



WiFi 使用指南

文档版本 07
发布日期 2019-05-30

Cogobuy Only For ShenZhen FuShi ChanJing Industrial Technology Co., Ltd.

版权所有 © 上海海思技术有限公司 2019。保留一切权利。

非经本公司书面许可，任何单位和个人不得擅自摘抄、复制本文档内容的部分或全部，并不得以任何形式传播。

商标声明



HISILICON、海思和其他海思商标均为海思技术有限公司的商标。

本文档提及的其他所有商标或注册商标，由各自的所有人拥有。

注意

您购买的产品、服务或特性等应受海思公司商业合同和条款的约束，本文档中描述的全部或部分产品、服务或特性可能不在您的购买或使用范围之内。除非合同另有约定，海思公司对本文档内容不做任何明示或默示的声明或保证。

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、信息和建议不构成任何明示或暗示的担保。

上海海思技术有限公司

地址：深圳市龙岗区坂田华为总部办公楼 邮编：518129

网址：<http://www.hisilicon.com/cn/>

客户服务邮箱：support@hisilicon.com



前言

概述

本文档主要介绍 Wi-Fi 需要使用到的配置，基本操作、调试方法，使用注意事项和常见问题处理。



说明

- 本文未做特殊说明，Hi3516D 与 Hi3516A 完全一致。
- 未做特殊说明，Hi3518EV201、Hi3516CV200 与 Hi3518EV200 完全一致。
- 未做特殊说明，Hi3516EV200、Hi3516EV300、Hi3516DV200、Hi3518EV300 完全一致。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本
Hi3516A	V100
Hi3516D	V100
Hi3518E	V200
Hi3518E	V201
Hi3516C	V200
Hi3559A	V100
Hi3559C	V100
Hi3519A	V100
Hi3516E	V200
Hi3516E	V300
Hi3518E	V300
Hi3559	V200
Hi3556	V200
Hi3516D	V200



读者对象

本文档（本指南）主要适用于以下工程师：

- 技术支持工程师
- 软件开发工程师

修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订日期	版本	修订说明
2019-05-30	08	第八次正式发布 全文删除不支持的 WIFI 型号相关内容
2019-05-15	07	第七次正式发布 Linux 版本增加 RTL8188FTV 章节
2019-02-28	06	第六次正式发布 1、删除 wifi_project 内容 2、增加 Boot 配置 3、增加 RTL8189FTV、AP6181 驱动移植 4、增加 WiFi 工具获取方法
2018-02-10	05	第五次正式发布 添加 Hi3559AV100 和 Hi3559CV100 相关内容 1.2 小节涉及修改 3.1.1、3.1.2、3.2.4 和 3.3.1 小节涉及修改
2018-01-15	04	第四次正式发布，删除 Hi3518EV20X 相关内容 每个章节均涉及修改，并新增第 4 章“测试”。
2015-09-25	03	添加 Hi3518EV200, Hi3518EV201 和 Hi3516CV200 的相关内容。
2015-02-10	02	3.3.3 节，将原来的步骤 3 和步骤 4 合并。



修订日期	版本	修订说明
2014-12-30	01	添加 Hi3516D 的相关内容。 2.1 节中新增 “配置 Wireless Extension”，删除 “配置 Netlink” 和 “配置 Wireless Extension”，新增“配置 IPv6” 一节。 2.2 删除 “wifi 驱动配置” 和 “wifi 无线管理工具”。 第 3 章 新增 “载入驱动文件”、“载入 firmware 文件”、“卸载驱动” 和 “适配国家或区域” 等小节。
2014-11-09	00B01	第 1 次临时版本发布。

Cogobuy Only For ShenZhen FuShi ChanJing Industrial Technology Co., Ltd.



目 录

前 言.....	i
1 概述.....	1
1.1 背景	1
1.2 范围	1
2 配置说明.....	3
2.1 内核配置	3
2.1.1 配置 CFG80211	3
2.1.2 配置 Wireless Extension.....	3
2.1.3 配置 USB 和 SDIO	4
2.1.4 SDIO 中断配置	4
2.1.5 配置通用 GPIO	4
2.2 Boot 配置	5
3 驱动移植.....	6
3.1 RTL8189FTV 驱动移植.....	6
3.2 RTL8188FTV 驱动移植.....	7
4 WiFi 工具.....	9
5 WiFi 基本操作.....	10
5.1 载入文件	10
5.1.1 载入驱动文件	10
5.1.2 载入工具	10
5.1.3 wpa_supplicant.conf 文件	10
5.1.4 hostapd.conf 文件	10
5.1.5 udhcpd.conf 文件	11
5.2 WiFi 设备检测	11
5.3 STA 模式基本操作示例	12
5.3.1 加载驱动	12
5.3.2 扫描 AP	12
5.3.3 连接 AP	13
5.3.4 卸载驱动	15



5.4 SoftAP 模式基本操作示例	16
5.4.1 检查 WiFi 设备、加载驱动	16
5.4.2 hostapd 配置和启动 SoftAP	16
5.4.3 开启 udhcpd.....	17
5.4.4 卸载驱动	17
5.5 适配国家或区域	17
6 测试.....	18
6.1 吞吐量测试	18
6.1.1 TCP 发送吞吐量测试	18
6.1.2 TCP 接收吞吐量测试	18
6.1.3 UDP 发送吞吐量测试	18
6.1.4 UDP 接收吞吐量测试	18
6.2 射频指标测试	18

Cogobuy Only For Shenzhen FuShi ChanJing Industrial Technology Co., Ltd.



插图目录

图 2-1 CFG80211 配置	3
图 2-2 Wireless Extension 配置	4
图 2-3 通用 GPIO 配置	5
图 2-4 通用 GPIO 配置 2	5
图 5-1 查看 SDIO 设备	11
图 5-2 查看 USB 设备	11
图 5-3 iwconfig 执行结果	12
图 5-4 扫描 AP 执行结果	13
图 5-5 wpa_cli 扫描 AP 结果	14
图 5-6 连接 AP	15
图 6-1 吞吐量测试组网环境	18
图 6-2 发送吞吐量测试示例	18
图 6-3 接收吞吐量测试示例	18

Cogobuy Only For Shenzhen Fushi ChanJing Industrial Technology Co., Ltd.



1 概述

1.1 背景

WiFi 行业有很多芯片厂家，这些厂家生产了各种型号的 WiFi 芯片，每种 WiFi 芯片都有它自己的驱动，驱动不具有普适性。由于 Linux 应用很普遍，市面上的 WiFi 芯片都有支持 Linux 平台的驱动，使用这些 WiFi 芯片要将对应的驱动移植到平台上。海思平台采用了标准 Linux，能适配所有 Linux 平台的 WiFi 驱动，不需要对平台做修改。但由于 Linux 版本不同，WiFi 对应的驱动版本也不同，需要向 WiFi 芯片厂家或模组厂家获取对应 Linux 版本的 WiFi 驱动进行移植。

对于不同的 WiFi 芯片，移植 WiFi 驱动和对 WiFi 进行操作的方法具有通用性，本文档以常用的几款 WiFi 芯片为例，介绍驱动移植和调试方法，用于适配其它 WiFi 芯片作为参考。

1.2 范围

Hisilicon、Realtek WiFi 芯片在 Camera 领域使用较多，Hisilicon Hi1131S 使用方法可参考 Hi1131S 相关文档，本文档主要以 Realtek RTL8189FTV 和 RTL8188FTV 两种 WiFi 为例进行说明，其它 WiFi 可以参考本文档。本文档只以芯片的一个驱动版本进行示例，实际使用时要从 WiFi 厂家获取最新驱动版本。

通常 WiFi 有四种模式，不同的 WiFi 设备可能支持其中一种或几种模式，四种模式如下：

- SoftAP: access point，一种将无线设备连接到一个网络的设备，可以理解为无线路由。
- STA: station，无线设备客户端，必须连接上一个 AP 才能使用。
- DIRECT: WiFi 直连，也称 P2P 模式。
- CONCURRENT: 同时支持 AP、STA 两种模式。

本文描述了 STA、SoftAP 两种模式的基本操作，DIRECT 的兼容性较差，用户体验不好，CONCURRENT 模式较少使用，因此本文档不对这两种模式进行说明。

WiFi 除基本功能测试外，性能是重要指标，本文档也对吞吐量、射频指标测试方法进行了简单描述。



Cogobuy Only For ShenZhen FuShi ChanJing Industrial Technology Co., Ltd.



2 配置说明

2.1 内核配置

2.1.1 配置 CFG80211

CFG80211 是内核中 WiFi 驱动和用户态进程的标准接口，在 CFG80211 出现之前是 WEXT，现在越来越多的使用 CFG80211，WiFi Direct 功能只有 CFG80211 才支持。

进入 Network support->Wireless，设置 cfg80211 和 mac80211 为 M，如图 2-1 所示。

图2-1 CFG80211 配置

```
--- Wireless
<M> cfg80211 - wireless configuration API
[ ] nl80211 testmode command (NEW)
[ ] enable developer warnings (NEW)
[ ] cfg80211 regulatory debugging (NEW)
[*] enable powersave by default (NEW)
[ ] cfg80211 wireless extensions compatibility (NEW)
<M> Generic IEEE 802.11 Networking Stack (mac80211)
Default rate control algorithm (Minstrel) --->
[ ] Enable mac80211 mesh networking (pre-802.11s) support (NEW)
[ ] Trace all mac80211 debug messages (NEW)
[ ] Select mac80211 debugging features (NEW) ----
```

2.1.2 配置 Wireless Extension

- WEXT 是内核中 WiFi 驱动和用户态进程的标准接口，调试工具 iwconfig、iwlist、iwpriv 需要使用该接口。如果没有配置该接口，有些驱动会有编译错误。
- WEXT 在内核配置中没有单独的配置项，只能通过打开依赖它的配置项来间接打开。配置了 CFG80211 后，进入 Device Drivers->Network device support->Wireless LAN，设置 USB ZD1201 based Wireless device support 为 M。如图 2-2 所示。



图2-2 Wireless Extension 配置

```
--- Wireless LAN
< > Marvell 8xxx Libertas WLAN driver support with thin firmware
< > Atmel at76c503/at76c505/at76c505a USB cards
< > USB ZD1201 based Wireless device support
< > Wireless RNDIS USB support
< > Realtek 8187 and 8187B USB support
< > Simulated radio testing tool for mac80211
[ ] Enable WiFi control function abstraction
< > Atheros Wireless Cards --->
< > Broadcom 43xx wireless support (mac80211 stack)
< > Broadcom 43xx-legacy wireless support (mac80211 stack)
< > Broadcom 4329/30 wireless cards support
< > Broadcom IEEE802.11n embedded FullMAC WLAN driver
< > IEEE 802.11 for Host AP (Prism2/2.5/3 and WEP/TKIP/CCMP)
< > Intel Wireless Multicom 3200 WiFi driver
< > Marvell 8xxx Libertas WLAN driver support
< > Softmac Prism54 support
< > Ralink driver support --->
< > Realtek RTL8192CU/RTL8188CU USB Wireless Network Adapter
< > TI wl1251 driver support --->
< > TI wl12xx driver support --->
< > ZyDAS ZD1211/ZD1211B USB-wireless support
< > Marvell WiFi-Ex Driver
```

如果找不到这项配置，需要先配置好 USB 再来配置该选项。

2.1.3 配置 USB 和 SDIO

请参考《外围设备驱动操作指南》中的 USB 和 SDIO 操作指南。

SDIO 的 IO 电压有 1.8V 和 3.3V 两种，请确保 WiFi 模块的 IO 电压和主控芯片 SDIO 电压一致。

2.1.4 SDIO 中断配置

Kernel 默认没有开启 SDIO 中断，使用不支持 OOB 的 WiFi 时需要将 SDIO 中断打开，开启方法如 Hi3516EV200 平台：文件 arch/arm/boot/dts/hi3516ev200.dtsi，在 WiFi 对接的 SDIO 端口属性中增加：cap-sdio-irq。

注意

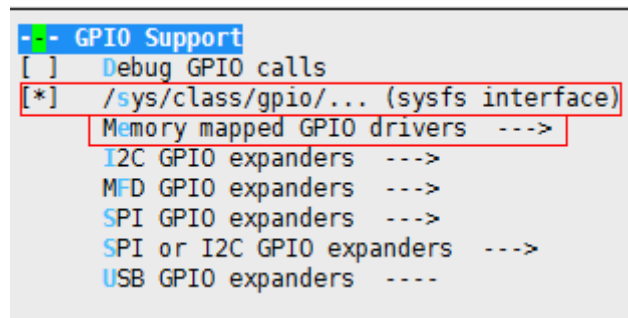
配置并编译完内核后，需要基于新的内核重新编译 WiFi 驱动，否则 WiFi 驱动在运行时会出现空指针或找不到内核符号错误。

2.1.5 配置通用 GPIO

打开通用 GPIO 配置，进入 Device Drivers，开启 GPIO Support，然后再进入 GPIO Support，按图 2-3 配置。

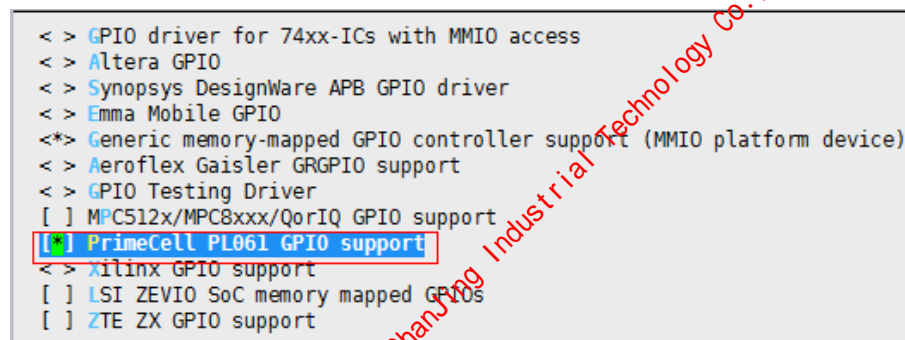


图2-3 通用 GPIO 配置



然后进入 Memory mapped GPIO drivers，按图 2-4 配置。

图2-4 通用 GPIO 配置 2



内核配置好后请编译一次内核，并烧写到单板。

2.2 Boot 配置

需要在 Boot 表格中配置 SDIO 和 USB 管脚复用和驱动电流，配置方法请参考 Boot 表格配置文档。配置完后编译生成新的 Boot 文件并烧写到单板。配置表格的位置在 SDK 包 package/osdrv.tgz 文件里的 tools/pc/u-boot_tools 目录。



3 驱动移植

3.1 RTL8189FTV 驱动移植

步骤 1 从 Realtek 或者模组厂家获取驱动，如：

rtl8189FS_linux_v4.3.24.11_26052.20180108.tar.gz。

步骤 2 对压缩包进行解压，得到 rtl8189FS_linux_v4.3.24.11_26052.20180108 代码目录。

```
tar zxvf rtl8189FS_linux_v4.3.24.11_26052.20180108.tar.gz
```

步骤 3 修改文件 rtl8189FS_linux_v4.3.24.11_26052.20180108/include/autoconf.h。

去掉驱动打印，注释掉以下这一行。

```
//#define CONFIG_DEBUG /* DBG_871X, etc... */
```

如果调试需要打印信息，可以不修改这一行。

步骤 4 修改文件 rtl8189FS_linux_v4.3.24.11_26052.20180108/Makefile。

设置：

```
CONFIG_REDUCE_TX_CPU_LOADING = y
```

步骤 5 修改文件 rtl8189FS_linux_v4.3.24.11_26052.20180108/hal/rtl8188f/sdio/rtl8189fs_xmit.c。

rtl8188fs_xmit_handler 函数中的两处 rtw_msleep_os(1)修改为 rtw_usleep_os(1)，以降低 CPU 占用率。

```
ret = xmit_xmitframes(padapter, pxmitpriv);
if (ret == -2) {
    /* here sleep 1ms will cause big TP loss of TX */
    /* from 50+ to 40+ */
    if (padapter->registrypriv.wifi_spec)
        rtw_msleep_os(1);
    else
#ifdef CONFIG_REDUCE_TX_CPU_LOADING
        // rtw_msleep_os(1);
        rtw_usleep_os(1);
#endif
}
```



```

#else
    rtw_yield_os();
#endif
    goto next;
}

_enter_critical_bh(&pxmitpriv->lock, &irq1);
ret = rtw_txframes_pending(padapter);
_exit_critical_bh(&pxmitpriv->lock, &irq1);
if (ret == 1) {
#ifdef CONFIG_REDUCE_TX_CPU_LOADING
    //rtw_msleep_os(1);
#endif
    goto next;
}

```

步骤 6 编译。

在 rtl8189FS_linux_v4.3.24.11_26052.20180108 的同级目录下，执行如下命令：

```

make -C rtl8189FS_linux_v4.3.24.11_26052.20180108 ARCH=arm
CROSS_COMPILE=arm-himix100-linux- KSRC=/home/kernel/linux-4.9.y

```

其中 arm-himix100-linux-要修改为编译内核采用的编译器，/home/kernel/linux-4.9.y 要修改为已经编译好的内核目录。

编译完后会在 rtl8189FS_linux_v4.3.24.11_26052.20180108 目录下生成驱动 8189fs.ko。

----结束

3.2 RTL8188FTV 驱动移植

步骤 1 从 Realtek 或者模组厂家获取驱动，如：

rtl8188FU_linux_v5.2.11.1_22924.20170703.tar.gz。

步骤 2 对压缩包进行解压，得到 rtl8188FU_linux_v5.2.11.1_22924.20170703 代码目录。

```

tar zxvf rtl8188FU_linux_v5.2.11.1_22924.20170703.tar.gz

```

步骤 3 修改文件 rtl8188FU_linux_v5.2.11.1_22924.20170703/include/autoconf.h。

1) 开启 CONFIG_IOCTL_CFG80211

取消以下这一行的注释：

```
#define CONFIG_IOCTL_CFG80211
```

2) 开启 RTW_USE_CFG80211_STA_EVENT

取消以下这一行的注释：



```
#define RTW_USE_CFG80211_STA_EVENT
```

步骤 4 修改文件 rtl8188FU_linux_v5.2.11.1_22924.20170703/Makefile。

去掉驱动打印，设置：

```
CONFIG_RTW_DEBUG = n
```

步骤 5 编译。

在 rtl8188FU_linux_v5.2.11.1_22924.20170703 的同级目录下，执行如下命令：

```
make -C rtl8188FU_linux_v5.2.11.1_22924.20170703 ARCH=arm CROSS_COMPILE=arm-himix100-linux- KSRC=/home/kernel/linux-4.9.y
```

其中 arm-himix100-linux-要修改为编译内核采用的编译器，/home/kernel/linux-4.9.y 要修改为已经编译好的内核目录。

编译完后会在 rtl8188FU_linux_v5.2.11.1_22924.20170703 目录下生成驱动 8188fu.ko。

-----结束

Cogobuy Only For Shenzhen FuShi ChanJing Industrial Technology Co., Ltd.



4 WiFi 工具

操作 WiFi 需要使用到 wpa_supplicant、libnl、iwconfig、iwlist、iwpriv、iperf 工具，这些工具都是开源软件，可以在网上下载最新版本使用。

表4-1 WiFi 工具获取路径

工具名称	下载路径	编译结果
wpa_supplicant	http://w1.fi/releases	wpa_supplicant、wpa_cli、hostapd
libnl	http://www.linuxfromscratch.org/blfs/view/svn/basicnet/libnl.html	libnl-genl.so.2.0.0、libnl.so.2.0.0
Wireless tools	http://www.linuxfromscratch.org/blfs/view/svn/basicnet/wireless_tools.html	iwconfig、iwlist、iwpriv
iperf	http://iperf.fr/iperf-download.php	iperf

编译方法请参考网址中的说明，其中 wpa_supplicant 的编译依赖 libnl 库，需要先编译 libnl 再编译 wpa_supplicant。



5 WiFi 基本操作

5.1 载入文件

5.1.1 载入驱动文件

将内核中的 `net/wireless/cfg80211.ko` 和编译出来的驱动拷贝到单板上，驱动在单板上的目录不重要，比如可以放在 `/kmod` 目录。

5.1.2 载入工具

- 将 `libnl-genl.so.2.0.0` 和 `libnl.so.2.0.0` 拷贝到单板的 `/lib` 目录。进入单板 `/lib` 目录，创建这两个文件的软链接：

```
ln -s libnl-genl.so.2.0.0 libnl-genl.so.2
ln -s libnl.so.2.0.0 libnl.so.2
```

- 拷贝 `iwconfig`、`iwlist`、`iwpriv`、`iperf` 拷贝到单板的 `/sbin` 目录下。这几个是调试工具，实际使用时可以不用拷贝这几个文件。
- STA 模式需要将 `wpa_supplicant`、`wpa_cli` 拷贝到单板的 `/sbin` 目录下。
- AP 模式需要将 `hostapd` 拷贝到单板的 `/sbin` 目录下。

工具拷贝到单板后，需要修改工具的可执行权限，如：

```
chmod a+x wpa_supplicant
```

5.1.3 wpa_supplicant.conf 文件

`wpa_supplicant.conf` 是启动 `wpa_supplicant` 进程时需要使用到的配置文件。可以在单板上新建一个，目录不重要，比如放在 `/etc/Wireless` 目录下。文件内容如下：

```
ctrl_interface=/var/wpa_supplicant
```

5.1.4 hostapd.conf 文件

`hostapd.conf` 是启动 `hostapd` 进程时需要使用到的配置文件。可以在单板上新建一个，目录不重要，比如放在 `/etc/Wireless` 目录下。文件内容请参考 [hostapd 配置和启动 SoftAP](#)。



5.1.5 udhcpd.conf 文件

udhcpd.conf 文件是 SoftAP 模式下 DHCP server 需要使用到的配置文件，可以参考网上资料创建一个 udhcpd.conf 文件。将 udhcpd.conf 到单板目录下，目录不重要，比如放在/etc/Wireless 目录下。

5.2 WiFi 设备检测

SDIO WiFi 需要在使用前先检测到 SDIO 设备，USB WiFi 需要在使用前先检测到 USB 设备。在 Boot 中配置好 SDIO 后，Realtek 的 SDIO WiFi 在开机时即可检测到 SDIO 设备，如果看到“mmc1: new SDIO card at address 0001”这样的打印，说明卡检测成功。

开机后也可以通过执行 cat /proc/mci/mci_info 命令看到是否检测到 SDIO 设备，如图 5-1 所示。

图5-1 查看 SDIO 设备

```
~ #  
~ # cat /proc/mci/mci_info  
MCI0: unplugged_disconnected  
MCI1: plugged_connected  
      Type: SDIO card Mode: HS  
      Speed Class: Class 0  
      Uhs Speed Grade: Less than 10MB/sec(0h)  
      Host work clock: 50MHz  
      Card support clock: 50MHz  
      Card work clock: 50MHz  
      Card error count: 0  
MCI2: invalid  
~ #  
~ #
```

USB WiFi 同样开机就可以检测到 USB 设备，通过执行 lsusb 命令可以看到是否检测到 USB 设备，如图 5-2 所示。

图5-2 查看 USB 设备

```
~ # lsusb  
Bus 001 Device 001: ID 1d6b:0002  
Bus 001 Device 002: ID 148f:7601  
Bus 002 Device 001: ID 1d6b:0003  
~ #
```

如果看到 WiFi 设备的 USB ID，那表示检测成功，USB ID 请向 WiFi 厂家询问。



5.3 STA 模式基本操作示例

5.3.1 加载驱动

步骤 1 加载驱动。

各个 WiFi 芯片请按以下对应关系加载相应的驱动模块。

- rtl8189ftv
insmod cfg80211.ko
insmod 8189fs.ko
- rtl8188ftv
insmod cfg80211.ko
insmod 8188fu.ko

步骤 2 查看驱动是否加载成功。

执行 shell 命令：

```
iwconfig
```

如果看到有一个 wlan0 网口，那说明驱动已经初始化成功，WiFi 设备可用。

图5-3 iwconfig 执行结果

```
# iwconfig
lo        no wireless extensions.

eth0      no wireless extensions.

wlan0     unassociated Nickname:"<WIFI@REALTEK>"
          Mode:Auto Frequency=2.412 GHz Access Point: Not-Associated
          Sensitivity:0/0
          Retry:off RTS thr:off Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality:0 Signal level:0 Noise level:0
          Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
          Tx excessive retries:0 Invalid misc:0 Missed beacon:0
```

步骤 3 启动 WiFi 网口。

执行 shell 命令：

```
ifconfig wlan0 up
```

执行完后，WiFi 是可用状态，可以进行扫描和连接操作了。

----结束

5.3.2 扫描 AP

执行 shell 命令：



```
iwlist wlan0 scan
```

图5-4 扫描 AP 执行结果

```
# iwlist wlan0 scan
wlan0    Scan completed :
          Cell 01 - Address: F4:EC:38:22:30:60
                   ESSID:"HiMMI"
                   Protocol:IEEE 802.11bg
                   Mode:Master
                   Frequency:2.412 GHz (Channel 1)
                   Encryption key:on
                   Bit Rates:54 Mb/s
                   Extra:wpa_ie=dd160050f20101000050f20401000050f20401000050f202
                   IE: WPA Version 1
                        Group Cipher : CCMP
                        Pairwise Ciphers (1) : CCMP
                        Authentication Suites (1) : PSK
                   Extra:rsn_ie=30140100000fac0401000000fac0401000000fac020100
                   IE: IEEE 802.11i/WPA2 Version 1
                        Group Cipher : CCMP
                        Pairwise Ciphers (1) : CCMP
                        Authentication Suites (1) : PSK
                        Preauthentication Supported
                   Quality=0/100  Signal level=42/100
```

扫描到的 AP 会以“Cell xx”的形式显示，一个 AP 对应一个“Cell xx”。

每个 AP 的信息包括：

- Address: MAC 地址。
- ESSID: AP 的名称，即 SSID。
- Protocol: IEEE80211 协议，11b/g/n。
- Frequency: 信道。
- 认证加密信息: WEP、WPA-PSK、WPA2-PSK、WPA、WPA2。
- Quality: 信号质量，该数据有些 WiFi 显示得不准确，可以忽略。
- Signal Level: 信号强度，数字越大，信号强度越高，WiFi 芯片不同，显示的方式有些区别，有的是以 xx/100 类型显示，有的是以 xx dBm 显示。

上述信息并不是所有 WiFi 都是以这种格式显示，WiFi 不同显示的格式也不一样。

注意

使用 iwlist 进行扫描时，iwlist 不会等驱动扫描完所有信道才返回扫描结果，所以经常会出现有些 AP 没有搜出来的情况，尤其是 MT7601U，由于在每个信道上停留的时间较长，所以第一次扫描时，只能搜到 1~2 个信道里的 AP。

5.3.3 连接 AP

连接 AP 是通过 wpa_supplicant 进程进行的。wpa_supplicant 是开源代码，Linux、Android 都是采用它负责 WiFi 的连接过程，它包含了 WEP、WPA/WPA2、WPA-PSK/WPA2-PSK、WAPI、WPS、P2P、EAP 等协议。



步骤 1 启动 wpa_supplicant 进程。

执行 shell 命令：

```
wpa_supplicant -iwlan0 -Dnl80211 -c/etc/Wireless/wpa_supplicant.conf&
```

- -iwlan0 表示使用 wlan0 网口；
- -Dnl80211 表示使用 cfg80211 接口（用户态的接口是 libnl，内核中是 cfg80211）；
- /etc/Wireless/wpa_supplicant.conf 是 wpa_supplicant 的配置文件，要保证该文件已经存在。

执行完后，用 ps 命令查看一下 wpa_supplicant 进程是否存在，存在表示工作正常。如果没有 wpa_supplicant 进程，可以增加 wpa_supplicant 的打印级别，从 log 看出现什么问题，如：

```
wpa_supplicant -iwlan0 -Dnl80211 -c/etc/Wireless/wpa_supplicant.conf -ddd  
&
```

步骤 2 启动 wpa_cli 进程。

执行 shell 命令：

```
wpa_cli -iwlan0 -p/var/wpa_supplicant
```

执行成功会出现 “>” 符号。

如果出现 “Could not connect to wpa_supplicant -re-trying”，那表示 wpa_cli 不能和 wpa_supplicant 建立 socket 连接，这时要检查 wpa_supplicant 进程是否还在，再看是否有 /var/wpa_supplicant/wlan0，然后检查 wpa_supplicant.conf 文件中是否是 ctrl_interface=/var/wpa_supplicant。

步骤 3 扫描。

在 “>” 后执行 “scan” 命令，收到 “CTRL-EVENT-SCAN-RESULTS” 后再执行 “scan_results”，会获得扫描结果。

图5-5 wpa_cli 扫描 AP 结果

```
> scan
OK
<3>CTRL-EVENT-SCAN-RESULTS
<3>WPS-AP-AVAILABLE

> scan_results
bssid / frequency / signal level / flags / ssid
78:a1:06:48:e2:e8      2472      -65      [WPA-PSK-CCMP] [WPA2-PSK-CCMP] [WPS] [ESS] B21-1
40:4d:8e:81:08:f1      2462      -69      [WPA-PSK-TKIP] [ESS] B25_chenxie
f4:ec:38:22:30:60      2412      -74      [WPA-PSK-CCMP] [WPA2-PSK-CCMP-preauth] [ESS] HiMMI
8c:21:0a:a5:cd:b2      2437      -48      [WEP] [ESS] B21
```

步骤 4 连接。

连接 OPEN 方式的 AP：

1. 在 “>” 后执行 add_network，假如返回网络 ID 为 0。
2. 配置网络的 SSID，执行 set_network 0 ssid “AP 的 SSID”。
3. 配置网络的加密方式，执行 set_network 0 key_mgmt NONE。



4. 启动网络，执行 `select_network 0`。
5. 收到 `CTRL-EVENT-CONNECTED` 表示连接成功。

图5-6 连接 AP

```
> add_network
0
> set_network 0 ssid "WINDSKY_WLAN"
OK
> set_network 0 key_mgmt NONE
OK
> enable_network 0
OK
> wlan0: Trying to associate with ac:f7:f3:e5:d7:33 (SSID='WINDSKY_WLAN' freq=2437 MHz)
<3>CTRL-EVENT-SCAN-RESULTS
<3>WPS-AP-AVAILABLE
<3>Trying to associate with ac:f7:f3:e5:d7:33 (SSID='WINDSKY_WLAN' freq=2437 MHz)
wlan0: Associated with ac:f7:f3:e5:d7:33
<3>Associated with ac:f7:f3:e5:d7
wlan0: CTRL-EVENT-CONNECTED - Connection to ac:f7:f3:e5:d7:33 completed (auth) [id=0 id_str=]
<3>CTRL-EVENT-CONNECTED - Connection to ac:f7:f3:e5:d7:33 completed (auth) [id=0 id_str=]
```

连接 WPA-PSK/WPA2-PSK 方式的 AP:

- 在“>”后执行 `add_network`，假如返回网络 ID 为 0。
- 配置网络的 SSID，执行 `set_network 0 ssid “AP 的 SSID”`。
- 配置网络的加密方式，执行 `set_network 0 psk “AP 的密码”`。
- 启动网络，执行 `select_network 0`。
- 收到 `CTRL-EVENT-CONNECTED` 表示连接成功。

步骤 5 获取 IP 地址。

输入 q 退出 `wpa_cli`，执行 shell 命令：`udhcpc -i wlan0`

配置了 IP 地址后，可以 ping 网关看是否能 ping 通。

-----结束

5.3.4 卸载驱动

各个 WiFi 芯片请按以下对应关系卸载相应的驱动模块。

- rtl8189ftv
ifconfig wlan0 down
rmmod 8189fs.ko
rmmod cfg80211.ko
- rtl8188ftv
ifconfig wlan0 down
rmmod 8188fu.ko
rmmod cfg80211.ko



5.4 SoftAP 模式基本操作示例

5.4.1 检查 WiFi 设备、加载驱动

rtl8189ftv、rtl8188ftv 加载驱动的方法和 STA 模式一样。

5.4.2 hostapd 配置和启动 SoftAP

配置 SoftAP 是通过 hostapd 进行的。hostapd 和 wpa_supplicant 类似，它包含了 AP 端的各种认证协议、连接流程，wpa_supplicant 是 STA 端的。

步骤 1 修改 hostapd.conf 文件。

hostapd 进程需要使用 hostapd.conf 配置文件，在配置文件里设置 SSID、信道、加密方式等。配置文件的内容举例如下：

- OPEN

```
interface=wlan0
driver=nl80211
ctrl_interface=/var/hostapd
ssid=HisiAP
channel=6
hw_mode=g
ieee80211n=1
ht_capab=[SHORT-GI-20][SHORT-GI-40][HT40-]
```
- WPA2-PSK

```
interface=wlan0
driver=nl80211
ctrl_interface=/var/hostapd
ssid=HisiAP
channel=6
hw_mode=g
ieee80211n=1
ht_capab=[SHORT-GI-20][SHORT-GI-40][HT40-]
wpa=3
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP CCMP
wpa_passphrase=12345678
```

hostapd 是开源代码，配置文件中的参数可以参考网络资源。

ht_capab 可以配置是否支持 40M 带宽，带有[SHORT-GI-40][HT40-]或者[SHORT-GI-40][HT40+]即支持 40M 带宽。当 channel 小于 6 时，用[SHORT-GI-40][HT40+]，当 channel 大于等于 6 时，用[SHORT-GI-40][HT40-]。

步骤 2 启动 hostapd 进程。

执行 Shell 命令：



```
hostapd /etc/Wireless/hostapd.conf &
```

执行完后，用 `ps` 命令查看一下 `hostapd` 进程是否存在，存在表示工作正常，用 STA 设备可以搜索到 SoftAP。如果没有，可以增加 `hostapd` 的打印级别，从 log 看出现什么问题，如：

```
hostapd -ddd /etc/Wireless/hostapd.conf &
```

----结束

5.4.3 开启 udhcpd

执行 Shell 命令：

```
ifconfig wlan0 192.168.1.1  
udhcpd -fS /etc/Wireless/udhcpd.conf
```

请确保 `/etc/Wireless/udhcpd.conf` 文件存在，并且配置的网段为 `192.168.1.x`。执行完后，用 STA 设备可以扫描和连接该 SoftAP，如果能成功连接且 ping 通网关表示此 AP 配置成功。

5.4.4 卸载驱动

卸载驱动的方法和 STA 模式一样。

5.5 适配国家或区域

不同的国家或区域，采用的频率范围有些不同，比如 2.4GHz 频段，美国支持 1~11 信道，中国和欧洲支持 1~13 信道，日本支持 1~14 信道。5GHz 频段也类似。WiFi 需要根据产品上市的国家或区域做相应的配置，以适用于该国家的频率范围。

不同的 WiFi 配置方法不一样，比如 RTL8188ftv：rtl8188ftv 配置成美国，在加载驱动时带上 `rtw_channel_plan=0x22` 参数：

```
insmod 8188fu.ko rtw_channel_plan=0x22
```

本文档无法列出所有的 WiFi 对应的国家配置，产品开发中请问模组厂家或者 WiFi 芯片厂家。



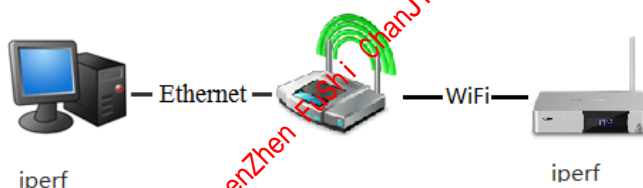
6 测试

6.1 吞吐量测试

吞吐量测试可以反映 WiFi 的性能，是目前芯片厂家、模组厂家、设备厂家普遍使用的测试方法，具有很高的认同度。吞吐量测试最常使用的工具是 iperf。

测试环境为 PC 机通过有线和 AP 连接，单板通过 WiFi 和 AP 连接，单板和 PC 机可以互相 ping 通。在 PC 机和单板上都有 iperf 工具，假设 PC 机的 IP 地址为 192.168.1.100，单板的 IP 地址为 192.168.1.101。

图6-1 吞吐量测试组网环境



6.1.1 TCP 发送吞吐量测试

发送吞吐量测试操作如下：

步骤 1 PC 机上命令行进入 iperf 工具目录，执行：

```
iperf -s
```

步骤 2 单板上通过 Shell 进入 iperf 工具目录，执行：

```
iperf -c 192.168.1.100 -t 10 -i 1
```



图6-2 发送吞吐量测试示例

```
# iperf -c 192.168.1.100 -t 10 -i 1
Client connecting to 192.168.1.100, TCP port 5001
TCP window size: 512 KByte (default)
[ 3] local 192.168.1.101 port 44753 connected with 192.168.1.100 port 5001
[ 3] 0.0- 1.0 sec 8.40 MBytes 70.5 Mbits/sec
[ 3] 1.0- 2.0 sec 8.57 MBytes 71.9 Mbits/sec
[ 3] 2.0- 3.0 sec 8.65 MBytes 72.5 Mbits/sec
[ 3] 3.0- 4.0 sec 8.52 MBytes 71.4 Mbits/sec
[ 3] 4.0- 5.0 sec 8.57 MBytes 71.9 Mbits/sec
[ 3] 5.0- 6.0 sec 8.52 MBytes 71.4 Mbits/sec
[ 3] 6.0- 7.0 sec 8.59 MBytes 72.1 Mbits/sec
[ 3] 7.0- 8.0 sec 8.52 MBytes 71.5 Mbits/sec
[ 3] 8.0- 9.0 sec 8.72 MBytes 73.1 Mbits/sec
[ 3] 9.0-10.0 sec 8.62 MBytes 72.4 Mbits/sec
[ 3] 0.0-10.0 sec 85.7 MBytes 71.6 Mbits/sec
```

其中，iperf -s 表示启动服务端，iperf -c 192.168.1.100 表示启动客户端，连接 192.168.1.100，-t 10 表示测试 10 秒钟，-i 1 表示每隔 1 秒钟打印一次结果。

最后打印的“0.0-10.0 sec 85.7 MBytes 71.6 Mbits/sec”表示这 10 秒钟的平均吞吐量为 71.6Mbps。

----结束

6.1.2 TCP 接收吞吐量测试

接收吞吐量测试操作如下：

步骤 1 单板上通过 Shell 进入 iperf 工具目录，执行：

```
iperf -s
```

步骤 2 PC 机上命令行进入 iperf 工具目录，执行：

```
iperf -c 192.168.1.101 -t 10 -i 1 -w 1M
```

图6-3 接收吞吐量测试示例

```
# iperf -s -i 1
Server listening on TCP port 5001
TCP window size: 1.00 MByte (default)
GetDesiredTssiAndCurrentTssi: BBP TSSI INFO is not ready. (BbpR47 = 0x94)
RT5390_AsicTxAlcGetAutoAgcOffset: Incorrect desired TSSI or current TSSI
[ 4] local 192.168.1.101 port 5001 connected with 192.168.1.100 port 59938
[ 4] 0.0- 1.0 sec 10.1 MBytes 85.0 Mbits/sec
[ 4] 1.0- 2.0 sec 10.3 MBytes 86.5 Mbits/sec
[ 4] 2.0- 3.0 sec 10.1 MBytes 84.4 Mbits/sec
[ 4] 3.0- 4.0 sec 9.86 MBytes 82.8 Mbits/sec
[ 4] 4.0- 5.0 sec 9.83 MBytes 82.4 Mbits/sec
[ 4] 5.0- 6.0 sec 9.92 MBytes 83.3 Mbits/sec
[ 4] 6.0- 7.0 sec 9.33 MBytes 78.3 Mbits/sec
[ 4] 7.0- 8.0 sec 9.99 MBytes 83.8 Mbits/sec
[ 4] 8.0- 9.0 sec 9.70 MBytes 81.4 Mbits/sec
[ 4] 9.0-10.0 sec 10.0 MBytes 84.2 Mbits/sec
[ 4] 0.0-10.1 sec 100 MBytes 83.3 Mbits/sec
```



iperf 也可以进行 UDP 测试，在有些 PC 机上单个 UDP 线程进行了限速，因此需要开启多个线程。

SoftAP 的吞吐量测试类似。

注意

有些 PC 机，由于安装了一些软件，对速率会有影响，一定要确保 PC 机没有速率的瓶颈。WEP 安全模式不能使用 802.11n 协议，因此速率比较低，一般只有 20+Mbps。

----结束

6.1.3 UDP 发送吞吐量测试

发送吞吐量测试操作如下：

步骤 1 PC 机上命令行进入 iperf 工具目录，执行：

```
iperf -s -u -l 32k
```

步骤 2 单板上通过 Shell 进入 iperf 工具目录，执行：

```
iperf -c 192.168.1.100 -u -t 10 -i 1 -l 32k -b 100M
```

----结束

6.1.4 UDP 接收吞吐量测试

接收吞吐量测试操作如下：

步骤 1 单板上通过 Shell 进入 iperf 工具目录，执行：

```
iperf -s -u
```

步骤 2 PC 机上命令行进入 iperf 工具目录，执行：

```
iperf -c 192.168.1.101 -u -t 10 -i 1 -l 32k -b 100M
```

----结束

6.2 射频指标测试

吞吐量测试可以反映 WiFi 的性能，在产品开发中必须要做的工作。有些有条件的公司还会进行射频指标测试，它可以准确的验证 WiFi 射频是否达标。因为模组厂家在生产模组时是必须做的工作，所以如果采用的是模组，这项工作可选的。但由于硬件设计时，有可能地线不干净、板上干扰等原因，影响 WiFi 射频性能，因此建议有条件的公司要进行该项测试。



射频指标包括：接收灵敏度、信道功率抑制、发送功率、发送载频容差、丢包率、EVM、接收杂散、发送杂散等。

测试仪器包含：频谱分析仪、功率测量仪、网络分析仪等。

测试方法较复杂，可以参考测量仪器说明书。

Cogobuy Only For ShenZhen FuShi ChanJing Industrial Technology Co., Ltd.