

V40 项目

SDK Quick Start Guide V1.0

文档履历

[illegible]

目 录

V40 项目	1
SDK Quick Start Guide V1.0	1
目 录	3
1. SDK 目录结构	4
2. 编译环境搭建指南	5
3. 编译指南	6
3.1. 编译内核.....	6
3.1.1. 编译方式 1.....	6
3.1.2. 编译方式 2.....	6
3.1.3. 编译方式 3.....	7
3.2. 编译 brandy(本节可选).....	7
3.2.1. 编译 u-boot	8
3.3. 编译 Android	8
4. 打包指南	9
4.1. Android 固件打包	9
4.2. Dragonboard/Linux 固件打包	9
4.2.1. 方法一	9
4.2.2. 方法二（建议使用）	9
5. 烧写指南	11
5.1. usb 方式.....	11
5.2. 卡量产方式.....	11
6. Declaration.....	12

1. SDK 目录结构

— android	
— abi	
— art	
— bionic	
— bootable	
— build	
— cts	
— dalvik	
— developers	
— development	
— device	
— softwinner	
— magton-common	平台公共目录
— magton-pl	方案定制目录
— docs	
— external	
— frameworks	
— hardware	
— libcore	
— libnativehelper	
— Makefile	
— ndk	
— packages	
— pdk	
— prebuilts	
— sdk	
— system	
— tools	
— lichee	
— brandy	boot 相关源码
— buildroot	编译脚本、交叉编译工具链
— build.sh	Top level 编译脚本
— linux-3.4	3.4 内核目录, v40 不使用
— linux-3.10	3.10 内核目录, v40 使用
— out	
— README	
— tools	打包脚本、工具和方案配置

2. 编译环境搭建指南

请查看 Android 网站 <http://source.android.com/> 说明。

3. 编译指南

3.1. 编译内核

3.1.1. 编译方式 1

/build.sh lunch 命令（建议使用）

```

su@kali:~$ cd /mnt/sda1/Android22.1/Workspace/Dev/product dev/lichee$ ./build.sh lunch
All available lichee lunch:
  0. sun50iw1p1-android
  1. sun50iw2p1-android
  2. sun8iw11p1-android
  3. sun8iw6p1-android-eagle
  4. sun8iw6p1-android-secure
  5. sun8iw6p1-android-aston
  6. sun8iw7p1-android-dolphin
  7. sun8iw7p1-android-secure
  8. sun8iw7p1-android-karaok
  9. sun8iw8p1-android
 10. sun9iw1p1-android-jaws
 11. sun9iw1p1-android-secure
 12. sun9iw1p1-android-optimus
Choice: 2

```

sun8iw11p1: V40芯片平台代号
android : 平台系统内核配置

3.1.2. 编译方式 2

./build.sh config 命令

```

su@kali:~$ cd /mnt/sda1/Android22.1/Workspace/Dev/product dev/lichee$ ./build.sh config
Welcome to mkscript setup progress
All available chips:
  0. sun50iw1p1
  1. sun50iw2p1
  2. sun8iw11p1
  3. sun8iw6p1
  4. sun8iw7p1
  5. sun8iw8p1
  6. sun9iw1p1
Choice: 2
All available platforms:
  0. android
  1. dragonboard
  2. linux
  3. eyeseeelinux
Choice: 0

```

sun8iw11p1: v40 平台芯片代号
android : 平台系统内核配置

提示：新增业务选择项，前提是配置对应选择的芯片平台，mkbusiness 文件

V40: 对应 sun8iw11p1

3.1.3. 编译方式 3

`./build.sh -p [chip]_[platform] -k [kernel version]`

```

v/lichee$ ./build.sh -p sun8iw11p1_android -k linux-3.10
not set business, to use default!
not set arch, to use default by kernel version
INFO: -----
INFO: build lichee ...
INFO: chip: sun8iw11p1
INFO: platform: android
INFO: business:
INFO: kernel: linux-3.10
INFO: arch: arm
INFO: board:
INFO: output: out/sun8iw11p1/android/
INFO: -----
INFO: build buildroot ...
external toolchain has been installed

```

注意:

- (1). 编译方式 1、2、3 每次编译都会重新拷贝内核配置到 `linux-3.10` 目录下,并命名为`.config`。
- (2). 如果采用过上述方式 1,2,3 进行编译,在下一一次编译的就可以使用命令: `./build.sh`。该命令不会重新拷贝一份内核配置到 `linux-3.10` 目录下。如下图所示命令的使用:

```

/wx3800/don/product dev,lichee$ ./build.sh
INFO: -----
INFO: build lichee ...
INFO: chip: sun8iw11p1
INFO: platform: android
INFO: business:
INFO: kernel: linux-3.10
INFO: arch: arm
INFO: board:
INFO: output: out/sun8iw11p1/android/
INFO: -----
INFO: build buildroot ...
external toolchain has been installed
external toolchain 32 has been installed
INFO: build buildroot OK.
INFO: build kernel ...
INFO: prepare toolchain ...
use last time build config
Building kernel

```

3.2. 编译 brandy(本节可选)

brandy 目录中存放的是 V40 平台的 bootloader, 该目录为 V40 启动代码, 默认不编译。

3.2.1. 编译 u-boot

方法一：brandy 目录下，可以快速完成 uboot 编译动作。

```
cd lichee/brandy/  
./build.sh -p sun8iw11p1
```

方法二：最常用。

```
cd lichee/brandy/u-boot-2014.07  
make distclean  
make sun8iw11p1_config  
make -j14
```

当编译成功，生成的 u-boot-sun8iw11p1.bin 文件会自动拷贝到对应的 tools 目录下，这时候可以直接打包或者其它操作。u-boot-sun8iw11p1.bin 是启动时 uboot 核心可执行程序。

3.3. 编译 Android

编译 android 前，请先确保内核编译完成过。

```
$ cd ~/workspace/carlet44/android  
$ source build/envsetup.sh  
$ lunch                               //选择 magton_p1-eng  
$ extract-bsp                         //拷贝 kenrel 和 modules  
$ make -j8                           //-j8 表示启用 8 个进程编译，一般情况下进程个数不用超过 cpu 核数。
```


4. 打包指南

4.1. Android 固件打包

cd 到 android 根目录

1. 打包 release 固件:

```
$ pack
```

2. 打包 debug 固件:

```
$ pack -d
```

4.2. Dragonboard/Linux 固件打包

4.2.1. 方法一

1. 配置（开启新的终端必须执行一次）

```
$ source buildroot/scripts/mksetup.sh
```

2. 打包

```
$ mkpack [-d [card0|uart0]] [-m [normal|dump]] [-f [android|prvt]]
```

参数:

- d card0|uart0, card0 表示 debug 固件, uart0 表示 release 固件（默认）
- m normal|dump, normal 表示普通固件（默认）, dump 表示读取机器分区数据固件
- f android|prvt, android 表示普通固件（默认）, prvt 表示烧写 private 分区的固件

4.2.2. 方法二（建议使用）

1. 配置（如果已经配置可以省略这个步骤）

```
$ ./build.sh config (选择编译 dragonboard 或 linux 固件)
```

2. 打包

- (1). 打包 release 固件

```
$ ./build.sh pack
```

- (2). 打包 debug 固件

```
$ ./build.sh pack_debug
```

- (3). 打包读取机器分区数据固件

```
$ ./build.sh pack_dump
```

(4). 打包烧写 private 分区的固件

```
$ ./build.sh pack_prvt
```

5. 烧写指南

5.1. usb 方式

电脑端可用于烧写的软件有：

- PhoenixUSBPro : USB 量产工具。
- PhoenixSuit : USB 升级工具。

这两个工具都位于 `lichee\tools\tools_win\USB_update_and_produce` 目录下，使用方法请参考软件安装完成后的使用说明文档。

5.2. 卡量产方式

卡量产即通过 SD 卡进行刷机，需要使用 PhoenixCard 制作量产卡，软件位于 `lichee\tools\tools_win\` 目录下，使用方法请参考软件安装完成后的使用说明文档。

6. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.