



## QY-1808EK LINUX用户手册

版本号 v1.0

2013 年 07 月

杭州启扬智能有限公司版权所有  
QIYANG TECHNOLOGY Co., Ltd  
Copyright Reserved

## 前言

欢迎使用杭州启扬智能科技有限公司产品 QY-1808EK，本产品 Linux 部分包含 3 份手册：QY-1808EK linux 用户手册、QY-1808EK 硬件说明以及 QY-1808EK Linux 测试手册。硬件相关部分可以参考 QY-1808EK 硬件说明，主板测试可以参考 QY-1808EK Linux 测试手册。

使用之前请仔细阅读 QY-1808EK Linux 用户手册以及 QY-1808EK 硬件说明！

### 公司简介

杭州启扬智能科技有限公司位于美丽的西子湖畔,是一家集研发、生产、销售为一体的高新技术产业。公司致力于成为嵌入式解决方案的专业提供商,为嵌入式应用领域客户提供软件开发工具和嵌入式系统完整解决方案。产品范围主要包括: Cirrus Logic EP93xx 系列 ARM9 主板、ATMEL AT91SAM926x 系列主板, FreeScale iMX 系列主板, TI Davinci 系列音/视频通用开发平台等等。可运行 Linux2.4/2.6、WinCE5.0/6.0 操作系统,并可根据客户需求开发各种功能组合的嵌入式硬件系统。应用领域涉及: 工业控制、数据采集、信息通讯、医疗设备、视频监控、车载娱乐等等。

客户的需求是公司发展的动力,公司将不断完善自身,与客户互助互惠,共同发展。

电话: 0571-87858811, 87858822

传真: 0571-87858822

技术支持 E-MAIL: support@qiyangtech.cn

网址: <http://www.qiyangtech.com>

地址: 杭州市西湖科技园西园一路8号3A幢5层

邮编: 310012

## 目 录

一、Linux 开发环境搭建 .....	4
1.1、虚拟机安装.....	4
1.2、Linux 主机安装 .....	4
1.3、构建 WinXP 和 Ubuntu 文件共享 .....	4
1.4、arm 交叉编译器安装.....	5
1.5、NFS 服务器安装 .....	6
1.6、tftp 服务器安装 .....	6
二、QY-1808EK 启动烧写步骤.....	11
2.1、UBL 与 U-boot 烧写 .....	11
2.2、内核文件 uImage 烧写 .....	12
2.3、Yaffs2 文件系统烧写 .....	14
三、AM1808 u-boot 及 kernel 编译 .....	16
3.1、uboot 编译.....	16
3.2、kernel 编译.....	17

## 一、Linux开发环境搭建

### 1.1、虚拟机安装

虚拟机可以选择 Vmwarework station.具体安装请参考本光盘提供的——《虚拟机安装 ubuntu 指导手册》。

### 1.2、Linux主机安装

Linux 主机可以选择 ubuntu 发行版. 具体安装请参考本光盘提供的——《虚拟机安装 ubuntu 指导手册》。

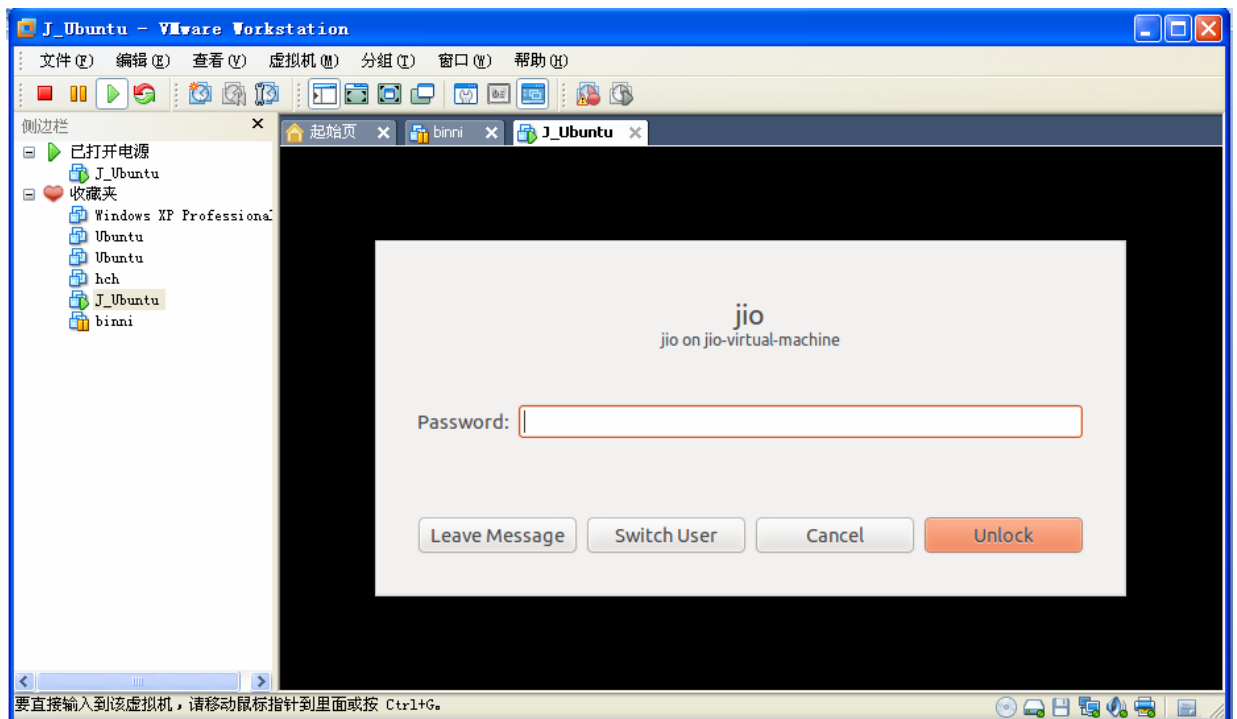
### 1.3、构建WinXP和Ubuntu文件共享

在 Winxp 和 Ubuntu 中实现文件共享可以搭建 SAMBA 服务器，也可以直接使用 Vmware 提供的文件共享功能。

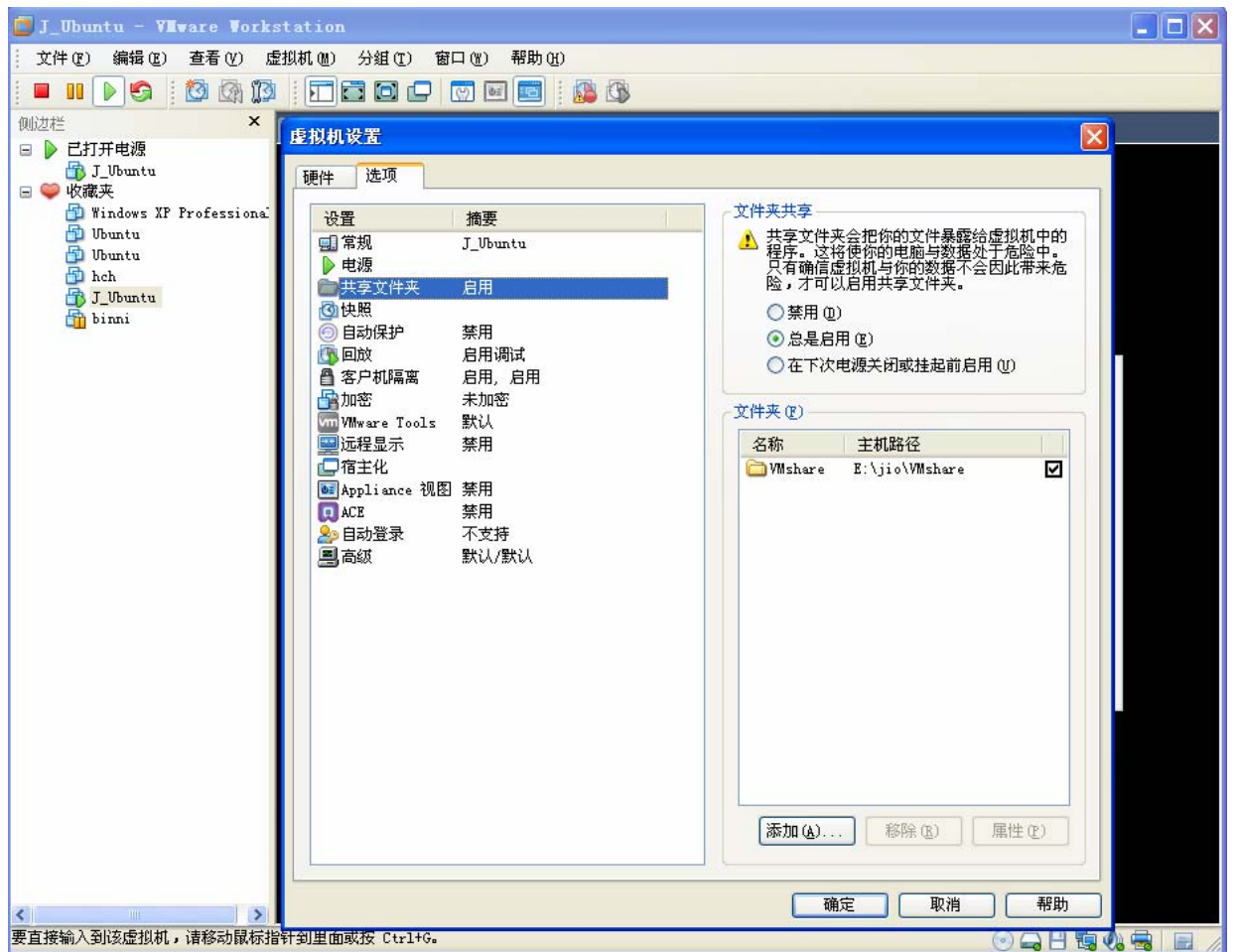
具体搭建请参考文光盘提供的——《虚拟机安装 ubuntu 指导手册》。

这里简单介绍一下直接使用 Vmwate 提供的文件共享功能：

打开安装好的虚拟机 Ubuntu,运行虚拟机



在虚拟机选项中选择设置，然后在虚拟接设置界面中选择选项卡，选择共享文件夹选项。在这里可以添加你要共享的文件夹。



在做完选项配置以后，进入 ubuntu 打开/mnt 目录，能够发现多了一个 hgfs 目录，在这个目录里可以找到刚刚共享的目录。至此 ubuntu 与 windows 文件共享完成。

## 1.4、arm交叉编译器安装

1、拷贝 arm-2007q1-10-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2 到虚拟机 ubuntu ~/opt 下

```
jio@ubuntu:~$ cp arm-2007q1-10-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2 ~/opt
```

```
jio@ubuntu:~$ cd ~/opt
```

```
jio@ubuntu:~$ tar xjvf arm-2007q1-10-arm-none-linux-gnueabi-i686-pc-linux-gnu.tar.bz2
```

这样在~/opt 目录下就会有 arm-2007q1 文件夹，这就是我们所要使用的交叉编译器。

2、把交叉编译器的路径添加到环境变量里

```
jio@ubuntu:~$ cd ~
```

```
jio@ubuntu:~$ vi .bashrc
```

编辑.bashrc 文件，在该文件最后后面添加以下路径

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有

```
export PATH=$PATH:~/opt/arm-2007q1/bin
```

3、使新的环境变量生效

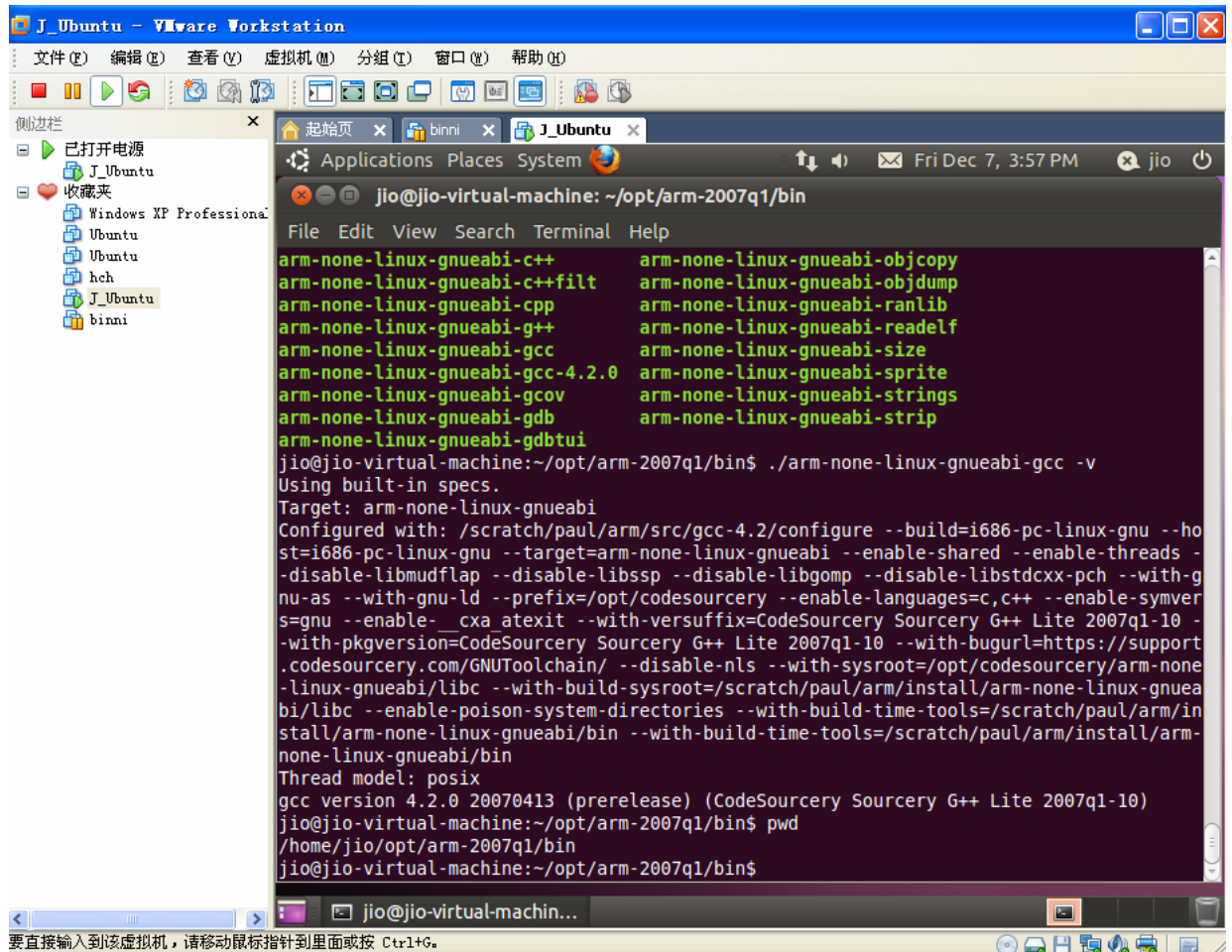
```
jio@ubuntu:~$ source ~/.bashrc
```

4、查看交叉编译器版本信息

在终端输入以下命令来查看交叉编译器是否安装正确。

```
jio@ubuntu:~$ arm-arago-linux-gnueabi-gcc -v
```

如图所示：



## 1.5、NFS服务器安装

具体安装请参考本光盘提供的——《虚拟机安装 ubuntu 指导手册》。

## 1.6、tftp服务器安装

1、建立 tftp 服务器目录，安装 tftp 的 server 和 client，还有 xinetd，进入用户执行如下命令：

```
jio@ubuntu:~$ cd /home/jio/
jio@ubuntu:~$ mkdir tftpboot
```

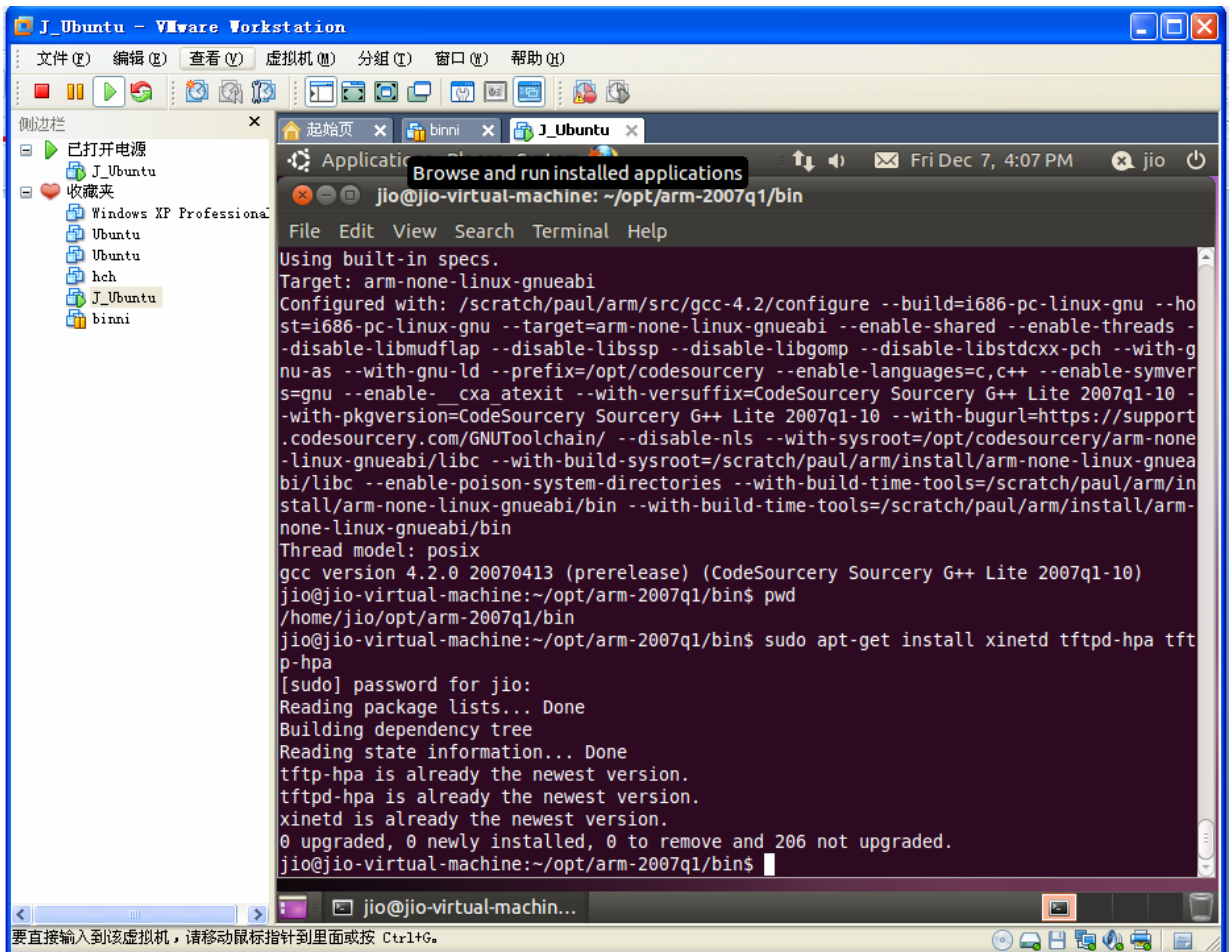
有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

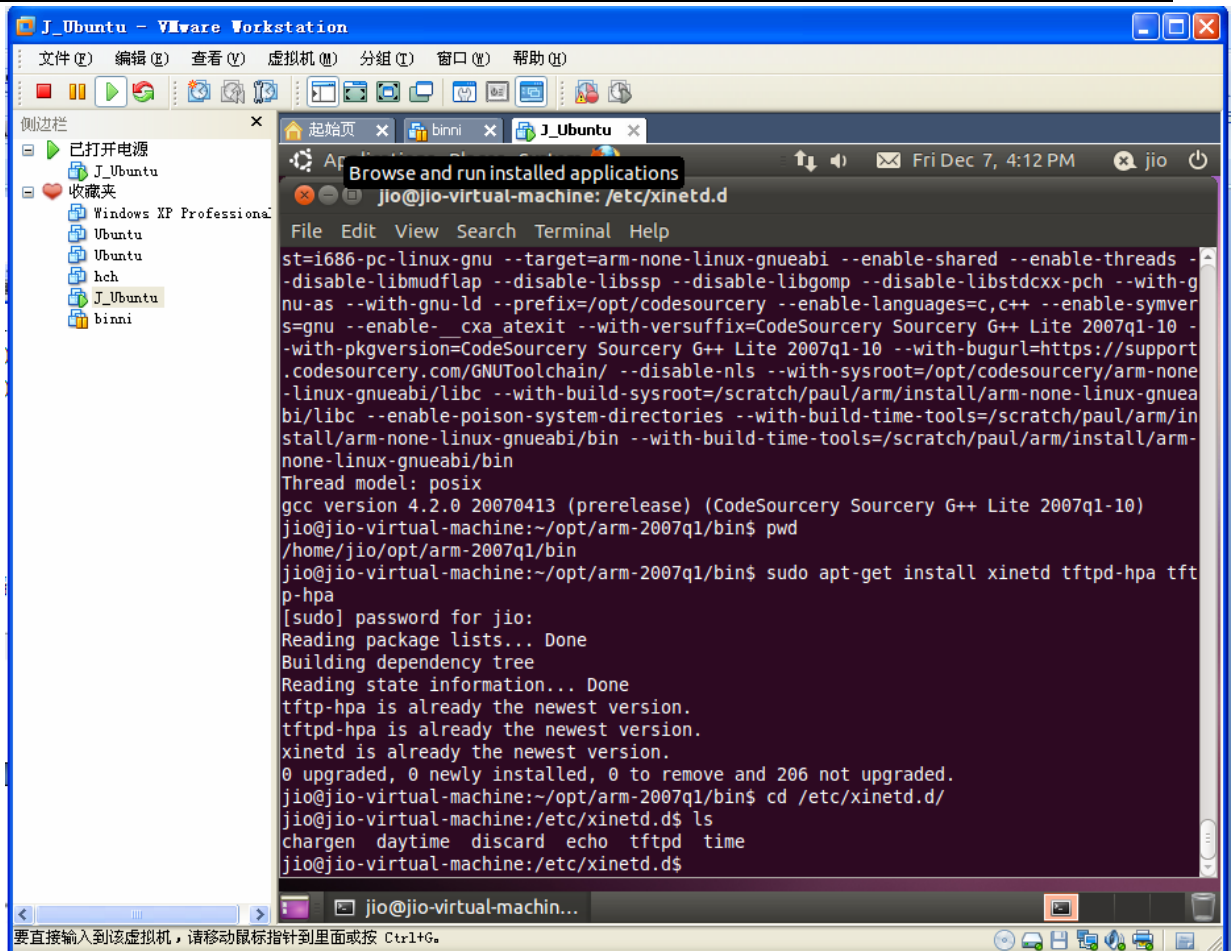
©2012 Qiyangtech 版权所有

```
jio@ubuntu:~$ chmod 777 tftpboot
jio@ubuntu:~$ sudo apt-get install xinetd tftpd-hpa tftp-hpa
```



2、xinetd 会建立目录/etc/xinetd.d/，在这个目录下新建文件 tftpd，执行如下命令

```
jio@ubuntu:~$ cd /etc/xinetd.d/
jio@ubuntu:~$ sudo touch tftpd
```



编辑 /etc/xinetd.d/tftpd，执行如下代码

```
jio@ubuntu:~$ sudo gedit /etc/xinetd.d/tftpd
```

然后加入以下内容：

```
service tftp
{
    disable = no
    socket_type = dgram
    wait = no
    user = root
    protocol = udp
    server = /usr/sbin/in.tftpd
    server_args = -s /home/jio/tftpboot
    log_on_success = PID HOST DURATION
    log_on_failure = HOST
}
```

注意：/home/jio/tftpboot 是你的 tftp 的根目录，其它的不用修改。

### 3、修改/etc/inetd.conf 文件，执行如下命令

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有

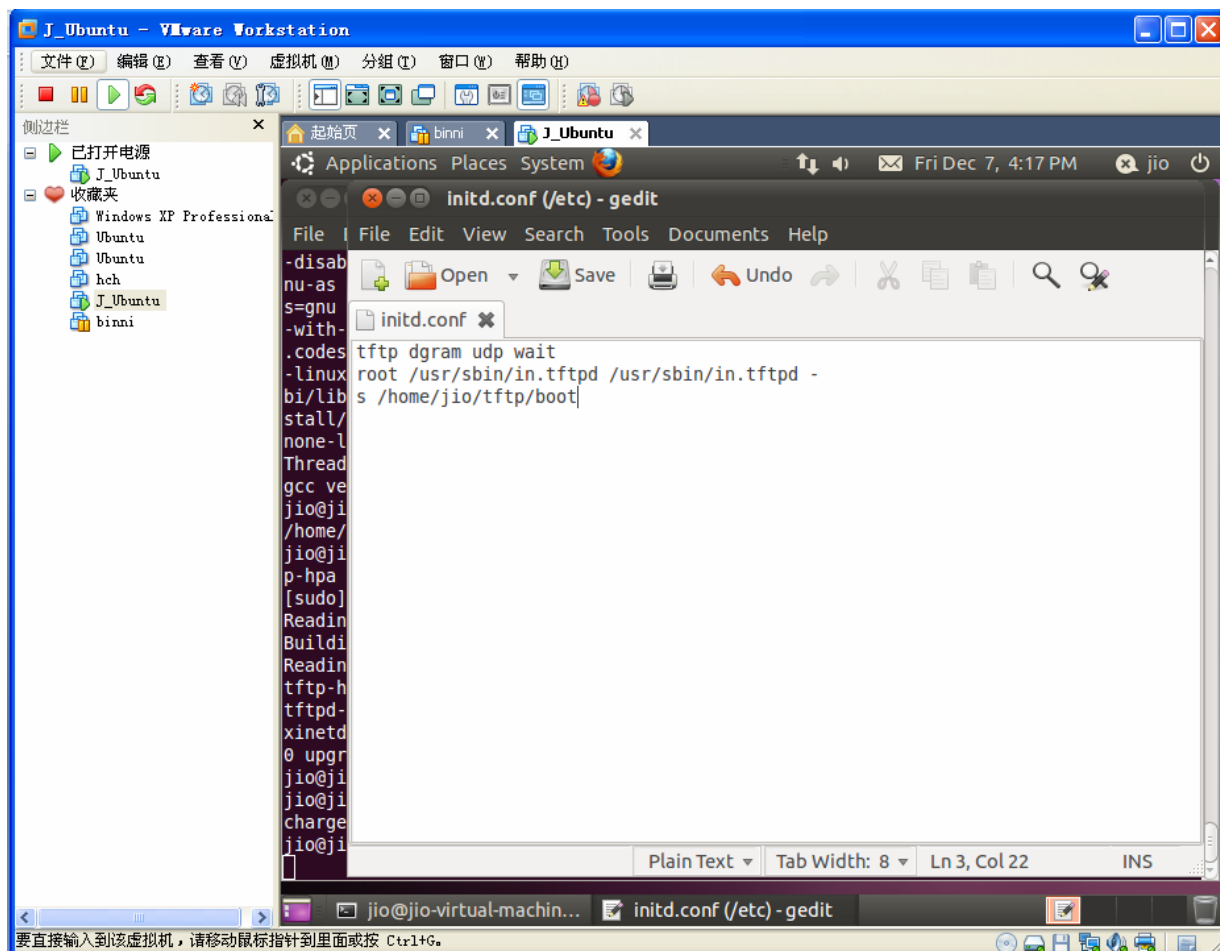


HOST \$ sudo gedit /etc/inetd.conf

然后加入以下代码

tftp dgram udp wait

root /usr/sbin/in.tftpd /usr/sbin/in.tftpd -  
s /home/jio/tftpboot



- 4、修改/etc/default/tftpd-hpa，执行如下命令

HOST \$ sudo gedit /etc/default/tftpd-hpa

修改成以下代码

#Defaults for tftpd-hpa

RUN\_DAEMON="yes"

OPTIONS="-l -s /home/jio/tftpboot"

- 5、重启服务 tftp 服务器，执行如下代码

HOST \$ sudo /etc/init.d/xinetd restart

HOST \$ sudo /etc/init.d/tftpd-hpa restart

- 6、到此 tftp 服务器建立完毕，测试

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有

在/home/jio/tftpboot 下新建一个文件，比如 uImage

然后重启开发板，在重启的开发板的时候在超级终端中会显示 uboot 启动信息，此时输入回车键，将进入 U-Boot 命令行，这个时候可以输入 U-Boot 命令了，这里我们使用 tftp 0xc0700000 uImage 命令，如下图测试成功。

```
U-Boot > tftp 0xc0700000 uImage
Using device
TFTP from server 192.168.1.50; our IP address is 192.168.1.20
Filename 'uImage'.
Load address: 0xc0700000
Loading: #
done
Bytes transferred = 14 (e hex)
U-Boot > █
```

## 二、QY-1808EK启动烧写步骤

TI DAVINCI 系列 CPU 采用 3 级启动 第一级 RBL 第二级 UBL 以及第三级 UBOOT。

RBL ( ARM ROM Boot Loader )在芯片出厂的时候就已经烧写到 ROM 里了, 因此对于 bootloader 这不需要大家关心, 我们需要烧写的有 UBL,UBOOT。

NOTE:板子的调试串口为 COM2(J11)。

### 2.1、UBL与U-boot烧写

在进行烧写之前, 需要安装.NET, 安装文件在光盘工具软件目录下的 dotNetFx40\_Full\_setup.exe, 这个是在线安装包, 具体安装不详细介绍了。

(注意: 核心板插到底板上的时候不能插反, 核心板的 J1 与底板的 J1 对应)

1、拨码开关设置如下 (从串口 2 启动):



1,3 (on) 2,4(off)

2、打开\镜像文件\bootloader\NandErase.bat, 如下图

```

C:\WINDOWS\system32\cmd.exe
E:\jio\AM1808<工大>\QY-AM1808-RELEASE\镜像文件\bootloader>sfh_OMAP-L138.exe -erase
-se -targetType AM1808 -flashType NAND -p COM3
-----
TI Serial Flasher Host Program for OMAP-L138
(C) 2012, Texas Instruments, Inc.
Ver. 1.67
-----

[TYPE] Global erase
[TARGET] AM1808
[DEVICE] NAND
[NAND Block] 1

Attempting to connect to device COM3...
Press any key to end this program at any time.

<AIS Parse>: Read magic word 0x41504954.
<AIS Parse>: Waiting for BOOTME... <power on or reset target now>
    
```

注意: 这个和下面的工具需要用到 PC 机端的串口, 如果当前超级终端占用了串口, 请确保先关闭超级终端。

3、此时给开发板上电, 会显示擦除进度条, 擦除完毕窗口和自动关闭。

有任何技术问题或需要帮助, 请联系: [supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品, 请联系销售: [sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问: <http://www.qiytech.com>

©2012 Qiyangtech 版权所有

4、等窗口关闭后，点击\镜像文件\bootloader\LinuxNandFlash.bat，会弹出如下图窗口

```

C:\WINDOWS\system32\cmd.exe
E:\jio\AM1808(工大)\QY-AM1808-RELEASE\镜像文件\bootloader>sfh_omap-l138.exe -flash -targetType AM1808 -flashType NAND -v -p COM3 ubl_AM1808_NAND.bin u-boot.bin

-----
TI Serial Flasher Host Program for OMAP-L138
(C) 2012, Texas Instruments, Inc.
Ver. 1.67
-----

[TYPE] UBL and application image
[UBL] ubl_AM1808_NAND.bin
[APP IMAGE] u-boot.bin
[TARGET] AM1808
[DEVICE] NAND
[NAND Block] 1

Attempting to connect to device COM3...
Press any key to end this program at any time.

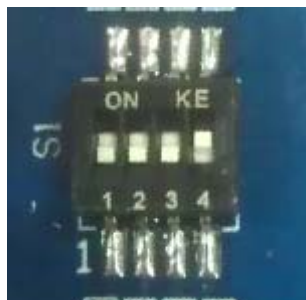
<AIS Parse>: Read magic word 0x41504954.
<AIS Parse>: Waiting for BOOTME... (power on or reset target now)
    
```

重新上电，系统会自动把 ubl\_AM1808\_NAND.bin 和 u-boot.bin 烧写完成，烧写完毕，窗口自动关闭。

注意:在此操作过程会使用电脑 PC 端的 COM1,如果使用其他的 COM 口,可以修改 NandErase.bat 和 LinuxNandFlash.bat 中把 COM1 修改成相应的端口号。

## 2.2、内核文件uImage烧写

1、拨码开关设置（从 nand 启动）



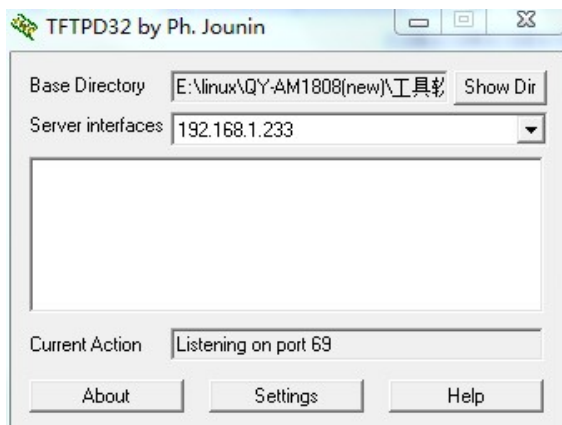
1,2,3 (off) 4(on)

2、打开串口终端，重新上电，此时在串口终端中会显示启动信息，出现 Hit any key to stop autoboot: 字符串时按下任意按键，进入 u-boot 终端，如下图所示

```

AM1808 initialization passed!
Booting TI User Boot Loader
    UBL Version: 1.65
    UBL Flashtype: NAND
Starting NAND Copy...
Valid magicnum, 0x55424CBB, found in block 0x00000002.
    DONE
D: 128 MiBentry point at 0xC1080000.
MMC:  davinci: 0
nand_bbt: ECC error while reading bad block table
In:    serial
Out:   serial
Err:   serial
Net:   DaVinci-EMAC
Hit any key to stop autoboot:  0
U-Boot >
    
```

3、打开在 windows 操作系统打开光盘-工具软件-tftpd32



- 4、把\镜像文件\kernel\uiImage 拷贝到 tftpd2 软件所在的目录。
- 5、在“U-Boot > ”提示设置开发板的 IP 地址和服务端的 IP 地址  
设置开发板的地址：

U-Boot > setenv ipaddr 192.168.1.103 （由自己设定）

将 Windows 的 IP 地址设置为服务端地址：

U-Boot > setenv serverip 192.168.1.233 （根据 TFTP 设定）

- 5、在“U-Boot > ”提示下输入：

U-Boot > run loadkernel

等待片刻，内核会烧写到 NAND Flash 中，烧写时候终端打印如下信息

```

U-Boot > run loadkernel
Using DaVinci-EMAC device
TFTP from server 192.168.1.100; our IP address is 192.168.1.103
Filename 'uImage'.
Load address: 0xc0700000
Loading: #####
#####
#####
done
Bytes transferred = 2346188 (23cccc hex)

NAND erase: device 0 offset 0x200000, size 0x300000
Erasing at 0x4e0000 -- 100% complete.
OK

NAND write: device 0 offset 0x200000, size 0x300000
3145728 bytes written: OK
U-Boot >
    
```

至此，内核烧写完毕。

## 2.3、Yaffs2文件系统烧写

- 1、把\镜像文件\rootf\rootf.yaffs2 复制到 tftpd2 软件所在的目录。
- 2、在“U-Boot# ”提示下输入：

```
U-Boot# run loadfs
```

文件系统比较大，需要等待片刻，内核会烧写到 NAND Flash 中，出现以下信息表示烧写 ok:

```

#####
#####
#####
#####
#####
#####
done
Bytes transferred = 55524480 (34f3c80 hex)

NAND erase: device 0 offset 0x600000, size 0x7a00000
Skipping bad block at 0x03a00000
Skipping bad block at 0x07f80000
Skipping bad block at 0x07fa0000
Skipping bad block at 0x07fc0000
Skipping bad block at 0x07fe0000

OK

NAND write: device 0 offset 0x600000, size 0x34f3c80
55524480 bytes written: OK
U-Boot >
    
```

至此，Yaffs2 文件烧写完毕。

- 3、设置系统环境变量，按顺序执行以下命令：

```
U-Boot> setenv bootfile uImage
```

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有

```
U-Boot>setenv bootargs console=ttyS2,115200n8 init=/linuxrc root=/dev/mtdblock4 rootfstype=yaffs2  
rootwait rw ip=192.168.1.103:192.168.1.100:192.168.1.1:255.255.255.0 mem=128M
```

```
U-Boot>setenv bootcmd 'nboot.e 0xc0700000 0 0x200000; bootm'
```

```
U-Boot>saveenv
```

出现以下信息:

```
U-Boot > setenv bootargs console=ttyS2,115200n8 init=/linuxrc root=/dev/mtdblock4 rootfstype=yaffs2  
0 mem=128M  
U-Boot > setenv bootcmd 'nboot.e 0xc0700000 0 0x200000; bootm'  
U-Boot > saveenv  
Saving Environment to NAND...  
Erasing Nand...  
Erasing at 0x0 -- 100% complete.  
Writing to Nand... done  
U-Boot >
```

4、在“U-Boot > ”提示下输入:

```
U-Boot > reset
```

此时系统能够解压启动内核，顺利进入文件系统。

### 三、AM1808 u-boot及kernel编译

#### 3.1、uboot编译

1、复制源码\uboot\linux-u-boot-2012.10.tar.gz 到/home/jio(linux 工作目录)下，执行如下命令(根据实际目录调整命令格式)

```
HOST $ cp u-boot-2012.10.tar.gz /home/jio
```

2、解压 u-boot-2012.10.tar.gz，执行如下命令

```
HOST $ cd /home/jio
```

```
HOST $ tar -zxvf u-boot-2012.10.tar.gz
```

```
HOST $ cd u-boot-2012.10/
```

```
uboot-nand-rmii
[hedj@QY-SVN uboot]$ cd uboot-nand-rmii/
[hedj@QY-SVN uboot-nand-rmii]$ ls
api                config.mk          doc                lib_arm           lib_i386           lib_nios2         MAKEALL           onenand_ip1      tools
board             COPYING           drivers           lib_avr32         lib_m68k           lib_ppc           Makefile         post             u-boot
CHANGELOG         cpu               examples         lib_blackfin     lib_microblaze    lib_sh            mkconfig         README           u-boot.bin
CHANGELOG-before-U-Boot-1.1.5  CREDITS           fs               libfdt           lib_mips           lib_sparc         nand_spi         rules.mk         u-boot.map
common           disk             include         lib_generic      lib_nios           MAINTAINERS      net              system.map       u-boot.src
```

3、U-boot 编译，执行如下命令

```
HOST $ make distclean CROSS_COMPILE=arm-none-linux-gnueabi-
```

```
HOST $ make qiyang_am18xxevm_config CROSS_COMPILE=arm-none-linux-gnueabi-
```

```
HOST $ make all CROSS_COMPILE=arm-none-linux-gnueabi-
```

```
(echo create libpost.a; for lib in ; \
do echo addlib $lib; done; echo save) \
| arm-none-linux-gnueabi-ar -M
make[1]: Leaving directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/post'
make -C board/davinci/da8xxevm/
make[1]: Entering directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/board/davinci/da8xxevm'
make[1]: Leaving directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/board/davinci/da8xxevm'
make[1]: Entering directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/board/davinci/da8xxevm'
arm-none-linux-gnueabi-gcc -g -os -fno-common -ffixed-r8 -msoft-float -fno-common -ffixed-r8 -msoft-float -D_KERNEL -DTEXT_BASE=0xc1080000 -I/home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/include -fno-built-in -ffreestanding -nostdinc -isystem /opt/arm-2007q1/bin/./lib/gcc/arm-none-linux-gnueabi/4.2.0/include -pipe -DCONFIG_ARM -D_ARM -marm -mabi=aapcs-linux -mno-thumb-interwork -march=armv5te -march=armv5te -Wall -Wstrict-prototypes -fno-stack-protector -o da850evm.o da850evm.c -c
da850evm.c: in function 'buzzer_hw_init':
da850evm.c:412: warning: control reaches end of non-void function
arm-none-linux-gnueabi-ar crv libda8xxevm.a da850evm.o
a - da850evm.o
make[1]: Leaving directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/board/davinci/da8xxevm'
make -C /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/cpu/arm926ejs/ u-boot.lds
make[1]: Entering directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/cpu/arm926ejs'
make[1]: Nothing to be done for u-boot.lds'.
make[1]: Leaving directory /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/cpu/arm926ejs'
arm-none-linux-gnueabi-gcc -E -g -os -fno-common -ffixed-r8 -msoft-float -fno-common -ffixed-r8 -msoft-float -D_KERNEL -DTEXT_BASE=0xc1080000 -I/home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/include -fno-built-in -ffreestanding -nostdinc -isystem /opt/arm-2007q1/bin/./lib/gcc/arm-none-linux-gnueabi/4.2.0/include -pipe -DCONFIG_ARM -D_ARM -marm -mabi=aapcs-linux -mno-thumb-interwork -march=armv5te -march=armv5te -include /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/include/u-boot/u-boot.lds.h -ansi -D_ASSEMBLY -P -c /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/cpu/arm926ejs/u-boot.lds u-boot.lds
UNDEF_SYM= arm-none-linux-gnueabi-objdump -x board/davinci/da8xxevm/libda8xxevm.a lib_generic/libgeneric.a lib_generic/lzma/liblzma.a lib_generic/lzo/liblzo.a board/davinci/common/libdavinci.a cpu/arm926ejs/libarm926ejs.a cpu/arm926ejs/davinci/libdavinci.a lib_arm/libarm.a fs/cramfs/libcramfs.a fs/fdos/libfdos.a fs/jffs2/libjffs2.a fs/reiserfs/libreiserfs.a fs/ext2/libext2fs.a fs/yaffs2/libyaffs2.a fs/ubiifs/libubiifs.a net/libnet.a disk/libdisk.a drivers/bios_emulator/libatlibioemu.a drivers/block/libblock.a drivers/dma/libdma.a drivers/fpga/libfpga.a drivers/gpio/libgpio.a drivers/hwmon/libhwmon.a drivers/i2c/libi2c.a drivers/input/libinput.a drivers/misc/libmisc.a drivers/mmc/libmmc.a drivers/mtd/libmtd.a drivers/mtd/nand/libnand.a drivers/mtd/onenand/libonenand.a drivers/mtd/ubi/libubi.a drivers/mtd/spi/libspi_flash.a drivers/net/libnet.a drivers/net/phy/libphy.a drivers/net/sk98lin/libsk98lin.a drivers/pci/libpci.a drivers/pccia/libpccia.a drivers/power/libpower.a drivers/spi/libspi.a drivers/rtc/librtc.a drivers/serial/libserial.a drivers/twserial/libtwserial.a drivers/usb/gadget/libusb_gadget.a drivers/usb/host/libusb_host.a drivers/usb/musb/libusb_musb.a drivers/video/libvideo.a drivers/watchdog/libwatchdog.a common/libcommon.a libfdt/libfdt.a api/libapi.a post/libpost.a | sed -n -e 's/^(u-boot_cmd.*)/u1/p | sort | uniq ; cd /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii && arm-none-linux-gnueabi-ld -Bstatic -T u-boot.lds -ttext 0xc1080000 $UNDEF_SYM cpu/arm926ejs/start.o --start-group lib_generic/libgeneric.a lib_generic/lzma/liblzma.a lib_generic/lzo/liblzo.a board/davinci/common/libdavinci.a cpu/arm926ejs/libarm926ejs.a cpu/arm926ejs/davinci/libdavinci.a lib_arm/libarm.a fs/cramfs/libcramfs.a fs/fat/libfat.a fs/fdos/libfdos.a fs/jffs2/libjffs2.a fs/reiserfs/libreiserfs.a fs/ext2/libext2fs.a fs/yaffs2/libyaffs2.a fs/ubiifs/libubiifs.a net/libnet.a disk/libdisk.a drivers/bios_emulator/libatlibioemu.a drivers/block/libblock.a drivers/dma/libdma.a drivers/fpga/libfpga.a drivers/gpio/libgpio.a drivers/hwmon/libhwmon.a drivers/i2c/libi2c.a drivers/input/libinput.a drivers/misc/libmisc.a drivers/mmc/libmmc.a drivers/mtd/libmtd.a drivers/mtd/nand/libnand.a drivers/mtd/onenand/libonenand.a drivers/mtd/ubi/libubi.a drivers/mtd/spi/libspi_flash.a drivers/net/libnet.a drivers/net/phy/libphy.a drivers/net/sk98lin/libsk98lin.a drivers/pci/libpci.a drivers/pccia/libpccia.a drivers/power/libpower.a drivers/spi/libspi.a drivers/rtc/librtc.a drivers/serial/libserial.a drivers/twserial/libtwserial.a drivers/usb/gadget/libusb_gadget.a drivers/usb/host/libusb_host.a drivers/usb/musb/libusb_musb.a drivers/video/libvideo.a drivers/watchdog/libwatchdog.a common/libcommon.a libfdt/libfdt.a api/libapi.a post/libpost.a board/davinci/da8xxevm/libda8xxevm.a --end-group /home/hedj/workspace/hedj/am1808/uboot/uboot-nand-rmii/lib_arm/eabi/compat.o -L /opt/arm-2007q1/bin/./lib/gcc/arm-none-linux-gnueabi/4.2.0 -l gcc -Map u-boot.map -o u-boot
arm-none-linux-gnueabi-objcopy -o src u-boot u-boot.src
arm-none-linux-gnueabi-objcopy --gap-fill=0xff --o binary u-boot u-boot.bin
[hedj@QY-SVN uboot-nand-rmii]$
```

4、编译完毕，会在 u-boot-2012.10 目录下生产可执行代码 u-boot.bin。

5、至此，u-boot 编译完成。

有任何技术问题或需要帮助，请联系：[supports@qiyangtech.com](mailto:supports@qiyangtech.com)

购买产品，请联系销售：[sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问：<http://www.qiytech.com>

©2012 Qiyangtech 版权所有



## 3.2、kernel编译

- 1、复制\源码\kernel\ linux-2.6.37-am1808.tar.gz 到/home/jio(linux 工作目录)目录下:

```
HOST $ cp linux-2.6.37-am1808.tar.gz /home/jio
```

```
HOST $ cd /home/jio
```

- 2、解压 linux-2.6.37-am1808.tar.gz, 执行如下命令:

```
HOST $ tar -zxvf linux-2.6.37-am1808.tar.gz
```

```
HOST $ cd linux-2.6.37-am1808
```

```
[hedj@QY-SVN kernel]$ ls
linux-2.6.33-am1808
[hedj@QY-SVN kernel]$ cd linux-2.6.33-am1808/
[hedj@QY-SVN linux-2.6.33-am1808]$ ls
arch                documentation      ipc                make_modules      pru-firmware-06-02-2011-v1.2.tar.gz  security  yaffs2
block               drivers            kbuild            make_uImage       pru-linux-060211-v1.2.patch          setup
change-version.txt  firmware          kernel            mm                 README                                sound
COPYING             fs                 lib               Module.symvers    REPORTING-BUGS                       tools
CREDITS             include           MAINTAINERS      net                samples                               usr
crypto              init              Makefile          pru                scripts                               virt
```

- 3、内核配置:

若 Ubuntu 未安装 ncurses 先安装一下 ncurses

```
HOST $ sudo apt-get install ncurses-dev
```

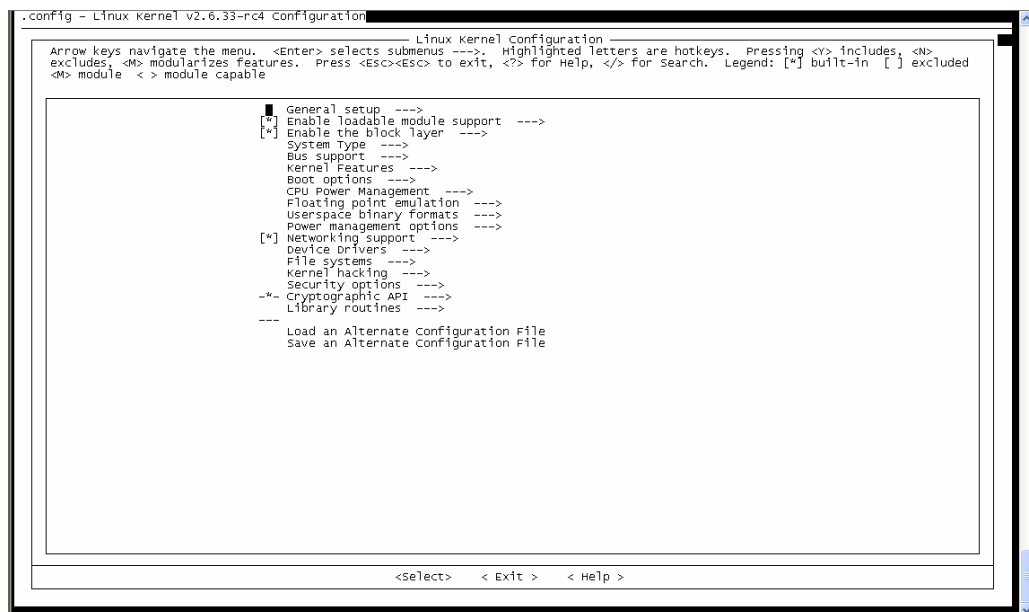
再执行如下命令:

```
HOST $ cd /home/jio/ linux-2.6.37-am1808
```

```
HOST $ make ARCH=arm CROSS_COMPILE=arm-none-linux-gnueabi- menuconfig
```

**注意:** menuconfig 之前有个空格

- 4、重新执行内核配置命令, 会进入内核配置界面, 如果不是这个界面, 则通过 exit 返回上一层, 直到看到这个界面, 如下图所示



有任何技术问题或需要帮助, 请联系: [supports@qiyangtech.com](mailto:supports@qiyangtech.com)

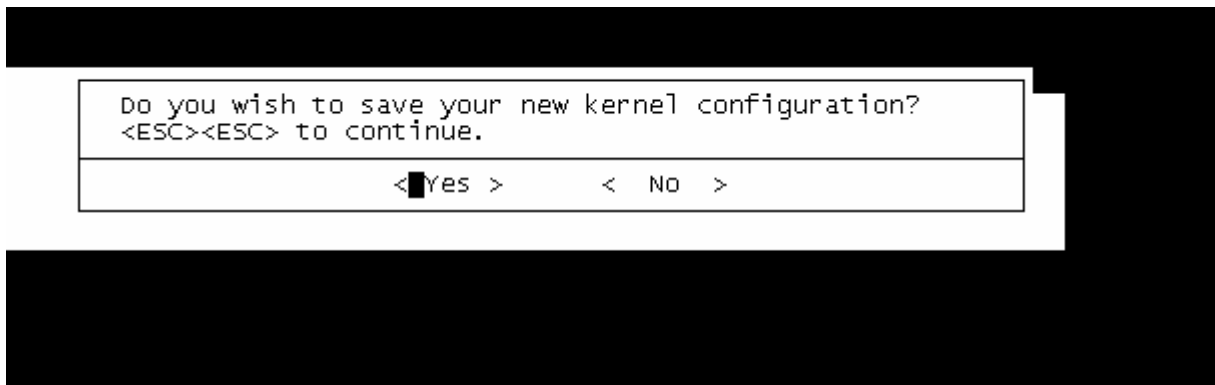
购买产品, 请联系销售: [sales@qiyangtech.com](mailto:sales@qiyangtech.com)

更多信息请访问: <http://www.qiytech.com>

©2012 Qiyangtech 版权所有

这里修改内核配置，对内核进行裁剪。

5、保存退出，执行内核编译命令



```
HOST $ make ARCH=arm CROSS_COMPILE=arm-none-linux-gnueabi- uImage
```

6、编译时间较长，编译完成之后，在内核目录的/arch/arm/boot 文件夹下会生成内核镜像文件 uImage。

注意：如果提示“mkimage” command not found，需要把在编译 u-boot 时编译出来的工具包添加到环境变量里面。执行如下命令：

```
HOST $ gedit ~/.profile
```

添加下面的语句

```
export PATH=$PATH:/home/jio/u-boot/u-boot-nand-rmii/tools
```

```
HOST $ source ~/.profile
```

重新编译内核

```
HOST $ make ARCH=arm CROSS_COMPILE=arm-none-linux-gnueabi- uImage
```

现在就可以看到 uImage 镜像了。

```

CC      lib/kobject_uevent.o
CC      lib/kref.o
CC      lib/plist.o
CC      lib/prio_heap.o
CC      lib/prio_tree.o
CC      lib/proportions.o
CC      lib/radix-tree.o
CC      lib/ratelimit.o
CC      lib/rbtree.o
CC      lib/reciprocal_div.o
CC      lib/rwsem-spinlock.o
CC      lib/sha1.o
CC      lib/show_mem.o
CC      lib/string.o
CC      lib/vsprintf.o
AR      lib/lib.a
LD      vmlinux.o
MODPOST vmlinux.o
GEN     .version
CHK     include/generated/compile.h
UPD     include/generated/compile.h
CC      init/version.o
LD      init/built-in.o
LD      .tmp_vmlinux1
KSYM    .tmp_kallsyms1.5
AS      .tmp_kallsyms1.o
LD      .tmp_vmlinux2
KSYM    .tmp_kallsyms2.5
AS      .tmp_kallsyms2.o
LD      vmlinux
SYSMAP  System.map
SYSMAP  .tmp_System.map
OBJCOPY arch/arm/boot/Image
Kernel: arch/arm/boot/Image is ready
AS      arch/arm/boot/compressed/head.o
GZIP    arch/arm/boot/compressed/piggy.gzip
AS      arch/arm/boot/compressed/piggy.gzip.o
CC      arch/arm/boot/compressed/misc.o
SHIPPED arch/arm/boot/compressed/lib1funcs.s
AS      arch/arm/boot/compressed/lib1funcs.o
LD      arch/arm/boot/compressed/vmlinux
OBJCOPY arch/arm/boot/zImage
Kernel: arch/arm/boot/zImage is ready
UIIMAGE arch/arm/boot/uImage
Image Name: Linux-2.6.33-rc4
Created:   Mon Dec 10 10:51:15 2012
Image Type: ARM Linux Kernel Image (uncompressed)
Data Size: 2092476 Bytes = 2043.43 kB = 2.00 MB
Load Address: c0008000
Entry Point: c0008000
Image arch/arm/boot/uImage is ready
[hedi@qy-svn linux-2.6.33-arm1808]$

```

7、编译内核模块，执行如下命令

HOST \$ make modules ARCH=arm CROSS\_COMPILE=arm-none-linux-gnueabi-

8、至此，内核编译完成。

## 杭州启扬智能科技有限公司

电话：0571-87858811 / 87858822

传真：0571-89935912

支持：0571-89935913

E-MAIL: [supports@qiyangtech.com](mailto:supports@qiyangtech.com)

网址： <http://www.qiytech.com>

地址： 杭州市西湖区西湖科技园西园 1 路 8 号 3A 幢 5F 楼

邮编： 310012