

Rtl8188EUS Driver For Hisi3518A 全部流程

一、编辑 Kernel

在 Linux 系统中，进入到 SDK 包中 Hi3518_SDK_V1.0.9.0/package/osdrv 目录下，然后进到 kernel 文件夹中解压 tar -zxf linux-3.0.y.tgz 并进入该目录。采用 arm-hisiv100nptl-linux 工具链编译内核：cp arch/arm/configs/hi3518a_full_defconfig .config
make ARCH=arm CROSS_COMPILE=arm-hisiv100nptl-linux- menuconfig 。

修改添加内核相较于默认的部分

File systems --->

[*] Miscellaneous filesystems --->

<*> SquashFS 4.0 - Squashed file system support

PPPOE 服务的支持

Device Drivers --->

Network device support --->

<*>PPP support 及以下所有的 ppp 服务

WIFI 的支持

Networking support --->

[*] Wireless --->

<*> cfg80211 - wireless configuration API

[*] Wireless extensions sysfs files

<*> Generic IEEE 802.11 Networking Stack(mac80211)

Device Drivers --->

[*]Network device support --->

[*] Wireless LAN --->

<*> IEEE 802.11 for Host AP (Prism2/2.5/3 andWEP/TKIP/CCMP)

[*] Support downloading firmware images withHost AP driver

[*] Support for non-volatile firmwaredownload

<M> Ralink driver support --->

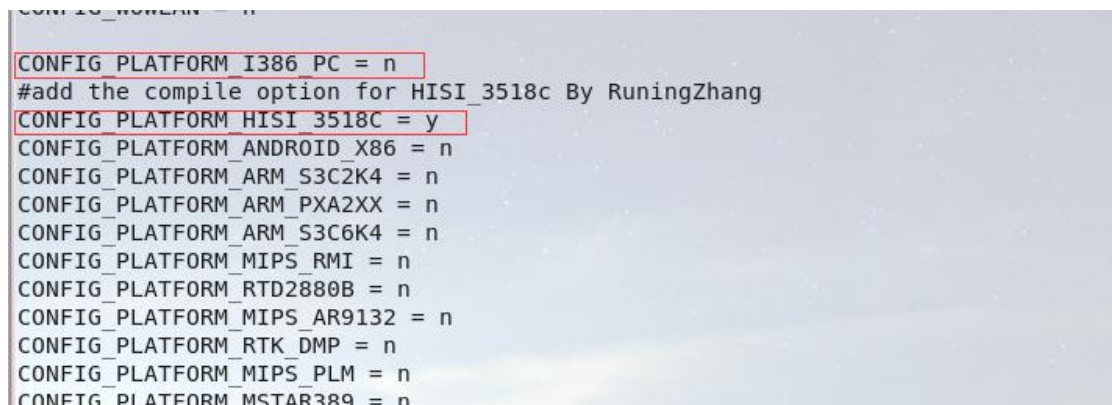
<M> Ralink rt27xx/rt28xx/rt30xx (USB) support

配置完保存，然后 make ARCH=arm CROSS_COMPILE=arm-hisiv200-linux- uImage。

最后将 arch/arm/boot/目录下的 uImage 烧录。

二、编辑 rtl8188eus.ko

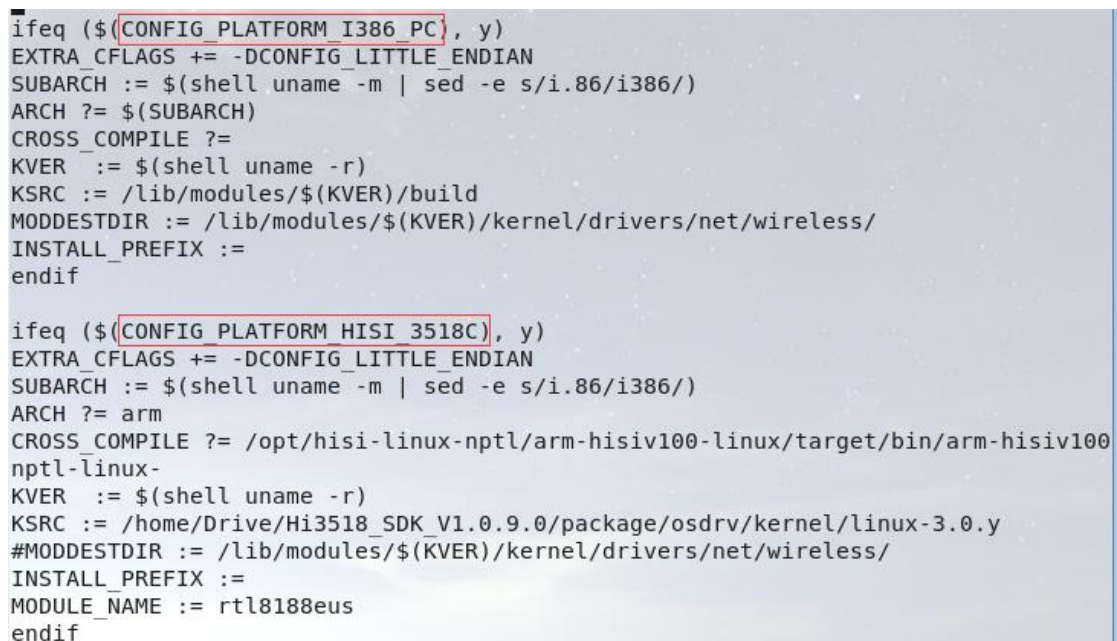
解压 driver 目录 tar zxvf rtl8188EUS_linux_v4.1.4-2_6773.20130326.tar.gz，进入该目录。更改 Makefile 文件中的编译选项，在 Makefile 里找到的目标编译平台开关宏，其中默认配置是 i386 平台，因为里面没有 hisi 的平台，手动添一个 3518，如下图 2-1。



```
CONFIG_PLATFORM_I386_PC = n
#add the compile option for HISI_3518c By RuningZhang
CONFIG_PLATFORM_HISI_3518C = y
CONFIG_PLATFORM_ANDROID_X86 = n
CONFIG_PLATFORM_ARM_S3C2K4 = n
CONFIG_PLATFORM_ARM_PXA2XX = n
CONFIG_PLATFORM_ARM_S3C6K4 = n
CONFIG_PLATFORM_MIPS_RMI = n
CONFIG_PLATFORM_RTD2880B = n
CONFIG_PLATFORM_MIPS_AR9132 = n
CONFIG_PLATFORM_RTK_DMP = n
CONFIG_PLATFORM_MIPS_PLM = n
CONFIG_PLATFORM_MSTAR389 = n
```

图 2-1 修改驱动源文件 Makefile 编译平台

接下来是创建新增的目标编译平台配置项，还是在 Makefile 里面，继续往下找，可以找到。更改 3518 平台时候可以参照 i386，如下图 2-1 所示。



```
ifeq ($(CONFIG_PLATFORM_I386_PC), y)
EXTRA_CFLAGS += -DCONFIG_LITTLE_ENDIAN
SUBARCH := $(shell uname -m | sed -e s/i.86/i386/)
ARCH ?= $(SUBARCH)
CROSS_COMPILE ?=
KVER := $(shell uname -r)
KSRC := /lib/modules/$(KVER)/build
MODDESTDIR := /lib/modules/$(KVER)/kernel/drivers/net/wireless/
INSTALL_PREFIX :=
endif

ifeq ($(CONFIG_PLATFORM_HISI_3518C), y)
EXTRA_CFLAGS += -DCONFIG_LITTLE_ENDIAN
SUBARCH := $(shell uname -m | sed -e s/i.86/i386/)
ARCH ?= arm
CROSS_COMPILE ?= /opt/hisi-linux-nptl/arm-hisiv100-linux/target/bin/arm-hisiv100-nptl-linux-
KVER := $(shell uname -r)
KSRC := /home/Drive/Hi3518_SDK_V1.0.9.0/package/osdrv/kernel/linux-3.0.y
#MODDESTDIR := /lib/modules/$(KVER)/kernel/drivers/net/wireless/
INSTALL_PREFIX :=
MODULE_NAME := rtl8188eus
endif
```

图 2-2 创建 3518 目标编译平台配置项

解释一下这些编译参数的含义。

EXTRA_CFLAGS : 附加编译选项, 可用值如下: **CONFIG_BIG_ENDIAN** 和 **CONFIG_LITTLE_ENDIAN**: 是大端存储模式和小端存储模式。两种存储模式在 ARM 上都支持, 而默认是小端存储, 所以这里选择了 **CONFIG_LITTLE_ENDIAN** **CONFIG_MINIMAL_MEMORY_USAGE** 是指定以小内存分配模式。为了在性能较强的平台上获得较好的性能, 驱动会分配大尺寸的连续内存空间作为 TX/RX 的 IO 缓冲区。这种分配模式在嵌入式平台中可能导致内存分配失败, 可以通过定义此项来避免这个问题。考虑到 RPi 的内存比较大, 所以不使用小内存分配模式, 故此项忽略。

ARCH : 目标编译平台。RPi 的 BCM2835 是 ARM 架构, 填"arm"

CROSS_COMPILE : 指定交叉编译工具链。直接在 RPi 上编译, 不需要指定此项

KVER : 内核版本。通过 shell 的 `uname -r` 获取即可

KSRC : 内核源代码路径。填 `/lib/modules/$(KVER)/build`

MODDESTDIR : 模块目标目录。这个选项是用于指定编译好的驱动模块的安装路径, 填 `/lib/modules/$(KVER)/kernel/drivers/net/wireless`。其实也可以不填此项, 等编译完成后将编译好的 .ko 文件复制到指定的目录也可以, 并使用 `depmod -a` 命令将编译好的驱动模块安装到内核中

INSTALL_PREFIX : 模块安装目录。

MODULE_NAME : 模块名称。最后编译出来的驱动文件的名称就是此名称加 .ko。如果没有设置此项, 则默认为 `rtl8188cus`, 即编译好的驱动文件名称为 `rtl8188eus.ko`。关闭 `CONFIG_POWER_SAVING = n`。

最后, 编译 `make` 将 `rtl8188eus.ko` copy 到平台下, 加载驱动 `insmod rtl8188eus.ko`。

三、编辑 wpa_supplicant, hostapd 以及 Wireless_tools

3.1 wpa_supplicant

解压 `tar zxvf wpa_supplicant_hostapd-0.8_rtw_r6747.20130222.tar.gz`, `cd Wpa_supplicant`, `Vim Makefile` 添加 海思的编译链。

```
add --> CC = arm-hisiv100nptl-linux-gcc
make
```

将编译生成好的 `wpa_cli`, `wpa_supplicant` 以及上以及目录中的 `wpa_0_8.conf` 一起拷贝到海思平台上。

`./wpa_supplicant -Dwext -iwlan0 -c /appex/wpa_0_8.conf -B` //我把所有拷贝到 18A 的/appex 下所以路径是/appex 下。

3.2 Hostapd

`cd hostapd/`, `vim Makefile` 同样是添加海思的编译链。

```
add CC = arm-hisiv100nptl-linux-gcc
make
```

将编译好的 `hostapd`, `hostapd_cli` 跟 `hostapd.conf` 拷贝到目标板/appex 下, 进到 /appex 运行 Daemon: `./hostapd rtl_hostapd.conf -B`

3.3 编译 wireless_tool 工具

解压 tar zxvf wireless_tools.30.rtl.tar.gz , cd wireless_tools.30.rtl
vi Makefile
修改 12 行为 : CC = arm-hisiv100nptl-linux-gcc
修改 14 行为 : AR = arm-hisiv100nptl-linux-ar
make
将生成的 iwlist iwconfig copy 到目标板\usr\bin 下
将 iwlib.so copy 到\lib 或者是\usr\lib

四、配置与测试

4.1 连接 wifi 热点

加载驱动 insmod rtl8188eus.ko ,
ifconfig wlan0 up
查看 wifi 驱动是否加载上 ,
ifconfig wlan0
查看附近有哪些 wifi 热点
iwlist wlan0 scan
运行 wpa_supplicant 在后台
//我的 wpa_supplicant 跟 wpa_0_8.conf 默认在/appex 目录下
./wpa_supplicant -Dwext -iwlan0 -c /appex/wpa_0_8.conf -B
./wpa_cli -p/var/run/wpa_supplicant scan
./wpa_cli -p/var/run/wpa_supplicant scan_results
./wpa_cli -p/var/run/wpa_supplicant remove_network 0
./wpa_cli -p/var/run/wpa_supplicant ap_scan 1
//add net work
./wpa_cli -p/var/run/wpa_supplicant add_network
//输入 wifissid
./wpa_cli -p/var/run/wpa_supplicant set_network 0 ssid "" WIFI"
//wifi 密码
./wpa_cli -p/var/run/wpa_supplicant set_network 0 psk "password"
./wpa_cli -p/var/run/wpa_supplicant select_network 0
最后 udhpcp -b -I wlan0 自动获取 ip,即可连接 wifi

4.2 配置 AP 热点

打开 hostapd.conf 文件 , 按照以下参数配置
interface=wlan0
ssid = SSID_NAME //你想设置成的 ssid 名 ssid 是无线终端搜索网络时看见的名字
channel = 6 //可以选择默认通道
hw_mode=g
bridge = br0
wpa=2
wpa_passphrase=password //你想配置的密码 , 但密码不能少于 8 位
wpa_key_mgmt=WPA_PSK //密钥管理算法

```
wpa_pairwise=CCMP
ieee80211n=1
ht_capab=[SHORT -GI-20][SHORT -GI-40][HT40]
driver=XXXX //driver 设置
```

上述配置完成以后，在终端/appex 下执行 `sudo hostapd hostapd.conf -B`(-B 是需要后台运行的时候添加)，到这里，怎表明了第一个过程结束了，也就是，我们在手机终端上可以搜索到 mytest 这个接入点，但是无法连接到这个接入，应该此时出现的情况是：正在获取 IP 地址，接下来给手机终端分配 IP；

4.3 配置 DHCP 服务器

如果安装好了 DHCP 服务器，配置 `dhcpd.conf` 文件，可根据实际情况参照如下配置参数：

```
subnet 10.5.5.0 netmask 255.255.255.224 {
    range 10.5.5.26 10.5.5.30; //这个范围表示你可以连接的终端数
    option domain-name-servers ns1.internal.example.org;
    option domain-name "internal.example.org";
    option routers 10.5.5.1; //设置无线网卡的 IP 地址
    option broadcast-address 10.5.5.31;
    default-lease-time 600;
    max-lease-time 7200;
}
```

最后，运行 `dhcpd dhcpd.conf` 即可用手机连接 AP 热点。