Rockchip

分片升级开发指南

发布版本:1.0.0

日期:2018.07

免责声明

本文档按“现状”提供，福州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 © 2018 福州瑞芯微电子股份有限公司

福州瑞芯微电子股份有限公司
Fuzhou 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： +86-591-83991906
客户服务传真： +86-591-83951833
客户服务邮箱： www.rock-chips.com

前言

概述

本文档主要介绍 Rockchip 处理器内针对小容量 flash 升级的一套升级方案，通过与手机 APP 配合通过 OTA 方式更新系统固件，以便升级系统。本文中详细介绍了该方案的开发过程以及注意事项。

产品版本

芯片名称	内核版本
RK3308	4.4

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
- 软件开发工程师

修订记录

日期	版本	作者	修改说明
2018.07.30	1.0.0	Chad.ma	初始版本

目录

插图目录

表格目录

1 分片升级

1.1 概述

分片升级方案是针对小容量存储设备提供一种安全、方便、快捷的系统升级方式。不同于普通 OTA 升级，需要借助其他存储介质，如 SD 卡、U 盘或自身的大容量存储空间，分片升级方案，只需要配合手机 APP，将手机与设备连接同一个网络，手机 APP 通过获取设备上系统固件各分区的版本信息，再通过网络与服务器上的升级文件版本描述文件做比较，将有更新的分区固件文件下载到手机存储空间上，然后再按分区将分区固件数据通过无线网络更新到设备的过程。

1.2 功能特点

- 小容量存储
- 单个或多个分区升级
- 手机 APP

1.3 升级流程

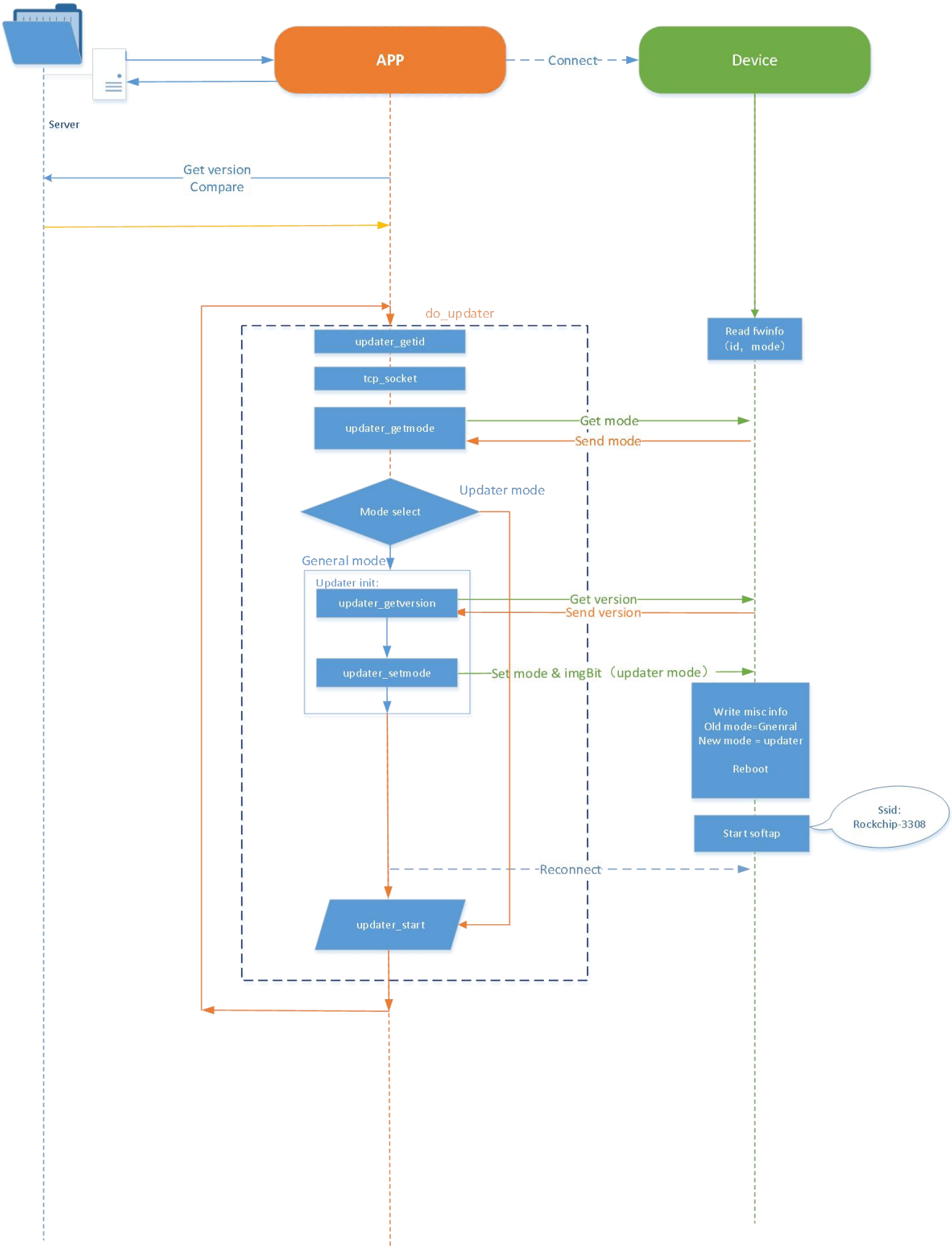

图 1 - 1 分片升级总体流程

图 1-1 中所示为分片升级方案中手机 APP 与设备及 APP 与服务器之间交互的一个简单流程说明。

承接上图，下图所示的是设备升级过程中 APP 与设备之间分片升级的具体流程：

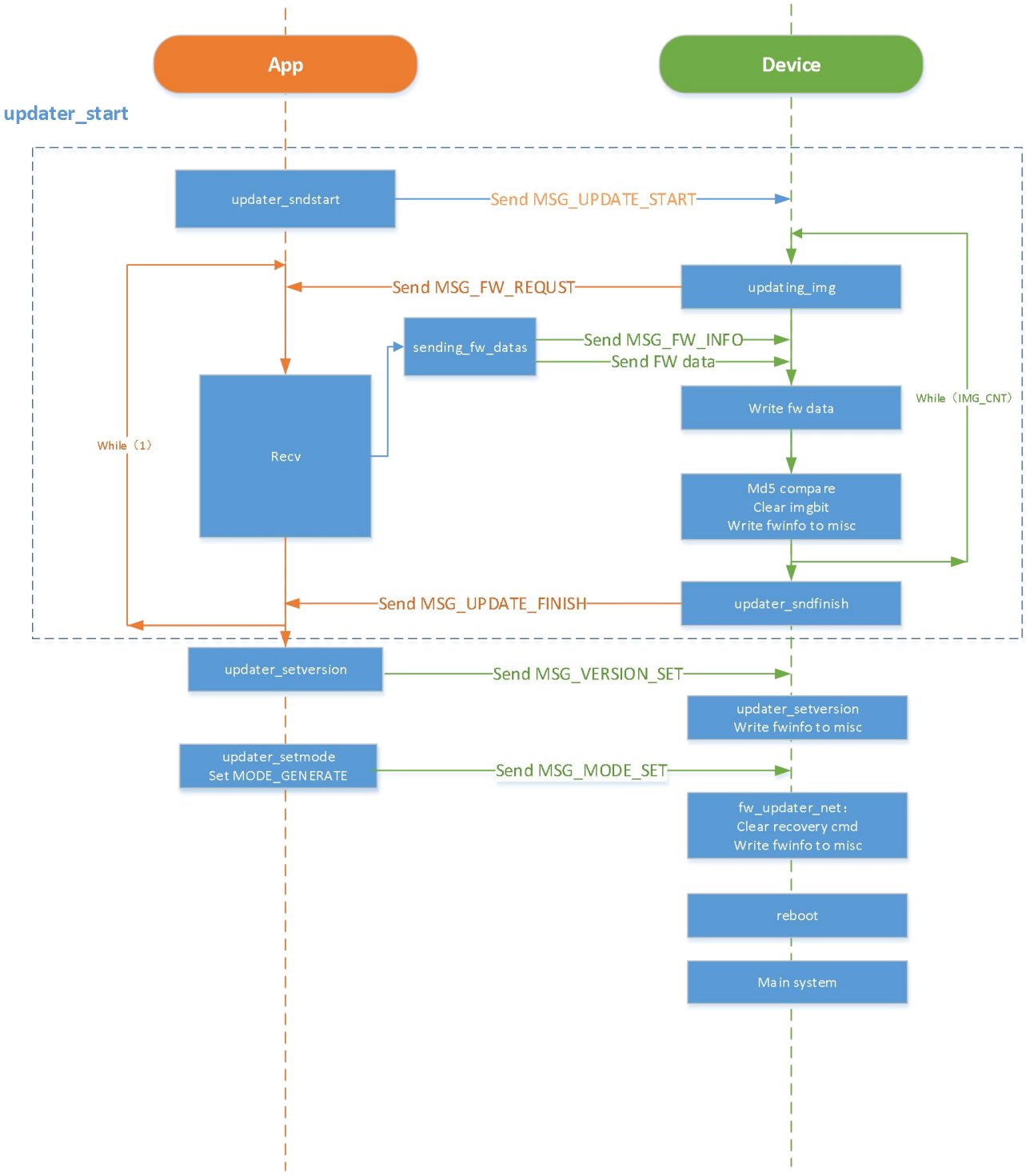

图 1 - 2 APP 与设备具体的分片升级流程

升级的详细流程，开发者可根据流程图，阅读升级方案的源码，这里不做进一步的展开说明。

注意事项

- 更新模式说明

  UPDATER_GNET：表示在正常系统下的普通模式。该模式下，手机 APP 与设备在同一局域网，

通过 socket 通讯，获取设备的分区固件版本信息，以及设置设备 misc 的一些信息，为在进入 recovery 模式后分片升级做准备。

UPDATER_UNET：表示在 recovery 下的分片升级模式。该模式下，设备启动 soft ap，APP 通过连接设备的 soft ap 后与设备通过 socket 通信，然后根据设备 misc 记录的固件升级信息，分片对设备需要升级的分区进行数据更新。

- 目前支持的分区包括："uboot","boot","rootfs","oem","trust","userdata"六个分区，顺序已经固定，这点可能不太灵活。
- 分片升级程序需要在正常系统与 recovery 系统下分别运行。
  正常系统下运行使用：frag_updater -g -d。
  Recovery 下运行使用：frag_updater -u -d。
  其中
  -g ： general mode
  -u ： update mode
  -d ： debug，open print log

1.4 编译方法

程序代码结构：external/frag-updater 目录下：

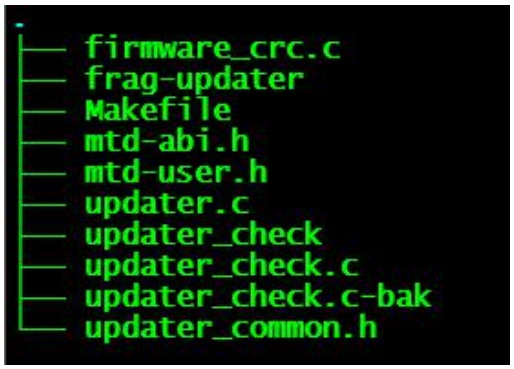

图 1 - 3 frag-update 代码结构

- 将 frag_updater 编译进正常系统
  确保 buildroot/package/rockchip/目录下存在 frag_updater 配置目录。

编译正常系统的 rootfs 时，选中该项配置。

```
make menuconfig
```

选中：

```
Target packages -->rockchip BSP packages -->Rockchip fragmetation updater for linux
```

保存，退出，

```
make savedefconfig
```

正常编译生成 rootfs。

- 将 frag_updater 编译进 recovery
  确保 buildroot/package/rockchip/目录下存在 frag_updater 配置目录。
  确保 buildroot/package/rockchip/目录下存在 recoverylaunch 配置目录。
  编译 recovery 时，选中该两项配置。

```
make menuconfig
```

选中：

```
Target packages -->rockchip BSP packages -->Rockchip fragmetation updater for linux
Target packages -->rockchip BSP packages --> Rockchip recovery launch for linux.
```

保存，退出

```
make savedefconfig
```

编译 recovery，生成 recovery 固件。

注意事项：

这里的 recoverylaunch 是根据 misc 分区中 bootloader cmd 的内容来启动执行哪一种升级方式，如果从 misc 分区读取到的 bootloader cmd 是"recovery\n--fragupdate"，则将启动分片升级的程序 frag_updater bin 程序，不然启动普通的 recovery bin 程序。

详见 external/recoverylaunch/recoverlaunch.c

图 1 - 4 recoverylaunch 代码结构

2 服务器搭建说明

2.1 搭建 Tomcat 服务器

关于如何搭建 Tomcat 服务器（也可以是 Apache 服务器，不做赘述）网上有比较多的教程，这里简单给出几个博客地址供参考。

1）Windows 环境下：
https://www.cnblogs.com/mfrank/p/7874177.html

2）Linux 环境下：
https://www.cnblogs.com/xdp-gacl/p/4097608.html

2.2 修改服务器端口

Tomcat 默认端口号是 8080，访问的时候需要加上端口号，不是特别方便，建议改成 80 后端口号可以省略。找到配置文件，将 8080 端口改成 80 端口，以 apache-tomcat-8.5.32 为例，配置文件在 apache-tomcat-8.5.32/conf/server.xml

```
<!-- A "Connector" represents an endpoint by which requests are received
     and responses are returned. Documentation at :
     Java HTTP Connector: /docs/config/http.html
     Java AJP  Connector: /docs/config/ajp.html
     APR (HTTP/AJP) Connector: /docs/apr.html
     Define a non-SSL/TLS HTTP/1.1 Connector on port 8080
-->
<Connector port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
<!-- A "Connector" using the shared thread pool-->
<!--
<Connector executor="tomcatThreadPool"
           port="8080" protocol="HTTP/1.1"
           connectionTimeout="20000"
           redirectPort="8443" />
-->
```

图 2 - 1 修改 Apache 配置文件中的端口

修改完后，重启 Tomcat 服务器生效。

2.3 相关文件上传

Tomcat 默认路径在 webapps/ROOT/目录下面，不需要作修改。只需要增加下图文件即可。
fw_version：Server 版本信息
firmware：固件文件，包含 3 个测试固件。

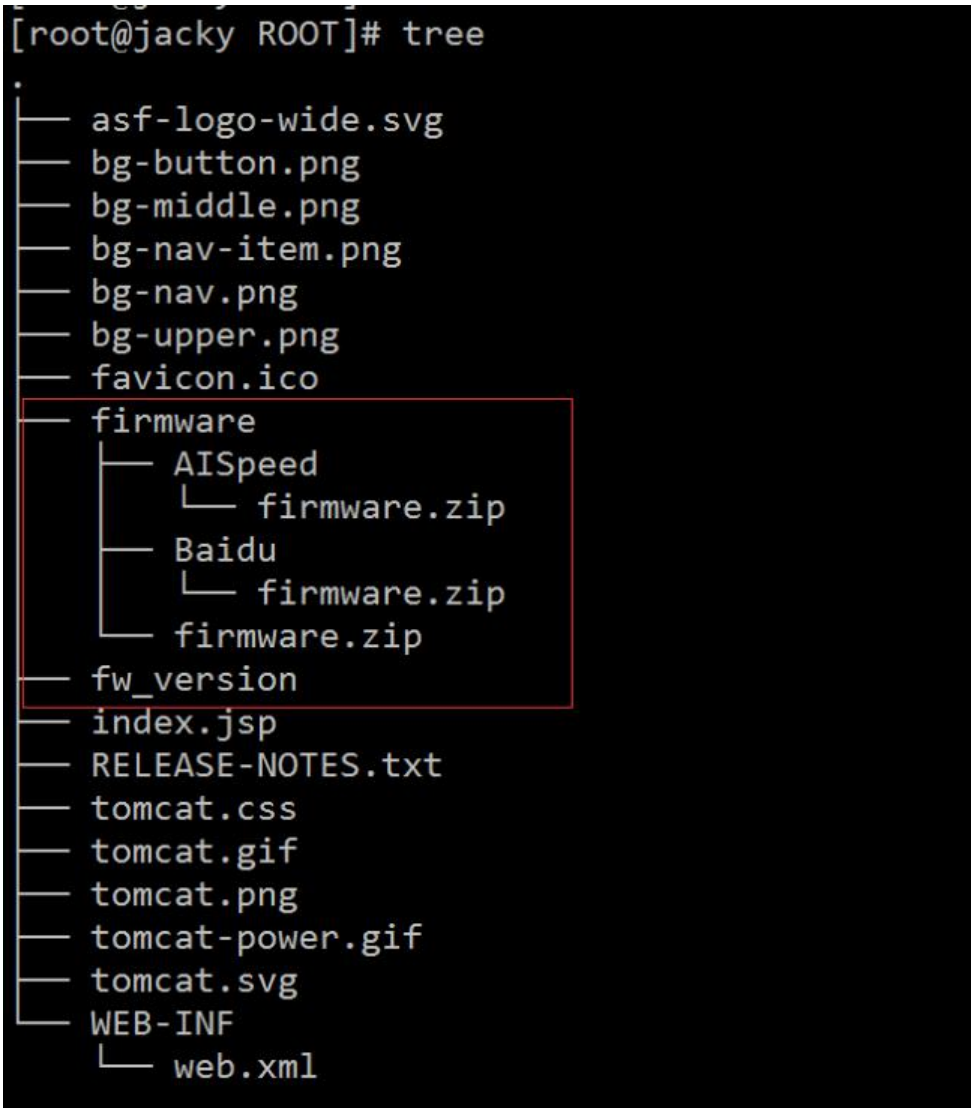

图 2 - 2 服务器固件文件结构