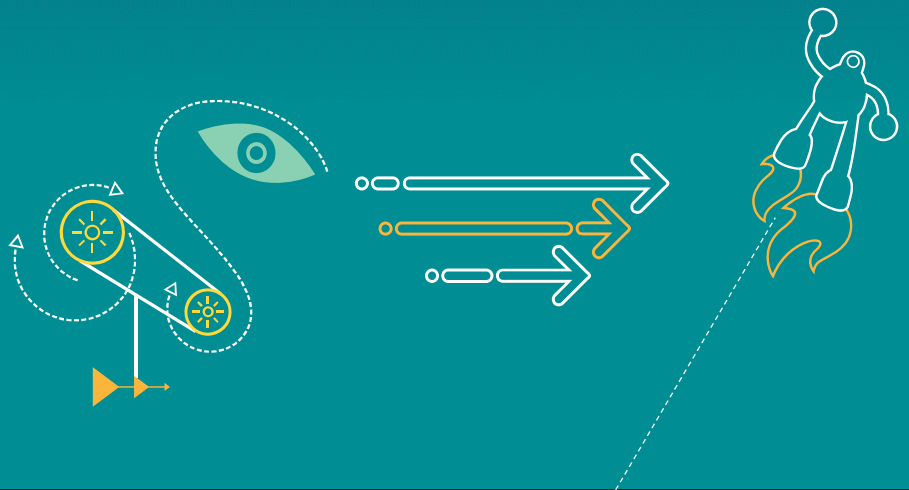
高通用户体验性能优化期刊

Qualcomm Technologies, Inc.



Confidential and Proprietary – Qualcomm Technologies, Inc.

Revision History

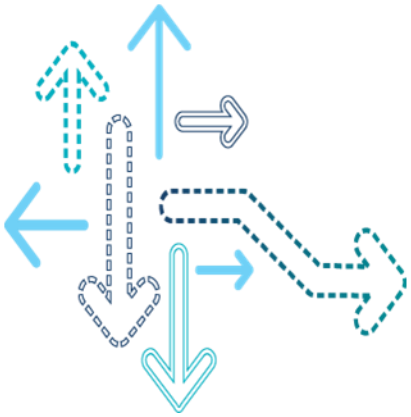
Revision	Date	Description
A	Apr. 2016	Initial release

Note: There is no Rev. I, O, Q, S, X, or Z per Mil. standards.

Contents

- **LCD蓝屏(under-run)**
- Adaptive LMK(ALMK)功能在8994/92平台使能

LCD蓝屏(under-run)问题分析



LCD蓝屏(under-run)问题分析

在开发过程中，遇见的LCD闪蓝屏问题，一般属于MDSS under-run。具体视觉效果差异较大，可以全屏闪，也可以局部闪动。有时难以确定复现路径，属于概率性问题。 其颜色可以通过panel dtsi文件(@file \kernel\arch\arm\boot\dts\qcom\dsi-panel-nt35590-720p-video.dtsi). <qcom,mdss-dsi-underflow-color>设置，以方便调试和最后阶段预防性的规避问题（可以设置成黑色）。

Under-run一般是因为display data flow 不能及时为MDSS 传送显示数据，MDSS HW自动填充数据以满足外部LCD panel的时序要求。

- 问题确认

有两种方法可以辨别under-run发生与否。

1.使用debugfs，如果under-run计数增加，那么就发生了under-run.

```
adb shell
 # cd d/mdp //d 是 debugfs 路径
# cat stat
 mdp:
intf2: play: 00010335    vsync: 00016858        underrun: 00000027
```

2. 在kernel文件中增加log信息，方便与特定frame信息一起打印

mdss_mdp_video_underrun_intr_done(void *arg)” (@ file \kernel\drivers\video\msm\mdss\mdss_mdp_intf_video.c).

更改 pr_debug 为 pr_err

```
- pr_debug("display underrun detected for ctl=%d count=%d\n", ctl->num, ctl->underrun_cnt);
+  pr_err("display underrun detected for ctl=%d count=%d\n", ctl->num, ctl->underrun_cnt);
```

LCD蓝屏(under-run)问题分析

- 问题debug

1.增加MDP的clock来确定是否与MDP clock计算错误有关

static void mdss_mdp_ctl_perf_update(struct mdss_mdp_ctl *ctl, int params_changed)” (@ file \kernel\drivers\video\msm\mdss\mdss_mdp_intf_ctl.c)

-- mdss_mdp_set_clk_rate(clk_rate);

++ mdss_mdp_set_clk_rate(320000000); // 320 MHZ 是 8916的最大值，每个芯片有所不同。

2. 增加DDR clock 和AXI bus clock的投票值

“int mdss_mdp_bus_scale_set_quota(u64 ab_quota, u64 ib_quota)” (@ file \kernel\drivers\video\msm\mdss\mdss_mdp_intf_ctl.c)

-- vect->ab = ab_quota;

-- vect->ib = ib_quota;

++ vect->ab = ab_quota *ab_fudge_factor;

++ vect->ib = ib_quota *ib_fudge_factor;

这些 factor可以是1.25, 1.5, 1.75，或者 中间数值。

以上两种方法可以初步定位问题，是否与MDSS clock以及系统带宽相关

LCD蓝屏(under-run)问题分析

3.增加VBP

如果VBP+VSYNC<6 ，可以考虑增加VBP，VBP的具体数值需要符合LCD panel的要求，否则LCD panel可能花屏或者黑屏。

4.添加实时logs，进行调试

https://www.codeaurora.org/cgit/quic/la//kernel/msm-3.10/commit/?id=50df44137b1ada21b4cb294fbb0c2fe07133d683

和入这个patch之后，抓取logs 一直到问题复现为止，然后提供logs给高通技术支持团队。

5.设置CPU performance mode

adb shell stop thermal-engine

adb shell "echo 1 > /sys/devices/system/cpu/cpuX/online" (cpuX : cpu0, cpu1 …)

adb shell "echo performance > /sys/devices/system/cpu/cpuX/cpufreq/scaling_governor" (cpuX: cpu0, cpu1, ...)

LCD蓝屏(under-run)问题分析

6.测试连接USB后，问题是否重现。

7. 参考solution 00028556 进行debug。

8.测试添加下面的patch后是否重现。

- msm_isp_update_IB_value_to_6400M.patch

```
--- a/drivers/media/platform/msm/camera_v2/isp/msm_isp_util.c
+++ b/drivers/media/platform/msm/camera_v2/isp/msm_isp_util.c
@@ -182,6 +182,9 @@ int msm_isp_update_bandwidth(enum msm_isp_hw_client client,
                                        isp_bandwidth_mgr.client_info[i].ib;
}
}
+    if (path->vectors[0].ib < 6400000000)
  +   {
+      path->vectors[0].ib = 6400000000;
+    }
msm_bus_scale_client_update_request(isp_bandwidth_mgr.bus_client,
isp_bandwidth_mgr.bus_vector_active_idx);

/* Insert into circular buffer */--
```

LCD蓝屏(under-run)问题分析

- msm: msm_bus: Mark certain rule transitions .patch

https://us.codeaurora.org/cgit/quic/la/kernel/msm-3.10/commit/?h=APSS.FSM.3.0&id=a4bae19890e5f02727b441e91fee49ebbaabe104

- L2PC delay patch.

  --- a/arch/arm/boot/dts/qcom/msm8916-pm.dtsi
  +++ b/arch/arm/boot/dts/qcom/msm8916-pm.dtsi
  @@ -149,7 +149,7 @@
  reg = <2>;
  label = "l2-pc";
  qcom,spm-l2-mode = "pc"; msm: msm_bus: Mark certain rule transitions .patch

  - qcom,latency-us = <11030>;
  + qcom,latency-us = <17500>;
  qcom,ss-power = <490>;
  qcom,energy-overhead = <972390>;
  qcom,time-overhead = <1580>;
  先测试蓝屏问题是否消失如果添加上面的三个patch。

LCD蓝屏(under-run)问题分析

如果问题依然存在则需要按情况测试下面的patch。

- Camera_PM_QOS_Request.patch //适用于camera preview ，recording 蓝屏问题。

pm_qos_request.patch

- Video_PM_QOS_Request.patch //使用于播放video 蓝屏问题。

VIDEO_disable_pm_qos_video_playback.diff

LCD蓝屏(under-run)问题分析

如果问题依然存在则需要准备机器做bus Profiling。

请参考下面的文档准备好机器邮寄给技术工程师。

80-NF341-1
80-NJ799-1

Adaptive LMK(ALMK)功能在8994/92平台使能

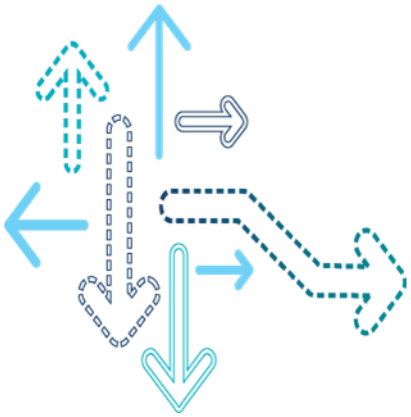
Adaptive LMK(ALMK)

ALMK特性可以改善低内存的场景：对于由于低内存导致的延迟和应用启动有改善。
在高通其他平台已经默认使能（8996/76/56/53/52/39/37）

Patch：

https://www.codeaurora.org/patches/quic/la/PATCH_146977_SBAforTaskidCaseid_20160407.tar.gz

如果对tuning ALMK参数有需求，请提Case高通支持



Questions?

https://support.cdmatech.com