

密级状态：绝密() 秘密() 内部() 公开(☒)

Rockchip Android AB 系统使用说明文档

(技术部，第二系统产品部)

文件状态： [] 正在修改 [<input checked="" type="checkbox"/>] 正式发布	当前版本：	V1.4
	作 者：	纪大峤
	完成日期：	2019-04-17
	审 核：	
	完成日期：	

福州瑞芯微电子股份有限公司

Fuzhou Rockchip Electronics 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
V1.0	纪大峤	2018/8/1	初始版本	
V1.1	纪大峤	2018/8/2	增加 2.4 节烧写工具说明	
V1.2	纪大峤	2018/9/10	Android 9.0 补充说明	
V1.3	纪大峤	2019/3/29	文档重构	
V1.4	纪大峤	2019/4/17	更新“4.验证方法”	

目 录

1	概述.....	2
2	系统配置说明.....	2
2.1	ANDROID 系统配置	2
2.2	KERNEL DTS 配置	4
2.3	UBOOT 配置.....	5
2.4	烧写工具 ANDROIDTOOL 增加分区.....	5
3	AB 系统与 OTA 包编译.....	6
4	验证方法（客户端与服务器）	6
4.1	无缝升级验证.....	7
5	注意事项.....	9

1 概述

本文档描述了 AB 系统升级的使用说明，可以使用在 Rockchip Android 9.0 及以上 Android 软件平台上。

使用该升级方法，所有待升级的分区都有 a 和 b 两份，相比传统的 Recovery 升级方式会大幅增加存储空间需求。其优点是 AB 升级是无缝升级，可以在 Android 系统运行过程中，根据客户定制的客户端和服务端之间的交互策略与协议来完成升级。

2 系统配置说明

在 Rockchip Android 平台上，AB 系统功能默认关闭，要使用 AB 系统，需要从 Android 系统、U-BOOT 和 kernel dts 三个方面进行配置。以下对此进行详细说明。

2.1 Android 系统配置

Android 系统的配置包括：

1.在 device\rockchip\common\BoardConfig.mk 中打开 AB 升级配置项。

将 BOARD_USES_AB_IMAGE 配置设置为 true，默认为 false.

```
BOARD_USES_AB_IMAGE := false
```

2.在对应的 device\rockchip\rkxxx 目录下，确认是否已经有 parameter_ab.txt 分区表文件，如果有，则直接跳过这一步；如果没有 parameter_ab.txt 文件，则按如下步骤执行：

（1）新建 parameter_ab.txt 分区表文件

该文件从对应 parameter.txt 拷贝一份，然后在此基础上，进行修改，核心是针对每个要升级的分区划分两个分区 a 和 b，最后删除 recovery 分区并且将 boot 分区的大小改为 64MB。

一个参考 parameter_ab.txt 如下：

FIRMWARE_VER:9.0

MACHINE_MODEL:RK3326

MACHINE_ID:007

MANUFACTURER: RK3326

MAGIC: 0x5041524B

ATAG: 0x00200800

MACHINE: 3326

CHECK_MASK: 0x80

PWR_HLD: 0,0,A,0,1

TYPE: GPT

CMDLINE:mtdparts=rk29xxnand:0x00002000@0x00004000(uboot_a),0x00002000@0x00006000(uboot_b),0x00002000@0x00008000(trust_a),0x00002000@0x0000a000(trust_b),0x00002000@0x0000c000(misc),0x00008000@0x0000e000(resource),0x00010000@0x00016000(kernel),0x00002000@0x00026000(dtb),0x00002000@0x00028000(dtbo_a),0x00002000@0x0002a000(dtbo_b),0x00000800@0x0002c000(vbmeta_a),0x00000800@0x0002c800(vbmeta_b),0x00020000@0x0002d000(boot_a),0x00020000@0x0004d000(boot_b),0x00038000@0x0006d000(backup),0x00002000@0x000a5000(security),0x000c0000@0x000a7000(cache),0x00300000@0x00167000(system_a),0x00300000@0x00467000(system_b),0x00008000@0x00767000(metadata),0x000c0000@0x0076f000(vendor_a),0x000c0000@0x0082f000(vendor_b),0x00040000@0x008ef000(oem_a),0x00040000@0x0092f000(oem_b),0x00000400@0x0096f000(frp),-@0x0096f400(userdata:grow)

uuid:system=af01642c-9b84-11e8-9b2a-234eb5e198a0

(2) 新增针对 AB 的 fstab 文件 fstab.rk30board_AB

AB 分区增加 slotselect 挂载参数 (system 分区增加 slotselect 参数, 删除 oem 分区项), 同时将 frp, parameter, baseparameter, resource 的分区节点添加进去。

一个参考文件如下:

```
# Android fstab file.
#<src>
# The filesystem that contains the filesystem checker binary (typically /system) cannot
# specify MF_CHECK, and must come before any filesystems that do specify MF_CHECK
/dev/block/by-name/system / ext4 ro,barrier=1 wait,avb,slotselect
/dev/block/by-name/metadata /mnt/vendor/metadata ext4 noatime,nodiratime,nosuid,nodev,noauto_da_alloc,discard wait
/dev/block/by-name/misc /misc emmc defaults defaults
/dev/block/by-name/frp /frp emmc defaults defaults
/dev/block/by-name/parameter /parameter emmc defaults defaults
/dev/block/by-name/baseparameter /baseparameter emmc defaults defaults
/dev/block/by-name/resource /resource emmc defaults defaults
/devices/platform/usb* auto vfat defaults voldmanaged=usb:auto
/dev/block/zram0 none swap defaults zramsize=50%
# For sdmmc
/devices/platform/ff370000.dwmcc/mmc_host* auto auto defaults voldmanaged=sdcard1:auto,encryptable=userdata
# Full disk encryption has less effect on rk3326, so default to enable this.
/dev/block/by-name/userdata /data f2fs noatime,nodiratime,nosuid,nodev,discard,inline_xattr wait,check,notrim,encrypt
able=/mnt/vendor/metadata/key_file
/dev/block/by-name/cache /cache ext4 noatime,nodiratime,nosuid,nodev,noauto_da_alloc,discard wait,check
```

(3) device\rockchip\rkxxx 下的 BoardConfig 配置 TARGET_RECOVERY_FSTAB，使其指向刚刚创建的 fstab.rk30board_AB 文件。

+

+ifeq (\$(strip \$(BOARD_USES_AB_IMAGE)), true)

+TARGET_RECOVERY_FSTAB := device/rockchip/rk3326/rk3326_evb/fstab.rk30board_AB

+endif

如下截图：

```
diff --git a/rk3326_evb/BoardConfig.mk b/rk3326_evb/BoardConfig.mk
index c03eafe..7890e9a 100755
--- a/rk3326_evb/BoardConfig.mk
+++ b/rk3326_evb/BoardConfig.mk
@@ -14,3 +14,7 @@
# limitations under the License.
#
include device/rockchip/rk3326/BoardConfig.mk

ifeq ($(strip $(BOARD_USES_AB_IMAGE)), true)
TARGET_RECOVERY_FSTAB := device/rockchip/rk3326/rk3326_evb/fstab.rk30board_AB
endif
```

2.2 kernel dts 配置

包括添加 oem 项，同时将 slotselect 标记添加到 vendor 的 fsmgr_flags，参考配置如下：

```
--- a/arch/arm64/boot/dts/rockchip/rk3326-863-lp3-v10-avb.dts
+++ b/arch/arm64/boot/dts/rockchip/rk3326-863-lp3-v10-avb.dts
@@ -24,7 +24,14 @@
dev = "/dev/block/platform/ff390000.dwmcc/by-name/vendor";
type = "ext4";
mnt_flags = "ro,barrier=1,inode_readahead_blks=8";
fsmgr_flags = "wait,avb";
fsmgr_flags = "wait,avb,slotselect";

};
+
+ oem {
+ compatible = "android,oem";
+ dev = "/dev/block/platform/ff390000.dwmcc/by-name/oem";
+ type = "ext4";
+ mnt_flags = "ro,barrier=1,inode_readahead_blks=8";
+ fsmgr_flags = "wait,slotselect";
+ };
};
```

2.3 uboot 配置

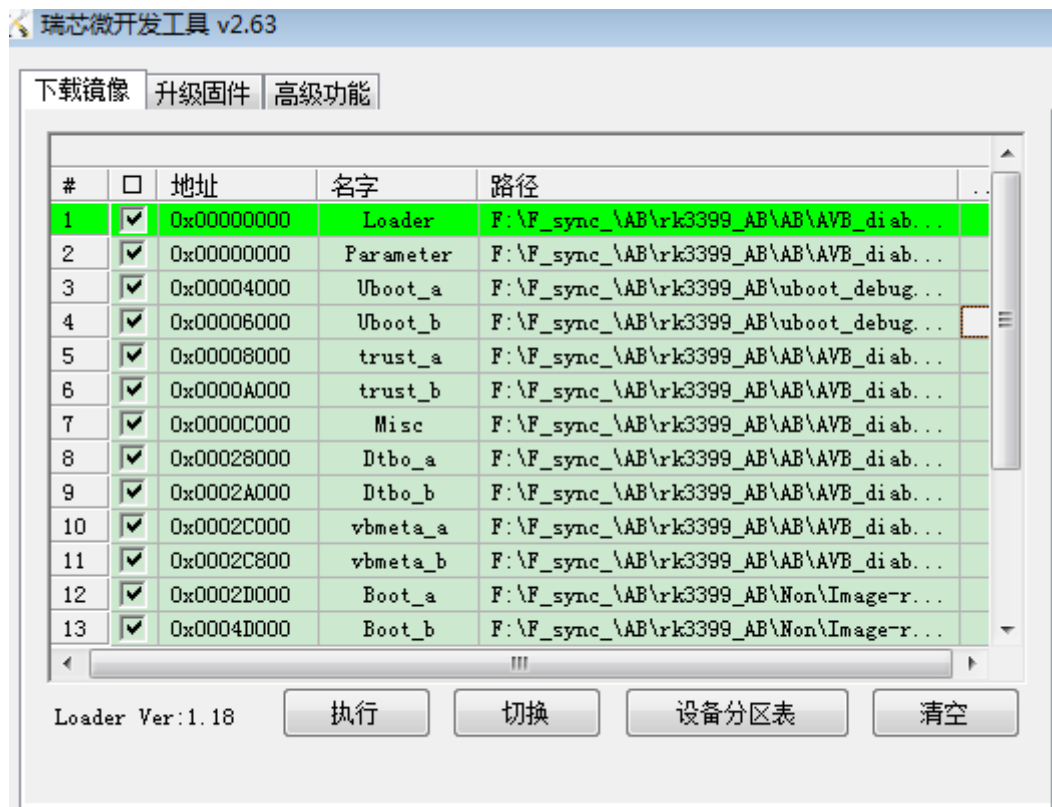
在 uboot 中，针对具体芯片的配置文件，添加 CONFIG_ANDROID_AB=y 配置项，参考配置如下截图所示：

```
diff --git a/configs/rk3326_defconfig b/configs/rk3326_defconfig
old mode 100644
new mode 100755
index 0465f23..d64648c
--- a/configs/rk3326_defconfig
+++ b/configs/rk3326_defconfig
@@ -114,3 +114,4 @@ CONFIG_OPTEE_CLIENT=y
 CONFIG_OPTEE_V2=y
 CONFIG_OPTEE_ALWAYS_USE_SECURITY_PARTITION=y
 CONFIG_TEST_ROCKCHIP=y
+CONFIG_ANDROID_AB=y
```

2.4 烧写工具 AndroidTool 增加分区

增加 B 分区的下载项，同时所有 AB 分区都烧写相同的固件。具体增加方法请参考《Rockchip Parameter File Format Ver1.3》

一个烧写工具截图如下：



3 AB 系统与 OTA 包编译

编译 AB 系统固件步骤（务必按如下步骤执行）：

```
lunch xxx
```

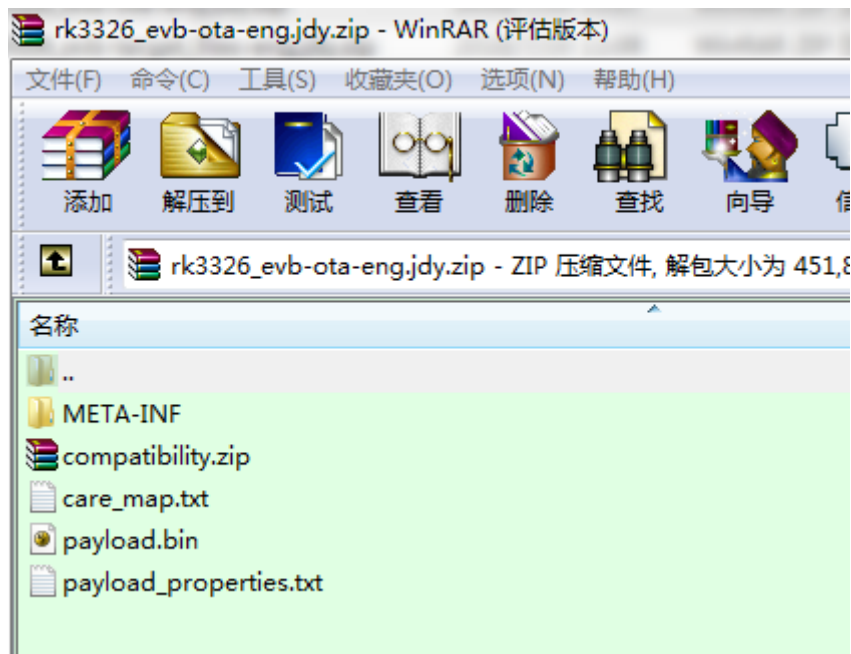
```
make -j32
```

```
make otapackage -j32
```

```
mkimage_ab.sh ota
```

注意：mkimage_ab.sh 文件从 device/rockchip/common 目录拷贝到 Android 根目录。

编译后的升级包组成参考如下：



4 验证方法（客户端与服务器）

Rockchip AB 系统支持正常系统下的无缝升级（一边下载升级包，一边升级）。

4.1 无缝升级验证

在无缝升级过程中，升级包可以一边下载，一边升级。这时候需要有一个 HTTP 服务器和一个升级客户端。

作为验证，有以下两种方法。至于产品化的升级客户端和升级服务器需要客户自行搭建，升级客户端可以参考 Android 默认提供的 update_engine_client。关于 update_engine_client 的使用方法请参考如下验证方法 update_device.py，该脚本就是通过 adb 最终调用 update_engine_client 来实现升级的。

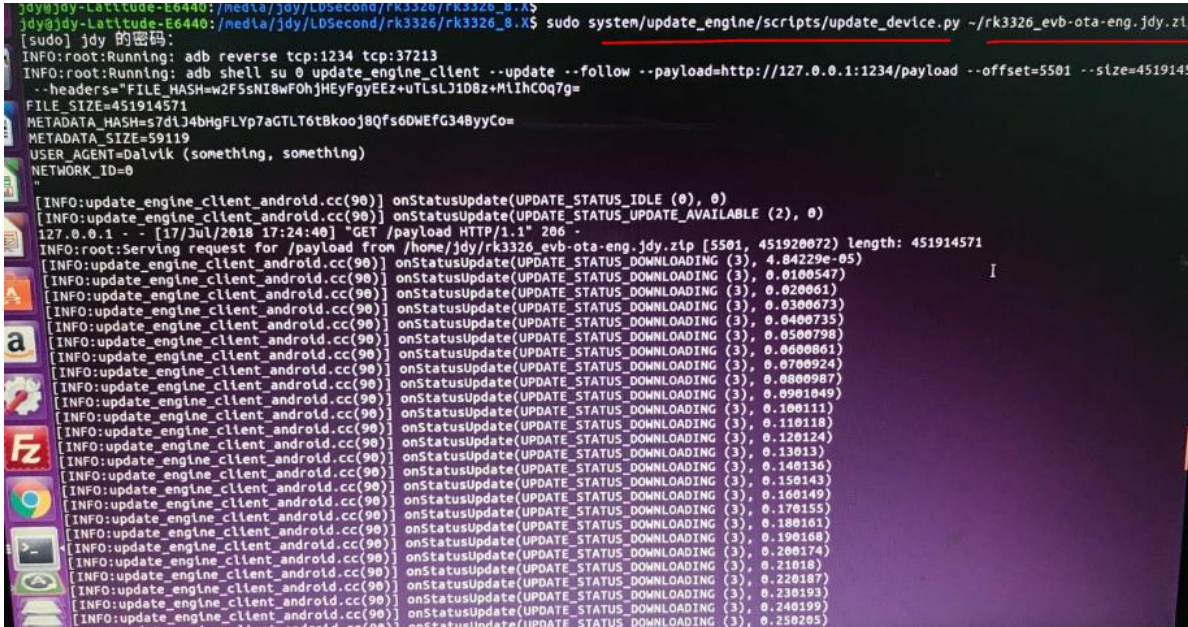
验证方法：update_device.py

update_device.py 脚本通过 adb 方式，将 Linux 主机变成 HTTP 服务器，然后调用 update_engine_client 来实现无缝升级。

使用方法如下：在 Linux 主机中执行如下命令（**要使用 Linux 主机,Windows 主机会有问题，同时确保该 Linux 主机 adb 功能正常**）：

system/update_engine/scripts/update_device.py {升级包名字}

示例如下：



```
jdy@jdy-Latitude-E6440: /media/jdy/LDSecond/rk3326/rk3326_0.X$ sudo system/update_engine/scripts/update_device.py ~/rk3326_evb-ota-eng.jdy.zip
[sudo] jdy 的密码:
INFO:root:Running: adb reverse tcp:1234 tcp:37213
INFO:root:Running: adb shell su 0 update_engine_client --update --follow --payload=http://127.0.0.1:1234/payload --offset=5501 --size=4519145
--headers="FILE_HASH=w2F5sNI8wFOhjHEYFgyEEZ+uTLsLj108z+MlIHCOq7g=
FILE_SIZE=451914571
METADATA_HASH=s7dLj4bHgFLYp7aGTLt6tBkooj8Qfs6DWEfG34ByyCo=
METADATA_SIZE=59119
USER_AGENT=Dalvik (something, something)
NETWORK_ID=0
"
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_IDLE (0), 0)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_UPDATE_AVAILABLE (2), 0)
127.0.0.1 - - [17/Jul/2018 17:24:40] "GET /payload HTTP/1.1" 206 -
INFO:root:Serving request for /payload from /home/jdy/rk3326_evb-ota-eng.jdy.zip [5501, 451920072] length: 451914571
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 4.84229e-05)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0100547)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.020061)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0300673)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0400735)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0500798)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0600861)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0700924)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0800987)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.0901049)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.100111)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.110118)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.120124)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.13013)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.140136)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.150143)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.160149)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.170155)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.180161)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.190168)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.200174)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.21018)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.220187)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.230193)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.240199)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_DOWNLOADING (3), 0.250205)
```


[illegible]


```
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.53)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.55)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.56)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.58)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.6)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.61)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.62)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.64)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.65)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.66)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.68)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.7)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.71)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.73)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.74)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.75)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.77)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.78)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.79)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.81)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.83)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.84)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.86)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.87)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.88)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.9)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.91)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.92)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.94)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.95)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.96)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.97)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 0.99)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 1)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 1)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_FINALIZING (5), 1)
[INFO:update_engine_client_android.cc(90)] onStatusUpdate(UPDATE_STATUS_UPDATED_NEED_REBOOT (6), 0)
[INFO:update_engine_client_android.cc(98)] onPayloadApplicationComplete(Error Code:rkSuccess (0))
INFO:root:Running: adb reverse --remove tcp:1234
INFO:root:Server Terminated
dy@jdy-Latitude-E6440:/media/jdy/LD5second/rk3326/rk3326_8.X$
```

上面的截图展示了完整的升级过程，升级成功后，会有 UPDATE_STATUS_UPDATED_NEED_REBOOT 地打印信息，如上截图所示。此时手动重启设备，就可以切换到新的升级后的系统。

5 注意事项

1.AB 需要 miniloader 支持，对于 rk3399 来说，miniloader 版本要求 V1.18 版本及以上，其他芯片对应的 miniloader 版本号，请向对应的项目接口人确认。