

H3 项目

wifi 模块移植说明文档 V1.0

文档履历

版本号	日期	制/修订人	内容描述
V1.0	2014-11-28		正式版本

confidential

目 录

1. 前言	4
1.1. 编写目的	4
1.2. 适用范围	4
1.3. 相关人员	4
2. 模块介绍	5
3. ap6181	6
3.1. wifi function	6
3.1.1. BoardConfig.mk	6
3.1.2. init.rc	7
3.1.3. eagle_fvd_p1.mk	7
3.1.4. config.xml	8
3.1.5. sys_config.fex	8
3.2. wifi wake host	10
3.3. ap6181 模组移植相关文件	10
4. ap6210	11
4.1. wifi function	11
4.1.1. BoardConfig.mk	11
4.1.2. init.rc	12
4.1.3. eagle_fvd_p1.mk	12
4.1.4. config.xml	13
4.1.5. sys_config.fex	13
4.2. bluetooth function	15
4.2.1. BoardConfig.mk	15
4.2.2. init.rc	15
4.2.3. eagle_fvd_p1.mk	16
4.2.4. libbt config	16
4.2.5. sys_config.fex	17
4.2.6. 蓝牙默认名字的修改	18
4.2.7. config.xml	18
4.3. wifi&bt wake host	18
4.4. ap6210 模组移植相关文件	19
5. rtl8188etv	20
5.1. BoardConfig.mk	20
5.2. init.rc	21
5.3. dolphin_fvd_p1.mk	21
5.4. config.xml	21
5.5. sys_config.fex	22
5.6. rtl8188etv 模组移植相关文件	23
6. ap6330	24
6.1. wifi function	24
6.1.1. BoardConfig.mk	24
6.1.2. init.rc	25
6.1.3. eagle_fvd_p1.mk	25

6.1.4. config.xml.....	26
6.1.5. sys_config.fex.....	26
6.2. bluetooth function.....	28
6.2.1. BoardConfig.mk	28
6.2.2. init.rc.....	28
6.2.3. jaws_optimus.mk	29
6.2.4. libbt config.....	29
6.2.5. sys_config.fex.....	30
6.2.6. 蓝牙默认名字的修改	31
6.2.7. config.xml.....	31
6.3. wifi&bt wake host.....	31
6.4. ap6330 模组移植相关文件.....	32
7. ap6335	33
7.1. wifi function	33
7.1.1. BoardConfig.mk	33
7.1.2. init.rc.....	34
7.1.3. eagle_fvd_p1.mk	34
7.1.4. config.xml.....	35
7.1.5. sys_config.fex.....	35
7.2. bluetooth function.....	36
7.2.1. BoardConfig.mk	37
7.2.2. init.rc.....	37
7.2.3. jaws_tvd_p1.mk.....	38
7.2.4. libbt config.....	38
7.2.5. sys_config.fex.....	39
7.2.6. 蓝牙默认名字的修改	40
7.2.7. config.xml.....	40
7.3. wifi&bt wake host.....	40
7.4. ap6335 模组移植相关文件.....	40
8. rtl8189etv.....	41
8.1. BoardConfig.mk	41
8.2. init.rc.....	42
8.3. eagle_fvd_p1.mk	42
8.4. config.xml.....	42
8.5. sys_config.fex.....	43
8.6. wifi wake host.....	44
8.7. rtl8189etv 模组移植相关文件	45
9. F&Q.....	46
9.1. 如何编译	46
9.2. 编译报错	46
9.3. 修改不起效	46
9.4. 蓝牙注意事项	46
10. Declaration.....	48

1. 前言

1.1. 编写目的

本文档编写目的是让相关人员能按照文档说明把 SDK 所支持的 wifi 使用起来，要使用 SDK 所支持 wifi 的其中一款只需要修改相关配置即可。

1.2. 适用范围

本文档适用于 H3 android4.4.2 平台移植 wifi 使用。

1.3. 相关人员

本文档适合需要使用或移植 wifi 的人员查阅。

Confidential

2. 模块介绍

目前 H3 android4.4.2 平台上已支持 wifi 模组有 6 款，本文档将以 H3 盒子原型机为例说明如何配置每款 wifi 模组。

wifi 模组可分 USB 接口和 SDIO 接口两种类型，部分模组带蓝牙功能，wifi 的全功能包括 station、softap 和 wifi direct，其中 station 是 wifi 的最基本功能，每款 wifi 模组均支持。

本文档会不断更新，文档和代码对应可能会稍有差别。

表 1 H3 android4.4.2 平台 wifi 模组支持列表

模组	功能（协议）	接口	wifi 功能		
			Station	Softap	wifi direct
ap6181	802.11b/g/n	SDIO	✓	✓	✓
ap6210	802.11 b/g/n Bluetooth4.0	SDIO/PCM/UART	✓	✓	✓
rtl8188etv	802.11b/g/n	USB	✓	✓	✓
ap6330	802.11 b/g/n Bluetooth 4.0	SDIO/PCM/UART	✓	✓	✓
ap6335	802.11 b/g/n/a/ac Bluetooth 4.0	SDIO/PCM/UART	✓	✓	✓
rtl8189etv	802.11b/g/n	SDIO	✓	✓	✓

注意：

- 1、到 linux-3.4 目录下，输入 make ARCH=arm menuconfig，然后选择 Device Drivers --->Network device support--->Wireless LAN，即可看到参与编译的 wifi 驱动。
- 2、内核默认是编译蓝牙，如果实际平台没有蓝牙功能可蓝牙去除编译，减少生成的内核大小，修改方法如下。

去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择[*] Networking support --->

<> Bluetooth subsystem support --->

.....

<> RF switch subsystem support --->

3. ap6181

功能：wifi station、softap、wifi direct

接口类型：SDIO

编译模式：编译成内核模块，修改路径如下

去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择 Device Drivers --->

Network device support --->

Wireless LAN --->

<M> Broadcom 4329/30 wireless cards support

[*] Use Customized firmware path

[*] Use Customized NVRAM path

[*] Use Customized config path

Interrupt type (Out-of-Band Interrupt) --->

进入 Interrupt type (Out-of-Band Interrupt)--->，选择

(X) Out-of-Band Interrupt

3.1. wifi function

3.1.1. BoardConfig.mk

BoardConfig.mk 文件决定 android 加载哪一款 wifi 模组，要配置成使用 ap6181 模组需要把 BoardConfig.mk 文件的相关代码修改成如下。

```
# wifi and bt configuration

# 1. Wifi Configuration
#BOARD_WIFI_VENDOR := realtek
BOARD_WIFI_VENDOR := broadcom

# 1.1 broadcom wifi support
ifeq ($(BOARD_WIFI_VENDOR), broadcom)
    BOARD_WPA_SUPPLICANT_DRIVER := NL80211
    WPA_SUPPLICANT_VERSION      := VER_0_8_X
    BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_bcmdhd
    BOARD_HOSTAPD_DRIVER       := NL80211
    BOARD_HOSTAPD_PRIVATE_LIB  := lib_driver_cmd_bcmdhd
    BOARD_WLAN_DEVICE           := bcmdhd
    WIFI_DRIVER_FW_PATH_PARAM   := "/sys/module/bcmdhd/parameters/firmware_path"

    SW_BOARD_USR_WIFI := AP6181

    include hardware/broadcom/wlan/bcmdhd/firmware/firmware-bcm.mk
endif
```

说明：

- 1、“#”符号起注释作用；
- 2、“#BOARD_WIFI_VENDOR := realtek”指明不使用 realtek 系统 wifi 模组；
- 3、“BOARD_WIFI_VENDOR := broadcom”指明使用 broadcom 系统 wifi 模组；
- 4、“SW_BOARD_USR_WIFI := AP6181”指明使用 AP6181 模块；

3.1.2. init.rc

init.rc 是资源和服务配置相关的文件，使用 ap6181 wifi 模组需要作如下修改（部分代码）。

添加以下内容加载 wifi 驱动：

```
#insmod boardcom wifi driver
insmod /system/vendor/modules/bcmdhd.ko
```

添加以下内容，启动 wpa_supplicant：

```
# broadcom wifi service
# 1 broadcom wifi station and softap
service wpa_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -I/system/etc/wifi/wpa_supplicant_overlay.conf \
    -O/data/misc/wifi/sockets \
    -e/data/misc/wifi/entropy.bin -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot

# 2 braodcom wifi sta p2p concurrent service
service p2p_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -I/system/etc/wifi/wpa_supplicant_overlay.conf \
    -O/data/misc/wifi/sockets -N \
    -ip2p0 -Dnl80211 -c/data/misc/wifi/p2p_supplicant.conf \
    -I/system/etc/wifi/p2p_supplicant_overlay.conf \
    -puse_p2p_group_interface=1 -e/data/misc/wifi/entropy.bin \
    -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot
```

3.1.3. eagle_fvd_p1.mk

添加以下内容，使得在设置界面显示 wifi 和 wifi 直连功能：

```
PRODUCT_COPY_FILES += \
    frameworks/native/data/etc/android.hardware.wifi.xml:system/etc/permissions/android.hardware.wifi.xml \
    frameworks/native/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct
```


.xml

3.1.4. config.xml

在 frameworks\base\core\res\res\values\config.xml 文件里添加以下内容，使得在设置界面显示“网络共享与便携式热点”功能：

```
<string-array translatable="false" name="config_tether_wifi_regexs">
    <item>"wlan0"</item>
</string-array>
```

3.1.5. sys_config.fex

要配置成使用 ap6181 模组需要把 sys_config.fex 文件修改成如下（部分代码）。

```
[mmc1_para]
sdc_used          = 1
sdc_detmode       = 4
sdc_buswidth      = 4
sdc_clk           = port:PG00<2><1><2><default>
sdc_cmd           = port:PG01<2><1><2><default>
sdc_d0            = port:PG02<2><1><2><default>
sdc_d1            = port:PG03<2><1><2><default>
sdc_d2            = port:PG04<2><1><2><default>
sdc_d3            = port:PG05<2><1><2><default>
sdc_det           =
sdc_use_wp        = 0
sdc_wp            =
sdc_isio          = 1
sdc_regulator     = "none"

;-----
;wifi/bt/fm/gps/nfc modules configuration
;module_num:
;          0- none
;          1- rtl8188eu(wifi)
;          2- rtl8723bs(wifi+bt)
;          3- ap6181(wifi)
;          4- ap6210(wifi+bt)
;          5- ap6330(wifi+bt)
;          6- ap6335(wifi)
;module_power0: axp used by module, "axp22_dldo1" - use dldo1, not use keep it empty
;module_power0_vol: power0 voltage, mv;
;chip_en:       enable chip io
;lpo_use_apclk: "" - not use, "losc_out"- a23/33, "ac10032k1"、"ac10032k2"、"ac10032k3"- a80
;-----
[module_para]
```

```

module_num          = 3
module_power0       = "axp22_dldo1"
module_power0_vol   = 3300000
module_power1       = "axp15_cldo3"
module_power1_vol   = 3000000
module_power2       = "axp22_lldoio0"
module_power2_vol   = 3000000
module_power3       =
module_power3_vol   =
chip_en             =
lpo_use_apclk       = "ac10032k2"

;-----
;wifi configuration
;wifi_used:         0-not use, 1- use
;wifi_sdc_id:       0- SDC0, 1- SDC1, 2- SDC2, 3- SDC3
;wifi_usbc_id:      0- USB0, 1- USB1, 2- USB2
;wifi_usbc_type:    1- EHCI(speed 2.0), 2- OHCI(speed 1.0)
;wl_reg_on:         wifi function enable io
;wl_host_wake:      wifi device wake-up host
;wl_host_wake_invert: whether wl_host_wake use inverter between ap and module
;
;                   0: not used, 1: used
;-----

[wifi_para]
wifi_used           = 1
wifi_sdc_id         = 1
wifi_usbc_id        = 1
wifi_usbc_type      = 1
wl_reg_on           = port:PL02<1><<default><default><0>
wl_host_wake        = port:PL03<0><<default><default><0>
wl_host_wake_invert = 0

```

说明：

- 1、“；”符号起注释作用；
- 2、“wifi_used”宏赋值为 1 表示使用 wifi，为 0 表示不使用；
- 3、“wifi_sdc_id”宏表示使用哪个 SD 接口连接 SDIO wifi；
- 4、“wifi_num”宏表示选用哪一款 wifi 模组；
- 5、“module_power0”宏表示 wifi 供电电源；
- 6、“module_power0_vol”宏表示 wifi 供电电源的电压值；
- 7、“module_power1”宏作用与 module_power0 宏一样；
- 8、“module_power1_vol”宏作用与 module_power0_vol 宏一样；
- 9、“module_power2”宏表示 wifi IO 口上拉电阻所接的电压源；
- 10、“module_power2_vol”宏表示 wifi IO 口上拉电阻所接的电压值；
- 11、“lpo_use_apclk”宏表示所选择的 32K 时钟源；

注意：

- 1、模组供电电源和使用哪些 pin 做为 ap6181 的控制需要根据具体平台而定；

3.2. wifi wake host

ap6181 支持 wifi 唤醒休眠主控功能，只需要在 sys_config.fex 的[wakeup_src_para]主键中做添加即可，具体修改如下。

```
[wakeup_src_para]
cpu_en           = 0
cpu_freq         = 48
; (cpu:apb:ahb)
pll_ratio        = 0x111
dram_selfresh_en = 1
dram_freq        = 36
wakeup_src0      =
wakeup_src_wl    = port:PL03<4><default><default><0>
bb_wake_ap       = port:PM01<4><default><default><0>
```

说明：

1、“wakeup_src_wl”子键是为 wifi 唤醒休眠主控而添加的；

注：

1、模组使用哪些 pin 做为 ap6181 的控制需要根据具体平台而定；

3.3. ap6181 模组移植相关文件

以下文件是与 ap6181 模组移植相关的，无需再对这些文件作修改，只需了解即可。

一、ap6181 驱动代码

\\linux-3.4\\drivers\\net\\wireless\\bcmdhd

二、电源及 GPIO 控制

lichee\\linux-3.4\\drivers\\misc\\rf_pm\\wifi_pm.c

lichee\\linux-3.4\\drivers\\misc\\rf_pm\\module_pm.c

lichee\\linux-3.4\\drivers\\misc\\rf_pmbt_pm.c

wifi_pm.c 是 ap6181 模组电源和 GPIO 控制的实现文件。

bt_pm.c 是蓝牙电源管理的实现文件。

注：

1、broadcom 的 ap6xxx 系列模组均是使用相同的 bcmdhd 驱动；

4. ap6210

功能：wifi station、softap、wifi direct、bluetooth

接口类型：SDIO、UART

参考方案：eagle_fvd_p1

编译模式：编译成内核模块，修改路径如下

去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择 Device Drivers --->

Network device support --->

Wireless LAN --->

<M> Broadcom 4329/30 wireless cards support

[*] Use Customized firmware path

[*] Use Customized NVRAM path

[*] Use Customized config path

Interrupt type (Out-of-Band Interrupt) --->

进入 Interrupt type (Out-of-Band Interrupt)--->，选择

(X) Out-of-Band Interrupt

4.1. wifi function

4.1.1. BoardConfig.mk

BoardConfig.mk 文件决定 android 加载哪一款 wifi 模组，要配置成使用 ap6210 模组需要把 BoardConfig.mk 文件的相关代码修改成如下。

```
# wifi and bt configuration
```

```
# 1. Wifi Configuration
```

```
#BOARD_WIFI_VENDOR := realtek
```

```
BOARD_WIFI_VENDOR := broadcom
```

```
# 1.1 broadcom wifi support
```

```
ifeq ($(BOARD_WIFI_VENDOR), broadcom)
```

```
BOARD_WPA_SUPPLICANT_DRIVER := NL80211
```

```
WPA_SUPPLICANT_VERSION      := VER_0_8_X
```

```
BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_bcmdhd
```

```
BOARD_HOSTAPD_DRIVER        := NL80211
```

```
BOARD_HOSTAPD_PRIVATE_LIB   := lib_driver_cmd_bcmdhd
```

```
BOARD_WLAN_DEVICE            := bcmdhd
```

```
WIFI_DRIVER_FW_PATH_PARAM    := "/sys/module/bcmdhd/parameters/firmware_path"
```

```
SW_BOARD_USR_WIFI := AP6210
```

```
include hardware/broadcom/wlan/bcmdhd/firmware/firmware-bcm.mk
```

```
endif
```

说明:

- 1、“#”符号起注释作用;
- 2、“#BOARD_WIFI_VENDOR := realtek”指明不使用 realtek 系统 wifi 模组;
- 3、“BOARD_WIFI_VENDOR := broadcom”指明使用 broadcom 系统 wifi 模组;
- 4、“SW_BOARD_USR_WIFI := AP6210”指明使用 AP6210 模块;

4.1.2. init.rc

init.rc 是资源和服务配置相关的文件, 使用 ap6210 wifi 模组需要作如下修改 (部分代码)。

添加以下内容加载 wifi 驱动:

```
#insmod boardcom wifi driver
insmod /system/vendor/modules/bcmdhd.ko
```

添加以下内容, 启动 wpa_supplicant:

```
# broadcom wifi service
# 1 broadcom wifi station and softap
service wpa_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -I/system/etc/wifi/wpa_supplicant_overlay.conf \
    -O/data/misc/wifi/sockets \
    -e/data/misc/wifi/entropy.bin -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot

# 2 broadcom wifi sta p2p concurrent service
service p2p_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -I/system/etc/wifi/wpa_supplicant_overlay.conf \
    -O/data/misc/wifi/sockets -N \
    -ip2p0 -Dnl80211 -c/data/misc/wifi/p2p_supplicant.conf \
    -I/system/etc/wifi/p2p_supplicant_overlay.conf \
    -puse_p2p_group_interface=1 -e/data/misc/wifi/entropy.bin \
    -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot
```

4.1.3. eagle_fvd_p1.mk

添加以下内容, 使得在设置界面显示 wifi 和 wifi 直连功能:

```
PRODUCT_COPY_FILES += \
    frameworks/native/data/etc/android.hardware.wifi.xml:system/etc/permissions/android.hardware.wifi.xml \
```

```
frameworks/native/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct.xml
```

4.1.4. config.xml

在 frameworks\base\core\res\res\values\config.xml 文件里添加以下内容，使得在设置界面显示“网络共享与便携式热点”功能：

```
<string-array translatable="false" name="config_tether_wifi_regexs">
    <item>"wlan0"</item>
</string-array>
```

4.1.5. sys_config.fex

要配置成使用 ap6210 模组需要把 sys_config.fex 文件修改成如下（部分代码）。

```
[mmc1_para]
sdc_used          = 1
sdc_detmode       = 4
sdc_buswidth      = 4
sdc_clk           = port:PG00<2><1><2><default>
sdc_cmd           = port:PG01<2><1><2><default>
sdc_d0            = port:PG02<2><1><2><default>
sdc_d1            = port:PG03<2><1><2><default>
sdc_d2            = port:PG04<2><1><2><default>
sdc_d3            = port:PG05<2><1><2><default>
sdc_det           =
sdc_use_wp        = 0
sdc_wp            =
sdc_isio          = 1
sdc_regulator     = "none"
sdc_power_supply  = "vcc-wifi-io"

;-----
;wifi/bt/fm/gps/nfc modules configuration
;module_num:
;          0- none
;          1- rtl8188eu(wifi)
;          2- rtl8723bs(wifi+bt)
;          3- ap6181(wifi)
;          4- ap6210(wifi+bt)
;          5- ap6330(wifi+bt)
;          6- ap6335(wifi)
;module_power0: axp used by module, "axp22_dldo1" - use dldo1, not use keep it empty
;module_power0_vol: power0 voltage, mv;
;chip_en:       enable chip io
;lpo_use_apclk: "" - not use, "losc_out"- a23/33, "ac10032k1", "ac10032k2", "ac10032k3"- a80
```

```

;-----
[module_para]
module_num          = 4
module_power0       = "axp81x_dcdc1"
module_power0_vol    = 3300000
module_power1       =
module_power1_vol    =
module_power2       =
module_power2_vol    =
module_power3       =
module_power3_vol    =
chip_en             =
lpo_use_apclk       = "ac10032k2"

;-----
;wifi configuration
;wifi_used:         0-not use, 1- use
;wifi_sdc_id:       0- SDC0, 1- SDC1, 2- SDC2, 3- SDC3
;wifi_usbc_id:      0- USB0, 1- USB1, 2- USB2
;wifi_usbc_type:    1- EHCI(speed 2.0), 2- OHCI(speed 1.0)
;wl_reg_on:         wifi function enable io
;wl_host_wake:      wifi device wake-up host
;wl_host_wake_invert: whether wl_host_wake use inverter between ap and module
;                   0: not used, 1: used
;-----
[wifi_para]
wifi_used           = 1
wifi_sdc_id         = 1
wifi_usbc_id        = 1
wifi_usbc_type      = 1
wl_reg_on           = port:PL02<1><default><default><0>
wl_host_wake        = port:PL03<0><default><default><0>
wl_host_wake_invert = 0

```

说明:

- 1、“;”符号起注释作用;
- 2、“wifi_used”宏赋值为 1 表示使用 wifi, 为 0 表示不使用;
- 3、“wifi_sdc_id”宏表示使用哪个 SD 接口连接 SDIO wifi;
- 4、“wifi_num”宏表示选用哪一款 wifi 模组;
- 5、“module_power0”宏表示 wifi 供电电源;
- 6、“module_power0_vol”宏表示 wifi 供电电源的电压值;
- 7、“module_power1”宏作用与 module_power0 宏一样;
- 8、“module_power1_vol”宏作用与 module_power0_vol 宏一样;
- 9、“module_power2”宏表示 wifi IO 口上拉电阻所接的电压源;
- 10、“module_power2_vol”宏表示 wifi IO 口上拉电阻所接的电压值;
- 11、“lpo_use_apclk”宏表示所选择的 32K 时钟源;

注意：

1、模组供电电源和使用哪些 pin 做为 ap6210 的控制需要根据具体平台而定；

4.2. bluetooth function

修改内核配置，把蓝牙休眠功能给选择上。去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig，把内核中把支持蓝牙休眠功能选择上。

```
选择    [*] Networking support --->
        <*> Bluetooth subsystem support--->
            Bluetooth device drivers--->
                <M> Broadcom Bluetooth Low Power Manager Support
```

注意：

这里将它编译成模块。

4.2.1. BoardConfig.mk

修改 BoardConfig.mk 文件，让其支持 ap6210 的蓝牙功能，修改如下。

```
# 2. Bluetooth Configuration
# make sure BOARD_HAVE_BLUETOOTH is true for every bt vendor
BOARD_HAVE_BLUETOOTH := true
BOARD_HAVE_BLUETOOTH_BCM := true
#BOARD_HAVE_BLUETOOTH_RTK := true
#BLUETOOTH_HCI_USE_RTK_H5 := true
SW_BOARD_HAVE_BLUETOOTH_NAME := ap6210
BOARD_BLUETOOTH_BDROID_BUILDCFG_INCLUDE_DIR :=
device/softwinner/eagle-tvd-perf3/bluetooth/
```

4.2.2. init.rc

修改 init.rc 文件，设置蓝牙属性信息，修改如下。

```
on boot
    insmod /system/vendor/modules/bcm_btlpm.ko

# bcm bluetooth
# uart device
chmod 660 /dev/ttyS1
chown bluetooth net_bt_stack /dev/ttyS1
mkdir /data/misc/bluedroid 770 bluetooth net_bt_stack

# power up/down interface
chmod 0660 /sys/class/rfkill/rfkill0/state
chmod 0660 /sys/class/rfkill/rfkill0/type
chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/state
chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/type
#write /sys/class/rfkill/rfkill0/state 0
```



```
# bluetooth MAC address programming
chown bluetooth net_bt_stack ro.bt.bdaddr_path
chown bluetooth net_bt_stack /system/etc/bluetooth
chown bluetooth net_bt_stack /data/misc/bluetooth
setprop ro.bt.bdaddr_path "/data/misc/bluetooth/bdaddr"

# bluetooth LPM
chmod 0220 /proc/bluetooth/sleep/lpm
chmod 0220 /proc/bluetooth/sleep/btwrite
chown bluetooth net_bt_stack /proc/bluetooth/sleep/lpm
chown bluetooth net_bt_stack /proc/bluetooth/sleep/btwrite
```

4.2.3. eagle_fvd_p1.mk

增加以下内容，使得在设置界面显示蓝牙选项。

```
PRODUCT_COPY_FILES+=
frameworks/native/data/etc/android.hardware.bluetooth.xml:system/etc/permissions/android.hardware.bluetooth.x
ml\
frameworks/native/data/etc/android.hardware.bluetooth_le.xml:system/etc/permissions/android.hardware.bluetooth
le.xml
```

增加以下内容，以便编译 bluetooth.apk

```
# build Bluetooth.apk
PRODUCT_PACKAGES += \
    Bluetooth
```

4.2.4. libbt config

在 device\softwinner\eagle-fvd-p1\bluetooth\目录下添加 libbt 文件夹，用于对 libbt 进行配置，libbt 文件夹包括三个文件：libbt\include\vnd_\$(TARGET_DEVICE).txt（在 eagle-tvd-perf3 方案中，该文件名是 vnd_eagle-tvd-perf3.txt）、libbt\conf\libbt_conf.mk、libbt\conf\bt_vendor.conf；

其中，vnd_eagle-tvd-perf3.txt（注意：该文件需要根据方案重新命名，格式为：vnd_\$(TARGET_DEVICE).txt）的内容如下：

```
BLUETOOTH_UART_DEVICE_PORT = "/dev/ttyS1"
FW_PATCHFILE_LOCATION = "/system/vendor/modules/"
LPM_IDLE_TIMEOUT_MULTIPLE = 5
UART_TARGET_BAUD_RATE = 1500000
BT_WAKE_VIA_PROC = TRUE
#LPM_SLEEP_MODE = FALSE
BTVND_DBG = TRUE
BTHW_DBG = TRUE
VNDUSERIAL_DBG = TRUE
UPIO_DBG = TRUE
```

```

SCO_PCM_ROUTING=0x00
SCO_PCM_IF_CLOCK_RATE=0x00
SCO_PCM_IF_FRAME_TYPE=0x00
SCO_PCM_IF_SYNC_MODE=0x01
SCO_PCM_IF_CLOCK_MODE=0x01
PCM_DATA_FMT_SHIFT_MODE=0x00
PCM_DATA_FMT_FILL_BITS=0x00
PCM_DATA_FMT_FILL_METHOD=0x00
PCM_DATA_FMT_FILL_NUM=0x00
PCM_DATA_FMT_JUSTIFY_MODE=0x00

```

说明:

- 1、BLUETOOTH_UART_DEVICE_PORT 表示蓝牙所使用的 uart 设备号，根据实际情况配置；
- 2、UART_TARGET_BAUD_RATE 表示蓝牙使用的 uart 的波特率，根据实际情况配置；

bt_vendor.conf 的内容如下:

```
# UART device port where Bluetooth controller is attached
```

```
UartPort = /dev/ttyS1
```

```
# Firmware patch file location
```

```
FwPatchFilePath = /system/vendor/modules/
```

```
# Firmware Name
```

```
FwPatchFileName = bcm20710a1.hcd
```

说明:

- 1、UartPort 表示蓝牙所使用的 uart 设备号，根据实际情况配置；
- 2、FwPatchFileName 表示 ap6210 所使用的 Firmware 文件是 bcm20710a1.hcd；

4.2.5. sys_config.fex

修改 sys_config.fex 文件，把使用蓝牙选项打开并且打开 uart1。

```

[uart1]
uart_used      = 1
uart_port      = 1
uart_type      = 4
uart_tx        = port:PG06<2><1><default><default>
uart_rx        = port:PG07<2><1><default><default>
uart_rts       = port:PG08<2><1><default><default>
uart_cts       = port:PG09<2><1><default><default>
uart_regulator = "vcc-io"

;-----
;bluetooth configuration
;bt_used:      0- no used, 1- used
;bt_uart_id:   0- uart0, 1- uart1, 2- uart2
;bt_rst_n:     bt function enable io

```

```

;bt_wake:      host wake-up bluetooth device
;bt_wak_host:  bt device wake-up host
;bt_host_wake_invert: whether bt_host_wake use inverter between ap and module
;
;              0: not used, 1: used
;-----
[bt_para]
bt_used          = 1
bt_uart_id       = 1
bt_rst_n         = port:PL04<1><default><default><0>
bt_wake          = port:PL06<1><default><default><0>
bt_host_wake     = port:PL05<0><default><default><0>
bt_host_wake_invert = 0

```

4.2.6. 蓝牙默认名字的修改

修改的文件路径: device\softwinner\eagle-fvd-p1\bluetooth\bdroid_buildcfg.h

```
#define BTM_DEF_LOCAL_NAME "eagle-p1"
```

4.2.7. config.xml

如需支持蓝牙共享网络功能, 修改 frameworks\base\core\res\res\values\config.xml 如下:

```

<string-array translatable="false" name="config_tether_bluetooth_regexs">
    <item>"bt-pan"</item>
</string-array>

```

4.3. wifi&bt wake host

ap6210 支持 wifi、蓝牙唤醒休眠主控功能, 只需要在 sys_config.fex 的[wakeup_src_para]主键中做添加即可, 具体修改如下。

```

[wakeup_src_para]
cpu_en          = 0
cpu_freq        = 48
; (cpu:apb:ahb)
pll_ratio       = 0x111
dram_selfresh_en = 1
dram_freq       = 36
wakeup_src0     =
wakeup_src_wl   = port:PL03<4><default><default><0>
wakeup_src_bt   = port:PL04<4><default><default><0>
bb_wake_ap      = port:PM01<4><default><default><0>

```

说明:

1、“wakeup_src_bt”、“wakeup_src_wl”两个子键是为蓝牙、wifi 唤醒主体睡眠主控而添加的;

注:

1、模组使用哪些 pin 做为 ap6210 的控制需要根据具体平台而定;

4.4. ap6210 模组移植相关文件

以下文件是与 ap6210 模组移植相关的，无需再对这些文件作修改，只需了解即可。

一、ap6210 驱动代码

\linux-3.4\drivers\net\wireless\bcmdhd

二、电源及 GPIO 控制

lichee\linux-3.4\drivers\misc\rf_pm\wifi_pm.c

lichee\linux-3.4\drivers\misc\rf_pm\module_pm.c

lichee\linux-3.4\drivers\misc\rf_pmbt_pm.c

wifi_pm.c 是 ap6210 模组电源和 GPIO 控制的实现文件。

bt_pm.c 是蓝牙电源管理的实现文件。

注：

1、broadcom 的 ap6xxx 系列模组均是使用相同的 bcmdhd 驱动；

Confidential

5. rtl8188etv

功能：wifi station、softap、wifi direct

接口类型：USB

参考方案：dolphin-fvd-p1

编译模式：编译成内核模块，修改路径如下

去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择 Device Drivers --->

Network device support --->

Wireless LAN --->

<M> Realtek 8188E USB Wifi

5.1. BoardConfig.mk

BoardConfig.mk 文件决定 android 加载哪一款 wifi 模组，要配置成使用 rtl8188eu 模组需要把 BoardConfig.mk 文件的相关代码修改成如下。

```
# wifi and bt configuration
# 1. Wifi Configuration
# 1.1 realtek wifi support
BOARD_WIFI_VENDOR := realtek
ifeq ($(BOARD_WIFI_VENDOR), realtek)
    WPA_SUPPLICANT_VERSION := VER_0_8_X
    BOARD_WPA_SUPPLICANT_DRIVER := NL80211
    BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_rtl
    BOARD_HOSTAPD_DRIVER := NL80211
    BOARD_HOSTAPD_PRIVATE_LIB := lib_driver_cmd_rtl

    SW_BOARD_USR_WIFI := rtl8188eu
    BOARD_WLAN_DEVICE := rtl8188eu
endif

# 1.2 broadcom wifi support
#BOARD_WIFI_VENDOR := broadcom
ifeq ($(BOARD_WIFI_VENDOR), broadcom)
    .....
endif
```

说明：

- 1、“#”符号起注释作用；
- 2、“BOARD_WIFI_VENDOR := realtek”指明使用 realtek 系统 wifi 模组；
- 3、“#BOARD_WIFI_VENDOR := broadcom”指明不使用 broadcom 系统 wifi 模组；
- 4、“SW_BOARD_USR_WIFI := rtl8188eu”、“BOARD_WLAN_DEVICE := rtl8188eu”宏指明使用 rtl8188eu 模组；

5.2. init.rc

init.rc 是资源和服务配置相关的文件，使用 rtl8188etv wifi 模组需要作如下修改（部分代码）。

```
# 1. realtek wifi service
# 1.1 realtek wifi sta service
service wpa_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -O/data/misc/wifi/sockets \
    -e/data/misc/wifi/entropy.bin -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot

# 1.2 realtek wifi sta p2p concurrent service
service p2p_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -ip2p0 -Dnl80211 -c/data/misc/wifi/p2p_supplicant.conf \
    -e/data/misc/wifi/entropy.bin -N \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -O/data/misc/wifi/sockets \
    -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot
```

注意：

1、init.rc 文件指定开启 wpa_supplicant 和 p2p_supplicant 两个服务；

5.3. dolphin_fvd_p1.mk

dolphin_fvd_p1.mk 文件会在编译时拷贝指定文件到特定目录下，使用 rtl8188eu 需要做如下修改。

```
# wifi & bt config file
PRODUCT_COPY_FILES += \
frameworks/native/data/etc/android.hardware.wifi.xml:system/etc/permissions/android.hardware.wifi.xml \
frameworks/native/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct.xml
```

注意：

- 1、以上是把 wifi direct 配置文件 android.hardware.wifi.direct.xml 拷贝到指定目录；
- 2、android.hardware.wifi.direct.xml 是在 wifi 设备中显示 wifi direct 选项；

5.4. config.xml

config.xml 文件路径：frameworks\base\core\res\res\values\config.xml

rtl8188eu 驱动代码是支持 softap 功能，需要作配置才能在设置界面显示 softap 功能，具体的配置是在 config.xml 中实现，修改的部份代码如下。

```
<!-- List of regexpressions describing the interface (if any) that represent tetherable
      Wifi interfaces.  If the device doesn't want to support tethering over Wifi this
      should be empty.  An example would be "softap.*" -->
<!--  default: disable Softap feature -->
<string-array translatable="false" name="config_tether_wifi_regexs">
  <item>"wlan0"</item>
</string-array>

<!-- List of regexpressions describing the interface (if any) that represent tetherable
```

5.5. sys_config.fex

sys_config.fex 文件决定 GPIO pin 的分配，要配置成使用 rtl8188eu 模组需要把 sys_config.fex 文件修改成如下（部分代码）。

```
;-----
;---      USB3 控制标志
;-----
[usbc3]
usb_used          = 1
usb_drv_vbus_gpio =
usb_restrict_gpio =
usb_host_init_state = 0
usb_restric_flag   = 0
usb_regulator_io   = "nocare"
usb_regulator_vol  = 0
usb_not_suspend    = 0

[module_para]
module_num         = 1
module_power0      =
module_power0_vol  = 0
module_power1      =
module_power1_vol  =
module_power2      =
module_power2_vol  =
module_power3      =
module_power3_vol  =
chip_en            =

;-----
;wifi configuration
;wifi_sdc_id:      0- SDC0, 1- SDC1, 2- SDC2, 3- SDC3
```

```

;wifi_usbc_id:    0- USB0, 1- USB1, 2- USB2
;wifi_usbc_type: 1- EHCI(speed 2.0), 2- OHCI(speed 1.0)
;wifi_mod_sel:   0- none, 1- ap6181, 2- ap6210(wifi+bt),
;                3 - rtl8188eu, 4- rtl8723au(wifi+bt),
;                5 - rtl8723bs, 6- esp8089,
;                7 - ap6476
;-----
[wifi_para]
wifi_used          = 1
wifi_sdc_id        = 1
wifi_usbc_id       = 3
wifi_usbc_type     = 1
wl_reg_on          = port:PL07<1><default><default><1>
wl_host_wake       =
wl_host_wake_invert = 0

```

说明:

- 1、“;”符号起注释作用;
- 2、“wifi_used”宏赋值为1表示使用wifi,为0表示不使用;
- 3、“wifi_usbc_id”宏表示使用哪个usb接口连接usb wifi;
- 4、“wifi_usbc_type”宏表示使哪种接口的usb;
- 5、“module_num”宏表示选用哪一款wifi模组;
- 6、“usb_host_init_state”宏设成0是wifi模组的电源由wifi控制,设成1wifi模组的电源将由usb控制;
- 7、“module_power0”宏表示wifi供电电源;
- 8、“module_power0_vol”宏表示wifi供电电源的电压值;
- 9、“module_power1”宏作用与module_power0宏一样;
- 10、“module_power1_vol”宏作用与module_power0_vol宏一样;
- 11、“module_power2”宏表示wifi IO口上拉电阻所接的电压源;
- 12、“module_power2_vol”宏表示wifi IO口上拉电阻所接的电压值;
- 13、“lpo_use_apclk”宏表示所选择的32K时钟源;

5.6. rtl8188etv 模组移植相关文件

以下文件是与rtl8188eu模组移植相关的,无需再对这些文件作修改,只需了解即可。

一、rtl8188eu 驱动代码

\\linux-3.4\\drivers\\net\\wireless\\rtl8188eu

二、电源控制

lichee\\linux-3.4\\drivers\\misc\\rf_pm\\wifi_pm.c

lichee\\linux-3.4\\drivers\\misc\\rf_pm\\module_pm.c

wifi_pm.c 是rtl8188eu模组电源控制的实现文件

6. ap6330

功能：wifi station、softap、wifi direct、bluetooth

接口类型：SDIO、UART

参考方案：jaws-optimus

编译模式：编译成内核模块，修改路径如下

去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择 Device Drivers --->

Network device support --->

Wireless LAN --->

<M> Broadcom 4329/30 wireless cards support

[*] Use Customized firmware path

[*] Use Customized NVRAM path

[*] Use Customized config path

Interrupt type (Out-of-Band Interrupt) --->

进入 Interrupt type (Out-of-Band Interrupt)--->，选择

(X) Out-of-Band Interrupt

6.1. wifi function

6.1.1. BoardConfig.mk

BoardConfig.mk 文件决定 android 加载哪一款 wifi 模组，要配置成使用 ap6330 模组需要把 BoardConfig.mk 文件的相关代码修改成如下。

```
# wifi and bt configuration
```

```
# 1. Wifi Configuration
```

```
#BOARD_WIFI_VENDOR := realtek
```

```
BOARD_WIFI_VENDOR := broadcom
```

```
# 1.1 broadcom wifi support
```

```
ifeq ($(BOARD_WIFI_VENDOR), broadcom)
```

```
BOARD_WPA_SUPPLICANT_DRIVER := NL80211
```

```
WPA_SUPPLICANT_VERSION      := VER_0_8_X
```

```
BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_bcmdhd
```

```
BOARD_HOSTAPD_DRIVER        := NL80211
```

```
BOARD_HOSTAPD_PRIVATE_LIB   := lib_driver_cmd_bcmdhd
```

```
BOARD_WLAN_DEVICE           := bcmdhd
```

```
WIFI_DRIVER_FW_PATH_PARAM    := "/sys/module/bcmdhd/parameters/firmware_path"
```

```
SW_BOARD_USR_WIFI := AP6330
```

```
include hardware/broadcom/wlan/bcmdhd/firmware/firmware-bcm.mk
```

```
endif
```

说明:

- 1、“#”符号起注释作用;
- 2、“#BOARD_WIFI_VENDOR := realtek”指明不使用 realtek 系统 wifi 模组;
- 3、“BOARD_WIFI_VENDOR := broadcom”指明使用 broadcom 系统 wifi 模组;
- 4、“SW_BOARD_USR_WIFI := AP6330”指明使用 AP6330 模块;

6.1.2. init.rc

init.rc 是资源和服务配置相关的文件, 使用 ap6330 wifi 模组需要作如下修改 (部分代码)。

添加以下内容加载 wifi 驱动:

```
#insmod boardcom wifi driver
insmod /system/vendor/modules/bcmdhd.ko
```

添加以下内容, 启动 wpa_supplicant:

```
# broadcom wifi service
# 1 broadcom wifi station and softap
service wpa_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
-iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
-I/system/etc/wifi/wpa_supplicant_overlay.conf \
-O/data/misc/wifi/sockets \
-e/data/misc/wifi/entropy.bin -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot

# 2 broadcom wifi sta p2p concurrent service
service p2p_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
-iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
-I/system/etc/wifi/wpa_supplicant_overlay.conf \
-O/data/misc/wifi/sockets -N \
-ip2p0 -Dnl80211 -c/data/misc/wifi/p2p_supplicant.conf \
-I/system/etc/wifi/p2p_supplicant_overlay.conf \
-puse_p2p_group_interface=1 -e/data/misc/wifi/entropy.bin \
-g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot
```

6.1.3. eagle_fvd_p1.mk

添加以下内容, 使得在设置界面显示 wifi 和 wifi 直连功能:

```
PRODUCT_COPY_FILES += \
frameworks/native/data/etc/android.hardware.wifi.xml:system/etc/permissions/android.hardware.wifi.xml \
```

```
frameworks/native/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct.xml
```

6.1.4. config.xml

在 frameworks\base\core\res\res\values\config.xml 文件里添加以下内容，使得在设置界面显示“网络共享与便携式热点”功能：

```
<string-array translatable="false" name="config_tether_wifi_regexs">
    <item>"wlan0"</item>
</string-array>
```

在 device\softwinner\jaws-optimus\overlay\frameworks\base\core\res\res\values\config.xml 文件里添加以下内容，使得在设置界面显示频段选择功能：

```
<!-- Boolean indicating whether the wifi chipset has dual frequency band support -->
<bool translatable="false" name="config_wifi_dual_band_support">true</bool>
```

6.1.5. sys_config.fex

要配置成使用 ap6330 模组需要把 sys_config.fex 文件修改成如下（部分代码）。

```
[mmc1_para]
sdc_used          = 1
sdc_detmode       = 4
sdc_buswidth      = 4
sdc_clk           = port:PG00<2><1><2><default>
sdc_cmd           = port:PG01<2><1><2><default>
sdc_d0            = port:PG02<2><1><2><default>
sdc_d1            = port:PG03<2><1><2><default>
sdc_d2            = port:PG04<2><1><2><default>
sdc_d3            = port:PG05<2><1><2><default>
sdc_det           =
sdc_use_wp        = 0
sdc_wp            =
sdc_isio          = 1
sdc_regulator     = "none"

;-----
;wifi/bt/fm/gps/nfc modules configuration
;module_num:
;          0- none
;          1- rtl8188eu(wifi)
;          2- rtl8723bs(wifi+bt)
;          3- ap6181(wifi)
;          4- ap6210(wifi+bt)
;          5- ap6330(wifi+bt)
;          6- ap6335(wifi)
```

```

;module_power0: axp used by module, "axp22_dldo1" - use dldo1, not use keep it empty
;module_power0_vol: power0 voltage, mv;
;chip_en:          enable chip io
;lpo_use_apclk: "" - not use, "lpsc_out"- a23/33, "ac10032k1"、 "ac10032k2"、 "ac10032k3"- a80
;-----
[module_para]
module_num          = 5
module_power0       = "axp22_dldo1"
module_power0_vol    = 3300000
module_power1       = "axp15_cldo3"
module_power1_vol    = 3000000
module_power2       = "axp22_ldoio0"
module_power2_vol    = 3000000
module_power3       =
module_power3_vol    =
chip_en             =
lpo_use_apclk       = "ac10032k2"
;-----
;wifi configuration
;wifi_used:         0-not use, 1- use
;wifi_sdc_id:       0- SDC0, 1- SDC1, 2- SDC2, 3- SDC3
;wifi_usbc_id:      0- USB0, 1- USB1, 2- USB2
;wifi_usbc_type:    1- EHCI(speed 2.0), 2- OHCI(speed 1.0)
;wl_reg_on:         wifi function enable io
;wl_host_wake:      wifi device wake-up host
;wl_host_wake_invert: whether wl_host_wake use inverter between ap and module
;                   0: not used, 1: used
;-----
[wifi_para]
wifi_used           = 1
wifi_sdc_id         = 1
wifi_usbc_id        = 1
wifi_usbc_type      = 1
wl_reg_on           = port:PL02<1><default><default><0>
wl_host_wake        = port:PL03<0><default><default><0>
wl_host_wake_invert = 0

```

说明:

- 1、“;”符号起注释作用;
- 2、“wifi_used”宏赋值为1表示使用wifi, 为0表示不使用;
- 3、“wifi_sdc_id”宏表示使用哪个SD接口连接SDIO wifi;
- 4、“wifi_num”宏表示选用哪一款wifi模组;
- 5、“module_power0”宏表示wifi供电电源;
- 6、“module_power0_vol”宏表示wifi供电电源的电压值;
- 7、“module_power1”宏作用与module_power0宏一样;

- 8、“module_power1_vol”宏作用与 module_power0_vol 宏一样；
- 9、“module_power2”宏表示 wifi IO 口上拉电阻所接的电压源；
- 10、“module_power2_vol”宏表示 wifi IO 口上拉电阻所接的电压值；
- 11、“lpo_use_apclk”宏表示所选择的 32K 时钟源；

注意：

- 1、模组供电电源和使用哪些 pin 做为 ap6330 的控制需要根据具体平台而定；

6.2. bluetooth function

修改内核配置，把蓝牙休眠功能给选择上。去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig，把内核中把支持蓝牙休眠功能选择上。

```
选择    [*] Networking support --->
        <*> Bluetooth subsystem support--->
            Bluetooth device drivers--->
                <M> Broadcom Bluetooth Low Power Manager Support
```

注意：

这里将它编译成模块。

6.2.1. BoardConfig.mk

修改 BoardConfig.mk 文件，让其支持 ap6330 的蓝牙功能，修改如下。

```
# 2. Bluetooth Configuration
# make sure BOARD_HAVE_BLUETOOTH is true for every bt vendor
BOARD_HAVE_BLUETOOTH := true
BOARD_HAVE_BLUETOOTH_BCM := true
#BOARD_HAVE_BLUETOOTH_RTK := true
#BLUETOOTH_HCI_USE_RTK_H5 := true
SW_BOARD_HAVE_BLUETOOTH_NAME := ap6330
BOARD_BLUETOOTH_BDROID_BUILDCFG_INCLUDE_DIR := device/softwinner/eagle-fvd-p1/bluetooth/
```

6.2.2. init.rc

修改 init.rc 文件，设置蓝牙属性信息，修改如下。

```
on boot
    insmod /system/vendor/modules/bcm_btlpm.ko

# bcm bluetooth
# uart device
    chmod 660 /dev/ttyS2
    chown bluetooth net_bt_stack /dev/ttyS2
    mkdir /data/misc/bluedroid 770 bluetooth net_bt_stack

# power up/down interface
    chmod 0660 /sys/class/rfkill/rfkill0/state
    chmod 0660 /sys/class/rfkill/rfkill0/type
```

```

chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/state
chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/type
#write /sys/class/rfkill/rfkill0/state 0

# bluetooth MAC address programming
chown bluetooth net_bt_stack ro.bt.bdaddr_path
chown bluetooth net_bt_stack /system/etc/bluetooth
chown bluetooth net_bt_stack /data/misc/bluetooth
setprop ro.bt.bdaddr_path "/data/misc/bluetooth/bdaddr"

# bluetooth LPM
chmod 0220 /proc/bluetooth/sleep/lpm
chmod 0220 /proc/bluetooth/sleep/btwrite
chown bluetooth net_bt_stack /proc/bluetooth/sleep/lpm
chown bluetooth net_bt_stack /proc/bluetooth/sleep/btwrite

```

6.2.3. jaws_optimus.mk

增加以下内容，使得在设置界面显示蓝牙选项。

```

PRODUCT_COPY_FILES+=
frameworks/native/data/etc/android.hardware.bluetooth.xml:system/etc/permissions/android.hardware.bluetooth.x
ml\
frameworks/native/data/etc/android.hardware.bluetooth_le.xml:system/etc/permissions/android.hardware.bluetooth
h_le.xml

```

增加以下内容，以便编译 bluetooth.apk

```

# build Bluetooth.apk
PRODUCT_PACKAGES += \
    Bluetooth

```

6.2.4. libbt config

在 device\softwinner\jaws-optimus\bluetooth\目录下添加 libbt 文件夹，用于对 libbt 进行配置，libbt 文件夹包括三个文件：libbt\include\vnd_\$(TARGET_DEVICE).txt（在 jaws-optimus 方案中，该文件名是 vnd_jaws-optimus.txt）、libbt\conf\libbt_conf.mk、libbt\conf\bt_vendor.conf；

其中，vnd_jaws-optimus.txt（注意：该文件需要根据方案重新命名，格式为：vnd_\$(TARGET_DEVICE).txt）的内容如下：

```

BLUETOOTH_UART_DEVICE_PORT = "/dev/ttyS2"
FW_PATCHFILE_LOCATION = "/system/vendor/modules/"
LPM_IDLE_TIMEOUT_MULTIPLE = 5
UART_TARGET_BAUD_RATE = 4000000
BT_WAKE_VIA_PROC = TRUE
#LPM_SLEEP_MODE = FALSE
BTVND_DBG = FALSE

```

```

BTHW_DBG = TRUE
VNDUSERIAL_DBG = FALSE
UPIO_DBG = FALSE
SCO_PCM_ROUTING=0x00
SCO_PCM_IF_CLOCK_RATE=0x00
SCO_PCM_IF_FRAME_TYPE=0x00
SCO_PCM_IF_SYNC_MODE=0x01
SCO_PCM_IF_CLOCK_MODE=0x01
PCM_DATA_FMT_SHIFT_MODE=0x00
PCM_DATA_FMT_FILL_BITS=0x00
PCM_DATA_FMT_FILL_METHOD=0x00
PCM_DATA_FMT_FILL_NUM=0x00
PCM_DATA_FMT_JUSTIFY_MODE=0x00

```

说明:

- 1、BLUETOOTH_UART_DEVICE_PORT 表示蓝牙所使用的 uart 设备号，根据实际情况配置；
- 2、UART_TARGET_BAUD_RATE 表示蓝牙使用的 uart 的波特率，根据实际情况配置；

bt_vendor.conf 的内容如下:

```
# UART device port where Bluetooth controller is attached
```

```
UartPort = /dev/ttyS2
```

```
# Firmware patch file location
```

```
FwPatchFilePath = /system/vendor/modules/
```

```
# Firmware Name
```

```
FwPatchFileName = bcm40183b2.hcd
```

说明:

- 1、UartPort 表示蓝牙所使用的 uart 设备号，根据实际情况配置；
- 2、FwPatchFileName 表示 ap6330 所使用的 Firmware 文件是 bcm40183b2.hcd;

6.2.5. sys_config.fex

修改 sys_config.fex 文件，把使用蓝牙选项打开并且打开 uart2。

```

[uart2]
uart_used      = 1
uart_port      = 2
uart_type      = 4
uart_tx        = port:PG06<2><1><default><default>
uart_rx        = port:PG07<2><1><default><default>
uart_rts       = port:PG08<2><1><default><default>
uart_cts       = port:PG09<2><1><default><default>

;-----
;bluetooth configuration
;bt_used:      0- no used, 1- used

```

```

;bt_uart_id:      0- uart0, 1- uart1, 2- uart2
;bt_rst_n:        bt function enable io
;bt_wake:         host wake-up bluetooth device
;bt_wak_host:     bt device wake-up host
;bt_host_wake_invert: whether bt_host_wake use inverter between ap and module
;
;                0: not used, 1: used
;-----
[bt_para]
bt_used           = 1
bt_uart_id       = 2
bt_rst_n          = port:PL05<1><default><default><0>
bt_wake           = port:PL08<1><default><default><0>
bt_host_wake      = port:PL04<0><default><default><0>
bt_host_wake_invert = 0

```

6.2.6. 蓝牙默认名字的修改

修改的文件路径: device\softwinner\eagle-fvd-p1\bluetooth\bdroid_buildcfg.h

```
#define BTM_DEF_LOCAL_NAME "eagle-p1"
```

6.2.7. config.xml

如需支持蓝牙共享网络功能, 修改 frameworks\base\core\res\res\values\config.xml 如下:

```

<string-array translatable="false" name="config_tether_bluetooth_regexs">
    <item>"bt-pan"</item>
</string-array>

```

6.3. wifi&bt wake host

ap6330 支持 wifi、蓝牙唤醒休眠主控功能, 只需要在 sys_config.fex 的[wakeup_src_para]主键中做添加即可, 具体修改如下。

```

[wakeup_src_para]
cpu_en           = 0
cpu_freq         = 48
; (cpu:apb:ahb)
pll_ratio        = 0x111
dram_selfresh_en = 1
dram_freq        = 36
wakeup_src0      =
wakeup_src_wl    = port:PL03<4><default><default><0>
wakeup_src_bt    = port:PL04<4><default><default><0>
bb_wake_ap       = port:PM01<4><default><default><0>

```

说明:

1、“wakeup_src_bt”、“wakeup_src_wl”两个子键是为蓝牙、wifi 唤醒主休眠主控而添加的;

注:

1、模组使用哪些 pin 做为 ap6330 的控制需要根据具体平台而定；

6.4. ap6330 模组移植相关文件

以下文件是与 ap6330 模组移植相关的，无需再对这些文件作修改，只需了解即可。

一、ap6330 驱动代码

linux-3.4\drivers\net\wireless\bcmdhd

二、电源及 GPIO 控制

lichee\linux-3.4\drivers\misc\rf_pm\wifi_pm.c

lichee\linux-3.4\drivers\misc\rf_pm\module_pm.c

lichee\linux-3.4\drivers\misc\rf_pmbt_pm.c

wifi_pm.c 是 ap6330 模组电源和 GPIO 控制的实现文件。

bt_pm.c 是蓝牙电源管理的实现文件。

注：

1、broadcom 的 ap6xxx 系列模组均是使用相同的 bcmdhd 驱动；

Confidential

7. ap6335

功能：wifi station、softap、wifi direct、bluetooth

接口类型：SDIO、UART

参考方案：jaws-tvd-p1

编译模式：编译成内核模块，修改路径如下

去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择 Device Drivers --->

Network device support --->

Wireless LAN --->

<M> Broadcom 4329/30 wireless cards support

[*] Use Customized firmware path

[*] Use Customized NVRAM path

[*] Use Customized config path

Interrupt type (Out-of-Band Interrupt) --->

进入 Interrupt type (Out-of-Band Interrupt)--->，选择

(X) Out-of-Band Interrupt

7.1. wifi function

7.1.1. BoardConfig.mk

BoardConfig.mk 文件决定 android 加载哪一款 wifi 模组，要配置成使用 ap6335 模组需要把 BoardConfig.mk 文件的相关代码修改成如下。

```
# wifi and bt configuration
```

```
# 1. Wifi Configuration
```

```
#BOARD_WIFI_VENDOR := realtek
```

```
BOARD_WIFI_VENDOR := broadcom
```

```
# 1.1 broadcom wifi support
```

```
ifeq ($(BOARD_WIFI_VENDOR), broadcom)
```

```
BOARD_WPA_SUPPLICANT_DRIVER := NL80211
```

```
WPA_SUPPLICANT_VERSION      := VER_0_8_X
```

```
BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_bcmdhd
```

```
BOARD_HOSTAPD_DRIVER        := NL80211
```

```
BOARD_HOSTAPD_PRIVATE_LIB   := lib_driver_cmd_bcmdhd
```

```
BOARD_WLAN_DEVICE            := bcmdhd
```

```
WIFI_DRIVER_FW_PATH_PARAM    := "/sys/module/bcmdhd/parameters/firmware_path"
```

```
SW_BOARD_USR_WIFI := AP6335
```

```
include hardware/broadcom/wlan/bcmdhd/firmware/firmware-bcm.mk
```

```
endif
```

说明:

- 1、“#”符号起注释作用;
- 2、“#BOARD_WIFI_VENDOR := realtek”指明不使用 realtek 系统 wifi 模组;
- 3、“BOARD_WIFI_VENDOR := broadcom”指明使用 broadcom 系统 wifi 模组;
- 4、“SW_BOARD_USR_WIFI := AP6335”指明使用 AP6335 模块;

7.1.2. init.rc

init.rc 是资源和服务配置相关的文件, 使用 ap6335 wifi 模组需要作如下修改 (部分代码)。

添加以下内容加载 wifi 驱动:

```
#insmod boardcom wifi driver
insmod /system/vendor/modules/bcmdhd.ko
```

添加以下内容, 启动 wpa_supplicant:

```
# broadcom wifi service
# 1 broadcom wifi station and softap
service wpa_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
-iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
-I/system/etc/wifi/wpa_supplicant_overlay.conf \
-O/data/misc/wifi/sockets \
-e/data/misc/wifi/entropy.bin -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot

# 2 broadcom wifi sta p2p concurrent service
service p2p_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
-iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
-I/system/etc/wifi/wpa_supplicant_overlay.conf \
-O/data/misc/wifi/sockets -N \
-ip2p0 -Dnl80211 -c/data/misc/wifi/p2p_supplicant.conf \
-I/system/etc/wifi/p2p_supplicant_overlay.conf \
-puse_p2p_group_interface=1 -e/data/misc/wifi/entropy.bin \
-g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot
```

7.1.3. eagle_fvd_p1.mk

添加以下内容, 使得在设置界面显示 wifi 和 wifi 直连功能:

```
PRODUCT_COPY_FILES += \
frameworks/native/data/etc/android.hardware.wifi.xml:system/etc/permissions/android.hardware.wifi.xml \
```

```
frameworks/native/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct.xml
```

7.1.4. config.xml

在 frameworks\base\core\res\res\values\config.xml 文件里添加以下内容，使得在设置界面显示“网络共享与便携式热点”功能：

```
<string-array translatable="false" name="config_tether_wifi_regexs">
    <item>"wlan0"</item>
</string-array>
```

在 device\softwinner\jaws-optimus\overlay\frameworks\base\core\res\res\values\config.xml 文件里添加以下内容，使得在设置界面显示频段选择功能：

```
<!-- Boolean indicating whether the wifi chipset has dual frequency band support -->
<bool translatable="false" name="config_wifi_dual_band_support">true</bool>
```

7.1.5. sys_config.fex

要配置成使用 ap6335 模组需要把 sys_config.fex 文件修改成如下（部分代码）。

```
[mmc1_para]
sdc_used          = 1
sdc_detmode       = 4
sdc_buswidth      = 4
sdc_clk           = port:PG00<2><1><2><default>
sdc_cmd           = port:PG01<2><1><2><default>
sdc_d0            = port:PG02<2><1><2><default>
sdc_d1            = port:PG03<2><1><2><default>
sdc_d2            = port:PG04<2><1><2><default>
sdc_d3            = port:PG05<2><1><2><default>
sdc_det           =
sdc_use_wp        = 0
sdc_wp            =
sdc_isio          = 1
sdc_regulator     = "none"
```

```
[module_para]
module_num        = 6
module_power0     = "axp15_cldo2"
module_power0_vol = 3300000
module_power1     = "axp15_cldo3"
module_power1_vol = 3300000
module_power2     = "axp15_aldo2"
module_power2_vol = 1800000
module_power3     =
module_power3_vol =
```

```

chip_en                =
lpo_use_apclk          = "ac10032k2"

;-----
;wifi configuration
;wifi_used:            0-not use, 1- use
;wifi_sdc_id:          0- SDC0, 1- SDC1, 2- SDC2, 3- SDC3
;wifi_usbc_id:         0- USB0, 1- USB1, 2- USB2
;wifi_usbc_type:       1- EHCI(speed 2.0), 2- OHCI(speed 1.0)
;wl_reg_on:            wifi function enable io
;wl_host_wake:         wifi device wake-up host
;wl_host_wake_invert:  whether wl_host_wake use inverter between ap and module
;                      0: not used, 1: used
;-----
[wifi_para]
wifi_used              = 1
wifi_sdc_id            = 1
wifi_usbc_id           = 1
wifi_usbc_type         = 1
wl_reg_on              = port:PM00<1><default><default><0>
wl_host_wake           = port:PM01<0><default><default><0>
wl_host_wake_invert    = 0

```

说明:

- 1、“;”符号起注释作用;
- 2、“wifi_used”宏赋值为 1 表示使用 wifi, 为 0 表示不使用;
- 3、“wifi_sdc_id”宏表示使用哪个 SD 接口连接 SDIO wifi;
- 4、“wifi_num”宏表示选用哪一款 wifi 模组;
- 5、“module_power0”宏表示 wifi 供电电源;
- 6、“module_power0_vol”宏表示 wifi 供电电源的电压值;
- 7、“module_power1”宏作用与 module_power0 宏一样;
- 8、“module_power1_vol”宏作用与 module_power0_vol 宏一样;
- 9、“module_power2”宏表示 wifi IO 口上拉电阻所接的电压源;
- 10、“module_power2_vol”宏表示 wifi IO 口上拉电阻所接的电压值;
- 11、“lpo_use_apclk”宏表示所选择的 32K 时钟源;

注意:

- 1、模组供电电源和使用哪些 pin 做为 ap6335 的控制需要根据具体平台而定;

7.2. bluetooth function

修改内核配置, 把蓝牙休眠功能给选择上。去到 linux-3.4 目录下, 输入 make ARCH=arm menuconfig, 把内核中把支持蓝牙休眠功能选择上。

```

选择    [*] Networking support --->
        <*> Bluetooth subsystem support--->
            Bluetooth device drivers--->
                <M> Broadcom Bluetooth Low Power Manager Support

```

注意：

这里将它编译成模块。

7.2.1. BoardConfig.mk

修改 BoardConfig.mk 文件，让其支持 ap6335 的蓝牙功能，修改如下。

```
# 2. Bluetooth Configuration
# make sure BOARD_HAVE_BLUETOOTH is true for every bt vendor
BOARD_HAVE_BLUETOOTH := true
BOARD_HAVE_BLUETOOTH_BCM := true
#BOARD_HAVE_BLUETOOTH_RTK := true
#BLUETOOTH_HCI_USE_RTK_H5 := true
SW_BOARD_HAVE_BLUETOOTH_NAME := ap6335
BOARD_BLUETOOTH_BDROID_BUILDCFG_INCLUDE_DIR := device/softwinner/eagle-fvd-p1/bluetooth/
```

7.2.2. init.rc

修改 init.rc 文件，设置蓝牙属性信息，修改如下。

```
on boot
    insmod /system/vendor/modules/bcm_btlpm.ko

# bcm bluetooth
# uart device
    chmod 660 /dev/ttyS2
    chown bluetooth net_bt_stack /dev/ttyS2
    mkdir /data/misc/bluedroid 770 bluetooth net_bt_stack

# power up/down interface
    chmod 0660 /sys/class/rfkill/rfkill0/state
    chmod 0660 /sys/class/rfkill/rfkill0/type
    chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/state
    chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/type
    #write /sys/class/rfkill/rfkill0/state 0

# bluetooth MAC address programming
    chown bluetooth net_bt_stack ro.bt.bdaddr_path
    chown bluetooth net_bt_stack /system/etc/bluetooth
    chown bluetooth net_bt_stack /data/misc/bluetooth
    setprop ro.bt.bdaddr_path "/data/misc/bluetooth/bdaddr"

# bluetooth LPM
    chmod 0220 /proc/bluetooth/sleep/lpm
    chmod 0220 /proc/bluetooth/sleep/btwrite
    chown bluetooth net_bt_stack /proc/bluetooth/sleep/lpm
    chown bluetooth net_bt_stack /proc/bluetooth/sleep/btwrite
```

7.2.3. jaws_tvd_p1.mk

增加以下内容，使得在设置界面显示蓝牙选项。

```
PRODUCT_COPY_FILES+=
frameworks/native/data/etc/android.hardware.bluetooth.xml:system/etc/permissions/android.hardware.bluetooth.xml\
frameworks/native/data/etc/android.hardware.bluetooth_le.xml:system/etc/permissions/android.hardware.bluetooth_le.xml
```

增加以下内容，以便编译 bluetooth.apk

```
# build Bluetooth.apk
PRODUCT_PACKAGES += \
    Bluetooth
```

7.2.4. libbt config

在 device\softwinner\jaws-tvd-p1\bluetooth\目录下添加 libbt 文件夹，用于对 libbt 进行配置，libbt 文件夹包括三个文件：libbt\include\vnd_\$(TARGET_DEVICE).txt（在 jaws-tvd-p1 方案中，该文件名是 vnd_jaws-tvd-p1.txt）、libbt\conf\libbt_conf.mk、libbt\conf\bt_vendor.conf；

其中，vnd_jaws-tvd-p1.txt（注意：该文件需要根据方案重新命名，格式为：vnd_\$(TARGET_DEVICE).txt）的内容如下：

```
BLUETOOTH_UART_DEVICE_PORT = "/dev/ttyS2"
FW_PATCHFILE_LOCATION = "/system/vendor/modules/"
LPM_IDLE_TIMEOUT_MULTIPLE = 5
UART_TARGET_BAUD_RATE = 4000000
BT_WAKE_VIA_PROC = TRUE
#LPM_SLEEP_MODE = FALSE
BTVND_DBG = FALSE
BTHW_DBG = TRUE
VNDUSERIAL_DBG = FALSE
UPIO_DBG = FALSE
SCO_PCM_ROUTING=0x00
SCO_PCM_IF_CLOCK_RATE=0x00
SCO_PCM_IF_FRAME_TYPE=0x00
SCO_PCM_IF_SYNC_MODE=0x01
SCO_PCM_IF_CLOCK_MODE=0x01
PCM_DATA_FMT_SHIFT_MODE=0x00
PCM_DATA_FMT_FILL_BITS=0x00
PCM_DATA_FMT_FILL_METHOD=0x00
PCM_DATA_FMT_FILL_NUM=0x00
PCM_DATA_FMT_JUSTIFY_MODE=0x00
```

说明：

- 1、BLUETOOTH_UART_DEVICE_PORT 表示蓝牙所使用的 uart 设备号，根据实际情况配置；
- 2、UART_TARGET_BAUD_RATE 表示蓝牙使用的 uart 的波特率，根据实际情况配置；

bt_vendor.conf 的内容如下:

```
# UART device port where Bluetooth controller is attached
```

```
UartPort = /dev/ttyS2
```

```
# Firmware patch file location
```

```
FwPatchFilePath = /system/vendor/modules/
```

```
#Firmware name
```

```
FwPatchFileName = bcm4339a0.hcd
```

说明:

- 1、UartPort 表示蓝牙所使用的 uart 设备号，根据实际情况配置;
- 2、FwPatchFileName 表示 ap6335 所使用的 Firmware 文件是 bcm4339a0.hcd;

7.2.5. sys_config.fex

修改 sys_config.fex 文件，把使用蓝牙选项打开并且打开 uart2。

```
[uart2]
uart_used      = 1
uart_port      = 2
uart_type      = 4
uart_tx        = port:PG06<2><1><default><default>
uart_rx        = port:PG07<2><1><default><default>
uart_rts       = port:PG08<2><1><default><default>
uart_cts       = port:PG09<2><1><default><default>

;-----
;bluetooth configuration
;bt_used:      0- no used, 1- used
;bt_uart_id:   0- uart0, 1- uart1, 2- uart2
;bt_rst_n:     bt function enable io
;bt_wake:      host wake-up bluetooth device
;bt_wak_host:  bt device wake-up host
;bt_host_wake_invert: whether bt_host_wake use inverter between ap and module
;
;              0: not used, 1: used
;-----

[bt_para]
bt_used        = 1
bt_uart_id     = 2
bt_rst_n       = port:PM03<1><default><default><0>
bt_wake        = port:PM04<1><default><default><0>
bt_host_wake   = port:PM02<0><default><default><0>
bt_host_wake_invert = 0
```


7.2.6. 蓝牙默认名字的修改

修改的文件路径: device\softwinner\eagle-fvd-p1\bluetooth\bdroid_buildcfg.h

```
#define BTM_DEF_LOCAL_NAME "jaws-p1"
```

7.2.7. config.xml

如需支持蓝牙共享网络功能, 修改 frameworks\base\core\res\res\values\config.xml 如下:

```
<string-array translatable="false" name="config_tether_bluetooth_regexs">
    <item>"bt-pan"</item>
</string-array>
```

7.3. wifi&bt wake host

ap6335 支持 wifi、蓝牙唤醒休眠主控功能, 只需要在 sys_config.fex 的[wakeup_src_para]主键中做添加即可, 具体修改如下。

```
[wakeup_src_para]
cpu_en            = 0
cpu_freq          = 48
; (cpu:apb:ahb)
pll_ratio         = 0x111
dram_selfresh_en  = 1
dram_freq         = 36
wakeup_src0       =
wakeup_src_wl     = port:PM01<4><default><default><0>
wakeup_src_bt     = port:PM02<4><default><default><0>
```

说明:

1、“wakeup_src_bt”、“wakeup_src_wl”两个子键是为蓝牙、wifi 唤醒主体睡眠主控而添加的;

注:

1、模组使用哪些 pin 做为 ap6335 的控制需要根据具体平台而定;

7.4. ap6335 模组移植相关文件

以下文件是与 ap6335 模组移植相关的, 无需再对这些文件作修改, 只需了解即可。

一、ap6335 驱动代码

linux-3.4\drivers\net\wireless\bcmdhd

二、电源及 GPIO 控制

lichee\linux-3.4\drivers\misc\rf_pm\wifi_pm.c

lichee\linux-3.4\drivers\misc\rf_pm\module_pm.c

lichee\linux-3.4\drivers\misc\rf_pmbt_pm.c

wifi_pm.c 是 ap6335 模组电源和 GPIO 控制的实现文件。

bt_pm.c 是蓝牙电源管理的实现文件。

注:

1、broadcom 的 ap6xxx 系列模组均是使用相同的 bcmdhd 驱动;

8. rtl8189etv

功能：wifi station、softap、wifi direct

接口类型：SDIO

参考方案：eagle-fvd-p1

编译模式：编译成内核模块，修改路径如下

rtl8189etv 和 rtl8189es 使用相同的驱动，去到 linux-3.4 目录下，输入 make ARCH=arm menuconfig

然后选择 Device Drivers --->

Network device support --->

Wireless LAN --->

<M> Realtek 8189E SDIO Wifi

8.1. BoardConfig.mk

BoardConfig.mk 文件决定 android 加载哪一款 wifi 模组，要配置成使用 rtl8189etv 模组需要把 BoardConfig.mk 文件的相关代码修改成如下。

```
# wifi and bt configuration

# 1. Wifi Configuration
BOARD_WIFI_VENDOR := realtek
#BOARD_WIFI_VENDOR := broadcom

# 1.1 realtek wifi support
ifeq ($(BOARD_WIFI_VENDOR), realtek)
    WPA_SUPPLICANT_VERSION := VER_0_8_X
    BOARD_WPA_SUPPLICANT_DRIVER := NL80211
    BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_rtl
    BOARD_HOSTAPD_DRIVER := NL80211
    BOARD_HOSTAPD_PRIVATE_LIB := lib_driver_cmd_rtl

    SW_BOARD_USR_WIFI := rtl8189es
    BOARD_WLAN_DEVICE := rtl8189es
endif

# 1.2 broadcom wifi support
#BOARD_WIFI_VENDOR := broadcom
ifeq ($(BOARD_WIFI_VENDOR), broadcom)
    .....
endif
```

说明：

- 1、“#” 符号起注释作用；
- 2、“BOARD_WIFI_VENDOR := realtek” 指明使用 realtek 系统 wifi 模组；
- 3、“#BOARD_WIFI_VENDOR := broadcom” 指明不使用 broadcom 系统 wifi 模组；
- 4、“SW_BOARD_USR_WIFI := rtl8189es”、“BOARD_WLAN_DEVICE := rtl8189es”

宏指明使用 rtl8189es 模组；

8.2. init.rc

init.rc 是资源和服务配置相关的文件，使用 rtl8189etv wifi 模组需要作如下修改（部分代码）。

```
# 1. realtek wifi service
# 1.1 realtek wifi sta service
service wpa_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -O/data/misc/wifi/sockets \
    -e/data/misc/wifi/entropy.bin -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot

# 1.2 realtek wifi sta p2p concurrent service
service p2p_supplicant /system/bin/logwrapper /system/bin/wpa_supplicant \
    -ip2p0 -Dnl80211 -c/data/misc/wifi/p2p_supplicant.conf \
    -e/data/misc/wifi/entropy.bin -N \
    -iwlan0 -Dnl80211 -c/data/misc/wifi/wpa_supplicant.conf \
    -O/data/misc/wifi/sockets \
    -g@android:wpa_wlan0
class main
socket wpa_wlan0 dgram 660 wifi wifi
disabled
oneshot
```

注意：

1、init.rc 文件指定开启 wpa_supplicant 和 p2p_supplicant 两个服务；

8.3. eagle_fvd_p1.mk

eagle_fvd_p1.mk 文件会在编译时拷贝指定文件到特定目录下，使用 rtl8189etv 需要做如下修改。

```
# wifi & bt config file
PRODUCT_COPY_FILES += \
frameworks/native/data/etc/android.hardware.wifi.xml:system/etc/permissions/android.hardware.wifi.xml \
frameworks/native/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct.xml
```

注意：

- 1、以上是把 wifi direct 配置文件 android.hardware.wifi.direct.xml 拷贝到指定目录；
- 2、android.hardware.wifi.direct.xml 是在 wifi 设备中显示 wifi direct 选项；

8.4. config.xml

config.xml 文件路径：frameworks\base\core\res\res\values\config.xml

rtl8189etv 驱动代码是支持 softap 功能，需要作配置才能在设置界面显示 softap 功能，具体的配置是在 config.xml 中实现，修改的部份代码如下。

```
<!-- List of regexpressions describing the interface (if any) that represent tetherable
      Wifi interfaces.  If the device doesn't want to support tethering over Wifi this
      should be empty.  An example would be "softap.*" -->
<!--  default: disable Softap feature -->
      <string-array translatable="false" name="config_tether_wifi_regexs">
        <item>"wlan0"</item>
      </string-array>

<!-- List of regexpressions describing the interface (if any) that represent tetherable
```

8.5. sys_config.fex

tools\pack\chips\sun8iw6p1\configs\eagle-p1\sys_config.fex 文件决定 GPIO pin 的分配，要配置成使用 rtl8189etv 模组需要把 sys_config.fex 文件修改成如下（部分代码）。

```
[mmc1_para]
sdc_used          = 1
sdc_detmode       = 4
sdc_buswidth      = 4
sdc_clk           = port:PG00<2><1><2><default>
sdc_cmd           = port:PG01<2><1><2><default>
sdc_d0            = port:PG02<2><1><2><default>
sdc_d1            = port:PG03<2><1><2><default>
sdc_d2            = port:PG04<2><1><2><default>
sdc_d3            = port:PG05<2><1><2><default>
sdc_det           =
sdc_use_wp        = 0
sdc_wp            =
sdc_isio          = 1
sdc_regulator     = "none"
sdc_power_supply  = "vcc-wifi-io"

[module_para]
module_num        = 7
module_power0     = "vcc-wifi-33"
module_power0_vol = 0
module_power1     =
module_power1_vol =
module_power2     =
module_power2_vol =
module_power3     =
module_power3_vol =
chip_en           =
```

```

;-----
;wifi configuration
;wifi_used:      0-not use, 1- use
;wifi_sdc_id:    0- SDC0, 1- SDC1, 2- SDC2, 3- SDC3
;wifi_usbc_id:   0- USB0, 1- USB1, 2- USB2
;wifi_usbc_type: 1- EHCI(speed 2.0), 2- OHCI(speed 1.0)
;wl_reg_on:      wifi function enable io
;wl_host_wake:   wifi device wake-up host
;wl_host_wake_invert: whether wl_host_wake use inverter between ap and module
;
;               0: not used, 1: used
;-----

[wifi_para]
wifi_used          = 1
wifi_sdc_id        = 1
wifi_usbc_id       = 0
wifi_usbc_type     = 1
wl_reg_on          = port:PL02<1><default><default><0>
wl_host_wake       = port:PL03<0><default><default><0>
wl_host_wake_invert = 0

```

说明:

- 1、“;”符号起注释作用;
- 2、“wifi_used”宏赋值为1表示使用wifi, 为0表示不使用;
- 3、“wifi_sdc_id”宏表示使用哪组mmc连接wifi;
- 4、“wifi_usbc_id”宏表示使用哪个usb接口连接usb wifi;
- 5、“wifi_usbc_type”宏表示使用哪种接口的usb;
- 6、“module_num”宏表示选用哪一款wifi模组;
- 7、“module_power0”宏表示wifi供电电源;
- 8、“module_power0_vol”宏表示wifi供电电源的电压值;
- 9、“module_power1”宏作用与module_power0宏一样;
- 10、“module_power1_vol”宏作用与module_power0_vol宏一样;
- 11、“module_power2”宏表示wifi IO口上拉电阻所接的电压源;
- 12、“module_power2_vol”宏表示wifi IO口上拉电阻所接的电压值;
- 13、“wl_reg_on”宏表示wifi模组所使用的使能引脚;
- 14、“wl_host_wake”宏表示wifi模组唤醒主控所使用的引脚;

8.6. wifi wake host

rtl8189etv 支持 wifi 唤醒休眠主控功能, 只需要在 sys_config.fex 的[wakeup_src_para]主键中做添加即可, 具体修改如下。

```

[wakeup_src_para]
cpu_en          = 0
cpu_freq        = 48
;(cpu:apb:ahb)
pll_ratio       = 0x111
dram_selffresh_en = 1

```

dram_freq	= 36
wakeup_src0	=
wakeup_src_wl	= port:PL03<4><default><default><0>

说明:

1、“wakeup_src_wl”子键是为 wifi 唤醒休眠主控而添加的;

注:

1、模组使用哪些 pin 做为 rtl8189etv 的控制需要根据具体平台而定;

8.7. rtl8189etv 模组移植相关文件

以下文件是与 rtl8189etv 模组移植相关的, 无需再对这些文件作修改, 只需了解即可。

一、rtl8189etv 驱动代码

\\linux-3.4\\drivers\\net\\wireless\\rtl8189es

二、电源控制

lichee\\linux-3.4\\drivers\\misc\\rf_pm\\wifi_pm.c

lichee\\linux-3.4\\drivers\\misc\\rf_pm\\module_pm.c

wifi_pm.c 是 rtl8189etv 模组电源控制的实现文件

9. F&Q

9.1. 如何编译

修改配置文件后有两种编译方法可以使得修改生效，第一种是整体编译重烧固件，第二种是单独的模块编译然后把生成的文件 push 进相应的目录，详细操作如下。

整体编译

整体编译有两种方法可选

方法一：在 android 主目录下先 “make clean” 后再 “make -j8”；

方法二：分别到 android4.4.2\hardware\libhardware_legacy\wifi\和 external\wpa_supplicant_8\wpa_supplicant\目录下使用 “touch *” 命令更新目录下所有文件的时间戳，再把 android 主目录下使用 “make -j8” 命令编译；

单独模块编译

先分别到 android4.4.2\hardware\libhardware_legacy\wifi\和 external\wpa_supplicant_8\wpa_supplicant\目录下使用 “touch *” 命令更新目录下所有文件的时间戳，然后在相应目录下使用 “mm” 命令进行编译，再把生的 libhardware_legacy.so 和 wpa_supplicant 推入目标机器目录，最后更改执行权限并重启。

若生成的 libhardware_legacy.so 和 wpa_supplicant 存放在电脑 D 盘下，则用如下命令把其推到目标机器目录下。

```
adb push D:\wap_supplicant /system/bin/wpa_supplicant
adb push D:\libhardware_legacy.so /system/lib/libhardware_legacy.so
adb shell chmod 777 /system/bin/ wpa_supplicant
adb shell chmod 666 /system/lib/libhardware_legacy.so
```

注意：单独模块编译适合调试使用。

9.2. 编译报错

修改配置文件后，如果编译的时候报与 wpa_supplicant_8 相关的错误，可用以下两种方法解决。

方法一：将 out/target/product/方案名/obj/EXECUTABLES/wpa_supplicant_intermediates 下面的中间生成文件删除，再次编译。

方法二：使用方法一后编译还是报错，则需要先使用 make clean 命令清除中间文件和目标文件后，再使用 make -j8 命令进行整体编译。

9.3. 修改不起效

修改配置文件并进行整体编译重烧固件或单独模块编译把生成文件推进目标机器后发现指定的 wifi 模组不能工作，修改配置前那款 wifi 模组还是能正常工作。该问题是因为由于只修改配置文件，wifi 相关的编译文件的时间戳没改变使用不会去重新编译，详细解决方法可参考“如何编译”。

9.4. 蓝牙注意事项

若平台不支持蓝牙，需要确定以下三点以保证不影响 wifi 的正常使用。

- 1、内核中把蓝牙相关的编译去除；
- 2、sys_config.fex 中 bt_used 项应该赋值为 0 以代码不使用蓝牙；
- 3、平台文件 init.rc 文件中以下的蓝牙属性设置是不存在或被注释起来的；

```
# bluetooth
#chmod 660 /dev/ttyS2
#chown bluetooth net_bt_stack /dev/ttyS2
#mkdir /data/misc/bluedroid 770 bluetooth net_bt_stack

# power up/down interface
#chmod 0660 /sys/class/rfkill/rfkill0/state
#chmod 0660 /sys/class/rfkill/rfkill0/type
#chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/state
#chown bluetooth net_bt_stack /sys/class/rfkill/rfkill0/type
#write /sys/class/rfkill/rfkill0/state 0

# bluetooth MAC address programming
#chown bluetooth net_bt_stack ro.bt.bdaddr_path
#chown bluetooth net_bt_stack /system/etc/bluetooth
#chown bluetooth net_bt_stack /data/misc/bluetooth
#setprop ro.bt.bdaddr_path "/data/misc/bluetooth/bdaddr"

# bluetooth LPM
#chmod 0220 /proc/bluetooth/sleep/lpm
#chmod 0220 /proc/bluetooth/sleep/btwrite
#chown bluetooth net_bt_stack /proc/bluetooth/sleep/lpm
#chown bluetooth net_bt_stack /proc/bluetooth/sleep/btwrite
```


10. Declaration

This document is the original work and copyrighted property of Allwinner Technology (“Allwinner”). Reproduction in whole or in part must obtain the written approval of Allwinner and give clear acknowledgement to the copyright owner.

The information furnished by Allwinner is believed to be accurate and reliable. Allwinner reserves the right to make changes in circuit design and/or specifications at any time without notice. Allwinner does not assume any responsibility and liability for its use. Nor for any infringements of patents or other rights of the third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Allwinner. This datasheet neither states nor implies warranty of any kind, including fitness for any particular application.

Confidential