

Lierda IC610 WIFI 应用指导

产品名称：ST-A35-IC610 工业核心板

产品型号：L-IDMIM0-AA185

版本：Rev1.0

日期：25/03/15

状态：受控版本

法律声明

若接收利尔达科技集团股份有限公司(以下称为“利尔达”)的此份文档,即表示您已经同意以下条款。若不同意以下条款,请停止使用本文档。

本文档版权归利尔达科技集团股份有限公司所有,保留任何未在本文档中明示授予的权利。文档中涉及利尔达的专有信息。未经利尔达事先书面许可,任何单位和个人不得复制、传递、分发、使用和泄漏该文档以及该文档包含的任何图片、表格、数据及其他信息。

本产品符合有关环境保护和人身安全方面的设计要求,产品的存放、使用和弃置应遵照产品手册、相关合同或者相关法律、法规的要求进行。

本公司保留在不预先通知的情况下,对此手册中描述的产品进行修改和改进的权利;同时保留随时修订或收回本手册的权利。



文件修订历史

文档版本	变更日期	修订人	审核人	变更内容
Rev1.0	25-03-15	YQA		初始版本

Lierda
利 尔 达

目录

法律声明	1
文件修订历史	2
目录	3
1 引言	4
2 WIFI 驱动开发	5
3 用户层操作	9
3.1 驱动加载	9
3.2 STA 模式	9
3.3 AP 模式	13
4 网络故障排查	15



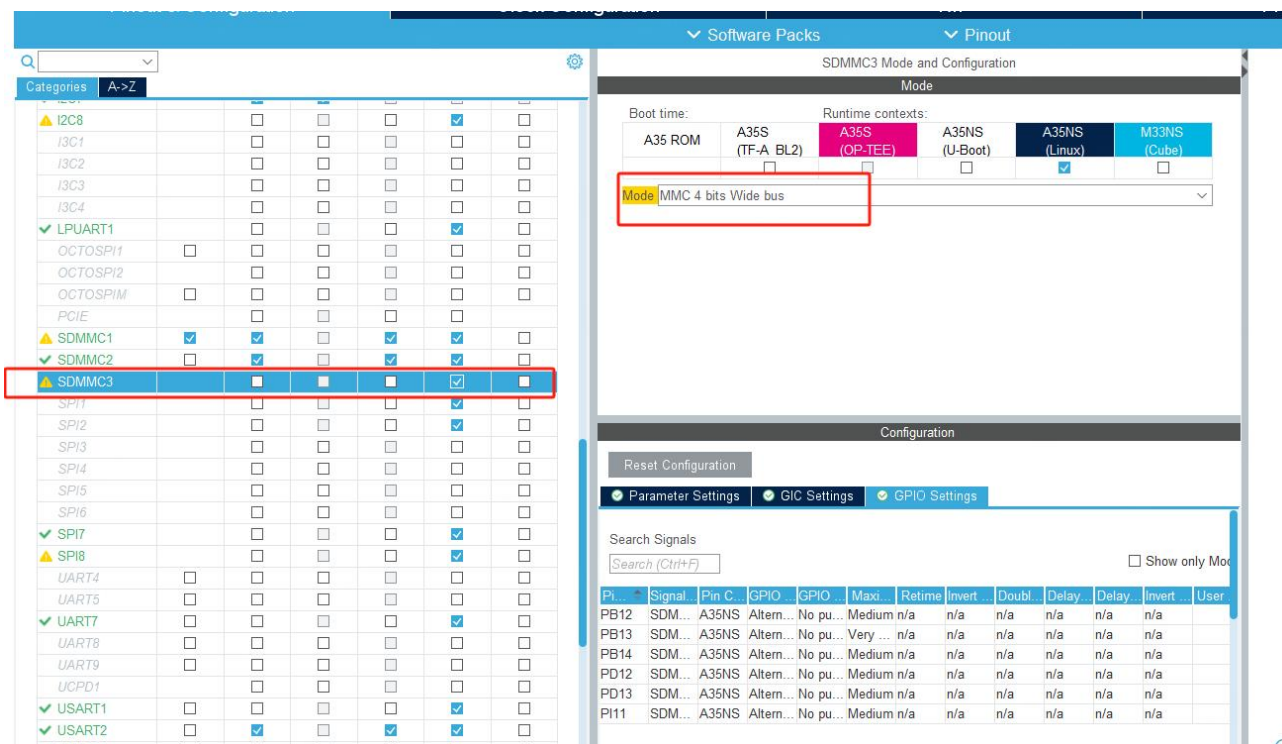
1 引言

本文档依托 IC610 evk，旨在使用 CUBEMX 构建 wifi 设备树及配置编 wifi 模组驱动，及用户层对 wifi 连接热点及作为 ap 热点相关测试等。

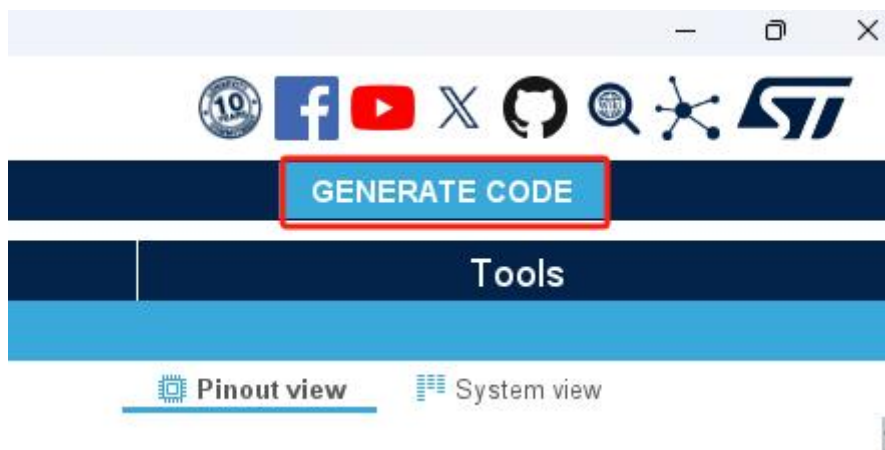


2 WIFI 驱动开发

Cubemx6.14 打开 ic610.ioc, SDIO 接口 wifi 默认使用 SDMMC3 接口。



Gpio 配置完成后点击 GENERATE CODE, 即可更新 dts。



代码生成完成后, 自动更新

ic610\CA35\DeviceTree\ic610\kernel\stm32mp255d-ic610-mx.dts

自动生成 gpio 复用

```
sdmmc3_pins_mx: sdmmc3_mx-0 {
    pins1 {
        pinmux = <STM32_PINMUX('B', 12, AF10)>, /* SDMMC3_D2 */
                <STM32_PINMUX('B', 14, AF10)>, /* SDMMC3_D0 */
                <STM32_PINMUX('D', 12, AF10)>, /* SDMMC3_CMD */
                <STM32_PINMUX('D', 13, AF10)>, /* SDMMC3_D1 */
                <STM32_PINMUX('I', 11, AF10)>; /* SDMMC3_D3 */
        bias-disable;
        drive-push-pull;
        slew-rate = <1>;
    };
    pins2 {
        pinmux = <STM32_PINMUX('B', 13, AF10)>; /* SDMMC3_CK */
        bias-disable;
        drive-push-pull;
        slew-rate = <3>;
    };
};

sdmmc3_opendrain_pins_mx: sdmmc3_opendrain_mx-0 {
    pins1 {
        pinmux = <STM32_PINMUX('B', 12, AF10)>, /* SDMMC3_D2 */
                <STM32_PINMUX('B', 14, AF10)>, /* SDMMC3_D0 */
                <STM32_PINMUX('D', 13, AF10)>, /* SDMMC3_D1 */
                <STM32_PINMUX('I', 11, AF10)>; /* SDMMC3_D3 */
        bias-disable;
        drive-push-pull;
        slew-rate = <1>;
    };
    pins2 {
        pinmux = <STM32_PINMUX('B', 13, AF10)>; /* SDMMC3_CK */
        bias-disable;
        drive-push-pull;
        slew-rate = <3>;
    };
    pins3 {
        pinmux = <STM32_PINMUX('D', 12, AF10)>; /* SDMMC3_CMD */
        bias-disable;
        drive-open-drain;
        slew-rate = <1>;
    };
};
```

及 sdmmc3 节点

在以下区域内添加相关配置信息即可。

```
wifi_pwrseq: wifi-pwrseq {
    compatible = "mmc-pwrseq-simple";
```

```

        reset-gpios = <&gpiof 4 GPIO_ACTIVE_LOW>;

};

&sdmmc3 {
    pinctrl-names = "default", "opendrain", "sleep";
    pinctrl-0 = <&sdmmc3_pins_mx>;
    pinctrl-1 = <&sdmmc3_opendrain_pins_mx>;
    pinctrl-2 = <&sdmmc3_sleep_pins_mx>;
    status = "okay";

    /* USER CODE BEGIN sdmmc3 */
    non-removable;

    st,neg-edge;
    bus-width = <4>;
    mmc-pwrseq = <&wifi_pwrseq>;
    vmmc-supply = <&scmi_v3v3>;
    cap-sdio-irq;
    #address-cells = <1>;
    #size-cells = <0>;

    bcmhdhd_wlan {
        compatible = "android,bcmdhd_wlan";
        status = "okay";
    };

    /* USER CODE END sdmmc3 */
};

```


wifi 驱动编译：

以 AP6256 为例，将 wifi 驱动源码复制到 linux-6.6.48/drivers/net/wireless/ap6256 下

```
cd drivers/net/wireless/ap6256
```

```
source
```

```
/opt/st/stm32mp2/5.0.3-openstlinux-6.6-yocto-scarthgap-mpu-v24.11.06/environment-setup-  
p-cortexa35-ostl-linux
```

```
make                                bcmdhd_sdio                CROS_COMPILE=aarch64-ostl-linux-  
LINUXDIR=/home/yqa/st/stm32mp2/sdk_v24.11.06/linux-6.6.48/linux-6.6.48/
```

编译完成后生成 bcmdhd.ko 文件。



3 用户层操作

3.1 驱动加载

将编译出来的 ko 文件及模组固件添加到文件系统下，如下

```
root@stm32mp2:~# ls /ap6256/
```

```
AP6256.hcd    ap6256.sh    bcmhdhd.ko    config.txt    fw_bcm43456c5_ag.bin
```

```
fw_bcm43456c5_ag_apsta.bin  hostapd.conf  nvram_ap6256.txt
```

ap6256.sh 为驱动加载脚本,自动在开机脚本/etc/rc.local 中执行

```
insmod /ap6256/bcmhdhd.ko firmware_path=/ap6256/fw_bcm43456c5_ag_apsta.bin
nvram_path=/ap6256/nvram_ap6256.txt

sleep 10

#ble

i2cset -f -y 0 0x51 0x0d 0x80 b

sleep 1

brcm_patchram_plus -d --enable_hci --no2bytes --tosleep 200000 --baudrate 1000000
--patchram /ap6256/AP6256.hcd /dev/ttySTM3 &

sleep 20

hciconfig hci0 up
```

3.2 STA 模式

配置文件为/etc/wpa_supplicant.conf，ssid 为要连接的热点名称，psk 为热点密码，

```
root@stm32mp2:~# vi /etc/wpa_supplicant.conf
```

热点信息修改后执行以下自动连接热点。

```
root@stm32mp2:~# wifi_sta_ap.sh sta
```

```

root@stm32mp2:~#
root@stm32mp2:~#
root@stm32mp2:~# wifi_sta_ap.sh sta
[12457.804311] [dhd] [wlan0] wl_cfg80211_disconnect : Reason 3, act 1, bssid b8:f8:83:c3:8b:ca
[12457.815115] [dhd] [wlan0] wl_fw_event : [1 times] disconnected with b8:f8:83:c3:8b:ca, event 11, reason 8
[12457.816105] [dhd] [wlan0] wl_handle_link_down : Link down: WLC_E_LINK(16), reason 2 from b8:f8:83:c3:8b:ca
[12457.819222] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_DISASSOC(11), reason 8
[12457.829373] [dhd] [wlan0] wl_handle_link_down : Disconnect event sent to upper layer:event=16 e->reason=33554432 reason=2 ie_len=0 loc_gen=1 from b8:f8:83:c3:8b:ca
[12457.839546] [dhd] [wlan0] wl_fw_event : Link Down with b8:f8:83:c3:8b:ca, reason=2
[12457.861313] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_LINK(16), reason 2
[12457.871143] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_DEAUTH(5), reason 7
[12458.057984] [dhd] [wlan0] dhd_notify_connect_status_sta : Unexpected event:5 in assoc idle state
[12458.058033] [dhd] [wlan0] dhd_pri_stop : TX queue stopped
[12458.058033] [dhd] [wlan0] dhd_stop : Enter
wlan0: CTRL-Event-DISCONNECTED bssid=b8:f8:83:c3:8b:ca reason=2 [12458.068575] [dhd] dhd_stop: ##### called for ifidx=0 #####
locally generated:
[12458.078600] [dhd] dhd_stop: making dhdpub up FALSE
[12458.080244] [dhd] dhd_tcpack_suppress_set: TCP ACK Suppress mode 1 -> mode 0
[12458.087322] [dhd] dhd_tcpack_suppress_set: TCPACK_INFO_MAXNUM=40, TCPDATA_INFO_MAXNUM=40
[12458.095455] [dhd] [wlan0] wl_android_wifi_off : g_wifi_on=1 force_off=1
[12458.102757] [dhd] dhd_wlfc_deinit(0)3992, maintain HOST RXRERODER flag in tv)
[12458.110789] [dhd] dhd_bus_devreset: == Power OFF ==
[12458.114182] [dhd] dhd_bus_stop: making DHD_BUS_DOWN
[12458.118956] [dhd] dhd_bus_devreset: making dhdpub up FALSE
[12458.124418] [dhd] dhd_txglom_enable: enable 0
[12458.129157] [dhd] dhd_bus_devreset: making DHD_BUS_DOWN
[12458.134196] [dhd] wifi_platform_set_power = 0, delay: 10 msec
[12458.139759] [dhd] == PULL WL_REG_ON(-1) Low! ==
[12458.156329] [dhd] wifi_platform_set_power = 0, sleep done: 10 msec
[12458.156964] [dhd] [wlan0] wl_android_wifi_off : out
[12458.161833] [dhd] [wlan0] dhd_stop : Exit
wlan0: do not deauthenticate as part of interface deinit since WOWLAN is enabled
[12458.296670] [dhd] dhd_pri_open : no mutex held
[12458.299196] [dhd] dhd_pri_open : set mutex lock
[12458.303978] [dhd] [wlan0] dhd_open : Enter
[12458.307756] [dhd] Dongle Host Driver, version 101.10.591.68.32 (20240712-1)(429fcb0)
[12458.307756] /home/yqar/stm32mp2/sdk_v24.11.06/linux-6.6.48/linux-6.6.48/drivers/net/wireless/ap6236 compiled on Jun 5 2025 at 09:42:47
[12458.307756] [dhd] dhd_open: ##### called for ifidx=0 #####
[12458.335507] [dhd] [wlan0] wl_android_wifi_on : in g_wifi_on=0
[12458.341209] [dhd] wifi_platform_set_power = 1, delay: 200 msec

```

```

[12531.029323] [dhd] [wlan0] wl_fw_event : Link UP with b8:f8:83:c3:8b:ca
[12531.030409] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link UP with b8:f8:83:c3:8b:ca
[12531.032598] [dhd] [wlan0] wl_bss_connect_done : Report connect result - connection succeeded
[12531.051377] [dhd] CFG80211-ERROR! wl_mkeep_alive_update : Get mkeep_alive failed (error=-2)
wlan0: Associated with b8:f8:83:c3:8b:ca
wlan0: CTRL-Event-SUBNET-STATUS-UPDATE status=0
[12531.139348] [dhd] [wlan0] wl_add_keyext : key index (0) for b8:f8:83:c3:8b:ca
wlan0: WPA: Key negotiation completed with b8:f8:83:c3:8b:ca [PTK=CCMP GTK=TKIP]
wlan0: CTRL-Event[12531.164311] [dhd] [wlan0] wl_cfg80211_set_suspend_bcn_li_dtim : bcn_li_dtim:0 lpas:0 b
T-CONNECTED - Connection to b8:f8:83:c3:8b:ca completed [id=0 id_str=]
udhcpc: started, v1.36.1
Dropped protocol specifier '.udhcpc' from 'wlan0.udhcpc'. Using 'wlan0' (ifindex=7).
udhcpc: broadcasting discover
udhcpc: broadcasting select for 192.168.2.101, server 192.168.2.1
udhcpc: lease of 192.168.2.101 obtained from 192.168.2.1, lease time 7200
/etc/udhcpc.d/50default: Adding DNS 114.114.114.114
/etc/udhcpc.d/50default: Adding DNS 8.8.8.8
Dropped protocol specifier '.udhcpc' from 'wlan0.udhcpc'. Using 'wlan0' (ifindex=7).
root@stm32mp2:~# ifconfig wlan0
wlan0: Link encap:Ethernet Hwaddr 9C:B8:B4:35:D2:6A
       inet addr:192.168.2.101 Bcast:192.168.2.255 Mask:255.255.255.0
       inet6 addr: fe80::9eb8:b4ff:fe35:d26a/64 Scope:Link
       UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
       RX packets:1546 errors:0 dropped:3837 overruns:0 frame:0
       TX packets:1986 errors:0 dropped:0 overruns:0 carrier:0
       collisions:0 txqueuelen:1000
       RX bytes:169478 (165.5 KiB) TX bytes:354142 (345.8 KiB)

root@stm32mp2:~#

```

Wifi ip 地址为 192.168.2.154

wlan0 ping 网关测试：：

```
root@stm32mp2:~# ping -I wlan0 192.168.2.1
```

如下表示板子与网关通信正常。

```

root@stm32mp2:~#
root@stm32mp2:~#
root@stm32mp2:~#
root@stm32mp2:~# ping -I wlan0 192.168.2.1
PING 192.168.2.1 (192.168.2.1) from 192.168.2.154 wlan0: 56(84) bytes of data.
64 bytes from 192.168.2.1: icmp_seq=1 ttl=64 time=4.41 ms
64 bytes from 192.168.2.1: icmp_seq=2 ttl=64 time=7.95 ms
64 bytes from 192.168.2.1: icmp_seq=3 ttl=64 time=12.9 ms
64 bytes from 192.168.2.1: icmp_seq=4 ttl=64 time=8.12 ms
64 bytes from 192.168.2.1: icmp_seq=5 ttl=64 time=11.4 ms

```

网速测试：

Pc 端安装下载 iperf3.exe，且 pc 的 wifi 与板子 wifi 在同一个局域网内。

如 D:\iperf-3.1.3-win64\存在 iperf3.exe 搜索框输入 cmd 进入命令提示符



进入 D:\iperf-3.1.3-win64

d:

cd iperf-3.1.3-win64

```
Microsoft Windows [版本 10.0.22631.5189]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\10274>d:

D:\>cd iperf-3.1.3-win64

D:\iperf-3.1.3-win64>
```

板子端运行 iperf3 作为 server

```
root@stm32mp2:~# iperf3 -s
warning: this system does not seem to support IPv6 - trying IPv4
-----
Server listening on 5201 (test #1)
-----
```

Pc 端作为 client

```
iperf3.exe -c 192.168.2.154
```

```
D:\iperf-3.1.3-win64>iperf3.exe -c 192.168.2.154
Connecting to host 192.168.2.154, port 5201
[ 4] local 192.168.2.124 port 53232 connected to 192.168.2.154 port 5201
[ ID] Interval      Transfer    Bandwidth
[ 4]  0.00-1.01    sec  1.00 MBytes  8.31 Mbits/sec
[ 4]  1.01-2.00    sec   512 KBytes  4.23 Mbits/sec
[ 4]  2.00-3.04    sec  1.88 MBytes 15.2 Mbits/sec
[ 4]  3.04-4.02    sec  1.25 MBytes 10.6 Mbits/sec
[ 4]  4.02-5.02    sec  1.25 MBytes 10.5 Mbits/sec
[ 4]  5.02-6.01    sec  1.50 MBytes 12.6 Mbits/sec
[ 4]  6.01-7.01    sec  1.25 MBytes 10.5 Mbits/sec
[ 4]  7.01-8.01    sec  1.00 MBytes  8.36 Mbits/sec
[ 4]  8.01-9.02    sec  1.38 MBytes 11.5 Mbits/sec
[ 4]  9.02-10.01   sec  1.25 MBytes 10.6 Mbits/sec
-----
[ ID] Interval      Transfer    Bandwidth
[ 4]  0.00-10.01   sec  12.2 MBytes 10.3 Mbits/sec
[ 4]  0.00-10.01   sec  12.0 MBytes 10.1 Mbits/sec
sender
receiver
```



```

rx bytes:23942 (23.3 Kib)  tx bytes:7780 (7.6 Kib)
root@stm32mp2:~# iperf3 -s
warning: this system does not seem to support IPV6 - trying IPV4
-----
Server listening on 5201 (test #1)
-----
Accepted connection from 192.168.2.124, port 53230
[ 5] local 192.168.2.154 port 5201 connected to 192.168.2.124 port 53232
[ ID] Interval           Transfer             Bitrate
[ 5] 0.00-1.00      sec    768 KBytes        6.28 Mbits/sec
[ 5] 1.00-2.00      sec    640 KBytes        5.25 Mbits/sec
[ 5] 2.00-3.00      sec    1.75 MBytes       14.7 Mbits/sec
[ 5] 3.00-4.00      sec    1.50 MBytes       12.6 Mbits/sec
[ 5] 4.00-5.00      sec    1.00 MBytes        8.39 Mbits/sec
[ 5] 5.00-6.00      sec    1.75 MBytes       14.7 Mbits/sec
[ 5] 6.00-7.00      sec    1.00 MBytes        8.40 Mbits/sec
[ 5] 7.00-8.00      sec    1.00 MBytes        8.39 Mbits/sec
[ 5] 8.00-9.00      sec    1.50 MBytes       12.6 Mbits/sec
[ 5] 9.00-10.00     sec    1.12 MBytes        9.45 Mbits/sec
-----
[ ID] Interval           Transfer             Bitrate
[ 5] 0.00-10.06     sec   12.0 MBytes       10.0 Mbits/sec
-----
Server listening on 5201 (test #2)
-----

```

3.3 AP 模式

Wifi 作为 ap 时，需要 hostapd 使能 wifi 发出热点信息和 dhcpd 服务用于自动分配 ip 地址。

/etc/hostapd.conf 为 hostapd 配置文件，包含 ssid 密码 channel 等信息，默认热点名称为 lierda-ic610，密码为 12345678。

/etc/kea/kea-dhcp4.conf 为 dhcpd 服务子网配置文件，现在默认子网为 192.168.1.1 。

```
root@stm32mp2:~# wifi_sta_ap.sh ap
```

```

root@stm32mp2:~#
root@stm32mp2:~# wifi_sta_ap.sh ap
wlan0: Do not deauthenticate as part of interface deinit since WOWLAN is enabled
[12671.320452] [dhd] [wlan0] wl_cfg80211_disconnect : Reason 3, act 1, bssid b8:f8:83:c3:8b:ca
[12671.332481] [dhd] [wlan0] wl_iw_event : [5 times] disconnected with b8:f8:83:c3:8b:ca, event 11, reason 8
[12671.332500] [dhd] [wlan0] wl_handle_link_down : Link down: WLC_E_LINK(16), reason 2 from b8:f8:83:c3:8b:ca
[12671.332564] [dhd] [wlan0] wl_handle_link_down : Disconnect event sent to upper layer event:16 e->reason=33554
[12671.336709] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_DISASSOC(11), rea
[12671.362271] [dhd] CFG80211-ERROR) wl_notify_connect_status_sta : unexpected event:5 in assoc idle state
[12671.370967] [dhd] [wlan0] wl_iw_event : Link Down with b8:f8:83:c3:8b:ca, reason=2
[12671.370999] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_LINK(16), reason
[12671.380559] [dhd] CFG80211-ERROR) wl_notify_connect_status_sta : unexpected event:5 in assoc idle state
[12671.388090] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_DEAUTH(5), reason
[12671.417829] [dhd] [wlan0] wl_ext_iapsta_link : [S] Link down with b8:f8:83:c3:8b:ca, WLC_E_DEAUTH(5), reason
[12671.581990] [dhd] [wlan0] dhd_pri_stop : tx queue stopped
[12671.582040] [dhd] [wlan0] dhd_stop : Enter
[12671.588745] [dhd] dhd_stop: ##### called for ifidx=0 #####
[12671.596656] [dhd] dhd_stop: making dhdpub up FALSE
[12671.596900] [dhd] dhd_tcpack_suppress_set: TCP ACK Suppress mode 1 -> mode 0
[12671.603878] [dhd] dhd_tcpack_suppress_set: TCPACK_INFO_MAXNUM=40, TCPDATA_INFO_MAXNUM=40
[12671.611978] [dhd] [wlan0] wl_android_wifi_off : g_wifi_on=1 force_off=1
[12671.619042] [dhd] dhd_wlfc_deinit():3992, maintain HOST RXRERODER flag in tvl
[12671.627146] [dhd] dhd_bus_devreset: == Power OFF ==
[12671.631275] [dhd] dhd_bus_stop: making DHD_BUS_DOWN
[12671.635553] [dhd] dhd_bus_devreset: making dhdpub up FALSE
[12671.641002] [dhd] dhd_txglom_enable: enable 0
[12671.645351] [dhd] dhd_bus_devreset: making DHD_BUS_DOWN
[12671.650686] [dhd] wifi_platform_set_power = 0, delay: 10 msec
[12671.656246] [dhd] ===== PULL WL_REG_ON(-1) LOW! =====
[12671.672841] [dhd] wifi_platform_set_power = 0, sleep done: 10 msec
[12671.673477] [dhd] [wlan0] wl_android_wifi_off : out
[12671.678350] [dhd] [wlan0] dhd_stop : Exit
wlan0: CTRL-EVENT-DSPP-POLICY clear_all
nl80211: deinit ifname=wlan0 disabled_11b_rates=0
wlan0: CTRL-EVENT-TERMINATING
[12671.756964] [dhd] dhd_pri_open : no mutex held
[12671.757012] [dhd] dhd_pri_open : set mutex lock

```



若需要修改子网网段，需要修改 /etc/kea/kea-dhcp4.conf 子网信息及 /usr/sbin/wifi_sta_ap.sh 中的 wlan0 网关地址，网关 wlan0 的 ip 必须与 /etc/kea/kea-dhcp4.conf 中设置的 ip 地址保持一致。

4 网络故障排查

1、ping ip 地址正常，ping 网站不同

问题原因：DNS 解析问题

解决办法：/etc/resolv.conf 内手动添加

```
nameserver 114.114.114.114
```

```
nameserver 8.8.8.8
```

2、pc ping 板子正常，板子 ping pc 无法 ping 通

问题原因：电脑防火墙问题

解决办法：关闭 pc 端防火墙

3、wifi 热点无法启动

检测子网配置与 wlan0 地址是否属于同一个子网。

Lierda
利 尔 达