

密级状态：绝密( )      秘密( )      内部( )      公开(√)

## RK32\_SDK, radical\_update 功能, 基本原理 和接口说明

(技术部, 图形系统组)

文件状态： [ ] 正在修改 [√] 正式发布	当前版本：	1.1
	作 者：	陈真
	完成日期：	2014/11/19
	审 核：	
	完成日期：	

福州瑞芯微电子有限公司

Fuzhou Rockchips Semiconductor Co., Ltd

(版本所有, 翻版必究)

## 版 本 历 史

版本号	作者	修改日期	修改说明	备注
1.0	陈真	2014/9/19	创建初始版本	
1.1	陈真	2014/11/19	添加对 parameter 文件的强调说明	



## 1 需求背景

目前 RK32\_SDK GPU 驱动程序尚未非常稳定, GPU 驱动版本发布 比 RK32\_SDK 整体版本发布要频繁.

为了终端用户可以尽快用上最新的 GPU 驱动, 必须提供一种机制, 使得最终的产品设备 可以从 RK 维护的服务器, 获取适合当前设备的 最新的 GPU 驱动.

因为这样的升级方式, 比一般的 ota 整体升级方更为 及时 或者 "激进", 故将该机制称为 radical\_update.

## 2 基本实现原理

### 2.1 涉及的功能实体

#### 2.1.1 ru\_pkg

radical\_update 机制使用的升级包. 记为 ru\_pkg. 目前也即 GPU 驱动的升级包.

Ru\_pkg 借用 ota 升级包的格式, 具体的安装也在 recovery 模式下中完成.

目前 ru\_pkg 中包含的 GPU 驱动二进制文件如下:

```
system/lib/modules/mali_kbase.ko
system/vendor/lib/egl/libGLES_mali.so
```

#### 2.1.2 ru\_server

RK 维护的网络 server, 主要用于对最终产品设备发布 ru\_pkg.

#### 2.1.3 ru\_client

运行在 最终产品设备上的应用程序. 用于向最终用户提供下载, 安装 ru\_pkg 的 UI.

在 SDK 中, 以 prebuilt\_apk 的形式提供, 具体路径:

```
device/rockchip/common/app/apk/RkExpe/RkExpe.apk
```

#### 2.1.4 ru\_partition

SDK 中设定的一个 flash 分区, 目前是 64M.

用于缓存 ru\_pkg 的内容, 以及备份被 ru\_pkg 替换的固件文件 (当前 ota 版本下的 GPU 驱动).

仅仅在 recovery 模式下挂载 正常模式下不可见.

参见 3.2.1.

## 2.1.5 recovery\_supporting\_ru\_pkg

支持安装 ru\_pkg 功能的 recovery 实现.

## 2.2 基本实现流程

1. RK 计划对最终产品设备发布新的 GPU 驱动; 将相关二进制文件和描述信息等, 打包到 ru\_pkg 中; 并部署到 ru\_server 上.
2. 某个产品设备上, 最终用户操作 ru\_client, 从 ru\_server 下载并安装该 ru\_pkg.
3. 设备将重启到 recovery 模式, recovery 将完成如下操作 :
  - a) 将 ru\_pkg 内容先安装到 ru\_partition 中,
  - b) 然后根据其中的配置信息, 先将 system\_partition 中 GPU 驱动程序的二进制固件文件备份到 ru\_partition 中. 这些固件文件的版本 标定在当前 ota 版本下, 所以将备份到 ru\_partition 中的这些固件文件, 记为 backup\_of\_fws\_in\_ota\_ver.
  - c) 将新的 GPU 驱动二进制文件部署到 system\_partition 中.
  - d) 设置 标识系统中已经 安装了某个 ru\_pkg 的标识 (flag\_of\_ru\_pkg\_applied).
  - e) 重启设备.
4. 重启之后, 系统将启用新版本的 GPU 驱动.

## 2.3 对 ota 差分升级机制的冲突

radical\_update 机制可以单独升级(修改) GPU 驱动的二进制文件, 假定之后设备要进行一个 ota 差分升级 (ota\_diff\_update), 且 ota 差分升级包 (ota\_diff\_pkg) 中也包含对 GPU 驱动的 patch, 若没有特别处理, 则 ota\_diff\_pkg 将失败. 因为 作为差分起点的文件的源状态已经被 radical\_update 修改, 这是维护产品设备 ota 版本的设备厂商无法知晓和无法控制的.

针对上述问题, recovery\_supporting\_ru\_pkg 在安装一般的 ota\_pkg 之前, 会先检查 flag\_of\_ru\_pkg\_applied, 若已经安装了 ru\_pkg, 则先从 ru\_partition 中将 backup\_of\_fws\_in\_ota\_ver 还原到 system\_partition 中, 然后再执行 ota\_update. 这样 ota\_update 就不会失败, 如同之前的 ru\_pkg 未被安装一样.

## 3 radical\_update 对 SDK 二次开发商的编译配置接口和要求

### 3.1 使能/禁用 radical\_update 的编辑配置

SDK 中 radical\_update 默认使能.

若要禁用该机制, 可以在 /work/cz/device/rockchip/rksdk/BoardConfig.mk 中 将 RECOVERY\_WITH\_RADICAL\_UPDATE 设置为 false. 这样编译出来的 recovery 将不支持

radical\_update, ru\_client (RkExpe.apk) 也不会被预置到 system\_partition 中。

## 3.2 使能 radical\_update 的注意事项

若二次开发商希望使用 radical\_update 机制, 还要注意如下事项 :

### 3.2.1 设备“必须”烧写 定义有 ru\_partition 的 parameter 文件

Radical\_update 机制依赖 ru\_partition, 所以系统的 flash 中“必须”定义有名称是 radical\_update 的分区。

一个定义有 ru\_partition 的 parameter 文件的内容如下, 在 CMDLINE 一行中, 定义 ru\_partition 的 起始地址和大小。

```
FIRMWARE_VER:4.4.2
MACHINE_MODEL:rk3288
MACHINE_ID:007
MANUFACTURER:RK30SDK
MAGIC:0x5041524B
ATAG:0x60000800
MACHINE:3066
CHECK_MASK:0x80
PWR_HLD:0.0.4.0.1
#KERNEL_IMG:0x62008000
#FDT_NAME:rk-kernel.dtb
#RECOVER_KEY:1.1.0.20.0
CMDLINE:console=tyFIQ0 androidboot.hardware=rk30board androidboot.console=tyFIQ0
board.ap.has_alsa=0 init=/init initrd=0x62000000,0x00800000
midparts=rk29xxnand:0x00002000@0x00002000(uboot),0x00002000@0x00004000(misc),0x00008000
@0x00006000(resource),0x00008000@0x0000e000(kernel),0x00010000@0x00016000(boot),0x0001000
0@0x00026000(recovery),0x0001u000@0x00036000(backup),0x00040000@0x00050000(cache),0x0000
2000@0x00090000(kpanic),0x00100000@0x00092000(system),0x00012000@0x00192000(metadata),0x
00200000@0x00194000(userdata),0x00020000@0x00394000(radical_update),-@0x003B4000(user)
```

### 3.2.2 产品设备中, gpu\_driver 依赖的模块的版本标识和实现, 必须尽量和 SDK 中一致

目前 ru\_pkg 承载的 gpu\_driver 对系统的如下模块 有版本依赖 :

kernel

libgui

gralloc

通过 radical\_update 机制发布的 gpu\_driver\_update, 基于 SDK 开发, 其特定版本依赖的 上述模块的版本, 也根据 SDK 来确定.

为了使得来自 SDK 的 gpu\_driver\_update 可以在产品设备上正常工作, **请保证具体产品中, 上述模块的版本标识和实现, 同 SDK 中尽量一致, "不要" 出现明显的背离和差异. 即, 将 SDK 中上述模块的更新引入产品时, 要完整而准确.** 否则 radical\_update 将无法正常工作, 甚至出现更新 gpu\_driver\_update 之后设备无法显示等问题.

### 3.2.3 请 "不要" 修改 SDK 中定义的如下系统 property :

ro.rk.soc

ro.build.version.release

上述 property 用来标识 产品设备 的软硬件平台信息, 请不要更改.