Rockchip Linux Network Config Documentation

发布版本：1.0

作者：CTF

日期：2019.4.29

文件密级：公开资料

概述

该文档旨在介绍Rockchip Linux各种配网方式。

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

对应DeviceIo库版本

V1.2.1以上，不包含V1.2.1

修订记录

日期	版本	作者	修改说明
2019-4-29	V1.0	CTF	初始版本

1、WIFI/BT配置

1.1 kernel配置

- kernel目录下执行`make menuconfig`，根据实际wifi选择相应配置

```
Symbol: WL_ROCKCHIP [=y]
Type  : boolean
Prompt: Rockchip Wireless LAN support
  Location:
    -> Device Drivers
      -> Network device support (NETDEVICES [=y])
(1)     -> Wireless LAN (WLAN [=y])
  Defined at drivers/net/wireless/rockchip_wlan/Kconfig:2
  Depends on: NETDEVICES [=y] && WLAN [=y]
  Selects: WIRELESS_EXT [=y] && WEXT_PRIV [=y] && CFG80211 [=y] && MAC80211
```

```
--------------------- Rockchip Wireless LAN support ----------------------
 Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty
 submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> includes,
 <N> excludes, <M> modularizes features.  Press <Esc><Esc> to exit, <?> for
 Help, </> for Search.  Legend: [*] built-in  [ ] excluded  <M> module  < >
+-----------------------------------------------------------------------------+
|    -*- Rockchip Wireless LAN support                                        |
|    [ ]   build wifi ko modules                                              |
|    [*]   Wifi load driver when kernel bootup                                |
|    [*]   Wifi generate random mac address and save to vendor storage for    |
|    < >   ap6xxx wireless sdio cards support                                 |
|    < >     Cypress wireless sdio cards support                              |
|    [ ]   Realtek Wireless Device Driver Support  ----                       |
|    < >   Realtek 8189F SDIO WiFi                                           |
|    < >   Realtek 8723B SDIO or SPI WiFi                                     |
|    < >   Realtek 8723C SDIO or SPI WiFi                                     |
|    <*>   Realtek 8723D SDIO or SPI WiFi                                     |
|    < >   Marvell 88W8977 SDIO WiFi                                          |
|          SouthSV 6XXX WLAN support  --->                                    |
```

- 退出配置框，make savedefconfig保存配置
- 重新编译kernel

1.2 buildroot配置

- 根目录下执行`make menuconfig`
- rkwifibt配置，根据实际WiFi选择对应配置，必须跟kernel配置一致

```
Symbol: BR2_PACKAGE_RKWIFIBT [=y]
Type  : boolean
Prompt: rkwifibt
  Location:
    -> Target packages
(1)   -> rockchip BSP packages (BR2_PACKAGE_ROCKCHIP [=y])
  Defined at package/rockchip/rkwifibt/Config.in:1
  Depends on: BR2_PACKAGE_ROCKCHIP [=y]
```

```
+---------------------- wifi chip support ----------------------+
|  Use the arrow keys to navigate this window or press the      |
|  hotkey of the item you wish to select followed by the <SPACE |
|  BAR>. Press <?> for additional information about this        |
|  +-------------------------------------------------------+    |
|  |       ( ) AP6181                                      |    |
|  |       ( ) AP6255                                      |    |
|  |       ( ) AP6212A1                                    |    |
|  |       ( ) AP6354                                      |    |
|  |       ( ) AP6236                                      |    |
|  |       (X) AW-CM256                                    |    |
|  +----------v(+)-----------------------------------------+    |
+---------------------------------------------------------------+
|                    <Select>      < Help >                   |
+---------------------------------------------------------------+
```

- 蓝牙配置
  - realtek模组建议使用bluez 协议，正基/海华模组建议使用bsa 协议。
  - 以下配置，根据模组类型三选一
    - realtek模组选择：`bluez-utils 5.x`，使用bluez需要同时开启`bluez-alsa` `readline`

```
Symbol: BR2_PACKAGE_BLUEZ5_UTILS [=y]
Type  : boolean
Prompt: bluez-utils 5.x
  Location:
    -> Target packages
(2)   -> Networking applications
  Defined at package/bluez5_utils/Config.in:1
  Depends on: BR2_USE_WCHAR [=y] && BR2_TOOLCHAIN_HAS_THREADS [=y] && BR2_U
  Selects: BR2_PACKAGE_DBUS [=y] && BR2_PACKAGE_LIBGLIB2 [=y]
  Selected by: BR2_PACKAGE_BLUEZ_ALSA [=y] && !BR2_STATIC_LIBS [=n] && !BR2
```

```
    [ ] bcusdk
    [ ] bind
    [ ] bluez-tools
    [ ] bluez-utils
    -*- bluez-utils 5.x
    [ ]   build OBEX support
    [*]   build CLI client
    [ ]     install GATT tool
    [ ]   build experimental plugins
    [ ]   build sixaxis plugin
    [ ]   build tests
    [ ] bmon
```

```
Symbol: BR2_PACKAGE_BLUEZ_ALSA [=y]
Type  : boolean
Prompt: bluez-alsa
  Location:
    -> Target packages
(9)   -> Audio and video applications
  Defined at package/rockchip/bluez-alsa/Config.in:1
  Depends on: !BR2_STATIC_LIBS [=n] && !BR2_PACKAGE_BLUEZ_UTILS [=n] && BR2
  Selects: BR2_PACKAGE_ALSA_LIB [=y] && BR2_PACKAGE_BLUEZ5_UTILS [=y] && BR
```

```
    [*] alsa-utils  --->
    [*] alsa-plugins  ----
    [ ] atest
    [ ] aumix
    [ ] bellagio
    [*] bluez-alsa
    [*]   hcitop
    [ ] dvblast
    [ ] dvdauthor
    [ ] dvdrw-tools
    [ ] espeak
    -*- faad2
```

```
Symbol: BR2_PACKAGE_READLINE [=y]
Type  : boolean
Prompt: readline
  Location:
    -> Target packages
      -> Libraries
(7)     -> Text and terminal handling
  Defined at package/readline/Config.in:1
  Selects: BR2_PACKAGE_NCURSES [=y]
  Selected by: BR2_PACKAGE_BLE_WIFICONFIG [=n] && BR2_PACKAGE_ROCKCHIP [=y]
```

```
    -*-   UTF-8/16/32 support in pcre
    -*-   Unicode properties support in pcre
    [ ] pcre2
    -*- popt
    -*- readline
    [ ] slang
    [ ] tclap
    [ ] ustr
```

- 正基模组选择： `broadcom(ampak) bsa server and app`

  进入 `wifi/bt chip support(XXX)--->` 选择实际的芯片型号，必须跟rkwifibt配置一致

- 海华模组选择 ： `broadcom(cypress) bsa server and app`

  进入 `wifi/bt chip support(XXX)--->` 选择实际的芯片型号，必须跟rkwifibt配置一致

```
                        rockchip BSP packages
Arrow keys navigate the menu.  <Enter> selects submenus ---> (or empty
submenus ----).  Highlighted letters are hotkeys.  Pressing <Y> selects a
feature, while <N> excludes a feature.  Press <Esc><Esc> to exit, <?> for
Help, </> for Search.  Legend: [*] feature is selected  [ ] feature is
----^(-)-------------------------------------------------------------------+
    [ ]   linux-serial-test                                                |
    [ ]   Simple iflytek voice process and cloud SDK                       |
    [*]   Equalizer and DRC process                                        |
    [*]   alsa plugin ladspa                                               |
    [ ]   stress test tools                                                |
    [ ]   rockchip modules                                                 |
    [ ]   broadcom(ampak) bsa server and app                               |
    [*]   broadcom(cypress) bsa server and app                             |
            wifi/bt chip support (AW-CM256)  --->                          |
    [ ]   pm suspend api & demo                                            |
    [ ]   realtek simple config                                            |
    [ ]   Rockchip recovery for linux                                      |
    [*]   Rockchip OTA update for linux                                    |
    [ ]   Rockchip ueventd for linux                                       |
    [ ]   Rockchip rkupdate for linux                                      |
----v(+)-------------------------------------------------------------------+
-----------------------------------------------------------------------------
        <Select>    < Exit >    < Help >    < Save >    < Load >
```

正基模组

海华模组

- 退出配置框，make savedefconfig保存配置

1.3 编译说明

- 根目录下执行：`make rkwifibt-dirclean && make rkwifibt-rebuild`
- 以下编译选项，根据模组类型三选一
  - realtek模组编译：`make bluez5_utils-rebuild`

    `make bluez-alsa-rebuild`
  - 正基模组编译：`make broadcom_bsa-rebuild`
  - 海华模组编译：`make cypress_bsa-rebuild`
- 根目录下执行：`make deviceio-dirclean && make deviceio-rebuild`
- 根目录下执行：`make`
- 打包固件：`./mkfirmware.sh`

2、命令行配网

- 首先确保WiFi的服务进程启动，串口输入：`ps | grep wpa_supplicant`

```
# ps | grep wpa_supplicant
  532 root       3380 S    wpa_supplicant -B -i wlan0 -c /data/cfg/wpa_supplica
  618 root       1836 R    grep wpa_supplicant
```

- 如果没启动，请手动启动：

  `wpa_supplicant -B -i wlan0 -c /data/cfg/wpa_supplicant.conf &`
- 修改`/data/cfg/wpa_supplicant.conf`文件，添加配置项

```
network={
        ssid="WiFi-AP"      // WiFi名字
        psk="12345678"      // WiFi密码
        key_mgmt=WPA-PSK    // 选填加密方式，不填的话可以自动识别
        # key_mgmt=NONE     // 不加密
}
```

- 重新读取上述配置：`wpa_cli reconfigure`
- 重新连接：`wpa_cli reconnect`

3、手机配网

3.1 ble 配网

- 简介

  目前ble配网已经集成到deviceio，接口位于RkBle.h。同时支持bluez ble配网和bsa ble配网，配置参照本文档的第一章节'WIFI/BT 配置'。
- 接口说明

  请参考/external/deviceio/doc目录下Rockchip_Developer_Guide_Rk3308_DeviceIo_Bluetooth_CN.pdf文档的第二章节'BLE接口介绍（RkBle.h）'。
- APP：Rkble.apk
- 配网步骤

- 该配网步骤以bsa ble配网为例进行说明，所有板端log均为bsa的配网log。bluez操作步骤相同，板端log不同。
- 确保wifi server进程启动 ，板端命令行执行：

  `wpa_supplicant -B -i wlan0 -c /data/cfg/wpa_supplicant.conf &`
- 板端命令行执行：`deviceio_test blewifi` 启动ble 配网，设置的ble广播设备名必须以RockChip为前缀，否则Rkble.apk无法检索到设备

```
BSA_trace 17@ 01/01 08h:03m:10s:758ms: BSA_DmSetConfigInit
DEBUG: app_mgr_set_bt_config: Enable:1
DEBUG: app_mgr_set_bt_config: Discoverable:1
DEBUG: app_mgr_set_bt_config: Connectable:1
DEBUG: app_mgr_set_bt_config: Name:RockChip
DEBUG: app_mgr_set_bt_config: Bdaddr cc:b8:a8:bf:ac:d5
DEBUG: app_mgr_set_bt_config: ClassOfDevice:00:04:24
DEBUG: app_mgr_set_bt_config: First host disabled channel:79
DEBUG: app_mgr_set_bt_config: Last host disabled channel:79
BSA_trace 18@ 01/01 08h:03m:10s:761ms: BSA_DmSetConfig
BSA_trace 19@ 01/01 08h:03m:10s:874ms: BSA_TmReadVersionInit
BSA_trace 20@ 01/01 08h:03m:10s:874ms: BSA_TmReadVersion
INFO: app_mgr_read_version: Server status:0
INFO: app_mgr_read_version: FW Version:3.1.25.269
INFO: app_mgr_read_version: BSA Server Version:BSA0107_00.26.00
```

```
>>>>> Start ble ....
DEBUG: app_ble_rk_server_open: app_ble_rk_server_open
[RK] ble status: RK_BLE_STATE_IDLE
INFO: app_ble_start: app_ble_start
BSA_trace 23@ 01/01 08h:03m:13s:879ms: BSA_BleEnableInit
BSA_trace 24@ 01/01 08h:03m:13s:879ms: BSA_BleEnable
DEBUG: app_ble_rk_server_set_device_name: app_ble_device_name: RockChipBle
INFO: app_ble_rk_server_gatt_server_init: wifi_introducer_gatt_server_init
BSA_trace 25@ 01/01 08h:03m:13s:881ms: BSA_BleSeAppRegisterInit
BSA_trace 26@ 01/01 08h:03m:13s:881ms: BSA_BleSeAppRegister
INFO: app_ble_rk_server_register: server_if:4
```

- 手机端打开apk：

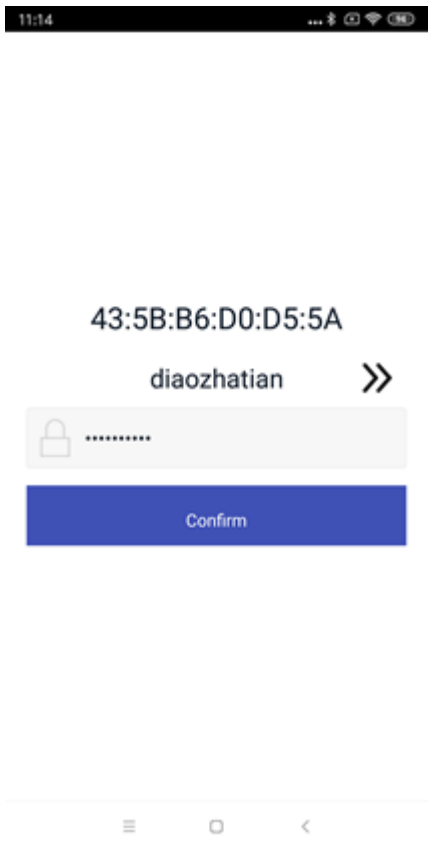
  点击CONTINUE -> START SCAN，扫描以RockChip为前缀命名的ble设备

- 点击想要连接的ble设备，开始连接设备，设备连接成功，板端log如下：

```
INFO: app_ble_rk_server_profile_cback: BSA_BLE_SE_OPEN_EVT status:0
INFO: app_ble_rk_server_profile_cback: app_ble_rk_server_conn_up conn_id:0x4
INFO: app_ble_rk_server_profile_cback: app_ble_rk_server_conn_up connected to [74:B6
:64:CC:DA:32]
DEBUG: app_dm_set_ble_visibility: Set BLE Visibility Discoverable:0 Connectable:0
BSA_trace 42@ 01/01 08h:12m:05s:210ms: BSA_DmSetConfigInit
BSA_trace 43@ 01/01 08h:12m:05s:210ms: BSA_DmSetConfig
[RK] ble status: RK_BLE_STATE_CONNECT
INFO: app_ble_rk_server_profile_cback: Stopping Advertisements
BSA_trace 44@ 01/01 08h:12m:05s:212ms: bsa_sec_event_hdlr event:0
DEBUG: app_mgr_security_callback: event:0
DEBUG: app_mgr_security_callback: BSA_SEC_LINK_UP_EVT bd_addr: 74:b6:64:cc:da:32
DEBUG: app_mgr_security_callback: ClassOfDevice:00:00:00 => Misc device
DEBUG: app_mgr_security_callback: LinkType: 2
DEBUG: bt_mgr_notify_callback: BT_LINK_UP_EVT

DEBUG: app_mgr_write_remote_devices: app_xml_write_db ok
```

- 设备连接成功，apk进入配网界面，输入ssid和psk，点击Confirm，发送配网信息



- 板端接收到ssid和psk后，启动配网

```
[RK] ble_data.cmd: wifisetup, ble_data.start: 1, ble_data.end: 4
01-01 08:06:50.057  2514  2542 D [RK] wifi ssid is diaozhatian
01-01 08:06:50.058  2514  2542 D [RK] wifi psk is 7788123456
[RK] rk_config_wifi_thread
[RK] controlWifi connect ...
[RKWIFI] exec1: wpa_cli -iwlan0 disable_network all
[RKWIFI] exec1: wpa_cli -iwlan0 add_network
format_wifiinfo ssid: 6469616f7a68617469616e
[RKWIFI] exec1: wpa_cli -iwlan0 set_network 0 ssid 6469616f7a68617469616e
format_wifiinfo password: \7\7\8\8\1\2\3\4\5\6
[RKWIFI] exec1: wpa_cli -iwlan0 set_network 0 psk \"\7\7\8\8\1\2\3\4\5\6\"
01-01 08:06:51.211  2514  2971 I RK_wifi_connect ssid:"diaozhatian" strlen(ssid):11;
 ori:"diaozhatian" strlen(ori):11; psk:"7788123456"
```

- 配网成功

```
wifi is connected.
OK
OK
[RK] rk_blewifi_state_callback state: 4
```

3.2 airkiss 配网

- 简介

  目前微信airkiss配网只支持realtek，请参照本文档第一章节 'WIFI/BT 配置'，正确配置kernel和rkwifibt；并且已集成到deviceio_test中。

- kernel 修改

  修改 `/drivers/net/wireless/rockchip_wlan/rtl8723ds/Makefile` 文件

  ```
  -CONFIG_WIFI_MONITOR = n
  +CONFIG_WIFI_MONITOR = y
  ```

- 接口说明

  `int RK_wifi_airkiss_config(char *ssid, char *password)`

  启动airkiss配网，并通过ssid、password参数返回手机端传输的wifi名称和密码，成功返回0，失败返回-1

- 示例程序

  示例程序的路径为：`external/deviceio/test/rk_wifi_test.c`

  该测试用例调用`RK_wifi_airkiss_config()`启动airkiss，获取ssid和password并启动wifi配网。

  主要接口：`void rk_wifi_airkiss()`， 在DeviceIOTest.cpp中调用。

- 配网步骤
  - 确保wifi server进程启动 ，命令行执行：

    `wpa_supplicant -B -i wlan0 -c /data/cfg/wpa_supplicant.conf &`
  - 手机必须开启wifi，并连接网络，微信扫描二维码，进入网络配置界面

    
  - 选择 '配置设备上网'，输入手机当前连接wifi的密码，点击连接

- 板端命令行执行：`deviceio_test airkiss` 启动airkiss 配网
  - airkiss 启动成功可以看到如下log

```
version:V1.2.0
===== rk_wifi_airkiss =====
=== RK_wifi_rtl_airkiss_config ===
rm: can't remove '/tmp/airkiss.conf': No such file or directory
killall: rk_airkiss: no process killed
airkiss key: h1oYDCkSsDHw4LbL
probe_chplan[0]:        ch=1,    bw_offset=1
probe_chplan[1]:        ch=6,    bw_offset=2
probe_chplan[2]:        ch=11,   bw_offset=2
probe_chplan[3]:        ch=2,    bw_offset=1
probe_chplan[4]:        ch=3,    bw_offset=1
probe_chplan[5]:        ch=4,    bw_offset=1
probe_chplan[6]:        ch=5,    bw_offset=2
probe_chplan[7]:        ch=7,    bw_offset=2
probe_chplan[8]:        ch=8,    bw_offset=2
probe_chplan[9]:        ch=9,    bw_offset=2
probe_chplan[10]:       ch=10,   bw_offset=2
probe_chplan[11]:       ch=12,   bw_offset=2
probe_chplan[12]:       ch=13,   bw_offset=2
Start airkiss!
Airkiss airkiss_set_key succeed!
Airkiss init succeed!
killall: wpa_supplicant: no process killed
[AIRKISS] start rk_airkiss success, cnt: 2
[AIRKISS] check airkiss conf, wait_cnt: 59
enabled wifi monitor mode.
```

  - 成功接收ssid和password，并开始配网

```
AirKiss decrypt pwd: length 16, pwd A0 B6 08 55 A4 64 69 F7 12 5A 0E 1D
AirKiss complete: ssid "diaozhatian", pwd "7788123456", random 0xb8
AIRKISS_STATUS_COMPLETE
airkiss_get_result() ok!
ssid = "diaozhatian", pwd = "7788123456", ssid_length = 11, "pwd_length
 = 0xb8
[AIRKISS] check airkiss conf, wait_cnt: 21
killall: wpa_supplicant: no process killed
[AIRKISS] check airkiss conf, wait_cnt: 20
[AIRKISS] geted airkiss data
[AIRKISS] ssid ret_buf: rk_ssid=diaozhatian

[AIRKISS] password ret_buf: rk_password=7788123456

[AIRKISS] SSID: diaozhatian[11], PSK: 7788123456[10]
[RK_AIRKISS] rk_config_wifi_thread
[RK_AIRKISS] rk_wifi_airkiss_state_callback state: 1
[RKWIFI] exec1: wpa_cli -iwlan0 disable network all
```

- 配网成功

```
wifi is connected.
OK
OK
[RK_AIRKISS] rk_wifi_airkiss_state_callback state: 4
[RK_AIRKISS] RK_WIFI_State_CONNECTED
sending commands to master dhcpcd process
21 Jan 21:12:26 ntpd[375]: Listen normally on 2 wlan0 192.168.31.123:12
Check wifi state with none state. try more 1/50,
Congratulation: wifi connected.
Selected interface 'wlan0'
OK
Selected interface 'wlan0'
OK
airkiss_confirm_connected
```

3.3 Softap 配网

- 简介

  首先，用SDK板的WiFi创建一个AP热点，在手机端连接该AP热点；其次，通过手机端apk获取SDK板的当前扫描到的热点列表，在手机端填入要连接AP的密码，apk会把AP的ssid和密码发到SDK板端；最后，SDK板端会根据收到的信息连接WiFi。

  目前Softap还未集成到deviceio_test中，后续会进一步更新！！！

- APP: /external/app/RkEcho.apk
- buildroot配置

```
Type  : boolean
Prompt: softap mode to setup wifi
  Location:
    -> Target packages
(1)   -> rockchip BSP packages (BR2_PACKAGE_ROCKCHIP [=y])
  Defined at package/rockchip/softap/Config.in:1
  Depends on: BR2_PACKAGE_ROCKCHIP [=y]
  Selected by: BR2_PACKAGE_SOFTAPSERVER [=y] && BR2_PACKAGE_ROCKCHIP [=y]


Symbol: BR2_PACKAGE_SOFTAPSERVER [=y]
Type  : boolean
Prompt: socket server based on softap
  Location:
    -> Target packages
(2)   -> rockchip BSP packages (BR2_PACKAGE_ROCKCHIP [=y])
  Defined at package/rockchip/softapServer/Config.in:1
  Depends on: BR2_PACKAGE_ROCKCHIP [=y]
  Selects: BR2_PACKAGE_SOFTAP [=y]
```

- 源码开发目录

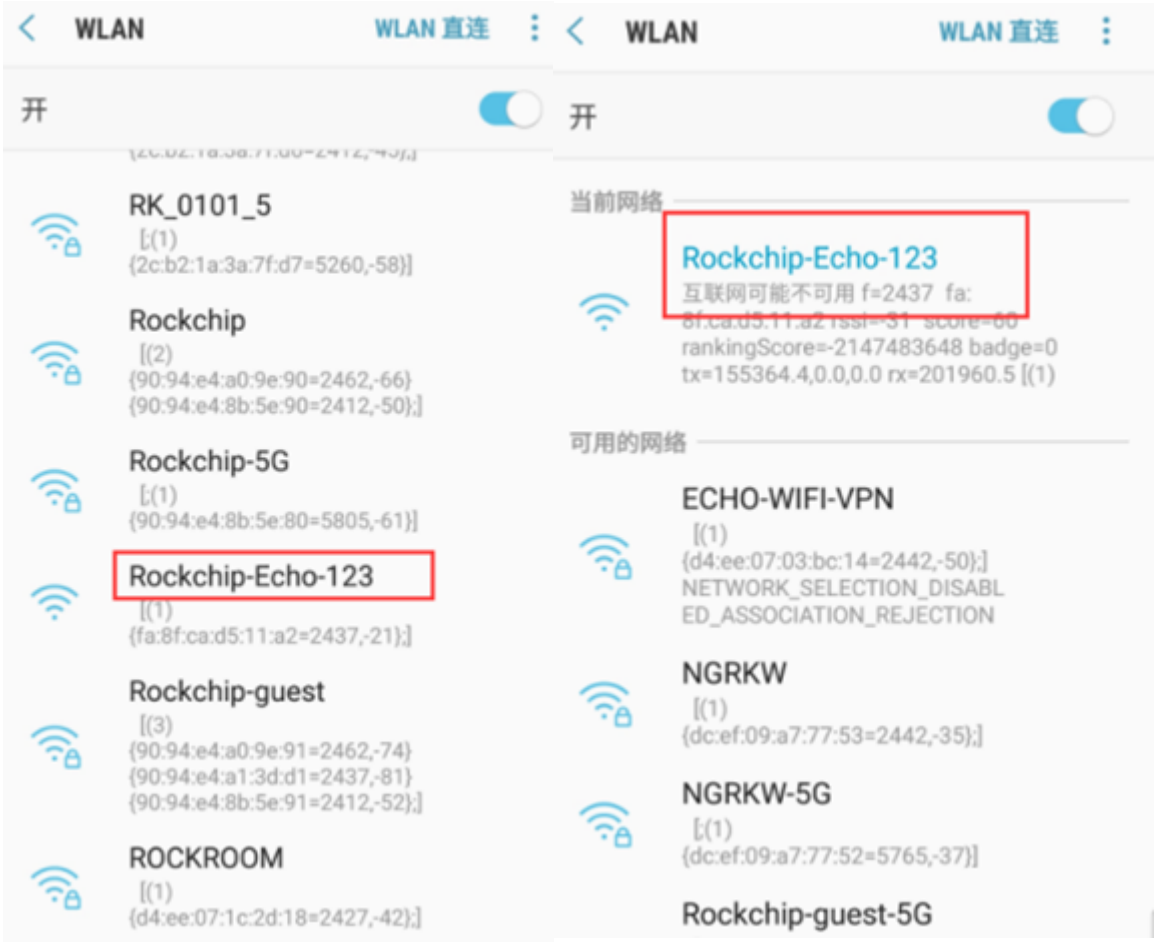
  wifi与apk端相关操作：/external/softapServer

  wifi相关操作：/external/softapDemo

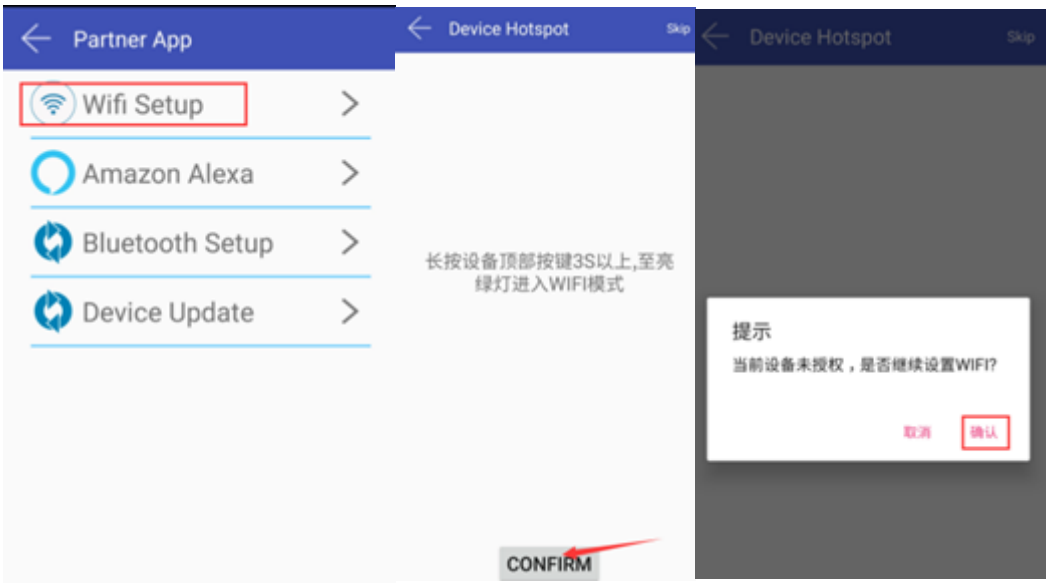
- 配网步骤
  - 确保wifi server进程启动 ，命令行执行：

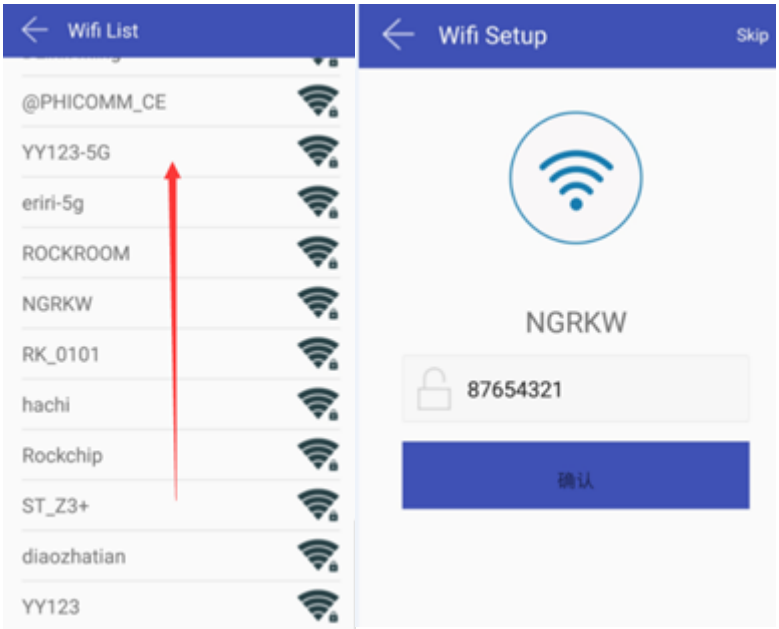
    `wpa_supplicant -B -i wlan0 -c /data/cfg/wpa_supplicant.conf &`
  - 板端命令行执行：

    `softapServer Rockchip-Echo-123`(wifi热点的名字，前缀必须为Rockchip-Echo-xxx)
  - 打开手机wifi setting界面，找到Rockchip-Echo-xxx , 点击连接 ：



- 打开apk，点击wifi setup->CONFIRM->确认->wifi列表->点击你要连接的网络名字->输入密码->点击确认





- 板子串口端显示

- 检查网络是否连通
  - 添加dns域名解析：`echo nameserver 8.8.8.8 > etc/resolv.conf`
  - 看下是否ping通：`ping www.baidu.com`

- 注意要点
  - softspServer Rockchip-Echo-123 执行后命令行是无法退出的，直到配网完成
  - 热点名千万不要写错，否则apk无法进入确认界面（Rockchip-Echo-xxx）