

# 君正® 君正 OTA 升级用户指南

Date: 2015. 8.17

Jim Gao [jian.gao@ingenic.com](mailto:jian.gao@ingenic.com)

---



北京君正集成电路股份有限公司  
Ingenic Semiconductor Co., Ltd.

君正 ®

## 君正 OTA 升级用户指南

Copyright © Ingenic Semiconductor Co. Ltd 2013. All rights reserved.

### Release history

Date	Revision	Change
August.17	1.0	First release
August .21	2.0	Add ota user interface

### Disclaimer

This documentation is provided for use with Ingenic products. No license to Ingenic property rights is granted. Ingenic assumes no liability, provides no warranty either expressed or implied relating to the usage, or intellectual property right infringement except as provided for by Ingenic Terms and Conditions of Sale.

Ingenic products are not designed for and should not be used in any medical or life sustaining or supporting equipment.

All information in this document should be treated as preliminary. Ingenic may make changes to this document without notice. Anyone relying on this documentation should contact Ingenic for the current documentation and errata.

北京君正集成电路股份有限公司

地址 : 北京市海淀区西北旺东路 10 号院东区 14 号楼 君正大厦

电话 : (86-10)56345000

传真 : (86-10)56345001

Http: //www.ingenic.cn

# OTA升级用户指南

## 1 硬件环境

- SPI-nor 16M; mem 32M
- 产品板可能不带 SD 卡
- 支持 WIFI 或者有线网络

## 2 基于无 SD OTA升级分区介绍

16M nor 基本使用，分为六个分区：

u-boot-with-spl.bin	256K
nv.img	128K
usrdata.jffs2	640K
zImage	3M
updater.cramfs	4.5M
appfs.cramfs	7.5M

- u-boot-with-spl.bin 是 uboot 引导程序。
- nv.img是升级配置信息，存放升级网址，版本信息等重要信息，不要轻易修改烧录位置。
- usrdata.jffs2 是 mozart 编译生成的文件，是音响的配置信息，挂载/usr/data，恢复出厂设置是会擦除这个分区。
- zImage 是内核 make zImage 生成的 image。
- updater.cramfs 是 mozart 编译生成的文件，是负责升级的部分。
- appfs.cramfs 是 mozart 编译生成的文件，是用户 app 的部分。

注意：

a. 不要自己随意调整分区的位置，特别是前三个分区的位置和大小，会导致 OTA 不能使用。后三个分区可以做简单调整，但是调整的原则是 zImage 分区和 updater.cramfs 分区的大小的和不能大于 appfs.cramfs 分区的大小。因为 ota 升级 zimage 和

updater.cramfs 时会暂时存在 appfs.cramfs 的位置，如果不按这个要求去做，OTA 会有越界。

b. 调整分区位置需要修改 update.c 和 mkzip.sh 的源码，里面有对分区的具体操作。

### 3 OTA 基本流程

- 目前不支持 spl 和 uboot 升级
- Spl 直接加载 kernel，uboot 只做升级 zImage 和 updater.cramfs 使用
- zImage 与 updater.cramfs 同时下载升级
- 如果有 zImage 与 updater.cramfs 升级，必须有 appfs.cramfs 升级
- 可以只有 appfs.cramfs 升级
- 下载以 1M block 为单元下载，支持断点续传
- **升级过程中支持掉电和拔网线等异常操作**，升级程序在 NV 分区设置了标志位，掉电或其他操作，只要标志位还在，会继续升级，直到升级成功，才会把标志位清除。前提是用户要保证升级 url 的正确性和升级软件版本的正确性，升级错误的软件包会导致升级失败或音响启动失败。如果误升级了错误的升级包，需要重新返回原厂烧录系统。

### 4 服务器端升级包规定

目录结构为：

- ../download/info
- ../download/20150811
- ../download/20150812
- Info 存放的为我当前目录的内容，例如：20150811

### 5 编译 OTA 文件系统：

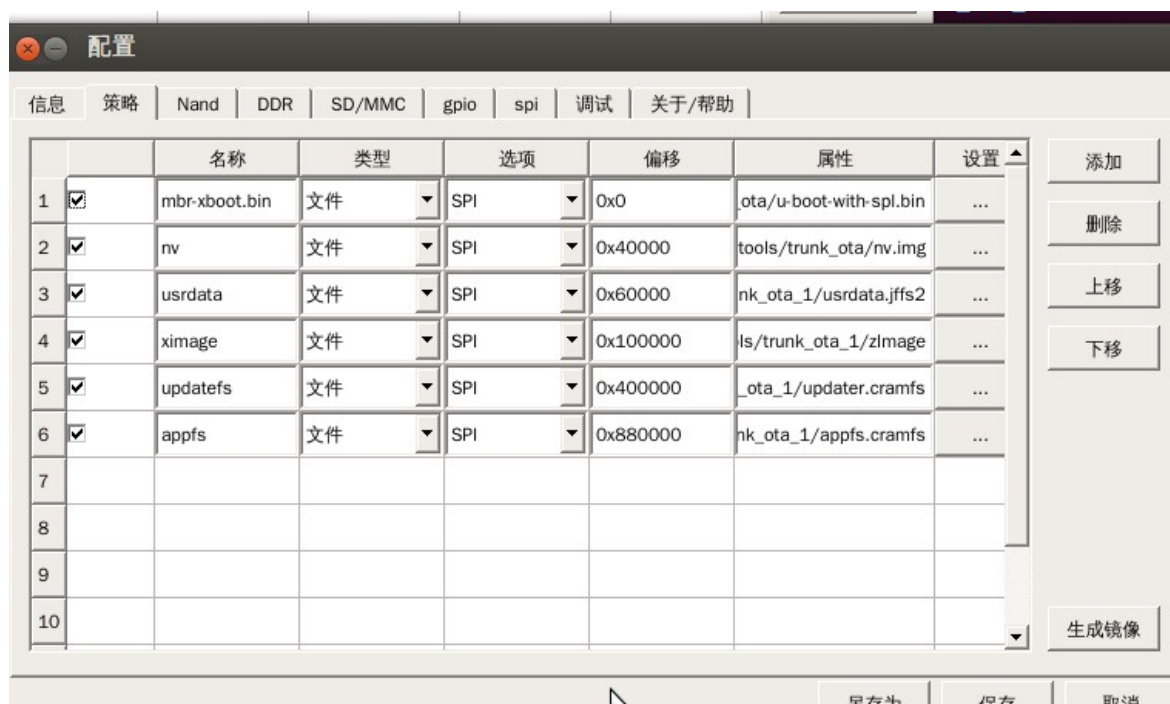
- OTA 源码位于目录 moztart/src/updater
- 先编译 update.c 和 wr\_flag.c，生成两个可执行文件 update 和 wr\_flag
- wr\_flag 是置标志位并启动升级，update 是重启后的升级程序。
- 客户可以根据需求修改升级程序的源码。

### 6 制作 OTA 升级包上传服务器

- 把 zImage, updater.cramfs, appfs.cramfs 放在 \$(MOZART\_PATH)/tools/host-tools/update\_pack\_legacy/updatezip/image 目录下。
- mkzip.sh 是制作 ota 升级包的脚本，把升级包拆分为 1M 的小升级包。

- 执行 `./mkzip.sh 20150808` 命令，传一个参数，版本号，目前以日期为例。
- `$(MOZART_PATH)/tools/host-tools/update_pack_legacy/output` 目录中的文件为生成的升级包文件，需要上传到升级服务器。
- 上传的 URL 保存在 `$(MOZART_PATH)/rootfs/updater/usr/data/system.ini` 中的 `[Update].remote-url` 项。

## 7 烧录系统



M150 的系统分为六个不同的分区，六个不同的文件。

第一次需要烧录六个文件，烧录位置为：

- u-boot-with-spl.bin 0
- nv.img 0x40000
- usrdata.jffs2 0x60000
- zImage 0x100000
- updater.cramfs 0x400000
- appfs.cramfs 0x880000

之后可以根据需求，只烧录修改过的分区。

## 8 OTA升级接口

应用层启动升级接口：

头文件位于：./output/molib/app/usr/include/ota\_interface.h

具体接口介绍：

打开 ota 设备节点，返回fd， seturl ， geturl ， getversion 等函数调用之前先调用该函数：

```
extern void mozart_ota_init();
```

关闭 ota 设备节点：

```
extern void mozart_ota_close(fd);
```

设置OTA服务器 url 接口：

```
extern int mozart_ota_seturl(char* url,int urlsize, int fd);
```

取得 url 接口：

```
extern char* mozart_ota_geturl(int fd);
```

验证 key 函数，预留接口，用户自己实现：返回值是验证成功或失败，path是下载文件的路径

```
extern bool mozart_ota_callback_checkkey(char* path);
```

取得音响当前版本接口：

```
extern unsigned int mozart_ota_get_version(int fd);
```

启动升级接口：（启动升级前一定先把正确的升级服务器 url 设置进去）

```
extern void mozart_ota_start_update();
```

启动恢复出厂设置接口：

```
extern void mozart_ota_start_recovery();
```

## 9 使用实例：

```
#include <stdio.h>
#include <linux/rtc.h>
#include <fcntl.h>
#include <sys/ioctl.h>
#include "ota_interface.h"
```

```
int main()
{
```

```
printf("start test ota\n");
int fd = mozart_ota_init();

printf("start test set url\n");
char *tmp = "http://192.168.1.200/ota/download2";
unsigned int size = sizeof("http://192.168.1.200/ota/download2");
mozart_ota_seturl(tmp,size,fd);

printf("start test ota get url\n");
mozart_ota_geturl(fd);

printf("start test ota get software version\n");
unsigned int version = mozart_ota_get_version(fd);
printf("version is %d\n",version);

printf("start test ota close\n");
mozart_ota_close(fd);

printf("start test ota update\n");
mozart_ota_start_update();

printf("start test ota recovery\n");
mozart_ota_start_recovery();

printf("test ota finish\n");

return 0;

}
```