

Rockchip DRM Direct Show 开发指南

文件标识: RK-YH-YF-454

发布版本: V1.0.0

日期: 2022-03-01

文件密级: 绝密 秘密 内部资料 公开

免责声明

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

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

商标声明

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

本文档可能提及的其他所有注册商标或商标, 由其各自拥有者所有。

版权所有 © 2022 瑞芯微电子股份有限公司

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

瑞芯微电子股份有限公司

Rockchip Electronics Co., Ltd.

地址: 福建省福州市铜盘路软件园A区18号

网址: www.rock-chips.com

客户服务电话: +86-4007-700-590

客户服务传真: +86-591-83951833

客户服务邮箱: fae@rock-chips.com

前言

文本主要介绍基于Rockchip DRM 显示驱动框架提供了内核态送显示的接口, 主要用于快速影像显示、kernel logo 显示以及一些自动测试等场景。

产品版本

芯片名称	内核版本
RK356X/RK3588	Linux kernel 5.10 及以上内核

读者对象

本文档 (本指南) 主要适用于以下工程师:

技术支持工程师

修订记录

版本号	作者	修改日期	修改说明
V1.0.0	黄家钊	2022-03-01	初始版本

目录

Rockchip DRM Direct Show 开发指南

Direct Show API 及使用说明

API 说明

获取 DRM 设备

申请内存

获取 crtc

获取 plane

提交显示

关闭图层

释放内存

使用流程

自动测试用例说明

Kernel Logo 显示说明

Direct Show API 及使用说明

API 说明

获取 DRM 设备

```
struct drm_device *rockchip_drm_get_dev(void);
```

用于获取 DRM 设备，后面的内存申请和释放，crtc 和 plane 的获取、提交显示等流程都需要用到 DRM 设备。

申请内存

```
int rockchip_drm_direct_show_alloc_buffer(struct drm_device *drm,  
                                          struct rockchip_drm_direct_show_buffer  
*buffer)
```

- **buffer 输入:**

变量	说明
width	buffer 的宽, 单位是 Pixel
height	buffer 的高, 单位是 Pixel
pixel_format	buffer 的格式, 具体定义可以参考: include/uapi/drm/drm_fourcc.h
flag	用于指定 buffer 的类型, 默认值0是离散非连续内存, ROCKCHIP_BO_CONTIG 是连续内存

- **buffer 输出:**

变量	说明
bpp	根据 buffer->pixel_format 获取的每个 Pixel bit 数量
pitch[3]	根据 buffer->width、buffer->pixel_format 以及一些对齐规则获取当前 buffer 的虚宽
vir_addr[3]	当前申请 buffer 内核态的虚拟地址
phy_addr[3]	当前申请 buffer 的物理地址
rk_gem_obj	rockchip drm gem 驱动用于管理当前 buffer 的抽象
fb	drm plane 显示中对当前buffer的抽象
dmabuf_fd	当前buffer export 出来的dmabuf_fd, 可能会被其他模块使用

对 buffer->pitch[3]、buffer->vir_addr[3]、buffer->phy_addr[3] 的说明:

以 buffer->pitch[3] 为例, 对于 RGB 格式默认使用 buffer->pitch[0], 如果是 NV12 格式有2个 plane 的, 使用 buffer->pitch[0] 和 buffer->pitch[1], 如果是有3个plane 的则使用: buffer->pitch[0]、buffer->pitch[1] 和 buffer->pitch[2]。

获取 crtc

```
struct drm_crtc *rockchip_drm_direct_show_get_crtc(struct drm_device *drm)
```

用于获取当前正在使用的 crtc。

获取 plane

```
struct drm_plane *rockchip_drm_direct_show_get_plane(struct drm_device *drm,
                                                    char *name)
```

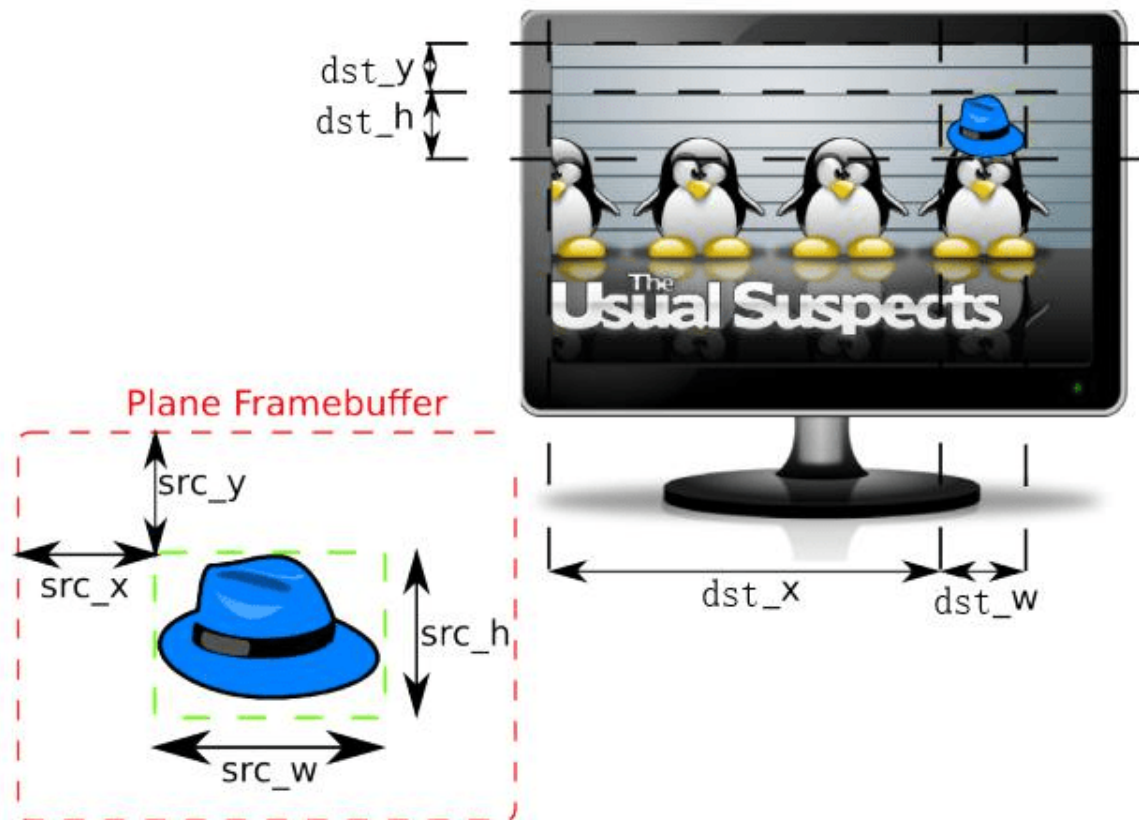
根据 name 信息获取对应的 plane, 获取的 plane 会被用于显示, 如获取 Esmart0-win0:

```
plane = rockchip_drm_direct_show_get_plane(drm_dev, "Esmart0-win0")
```

提交显示

```
int rockchip_drm_direct_show_commit(struct drm_device *drm,
                                   struct rockchip_drm_direct_show_commit_info
                                   *commit_info)
```

commit_info 变量	说明
crtc	指定提交的 crtc
plane	指定提交的 plane
buffer	提交用于显示的 buffer
top_zpos	false: 按默认配置, true: 设定最顶层
src_x/src_y/src_w/src_h	buffer 中用于显示部分的偏移和大小, 具体可以参考下图说明
dst_x/dst_y/dst_w/dst_h	当前图层显示在屏幕上的位置和大小, 具体可以参考下图说明



关闭图层

```
int rockchip_drm_direct_show_disable_plane(struct drm_device *drm,
                                           struct drm_plane *plane)
```

如不需要继续显示当前图层的信息可以通过该接口关闭图层显示。

释放内存

```
void rockchip_drm_direct_show_free_buffer(struct drm_device *drm,
                                          struct rockchip_drm_direct_show_buffer
                                          *buffer)
```

图层关闭后，如 buffer 无需继续使用，可以通过该接口释放和回收内存。

使用流程

Rockchip DRM Direct Show 默认是关闭的，如需要使用该功能，可以在 rockchip_defconfig 将其打开：

```
diff --git a/arch/arm64/configs/rockchip_defconfig
b/arch/arm64/configs/rockchip_defconfig
index b1c92060c6f9..2e3091f8631f 100644
--- a/arch/arm64/configs/rockchip_defconfig
+++ b/arch/arm64/configs/rockchip_defconfig
@@ -588,6 +588,8 @@ CONFIG_DRM_IGNORE_IOTCL_PERMIT=y
 CONFIG_DRM_DP_AUX_CHARDEV=y
 CONFIG_DRM_LOAD_EDID_FIRMWARE=y
 CONFIG_DRM_ROCKCHIP=y
+CONFIG_ROCKCHIP_DRM_DIRECT_SHOW=y
```

使用流程基本可以按以下7个步骤进行：

1. 获取 drm 设备；
2. 根据需要申请对应格式和大小的 buffer；
3. 向申请的 buffer 中绘制需要的图像；
4. 获取 crtc 和 plane；
5. 提交显示；
6. 如不需要继续显示，可以关闭图层；
7. 释放和回收内存；

自动测试用例说明

基于 Rockchip DRM Direct Show 实现了 Rockchip DRM 自动测试程序，可以用于开发者参考使用，也可以用于作为 Rockchip DRM 驱动的自动测试程序。该功能默认是关闭的，可以在 rockchip_defconfig 将其打开：

```
diff --git a/arch/arm64/configs/rockchip_defconfig
b/arch/arm64/configs/rockchip_defconfig
index b1c92060c6f9..2e3091f8631f 100644
--- a/arch/arm64/configs/rockchip_defconfig
+++ b/arch/arm64/configs/rockchip_defconfig
@@ -588,6 +588,8 @@ CONFIG_DRM_IGNORE_IOTCL_PERMIT=y
 CONFIG_DRM_DP_AUX_CHARDEV=y
 CONFIG_DRM_LOAD_EDID_FIRMWARE=y
 CONFIG_DRM_ROCKCHIP=y
 CONFIG_ROCKCHIP_DRM_DIRECT_SHOW=y
+CONFIG_ROCKCHIP_DRM_SELF_TEST=y
```

Kernel Logo 显示说明

部分产品没有使用 rockchip 官方的 uboot 或者因为某种原因无法打开 uboot logo 显示，但是又需要 kernel logo 显示的，可以通过修改 Direct Show 自动测试程序实现该功能：

```
diff --git a/drivers/gpu/drm/rockchip/rockchip_drm_self_test.c
b/drivers/gpu/drm/rockchip/rockchip_drm_self_test.c
index 74c395441e96..355632125914 100644
```

```

--- a/drivers/gpu/drm/rockchip/rockchip_drm_self_test.c
+++ b/drivers/gpu/drm/rockchip/rockchip_drm_self_test.c
@@ -20,7 +20,7 @@
#include "kernel_logo_img.h"

-#define USE_BUFFER_NUM 2
+#define USE_BUFFER_NUM 1
#define BUFFER_WIDTH 652
#define BUFFER_HEIGHT 268
#define BUFFER_FORMAT DRM_FORMAT_RGB565 /*
DRM_FORMAT_RGB565/DRM_FORMAT_XRGB8888/DRM_FORMAT_NV12 */
@@ -159,7 +159,7 @@ static void rockchip_drm_self_test_commit(struct work_struct
*work)
        goto free_buffer;
    }

-#if 1 /* for self test pattern */
+#if 0 /* for self test pattern */
    /* commit to display */
    do {
        u32 i = 0;

```

其中 kernel_logo_img.h 是通过使用bmp2hex等工具将 bmp 文件转成对应的数据保存的文件，并根据实际使用的图片修改：BUFFER_WIDTH、BUFFER_HEIGHT 以及BUFFER_FORMAT 的宏定义。

如果无法找到合适的 bmp2hex 工具可以从 rockchip redmine 以下链接搜索 bmp2hex 获取：<https://redmine.rock-chips.com/documents/97>。