

# **Ingenic<sup>®</sup> Newton**

## **Linux Development Guide**

---

Date: Aug. 2014



**北京君正集成电路有限公司**  
**Ingenic Semiconductor Co. Ltd**

# Ingenic<sup>®</sup> Newton

## Linux Development Guide

Copyright © Ingenic Semiconductor Co. Ltd 2005 - 2014. All rights reserved.

### Release history

Date	Revision	Change
Jul. 2014	1.02	Modified the method to download source code Modified the description of toolchain Modified the description of uboot and kernel Modified the method to make rootfs Modified the description of flashing
Apr. 2014	1.01	First release

### 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.

**Ingenic Semiconductor Co., Ltd.**

**Ingenic Headquarters, East Bldg. 14, Courtyard #10,  
Xibeiwang East Road, Haidian Dist., Beijing 100193, China  
Tel: 86-10-56345000  
Fax: 86-10-56345001  
Http: //www.ingenic.cn**

---

CONTENT

1	OVERVIEW .....	1
2	SETUP THE ENVIRONMENT .....	1
2.1	HOST COMPUTER REQUIREMENTS.....	1
2.2	SETUP THE DEVELOPMENT ENVIRONMENT.....	1
3	GET SOURCE CODE .....	1
3.1	INSTALLING REPO .....	1
3.2	DOWNLOAD THE LINUX SOURCE .....	1
4	BUILD THE NEWTON LINUX SDK .....	1
4.1	SDK DIRECTORY TREE .....	1
4.1.1	u-boot.....	2
4.1.2	kernel .....	3
4.2	SET ENVIROMENT FOR TOOLCHAIN .....	4
4.3	COMPILE THE U-BOOT.....	4
4.4	COMPILE THE LINUX KERNEL.....	5
4.5	CREATE AN EXT4 ROOT FS IMAGE.....	5
4.6	PARTITIONS TABLE .....	5
4.7	BURN BINARY FILES .....	6
4.8	BOOT NEWTON BOARD .....	6
4.9	READMES .....	7



# 1 Overview

This guide describes how to on how to set up the environment for Linux developing and how to use the Newton Linux SDK. This document is written for the system software engineers.

Before reading this document, you are suggested that:

- Familiar with the Ubuntu, Linux environment variables and shell commands
- Familiar with the Windows OS, and know how to install a device driver on it

## 2 Setup the Environment

### 2.1 Host Computer Requirements

The host computer requirements are:

- a. The Hard Disk capacity is not less than 128GB.
- b. The DRAM size is not less than 4GB.
- c. Ubuntu-12.04, 64-bit is recommended.

### 2.2 Setup the Development Environment

- a. Install Ubuntu-12.04 on your computer.
- b. Install required packages:  
`$ sudo apt-get install git build-essentials u-boot-tools`
- c. Install ia32-libs (It is required to run 32-bit programs on an Ubuntu-12.04 64-bit system)  
`$ sudo apt-get install ia32-libs`

## 3 Get Source Code

### 3.1 Installing Repo

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

### 3.2 Download the Linux source

```
$ ./repo init -u http://git.ingenic.cn:8082/gerrit/linux/manifest.git -b newton-master
$ ./repo sync
```

## 4 Build the Newton Linux SDK

### 4.1 SDK Directory Tree

The Newton Linux SDK contains binaries and tools, help documents, and source codes of the

bootloader, kernel and user space applications. The top-level SDK directory trees are:

binaries/:

- host-tool/: Tools maybe needed by PC
- rootfs/rootfs-newton/mkrootfs: Scripts and tools for making rootfs
- rootfs/rootfs-newton/rootfs.ext2: rootfs which can bootup already
- toolchain/mips-gcc472-glibc216/: Tool chain

documents/: Readmes

sources/: Codes

- bootloader/u-boot/: u-boot
- kernel/linux-3.0.8/: Linux3.0.8
- application/: Third party Apps

#### 4.1.1 u-boot

```
[~yyhuang:u-boot]:$ ls
api      config.mk  drivers   lib        mkconfig  rules.mk   tools     u-boot.srec
arch     COPYING   dts       MAINTAINERS nand_spl  snapshot.commit u-boot    u-boot-with-spl.bin
board    CREDITS   examples  MAKEALL    net       spl        u-boot.bin u-boot-with-spl-mbr.bin
boards.cfg disk      fs        makecp.sh  post      System.map u-boot.lds
common   doc       include   Makefile   README    test       u-boot.map
```

Figure 4-1 u-boot

api: apis

arch:

- cpu: Jz4775 related files are located in arch/mips/cpu/xburst/:
  - cpu.c Initializing, caches
  - jz4775/jz4775.c timer, PLL
  - jz\_serial.c UART
  - interrupt.c Interrupts
  - start.S The enter of u-boot
- lib: Libs
- Include: Heads

board: Configuration for boards, Newton related files are in ingenic/newton

Boards.cfg: Register for boards

common: Commands

Config.mk: Config for building and others

disk: Codes for disk

doc: Documents for uboot

drivers: Drivers

fs: FS

include: Heads, Newton related files are in include/configs/newton.h

lib: Libs

nand\_spl: Nand related

tools: Tools

Test: Tests

net: Net

## 4.1.2 kernel

```
-----[ PWD = ~/work/dev_newton_master/sources/kernel/linux-3.0.8 ]
[-yyhuang:linux-3.0.8]:$ ls
arch      crypto    fs        Kbuild    MAINTAINERS  mm          REPORTING-BUGS  sound      virt
block     Documentation  include  Kconfig    makecp.sh    Module.symvers  samples         System.map  vmlinux
COPYING   drivers    init      kernel     Makefile     net           scripts         tools       vmlinux.o
CREDITS   firmware   ipc       lib        mklinux.sh   README        security        usr
```

Figure 4-2 kernel

arch/mips/: MIPS

```
-----[ PWD = ~/work/dev_newton_master/sources/kernel/linux-3.0.8/arch/mips ]
[-yyhuang:mips]:$ ls
alchemy    boot      dec       jz4740     kernel     Makefile    mti-sead3    pnx833x    sgi-ip22    txx9
ar7        built-in.o  emma      Kbuild     lantiq     math-emu    netlogic     pnx8550    sgi-ip27    vr4lxx
ath79      cavium-octeon fw        Kbuild.platforms  lasat     mipssim     oprofile     power      sgi-ip32    wrppmc
bcm47xx    cobalt     include   Kconfig    lib        mm          pci          powertv    sibyte     xburst
bcm63xx    configs    jazz      Kconfig.debug  loongson  mti-malta   pmc-sierra   rb532     sni
```

Figure 4-3 arch/mips 目录

- kernel/: Common codes for kernel
- mm/: Memory Manager
- lib/: Libs
- xburst/soc-4775/: JZ4775 related
  - board/s2523b\_15m/: Newton related
  - common/: Common files for Jz4775
  - include /: Heads for Jz4775
- boot/compressed/: ulmage will be created here
- Kconfig: MIPS 体系配置文件
- Makefile: MIPS 通用 makefile
- configs/: Configs

include/asm-generic/: Generic heads

Sound:

- oss/jzsound/: OSS driver
- devices/codecs: Codecs driver
- interface/: Interface

kernel: Common codes for kernel

mm/: Memory Manager

lib/: Libs

init/: Init

ipc/: IPC

net/: Net

fs/: FS

-jffs2/: JFFS/JFFS2

-ubifs/: UBIFS

drivers/:

```
-----[ PWD = ~/work/dev_newton_master/sources/kernel/linux-3.0.8/drivers ]
[-yyhuang:drivers]:$ ls
accessibility bluetooth crypto gpu input media nfc pnp sbus switch
acpi built-in.o dca hid isdn memstick nubus power scsi target
amba cdrom dio hwmon Kconfig message of pps sfi tc
ata char dma hwspinlock leds mfd oprofile ps3 sh telephony
atm clk edac i2c lguest misc parisc ptp slpt thermal
auxdisplay clocksource eisa ide macintosh mmc parport rapidio sn tty
base connector firewire idle Makefile mtd pci regulator spi uio
bcma cpufreq firmware ieee802154 mca nand pcmcia rtc ssb usb
block cpuidle gpio infiniband md net platform s390 staging uwb
```

**Figure 4-4 drivers 目录**

- block/: Block drivers
- char/: Char drivers
- cpufreq: Freq related drivers
- input/: Input device drivers(keyboard, mouse, touchscreen...)
- mmc/: MMC/SD
- mtd/: MTD
- mtd/ubi/: UBI
- net/: Net
- tty/serial/: UART
- spi/: SPI
- usb/host: USB host
- usb/otg: USB otg
- usb/dwc2: USB dwc2
- usb/gadget: USB device gadget
- video/jz4780\_fb: LCD framebuffer
- misc/jz\_cim: Camera

## 4.2 Set Enviroment for Toolchain

\$ export PATH= SDK\_ROOT/binaries/toolchain/ mips-gcc472-glibc216/bin:\$PATH

\$ export CROSS\_COMPILE=mips-linux-gnu-

Then use " which mips-linux-gnu-gcc"check the toolchain.

```
[-yyhuang:dev_newton_master]:$ which mips-linux-gnu-gcc
~/work/dev_newton_master/binaries/toolchain/mips-gcc472-glibc216/bin/mips-linux-gnu-gcc
```

**Figure 4-5 Android Home**

## 4.3 Compile the U-Boot

\$ make distclean

\$ make newton\_msc\_config

\$ make

This will create u-boot-with-spl-mbr.bin.



## 4.4 Compile the Linux Kernel

```
$ cd sources/kernel/linux-3.0.8
```

```
$ make newton_msc_defconfig
```

```
$ make ulmage
```

This will create ulmage in arch/mips/boot/compressed.

## 4.5 Create an EXT4 Root FS Image

Binary of rootfs which can be used for bootup has been uploaded into the SDK: binaries/rootfs/rootfs-newton/rootfs.ext2. But most of time root must be modified to fit the certain situation. If you just want to put some binaries into your root, please refer the followint steps, **note that all the steps MUST be run as root:**

```
$ sudo -s
# mkdir rootfs
# cd rootfs/
# tar xvf ../rootfs.tar.bz2
# cd ../
# ./mk_rootfsimg.sh rootfs
```

Rootfs is based on buildroot, please refer “How to use BuildRoot” for more detail.

## 4.6 Partitions Table

**Newton Board Partitions Table**

Board	Hardware	File	Offset(B)	Option	Configuration
Newton	Storage: EMMC 4G, 512B/Sector Mem: Mobile DDR	u-boot-with-spl-mbr.bin	0	MMC0	Newton_mmc_lpddr.cfg
		ulmage	0x300000	MMC0	
		rootfs.img/rootfs.ext2	0x3800000	MMC0	

To configure this parameters, click the “Configure” button:

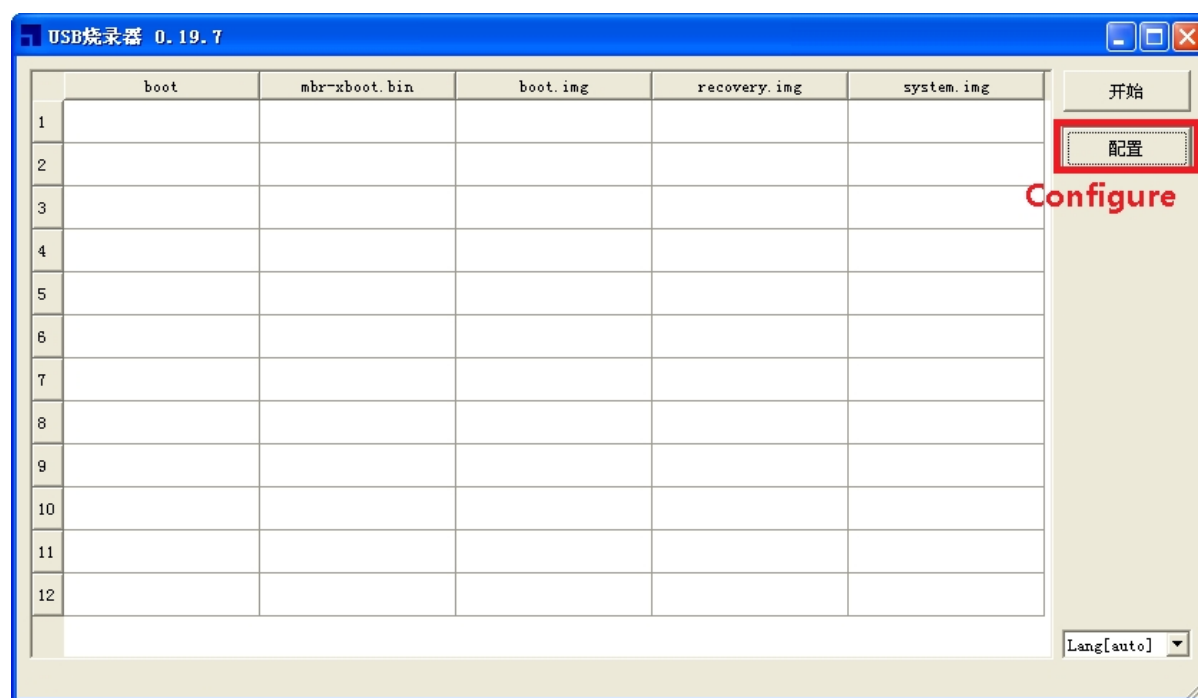


Figure 4-6 Configure Button

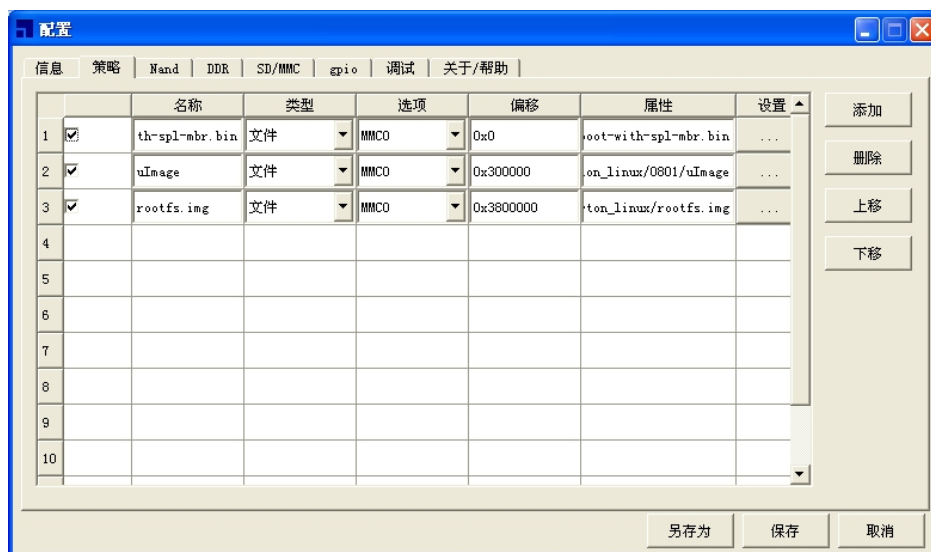


Figure 4-7 Configure Menu

## 4.7 Burn binary files

u-boot-with-spl-mbr.bin, ulmage, rootfs.img/rootfs.ext2 should be flashed into Newton.

Another document “How to Burn Newton Demo” provides help for you on how to install the burning tool driver and how to burn Newton.

## 4.8 Boot Newton board

After being flashed successfully, Newton will reboot automaticly and Serial Port(Baud rate: 57600, Data Bits:8, Parity: None, Stop Bits: 1, RTS/DTR Control: NO) will show following message:

```

sdram init ok
MMC init ok
Starting U-Boot ...

U-Boot 1.1.6-g5352e480 (Apr 23 2014 - 12:02:44)

Board: Ingenic NEWTON (4775 SOC CPU Speed 1008 MHz)
MEM Clock: 168 MHz
DRAM: 512 MB Ram size > EMC_LOW_SDRAM_SPACE_SIZE, set ram size = EMC_LOW_SDRAM_SPACE_SIZE: 256 MB
Error: Unknown flash ID, force set to 'SST_ID_39SF040'
Flash: 512 kB
MMC init ok
*** Warning - MMC/SD first load, using default environment

--==--== 0x8fe88000 --==--
d2041 set_lcd_power_on
Line is 762
DEFAULT_BACKLIGHT_LEVELIn: serial
Out: lcd
Err: lcd
Net: JZ ETHERNET
Hit any key to stop autoboot: 0
MMC init ok
6291456 bytes : OK
## Booting image at 80600000 ...
Image Name: Linux-3.0.8-00132-gde62c9c
Image Type: MIPS Linux Kernel Image (gzip compressed)
Data Size: 2580324 Bytes = 2.5 MB
Load Address: 80010000
Entry Point: 80425700
Verifying Checksum ... OK
Uncompressing Kernel Image ... OK

Starting kernel ...

```

**Figure 4-8 Boot Message**

If serial port shows as below, kernel is OK:

```

[ 4.943623] x2d x2d: Virtual Driver of JZ X2D registered
[ 4.954509] Virtual Driver of JZ X2D registered
[ 4.964442] regulator_init_complete: LDO_AUD: incomplete constraints, leaving on
[ 4.985821] jz-rtc jz-rtc.0: setting system clock to 2013-03-01 07:59:44 UTC (1362124784)
[ 5.007577] EXT4-fs (mmcblk0p1): couldn't mount as ext3 due to feature incompatibilities
[ 5.043737] EXT4-fs (mmcblk0p1): couldn't mount as ext2 due to feature incompatibilities
[ 5.080652] EXT4-fs (mmcblk0p1): mounted filesystem without journal. Opts: (null)
[ 5.096083] VFS: Mounted root (ext4 filesystem) on device 179:1.
[ 5.109367] Freeing unused kernel memory: 212k freed
Starting udevd ...

```

**Figure 4-9 Boot Message**

If serial port shows as below, mount root file system is OK. Enter 'root' for user name:

```

[ 7.166393] dhdsdio_write_vars: Download, Upload and compare of NVRAM succeeded.
[ 7.340273] dhd_bus_init: enable 0x06, ready 0x06 (waited 0us)
[ 7.403438] Firmware up: op_mode=0x0015, Broadcom Dongle Host Driver mac=44:39:c4:45:32:c6
[ 7.643473] Firmware version = wl0: Dec 26 2013 20:38:09 version 6.10.190.43 (r442147) FWID 01-32458bd3
[ 7.664763] Broadcom Dongle Host Driver: register interface [wlan0] MAC: 44:39:c4:45:32:c6
[ 7.685505] dhdsdio_probe : the lock is released.
[ 7.843555] wl_host_event: Invalid ifidx 0 for wl0
[ 7.853439] wl_android_wifi_on in
[ 7.883512] CFG80211-ERROR) wl_cfg80211_attach_post : p2p0: p2p_dev_addr=46:39:c4:45:32:c6
udhcpd (v1.21.0) started
[root@Ingenic /]#

```

## 4.9 Readmes

Under SDK\_ROOT/documents directory, there are some help documentations for Newto, read them to

---

find more details.

README\_ WIFI:               /\* For WIFI device configuration and testing \*/

README\_ BLUETOOTH:       /\* For Bluetooth device configuration and testing \*/

README\_ SENSOR:           /\* For Sensor devices configuration and testing \*/

README\_ MISC:             /\* For other devices's configuration and testing \*/