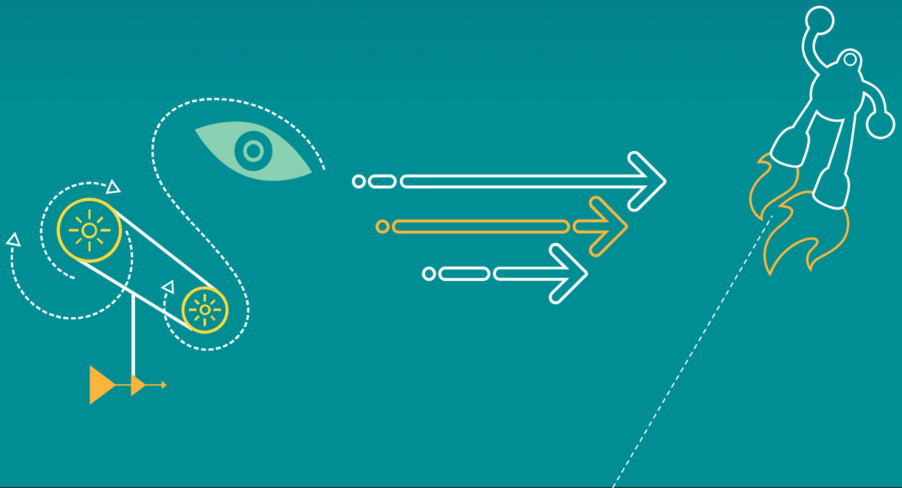
高通多媒体技术期刊　20151028

Qualcomm Technologies, Inc.



Confidential and Proprietary – Qualcomm Technologies, Inc.

Confidential and Proprietary – Qualcomm Technologies, Inc.

NO PUBLIC DISCLOSURE PERMITTED: Please report postings of this document on public servers or web sites to: DocCtrlAgent@qualcomm.com. **禁止公开：**如在公共服务器或网站上发现本文档，请报告至： DocCtrlAgent@qualcomm.com.

Restricted Distribution: Not to be distributed to anyone who is not an employee of either Qualcomm or its affiliated without the express approval of Qualcomm‘s Configuration Management. **限制分发：**未经高通配置管理部门的明示批准，不得发布给任何非高通或高通附属及关联公司员工的人。
Not to be used, copied, reproduced, or modified in whole or in part, nor its contents revealed in any manner to others without the express written permission of Qualcomm Technologies, Inc. 未经高通技术股份有限公司明示的书面允许，不得使用、复印、 复制、或修改全部或部分文档，不得以任何形式向他人透露其内容。
The user of this documentation acknowledges and agrees that any Chinese text and/or translation herein shall be for reference purposes only and that in the event of any conflict between the English text and/or version and the Chinese text and/or version, the English text and/or version shall be controlling. 本文档的用户知悉并同意中文文本和/或翻译仅供参考之目的，如英文 文本和/或版本和中文文本和/或版本之间存在冲突，以英文文本和/或版本为准。
This document contains confidential and proprietary information and must be shredded when discarded. 未经高通明示的书面允 许，不得使用、复印、复制全部或部分文档，不得以任何形式向他人透露其内容。本文档含有高通机密和专有信息，丢弃时必须粉碎销毁。
Qualcomm reserves the right to make changes to the product(s) or information contained herein without notice. No liability is assumed for any damages arising directly or indirectly by their use or application. The information provided in this document is provided on an “as is” basis. 高通保留未经通知即修改本文档中提及的产品或信息的权利。本公司对使用或应用本文档所产生的直接或间接损失概不负责。本文档中的信息为基于现状所提供，使用风险由用户自行承担。
Qualcomm is a trademark of QUALCOMM Incorporated, registered in the United States and other countries. All QUALCOMM Incorporated trademarks are used with permission. Other product and brand names may be trademarks or registered trademarks of their respective owners. Qualcomm是高通公司在美国及其它国家注册的商标。所有高通公司的商标皆获得使用许可。 其它产品和品牌名称可能为其各自所有者的商标或注册商标。
This technical data may be subject to U.S. and international export, re-export, or transfer (“export”) laws. Diversion contrary to U.S. and international law is strictly prohibited. 本文档及所含技术资料可能受美国和国际出口、再出口或转移出口法律的 限制。严禁违反或偏离美国和国际的相关法律。

Qualcomm Technologies, Inc. 5775 Morehouse Drive San Diego, CA 92121 U.S.A.
高通技术股份有限公司，美国加利福尼亚州圣地亚哥市莫豪斯路 **5775** 号，邮编 **92121**

Revision History

Revision	Date	Description
A	Oct 2015	Initial release

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

Contents

- Graphics
  - Google AOB bug: Youdao apk force close
  - Qualcomm issue: 8939 R3 UI corruption occurs randomly in the left-top corner of screen
  - Qualcomm issue: CTS test fail issue in cts.NativeMediaTest
  - Third party apk issue: display abnormal with wechat
- Camera
  - Camera PDAF Bring up

Graphics

Google AOB bug: Youdao apk force close

- Introduction：

When run Youdao dictionary APK, enter english words to search, APK reported error or force close.

From log we can see below crash backtrace:

```
F/libc (20364): Fatal signal 11 (SIGSEGV), code 1, fault addr 0x0 in tid 20364 (com.youdao.dict)
V/MemoryUtil( 3492): totalMemory MB = 0
I/DEBUG ( 450): *** *** *** *** *** *** *** *** *** *** *** *** *** *** *** ***
I/DEBUG ( 450): Build fingerprint: 'Letv/x1/x1:5.0.2/BEXCNOP5011907221D/37:userdebug/test-keys'
I/DEBUG ( 450): Revision: '0'
I/DEBUG ( 450): ABI: 'arm'
I/DEBUG ( 450): pid: 20364, tid: 20364, name: com.youdao.dict >>> com.youdao.dict <<<
I/DEBUG ( 450): signal 11 (SIGSEGV), code 1 (SEGV_MAPERR), fault addr 0x0
<SNIP>
I/DEBUG ( 450): backtrace:
I/DEBUG ( 450): #00 pc 00089ba4 /system/lib/libandroid_runtime.so (GraphicsJNI::getNativeCanvas(_JNIEnv*, _jobject*)+11)
I/DEBUG ( 450): #01 pc 000012a7 /system/lib/libwebviewchromium_plat_support.so
I/DEBUG ( 450): #02 pc 001ebc31 /system/lib/libwebviewchromium.so
I/DEBUG ( 450): #03 pc 001d0591 /system/lib/libwebviewchromium.so
I/DEBUG ( 450): #04 pc 001dd339 /system/lib/libwebviewchromium.so
I/DEBUG ( 450): #05 pc 00009c79 /data/dalvik-cache/arm/system@app@webview@webview.apk@classes.dex
```

对下面的函数分析并打印日志

```
SkCanvas* GraphicsJNI::getNativeCanvas(JNIEnv* env, jobject canvas) {
364 SkASSERT(env);
365 SkASSERT(canvas);
366 SkASSERT(env->IsInstanceOf(canvas, gCanvas_class));
367 jlong canvasHandle = env->GetLongField(canvas, gCanvas_nativeInstanceID);
368 SkCanvas* c = reinterpret_cast<android::Canvas*>(canvasHandle)->getSkCanvas();
369 SkASSERT(c);
370 return c;
371}
```

经日志分析于368行得到空指针。

This is known Google issue, google bug: 18261928, the formal fix is from Google.

Google AOB bug: Youdao apk force close

- Fix：
  This fix is the Google formal fix
  https://android.googlesource.com/platform/frameworks/base/+/de92f4c%5E!/#F0

```
commit de92f4c72be8537d2efff6024390b9ab706286e1
Author: Bo Liu <boliu@google.com>
Date: Mon Nov 24 10:53:52 2014 -0800

Fix crash in getNativeCanvas when canvasHandle is NULL

BUG: 18261928
Change-Id: I01a5af201fe829d5752433e1bb0db7edc01733d4

diff --git a/core/jni/android/graphics/Graphics.cpp b/core/jni/android/graphics/Graphics.cpp
index d7b75db..2eccfbd 100644
--- a/core/jni/android/graphics/Graphics.cpp
+++ b/core/jni/android/graphics/Graphics.cpp
@@ -365,6 +365,9 @@ SkCanvas* GraphicsJNI::getNativeCanvas(JNIEnv* env, jobject canvas) { SkASSERT(canvas); SkASSERT(env->IsInstanceOf(canvas, gCanvas_class)); jlong canvasHandle = env->GetLongField(canvas, gCanvas_nativeInstanceID);
+ if (!canvasHandle) {
+ return NULL;
+ }
SkCanvas* c = reinterpret_cast<android::Canvas*>(canvasHandle)->getSkCanvas();
SkASSERT(c);
return c;
```

Qualcomm issue: 8939 R3 UI corruption occurs randomly in the left-top corner of screen

- Introduction：

  8939 R3芯片的个别手机上面概率性出现左上角显示花屏现象, 经检查确认是与CR812357 有关，属于regression issue.

- Fix： This Qualcomm known issue, the related CR is CR885073.

  We requested to propagate these CR885073 to all the customer 8939 branch.

  If you meet the similar issue, you can apply below patch.

clk: qcom: 8936: Disable GMEM dynamic clock gating

https://www.codeaurora.org/cgit/quic/la//kernel/msm-3.10/commit/?id=5d766c7636a7dbeaf5e2bc729506b5666d4d400

Qualcomm issue: CTS test fail issue in cts.NativeMediaTest

- Introduction：
- CTS test failed on the 32 and 64 Bit test:

```
run cts -c android.mediastress.cts.NativeMediaTest -m test1080pPlay
run cts -c android.mediastress.cts.NativeMediaTest -m test480pPlay
run cts -c android.mediastress.cts.NativeMediaTest -m test720pPlay
run cts -c android.mediastress.cts.NativeMediaTest -m testDefaultPlay
```

- Fix：

This Qualcomm known issue, the related CR are CR800068 and CR863780.

We requested to propagate these CR800068, CR863780 to all the customer 8916/8939/8974/8909 branch.

If you meet the similar issue, you can ask for the GFX SBA lib on CR800068 or CR863780.

Third party apk issue: display abnormal with wechat

- Introduction：

  在微信朋友圈分享视频链接，播放时出现严重的画屏现象。

  这个问题只出现在基于Tencent QQ的webview的应用，如果改用系统的webview 就没有问题。

- Fix：

  这个是Tencent X5 webkit的问题, 已经跟Tencent沟通并确认此问题, Tencent会在后面的版本上面修正。

  Qualcomm provided one workaround change in framework HWUI code

```
void CanvasContext::trimMemory(RenderThread& thread, int level)
 if (level >= TRIM_MEMORY_COMPLETE) {
         Caches::getInstance().flush(Caches::kFlushMode_Full);
-        thread.eglManager().destroy();
+        const char* pro_name = get_process_name();
+        if(strcmp(pro_name, "com.tencent.mm:tools")) {
+            thread.eglManager().destroy();
+        }
   } else if (level >= TRIM_MEMORY_UI_HIDDEN) {
     Caches::getInstance().flush(Caches::kFlushMode_Moderate);
}
```

Camera PDAF Bring up

PDAF Type2/Type3 Module OTP

- PADF模组校准包含两部分 - PDAF Gain Map 和 PD Conversion Coefficient (K)
- PDAF Gain Map 分为1D Gain Map和2D Gain Map:
  - 1D Gain Map结构体定义:

```
typedef struct GainMap1DDataStructtag
{
    int VersionNum;
    unsigned short DSRatio;
    unsigned short ActualLength;
    unsigned short DSLength;
    unsigned short Left_GainMap[MAXLENGTH];
    unsigned short Right_GainMap[MAXLENGTH];
}  GainMap1DDataStruct;
```

PDAF Type2/Type3 Module OTP (2)

- 2D Gain Map结构体定义:

```
typedef struct GainMap2DDataStructtag
{
        int VersionNum;
        unsigned short OffsetX;
        unsigned short OffsetY;
        unsigned short RatioX;
        unsigned short RatioY;
        unsigned short MapWidth;
        unsigned short MapHeight;
        unsigned short Left_GainMap[MAXLENGTH2D];
        unsigned short Right_GainMap[MAXLENGTH2D];
}GainMap2DDataStruct;
```

- 1D和2D Gain Map结构体最开始都有VersionNum，虽然VersionNum是可选的，但是推荐写入这个字段，这样可以在OTP 驱动中区分1D还是2D Gain Map. 如果是1D, VersionNum 写入0x1；如果是2D，写入0x2
- 使用1D还是2D Gain Map，请在烧录OTP前与我们确认

PDAF Type2/Type3 Module OTP (3)

- PD Conversion Coefficient (K值) 校准后在OTP里写入一个2字节的校准值
- 所以最终的PDAF OTP内容如下

PDAF Gain Map 校准区	
VersionNumber (1D: 0x1, 2D: 0x2)	

1D:
unsigned short DSRatio;
unsigned short ActualLength;
unsigned short DSLength;
unsigned short Left_GainMap[PDGAIN_WITDH*PDGAIN_HEIGHT];
unsigned short Right_GainMap[PDGAIN_WITDH*PDGAIN_HEIGHT];
	2D:
	unsigned short OffsetX;
	unsigned short OffsetY;
	unsigned short RatioX;
	unsigned short RatioY;
	unsigned short MapWidth;
	unsigned short MapHeight;
	unsigned short Left_GainMap[PDGAIN_WITDH*PDGAIN_HEIGHT];
	unsigned short Right_GainMap[PDGAIN_WITDH*PDGAIN_HEIGHT];

PDAF PD Conversion Coefficient 校准区	
unsigned short PD_ConversionCoeff	

- PDAF校准方法和工具请参考文档 80-NV125-1 PDAF_Module_Calibration_Guide，请以最新H版本为准。

PDAF Type2/Type3 Module OTP (4)

- OTP驱动部分需要实现函数 xxx_eeprom_format_pdinfo(), xxx_eeprom_format_pdafgain()
  - xxx_eeprom_format_pdinfo(): PD 像素的坐标信息，一般相同的sensor，PD 像素的坐标是固定的。甚至同一个sensor vendor的不同sensor PD 像素的坐标都是一样的（比如S5K3M2 和S5K3L8）。该信息由sensor vendor提供
  - xxx_eeprom_format_pdafgain(): 从OTP buffer中读出Gain Map和PD Conversion Coefficient (K值) 校准值。
    - 1D参考dw9761b_eeprom.c
    - 2D参考imx258_gt24c16_eeprom.c

- xxx_eeprom_format_pdinfo()和xxx_eeprom_format_pdafgain 被xxx_eeprom_format_calibration_data()调用

- xxx_eeprom_get_calibration_items()中加入e_items->is_pdaf_sensor = TRUE;

PDAF Type2/Type3 Module OTP (5)

- do_af_calibration不使用默认的eeprom_autofocus_calibration(), 重载一个实现: xxx_autofocus_calibration(), 请参考dw9761b_autofocus_calibration()。留意在dw9761b_autofocus_calibration()对qvalue的使用。

- qvalue用于提高code_per_step计算结果的精度。 code_per_step是一个整型，但是AF OTP校准后计算得到的可能是小数。乘以qvalue可以提高精度。(在AF kernel驱动msm_actuator.c中对code的计算结果会除以qvalue还原真实的code)

PDAF Sensor 驱动部分修改

- 实现sensor_lib_ptr. sensorlib_pdaf_api中定义的callback函数

```
.sensorlib_pdaf_api =
 {
   .init = "xxx_init_pdaf",
   .deinit = "xxx_deinit_pdaf",
   .calcdefocus = "xxx_pdaf_calculate_defocus",
   .libname = “…",
   .pdaf_get_defocus_API = “…",
   .get_version_API = “…",
 },
```

- .libname, .pdaf_get_defocus_API和.get_version_API只用于PDAF Type-1中对外部PD算法库的引用。对于Type-2, Type-3只要实现.init, .deinit和calcdefocus.
- 请参考s5k3m2xm_lib.c 中的“s5k3m2xm_init_pdaf”，“s5k3m2xm_deinit_pdaf”和“s5k3m2xm_pdaf_calculate_defocus”。Type-2, Type-3这些函数是类似的.

PDAF Sensor 驱动部分修改 (2)

- 在xxx_init_pdaf函数中，注意对input.sensor_pattern和input.cali_version的赋值
  - s5k3m2xm_init_pdaf():
    - input.sensor_pattern = SENSOR_PATTERN0;
    - input.cali_version = CALIBRATION_VERSION_1D;
  - imx258_init_pdaf():
    - input.sensor_pattern = SENSOR_PATTERN1;
    - input.cali_version = CALIBRATION_VERSION_2D;
  - s5k3p3_init_pdaf():
    - input.sensor_pattern = SENSOR_PATTERN4;
    - input.cali_version = CALIBRATION_VERSION_2D;
- 设置sensor_out_info[0] .is_pdaf_supported = 1
  - 特别注意的是对于Type 2, Type3 PDAF，只有full size resolution支持PDAF所以只有sensor_out_info[0] .is_pdaf_supported = 1，其他resolution不支持，要设置sensor_out_info[n] .is_pdaf_supported = 0 (n != 0)。 这里设置错误会导致稳定性测试很多问题。
  - 注: Type-1 PDAF 其他resolution可能是支持PDAF的，比如imx230.

HAF tuning文件修改

- 对于msm8916/39使用老版本chromatix 头文件的平台，HAF tuning 文件在af_haf_tuning_data_camera.h，需要使能下面设置:
  .enable = 1,
  .algo_enable = {0,1,0,0},

- 对于msm8952/56这些使用0308 chromatix 头文件版本的平台, HAF tuning 已经是Chromatix头文件的一部分，但设置与以上一致

```
/***********************************************************************/
// Feature: Hybrid AF related tuning params.
//
// Variable name: enable.
// Enable flag for HAF.
// Default value: 0.
// Data range: 0 or 1.
// Constraints: None.
//
// Variable name: algo_enable[AF_HAF_ALGO_MAX].
// Array of enable flags for each haf algorithm.
// Default value: {0, 0, 0, 0}.
// Data range: 0 or 1.
// Constraints: None.
```

HAF tuning文件修改 (2)

- 0308 chromatix 头文件版本的平台, PDAF tuning的头文件由驱动res0的 a3_camera_chromatix 决定, 以S5K3M2为例：

```
s5k3m2xm_lib.h
.sensor_lib_chromatix =
   {
     /* RES 0 */
     {
…
.a3_camera_chromatix =
         SENSOR_LOAD_CHROMATIX(SENSOR_MODEL, zsl_preview_dw9761b),
     },
```

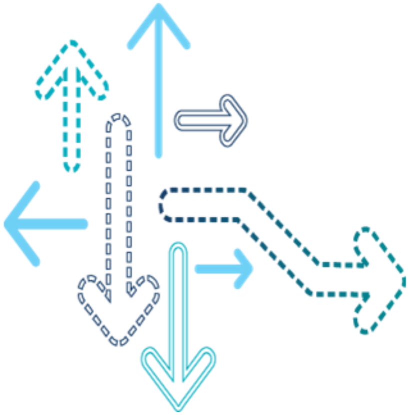
所以对应的PDAF tuning 头文件为 chromatix_s5k3m2xm_zsl_preview_dw9761b.h

注意事项

- 当前PDAF Type-2/3 Camera请不要有sensor mirror/flip。所以在项目前期模组设计时就要保证后面的sensor驱动不需做mirror/flip
- 有些EEPROM/OTP的datasheet没有描述清楚2字节数据是高位在前还是低位在前，在写OTP的驱动前一定要和模组厂确认清楚。PD gain map的数值就是2字节的，这里搞错PD就不能正常工作
- 打开OTP驱动的log，8952/56的新版本软件在打开camera的时候会发现xxx_eeprom_format_calibration_data()不会被调用。这里是我们对OTP和tuning 参数读取的优化，我们只在mm-qcamera-daemon进程启动的时候读取一次，然后会缓存起来。所以在camera 打开的时候读取缓存而不会再次调用OTP驱动去读取。可以把mm-qcamera-daemon进程杀掉让它自动重启服务，在adb log中就可以看到xxx_eeprom_format_calibration_data()的log了
- 在sensor驱动里面定义的xxx_init_pdaf() 中，请打印出PD lib的版本，有些新加入支持的sensor只有使用较新的PD lib才能支持。我们需要PD lib版本信息确认是否能正常支持当前sensor
- xxx_init_pdaf() 对cali_version, sensor_pattern的赋值不要搞错. 请参考pdaf_sensor_pattern_t定义里面的注释
- xxx_init_pdaf() 里设置的defocus_confidence_th不同sensor是不同的，有疑问请咨询我们的tuning工程师

说明

- 以上介绍是怎样使能Camera PDAF。至于PDAF 工作是否正常，需要做PDAF gating (Performance Checking)测试，我们会在后续的文档中介绍。



Questions?

https://support.cdmatech.com