

文章编号:1003-5850(2008)11-0031-03

基于 OMAP 的 PDA 的 Linux 2.6 移植

Linux 2.6 Porting to PDA based on OMAP

韩金燕 卢素锋

(淮安信息职业技术学院 淮安 223003)

【摘要】在嵌入式应用开发中,操作系统的建立是上层应用开发的基础,具有重要的地位。介绍了由 OMAP 处理器构成的 PDA 模型及其系统固件的建立过程,包括开发环境的配置、存储空间的分配、U-boot 的移植、GNU/Linux 2.6 的移植、根文件系统的制作等。移植后,系统运行良好,为进一步的开发提供了基本条件。

【关键词】PDA, OMAP, Linux

中图分类号: TP316

文献标识码: A

ABSTRACT In embedded system development, building of the embedded operating system is basis for upper user application soft wares, it has very important significance. Building process of system's firmware is described, which is a PDA model based on TI OMAP proccesser, such as configuration of development environment, memory space arrangement, porting of U-boot, porting of GNU/Linux 2.6, building of root file system, etc. After such tasks, the PDA model worked well, and got a basic condition to support future development tasks.

KEYWORDS PDA, OMAP, Linux

1 系统软件的开发环境

1.1 交叉工具链的要求

Linux 2.6 的代码需要 GCC3.4.0 以上的编译器才可以编译,用于 Linux 2.4 的 GCC 2.95.3 或 GCC 3.3.2 不能胜任高版本的 Linux 2.6.1x 或 Linux 2.6.2x 的编译任务。本实验采用的交叉工具链是 ARM-Linux-GCC 3.4.5,该交叉工具链可通过配置编译 GNU 的工具链得到。

1.2 OMAP 处理器的 bootloader

上电后,OMAP5912 的 MPU 从 0x0 开始执行代码,由于设置的不同,这个 0x0 可以是 CS0 映射的地址空间,也可以是 CS3 映射的地址空间,如果是 CS0 映射的地址空间,则 MPU 执行片内 0x0-0xFFFF 固化的 Boot ROM,并接受此 Boot ROM 引导的进一步的寻址操作。如果是 CS3 映射的地址空间,由于 CS3 一般片外扩展 NOR Flash,而 NOR Flash 的起始位置一般放置 U-Boot(对于 Linux),于是 MPU 执行 NOR Flash 中的 U-Boot 代码。

1.3 裸板的 U-boot 烧入

为了后续工作的进行,必须先给 PDA 的 NOR Flash 烧入一个可以使用的 u-boot,烧入前需通过跳线将 OMAP5912 的 GPIO13 设为高电平,使能通用目的处理器的完全加载功能。

初始 u-boot 的烧入方法很多,可以像常规的

ARM 嵌入式系统开发那样,由 ADS1.2 环境和 Multi-ICE 仿真器烧录 u-boot 映像,也可以通过 JTAG 仿真器和 TI CCS 环境完成,这里选择后者,在 CCS 环境下建立一个 NOR Flash 驱动工程,将修改好的 u-boot 文件加入,编译生成 .out 格式的可执行文件,运行后即完成任务。由于不同的 Linux 2.6 对 bootloader 的需求不同,因此在后续过程中仍需要进一步更新 U-boot,详见本文第 3 节。

1.4 存储空间的分配

如图 1 所示,PDA 共有 32MB NOR Flash, 64MB NAND Flash, 64MB Mobile DDR SDRAM, NOR Flash 的地址范围是 0x0-0x2000000, NAND Flash 的地址范围是 0x8000000-0xC000000, Mobile DDR SDRAM 的地址范围是 0x10000000-0x14000000。

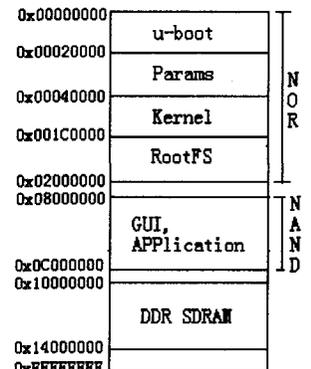


图 1 存储空间的分配

对于 32MB 的 NOR Flash,0x0-0x20000 空间存放系统的 Bootloader U-boot 映像,0x20000-0x40000 空间存放 Linux 内核的启动传递参数 Params, 0x40000-0x1C0000 空间存放 Linux 内核映像, 0x1C0000-0x2000000 空间存放 Busybox 等基本文件

* 2008-05-16 收到,2008-09-13 改回
* * 韩金燕,女,1980 生,硕士研究生,研究方向:嵌入式系统。

系统映像。

64MB NAND Flash 作为另一个大分区空间存放 PDA 的 GUI、应用程序等映像。

2 U-boot 的移植

U-boot 最初是德国工程师 Wolfgang Denk 开发的一个针对嵌入式应用的支持多处理器多开发板的 bootloader, 目前演变成一个开源项目, 其代码可以在官方网站下载到, 最新版本是 1.2.x, 本次实验采用的是 1.2.0, 它有直接支持接近本 PDA 硬件结构的 OMAP5912OSK 选项, 采用该选项配置, 移植非常方便。

为了使用 PDA 的 64MB DDR SDRAM 空间来暂存下载的体积庞大的根文件系统映像文件, 需要修改 U-boot 1.2.0 的 include/configs 目录下的 omap5912osk.h 文件, 修改 CFG_MEMTEST_END 和 PHYS_SDRAM_1_SIZE 的值为 0x14000000。

U-boot 的编译命令如下:

```
export PATH =/opt/crosstool/gcc-3.4.5-glibc-2.3.6/
arm-softfloat-linux-\
gnu/bin:$PATH /* 设置交叉编译器路径 */
cd /usr/src
tar jxvf /lusufeng/u-boot-1.2.0.tar.bz2
cd u-boot-1.2.0/include/configs
vi omap5912osk.h /* 修改 CFG_MEMTEST_END 和
PHYS_SDRAM_1_SIZE */
make distclean
make omap5912osk_config /* 配置 */
make /* 编译 */
cp u-boot.bin /tftpboot
```

编译结束后, 会在当前目录生成 u-boot.bin 文件, 这就是需要的 u-boot 映像, 对于已经具有一个可以运行的 u-boot 的 OMAP5912 电路板, 更新 u-boot 映像比较容易, 当 u-boot 启动时, 在 PC 机的 minicom 下按空格键进入 u-boot 控制界面, 输入如下命令:

```
setenv ethaddr 00:0E:99:02:0A:52 /* 必须和 PDA 的网
卡 MAC 一致 */
setenv ipaddr 192.168.0.156 /* PDA 实验板的 IP 地址
*/
setenv serverip 192.168.0.201 /* PC 机的 IP 地址 */
setenv netmask 255.255.255.0
setenv gatewayip 192.168.0.1
saveenv /* 以上是配置 u-boot 的网络环境 */
tftpboot 0x10000000 u-boot.bin
protect off 1:0
erase 1:0
cp.b 0x10000000 0x0 192c8 /* 192c8 是 16 进制表示的 u-
```

boot.bin 的体积 */

3 GNU/Linux 2.6 的移植

3.1 OMAP Linux 的选择

OMAP5912 支持目前主流的嵌入式操作系统, 如 VxWorks、Windows CE、Montavista Linux 和普通 Linux, 本实验采用的是普通的 GNU/Linux 源代码文件 linux-2.6.21.tar.bz2, OMAP 补丁文件是 patch-2.6.21-omap1.bz2。

3.2 编译过程

为 Linux 打 OMAP 补丁的命令如下:

```
cd /usr/src
tar jxvf /linux-2.6.21.tar.bz2
cp /patch-2.6.21-omap1.bz2
bunzip2 patch-2.6.21-omap1.bz2
cd linux-2.6.21
cat ... /patch-2.6.21-omap1 | patch-p1
```

配置命令如下:

```
make clean
make omap_osk_5912_defconfig
make xconfig /* 可根据实际情况进一步修改 OMAP 缺
省配置选项, NFS 必选
```

make

linux 映像生成命令如下:

```
arm-linux-objcopy-O binary-R.note-R.comment-S arch/
arm/boot/compres\sed/vmlinux linux.bin
gzip -9 linux.bin
/usr/src/u-boot-1.2.0/tools/mkimage -A arm -O linux -
T kernel -C gzip -a 0x
\ 10c08000 -e 0x10c08000 -n 'OMAPlinux 2.6 kernel
image' -d linux.bin.gz uImage
cp uImage /tftpboot
```

3.3 下载内核映像

```
tftpboot 0x10000000 uImage
erase 1:2-12
cp.b 0x10000000 0x00040000 17d83a /* 17d83a 是 16 进
制表示的 uImage 体积
setenv bootcmd bootm 0x00040000 /* 以下配置内核工作
环境 */
setenv bootdelay 3
setenv bootfile uImage
setenv bootargs console=ttyS0,115200n8 nointrd rw ip
=dhcp root=/dev/nfs \ nfsroot=192.168.0.201:/data/
rootfs2.6,nolock mem=62M
saveenv
```

上述命令用来测试刚生成的 Linux 内核是否可以正常运行, 其中需要主机 IP 地址是 192.168.0.201 的 /data/rootfs2.6 文件系统的支持, 该文件系统采用如

下方法得到。

4 根文件系统的建立

根文件系统包含支持 Linux 系统运行的所有文件,通常包括基本的文件系统结构和基本的运行库。

对于本 PDA,需要的基本目录有/dev,/proc,/bin,/sbin,/lib,/etc,/usr,/tmp 等。基本的工具有 cp, cd,mv,ls,export,echo,mkdir,vi 等。基本的配置文件有 rc,inittab,fstab 等。基本的设备有/dev/tty *, /dev/ttyS *, /dev/console 等,这些共同组成一个简单的文件系统。

Busybox 是一个由 Bruce Perens 和 Erik Andersen 负责的开源项目,它提供小体积的可配置的 ash shell 功能,本 PDA 的文件系统可以在 Busybox 的基础上修改而来^[2],其中,/etc/init.d/rcS 文件修改如下:

```

PATH=/bin:/sbin:/usr/bin:/usr/sbin
HOSTNAME=OMAP5912PDA
hostname $HOSTNAME
echo ""
echo " * * * * * "

```

```

* * * * * "
echo "Starting System Init"
echo " * * * * * "
* * * * * "
* * * * * "
* * * * * "
cd /etc/init.d
mount-a

```

将上述文件系统中所有文件用 mkfs.jffs2 工具制成一个映像文件,烧入 NOR Flash 中 0x001C0000-0x02000000 空间,至此,OMAP Linux 2.6 基本移植完毕,PDA 上电复位后,运行画面如图 2 所示。



图 2 OMAP Linux 2.6 运行画面

参考文献

[1] 韩金燕,卢素锋,王胜坤.一种基于 TI OMAP 的 PDA 的设计[J].电脑开发与应用,2007,20(2):47-49.
[2] 杨延军.用 busybox 制作嵌入式 Linux 的文件系统[J].单片机与嵌入式系统应用,2005,4(4):8-10.

(上接第 30 页)

6 结论

PSO 算法是一种相当简单的算法,只需很少的代码和参数,但在各种问题的求解与应用中却展现了它的特点和魅力。该算法为人们提供了如下一种思路:使智慧出现而不是努力强迫它;模拟自然而不是力图控制它;寻求使事情简单化而不是让它复杂。

目前我国针对 PSO 算法的研究还很少,而针对具体应用领域的研究则更为重要,以便促进我国高新技术的迅速发展。

参与文献

[1] Kennedy J,Eberhart R. Particle Swarm Optimization [A]. Proc IEEE Int Conf on Neural Networks[C]. Perth,1995;1 942-1 948.
[2] Eberhart R, Kennedy J. A New Optimizer Using Particle Swarm Theory [A]. Proc 6th Int Symposium on Micro Machine and Human Science [C]. Nagoya, 1995;39-43.
[3] 谢晓锋,张文俊,杨之廉.微粒群算法综述[J].控制与决策,2003,18(2):129-134.
[4] Fogel L J, Walsh M J. Artificial Intelligence Through

Simulated Evolution [M]. Chichester: John Wiley, 1966.

[5] Holland J H. A daptation in Natural and Artificial System s[M]. Ann Arbor: University of Michigan Press, 1975.
[6] 张千里,李 星.基于粒子群优化算法的模糊模拟[J].计算机工程,2006,32(21):33-34.
[7] 乔立岩,马云彤,彭喜元.离散二进制微粒群算法[J].电子测量与仪器学报,2005(增刊):16-18.
[8] Ray T, Liew K M. A Swarm with an Effective Information Sharing Mechanism for Unconstrained and Const Rained Single Objective Optimization Problems[A]. P roc IEEE Int Conf on Evolutionary Computation [C]. Seoul, 2001;75 280.
[9] Yoshida H, Kawata K, Fukuyama Y, et al. A Particle Swarm Optimization for Reactive Power and Voltage Control Considering Voltage Security Assessment [J]. Trans of the Institute of Electrical Engineers of Japan, 1999,119-B(12):1 462-1 469.
[10] Eberhart R, Shi Y H. Tracking and Optimizing Dynamic Systems with Particle Swarms [A]. Proc IEEE Int Conf on Evolutionary Computation [C]. Hawaii,2001;94-100