

君正®

## **Newton Android 开发指南**

---

Date: Aug. 2014



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

君正®

## Newton Android 开发指南

Copyright © Ingenic Semiconductor Co. Ltd 201. All rights reserved.

### 版本发布历史

Date	Revision	Revision History
八月 5, 2014	1.03	<ul style="list-style-type: none"><li>- 修改烧录时的偏移地址</li><li>- 修改烧录工具界面</li><li>- 修改制作 logo 的方法</li></ul>
七月 12, 2014	1.02	<ul style="list-style-type: none"><li>- 修改获取 Android 代码的命令</li><li>- 修改编译 Android 的过程和命令</li><li>- 修改烧录工具界面</li></ul>
四月 30, 2014	1.01	<ul style="list-style-type: none"><li>- 第一次发布</li></ul>

### 声明

本文档是提供给使用君正产品，在没有君正授权的情况下，君正不负任何责任，不提供任何明确或暗示的保证供其使用，或者知识产权的侵权除非拥有君正开发团队或者销售部门的授权。

君正产品的不是为医疗或相关设备而设计，并且也不能应用于医疗和相关的设备的应用。

所有这篇文档的内容都是初步的。君正有权在不提供提示的情况下作出修改。任何人员依赖于这篇文档的人员应该及时的联系君正获取最新的文档和勘误表。

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

---

目录

1	概述 .....	1
2	准备开发环境 .....	1
2.1	准备开发主机环境 .....	1
2.2	准备烧录主机环境 .....	3
3	获取 Newton Android 源码 .....	3
3.1	下载 repo 脚本 .....	3
3.2	下载 Android 源码 .....	3
3.3	同步至最新 tag 版本 .....	3
4	编译及烧录 .....	3
4.1	Android 工程根目录结构 .....	4
4.2	全部编译 .....	4
4.2.1	编译 bootloader .....	4
4.2.2	编译 kernel .....	4
4.2.3	编译 Android 系统文件 .....	4
4.3	单独编译 boot.img .....	4
4.4	单独编译 system.img .....	5
4.5	单独编译模块 .....	5
4.6	烧录二进制镜像文件 .....	5
5	安装 ADB 工具 .....	9
5.1	Windows 下 ADB 环境配置方法 .....	9
5.1.1	下载 SDK .....	9
5.1.2	设置运行 ADB 的环境变量 .....	9
5.1.3	查看 ADB 版本 .....	9
5.2	Windows 环境的 ADB 驱动 .....	9
5.2.1	下载 Google USB 驱动 .....	9
5.2.2	安装 ADB 驱动 .....	10
5.3	Linux 主机 ADB 环境配置方法 .....	10
5.3.1	添加/etc/udev/rules.d/51-android.rules 文件 .....	10
5.3.2	下载 SDK .....	10
5.3.3	安装 SDK .....	11
6	个性化定制 .....	11
6.1	制作 bootloader logo .....	11
6.1.1	准备原始 jpg 图片 .....	11
6.1.2	拷贝 logo 图片 .....	11
6.1.3	编译到 u-boot 中 .....	11
6.1.4	重新烧录 u-boot-with-spl-mbr-gpt.bin .....	11

---

6.2	制作开机动画.....	11
6.2.1	准备开机动画图片 .....	11
6.2.2	准备用于压缩的 bootanimation 目录 .....	11
6.2.3	将 part1, part2 文件夹和 desc.txt 压缩成 bootanimation.zip .....	12
6.2.4	替换已经制作好的动画 .....	12
7	关注君正 .....	12
7.1	新浪微博号：北京君正官微.....	13
7.2	微信号：北京君正.....	13
7.3	北京君正官网.....	13

# 1 概述

本开发指南将向读者介绍如何为君正 Newton 平台搭建 Android 的开发环境，以及如何做定制化功能开发。本指南适合系统软件开发人员使用。

本指南主要介绍以下几个方面的内容：

- 1) 如何基于 64 位 PC 机搭建基础开发和编译环境；
- 2) 如何获取 Android 源码；
- 3) 如何编译 Android 源码；
- 4) 如何安装和使用 ADB 工具；
- 5) 如何做个性化定制。

阅读本指南前，需要具有以下知识：

- 1) 熟悉 Ubuntu 操作系统，熟悉 Linux 环境变量配置，熟悉 shell 命令和 Linux 命令行操作；
- 2) 熟悉 Windows XP/7 操作系统，熟悉 Windows 上安装设备驱动的过程。

## 2 准备开发环境

本节介绍搭建 Linux 开发主机环境以及 Windows 烧录主机环境的方法。

### 2.1 准备开发主机环境

需要准备一台安装 Linux 系统的电脑作为开发主机。

Linux 主机系统要求：

- 1) 硬盘容量不少于 128GB；
- 2) RAM 不小于 4GB；
- 3) 安装 Ubuntu-12.04 64 位系统（Android 编译环境要求必须是 64 位系统）。

安装步骤：

- 1) 在开发主机上安装好 64 位 Ubuntu-12.04 系统；
- 2) 安装必需的软件包：

```
$ sudo apt-get install cpp-4.6 g++-4.6 gcc-4.6 gcc-4.6-multilib gcc g++ cpp gcc-multilib  
g++-4.6-multilib git-core git gnupg flex bison gperf build-essential zip curl libc6-dev  
libncurses5-dev:i386 x11proto-core-dev libx11-dev:i386 libreadline6-dev:i386 libgl1-mesa-glx:i386  
libgl1-mesa-dev g++-multilib mingw32 tofrodos python-markdown libxml2-utils xsltproc  
zlib1g-dev:i386 ia32-libs gawk qt4-dev-tools libgl1-mesa-dri:i386 libglapi-mesa:i386  
libncurses5-dev libqt3-mt-dev u-boot-tools
```

Newton Android SDK 开发环境与 Google 官方完全兼容，更详细的编译环境建立方法，请参考 android 的官方网站：<http://source.android.com/source/initializing.html>

- 3) 安装 JDK

推荐安装 JDK 版本为 jdk-6u37-linux-x64，下载页面为：

<http://www.oracle.com/technetwork/java/javase/downloads/java-archive-downloads-javase6-419409.html>


















Java SE Development Kit 6u37		
You must accept the <a href="#">Oracle Binary Code License Agreement for Java SE</a> to download this software.		
<input type="radio"/> Accept License Agreement <input checked="" type="radio"/> Decline License Agreement		
Product / File Description	File Size	Download
Linux x86	65.43 MB	 <a href="#">jdk-6u37-linux-i586-rpm.bin</a>
Linux x86	68.44 MB	 <a href="#">jdk-6u37-linux-i586.bin</a>
Linux x64	65.65 MB	 <a href="#">jdk-6u37-linux-x64-rpm.bin</a>
Linux x64	68.71 MB	 <a href="#">jdk-6u37-linux-x64.bin</a>
Solaris x86	68.35 MB	 <a href="#">jdk-6u37-solaris-i586.sh</a>
Solaris x86	119.94 MB	 <a href="#">jdk-6u37-solaris-i586.tar.Z</a>
Solaris SPARC	73.36 MB	 <a href="#">jdk-6u37-solaris-sparc.sh</a>
Solaris SPARC	124.71 MB	 <a href="#">jdk-6u37-solaris-sparc.tar.Z</a>
Solaris SPARC 64-bit	12.13 MB	 <a href="#">jdk-6u37-solaris-sparcv9.sh</a>
Solaris SPARC 64-bit	15.42 MB	 <a href="#">jdk-6u37-solaris-sparcv9.tar.Z</a>
Solaris x64	8.45 MB	 <a href="#">jdk-6u37-solaris-x64.sh</a>
Solaris x64	12.18 MB	 <a href="#">jdk-6u37-solaris-x64.tar.Z</a>
Windows x86	69.72 MB	 <a href="#">jdk-6u37-windows-i586.exe</a>
Windows x64	59.73 MB	 <a href="#">jdk-6u37-windows-x64.exe</a>
Linux Intel Itanium	53.95 MB	 <a href="#">jdk-6u37-linux-ia64-rpm.bin</a>
Linux Intel Itanium	60.67 MB	 <a href="#">jdk-6u37-linux-ia64.bin</a>
Windows Intel Itanium	57.89 MB	 <a href="#">jdk-6u37-windows-ia64.exe</a>
<a href="#">Back to top</a>		

图2-1 Java JDK 下载页面

选择下载 `jdk-6u37-linux-x64.bin`

JDK 安装步骤:

- 1) 拷贝 `jdk-6u37-linux-x64.bin` 到 `/usr/java/` 目录中;
- 2) 修改可执行权限:  
\$ `sudo chmod u+x /usr/java/jdk-6u37-linux-x64.bin`
- 3) 执行下面命令开始安装:  
\$ `sudo /usr/java/jdk-6u37-linux-x64.bin`
- 4) 安装完毕, 配置系统环境变量, 在这里介绍两种方法:

方法 1 使用超级用户权限在 `/etc/environment` 中增加:

```
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/java/jdk1.6.0_37/bin"
```

```
JAVA_HOME=/usr/java/jdk1.6.0_37
```

```
JRE_HOME=/usr/java/jdk1.6.0_37/jre
```

```
CLASSPATH=$CLASSPATH:$JAVA_HOME/lib:$JRE_HOME/lib
```

保存后退出。重新登录用户, 环境变量会自动生效。

方法 2. 使用超级用户权限在 `/etc/profile` 中增加:

```
export JAVA_HOME=/usr/java/jdk1.6.0_37
```

```
export JRE_HOME=/usr/java/jdk1.6.0_37/jre
```

```
export CLASSPATH=$CLASSPATH:$JAVA_HOME/lib:$JRE_HOME/lib
```

```
export PATH=$JAVA_HOME/bin:$JAVA_HOME/jre/bin:$PATH
```

保存后退出。重新登录用户，环境变量会自动生效。

### 5) 查看 Java 版本

重新登陆后，在任一目录下执行下面的命令查看 Java 版本，显示下图所示信息，说明已经配置成功。

```
root@Ubuntu-1204:~# java -version
java version "1.6.0_37"
Java(TM) SE Runtime Environment (build 1.6.0_37-b06)
Java HotSpot(TM) 64-Bit Server VM (build 20.12-b01, mixed mode)
root@Ubuntu-1204:~#
```

图2-2 Java JDK 版本信息

## 2.2 准备烧录主机环境

君正提供运行在 Windows XP 和 Windows 7 上的 USB 烧录工具。详情请阅读“[Howto Burn Newton Demo](#)”。

# 3 获取 Newton Android 源码

Newton Android 源码可从君正 GIT 仓库下载，并使用 repo 对多个 GIT 仓库进行管理。按照下面步骤进行操作：

### 3.1 下载 repo 脚本

```
$ mkdir newton-android
$ cd newton-android
$ wget http://git.ingenic.cn:8082/bj/repo
$ chmod +x repo
```

### 3.2 下载 Android 源码

```
$ ./repo init -u http://git.ingenic.cn:8082/gerrit/AOSP/platform/manifest -b android-4.3-newton-4775
$ ./repo sync
```

以上操作开始下载代码，代码大约有 20~30GB，需要等待一段时间，下载速度依赖于实际网络条件。

### 3.3 同步至最新 tag 版本

```
$ ./repo forall -c "git reset --hard ingenic-android4.3.0-kernel3.0.8-newton-vx.x-yyyymmdd"
```

其中“ingenic-android4.3.0-kernel3.0.8-newton-vx.x-yyyymmdd”代表某一个发布版本的 tag 标签。通过下列命令查看具体 tag 信息，并使用上面命令同步到最新的 tag 版本。

```
$ cd kernel
$ git tag | grep newton
```

# 4 编译及烧录

本节介绍如何编译 Android 代码以及如何进行烧录。如果还没有下载 Android 代码，请回到上章。假设已经把 Android 代码下载至 newton-android 目录下，后面很多操作命令会在该目录下执行。为了方便理解，在之后的文档描述中，有时会用 Android 工程根目录代指该目录。

## 4.1 Android 工程根目录结构

代码下载成功后，会得到如下图所示的一个目录结构：

```
root@Ubuntu-1204:~/newton-android# ls
abi          dalvik       docs          kernel        ndk           system
bionic       developers   external      libcore       packages      vendor
bootable     development  frameworks    libnativehelper  pdk
build        device       hardware      Makefile      prebuilts
root@Ubuntu-1204:~/newton-android#
```

图4-1 Android 工程根目录结构

其中：

Bootloader 的源码位于：./bootable/bootloader/uboot

Android 的 linux 内核源码位于：./kernel

Android 的设备配置位于：./device/ingenic/newton

## 4.2 全部编译

### 4.2.1 编译 bootloader

bootloader 的源码位于：bootable/bootloader/uboot。在 Android 工程根目录下执行以下命令：

```
$ source build/envsetup.sh
```

```
$ lunch full_newton-userdebug
```

```
$ cd bootable/bootloader/uboot/
```

```
$ make distclean
```

```
$ make newton_android_msc0
```

编译完成后，会在当前目录下生成 u-boot-with-spl-mbr-gpt.bin 文件。

### 4.2.2 编译 kernel

在 Android 工程根目录下执行以下命令：

```
$ cd ../../kernel
```

```
$ make newton_android_msc_defconfig
```

```
$ make zImage
```

```
$ cp arch/mips/boot/compressed/zImage ../device/ingenic/newton/kernel
```

把编译生成的 zImage 更新到 device/ingenic/newton 目录下相对应的文件。

### 4.2.3 编译 Android 系统文件

进到 Android 工程根目录下，执行以下命令：

```
$ cd ..
```

```
$ make
```

整个编译过程需要持续几个小时，具体的时间取决于主机的性能情况。编译完成后，会在 out/target/product/newton/ 目录下生成可以烧录到开发板的文件：boot.img，recovery.img 和 system.img。

## 4.3 单独编译 boot.img

在 Android 工程根目录下执行以下命令：

```
$ source build/envsetup.sh
```



```
$ lunch full_newton-userdebug
$ cd kernel/
$ make newton_android_msc_defconfig
$ make zImage
$ cp arch/mips/boot/compressed/zImage ../device/ingenic/newton/kernel
$ cd ..
$ make bootimage
```

生成的 boot.img 位于 out/target/product/newton/目录下。

#### 4.4 单独编译 system.img

在 Android 工程根目录下执行以下命令：

```
$ source build/envsetup.sh
$ lunch full_newton-userdebug
$ make systemimage
```

#### 4.5 单独编译模块

在 Android 工程根目录下先执行以下命令：

```
$ source build/envsetup.sh
$ lunch full_newton-userdebug
```

进入要编译的模块目录下，然后执行 mm 命令，例如：

```
$ cd packages/apps/Calendar
$ mm
```

#### 4.6 烧录二进制镜像文件

所有文件编译成功后，就可以进行烧录。请关注 Ingenic 官网获取最新的烧录工具进行烧录。对于 Android，需要烧录的二进制文件为：

- u-boot-with-spl-mbr-gpt.bin
- boot.img
- recovery.img
- system.img

烧录文件的分区信息：

JZ4775 Newton 开发板烧录文件分区表

Board	Storage	File	Offset(B)	Option	Configuration
Newton	Storage: EMMC 4G, 512B/Sector Mem: Mobile DDR	u-boot-with-spl-mbr-gpt.bin	0	MMC0	Newton_mmc_lpddr.cfg
		boot.img	0x300000	MMC0	
		recovery.img	0xb00000	MMC0	
		system.img	0x3800000	MMC0	

在烧录工具主界面，点击配置按钮，选择配置菜单：

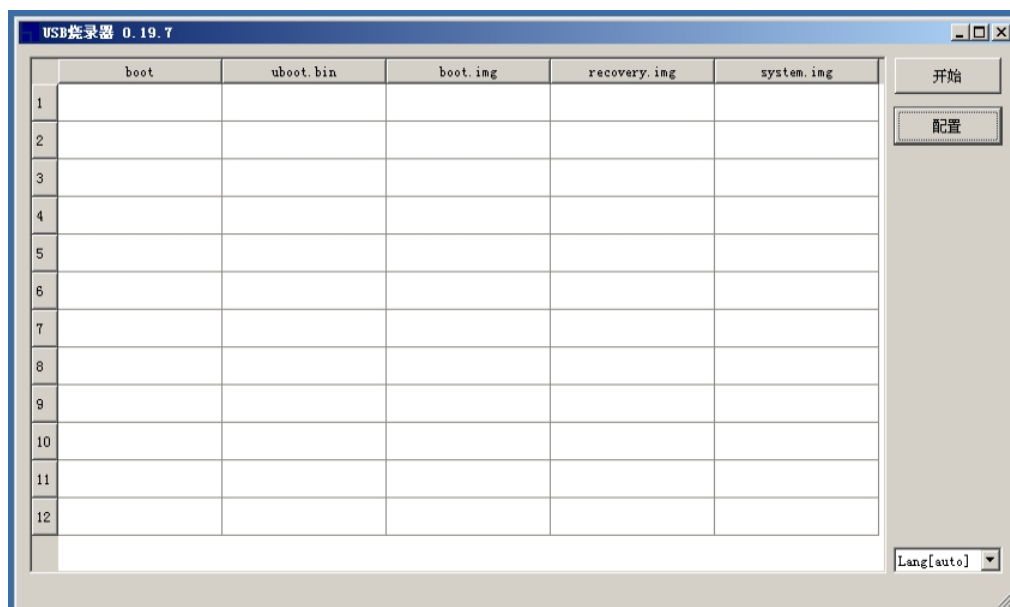


图4-2 烧录工具主界面

确认配置文件是 newton\_mmc\_lpddr.cfg。

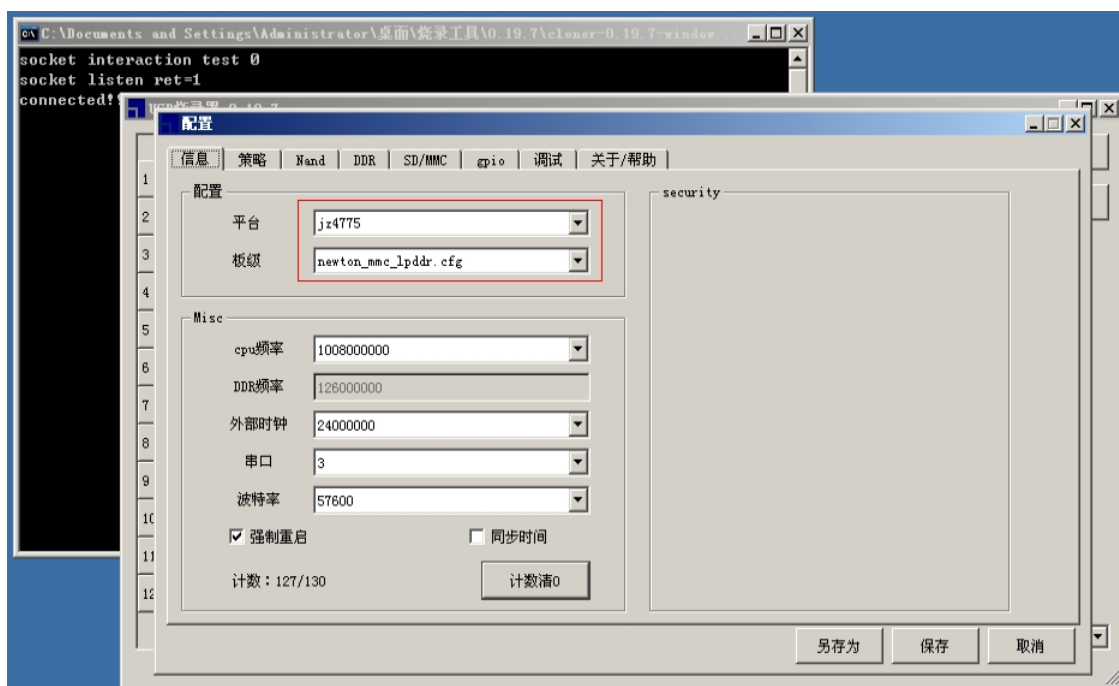


图4-3 确认配置文件

确认烧录的文件名及其对应的偏移位置，可以按照需要选择或者取消选择要烧录的文件。

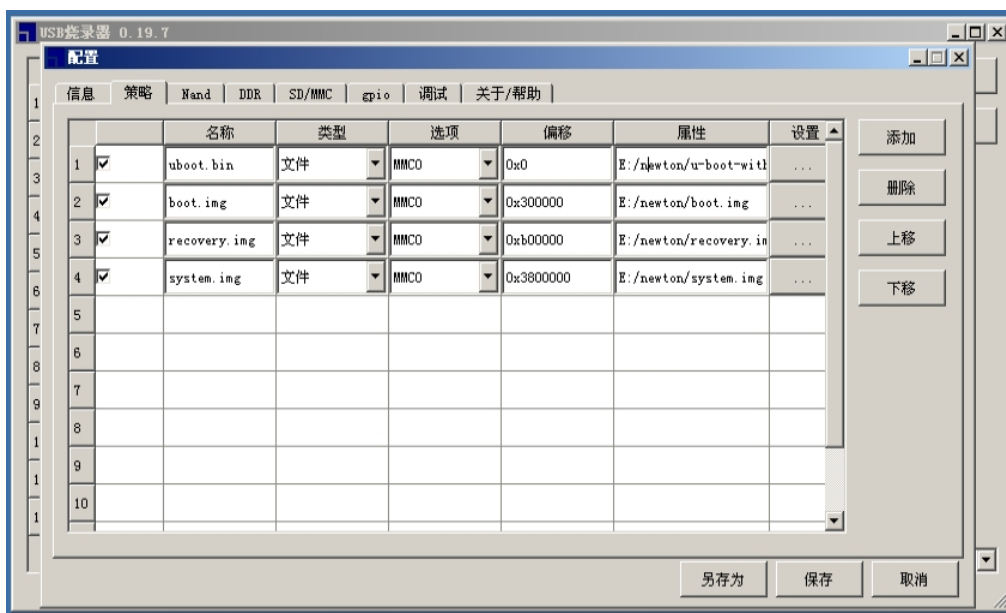


图4-4 烧录位置配置界面-全部烧录

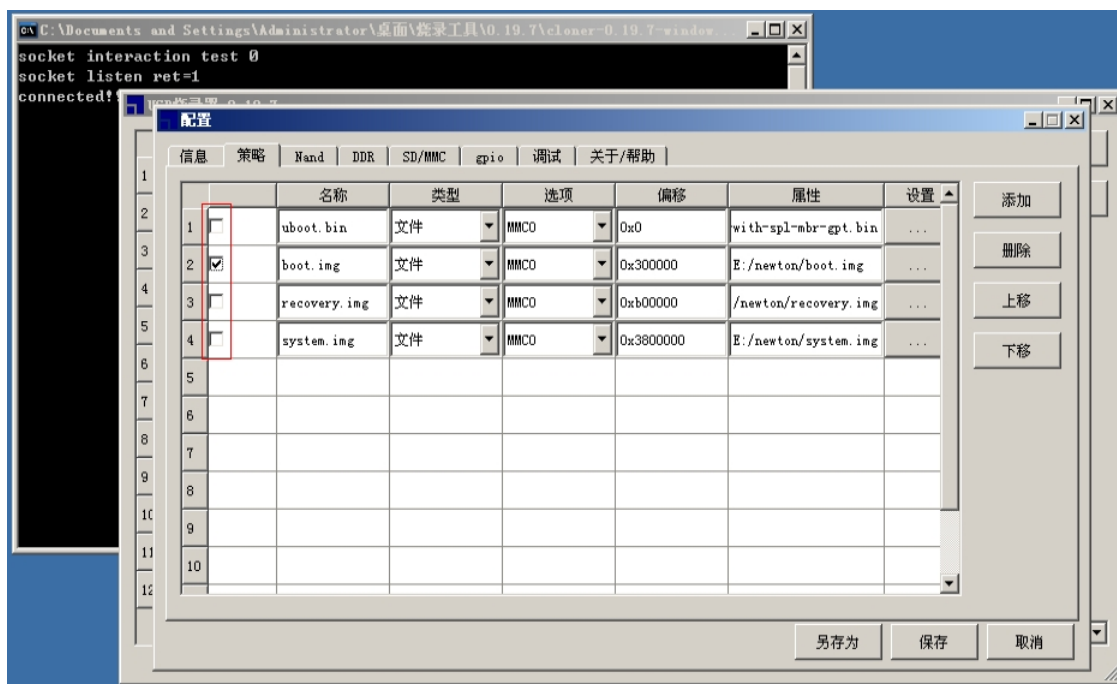


图4-5 烧录位置配置界面-部分烧录（只烧录 boot.img）

如果需要完全重新烧录，请选择强制全部擦除。

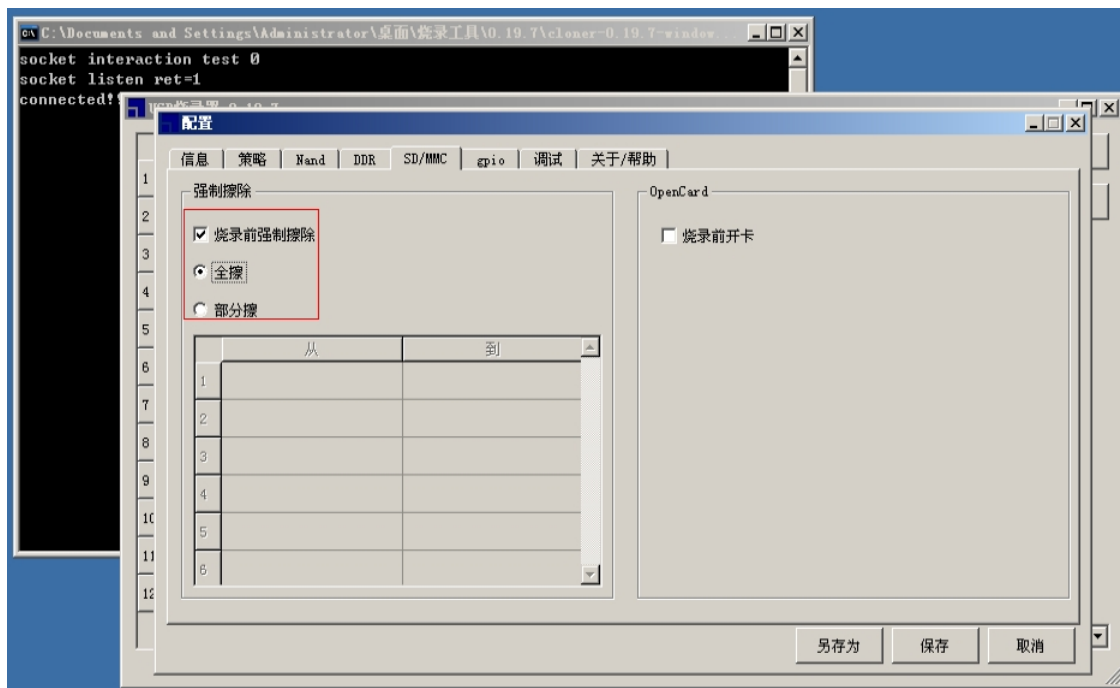


图4-6 强制全擦

如果不需要完全重新烧录，请取消选择强制全部擦除。

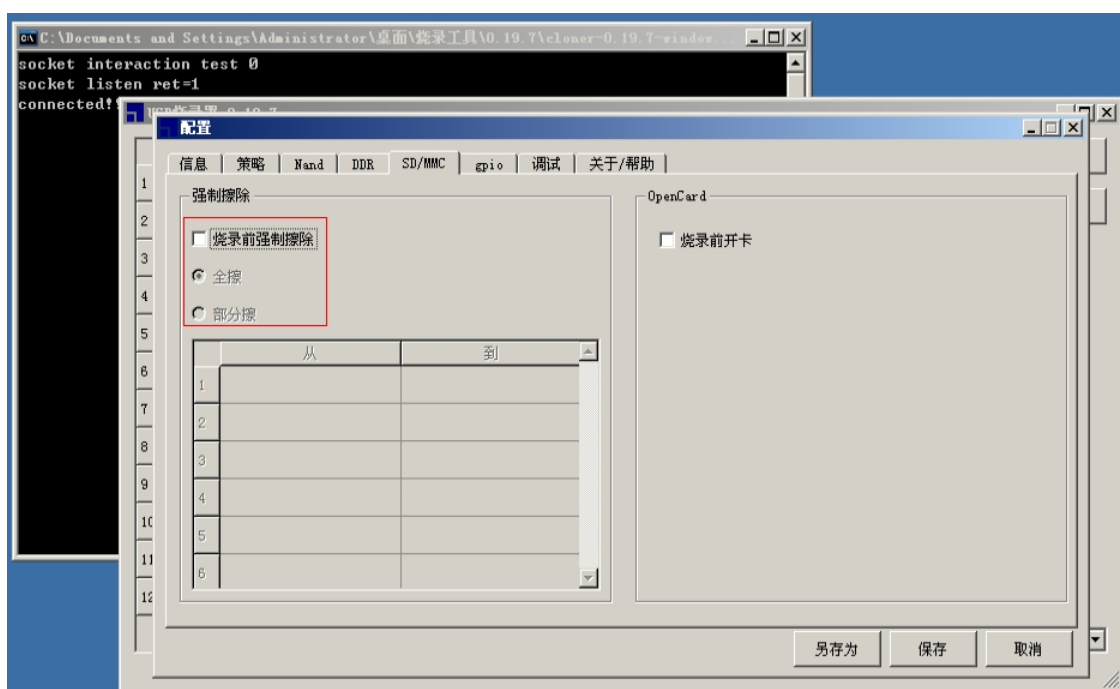


图4-7 取消强制全擦

## 5 安装 ADB 工具

### 5.1 Windows 下 ADB 环境配置方法

#### 5.1.1 下载 SDK

从 Android 开发网站选择合适的 Windows 版本的 SDK 并下载，SDK 的网址为：

<http://developer.android.com/sdk/index.html>

ADT Bundle			
Platform	Package	Size	MD5 Checksum
Windows 32-bit	<a href="#">adt-bundle-windows-x86-20140321.zip</a>	535085536 bytes	b61495a6bf591cc374c31bce4fc46ec0
Windows 64-bit	<a href="#">adt-bundle-windows-x86_64-20140321.zip</a>	535287324 bytes	a6f4699bbdc5a29b371ed60610535651
Mac OS X 64-bit	<a href="#">adt-bundle-mac-x86_64-20140321.zip</a>	501955296 bytes	4a08649cea9b098cdf7349f452294014
Linux 32-bit	<a href="#">adt-bundle-linux-x86-20140321.zip</a>	527971926 bytes	943ae4d28fe7c79108c8bf2aafd5e6d2
Linux 64-bit	<a href="#">adt-bundle-linux-x86_64-20140321.zip</a>	528187678 bytes	f2a2153b5c7dbaeb86b550bf4f770c36

图5-1 Android Windows SDK 下载页面

例如，对于 Windows 32 位系统，选择下载 `adt-bundle-windows-x86-20140321.zip`，解压到目录。

#### 5.1.2 设置运行 ADB 的环境变量

以 Windows7 为例，右击“计算机”，选择“属性”，点击“高级系统设置”弹出“系统属性”对话框，选择“高级”，点击“环境变量”弹出对话框，在“系统变量(S)”中找到“Path”变量，然后“编辑”，将 `adb` 所在目录的路径加到“变量值”结尾处，前面用“;”隔开。

#### 5.1.3 查看 ADB 版本

运行 cmd，可以用 `adb version` 命令查看 `adb` 的版本，若显示下图所示信息，说明 `adb` 已经配置成功。

```
C:\Users\Administrator>adb version
Android Debug Bridge version 1.0.31
```

图5-2 ADB 的版本查看

### 5.2 Windows 环境的 ADB 驱动

#### 5.2.1 下载 Google USB 驱动

下载地址：<http://developer.android.com/sdk/win-usb.html>

## Google USB Driver

The Google USB Driver is **required for Windows only** in order to perform [adb](#) debugging with any of the **Google Nexus devices**. The one exception is the Galaxy Nexus: the driver for Galaxy Nexus is distributed by [Samsung](#) (listed as model SCH-I515).

Windows drivers for all other devices are provided by the respective hardware manufacturer, as listed in the [OEM USB Drivers](#) document.

**Note:** If you're developing on Mac OS X or Linux, then you **do not** need to install a USB driver. To start developing with your device, read [Using Hardware Devices](#).

### Downloading the Google USB Driver

The Google USB Driver for Windows is available for download as an optional SDK component. You need the driver only if you are developing on Windows and want to connect a Google Android-powered device (such as a Nexus 7) to your development environment over USB.

#### IN THIS DOCUMENT

[Downloading the Google USB Driver](#)

#### SEE ALSO

[Installing a USB Driver](#)  
[Using Hardware Devices](#)

#### GET IT

[Download Google USB Driver](#)

latest\_usb\_driver\_windows.zip

图5-3 Google USB 驱动下载页面

点击“Download Google USB Driver”下载最新的驱动 latest\_usb\_driver\_windows.zip

## 5.2.2 安装 ADB 驱动

关于如何安装 Windows 环境下的 ADB 驱动，请参考 Android 官方网站：

<http://developer.android.com/tools/extras/oem-usb.html#InstallingDriver>

## 5.3 Linux 主机 ADB 环境配置方法

### 5.3.1 添加/etc/udev/rules.d/51-android.rules 文件

对于 Linux 用户，需要添加/etc/udev/rules.d/51-android.rules 文件，内容如下：

```
SUBSYSTEM=="usb|usb_device", ATTR{idVendor}=="18d1", MODE="0666", GROUP="plugdev"
```

### 5.3.2 下载 SDK

从 Android 开发网站选择合适的 Linux 版本的 SDK 并下载，SDK 的网址为：

<http://developer.android.com/sdk/index.html>

#### ADT Bundle

Platform	Package	Size	MD5 Checksum
Windows 32-bit	<a href="#">adt-bundle-windows-x86-20140321.zip</a>	535085536 bytes	b61495a6bf591cc374c31bce4fc46ec0
Windows 64-bit	<a href="#">adt-bundle-windows-x86_64-20140321.zip</a>	535287324 bytes	a6f4699bbdc5a29b371ed60610535651
Mac OS X 64-bit	<a href="#">adt-bundle-mac-x86_64-20140321.zip</a>	501955296 bytes	4a08649cea9b098cdf7349f452294014
Linux 32-bit	<a href="#">adt-bundle-linux-x86-20140321.zip</a>	527971926 bytes	943ae4d28fe7c79108c8bf2aafd5e6d2
Linux 64-bit	<a href="#">adt-bundle-linux-x86_64-20140321.zip</a>	528187678 bytes	f2a2153b5c7dbaeb86b550bf4f770c36

图5-4 Android Linux SDK 下载页面

例如，对于 64 位的 Ubuntu 系统，选择下载 adt-bundle-linux-x86\_64-20140321.zip

### 5.3.3 安装 SDK

将下载的 SDK 解压安装到任意目录下。然后修改`~/.bash_profile` 文件，设置运行环境指向 `sdk` 的 `platform-tools/`目录。具体是打开`~/.bash_profile` 文件，在里面加入一行：

```
export PATH=<ANDROID_SDK_INSTALL_PATH>/platform-tools: ${PATH}
```

设置好以后，就可以在任意目录下使用 `adb` 命令了。

## 6 个性化定制

### 6.1 制作 bootloader logo

#### 6.1.1 准备原始 jpg 图片

建议使用与屏幕分辨率相同或稍小的 jpg 图片制作 logo，起名为 `logo.jpg`。例如屏幕的分辨率为 `288x192`，将图片制作成 `288x192` 的 jpg 图片，保存为 `logo.jpg`。

注意：如果显示有问题，请将图片的分辨率的宽高改为偶数，如 `100x100`。

#### 6.1.2 拷贝 logo 图片

将第一步做的 `logo.jpg` 文件拷贝到 `u-boot` 工程目录下的 `bootable/bootloader/uboot/tools/logos/` 目录下。

```
$ cp logo.jpg bootable/bootloader/uboot/tools/logos/logo.jpg
```

#### 6.1.3 编译到 u-boot 中

1) 修改相关文件

进入 `u-boot` 工程根目录，执行以下命令

```
$ vim tools/ingenic-tools/logo.mk
```

将 `BOOT_LOGO_JPG ?= $(TOPDIR)/tools/logos/ingenic.jpg` 中的 `ingenic.jpg` 修改成为 `logo.jpg`。

2) 编译 `u-boot`

```
$ make clean
```

```
$ make newton_android_msc0
```

#### 6.1.4 重新烧录 u-boot-with-spl-mbr-gpt.bin

定制完成，启动后，即可显示最新的 logo。

### 6.2 制作开机动画

#### 6.2.1 准备开机动画图片

根据屏幕的分辨率制作同样的图片，动画有几帧，将对应的几张 jpg 图片，按播放顺序命名，如：`all_0001.jpg` , `all_0002.jpg` ..... 等。

#### 6.2.2 准备用于压缩的 bootanimation 目录

```
$ mkdir bootanimation
```

```
$ cd bootanimation
```

```
$ mkdir part1 part2
```

```
$ cp ../*.jpg part1
```

```
$ cp ../*.jpe part2
```

```
$ touch desc.txt
```

两个目录 **part1**, **part2**, 分别是播放一遍的部分和循环播放直到开机部分。

按照以下内容编辑 **desc.txt**, 保存退出:

```
288 192 30
```

```
p 1 0 part1
```

```
p 0 0 part2
```

第一行分别为图片分辨率宽、图片分辨率高、每秒帧率。

第二行中 **p** 标识符, 不必理会, **1** 代表循环次数为 1 次, **0** 代表阶段间隔时间为 0, **part1** 是文件夹名字。

第三行中 **p** 标识符, 不必理会, **0** 代表无限循环, **0** 代表阶段间隔时间为 0, **part2** 是文件夹名字。

确认 **bootanimation** 目录下除了上述文件外, 没有其他文件存在。**Bootanimation** 目录树:

```
├── desc.txt
├── part1
│   ├── all_0001.jpg
│   ├── all_0002.jpg
│   └── all_0003.jpg
└── part2
    ├── all_0001.jpg
    ├── all_0002.jpg
    └── all_0003.jpg
```

### 6.2.3 将 **part1**, **part2** 文件夹和 **desc.txt** 压缩成 **bootanimation.zip**

```
$ zip -0 bootanimation.zip part1/*.jpg part2/*.jpg desc.txt
```

### 6.2.4 替换已经制作好的动画

如果已经有制作好的动画 **bootanimation.zip**, 将其拷贝到 **/system/media/bootanimation.zip** 或 **/data/local/bootanimation.zip**, 即可实现修改开机动画。

## 7 关注君正

欢迎关注君正的微博和微信, 第一时间获得君正最新动态。



## 7.1 新浪微博号：北京君正官微



图7-1 北京君正新浪微博

## 7.2 微信号：北京君正



图7-2 北京君正微信

## 7.3 北京君正官网

官网主页：<http://www.ingenic.cn/>

芯片产品主页：<http://www.ingenic.cn/cn/cn/product.php?fid=782>

开发支持主页：<http://www.ingenic.cn/cn/cn/kfzc.php>

如有技术方面问题和建议，请联系君正技术支持：

邮箱：[support@ingenic.cn](mailto:support@ingenic.cn)

电话：4008-900-223