

Android 内核的简单分析

Android 内核的简单分析

NeilWong (neilwong@hotmail.com)

Google 的 Android 操作系统已经成功面市,因为 Android 操作系统是基于 Linux-2.6.25 内核的,并且目前所有的源代码也都全部开放,因此可以简单地加以分析以方便以后向其他平台的移植工作.

首先需要了解的就是 Android 的一些基本情况,并下载 Android 的源代码以方便分析.

Android 的官方站点:

<http://www.android.com/>

Android Source 的官方安装方法:

<http://source.android.com/download>

Android 的中文站点:

<http://www.androidin.com/>

源代码包的相应联结地址:

<http://www.androidin.com/bbs/viewthread.php?tid=2337>

我们可以采用 Android 官方站点推荐方式来下载 Android 的源代码,也可以从 Android 中文站点直接下载源代码程序包,随后再更新和同步.

当采用 Android 官方推荐方式下载或者需要更新你的源代码时,都需要 git 工具支持, 当你的系统没有安装 git 或者 git 版本过低时,建议从下面的地址更新到最新的 git 包.(Android 要求 git 版本必须大于 1.5.4)

<http://git.or.cz/>

给 git 自动升级的方式:

```
git clone git://git.kernel.org/pub/scm/git/git.git
```

如果采用下载源代码包的方式,下载并展开后,在 .repo/repo 目录下有同步执行的文件 repo, 你可以将其拷贝到你的 PATH 所在路径以备使用,例如/usr/bin 等等. 或者也可以从 Android 站点得到最新的 repo 文件并复制到 PATH 所在路径.

```
wget http://android.git.kernel.org/repo
```

源代码包解开后,其主目录下存在 repo_sync.sh 文件, 其主要是执行 repo sync 以实现和 Android 站点同步代码包的目的, 当执行出错后将再一次运行 repo sync 命令直到同步正常结束,但原文件包含错误,建议改成下面的程序:(原文件中少了一个空格)

```
#!/bin/bash
echo "=====start repo sync===== "
repo sync
while [ $? = 1 ]; do
echo "=====sync failed, re-sync again ===== "
sleep 3
repo sync
done
```

这样, 运行 `sh repo_sync.sh` 就可以自动实现本地代码包的同步和更新动作了.

我在 Fedora Core 6 上经过同步之后, `du -h ./mydroid`, 占用空间大约是 2.1G

以下为 Android 官方公布的架构图:

在 `kernel` 子目录下存放的就是 Android 的 Linux Kernel 了, 通过和标准的 Linux 2.6.25 Kernel 的对比, 我们可以发现, 其主要增加了以下的内容:

1. 基于 ARM 架构增加 Gold-Fish 平台,相应增加的目录如下:

kernel/arch/arm/mach-goldfish

kernel/include/asm-arm/arch-goldfish

目前 Google 的基于 Gold-Fish 的手机为 G1 手机, 明年 G2 手机将面世。

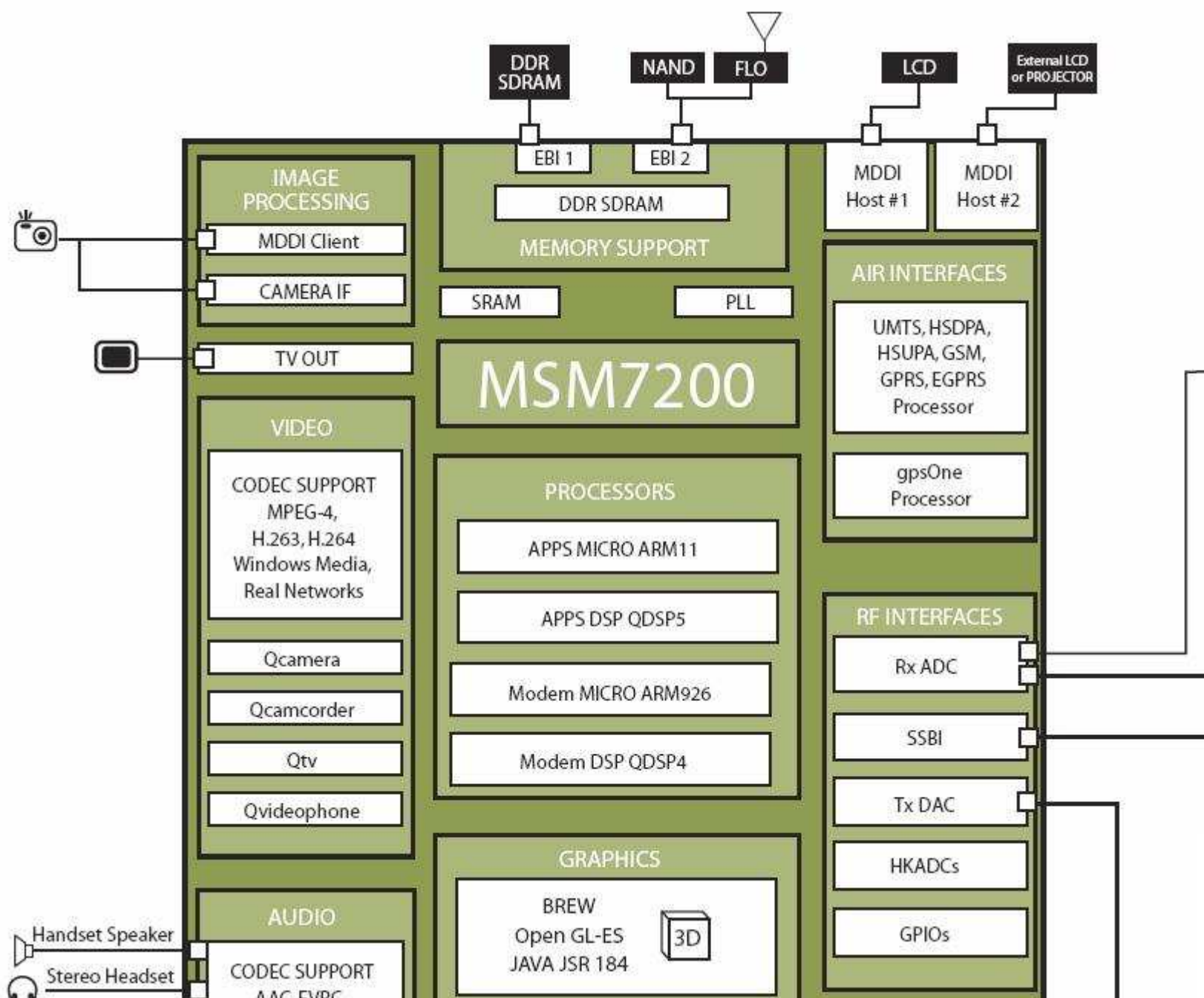
该手机采用 Qualcomm 公司的高性能处理芯片 MSM7201A, 该芯片以 ARM 11 作为 Application 应用处理器, 以 ARM926T 作为 BaseBand 主处理器(主要承载 GSM / GPRS / EDGE / 3G 等协议栈处理), 支持 JAVA 硬加速(ARM 本身也自带 JAVA 硬件处理), 包含 GPS Processor, 支持 2D/3D 图形加速(每秒可处理 4 百万个多边形), 另外还支持最大 800 万像素的 Sensor, 支持 MPEG 4/H.263/H.264/Real Media 等多种 Codec。

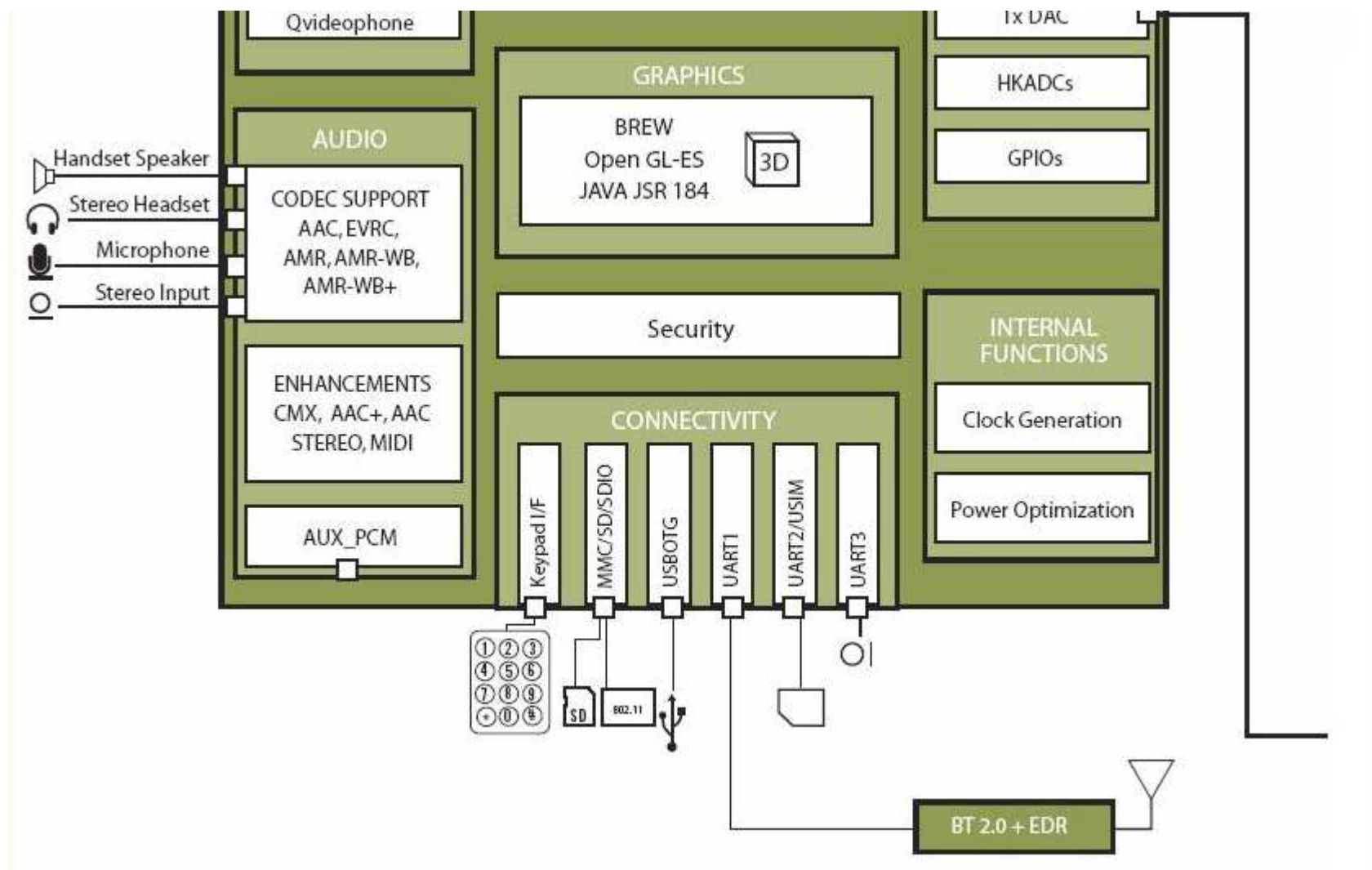
其中 ARM926T 采用的主频为 274M HZ, ARM11 的主频为 528MHZ. (MSM7200 的 ARM11 的主频为 400M HZ)

因为没有找到 MSM7201A 的详细 DataSheet, 因此可以简单地参考 MSM7200 的规格:

[MSM7200_DataSheet.pdf](#)

下面为 MSM7200 架构图:





2. 增加了 yaffs2 FLASH 文件系统,相应增加的目录为:

kernel/fs/yaffs2

实际上, Android 包经过编译后生成的 system.img 和 ramdisk.img 文件就是 yaffs2 格式的包.

3. 增加了 Android 的相关 Driver,相应目录为:

kernel/drivers/android

主要分为:

Android IPC 系统: Binder (binder.c)

Android 日志系统: Logger (logger.c)

Android 电源管理: Power (power.c)

Android 闹钟管理: Alarm (alarm.c)

Android 内存控制台: Ram_console (ram_console.c)

Android 时钟控制的 gpio: Timed_gpio (timed_gpio.c)

4. 增加了 switch 处理, 相应的目录为:

`kernel/drivers/switch/`

5. 增加了一种新的共享内存处理方式, 相应增加的文件为:

`kernel/mm/ashmem.c`

6. 其他为 Linux-2.6.25 内核所做的补丁等等, 例如 BlueTooth, 在此不做详细分析

另外 GoldFish 平台相关的驱动文件如下:

1. 字符输出设备:

`kernel/drivers/char/goldfish_tty.c`

2. 图象显示设备: (Frame Buffer)

kernel/drivers/video/goldfishfb.c

3. 键盘输入设备:

kernel/drivers/input/keyboard/goldfish_events.c

4. RTC 设备: (Real Time Clock)

kernel/drivers/rtc/rtc-goldfish.c

5. USB Device 设备:

kernel/drivers/usb/gadget/android_adb.c

6. SD 卡设备:

kernel/drivers/mmc/host/goldfish.c

7. FLASH 设备:

kernel/drivers/mtd/devices/goldfish_nand.c

kernel/drivers/mtd/devices/goldfish_nand_reg.h

8. LED 设备:

kernel/drivers/leds/ledtrig-sleep.c

9. 电源设备:

kernel/drivers/power/goldfish_battery.c

10. 音频设备:

kernel/arch/arm/mach-goldfish/audio.c

11. 电源管理:

kernel/arch/arm/mach-goldfish/pm.c

12. 时钟管理:

kernel/arch/arm/mach-goldfish/timer.c

下面的链接为 **Android Kernel** 和标准 **Linux-2.6.25** 包相比较所产生的 **PATCH** 包, 其中比较小的 **other patch** 包是除了上面列出的增加文件之外所修改的文件细节, 供大家参考。

<http://blogimg.chinaunix.net/blog/upfile2/090102095410.gz>

以下为 Android 内核启动信息:

Uncompressing Linux..... done, booting the kernel.

emulator: emulator window was out of view and was recentred

Linux version 2.6.25-00350-g40fff9a (android-build@apa27.mtv.corp.google.com) (gcc version 4.2.1) #1 Wed Jul 23 18:10:44 PDT 2008

CPU: ARM926EJ-S [41069265] revision 5 (ARMv5TEJ), cr=00003137

Machine: Goldfish

Memory policy: ECC disabled, Data cache writeback

CPU0: D VIVT write-through cache

CPU0: I cache: 4096 bytes, associativity 4, 32 byte lines, 32 sets

CPU0: D cache: 65536 bytes, associativity 4, 32 byte lines, 512 sets

Built 1 zonelists in Zone order, mobility grouping on. Total pages: 24384

Kernel command line: qemu=1 console=ttyS0 android.checkjni=1 android.qemud=ttyS1 android.ndns=2

Unknown boot option `android.checkjni=1': ignoring

Unknown boot option `android.qemud=ttyS1': ignoring

Unknown boot option `android.ndns=2': ignoring

PID hash table entries: 512 (order: 9, 2048 bytes)

Console: colour dummy device 80x30

Dentry cache hash table entries: 16384 (order: 4, 65536 bytes)

Inode-cache hash table entries: 8192 (order: 3, 32768 bytes)

Memory: 96MB = 96MB total

Memory: 94268KB available (2380K code, 445K data, 100K init)

Mount-cache hash table entries: 512

CPU: Testing write buffer coherency: ok
net_namespace: 152 bytes
android_power_init
android_power_init done
NET: Registered protocol family 16
NET: Registered protocol family 2
IP route cache hash table entries: 1024 (order: 0, 4096 bytes)
TCP established hash table entries: 4096 (order: 3, 32768 bytes)
TCP bind hash table entries: 4096 (order: 2, 16384 bytes)
TCP: Hash tables configured (established 4096 bind 4096)
TCP reno registered
checking if image is initramfs... it is
Freeing initrd memory: 136K
goldfish_new_pdev goldfish_interrupt_controller at ff000000 irq -1
goldfish_new_pdev goldfish_device_bus at ff001000 irq 1
goldfish_new_pdev goldfish_timer at ff003000 irq 3
goldfish_new_pdev goldfish_rtc at ff010000 irq 10
goldfish_new_pdev goldfish_tty at ff002000 irq 4
goldfish_new_pdev goldfish_tty at ff011000 irq 11
goldfish_new_pdev smc91x at ff012000 irq 12
goldfish_new_pdev goldfish_fb at ff013000 irq 13
goldfish_new_pdev goldfish_audio at ff004000 irq 14
goldfish_new_pdev goldfish_memlog at ff006000 irq -1
goldfish_new_pdev goldfish-battery at ff014000 irq 15
goldfish_new_pdev goldfish_events at ff015000 irq 16
goldfish_new_pdev goldfish_nand at ff016000 irq -1

goldfish_new_pdev goldfish-switch at ff017000 irq 17
goldfish_new_pdev goldfish-switch at ff018000 irq 18
goldfish_pdev_worker registered goldfish-switch
goldfish_pdev_worker registered goldfish-switch
goldfish_pdev_worker registered goldfish_nand
goldfish_pdev_worker registered goldfish_events
goldfish_pdev_worker registered goldfish-battery
goldfish_pdev_worker registered goldfish_memlog
goldfish_audio_probe
goldfish_pdev_worker registered goldfish_audio
goldfish_pdev_worker registered goldfish_fb
goldfish_pdev_worker registered smc91x
goldfish_pdev_worker registered goldfish_tty
goldfish_pdev_worker registered goldfish_tty
goldfish_pdev_worker registered goldfish_rtc
goldfish_pdev_worker registered goldfish_timer
goldfish_pdev_worker registered goldfish_device_bus
goldfish_pdev_worker registered goldfish_interrupt_controller
ashmem: initialized
Installing knfsd (copyright (C) 1996 okir@monad.swb.de).
yaffs Jul 23 2008 18:10:35 Installing.
io scheduler noop registered
io scheduler anticipatory registered (default)
io scheduler deadline registered
io scheduler cfq registered
allocating frame buffer 320 * 480, got ffc10000

Console: switching to colour frame buffer device 40x30
console [ttyS0] enabled
brd: module loaded
loop: module loaded
nbd: registered device at major 43
smc91x.c: v1.1, sep 22 2004 by Nicolas Pitre <nico@cam.org>
No IRQF_TRIGGER set_type function for IRQ 12 (goldfish)
eth0: SMC91C11xFD (rev 1) at c6800000 IRQ 12 [nowait]
eth0: Ethernet addr: 52:54:00:12:34:56
goldfish nand dev0: size 4000000, page 2048, extra 64, erase 131072
goldfish nand dev1: size 4000000, page 2048, extra 64, erase 131072
goldfish nand dev2: size 4000000, page 2048, extra 64, erase 131072
mice: PS/2 mouse device common for all mice
*** events probe ***
events_probe() addr=0xc6804000 irq=16
events_probe() keymap=qwerty2
input: qwerty2 as /class/input/input0
goldfish_rtc goldfish_rtc: rtc core: registered goldfish_rtc as rtc0
logger: created 64K log 'log_main'
logger: created 64K log 'log_events'
logger: created 64K log 'log_radio'
IPv4 over IPv4 tunneling driver
GRE over IPv4 tunneling driver
TCP cubic registered
NET: Registered protocol family 1
NET: Registered protocol family 17

```
RPC: Registered udp transport module.
RPC: Registered tcp transport module.
802.1Q VLAN Support v1.8 Ben Greear <greearb@candelatech.com>
All bugs added by David S. Miller <davem@redhat.com>
VFP support v0.3: implementor 41 architecture 1 part 10 variant 9 rev 0
goldfish_rtc goldfish_rtc: setting system clock to 2009-01-02 07:43:01 UTC (1230882181)
Freeing init memory: 100K
init: cannot open '/initlogo.rle'
yaffs: dev is 32505856 name is "mtdblock0"
yaffs: passed flags ""
yaffs: Attempting MTD mount on 31.0, "mtdblock0"
yaffs: dev is 32505857 name is "mtdblock1"
yaffs: passed flags ""
yaffs: Attempting MTD mount on 31.1, "mtdblock1"
yaffs: dev is 32505858 name is "mtdblock2"
yaffs: passed flags ""
yaffs: Attempting MTD mount on 31.2, "mtdblock2"
sh: can't access tty; job control turned off
# init: cannot find '/system/bin/playmp3', disabling 'bootsound'
eth0: link up
warning: `rild' uses 32-bit capabilities (legacy support in use)
init: sys_prop: mis-match msg size recieved: -1 expected: 128
```

从启动信息可以看出，其主 CPU 为 ARM926EJ-S,并非 ARM11 CPU, 说明下载的 Emulator 内核并非是针对 G1 手机的，估计只是实现了对 ARM926EJ-S CPU 的模拟。

以上为 Android 内核的大致分析，希望能给有兴趣的人员以简单帮助。

