KBA_QuecOpen_NF3303 驱动适配手册_V1.0

www.quectel.com

About the Document

本文档适用于 MDM9628 和 MDM9X07 平台

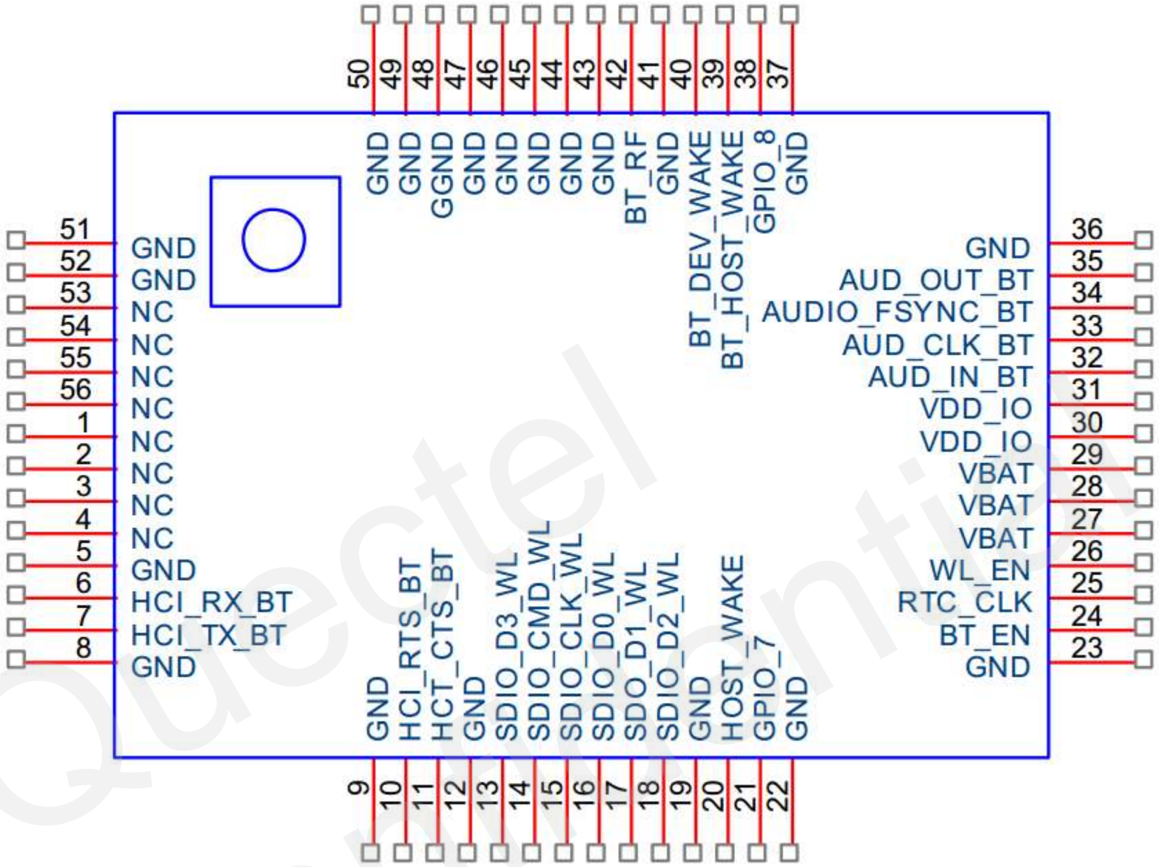
History

Revision	Date	Author	Description
1.0	2018-03-07	Edison	Initial

目录

本文档仅适用于 Linux3.18.20 内核

1. NF3303 硬件连接图及 GPIO 配置



NF3303		MDM9x07		
PIN NAME	PIN NO.	SDIO 接口	PIN NO.	GPIO
SDIO_D3_WL	13	SDC1_DATA3	129	GPIO_12
SDIO_D2_WL	18	SDC1_DATA2	130	GPIO_13
SDIO_D1_WL	17	SDC1_DATA1	131	GPIO_14
SDIO_D0_WL	16	SDC1_DATA0	132	GPIO_15
SDIO_CLK_WL	15	SDC1_CLK	133	GPIO_16
SDIO_CMD_WL	14	SDC1_CMD	134	GPIO_17
WL_EN	26	WLAN_EN	136	GPIO_38
HOST_WAKE	20	WAKE_ON_WIRELESS	135	GPIO_59
VBAT	27/28/29	PM_ENABLE	127	GPIO_1020
VDD_IO	30/31	VDD_EXT	7	

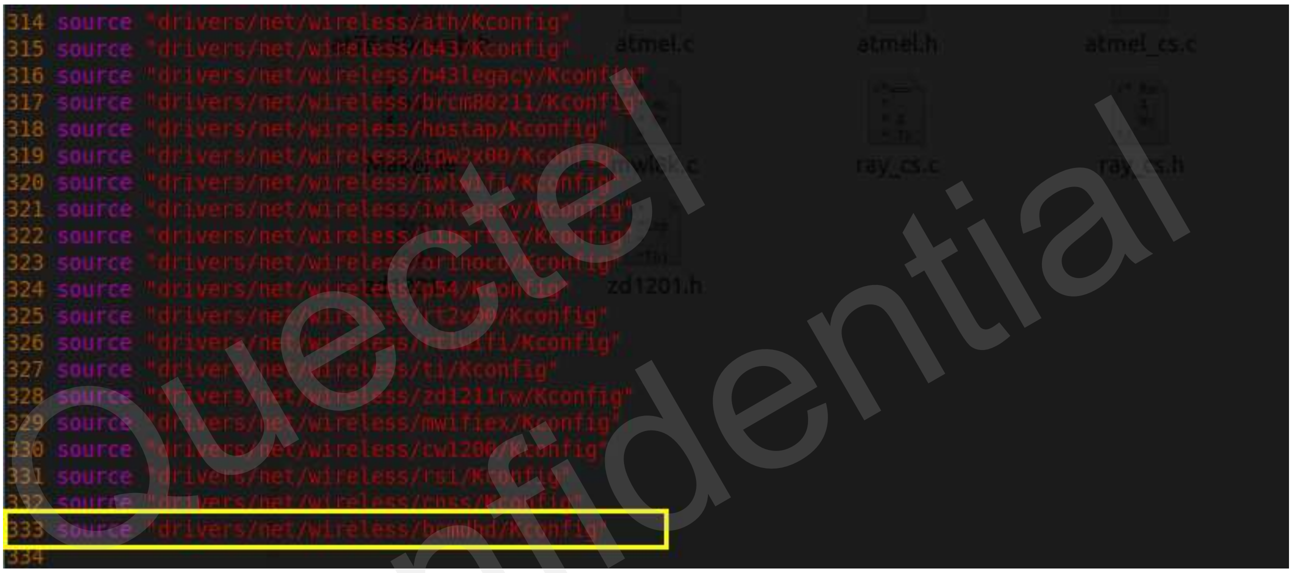
2. NF3303 驱动添加方法

2.1 添加内核驱动代码

2.1.1 将 NF3303 驱动源码拷贝到 ql-ol-sdk/ql-ol-kernel/drivers/net/wireless/bcmdhd 目录下替换：

2.1.2 修改 ql-ol-sdk/ql-ol-kernel/drivers/net/wireless/Kconfig，参考下图：

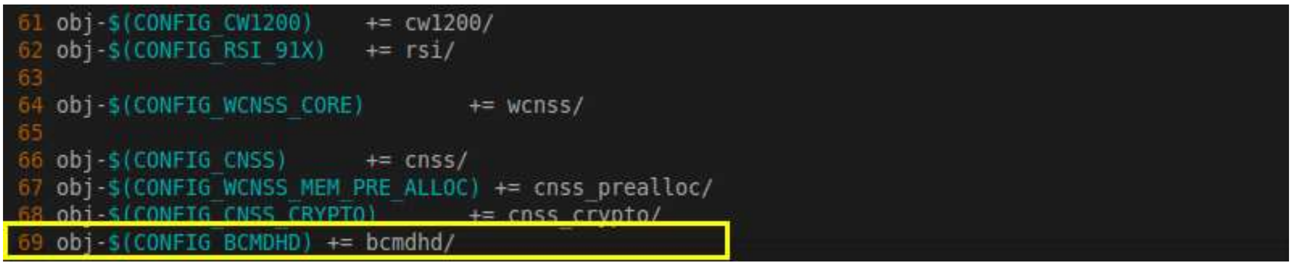
增加 source “drivers/net/wireless/bcmdhd/Kconfig”



```
source "drivers/net/wireless/ath/Kconfig"
source "drivers/net/wireless/b43/Kconfig"
source "drivers/net/wireless/b43legacy/Kconfig"
source "drivers/net/wireless/brcm80211/Kconfig"
source "drivers/net/wireless/hostap/Kconfig"
source "drivers/net/wireless/ipw2x00/Kconfig"
source "drivers/net/wireless/iwlwifi/Kconfig"
source "drivers/net/wireless/iwlegacy/Kconfig"
source "drivers/net/wireless/libertas/Kconfig"
source "drivers/net/wireless/orinoco/Kconfig"
source "drivers/net/wireless/p54/Kconfig"
source "drivers/net/wireless/rt2x00/Kconfig"
source "drivers/net/wireless/rtlwifi/Kconfig"
source "drivers/net/wireless/ti/Kconfig"
source "drivers/net/wireless/zd1211rw/Kconfig"
source "drivers/net/wireless/mwifiex/Kconfig"
source "drivers/net/wireless/cw1200/Kconfig"
source "drivers/net/wireless/rsi/Kconfig"
source "drivers/net/wireless/cnss/Kconfig"
source "drivers/net/wireless/bcmdhd/Kconfig"

```

2.1.3 修改 ql-ol-sdk/ql-ol-kernel/drivers/net/wireless/Makefile，参考下图：

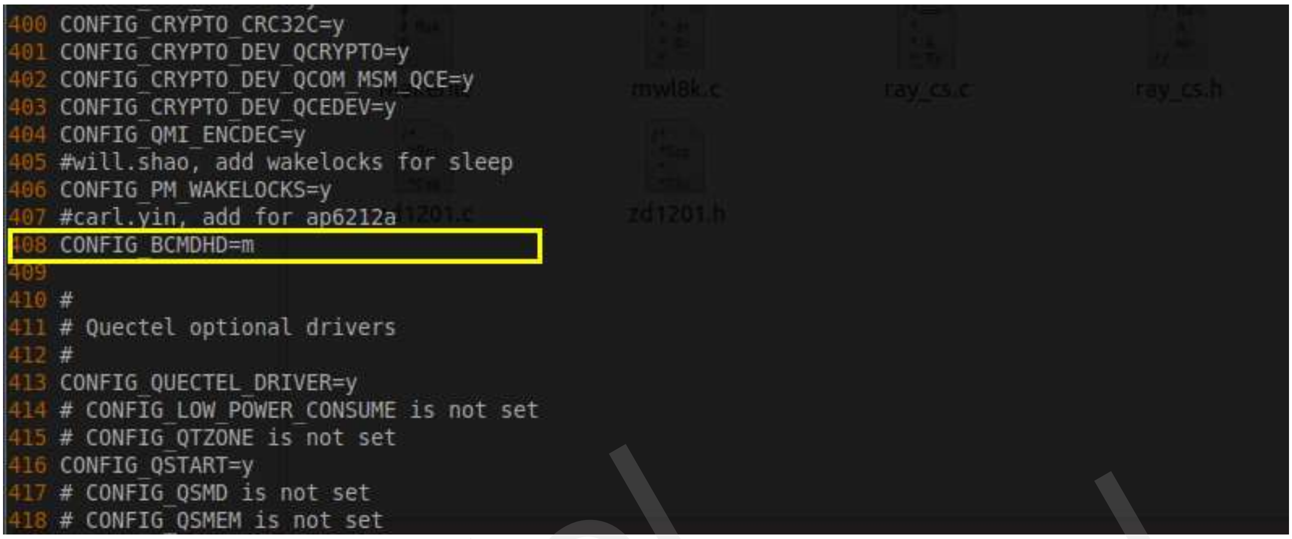
增加 obj-$(CONFIG_BCMDHD) += bcmdhd 配置。



```
obj-$(CONFIG_CW1200)     += cw1200/
obj-$(CONFIG_RSI_91X)    += rsi/

obj-$(CONFIG_WCNSS_CORE)          += wcnss/

obj-$(CONFIG_CNSS)       += cnss/
obj-$(CONFIG_WCNSS_MEM_PRE_ALLOC) += cnss_prealloc/
obj-$(CONFIG_CNSS_CRYPTO)         += cnss_crypto/
obj-$(CONFIG_BCMDHD) += bcmdhd/
```

2.1.4 修改 ql-ol-sdk/ql-ol-kernel/arch/arm/configs/mdm9607-perf_defconfig，参考下图：

增加 CONFIG_BCMDHD=m 配置。



```
CONFIG_CRYPTO_CRC32C=y
CONFIG_CRYPTO_DEV_QCRYPTO=y
CONFIG_CRYPTO_DEV_QCOM_MSM_QCE=y
CONFIG_CRYPTO_DEV_QCEDEV=y
CONFIG_QMI_ENCDEC=y
#will.shao, add wakelocks for sleep
CONFIG_PM_WAKELOCKS=y
#carl.yin, add for ap6212a
CONFIG_BCMDHD=m

#
# Quectel optional drivers
#
CONFIG_QUECTEL_DRIVER=y
# CONFIG_LOW_POWER_CONSUME is not set
# CONFIG_QTZONE is not set
CONFIG_QSTART=y
# CONFIG_QSMD is not set
# CONFIG_QSMEM is not set
```

2.1.5 重新编译内核，步骤如下：

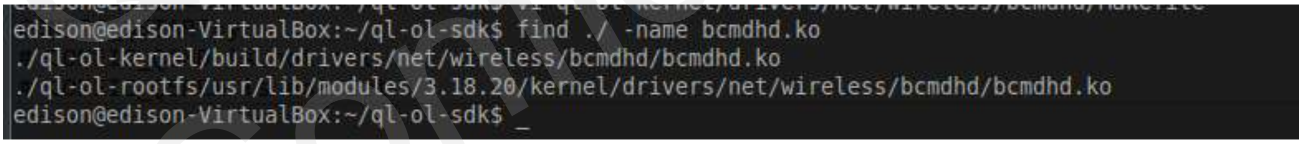
1、#cd ~/ql-ol-sdk

2、#make clean

3、#make kernel_menuconfig

4、#make

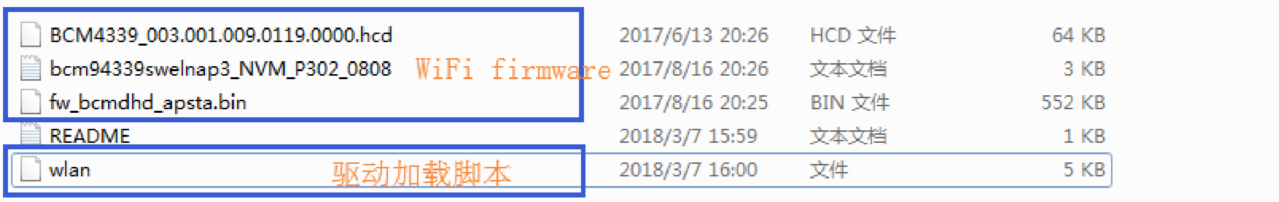
5、驱动生成路径在：



```
edison@edison-VirtualBox:~/ql-ol-sdk$ vi ql-ol-kernel/drivers/net/wireless/bcmdhd/Makefile
edison@edison-VirtualBox:~/ql-ol-sdk$ find ./ -name bcmdhd.ko
./ql-ol-kernel/build/drivers/net/wireless/bcmdhd/bcmdhd.ko
./ql-ol-rootfs/usr/lib/modules/3.18.20/kernel/drivers/net/wireless/bcmdhd/bcmdhd.ko
edison@edison-VirtualBox:~/ql-ol-sdk$ _
```

2.1.6 NF3303 固件及驱动加载脚本配置路径：

2.1.6.1 拿到 NF3303 固件清单，首先检查文件是否全：



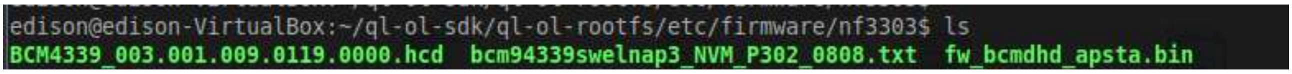
BCM4339_003.001.009.0119.0000.hcd	2017/6/13 20:26	HCD 文件	64 KB
bcm94339swelnap3_NVM_P302_0808	2017/8/16 20:26	文本文档	3 KB
fw_bcmdhd_apsta.bin	2017/8/16 20:25	BIN 文件	552 KB
README	2018/3/7 15:59	文本文档	1 KB
wlan	2018/3/7 16:00	文件	5 KB

WiFi firmware

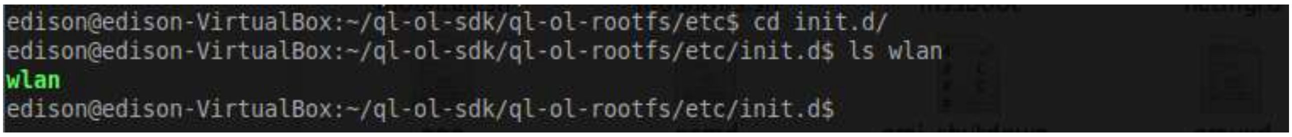
驱动加载脚本

2.1.6.2 EC2X 模块 删除 ql-ol-rootfs/etc/rc5.d/s91start_shortcut_fe_le 文件

2.1.6.3 将以上文件放置在对应 rootfs 文件路径下，重新 make rootfs：



```
edison@edison-VirtualBox:~/ql-ol-sdk/ql-ol-rootfs/etc/firmware/nf3303$ ls
BCM4339_003.001.009.0119.0000.hcd  bcm94339swelnap3_NVM_P302_0808.txt  fw_bcmdhd_apsta.bin
```



```
edison@edison-VirtualBox:~/ql-ol-sdk/ql-ol-rootfs/etc$ cd init.d/
edison@edison-VirtualBox:~/ql-ol-sdk/ql-ol-rootfs/etc/init.d$ ls wlan
wlan
edison@edison-VirtualBox:~/ql-ol-sdk/ql-ol-rootfs/etc/init.d$
```

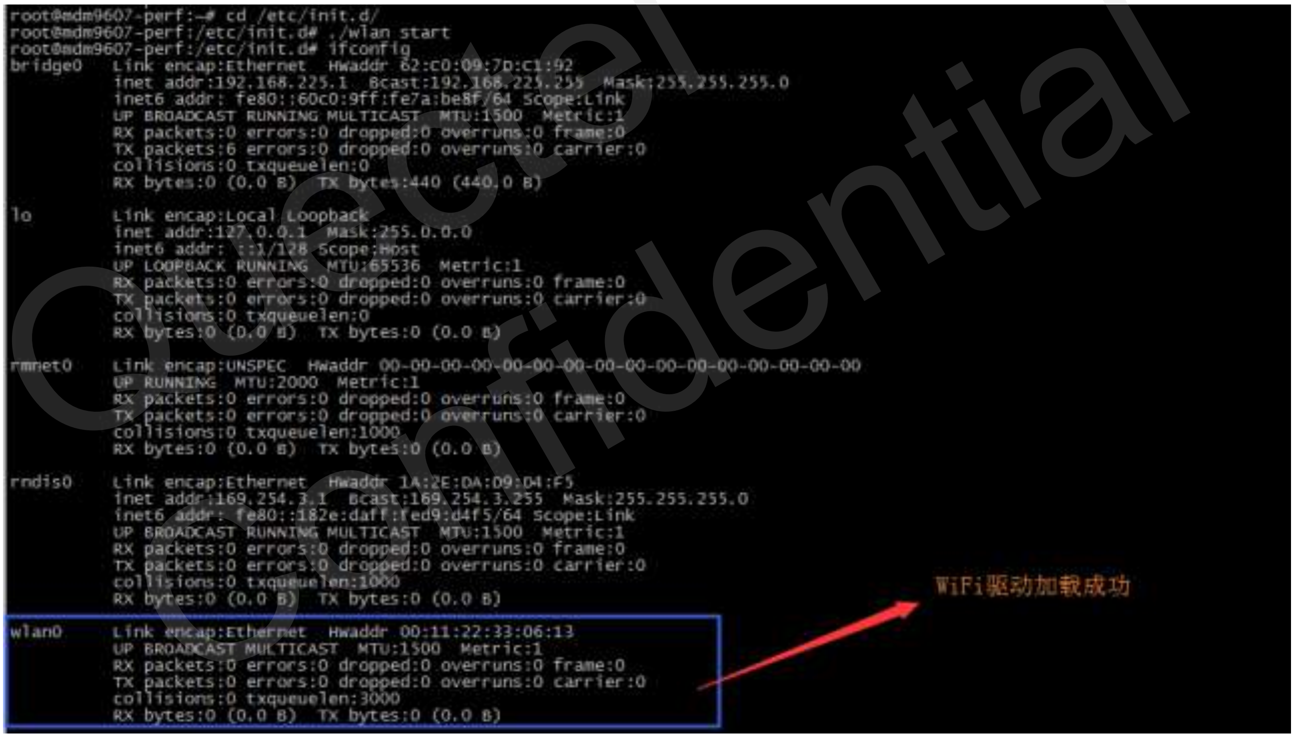
3. WIFI 功能测试

目前有三种方式，AT 命令、API 方式或命令行方式进行 NF3303 的测试。

AT 命令和 API 参考 Quectel 提供的 KBA 文档，这里仅对命令行方式进行说明。

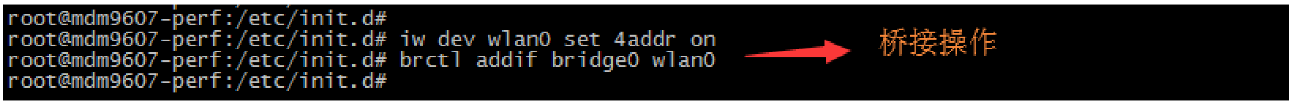
3.1 加载 WIFI 驱动

# cd /etc/init.d
# ./wlan start
# iw dev wlan0 set 4addr on
# brctl addif bridge0 wlan0



```
root@mdm9607-perf:~# cd /etc/init.d/
root@mdm9607-perf:/etc/init.d# ./wlan start
root@mdm9607-perf:/etc/init.d# ifconfig
bridge0   Link encap:Ethernet  HWaddr 62:C0:09:7D:C1:92
          inet addr:192.168.225.1  Bcast:192.168.225.255  Mask:255.255.255.0
          inet6 addr: fe80::60c0:9ff:fe7a:be8f/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:440 (440.0 B)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

rmnet0    Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          UP RUNNING  MTU:2000  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

rndis0    Link encap:Ethernet  HWaddr 1A:2E:DA:D9:D4:F5
          inet addr:169.254.3.1  Bcast:169.254.3.255  Mask:255.255.255.0
          inet6 addr: fe80::182e:daff:fed9:d4f5/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

wlan0     Link encap:Ethernet  HWaddr 00:11:22:33:06:13
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:3000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

WiFi驱动加载成功



```
root@mdm9607-perf:/etc/init.d#
root@mdm9607-perf:/etc/init.d# iw dev wlan0 set 4addr on
root@mdm9607-perf:/etc/init.d# brctl addif bridge0 wlan0
root@mdm9607-perf:/etc/init.d#
```

桥接操作

3.2 WIFI 开启 AP 模式

开启 WiFi 热点

EC2x 模块：

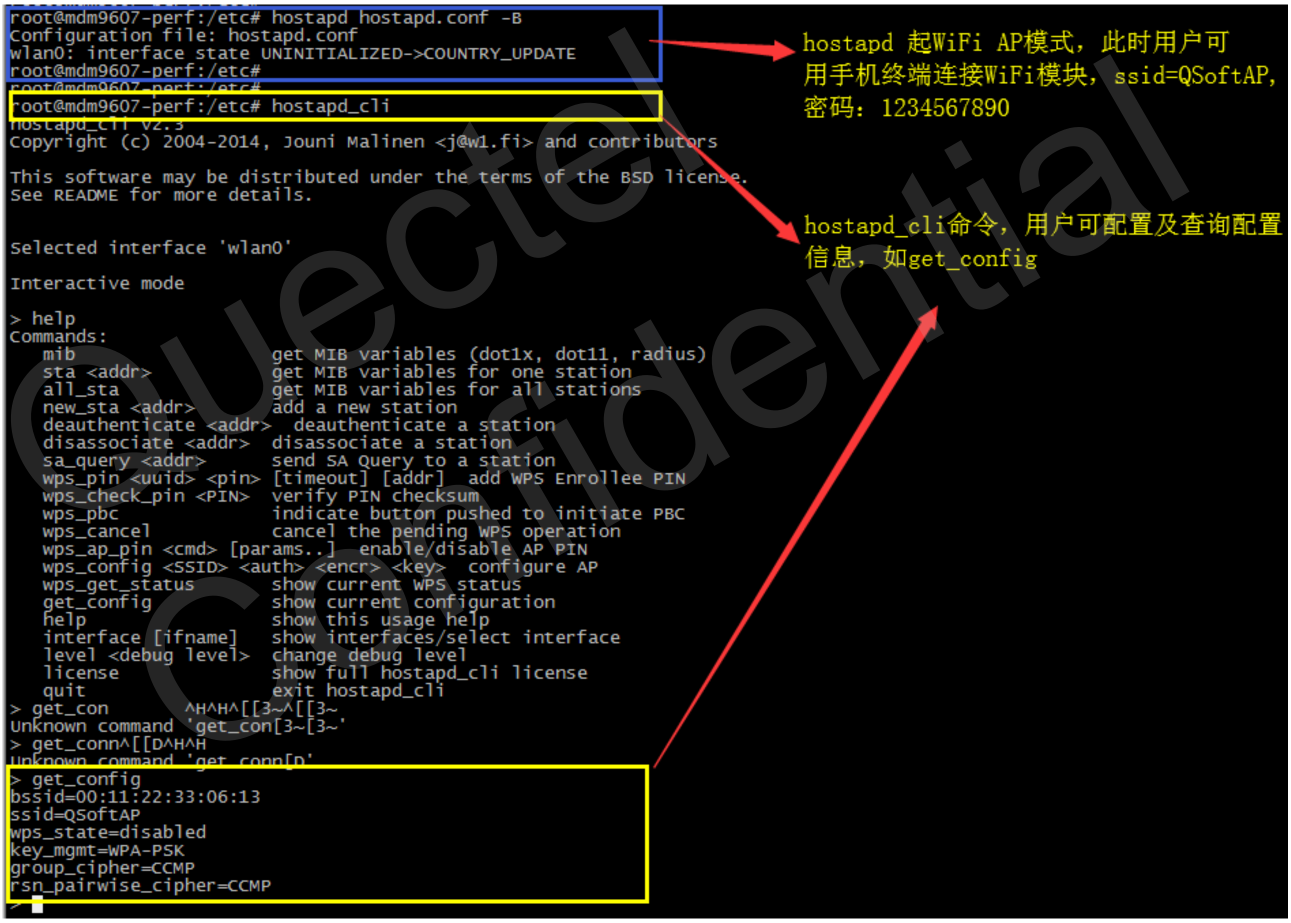
# hostapd /etc/hostapd.conf –B（此时用户即可用手机终端连接 WiFi 模块，ssid=QSoftAP，密码：1234567890）

# hostapd_cli（hostapd 命令行模式，可配置及查询操作）

AG35 模块：

# hostapd /data/misc/wifi/hostapd.conf –B

# hostapd_cli



```
root@mdm9607-perf:/etc# hostapd hostapd.conf -B
Configuration file: hostapd.conf
wlan0: interface state UNINITIALIZED->COUNTRY_UPDATE
root@mdm9607-perf:/etc#
root@mdm9607-perf:/etc#
root@mdm9607-perf:/etc# hostapd_cli
hostapd_cli v2.3
Copyright (c) 2004-2014, Jouni Malinen <j@w1.fi> and contributors

This software may be distributed under the terms of the BSD license.
See README for more details.


Selected interface 'wlan0'

Interactive mode

> help
Commands:
   mib                  get MIB variables (dot1x, dot11, radius)
   sta <addr>           get MIB variables for one station
   all_sta              get MIB variables for all stations
   new_sta <addr>       add a new station
   deauthenticate <addr>  deauthenticate a station
   disassociate <addr>  disassociate a station
   sa_query <addr>      send SA Query to a station
   wps_pin <uuid> <pin> [timeout] [addr]  add WPS Enrollee PIN
   wps_check_pin <PIN>  verify PIN checksum
   wps_pbc              indicate button pushed to initiate PBC
   wps_cancel           cancel the pending WPS operation
   wps_ap_pin <cmd> [params..]  enable/disable AP PIN
   wps_config <SSID> <auth> <encr> <key>  configure AP
   wps_get_status       show current WPS status
   get_config           show current configuration
   help                 show this usage help
   interface [ifname]   show interfaces/select interface
   level <debug level>  change debug level
   license              show full hostapd_cli license
   quit                 exit hostapd_cli
> get_con        ^H^H^[[3~^[[3~
Unknown command 'get_con[3~[3~'
> get_conn^[[D^H^H
Unknown command 'get_conn[D'
> get_config
bssid=00:11:22:33:06:13
ssid=QSoftAP
wps_state=disabled
key_mgmt=WPA-PSK
group_cipher=CCMP
rsn_pairwise_cipher=CCMP
>
```

hostapd 起WiFi AP模式，此时用户可用手机终端连接WiFi模块，ssid=QSoftAP，密码：1234567890

hostapd_cli命令，用户可配置及查询配置信息，如get_config

3.3 WIFI 开启 STA 模式

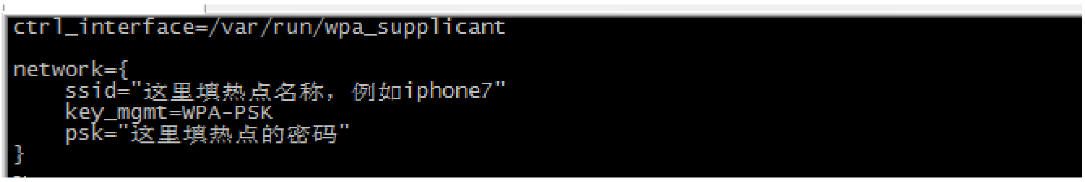
3.3.1 修改 wpa_supplicant 文件

EC2x 模块：

可将/etc/wpa_supplicant.conf 配置文件进行修改

AG35 模块：

可将/data/misc/wifi/wpa_supplicant.conf 配置文件进行修改，例如下图：

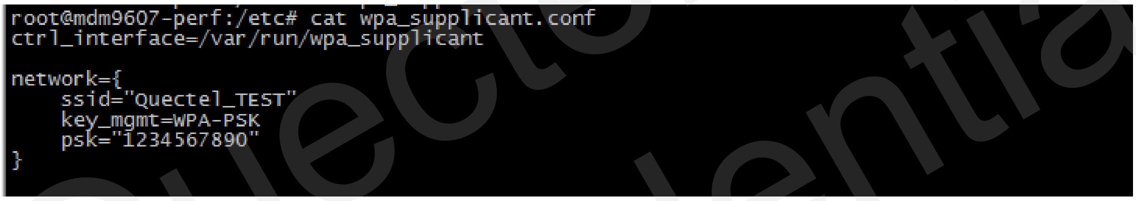


```
ctrl_interface=/var/run/wpa_supplicant

network={
    ssid="这里填热点名称，例如iphone7"
    key_mgmt=WPA-PSK
    psk="这里填热点的密码"
}
```

将 ssid=””，双引号内填充热点的 WiFi 名称，建议使用英文字符 ssid

将 psk=””，双引号内填充热点的 WiFi 密码

修改后如下图：



```
root@mdm9607-perf:/etc# cat wpa_supplicant.conf
ctrl_interface=/var/run/wpa_supplicant

network={
    ssid="Quectel_TEST"
    key_mgmt=WPA-PSK
    psk="1234567890"
}
```

3.3.2 开启用户自己的 WiFi 热点，进行组网测试，操作步骤如下：

第一步：

EC2x 模块：修改/etc/init.d/wlan 脚本，将图片中红色框路径改为图片所示；

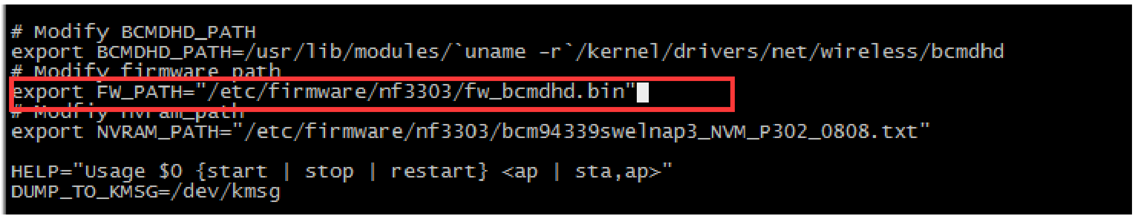
加载驱动操作：

# /etc/init.d/wlan start

AG35 模块：由于是只读文件系统，需要将**/etc/init.d/wlan** 脚本拷贝到**/data** 目录，然后修改红色框路劲如图片所示。

加载驱动操作：

# /data/wlan start



```
# Modify BCMDHD_PATH
export BCMDHD_PATH=/usr/lib/modules/`uname -r`/kernel/drivers/net/wireless/bcmdhd
# Modify firmware path
export FW_PATH="/etc/firmware/nf3303/fw_bcmdhd.bin"
# Modify nvram_path
export NVRAM_PATH="/etc/firmware/nf3303/bcm94339swelnap3_NVM_P302_0808.txt"

HELP="Usage $0 {start | stop | restart} <ap | sta,ap>"
DUMP_TO_KMSG=/dev/kmsg
```

第二步：

EC2x 模块：

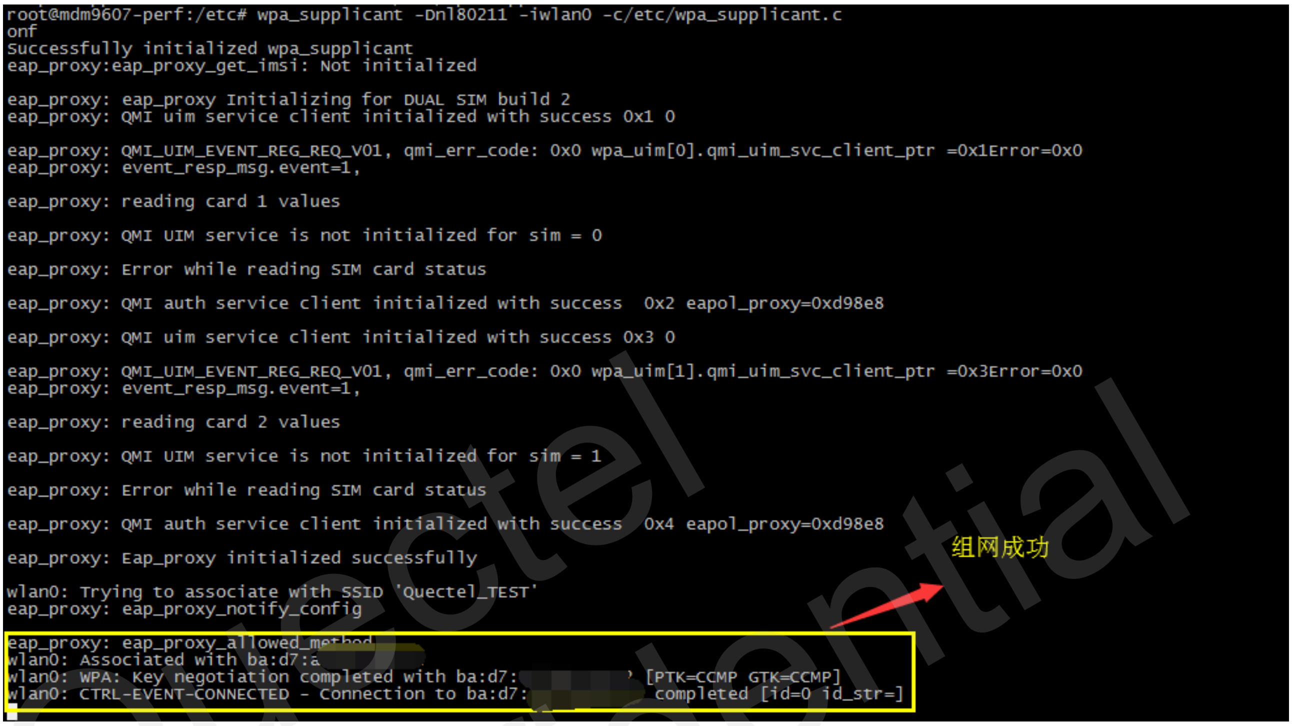
# wpa_supplicant -Dnl80211 -iwlan0 -c/etc/wpa_supplicant.conf

AG35 模块：

# wpa_supplicant -Dnl80211 -iwlan0 -c/data/misc/wifi/wpa_supplicant.conf

STA 组网成功，若想进行 AP 与 STA 之间互 ping 操作，需要通过 dhcp 获取 IP 操作，具体命令如下：

# udhcpc -i wlan0 -s /etc/udhcpc.d/50default &



```
root@mdm9607-perf:/etc# wpa_supplicant -Dnl80211 -iwlan0 -c/etc/wpa_supplicant.c
onf
Successfully initialized wpa_supplicant
eap_proxy:eap_proxy_get_imsi: Not initialized

eap_proxy: eap_proxy Initializing for DUAL SIM build 2
eap_proxy: QMI uim service client initialized with success 0x1 0

eap_proxy: QMI_UIM_EVENT_REG_REQ_V01, qmi_err_code: 0x0 wpa_uim[0].qmi_uim_svc_client_ptr =0x1Error=0x0
eap_proxy: event_resp_msg.event=1,

eap_proxy: reading card 1 values

eap_proxy: QMI UIM service is not initialized for sim = 0

eap_proxy: Error while reading SIM card status

eap_proxy: QMI auth service client initialized with success  0x2 eapol_proxy=0xd98e8

eap_proxy: QMI uim service client initialized with success 0x3 0

eap_proxy: QMI_UIM_EVENT_REG_REQ_V01, qmi_err_code: 0x0 wpa_uim[1].qmi_uim_svc_client_ptr =0x3Error=0x0
eap_proxy: event_resp_msg.event=1,

eap_proxy: reading card 2 values

eap_proxy: QMI UIM service is not initialized for sim = 1

eap_proxy: Error while reading SIM card status

eap_proxy: QMI auth service client initialized with success  0x4 eapol_proxy=0xd98e8

eap_proxy: Eap_proxy initialized successfully

wlan0: Trying to associate with SSID 'Quectel_TEST'
eap_proxy: eap_proxy_notify_config

eap_proxy: eap_proxy_allowed_method
wlan0: Associated with ba:d7:a[redacted]
wlan0: WPA: Key negotiation completed with ba:d7:[redacted] [PTK=CCMP GTK=CCMP]
wlan0: CTRL-EVENT-CONNECTED - Connection to ba:d7:[redacted] completed [id=0 id_str=]
```