

K9 Linux_2.4.19 移植

1. 准备工作

- a) **交叉开发环境的安装** 交叉开发环境, 安装在位置/usr/local/arm/2.95.3/bin。
- b) **内核原码的安装** 源码程序 linux-2.4.19-rmk7, 拷贝到/usr/local/arm/k9l 下面, 并且解压缩, 得到内核源码
- c) **安放 Uboot 工具** 源码程序 k9uboot, 拷贝到/usr/local/arm/k9s 下面。

2. 预先设置

主要在内核源码中设置 makefile 文件, 主要设置两个地方 ARCH 和 CROSS_COMPILE。

ARCH :=arm ; 表示目标板为 arm

CROSS_COMPILE=/usr/local/arm/2.95.3/bin/arm-linux- ; 设置交叉编译工具的地址。

3. 内核配置

在内核源码/usr/local/arm/k9l/ linux-2.4.19-rmk7 下

make menuconfig ; 菜单界面

或

make xconfig ; 图形界面, 通常采用

4. 内核编译

在内核源码/usr/local/arm/k9l/ linux-2.4.19-rmk7 下

make clean

make dep

make

5. 建立内核的影象

在内核源码/usr/local/arm/k9l/ linux-2.4.19-rmk7 下

make Image ; 形成 vmlinux

cp vmlinux /usr/local/arm/2.95.3/bin ; copy

在编译工具/usr/local/arm/2.95.3/bin 下

./arm-linux-objcopy -O binary -S vmlinux linux.bin ; 形成 linux.bin

gzip -v9 linux.bin ; 形成 linux.bin.gz

cp linux.bin.gz /usr/local/arm/k9s/k9uboot/tools ; copy

在 k9uboot 的目录/usr/local/arm/k9s/k9uboot/tools 下

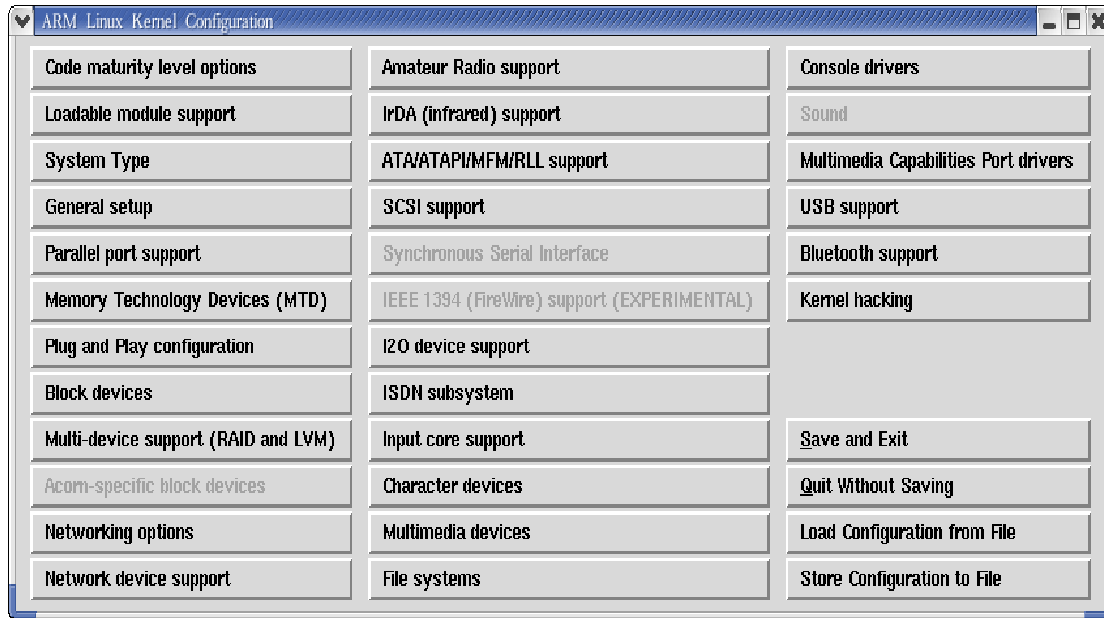
./mkimage -A arm -O linux -C gzip -a 0x20008000 -e 0x20008000 -d linux.bin.gz uImage ; 把 linux.bin.gz 文件变为最终的影象文件 uImage

最后, 检查 uImage 大小, 尽量使 uImage 小于 8*128=768(KB)。

至此, Linux 内核移植完成, 我们生成了 linux-2.4.19-rmk7 内核 for K9 的 linux 内核烧写文件 uImage。

6. 如何配置和裁剪 Linux 的内核

在 Linux 下, 用 `make xconfig` 或者 `make menuconfig` 进入配置的界面。在内核配置中, 一般有四种选择: Y (选种)、N (不选)、M (模块) 和数字, 用户可以根据裁剪需要进行设置, 最后配置完毕, 选择是否对配置结果进行保存? 图形配置界面如下:



Block Devices:

→ Network block device support : n

→ -> Ramdisk support: y

→ -> Default Ramdisk size: 15360

好像不能自行改为 4096

真实的 Ramdisk 大小在 Uboot ENV 里面配置

→ -> Initial Ramdisk (initrd) support: y

File System -> Network File systems:

→ NFS file system support : n

→ -> NFS server support: n

Console drivers -> Frame Buffer Support:

→ support for frame buffer devices (exp): y # K9 不支持屏, 可选 n

→ -> epson LCD/CR/TV controller support: y

→ -> epson S1 D1 3806 support for AT91RM9200DK: y

→ -> virtual frame buffer support (only for testing!): n

→ -> advanced low level driver option: y

→ -> 16 bpp packed pixel support: y (others n)

USB Support:

→ support for USB: y # K9 不支持 USB, 可选 n

→ -> AT91RM9200 OHCI- compatible host interface: y

→ -> USB mass storage support : y

→ -> USB Human Interface device (full HID) support: y

→ -> HID input layer support : y

Kernel configuration when using NFS ramdisk:

如需支持 NFS

Below the details of the parameters when typing: make xconfig:

General setup:

-> Default kernel string : Erase its contents

Block Devices:

➔ Network block device support : y

➔ -> Ramdisk support: n

File System:

➔ Quota support: n

➔ -> Kernel automounter support: y

➔ -> DOS FAT fs support: y

➔ -> VFAT (Windows 95) fs support: y

➔ -> Journalling flash file system v2 (JFFS2) support: 0

➔ -> /proc file system support: y

➔ -> /dev file system support (EXP): y

➔ -> Automatically mount a boot: y

➔ -> Second extended fs support: y

File System -> Network File systems:

➔ NFS file system support : y

➔ -> provide NFSv3 client support: y

➔ -> root file system on NFS: y

➔ NFS server support: n