

密级状态： 绝密( ) 秘密( ) 内部资料( ) 公开(√)

# 图形显示客户支持与案例分析

(图形计算平台中心)

文件状态： [ ] 草稿 [ ] 正在修改 [√] 正式发布	文件标识:	RK-PC-YF-0001
	当前版本:	1.4.0
	作    者:	图形计算平台中心 GPU 组
	完成日期:	2020-07-29
	审    核:	熊伟
	审核日期:	2020-07-29



## 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 © 2019 瑞芯微电子股份有限公司

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

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd.

地址：福建省福州市铜盘路软件园 A 区 18 号

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

客户服务电话：+86-4007-700-590

客户服务传真：+86-591-83951833

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

# 前言

---

## 概述

本文档主要列举图形显示相关问题的总结与解决补丁，旨在指导相关工程师快速定位与解决图形相关问题。

本文档所有的补丁目录均以该文档发布的版本所在目录为基目录，向下链接补丁，故建议使用本文档的工程师不要移动本文档，否则可能无法找到对应的补丁目录。

## 产品版本

芯片名称	系统版本
RK3399 / RK3399 pro	Android / Linux
RK3326 / PX30	Android / Linux
RK3368 / PX5	Android
RK3288	Android / Linux
RK3328 / RK3228h	Android / Linux
RK322x	Android
RK3126c	Android
RK3128	Android / Linux
RK3188 / PX3	Android / Linux

## 读者对象

本文档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
2020/04/27	1.0.0	GPU图形组	初始版本
2020/05/14	1.1.0	GPU图形组	根据第一次审批意见修改版本
2020/05/22	1.2.0	GPU图形组	输出可发布版本1.2.0
2020/06/01	1.3.0	GPU图形组	文档格式修改为Markdown
2020/07/29	1.4.0	GPU图形组	输出可发布版本1.4.0

# 目录

---

---

前言

目录

1. SurfaceFlinger

  1.1. 镜像输出

    1.1.1. 显示方案-镜像功能-系统支持左右(上下)镜像输出

  1.2. 梯形矫正

    1.2.1. 显示方案-梯形矫正-功能验证

    1.2.2. 显示方案-梯形矫正-快速更新梯形坐标

    1.2.3. 显示方案-梯形矫正-实现正投影(背投影)镜像变换

    1.2.4. 显示问题-梯形矫正-双屏同显副屏画面缩小

  1.3. 显示问题

    1.3.1. 显示问题-开机动画-预旋转后开机动画显示不全问题

2. UI frameworks

  2.1. Toast-"app快速不停地show Toast"- "app crash"

3. GUI

  3.1. GUI-fd泄露-Surface未析构导致FenceFd泄露

4. HWUI

  4.1. 崩溃问题

    4.1.1. 崩溃问题-HWUI崩溃相关问题

  4.2. 调试方法

    4.2.1. 调试方法-关闭硬件加速-关闭HWUI方法

5. Hardware Composer ( HWC )

  5.1. HDMI相关问题

    5.1.1. 热插拔相关问题

      5.1.1.1. HDMI-主屏热插拔-副屏显示不全

      5.1.1.2. HDMI-切换分辨率-显示不满屏问题-RK3188-5.1

      5.1.1.3. HDMI-双屏同显-RK3128支持双屏同显方案

      5.1.1.4. HDMI-主副屏配置问题导致显示问题

  5.2. 显示问题

    5.2.1. 视频

      5.2.1.1. 视频-显示异常-色彩空间设置导致显示偏红

      5.2.1.2. 视频-HDR-显示效果与视频流畅相关问题

    5.2.2. 显示异常

      5.2.2.1. 显示抖动-RK3326显示多层1080p图层出现抖动

      5.2.2.2. 亮度异常-屏幕亮度忽明忽暗-RK3399

      5.2.2.3. 亮度异常-屏幕亮度忽明忽暗-RK3126c

      5.2.2.4. 显示闪屏-局部区域-Toast弹窗导致屏幕局部闪屏

      5.2.2.5. 显示闪屏-RK3288\_5.1显示闪屏问题

    5.2.3. 显示花屏

      5.2.3.1. 显示花屏-DRM-多区域格式配置错误导致显示花屏

      5.2.3.2. 显示花屏-RK\_FB-多区域格式配置错误导致显示花屏

    5.2.4. 显示黑屏

      5.2.4.1. 开机动画-显示黑屏-Android动画至Luncher存在黑屏

      5.2.4.2. 开机动画-显示黑屏-Kernel\_logo至Android动画存在黑屏-RK3128-4.4

  5.3. 显示卡顿

    5.3.1. 显示卡顿-4K视频播放卡顿问题

  5.4. 显示延迟

    5.4.1. 显示延迟-Camera码流显示延迟问题分析

  5.5. 系统异常

    5.5.1. 系统卡死

      5.5.1.1. 系统卡死-Monkey拷机-HDMI频繁插拔拷机导致系统卡死问题

    5.5.2. 系统崩溃

      5.5.2.1. 系统崩溃-Monkey拷机-HDMI频繁插拔导致系统重启

### 5.5.2.2. 系统崩溃-休眠唤醒拷机-HDMI频繁插拔导致系统重启

## 5.6. 功能说明

### 5.6.1. HWC属性说明

#### 5.6.1.1. Overscan属性

### 5.6.2. 预留图层功能

#### 5.6.2.1. 预留Plane-快速倒车说明与调试

## 6. Gralloc

### 6.1. Gralloc-CPU读写graphics\_buffer-显示有异常横条纹

### 6.2. 进程内存异常-HWC内存占用大

## 7. RGA

### 7.1. 咨询问题

#### 7.1.1. 咨询问题-RGA功能咨询

#### 7.1.2. 咨询问题-RGA性能问题咨询

#### 7.1.3. 咨询问题-RGA耗时是否正常

### 7.2. 需求支持

#### 7.2.1. 需求支持-客户SDK需要添加librga

#### 7.2.2. 需求支持-RGA用例使用物理地址说明

### 7.3. RGA运行问题

#### 7.3.1. librga问题

##### 7.3.1.1. 色彩偏差-RGA输出数据色域空间问题

##### 7.3.1.2. 调用失败-YUV宽度对齐约束条件不合理调用调用失败

#### 7.3.2. kernel 相关问题

##### 7.3.2.1. 3288-4.19内核-RGA出现TimeOut问题

##### 7.3.2.2. RGB888格式奇数宽输出存在黑色斜线的问题

##### 7.3.2.3. 显示黑线-RGA驱动BUG导致RGA合成输出图像随机黑线

##### 7.3.2.4. 内存重入-RGA内存访问越界

## 8. Frameworks

### 8.1. RK3288 双屏拔插物理输入设备APP出现闪黑的问题

## 9. GPU硬件报错排查流程

## 10. GPU 平台适配

### 10.1. GPU驱动初始化问题

#### 10.1.1. GPU驱动-初始化失败-mali\_ko与客户编译的kernel不兼容

#### 10.1.2. GPU驱动-初始化失败-无法获取regulator\_gpu

#### 10.1.3. GPU频率电压适配

##### 10.1.3.1. GPU异常-DTS配置问题导致GPU异常

##### 10.1.3.2. GPU频率配置-固定GPU频率导致系统不能启动

##### 10.1.3.3. 系统卡死-系统花屏-RK3368串口日志出现FenceTimeOut

### 10.2. GPU版本兼容问题

#### 10.2.1. 版本兼容-ko与so版本不兼容导致Android启动失败

#### 10.2.2. 版本兼容-so与libgralloc不兼容导致Android启动失败

#### 10.2.3. 版本兼容-RK3288不支持Android8.1及更高Android版本说明

### 10.3. Kernel崩溃问题

#### 10.3.1. Kernel崩溃-vdd\_gpu 未配置为 always-on

#### 10.3.2. Mali400\_GPU-设备上没有外置RTC-kernel提示BUG,死机

## 11. GPU 性能相关

### 11.1. 性能不足问题

#### 11.1.1. 性能问题-低温测试-RK3288(5.1)GPU低温测试性能差

#### 11.1.2. 温控异常-RK3399(7.1)GPU温控行为异常

#### 11.1.3. 性能问题-代码BUG-RK3288(6.0)某提交导致GPU性能差

### 11.2. 性能优化

#### 11.2.1. 性能优化-调整GPU变频行以提高部分场景性能

### 11.3. 性能跑分

## 12. GPU 功能支持

### 12.1. OpenCL支持

#### 12.1.1. OpenCL支持-可执行用例

#### 12.1.2. 应用异常-并发调用 OpenGL 和 OpenCL 时应用异常

#### 12.1.3. 兼容问题-不同版本 DDK 编译的 OpenCL binary kernel 不兼容

### 12.2. Vulkan支持

12.2.1. Vulkan支持-Mali

12.2.2. Vulkan-应用卡住-PVR DDK 1.8 Vulkan相关应用卡住的问题

## 13. GPU问题

### 13.1. GPU 第三方应用相关问题

#### 13.1.1. Antutu

13.1.1.1. Antutu-Vulkan-安兔兔V8相关测试无法运行

13.1.1.2. Antutu-安兔兔V7 GPU项0分

13.1.1.3. Antutu-PVR-安兔兔精炼厂测试无法运行

13.2. 应用卡住-Mali4x0缺少重要提交导致应用无响应问题

13.3. 驱动报错-拷机出现"job status 0x00000004 (TERMINATED)"

13.4. PVR-DDK1.5升级到1.8版本 ( 针对部分webview 场景 )

13.5. PVR-渲染错误-应用界面出现非预期黑色区域

## 14. Linux

### 14.1. 系统问题

#### 14.1.1. 系统崩溃

14.1.1.1. 系统崩溃-SDK错误使能MALI\_BIFROST\_FOR\_LINUX

14.1.1.2. 系统崩溃-客户错误使用MALI\_BIFROST\_FOR\_LINUX

#### 14.2. 咨询问题

14.2.1. 咨询问题-GPU是否启用

#### 14.3. Benchmark相关

14.3.1. Benchmark跑分-Glmark2 跑分较低

#### 14.4. OpenCL

14.4.1. Linux-OpenCL-可执行用例

14.4.2. Linux-运行某些OpenCL应用-应用卡死

#### 14.5. 第三方应用问题

14.5.1. Chromium

14.5.1.1. Chromium-GPU 报错导致Chromium-Wayland启动失败

14.5.1.2. Chromium-部分版本无法load gpu 库

#### 14.6. Linux librga

14.6.1. Librga-部分YUV 格式存在uv 变反的问题

14.6.2. Librga-X11上窗口拖动卡顿，存在残影

14.6.3. Librga-Weston 上使用RGA 渲染

---

# 1. SurfaceFlinger

## 1.1. 镜像输出

### 1.1.1. 显示方案-镜像功能-系统支持左右(上下)镜像输出

归属模块：

SurfaceFlinger

关键字：

左右镜像、上下镜像

现象说明：

产品需要对显示输出图像进行左右(上下)镜像

解决方法：

补丁修改如下：

```
1 --- a frameworks/native/services/surfaceflinger/RenderEngine/ProgramCache.cpp
2 +++ b frameworks/native/services/surfaceflinger/RenderEngine/ProgramCache.cpp
@@ -274,6 +274,7 @@ String8 ProgramCache::generateVertexShader(const Key& needs) {
3         << "gl_Position = projection * position;" ;
4 +         << "gl_Position.x = float(0) - gl_Position.x;" ;
5     if (needs.isTexturing()) {
6 }
```

补丁说明：

1. 关闭HWC，强制使用GPU合成输出。操作方法：

```
1 setprop sys.hwc.compose_policy 0
2 setprop sys.hwc.enable 0
```

或者在build.prop修改对应字段如下，然后重启系统生效：

```
1 setprop sys.hwc.compose_policy=0
2 setprop sys.hwc.enable=0
```

2. 如果要做上下镜像，以此类推把上面x改成y。如下：

```
1 + << "gl_Position.y = float(0) - gl_Position.y;" ;
```

SDK commit:

1. 未提交SDK
2. 此功能针对个别客户特殊需求，修改较为简单。

相关Redmine:

Defect #240782 : <https://redmine.rock-chips.com/issues/240782>

Defect #163718 : <https://redmine.rock-chips.com/issues/163718>

## 1.2. 梯形矫正

### 1.2.1. 显示方案-梯形矫正-功能验证

**归属模块：**

libkeystone

**关键字：**

梯形校正无效果

**现象说明：**

客户在移植完梯形校正补丁后，发现开机并无校正效果，寻求帮助。

**解决方法：**

1. 检查是否有按照readme.txt中的说明正确移植；
2. 开机后，检查HWC属性是否关闭：getprop|grep hwc；
3. 多屏状态的话，检查persist.sys.keystone.display.id属性是否与目标id一致；
4. 检查使用是否规范，各个属性值的作用在补丁readme.txt中有说明；

**补丁说明：**

各版本keystone补丁已提交至以下目录：

<Android/frameworks/redmine/feature/keystone>

**SDK commit:**

不建议此功能合并至主分支，默认的代码改动较多，可能会影响正常的版本，建议以补丁形式提供。

**相关Redmine:**

Defect #185642 : <https://redmine.rock-chips.com/issues/185642>

Defect #197931 : <https://redmine.rock-chips.com/issues/197931>

Defect #196112 : <https://redmine.rock-chips.com/issues/196112>

Defect #179256 : <https://redmine.rock-chips.com/issues/179256>

## 1.2.2. 显示方案-梯形矫正-快速更新梯形坐标

**归属模块：**

libkeystone

**关键字：**

画面刷新

**现象说明：**

1. 梯形功能能够正常使用，但是每次要画面过很久才生效；
2. Demo apk中由于按键会修改到坐标值，所以画面更新，能直接生效；
3. 如果用属性值，就需要动下光标才行。

**解决方法：**

在HWC模块中打上强制刷新补丁，系统每隔若干时间刷新一帧，触发梯形矫正新坐标生效。

**补丁说明：**

1. 补丁已提交至以下路径：

<Android/frameworks/redmine/feature/keystone/hwc force refresh 4.4 and 7.1>

2. 根据不同HWC版本，在/hardware/rockchip/hwcomposer打上对应的补丁

**SDK commit:**

无

**相关Redmine:**

Defect #180625：<https://redmine.rock-chips.com/issues/180625>

Defect #198822：<https://redmine.rock-chips.com/issues/198822>

### 1.2.3. 显示方案-梯形矫正-实现正投影（背投影）镜像变换

**归属模块：**

libkeystone

**关键字：**

投影机，镜像变换

**现象说明：**

客户在投影机设备上有镜像需求。

**解决方法：**

1. 梯形校正补丁自带镜像的功能；
2. 让客户按照《投影梯形校正补丁说明.txt》中的说明，先熟悉梯形校正四个顶点坐标设置的方式：

<Android/frameworks/redmine/feature/keystone>

**投影梯形校正补丁说明.txt**

3. 要实现左右镜像，可对顶点的属性设置成如下值（假设分辨率为1280x720）

```
1 persist.sys.keystone.lt=1280,0  
2 persist.sys.keystone.rt=-1280,0  
3 persist.sys.keystone.lb=1280,0  
4 persist.sys.keystone.rb=-1280,0
```

简单来说就是将四个顶点左右调换位置。

4. 此功能将会在最新的梯形校正补丁中，做成一个属性，进行开关控制。预计属性将会命名为：

```
1 persist.sys.keystone.mirror_x  
2 persist.sys.keystone.mirror_y
```

**补丁说明：**

无

**SDK commit:**

该修改未合并SDK内部分支，建议以补丁形式提供。

**相关Redmine:**

Defect #240782 : <https://redmine.rock-chips.com/issues/240782>

Feature #240681 : <https://redmine.rock-chips.com/issues/240681>

## 1.2.4. 显示问题-梯形矫正-双屏同显副屏画面缩小

**归属模块：**

libkeystone

**关键字：**

梯形校正，画面缩小

**现象说明：**

单屏时，梯形校正功能正常，插上hdmi等第二个屏，梯形整个屏幕显示不完全。

**解决方法：**

1. 在SurfaceFlinger.cpp的doDisplayComposition() 函数中加log进行调查；
2. 打印出宽高，发现宽高在变换，第二个屏始终以第一个屏的宽高进行计算；
3. 分析后得出原因：这是由于第二个屏与第一个屏的屏幕分辨率不同，在做合成时，两个屏交替做合成，而梯形校正取的宽高是上一帧的宽高。

**补丁说明：**

1. 补丁已提交至以下目录的对应文件：

[Android/frameworks/redmine/feature/keystone](#)

0001-keystone-to-solve-HDMI-can-not-do-keystone.-add-new-.patch

2. 请在/frameworks/native/ 处打上改patch可以解决问题

**SDK commit:**

该补丁未合并入SDK内部分支，目前以补丁形式提供。

**相关Redmine:**

Defect #190074 : <https://redmine.rock-chips.com/issues/190074>

Defect #197958 : <https://redmine.rock-chips.com/issues/197958>

## 1.3. 显示问题

### 1.3.1. 显示问题-开机动画-预旋转后开机动画显示不全问题

**归属模块：**

SurfaceFlinger

**关键字：**

预旋转，开机动画显示不全

**现象说明：**

某些广告机客户因安装需要，希望能有display预旋转功能。

Android 9.0与Android 10.0 使用google推荐流程达到display预旋转，在开机动画时，会有显示不全问题。现象是Android的字母d消失，或者客户定制的开机动画被截断，另一半变黑。

**解决方法：**

1. 此现象是google显示框架的BUG，由于开机那段时间获取到的宽高非预期的宽高，导致此问题。
2. 目前RK已经修复此BUG，提交至SDK工程中。

**补丁说明：**

1. 补丁存放目录为：

[Android/frameworks/redmine/display\\_issue/pre\\_rotate\\_cause\\_bootanimaltion\\_display\\_err](https://github.com/rock-chips/rockchip-android/tree/master/Android/frameworks/redmine/display_issue/pre_rotate_cause_bootanimaltion_display_err)

0001-Android-9.0-SurfaceFlinger-Fix-a-prerotation-bug.patch

0001-Android-10.0-SurfaceFlinger-fix-setProjection-get-Width-and-Heigh.patch

2. 若是客户不愿更新SDK，想要我们提供patch，此时需要确认清楚是否有版本冲突问题。

**SDK commit:**

补丁已提交至SDK中，可让客户更新sdk解决此问题。Android\_9.0更新至20190627及之后版本；  
Android\_10.0更新至RKR6及之后版本。

**相关Redmine:**

Android 10.0：

Defect #242380：<https://redmine.rock-chips.com/issues/242380>

Defect #242255：<https://redmine.rock-chips.com/issues/242255>

Android 9.0：

Defect #242690：<https://redmine.rock-chips.com/issues/242690>

Defect #212046：<https://redmine.rock-chips.com/issues/212046>

Defect #209873：<https://redmine.rock-chips.com/issues/209873>

## 2. UI frameworks

### 2.1. Toast-"app快速不停地show Toast"- "app crash"

问题描述：

app 快速不停地 show 不同的 Toast 实例, surfaceflinger 进程中 fd 被耗尽, 导致 app 发起的 dequeueBuffer() 失败, app crash.

对应的 log :

```
1 04-02 08:23:07.221 264 264 E Fence : merge:  
sync_merge("Framebuffersurface:0", 1021, 1019) returned an error: Too many  
open files (-24)  
2 ...  
3 04-02 08:23:06.449 264 384 E BufferQueueProducer: [Toast#72]  
dequeueBuffer: createGraphicBuffer failed  
4 04-02 08:23:06.449 3861 3877 F OpenGLRenderer: Failed to set damage  
region on surface 0x7a753cd140, error=EGL_BAD_ACCESS  
5 ----- beginning of crash  
6 04-02 08:23:06.450 3861 3877 F libc : Fatal signal 6 (SIGABRT), code  
-6 in tid 3877 (RenderThread), pid 3861 (d.helloactivity)  
7 ...  
8 04-02 08:23:06.560 3910 3910 F DEBUG : pid: 3861, tid: 3877, name:  
RenderThread >>> com.example.android.helloactivity <<<  
9 ...  
10 04-02 08:23:06.570 3910 3910 F DEBUG : Abort message: 'Failed to set  
damage region on surface 0x7a753cd140, error=EGL_BAD_ACCESS'
```

另, 在 dumpsys SurfaceFlinger 的输出中, 可以看到 数量异常多的名称为 Toast#X 的 layer 和对应的 buffer.

上面的 "Toast#X" 中的 X 是大于等于 0 的数值.

相关Redmine :

Defect #256742: <https://redmine.rock-chips.com/issues/256742>

Defect #253364: <https://redmine.rock-chips.com/issues/253364>

解决方法：

frameworks/base, pick Android 9.0 中的 commit 99298431b, 可以解决本异常.

### 3. GUI

#### 3.1. GUI-fd泄露-Surface未析构导致FenceFd泄露

问题描述：

客户平台设备为 RK3368 Android 8.1

客户Monkey拷机过程中，发现系统显示异常，卡死，崩溃等现象，发现相关日志如下：

```
1 # 出现 Too many open files 报错，是典型的fd泄露问题
2 E/Fence    ( 279): merge: (...)returned an error: Too many open files (-24)
3
4 # Fence fd 索引达到1023，进一步说明fd存在泄露问题，因为fd的最大值为1024
5 W/HwcComposer( 279): failed to dup fence 1018
6 W/HwcComposer( 279): failed to dup fence 1019
7 W/HwcComposer( 279): failed to dup fence 1023
8
9 # dup fd 失败导致Pvr driver 报错
10 E/Parcel   (21148): fcntl(F_DUPFD_CLOEXEC) failed in Parcel::read, i is 0,
11   fds[i] is 170, fd_count is 1, error: Unknown error 2147483647
12 E/IMGSRV   (21148): android_ws.c:374: DequeueBufferWrapper: Failed to de-
13   queue buffer (-22)
14 E/IMGSRV   (21148): generic_ws.c:350: KEGLGetDrawableParameters: Native
15   window is invalid
16 E/IMGSRV   (21148): eglglue.c:2538: GLES3MakeCurrentGC: Invalid write
17   drawable - what do we do?
18 E/IMGSRV   (21148): khronos_egl.c:6155: IMGeeglMakeCurrent: unexpected error
19   code
20
21 # 最终导致 Fatal 报错，系统出现crash
22 F/OpenGLRenderer(21148): Failed to make current on surface 0x7e479262c0,
23   error=EGL_SUCCESS
24 F/libc     (21148): Fatal signal 6 (SIGABRT), code -6 in tid 21266
25   (RenderThread), pid 21148 (oid.m1.launcher)
26
27 # 结论：由fd泄露导致的系统crash问题
```

相关Redmine：

Defect #256905：<https://redmine.rock-chips.com/issues/255308#change-2386256>

详细文档及补丁文件：

结论：由fd泄露导致的系统crash问题，最终确定为客户播放库相关结构未释放导致问题，详细的文档分析如下：

[Android/gui/surface\\_object\\_undestruct\\_cause\\_fence\\_fd\\_leak/](#)

## 4. HWUI

### 4.1. 崩溃问题

#### 4.1.1. 崩溃问题-HWUI崩溃相关问题

##### 现象说明

APK 存在crash、闪退等现象。查看logcat，能发现crash log，最终堆栈指向libhwui。

##### 典型log

1、Toast 异常引起 bufferqueue has been abandoned ( 多见于Android 8.1 )

```
1 | 05-08 20:23:15.926 239 445 E BufferQueueProducer: [Toast#0]
|   dequeueBuffer: BufferQueue has been abandoned
```

2、其他异常引起 bufferqueue has been abandoned

```
1 | F DEBUG    :      #02 pc 0004d12b /system/lib/libhwui.so
|   (Android::uirenderer::renderthread::EglManager::damageFrame(android::uirender
|   er::renderthread::Frame const&, SkRect const&)+98)
```

3、fd泄漏

```
1 | 06-06 00:55:47.780 E/Parcel  (21148): fcntl(F_DUPFD_CLOEXEC) failed in
|   Parcel::read, i is 0, fds[i] is 170, fd_count is 1, error: unknown error
|   2147483647
```

##### 详细文档及补丁文件：

详情请阅读以下文档：

[Android/hwui/assert crash issue/](#)

## 4.2. 调试方法

### 4.2.1. 调试方法-关闭硬件加速-关闭HWUI方法

**归属模块：**

hwui

**关键字：**

针对应用关闭hwui，应用图标显示异常，布局异常

**现象说明：**

1. 某些应用有布局问题，如：图标过大，图标重叠，字体乱码。
2. 在开发者选项中设置为 **尝试关闭hwui硬件加速** 后，显示异常消失。可定位为hwui 硬件加速的问题。

**解决方法：**

针对问题应用，通过判定其进程名，做关闭hwui的处理。

**可能的风险：**针对应用hwui硬件加速后，可能会在运行该应用时，cpu负载稍微上升。

**补丁说明：**

1. 补丁已提交至

[Android/hwui/debug\\_method/disable\\_hwui\\_method](#)

**ViewRootImpl\_diff.patch**

2. 请在frameworks\base\libs\hwui打上该补丁，并修改apk进程名的字符串；
3. 编译更新重启后，检查logcat是否有rk-debug相关字段。

**SDK commit:**

该问题补丁没有合并至SDK的必要，这里只是记录一个调试手段，而且在 **设置->开发者选项->尝试关闭hwui硬件加速** 可以达到关闭所用应用hwui硬件加速 的效果。

**相关Redmine:**

Defect #243982：<https://redmine.rock-chips.com/issues/243982>

## 5. Hardware Composer ( HWC )

### 5.1. HDMI相关问题

#### 5.1.1. 热插拔相关问题

##### 5.1.1.1. HDMI-主屏热插拔-副屏显示不全

现象说明：

1. 主屏插入，副屏出现显示不全的问题；
  1. 显示超出屏幕，仅显示完整图像的一部分；
  2. 显示不满屏，仅屏幕左上角有图像，外围黑屏；
2. 副屏插入后现象恢复；
3. 现象如下：屏幕显示完整图像的一部分



相关Redmine:

Defect #221917 : <https://redmine.rock-chips.com/issues/221917>

Defect #244758 : <https://redmine.rock-chips.com/issues/244758>

详细文档及补丁文件：

问题最终定位为HWC代码BUG，详情请阅读以下文档：

[Android/hwcomposer/redmine/hdmi\\_issues/hdmi\\_hotplug\\_cause\\_display\\_error/](Android/hwcomposer/redmine/hdmi_issues/hdmi_hotplug_cause_display_error/)

### 5.1.1.2. HDMI-切换分辨率-显示不满屏问题-RK3188-5.1

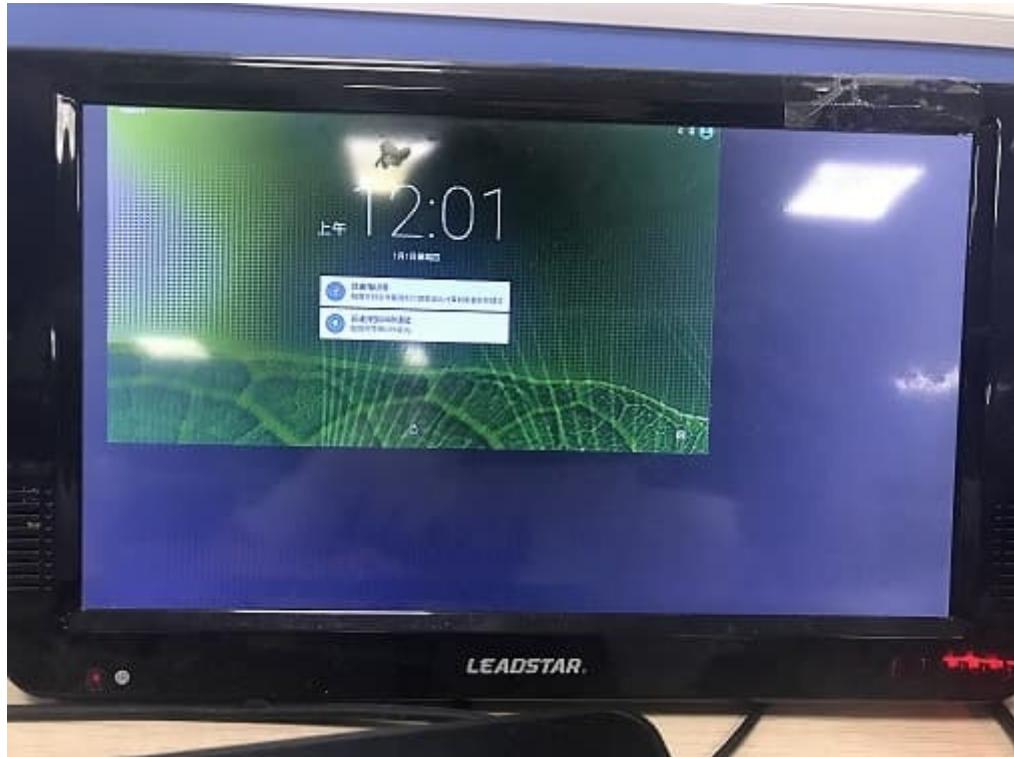
现象说明：

- 客户环境仅使用主屏幕，原始分辨率为1280x720p60；

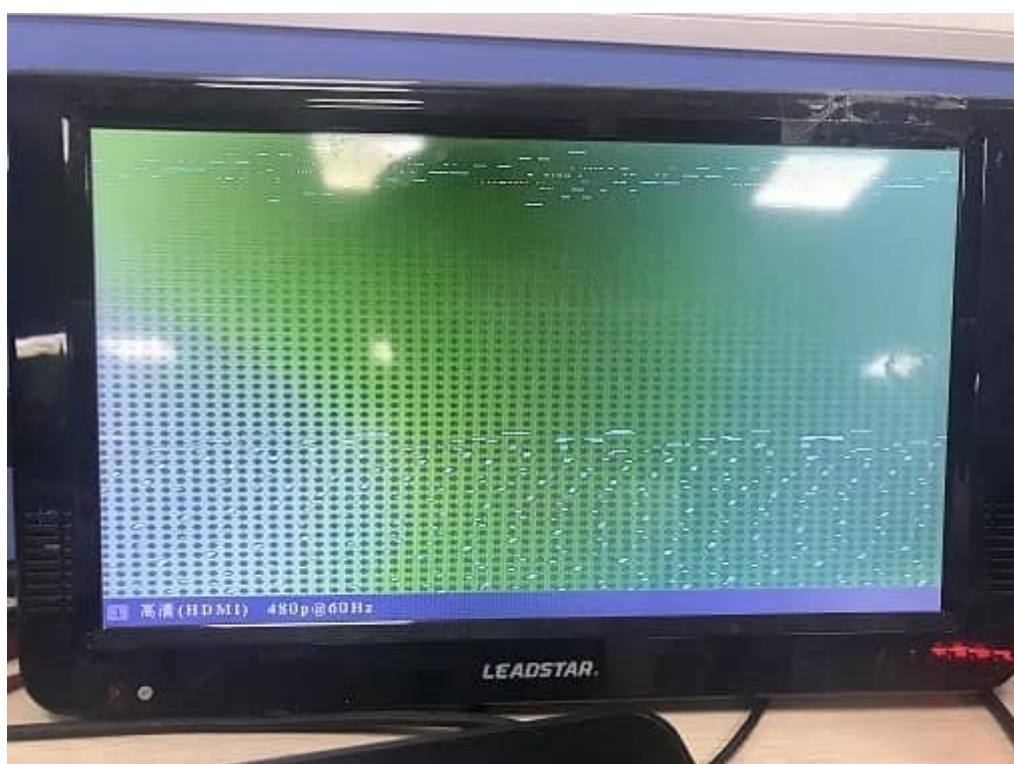
使用一下两个命令切换分辨率

```
1 | echo 1920x1080p-60 > /sys/class/display/HDMI	mode
2 | echo 720x480p-60 > /sys/class/display/HDMI	mode
```

1.调整屏幕分辨率为1920x1080p60后，画面大小集中在屏幕左上角，约屏幕1/4大小，现象如下：



2.调整屏幕分辨率为720x480p60后，画面放大，且屏幕闪烁明显，现象如下：



**相关Redmine:**

Support #250075 : <https://redmine.rock-chips.com/issues/250075>

**详细文档及补丁文件 :**

问题最终定位为RK3188 Android5.1平台HWC代码缺少针对切换分辨率进行缩放的功能，详情请阅读以下文档：

[Android/hwcomposer/redmine/hdmi\\_issues/hdmi\\_switch\\_resolution\\_incomplete\\_display\\_rk3188\\_5\\_1/](Android/hwcomposer/redmine/hdmi_issues/hdmi_switch_resolution_incomplete_display_rk3188_5_1/)

### 5.1.1.3. HDMI-双屏同显-RK3128支持双屏同显方案

**现象说明：**

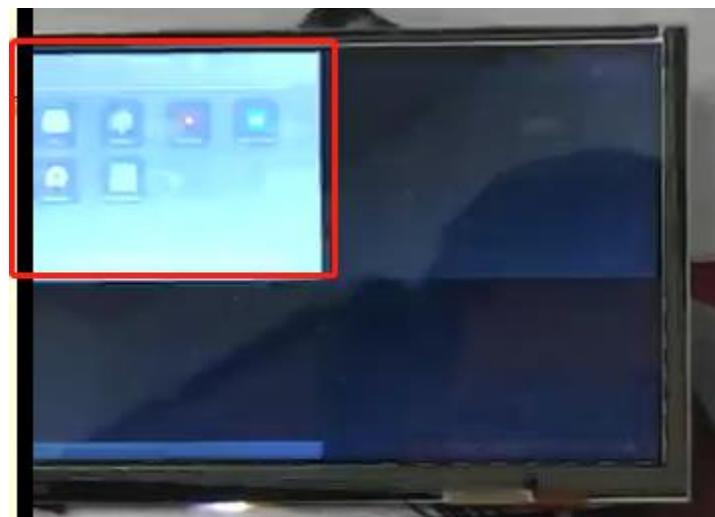
RK3128 Android 7.1

HDMI插拔过程中，LCD屏幕会出现画面缩放右上角，显示不全，花屏等现象，现象如下：

**正常显示：**



**显示缩放，显示不全：**



**花屏：**



**相关Redmine:**

Defect #243080 : <https://redmine.rock-chips.com/issues/243080>

**详细文档及补丁文件 :**

详情请阅读以下文档 :

[Android/hwcomposer/redmine/hdmi\\_issues/hdmi\\_support\\_dual\\_display\\_rk3128/](Android/hwcomposer/redmine/hdmi_issues/hdmi_support_dual_display_rk3128/)

#### 5.1.1.4. HDMI-主副屏配置问题导致显示问题

**归属模块：**

Hardware Composer (HWC)

**关键字：**

分辨率切换、显示分辨率、渲染分辨率

**现象说明：**

主副屏配置，解决可能存在的显示不满屏问题，闪屏问题等，均是由于主副屏没有正确配置导致

**解决方法：**

主副屏配置属性如下：

```
1 // Android 7.1 / 8.1
2 sys.hwc.device.primary //设置主屏设备类型，由用户设置
3 sys.hwc.device.extend //设置副屏设备类型，由用户设置
4 // Android 9.0 以上版本
5 vendor.hwc.device.primary //设置主屏设备类型，由用户设置
6 vendor.hwc.device.extend //设置副屏设备类型，由用户设置
```

详情请参考以下目录文档：

[Android/hwcomposer/feature/hwc\\_property/primary\\_extend\\_property](#)

**SDK commit:**

该修改已合并内部开发分支，相关commit message如下：

```
1 commit 97adadfaf5e8c2137d2a32fd457ad158ada56e8b
2 Author: Mark Yao <mark.yao@rock-chips.com>
3 Date:   wed Aug 16 10:29:54 2017 +0800
4
5     support configure primary/extend displays with properties
6
7     Change-Id: I3c51757e62c796ab4ab75d9edf4928087dedf7d6
8     Signed-off-by: Mark Yao <mark.yao@rock-chips.com>
```

HWC更新至 0.31 版本即可包含该提交，版本信息可查询以下属性获得：

sys.ghwc.version 或 vendor.ghwc.version

**相关Redmine:**

Defect #242890 : <https://redmine.rock-chips.com/issues/242890>

Support #234212 : <https://redmine.rock-chips.com/issues/234212>

## 5.2. 显示问题

### 5.2.1. 视频

#### 5.2.1.1. 视频-显示异常-色彩空间设置导致显示偏红

现象说明：

RK3399-10.0 客户更新最新HWC版本后，出现主界面滑动色彩偏红的问题，现象如下：



进入主界面后，静态桌面显示正常，滑动桌面的情况下颜色显示偏红，如上图，正常显示情况如下图：



复现问题关键信息：

- HWC 启动的情况下，静态桌面色彩恢复正常，滑动桌面复现色彩偏红异常
- HWC关闭的情况下采用GPU合成，则无法复现问题

相关Redmine:

Defect #255877 : <https://redmine.rock-chips.com/issues/255877>

详细问题说明文档：

问题定位为Kernel driver没有支持 colorspace 类型导致显示偏红，详细的文档请参考：

[Android/hwcomposer/redmine/display\\_issues/video/colorspace\\_config\\_cause\\_display\\_error/](Android/hwcomposer/redmine/display_issues/video/colorspace_config_cause_display_error/)

### 5.2.1.2. 视频-HDR-显示效果与视频流畅相关问题

**现象说明：**

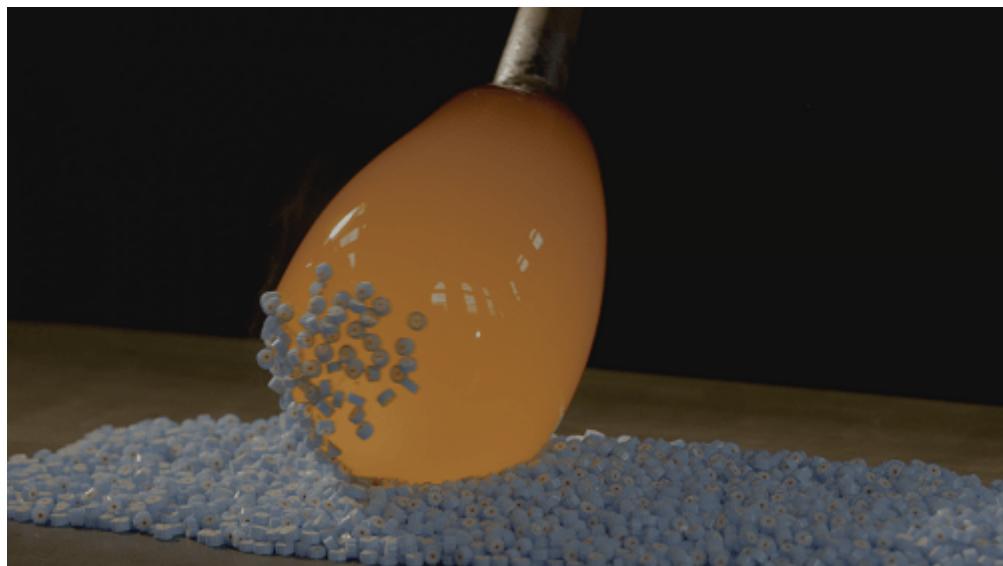
客户RK3399-10.0平台播放HDR视频发现如下现象：

- Android 10.0 HDMI显示器支持HDR，视频显示效果正常；
- Android 10.0 HDMI显示器不支持HDR，视频显示效果异常，并且LVDS屏幕显示也异常；
- Android 7.1 HDMI显示器支持HDR，视频显示效果正常；
- Android 7.1 HDMI显示器不支持HDR，视频显示效果正常，并且LVDS屏幕显示也正常；

HDR正常显示效果如下：



HDR异常显示效果如下：



**相关Redmine:**

Defect #254008 : <https://redmine.rock-chips.com/issues/254008>

**详细问题说明文档：**

详细的文档请参考：

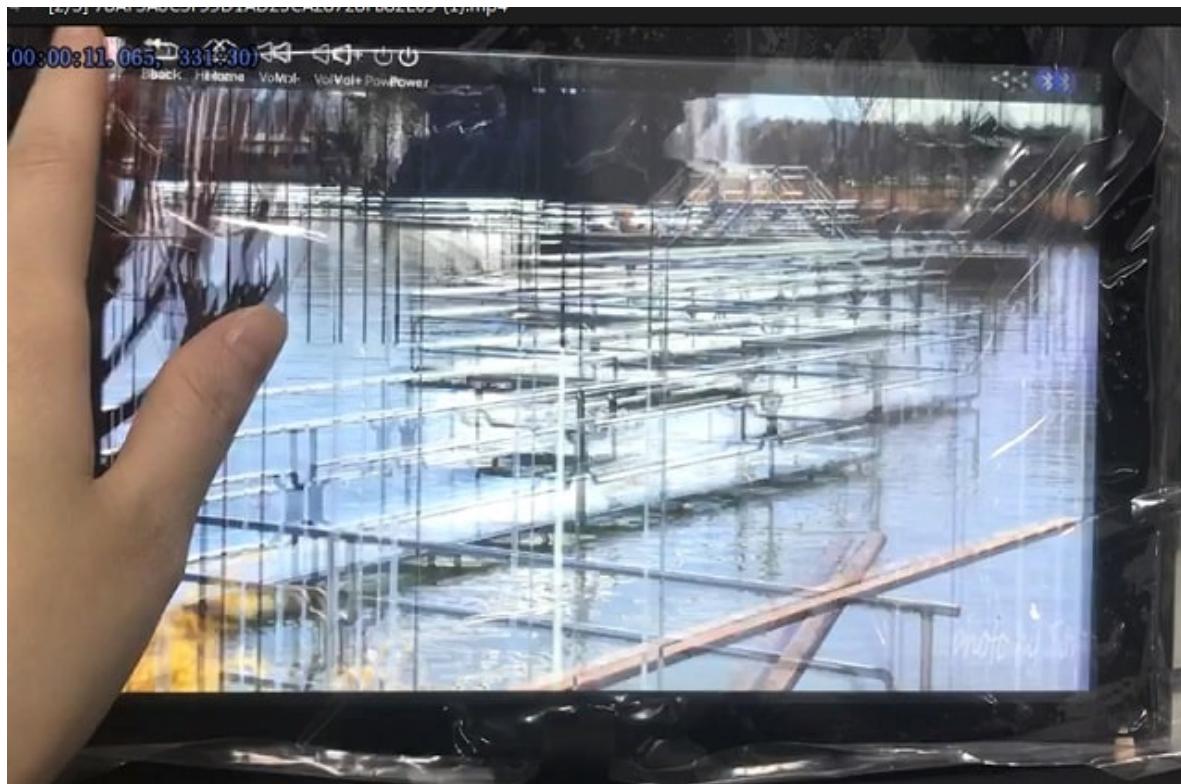
[Android/hwcomposer/redmine/display\\_issues/video/hdr\\_video\\_display\\_problem\\_in\\_rk3399](Android/hwcomposer/redmine/display_issues/video/hdr_video_display_problem_in_rk3399)

## 5.2.2. 显示异常

### 5.2.2.1. 显示抖动-RK3326显示多层1080p图层出现抖动

现象说明：

客户在目标场景出现闪条纹的抖动问题，现象如下，详情请参考Defect #256905提供的现象视频：



并且串口伴随以下日志输出：

```
1 [ 361.084769] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
2 [ 370.557633] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
3 [ 370.557775] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
4 [ 370.690141] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
5 [ 370.690915] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
6 [ 370.690997] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
7 [ 370.691125] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
8 [ 370.691149] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
9 [ 370.691167] rockchip-vop ff460000.vop: [drm:vop_isr] *ERROR*
POST_BUF_EMPTY irq err
```

相关Redmine:

Defect #256905 : <https://redmine.rock-chips.com/issues/256905>

Defect #225167 : <https://redmine.rock-chips.com/issues/225167>

详细问题说明文档：

详细的文档请参考：

[Android/hwcomposer/redmine/display\\_issues/screen\\_tear/fhd\\_screen\\_tear\\_on\\_rk3326/](https://Android/hwcomposer/redmine/display_issues/screen_tear/fhd_screen_tear_on_rk3326/)

### 5.2.2.2. 亮度异常-屏幕亮度忽明忽暗-RK3399

**归属模块：**

Hardware Composer (HWC)

**关键字：**

屏幕亮度闪烁、忽明忽暗、RK3399

**现象说明：**

部分场景屏幕亮度忽明忽暗，关闭 HWC 后问题消失；

**问题分析：**

VOP win2/win3 做 YUV2RGB or RGB2YUV 时转换系数不对导致 vop overlay 的数据与 GPU 合成输出的数据颜色不一致，导致效果差异，关闭 HWC 图像均由 GPU 输出，故现象消失；

**解决方法：**

在 Kernel 检查以下补丁是否存在，若不存在则打上补丁：

[Android/hwcomposer/redmine/display\\_issues/commit/9](https://github.com/rock-chips/Android/hwcomposer/redmine/display_issues/commit/9)

e9ee59b.diff.zip

**SDK commit:**

该修改已经合并内部主分支，相关commit message 如下：

```
1 commit e9ee59b0ce8cca88000fe21285b31fa1d9d9056a
2 Author: Sandy Huang <hjc@rock-chips.com>
3 Date:   Thu, 24 Jan 2019 18:04:37 +0800
4 drm/rockchip: vop: add extern csc for rk3399 win lite
5
6 Change-Id: I7a92e64e2bcb9695b23895a6fd79ca5434852d55
7 Signed-off-by: Sandy Huang <hjc@rock-chips.com>
```

请检查客户kernel 是否包含该修改，或者检查补丁修改内容是否在客户工程中已包含。

**相关Redmine:**

Defect #203740 : <https://redmine.rock-chips.com/issues/203740>

Defect #206634 : <https://redmine.rock-chips.com/issues/206634>

### **5.2.2.3. 亮度异常-屏幕亮度忽明忽暗-RK3126c**

**归属模块：**

Hardware Composer (HWC)

**关键字：**

屏幕亮度闪烁、忽明忽暗、RK3126c

**现象说明：**

部分场景屏幕亮度忽明忽暗，关闭 HWC 后问题消失；

**问题分析：**

vop alpha 预乘值计算不对，导致颜色差异

**解决方法：**

在 Kernel 检查以下补丁是否存在，若不存在则打上补丁：

[Android/hwcomposer/redmine/display\\_issues/display\\_config\\_err/display\\_brightness\\_flash/rk3126c](https://Android/hwcomposer/redmine/display_issues/display_config_err/display_brightness_flash/rk3126c)

**SDK commit:**

该修改已经合并内部kernel分支，可以检查kernel 提交中是否存在该补丁修改，如果没有建议客户打上补丁。

**相关Redmine:**

Defect #205279：<https://redmine.rock-chips.com/issues/205279>

#### 5.2.2.4. 显示闪屏-局部区域-Toast弹窗导致屏幕局部闪屏

归属模块：

Hardware Composer (HWC)

关键字：

Toast、弹窗、闪屏

现象说明：

Toast弹窗右侧区域屏幕出现闪屏，关闭HWC后问题消失；

目前发现在 RK3288W-5.1 与 RK3288W-7.1



问题分析：

在图层使用较小区域显示时，如果使用IOMMU 需要使能 Axi outstanding 功能,经过测试需要设置Axi outstanding max number 为30可以解决 win flush 的问题

解决方法：

补丁文件位于以下目录，分为 kernel 3.10 版本与kernel 4.4 版本，请在kernel 目录打包：

[Android/hwcomposer/redmine/display\\_issues/display\\_config\\_err/toast\\_cause\\_screen\\_display\\_err](https://redmine.rock-chips.com/projects/android-hwcomposer/issues/177855)

**SDK commit:**

内部分支均已提交，请下载补丁检查对应文件是否包含该修改。

**相关Redmine:**

Defect #177855 : <https://redmine.rock-chips.com/issues/177855>

Defect #229685 : <https://redmine.rock-chips.com/issues/229685>

### 5.2.2.5. 显示闪屏-RK3288\_5.1显示闪屏问题

**归属模块：**

Hardware Composer (HWC)

**关键字：**

Toast、弹窗、闪屏、RK3288、Android 5.1

**现象说明：**

显示闪屏，串口出现如下报错：

```
1 <3>[ 1459.801794] win[2]->area[0],not support scale
2 <3>[ 1459.801818] xact=70,yact=1080,xsize=71,ysize=1080
3 <3>[ 1459.817012] win[2]->area[0],not support scale
4 <3>[ 1459.817038] xact=70,yact=1080,xsize=71,ysize=1080
```

**问题分析：**

kernel rk\_fb driver 有对所有格式 offset 作偶数对齐，进而导致需要缩放，而HWC配置的win不支持缩放，最终导致显示问题

**解决方法：**

在kernel 目录打包，补丁文件位于以下目录：

[Android/hwcomposer/redmine/display\\_issues/display\\_config\\_err/display\\_flash/rk3288\\_5.1\\_display\\_flash](https://Android/hwcomposer/redmine/display_issues/display_config_err/display_flash/rk3288_5.1_display_flash)

**SDK commit:**

该修改已经合并入kernel主分支，相应的commit message 如下：

```
1 commit 9f2ef4bf49aedbe25d3d176be7f612b6e9435237
2 Author: Huang Jiachai <hjc@rock-chips.com>
3 Date:   Wed Oct 14 10:41:00 2015 +0800
4
5     video: rockchip: fb: distinguish between RGB YUV and FBDC data format
6
7     Change-Id: Ia351340cf05e5264d798348178b488917ada4482
8     Signed-off-by: Huang Jiachai <hjc@rock-chips.com>
```

请下载补丁查看工程是否包含该修改。

**相关Redmine:**

Support #206803 : <https://redmine.rock-chips.com/issues/206803>

### 5.2.3. 显示花屏

#### 5.2.3.1. 显示花屏-DRM-多区域格式配置错误导致显示花屏

归属模块：

Hardware Composer (HWC)

关键字：

花屏，小区域，多区域配置错误

现象说明：

屏幕小区域面积出现花屏，现象如下：



解决方法：

1. 尝试关闭HWC，查看问题是否复现：

```
1 // Android 8.1 及以下版本:  
2 adb shell setprop sys.hwc.compose_policy 0  
3 adb shell setprop sys.hwc.enable 0  
4 // Android 9.0 及以上版本:  
5 adb shell setprop vendor.hwc.compose_policy 0  
6 adb shell setprop vendor.hwc.enable 0
```

2. 复现问题后抓打印以下log:

```
1 // Android 8.1 及以下版本:  
2 // 复现问题，需要将 正常->异常 完整的过程都抓打印下来:  
3 adb shell setprop sys.hwc.log 511;logcat -c ;logcat > hwc.log  
4 // 正常情况与异常情况分别打印以下log，提供分析:  
5 adb shell cat /d/dri/0/summary > summary.log  
6 adb shell dumpsys SurfaceFlinger > sf.log  
7 // Android 9.0 及以上版本:  
8 adb shell setprop vendor.hwc.log 511;logcat -c ;logcat > hwc.log  
9 // 正常情况与异常情况分别打印以下log，提供分析:  
10 adb shell cat /d/dri/0/summary > summary.log  
11 adb shell dumpsys SurfaceFlinger > sf.log
```

3. 分析相关log，重点查看summary.log

#### 4. 提供补丁验证问题

详细过程可阅读以下目录文档：

[Android/hwcomposer/redmine/display\\_issues/display\\_config\\_err/drm\\_multi\\_areas\\_config\\_error\\_cause\\_display\\_err](#)

#### SDK commit:

该修改已经合并内部开发分支，相关commit message 如下：

```
1 commit 8e4f21b8aedcce05b94795e1c8022b55191b18d9
2 Author: libin <bin.li@rock-chips.com>
3 Date:   Sun Sep 29 14:36:34 2019 +0800
4
5 RK3288 RK3326 multi region format must be the same.
6
7 Bug:
8     Defect #223454
9
10 Change-Id: I5278bc1d914645ed2aefe62e073b4bfc115b7248
11 Signed-off-by: libin <bin.li@rock-chips.com>
```

HWC 更新至 0.59 版本即可包含该提交，版本信息可查询以下属性获得：

sys.ghwc.version 或 vendor.ghwc.version

#### 相关Redmine:

Defect #238911 : <https://redmine.rock-chips.com/issues/238911>

Defect #223454 : <https://redmine.rock-chips.com/issues/223454>

### 5.2.3.2. 显示花屏-RK\_FB-多区域格式配置错误导致显示花屏

**归属模块：**

Hardware Composer (HWC)

**关键字：**

花屏，小区域，多区域配置错误，RK3288，Android 6.0

**现象说明：**

屏幕小区域面积出现花屏，并且kernel 出现如下报错：

```
1 [ 35.100086] rk3288-lcdc lcdc0: win2_set_par:un supported format!
2 [ 35.118283] rk3288-lcdc lcdc0: win2_set_par:un supported format!
3 [ 35.118305] rk3288-lcdc lcdc0: win2_set_par:un supported format!
4 [ 35.132584] rk3288-lcdc lcdc0: win2_set_par:un supported format!
```

**解决方法：**

1. 尝试关闭HWC，查看问题是否复现：

```
1 adb shell setprop sys.hwc.compose_policy 0
2 adb shell setprop sys.hwc.enable 0
```

2. 复现问题后，尝试抓打印以下log:

```
1 //打印HWClog
2 adb shell setprop sys.hwc.log 511
3 adb shell logcat -c ;logcat > hwc.log
```

3. 确认vop配置：

```
1 adb shell cat /sys/class/graphics/fb0/disp_info
```

4. 可发现多区域配置了不同的格式，多区域只允许相同格式配置到同一图层

补丁文件位于以下目录：

[Android/hwcomposer/redmine/display\\_issues/display\\_config\\_err/rk\\_fb\\_multi\\_areas\\_config\\_err\\_use\\_display\\_err](#)

**SDK commit:**

该修改已经合并内部开发分支，相关commit message 如下：

```
1 commit 2fad96dd265624e6bc6d914cd0c8d316e51705a1
2 Author: libin <bin.li@rock-chips.com>
3 Date:   Tue Apr 7 10:48:23 2020 +0800
4
5     Multi areas format limit, multi areas not support YUV
6     format.
7
8     Bug:
9         Defect #216747
10
11    Change-Id: i45dd83ce85bf8f0640c824c6e56ef0e65ec15345
12    Signed-off-by: libin <bin.li@rock-chips.com>
```

目前对外的版本均不包含该提交，建议直接采用打补丁的方式进行修复问题。

**相关Redmine:**

Defect #216747 : <https://redmine.rock-chips.com/issues/216747>

## 5.2.4. 显示黑屏

### 5.2.4.1. 开机动画-显示黑屏-Android动画至Luncher存在黑屏

归属模块：

Hardware Composer (HWC)

关键字：

开机动画、黑屏、闪屏

现象说明：

升级后第一次启动没有问题,reset后重启进入系统会出现闪开机动画的现象，现象出现在第二次第三次开机,再后面的开机不会在出现。

异常情况：

开机logo-开机动画--android正在启动--(闪开机动画)--进入安卓桌面

正常情况：

开机logo-开机动画--android正在启动--进入安卓桌面

解决方法：

问题原因由于是因为开机过程中系统负荷比较大，开机动画作为过渡到系统luncher的动画在系统luncher还未正常送显的情况下，过早的结束开机动画，导致一段时间的系统黑屏，解决办法是在关闭开机动画流程添加合适的延迟，延迟开机动画结束的时间点，存在的风险就是开机动画时间会相应延长。

详情请参考以下目录文档：

[Android/hwcomposer/redmine/display\\_issues/display\\_black/bootanimaltion\\_black\\_android\\_to\\_luncher](https://Android/hwcomposer/redmine/display_issues/display_black/bootanimaltion_black_android_to_luncher)

SDK commit:

目前该问题仅在少数客户项目中出现，故此方法没有合并主分支，采用补丁的形式提供客户。

相关Redmine:

Defect#230712：<https://redmine.rock-chips.com/issues/230712>

Support#246319：<https://redmine.rock-chips.com/issues/246319>

## 5.2.4.2. 开机动画-显示黑屏-Kernel\_logo至Android动画存在黑屏-RK3128-4.4

**归属模块：**

Hardware Composer (HWC)

**关键字：**

开机动画、黑屏、闪屏、Android 4.4、RK3128

**现象说明：**

Kernel 动画 至 Android 动画过程中出现黑屏

**问题分析：**

SurfaceFlinger对第一帧Andorid动画进行绘制时，获取Framebuffer时正好获取到的是kernel logo所在的buffer，而这个buffer的format是RGB565，SurfaceFlinger绘制的Android动画logo格式为RGBA8888，系统在切换显示格式之前会将对应buffer的数据清空，就是这个行为导致出现黑屏。

**解决方法：**

错开kernel logo 与 Android logo申请的buffer地址，这样就不需要去清空kernel logo的数据

补丁文件请从以下目录获取：

[Android/hwcomposer/redmine/display\\_issues/display\\_black/bootanimaltion\\_black\\_android\\_to\\_launcher\\_rk3128\\_4.4](https://Android/hwcomposer/redmine/display_issues/display_black/bootanimaltion_black_android_to_launcher_rk3128_4.4)

**SDK commit:**

该补丁尚未提交内部分支，建议以补丁形式提供。

**相关Redmine:**

Defect #230712 : <https://redmine.rock-chips.com/issues/230712>

Support #246319 : <https://redmine.rock-chips.com/issues/246319>

## 5.3. 显示卡顿

### 5.3.1. 显示卡顿-4K视频播放卡顿问题

归属模块：

Hardware Composer (HWC)

关键字：

4K、视频播放、卡顿、不流畅

现象说明：

1. 播放视频卡顿，不流畅
2. 音视频不同步

解决方法：

1. 打印SurfaceFlinger 服务信息

```
1 | adb shell dumpsys SurfaceFlinger
```

2. 打印HWC 调试日志

```
1 | //Android 7.1 / 8.1
2 | adb shell setprop sys.hwc.log 511
3 | adb shell logcat
4 | //Android 9.0 / 10.0
5 | adb shell setprop vendor.hwc.log 511
6 | adb shell logcat
```

3. 查看VOP配置信息

```
1 | adb shell cat /d/dri/0/summary
```

4. 该问题最终定位为HWC软件逻辑导致视频图层没有采用overlay的模式输出，导致视频卡顿，修改相应判断逻辑后视频播放流畅。

详情可阅读以下目录文档：

[Android/hwcomposer/redmine/display\\_caton/video\\_4k](#)

SDK commit:

该修改已合并内部开发分支，对应commit message如下：

```
1 | commit a56851d28019c4200fd27b2f2da392794d1881c2
2 | Author: libin <bin.li@rock-chips.com>
3 | Date:   Thu Jan 24 10:16:41 2019 +0800
4 |
5 |     From System Product 1 Fang Xingwen:
6 |     Give priority to frame rate stability, regardless of the HDR effect.
7 |     HWC will overlaye the video instead of rolling back the GPU.
8 |
9 |     Change-Id: I30ef6c1563682b6a555b5e6e48d1faaec1d846a5
10 |    Signed-off-by: libin <bin.li@rock-chips.com>
```

HWC 更新至0.55版本即可包含该提交，版本信息可查询以下属性获得：

sys.ghwc.version 或 vendor.ghwc.version

**相关Redmine:**

Defect #179655 : <https://redmine.rock-chips.com/issues/179655>

## 5.4. 显示延迟

### 5.4.1. 显示延迟-Camera码流显示延迟问题分析

---

#### 现象说明：

客户 Camera App 在HDMI不同刷新率的分辨率下显示延迟程度不同

- 1080p 60帧：显示延迟现象可以接受
- 1080p 30帧：显示延迟现象严重，需要等待一段时间才能将实际的画面显示出来

#### 相关Redmine:

Defect #253176 : <https://redmine.rock-chips.com/issues/253176>

Defect #182656 : <https://redmine.rock-chips.com/issues/182656>

Defect #258100 : <https://redmine.rock-chips.com/issues/258100>

#### 详细功能说明文档：

问题详细解决文档如下：

[Android/hwcomposer/redmine/display\\_delay/ipc\\_or\\_camera\\_video\\_display\\_delay/](Android/hwcomposer/redmine/display_delay/ipc_or_camera_video_display_delay/)

## 5.5. 系统异常

### 5.5.1. 系统卡死

#### 5.5.1.1. 系统卡死-Monkey拷机-HDMI频繁插拔拷机导致系统卡死问题

**归属模块：**

Hardware Composer (HWC)

**关键字：**

Monkey、HDMI插拔、系统卡死

**现象说明：**

Monkey + HDMI插拔拷机场景，小概率出现系统卡死，系统无响应问题

**问题分析：**

1. 确定复现场景；
2. 分析相关日志：
  - 1) ANR日志，traces.txt文件
  - 2) addr2line命令分析目标现场堆栈信息
  - 3) 分析堆栈代码上下文
3. 定位问题；
4. 解决问题。

详情可参考以下目录文档：

[Android/hwcomposer/redmine/system\\_err/system\\_stuck/monkey\\_and\\_hdmi\\_hotplug\\_cause\\_system\\_stuck](#)

**解决方法：**

最终在代码逻辑上添加补丁修改，加入超时机制来解决死锁问题，具体修改请参考以下目录文档：

[Android/hwcomposer/redmine/system\\_err/system\\_stuck/monkey\\_and\\_hdmi\\_hotplug\\_cause\\_system\\_stuck](#)

**SDK commit:**

该修改已经合并内部开发分支：

1 | 仓库路径: hardware/rockchip/hwcomposer  
2 | 分支名: rk33/mid/6.0/develop\_drm

对应的commit message 如下：

```
1 commit 6ce613ced2b87f6de41b7fd46e7768929cb0ecb8
2 Author: libin <bin.li@rock-chips.com>
3 Date:   Mon Oct 22 17:28:23 2018 +0800
4
5     Solve HWC software BUG:
6         This BUG may cause Android crash because of operating NULL pointer
7         when hdmi hotplug at switching vop b/l.
8
9     Issue:  Defect #177719
10
11    Change-Id: Ifa4c0eaf043f2094b75ef3fe74eff7bc7dd2b2b1
12    Signed-off-by: libin <bin.li@rock-chips.com>
```

HWC 更新至 0.53 版本即可包含该提交，版本信息可查询以下属性获得：

sys.ghwc.version 或 vendor.ghwc.version

**相关Redmine:**

Defect #177719 : <https://redmine.rock-chips.com/issues/177719>

## 5.5.2. 系统崩溃

### 5.5.2.1. 系统崩溃-Monkey拷机-HDMI频繁插拔导致系统重启

归属模块：

Hardware Composer (HWC)

关键字：

Monkey、HDMI插拔、系统重启

现象说明：

Monkey + HDMI插拔拷机场景，大概率出现系统重启问题。

问题分析：

1. 确定问题复现场景；
2. 分析相关日志：
  - 1) Logcat 日志，发现存在crash打印，提供crash 堆栈信息；
  - 2) addr2line 命令反编译堆栈；
  - 3) 分析堆栈上下文
3. 定位问题；
4. 解决问题。

详情可参考以下目录文档：

[Android/hwcomposer/redmine/system\\_err/system\\_crash/monkey\\_and\\_hdmi\\_hotplug\\_cause\\_system\\_crash](#)

解决方法：

问题最终定位为访问数据重入问题导致代码执行过程访问到空指针，造成系统crash，导致重启。

SDK commit:

该修改已经合并内部开发分支：

- |   |                                    |
|---|------------------------------------|
| 1 | 仓库路径: hardware/rockchip/hwcomposer |
| 2 | 分支名: rk33/mid/6.0/develop_drm      |

对应的commit message如下：

1	commit 6ce613ced2b87f6de41b7fd46e7768929cb0ecb8
2	Author: libin <bin.li@rock-chips.com>
3	Date: Mon Oct 22 17:28:23 2018 +0800
4	
5	Solve HWC software BUG:
6	This BUG may cause Android crash because of operating NULL pointer
7	when hdmi hotplug at switching vop b/l.
8	
9	Issue: Defect #177719
10	
11	Change-Id: Ifa4c0eaf043f2094b75ef3fe74eff7bc7dd2b2b1
12	Signed-off-by: libin <bin.li@rock-chips.com>

HWC 更新至 0.53 版本即可包含该提交，版本信息可查询以下属性获得：

sys.ghwc.version 或 vendor.ghwc.version

**相关Redmine:**

Defect #177719 : <https://redmine.rock-chips.com/issues/177719>

## 5.5.2.2. 系统崩溃-休眠唤醒拷机-HDMI频繁插拔导致系统重启

**归属模块：**

Hardware Composer (HWC)

**关键字：**

Monkey、HDMI插拔、系统重启、Android 8.1 及后续版本

**现象说明：**

Logcat出现crash信息，系统重启，存在以下两个场景可复现问题：

1. 频繁切换HDMI，目的是频繁上报热插拔事件
2. 频繁休眠唤醒

**问题分析：**

1. 确定问题复现场景；
2. 分析相关日志：
  - 1) 分析logcat日志，发现crash堆栈信息
  - 2) addr2line 命令反编译堆栈信息，定位问题代码
  - 3) 分析目标代码上下文
3. 定位问题
4. 解决问题

详情请参考以下目录文档：

[Android/hwcomposer/redmine/system\\_err/system\\_crash/sleep\\_wake\\_and\\_hdmi\\_hotplug\\_ca\\_use\\_system\\_crash](#)

**解决方法：**

在对应的代码逻辑加入异常检测逻辑，若存在异常则直接执行异常处理流程，避免出现空指针crash.

**SDK commit:**

该修改已经合并入AOSP分支，相关commit message 如下：

```
1 commit b35cb8583bfab542fce46820997fc7b063ea04b
2 Author: libin <bin.li@rock-chips.com>
3 Date:   Fri Mar 13 09:26:07 2020 +0800
4
5     SurfaceFlinger : Fix crash when plug out HDMI.
6
7     Frequent HDMI unplug event may cause SurfaceFlinger
8     create hwclayer failure, further causing SurfaceFlinger
9     crash problems by NULL pointer.
10
11    Bug:
12        Defect #243456
13        Defect #240985
14
15    Signed-off-by: libin <bin.li@rock-chips.com>
```

**相关Redmine:**

Defect #243456 : <https://redmine.rock-chips.com/issues/243456>

Defect #240985 : <https://redmine.rock-chips.com/issues/240985>

## 5.6. 功能说明

### 5.6.1. HWC属性说明

#### 5.6.1.1. Overscan属性

**归属模块：**

Hardware Composer (HWC)

**关键字：**

overscan、缩放、过扫描

**现象说明：**

1. 显示超出屏幕
2. 显示不全
3. HDMI过扫描问题

缩放屏幕需求

**解决方法：**

通过设置 overscan 来实现功能，涉及属性如下：

```
1 //Android 8.1 及以下版本:  
2 persist.sys.overscan.main //主屏  
3 persist.sys.overscan.aux //副屏, 通常为热插拔屏幕  
4  
5 //Android 9.0 及以上版本:  
6 persist.vendor.overscan.main //主屏  
7 persist.vendor.overscan.aux //副屏, 通常为热插拔屏幕
```

详情请参以下文档：

[Android/hwcomposer/feature/hwc\\_property/overscan\\_property](#)

**SDK commit:**

该修改已经合并SDK主分支，相关commit message 如下：

```
1 仓库: hardware/rockchip/hwcomposer  
2  
3 分支: rk33/mid/6.0/develop_drm  
4 commit 370d183ff247b9e45e6216a389bbde030d12e58c  
5 Author: Mark Yao <mark.yao@rock-chips.com>  
6 Date: Wed Jan 11 11:43:04 2017 +0800  
7 support overscan with property setting  
8 primary display:  
9 setprop persist.sys.overscan.main "overscan 80,80,80,80"  
10 external display:  
11 setprop persist.sys.overscan.aux "overscan 80,80,80,80"  
12 rules: overscan left,top,right,bottom  
13  
14 Change-Id: 1a93e0ec8c48640cbe68dc9fdb63e30cb16b3e74  
15  
16 Signed-off-by: Mark Yao <mark.yao@rock-chips.com>
```

**相关Redmine:**

Defect #241930 : <https://redmine.rock-chips.com/issues/241930>

## 5.6.2. 预留图层功能

### 5.6.2.1. 预留Plane-快速倒车说明与调试

---

#### 功能说明:

利用预留图层功能实现车载产品的快速倒车功能，在系统启动瞬间即可显示倒车画面

#### 相关Redmine:

暂无

#### 详细功能说明文档 :

详细文档请参考：[Android/hwcomposer/feature/reserved\\_plane/vehicle\\_reverse\\_video/](Android/hwcomposer/feature/reserved_plane/vehicle_reverse_video/)

## 6. Gralloc

### 6.1. Gralloc-CPU读写graphics\_buffer-显示有异常横条纹

现象说明：

将 graphics\_buffer(比如来自 camera) 送显示, 出现很多异常的细横条纹, 如下图：



相关Redmine：

Defect #253054 : <https://redmine.rock-chips.com/issues/253054>

解决方法：

要求客户检查整体流程中 是否有 使用 CPU 读写 graphics\_buffer, 比如用 CPU 实现 graphics\_buffer 的拷贝. 预期是有的.

再要求客户确认 "在 CPU 对 graphics\_buffer 所有读写操作之前, 调用了 gralloc\_module 的 lock() 方法; 在所有读写操作之后, 调用了 gralloc\_module 的 unlock() 方法".

## 6.2. 进程内存异常-HWC内存占用大

**归属模块：**

libgralloc(utgard)

**关键字：**

HWC进程内存占用，Utgard

**现象说明：**

1. 最初是在HWC处发现内存占用过大，指令是：showmap pgrep composer
2. 查看该进程的内存，发现主要是/dev/dri/renderD128这个节点占用
3. 此现象只在utgard的平台出现

**解决方法：**

1. 搜索相关模块，发现此节点在gralloc处打开
2. 在HWC处加代码，发现只要new GraphicBuffer就会有内存占用
3. 此现象较为反常，理论上在map时，才会将内存映射出来
4. 跟进gralloc模块中的alloc流程，发现其new GraphicBuffer时有去rockchip\_bo\_map。经讨论后，此问题是历史遗留问题，可以去掉。
5. 因此最终的解决方案是：Utgard 的libgralloc在new GraphicBuffer 时多加了map的操作，将该操作删去。

**SDK commit:**

已提交至SDK，相关commit 信息如下：

```
1 仓库: hardware/rockchip/libgralloc/utgard
2 分支: rk/rk33/mid/10.0/develop
3 Commit message:
4 commit e2406910c089cabd14f7dec9eb77d03da772b265
5 Author: zhixiong Lin <zhixiong.lin@rock-chips.com>
6 Date: Thu Dec 5 16:14:06 2019 +0800
7 Remove calling to rockchip_bo_map() when allocating or importing graphic
buffers
8
9 It was useless and occupied extra virtual memory.
10
11 Change-Id: Icc688d14fcabf1bf9a99a4ca885b72e22c6018c5
12 Signed-off-by: zhixiong Lin <zhixiong.lin@rock-chips.com>
```

**相关Redmine:**

无

## 7. RGA

---

### 7.1. 咨询问题

#### 7.1.1. 咨询问题-RGA功能咨询

**归属模块：**

librga

**关键字：**

RGA，格式支持，缩放倍数，硬件功能查询

**现象说明：**

客户关于各soc 平台RGA 功能支持以及详细功能的咨询类问题。比如格式支持，缩放倍数等相关问题咨询。

**问题分析：**

已整理详细的查询文档，针对soc以及RGA硬件类型分析，根据客户问题，可自行查表回复。

**解决方法：**

文档在当前 git 工程的如下位置：

[android\RGA\document](#)

**相关Redmine:**

无

## 7.1.2. 咨询问题-RGA性能问题咨询

**归属模块：**

librga

**关键字：**

RGA，性能咨询

**现象说明：**

关于RGA的性能咨询

**问题分析：**

已整理详细的查询文档，根据客户问题，可自行查表回复。

**解决方法：**

文档在当前 git 工程的如下位置：

[android\RGA\document](#)

也可以通过最新版本librga IM2D的querystring 接口进行查询，详细API说明参见压缩包 librga\_im2d.tar.gz 内readme。在本git 工程如下路径可以找到新版本librga：

[Android/RGA/demand\\_support/release\\_librga](#)

**相关Redmine:**

无

### 7.1.3. 咨询问题-RGA耗时是否正常

**归属模块：**

librga

**关键字：**

耗时

**现象说明：**

客户对RGA 的耗时有疑虑。

**解决方法：**

优先考虑使用RGA的 demo 测试耗时，以排除客户本身修改或者使用不当导致的耗时异常。

具体耗时是否正常，需要根据“RGA\_All\_Version comparision.xlsx” 文档先确定rga 硬件类型是1代还是2代。以1080p, 300M aclk为例子，理论时间为：

RGA1:  $1920 \times 1080 / (300 \times 1000000) = 0.0069s$

RGA2:  $1920 \times 1080 / (300 \times 1000000 \times 2) = 0.0034s$

这里理论值的计算以数据copy为例子，如果是缩放的情况，耗时计算以源数据或者目标数据的最大值来计算，而且需要加上缩放算法耗时，会比copy的耗时更长。实际场景应用中，考虑到情况复杂，可能存在ddr 不是很充足、使用了虚地址等情况，耗时会比理论上长一些存在浮动，这都是正常。

下图展示了使用RK3399 在使用rga demo 在DDR 800M , rga aclk 400M情况下的测试数据：

数据 RGBA8888 ; DDR=800M;ACLK=400M ;	耗时 ( ms )
3840x2160->3840x2160	11.3
3840x2160->1920x1080	11.4
1920x1080->3840x2160	13.3

更详细参考数据可以参考RGA\_benchmark 文档，分别在不同的设备上详细测量了RGA1 以及 RGA2的耗时数据。

可以使用Dr-G 进行快速测试，命令如下：

```
1 | dr-g -rga perf
```

**补丁说明:**

RGA\_benchmark 文档路径：

[android\RGA\document](#)

**相关Redmine:**

Defect #238654 : <https://redmine.rock-chips.com/issues/238654>

## 7.2. 需求支持

### 7.2.1. 需求支持-客户SDK需要添加librga

**归属模块：**

librga

**关键字：**

版本更新，集成

**现象说明：**

librga 版本太旧问题，较早期sdk 中在hardware/rockchip/目录未找到librga模块。

**问题分析：**

1. 尝试更新SDK。
2. 如果不愿意更新SDK，可以尝试“解决办法”中提到的压缩包，在SDK/hardware/rockchip/ 目录下解压或者替换。

**解决方法：**

新版本带IM2D 接口的librga，支持Android 7.1 即以上，包含Linux，NDK编译支持，详细使用说明参看压缩包 librga\_im2d.tar.gz 内readme，在本git 工程如下路径可以找到：

[Android/RGA/demand\\_support/release\\_librga](#)

旧版接口librga 更新压缩包按Android版本划分，在本git 工程如下路径可以找到

[Android/RGA/demand\\_support/release\\_librga](#)

Android 7.1及以下：

librga\_for\_android\_7.1.tar.gz

Android 8.1:

librga\_for\_Android\_8\_1.tar.gz

Android 9.0:

librga\_for\_Android\_9\_0.tar.gz

**相关Redmine:**

无

## 7.2.2. 需求支持-RGA用例使用物理地址说明

**归属模块：**

librga

**关键字：**

物理地址，MMU

**现象说明：**

某些客户需要用物理地址进行RGA的操作。且使用物理地址，RGA处理效率最高，这里做一下简要说明。

**解决方法：**

( 1 ) 使用librga 旧版本接口：

物理地址需要配置给rga\_info 结构体中phyAddr 成员即可，并且需要关闭MMU，不要对mmuFlag 成员进行赋值。其他操作和虚地址，fd 方式一致。

( 2 ) 使用librga 新版本IM2D接口：

调用配置wrapbuffer\_physicaladdr 函数即可。详细使用说明参看压缩包 librga\_im2d.tar.gz 内 readme，在本git 工程如下路径可以找到新版本librga：

[Android/RGA/demand\\_support/release\\_librga](#)

**SDK commit:**

无

**相关Redmine:**

无

## 7.3. RGA运行问题

### 7.3.1. librga问题

#### 7.3.1.1. 色彩偏差-RGA输出数据色域空间问题

归属模块：

librga

关键字：

色域空间，YUV，颜色偏差

现象说明：

通常出现在YUV2RGB，或者RGB2YUV的过程中。

YUV2RGB的过程中RGA 默认色域空间设置为BT 601 limit range。颜色范围[16-235]。

以RGBA值000000AA（黑色）为例子，会被转化为101010AA（深灰），从而出现色差。

RGB2YUV 的过程中RGA默认色域空间设置为BT601 full range 则不会出现色差的问题，但是可能也有会有客户期望BT601 limit range的效果。

解决方法：

( 1 ) 使用librga 新版本IM2D接口：

使用imcvtcolor 接口时配置mode 参数，传入对应的IM\_COLOR\_SPACE\_MODE 类型即可。详细使用说明参看压缩包 librga\_im2d.tar.gz 内readme，在本git 工程如下路径可以找到新版本librga：

[Android/RGA/demand\\_support/release\\_librga](#)

( 2 ) 使用librga 旧版本接口：

针对客户需求，可以修改RGA 默认色域空间的设置(yuvToRgbMode)。主要支持以下四种：

bypass 对应修改值 0x0 << 0 或 0x0 << 4

bt601 limit range 对应修改值 0x1 << 0 或 0x1 << 4

bt601 full range 对应修改值 0x2 << 0 或 0x2 << 4

bt709 limit range 对应修改值 0x3 << 0 或 0x3 << 4

补丁说明：

YUV2RGB 默认BT601 limit range 提供以下两个patch:

[Android/RGA/librga/colorspace\\_config](#)

**BT601 full range:yuv\_to\_rgb\_601\_full.patch**

**BT709 limit range: yuv\_to\_rgb\_709\_limit.patch**

rgb->yuv 默认bt601 full range 提供以下两个patch:

**BT601 limit range: rgb\_to\_yuv\_601\_limit.patch**

**BT709 limit range: rgb\_to\_yuv\_709\_limit.patch**

**SDK commit:**

目前该修改还未合并入SDK，后续会安排提供相关接口供客户调用。

**相关Redmine:**

无

### 7.3.1.2. 调用失败-YUV宽度对齐约束条件不合理调用失败

**归属模块：**

librga

**关键字：**

对齐

**现象说明：**

logcat 中可以发现以下报错：

```
1 | "err wstride is not align to 8 or yuv not align to 2"
```

**问题分析：**

1. 请确认YUV数据虚宽实宽是否已经2对齐
2. 如果YUV 数据虚宽实宽已经2对齐，wstride 虚宽8对齐不是必须的，可以打上补丁  
`remove_assert_for_wstride_align_to_8.patch`，重新编译librga 验证

**解决方法：**

Patch以及相关文件在本git 工程如下目录：

[Android/RGA/librga/wstride\\_align\\_err\\_cause\\_call\\_librga\\_fail](#)

`remove_assert_for_wstride_align_to_8.patch`

**SDK commit:**

Android 10.0:

```
1 仓库: hardware/rockchip/librga
2 分支: rk/rk33/mid/10.0/develop
3 Commit message:
4 commit 3c907fbb04338069ec07dc615f390fab36f673e8
5 Author: Li Huang <putin.li@rock-chips.com>
6 Date: Thu Jan 16 14:52:56 2020 +0800
7
8 Remove assert about wstride align to 8
9
10 Change-Id: 1e94c4497ae3dbf019dab8c3c8eae704f92c8e854
11 Signed-off-by: Li Huang <putin.li@rock-chips.com>
```

Android 其余版本还未合并开发分支，可参考Android 10.0修改直接提供补丁

Linux 已合并内部开发分支，commit message如下：

```
1 仓库: external/linux-rga
2 分支: rk/master
3 commit 87737c0b4425e2372086385de7fe9c54b9dbbace
4 Author: Putin Lee <putin.li@rock-chips.com>
5 Date: Thu May 14 20:37:03 2020 +0800
6
7 Remove assert about wstride align to 8
8
9 Signed-off-by: Putin Lee <putin.li@rock-chips.com>
10 Change-Id: I26e045fb4a4faeeab83181797bb2f1562bed6a5e
```

**相关Redmine:**

无

## 7.3.2. kernel 相关问题

### 7.3.2.1. 3288-4.19内核-RGA出现TimeOut问题

#### 现象说明

3288上调用RGA的所有场景卡顿，查看dmesg有RGA TimeOut相关打印。尝试使用RGA Demo进行测试，同样存在问题。且与DDR无关，定频后仍有问题。

相关log：

```
1 | [ 1315.728115] rga: Rga sync pid 1527 wait 1 task done timeout
```

#### 详细问题说明文档：

问题定位为RGA CLK异常，详细的文档请参考：

[3288-4.19内核-RGA出现TimeOut问题](#)

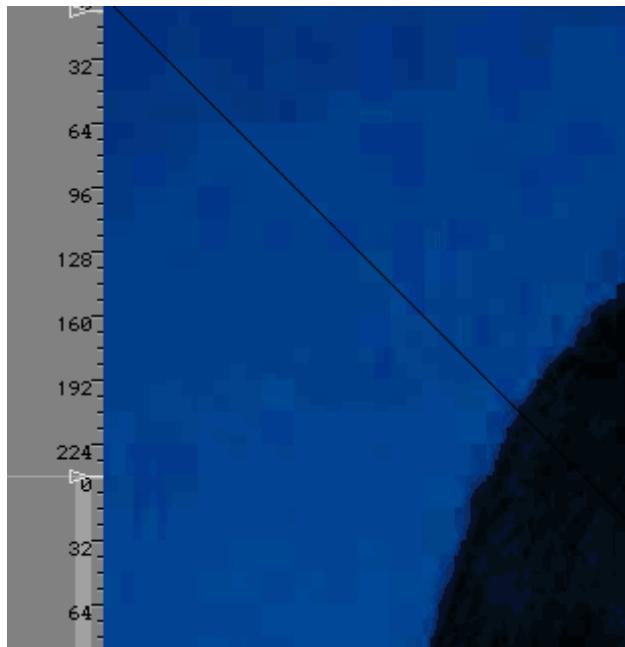
#### 相关redmine

无

### 7.3.2.2. RGB888格式奇数宽输出存在黑色斜线的问题

#### 现象说明

RGA目标Dst格式为RGB888 或者 BGR888 时，且输出图像的宽为奇数宽时，图像上会出现非预期的黑色斜线。如下图：



#### 详细问题说明文档：

问题定位为RGA CLK异常，详细的文档请参考：

[Android/RGA/kernel/output\\_error\\_issue/RGB888\\_odd\\_width\\_cause\\_black\\_line/](#)

#### 相关redmine

无

### 7.3.2.3. 显示黑线-RGA驱动BUG导致RGA合成输出图像随机黑线

现象说明：

使用hwc合成时，进入某些APK场景，会有黑色直线问题。如下图：



详细问题说明文档：

问题定位为 在调用RGA时，源数据同时传入fd值与viraddr导致的，详细的文档请参考：

[Android/RGA/kernel/input\\_fd\\_and\\_viraddr\\_issue/](https://redmine.rock-chips.com/issues/239256)

相关redmine

<https://redmine.rock-chips.com/issues/239256>

#### 7.3.2.4. 内存重入-RGA内存访问越界

**归属模块：**

librga

**关键字：**

冲内存，kernel 4.4

**现象说明：**

通常是camera 数据被冲，kernel 出现 Bad page map。且camera 关闭RGA 使用cpu后不会复现该问题。内存数据可以看到类似图像数据的结构。

**解决方法：**

kernel 需要打上如下patch：

Patch 位于本git 工程如下目录：

[Android/RGA/kernel/out\\_of\\_bounds\\_memory\\_access](#)

**cf5f6c0.diff.zip**

**SDK commit:**

Gerrit 提交如下：

<https://10.10.10.29/c/rk/kernel/+/77455>

该补丁尚未合并kernel 主分支，由于可能存在未知的风险，故未合并。

**相关Redmine:**

Defect #229739 : <https://redmine.rock-chips.com/issues/229739>

## 8. Frameworks

### 8.1. RK3288 双屏拔插物理输入设备APP出现闪黑的问题

#### 现象说明

在双屏的情况下，部分APP比如计算器，Setting等，拔插物理键盘或者蓝牙鼠标等物理输入设备，会出现闪黑的现象。

相关log：

```
1 | 06-28 04:13:26.945  418  490 I ActivityManager: override config changes=30
{1.0 ?mcc?mnc [en_us] ldltr sw720dp w1280dp h648dp 240dpi lrg long land
finger -keyb/v/h -nav/h winConfig={ mBounds=Rect(0, 0 - 1920, 1008)
mAppBounds=Rect(0, 0 - 1920, 1008) mWindowingMode=fullscreen
mActivityType=undefined} s.24} for displayId=0
```

详细问题说明文档：

[Android/frameworks/redmine/display\\_issue/phy\\_input\\_device\\_unplug\\_cause\\_splash\\_screen/](https://Android/frameworks/redmine/display_issue/phy_input_device_unplug_cause_splash_screen/)

相关redmine

<https://redmine.rock-chips.com/issues/258309#change-2401068>

<https://redmine.rock-chips.com/issues/244750>

## 9. GPU硬件报错排查流程

此类问题较为常见，表现为卡死，花屏等问题。GPU 驱动输出硬件状态异常信息。

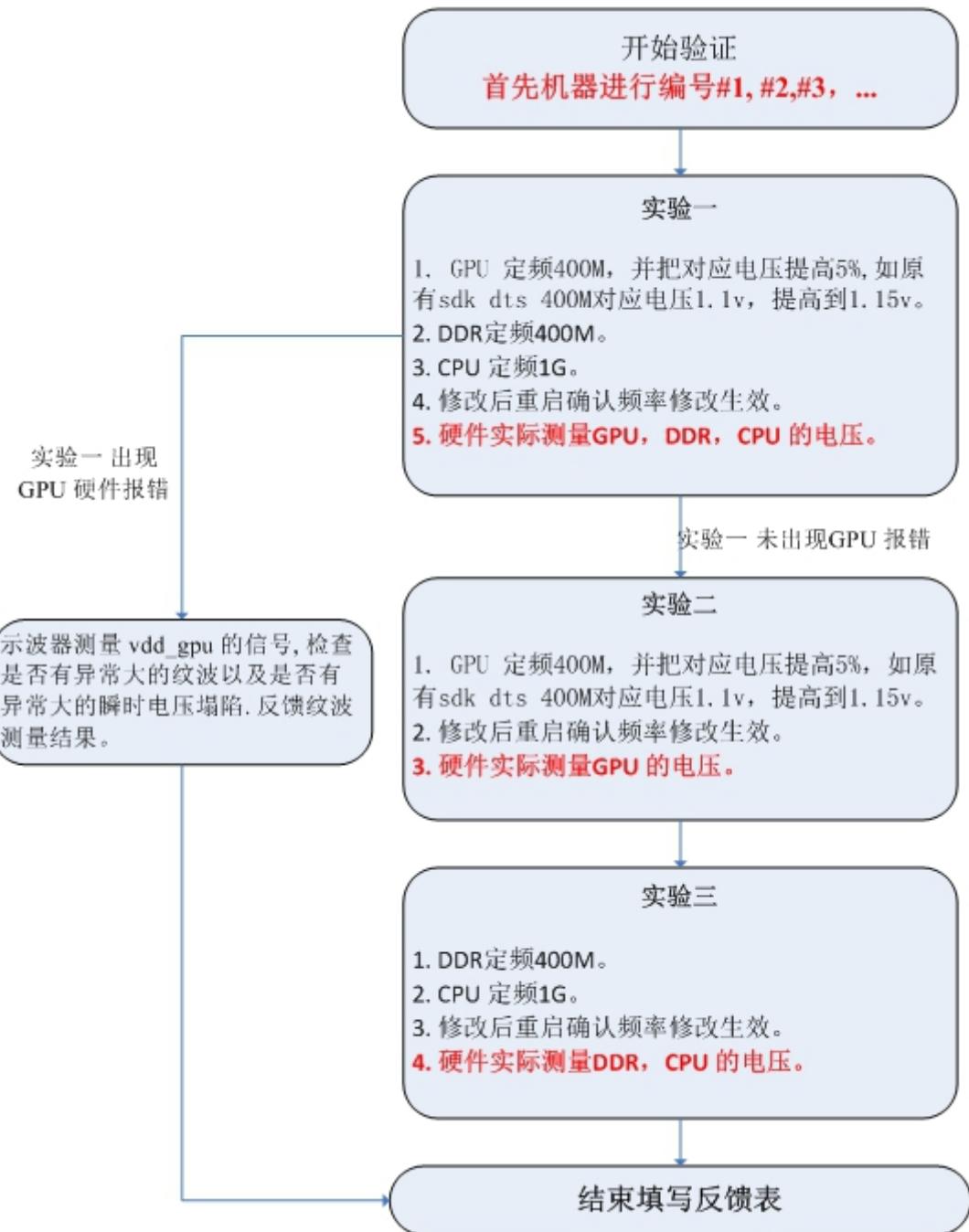
kernel\_log 中有 fence timeout, 以及 mali的报错 log, 诸如:

```
1 mali ff9a0000.gpu: Failed to soft-reset GPU (timed out after 500 ms), now
attempts a hard reset
2 ...
3 mali ff9a0000.gpu: JS: Job Hard-Stopped (took more than 50 ticks at 100
ms/tick)
4 mali ff9a0000.gpu: JS: Job has been on the GPU for too long
5 (JS\_RESET\_TICKS\_SS/DUMPING timeout hit). Issueing GPU soft-reset to
resolve.
6 mali ff9a0000.gpu: Preparing to soft-reset GPU: Waiting (upto 3000 ms) for
all
7 jobs to complete soft-stop
8 mali ff9a0000.gpu: Resetting GPU (allowing up to 500 ms)
9 mali ff9a0000.gpu: Register state:
10 ...
```

PVR GPU 如下log:

```
[ 312.513328] PVR_K: 157: -----[ RGX summary ]-----
[ 312.513467] PVR_K: 157: RGX BVNC: 5.9.1.46
[ 312.513508] PVR_K: 157: RGX Device State: Active
[ 312.513542] PVR_K: 157: RGX Power State: ON
[ 312.513580] PVR_K: 157: BIF0 - OK
[ 312.513613] PVR_K: 157: BIF1 - OK
[ 312.513653] PVR_K: 157: RGX FW State: OK (HWRState 0x00000001)
[ 312.513703] PVR_K: 157: RGX FW Power State: RGXFWIF_POW_ON (APM enabled: 142 ok, 0 denied, 0 other, 142 total)
[ 312.513752] PVR_K: 157: RGX FW OS 0 State: 0x00000001 (Active: Yes, Freelists: Not Ok)
[ 312.513858] PVR_K: 157: Number of HWR: GP(0/0+0), 2D(0/0+0), TA(0/0+0), 3D(1/1+0), CDM(0/0+0), FALSE(0,0,0,0,0)
[ 312.513906] PVR_K: 157: DM 0 (GP)
[ 312.513951] PVR_K: 157: DM 1 (HWRflags 0x00000000)
[ 312.513993] PVR_K: 157: DM 2 (HWRflags 0x00000000)
[ 312.514034] PVR_K: 157: DM 3 (HWRflags 0x00000000)
[ 312.514087] PVR_K: 157: Recovery 1: PID = 1093, frame = 0, HWRTData = 0x00000000, EventStatus = 0x00000010, Guilty Lockup
[ 312.514144] PVR_K: 157: CRTimer = 0x000003475E23, OSTimer = 312.512793804, CyclesElapsed = 40089856
[ 312.514198] PVR_K: 157: PreResetTimeInCycles = 20736, HWResetTimeInCycles = 16896, TotalRecoveryTimeInCycles = 37632
[ 312.514249] PVR_K: 157: DM 4 (HWRflags 0x00000000)
[ 312.514289] PVR_K: 157: -----[ RGX registers ]-----
```

此类问题判定为硬件不稳定。按如下流程做实验，然后填写实验反馈表。把反馈表发给GPU小组。



### 实验反馈表格：

实验类型	现象是否有问题	几台拷机	拷机时长	有问题的机器编号	出现问题是否固定编号	实际模块电压		实际频率
						GPU(v)	DDR(v)	
实验一						CPU(v)		
						GPU(v)		
						DDR(v)		
实验二						CPU(v)		
						GPU(v)		
						DDR(v)		
实验三						CPU(v)		
						GPU(v)		
						DDR(v)		

实验结果反馈表 1

# 10. GPU 平台适配

## 10.1. GPU驱动初始化问题

### 10.1.1. GPU驱动-初始化失败-mali\_ko与客户编译的kernel不兼容

归属模块：

mali device driver

关键字：

mali\_kbase.ko 加载失败; 开机动画不显示; SurfaceFlinger 初始化失败

现象说明：

RK3326 的 SDK 上, 客户对 kernel 有定制修改, 特别是修改 config, 比如:

<https://redmine.rockchip.com.cn/issues/220290>

中将 CONFIG\_VLAN\_8021Q 修改为 y, 导致 "编译得到的 kernel" 和 "SDK 中 prebuilt 的 mali\_kbase.ko(mali\_device\_driver)" 不兼容, init 对 mali\_kbase.ko 的 insmod 操作将失败, 可能的对应的 kernel 报错 log :

```
1 [ 4.682419] init: finit_module for "/vendor/lib/modules/mali_kbase.ko"  
failed: Invalid argument
```

mali GPU 设备未被加载, mali\_driver 无法成功初始化, 对应的 logcat log :

```
1 E mali : Failed creating base context during opening of kernel driver.  
2 E mali : Kernel module may not have been loaded
```

依赖 mali\_driver 的 SurfaceFlinger(SF) 初始化失败, 认为是 fatal, crash :

```
1 W libEGL : eglInitialize(0x7b5a235380) failed (EGL_NOT_INITIALIZED)  
2 ...  
3 W SurfaceFlinger: no suitable EGLConfig found, trying a simpler query  
4 F SurfaceFlinger: no suitable EGLConfig found, giving up
```

解决方法：

在 kernel/ 下, 使用 0001-TMP-arm64-rockchip\_defconfig-set-CONFIG\_MALI\_BIFROST.patch .

该 patch 将 RK3326 的 GPU device driver "kernel/drivers/gpu/arm/bifrost" build in 到 kernel 或 boot 镜像, 从而解决上面的问题.

标识 Mali Bifrost GPU 被成功加载的 kernel log :

```
1 [ 1.514036] mali ff400000.gpu: Probed as mali0
```

对应的设备文件是 /dev/mali0.

在 RK3399 的某些 SDK 上, 也可能存在类似的问题.

此时, 可以使用 0001-TMP-arm64-rockchip\_defconfig-set-CONFIG\_MALI\_MIDGARD.patch

补丁说明:

上述两个 patch 文件都在以下目录：

[Android/gpu/mali/mali\\_device\\_driver\\_init\\_err](Android/gpu/mali/mali_device_driver_init_err)

**相关Redmine:**

Defect #220290 : <https://redmine.rock-chips.com/issues/220290>

## 10.1.2. GPU驱动-初始化失败-无法获取regulator\_gpu

**归属模块：**

midgard device driver

**关键字：**

mali ffa30000.gpu

**现象说明：**

kernel 4.4 和 4.10, RK3288(W), "midgard device driver" probe "mali 设备" 失败, 依赖 mali driver 的 SurfaceFlinger 无法正常启动.

kernel log 中有多次提示 mali GPU device 无法获取 依赖的 regulator\_gpu 的报错 log, 比如:

```
1 [ 0.564010] mali ffa30000.gpu: Failed to get regulator
2 [ 0.564031] mali ffa30000.gpu: Power control initialization failed
3 ...
4 [ 2.276002] mali ffa30000.gpu: Failed to get regulator
5 [ 2.276032] mali ffa30000.gpu: Power control initialization failed
6 ...
7 [ 2.355706] mali ffa30000.gpu: Failed to get regulator
8 [ 2.355720] mali ffa30000.gpu: Power control initialization failed
9 ...
10 [ 2.371909] mali ffa30000.gpu: Failed to get regulator
11 [ 2.371930] mali ffa30000.gpu: Power control initialization failed
12 ...
13 [ 2.387183] mali ffa30000.gpu: Failed to get regulator
14 [ 2.387206] mali ffa30000.gpu: Power control initialization failed
```

且 "没有" 提示 probe 成功的 log :

```
1 | gpu: Probed as malio
```

RK3399 和 RK3326 上也可能出现类似的异常.

**问题分析：**

无

**解决方法：**

要求 FAE 或 kernel PM组 配合客户保证 "regulators 配置正确, 且初始化正确.". 后再查看是否还有该问题。

**SDK commit:**

无

**相关Redmine:**

Defect #221844 : <https://redmine.rock-chips.com/issues/221844>

Defect #215923 : <https://redmine.rock-chips.com/issues/215923>

### 10.1.3. GPU频率电压适配

#### 10.1.3.1. GPU异常-DTS配置问题导致GPU异常

归属模块：

GPU device driver

关键字：

mali ffa30000.gpu

现象说明：

1. RK3288-CGT版本的cpu 添加补丁后log会报gpu错误

```
1 "mali ffa30000.gpu: Continuing without Mali clock control"
2 "mali ffa30000.gpu: failed to runtime resume device: -22"
```

2. 显示界面切换时会闪屏

问题分析：

无

解决方法：

1. 根据客户描述，更新补丁后出现的问题，因此可以猜测是否是打补丁步骤有问题；
2. 根据打印的log追踪源码，该打印是GPU的驱动，在读取dts配置的电压时，获取到的非驱动想要的clock，所以报错；
3. 最终客户确认到，是自己打补丁时的问题，详见redmine #221844的处理流程；

SDK commit:

无

相关Redmine:

Defect #221844 : <https://redmine.rock-chips.com/issues/221844>

### 10.1.3.2. GPU频率配置-固定GPU频率导致系统不能启动

**归属模块：**

GPU , power

**关键字：**

固定GPU频率

**现象说明：**

1. 在RK3399 9.0 dts配置固定某个GPU的频率，系统不能正常启动；
2. logcat信息带有android.hardware.power服务相关堆栈:

[image]

3. 设置两个及两个以上频率，能够正常启动。

**问题分析：**

无

**解决方法：**

1. 该问题的根本原因是：该固件HAL层的power模块有bug，在gpu\_available\_freqs配置成1个时无法被split，进一步导致系统挂掉。
2. 此问题不属于GPU模块问题，最终由产品部出解决补丁。
3. 由于是涉及GPU电压配置，才出现该现象，特此做个记录。

**补丁说明：**

1. 补丁已提交以下路径：

[Android/gpu/mali/only\\_one\\_gpu\\_clk\\_cause\\_sys\\_init\\_err](#)

**0001-rk3399-fix-single-freq-crash.patch**

2. 请在hardware\rockchip\power\ 路径打上该补丁，编译后，进行测试

**相关Redmine:**

Defect #239815 : <https://redmine.rock-chips.com/issues/239815>

### 10.1.3.3. 系统卡死-系统花屏-RK3368串口日志出现FenceTimeOut

归属模块：

PVR

关键字：

频率电压，fence timeout

现象说明：

通常表现为卡死，花屏等问题。内核可以在复现时kernel看到以下log：

3000ms 的Fence timeout：

fence timeout on [fffffc02cfb0200] after 3010ms

硬件检测信息：

```
[ 312.513328] PVR_K: 157: -----[ RGX summary ]-----
[ 312.513467] PVR_K: 157: RGX BVNC: 5.9.1.46
[ 312.513508] PVR_K: 157: RGX Device State: Active
[ 312.513542] PVR_K: 157: RGX Power State: ON
[ 312.513580] PVR_K: 157: BIF0 - OK
[ 312.513613] PVR_K: 157: BIF1 - OK
[ 312.513653] PVR_K: 157: RGX FW State: OK (HWRState 0x00000001)
[ 312.513703] PVR_K: 157: RGX FW Power State: RGXFWIF_POW_ON (APM enabled: 142 ok, 0 denied, 0 other, 142 total)
[ 312.513752] PVR_K: 157: RGX FW OS 0 State: 0x00000001 (Active: Yes, Freelist: Not Ok)
[ 312.513858] PVR_K: 157: Number of HWR: GP(0/0+0), 2D(0/0+0), TA(0/0+0), 3D(1/1+0), CDM(0/0+0), FALSE(0,0,0,0)
[ 312.513906] PVR_K: 157: DM 0 (GP)
[ 312.513951] PVR_K: 157: DM 1 (HWRflags 0x00000000)
[ 312.513993] PVR_K: 157: DM 2 (HWRflags 0x00000000)
[ 312.514034] PVR_K: 157: DM 3 (HWRflags 0x00000000)
[ 312.514087] PVR_K: 157: Recovery 1: PID = 1093, frame = 0, HWRTData = 0x00000000, EventStatus = 0x00000010, Guilty Lockup
[ 312.514144] PVR_K: 157: CRTimer = 0x000003475E23, OSTimer = 312.512793804, CyclesElapsed = 40089856
[ 312.514198] PVR_K: 157: PreResetTimeInCycles = 20736, HWResetTimeInCycles = 16896, TotalRecoveryTimeInCycles = 37632
[ 312.514249] PVR_K: 157: DM 4 (HWRflags 0x00000000)
[ 312.514289] PVR_K: 157: -----[ RGX registers ]-----
```

问题分析：

无

解决方法：

优先要求定频，GPU、DDR、CPU都定频进行测量。如果还有问题详见参考文档处理。

参考文档

[Android/gpu/pvr/fence\\_time\\_out\\_issues](#)

**Fence\_time\_out\_approach.md**

补丁说明：

Patch以及相关文件在本git工程如下目录：

[Android/gpu/pvr/fence\\_time\\_out\\_issues](#)

1. powervr.ini

2. pvr\_power\_always\_on.patch

SDK commit:

powervr.ini文件需要针对应用指定修改，不建议全局开启，否则会造成一定性能损失。Power always on 修改只限不关心功耗的客户，同样无法合入SDK

相关Redmine:

Defect #208019 : <https://redmine.rock-chips.com/issues/208019>

Defect #219843 : <https://redmine.rock-chips.com/issues/219843>

Defect #225851 : <https://redmine.rock-chips.com/issues/225851>

## 10.2. GPU版本兼容问题

### 10.2.1. 版本兼容-ko与so版本不兼容导致Android启动失败

**归属模块：**

GPU

**关键字：**

启动失败，版本兼容

**现象说明：**

客户更新代码后无法进入Android系统

**问题分析：**

1. 客户有做过SDK工程更新的操作；
2. logcat信息带有DDK版本与device driver版本不一致的相关提示信息：

```
1 | E mali : ERROR: The DDK (built for 0x750 r0p0 status range [1..1]) is  
2 | not compatible with this Mali GPU device, /dev/mali0 detected as 0x750  
3 | r1p0 status 0.  
2 |  
3 | E mali : Failed creating base context during DDK compatibility check."
```

3. 更新时，是部分代码更新，如：只更新kernel。

**解决方法：**

1. 建议客户，重新建立一个目录，同步新工程编译。然后将客户自己的修改pick上去。
2. 牢记，GPU 相关模块版本升级，需要kernel、mali so与gralloc同步升级。

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #232364 : <https://redmine.rock-chips.com/issues/232364>

Defect #214427 : <https://redmine.rock-chips.com/issues/214427>

Defect #201838 : <https://redmine.rock-chips.com/issues/201838>

## 10.2.2. 版本兼容-so与libgralloc不兼容导致Android启动失败

**归属模块：**

mali driver

**关键字：**

gralloc private structure does not match

**现象说明：**

客户在升级 RK3399 7.1 industry SDK 某个版本之后, SurfaceFlinger 遇到 fatal 错误, crash.

crash 之前, logcat log 中有如下的 mali 报错：

```
1 | E mali_so : invalid handle: version=12, numInts=31, numFds=2, magic=1a00
2 | E mali_so : invalid handle: right version=12, numInts=35, numFds=2, magic=123
45678
3 | E mali_so : Runtime and build version of gralloc private structure does not m
atch for handle 0x7d77242820. Errors will likely follow.
```

**问题分析：**

无

**解决方法：**

该异常 通常是 "hardware/rockchip/libgralloc/ 下的源码" 和  
"vendor/rockchip/common/gpu/MaliT860/ 下的两个 libGLES\_mali.so" 的版本不匹配导致.

按照正常流程 release 的 SDK, 不会有这样的问题; 通常是 客户在 升级 SDK 的时候引入.

所以, 要请 客户 确认其产品的 Android 源码工程中的 "所有" 子 project 都被 rebase 或者 merge 到 预期的 SDK 的新版本中.

尤其是如下几个 子 project :

1. kernel/
2. hardware/rockchip/libgralloc/
3. vendor/rockchip/common/

RK3288, RK3326 上同样可能出现这样的异常, 解决方法也相同.

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #213361 : <https://redmine.rock-chips.com/issues/213361>

### 10.2.3. 版本兼容-RK3288不支持Android8.1及更高Android版本说明

**归属模块：**

midgard driver

**关键字：**

RK3288

**现象说明：**

客户尝试在 RK3288 SoC 上使用 RK Android 8.1 及之后的 SDK, 但是无法正常启动.

logcat log 中有如下的报错：

```
1 | E/mali ( 323): ERROR! The DDK (built for 0x750 r1p0 status range [0..15]) is not compatible with this Mali GPU device, /dev/mali0 detected as 0x750 r0 ep0 status 1.
```

**问题分析：**

无

**解决方法：**

告知客户 "RK Android 8.1 及之后的 SDK, 不再支持 RK3288", 仅支持 RK3288W.

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #219114 : <https://redmine.rock-chips.com/issues/219114>

Defect #197183 : <https://redmine.rock-chips.com/issues/197183>

## 10.3. Kernel崩溃问题

### 10.3.1. Kernel崩溃-vdd\_gpu 未配置为 always-on

归属模块：

midgard device driver

关键字：

regulator-always-on, panic

现象说明：

RK3399 9.0, 为进一步降低功耗, 在 dts 中, 将 vdd\_gpu 节点的 "regulator-always-on;" 属性注释掉, 导致 kernel panic :

```
1 [ 2.217095] Kernel panic - not syncing: panic_on_set_idle set ...
2 [ 2.217095]
3 [ 2.218765] CPU: 0 PID: 207 Comm: insmod Not tainted 4.4.167 #149
4 [ 2.219306] Hardware name: rockchip,rk3399-mid (DT)
5 [ 2.219744] Call trace:
6 [ 2.219991] [<ffffffff8008089f70>] dump_backtrace+0x0/0x1ec
7 [ 2.220482] [<ffffffff800808a170>] show_stack+0x14/0x1c
8 [ 2.220942] [<ffffffff80083ad598>] dump_stack+0x94/0xbc
9 [ 2.221399] [<ffffffff800816dc8>] panic+0xec/0x240
10 [ 2.221834] [<ffffffff8008443e68>] rockchip_pmu_set_idle_request+0x194/0x1
ac
11 [ 2.222449] [<ffffffff80084440ec>] rockchip_pd_power+0x208/0x250
12 [ 2.222973] [<ffffffff800844414c>] rockchip_pd_power_on+0x18/0x28
13 [ 2.223510] [<ffffffff8008521844>] genpd_power_on.part.8+0x28/0x64
14 [ 2.224044] [<ffffffff8008521cdc>] __genpd_poweron.part.10+0xf0/0xfc
15 [ 2.224601] [<ffffffff8008522700>] genpd_dev_pm_attach+0x170/0x1b4
16 [ 2.225138] [<ffffffff8008516678>] dev_pm_domain_attach+0xc/0x14
17 [ 2.225662] [<ffffffff800850fca8>] platform_drv_probe+0x38/0xa8
18 [ 2.226185] [<ffffffff800850dff4>] driver_probe_device+0x188/0x26c
19 [ 2.226718] [<ffffffff800850e124>] __driver_attach+0x4c/0x90
20 [ 2.227220] [<ffffffff800850c460>] bus_for_each_dev+0x9c/0xbc
21 [ 2.227722] [<ffffffff800850dac4>] driver_attach+0x20/0x28
22 [ 2.228200] [<ffffffff800850d6e4>] bus_add_driver+0x198/0x1dc
23 [ 2.228698] [<ffffffff800850ecc0>] driver_register+0x98/0xd0
24 [ 2.229198] [<ffffffff800850fc1c>] __platform_driver_register+0x48/0x50
25 [ 2.230332] [<ffffffff8000c06018>] kbase_platform_driver_init+0x18/0x1000
[ midgard_kbase]
26 ...
```

问题分析：

无

解决方法：

[Android/gpu/mali/vdd\\_gpu\\_disable\\_always-on\\_cause\\_panic](#) 下, 使用 0001-TMP-arm64-rockchip\_defconfig-set-CONFIG\_MALI\_MIDGARD.patch 将 midgard device driver build-in 到 kernel 镜像中.

补丁说明：

0001-TMP-arm64-rockchip\_defconfig-set-CONFIG\_MALI\_MIDGARD.patch 在Patch 文件都在以下目录：

[Android/gpu/mali/vdd\\_gpu\\_disable\\_always-on\\_cause\\_panic](Android/gpu/mali/vdd_gpu_disable_always-on_cause_panic)

**相关Redmine:**

Defect #209883 : <https://redmine.rock-chips.com/issues/209883>

### 10.3.2. Mali400\_GPU-设备上没有外置RTC-kernel提示BUG,死机

现象说明：

RK3188(RKPx3) Android 5.1, 设备上没有外置RTC, 设备运行一段时间后会死机, 并有如下 log :

```
1 [192152.062540] kernel BUG at kernel/timer.c:853!
2 [192152.062665] Unable to handle kernel NULL pointer dereference at virtual
address 00000000
3 ...
4 [192152.967428] [<c05b422c>] (_bug+0x1c/0x28) from [<c05f673c>]
(add_timer+0x28/0x2c)
5 [192152.975164] [<c05f673c>] (add_timer+0x28/0x2c) from [<bf019df8>]
(mali_control_timer_callback*+0x60/0x70 [mali])
6 [192152.985412] [<bf019df8>] (mali_control_timer_callback+0x60/0x70 [mali])
from [<c05f6480>] (run_timer_softirq+0x12c/0x3c0)
7 [192152.996486] [<c05f6480>] (run_timer_softirq+0x12c/0x3c0) from
[<c05ef07c>] (_do_softirq+0xc8/0x25c)
8 [192153.005746] [<c05ef07c>] (_do_softirq+0xc8/0x25c) from [<c05ef6c4>]
(irq_exit+0x98/0xa0)
9 [192153.014049] [<c05ef6c4>] (irq_exit+0x98/0xa0) from [<c05ab050>]
(asn_do_IRQ+0x50/0xac)
10 [192153.022098] [<c05ab050>] (asn_do_IRQ+0x50/0xac) from [<c05b0ac8>]
(__irq_svc+0x48/0xe0)
```

"mali\_control\_timer\_callback" 是 Mali400 device driver 中的函数.

相关Redmine：

Defect #252934: <https://redmine.rock-chips.com/issues/252934>

详细文档及补丁文件：

在 mali.ko 中加入 workaround 的 patch, 可解决本异常.

相关 patch 和使用步骤, 详见如下文档 : [如何处理" Mali400\\_GPU-设备上没有外置RTC-kernel提示BUG,死机"](#)

# 11. GPU 性能相关

## 11.1. 性能不足问题

### 11.1.1. 性能问题-低温测试-RK3288(5.1)GPU低温测试性能差

**归属模块 :**

midgard device driver

**关键字 :**

低温下, GPU 性能变差

**现象说明 :**

RK3288 5.1 SDK 的某个历史版本, 当温度下降到 0 度以下, GPU 性能变得很差, 比如较大的 UI 动画场景(旋转等)的 FPS 变得很低.

且 kernel 输出如下指示“温度过高”的 log :

```
1 [ 6143.000761] w : [File] : drivers/gpu/arm/midgard/platform/rk/mali_kbase_dv  
fs.c; [Line] : 276; [Func] : mali_dvfs_event_proc(); overheated! temp:-1, but  
can't jump down anymore.
```

**问题分析 :**

无

**解决方法 :**

该异常是 midgard device drivers 中的RK内部提交的BUG导致 : 将零度以下的温度识别为温度过高, 将 clk\_gpu 限制在 最低频点.

为处理该异常, 可以将产品的源码工程升级到 SDK 到最新版本(git rebase 或者 merge), 或者 在 kernel/ 下使用 0001-MALI-midgard-RK-specific-code-use-type-int-for-tempe.patch 来修复.

**补丁说明 :**

0001-MALI-midgard-RK-specific-code-use-type-int-for-tempe.patch

在以下目录 :

[Android/gpu/performance\\_issues/low\\_temperature\\_low\\_performance\\_rk3288\\_5.1](Android/gpu/performance_issues/low_temperature_low_performance_rk3288_5.1)

**SDK commit:**

0001-MALI-midgard-RK-specific-code-use-type-int-for-tempe.patch 已经被集成到 kernel 3.10 的主分支 :

仓库 : ssh://10.10.10.29:29418/rk/kernel

分支 : develop-3.10

commit ; f37b404 MALI: midgard: RK specific code: use type 'int' for temperature data

**相关Redmine:**

Defect #210733 : <https://redmine.rock-chips.com/issues/210733>

## 11.1.2. 温控异常-RK3399(7.1)GPU温控行为异常

**归属模块：**

devfreq GPU

**关键字：**

GPU 温度异常高

**现象说明：**

RK3399 7.1 industry SDK, 更新到V2.4后, 老化测试CPU/GPU温度比更新前 ( V1.9 ) 高出十多度 , 最高温度达到100度.

**问题分析：**

由于缺少必要的提交, 统筹系统性能和功耗的 IPA 机构行为异常, 未能有效控制 GPU 的温度.

**解决方法：**

更新到 SDK 最新版本.

或者, 在 kernel下使用如下 patches :

1. 0001-MALI-midgard-Fix-power-coefficient-overflow.patch
2. 0001-thermal-devfreq\_cooling-Add-support-to-update-stats-.patch
3. 0001-arm64-dts-rockchip-modify-gpu\_power\_model-for-Midgar.patch

**补丁说明：**

上述 3 个 patch 在以下目录 :

[Android/gpu/performance issues/temperature control issues rk3399 7.1](https://github.com/rock-chips/rk3399-n-all/tree/develop-4.4-release/kernel/patches/0001-Android/gpu/performance%20issues/temperature%20control%20issues%20rk3399%207.1)

**SDK commit:**

上述 3 个 patch 都已经集成到 RK3399 7.1 industry SDK 中 :

```
1 | 仓库 : ssh://git@10.10.10.78:2222/rk3399-n-all/rk/kernel
2 | 分支 : develop-4.4-release
3 | commit id 依次是 :
4 |     b1dd3412e0b3cd46123dc794c0522f93b539e44b,
5 |     b69101370a99b18468b2528e816704a2ba6d3310,
6 |     79fe6a4cec24c3cf61c9f618c867adaa2d647ee2
```

对应的 Change-ID 依次是 :

```
1 | Ie0ceca32a33a5a8da56d3a32913e819a7e31e45b
2 | Ic98fabf46f693a60b0f07094c59e75e4d141e42c
3 | I1843b999a3b93fd5791e556db8733596c75ef8ac
```

**相关Redmine:**

Defect #222487 : <https://redmine.rock-chips.com/issues/222487>

### 11.1.3. 性能问题-代码BUG-RK3288(6.0)某提交导致GPU性能差

**归属模块：**

midgard device driver

**关键字：**

GPU 性能变差

**现象说明：**

RK3288 6.0, 内部 kernel 3.10 主分支上的某个状态, GPU 性能较差, 比如 "从 launcher 进入 app 列表的动画的 FPS 变低".

**问题分析：**

无

**解决方法：**

该异常由 RK 内部 kernel 3.10 主分支上的 commit "e062505 Mali: Midgard: RK: set clk\_gpu lowest and disable DVFS before powering off GPU" 引入. 该提交存在 BUG : 某些场景(比如 launcher 调出 app 列表)中, "将 clk\_gpu 设置为 lowest 的操作" 会被频繁地调用, 导致 GPU 实际渲染性能下降, 帧率下降.

上述提交是非必须的, 可以直接移除或 revert.

RK3288 6.0 对外 SDK 没有包含 引入该问题的 commit.

但是确实有客户 以 patch 的方式 收到过该 commit, 参见 <https://redmine.rockchip.com.cn/issues/218502>, #10 .

此时, 可以让客户 去除 上述 patch; 或者 加上修复本异常的 patch : 0001-Midgard-RK-set-clk\_gpu-lowest-highest-and-disable-en.patch

**补丁说明：**

0001-Midgard-RK-set-clk\_gpu-lowest-highest-and-disable-en.patch

在以下目录 :

[Android/gpu/performance\\_issues/gpu\\_low\\_performance\\_rk3288\\_6.0](Android/gpu/performance_issues/gpu_low_performance_rk3288_6.0)

**SDK commit:**

0001-Midgard-RK-set-clk\_gpu-lowest-highest-and-disable-en.patch 已经被合并到 内部 kernel 3.10 主分支 :

仓库 : ssh://10.10.10.29:29418/rk/kernel

分支 : develop-3.10

commit : 8109523 Midgard: RK: set clk\_gpu lowest/highest and disable/enable DVFS in right place

**相关Redmine:**

Defect #218502 : <https://redmine.rock-chips.com/issues/218502>

## 11.2. 性能优化

### 11.2.1. 性能优化-调整GPU变频行以提高部分场景性能

**归属模块：**

devfreq GPU

**关键字：**

GPU 性能不足, 变频策略

**现象说明：**

kernel\_4.4/4.19, 调整 Mali GPU 变频行为的性能/节能偏好；

某些场景(3D 游戏等)下, GPU 定最高频点, 性能满足要求, 但若 GPU 使用默认的 'simple\_onDemand' devfreq governor 时, 被测试部或者客户评估为性能不足, 或较之前的版本差。

**问题分析：**

**解决方法：**

调整 dts 中 gpu node 的 upthreshold 和 downdifferential 属性的值, 比如 "upthreshold = <40>; downdifferential = <10>;"

对这两个属性(参数)的说明如下：

在 governor\_simpleondemand 的具体实现中,

若当前的 utilization > (upthreshold / 100), clk\_gpu 将跳到最高频点,

这里的 utilization 是 "当前的 GPU 使用率", 取值从 0 到 1(100),

而 upthreshold 的有效取值范围是 (0, 100);

否则, next\_frequency = current\_frequency \* utilization \* 100 / ((upthreshold - downdifferential) / 2 ),

current\_frequency 是当前的频率,

next\_frequency 是之后将使用的频率,

downdifferential 必须是非负整数, 且必须小于 upthreshold, 若 downdifferential 增大, ((upthreshold - downdifferential) / 2 ) 将减小, next\_frequency 将变大.

**补丁说明：**

3399\_10.0 中, 处理本问题的补丁：

[Android/gpu/performance\\_issues/modify\\_gpu\\_dynamic\\_frequency\\_to\\_improve\\_performance](https://github.com/Android/platform_frameworks_base/blob/master/Android/gpu/performance_issues/modify_gpu_dynamic_frequency_to_improve_performance)

**0001-arm64-dts-rk3399-set-gpu-frequency-conversion-thresh.patch**

**SDK commit:**

上述针对 RK3399\_10.0 的 patch 已经提交并合并：

```
1 | 仓库 : ssh://10.10.10.29:29418/rk/kernel
2 | 分支 : develop-4.19
3 | commit id : ffd7b5d0816db9e3525e1868880e34a2be7d0ded
4 | change id : I492b973ef7401699dbeecde8df6d66650155f14
```

### 相关Redmine:

Defect #237656 : <https://redmine.rock-chips.com/issues/237656>

### 11.3. 性能跑分

BenchMark测试数据已上传至附件：

[Android/gpu/performance\\_issues/performance\\_benchmark](#)

## 12. GPU 功能支持

### 12.1. OpenCL支持

#### 12.1.1. OpenCL支持-可执行用例

**归属模块：**

GPU 平台适配

**关键字：**

Android OpenCL demo

**现象说明：**

客户要求提供Andorid 能运行OpenCL 的demo

**问题分析：**

[Android/gpu/feature\\_support/opencl\\_demo](#)

目录下为android 可以编译通过并且可以运行的demo。

**解决方法：**

无

**补丁说明：**

1. 默认是64bit。如果是32bit , 请把libGLES\_mali.so.32bit 替换libGLES\_mali.so。
2. 把android-opencl-demo-blur 目录拷贝到 frameworks/native/opengl/tests/
3. 编译
4. 在当前目录下生成 test\_cl
5. adb push test\_cl /system/bin
6. adb push kernel.cl /system/bin
7. adb push image3.BMP /system/bin
8. 在目标板子上的shell
  - 1) chmod 777 /system/bin/test\_cl
  - 2) test\_cl -i /system/bin/image3.BMP
  - 3) 在当前目录会生成 gpu\_blur.bmp
  - 4) 这个gpu\_blur.bmp图片与原图image3.BMP比较 , 有模糊效果。
  - 5) test\_cl -i /system/bin/image3.BMP -g 21 -s 1.8 // 指定参数 , 做得更模糊

**SDK commit:**

无

**相关Redmine:**

无

## 12.1.2. 应用异常-并发调用 OpenGL ES 和 OpenCL 时应用异常

**归属模块：**

midgard driver

**关键字：**

并发调用 OpenGL ES 和 OpenCL, 应用异常

**现象说明：**

RK3399 7.1, app 对 OpenGL ES 和 OpenCL 并发调用, 且 OpenCL 占用 GPU 资源较多, 出现 app 卡死或者 crash 在 libGLES\_mali.so 中.

**问题分析：**

无

**解决方法：**

将 libGLES\_mali.so 升级到 r18p0-01rel0\_x\_4@0\_for\_3399\_7.1 之后.

kernel 中的 midgard\_device\_driver(kernel/drivers/gpu/arm/midgard), 也要对应升级, 至少要包含 kernel 4.4 主分支中的 commit "5cf27d0 Mali: midgard: changes to enlarge BASE\_JD\_ATOM\_COUNT to 512, for defect 184210"

**补丁说明：**

无

**SDK commit:**

libGLES\_mali.so :

```
1 仓库 : ssh://10.10.10.29:29418/android/rk/platform/vendor/rockchip/common  
2 分支 : rk32/mid/7.0/develop  
3 commit : a51fcfd5 GPU: 3399: drm_mali_so_with_c1_and_vulkan_of_r18p0-  
01rel0_x_4@0_for_3399_7.1
```

midgard\_device\_driver :

```
1 仓库 : ssh://10.10.10.29:29418/rk/kernel  
2 分支 : develop-4.4  
3 commit : 5cf27d0 Mali: midgard: changes to enlarge BASE_JD_ATOM_COUNT to 512,  
for defect 184210
```

**相关Redmine:**

Defect #184210 : <https://redmine.rock-chips.com/issues/184210>

Defect #200085 : <https://redmine.rock-chips.com/issues/200085>

### 12.1.3. 兼容问题-不同版本 DDK 编译的 OpenCL binary kernel 不兼容

**归属模块：**

mali\_driver

**关键字：**

OpenCL 报错或 crash

**现象说明：**

客户在 RK3399 7.1 industry SDK 的较老版本上开发 OpenCL 应用, 该 SDK 中的 mali\_driver 的 DDK 版本是 r14.

且客户 的 OpenCL app 中使用了在 mali\_driver r14 上编译得到的 OpenCLbinary kernel.

客户将 SDK 升级到最新, 但是上述 app 出现 OpenCL API 报错 或者 crash的异常.

**问题分析：**

无

**解决方法：**

最新 SDK 中的 mali\_driver 的 DDK 版本是 r18, 运行时不兼容 在 r14 上编译得到的 OpenCL binary kernel, 导致上述异常.

为解决本异常, 必须要求客户在 最新 SDK 上(mali\_driver r18) 上重新编译生成 OpenCL app 需要的 OpenCL binary kernel.

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #228668 : <https://redmine.rock-chips.com/issues/228668>

Defect #225523 : <https://redmine.rock-chips.com/issues/225523>

## 12.2. Vulkan支持

### 12.2.1. Vulkan支持-Mali

**归属模块：**

Vulkan

**关键字：**

no vulkan, Vulkan 应用无法执行

**现象说明：**

RK3399 7.1 industry SDK 早期版本, 未支持 Vulkan , 因此使用 Vulkan API 的 app 无法正常运行；

比如 squeezencnn-release .apk, 安装后启动, 点击识别GPU, 提示 "no vulkan capable gpu".

**问题分析：**

无

**解决方法：**

该 SDK 编译得到的 固件中缺少 支持 Vulkan 需要的 符号链接和 feature 声明, 可以使用如下两个补丁修复.

在 device/rockchip/rk3399 下, 使用 0001-device.mk-Vulkan-copy-xml-files-for-Vulkan-features-.patch;

在 vendor/rockchip/common 下, 使用 0001-rk3399-GPU-Vulkan-add-symbols-links-for-Vulkan.patch.

**补丁说明：**

上述两个 patch 文件, 在当前 git 工程中的路径如下 :

[Android/gpu/feature\\_support/vulkan/vulkan\\_support](#)

0001-device.mk-Vulkan-copy-xml-files-for-Vulkan-features-.patch

0001-rk3399-GPU-Vulkan-add-symbols-links-for-Vulkan.patch

**SDK commit:**

上述两个 patch 也已经提交并合并到 SDK 相关主分支中 :

```
1 device/rockchip/rk3399/
2   仓库: ssh://10.10.10.29:29418/android/rk/device/rockchip/rk3399
3   分支: rk33/mid/7.0/develop
4   commit: 5ea4bc316d20e95985ba737ccd28845743340a06
5 vendor/rockchip/common
6   仓库: ssh://10.10.10.29:29418/android/rk/platform/vendor/rockchip/common
7   分支: rk32/mid/7.0/develop
8   commit: f998983f829b046fac69b085d96620ccd7dfda95
```

**相关Redmine:**

Defect #199659 : <https://redmine.rock-chips.com/issues/199659>

Defect #201919 : <https://redmine.rock-chips.com/issues/201919>

Defect #200869 : <https://redmine.rock-chips.com/issues/200869>

## 12.2.2. Vulkan-应用卡住-PVR DDK 1.8 Vulkan相关应用卡住的问题

**归属模块：**

PVR

**关键字：**

RK3368, Vulkan

**现象说明：**

Android 8.1 以上，出现Vulkan 相关应用运行卡住

**问题分析：**

通过“debuger -b”命令确认卡住进程的堆栈信息，最终指向vkCreateSwapchainKHR 函数。

**解决方法：**

1.确认客户PVR版本，通过以下命令确认版本：

```
1 | getprop | grep pvr
```

确认客户ddk版本是否为

```
1 | Android 8.1:  
2 | [sys.gpvr.version]: [UM N 5.33]  
3 | Android 9.0 && 10.0:  
4 | [sys.gpvr.version]: [UM P 5.45]
```

如果低于此版本，需要更新sdk/vendor/rockchip/common/gpu 目录

**补丁说明：**

无

**SDK commit:**

Andoird 9.0 && Android 10.0:

```
1 | 仓库: vendor/rockchip/common  
2 | 分支: rk/rk33/mid/9.0/develop  
3 | commit 715d4d2954f26a886e5b5ea51dd4d56577652f85  
4 | Author: Li Huang <putin.li@rock-chips.com>  
5 | Date:   wed Aug 7 11:03:47 2019 +0800  
6 |  
7 |     RK3368 GPU G6110_64 version: UM P 5.42  
8 |  
9 |     1.Solve vulkan stuck  
10 |    2.Fixup eglCreateSyncKHR fail  
11 |  
12 |    Change-Id: I654d13496b0bb464036a2ad62f14dd342c1d502b  
13 |    Signed-off-by: Li Huang <putin.li@rock-chips.com>
```

Android 8.1:

```
1 仓库: vendor/rockchip/common
2 分支: rk/rk33/mid/8.1/develop
3 commit 55593f709b2e259041fcb41250352cf9e6942dac
4 Author: Li Huang <putin.li@rock-chips.com>
5 Date:   Wed Mar 4 18:09:40 2020 +0800
6
7 RK3368 GPU G6110_64 version: UM N5.32
8
9 Solve vulkan stuck
10
11 Signed-off-by: Li Huang <putin.li@rock-chips.com>
12 Change-Id: 1e507cd6ae45fc2a34a938cead33ec008f5f2b154
```

#### 相关Redmine:

无

# 13. GPU问题

## 13.1. GPU 第三方应用相关问题

### 13.1.1. Antutu

#### 13.1.1.1. Antutu-Vulkan-安兔兔V8相关测试无法运行

**归属模块：**

GPU 平台适配

**关键字：**

安兔兔 antutu V8无法运行 , “兵马俑”无法运行

**现象说明：**

1. 安兔兔 V8 版本 , “海岸线”“兵马俑” 无法测试。
2. Terracotta - Vulkan 和 Coastline -Vulkan 无法测试。

**问题分析：**

**解决方法：**

1. 要求平台支持Vulkan。
2. 确认以下2个软链接有创建。

```
1 ln -sf /vendor/lib/egl/libGLES_mali.so /vendor/lib/hw/vulkan.rk3399.so
2 ln -sf /vendor/lib64/egl/libGLES_mali.so /vendor/lib64/hw/vulkan.rk3399.so
```

3. Andorid 7.1 平台 ls -l /system/etc/permissions | grep vulkan 确认有version 的xml 文件如 android.hardware.vulkan.version-1\_0\_3.xml。
4. 手动把frameworks/native/data/etc/android.hardware.vulkan.\* , 3个文件推到/system/etc/permissions , 验证安兔兔V8 vulkan 测试项。
5. 在第4点确认可行后 , 在工程目录device/rockchip/common打入如下补丁 Android\device\rockchip\rk3399\0001-device.mk-Vulkan-copy-xml-files-for-Vulkan-features-.patch , 重新编译烧写后 , 确认第3点的xml已经存在。
6. Android 8.0 及以上平台 , 请确认 /vendor/etc/permissions/ 目录。对应补丁目标路径修改为 /vendor/etc/permissions/ 。
7. 在GPU支持Vulkan的情况下 , 只有物理内存大于等于6GB或者实际剩余内存大于2GB的手机才能运行《兵马俑》场景 , 否则将自动跳过这一场景 , 最终总成绩中《兵马俑》场景子项目测试成绩为 0 , 其余测试项目均可正常运行

安兔兔官网说明<https://www.antutu.com/doc/118805.htm>

**补丁说明：**

无

**SDK commit:**

参考9.4 Vulkan 支持

**相关Redmine:**

Support #238001 : <https://redmine.rock-chips.com/issues/238001>

Defect #227145 : <https://redmine.rock-chips.com/issues/227145>

### 13.1.1.2. Antutu-安兔兔V7 GPU项0分

**归属模块：**

GPU 平台适配

**关键字：**

安兔兔antutu V7. GPU 项 0分

**现象说明：**

RK3126C 安兔兔V7.1.6 跑GPU 零分。

**问题分析：**

无

**解决方法：**

1. 要求GPU 支持 GLES3.0 。 Mali400 或450 系列均不支持GLES3.0。
2. 平台内存2G 以上。
3. Mali400 或 450 平台，请用antutu V6。

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #221354 : <https://redmine.rock-chips.com/issues/221354>

### 13.1.1.3. Antutu-PVR-安兔兔精炼厂测试无法运行

**归属模块：**

PVR

**关键字：**

RK3368 安兔兔，Antutu v7 精炼工厂，

**现象说明：**

部分客户在RK3368上运行Antutu v7 跑分测试，精炼厂项（Refinery）该项跑分为0分

**问题分析：**

无

**解决方法：**

PVR 缺少以下扩展，无法支持精炼厂该项测试。

GL\_EXT\_tessellation\_shader

GL\_OES\_tessellation\_shader

GL\_ANDROID\_extension\_pack\_es31a

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #228670 : <https://redmine.rock-chips.com/issues/228670>

## 13.2. 应用卡住-Mali4x0缺少重要提交导致应用无响应问题

**归属模块：**

Mali400/450

**关键字：**

Mali 400/450 GPU 报错, Mali: Executor GP: Job 19916 Timeout,

**现象说明：**

kernel 3.10, 使用 Mali400/450 GPU 的平台上, mali\_device\_driver 输出明确的硬件状态异常报错 log, 诸如：

```
1 | Mali: Executor GP: Job 396148 Timeout on Mali_GP
2 | [86304.547337] Mali: ERR: drivers/gpu/arm/mali400/mali/common/mali_gp.c
3 | [86304.547355] mali_gp_hard_reset() 140
4 | Mali GP: The hard reset loop didn't work, unable to recover
5 | [86304.547380]
6 | [86304.547503] Mali: ERR: drivers/gpu/arm/mali400/mali/common/mali_mmu.c
7 | [86304.547521] mali_mmu_raw_reset() 279
8 | Reset request failed, MMU status is 0xFFFFFFFF
9 | ...
```

在上述 log 前, 通常有 fence timeout 的 log:

设备出现 app 无响应等异常行为.

**问题分析：**

无

**解决方法：**

在 kernel/ 下使用如下补丁：

1. 0001-MALI-Utgard-RK-fix-some-potential-problems-about-pm\_.patch .

若 SoC 是 RK3229, 保险起见, 还可以在 kernel/ 下使用如下两个 patch :

2. 0001-WORKAROUND-MALI-utgard-rk322x-add-delay-after-enabli.patch

3. 0001-MALI-utgard-RK-not-to-operate-clk\_gpu-and-pd\_gpu-whe.patch

**补丁说明：**

上述 3 个 patch 位于如下位置：

[Android/gpu/issues/mali4x0\\_lack\\_commit\\_cause\\_apk\\_anr](https://github.com/Android/gpu/issues/mali4x0_lack_commit_cause_apk_anr)

**SDK commit:**

上述 3 个 patch 都已经合并到 3.10 kernel 主分支：

```
1 | 仓库 : ssh://10.10.10.29:29418/rk/kernel
2 | 分支 : develop-3.10
```

0001-MALI-Utgard-RK-fix-some-potential-problems-about-pm\_.patch 对应的 commit :

```
1 | 5767e2a MALI Utgard: RK: fix some potential problems about pm_runtime
```

0001-WORKAROUND-MALI-utgard-rk322x-add-delay-after-enabli.patch :

```
1 | 5aeb486 WORKAROUND: MALI: utgard: rk322x: add delay after enabling clk_gpu
```

0001-MALI-utgard-RK-not-to-operate-clk\_gpu-and-pd\_gpu-whe.patch :

```
1 | e9a8cbe MALI: utgard: RK: not to operate clk_gpu and pd_gpu when rebooting
```

### 相关Redmine:

Defect #221637 : <https://redmine.rock-chips.com/issues/221637>

Defect #219145 : <https://redmine.rock-chips.com/issues/219145>

Defect #212882 : <https://redmine.rock-chips.com/issues/212882>

Defect #200019 : <https://redmine.rock-chips.com/issues/200019>

Defect #193710 : <https://redmine.rock-chips.com/issues/193710>

Defect #189156 : <https://redmine.rock-chips.com/issues/189156>

### 13.3. 驱动报错-拷机出现"job status 0x00000004 (TERMINATED)"

**归属模块：**

Midgard/Bifrost device driver

**关键字：**

TERMINATED, GPU fault 0x04

**现象说明：**

RK3288, RK3399 和 RK3326 上, 长时间运行 3D 应用(比如 antutu 烤机), kernel log 中可能出现类似如下的 "TERMINATED" log :

```
1 | mali ff9a0000.gpu: error detected from slot 1, job status 0x00000004  
  | (TERMINATED)  
2 | mali ff9a0000.gpu: t6xx: GPU fault 0x04 from job slot 1
```

**问题分析：**

无

**解决方法：**

1. 若上述 log 是单独出现的, 那么它实际上并 "不是" 报错 , 按照 GPU 的 TRM, 该 log 对应 "The job was interrupted by a HARD\_STOP command."这个 HARD\_STOP command 是 driver 主动下发给 GPU 的, 并不是错误.
2. 若上述 log 之前有 "fence timeout" 和 "JS: Job Hard-Stopped (took more than 50 ticks at 100 ms/tick)" 的报错 log, 则需要图形组来特别调查和处理, 比如 redmine #236263.

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Support #236263 : <https://redmine.rock-chips.com/issues/236263>

## 13.4. PVR-DDK1.5升级到1.8版本（针对部分webview 场景）

**归属模块：**

PVR

**关键字：**

RK3368, webview

**现象说明：**

早期比如3368 Android 5.1 && 6.0 , 部分客户webView 根据 [Android/gpu/pvr/browser\\_faq](#) 目录的浏览器FAQ.pdf 文档中要求需要升级ddk 从1.5到1.8。

**问题分析：**

无

**解决方法：**

需首先确认客户pvr 版本是否为1.5

getprop | grep pvr

确认客户ddk版本是否为

[sys.gpvr.version]: [UM M 1.34]

请直接提供“补丁说明”标示位置的压缩包给客户，让其参照readme自行操作移植。

正常完成移植后，请重新确认ddk版本信息

[sys.gpvr.version]: [UM N 5.14]

注：小部分客户kernel 环境可能和rk目前发布的SDK中kernel的代码相差较大，移植补丁后无法正常开机，请让客户打包上传目前使用的kernel代码，并移交GPU组处理。

**补丁说明：**

压缩包位于如下目录：

[Android/gpu/pvr/browser\\_faq/patch](#)

DDK\_1.8\_on\_rk3368\_6.0\_v5.tar.gz

**SDK commit:**

无

**相关Redmine:**

Defect #194515 : <https://redmine.rock-chips.com/issues/194515>

## 13.5. PVR-渲染错误-应用界面出现非预期黑色区域

**归属模块：**

PVR

**关键字：**

RK3368, 黑色区域

**现象说明：**

Android 8.1 以上会出现该问题，部分场景alpha计算有问题，导致出现黑色区域。logcat 中有以下 log：

```
1 | alpha is NOT supported, to set it to 1
```

**问题分析：**

无

**解决方法：**

需首先确认客户PVR版本是否最新

```
1 | getprop | grep pvr
```

确认客户ddk版本是否为

Android 8.1:

```
1 | [sys.gpvr.version]: [UM N 5.33]
```

Android 9.0 && 10.0:

```
1 | [sys.gpvr.version]: [UM P 5.45]
```

如果低于此版本，需要更新sdk/vendor/rockchip/common/gpu 目录

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #235814 : <https://redmine.rock-chips.com/issues/235814>

# 14. Linux

## 14.1. 系统问题

### 14.1.1. 系统崩溃

#### 14.1.1.1. 系统崩溃-SDK错误使能MALI\_BIFROST\_FOR\_LINUX

归属模块：

midgard\_device\_driver for Linux

关键字：

GPU fault、RK3399 Linux

现象说明：

客户使用 RK3399 Linux SDK V2.07, X server 等用户空间进程卡住 或者 crash,

kernel 有类似如下的 GPU 报错：

```
1 [ 209.738782] mali ff9a0000.gpu: Unhandled Page fault in AS0 at VA 0x0000000
0E741950C0
2 [ 209.738782] Reason: Memory is not mapped on the GPU
3 [ 209.738782] raw fault status: 0x9003C1
4 [ 209.738782] decoded fault status: SLAVE FAULT
5 [ 209.738782] exception type 0xC1: TRANSLATION_FAULT
6 [ 209.738782] access type 0x3: WRITE
7 [ 209.738782] source id 0x90
8 [ 209.738782] pid: 605
9 [ 209.742760] mali ff9a0000.gpu: error detected from slot 1, job status 0x0
0000042 (JOB_READFAULT)
10 [ 209.746007] mali ff9a0000.gpu: error detected from slot 1, job status 0x0
0000042 (JOB_READFAULT)
11 [ 209.747066] mali ff9a0000.gpu: t6xx: GPU fault 0x42 from job slot 1
12 [ 209.747792] mali ff9a0000.gpu: t6xx: GPU fault 0x42 from job slot 1
13 [ 209.748436] mali ff9a0000.gpu: t6xx: GPU fault 0x42 from job slot 0
14 [ 209.749094] mali ff9a0000.gpu: t6xx: GPU fault 0x42 from job slot 0
```

问题分析：

无

解决方法：

将产品工程整体 rebase 到 SDK 的最新版本;

或者 在 kernel/ 中使用如下 patch :

1. 0001-3399-only-arm64-rockchip\_linux\_defconfig-enable-CONF.patch
2. 0002-Revert-MALI-rockchip-upgrade-midgard-DDK-to-r18p0-01.patch
3. 0003-Revert-arm64-dts-rockchip-modify-gpu\_power\_model-for.patch

补丁说明：

上述 3 个 patch 文件以下目录：

[Linux/issues/system\\_err/system\\_crash/sdk\\_ser\\_err\\_config\\_in\\_rk3399\\_linux](#)

**SDK commit:**

无

**相关Redmine:**

Defect #182849 : <https://redmine.rock-chips.com/issues/182849>

Defect #189044 : <https://redmine.rock-chips.com/issues/189044>

#### 14.1.1.2. 系统崩溃-客户错误使用MALI\_BIFROST\_FOR\_LINUX

**归属模块：**

midgard\_device\_driver for Linux

**关键字：**

应用闪退、 GPU fault、 RK3399 linux

**现象说明：**

调用 GLES 的 app 闪退, kernel "GPU fault" log :

```
1 mali ff9a0000.gpu: error detected from slot 1, job status 0x00000058 (DATA_IN  
VALID_FAULT)  
2 mali ff9a0000.gpu: t6xx: GPU fault 0x58 from job slot 1  
3 ...  
4 mali ff9a0000.gpu: t6xx: GPU fault 0x58 from job slot 0
```

**问题分析：**

RK3399 Linux 平台错误配置了RK3326的 kernel config , 导致问题

**解决方法：**

1. 要求客户将其使用 SDK 更新到最新.
2. 检查kernel config, 确保其中是 :

```
1 CONFIG_MALI_MIDGARD_FOR_LINUX=y  
2 CONFIG_MALI_MIDGARD=y
```

而不是 :

```
1 CONFIG_MALI_BIFROST_FOR_LINUX=y  
2 CONFIG_MALI_BIFROST=
```

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

Defect #235855 : <https://redmine.rock-chips.com/issues/235855>

Defect #202759 : <https://redmine.rock-chips.com/issues/202759>

## 14.2. 咨询问题

### 14.2.1. 咨询问题-GPU是否启用

归属模块：

Linux

关键字：

Mali

现象说明：

部分场景卡顿严重，查看gpu 负载：

```
1 | cat /sys/devices/platform/*.gpu/utilisation
```

多次敲入，均为0，无变化

问题分析：

无

解决方法：

1. 确认内核是否有mali 相关异常报错信息。如果有，请优先建议客户同步SDK 至最新（尤其是 kernel 以及external/libmali 目录）
2. 确认GPU 相关软链接建立是否正常，部分环境可能会被MESA 配置覆盖。手动建立软链接请参考：

GLES 及 EGL 需要建立以下软链接

```
1 | ln -s libmali.so libEGL.so
2 | ln -s libmali.so libGLESv1_CM.so
3 | ln -s libmali.so libGLESv2.so
4 | Gbm 相关:
5 | ln -s libmali.so libgbm.so
6 | wayland:
7 | ln -s libmali.so libwayland-egl.so
```

正常建立后，Glmark2 测试可以看到如下Mali 相关的信息

```
1 | GL_RENDERER: Mali-*
```

补丁说明：

无

SDK commit:

无

相关Redmine:

无

## 14.3. Benchmark相关

### 14.3.1. Benchmark跑分-Glmark2 跑分较低

**归属模块：**

Linux

**关键字：**

Mali , Glmark2 , 跑分

**现象说明：**

客户使用Glmark2 评估GPU 性能，反映得分较低。

**问题分析：**

无

**解决方法：**

跑分较低存在两种情况：

1、Mali 库未正常启用，使用软件方式渲染。这时候请优先确认mali so相关软链接是否正常建立。这种情况下跑分通常远低于正常分数。参考11.6.1 章节描述。

2、最终跑分接近屏幕刷新率，不超过60。通常使因为Glmark on screen 渲染受到屏幕刷新率及分辨率影响，准确测试评估请用off screen 方式测试。测试命令如下(--size 用于指定分辨率大小)：

```
glmark2-es2 --size 3840x2160 --off-screen
```

以RK3399 测试数据为例：

off screen : 1080p分数为290， 4k分数为92 ( 测试条件为ddr : 800M , GPU : 800M , CPU0 : 1.4G , CPU4 : 1.8G ) ,设备不同分数会存在差异

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

无

## 14.4. OpenCL

### 14.4.1. Linux-OpenCL-可执行用例

**归属模块：**

Linux

**关键字：**

Linux Opencl demo

**现象说明：**

客户要求提供Linux 能运行cl 的demo

**问题分析：**

无

**解决方法：**

[\Linux\linux-opencl-demo-blur](#) , 该目录下为Linux buildroot下可运行的demo。

**补丁说明：**

1. 把linux-opencl-demo-blur 目录拷贝到buildroot目录下，如 buildroot\output\build\linux-opencl-demo-blur。
2. 修改makefile gcc , 指定交叉编译工具 .CC=aarch64-linux-gcc 。如果要编译32bit , 指定 .CC=arm-linux-gcc
3. make
4. 在当前目录下生成 opengl-tutorial
5. adb push opengl-tutorial /usr/bin
6. adb push kernel.cl /usr/bin
7. adb push image3.BMP /usr/bin
8. 在目标板子上的shell
  - 1 ) chmod 777 /usr/bin/opengl-tutorial
  - 2 ) opengl-tutorial -i /usr/bin/image3.BMP
  - 3 ) 在当前目录会生成2个文件 , cpu\_blur.bmp 和 gpu\_blur.bmp
  - 4 ) 这2个图片与原图比较 , 就是做过模糊处理的。
  - 5 ) opengl-tutorial -i /usr/bin/image3.BMP -g 21 -s 1.8

// 指定参数 , 做得更模糊

**SDK commit:**

否

**相关Redmine:**

否

## 14.4.2. Linux-运行某些OpenCL应用-应用卡死

---

### 现象说明：

使用 Midgard DDK r14 的 rk3399 Linux SDK 上, 客户的某些 OpenCL 应用执行一段时间后卡住, API 没有显式报错, 也没有 crash.

### 相关Redmine:

Defect #189171: <https://redmine.rock-chips.com/issues/189171>

Defect #254482: <https://redmine.rock-chips.com/issues/254482>

### 详细文档及补丁文件：

将设备中的 mali\_driver 升级到 r18 可以解决本异常.

相关 patch 和使用步骤, 详见如下文档: [Linux/OpenCL/OpenCL app is stuck/](#)

## 14.5. 第三方应用问题

### 14.5.1. Chromium

#### 14.5.1.1. Chromium-GPU 报错导致Chromium-Wayland启动失败

##### 现象说明

启动Chromium 出现GPU报错，且无法正常启动。

相关log：

```
1 [573:573:0703/143353.020636:ERROR:g1_image_egl.cc(37)] Error creating EGLImage: EGL_BAD_PARAMETER
2 [573:573:0703/143353.020939:ERROR:gpu_memory_buffer_factory_native_pixmap.cc(43)] Failed to create GLImage 508x1260, BGRA_8888
3 [573:573:0703/143353.021267:ERROR:shared_image_backing_factory_g1_texture.cc(996)] Failed to create image.
4 [573:573:0703/143353.021486:ERROR:shared_image_factory.cc(422)] CreateSharedImage: could not create backing.
5 [573:573:0703/143353.026533:ERROR:shared_image_factory.cc(228)] DestroySharedImage: Could not find shared image mailbox
6 [573:573:0703/143353.047241:ERROR:buffer_manager.cc(488)] [.DisplayCompositor]GL ERROR :GL_INVALID_FRAMEBUFFER_OPERATION : glBufferData: <- error from previous GL command
```

##### 相关Redmine：

<https://redmine.rock-chips.com/issues/259237#change-2410623>

##### 详细文档及补丁文件：

将设备中的mali\_driver 更新到对应提交可以解决本异常

相关 patch 和使用步骤, 详见如下文档：

[Linux/gpu/application\\_issue/](#)

#### 14.5.1.2. Chromium-部分版本无法load gpu 库

**归属模块：**

mali

**关键字：**

Mali , EGL\_EXT\_image\_dma\_buf\_import , chrome

**现象说明：**

Chromium 73 以上版本无法正常load gpu库。

打开Chromium GUP log开关抓取完整log

运行Chromium 带上以下选项:

```
1 | --enable-gpu-service-logging --enable-gpu-service-tracing  
2 | --enable-logging --v=1
```

能看到无法load gpu 报错的log

**问题分析：**

无

**解决方法：**

请更新SDK的externl/libmali 目录，最新的so已经支持EGL\_EXT\_image\_dma\_buf\_import扩展。

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

无

## 14.6. Linux librga

### 14.6.1. Librga-部分YUV 格式存在uv 变反的问题

**归属模块：**

Linux librga

**关键字：**

颜色错误，uv颜色变反

**现象说明：**

RGA 部分yuv 格式拷贝出现uv通互换道的问题，问题图像在7yuv内点击swap uv选项后可以显示正常。通常是如下格式存在问题：yuv420\_p、yuv422\_p等

**问题分析：**

无

**解决方法：**

1. 直接替换最新的RGA device driver。将压缩包解压替换掉kernel/driver/rockchip/rga2 或者 kernel/driver/rockchip/rga 目录
2. 或者提供patch

**补丁说明：**

在git 工程如下目录可以找到：

[Linux/issues/librga/release\\_version](#)

kernel 3.10 && 4.4 兼容，rga2: **device\_driver\_rga2.tar.gz**

kernel 3.10 && 4.4 兼容，rga1: **device\_driver\_rga.tar.gz**

相关patch

[Linux/issues/librga/rga\\_output\\_yuv\\_format\\_uv\\_swap\patch](#)

0001-video-rockchip-rga2-Fixup-wrong-swap-uv-on-YCrCb\_420.patch

**SDK commit:**

Kernel 4.4: b791aba486285310aa9c2ced04df5ecd080a4548

**相关Redmine:**

无

## 14.6.2. Librga-X11上窗口拖动卡顿，存在残影

**归属模块：**

Linux librga

**关键字：**

Glamor , EXA-RGA , Xserver

**现象说明：**

提供切换Glamor 和 EXA-RGA的方法，用于验证解决一些场景的性能问题，比如窗口拖动卡顿、残影等。

**问题分析：**

无

**解决方法：**

1. 确认sdk版本为v2.2以上
2. 修改设备上该文件：

```
1 | /etc/X11/xorg.conf.d/20-modesetting.conf
```

使用EXA 配置：

```
1 | Option "AccelMethod" "exa"
```

使用Glamor 配置

```
1 | Option "AccelMethod" "glamor"
```

**补丁说明：**

无

**SDK commit:**

无

**相关Redmine:**

无

### 14.6.3. Librga-Weston 上使用RGA 渲染

**归属模块：**

Linux librga

**关键字：**

RGA , Pixmen , Weston

**现象说明：**

Weston上的性能优化，部分场景Weston可能会额外调用GPU 合成，导致性能不理想，这部分可以交给RGA 做加速。

**问题分析：**

需要注意Weston改成RGA，那么就不支持Wayland-EGL，EGL 相关可执行程序需要使用offscreen egl 渲染，或者rga合成。

Linux SDK需要更新到最新。

config确认开启BR2\_PACKAGE\_LINUX\_RGA，重编Pixman，之后配置PIXMAN\_USE\_RGA环境变量为1，并且Weston启动参数加入--use-pixman

比如：

```
1 # /etc/init.d/ss501auncher
2
3 start)
4
5 ...
6 export PIXMAN_USE_RGA=1
7 ...
8 weston --tty=2 -B=drm-backend.so --idle-time=0 --use-pixman&
```

**解决方法：**

无

**补丁说明：**

无

**SDK commit:**

Buildroot: 6873e04dd246c0b969c19bcc38549c3e012a4b20

**相关Redmine:**

无