

OK6410, Emdebian Linux 日志

Copyright (C) 2015 Zhou Mo

LICENSE : MIT

May 12, 2015

Contents

1	Installing base Emdebian/Debian	2
1.1	Prepare the rootfs	2
1.1.1	debootstrap	2
1.1.2	squash rootfs into yaffs2	2
1.2	Prepare the SD Card	2
1.2.1	Format SD card as FAT32	2
1.2.2	Program SD card and copy files	2
1.3	Switch jumper: boot from SD card	3
1.4	Burn files from SD to NAND	3
1.5	Switch jumper: boot from NAND	3
1.6	Finalize system on mtblock2	3
1.6.1	set u-boot arg	3
1.6.2	stage2	3
1.6.3	finalize	4
1.7	Make system on NAND:mtblock3	4
1.7.1	send ok6410.tar.gz to board	4
1.7.2	prepare mtblock3	4
1.7.3	untar	4
1.7.4	reboot, tweak bootargs	4
1.7.5	do stage2 as above	4
1.7.6	do finalize as above	4
1.7.7	tweak bootargs	4
1.8	Smoke test!	4
2	Beyond base system	4
2.1	installing more software	5
2.1.1	tasksel	5
2.1.2	dpkg,apt	5
2.2	Calibrate the Touchscreen	5
2.2.1	Set env	5
2.2.2	Calibrate	5
2.3	Punch the face of X	5
2.3.1	Sample script	5
2.3.2	Note	5
2.4	Kick the ass of ALSA	5
2.4.1	the right way	6

3	Maintain & Troubleshoot & Hint	6
3.1	Merge backup rootfs into restored FS	6
3.1.1	Method 1: rsync + ssh	6
3.1.2	Method 2: tar + nc	6
3.2	Dirty hint: Stage1 rootfs to Stage1.5	6
3.3	Hint: bootargs	7
3.4	Manipulate broken ext3 FS	7
3.4.1	Boot another OS	7
3.4.2	Check filesystem	7
3.4.3	Change bootargs and boot	8
3.5	Hint: Virtualbox USB Driver	8
3.6	Unable to load mmc:1 *fatload*	8
3.7	Trsh: sshd: Connection Timeout.	8
3.7.1	Syndrome	8
3.7.2	Solution	8
4	Porting Kernel(failed)	8
A	Reference	9
B	Hardware	9
C	ch341.ko	9
D	Changelog	9
D.1	3 Mar 2015 - 11 Mar 2015	9
D.2	19 Mar 2015	9
D.3	20 Mar 2015	9

1 Installing base Emdebian/Debian

Note: no porting linux kernel in this section.

Note: the procedure is according to user manual and some web resource.

1.1 Prepare the rootfs

1.1.1 debootstrap

首先在PC上完成Base system 的 bootstrap, 生成 Stage 1 系统。

```
# sudo debootstrap --arch=armel --foreign --no-check-gpg \  
  --include=vim,openssh-server squeeze rootfs/ \  
  http://ftp.uk.debian.org/emdebian/grip/
```

此处Debootstrap参数根据需要自行调整。其中include是指额外安装到Base system中的程序, 推荐vim,openssh-server,busybox.

完成之后可以对此基本系统进行初步设置。

```
# echo "proc /proc proc none 0 0" >> rootfs/etc/fstab  
# echo "OK6410" > rootfs/etc/hostname  
# sudo mkdir -p rootfs/usr/share/man/man1/  
# sudo mknod rootfs/dev/console c 5 1  
# sudo tar zcvf ../ok6410.tar.gz .
```

妥善保管ok6410.tar.gz这个stage1归档。

1.1.2 squash rootfs into yaffs2

由于飞凌提供的手册上使用了yaffs2文件系统, 所以此处跟随使用该文件系统。

本章将把制作好的stage1系统包装成可烧写到NANDFlash上的yaffs2文件, 然后进行烧写。

find the yaffs2 making tool¹, then

```
# mkyaffs2image-nand2g rootfs/ rootfs.yaffs2
```

注意, 资料中编译好的对应二进制程序应用i386体系结构, 用户可以使用 Arch Linux或者Gentoo Linux的iso在i386虚拟机中运行该程序。multi-arch亦可。qemu亦可。

1.2 Prepare the SD Card

准备一张干活的SD卡, 用它来烧写rootfs (此处仅烧写rootfs)。

1.2.1 Format SD card as FAT32

```
# parted ...  
# mkfs.vfat ...
```

1.2.2 Program SD card and copy files

1. Program SD card with SD_Writer.exe
Refer the Appendix:reference :: 1 please.
2. Copy u-boot.bin into SD card

¹./Linux-3.0.1/filesystem/Yaffs2文件系统制作工具/mkyaffs2image-nand2g

3. Copy zImage into SD card
use the one included in demo/²
4. Copy rootfs.yaffs2 into SD card
use the one you created just now.

1.3 Switch jumper: boot from SD card

switch jumper 6 and 7 off:

```
Num      : 1234 5678
ShouldBe: 0001 1111
Original: 0001 1001
```

1.4 Burn files from SD to NAND

these are u-boot automatic processes.³

1. *Burn u-boot
2. *Burn kernel
3. Burn system

This takes a long time.

1.5 Switch jumper: boot from NAND

switch jumper 6 and 7 on.

1.6 Finalize system on mtddblock2

1.6.1 set u-boot arg

首先设置内核参数。

```
# setenv bootargs \root=/dev/mtddblock2 noinitrd rootfstype=yaffs2 \
  consle=/dev/ttySAC0,115200 init=/bin/sh"
# saveenv
# reset
```

1.6.2 stage2

使用debootstrap来进行stage2.

```
# /debootstrap/debootstrap --second-stage
```

This also takes a very long time.

²zImage of linux-3.0.1 heavily customized

³items headed with '*' is optional

1.6.3 finalize

最后的一些设置，确定系统重启之后可以正常工作。

```
# echo "ttySAC0" >> /etc/securetty
# printf "T0:123:respawn:/sbin/getty 115200 ttySAC0\n" >> /etc/inittab
# printf "auto eth0\nifconfig eth0 inet dhcp\n" >> /etc/network/interfaces
(this step is optional)
# passwd
# reboot -f
```

Then tweak bootargs with u-boot

```
# setenv bootargs \root=/dev/mtdblock2 noinitrd rootfstype=yaffs2 \
    consle=/dev/ttySAC0,115200 init=/sbin/init"
# saveenv
# reset
```

Then the system on mtblock2 is in stage3 now, and it works.

1.7 Make system on NAND:mtblock3

1.7.1 send ok6410.tar.gz to board

1. nc; 2.ssh; 3. rsync;

1.7.2 prepare mtblock3

```
ok# mkfs.ext3 /dev/mtblock3
```

1.7.3 untar

```
ok# mount /dev/mtblock3 /mnt ok# tar zxvf ok.tar.gz -C /mnt
```

1.7.4 reboot, tweak bootargs

```
root=/dev/mtblock3 noinitrd rootfstype=ext3 consle=/dev/ttySAC0,115200 init=/bin/sh
```

1.7.5 do stage2 as above

1.7.6 do finalize as above

1.7.7 tweak bootargs

```
root=/dev/mtblock3 noinitrd rootfstype=ext3 consle=/dev/ttySAC0,115200 init=/sbin/init
```

1.8 Smoke test!

Boot, then see if the machine works.

2 Beyond base system

This section is focused on applications beyond the base system.

2.1 installing more software

2.1.1 tasksel

install touch screen programs easily, this is optional.

2.1.2 dpkg,apt

view the man page.

2.2 Calibrate the Touchscreen

2.2.1 Set env

```
# export TSLIB_TSDEVICE=/dev/input/event2
```

可以参考飞凌的手册。

2.2.2 Calibrate

please start the X, then do the calibration with

```
# ts_calibrate
```

if you want to run a test, type

```
# ts_test
```

2.3 Punch the face of X

2.3.1 Sample script

```
#!/bin/bash
#set -e
```

```
X :1 -noreset -quiet&
export DISPLAY=:1
echo "X start success."
```

2.3.2 Note

X 与 Xorg 有区别。

Xorg 似乎更加底层，但区别到底在哪呢。

2.4 Kick the ass of ALSA

On wiki.archlinux.org, page "ALSA", there are lots of solutions to make alsa work. However, nothing works, totally.

2.4.1 the right way

The demo rootfs included in Material has a working alsa instance installed. So the right way is to copy the working alsa instance into the Endebian.

Demo's init program is /linuxrc, which is a symbol link to "/bin/busybox". Inspected /etc/{init.d,rc.d} matter, I copied some files to emdebian.

1. /etc/wm9714.conf⁴
2. /usr/alsa/*
3. ?/alsaconf (shell script)

Had those files copied to the board, you invoke

```
# alsaconf start
```

Then alsa works.⁵

3 Maintain & Troubleshoot & Hint

3.1 Merge backup rootfs into restored FS

3.1.1 Method 1: rsync + ssh

using "rsync". From the PC side:

```
# rsync -av --ignore-existing the/backup/dir root@IP_OF_BOARD:/
```

note the '--ignore-existing' option of rsync.

3.1.2 Method 2: tar + nc

firstly the board side:

```
# nc -l -p PORT_BOARD | tar xvf - -C /
```

then the PC side:

```
# cd /rootfs_ok/  
# tar cvf - . | nc IP_BOARD PORT_BOARD
```

and please note the '-k' '-z' option of tar.

3.2 Dirty hint: Stage1 rootfs to Stage1.5

If you can't wait the slow process that dpkg bootstrapping the base system, you can directly extrace those files in software packages into your rootfs, by invoking dpkg -X.

e.g.

```
# cd /var/cache/apt/archive/  
# for DEB in $(ls *.deb); do dpkg -X ${DEB} / ; done
```

Then the system is nearly the same as stage1, but contains more commands for you to invoke. Maybe the bootstrapping procedure could get painless in this way.

⁴the alsa configuration file for the board

⁵but still don't know why it works.

3.3 Hint: bootargs

此处可能出现低级错误。在uboot下printenv可以看见类似这样的一个bootargs:

```
bootargs=root=/dev/mtdblock2 rootfstype=yaffs2 init=/bin/sh ...
```

然而可能由各种原因导致写入新值时写成如下形式:

```
bootargs=/dev/mtdblock2 rootfstype=yaffs2 init=/bin/sh ...
```

此时容易漏掉“root=”几个字符, 导致内核无法加载rootfs从而kernel panic。如果犯了这种错误, 内核会尝试从NFS中加载rootfs, 此特征值得注意。

3.4 Manipulate broken ext3 FS

使用备用操作系统的e2fsck来对mtdblock3这个设备上的文件系统执行离线检查。

3.4.1 Boot another OS

In u-boot command line, change the bootargs variable

```
# echo $bootargs
root=/dev/mtdblock3 noinitrd rootfstype=ext3 init=/sbin/init \
    lcdsize=43 console=ttySAC0,115200
```

then modify it

```
# setenv bootargs "root=/dev/mtdblock2 noinitrd rootfstype=yaffs2 \
    init=/sbin/init lcdsize=43"
# saveenv
# reset
```

, then we can boot the system on mtdblock2

3.4.2 Check filesystem

don't mount /dev/mtdblock3, and just type

```
# fsck.ext3 /dev/mtdblock3 -y
if success: goto next subsection; else
# fsck.ext3 -f /dev/mtdblock3 -y
if success: goto next subsection; else
# mkfs.ext3 -n /dev/mtdblock3
```

then find the place of backup superblock, e.g. "32768". Next, use the backup superblock

```
( change the following 32768 to the number you exactly see
# fsck.ext3 -b 32768 /dev/mtdblock3 -f -y
```

if success: goto next subsection; else

```
# mkfs.ext3 -S /dev/mtdblock3
( rightly after that, type next command
# fsck.ext3 -f /dev/mtdblock3 -y
```

this method must success.

3.4.3 Change bootargs and boot

3.5 Hint: Virtualbox USB Driver

<http://download.virtualbox.org/virtualbox>

下载Extension_Pack并安装，添加当前用户到vboxusers中即可。

<http://download.virtualbox.org/virtualbox>

3.6 Unable to load mmc:1 *fatload*

换SD卡。

3.7 Trsh: sshd: Connection Timeout.

3.7.1 Syndrome

1. sshd service is running, other machines have ssh access to it. However even if the password is definitely correct, ssh would say "Timeout, server 1.1.1.2 not responding."
2. Even from the machine running sshd, ssh can not login itself. It says "ssh: connect to host localhost port 22: Network is unreachable"

3.7.2 Solution

Seems that the loopback network device "lo" is DOWN. And then turn on the device "lo" indeed solves this issue.

1. Edit /etc/network/interfaces (Debian)

```
+ auto lo
+ iface lo inet loopback
```

2. Start service "networking"
3. Turn on the device manually.

```
# ip link set dev lo up
```

Now the sshd works.

4 Porting Kernel(failed)

This plan is aborted. (20, Mar, 2015)

<https://code.google.com/p/linux-ok6410>

A Reference

1. OK6410-A开发板LINUX3.0.1-2014-09用户手册.pdf
2. http://www.arm9board.net/wiki/index.php?title=Debian_on_OK6410
3. Archwiki: ALSA
4. Archwiki: ALSA/Troubleshoot
5. virtualbox.org
6. Man pages

B Hardware

CPU: Samsung S3C6410, armv7
RAM: 256 MB
NANDFlash: 1GiB
Board Machine ID: 1626

C ch341.ko

D Changelog

D.1 3 Mar 2015 – 11 Mar 2015

安装Emdebian，记录简单日志。

D.2 19 Mar 2015

调整文档结构，添加Changelog，添加后期发现的点子和一些经验。

D.3 20 Mar 2015

板子调整和测试完成。完善文档。