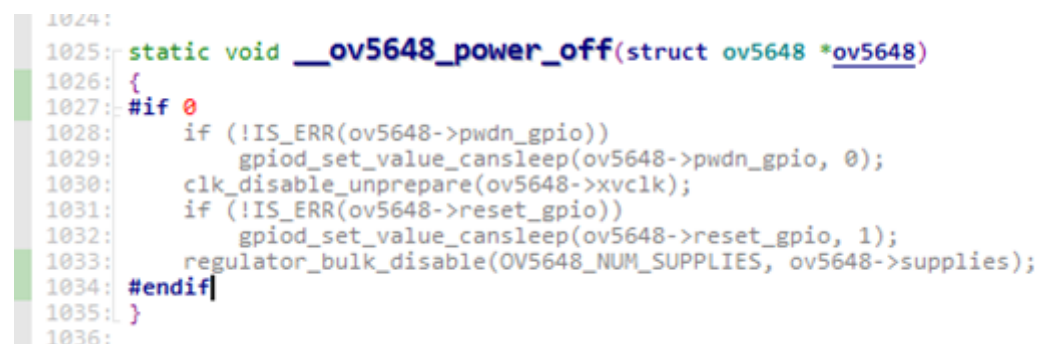
```
sgm3784: sgm3784@30 {
    compatible = "sgmicro,gsm3784";
    .....
    rockchip,camera-module-index = <0>;
    .....
};
```

I2C不通，如何确认相关信号？

Camera Sensor的I2C不通，则需要测量确认如下信号：Power/PowerDown/Reset/Mclk/I2cBus，

以OV5648正常调试为例进行说明。

需要将__ov5648_power_off()函数中相关下电操作屏蔽如下：



```
1024:
1025: static void __ov5648_power_off(struct ov5648 *ov5648)
1026: {
1027:     #if 0
1028:         if (!IS_ERR(ov5648->pwdn_gpio))
1029:             gpiod_set_value_cansleep(ov5648->pwdn_gpio, 0);
1030:         clk_disable_unprepare(ov5648->xvclk);
1031:         if (!IS_ERR(ov5648->reset_gpio))
1032:             gpiod_set_value_cansleep(ov5648->reset_gpio, 1);
1033:         regulator_bulk_disable(OV5648_NUM_SUPPLIES, ov5648->supplies);
1034:     #endif
1035: }
1036:
```

图 15 代码下电位置屏蔽示例图

测量Power情况

查看OV5648的Datasheet： AVDD、DVDD、DOVDD的供电要求分别如下：

01	DVDD	power	digital circuit power	1.5V
02	AGND	ground	analog ground	
03	AVDD	power	analog power	2.8V
04	DOGND	ground	I/O ground	
05	SIOD	I/O	SCCB interface data	
06	SIOC	input	SCCB interface input clock	
07	DVDD	power	digital circuit power	1.5V
08	SCK	I/O	SPI interface input clock	
09	SDI0	I/O	SPI interface data input 0	
10	SDI1	I/O	SPI interface data input 1	
11	DOVDD	power	I/O power	1.8/2.8V

图 16 OV5648供电要求

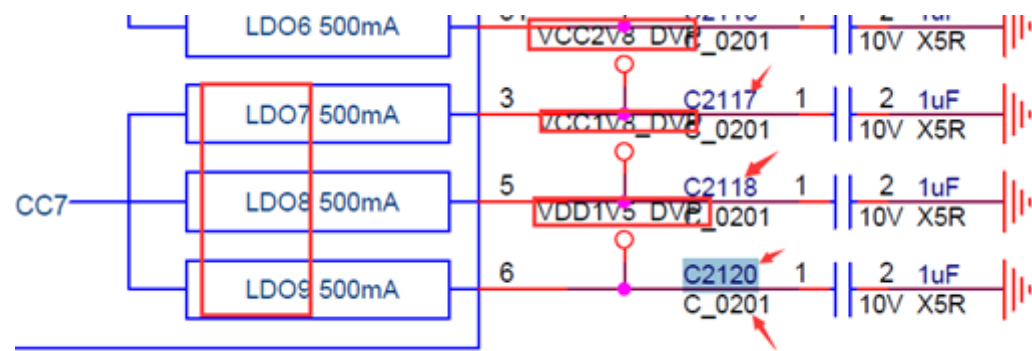


图 17 原理图供电电源

依次测量结果如下：

VCC2V8_DVP： 2.8V， 对应OV5648的AVDD；

VCC1V8_DVP： 1.8V， 对应OV5648的DOVDD；

VDD1V5_DVP： 1.5V， 对应OV5648的DVDD；

和Datasheet的供电要求一致， 正常。

异常分析

如果测量的电压和实际的不符， 需要硬件工程师检查各个供电分压是否正常；

PowerDown

查看OV5648的Datasheet： 有效时候为低电平：

18	PWDNB	input	power down (active low with pull down resistor)
----	-------	-------	---

图 18 PWDNB

那说明Sensor工作的时候需要为： 高电平

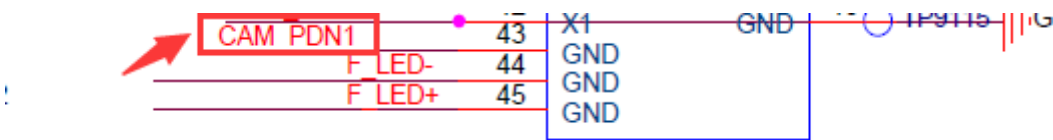


图 19 原理图PWD引脚

测量结果为：1.8V；正常

异常分析

如果测量的PowerDown电平和实际配置的GPIO输出电压不一致，那需要查看如下两个方面原因：

- 1、查看对应GPIO是否是被其他模块占用，是否配置为GPIO功能
- 2、检查硬件电路，查看中间是否有电阻虚焊或者漏焊问题；

Reset

查看OV5648的Datasheet：有效时候为低电平

17	RESETB	input	system reset (active low with pull up resistor)
----	--------	-------	---

图 20 RESET

那说明Sensor工作的时候需要为：高电平

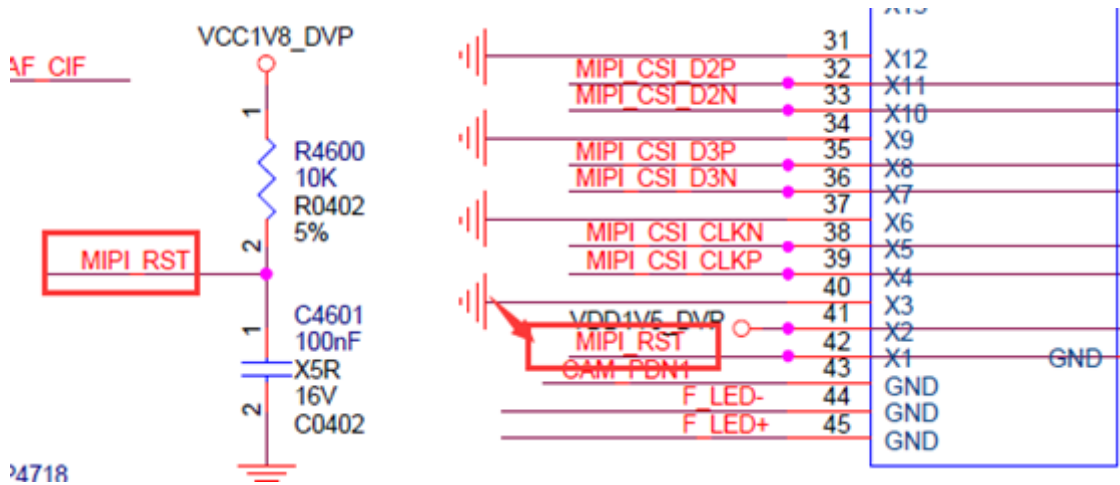


图 21 原理图RESET引脚

测量结果为：1.8V，被固定拉成1.8V了；正常。

异常分析

如果测量的Reset电平和实际配置的GPIO输出电压不一致，那需要查看如下两个方面原因：

- 1、查看对应GPIO是否是被其他模块占用，是否配置为GPIO功能
- 2、检查硬件电路，查看中间是否有电阻虚焊或者漏焊问题；

Mclk

测量结果如图：

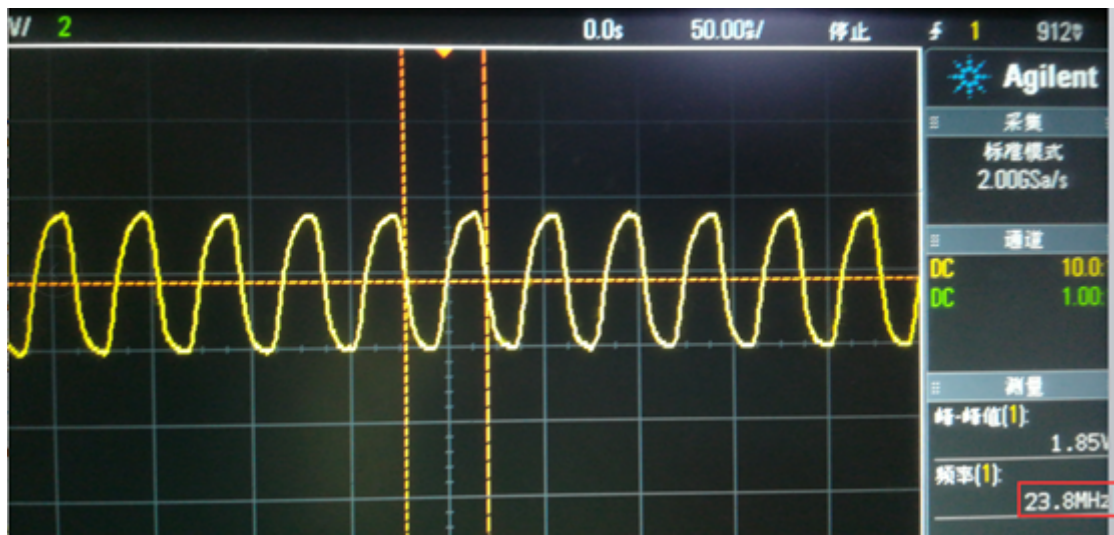


图 22 正常MCLK示例

24MHZ，1.8V左右，正常。

MCLK异常分析

测量到 MCLK 输出如图：结果只有 200mv 左右，或者有时直接没有

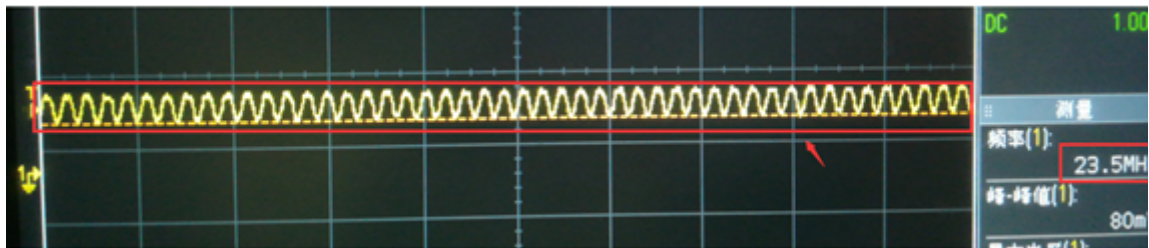


图 23 异常MCLK示例

MCLK异常的可能原因：

1、查看CIF_CLKOUT引脚复用没有配置正确

cat /sys/kernel/debug/pinctrl/pinctrl/pinmux-pins

```
pin 65 (gpio2-9): ff950000.cif-new (GPIO UNCLAIMED) function isp_pin group isp-d2d9
pin 66 (gpio2-10): ff950000.cif-new (GPIO UNCLAIMED) function isp_pin group isp-d2d9
pin 67 (gpio2-11): ff950000.cif-new (GPIO UNCLAIMED) function isp_pin group isp-mipi
pin 68 (gpio2-12): (MUX UNCLAIMED) (GPIO UNCLAIMED)
pin 69 (gpio2-13): (MUX UNCLAIMED) (GPIO UNCLAIMED)
```

图 24 rk3288 cif_clkout引脚复用查看示例

2、硬件上 CIF_CLKOUT对应的 IO 电源域未供电；

直接测量硬件上 CIF_CLKOUT 的 IO 电源供电情况；

3. 硬件上 CIF_CLKOUT 对应的 IO 电源域与软件配置 IO-domain 不匹配；

在：kernel/drivers/power/avs/rockchip-io-domain.c中可以查看原理图上电源域名称和

dts中配置的电源域名称对应关系；以下以RK3326为例说明如何正确配置 MCLK 的 IO-domain。

RK3326:

CIF_CLKO_MO 对应的电源域如图：为 VCCIO3；

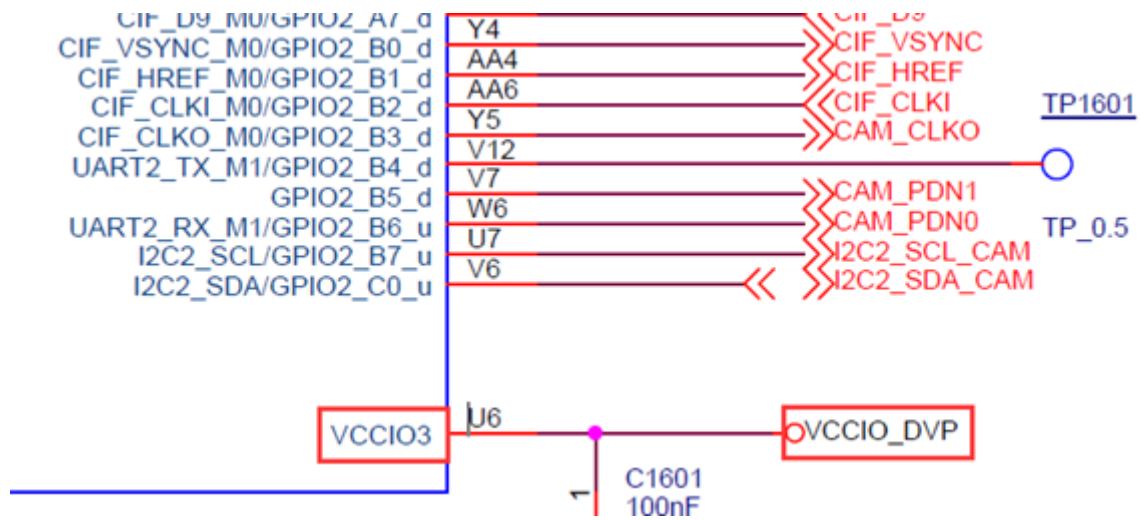


图 25 RK3326 MCLK对应电源域示例图

a.硬件VCCIO_DVP连接1.8V供电如图:

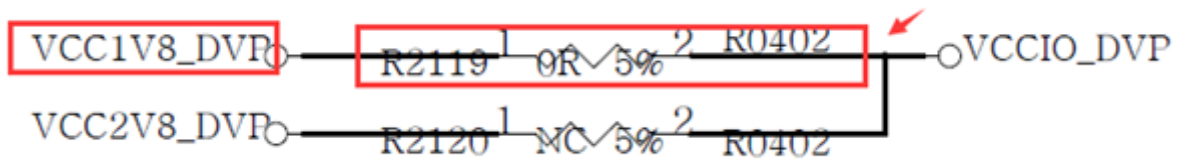


图 26 VCCIO_DVP连接1.8V供电

此时需要将vccio3-supply配置成 1.8v 电源域如下:

```

55
56 &io_domains {
57     status = "okay";
58
59     vccio1-supply = <&vcc_3v0>;
60     vccio2-supply = <&vccio_sd>;
61     vccio3-supply = <&vcc1v8_dvp>;
62     vccio4-supply = <&vcc_3v0>;
63     vccio5-supply = <&vcc_3v0>;
64 };

```

b.硬件VCCIO_DVP连接2.8V供电如图:

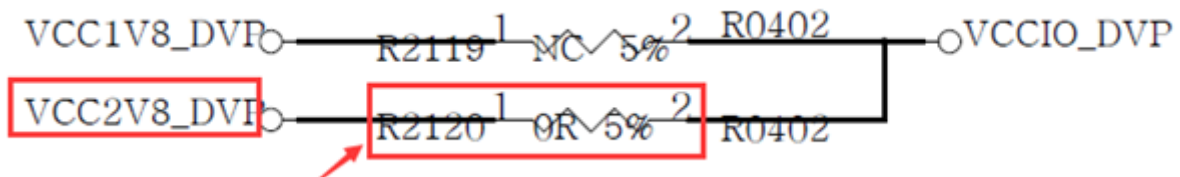


图 27 VCCIO_DVP连接2.8V供电

此时需要配置成 2.8V 电源域如下:

```

539 &io_domains {
540     status = "okay";
541
542     vccio1-supply = <&vcc_3v0>;
543     vccio2-supply = <&vccio_sd>;
544     vccio3-supply = <&vcc2v8_dvp>;
545     vccio4-supply = <&vcc_3v0>;
546     vccio5-supply = <&vcc_3v0>;
547 };

```

RK3566-Android11.0-MCLK测量没有

很大原因是没有添加power-domain，需要添加，示例如下：

```
&i2c4 {
    status = "okay";
    gc2145: gc2145@3c {
        status = "okay";
        compatible = "galaxycore,gc2145";
        reg = <0x3c>;
        clocks = <&cru CLK_CIF_OUT>;
        clock-names = "xvclk";
+       power-domains = <&power RK3568_PD_VI>;
        pinctrl-names = "default";
        pinctrl-0 = <&cif_dvp_8bit>;
        .....
    }
}
```

I2cBus

根据原理图确认是I2C2。

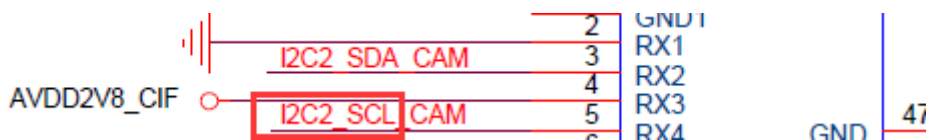


图 28 原理图I2C编号示例

查看DTS中OV5648确实配置在I2C2结点下，配置正确。

```
32
33 &i2c2 {
34     status = "okay";
35
36     ov5648: ov5648@36 {
37         status = "okay";
38         compatible = "ovti,ov5648";
39         reg = <0x36>;
40         clocks = <&cru SCLK_CIF_OUT>;
41         clock-names = "xvclk";
42     }
```

图 29 I2CBus正确示例图

异常分析

如果测量的I2C电平不对，需要检查I2C的供电是否正常，在无信号传输的情况下，I2C的两条总线是处于高电平状态；

如何使用v4l2获取底层数据流？

rkisp1配置链路 示例

目前RK的Camera驱动使用的media-ctl 框架，有以下四种常见链路：

- 1、 MIPI-RAW-Sensor -> MIPI接口->ISP
- 2、 MIPI-YUV-Sensor -> MIPI接口->ISP
- 3、 Parallel-Sensor ->ISP

4、Parallel-Sensor -> CIF

MIPI-RAW-Sensor -> MIPI interface->ISP

图 30 示例拓扑图1

以ov5648为例：配置链路命令如下：

```
media-ctl -l '"ov5648 2-0036":0->"rockchip-mipi-dphy-rx":0[1]'
```

配置好后，使用media-ctl -p命令查看如下：会显示ENABLED（下同），说明链路配置连接成功。

```
- entity 7: m01_b_ov5648 2-0036 <1 pad, 1 link>
  type V4L2 subdev subtype Sensor
  device node name /dev/v4l-subdev2
  pad0: Source
    [fmt:SBGGR10/2592x1944]
    -> "rockchip-mipi-dphy-rx":0 [ENABLED]
```

图 31 链路配置OK示例图

其他所需链路配置的命令：

```
media-ctl -l '"rockchip-mipi-dphy-rx":1->"rkisp1-isp-subdev":0[1] '
media-ctl -l '"rkisp1-input-params":0->"rkisp1-isp-subdev":1[1] '
media-ctl -l '"rkisp1-isp-subdev":2->"rkisp1_selfpath":0[1] '
media-ctl -l '"rkisp1-isp-subdev":2->"rkisp1_mainpath":0[1] '
media-ctl -l '"rkisp1-isp-subdev":3->"rkisp1-statistics":0[1]'
```

MIPI-YUV-Sensor -> MIPI interface->ISP

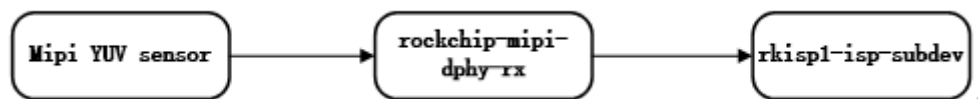


图 32 示例拓扑图2

以GC2145-MIPI-YUV为例：配置链路命令如下：

```
media-ctl -l '"gc2145 2-003c":0->"rockchip-mipi-dphy-rx":0[1] '
media-ctl -l '"rockchip-mipi-dphy-rx":1->"rkisp1-isp-subdev":0[1]'
```

其他同1

Parallel-Sensor -> ISP



图 33 示例拓扑图3

以GC2145-DVP为例：

```
media-ctl -l '"gc2145 2-003c":0->"rkisp1-isp-subdev":0[1]'

media-ctl -l '"rkisp1-input-params":0->"rkisp1-isp-subdev":1[1]'

media-ctl -l '"rkisp1-isp-subdev":2->"rkisp1_selfpath":0[1]'

media-ctl -l '"rkisp1-isp-subdev":2->"rkisp1_mainpath":0[1]'

media-ctl -l '"rkisp1-isp-subdev":3->"rkisp1-statistics":0[1]'
```

Parallel-Sensor -> CIF

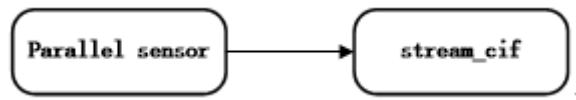


图 34 示例拓扑图4

以GC0329为例：

```
media-ctl -l '"gc0329 0-0031":0->"stream_cif":0[1]'
```

配置各entity格式

MIPI-RAW-Sensor -> MIPI interface->ISP

以ov5648为例：全分辨率2592x1944输出，配置各entity和pad的格式如下：

```
media-ctl --set-v4l2 '"ov5648 2-0036":0[fmt:SBGGR10/2592x1944]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":0[fmt:SBGGR10/2592x1944]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":0[fmt:SBGGR10/2592x1944]' --set-v4l2
'"rkisp1-isp-subdev":0[crop:(0,0)/2592x1944]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":2[fmt:YUYV2X8/2592x1944]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":2[fmt:YUYV2X8/2592x1944]' --set-v4l2
'"rkisp1-isp-subdev":2[crop:(0,0)/2592x1944]'
```

使用命令：

```
media-ctl --get-v4l2 '"rkisp1-isp-subdev":0' 或 media-ctl -p
```

查看设置的格式是否正确

```
127lr8638_mid:/ # media-ctl --get-v4l2 '"rkisp1-isp-subdev":0'
[fmt:SBGGR10/2592x1944
crop.bounds:(0,0)/2592x1944
crop:(0,0)/2592x1944]
```

图 35 分辨率配置正确示例图

MIPI-YUV-Sensor -> MIPI interface->ISP

以GC2145-MIPI-YUV为例：800x600输出

```
media-ctl --set-v4l2 '"gc2145 3-003c":0[fmt:UYVY2X8/800x600]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":0[fmt:UYVY2X8/800x600]'
media-ctl --set-v4l2 '"rkisp1-isp-subdev":0[fmt:UYVY2X8/800x600]' --set-v4l2
'"rkisp1-isp-subdev":0[crop:(0,0)/800x600]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":2[fmt:YUYV2X8/800x600]'

media-ctl --set-v4l2 '"rkisp1-isp-subdev":2[fmt:YUYV2X8/800x600]' --set-v4l2
'"rkisp1-isp-subdev":2[crop:(0,0)/800x600]'
```

MParallel-Sensor ->ISP

以GC2145-DVP为例：同2

MParallel-Sensor ->CIF

```
media-ctl --set-v4l2 '"gc0329 0-0031":0[fmt:YUYV2X8/640x480]'
```

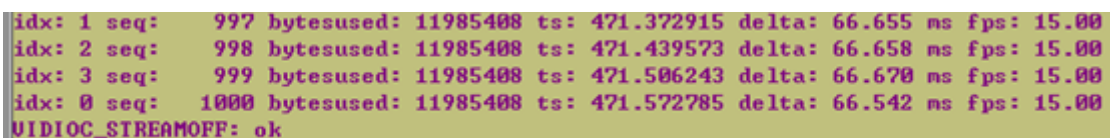
使用V4l2命令获取数据流

以上四种通路最后获取数据流的命令基本一致：使用命令：v4l2-ctl -h，查看使用方法，以ov5648为例：

```
v4l2-ctl --verbose -d /dev/video0 --set-fmt-
video=width=2592,height=1944,pixelformat='NV12' --stream-mmap=4 --set-
selection=target=crop,flags=0,top=0,left=0,width=2592,height=1944
```

如果是其他分辨率，需要修改下width和height；

如果数据流正常，打印如下：会打印出帧率。



```
idx: 1 seq: 997 bytesused: 11985408 ts: 471.372915 delta: 66.655 ms fps: 15.00
idx: 2 seq: 998 bytesused: 11985408 ts: 471.439573 delta: 66.658 ms fps: 15.00
idx: 3 seq: 999 bytesused: 11985408 ts: 471.506243 delta: 66.670 ms fps: 15.00
idx: 0 seq: 1000 bytesused: 11985408 ts: 471.572785 delta: 66.542 ms fps: 15.00
VIDIOC_STREAMOFF: ok
```

图 36 正常数据流输出示例图

v4l2-ctl及media-ctl工具说明

media-ctl 以及 v4l2-ctl是v4l-utils中包含的两个命令行工具。

l media-ctl，用以查看、配置拓扑结构

l v4l2-ctl，用以配置 v4l2 controls，可抓帧，设置 cif，isp，sensor 参数

v4l-utils 工具是由 Linuxtv维护的一个 V4L2 开发套件，它提供一系列 V4L2 及 media framework 相关的工具，用来配置 V4L2 子设备的属性，测试 V4L2 设备，并提供如 libv4l2.so开发库等等。代码下载地址：<https://www.linuxtv.org/downloads/v4l-utils/>

media-ctl 以及 v4l2-ctl两个命令工具可以通过上述地址自己下载编译或找FAE获取RK平台编译好的可执行文件。

v4l2命令获取数据流失败问题常见原因及调试方法

可能原因1：DVP接口相关的IOMUX未进行配置

以RK3368上调试GC2145-DVP为例进行说明。

问题现象

使用如下命令进行数据流获取：

```
v4l2-ctl --verbose -d /dev/video0 --set-fmt-  
video=width=800,height=600,pixelformat='NV12' --stream-mmap=4 --set-  
selection=target=crop,flags=0,top=0,left=0,width=800,height=600
```

但是结果如下：

```
VIDIOC_QUERYCAP: ok  
VIDIOC_G_FMT: ok  
VIDIOC_S_FMT: ok  
Format Video Capture Multiplanar:  
    Width/Height      : 800/600  
    Pixel Format       : 'NV12'  
    Field             : None  
    Number of planes  : 1  
    Flags              :  
    Colourspace       : Default  
    Transfer Function  : Default  
    YCbCr Encoding    : Default  
    Quantization      : Full Range  
    Plane 0           :  
        Bytes per Line : 800  
        Size Image     : 720000  
VIDIOC_REQBUFS: ok  
VIDIOC_QUERYBUF: ok  
VIDIOC_QUERYBUF: ok  
VIDIOC_QBUF: ok  
VIDIOC_QUERYBUF: ok  
VIDIOC_QBUF: ok  
VIDIOC_QUERYBUF: ok  
VIDIOC_QBUF: ok  
VIDIOC_QUERYBUF: ok  
VIDIOC_QBUF: ok  
VIDIOC_STREAMON: ok  
select timeout
```

就是没有数据流抓出来；

测量GC2145输出及输入的信号：CIF_CLKIN = 61MHZ

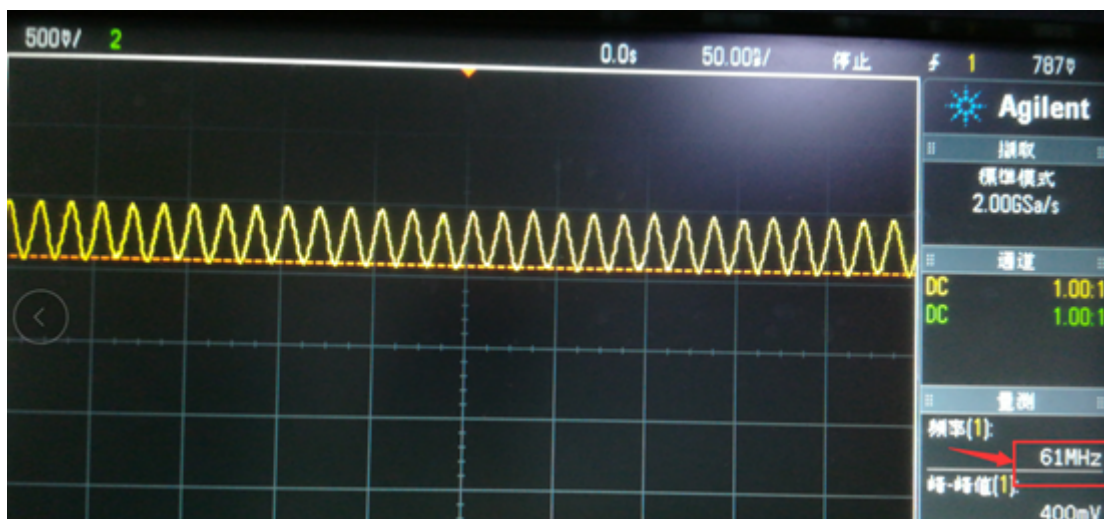


图 37 PCLK信号

输入给GC2145的CIF_CLKOUT = 24MHZ 正常，CIF_DATA引脚均有数据输出。

调试方法

使用IO命令查看当前DVP相关的GPIO的IOMMU是否正确

```
>isp_dvp_d2d9: isp-dvp-d2d9 {
>   rockchip,pins =
>   <1 0 RK_FUNC_1 &pcfg_pull_none>, //cif_data2
>   <1 1 RK_FUNC_1 &pcfg_pull_none>, //cif_data3
>   <1 2 RK_FUNC_1 &pcfg_pull_none>, //cif_data4
>   <1 3 RK_FUNC_1 &pcfg_pull_none>, //cif_data5
>   <1 4 RK_FUNC_1 &pcfg_pull_none>, //cif_data6
```

图 38 DVP接口部分GPIO示例

查看RK3368的Datasheet找到对应GPIO口IOMUX的寄存器地址：

GRF基地址：

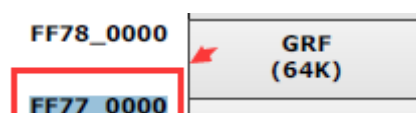


图 39 GRF基地址

对应GPIO_IOMUX地址：

GRF_GPIO1A_IOMUX	0x00000	W	0x00000000	GPIO1A iomux control
GRF_GPIO1B_IOMUX	0x00004	W	0x00000000	GPIO1B iomux control

图 40 GPIO_IOMUX地址

$$0xFF770000 + 0x00000 = 0xff770000$$

3:2	RW	0x0	gpio1a1_sel GPIO1A[1] iomux select 2'b00: gpio 2'b01: cif_data3 2'b10: ts_data1 2'b11: reserved
1:0	RW	0x0	gpio1a0_sel GPIO1A[0] iomux select 2'b00: gpio 2'b01: cif_data2

图 41 IOMUX复用位说明

使用命令：io -4 0xff770000，查看GPIO复用情况，结果如下：没有被赋值为：CIF_DATA；

```
rk3368:/ # io -4 0xff770000
ff770000: 00000000
rk3368:/ #
```

图 42异常IOMUX结果

而正常的如下：IOMUX正常赋值成了CIF_DATA引脚；

```
D:\Install Program Packages\plat
rk3368:/ # io -4 0xff770000
ff770000: 00005555
```

图 43 正常IOMUX结果

解决办法

在dts的rkisp1结点中把相应的pinctrl加上即可。

```
&rkisp1 {
    status = "okay";
    pinctrl-names = "default";
    pinctrl-0 = < &cif_clkout &isp_dvp_d2d9 &isp_dvp_d10d11 >;
```

可能原因2：模组供电少了

问题现象

以调试OV5648为例，导致Check ID虽然成功了，但后面寄存器写失败。

调试方法

在static int ov5648_write_reg()函数中添加调试LOG如下：

```
687: static int ov5648_write_reg(struct i2c_client *client, u16 reg,
688:                             u32 len, u32 val)
689: {
690:     u32 buf_i, val_i;
691:     u8 buf[6];
692:     u8 *val_p;
693:     __be32 val_be;
694:
695:     dev_info(&client->dev, "%s(%d) enter!\n", __func__, __LINE__);
696:     dev_info(&client->dev, "ov5648 write reg(0x%x val:0x%x)!\n", reg, val);
697:
```

图 44打印信息添加示例

然后核对打印信息，查看寄存器值是否写入OK。因为OV5648模组需要1.5v供电；而dts中如果未提供，可能导致Check ID虽然成功了，但后面寄存器写失败。

解决办法

参考原理图，在Sensor的结点配置加上对应供电：dvdd-supply = <&vcc1v5_dvp>。

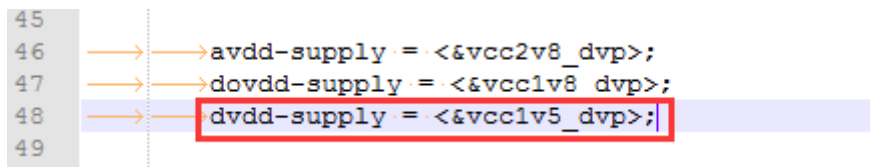


图 45 添加DVDD供电

可能原因3：驱动中配置的MIPI速率不对

问题现象

以RK3326-Android9.0上调试OV5648为例。

使用命令：cat /proc/interrupts | grep isp，查看ISP中断情况如下：

```
console:/ # cat /proc/interrupts | grep isp
37:          0          0          0          0      GICv2 102 Level   rkisp1
console:/ #
console:/ #
console:/ #
```

图 46 isp中断

没有中断过来，说明isp这边没有收到数据。

调试方法

使用如下命令进行数据流获取：

```
v4l2-ctl --verbose -d /dev/video0 --set-fmt-
video=width=2592,height=1944,pixelformat='NV12' --stream-mmap=4 --set-
selection=target=crop,flags=0,top=0,left=0,width=2592,height=1944
```

配置好后，使用命令：io -4 -l 0x100 0xff4a1c00，查看ISP识别数据情况：

```
r8638_mid:/ # io -4 -l 0x100 0xff4a1c00
ff4a1c00: 00021001 00002c00 03f33333 00200332
ff4a1c10: 00200332 00000000 00000000 000000c2
ff4a1c20: 0000002b 000000ff 000000ff 000000ff
ff4a1c30: 000000ff 00000000 00000000 00000000
ff4a1c40: 00000000 00000000 00000000 00000000
ff4a1c50: 00000000 00000000 001f6ad3 22222222
ff4a1c60: 000002f7 00000000 00000000 00000000
ff4a1c70: 00000000 00000000 00000000 00000000
ff4a1c80: 00000000 00000000 00000000 00000000
```

图 47 ISP识别数据类型情况

识别到的数据类型异常。

尝试解决办法

将MIPI_FREQ改成210MHz。

```
40:
41: /* pixel_rate = link_frequency * 2 * lanes / BITS_PER_SAMPLE */
42: #define MIPI_FREQ 210000000U
...
471:
472: static const s64 link_freq_menu_items[] = {
473:     MIPI_FREQ
474: };
475:
```

图 48 MIPI速率配置修改

结果如下：数据流通了；

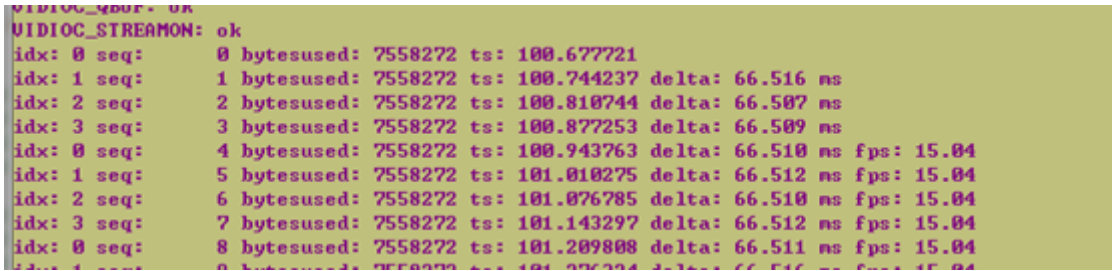


图 49 数据流通示例

可能原因4：模组的MIPI CLK P/N或data P/N 和主板的P/N反了；

问题现象

使用io命令查看phy stop state的跳变；

调试方法

找硬件配合核对原理图；

解决办法

如果确实反了，可以查看Sensor是否支持MIPI CLK/Data P/N反向；

如果不行只能重新打样模组；

常见更新SDK后，摄像头打不开问题分析排查步骤

先将camera3_profiles.xml中 先改成SOC类型

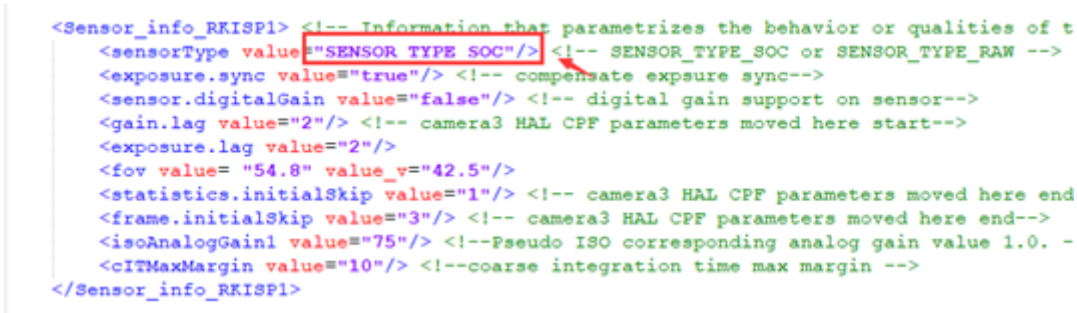


图 50 SENSOR TYPE修改示例图

如果不会报错，说明tuning文件版本不对；请联系对应tuning调试工程师进行版本更新。因为对于输出RAW格式的Camera来说，SDK更新过程中，camera_engine_rkisp的代码有更新tuning参数结构可能有修改，从而导致更新SDK后摄像头打不开问题。

如果是SOC的Camera，可以忽略这步，跳到下一步排查。

如果还报错，先重启，然后使用v4l2-ctl工具底层抓数据流,看下是否正常

如果正常，说明是camera3_profiles.xml配置问题，对照最新SDK目录下对应平台的xml配置进行修改；

如果不正常，说明底层驱动问题，排查dts是否配置正确，驱动和最新SDK 目录下驱动对照修改。

如何判断驱动中摄像头配置是RAW还是SOC类型？

如果驱动中xxxx_enum_mbus_code配置为MEDIA_BUS_FMT_UYVY/YUYV/VYUY等，则为SOC摄像头；如果需要调试效果，找对应Sensor原厂。

例如：GC2145输出YUV数据配置如下：

```
1935: static const struct gc2145_pixfmt gc2145_formats[] = {
1936: {
1937:     .code = MEDIA_BUS_FMT_UYVY8_2X8,
1938: }
1939: };
1940:
```

图 51 SOC类型示例

如果驱动中xxxx_enum_mbus_code配置为MEDIA_BUS_FMT_SBGGR/SRGGB/SGBRG等，则为RAW摄像头，如果需要调试效果，找业务申请；

例如：OV5648输出RAW数据配置如下：

```
685: static int ov5648_enum_mbus_code(struct v4l2_subdev *sd,
686:                                struct v4l2_subdev_pad_config *cfg,
687:                                struct v4l2_subdev_mbus_code_enum *code)
688: {
689:     if (code->index != 0)
690:         return -EINVAL;
691:     code->code = MEDIA_BUS_FMT_SBGGR10_1X10;
692:     return 0;
693: }
694:
```

图 52 RAW类型示例

如何查看驱动注册后各video/v4l-subdev结点名称信息？

查看video结点名称信息：

使用命令：

```
grep -H '' /sys/class/video4linux/video*/name
```

```
C:\Users\lzw>adb shell
rk3288_Android10:/ # grep -H '' /sys/class/video4linux/video*/name
/sys/class/video4linux/video0/name:rkisp_mainpath
/sys/class/video4linux/video1/name:rkisp_selfpath
/sys/class/video4linux/video2/name:rkisp-statistics
/sys/class/video4linux/video3/name:rkisp-input-params
/sys/class/video4linux/video4/name:rkisp-mipi-luma
rk3288_Android10:/ #
```

v4l-subdev结点名称信息

使用命令：

```
grep -H '' /sys/class/video4linux/v4l-subdev*/name
```

```
CRTB1001:/ #
CRTB1001:/ # grep -H '' /sys/class/video4linux/v4l-subdev*/name
/sys/class/video4linux/v4l-subdev0/name:rkisp1-isp-subdev
/sys/class/video4linux/v4l-subdev1/name:m01_f_gc0312 2-0021
/sys/class/video4linux/v4l-subdev2/name:m00_b_gc2145 2-003c
CRTB1001:/ #
```

RK356x-接DVP-RAW要求注意事项

如果接isp注意如下限制

Sensor的VSYNC有效必须为高有效，如果不是，需修改sensor寄存器改polarity成高：

RK356x dvp-raw接isp，sensor输出改成高有效

sensor输出改成高有效，flag对应改，其他芯片也兼容。

示例：sc031gs：

16'h3d08	DVP_POL_CTRL	8'h01	RW	Bit[2]: LREF polarity Bit[1]: FSYNC polarity Bit[0]: PCLK polarity
----------	--------------	-------	----	--

尝试修改sc031gs的极性，前面是hsync: 1; vsync: 0,pclk: falling有效；值为0x3d08: 0x00

改成：0x3d08: 0x03 即hsync:1 vsync:1, pclk: rising有效；

```
@@ -208,20 +208,29 @@ static const struct regval sc031gs_global_regs[] = {
    {0x36fa, 0x3a},
    {0x36fc, 0x01},
    {0x3908, 0x91},
-   {0x3d08, 0x00}, //0x01
+   {0x3d08, 0x03}, //0x01
```

DVP-RAW摄像头对应所需的提交

Kernel:

Isp所需提交

```
commit 3fe01f2b12f4f5d82671014732c5056b99a73eaa
Author: Cai YiWei <cyw@rock-chips.com>
Date:   Fri Oct 8 14:16:45 2021 +0800

media: rockchip: isp: rawwr and rawrd memory mode

Three mode:
0: raw12/raw10/raw8 8bit memory compact
1: raw12/raw10 16bit memory one pixel
   big endian for rv1126/rv1109
   |15|14|13|12|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
   | 3| 2| 1| 0| -| -| -| -|11|10| 9| 8| 7| 6| 5| 4|
   little align for rk356x
   |15|14|13|12|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
   | -| -| -| -|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
2: raw12/raw10 16bit memory one pixel
   big align for rv1126/rv1109/rk356x
   |15|14|13|12|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
   |11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0| -| -| -| -|

Change-Id: Iabd5600d1a880057f0a20e187b15d337079a14c6
Signed-off-by: Cai YiWei <cyw@rock-chips.com>
Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
```

cif驱动所需提交:

```
commit 74d87ea5ea8b596b23e6fa022511bc6aed4b1230
Author: Zefa Chen <zefa.chen@rock-chips.com>
Date: Sat Oct 9 17:23:28 2021 +0800

media: rockchip: cif support config memory mode

cif memory mode
0: raw12/raw10/raw8 8bit memory compact
1: raw12/raw10 16bit memory one pixel
   low align for rv1126/rv1109/rk356x
   |15|14|13|12|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
   | -| -| -| -|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
2: raw12/raw10 16bit memory one pixel
   high align for rv1126/rv1109/rk356x
   |15|14|13|12|11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0|
   |11|10| 9| 8| 7| 6| 5| 4| 3| 2| 1| 0| -| -| -| -|

note: rv1109/rv1126/rk356x dvp only support uncompact mode,
      and can be set low align or high align

Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
Change-Id: I59d619645650dfa10c9b2c168d8c741292f9f90f
```

RK356x-BT1120-4路VC芯片NVP6158-DTS配置说明

参考示例: `SDK/kernel/Documentation/devicetree/bindings/media/i2c/nvp6158.txt`

完整示例配置

```
nvp6158: nvp6158@30 {
    compatible = "nvp6158-v4l2";
    status = "okay";
    reg = <0x30>;
    clocks = <&cru CLK_CIF_OUT>;
    clock-names = "xvclk";
    power-domains = <&power RK3568_PD_VI>;
    pinctrl-names = "default";
    pinctrl-0 = <&cif_clk &cif_dvp_clk &cif_dvp_bus8 &cif_dvp_bus16>;
    pwr-gpios = <&gpio4 RK_PB2 GPIO_ACTIVE_HIGH>;
    pwr2-gpios = <&gpio4 RK_PB1 GPIO_ACTIVE_HIGH>;
    rst-gpios = <&gpio4 RK_PB5 GPIO_ACTIVE_HIGH>;
    /*rst2-gpios = <&gpio2 RK_PC5 GPIO_ACTIVE_HIGH>;*/
    /*pwn-gpios = <&gpio4 RK_PA6 GPIO_ACTIVE_HIGH>;*/
    /*pwn2-gpios = <&gpio4 RK_PA6 GPIO_ACTIVE_HIGH>;*/
    rockchip,camera-module-index = <0>;
    rockchip,camera-module-facing = "back";
    rockchip,camera-module-name = "default";
    rockchip,camera-module-lens-name = "default";
    rockchip,dvp_mode = "BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <4>; //channel nums, 1/2/4
    rockchip,dual_edge = <1>; // pclk dual edge, 0/1
    rockchip,default_rect= <1920 1080>; // default resolution
    port {
```

```

        nvp6158_out: endpoint {
            remote-endpoint = <&dvp_in_bcam1>;
        };
    };

    &rkcif {
        status = "okay";
    };

    &rkcif_dvp {
        status = "okay";
        ports {
            #address-cells = <1>;
            #size-cells = <0>;

            port@0 {
                #address-cells = <1>;
                #size-cells = <0>;
                /* Parallel bus endpoint */
                dvp_in_bcam1: endpoint@1 {
                    reg = <1>;
                    remote-endpoint = <&nvp6158_out>;
                    bus-width = <16>;
                };
            };
        };
    };

    &rkcif_mmu {
        status = "disabled";
    };

    &rkcif_mipi_lvds {
        status = "disabled";
    };

    &rkcif_mipi_lvds_sditf {
        status = "disabled";
    };

```

各部分配置说明

如下部分是和camera相同的配置

参考Rockchip_Driver_Guide_ISP2x_CN_v1.x.x.pdf文件说明即可

```

nvp6158: nvp6158@30 {
    compatible = "nvp6158-v4l2";
    status = "okay";
    reg = <0x30>;
    clocks = <&cru CLK_CIF_OUT>;
    clock-names = "xvclk";
    power-domains = <&power RK3568_PD_VI>;
    pinctrl-names = "default";
    pinctrl-0 = <&cif_clk &cif_dvp_clk &cif_dvp_bus8 &cif_dvp_bus16>;
    pwr-gpios = <&gpio4 RK_PB2 GPIO_ACTIVE_HIGH>;

```

```

        pwr2-gpios = <&gpio4 RK_PB1 GPIO_ACTIVE_HIGH>;
        rst-gpios = <&gpio4 RK_PB5 GPIO_ACTIVE_HIGH>;
        /*rst2-gpios = <&gpio2 RK_PC5 GPIO_ACTIVE_HIGH>;*/
        /*pwn-gpios = <&gpio4 RK_PA6 GPIO_ACTIVE_HIGH>;*/
        /*pwn2-gpios = <&gpio4 RK_PA6 GPIO_ACTIVE_HIGH>;*/
        rockchip,camera-module-index = <0>;
        rockchip,camera-module-facing = "back";
        rockchip,camera-module-name = "default";
        rockchip,camera-module-lens-name = "default";
        .....
    };

    &rkcif {
        status = "okay";
    };

    &rkcif_dvp {
        status = "okay";
        ports {
            #address-cells = <1>;
            #size-cells = <0>;

            port@0 {
                #address-cells = <1>;
                #size-cells = <0>;
                /* Parallel bus endpoint */
                dvp_in_bcam1: endpoint@1 {
                    reg = <1>;
                    remote-endpoint = <&nvp6158_out>;
                    bus-width = <16>;
                };
            };
        };
    };

    &rkcif_mmu {
        status = "disabled";
    };

    &rkcif_mipi_lvds {
        status = "disabled";
    };

    &rkcif_mipi_lvds_sditf {
        status = "disabled";
    };

```

nvp6158驱动私有配置说明

1、配置nvp6158的输出模式

```
rockchip,dvp_mode = "BT1120"; //BT656 or BT1120 or BT656_TEST
```

因为nvp6158支持BT656 P格式，BT656 I测试格式，及BT1120 P格式的输出，通过上面可以配置相应的输出模式；

2、配置nvp6158的输出通道数

```
rockchip,channel_nums = <4>; //channel nums, 1/2/4
```

3、配置nvp6158输出的pclk采样极性

```
rockchip,dual_edge = <1>; // pclk dual edge, 0/1
```

支持单边沿和双边沿配置，0表示单边沿输出模式；1表示双边沿输出模式；

4、配置默认分辨率

```
rockchip,default_rect= <1920 1080>; // default resolution
```

支持：1080P，720P，及测试用的BT656I格式，具体见驱动；

BT656和BT1120配置区别

BT1120格式是16bit的数据位：pinctrl配置如下：

```
pinctrl-names = "default";

pinctrl-0 = <&cif_clk &cif_dvp_clk &cif_dvp_bus8 &cif_dvp_bus16>;
```

BT656格式是8bit的数据位：pinctrl配置如下：

```
pinctrl-names = "default";

pinctrl-0 = <&cif_clk &cif_dvp_clk &cif_dvp_bus16>;
```

BT656一路I格式输出配置示例

分辨率：720x480I

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT656_TEST "; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <720 480>; // default resolution
};

&rkCIF_dvp {
    status = "okay";

    port {
        #address-cells = <1>;
        #size-cells = <0>;
        /* Parallel bus endpoint */
        dvp_in_bcam1: endpoint@1 {
            reg = <1>;
            remote-endpoint = <&nvp6158_out>;
            bus-width = <8>;
        };
    };
};
```

分辨率：720x576I

rkcif_dvp 配置同上：

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT656_TEST "; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <720 576>; // default resolution
};
```

分辨率：960x576I

rkcif_dvp 配置同上：

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT656_TEST "; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <960 576>; // default resolution
};
```

分辨率：1920x576I

rkcif_dvp 配置同上：

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT656_TEST "; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1920 576>; // default resolution
};
```

BT656一路P格式输出配置示例

分辨率：1280x720P

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT656"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1280 720>; // default resolution
};

&rkcif_dvp {
    status = "okay";
    port {
```

```

        #address-cells = <1>;
        #size-cells = <0>;
        /* Parallel bus endpoint */
        dvp_in_bcam1: endpoint@1 {
            reg = <1>;
            remote-endpoint = <&nvp6158_out>;
            bus-width = <8>;
        };
    };
};

```

分辨率：1920x1080P

rkcif_dvp 配置同上：

```

nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT656"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1920 1080>; // default resolution
};

```

BT656-2路无支持格式

BT656-4路无支持格式

BT1120一路P格式输出配置示例

分辨率：1280x720P

```

nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1280 720>; // default resolution
};

&rkcif_dvp {
    status = "okay";
    port {
        #address-cells = <1>;
        #size-cells = <0>;
        /* Parallel bus endpoint */
        dvp_in_bcam1: endpoint@1 {
            reg = <1>;
            remote-endpoint = <&nvp6158_out>;
            bus-width = <16>;
        };
    };
};
};

```

分辨率：1920x1080P

rkcif_dvp 配置同上：

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <1>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1920 1080>; // default resolution
};
```

BT1120-2路P格式输出配置示例

分辨率：1280x720P只支持只能识别一路，两路相位差180度

```
i2ctransfer -y -f 1 w2@0x30 0xff 0x01
```

```
i2ctransfer -y -f 1 w1@0x30 0xcd r1
```

调整时序: `i2ctransfer -y -f 1 w2@0x30 0xcd 0x`

0xcd/0xce 为0x1f的时候，识别到ID0， 为0x06的时候，识别到ID1

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <2>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1280 720>; // default resolution
};

&rkcif_dvp {
    status = "okay";
    port {
        #address-cells = <1>;
        #size-cells = <0>;
        /* Parallel bus endpoint */
        dvp_in_bcam1: endpoint@1 {
            reg = <1>;
            remote-endpoint = <&nvp6158_out>;
            bus-width = <16>;
        };
    };
};
```

分辨率：1920x1080P

rkcif_dvp 配置同上：

```

nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <2>; //channel nums, 1/2/4
    rockchip,dual_edge = <0>; // pclk dual edge, 0/1
    rockchip,default_rect= <1920 1080>; // default resolution
};

```

BT1120-4路P格式输出配置示例

分辨率：1280x720P

```

nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <4>; //channel nums, 1/2/4
    rockchip,dual_edge = <1>; // pclk dual edge, 0/1
    rockchip,default_rect= <1280 720>; // default resolution
};

&rkcif_dvp {
    status = "okay";
    port {
        #address-cells = <1>;
        #size-cells = <0>;
        /* Parallel bus endpoint */
        dvp_in_bcaml: endpoint@1 {
            reg = <1>;
            remote-endpoint = <&nvp6158_out>;
            bus-width = <16>;
        };
    };
};

```

注意：这里需要配置pclk为双边沿采样才行；

如果配置成单边沿：nvp6158_video.c—》nvp6158_set_portmode

NVP6158_OUTMODE_4MUX_BT1120S中需要使用如下配置才行；

```

1531:  */
1532:  */
1533:  */
1534:  */
1535:  */
1536:  */
1537:  */
1538:  */
1539:  */
1540:  */
1541:  */
1542:  */
1543:  */
1544:  */
1545:  */
1546:  */
1547:  */
1548:  */
1549:  */
1550:  */

case NVP6158_OUTMODE_4MUX_BT1120S:
    gpio_i2c_write(chipaddr, 0xFF, 0x00);
    gpio_i2c_write(chipaddr, 0x56, 0x32);
    gpio_i2c_write(chipaddr, 0xFF, 0x01);
    if(chip_id[chip] == NVP6158C_R0_ID || chip_id[chip] == NVP6168C_R0_ID)
    {
        /* 6158C makes 2 bt656 ports to 1 bt1120 port. portsel=[1,2] to ch
        gpio_i2c_write(chipaddr, 0xC2, 0x54);
        gpio_i2c_write(chipaddr, 0xC3, 0x76);
        gpio_i2c_write(chipaddr, 0xC4, 0xdc);
        gpio_i2c_write(chipaddr, 0xC5, 0xfe);
        gpio_i2c_write(chipaddr, 0xC8, 0x88);
        gpio_i2c_write(chipaddr, 0xC9, 0x88);
        //dual_edge
        gpio_i2c_write(chipaddr, 0xCD, 0x06); //74.25MHz clock
        gpio_i2c_write(chipaddr, 0xCE, 0x06); //74.25MHz clock
        //single edge
        gpio_i2c_write(chipaddr, 0xCD, 0x46); //148.5MHz clock
        gpio_i2c_write(chipaddr, 0xCE, 0x46); //148.5MHz clock
    }

```

分辨率：1920x1080P

rkcif_dvp 配置同上：

```
nvp6158: nvp6158@30 {
    .....
    rockchip,dvp_mode = " BT1120"; //BT656 or BT1120 or BT656_TEST
    rockchip,channel_nums = <4>; //channel nums, 1/2/4
    rockchip,dual_edge = <1>; // pclk dual edge, 0/1
    rockchip,default_rect= <1920 1080>; // default resolution
};
```

注意：这里需要配置pclk为双边沿采样才行；

注意：以上均是color Bar输出，接AHD的只验证过1080P；

RK356x-BT1120-4路VC芯片NVP6158-驱动配置注意事项

前面讲述dts中将bit位配置为16bit，用于BT1120传输；

而实现BT1120传输，对应sensor的驱动驱动中需要实现：*_querystd函数，

从而提供参数给vicap，告诉vicap当前sensor输出的是bt656还是bt1120模式；

```
231: static int nvp6158_querystd(struct v4l2_subdev *sd, v4l2_std_id *std)
232: {
233:     struct nvp6158 *nvp6158 = to_nvp6158(sd);
234:
235:     if ((nvp6158->mode->BT656I_TEST_MODES) &&
236:         (nvp6158->mode-<NVP6158_DVP_MODES_END)) {
237:         /* for vicap detect bt1120 */
238:         *std = V4L2_STD_ATSC;
239:     } else {
240:         *std = V4L2_STD_PAL;
241:     }
242:     return 0;
243: }
```

其中*std = V4L2_STD_ATSC;表示输出的是BT1120模式；其他则表示BT656，见如下匹配代码：

```
2116:
2117: ret = v4l2_subdev_call(sensor_info->sd, video, querystd, &std);
2118: if (ret == 0) {
2119:     /* retrieve std from sensor if exist */
2120:     switch (std) {
2121:         case V4L2_STD_NTSC:
2122:             mode = INPUT_MODE_NTSC;
2123:             break;
2124:         case V4L2_STD_PAL:
2125:             mode = INPUT_MODE_PAL;
2126:             break;
2127:         case V4L2_STD_ATSC:
2128:             mode = INPUT_MODE_BT1120;
2129:             break;
2130:         default:
2131:             v4l2_err(&dev->v4l2_dev,
2132:                 "std: %lld is not supported", std);
2133:     }
```

RK356x-MIPI-4路VC芯片NVP6324-DTS配置说明

RK356x-11.0 –AHD-MIPI多路驱动调试问题

问：是否有调过4路AHD转MIPI输出接口的VC4通道芯片

答：我司调过的这类芯片有NVP6324；

问：能提供下NVP6324相关的配置说明吗，dts配置、参考代码之类的？

答：见rk3568-evb1-bt1120-nvp6158.dtsi

VC通道是MIPI CSI本身支持的功能，根据DI决定就行了，只要按照标准传输各个通道数据就能正常接收；数据标识（DI）高2bit用于虚拟通道，低6bit是数据类型。

参考示例：`SDK/kernel/Documentation/devicetree/bindings/media/i2c/nvp6324.txt`

示例配置

```
&i2c4 {
    status = "okay";

    jaguar1: jaguar1@30 {
        compatible = "jaguar1-v4l2";
        status = "okay";
        reg = <0x30>;
        clocks = <&cru CLK_CAM0_OUT>;
        clock-names = "xvclk";
        power-domains = <&power RK3568_PD_VI>;
        pd-gpios = <&gpio4 RK_PB4 GPIO_ACTIVE_HIGH>;
        rst-gpios = <&gpio3 RK_PB6 GPIO_ACTIVE_HIGH>;
        rockchip,camera-module-index = <0>;
        rockchip,camera-module-facing = "back";
        rockchip,camera-module-name = "jaguar1";
        rockchip,camera-module-lens-name = "jaguar1";
        rockchip,default_rect= <1920 1080>; // default resolution
        port {
            n4_out: endpoint {
                remote-endpoint = <&mipi_in_ucam3>;
                data-lanes = <1 2 3 4>;
            };
        };
    };
};
```

N4驱动私有配置说明

配置默认分辨率

```
rockchip,default_rect= <1920 1080>; // default resolution
```

支持：1080P，720P；

RK356x-多路VC芯片常见问题

问1: 现在 mipi vc 4路进来有进isp吗?

答: 没进isp, 用的vicap采

问2: 那的bt1120 vc 也用vicap采?

答: 是的;

问3: 那不分时搞?

答: 不用, 可以同时使用;

问4: 这样可以? 没有明白

答:

1、BT1120和MIPI接口是独立的, 所以支持同时采集;

2、vicap中bt1120和mipi相关的寄存器也是独立的, 支持同时配置;

所以支持同时采集

就类似于两个不同接口的, 但型号相同的camera, 例如MIPI-GC2145+ DVP GC2145;

CIF控制器根据dts配置中接口类型, 注册多个media结点, 将MIPI和BT1120收取独立处理

问5: 那类似有8个camera 节点了

答: 对, 实际出流的video数量是有8个

有点不足的就是bt1120采集的4个VC通道没有单独控制enable disable寄存器, MIPI采集的4路VC就可以单独控制开关

RK356x-11.0-camera如何打开HDR?

确认CMOS sensor是否本身支持HDR模式?

选择CMOS sensor前询问sensor原厂是否支持HDR模式, 如果不支持, 肯定不能配置成HDR模式; 只能配置成NORMAL的线性模式;

如何配置camera3_profiles.xml支持HDR?

关键字段: aiq.workingMode;

正常线性模式配置示例:

```
<aiq.workingMode value=" NORMAL "/> <!-- NORMAL or HDR2 or HDR3 -->
```

HDR2模式配置示例: 即2帧合成一帧的hdr模式

```
<aiq.workingMode value=" HDR2 "/> <!-- NORMAL or HDR2 or HDR3 -->
```

HDR3模式配置示例: 即3帧合成一帧的hdr模式

```
<aiq.workingMode value=" HDR3"/> <!-- NORMAL or HDR2 or HDR3 -->
```

RK356x-11.0-AHD to DVP接口常见转换芯片

cvbs转BT1120的有： TP9930、nvp6158c

cvbs转bt656的有客户使用： rn6752/tw9930/pr2000;

录像相关常见问题

跟录像相关的文件？

板子上位置： /vendor/etc/media_profiles_V1_0.xml

SDK位置： sdk/device/rockchip/rkxxx/media_profiles_default.xml

如何修改录像分辨率？

默认录像配置最高分辨率为480P，如想修改，先确认Camera是否支持更高分辨率，

如果支持，可以尝试修改对应的分辨率。

Android-9.0-13.0版本的USB camera常见问题咨询

Android9.0及以上版本的USB camera代码位置：

```
Platform/hardware/interfaces/camera/device/3.4/default
hardware/interfaces/camera/device/3.4/default$ mm -j12
```

编译生成： camera.device@3.4-external-impl.so

在： vendor/lib目录下，所以修改库后，直接编译替换这个库即可；

sdk中添加usb camera支持的参考文档：

<https://redmine.rock-chips.com/documents/53>中的

USB_UVC_Integrated_Cameras.pdf

usb camerahal 的log打开支持：

```
diff --git a/camera/device/3.4/default/ExternalCameraDevice.cpp
b/camera/device/3.4/default/ExternalCameraDevice.cpp
old mode 100644
new mode 100755
index 9a2fddf..ac3b8e7
--- a/camera/device/3.4/default/ExternalCameraDevice.cpp
+++ b/camera/device/3.4/default/ExternalCameraDevice.cpp
@@ -15,7 +15,7 @@
 */
\#define LOG_TAG "ExtCamDev@3.4"
-//#define LOG_NDEBUG 0
+#define LOG_NDEBUG 0
\#include <log/log.h>
\#include <algorithm>
diff --git a/camera/device/3.4/default/ExternalCameraDeviceSession.cpp
b/camera/device/3.4/default/ExternalCameraDeviceSession.cpp
index ff15e65..91f6215 100755
--- a/camera/device/3.4/default/ExternalCameraDeviceSession.cpp
+++ b/camera/device/3.4/default/ExternalCameraDeviceSession.cpp
```

```

@@ -14,7 +14,7 @@
  \* limitations under the License.
  */
\#define LOG_TAG "ExtCamDevSsn@3.4"
-//\#define LOG_NDEBUG 0
+\#define LOG_NDEBUG 0
\#define ATRACE_TAG ATRACE_TAG_CAMERA
\#include <log/log.h>
diff --git a/camera/device/3.4/default/ExternalCameraUtils.cpp
b/camera/device/3.4/default/ExternalCameraUtils.cpp
old mode 100644
new mode 100755
index e25deff..6976185
--- a/camera/device/3.4/default/ExternalCameraUtils.cpp
+++ b/camera/device/3.4/default/ExternalCameraUtils.cpp
@@ -14,7 +14,7 @@
  \* limitations under the License.
  */
\#define LOG_TAG "ExtCamUtils@3.4"
-//\#define LOG_NDEBUG 0
+\#define LOG_NDEBUG 0
\#include <log/log.h>
\#include <cmath>

```

如何配置USB camera为前后置？

Platform/hardware/interfaces/camera/device/3.4/default 目录：

示例根据查询uvc差异信息修改前后置：

```

diff --git a/camera/device/3.4/default/ExternalCameraDevice.cpp
b/camera/device/3.4/default/ExternalCameraDevice.cpp
old mode 100644
new mode 100755
index 6dfe9d7..e7a87f1
--- a/camera/device/3.4/default/ExternalCameraDevice.cpp
+++ b/camera/device/3.4/default/ExternalCameraDevice.cpp
@@ -82,8 +82,28 @@ Return<void> ExternalCameraDevice::getCameraCharacteristics(
    _hidl_cb(Status::INTERNAL_ERROR, hidlChars);
    return Void();
}

-
+ ALOGD("%s(%d): open camera %s. !", __FUNCTION__, __LINE__, mCameraId.c_str());
+ const camera_metadata_t* rawMetadata = mCameraCharacteristics.getAndLock();
+
+ camera_metadata_entry facing =
+     mCameraCharacteristics.find(ANDROID_LENS_FACING);
+ if (facing.count == 1) {
+     switch (facing.data.u8[0]) {
+     case ANDROID_LENS_FACING_BACK:
+         // info->facing = hardware::CAMERA_FACING_BACK;
+         ALOGD("%s(%d) enter, Facing BACK.\n", __FUNCTION__, __LINE__);
+         break;
+     case ANDROID_LENS_FACING_EXTERNAL:
+         // Map external to front for legacy API
+     case ANDROID_LENS_FACING_FRONT:
+         //info->facing = hardware::CAMERA_FACING_FRONT;

```

```

+         ALOGD("%s(%d) enter, Facing Front.\n", __FUNCTION__, __LINE__);
+         break;
+     }
+ } else {
+     ALOGE("%s: Unable to find android.lens.facing static metadata",
+__FUNCTION__);
+ }
+
+     V3_2::implementation::convertToHidl(rawMetadata, &hidlChars);
+     _hidl_cb(Status::OK, hidlChars);
+     mCameraCharacteristics.unlock(rawMetadata);
@@ -205,7 +225,7 @@ status_t ExternalCameraDevice::initCameraCharacteristics() {
+
+     status_t ret;
-     ret = initDefaultCharsKeys(&mCameraCharacteristics);
+     ret = initDefaultCharsKeys(fd.get(), &mCameraCharacteristics);
+     if (ret != OK) {
+         ALOGE("%s: init default characteristics key failed: errorno %d",
+__FUNCTION__, ret);
+         mCameraCharacteristics.clear();
@@ -238,7 +258,7 @@ do {
+     }
+ } while (0)

-status_t ExternalCameraDevice::initDefaultCharsKeys(
+status_t ExternalCameraDevice::initDefaultCharsKeys(int fd,
+    ::android::hardware::camera::common::V1_0::helper::CameraMetadata*
metadata) {
+     const uint8_t hardware_level =
+     ANDROID_INFO_SUPPORTED_HARDWARE_LEVEL_EXTERNAL;
+     UPDATE(ANDROID_INFO_SUPPORTED_HARDWARE_LEVEL, &hardware_level, 1);
@@ -309,8 +329,27 @@ status_t ExternalCameraDevice::initDefaultCharsKeys(
+     UPDATE(ANDROID_LENS_INFO_AVAILABLE_OPTICAL_STABILIZATION,
+         &opticalStabilizationMode, 1);

-     const uint8_t facing = ANDROID_LENS_FACING_BACK;
-     UPDATE(ANDROID_LENS_FACING, &facing, 1);
+     struct v4l2_capability capability;
+     if (ioctl(fd, VIDIOC_QUERYCAP, &capability) < 0) {
+         ALOGE("Video device(%d): query capability not supported.\n", fd);
+     } else {
+         ALOGD("%s(%d): capability.driver is %s.\n", __FUNCTION__, __LINE__,
+             capability.driver);
+         ALOGD("%s(%d): capability.bus_info is %s.\n", __FUNCTION__, __LINE__,
+             capability.bus_info);
+         ALOGD("%s(%d): capability.card is %s.\n", __FUNCTION__, __LINE__,
+             capability.card);
+         if(strcmp((char*)&capability.card[0], "aoni Webcam") == 0) {
+             ALOGD("%s(%d): setting facing Back!\n", __FUNCTION__, __LINE__);
+             const uint8_t facing = ANDROID_LENS_FACING_BACK;
+             UPDATE(ANDROID_LENS_FACING, &facing, 1);
+         } else {
+             ALOGD("%s(%d): setting facing Front!\n", __FUNCTION__, __LINE__);
+             const uint8_t facing = ANDROID_LENS_FACING_FRONT;
+             UPDATE(ANDROID_LENS_FACING, &facing, 1);
+         }
+     }
+ }

```

```

// android.noiseReduction
const uint8_t noiseReductionMode = ANDROID_NOISE_REDUCTION_MODE_OFF;
diff --git
a/camera/device/3.4/default/include/ext_device_v3_4_impl/ExternalCameraDevice_3_4
.h
b/camera/device/3.4/default/include/ext_device_v3_4_impl/ExternalCameraDevice_3_4
.h
old mode 100644
new mode 100755
index a52f0e4..ded1228
---
a/camera/device/3.4/default/include/ext_device_v3_4_impl/ExternalCameraDevice_3_4
.h
+++
b/camera/device/3.4/default/include/ext_device_v3_4_impl/ExternalCameraDevice_3_4
.h
@@ -83,7 +83,8 @@ protected:

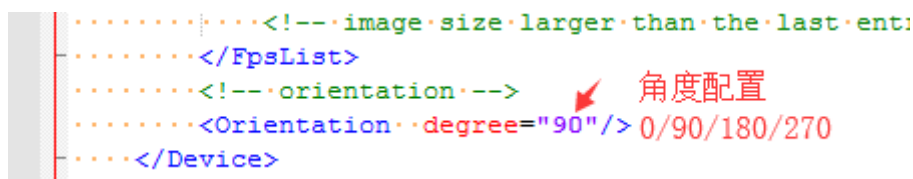
    status_t initCameraCharacteristics();
    // Init non-device dependent keys
-    status_t
initDefaultCharsKeys(::android::hardware::camera::common::V1_0::helper::CameraMet
adata*);
+    status_t initDefaultCharsKeys(int fd,
+    ::android::hardware::camera::common::V1_0::helper::CameraMetadata*);
    // Init camera control chars keys. Caller still owns fd
    status_t initCameraControlsCharsKeys(int fd,
        ::android::hardware::camera::common::V1_0::helper::CameraMetadata*);

```

RK3566-11.0-如何旋转usb camera角度

11.0之后usb camera角度配置放在了external_camera_config.xml文件中的Orientation字段：

示例如图：



The image shows a snippet of the external_camera_config.xml file. The relevant lines are:

 <!-- image size larger than the last ent:

 </FpsList>

 <!-- orientation -->

 <Orientation degree="90"/>

 </Device>

 A red arrow points to the "90" value in the Orientation tag, with the text "角度配置" (Angle Configuration) next to it.

external_camera_config.xml在板子上的位置：/vendor/etc/external_camera_config.xml

在SDK上的位置：sdk/device/rockchip/common/external_camera_config.xml

如何查看usb camera的格式及分辨率

使用命令：

```
grep -H '' /sys/class/video4linux/video*/name
```

看下哪个是usb 摄像头的video结点，然后使用如下命令看下usb 摄像头对应支持的格式信息

```
v4l2-ctl -d /dev/video* --list-formats-ext
```

Android 14.0重启usb camerahal provider命令

```
kill -9 `pidof android.hardware.camera.provider-V1-external-service-rk`
```

Android 14.0如何抓取iep deinterlaced前后数据

```
mkdir /data/camera/  
chmod 777 /data/camera/
```

使能数据抓取

```
setprop vendor.usbcamerahal.dil.dumpenable true
```

调整开始抓取数据帧，x~x+15，例如：1~15

```
setprop vendor.usbcamerahal.dil.dumpstart 0
```

使能或者关闭使用iep做deinterlace

```
setprop vendor.camera.useiep true/false
```

PDAF功能调试相关

先确认CMOS sensor本身是否支持PD像素使用mipi virtual channel输出；

查看示例说明：

1、查看OV50C40的Datasheet支持PD像素单独输出；

3.5 PDAF control

The OV50C40 supports PDAF data output through MIPI by programmable data type mode or virtual channel mode. All PDAF data or partial PDAF data are selected to output. In mirror mode, PDAF data can also be mirrored.

3.5.1 PDAF data output: option 1 (programmable data type)

Option 1 uses programmable data type for PDAF RAW data. Set registers 0x3684[0] to 1, 0x3684[2] to 0, and 0x3684[1] to 1 to enable PDAF output timing; set register 0x480E[3] to 1 to enable MIPI data type mode. Register 0x4815[5:0] is the data type of PD data. PDAF data is output during image data horizontal blanking. Maximum PDAF pixel number in one package is configured by registers {0x4640, 0x4641}*8.

3.5.2 PDAF data output: option 2 (MIPI virtual channel)

Option 2 is the MIPI virtual channel for PDAF RAW data. Set registers 0x3684[0] to 1, 0x3684[2] to 1, and 0x3684[1] to 1 to enable PDAF output timing; set register 0x480E[2] to 1 to enable MIPI virtual channel mode. A normal image outputs through MIPI virtual channel 0. PDAF data outputs through MIPI virtual channel 1. PDAF data is output during image data horizontal blanking. Maximum PDAF pixel number in one package is configured by registers {0x4640, 0x4641}*8.

2、S5KJN1的PDAF文档查看

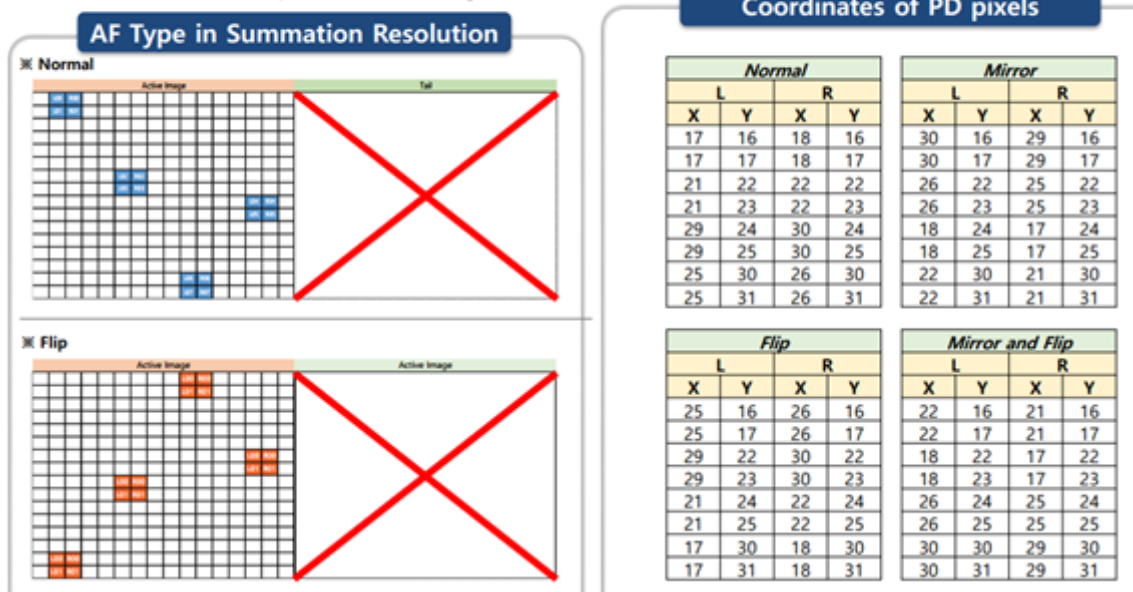
12.5Mp即4080x3072@30FPS 的分辨率支持tail mode即单独virtual channel输出；

AF1* : A1(Left Pixel), AF2** : A2(Right Pixel)

^{**} Be able to modify by parameter rule of AP's if needed

PDAF Tail Mode Type & Coordinates (Full 8160x6144)

- Full mode - DOES NOT SUPPORT TAIL MODE
- The coordinates of PDAF pixels in basic unit (Figure2).



3、OV13855的Datasheet写着支持pdaF，但pd像素是嵌入在正常像素中的；

这种RK平台不支持pdaF；

- 33.15° CRA
- support for PDAF
- 13MP at 30 fps

支持pdaF的dtsi所需配置

在对应sensor下面参考如下配置添加pdaF像素支持

```
--- a/arch/arm64/boot/dts/rockchip/rk3588s-tablet.dtsi
+++ b/arch/arm64/boot/dts/rockchip/rk3588s-tablet.dtsi
@@ -547,6 +547,7 @@
        rockchip,camera-module-facing = "back";
        rockchip,camera-module-name = "HZGA06";
        rockchip,camera-module-lens-name = "ZE0082C1";
+       rockchip,spd-id = <1>;
        eeprom-ctrl = <&otp_eeprom>;
        lens-focus = <&aw8601>;
        port {
```

底层确认pd像素输出

dtsi中如果配置rockchip,spd-id = <1>; 说明是配置成virtual channel 1输出pd像素

例如：ov50c40抓取pd像素命令如下：

```
v4l2-ctl --verbose -d /dev/video1 --set-fmt-
video=width=1016,height=760,pixelformat='SPD6' --stream-mmap=4 --set-
selection=target=crop,flags=0,top=0,left=0,width=1016,height=760
```

输入命令后示例输出：

```
rk3588s_s:/data/pdaf #
ap=4 --set-selection=target=crop,flags=0,top=0,left=0,width=1016,height=760 <
VIDIOC_QUERYCAP: ok
VIDIOC_G_FMT: ok
VIDIOC_S_FMT: ok
Format Video Capture Multiplanar:
    Width/Height      : 1016/760
    Pixel Format       : 'SPD6'
    Field             : None
    Number of planes   : 1
    Flags             :
    Colorspace        : Default
    Transfer Function  : Default
    YCbCr Encoding    : Default
    Quantization       : Default
    Plane 0           :
        Bytes per Line : 2032
        Size Image      : 1544320
VIDIOC_G_SELECTION: ok
VIDIOC_S_SELECTION: failed: Inappropriate ioctl for device
VIDIOC_REQBUFS: ok
VIDIOC_QUERYBUF: ok
VIDIOC_QUERYBUF: ok
VIDIOC_QBUF: ok
VIDIOC_QUERYBUF: ok
VIDIOC_QBUF: ok
VIDIOC_QUERYBUF: ok
VIDIOC_QBUF: ok
VIDIOC_QUERYBUF: ok
VIDIOC_QBUF: ok
VIDIOC_STREAMON: ok
idx: 0 seq:      0 bytesused: 1544320 ts: 6037.805572
idx: 1 seq:      1 bytesused: 1544320 ts: 6037.838898 delta: 33.326 ms
idx: 2 seq:      2 bytesused: 1544320 ts: 6037.872124 delta: 33.226 ms
idx: 3 seq:      3 bytesused: 1544320 ts: 6037.905555 delta: 33.431 ms
idx: 0 seq:      4 bytesused: 1544320 ts: 6037.938891 delta: 33.336 ms fps: 30.00
idx: 1 seq:      5 bytesused: 1544320 ts: 6037.972205 delta: 33.314 ms fps: 30.01
idx: 2 seq:      6 bytesused: 1544320 ts: 6038.005532 delta: 33.327 ms fps: 30.01
idx: 3 seq:      7 bytesused: 1544320 ts: 6038.038859 delta: 33.327 ms fps: 30.01
idx: 0 seq:      8 bytesused: 1544320 ts: 6038.072185 delta: 33.326 ms fps: 30.01
idx: 1 seq:      9 bytesused: 1544320 ts: 6038.105514 delta: 33.329 ms fps: 30.01
idx: 2 seq:     10 bytesused: 1544320 ts: 6038.138852 delta: 33.338 ms fps: 30.00
idx: 3 seq:     11 bytesused: 1544320 ts: 6038.172169 delta: 33.317 ms fps: 30.01
idx: 0 seq:     12 bytesused: 1544320 ts: 6038.205398 delta: 33.229 ms fps: 30.01
```

说明pd像素输出正常；

pdaf上层配置及调试支持

1、首先查看iq文件中是否将pdaf支持打开

```
{
  "pdaf": {
    "enable": 1,
    "pdVsCdDebug": 0
  }
}
```

2、上层如何获取并保存pd像素

Sdk/external/camera_engine_rkaiq目录:

添加如下修改

```
diff --git a/aiq_core/RkAiqResourceTranslator.cpp
b/aiq_core/RkAiqResourceTranslator.cpp

old mode 100644

new mode 100755

index cda8745..d96d178

--- a/aiq_core/RkAiqResourceTranslator.cpp
+++ b/aiq_core/RkAiqResourceTranslator.cpp

@@ -16,7 +16,7 @@

 \* limitations under the License.

 \*

 \*/

-// #define PDAF_RAW_DUMP

+ #define PDAF_RAW_DUMP

\#include "isp20/Isp20Evts.h"

\#include "isp20/Isp20StatsBuffer.h"
```

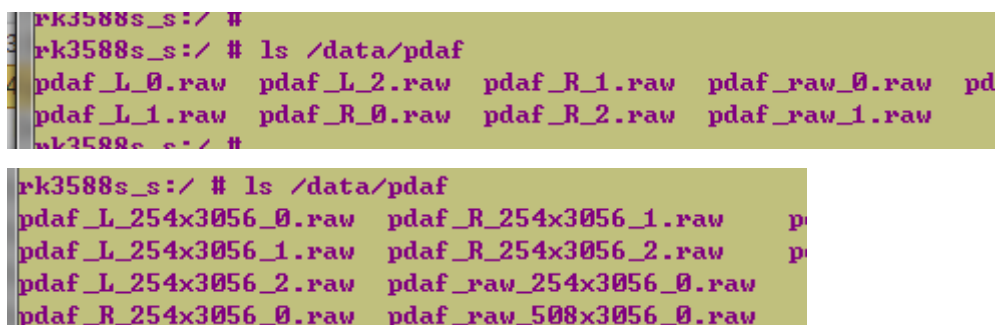
然后:

```
mkdir -p /data/pdaf/
chmod 777 /data/pdaf/
```

修改后打开摄像头, 如果pdaf运行正常

rk3588s_s:/ # ls /data/pdaf

会保存如下的pd完整像素raw图, pd L/R的像素raw图



```
rk3588s_s:/ #
rk3588s_s:/ # ls /data/pdaf
pdaf_L_0.raw  pdaf_L_2.raw  pdaf_R_1.raw  pdaf_raw_0.raw  pd
pdaf_L_1.raw  pdaf_R_0.raw  pdaf_R_2.raw  pdaf_raw_1.raw
rk3588s_s:/ #

rk3588s_s:/ # ls /data/pdaf
pdaf_L_254x3056_0.raw  pdaf_R_254x3056_1.raw  p
pdaf_L_254x3056_1.raw  pdaf_R_254x3056_2.raw  p
pdaf_L_254x3056_2.raw  pdaf_raw_254x3056_0.raw
pdaf_R_254x3056_0.raw  pdaf_raw_508x3056_0.raw
```

从板子上pull 出来后, 使用raw图查看工具, 可以看到图像;

或者: 参考: Rockchip_Tuning_Guide_ISP30_CN_v1.2.1.pdf文件

将iq文件中pdDumpDebug使能为1:

```
→ "pdaf" : {  
→   "enable" : 1,  
→   "pdVsCdDebug" : 0,  
→   "pdDumpDebug" : 0,  
→   "pdDataBit" : 10  
→ }
```

打开这个开关打开, /data/pdafdebug/要手动创建一下

4.11.2.7 pdaf

【描述】

pdaf算法参数设置

【成员】

enable为pdaf算法开关;

pdVsCdDebug为全扫开关, 打开此开关后, 会遍历每个马达位置, 同时打印fv值和相位差值;

pdDumpDebug为debug开关, 打开此开关后, 会dump pdaf调试数据到/data/pdafdebug/下;

pdDataBit为pd像素bit数;

mkdir -p /data/pdafdebug

chmod 777 /data/pdafdebug

打开摄像头跑一下, /data/pdafdebug里面就会保存pdaf数据

上层pdaf像素未抓取到问题排查步骤

1、先确认aiq中获取pdaf信息正常:

hwi/isp20/CamHwIsp20.cpp中CamHwIsp20::get_sensor_pdafinfo

```
CamHwIsp20.cpp x LogHelperAndroid.cpp LogHelperAndroid.h PlatformData.cpp  
2154:  
2155: XCamReturn  
2156: CamHwIsp20::get_sensor_pdafinfo(rk_sensor_full_info_t*sensor_info,  
2157: .....rk_sensor_pdaf_info_t*pdaf_info){  
2158: ....XCamReturn.ret=XCAM_RETURN_NO_ERROR;  
2159: ....struct rkmodule_channel_info channel;  
2160: ....memset(&channel, 0, sizeof(struct rkmodule_channel_info));  
2161:  
2162: ....V4l2SubDevice vdev(sensor_info->device_name.c_str());  
2163: ....ret=vdev.open();  
2164: ....if(ret!=XCAM_RETURN_NO_ERROR){  
2165: .....LOGE_CAMHW_SUBM(ISP20HW_SUBM, "failed to open dev (%s)", sensor_info->dev_name.c_str());  
2166: .....return XCAM_RETURN_ERROR_FAILED;  
2167: ....}  
2168:  
2169: ....pdaf_info->pdaf_support=false;  
2170: ....for(int i=0; i<4; i++){  
2171: .....channel.index=i;  
2172: .....LOGI_CAMHW_SUBM(ISP20HW_SUBM, "channel.index before: 0x%x", channel.index);  
2173:  
2174: .....if(vdev.io_control(RKMODULE_GET_CHANNEL_INFO, &channel)==0){  
2175: .....LOGI_CAMHW_SUBM(ISP20HW_SUBM, "channel.bus_fmt: 0x%x, channel.index: %d", channel.bus_fmt, channel.index);  
2176: .....channel.bus_fmt, channel.index, channel.width, channel.height);  
2177:  
2178: .....if(channel.bus_fmt==MEDIA_BUS_FMT_SPD_2X8){  
2179: .....pdaf_info->pdaf_support=true;  
2180: .....pdaf_info->pdaf_vc=i;  
2181: .....pdaf_info->pdaf_code=channel.bus_fmt;
```

添加示例打印:

```
diff --git a/hwi/isp20/CamHwIsp20.cpp b/hwi/isp20/CamHwIsp20.cpp
old mode 100644
new mode 100755
index 57bd81c..7770e2d
--- a/hwi/isp20/CamHwIsp20.cpp
+++ b/hwi/isp20/CamHwIsp20.cpp
@@ -2169,7 +2169,12 @@ CamHwIsp20::get_sensor_pdatainfo(rk_sensor_full_info_t
 *sensor_info,
     pdata_info->pdata_support = false;
     for (int i = 0; i < 4; i++) {
         channel.index = i;
+        LOGI_CAMHW_SUBM(ISP20HW_SUBM, "channel.index before: 0x%x ",
+channel.index);
+
         if (vdev.io_control(RKMODULE_GET_CHANNEL_INFO, &channel) == 0) {
+            LOGI_CAMHW_SUBM(ISP20HW_SUBM, "channel.bus_fmt 0x%x, channel.index:
0x%x channel.widthxheight: %dx%d",
+            channel.bus_fmt, channel.index, channel.width, channel.height);
+
             if (channel.bus_fmt == MEDIA_BUS_FMT_SPD_2X8) {
                 pdata_info->pdata_support = true;
                 pdata_info->pdata_vc = i;

```

2、跟踪RKMODULE_GET_CHANNEL_INFO这个ioctl调用到sensor驱动位置：

以s5kjin1为例调用位置如下：自行加log跟踪查看哪里调用异常，然后修改

```
1259: static int s5kjin1_get_channel_info(struct s5kjin1 *s5kjin1, struct rkmodule_channel_info *ch_
1260: {
1261:     const struct s5kjin1_mode *mode = s5kjin1->cur_mode;
1262:
1263:     if (ch_info->index < PAD0 || ch_info->index >= PAD_MAX)
1264:         return -EINVAL;
1265:
1266:     if (mode->spd)
1267:         dev_info(&s5kjin1->client->dev, "ch_info->index:%d, s5kjin1->spd_id: %d\n",
1268:             ch_info->index, s5kjin1->spd_id);
1269:
1270:     if (ch_info->index == s5kjin1->spd_id && mode->spd) {
1271:         ch_info->vc = V4L2_MBUS_CSI2_CHANNEL_1;
1272:         ch_info->width = mode->spd->width;
1273:         ch_info->height = mode->spd->height;
1274:         ch_info->bus_fmt = mode->spd->bus_fmt;
1275:         ch_info->data_type = mode->spd->data_type;
1276:         ch_info->data_bit = mode->spd->data_bit;
1277:     } else {
1278:         ch_info->vc = s5kjin1->cur_mode->vc[ch_info->index];
1279:         ch_info->width = s5kjin1->cur_mode->width;
1280:         ch_info->height = s5kjin1->cur_mode->height;
1281:         ch_info->bus_fmt = s5kjin1->cur_mode->bus_fmt;
1282:     }
1283:     return 0;
1284: } // end s5kjin1_get_channel_info
```

RK3576支持OV13855

OV13855 Camera Application Notes R1.02_Rockchip.pdf中介绍了OV13855的PD像素排列；

OV13855 is a type 3 PDAF sensor。PD像素排列如图：

6.1.1 PD Pattern of Full Resolution

PD pattern is shown as below.

Figure 6 - 1 PD Pattern of Full Resolution with mirror normal/flip normal

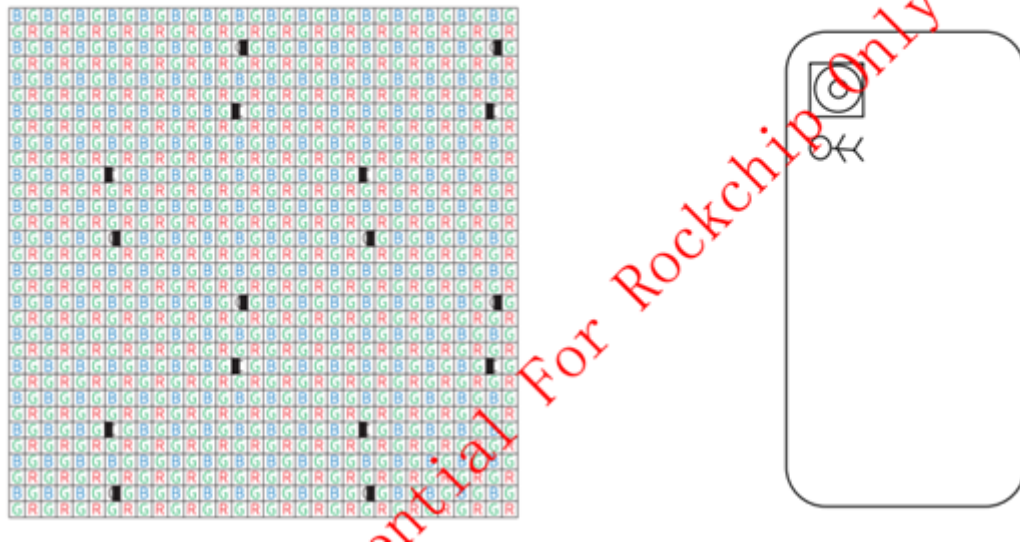
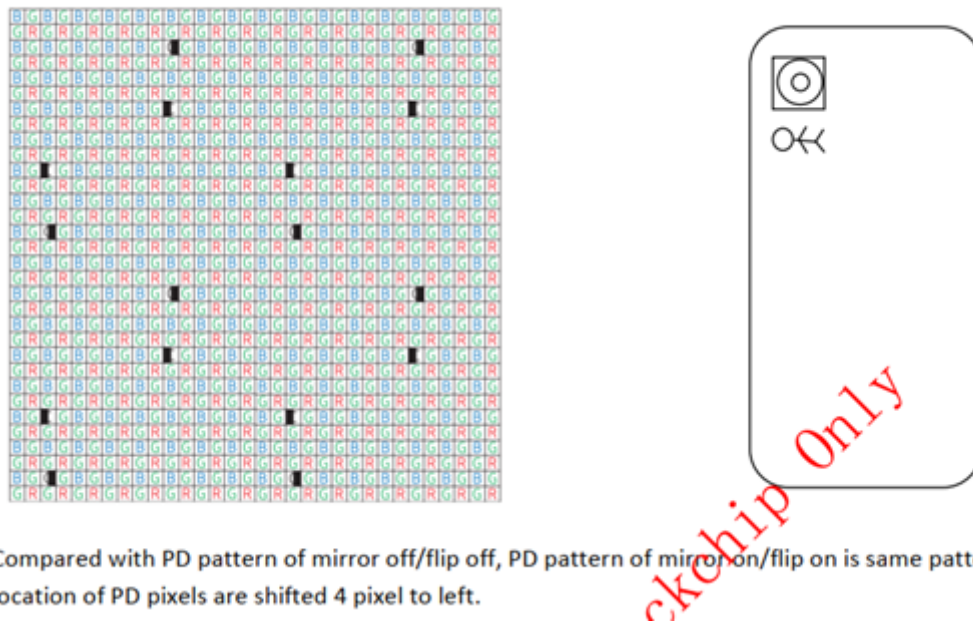


Figure 6-2A PD Pattern of Full Resolution with mirror on/flip on



iq中的dpcc和af配好以后,

也是RkAiqResourceTranslator::translatePdafStats

dump pdaf数据

```
},  
    "Dpcc_pdaf": {  
        "en": 0,  
        "point_en": [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],  
        "offsetx": 0,  
        "offsety": 0,  
        "wrapx": 0,  
        "wrapy": 0,  
        "wrapx_num": 0,  
        "wrapy_num": 0,  
        "point_x": [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],  
        "point_y": [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],  
        "forward_med": 0  
    },  
}
```

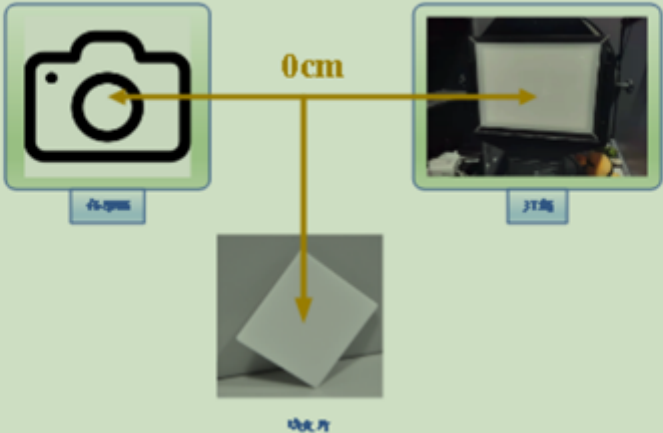
```

"pdResoInf": [{
    "pdType": "CalibDbv2_PDAF_SENSOR_TYPE3",
    "imageWidth": 4608,
    "imageHeight": 3456,
    "pdOutWidth": 512,
    "pdOutHeight": 384,
    "pdCropX": 0,
    "pdCropY": 0,
    "pdBaseWidth": 512,
    "pdBaseHeight": 384
}],
},
"pda": {
    "enable": 0,
    "pdVsCdDebug": 0,
    "pdDumpDebug": 0
}

```

PDAF调试拍照规范:

2、拍摄方式



1. 将模组放置在均匀面光源前，模组端面与光源面保持平行（eg. 实验时采用的积分球，面板亮度数值调整为6.6），均光片置于镜头前端，使得面光源充满画面；
2. sensor的gain-对于 metal-shield和2x10CL类型的sensor的again和dgain设置为1x, dual-pd类型的sensor, again设置为2x, dgain设置为1x, 曝光时间设置为10的倍数；
3. 关闭sensor中的mirro/flip/OB等设置
4. 全尺寸显示时画面中心1/32w*1/32H的ROI区域的TOP 10%G通道的平均亮度值为 800LSB(raw10)或200LSB (raw8); dual pd G通道的均值为 700LSB(raw10)或175LSB (raw8);
5. 打开camera预热3分钟，将马达移动到中置位置连续拍摄3张raw图，首先检测坏点个数是否超过允许的范围，若坏点个数超过允许范围，则需要降低画面亮度重新拍摄；若坏点检测通过，再得到平均图像数据；

具体从redmine或ISP窗口获取规范文档: Rockchip_Development_Guide_PDAF.pdf

明显带PD像素的RAW图抓取示例图:

Rockchip OTP常见问题

CMOS sensor添加OTP支持步骤

1、dtsi添加eeprom-ctrl关键字；ov08d10模组添加otp支持示例如下：

```
ov08d10: ov08d10@10 {
    . . . . .
    eeprom-ctrl = <&otp_eeprom_b>;
    . . . . .
};
otp_eeprom_b: otp_eeprom_b@53 {
    compatible = "rk,otp_eeprom";
    status = "okay";
    reg = <0x53>;
};
```

0x53为示例otp的i2c地址，具体根据模组自行修改；

2、查看CMOS sensor驱动是否有添加otp驱动支持

包含OTP驱动支持示例驱动：OV50C40.c

部分示例代码：

```
eeprom_ctrl_node = of_parse_phandle(node, "eeprom-ctrl", 0);
if (eeprom_ctrl_node) {
    eeprom_ctrl_client =
        of_find_i2c_device_by_node(eeprom_ctrl_node);
    of_node_put(eeprom_ctrl_node);
    if (IS_ERR_OR_NULL(eeprom_ctrl_client)) {
        dev_err(dev, "can not get node\n");
    }
}
```

```

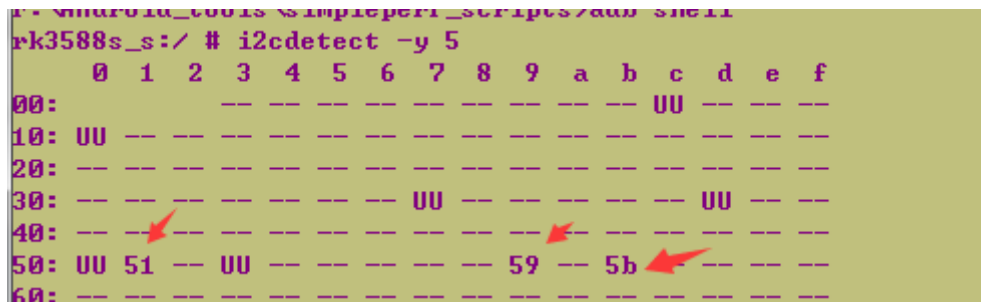
        goto continue_probe;
    }
    eeprom_ctrl = i2c_get_clientdata(eeprom_ctrl_client);
    if (IS_ERR_OR_NULL(eeprom_ctrl)) {
        dev_err(dev, "can not get eeprom i2c client\n");
    } else {
        otp_ptr = devm_kzalloc(dev, sizeof(*otp_ptr), GFP_KERNEL);
        if (!otp_ptr)
            return -ENOMEM;
        ret = v4l2_subdev_call(eeprom_ctrl,
                                core, ioctl, 0, otp_ptr);
        if (!ret) {
            ov50c40->otp = otp_ptr;
        } else {
            ov50c40->otp = NULL;
            devm_kfree(dev, otp_ptr);
        }
    }
}

```

如何查看模组是否有正确按照ROCKCHIP规范烧录OTP信息

将摄像头打开，然后使用i2cdetec -y i2cnum 命令查看otp对应的i2c地址；

例如：i2cdetec -y 5检测到的地址很多；



```

rk3588s_s:/ # i2cdetec -y 5
  0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10: UU  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20: --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30: --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40: --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50: UU 51 -- UU -- -- -- -- 59 -- 5b -- -- -- -- --
60: --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --

```

然后使用i2ctransfer工具，查看这些地址前8个值；

读取ASCII码值，和”ROCKCHIP”对应的ASCII码值对比看是否一致，

如果一致，说明是按照rockchip烧录规范烧写的otp信息；

如果不一致，说明这个不是otp或者不是按照rk规范烧录的otp信息；

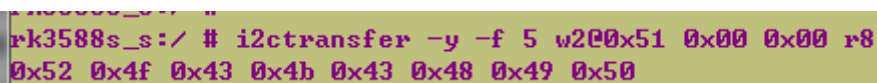
换地址查看，直到找到对应otp地址即可；

1) 读取的是”ROCKCHIP”对应的ASCII码值示例：

```

rk3588s_s:/ # i2ctransfer -y -f 5 w2@0x51 0x00 0x00 r8
0x52 0x4f 0x43 0x4b 0x43 0x48 0x49 0x50

```



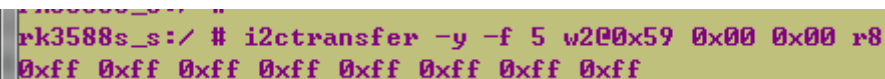
```

rk3588s_s:/ # i2ctransfer -y -f 5 w2@0x51 0x00 0x00 r8
0x52 0x4f 0x43 0x4b 0x43 0x48 0x49 0x50

```

2) 读取不一致示例：

例如：i2ctransfer -y -f 5 w2@0x51 0x00 0x00 r8



```

rk3588s_s:/ # i2ctransfer -y -f 5 w2@0x51 0x00 0x00 r8
0xff 0xff 0xff 0xff 0xff 0xff 0xff 0xff

```

RK3588-12.0-多路帧同步功能调试

1、参考Rockchip_Driver_Guide_VI_CN_v1.1.1.pdf文档中

多摄同步机制 章节，将驱动适配成同步模式，同时硬件按照文档要求进行同步互联；

2、上层AIQ同步调试；参考

SDK\RKDocs\common\camera\HAL3\Rockchip_Development_Guide_ISP\ISP30\CN\Rockchip_Development_Guide_ISP30_CN_v1.2.3.pdf文档中Camera组相关api调试；

概述

```
rk_aiq_uapi2_camgroup_create  
  
rk_aiq_uapi2_camgroup_prepare  
  
rk_aiq_uapi2_camgroup_start  
  
rk_aiq_uapi2_camgroup_stop  
  
rk_aiq_uapi2_camgroup_destroy  
  
rk_aiq_uapi2_camgroup_getOverlapMap  
  
rk_aiq_uapi2_camgroup_getOverlapMap  
  
rk_aiq_uapi2_camgroup_getAiqCtxBySnsNm  
  
rk_aiq_uapi2_camgroup_getCamInfos  
  
rk_aiq_uapi2_camgroup_bind  
  
rk_aiq_uapi2_camgroup_unbind
```

拍照问题

如何区分使用的是硬编还是软编

将camerahal的log打开，然后

```
logcat | grep ImgHWEncoder
```

能抓到说明是硬编；反之是软编

Camera MIPI报错问题排查

1、sensor驱动link_req

2、检测硬件是否有虚焊，排线是否松动或解除不良；

3、排查主板mipi走线是否有干扰信号，比如mipi走线旁边是否有高频信号，中间是否隔地；走线上方是否有高频信号，如wifi等，可先用排除法排除下；除了板上走线，排线也可能有影响，可以包屏蔽纸测试；

4、尝试增加sensor hts确认是否有改善；

5、测量硬件信号，测量点尽量靠近主控端，确认信号是否符合mipi标准协议；

6、排查主板mipi走线是否偏长，走线偏长可能削弱信号导致数据识别错误；

- 7、如果是时序不满足要改sensor驱动，修改mipi输出时序；
- 8、如果是信号削弱，可以修改sensor的寄存器，增加mipi输出驱动强度；
- 9、如果是信号干扰，要做好屏蔽措施。

SDK自带驱动更新SDK后打不开问题排查步骤

- 1、v4l2-ctl抓数据流，查看底层是否正常；
- 2、camera3_profiles.xml中sensorType改成soc类型，查看HAL到底层的通路是否正常？
- 3、sensorType再改成RAW类型，排查IQ文件是否匹配？

示例IQ文件定义规则

RK3288/ RK3326/PX30/RK3368/RK3399平台

<sensor 名称>_<模组名称>_<lens 名称>.xml

RK3566/RK3568/RK3588/RK3562/RK3576平台

<sensor 名称>_<模组名称>_<lens 名称>.json

其中：“模组名称”和“lens名称”分别对应dts中如下关键字

```
rockchip,camera-module-name = "default";
```

及

```
rockchip,camera-module-lens-name = "default";
```

示例dts:

```
imx415: imx415@1a {
    .....
    rockchip,camera-module-name = "CMK-OT2022-PX1";
    rockchip,camera-module-lens-name = "IR0147-50IRC-8M-F20";
    .....
};
```

则RK3288/ RK3326/PX30/RK3368/RK3399平台的IQ文件名称为:

imx415_CMK-OT2022-PX1_IR0147-50IRC-8M-F20.xml

RK3566/RK3568/RK3588/RK3562/RK3576平台的IQ文件名称为:

imx415_CMK-OT2022-PX1_IR0147-50IRC-8M-F20.json

IQ场景参数切换

现在不同分辨率用的是同一套iq参数？

期望用不同iq参数，是需要自己切换的场景；

整个iq模块，只有lsc模块跟分辨率有关，其他都无关

所以如下想要区分降噪，锐化参数，就需要切换场景；

添加支持场景切换示例修改

需要单独添加补丁：

0001-camera-rkisp2-add-scene-switch-support.patch

0001-hal_interface2.0-add-scene-switch-support.patch

如需要找FAE搜索redmine获取：

Camera第三方应用，方向兼容性问题

相机360/小红书，调用前置/后置摄像头图像180°倒置问题排查

问题现象

步骤：

- 测试机已开启自动旋转功能
- 自行安装 小红书_8.43.1.5 或 相机360_9.9.37 最新版,进行相机测试
- 小红书（发布-拍摄-调用相机-切换到前置摄像头）
- 相机360（拍摄-切换到前置摄像头）

现象：预览窗口和照片都是180°倒置，后置摄像头正常，系统相机预览正常。

排查步骤

1. 首先确认camera3_profiles.xml中摄像头方向配置是否满足如下规则

竖屏机器：前置摄像头配置270°，后置摄像头配置90°；

横屏机器：前置摄像头配置180°，后置摄像头配置0°；

关键字：<sensor.orientation value="0"/>

如果不符合，先修改，sync后，推回板子确认方向；

2. 修改CMOS sensor驱动，调整方向

参考CMOS sensor的Datasheet，找到对应Flip、Mirror寄存器，使用i2c命令进行调整

例如：GC05A2 FLIP/MIRROR寄存器如下为：0x0101

Table 4-1 Mirror and Flip Information

Function	Register Address	Register Value	First Pixel
Normal	0x0101[1:0]	00	Gr
Horizontal mirror	0x0101[1:0]	01	R
Vertical Flip	0x0101[1:0]	10	B
Horizontal Mirror and Vertical Flip	0x0101[1:0]	11	Gb

```
i2ctransfer -y -f 4 w2@0x37 0x01 0x01 r1 //读取原有寄存器值
i2ctransfer -y -f 4 w3@0x37 0x01 0x01 0x00 //normal
i2ctransfer -y -f 4 w3@0x37 0x01 0x01 0x01 //mirror
i2ctransfer -y -f 4 w3@0x37 0x01 0x01 0x02 //flip
i2ctransfer -y -f 4 w3@0x37 0x01 0x01 0x03 //flip & mirror
```

调整符和方向的寄存器配置后，可以根据如下 两种情况修改驱动完成：

- CMOS sensor仅输出YUV格式数据，则直接修改驱动中初始化寄存器配置调整方向；

- CMOS sensor仅输出RAW格式数据，则还可以驱动实现V4L2标准ioctl调用

V4L2_CID_HFLIP和V4L2_CID_VFLIP，从而使用IQ文件实现方向调整

示例解决方法

以输出RAW格式的GC05A2驱动调整方向为例说明

1. gc05a2添加V4L2_CID_HFLIP和V4L2_CID_VFLIP支持

参考patch:

[0001-media-i2c-gc05a2-add-set-flip-mirror-support.patch](#)

2. camera3_profiles.xml修改正确orientation

```
diff --git a/etc/camera/camera3_profiles_rk3576.xml
b/etc/camera/camera3_profiles_rk3576.xml
index 746dea9..2220141 100644
--- a/etc/camera/camera3_profiles_rk3576.xml
+++ b/etc/camera/camera3_profiles_rk3576.xml
@@ -1484,7 +1484,7 @@
         <sensor.baseGainFactor value="0,1"/> <!-- HAL may override this
value from CMC for RAW sensors -->
         <sensor.blackLevelPattern value="0,0,0,0"/>
         <sensor.maxAnalogSensitivity value="2400"/> <!-- HAL may override
this value from CMC for RAW sensors -->
-         <sensor.orientation value="90"/>
+         <sensor.orientation value="270"/>
         <sensor.profileHueSatMapDimensions value="0,0,0"/>
         <sensor.availableTestPatternModes value="OFF,COLOR_BARS"/>
         <!-- Info -->
```

3. IQ文件调整Sensor_info中FLIP配置

```
diff --git a/iqfiles/isp39/gc05a2_KYT-11210-V2_default.json
b/iqfiles/isp39/gc05a2_KYT-11210-V2_default.json
index 4ad124c3..5d7d9403 100644
--- a/iqfiles/isp39/gc05a2_KYT-11210-V2_default.json
+++ b/iqfiles/isp39/gc05a2_KYT-11210-V2_default.json
@@ -111,7 +111,7 @@
         }
         },
         "CISMinFps": 10,
-        "CISFlip": 0
+        "CISFlip": 3
     },
     "module_calib": {
         "sensor_module": {
```

qq/微信等第三方应用 摄像头方向问题

1. 系统配置为默认竖屏，像手机系统那样 第三方应用兼容性最好

/device/rockchip/rkxx/BoardConfig.mk:42:SF_PRIMARY_DISPLAY_ORIENTATION := 0 改为 90或者270

2. 系统默认竖屏后，配置camera orientation:

mipi camera 在camera3_profiles.xml里 配置orientation，前摄270，后摄90

usb camera external_camera_config.xml里 <Orientation degree="90"/>，前摄270，后摄90

按如上配置后如果图像倒180或者镜像了，mipi camera 要去改寄存器flip 或者mirror，usb camera 确认是否安装反了。

按如上配置后如果图像倒90或者270，那就是模组打样方向不对，请重新打样。

如果客户不想重新打样，可rga 裁剪旋转图像，但是会损失fov。

3. 如果客户开机动画和开机桌面是横屏的，可以打

上./Rockchip_Trouble_Shooting_CameraHAL3_CN/patches/an12_boot_rotation.patch

4. qq 视频通话gsensor 旋转有问题，打

上./Rockchip_Trouble_Shooting_CameraHAL3_CN/patches/framework_base_qq.diff

5. 腾讯会议usb camera无法打开问题，打

上./Rockchip_Trouble_Shooting_CameraHAL3_CN/patches/usb_camera_id_map_11.0_2.diff

RK356x-11.0-双摄问题汇总

RK356x-11.0-R9/R10/R11版本双摄同时打开异常所需补丁包

补丁包位置：

<https://redmine.rock-chips.com/documents/104>

RK3568-mipi-2lane+2lane双摄R9版本所需补丁包.rar

RK3568-mipi-2lane+2lane双摄R10版本所需补丁包.rar

RK3568-mipi-2lane+2lane双摄R11版本所需补丁包.rar

RK3568-mipi-2lane+2lane双摄R12版本所需补丁包.rar

Camera相关的目录：

CameraHal: sdk/hardware/rockchip/camera

aiq: sdk/external/camera_engine_rkaiq

kernel: drivers/media/platform/rockchip/isp

kernel: drivers/media/platform/rockchip/cif

kernel: include/uapi/linux/ rk-camera-module.h/ rkCIF-config.h/ rk_vcm_head.h/ rkisp1-config.h

rkisp2-config.h/ rkisp21-config.h/ rk-led-flash.h

kernel: drivers/iommu/rockchip-iommu.c include/soc/rockchip/rockchip_iommu.h drivers/media/i2c

rk356x-双摄链路组合：

mipi raw+mipi raw

链接关系1: raw sensor1->csi2_dphy1->isp_vir0

链接关系2: raw sensor2->csi2_dphy2->mipi_csi2->vicap->isp_vir1

mipi soc+mipi raw

链接关系1: soc sensor1->csi2_dphy1-> mipi_csi2->vicap

链接关系2: raw sensor2->csi2_dphy2 ->isp_vir0

dvp raw +mipi raw

链接关系1: raw sensor1->dvp->isp_vir0

链接关系2: raw sensor2->csi2_dphy0->mipi_csi2->vicap->isp_vir1

dvp soc+mipi raw

链接关系1: soc sensor1->dvp->vicap

链接关系2: raw sensor2->csi2_dphy0->mipi_csi2->vicap->isp_vir1

dvp raw+mipi soc

链接关系1: raw sensor1->dvp->isp_vir0

链接关系2: soc sensor2->csi2_dphy0->mipi_csi2->vicap

dvp soc+mipi soc

链接关系1: soc sensor1->dvp->isp_vir0

链接关系2: soc sensor2->csi2_dphy0->mipi_csi2->vicap

mipi soc+mipi soc

链接关系1: soc sensor1->csi2_dphy1->isp_vir0

链接关系2: soc sensor2->csi2_dphy2->mipi_csi2->vicap->isp_vir1

rk356x-11.0-双目摄像头调试底层常见问题

双目摄像头示例链路连接

链接关系1: sensor1->csi2_dphy1->isp_vir0

链接关系2: sensor2->csi2_dphy2->mipi_csi2->vicap->isp_vir1

如何确定链路正确

使用 `grep -H "/sys/class/video4linux/video*/name"`

这个命令看下哪个video结点是stream_cif_mipi_id0和rkisp-vir0的mainpath

一般是video0和video5结点;

```
130@rk3568_r:/ # grep -H "/sys/class/video4linux/video*/name"
/sys/class/video4linux/video0/name:stream_cif_mipi_id0
/sys/class/video4linux/video1/name:stream_cif_mipi_id1
/sys/class/video4linux/video10/name:rkisp_rawrd0_m
/sys/class/video4linux/video11/name:rkisp_rawrd2_s
/sys/class/video4linux/video12/name:rkisp-statistics
/sys/class/video4linux/video3/name:stream_cif_mipi_id3
/sys/class/video4linux/video4/name:rkcif-mipi-luma
/sys/class/video4linux/video5/name:rkisp_mainpath
/sys/class/video4linux/video6/name:rkisp_selfpath
/sys/class/video4linux/video7/name:rkisp_rawwr0
/sys/class/video4linux/video8/name:rkisp_rawwr2
/sys/class/video4linux/video9/name:rkisp_rawwr3
rk3568_r:/ #
```

使用如下两个命令同时抓取两个结点的数据流, 如果正常, 说明底层通路OK

```
v4l2-ctl --verbose -d /dev/video0 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --set-selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080
```

```

VIDIOC_QBUF: ok
VIDIOC_STREAMON: ok
idx: 0 seq:      0 bytesused: 3110400 ts: 316.198903
idx: 1 seq:      1 bytesused: 3110400 ts: 316.232269 delta: 33.366 ms
idx: 2 seq:      2 bytesused: 3110400 ts: 316.265570 delta: 33.301 ms
idx: 3 seq:      3 bytesused: 3110400 ts: 316.298910 delta: 33.340 ms
idx: 0 seq:      4 bytesused: 3110400 ts: 316.332237 delta: 33.327 ms fps: 30.00
idx: 1 seq:      5 bytesused: 3110400 ts: 316.365578 delta: 33.341 ms fps: 30.00
idx: 2 seq:      6 bytesused: 3110400 ts: 316.398911 delta: 33.333 ms fps: 30.00
idx: 3 seq:      7 bytesused: 3110400 ts: 316.432243 delta: 33.332 ms fps: 30.00

```

v4l2-ctl --verbose -d /dev/video5 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --set-selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080

```

VIDIOC_QBUF: ok
VIDIOC_STREAMON: ok
idx: 0 seq:      0 bytesused: 3110400 ts: 322.022796
idx: 1 seq:      1 bytesused: 3110400 ts: 322.055822 delta: 33.026 ms
idx: 2 seq:      2 bytesused: 3110400 ts: 322.089435 delta: 33.613 ms
idx: 3 seq:      3 bytesused: 3110400 ts: 322.122786 delta: 33.351 ms
idx: 0 seq:      4 bytesused: 3110400 ts: 322.156117 delta: 33.331 ms fps: 30.00
idx: 1 seq:      5 bytesused: 3110400 ts: 322.189384 delta: 33.267 ms fps: 30.01
idx: 2 seq:      6 bytesused: 3110400 ts: 322.222782 delta: 33.398 ms fps: 30.00
idx: 3 seq:      7 bytesused: 3110400 ts: 322.255853 delta: 33.071 ms fps: 30.04

```

如果异常，说明底层通路就异常了，需要先按照[v4l2命令获取数据流失败问题常见原因及调试方法](#)说明进行排查；

rk356x-11.0-双目摄像头调试上层常见问题

上层未识别到摄像头，使用APK打不开摄像头

使用命令：

```

dumpsy media.camera

Number of camera devices: 0

```

说明framework层没有识别到摄像头；

则先参考将CameraHal3的log开关打开；

然后使用命令：

```
logcat -G 128M && logcat -c && pkill provider && logcat
```

抓取相应的log，查看是否有：E RkCamera:报错，根据报错原因进行排查；

一般客户出现上述原因，基本是camera3_profiles.xml配置不对导致异常；

参考[camera_hal3_user_manual_vx.x.pdf](#)文档，进行排查。

上层识别到摄像头，使用APK打不开摄像头

```
Number of camera devices: 2
```

，说明framework层识别到摄像头，但是使用APK还是打不开；

这种情况一般是客户缺少tuning效果文件导致，参考logcat中提示将板子推对应的效果文件：

```

E rkisp : [XCORE]:XCAM ERROR RkAiqCalibDb.cpp:186: calibdb
/vendor/etc/camera/rkisp2/xxxxxx_default.xml and bin are all not exist!

```

rk3566-rkisp2x 2lane+2lane双mipi硬件连接注意事项

前后摄切换使用的场景：

可以只使用MIPI_RX_CLK0这组，data lane都从MIPI_RX_D0开始；

或者使用CLK0和CLK1：

MIPI_RX_CLK0和MIPI_RX_D0-1一组；

MIPI_RX_CLK1和MIPI_RX_D2-3一组

双MIPI需要同时使用的场景：

MIPI_RX_CLK0和MIPI_RX_D0-1一组；

MIPI_RX_CLK1和MIPI_RX_D2-3一组

Android9.0调试案例

SDK自带驱动，使用 v4l2-ctl命令获取数据流出现kernel崩溃问题

问题描述

在Android 9.0中使用已调试好的驱动，配置好链路后，使用 v4l2-ctl命令获取数据流出现kernel崩溃问题。关键崩溃LOG如下：

```
[ 37.667090] Internal error: Oops: 96000005 [#1] PREEMPT SMP
[ 37.669853] dwmmc_rockchip ff0f0000.dwmmc: Unexpected interrupt latency
[ 37.679306] Modules linked in: 8723cs
[ 37.683058] CPU: 0 PID: 0 Comm: swapper/0 Not tainted 4.4.167 #13
[ 37.689164] Hardware name: Rockchip rk3368 xkp avb board (DT)
[ 37.694936] task: ffffffff80091757d0 task.stack: ffffffff8009160000
[ 37.700893] PC is at camsys_mrv_irq+0x2c/0x308
[ 37.705378] LR is at handle_irq_event_percpu+0x9c/0x250
```

问题分析

查看LOG中有camsys_mrv_irq字样，camsys_mrv这个为Android8.1及更低版本使用的isp驱动代码中包含的；所以分析是dts中旧的isp结点和新的rkisp1结点同时配置，出现冲突导致异常。

解决办法

如果是RK3326平台，需要将如下结点disabled掉：

```
&rk_isp {
    status = "disabled";
};
```

如果是RK3288、RK3368平台，需要disabled的结点如下：

```
&isp {
    status = "disabled";
};
```

如果是RK3399平台，因为包含两个ISP，需要disabled的结点如下：

```
&isp 0 or 1{
    status = "disabled";
};
```

RK3126C-9.0-GC0312-预览必现分屏显示问题

问题描述

GC0312预览分屏显示异常如图，并且每次打开都是分屏。

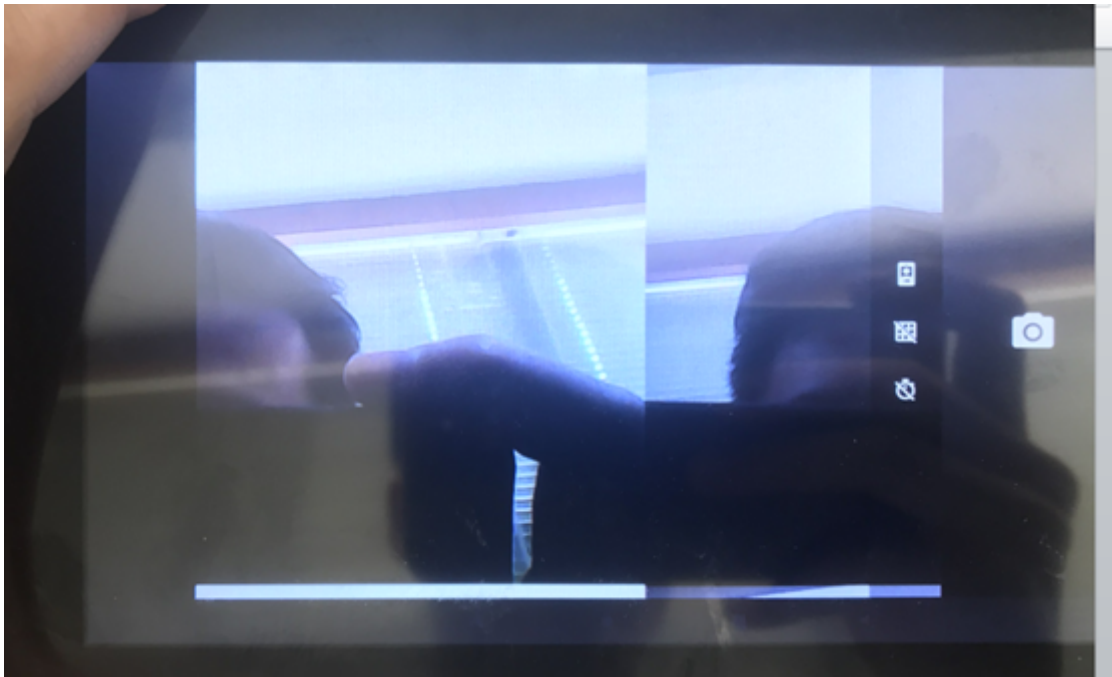


图 57 分屏显示异常图

问题分析

因为是必现，分析可能是vsync/hsync极性配置不对导致，尝试修改DTS中配置的极性试试。

解决办法

修改dts中vsync极性，原始极性配置如下：

```
209     ports {
210     → port@0 {
211     →     cif_in_fcaml endpoint@0 {
212     →     → remote-endpoint = <&gc0312_out>;
213     →     → vsync-active = <1>;
214     →     → hsync-active = <1>;
215     →     → };
```

图 58 原始vsync极性

改成如下：

```

19  → ports {
20  →   port@0 {
21  →     cif_in_fcaml endpoint@0 {
22  →       remote-endpoint = <&gc0312_out>;
23  →       vsync-active = <0>;
24  →       hsync-active = <1>;
25  →     };

```

图 59 修改后的vsync极性

RK3126C-9.0-GC2145-预览概率性分屏显示问题

问题描述

GC2145-预览概率性分屏显示如图，而且如果打开Touch Point触摸屏划线，概率会加大。



图 60 分屏显示示例图1



图 61 分屏显示示例图2

问题分析

和5.2中现象不同的是，打开Camera预览是概率性的分屏显示；而且打开TP触摸屏划线概率增大。概率性出现分屏显示时候，rkCIF驱动有如下LOG打印：

```
rkCIF: Bad frame, irq:0xf frmst:0x10001 size:605x3200
```

初步分析可能原因：

1、跟DDR带宽有关系；

2、VSYNC和HSYNC可能配置不对没问题；

3、CIF控制器和CIS（CMOS IMAGE SENSOR）的开启时序可能不对；

根据分析的可能原因做如下尝试：

1、尝试提高DDR频率，概率性分屏显示问题有改善，但多测试还是会出现；

2、尝试修改VSYNC和HSYNC极性，出现打不开摄像头，说明跟VSYNC和HSYNC无关；

3、尝试调整CIF控制器和CIS（CMOS IMAGE SENSOR）的开启数据流的时序，有效；

解决办法

调整CIF控制器和CIS（CMOS IMAGE SENSOR）开启数据流的时序，先将SENSOR的数据流开启，然后再开启CIF控制器，CIF控制器驱动代码修改如下：

```
--- a/drivers/media/platform/rockchip/cif/capture.c
+++ b/drivers/media/platform/rockchip/cif/capture.c
@@ -536,9 +536,6 @@ static int rkCIF_start_streaming(struct vb2_queue *queue,
unsigned int count)
    {
        v4l2_err(v4l2_dev, "Failed to get runtime pm, %d\n", ret);
        goto destroy_dummy_buf;
    }
-    ret = rkCIF_stream_start(stream);
-    if (ret < 0)
-        goto runtime_put;

    /* start sub-devices */
    sd = dev->active_sensor->sd;
@@ -549,6 +546,10 @@ static int rkCIF_start_streaming(struct vb2_queue *queue,
unsigned int count)
    if (ret < 0)
        goto subdev_poweroff;

+    ret = rkCIF_stream_start(stream);
+    if (ret < 0)
+        goto runtime_put;
+
    return 0;

subdev_poweroff:
```

RK3126C-9.0-GC2145录像帧率只有20fps左右，不到30fps问题

问题描述

打开CameraHal3的debug开关，然后打开Camera APK切换到录像预览界面；使用命令查看帧率：logcat | grep FPS，结果如下，只有20fps左右。

```

04-10 12:35:55.700 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.6945: nFrameCount=121
04-10 12:35:56.715 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7134: nFrameCount=141
04-10 12:35:57.732 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.6630: nFrameCount=161
04-10 12:35:58.747 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7015: nFrameCount=181
04-10 12:35:59.763 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.6871: nFrameCount=201
04-10 12:36:00.775 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7646: nFrameCount=221
04-10 12:36:01.788 7870 8287 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7220: nFrameCount=241
04-10 12:36:03.460 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 0.0004: nFrameCount=1
04-10 12:36:04.497 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.2748: nFrameCount=21
04-10 12:36:05.544 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.1008: nFrameCount=41
04-10 12:36:06.590 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.1130: nFrameCount=61
04-10 12:36:07.638 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.8941: nFrameCount=81
04-10 12:36:08.684 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.1121: nFrameCount=101
04-10 12:36:09.731 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.1005: nFrameCount=121
04-10 12:36:10.740 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 18.8419: nFrameCount=140
04-10 12:36:11.778 7870 8323 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 0.0000: nFrameCount=160
04-10 12:36:13.828 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 0.0831: nFrameCount=1
04-10 12:36:14.834 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.8789: nFrameCount=21
04-10 12:36:15.850 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.6783: nFrameCount=41
04-10 12:36:16.860 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.8138: nFrameCount=61
04-10 12:36:17.873 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7316: nFrameCount=81
04-10 12:36:18.889 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.6828: nFrameCount=101
04-10 12:36:19.906 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.6805: nFrameCount=121
04-10 12:36:20.920 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7232: nFrameCount=141
04-10 12:36:21.933 7870 8370 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.7319: nFrameCount=161
04-10 12:36:24.048 7870 8404 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 0.0815: nFrameCount=1
04-10 12:36:25.077 7870 8404 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.3870: nFrameCount=21
04-10 12:36:26.127 7870 8404 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.0920: nFrameCount=41
04-10 12:36:27.175 7870 8404 I RkCamera: <HAL> Stream: shouDebugFPS: Preview FPS : 19.8865: nFrameCount=61

```

图 62 录像预览帧率图

问题分析

RK3126C-9.0的SDK代码中，GC2145驱动有1600x1200@20fps和800x600@20fps这两组20fps的寄存器配置，在GC2145驱动里面加下如下打印信息；

```

@@ -2095,7 +2098,11 @@ static int gc2145_s_stream(struct v4l2_subdev *sd, int on)
    struct gc2145 *gc2145 = to_gc2145(sd);
    int ret = 0;

-    dev_dbg(&client->dev, "%s: on: %d\n", __func__, on);
+    //dev_dbg(&client->dev, "%s: on: %d\n", __func__, on);
+    dev_info(&client->dev, "%s: on: %d, %dx%d@%d\n", __func__, on,
+            gc2145->frame_size->width,
+            gc2145->frame_size->height,
+            gc2145->frame_size->fps);

    mutex_lock(&gc2145->lock);

```

看录像预览调下来的分辨率和帧率是多少，结果如下：

```

gc2145 2-003c: gc2145_s_stream: on: 1, 800x600@20
yujian rk816_lldo4_disable
gc2145 2-003c: gc2145_s_stream: on: 0, 800x600@20

```

图 63 录像预览分辨率帧率选择

说明录像预览的时候Camera Hal层下发的预览帧率只有20fps，经定位分析是camera3_profiles.xml中的entity没配对导致异常。

解决办法

camera3_profiles.xml中的原始配置的entity name如下：为gc2145 0-003c

```
<MediaCtl_elements RKISP1>
<element name="gc2145 0-003c" type="pixel_array"/>
<!--
<element name="rockchip-mipi-dphy-rx" type="csi_receiver"/>
-->
<element name="stream_cif" type="isys_backend"/>
</MediaCtl_elements_RKISP1>
```

图 64 原始配置的entity name

而通过media-ctl -p命令查看结果如下:

```
Device topology
- entity 1: stream_cif <1 pad, 2 links>
  type Node subtype U4L
  device node name /dev/video0
  pad0: Sink
    <- "gc0312 2-0021":0 [ENABLED]
    <- "gc2145 2-003c":0 [ ]
- entity 2: gc0312 2-0021 <1 pad, 1 link>
```

图 65 板子查看的entity name

所以上面的entity name应该修改成: gc2145 2-003c。

RK3326-9.0-GC2145烧完机后进camera概率性报错问题

问题描述

RK3326-9.0烧完机后进camera概率性报错, 刷5台有1台的样子报错, 重启机器、或恢复出厂设置就好了, 2019年8月5号的安全补丁。

问题分析

将CameraHal的log打开, 查看有如下关键LOG:

```
D RkCamera: <HAL> FlashLight: @getFlashLightInfo : hasFlash 1, flashNode0: ,
flashNode1:

E RkCamera: <HAL> FlashLight: init: Unable to open node ''
```

分析可能跟FLASH相关参数初始化有关。

解决办法

在SDK/hardware/rockchip/camera目录查看是否有如下提交: 如果没有找FAE获取

```
commit 2fb29b54b6e94ee6303bc4988abaf085247fb87b
Author: Zhong Yichong <zyc@rock-chips.com>
Date: Thu Oct 10 08:28:24 2019 +0800

    fix hal crashed by uninitialized var mFlashNum

Change-Id: I3fc16e8dada0201eb2158ab500e5c95a1ec401bc
Signed-off-by: Zhong Yichong <zyc@rock-chips.com>
```

图 66 问题解决提交点

或者直接使用如下增加mFlashNum初始化修改:

```

diff --git a/common/platformdata/PlatformData.cpp
b/common/platformdata/PlatformData.cpp

index b183dff..a416061 10diff --git a/common/platformdata/PlatformData.cpp
b/common/platformdata/PlatformData.cpp
index b183dff..a416061 100644
--- a/common/platformdata/PlatformData.cpp
+++ b/common/platformdata/PlatformData.cpp
@@ -747,6 +747,8 @@ status_t CameraHWInfo::findAttachedSubdevs(const std::string
&mcPath,

    LOGI("@%s", __FUNCTION__);

+   drv_info.mFlashNum = 0;
+
    int fd = open(mcPath.c_str(), O_RDONLY);
    if (fd == -1) {
        LOGW("Could not open media controller device: %s!", strerror(errno));

```

RK3326-9.0-打开camera预览界面概率性出现绿屏

问题描述

恢复出厂设置后第一次打开Camera的概率更大，一般重新开机第一次打开camera后容易复现这个问题。

测试过程中发现，深度休眠后，打开摄像头会整个预览界面都是绿色的，切换一下模式（拍照->录像）或者退出再进，预览就会正常，附件是图片和log，请看下

问题分析

打开相机进入深度休眠，然后唤醒后就会出现绿屏；可能跟IOMMU有关；

深度休眠后，ISP_IOMMU被关闭了；唤醒后没有被打开，从而导致绿屏；

解决办法

在SDK/kernel目录查看是否有如下提交：如果没有找FAE获取

```

commit 4b00ec7c60d66c36608504d9f453e4799f6cf122
Author: Cai YiWei <cyw@rock-chips.com>
Date: Mon Jun 17 08:53:27 2019 +0800

    media: rockchip: isp1: fix isp iommu work after suspend

Change-Id: Ie89ec58c6f99a751bc1fdf681a85a6595716c649
Signed-off-by: Cai YiWei <cyw@rock-chips.com>

```

RK3326-9.0-mipi camera预览卡顿问题

问题描述

打开系统相机，预览画面很卡顿，点击拍照也是很卡；

问题分析

分析目前预览画面卡顿可能原因有如下几个：

- 1、本身摄像头输出的帧率比较低，看着预览卡顿；
- 2、rga 没有打开，导致使用软件做裁剪和缩放,从而卡顿；

问题分析所需log：

- 1、参考附4.2把camera_hal的log打开， logcat | grep FPS看下预览帧率；

抓取完整的logcat；

- 2、kernel的完整log；

经查看关键LOG如下：有rga报错，很大可能就是rga未使能

```
E rockchiprga: This output the user patamaters when rga call blit fail
E          : CropScaleNV12Or21:rga blit failed
```

解决办法

客户参考rk3326-863-lp3-v10-avb.dts配置把rga相关的结点打开就正常了；

```
&rk_rga {
    status = "okay";
};
```

RK3326-9.0-支持同时打开两路camera(Defect #258741)

问题描述

基于rkpx30 android9.0客户需要支持同时打开两路camera,目前测试只能单独打开，当打开第二个的时候报错；

问题分析

因为双摄需要同时工作，则需要两个控制器+两个接口同时进行接收和处理。如果RK3326需要支持双摄，则camera的连接方式要求如下：

- 1、ISP控制器使用MIPI接口camera；
- 2、CIF控制器使用DVP接口camera；

经查看客户原理图，连接方式符合要求；那就是dts中cif控制器和isp控制器配置了；及CameraHal3对于支持多个摄像头需要做些修改；

解决办法

- 1) 示例dts配置如下：

```
&rkisp1 {
    status = "okay";
```


RK3326-9.0-SOC摄像头录像过程拍照，录像视频拍照时间点会闪一下问题

问题描述

APK退出后第一次打开，录像点击拍照第一张照片异常问题：



问题分析

使用命令：while true;do cat /proc/kmsg; done;

关键LOG：

```
61 +[33mrkisp1+[]n: end kfifo size: 0, frame_id 0
81 +[33mrkisp1+[]n: end kfifo size: 0, frame_id 0
61 +[33mgc2145 2-003c+[]n: gc2145_s_frame_interval match wxh@FPS is 800x600@20
71 +[33mrkisp1+[]n: end kfifo size: 0, frame_id 0
11 +[33mgc2145 2-003c+[]n: gc2145_s_frame_interval match wxh@FPS is 800x600@30
51 +[33mrkisp1+[]n: end kfifo size: 0, frame_id 0
```

为什么录像的时候点击拍照会调用20fps帧率去？

gc2145 2-003c: gc2145_s_frame_interval match wxh@FPS is 800x600@20

跟踪CameraHal代码

经跟踪CameraHal中psl/rkisp1/ControlUnit.cpp中录像第一次拍照点击拍照的时候

```
557: ControlUnit::processSoCSettings(const CameraMetadata *settings)
558: {
559:     // fill target fps range, it needs to be proper in results anyway
560:     camera_metadata_ro_entry entry =
561:         settings->find(ANDROID_CONTROL_AE_TARGET_FPS_RANGE);
562:     ALOGE("%s:%d enter!\n ", __FUNCTION__, __LINE__);
563:     if (entry.count == 2) {
564:         int32_t maxFps = entry.data.i32[1];
565:         // set to driver
566:         ALOGE("%s:%d enter maxFps = %d!\n ", __FUNCTION__, __LINE__, maxFps);
567:         if (mSensorSubdev.get())
568:             mSensorSubdev->setFramerate(0, maxFps);
569:     }
570:
571:     return OK;
572: }
573:
574: /**
```

```
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 20!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
2 E ControlUnit: processSoCSettings:566 enter maxFps = 30!
```

是上层设置下来的，第一次录像点击拍照会设置如下参数：

```

E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 15 maxFps = 20!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!
E ControlUnit: processSoCSettings:567 enter: minFps= 30 maxFps = 30!

```

尝试获取：ANDROID_CONTROL_CAPTURE_INTENT的entry看看

```

android_control_capture_intent_preview );
./common/platformdata/metadataAutoGen/6.0.1/MetadataInfoAutoGen.h:466: {"control.captureIntent", AND
ROID_CONTROL_CAPTURE_INTENT},
wpzz@ubuntu:~/4 rk3399 9.0 Gerrit code/hardware/rockchip/camera$

```

有如下选项：

android.control.captureIntent	byte (enum) [public] [legacy] CUSTOM (v3.2) - The goal of this request doesn't fall into the other categories. The camera device will default to preview-like behavior. PREVIEW (v3.2) - This request is for a preview-like use case. The precapture trigger may be used to start off a metering w/flash sequence. STILL_CAPTURE (v3.2) - This request is for a still capture-type use case. If the flash unit is under automatic control, it may fire as needed. VIDEO_RECORD (v3.2) - This request is for a video recording use case. VIDEO_SNAPSHOT (v3.2) - This request is for a video snapshot (still image while recording video) use case.
-------------------------------	---

对应：

```
hardware\rockchip\camera_engine_rkisp\metadata\libcamera_metadata\include\system
```

中 camera_metadata_tags.h 文件的如下位置：

```

// ANDROID_CONTROL_CAPTURE_INTENT
typedef enum camera_metadata_enum_android_control_capture_intent {
    ANDROID_CONTROL_CAPTURE_INTENT_CUSTOM,
    ANDROID_CONTROL_CAPTURE_INTENT_PREVIEW,
    ANDROID_CONTROL_CAPTURE_INTENT_STILL_CAPTURE,
    ANDROID_CONTROL_CAPTURE_INTENT_VIDEO_RECORD,
    ANDROID_CONTROL_CAPTURE_INTENT_VIDEO_SNAPSHOT,
    ANDROID_CONTROL_CAPTURE_INTENT_ZERO_SHUTTER_LAG,
    ANDROID_CONTROL_CAPTURE_INTENT_MANUAL,
    ...
}

```

尝试解决办法 Solution

尝试根据ANDROID_CONTROL_CAPTURE_INTENT的获取的entry值区分录像拍照

当获取的是录像时候拍照情况，控制SOC的camera不再设置帧率；

获取ANDROID_CONTROL_CAPTURE_INTENT的获取的entry值和metadata是对应的：

PREVIEW为：1；

VIDEO_RECORD为：3；

VIDEO_SNAPSHOT为: 4;

尝试修改后初步OK的提交:

```
commit a52845a91c728aa10cfe80b94a1a3cfb6f047150
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date:   Fri Mar 6 09:52:12 2020 +0800

    fix soc camera video Snapshot issues

    The framerate changes when take video snapshot for the first time;
    it will causes the recorded video flashed in the snapshot moment;
    so fix it.

Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Change-Id: I445af271bb50845a42b3a34e9c8c077faf436c15
```

RK3326-9.0-打开补光灯录像模式，重新进入camera app会闪一下灯再常亮补光灯(Defect ## 254851)

问题描述

打开补光灯录像模式，重新进入camera app会闪一下灯再常亮补光灯，

问题分析

因为camera的cts都过了，所以应该是上层APK调用问题；需要分析APK调用哪里不对；

解决办法

查看SDK/packages/apps/Camera2目录是否有如下提交：如果没有找FAE获取；

```
commit 9ccd5504ce70669f526343c48ff200f7cbad0e76
Author: chenxiao <cx@rock-chips.com>
Date:   Thu Jun 18 17:18:19 2020 +0800

    Camera v1.0.16:

    fix flash torch->off->torch status change error when video Module.
    refs for redmine#254851

Signed-off-by: chenxiao <cx@rock-chips.com>
Change-Id: I92c6d098d45a1b8a5c4761a5792b1390ffceb641
```

10.0对应提交点:

```
commit ef0ca6db282e5b3bb245a2deae199e0a0b901716
Author: chenxiao <cx@rock-chips.com>
Date: Thu Jun 18 17:18:19 2020 +0800

    Camera v1.0.16:
        fix flash torch->off->torch status change error when video Module.
        refs for redmine#254851

Signed-off-by: chenxiao <cx@rock-chips.com>
Change-Id: I2b680ede456488a56e4af589952d8e7f3de56862
```

RK3288-9.0/10.0-IMX327/214摄像头拍照及录像预览闪绿屏问题

问题描述

Defect #274492、Defect #278358

客户反馈：使用原生自动的APK预览，都显示绿色或者预览画面与绿色交替显示，拍照可以保存图片，图片颜色正常。但如果用客户自己写的app预览都是正常的。

问题分析

逻辑上不管使用的哪个APK，都是会调到驱动和效果文件的，除非客户写的APK调的另外一个摄像头；建议客户把用他们APP预览的kernel的log也抓出来看下。

先定位是不是hwc问题：

1、adb 下 敲screenrecord /sdcard/1.mp4 然后打开系统camera app，几秒后退出， 暂停screenrecord命令，然后打开1.mp4 看下录屏是否也有绿屏；

如果没有，那就是hwc模块有问题，

如果也有绿屏请按下面第2点抓下camera图像数据。

2、执行以下指令后，打开系统camera app，几秒后退出app，camera数据会保存到/data/dump目录下，请打包上来分析

```
adb root
adb remount
adb shell setprop persist.vendor.camera.dump 1
adb shell setenforce 0
adb shell mkdir /data/dump
adb shell chmod 777 /data/dump/
adb shell setprop persist.vendor.camera.dump.skip 10
adb shell setprop persist.vendor.camera.dump.cnt 10
adb shell setprop persist.vendor.camera.dump.path /data/dump/
dump 出来的 就是绿的。
```

使用v4l2-ctl工具抓取原始数据查看

```
v4l2-ctl -d /dev/video0 --set-fmt-video=width=1280,height=720,pixelformat=NV12 \
--stream-mmap=8 --stream-count=5 --stream-to=/data/dump/cap_720p.yuv --stream-
poll

v4l2-ctl -d /dev/video0 --set-fmt-video=width=1920,height=1080,pixelformat=NV12 \
--stream-mmap=8 --stream-count=5 --stream-to=/data/dump/cap_1080p.yuv --stream-
poll
```

不用打开camera，直接敲下这两个指令，把保存下来的cap_720p.yuv和cap_1080p.yuv发上来,经查看v4l2抓的数据也没问题;

对比客户APP预览和原生APP预览的pipeline配置差异

在客户应用camera预览时抓下 `media-ctl -p /dev/media0` 和

系统camera app 预览时的 `media-ctl -p /dev/media0`

```
16 Device topology
17 - entity 1: rkisp1-isp-subdev (4 pads, 6 links)
18     type V4L2 subdev subtype Unknown
19     device node name /dev/v4l-subdev0
20     pad0: Sink
21         [fmt:SRGB10/1920x1080
22         crop.bounds:(0,0)/1920x1080
23         crop:(0,0)/1920x1080]
24         <- "rkisp1_dmapath":0 []
25         <- "rockchip-mipi-dphy-rx":1 [ENABLED]
26     pad1: Sink
27         <- "rkisp1-input-params":0 [ENABLED]
28     pad2: Source
29         [fmt:YUYV2X8/1920x1080
30         crop.bounds:(0,0)/1920x1080
31         crop:(0,0)/1920x1080]
32         -> "rkisp1_selfpath":0 [ENABLED]
33         -> "rkisp1_mainpath":0 [ENABLED]
```

图 67 原生apk media-ctrl

```
16 Device topology
17 - entity 1: rkisp1-isp-subdev (4 pads, 6 links)
18     type V4L2 subdev subtype Unknown
19     device node name /dev/v4l-subdev0
20     pad0: Sink
21         [fmt:SRGB10/1920x1080
22         crop.bounds:(0,0)/1920x1080
23         crop:(0,0)/1920x1080]
24         <- "rkisp1_dmapath":0 []
25         <- "rockchip-mipi-dphy-rx":1 [ENABLED]
26     pad1: Sink
27         <- "rkisp1-input-params":0 [ENABLED]
28     pad2: Source
29         [fmt:YUYV2X8/1920x1080
30         crop.bounds:(0,0)/1920x1080
31         crop:(0,0)/1920x1080]
32         -> "rkisp1_selfpath":0 []
33         -> "rkisp1_mainpath":0 [ENABLED]
34     pad3: Source
```

经对比发现在没有打开selfpath数据流情况下是正常的;

分析是如果MP和SP同时输出的话，会增加DDR带宽;

客户内部跟硬件确认了下

三星的内存 -- 没有问题，没有补丁，摄像头预览也是正常

南亚的内存 -- 有问题，只有打了补丁，摄像头预览才是正常

说明确实是DDR问题;

解决办法

只用mainpath输出, 而不用selfpath输出, 因为如果MP和SP同时输出的话, 会增加DDR带宽;

SDK/hardware/rockchip/camera目录修改:

```
diff --git a/psl/rkisp1/RKISP1CameraHw.cpp b/psl/rkisp1/RKISP1CameraHw.cpp
index 8bf6bc5..04965f5 100644
--- a/psl/rkisp1/RKISP1CameraHw.cpp
+++ b/psl/rkisp1/RKISP1CameraHw.cpp
@@ -596,7 +596,9 @@ status_t RKISP1CameraHw::doConfigureStreams(UseCase
newUseCase,
        newUseCase == USECASE_STILL ? "USECASE_STILL" : "USECASE_TUNING",
        streams.size());

-    mGCM.enableMainPathOnly(newUseCase == USECASE_STILL ? true : false);
+    //mGCM.enableMainPathOnly(newUseCase == USECASE_STILL ? true : false);
+    mGCM.enableMainPathOnly(true);
+    LOGE("wpzz====, enable Main Path Only\n");

    status_t status = mGCM.configStreams(streams, operation_mode,
testPatternMode);
    if (status != NO_ERROR) {
```

Android9.0 CaptureRequestTest#testEdgeModeControl 失败问题

问题描述

RAW Camera进行CtsCameraTestCases

```
android.hardware.camera2.cts.CaptureRequestTest#testEdgeModeControl[1]
android.hardware.camera2.cts.CaptureRequestTest#testNoiseReductionModeControl[1]
```

失败LOG:

```
camera 1: Frame duration must be in the range of [33333333, 66666666], value
32813000 is out of range [32833332, 67666664])
camera 1: Frame duration must be in the range of [33333333, 66666666], value
32815000 is out of range [32833332, 67666664])
```

问题分析

32815000这个值很接近33333333了, 只是后面三位可能因为精度问题被过滤掉了; 需要修改下;

解决办法

查看SDK/hardware/rockchip/camera_engine_rkisp\$需要包含如下提交即可;

```
commit f636d4ddde37e6ecbb52778af05d4d64c514bf6b
Author: Zhong Yichong <zyc@rock-chips.com>
Date: Mon Jun 10 10:39:14 2019 +0800

    fix cts frame_duration bug caused by precision

Change-Id: I3fdcc2a1b9cc73806cf02b6f6bd63f9fc831c0a5
Signed-off-by: Zhong Yichong <zyc@rock-chips.com>
```

RK3399-9.0-IMX214调试中binning分辨率2104x1560拍照，一拍照相机APK立马退出问题

问题描述

相机中选择拍照分辨率：3.3M

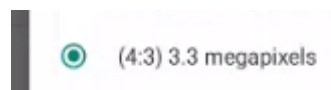


图 67 APK选择3.3M

然后点击拍照，相机立马退出。

问题分析

查看问题LOG:

```
D      : (pCfgFull->inputWidth & (15)) != 0
D      : JpegEncSetPictureSize: ERROR Out of range image dimension(s)
E hw_jpeg_encode: JPEGENCDOER: JpegEncSetPictureSize fail.
D      : JpegEncRelease#
D      : JpegEncRelease: OK
D hw_jpeg_encode: --- RUN OUT JPEGENCDOER. jpeg len : 0
E RkCamera: <HAL> ImgHWEncoder: @encodeSync 263: hw jpeg encode fail.
E RkCamera: <HAL> PostProcessPipeline: @processFrame, JPEG conversion failed!
[-2147483648]!
```

关键LOG: (pCfgFull->inputWidth & (15)) != 0

分析应该是因为HWENC编码JPEG要求16位对齐。

解决办法

把camera3_profiles.xml里面的拍照分辨率：2104x1560改成：2096x1560，修改后3.3M拍照OK；

RK3399-9.0-IMX214-配置为RAW之后，预览出现后马上卡死问题

问题描述

单独点前摄IMX214，配置为SOC的时候，预览拍照都正常，但是配置为RAW之后，预览出现之后就马上卡死。

问题分析

查看问题LOG:

```
E rkisp : [NO]:XCAM ERROR rkisp_control_loop_impl.cpp:358: can't access iq file  
/vendor/etc/camera/rkisp1/imx214_f_lens1_50013A7_OTP1.xml !
```

从LOG看出是缺少效果文件;

解决办法

复制一份对应版本的imx214效果文件, 改成复制一份示例效果文件改成

imx214_f_lens1_50013A7_OTP1.xml, 然后推到板子/vendor/etc/camera/rkisp1/目录下重启下应该就能正常打开了。

RK3399-9.0-IMX214-4096x3120分辨率拍照竖条纹问题

问题描述

Imx214 使用4096x3120分辨率拍照如图有竖条纹



问题分析

查看问题LOG:

```
04-29 14:45:24.996 7068 7245 E      : CropScaleNV12Or21(52): out wxh 4208x3120  
beyond rga capability
```

从LOG看出是使用调用rga超出范围, 但默认camera3_profiles.xml已经将支持的输出分辨率设置为了: 4096x3120了; 继续分析原因;

解决办法

Sdk/hardware/rockchip/camera\$目录需要有如下提交;

c

```
commit 67185f69a1dc170ed27f9da5501bd6dacc66acc9
Author: Zhong Yichong <zyc@rock-chips.com>
Date:   Fri Aug 23 11:11:05 2019 +0800

    limit isp path output size to the capacity of rga

Change-Id: Icf5d2c94ede630a24eba96d2bdb117c7363ef4aa
Signed-off-by: Zhong Yichong <zyc@rock-chips.com>
```

如没有，更新代码后cherry-pick过来或者找FAE获取；

RK3399-9.0-OV8858-4lane录像1080P帧率只有15FPS问题

问题描述

RK3399上两个ISP均连接有Camera，一个ISP接hm2051，一个ISP接OV8858。如果dts中两个都配置上，OV8858-4lane只能到15FPS；而如果不配置H2051，只配置OV8858-4lane就能到30FPS。

问题分析

逻辑上Sensor输出的帧率只跟写入OV8858的寄存器配置一样，驱动及配置一样的话，帧率应该一样的，分析可能跟MCLK有关。经分析定位最终发现前置hm2051驱动里面用的MCLK是用12MHZ，而OV8858所需MCLK为24MHZ；但是摄像头驱动中只在probe的时候设置了一次MCLK，导致后置摄像头中的MCLK设置被前置覆盖，从而导致OV8858的输出帧率减半。

解决办法

参考ov5695驱动在xxxx_power_on或xxxx_s_power中重新设置clk；

```
ret = clk_set_rate(ov5695->xvclk, OV5695_XVCLK_FREQ);
if (ret < 0) {
    dev_err(dev, "Failed to set xvclk rate (24MHz)\n");
    return ret;
}
if (clk_get_rate(ov5695->xvclk) != OV5695_XVCLK_FREQ)
    dev_warn(dev, "xvclk mismatched, modes are based on 24MHz\n");
ret = clk_prepare_enable(ov5695->xvclk);
if (ret < 0) {
    dev_err(dev, "Failed to enable xvclk\n");
    return ret;
}
```

Android10.0调试案例

RK3126C-10.0-GC0312-预览必现分屏显示问题

问题描述

GC0312预览分屏显示异常如图，并且每次打开都是分屏。



图 70 分屏显示异常图

问题分析

因为是必现，分析可能是vsync/hsync极性配置不对导致，尝试按照5.2中修改，无效；

查看代码中还有哪里配置vsync/hsync极性，跟踪发现gc0312.c驱动中gc0312_g_mbus_config这个函数中会配置

```

895: static int gc0312_g_mbus_config(struct v4l2_subdev *sd,
896:                                struct v4l2_mbus_config *config)
897: {
898:     config->type = V4L2_MBUS_PARALLEL;
899:     config->flags = V4L2_MBUS_HSYNC_ACTIVE_HIGH |
900:                   V4L2_MBUS_VSYNC_ACTIVE_HIGH |
901:                   V4L2_MBUS_PCLK_SAMPLE_RISING;
902:
903:     return 0;
904: }
905:

```

这里面VSYNC的极性配置反了，需要修改；

7.1.2 解决办法

```

--- a/drivers/media/i2c/gc0312.c
+++ b/drivers/media/i2c/gc0312.c
@@ -897,7 +897,7 @@ static int gc0312_g_mbus_config(struct v4l2_subdev *sd,
 {
     config->type = V4L2_MBUS_PARALLEL;
     config->flags = V4L2_MBUS_HSYNC_ACTIVE_HIGH |
-                   V4L2_MBUS_VSYNC_ACTIVE_HIGH |
+                   V4L2_MBUS_VSYNC_ACTIVE_LOW |
                     V4L2_MBUS_PCLK_SAMPLE_RISING;

     return 0;

```

RK3326-10.0-OV2680拍照预览帧率一直为30FPS问题

问题描述

拍照预览帧率是30FPS，而不是变动的16.6到30FPS

```
showDebugFPS: Preview FPS : 29.9151: mFrameCount=116
showDebugFPS: Preview FPS : 29.9789: mFrameCount=146
showDebugFPS: Preview FPS : 30.0605: mFrameCount=177
showDebugFPS: Preview FPS : 29.9664: mFrameCount=208
showDebugFPS: Preview FPS : 30.0456: mFrameCount=239
showDebugFPS: Preview FPS : 29.9766: mFrameCount=270
showDebugFPS: Preview FPS : 30.0216: mFrameCount=301
```

问题分析

CameraHal3_v2.1.0之前的camerahal，是profile优先；

CameraHal3_v2.1.0&& camera_engine_rkisp_v2.3.0之后是tuning文件优先。

解决办法

修改效果文件：

```
.....<ExposureSeparate index="1" type="cell" size="[1.2]">
.....<cell index="1" type="struct" size="[1.7]">
.....<name index="1" type="char" size="[1.6]">
.....normal
.....</name>
.....<TimeDot index="1" type="double" size="[1.6]">
.....[0.0.03.0.03.0.03.0.03.0.03]
.....</TimeDot>
.....<LTimeDot index="1" type="double" size="[1.6]">
.....[0.0.03.0.03.0.03.0.03.0.03]
.....</LTimeDot>
.....<STimeDot index="1" type="double" size="[1.6]">
.....[0.0.03.0.03.0.03.0.03.0.03]
.....</STimeDot>
```

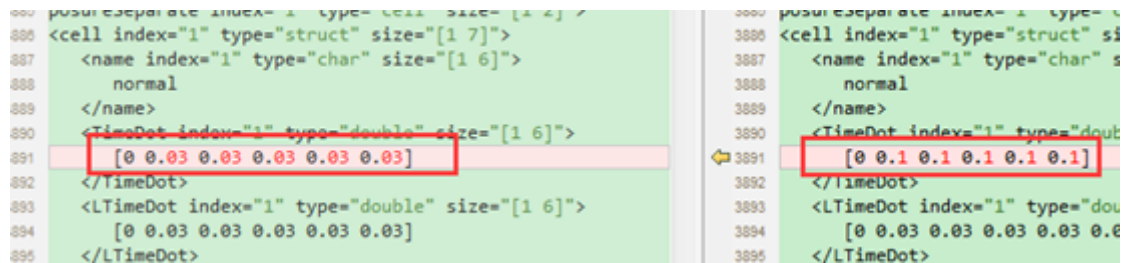
修改成如下：

```
.....<ExposureSeparate index="1" type="cell" size="[1.2]">
.....<cell index="1" type="struct" size="[1.7]">
.....<name index="1" type="char" size="[1.6]">
.....normal
.....</name>
.....<TimeDot index="1" type="double" size="[1.6]">
.....[0.0.1.0.1.0.1.0.1.0.1]
.....</TimeDot>
.....<LTimeDot index="1" type="double" size="[1.6]">
.....[0.0.03.0.03.0.03.0.03.0.03]
.....</LTimeDot>
```

第一个元素认为是最小帧间隔

最后一个元素认为是最大帧间隔

```
<TimeDot index="1" type="double" size="[1.6]">
  [0.0.1.0.1.0.1.0.1.0.1]
</TimeDot>
```



RK3326-10.0-MIPI和DVP摄像头兼容配置问题

问题描述

后摄的dvp和miipi的摄像头都兼容进来，但兼容了miipi的摄像头后摄的dvp就都不亮了，把miipi摄像头去掉又能亮了。

问题LOG:

```
gc2145 2-003c: Found GC2145 sensor
gc2145 2-003c: gc2145_detect success!
gc2145 2-003c: gc2145->module_index = 00, facing = b, dev_name = 2-003c
gc2145 2-003c: async_register_subdev_sensor_return = -16
```

问题分析

查看LOG，GC2145的ID已经识别到了，但async_register_subdev_sensor_return = -16；说明是注册失败，这个LOG打印地方为：

v4l2_async_register_subdev_sensor_common这个函数返回-16，

```
#define ENOTBLK      15 /* Block device required */
#define EBUSY        16 /* Device or resource busy */
```

对应device or resource busy

跟踪查看v4l2_async_register_subdev_sensor_common调用的哪个函数可能返回-16；

即return -EBUSY；

一步步分析定位：n->ops->bound(n, subdev, asd);调用的是：

drivers/media/platform/rockchip/isp/dev.c中的subdev_notifier_bound函数；

```
468: static int subdev_notifier_bound(struct v4l2_async_notifier *notifier,
469: >> >> >> struct v4l2_subdev *subdev,
470: >> >> >> struct v4l2_async_subdev *asd)
471: {
472: >> struct rkisp1_device *isp_dev = container_of(notifier,
473: >> >> >> struct rkisp1_device, notifier);
474: >> struct rkisp1_async_subdev *s_asd = container_of(asd,
475: >> >> >> struct rkisp1_async_subdev, asd);
476:
477: >> if (isp_dev->num_sensors == ARRAY_SIZE(isp_dev->sensors))
478: >> return -EBUSY;
```

这里返回-EBUSY；

分析是ARRAY_SIZE(isp_dev->sensors)这个值太小；

```

139: » struct v4l2_ctrl_handler ctrl_handler;
140: » struct media_device media_dev;
141: » struct v4l2_async_notifier notifier;
142: » struct v4l2_subdev *subdevs[RKISP1_SD_MAX];
143: » struct rkisp1_sensor_info *active_sensor;
144: » struct rkisp1_sensor_info sensors[RKISP1_MAX_SENSOR];
145: » int num_sensors;
146: ..

58: #define RKISP1_MAX_BUS_CLK» 8
59: #define RKISP1_MAX_SENSOR» 2
60: #define RKISP1_MAX_PIPELINE» 4

```

RKISP1_MAX_SENSOR为2，而如果有mipi camera的情况，可能mipi sensor在后面注册；

但是phy-rockchip-mip-rx已经先有结点了，导致isp_dev->num_sensors的值达到了2，从而导致异常。通过添加如下打印信息：

```

diff --git a/drivers/media/platform/rockchip/isp1/dev.c
b/drivers/media/platform/rockchip/isp1/dev.c
old mode 100644
new mode 100755
index a865baaea6f7..ff527c67e79a
--- a/drivers/media/platform/rockchip/isp1/dev.c
+++ b/drivers/media/platform/rockchip/isp1/dev.c
@@ -474,6 +474,8 @@ static int subdev_notifier_bound(struct v4l2_async_notifier
 *notifier,
     struct rkisp1_async_subdev *s_asd = container_of(asd,
                                                         struct rkisp1_async_subdev, asd);

+    v4l2_info(subdev, "isp_dev->num_sensors(%d), ARRAY_SIZE(isp_dev->sensors)
(%d)\n",
+                isp_dev->num_sensors, ARRAY_SIZE(isp_dev->sensors));
     if (isp_dev->num_sensors == ARRAY_SIZE(isp_dev->sensors))
         return -EBUSY;

```

结果如下，和分析一致；所以解决办法是将RKISP1_MAX_SENSOR值加大。

```

RK3326-10.0\白 能达-RK3326-10.0 摄像头兼容问题(bug #283140);\双GC2145兼容\双21
| rockchip-mipi-dphy-rx: isp_dev->num_sensors(0), ARRAY_SIZE(isp_dev->sensors) (4)
| m01_f_gc0312 2-0021-1: isp_dev->num_sensors(1), ARRAY_SIZE(isp_dev->sensors) (4)
| m01_f_gc2145 2-003c: isp_dev->num_sensors(2), ARRAY_SIZE(isp_dev->sensors) (4)

```

解决办法

```

diff --git a/drivers/media/platform/rockchip/isp1/dev.h
b/drivers/media/platform/rockchip/isp1/dev.h
old mode 100644
new mode 100755
index 980f7711302d..762c9d9ff498
--- a/drivers/media/platform/rockchip/isp1/dev.h
+++ b/drivers/media/platform/rockchip/isp1/dev.h
@@ -56,7 +56,7 @@
 #define GRP_ID_ISP_DMARX BIT(5)

 #define RKISP1_MAX_BUS_CLK 8
-#define RKISP1_MAX_SENSOR 2
+#define RKISP1_MAX_SENSOR 4
 #define RKISP1_MAX_PIPELINE 4

 #define RKISP1_MEDIA_BUS_FMT_MASK 0xF000

```

RK3326-10.0-camera-GTS错误

问题描述

```
com.google.android.assistant.gts.AssistanTest#testAssistantTakeSelfieWithoutVoiceInteraction
com.google.android.assistant.gts.AssistanTest#testAssistantOpenFrontCameraWithVoiceInteraction
com.google.android.assistant.gts.AssistanTest#testAssistantOpenRearCameraWithoutVoiceInteraction
com.google.android.assistant.gts.AssistanTest#testAssistantTakePhotoWithoutVoiceInteraction
com.google.android.assistant.gts.AssistanTest#testAssistantOpenFrontCameraWithVoiceInteraction
```

armeabi-v7a GtsAssistantHostTestCases		
Test	Result	
com.google.android.assistant.gts.AssistanTest#testAssistantTakeSelfieWithoutVoiceInteraction	fail	java.lang.AssertionError
com.google.android.assistant.gts.AssistanTest#testAssistantOpenFrontCameraWithVoiceInteraction	fail	java.lang.AssertionError
com.google.android.assistant.gts.AssistanTest#testAssistantOpenRearCameraWithoutVoiceInteraction	fail	java.lang.AssertionError
com.google.android.assistant.gts.AssistanTest#testAssistantTakePhotoWithoutVoiceInteraction	fail	java.lang.AssertionError
com.google.android.assistant.gts.AssistanTest#testAssistantOpenRearCameraWithVoiceInteraction	fail	java.lang.AssertionError
com.google.android.assistant.gts.AssistanTest#testAssistantOpenFrontCameraWithoutVoiceInteraction	fail	java.lang.AssertionError

解决办法

packages/apps/Camera2这下面打上这个补丁：

[./Rockchip_Trouble_Shooting_CameraHAL3_CN/patches/0001-camera-support-voice-action-from-Google-Assistant.patch](#)

如没有，找FAE获取：

RK3326-10.0-camera-vts错误

问题描述

报错log 如下：

```
03-30 04:47:02.711 22197 22197 W ServiceManagement: Waited one second for android.hardware.camera.provider@2.4::ICameraProvider/legacy/0

03-30 04:47:02.712 22197 22197 I ServiceManagement: getService: Trying again for android.hardware.camera.provider@2.4::ICameraProvider/legacy/0...

03-30 04:47:03.713 22197 22197 W ServiceManagement: Waited one second for [android.hardware.camera.provider@2.4::ICameraProvider/legacy/0] (mailto:android.hardware.camera.provider@2.4::ICameraProvider/legacy/0)
```

问题分析

和前面调试RK3399-10.0-CTS的有点类似：

```
'android.hardware.camera.provider@2.4::ICameraProvider/external/0' for
ctl.interface_start

[ 175.768249] init: Received control message 'interface_start' for
'android.hardware.camera.provider@2.4::ICameraProvider/external/0' from pid: 231
(/system/bin/hwservicemanager)
```

只是这个是external camera的服务；

尝试正常的时候复现如下错误：

```
W ServiceManagement: Waited one second for
android.hardware.camera.provider@2.4::ICameraProvider/legacy/0

I ServiceManagement: getService: Trying again for
[android.hardware.camera.provider@2.4::ICameraProvider/legacy/0]
(mailto:android.hardware.camera.provider@2.4::ICameraProvider/legacy/0) ..
```

如何在正常版本复现上述报错：

尝试把\vendor/etc/init目录的android.hardware.camera.provider@2.4- service.rc删除，然后重启后，

板子的logcat就出现上面循环报错：

解决办法

不是camera相关问题，是系统本身异常导致；

Android10.0-CTS和GSI CaptureRequestTest#testFlashTurnOff失败问题

问题描述

CtsCameraTestCases

```
android.hardware.camera2.cts.CaptureRequestTest#testFlashTurnOff
```

这个闪光灯的fail过不了

问题分析

android.hardware.camera2.cts.CaptureRequestTest#testFlashTurnOff 这项为10.0新增的测试；

CX分析问题LOG：

```
E TestRunner: failed:
testFlashTurnOff(android.hardware.camera2.cts.CaptureRequestTest)

E TestRunner: ----- begin exception -----

E TestRunner: java.lang.Exception: There were 6 errors:

E TestRunner: java.lang.Throwable(Test failed for camera 0: Invalid state 3 not
in expected list[4, 2] (expected = -1, actual = -1) )
```

```

E TestRunner: java.lang.Throwable(Test failed for camera 0: state 3 index -1 is
expected to be >= 0: (expected = 0 was not greater than actual = -1) )

E TestRunner: java.lang.Throwable(Test failed for camera 0: Invalid state 3 not
in expected list[4, 2] (expected = -1, actual = -1) )

E TestRunner: java.lang.Throwable(Test failed for camera 0: state 3 index -1 is
expected to be >= 0: (expected = 0 was not greater than actual = -1) )

E TestRunner: java.lang.Throwable(Test failed for camera 0: Invalid state 3 not
in expected list[4, 2] (expected = -1, actual = -1) )

E TestRunner: java.lang.Throwable(Test failed for camera 0: state 3 index -1 is
expected to be >= 0: (expected = 0 was not greater than actual = -1) )

```

解决办法

如果是SOC的camera如下修改:

sdk/hardware/rockchip/camera\$ git diff

```

diff --git a/psl/rkisp1/ControlUnit.cpp b/psl/rkisp1/ControlUnit.cpp
index d6a6a97..56f7229 100755
--- a/psl/rkisp1/ControlUnit.cpp
+++ b/psl/rkisp1/ControlUnit.cpp
@@ -155,9 +155,14 @@ int SocCamFlashCtrUnit::updateFlashResult(CameraMetadata
*result)

    uint8_t flashState = ANDROID_FLASH_STATE_READY;
    if (mV4lFlashMode == V4L2_FLASH_LED_MODE_FLASH ||
-       mV4lFlashMode == V4L2_FLASH_LED_MODE_TORCH)
+       mV4lFlashMode == V4L2_FLASH_LED_MODE_TORCH) {
        flashState = ANDROID_FLASH_STATE_FIRED;

+       if (mAeMode >= ANDROID_CONTROL_AE_MODE_ON
+           && mAeFlashMode == ANDROID_FLASH_MODE_OFF) {
+           flashState = ANDROID_FLASH_STATE_PARTIAL;
+       }
+   }

    // # ANDROID_METADATA_Dynamic android.flash.state done
    result->update(ANDROID_FLASH_STATE, &flashState, 1);

```

如果是RAW的camera如下修改:

sdk/hardware/rockchip/camera_engine_rkisp\$ git diff

```

diff --git a/modules/isp/rkiq_handler.cpp b/modules/isp/rkiq_handler.cpp
index 5dec438..3612b9b 100755
--- a/modules/isp/rkiq_handler.cpp
+++ b/modules/isp/rkiq_handler.cpp
@@ -698,6 +698,7 @@ AiqCommonHandler::processMiscMetaResults(struct
CamIA10_Results &ia10_results, X
    const CameraMetadata* settings =
        &_aiq_compositor->getAiqInputParams()->settings;
    uint8_t flash_mode = ANDROID_FLASH_MODE_OFF;
+    uint8_t ae_mode = ANDROID_CONTROL_AE_MODE_ON;

```

```

        camera_metadata_ro_entry entry_flash =
            settings->find(ANDROID_FLASH_MODE);

@@ -705,7 +706,9 @@ AiqCommonHandler::processMiscMetaResults(struct
CamIA10_Results &ia10_results, X
            flash_mode = entry_flash.data.u8[0];
        }
        metadata->update(ANDROID_FLASH_MODE, &flash_mode, 1);
-
+        camera_metadata_ro_entry entry_ae_mode = settings-
>find(ANDROID_CONTROL_AE_MODE);
+        if (entry_ae_mode.count == 1)
+            ae_mode = entry_ae_mode.data.u8[0];
        uint8_t flashState = ANDROID_FLASH_STATE_READY;

        struct CamIA10_Stats& camia10_stats =
@@ -715,8 +718,12 @@ AiqCommonHandler::processMiscMetaResults(struct
CamIA10_Results &ia10_results, X
            camia10_stats.flash_status.flash_mode == HAL_FLASH_TORCH ||
            /* CTS required */
            flash_mode == ANDROID_FLASH_MODE_SINGLE ||
-            flash_mode == ANDROID_FLASH_MODE_TORCH)
+            flash_mode == ANDROID_FLASH_MODE_TORCH) {
        flashState = ANDROID_FLASH_STATE_FIRED;
+        if (ae_mode >= ANDROID_CONTROL_AE_MODE_ON
+            && flash_mode == ANDROID_FLASH_MODE_OFF)
+            flashState = ANDROID_FLASH_STATE_PARTIAL;
+    }
        else if (camia10_stats.frame_status ==
CAMIA10_FRAME_STATUS_FLASH_PARTIAL)
            flashState = ANDROID_FLASH_STATE_PARTIAL;
        metadata->update(ANDROID_FLASH_STATE, &flashState, 1);

```

RK3326-10.0接后摄MIPI口GC2145，前摄DVP口GC032A出现切换报错

问题描述

打开gc2145出现：如下报错

```

[ 170.987576] gc2145 2-003c: gc2145_s_stream: on: 1, 800x600@16

[ 171.238218] rkisp1: MIPI error: err2: 0x00010000

[ 171.293869] rkisp1: MIPI error: err2: 0x00010000

```

切换到gc032a出现如下报错：

```

[ 93.394564] rkisp1 ff4a0000.rkisp1: can not get first iq setting in stream on

[ 93.395319] rkisp1: CIF_ISP_PIC_SIZE_ERROR (0x00000001)

[ 93.395337] rkisp1: CIF_ISP_PIC_SIZE_ERROR (0x00000001)

[ 93.395930] rkisp1: CIF_ISP_PIC_SIZE_ERROR (0x00000001)

```

```
[ 93.396494] rkisp1: isp icr v_statr err: 0x48

[ 93.397173] gc032a 2-0021: gc032a_s_stream: on: 1

[ 93.397396] rkisp1: CIF_ISP_PIC_SIZE_ERROR (0x00000001)

[ 93.397420] rkisp1: CIF_ISP_PIC_SIZE_ERROR (0x00000001)

[ 93.398397] rkisp1: CIF_ISP_PIC_SIZE_ERROR (0x00000001)
```

问题分析

这些报错原因基本就两种：

- 1、本身sensor输出信号质量差，导致报错；
- 2、sensor输出相互影响，导致信号异常；

排查方法：

- 1、尝试更换板子或者模组试试；
- 2、怀疑可能是gc032a影响gc2145信号，将gc032a驱动stream off后等待一帧时间，然后在切换到gc2145试试；
- 3、gc2145驱动中，`clk_set_rate(gc2145->xvclk, 24000000)`;改成120000000或者60000000试试；
- 4、gc2145的供电电压,要客户根据自身硬件及根据gc2145的Datasheet再核对下；

解决办法

两camera 硬件的i2c io电源域不同，一个1.8v，一个2.8v，解决办法是客户修改硬件。

RK3326-10.0-DVP和MIPI摄像头都兼容问题

问题描述

Android10.0系统，想把后摄的dvp和mipi的摄像头都兼容进来，但兼容了mipi的摄像头后摄的dvp就都不亮了，把mipi摄像头去掉又能亮了，能不能做到所有mipi和dvp摄像头都兼容进来？

问题分析

经分析问题原因：

是MIPI camera中的rockchip-mipi-dphy-rx:有先注册占用了`isp_dev->num_sensors`，

然后DVP注册的时候如果已经注册了一个，

但只要多一个就超了2的界限，而这时MIPI camera还没有进行解析，所以相关资源还没有释放，所以导致异常了。

kernel添加打印LOG位置：

```
diff --git a/drivers/media/platform/rockchip/isp1/dev.c
b/drivers/media/platform/rockchip/isp1/dev.c
old mode 100644
new mode 100755
index a865baaea6f7..ff527c67e79a
--- a/drivers/media/platform/rockchip/isp1/dev.c
+++ b/drivers/media/platform/rockchip/isp1/dev.c
```

```

@@ -474,6 +474,8 @@ static int subdev_notifier_bound(struct v4l2_async_notifier
*notifier,
    struct rkisp1_async_subdev *s_asd = container_of(asd,
                                                    struct rkisp1_async_subdev, asd);

+    v4l2_info(subdev, "isp_dev->num_sensors(%d), ARRAY_SIZE(isp_dev->sensors)
(%d)\n",
+                isp_dev->num_sensors, ARRAY_SIZE(isp_dev->sensors));
    if (isp_dev->num_sensors == ARRAY_SIZE(isp_dev->sensors))
        return -EBUSY;

```

解决办法

kernel按照如下修改:

```

diff --git a/drivers/media/platform/rockchip/isp1/dev.h
b/drivers/media/platform/rockchip/isp1/dev.h
old mode 100644
new mode 100755
index 980f7711302d..762c9d9ff498
--- a/drivers/media/platform/rockchip/isp1/dev.h
+++ b/drivers/media/platform/rockchip/isp1/dev.h
@@ -56,7 +56,7 @@
#define GRP_ID_ISP_DMARX                BIT(5)

#define RKISP1_MAX_BUS_CLK              8
-#define RKISP1_MAX_SENSOR              2
+#define RKISP1_MAX_SENSOR              4
#define RKISP1_MAX_PIPELINE            4

#define RKISP1_MEDIA_BUS_FMT_MASK       0xF000

```

Android11.0调试案例

RK3566/68- android11.0 如何配置MIPI YUV camera问题

问题描述及问题解答

使用的MIPI摄像头，在RK3399-7.1上配置的是CamSys_Fmt_Yuv422_8b，在rk3566上应该怎么配置呢？

CamSys_Fmt_Yuv422_8b是7.1上用的，9.0及以上的框架不一样，需要参考新的文档进行配置。

有没文档或类似的例程？

可以参考SDK/RKDocs/common/camera/HAL3中RKISP_Driver_User_Manual_v1.xx.pdf和amera_hal3_user_manual_vx.x.pdf文档；rk3566比较新，结点名称和文档有些差异；

是否有参考的dts配置？

可参考如下dts或dtsi配置：

1、rk3566-evb2-lp4x-v10.dtsi：包含MIPI-4lane及2lane+2lane配置；

其中csi2_dphy0对应正常MIPI-4lane配置；

2lane+2lane配置参考：csi2_dphy1和csi2_dphy2示例连接；

2、rk3566-evb3-ddr3-v10.dtsi：包含ov5695-mipi camera+ gc2145-dvp camera配置；

3、rk3566-rk817-tablet.dts 、 rk3566-rk817-tablet-k108.dts、 rk3566-rk817-tablet-rkg11.dts、

rk3566-rk817-eink-w6.dts等这些dts中也包含camera配置，只是可能使用的时钟和模组有些差异客户可自行参考；

具体参考：[rk356x平台摄像头DTS注册](#)

camera_hal3_user_manual_v2.3.pdf这个文档提到的 RAW sensor 及:SOC sensor 其中SOC sensor类型就是YUV输出的sensor吗？

是的，RAW Sensor指的就是输出Bayer Pattern格式的sensor；

而SOC Sensor指的是输出YUV 格式的sensor；

dts中，这些name 要与那些代码或配置保持一致？

```
rockchip,camera-module-name = "HS5885-BNSM1018-V01";
rockchip,camera-module-lens-name = "default";
```

YUYV输出的SOC sensor，没影响；只有RAW输出的sensor需要考虑，

具体参考camera_engine_rkisp_user_manual_vx.x.pdf文档说明；

不能获取到数据大概是什么原因，模组测量是有数据输出的？

参考本文档4.9部分。

RK356x-android11.0-gc5035-rkisp0: MIPI error: packet: 0x01000000 报错问题

问题描述

打开camera后报如下mipi 错误：

```
824.486548] rkisp0: MIPI error: packet: 0x01000000
824.486630] rkisp0: MIPI error: packet: 0x01000000
824.486750] rkisp0: MIPI error: packet: 0x01000000
824.487049] rkisp0: MIPI error: packet: 0x01000000
824.487549] rkisp0: MIPI error: packet: 0x01000000
824.487619] rkisp0: MIPI error: packet: 0x01000000
```

问题分析

查找报错位置：

```
./drivers/media/platform/rockchip/isp/capture_v21.c:1468:
v4l2_warn(v4l2_dev, "MIPI error: packet: 0x%08x\n", packet);
packet = readl(hw_dev->base_addr + CSI2RX_ERR_PACKET);
```

查找rk3568.dts中 isp的基地址为：0xfdff0000

对应寄存器偏移地址：0x1c00+0x0024 = 0x1c24

```

500: #define CSI2RX_BASE 0x00001C00
501: #define CSI2RX_CTRL0 (CSI2RX_BASE + 0x000000)
502: #define CSI2RX_CTRL1 (CSI2RX_BASE + 0x000004)
503: #define CSI2RX_CTRL2 (CSI2RX_BASE + 0x000008)
504: #define CSI2RX_CSI2_RESETN (CSI2RX_BASE + 0x000010)
505: #define CSI2RX_PHY_STATE_RO (CSI2RX_BASE + 0x000014)
506: #define CSI2RX_DATA_IDS_1 (CSI2RX_BASE + 0x000018)
507: #define CSI2RX_DATA_IDS_2 (CSI2RX_BASE + 0x00001c)
508: #define CSI2RX_ERR_PHY (CSI2RX_BASE + 0x000020)
509: #define CSI2RX_ERR_PACKET (CSI2RX_BASE + 0x000024)
510: #define CSI2RX_ERR_OVERFLOW (CSI2RX_BASE + 0x000028)
511: #define CSI2RX_ERR_STAT (CSI2RX_BASE + 0x00002c)

```

0x01000000对应bit[27:24] = 1，即CRC checksum error，就是mipi信号质量问题导致出错；

CSI2RX_1C00_ERR_PACKET			
Address: Operational Base + offset (0x0024)			
Bit	Attr	Reset Value	Description
31:28	RO	0x0	reserved
27:24	RC	0x0	checksum_error CRC checksum error
23:21	RO	0x0	reserved
20	RC	0x0	ecc_2bit_error ecc 2bit error
19:16	RC	0x0	ecc_1bit_error ecc 1bit error
15:12	RC	0x0	err_id id err
11:8	RC	0x0	err_frame_data The frame whose transfer is being finished had at last one CRC err detected
7:4	RC	0x0	err_f_seq The Frame Number of two consecutive packets does not follow the expcted order

尝试方法：

尝试将MCLK降低到12MHZ后，没有出现上面报错，说明就是mipi信号质量的问题；

需要模组重新打样。

解决办法

模组优化mipi走线，重新打样。

RK356x-android11.0-ov8858-rkisp0: MIPI error: overflow: 0x00000001报错问题

问题描述

打开camera后报如下mipi 错误：

```
[ 498.458507] rkisp0: MIPI error: overflow: 0x00000001
[ 498.458519] rkisp0: MIPI error: packet: 0x01100000
[ 498.458530] rkisp0: MIPI error: packet: 0x00100000
[ 498.466320] rkisp0: MIPI error: packet: 0x01000000
[ 498.466326] rkisp0: MIPI error: packet: 0x00010000
```

问题分析

查找报错位置:

```
./drivers/media/platform/rockchip/isp/capture_v2l.c:1470:
v4l2_warn(v4l2_dev, "MIPI error: overflow: 0x%08x\n", overflow);

./drivers/media/platform/rockchip/isp/capture_v2l.c:1468:
v4l2_warn(v4l2_dev, "MIPI error: packet: 0x%08x\n", packet);

packet = readl(hw_dev->base_addr + CSI2RX_ERR_PACKET);

overflow = readl(hw_dev->base_addr + CSI2RX_ERR_OVERFLOW);
```

查找rk3568.dts中 isp的基地址为: 0xfdff0000

对应寄存器偏移地址: 0x1c00+0x0024 = 0x1c24和0x1c28

```
512:
513: #define CSI2RX_BASE 0x00001C00
514: #define CSI2RX_CTRL0 (CSI2RX_BASE + 0x000000)
515: #define CSI2RX_CTRL1 (CSI2RX_BASE + 0x000004)
516: #define CSI2RX_CTRL2 (CSI2RX_BASE + 0x000008)
517: #define CSI2RX_CSI2_RESETN (CSI2RX_BASE + 0x000010)
518: #define CSI2RX_PHY_STATE_RO (CSI2RX_BASE + 0x000014)
519: #define CSI2RX_DATA_IDS_1 (CSI2RX_BASE + 0x000018)
520: #define CSI2RX_DATA_IDS_2 (CSI2RX_BASE + 0x00001c)
521: #define CSI2RX_ERR_PHY (CSI2RX_BASE + 0x000020)
522: #define CSI2RX_ERR_PACKET (CSI2RX_BASE + 0x000024)
523: #define CSI2RX_ERR_OVERFLOW (CSI2RX_BASE + 0x000028)
524: #define CSI2RX_ERR_STAT (CSI2RX_BASE + 0x00002c)
525: #define CSI2RX_MASK_PHY (CSI2RX_BASE + 0x000030)
```

解决办法

经分析排查, 是OV8858驱动中添加sensor_g_mbus_config中4lane配置出错了:

4lane配置成了2lane:

```
+static int sensor_g_mbus_config(struct v4l2_subdev *sd,
+                               struct v4l2_mbus_config *config)
+{
+    struct ov8858 *sensor = to_ov8858(sd);
+    struct device *dev = &sensor->client->dev;
+
+    dev_info(dev, "%s(%d) enter!\n", __func__, __LINE__);
+
+    if (2 == sensor->lane_num) {
+        config->type = V4L2_MBUS_CSI2;
+        config->flags = V4L2_MBUS_CSI2_2_LANE |
+                        V4L2_MBUS_CSI2_CHANNEL_0 |
+                        V4L2_MBUS_CSI2_NONCONTINUOUS_CLOCK;
+    } else if (4 == sensor->lane_num) {
+        config->type = V4L2_MBUS_CSI2;
+        config->flags = V4L2_MBUS_CSI2_2_LANE |
```

```
+
+             V4L2_MBUS_CSI2_CHANNEL_0 |
+             V4L2_MBUS_CSI2_NONCONTINUOUS_CLOCK;
+     } else {
+         dev_err(&sensor->client->dev,
+             "unsupported lane_num(%d)\n", sensor->lane_num);
+     }
+     return 0;
+}
```

改成如下即可：

```
+static int sensor_g_mbus_config(struct v4l2_subdev *sd,
+
+    struct v4l2_mbus_config *config)
+{
+    struct ov8858 *sensor = to_ov8858 (sd);
+    struct device *dev = &sensor->client->dev;
+
+    dev_info(dev, "%s(%d) enter!\n", __func__, __LINE__);
+
+    if (2 == sensor->lane_num) {
+        config->type = V4L2_MBUS_CSI2;
+        config->flags = V4L2_MBUS_CSI2_2_LANE |
+            V4L2_MBUS_CSI2_CHANNEL_0 |
+            V4L2_MBUS_CSI2_NONCONTINUOUS_CLOCK;
+    } else if (4 == sensor->lane_num) {
+        config->type = V4L2_MBUS_CSI2;
+        config->flags = V4L2_MBUS_CSI2_4_LANE |
+            V4L2_MBUS_CSI2_CHANNEL_0 |
+            V4L2_MBUS_CSI2_NONCONTINUOUS_CLOCK;
+    } else {
+        dev_err(&sensor->client->dev,
+            "unsupported lane_num(%d)\n", sensor->lane_num);
+    }
+    return 0;
+}
```

RK356x-11.0-Camera配置为RAW之后，预览出现后马上卡住问题

问题描述

单独点camera，配置为SOC的时候，预览拍照都正常，但是配置为RAW之后，预览出现之后就马上卡死。

问题分析

使用如下命令抓取LOG：

```
adb shell logcat -c && adb shell pkill provider && adb shell logcat > logcat.txt
```

查看问题LOG：

Check LOG of the issue:

```
E rkisp : [XCORE]:XCAM ERROR RkAiqCalibDb.cpp:186: calibdb
/vendor/etc/camera/rkisp2/ov8858_KYT-4878-V01_default.xml and bin are all not
exist!
```

从LOG看出是缺少效果文件；

解决办法

因为rk356x是rkisp2x版本，效果文件位于：

```
sdk\external\camera_engine_rkaiq\iqfiles
```

复制一份分辨率相同，版本相同的效果文件，改成一份示例效果文件，名字改成：

ov8858_KYT-4878-V01_default.xml，然后推到板子/vendor/etc/camera/rkisp2/目录后重启下应该就能正常打开了。

RK356x-11.0- CTS和GSI CaptureRequestTest#testFlashTurnOff[1]失败问题

问题描述

CtsCameraTestCases

android.hardware.camera2.cts.CaptureRequestTest#testFlashTurnOff[1]

这个闪光灯的fail过不了

问题LOG：

```
java.lang.Throwable(Test failed for camera 0: Invalid state 3 not in expected
list[4, 2] (expected = -1, actual = -1) )

java.lang.Throwable(Test failed for camera 0: state 3 index -1 is expected to be
>= 0: (expected = 0 was not greater than actual = -1) )

java.lang.Throwable(Test failed for camera 0: Invalid state 3 not in expected
list[4, 2] (expected = -1, actual = -1) )

java.lang.Throwable(Test failed for camera 0: Invalid state 3 not in expected
list[4, 2] (expected = -1, actual = -1) )

java.lang.Throwable(Test failed for camera 0: Invalid state 3 not in expected
list[2, 4] (expected = -1, actual = -1) )
```

问题分析

同5.19类似；只是目前RK356x的RAW和SOC都是类似5.19中SOC修改；

解决办法

修改如下：

```
sdk/hardware/rockchip/camera$ git diff
diff --git a/psl/rkisp2/RKISP2ControlUnit.cpp b/psl/rkisp2/RKISP2ControlUnit.cpp
index b8fbbb1..bdd9021 100755
```

```

--- a/psl/rkisp2/RKISP2ControlUnit.cpp
+++ b/psl/rkisp2/RKISP2ControlUnit.cpp
@@ -180,6 +180,12 @@ int SocCamFlashCtrUnit::updateFlashResult(CameraMetadata
*result)
    }
}

+    /* Using android.flash.mode == TORCH or SINGLE will always return FIRED.*/
+    if (mAeFlashMode == ANDROID_FLASH_MODE_TORCH ||
+        mAeFlashMode == ANDROID_FLASH_MODE_SINGLE) {
+        ALOGD("%s:%d mAeFlashMode: %d, set flashState FIRED!", __FUNCTION__,
+            __LINE__, mAeFlashMode);
+        flashState = ANDROID_FLASH_STATE_FIRED;
+    }
+
    // # ANDROID_METADATA_DYNAMIC android.flash.state done
    result->update(ANDROID_FLASH_STATE, &flashState, 1);

```

RK356x-11.0- CTS和GSI CaptureRequestTest#testFlashControl[1]失败问题

问题描述

CtsCameraTestCases

android.hardware.camera2.cts.CaptureRequestTest#testFlashControl[1]

这个闪光灯的fail过不了

问题LOG:

```

java.lang.Exception: There were 2 errors:

    java.lang.Throwable(Test failed for camera 0: Flash state result must be FIRED
(expected = 3, actual = 2) )

    java.lang.Throwable(Test failed for camera 0: Flash state result must be FIRED
(expected = 3, actual = 2) )

```

加上android.hardware.camera2.cts.CaptureRequestTest#testFlashTurnOff[1]的解决修改后:

剩余一个:

```

java.lang.Exception: Test failed for camera 0: Flash state result must be FIRED
(expected = 3, actual = 2)

```

问题分析

分析是android.flash.state的状态返回不对;

应该是FIRED 即3状态, 而实际返回的时候READY状态;

android.flash.state	byte (enum) [public] [limited] • UNAVAILABLE (v3.2) - No flash on camera. • CHARGING (v3.2) - Flash is charging and cannot be fired. • READY (v3.2) - Flash is ready to fire. • FIRE (v3.2) - Flash fired for this capture. • PARTIAL (v3.2) - Flash partially illuminated this frame. This is usually due to the next or previous frame having the flash fire, and the flash spilling into this capture due to hardware limitations.	Current state of f
---------------------	---	--------------------

返回FIRED状态有如下两种情况

When the camera device doesn't have flash unit (i.e. `android.flash.info.available == false`), this state will always be current flash status.

In certain conditions, this will be available on LEGACY devices:

- Flash-less cameras always return UNAVAILABLE.
- Using `android.control.aeMode == ON_ALWAYS_FLASH` will always return **FIRED**.
- Using `android.flash.mode == TORCH` will always return **FIRED**.

In all other conditions the state will not be available on LEGACY devices (i.e. it will be `null`).

解决办法

RK356x的camera如下修改:

```

sdk/hardware/rockchip/camera$ git diff
diff --git a/psl/rkisp2/RKISP2ControlUnit.cpp b/psl/rkisp2/RKISP2ControlUnit.cpp
index b8fbbb1..bdd9021 100755
--- a/psl/rkisp2/RKISP2ControlUnit.cpp
+++ b/psl/rkisp2/RKISP2ControlUnit.cpp
@@ -180,6 +180,12 @@ int SocCamFlashCtrUnit::updateFlashResult(CameraMetadata
 *result)
     }
 }

+ /* Using android.flash.mode == TORCH or SINGLE will always return FIRED.*/
+ if (mAeFlashMode == ANDROID_FLASH_MODE_TORCH ||
+     mAeFlashMode == ANDROID_FLASH_MODE_SINGLE) {
+     ALOGD("%s:%d mAeFlashMode: %d, set flashState FIRED!", __FUNCTION__,
+ __LINE__, mAeFlashMode);
+     flashState = ANDROID_FLASH_STATE_FIRED;
+ }
+
 // # ANDROID_METADATA_DYNAMIC android.flash.state done
 result->update(ANDROID_FLASH_STATE, &flashState, 1);

```

RK356x-11.0- CTS FlashlightTest失败问题

问题描述

```
CtsCameraTestCases
```

```
android.hardware.camera2.cts.FlashlightTest#testTorchCallback[1]
```

```
android.hardware.camera2.cts.FlashlightTest#testTorchModeExceptions[1]
```

```
android.hardware.camera2.cts.FlashlightTest#testCameraDeviceOpenAfterTorchOn[1]
```

```
android.hardware.camera2.cts.FlashlightTest#testSetTorchModeOnOff[1]
```

这几个闪光灯的fail过不了

失败LOG:

```
java.lang.IllegalArgumentException: setTorchMode:1948: Camera "0" has no
flashlight

at
android.hardware.camera2.CameraManager.throwAsPublicException(CameraManager.java:
1001)

at
android.hardware.camera2.CameraManager$CameraManagerGlobal.setTorchMode(CameraMan
ager.java

at android.hardware.camera2.CameraManager.setTorchMode(CameraManager.java:772)

at
android.hardware.camera2.cts.FlashlightTest.resetTorchModeStatus(FlashlightTest.j
ava:291)

at
android.hardware.camera2.cts.FlashlightTest.testTorchCallback(FlashlightTest.java
:148)

at
android.hardware.camera2.cts.FlashlightTest.testTorchCallback(FlashlightTest.java
:137)
```

问题分析

上面就是打印了调用流程:

```
FlashlightTest.java:137-> FlashlightTest.java:148à FlashlightTest.java:291à
CameraManager.java:772à
```

添加打印信息:

```
setprop log.tag.CameraManagerGlobal V
```

按照如下添加LOG定位:

```
--- a/services/camera/libcameraservice/CameraFlashlight.cpp
+++ b/services/camera/libcameraservice/CameraFlashlight.cpp
@@ -16,7 +16,7 @@

#define LOG_TAG "CameraFlashlight"
```

```

#define ATRACE_TAG ATRACE_TAG_CAMERA
-// #define LOG_NDEBUG 0
+#define LOG_NDEBUG 0

#include <utils/Log.h>
#include <utils/Trace.h>
diff --git a/services/camera/libcameraservice/CameraService.cpp
b/services/camera/libcameraservice/CameraService.cpp
old mode 100644
new mode 100755
index 4bfb441836..0ed116c6d2
--- a/services/camera/libcameraservice/CameraService.cpp
+++ b/services/camera/libcameraservice/CameraService.cpp
@@ -16,7 +16,7 @@

#define LOG_TAG "CameraService"
#define ATRACE_TAG ATRACE_TAG_CAMERA
-// #define LOG_NDEBUG 0
+#define LOG_NDEBUG 0

#include <algorithm>
#include <climits>

```

如下报错:

```

V CameraFlashlight: createFlashlightControl: creating a flash light control for
camera 0
V CameraFlashlight: setTorchMode: set camera 0 torch mode to 1
D RkCamera: Debug log file is not enabled
D RkCamera: Debug log file is not enabled
E CameraService: setTorchMode: Camera "0" has no flashlight

```

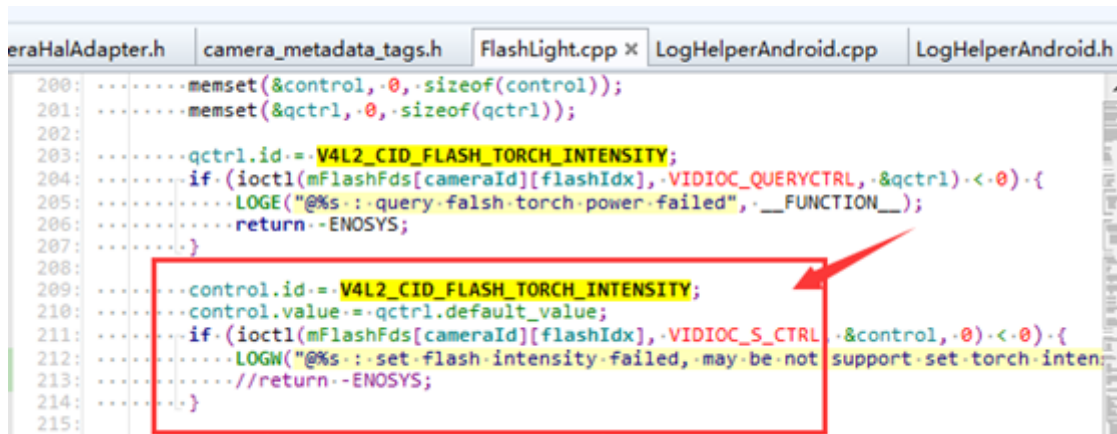
继续定位如下:

```

E RkCamera: <HAL> FlashLight: @setFlashMode : set flash intensity failed
D Camera3HALModule: EXIT-hal_set_torch_mode
D CameraFlashlight: setTorchMode(123): wpzz Camera device 0: res(-38).

```

FlashLight.cpp定位是这里报错:



使用命令:

```
grep -H '' /sys/class/video4linux/v4l-subdev*/name
```

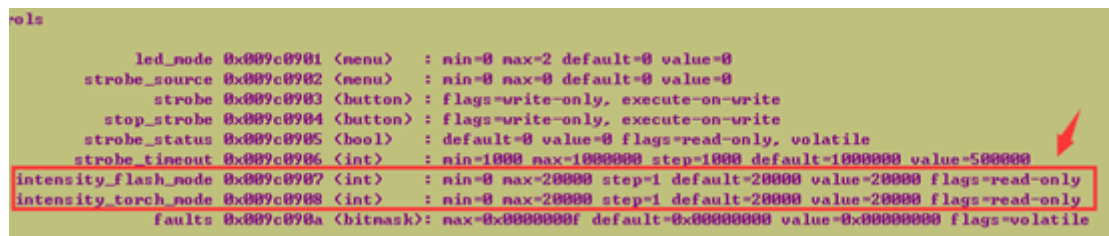
查看闪光灯对应的结点:

```
RK3566:/ # grep -H '' /sys/class/video4linux/v4l-subdev*/name
/sys/class/video4linux/v4l-subdev0/name:rkisp-isp-subdev
/sys/class/video4linux/v4l-subdev1/name:rkisp-csi-subdev
/sys/class/video4linux/v4l-subdev2/name:rockchip-csi2-dphy2
/sys/class/video4linux/v4l-subdev3/name:rockchip-csi2-dphy1
/sys/class/video4linux/v4l-subdev4/name:m01_f_gc5025 2-0037-2
/sys/class/video4linux/v4l-subdev5/name:m00_b_hi846 2-0020-1
/sys/class/video4linux/v4l-subdev6/name:m00_b_gpio-flash
/sys/class/video4linux/v4l-subdev7/name:m00_b_dw9714 2-000c
```

然后使用命令: `v4l2-ctl -d /dev/v4l-subdev6 --all`, 查看该结点支持的控制

`V4L2_CID_FLASH_TORCH_INTENSITY`对应的ioctl仅支持读, 不支持写;

因为`leds-rgb13h.c`驱动是gpio控制的闪光灯, 确实不支持配置亮度;



```
ols
    led_mode 0x009c0901 <menu> : min=0 max=2 default=0 value=0
    strobe_source 0x009c0902 <menu> : min=0 max=0 default=0 value=0
    strobe 0x009c0903 <button> : flags=write-only, execute-on-write
    stop_strobe 0x009c0904 <button> : flags=write-only, execute-on-write
    strobe_status 0x009c0905 <bool> : default=0 value=0 flags=read-only, volatile
    strobe_timeout 0x009c0906 <int> : min=1000 max=1000000 step=1000 default=1000000 value=500000
    intensity_flash_mode 0x009c0907 <int> : min=0 max=20000 step=1 default=20000 value=20000 flags=read-only
    intensity_torch_mode 0x009c0908 <int> : min=0 max=20000 step=1 default=20000 value=20000 flags=read-only
    faults 0x009c090a <bitmask>: max=0x0000000f default=0x00000000 value=0x00000000 flags=volatile
```

只能将`camerahal`中根据是否支持设置亮度, 进行处理就行了;

解决办法

SDK/hardware/rockchip/camera/目录添加修改

```
diff --git a/common/FlashLight.cpp b/common/FlashLight.cpp
index 4383b8d..99915f4 100644
--- a/common/FlashLight.cpp
+++ b/common/FlashLight.cpp
@@ -205,12 +205,14 @@ int32_t FlashLight::setFlashMode(const int cameraId, int
flashIdx, const bool mo
    LOGE("%s : query falsh torch power failed", __FUNCTION__);
    return -ENOSYS;
}

-
-    control.id = V4L2_CID_FLASH_TORCH_INTENSITY;
-    control.value = qctrl.default_value;
-    if (ioctl(mFlashFds[cameraId][flashIdx], VIDIOC_S_CTRL, &control, 0) <
0) {
-        LOGE("%s : set flash intensity failed", __FUNCTION__);
-        return -ENOSYS;
+    LOGD(" qctrl.flags(0x%08x)", qctrl.flags);
+    if (qctrl.flags != V4L2_CTRL_FLAG_READ_ONLY) {
+        control.id = V4L2_CID_FLASH_TORCH_INTENSITY;
+        control.value = qctrl.default_value;
+        if (ioctl(mFlashFds[cameraId][flashIdx], VIDIOC_S_CTRL, &control, 0)
< 0) {
```

```
+         LOGE("@%s : set flash intensity failed, may be not support set  
torch intensity.", __FUNCTION__);  
+         return -ENOSYS;  
+     }  
}  
  
memset(&control, 0, sizeof(control));
```

RK356x-11.0 -CTS StaticMetadataTest#testCapabilities[1]失败问题

问题描述

CtsCameraTestCases

android.hardware.camera2.cts.StaticMetadataTest#testCapabilities[1]

问题LOG:

```
java.lang.Exception: Test failed for camera 1: CONTROL_AF_REGIONS is available  
but afMaxRegion is 0  
    at  
    android.hardware.camera2.cts.testcases.Camera2AndroidTestCase.tearDown(Camera2And  
roidTestCase.java:155)  
    at java.lang.reflect.Method.invoke(Native Method)  
    at  
    org.junit.runners.model.FrameworkMethod$1.runReflectiveCall(FrameworkMethod.java:  
50)
```

问题分析

从上述LOG分析应该是马达相关的一些配置没去掉;

解决办法

主要是sensor不支持AF的话需要去掉相应字段:

<request.availableRequestKeys 中需要去掉: control.afRegions 字段

<request.availableResultKeys 中需要去掉: control.afRegions 字段

RK3126-11.0-2G-rgo版单前摄2145-CTS的CtsAppTestCases失败问题

问题描述

CtsAppTestCases

android.app.cts.SystemFeaturesTest#testCameraFeatures

单测case项失败

问题分析

跟camera的配置有关系;

解决办法

单前摄2145不支持聚焦，机器中关于camera的feature只需以下两个feature:

```
rkxxxx:$ pm list features |grep camera

feature:android.hardware.camera.any

feature:android.hardware.camera.front
```

所以需要如下修改配置才能pass:

1. CAMERA_SUPPORT_AUTOFOCUS = false
2. device/rockchip/common/device.mk文件中的
frameworks/native/data/etc/android.hardware.camera.xml:\$(TARGET_COPY_OUT_VENDOR)/etc/permissions/android.hardware.camera.xml去掉
3. frameworks/native/data/etc/go_handheld_core_hardware.xml文件中的也去掉

RK356x-11.0-AHD to DVP接口转换芯片报错调试建议

问题描述

```
[ 139.967158] rkCIF_dvp: ERROR: DVP_ALL_ERROR_INTEN:0x706!!

[ 139.986431] rkCIF_dvp: Bad frame, pp irq:0x403 frmst:0x10001 size:720x24

[ 139.987065] rkCIF_dvp: dvp line err
```

问题分析

看LOG是收到的行数不对；这输出的是逐行还是隔行的数据？

如下几点调试建议：

- 1、将VSYNC和HSYNC极性修改试试：

```
static int rn6752_g_mbus_config(struct v4l2_subdev *sd,
                               struct v4l2_mbus_config *config)
{
    config->type = V4L2_MBUS_PARALLEL;
    config->flags = V4L2_MBUS_HSYNC_ACTIVE_HIGH |
                   V4L2_MBUS_VSYNC_ACTIVE_LOW |
                   V4L2_MBUS_PCLK_SAMPLE_RISING;
    return 0;
}
```

- 2、如是bt656传输格式，参考adv7181实现querystd函数；

```
.querystd = adv7181_querystd,
```

一般配置：V4L2_STD_PAL

如果是BT601就不用实现querystd函数；

- 3、驱动中根据情况配置，隔行和逐行

```
format->field = V4L2_FIELD_NONE; //逐行
```

或

```
format->field = V4L2_FIELD_INTERLACED; //隔行
```

RK356x-11.0-开启双摄支持修改

问题描述

更新最新R7版本平台双摄用不了，

配置的这个链路：sensor2->csi2_dphy2->mipi_csi2->vicap->isp_vir1，出现打不开的问题，报错如下：

```
E rkCIF_mipi_lvds: get_remote_sensor: remote pad is null

E rkisp : [XCORE]:XCAM ERROR v4l2_device.cpp:184: open device(/dev/video0)
failed

E rkCIF_mipi_lvds: rkCIF_update_sensor_info: stream[0] get remote sensor_sd
failed!

E rkisp : [XCORE]:XCAM ERROR v4l2_device.cpp:184: open device(/dev/video1)
failed

E stream_cif_mipi_id0: update sensor info failed -19

E rkCIF_mipi_lvds: get_remote_sensor: remote pad is null

E rkCIF_mipi_lvds: rkCIF_update_sensor_info: stream[1] get remote sensor_sd
failed!

E stream_cif_mipi_id1: update sensor info failed -19

E RkCamera: <HAL> RKISP2ControlUnit: Could not retrieve media entity rkisp-isp-
subdev
```

解决办法

只需要SDK/hardware/rockchip/camera\$目录如下修改即可；

```
diff --git a/common/platformdata/PlatformData.cpp
b/common/platformdata/PlatformData.cpp

index 4d9257f..91cf277 100755

--- a/common/platformdata/PlatformData.cpp
+++ b/common/platformdata/PlatformData.cpp

@@ -1047,7 +1047,7 @@ CameraHWInfo::CameraHWInfo() :

    mProductName = "<not_set>";

    mManufacturerName = "<not_set>";
```

```
mCameraDeviceAPIVersion = CAMERA_DEVICE_API_VERSION_3_3;

- mSupportDualVideo = false;

+ mSupportDualVideo = true;

mSupportExtendedMakernote = false;

mSupportFullColorRange = true;

mSupportIPUAcceleration = false;
```

示例redmine: #305756

RK356x-11.0-R6版本SOC类型的camera点不亮问题

问题描述

RK356x-11.0-发布的R6版本, 如果camera3_profiles.xml里面的sensorType配置为SENSOR_TYPE_SOC点不亮, 但是底层直接参考[如何使用v4l2获取底层数据流](#)使用v4l2-ctl工具获取数据流正常。

问题分析

关键报错LOG如下:

```
F DEBUG : #00 pc 00007286
/apex/com.android.vndk.v30/lib/libcamera_metadata.so
F DEBUG : #01 pc 0001e22d
/vendor/lib/android.hardware.camera.provider@2.4-legacy.so
F DEBUG : #02 pc 0001cb75
/vendor/lib/android.hardware.camera.provider@2.4-legacy.so
F DEBUG : #03 pc 00013b71
/vendor/lib/android.hardware.camera.provider@2.4-legacy.so
F DEBUG : #04 pc 00013993
/vendor/lib/android.hardware.camera.provider@2.4-legacy.so
F DEBUG : #05 pc 00002d27
/vendor/lib/hw/android.hardware.camera.provider@2.4-impl.so
F DEBUG : #06 pc 00044119 /apex/com.android.vndk.v30/lib/libhidlbase.so
F DEBUG : #07 pc 000404d5 /apex/com.android.vndk.v30/lib/libhidlbase.so
F DEBUG : #08 pc 00042c9b /apex/com.android.vndk.v30/lib/libhidlbase.so
F DEBUG : #09 pc 0004136d /apex/com.android.vndk.v30/lib/libhidlbase.so
F DEBUG : #10 pc 0003f655 /apex/com.android.vndk.v30/lib/libhidlbase.so
F DEBUG : #11 pc 0003f91b /apex/com.android.vndk.v30/lib/libhidlbase.so
F DEBUG : #12 pc 00001a51
/vendor/bin/hw/android.hardware.camera.provider@2.4-service
F DEBUG : #13 pc 00001981
/vendor/bin/hw/android.hardware.camera.provider@2.4-service
```

跟libcamera_metadata.相关, 查看sdk/hardware/rockchip/camera: 目录

最近跟metadata相关的提交如下:

```
commit 8bb7b62d77462253b1d25352df681c2f706054fb
Author: Jian Qiu <qiu Jian@rock-chips.com>
Date: Fri Mar 12 16:15:06 2021 +0800
```

```
rkisp2: support aiq control vendor tags

Tag for hal:
RK_CONTROL_AIQ_BRIGHTNESS
RK_CONTROL_AIQ_CONTRAST
RK_CONTROL_AIQ_SATURATION

Key for apk:
com.rockchip.control.aiq.brightness
com.rockchip.control.aiq.contrast
com.rockchip.control.aiq.saturation

Signed-off-by: Jian Qiu <qiujian@rock-chips.com>
Change-Id: Iabb418fe3b4ed0b84914a064feab07b3be709336
```

经分析，这个提交只考虑了SENSOR_TYPE_RAW类型的Meta，没有将SOC类型的meta上传给上层；需要修复即可。

解决办法

在 `sdk/hardware/rockchip/camera:` 目录添加如下提交：

```
commit 8e4e7d859df8588a239ec3e6f6fde7a6810c0d63
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date: Tue Apr 6 11:10:34 2021 +0800

    rkisp2: fix soc camrea can't open issue
Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Change-Id: Ia12f85a4bb074f4803cbd1d0463c72a15b891ff9
```

如果没有，请找FAE获取，或者将SDK更新到R7及以上版本就有上述提交了。

RK356x-11.0-更新到R8后SOC-DVP接口的camera用不了

问题描述

更新最新R8版本，平台SOC-DVP接口的camera用不了，报错LOG如下：

```
E RkCamera: <HAL> RKISP2MediaCtlHelper: Getting MediaEntity "stream_cif" failed
E RkCamera: <HAL> RKISP2MediaCtlHelper: Cannot open video node (staitus =
0x80000000)
E RkCamera: <HAL> RKISP2MediaCtlHelper: Failed to open video
```

问题分析

从上面报错LOG看是没有找到stream_cif对应的video结点，导致应用打开失败；

应该是R8的rkcif驱动因为兼容多路VC通道，将video结点名称改了，从stream_cif改成stream_cif_dvp_id0/1/2/3。

解决办法

Kernel添加如下修改:

```
diff --git a/drivers/media/platform/rockchip/cif/capture.c
b/drivers/media/platform/rockchip/cif/capture.c
old mode 100644
new mode 100755
index 83636607e251..501f33652712
--- a/drivers/media/platform/rockchip/cif/capture.c
+++ b/drivers/media/platform/rockchip/cif/capture.c
@@ -3541,7 +3541,7 @@ static int rkCIF_register_stream_vdev(struct rkCIF_stream
*stream,
        } else {
            switch (stream->id) {
                case RKCIF_STREAM_MIPI_ID0:
-                   vdev_name = CIF_DVP_ID0_VDEV_NAME;
+                   vdev_name = CIF_VIDEODEVICE_NAME;
                    break;
                case RKCIF_STREAM_MIPI_ID1:
                    vdev_name = CIF_DVP_ID1_VDEV_NAME;
```

或者SDK/ hardware/rockchip/camera添加如下适配修改: 如没有, 找FAE获取;

```
ccommit 57cc58e5de7a1c8351a19b49ff6bc541ce69709d (HEAD)
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date: Thu Sep 2 11:13:59 2021 +0800

    rkisp2: adapt to rk356x vicap dvp interface new registered name

Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Change-Id: I2c592ea51b7a5be84bad1927e222818c8533f124
```

RK356x-11.0- some iq features are not supported报错

问题描述

```
E rkisp-vir0: some iq features(0xffffffff7fffffff, 0x3fbf7fe67ff) are not
supported

E rkisp-vir0: some iq features(0xffffffff7fffffff, 0x3fbf7fe67ff) are not
supported
```

问题分析

正常不影响;

RK356x-11.0-双摄开关拷机出现卡住黑屏问题

问题描述

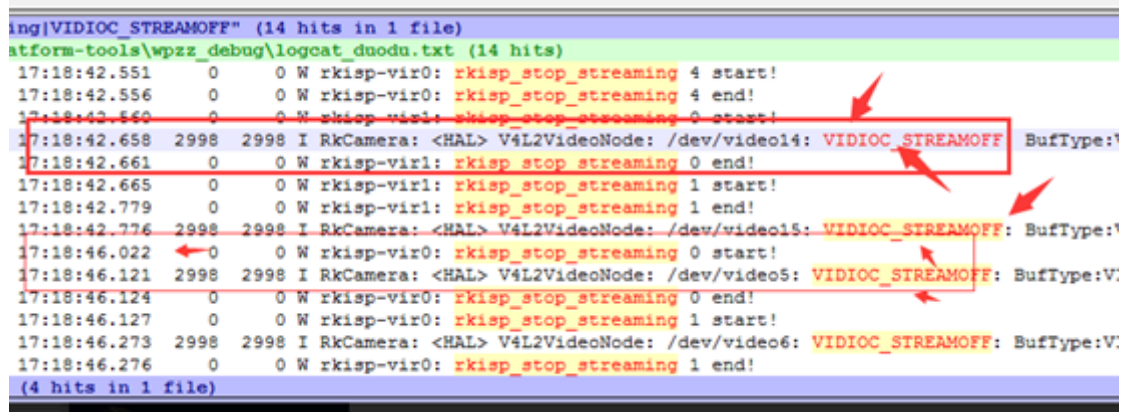
RK356x-11.0-两个摄像头进行开关拷机，很容易出现卡住黑屏问题。

问题分析

分析1:

经反复开关查看问题现象，初步分析是APK关闭后，重新打开前上一次关闭流程还没结束，从而导致异常。这个需要分析关闭流程哪里耗时太多，或者哪里异常，导致关闭太慢。

跟踪LOG发现，HAL层调用一个摄像头的VIDIOC_STREAMOFF后，过了接近4s才调到另外一个摄像头的VIDIOC_STREAMOFF，从而导致退出比较慢；



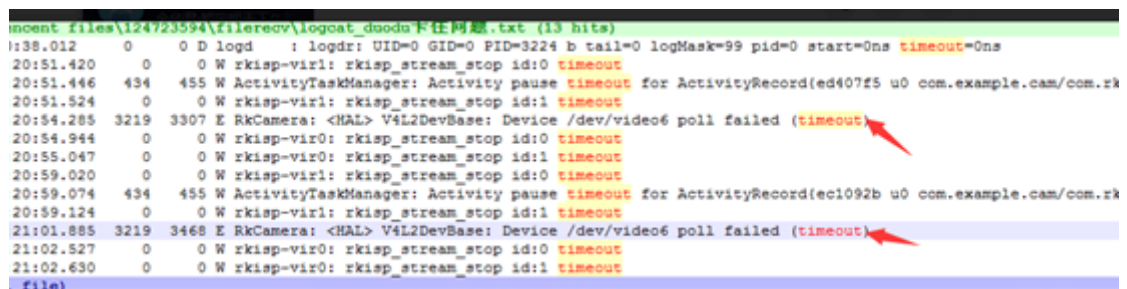
```
ing|VIDIOC_STREAMOFF" (14 hits in 1 file)
atform-tools\wpzz_debug\logcat_duodu.txt (14 hits)
17:18:42.551 0 0 W rkisp-vir0: rkisp_stop_streaming 4 start!
17:18:42.556 0 0 W rkisp-vir0: rkisp_stop_streaming 4 end!
17:18:43.560 0 0 W rkisp-vir1: rkisp_stop_streaming 0 start!
17:18:42.658 2998 2998 I RkCamera: <HAL> V4L2VideoNode: /dev/video14: VIDIOC_STREAMOFF: BufType:
17:18:42.661 0 0 W rkisp-vir1: rkisp_stop_streaming 0 end!
17:18:42.665 0 0 W rkisp-vir1: rkisp_stop_streaming 1 start!
17:18:42.779 0 0 W rkisp-vir1: rkisp_stop_streaming 1 end!
17:18:42.776 2998 2998 I RkCamera: <HAL> V4L2VideoNode: /dev/video15: VIDIOC_STREAMOFF: BufType:
17:18:46.022 0 0 W rkisp-vir0: rkisp_stop_streaming 0 start!
17:18:46.121 2998 2998 I RkCamera: <HAL> V4L2VideoNode: /dev/video5: VIDIOC_STREAMOFF: BufType:V.
17:18:46.124 0 0 W rkisp-vir0: rkisp_stop_streaming 0 end!
17:18:46.127 0 0 W rkisp-vir0: rkisp_stop_streaming 1 start!
17:18:46.273 2998 2998 I RkCamera: <HAL> V4L2VideoNode: /dev/video6: VIDIOC_STREAMOFF: BufType:V.
17:18:46.276 0 0 W rkisp-vir0: rkisp_stop_streaming 1 end!
(4 hits in 1 file)
```

继续分析查看LOG发现，关闭其中一个后，另外一个还在poll数据，会出现poll timeout异常；

关键LOG:

```
E RkCamera: <HAL> V4L2DevBase: Device /dev/video6 poll failed (timeout)
```

```
E RkCamera: <HAL> RKISP2ImguUnit: Device poll failed
```



```
ment files\124723594\filerectv\logcat_duodu卡住问题.txt (13 hits)
17:38.012 0 0 D logd : logdr: UID=0 GID=0 PID=3224 b tail=0 logMask=99 pid=0 start=0ns timeout=0ns
20:51.420 0 0 W rkisp-vir1: rkisp_stream_stop id:0 timeout
20:51.446 434 455 W ActivityTaskManager: Activity pause timeout for ActivityRecord(ed407f5 u0 com.example.cam/com.rk
20:51.524 0 0 W rkisp-vir1: rkisp_stream_stop id:1 timeout
20:54.285 3219 3307 E RkCamera: <HAL> V4L2DevBase: Device /dev/video6 poll failed (timeout)
20:54.944 0 0 W rkisp-vir0: rkisp_stream_stop id:0 timeout
20:55.047 0 0 W rkisp-vir0: rkisp_stream_stop id:1 timeout
20:59.020 0 0 W rkisp-vir1: rkisp_stream_stop id:0 timeout
20:59.074 434 455 W ActivityTaskManager: Activity pause timeout for ActivityRecord(ec1092b u0 com.example.cam/com.rk
20:59.124 0 0 W rkisp-vir1: rkisp_stream_stop id:1 timeout
21:01.885 3219 3468 E RkCamera: <HAL> V4L2DevBase: Device /dev/video6 poll failed (timeout)
21:02.527 0 0 W rkisp-vir0: rkisp_stream_stop id:0 timeout
21:02.630 0 0 W rkisp-vir0: rkisp_stream_stop id:1 timeout
file)
```

而这个timeout超时时间就是3s左右，从而导致卡住；

分析2:

现在就是定位是底层关闭其中一个摄像头，导致另外一个poll timeout异常；

还是上层逻辑不对导致，关闭其中一个摄像头，导致另外一个异常；

需要运行rkaiq_3A_server单独将两路运行起来，然后使用v4l2-ctl工具模拟单独关闭场景，查看是否影响另外一路。模拟场景步骤如下：

1) 先将CameraHal3库中对camera的调用去掉

```
rk3566_rgo:/ # cd /vendor/etc/camera/
```

```
rk3566_rgo:/vendor/etc/camera # mv camera3_profiles.xml camera3_profiles.xml.bak
```

2) 对应sensor运行两个rkaiq_3A_server

```
adb shell "rkaiq_3A_server --sensor_index=0 --hdrmode=HDR2"
adb shell "rkaiq_3A_server --sensor_index=1 --hdrmode=NORMAL"
```

其中: sensor 0是HDR模式的, sensor1是NORMAL模式配置

rkaiq_3A_server 参数配置参考:

sdk/external/camera_engine_rkaiq/hal_interface2.0/rkaiq_3A_server/readme.txt说明

3) 使用v4l2-ctl工具抓取数据流

```
adb shell "v4l2-ctl --verbose -d /dev/video14 --set-fmt-
video=width=1920,height=1080,pixelformat='NV12' \

--stream-mmap=4 --set-
selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080"

adb shell "v4l2-ctl --verbose -d /dev/video5 --set-fmt-
video=width=1920,height=1080,pixelformat='NV12' \

--stream-mmap=4 --set-
selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080"
```

同时打开上面两路sensor的数据流后, 然后再关闭;

尝试多次开关其中一路, 不会影响另外一路, 应该是HAL层poll的函数调用参数哪里不对;

如其他同事分析, 回读模式isp数据流是受aiq驱动 (raw数据走向 mipi->aiq->isp), aiq被关掉isp是没输出的。从log上看, isp1这路关闭, isp0这路mipi tx也被关了, 所以会poll不到数据, 需检查上层逻辑控制。

```
15 D/rkisp-vir0( 0): rkisp_stop_streaming 0
12 D/rkisp-vir0( 0): rkisp_stop_streaming 1
17 D/rkisp-vir0( 0): rkisp_isp_stop refcnt:2
18 D/rkisp-vir0( 0): rkisp_isp_sd s power on:0
17 D/rkisp-vir0( 0): rkisp_isp_sd s power on:1
12 D/rkisp-vir0( 0): rkisp_start_streaming 4
13 D/rkisp-vir0( 0): rkisp_start_streaming 1
18 D/rkisp-vir0( 0): rkisp_isp_start refcnt:2
10 D/rkisp-vir0( 0): rkisp_start_streaming 0
15 D/rkisp-vir0( 0): rkisp_stop_streaming 4
10 D/rkisp-vir1( 0): rkisp_stop_streaming 0
12 D/rkisp-vir1( 0): rkisp_stop_streaming 1
17 D/rkisp-vir1( 0): rkisp_isp_stop refcnt:2
13 D/rkisp-vir1( 0): rkisp_isp_sd s power on:0
16 D/rkisp-vir1( 0): rkisp_isp_sd s power on:1
12 D/rkisp-vir1( 0): rkisp_start_streaming 1
12 D/rkisp-vir1( 0): rkisp_isp_start refcnt:2
12 D/rkisp-vir1( 0): rkisp_start_streaming 0
11 E/RkCamera( 460): <HAL> V4L2DevBase: Device /dev/video6 poll failed (timeout)
18 D/rkisp-vir0( 0): rkisp_start_streaming 0
```

stream 4是mipi tx受控于aiq

分析定位3

将aiq的LOG加上后发现, 确实是关闭其中1个sensor的时候, aiq被释放掉了两次, 即把另外一个sensor对应的aiq服务也释放掉了;

而此时HAL还在处理Flush, 导致后面的poll流程卡住了, 这个才是问题所在。

```

0023 D RkCamera: <HAL> V4L2DevBase: @pollDevices(358) flushFd(136), numFds(1)!
0036 D RkCamera: <HAL> RequestThread: @handleProcessCaptureRequest : mRequestsInHAL :4
0023 D RkCamera: <HAL> V4L2DevBase: @pollDevices(358) flushFd(136), numFds(1)!
0036 D RkCamera: <HAL> RequestThread: @handleProcessCaptureRequest : mRequestsInHAL :4
0023 D RkCamera: <HAL> V4L2DevBase: @pollDevices(358) flushFd(136), numFds(1)!
0036 D RkCamera: <HAL> RequestThread: @handleProcessCaptureRequest : mRequestsInHAL :4
9502 D rkisp : [XCORE]:XCAM DEBUG rkisp_control_loop_impl.cpp:441: -----rkisp_cl_deinit
9502 D rkisp : [XCORE]:XCAM DEBUG rkisp_control_loop_impl.cpp:445: -----rkisp_cl_deinit done
9502 D rkisp : [XCORE]:XCAM DEBUG rkisp_control_loop_impl.cpp:441: -----rkisp_cl_deinit
9502 D rkisp : [XCORE]:XCAM DEBUG rkisp_control_loop_impl.cpp:445: -----rkisp_cl_deinit done
9502 D RequestThread: ENTER-flush
9502 I RkCamera: <HAL> RequestThread: @flush, line:468, mRequestsInHAL:4
0023 D RkCamera: <HAL> V4L2DevBase: @pollDevices(358) flushFd(136), numFds(1)!
9502 I RkCamera: <HAL> RequestThread: @flush, line:479, mRequestsInHAL:0, time spend:1113697us
9502 D RequestThread: EXIT-flush
9502 D RequestThread: ENTER-flush
9502 I RkCamera: <HAL> RequestThread: @flush, line:468, mRequestsInHAL:0
9502 I RkCamera: <HAL> RequestThread: @flush, line:479, mRequestsInHAL:0, time spend:0us
9502 D RequestThread: EXIT-flush
9502 D RequestThread: ENTER-flush
9502 I RkCamera: <HAL> RequestThread: @flush, line:468, mRequestsInHAL:0
9502 I RkCamera: <HAL> RequestThread: @flush, line:479, mRequestsInHAL:0, time spend:0us
9502 D RequestThread: EXIT-flush

```

经定位是如下提交导致：加个了sMap_aiq_ctx的静态变量；

```

commit ecf342a1cbd1d3b120dcf4c486963b320de062e0
Author: Jian Qiu <qiujian@rock-chips.com>
Date: Wed Mar 10 15:20:21 2021 +0800

    rkisp2: support dual camera

Signed-off-by: Jian Qiu <qiujian@rock-chips.com>
Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Change-Id: I61f23e604087b3668968a85bc6c0d682497e080b

```

释放aiq的时候，第一个sensor释放的时候就把两个都同时释放掉了，导致后面另外一个sensor的HAL层poll异常；

```

156
157 status_t RKISP2CtrlLoop::stop()
158 {
159     HAL_TRACE_CALL(CAM_GLBL_DBG_INFO);
160     PERFORMANCE_ATRACE_NAME("RKISP2CtrlLoop::stop");
161
162     if (mIsStarted == false)
163         return OK;
164
165     int ret = 0;
166
167     std::map<int, void*>::const_iterator it = sMap_aiq_ctx.begin();
168     for (; it != sMap_aiq_ctx.end(); ++it) {
169         ret = rkisp_cl_stop(it->second);
170     }
171     if (ret < 0) {
172         LOGE("%s: rkisp control loop stop failed !", __FUNCTION__);
173         return UNKNOWN_ERROR;
174     }

```

解决办法

sdk/hardware/rockchip/camera\$ 目录的如下提交已有解决；

```
commit bdfbca8f27b19e7e47388841a0ab1c725c3fb28e (HEAD)
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date: Mon Jul 26 09:55:36 2021 +0800

    rkisp2: fix dual camera deinit aiq issue

    Currently close one camera will deinit two camera's aiq,
    so fix it

Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Change-Id: Iaf740878c0c822f5717a38025f0940ee00fa0bd8
```

RK356x-11.0-有关双摄更新代码打不开排查流程

1、v4l2-ctl --verbose -d /dev/video0 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --set-selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080

看是否有如下数据流，如果没有，说明vicap代码更新导致出现异常，对vicap代码最近提交二分排查：

```
VIDIOC_QBUF: ok
VIDIOC_STREAMON: ok
idx: 0 seq:      0 bytesused: 3110400 ts: 316.198903
idx: 1 seq:      1 bytesused: 3110400 ts: 316.232269 delta: 33.366 ms
idx: 2 seq:      2 bytesused: 3110400 ts: 316.265570 delta: 33.301 ms
idx: 3 seq:      3 bytesused: 3110400 ts: 316.298910 delta: 33.340 ms
idx: 0 seq:      4 bytesused: 3110400 ts: 316.332237 delta: 33.327 ms fps: 30.00
idx: 1 seq:      5 bytesused: 3110400 ts: 316.365578 delta: 33.341 ms fps: 30.00
idx: 2 seq:      6 bytesused: 3110400 ts: 316.398911 delta: 33.333 ms fps: 30.00
idx: 3 seq:      7 bytesused: 3110400 ts: 316.432243 delta: 33.332 ms fps: 30.00
```

v4l2-ctl --verbose -d /dev/video5 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --set-selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080

看是否有如下数据流，如果没有说明isp代码更新导致出现异常，对isp代码最近提交进行排查：

```
VIDIOC_QBUF: ok
VIDIOC_STREAMON: ok
idx: 0 seq:      0 bytesused: 3110400 ts: 322.022796
idx: 1 seq:      1 bytesused: 3110400 ts: 322.055822 delta: 33.026 ms
idx: 2 seq:      2 bytesused: 3110400 ts: 322.089435 delta: 33.613 ms
idx: 3 seq:      3 bytesused: 3110400 ts: 322.122786 delta: 33.351 ms
idx: 0 seq:      4 bytesused: 3110400 ts: 322.156117 delta: 33.331 ms fps: 30.00
idx: 1 seq:      5 bytesused: 3110400 ts: 322.189384 delta: 33.267 ms fps: 30.01
idx: 2 seq:      6 bytesused: 3110400 ts: 322.222782 delta: 33.398 ms fps: 30.00
idx: 3 seq:      7 bytesused: 3110400 ts: 322.255853 delta: 33.071 ms fps: 30.04
```

2、开一个窗口运行

```
adb shell "rkaiq_3A_server --sensor_index=0 --hdrmode=NORMAL"
```

v4l2-ctl --verbose -d /dev/video14 --set-fmt-video=width=1920,height=1080,pixelformat='NV12' --stream-mmap=4 --set-selection=target=crop,flags=0,top=0,left=0,width=1920,height=1080

如果无数据流说明aiq代码更新导致出现异常，需对aiq代码最近提交进行排查：

RK356x-11.0-更新到R9后SOC-DVP接口的camera点击拍照录像异常问题

示例redmine号：Defect #308090

问题描述

DVP接口摄像头拍照报错如下

```
[ 131.609570] gc2145 2-003c: gc2145_set_fmt enter
[ 131.609916] gc2145 2-003c: gc2145_power(2638) on(1)
[ 131.612166] gc2145 2-003c: gc2145_init(2622)
[ 132.098197] gc2145 2-003c: gc2145_s_stream: on: 1, 800x600@16
[ 132.117679] rkCIF_dvp: set dual edge mode(off,0x2000000)!!!
[ 134.379484] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.440845] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.502196] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.563592] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.624951] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.686308] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.747662] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.809024] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.870450] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.931715] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 134.993093] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
[ 135.054514] rkCIF_dvp: not active buffer, skip current frame, dvp stream[0]
```

问题分析

从LOG看是cif驱动中buf出现异常了，查看最近提交，是将dummy buffer去掉的：

```
commit ea02130e2fd20c27ee0e94408c4224cf7e24be5a
Author: Zefa Chen <zefa.chen@rock-chips.com>
Date:   Fri Jun 18 21:09:34 2021 +0800

    media: rockchip: cif: remove dummy buffer

Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
Change-Id: Ib1cdf85020d37a7a4a06b726fb88597797806411
```

经尝试分析，确实跟这个有关系：

解决办法

kernel这个提交revert掉这个提交即可：

```
commit ea02130e2fd20c27ee0e94408c4224cf7e24be5a
Author: Zefa Chen <zefa.chen@rock-chips.com>
Date:   Fri Jun 18 21:09:34 2021 +0800

    media: rockchip: cif: remove dummy buffer

Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
Change-Id: Ib1cdf85020d37a7a4a06b726fb88597797806411
```

后面R10版本有修复这个提交问题，就不用revert掉了；

RK356x-11.0用2lane+2lane mipi yuv摄像头报错问题

示例redmine号：Defect #316820

问题描述

有款摄像头xc7160配置2lane输出，使用rk3568 2lane+2lane的mipi摄像头配置，走csi_dphy1。有一个问题，如果设备树中使能了rkisp_vir1节点，抓帧会报如下错误LOG：

```
[ 535.856420] rk_iommu fdff1a00.iommu: Enable stall request timed out,
retry_count = 0, status: 0x00022b
[ 535.856522] rk_iommu fdff1a00.iommu: Page fault at 0x00000000fe6a3000 of type
write
[ 535.856567] rk_iommu fdff1a00.iommu: iova = 0x00000000fe6a3000: dte_index:
0x3f9 pte_index: 0x2a3 page_offset: 0x0
[ 535.856603] rk_iommu fdff1a00.iommu: mmu_dte_addr: 0x000000007b3f2000
dte@0x000000007b3f2fe4: 0x1d347001 valid: 1 pte@0x000000001d347a8c: 0x000000
valid: 0 page@0x0000000000000000 flags: 0x0
[ 535.856628] rk_iommu fdff1a00.iommu: Page fault while iommu not attached to
domain?
[ 535.856667] rk_iommu fdff1a00.iommu: Page fault at 0x00000000fe6a3000 of type
write
[ 535.856726] rk_iommu fdff1a00.iommu: iova = 0x00000000fe6a3000: dte_index:
0x3f9 pte_index: 0x2a3 page_offset: 0x0
[ 535.856782] rk_iommu fdff1a00.iommu: mmu_dte_addr: 0x000000007b3f2000
dte@0x000000007b3f2fe4: 0x1d347001 valid: 1 pte@0x000000001d347a8c: 0x000000
valid: 0 page@0x0000000000000000 flags: 0x0
[ 535.856825] rk_iommu fdff1a00.iommu: Page fault while iommu not attached to
domain?
[ 535.857596] rk_iommu fdff1a00.iommu: Disable paging request timed out,
retry_count = 0, status: 0x00022b
```

问题分析

分析说明：

isp处理多路sensor，目前只支持多路raw

yuv sensor可以用vicap接收

rk356x isp可以接收单路yuv，多路不行。

解决办法

2lane+2lane的双目摄像头都是yuv情况：需要isp和vicap(也叫cif)各接收一个。

如果一个raw格式一个yuv的，raw格式的由isp接收，yuv格式的vicap接收；

RK356x-11.0-sensor->csi2_dphy2->mipi_csi2->vicap->isp_vir1链路兼容多个sensor问题

示例redmine号：Defect #336306

问题描述

客户需求: sensor->csi2_dphy2->mipi_csi2->vicap->isp_vir1这条配置链路, 兼容多个sensor;

即sensor0/sensor1 -> csi2_dphy2根据注册的情况

问题分析

对比cif只配置一个sensor和2个sensor的 media情况,

```
\- entity 67: rkCIF_mipi_lvds (1 pad, 1 link)

type V4L2 subdev subtype Unknown

device node name /dev/v4l-subdev5

pad0: Source

[fmt:SBGGR10/1600x1200]

-> "rkisp-isp-subdev":0 [ENABLED]
```

2个sensor比1个sensor的media2下, 少了rkCIF_mipi_lvds这个entity导致分辨率相关参数获取失败, 从而导致异常;

解决办法

如下修改验证OK

```
diff --git a/drivers/media/platform/rockchip/cif/dev.c
b/drivers/media/platform/rockchip/cif/dev.c
old mode 100644
new mode 100755
index 33310e0bb9be..747c625738fc
--- a/drivers/media/platform/rockchip/cif/dev.c
+++ b/drivers/media/platform/rockchip/cif/dev.c
@@ -1238,8 +1238,8 @@ static int __maybe_unused __rkCIF_clr_unready_dev(void)
     mutex_lock(&rkCIF_dev_mutex);

    list_for_each_entry(cif_dev, &rkCIF_device_list, list) {
-        subdev_asyn_register_itf(cif_dev);
        v4l2_async_notifier_clr_unready_dev(&cif_dev->notifier);
+        subdev_asyn_register_itf(cif_dev);
    }

    mutex_unlock(&rkCIF_dev_mutex);
```

对应kernel下的提交点如下: 如果不包含, 手动按照上面修改;

```
commit 2ecefbd713f228b315bfe4f985f2c5b35bcb760f (HEAD)
Author: Zefa Chen <zefa.chen@rock-chips.com>
Date: Wed Nov 24 19:48:38 2021 +0800

    media: rockchip: cif clean unready subdev before async subdev-itf

Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
Change-Id: I2049f83a9d552109acca86d37d81ef0ceb2e9687
```

RK3326-11.0-raw-sensor打开闪光灯拍照卡住问题

示例redmine号: Defect #337083

问题描述

rk3326/rk3326s 按照之前10.0系统配置闪光灯现在打不开, 图标有了, 打开闪光灯拍照会卡住, 单独去控制闪光灯的gpio口强制拉高是可以亮的, 使用v4l2-ctl命令也可以正常控制三种模式; 将camera3_profiles.xml中sensorType改成SENSOR_TYPE_SOC也可以正常打开闪光灯拍照; 但配置SENSOR_TYPE_RAW, 打开闪光灯拍照就卡住, 录像预览打开闪光灯也不亮;

问题分析

初步分析因为v4l2-ctl可以正常控制, 说明底层驱动初步正常; soc类型正常, raw类型异常, 说明是rkisp库这边逻辑可能出问题了;

查找关键异常LOG如下:

```
E rkisp : [XCORE]:XCAM ERROR isp_controller.cpp:1456: set fl
V4L2_CID_FLASH_TORCH_INTENSITY to 4000 failed
```

说明是isp_controller.cpp文件设置V4L2_CID_FLASH_TORCH_INTENSITY模式失败, 跟踪代码发现set_fl_contol_to_dev失败后, 直接返回, 没有继续下面流程导致流程异常; 所以录像模式灯也不亮; set_fl_contol_to_dev(fl_device, V4L2_CID_FLASH_LED_MODE, V4L2_FLASH_LED_MODE_TORCH);

问题原因:

10.0可以, 是因为10.0的闪光灯驱动版本比较旧, 支持设置上面报错模式;

11.0的驱动, 闪光灯驱动有新提交修改东西, 如下提交: 导致上面几个模式只读, 无法设置;

```
commit 46cff32208681605f109f9d22ebc91973cd1f763
Author: Zefa Chen <zefa.chen@rock-chips.com>
Date: Thu Jun 4 20:22:36 2020 +0800

    leds: leds-rgb13h add pwm control

Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
Change-Id: Id8800dc98543d4e082182a027f3fb25dbd30295c
```

```

ctrl 0x%x not supported\n", ctrl->id);
@@ -358,18 +410,24 @@ static int rgb13h_init_controls(struct rgb13h_led *led)
    led->max_flash_tm;
    led->timeout = led->max_flash_tm;
    /* V4L2_CID_FLASH_INTENSITY */
    v4l2_ctrl_new_std(&led->ctrls, &rgb13h_ctrl_ops,
+   ctrl = v4l2_ctrl_new_std(&led->ctrls, &rgb13h_ctrl_ops,
        V4L2_CID_FLASH_INTENSITY, 0,
        led->max_flash_current,
        led->max_flash_current,
        1,
        led->max_flash_current);
+   if (ctrl && IS_ERR(ctrl)) {
+       ctrl->flags |= V4L2_CTRL_FLAG_READ_ONLY;
+   }
    led->intensity = led->max_flash_current;
    /* V4L2_CID_FLASH_TORCH_INTENSITY */
    v4l2_ctrl_new_std(&led->ctrls, &rgb13h_ctrl_ops,
+   ctrl = v4l2_ctrl_new_std(&led->ctrls, &rgb13h_ctrl_ops,
        V4L2_CID_FLASH_TORCH_INTENSITY, 0,
        led->max_mm_current,
        led->max_mm_current,
        1,
        led->max_mm_current);
-   led->intensity = led->max_mm_current;
+   if (ctrl && IS_ERR(ctrl)) {
+       ctrl->flags |= V4L2_CTRL_FLAG_READ_ONLY;
+   }
    led->intensity_torch = led->max_mm_current;

```

解决办法

按照如下修改验证OK

sdk/hardware/rockchip/camera_engine_rkisp目录

```

diff --git a/modules/isp/isp_controller.cpp b/modules/isp/isp_controller.cpp
index f93ca14..e2c6c83 100755
--- a/modules/isp/isp_controller.cpp
+++ b/modules/isp/isp_controller.cpp
@@ -1393,8 +1393,7 @@ IspController::set_3a_fl (int fl_mode, float
fl_intensity[],
    control.id = control_id; \
    control.value = val; \
    if (fl_dev->io_control (VIDIOC_S_CTRL, &control) < 0) { \
-       XCAM_LOG_ERROR (" set fl %s to %d failed", #control_id, val); \
-       return XCAM_RETURN_ERROR_IOCTL; \
+       XCAM_LOG_WARNING (" set fl %s to %d failed, maybe not
supported", #control_id, val); \
    } \
    XCAM_LOG_DEBUG (" sof seq %d, set fl %p, cid %s to %d, success", \
        _frame_sequence, fl_dev.ptr(), #control_id, val); \

```

这样保证一些只读模式设置失败后也可以正常走后续流程;

RK3326-11.0-gc02m2切到gc032a概率报错

示例Defect #346316

问题描述

旋转rk3326 android11 gc032a+gc02m2 摄像头，生产发现部分机器从gc02m2 切换到gc032a时报错，测试发现跟着板子走

问题分析

报错LOG:

```
E rkispl : CIF_ISP_PIC_SIZE_ERROR (0x00000001)
E rkispl : CIF_ISP_PIC_SIZE_ERROR (0x00000000)
E rkispl : CIF_ISP_PIC_SIZE_ERROR (0x00000001)
E rkispl : CIF_ISP_PIC_SIZE_ERROR (0x00000000)
E rkispl : CIF_ISP_PIC_SIZE_ERROR (0x00000001)
```

分析可能原因:

- 1、gc032a前期输出异常到rkisp报错;
- 2、gc02m2没有正确关闭,影响了rkisp接收gc032a信号
- 3、部分板子干扰较大, gc032a驱动强度不够, 导致接收异常;
- 4、未正确配置电源域, 导致mclk赋值不对;

调试修改方法

- 1、核对下powerdown引脚, 测量打开gc032a时候gc02m2的信号是否已经关闭;
- 2、核对硬件原理图, 查看电源域: `vccio3-supply = <&vcc2v8_dvp>`;是否配置正确;
- 3、看如下提交是否有, 没有参考修改上

```
commit 34913157111b3a8ae1631daeb2d857959ec299cf
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date: Tue Aug 3 11:48:45 2021 +0800

    media: i2c: gc032a: set default stream off state

    Set sensor in stream off state by default,
    to avoid sending abnormal data in the early stage.

    Signed-off-by: Wang Panzhenzhuan <randy.wang@rock-chips.com>
    Change-Id: Iaae2fd3be95a5fcac05ebaf39a68614b7b807d4a
```

- 4、找GC原厂询问看是否能提高gc032a驱动强度;
- 5、把DVP相关的IO默认拉低:

```
&dvp_d2d9_m0 {
    rockchip,pins =
        <2 RK_PA0 1 &pcfg_pull_down>,/* cif_data2 */
        <2 RK_PA1 1 &pcfg_pull_down>,/* cif_data3 */
        <2 RK_PA2 1 &pcfg_pull_down>,/* cif_data4 */
        <2 RK_PA3 1 &pcfg_pull_down>,/* cif_data5 */
        <2 RK_PA4 1 &pcfg_pull_down>,/* cif_data6 */
        <2 RK_PA5 1 &pcfg_pull_down>,/* cif_data7 */
        <2 RK_PA6 1 &pcfg_pull_down>,/* cif_data8 */
        <2 RK_PA7 1 &pcfg_pull_down>,/* cif_data9 */
        <2 RK_PB0 1 &pcfg_pull_down>,/* cif_sync */
        <2 RK_PB1 1 &pcfg_pull_down>,/* cif_href */
        <2 RK_PB2 1 &pcfg_pull_down>,/* cif_clkkin */
        <2 RK_PB3 1 &pcfg_pull_none_12ma>;/* cif_clkout */
};
```

解决办法

按照5、把DVP相关的IO默认拉低就可以了；说明这个问题原因为DVP引脚初始的电平可能被sensor端影响，导致起始异常信号被错当做像素信号检测了，从而导致pic size error。

Android12.0调试案例

RK3566-12.0-更新贵司SDK后，后摄预览不停的对焦

(Defect #366374);

解决办法

sdk/external/camera_engine_rkaiq\$ 目录:

参考这个提交，自行对比修改下iq文件，主要是因为af结构体有修改，需同步更新iq:

```
commit 9166b0821597eb4416b745a6cefbbe7c000f18ed (tag: android-12.1-mid-rkr8, tag: android-12.1-mid-rkr7)
Author: Wang Panzhenzhuan <randy.wang@rock-chips.com>
Date: Mon Apr 11 11:12:07 2022 +0800

    iqfile: isp21: ov8858_HS5885-BNSM1018-V01_default.json fix af issue

    fix the auto focus cannot be fixed all the time issue

Signed-off-by: Hu Kejun <william.hu@rock-chips.com>
Change-Id: Ibadad246c6777bf4f565cbab4d78ec15e8631f5
```

RK3588-12.0-R10版本拍照报错重启问题

示例: Defect #372974

解决办法

sdk/hardware/rockchip/libhwjpeg\$ 目录加这个两个提交

提交1:

```
commit 0ecd3d456000d47eccc28063d98c2d5055b9a983 (tag: android-12.1-mid-rkr11, rk/master, m/rk35/mid/12.0/develop)
Author: Chen Jinsen <kevin.chen@rock-chips.com>
Date: Fri Jul 15 14:18:56 2022 +0800

    [MpiJpegEncoder]: Fix RGA blit page fault issue

    Alloc enough buffer size to make sure the scene crop twice

Signed-off-by: Chen Jinsen <kevin.chen@rock-chips.com>
Change-Id: Ibba4e85edaff558e980449f7ce2290a85dcdd9e3
```

提交2:

```
commit f2e2666da5401895db3a9efab9cbca011b11a9f2 (HEAD)
```

```
Author: Chen Jinsen <kevin.chen@rock-chips.com>
```

```
Date: Sat Aug 13 16:31:01 2022 +0800
```

```
[MpiJpegEncoder]: add support for import buffer
```

```
Avoid rga2 accessing a physical address greater than 32 bits
```

```
Signed-off-by: Chen Jinsen <kevin.chen@rock-chips.com>
```

```
Change-Id: Ic0353e2e85d0a7437c7e3765c678c7e08ca177fd
```

RK3588-12.0-R10版本预览手指晃动，会在运动区域出现红黄蓝的条状物

示例：Defect #391525

问题描述

示例现象如下图：



解决办法

Kernel所需提交：

```
From b15006636a0f2cf1e3eff8e11dd9687b24e18275 Mon Sep 17 00:00:00 2001
```

```
From: Cai YiWei <cyw@rock-chips.com>
```

```
Date: Tue, 26 Jul 2022 10:39:17 +0800
```

```
Subject: [PATCH] media: rockchip: isp: fix error of lsc repeat switch
```

```
Change-Id: I8b7948ad784c34d05a2495f9dde535cb4a265d7f
```

```
Signed-off-by: Cai YiWei <cyw@rock-chips.com>
```

sdk/external/camera_engine_rkaiq\$目录提交

```
From 02c775a9b2d21b30c3e5ccab746a875ae9bcb7e4 Mon Sep 17 00:00:00 2001
From: WuQiang <xianlee.wu@rock-chips.com>
Date: Wed, 8 Jun 2022 07:49:03 +0000
Subject: [PATCH] hwi: update lsc only when status change.

Change-Id: Idb11b43879ca543ca1b2ff44edd3fa8f50596e79
Signed-off-by: WuQiang <xianlee.wu@rock-chips.com>
(cherry picked from commit 48b0e696812df5b539f36596f845dd7495f89c41)
Signed-off-by: ZhongYichong <zyc@rock-chips.com>

Change-Id: Ia879610d32c79ae11657f36724073e229c6b0a63
```

Android-12.0-版本HDR出现休眠唤醒sensor打开报错问题

解决办法

Kernel所需提交:

```
commit 933af6b20cb4e050ccf2ccbafa1608634b6e7564
Author: Jianwei Fan <jianwei.fan@rock-chips.com>
Date: Wed Nov 16 07:33:30 2022 +0000

    media: i2c: sensors: fix pm_runtime_put issue in HDR mode

Change-Id: I11189e30eeb5ac2b7073888329e819a9a7f6b074
Signed-off-by: Jianwei Fan <jianwei.fan@rock-chips.com>
```

RK3588S-android 12-R12版本前摄休眠唤醒后预览闪烁

示例redmine Defect #408163

问题描述

前摄使用的是OV5648，已经请贵司调试过tunning

目前开机后不休眠的情况下打开相机预览拍照均正常

机器深度休眠再唤醒后，打开前摄预览一直闪烁，拍出来的照片是正常的

问题分析

分析可能原因:

1、OV5648休眠唤醒后供电不稳定，导致寄存器没写对

试验方法：将OV5648寄存器重新读取对比；

2、OV5648没下电，就重新上电导致异常

添加打印跟踪看下没下电原因；

解决办法

定位原因是休眠close的时候先将pipeline链路清除了，导致vicap调用下电时候没有调用到sensor相关下电流程，vicap调用的是 V4L2 pipeline pm put下电，但是链路在断开的时候，会下电不成功，必须是link的时候，才能正常上下电；所以在close的时候，不去清链路才行；

sdk/hardware/rockchip/camera: 目录添加如下修改:

```
diff --git a/psl/rkisp2/RKISP2MediaCtlHelper.cpp
b/psl/rkisp2/RKISP2MediaCtlHelper.cpp
index 91b0481..d878394 100755
--- a/psl/rkisp2/RKISP2MediaCtlHelper.cpp
+++ b/psl/rkisp2/RKISP2MediaCtlHelper.cpp
@@ -52,8 +52,8 @@
RKISP2MediaCtlHelper::RKISP2MediaCtlHelper(std::shared_ptr<MediaController> sens
RKISP2MediaCtlHelper::~RKISP2MediaCtlHelper()
{
    closeVideoNodes();
-    resetLinks(&mConfigedMediaCtlConfigs[RKISP2IStreamConfigProvider::CIO2]);
-
+    resetLinks(&mConfigedMediaCtlConfigs[RKISP2IStreamConfigProvider::IMGU_COMMON]);
+    // resetLinks(&mConfigedMediaCtlConfigs[RKISP2IStreamConfigProvider::CIO2]);
+    //
+    resetLinks(&mConfigedMediaCtlConfigs[RKISP2IStreamConfigProvider::IMGU_COMMON]);
}

void RKISP2MediaCtlHelper::getConfigedHwPathSize(const char* pathName, uint32_t
&size) {
--
```

RK3588-12.0-R12版本进入相机预览会闪一下绿屏问题

示例redmine Defect #407056

问题描述

在打开相机预览时，进入画面前会闪一下绿屏，之后正常。尝试跳过前面5帧，修改白平衡现象还是存在；

问题分析

试验烧写R10版本boot.img正常，先用二分法找到异常修改；

解决办法

如下提交解决，如没有找FAE获取；

```
commit caf146538fd8602d16f905a84a24f807ba978081
Author: Cai YiWei <cyw@rock-chips.com>
Date: Wed Nov 23 17:41:52 2022 +0800

    media: rockchip: isp: fix self update config for isp30

Change-Id: I882affcf17ab0dc6e6497ea1fb1e9683ff88c3c9
Signed-off-by: Cai YiWei <cyw@rock-chips.com>
```

RK3588-12.0-R12版本摄像头imx415预览分层问题

示例redmine Defect #409087

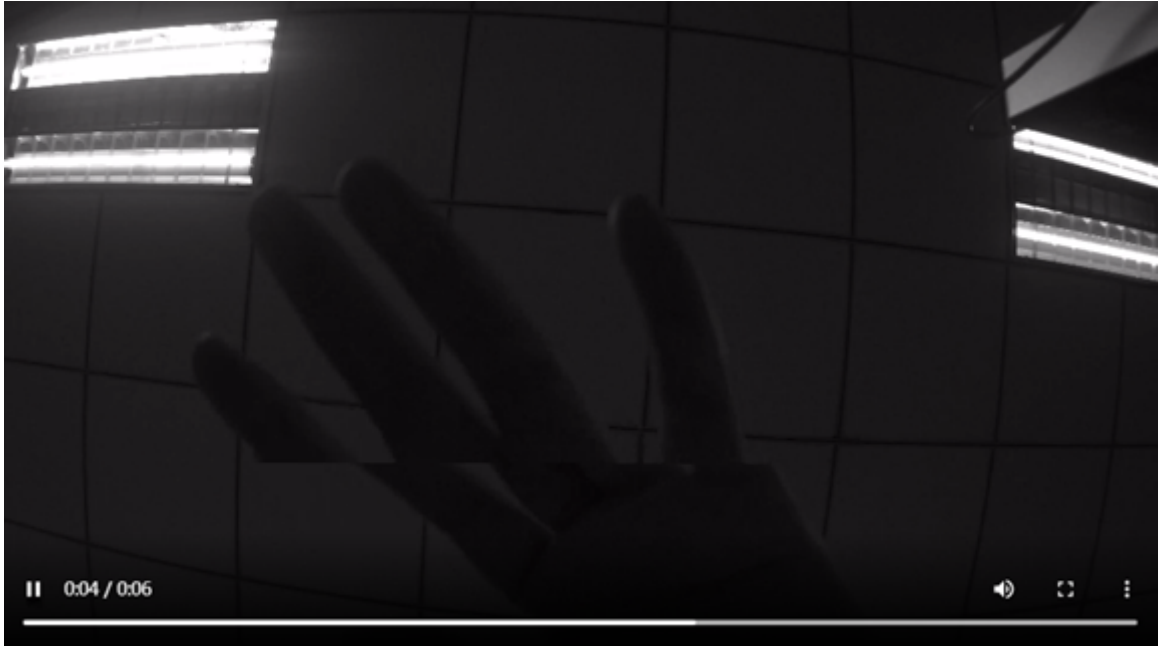
问题描述

驱动30帧+原生效果文件，不分层

驱动60帧+原生效果文件，分层

驱动60帧+修改后的效果文件，分层

分层如图：



问题分析

尝试试验，先将sensor type改成SOC类型看是否没有分层现象，排除aiq影响；

经客户反馈，改成SOC类型后没有分层现象，说明是效果相关参数更新导致，经调试沟通发现，需要如下提交才行。

解决办法

如下提交解决，如没有找FAE获取；

```
commit caf146538fd8602d16f905a84a24f807ba978081
Author: Cai YiWei <cyw@rock-chips.com>
Date:   Wed Nov 23 17:41:52 2022 +0800

    media: rockchip: isp: fix self update config for isp30

Change-Id: I882affcf17ab0dc6e6497ea1fb1e9683ff88c3c9
Signed-off-by: Cai YiWei <cyw@rock-chips.com>
```

RK3588-12.0-R12版本修改selfpath裁图的起始位置不生效问题

示例redmine Defect #412473

问题描述

CameraHal层设置从（0，0）的开始裁减图像，cat /proc/rkisp*信息显示从(0,720)位置截取；

```
Output rkisp_selfpath Format:NV12 Size:1920x1080 Dcrop(0,720|3840x2160)
(frame:701 rate:30ms delay:15ms frameloss:2)
```

问题分析

打开CameraHal的LOG，查看配置的selection参数是正确的，但是到isp驱动中确异常了；分析是isp驱动的selfpath相关配置异常；

解决办法

查看kernel是否有如下提交，如没有找FAE获取；

```
From d1c880c697f24f2d2e0008fe1033768c3d4aab49 Mon Sep 17 00:00:00 2001
From: Cai YiWei <cyw@rock-chips.com>
Date: Thu, 1 Sep 2022 09:35:14 +0800
Subject: [PATCH] media: rockchip: isp: fix limit of dcrop

Change-Id: Id378a98d65bc3d3d39b1f21387e6c94822aedbcd
Signed-off-by: Cai YiWei <cyw@rock-chips.com>
```

RK3588-12.0-R12版本多摄打开，关闭其中一个，其他所有都卡住问题

示例redmine Defect #414870

问题描述

多摄同时打开，关闭其中一个，其他所有都卡住问题，librkaiq库崩溃

LOG如下：

```
005-15 07:52:43.950 9693 9693 F DEBUG : Cmdline:
/vendor/bin/hw/android.hardware.camera.provider@2.4-service
05-15 07:52:43.950 9693 9693 F DEBUG : pid: 9244, tid: 9674, name: xc:g-
8:41001 >>> /vendor/bin/hw/android.hardware.camera.provider@2.4-service <<<
05-15 07:52:43.950 9693 9693 F DEBUG : uid: 1047
05-15 07:52:43.950 9693 9693 F DEBUG : signal 11 (SIGSEGV), code 1
(SEGV_MAPERR), fault addr 0x0
05-15 07:52:43.950 9693 9693 F DEBUG : Cause: null pointer dereference
05-15 07:52:43.950 9693 9693 F DEBUG : r0 00000000 r1 00000000 r2
00000000 r3 4423685a
05-15 07:52:43.950 9693 9693 F DEBUG : r4 e9b4b830 r5 e9b4e55c r6
e9d48514 r7 b86d9f8c
05-15 07:52:43.950 9693 9693 F DEBUG : r8 f2943790 r9 f39419e0 r10
f5c99138 r11 e9b4efd4
05-15 07:52:43.950 9693 9693 F DEBUG : ip f3931910 sp b86d9f38 lr
f37f3d07 pc f3716c02
```

```

05-15 07:52:43.950 9693 9693 F DEBUG : backtrace:
05-15 07:52:43.950 9693 9693 F DEBUG : #00 pc 00187c02
/vendor/lib/librkaiq.so (interpolation(float const*, float const*, int, float,
float*)+2) (BuildId: ce5060cfa6250e920b95f5c9ac54058f)
05-15 07:52:43.950 9693 9693 F DEBUG : #01 pc 00264d03
/vendor/lib/librkaiq.so (AwbPreProcV201(rk_aiq_awb_stat_res_v201_s,
awb_contex_s*)+502) (BuildId: ce5060cfa6250e920b95f5c9ac54058f)
05-15 07:52:43.950 9693 9693 F DEBUG : #02 pc 003b09dc
/vendor/lib/librkaiq.so!librkaiq.so (BuildId: ce5060cfa6250e920b95f5c9ac54058f)

```

问题分析

尝试使用server调用AIQ排除hal影响，结果还是一样；说明跟CameraHal无关；

需排查librkaiq相关信息；

```

sdk/external/camera_engine_rkaiq/uAPI/rk_aiq_user_api_sysctl.cpp
rk_aiq_uapi_sysctl_deinit_locked 中有没去调用:
RkAiQCalibDbV2::releaseCalibDbProj(); 或者
rk_aiq_deinit_lib

```

有的话注释掉；

问题原因为关闭多摄中一个相机，aiq会将其他相机的iq文件都释放掉了，导致异常。

解决办法

sdk/external/camera_engine_rkaiq\$需添加如下提交，没有找FAE获取

```

commit d6a06ef5d552e75a89e975b551ac17daa76b624f (HEAD)
Author: Xu Hongfei <xuhf@rock-chips.com>
Date: Fri Oct 21 14:20:58 2022 +0800

    release v3.0x8.8-p3

    9c47bd5 aiq_core: keep same iso for Android
    Rockchip_Trouble_Shooting_CameraHAL3_CN taken when resolution changed
    5b995d3 AE: fix reg2float accuracy problem
    bd33719 sysctl: should not release all sensor's calibs when deinit
    ddl606b update the file of j2s_generated_v*.h
    1e061bb Adrc, amerge and adehaze: use RK_AIQ_ALGO_CONFTYPE_CHANGERES for
capture
    aa94164 algos: alsc: apply lsc otp in processing
    a2af394 awb otp : apply otp in awb stat
    c1296a2 awb: translator awb stat based on pre_wbgain_inv_r and bls1
    97fb30b tuning: fix double free.

Signed-off-by: Xu Hongfei <xuhf@rock-chips.com>
Change-Id: I51f60de350031e2dab542f0bfd287e69bc2c2a47

```

及如下修改：

```

diff --git a/uAPI/rk_aiq_user_api_sysctl.cpp b/uAPI/rk_aiq_user_api_sysctl.cpp
index 9e96991c..a2d81706 100755
--- a/uAPI/rk_aiq_user_api_sysctl.cpp

```

```

+++ b/uAPI/rk_aiq_user_api_sysctl.cpp
@@ -465,10 +465,6 @@ rk_aiq_uapi_sysctl_deinit_locked(rk_aiq_sys_ctx_t* ctx)
    ctx->_rkAiqManager.release();
    ctx->_camHw.release();
    if (ctx->_calibDbProj) {
-       // TODO:public common resource release
-       if (ctx->cam_type == RK_AIQ_CAM_TYPE_SINGLE) {
-           RkAiqCalibDbV2::releaseCalibDbProj();
-       }
    }

    if (ctx->next_ctx) {

```

RK3588-12.0-R12版本-GMS-Camera VTS失败问题

示例redmine Defect #417409

问题描述

VTS测试时出现如下测试问题，还请帮忙看下。附件为测试logs、测试报告

```

E/GTestResultParser: Name for current test does not match test we started with,
expected: configureStreamsInvalidOutputs/1_legacy_0 but got:
configureStreamsInvalidOutputs/1_legacy_0, where GetParam() = "legacy/0"

05-29 10:54:25 I/ModuleListener: [1/1] HLDA6IXB1B
VtsHalCameraProviderV2_4TargetTest
PerInstance/CameraHidlTest#configureStreamsInvalidOutputs/1_legacy_0 FAILURE:
hardware/interfaces/camera/provider/2.4/vts/functional/VtsHalCameraProviderV2_4Ta
rgetTest.cpp:3638: Failure

Value of: ret.isOk()

Actual: false

Expected: true

```

有如下库崩溃：

报错LOG：

```

/vendor/lib/camera.device@3.3-impl.so
(android::hardware::camera::device::V3_3::implementation::CameraDeviceSession::co
nfigureStreams_3_3(android::hardware::camera::de
/vendor/lib/camera.device@3.3-impl.so
(android::hardware::camera::device::V3_3::implementation::CameraDeviceSession::Tr
ampolineSessionInterface_3_3::configureStreams_3
/vendor/lib/android.hardware.camera.device@3.3.so
(android::hardware::camera::device::V3_3::BnHwCameraDeviceSession::_hidl_configur
eStreams_3_3(android::hidl::base::V1
/vendor/lib/android.hardware.camera.device@3.3.so
(android::hardware::camera::device::V3_3::BnHwCameraDeviceSession::onTransact(uns
igned int, android::hardware::Parcel

```

解决办法

sdk/hardware/interfaces/ 参考如下修改:

```
diff --git a/camera/device/3.2/default/CameraDeviceSession.cpp
b/camera/device/3.2/default/CameraDeviceSession.cpp
index 97f402c2a..734d7a7f0 100755
--- a/camera/device/3.2/default/CameraDeviceSession.cpp
+++ b/camera/device/3.2/default/CameraDeviceSession.cpp
@@ -1092,8 +1092,10 @@ Return<void> CameraDeviceSession::configureStreams(
    mFirstRequest = true;
}

-   for (size_t i = 0; i < requestedConfiguration.streams.size(); i++) {
-       outStreams.streams[i].producerUsage |= RK_GRALLOC_USAGE_RANGE_FULL |
RK_GRALLOC_USAGE_YUV_COLOR_SPACE_BT601;
+   if (ret == OK) {
+       for (size_t i = 0; i < requestedConfiguration.streams.size(); i++) {
+           outStreams.streams[i].producerUsage |= RK_GRALLOC_USAGE_RANGE_FULL |
RK_GRALLOC_USAGE_YUV_COLOR_SPACE_BT601;
+       }
+   }

    _hidl_cb(status, outStreams);
diff --git a/camera/device/3.3/default/CameraDeviceSession.cpp
b/camera/device/3.3/default/CameraDeviceSession.cpp
index 44998f31b..7efd792d1 100755
--- a/camera/device/3.3/default/CameraDeviceSession.cpp
+++ b/camera/device/3.3/default/CameraDeviceSession.cpp
@@ -106,10 +106,11 @@ Return<void> CameraDeviceSession::configureStreams_3_3(
}

//outStreams.streams.resize(requestedConfiguration.streams.size());
-
-   for (size_t i = 0; i < requestedConfiguration.streams.size(); i++) {
-       outStreams.streams[i].v3_2.producerUsage |= RK_GRALLOC_USAGE_RANGE_FULL
| RK_GRALLOC_USAGE_YUV_COLOR_SPACE_BT601;
-       ALOGD(" @s(%d), overrideUsage add RANGE_FULL&COLOR_SPACE_BT601!\n.",
__FUNCTION__, __LINE__);
+   if (ret == OK) {
+       for (size_t i = 0; i < requestedConfiguration.streams.size(); i++) {
+           outStreams.streams[i].v3_2.producerUsage |=
RK_GRALLOC_USAGE_RANGE_FULL | RK_GRALLOC_USAGE_YUV_COLOR_SPACE_BT601;
+           ALOGD(" @s(%d), overrideUsage add
RANGE_FULL&COLOR_SPACE_BT601!\n.", __FUNCTION__, __LINE__);
+       }
+   }

    _hidl_cb(status, outStreams);
```

RK3588-12.0-R12-系统相机打开正常，微信前置相机拍照预览很黑

问题描述

示例redmine: Defect #428372

相机在相机应用打开是正常的，微信视频通话和其他第三方相机也正常，但是微信拍照的时候使用前摄拍照看预览效果非常的黑。

问题分析

图像亮暗，跟AE相关，因为系统相机正常，说明rkaiq应该是正常工作，而微信拍照预览异常，说明是Android设置参数差异导致异常，查看camera3_profiles.xml配置是否正确；

关键字段: `control.aeAvailableModes`

参考 `sdk/system/media/camera/docs/docs.html` 中`android.control.aeAvailableModes`规定：

所有设备 `control.aeAvailableModes` 需要支持ON模式；如果支持手动控制则支持OFF模式；

解决办法

camera3_profiles.xml中如下关键字修改：

```
<control.aeAvailableModes value="OFF"/>
```

需要改成：

```
<control.aeAvailableModes value="ON,OFF"/>
```

Android13.0调试案例

RK3562-13.0-R5版本-CTSV-R4 ITS-scene4和scene6曝光严重导致fail问题

问题描述

camera_scene4 和scene6曝光严重导致fail，摄像头模组02m1+8858

解决办法

sdk/external/camera_engine_rkaiq\$目录需要如下修改：

```
--- a/hal_interface2.0/interface/AiqCameraHalAdapter.cpp
+++ b/hal_interface2.0/interface/AiqCameraHalAdapter.cpp
@@ -481,8 +481,8 @@ AiqCameraHalAdapter::updateAeMetaParams(XCamAeParam
 *aeParams, int reqId){
     uint16_t w_sum = (w_x_end - w_x + 1) * (w_y_end - w_y + 1);

    // stExpSwAttr.metering_mode = aeParams->metering_mode
-   if (aeParams->window.x_end - aeParams->window.x_start > 0) {
-       LOGD(
+   if (aeParams->window.x_end - aeParams->window.x_start > 0 && (w_sum < 225))
+   {
+       LOGE(
        "%s: Update AE ROI weight = %d WinIndex: x:%d, y:%d, x end:%d, y "
        "end:%d, win_sum:%d\n",
        __FUNCTION__, aeParams->window.weight, w_x, w_y, w_x_end, w_y_end,
```

RK3562-13.0-R5版本-vicap接两路4lane-mipi，打开一路后再打开一路出现mipi-csi2 ERR2:0x10000000报错

问题描述

vicap接两路4lane-mipi，打开一路后再打开一路出现mipi-csi2 ERR2:0x10000000报错；

解决办法

sdk/kernel-5.10目录查看是否有如下提交：如果没有找FAE获取

```
commit d48ecfc5f703b5c764f15829202cf6131ea51d76
Author: Zefa Chen <zefa.chen@rock-chips.com>
Date:   Fri May 19 09:35:01 2023 +0800

    phy: rockchip: mipi csi2 dphy fixes grf write error of val for lane select

Signed-off-by: Zefa Chen <zefa.chen@rock-chips.com>
Change-Id: Id56eb83d9a25c15b9d611232e6f82221a55c7e64
```

RK3562-13.0-R5版本-vicap出现Failed to create dummy_buf问题

问题描述

示例报错LOG：

```
177.488928][ T1871] rkCIF_dvp: stream[0] start streaming

VIDIOC_STREAMON: failed: Out of memory

[ 177.489079][ T1871] vb2_cma_sg_alloc_contiguous: cma_en:1 alloc pages fail

[ 177.489126][ T1871] rkCIF rkCIF_dvp: rkCIF_alloc_buffer failed ret:-12

[ 177.489140][ T1871] rkCIF_dvp: Failed to allocate the memory for dummy buffer

[ 177.489152][ T1871] rkCIF_dvp: Failed to create dummy_buf, -12

[ 177.492215][ T1871] nvp6158 1-0030: nvp6158_power: on 0

[ 177.492265][ T1871] nvp6158 1-0030: __nvp6158_power_off(446)

[ 177.492304][ T1871] stream_cif_dvp_id0: close video, entity use_count 0
```

解决办法

问题原因是有些驱动没有实现xxxx_enum_frame_interval 函数，参考已有驱动实现

struct v4l2_subdev_pad_ops中的.enum_frame_interval函数，例如：

```
.enum_frame_interval = jaguar1_enum_frame_interval,aguar1_enum_frame_interval
```

