

Realtek Wi-Fi SDK for Android ICS

ver. 1.0.1

Contents

Release History	1
SDK packages	1
Introduction.....	2
1. Copy Necessary Files into SDK	2
2. Platform Related Files.....	2
2.1. BoardConfig.mk.....	2
2.2. init.xxx.rc	4
3. libhardware_legacy	5
3.1. Include wifi_realtek.c into source files	5
4. netd.....	5
4.1. Apply hostapd-0.6.9_rtl	5
4.2. Include SoftapController_realtek.cpp into source files	6
4.3. Include libhostapd_client into static libraries	6
5. Opening Portable Wi-Fi Hotspot UI	6

Release History

ver. 1.0.0	1. Support STA and SoftAP mode for Android 4.0 (ICS)
ver. 1.0.1	1. Correct the document error in 4.2 Include SoftapController_realtek.cpp into source files

SDK packages

- hardware/realtek/
Folder to store config file, private code from Realtek.
- external/hostapd-0.6.9_rtl/
A patched version hostapd provided from Realtek.
- device/ti/panda/BoardConfig.mk
- device/ti/panda/init.omap4pandaboard.rc
Reference codes for platform related files, which is retrieved from panda board and has been patched with Realtek Wi-Fi SDK
- hardware/libhardware_legacy/wifi/Android.mk
- system/netd/Android.mk

Reference codes for applying wifi_realtek.c and SoftapController_realtek.cpp.

- frameworks/base/core/res/res/values/config.xml

Reference codes for setting the two system resource configurations:
config_tether_wifi_regexs and config_tether_upstream_types

Introduction

This document provides a simple guide to help engineers to apply Realtek Wi-Fi solution onto their Android 4.0 (ICS) system. For now, we have supported the following Wi-Fi functionality:

- Standard STA mode
- Portable Wi-Fi Hotspot(SoftAP mode)

To port Realtek Wi-Fi driver onto Android platform, you can go through the following guide with reference codes within our driver package's realtek_wifi_SDK_for_android_ICS_20111229.tar.gz.

Because Android's SDK may differ from platform to platform, our reference codes may not be applied on every platform without modifications. You should check if our reference code is suitable for you to use.

1. Copy Necessary Files into SDK

You need to copy the following folder into your target Android ICS SDK folder:

- hardware/realtek/
- external/hostapd-0.6.9_rtl/

2. Platform Related Files

2.1. BoardConfig.mk

To apply Realtek Wi-Fi solution onto your Android ICS system, define the following compile-time variables in BoardConfig.mk of your platform:

```

BOARD_WIFI_VENDOR := realtek
ifeq ($(BOARD_WIFI_VENDOR), realtek)
    WPA_SUPPLICANT_VERSION := VER_0_8_X
    BOARD_WPA_SUPPLICANT_DRIVER := WEXT
    BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_wext_rtl

    #BOARD_HOSTAPD_DRIVER      :=
    #BOARD_HOSTAPD_PRIVATE_LIB :=

    BOARD_WLAN_DEVICE := rtl8192cu
    #BOARD_WLAN_DEVICE := rtl8192du
    #BOARD_WLAN_DEVICE := rtl8192ce
    #BOARD_WLAN_DEVICE := rtl8192de
    #BOARD_WLAN_DEVICE := rtl8723as
    #BOARD_WLAN_DEVICE := rtl8723au
    #BOARD_WLAN_DEVICE := rtl8188es

    WIFI_DRIVER_MODULE_NAME    := wlan
    WIFI_DRIVER_MODULE_PATH    := "/system/lib/modules/wlan.ko"
    WIFI_DRIVER_MODULE_ARG     := ""

    WIFI_FIRMWARE_LOADER      := ""
    WIFI_DRIVER_FW_PATH_STA    := ""
    WIFI_DRIVER_FW_PATH_AP     := ""
    WIFI_DRIVER_FW_PATH_P2P    := ""
    WIFI_DRIVER_FW_PATH_PARAM := ""
endif

```

- **BOARD_WIFI_VENDOR := realtek**

To distinguish and group the platform Wi-Fi device from products of other companies, we define variable BOARD_WIFI_VENDOR as realtek. This is for compile-time choices to be applied for Realtek Wi-Fi solutions.

- **WPA_SUPPLICANT_VERSION := VER_0_8_X**

For Android ICS, please set WPA_SUPPLICANT_VERSION as VER_0_8_X to use wpa_supplicant_8.

- **BOARD_WPA_SUPPLICANT_DRIVER := WEXT**

- **BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_wext_rtl**

For now, we use wext as the driver interface for wpa_supplicant to communicate with driver and provide lib_driver_cmd_wext_rtl as the private processing library.

TODO: we plan to shift the driver interface to nl80211 for Android ICS's Wi-Fi-Direct (Wi-Fi P2P) framework.

- **BOARD_HOSTAPD_DRIVER**
- **BOARD_HOSTAPD_PRIVATE_LIB**

As for Android Hotspot (SoftAP) functionality, now, we provide our own hostapd-0.6.9_rtl package and there is no need to set BOARD_HOSTAPD_DRIVER and BOARD_HOSTAPD_PRIVATE_LIB.

- **BOARD_WLAN_DEVICE**

Realtek provide a variety of Wi-Fi solutions to choose. For now, BOARD_WLAN_DEVICE is not used for any purpose but we suggest setting this variable for your Wi-Fi solution you used.

- **WIFI_DRIVER_MODULE_NAME**
- **WIFI_DRIVER_MODULE_PATH**
- **WIFI_DRIVER_MODULE_ARG**

These three variables will be used in libhardware_legacy (wifi.c) to do insmod and remmod. The value of WIFI_DRIVER_MODULE_NAME should match the value of MODULE_NAME specified in our driver's Makefile at compile-time. Please refer to "Platform Setting Section in Detail" of:

document/Quick_Start_Guide_for_Driver_Compilation_and_Installation.doc

- **WIFI_FIRMWARE_LOADER**
- **WIFI_DRIVER_FW_PATH_STA**
- **WIFI_DRIVER_FW_PATH_AP**
- **WIFI_DRIVER_FW_PATH_P2P**
- **WIFI_DRIVER_FW_PATH_PARAM**

Because our driver has FW embedded inside, and will automatically load FW at NIC initialization process, there is no need to set these 5 variables, just keep them empty.

2.2. init.xxx.rc

For STA mode and SoftAP mode to operate, we need wpa_supplicant and hostapd daemon to be defined as service inside init.xxx.rc. Please refer to the service definitions below:

```
service wpa_supplicant /system/bin/wpa_supplicant -Dwext -iwlan0 -c/data/misc/wifi/wpa_supplicant.conf
    group wifi inet
    disabled
    oneshot

service hostapd /system/bin/hostapd_wps /data/misc/wifi/hostapd.conf
    group wifi inet
    disabled
    oneshot
```

3. libhardware_legacy

The libhardware_legacy library includes functionality for Wi-Fi to operate. We have made modifications and extensions for our Wi-Fi solutions. To apply this, please go through the following instructions:

3.1. Include wifi_realtek.c into source files

Modify hardware/libhardware_legacy/wifi/Android.mk to include wifi_realtek.c instead of wifi.c into LOCAL_SRC_FILES. For example:

```
ifeq ($(BOARD_WIFI_VENDOR), realtek)
LOCAL_SRC_FILES += ../realtek/wlan/libhardware_legacy/wifi/wifi_realtek.c
else
LOCAL_SRC_FILES += wifi/wifi.c
endif
```

4. netd

The Portable Wi-Fi hotspot functionality is controlled by the SoftapController subfunction of netd. We provide our own SoftapController implementation to run our hostapd daemon for configure driver into softap mode and handle the WPA/WPA2 handshaking. To apply this, please go through the following instructions:

4.1. Apply hostapd-0.6.9_rtl

We provide hostapd-0.6.9_rtl as the daemon instead of the hostapd from wpa_supplicant_8. To apply this, just copy the hostapd-0.6.9_rtl folder under Android SDK's external folder. After compilation, you will get the following necessary object:

```
/system/bin/hostapd
/system/bin/hostapd_wps
```

/system/etc/wifi/hostapd.conf

4.2. Include SoftapController_realtek.cpp into source files

Modify system/netd/Android.mk to include SoftapController_realtek.cpp instead of SoftapController.cpp into LOCAL_SRC_FILES. For example:

```
ifeq ($(BOARD_WIFI_VENDOR), realtek)
    LOCAL_SRC_FILES += ../../hardware/realtek/wlan/netd/SoftapController_realtek.cpp
else
    LOCAL_SRC_FILES += SoftapController.cpp
endif
```

4.3. Include libhostapd_client into static libraries

Modify system/netd/Android.mk to include libhostapd_client into LOCAL_STATIC_LIBRARIES. For example:

```
ifeq ($(BOARD_WIFI_VENDOR), realtek)
    LOCAL_STATIC_LIBRARIES := libhostapd_client
endif
```

5. Opening Portable Wi-Fi Hotspot UI

We should set the following two resource configurations of your platform to show the UI for Portable Wi-Fi Hotspot. In general you can set these two configurations in frameworks/base/core/res/res/values/config.xml

- **config_tether_wifi_regexs**

The interfaces set here are used as the interfaces for Wi-Fi LAN port. We use 'wlan0' by default when our Wi-Fi is set as softap mode. So it needs to set 'wlan0' here for system to recognized 'wlan0' as Wi-Fi LAN port. For example:

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

- **config_tether_upstream_types**

The connection types set here are used as the interfaces for WAN port to connect to internet. Please mask the item 4 (TYPE_MOBILE_DUN) or the Portable Wi-Fi Hostapd UI will not be shown. For example:

```
<integer-array translatable="false" name="config_tether_upstream_types">  
  <item>1</item>  
  <!-- <item>4</item> -->  
</integer-array>
```

To know the definition and set other upstream connection types, please refer to `frameworks/base/core/java/android/net/ConnectivityManager.java`.