

# Realtek Wi-Fi SDK for Android ICS

## ver. 2.1.0

### Contents

Release History .....	1
SDK packages .....	1
Introduction.....	2
1. Copy Necessary Files into SDK .....	3
2. Platform Related Files.....	3
2.1. BoardConfig.mk.....	3
2.2. init.xxx.rc .....	5
2.3. device/ti/panda/device.mk .....	5
3. System Resource Configurations .....	5
4. libhardware_legacy .....	7
4.1. Include wifi_realtek.c into source files .....	7
5. Apply Realtek patch for NL80211 interface and WiFi-Direct solution.....	7
5.1. Patch for wpa_supplicant_8 .....	7
5.2. Patch for WifiP2pService.java .....	8
5.3. Patch for WifiP2pDevice.java.....	9
5.4. Patch for WirelessSettings.java.....	10

### Release History

ver. 1.0.0	1. Support STA and SoftAP mode for Android 4.0 (ICS)
ver. 2.0.0	1. Support NL80211 interface 2. Support WiFi-Direct
ver. 2.1.0	1. Update patch for wpa_supplicant_8 2. Update patch for WifiP2pService.java 3. Update patch for WifiP2pDevice.java

### SDK packages

- hardware/realtek/  
Folder to store config file, private code from Realtek.
- device/ti/panda/BoardConfig.mk
- device/ti/panda/init.omap4pandaboard.rc
- device/ti/panda/device.mk
- device/ti/panda/overlay/frameworks/base/core/res/res/values/config.xml  
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  
Reference codes for applying wifi\_realtek.c
- external/wpa\_supplicant\_8/src/drivers/driver\_nl80211.c \
- external/wpa\_supplicant\_8/src/p2p/p2p.c \
- external/wpa\_supplicant\_8/src/p2p/p2p.h \
- external/wpa\_supplicant\_8/src/p2p/p2p\_group.c \
- external/wpa\_supplicant\_8/hostapd/Android.mk \
- external/wpa\_supplicant\_8/wpa\_supplicant/Android.mk \
- external/wpa\_supplicant\_8/wpa\_supplicant/ctrl\_iface.c \
- external/wpa\_supplicant\_8/wpa\_supplicant/events.c \
- external/wpa\_supplicant\_8/wpa\_supplicant/p2p\_supplicant.c \
- Reference code for apply Realtek patch for NL80211 interface and WiF-Direct
- frameworks/base/wifi/java/android/net/wifi/p2p/WifiP2pDevice.java
- frameworks/base/wifi/java/android/net/wifi/p2p/WifiP2pService.java
- Reference code to patch for Realtek WiFi-Direct solution
- packages/apps/Settings/src/com/android/settings/WirelessSettings.java
- Reference code to enable WiFi-Direct Settings UI

## 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)
- WiFi-Direct

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_20120217.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/

## 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 := NL80211
    BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_rtl
    BOARD_HOSTAPD_DRIVER      := NL80211
    BOARD_HOSTAPD_PRIVATE_LIB := lib_driver_cmd_rtl

    BOARD_WLAN_DEVICE := rtl8191su
    #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 := ""
```

- **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 := NL80211`**

- **`BOARD_WPA_SUPPLICANT_PRIVATE_LIB := lib_driver_cmd_rtl`**

- **`BOARD_HOSTAPD_DRIVER := NL80211`**

- **`BOARD_HOSTAPD_PRIVATE_LIB := lib_driver_cmd_rtl`**

We use `NL80211` as the driver interface for `wpa_supplicant` and `hostapd` to communicate with driver and provide `lib_driver_cmd_rtl` as the private processing library.

- **`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 WiFi to operate properly, we need wpa\_supplicant daemon to be defined as service inside init.xxx.rc. Please refer to the service definitions below:

```
service wpa_supplicant /system/bin/wpa_supplicant -Dnl80211 -iwlan0 -c/data/misc/wifi/wpa_supplicant.conf
    socket wpa_wlan0 dgram 660 wifi wifi
    group wifi inet
    disabled
    oneshot
```

## 2.3. device/ti/panda/device.mk

### ● Add android.hardware.wifi.direct.xml

If you want to use WiFi-Direct functionality, please add the rule in the PRODUCT\_COPY\_FILES variable in your device platform related file to copy the permission definition file, frameworks/base/data/etc/android.hardware.wifi.direct.xml to the system/etc/permissions/ folder of your system image. For example:

```
PRODUCT_COPY_FILES += \
    frameworks/base/data/etc/android.hardware.wifi.direct.xml:system/etc/permissions/android.hardware.wifi.direct.xml
```

### ● wifi.interface

To specify the wifi interface name in Android, a product property named “wifi.interface” is used. For Realtek wifi driver, wifi interface name is assigned with “wlan%d”. In general, you should set wifi.interface as “wlan0”. For example:

```
PRODUCT_PROPERTY_OVERRIDES += \
    wifi.interface=wlan0
```

## 3. System Resource Configurations

We should set the following three resource configurations of your platform to configure the network function and enable the corresponding UI interface. In general you can set the following configurations in your platform dependent config file such as:

device/ti/panda/overlay/frameworks/base/core/res/res/values/config.xml

Or the global config file:

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

- **networkAttributes**

To define the system's available network interfaces, we need to define interface items in the networkAttributes resource configuration. For example:

```
<string-array translatable="false" name="networkAttributes">
    <item>"wifi,1,1,1,-1,true"</item>
    <item>"bluetooth,7,7,0,-1,true"</item>
    <item>"ethernet,9,9,2,-1,true"</item>
    <item>"wifi_p2p,13,1,0,-1,true"</item>
</string-array>
```

- **radioAttributes**

To define the system's available network interfaces, we need to define interface items in the networkAttributes resource configuration. For example:

```
<string-array translatable="false" name="radioAttributes">
    <item>"1,1"</item>
    <item>"7,1"</item>
    <item>"9,1"</item>
</string-array>
```

- **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>"wlan\\d"</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) in global config file 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>
```

Or you could define another entry in your platform dependent config file to override the global definition. For example in our reference code:

```
<integer-array translatable="false" name="config_tether_upstream_types">
    <item>9</item>
</integer-array>
```

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

#### **4. 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:

##### **4.1. 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
```

#### **5. Apply Realtek patch for NL80211 interface and WiFi-Direct solution**

##### **5.1. Patch for wpa\_supplicant\_8**

Replace the following file with our reference code into your SDK `external/wpa_supplicant_8` folder:

```
external\wpa_supplicant_8\hostapd\Android.mk
external\wpa_supplicant_8\src\drivers\driver_nl80211.c
external\wpa_supplicant_8\src\p2p\p2p.c
external\wpa_supplicant_8\src\p2p\p2p.h
external\wpa_supplicant_8\src\p2p\p2p_group.c
external\wpa_supplicant_8\wpa_supplicant\Android.mk
external\wpa_supplicant_8\wpa_supplicant\ctrl_iface.c
external\wpa_supplicant_8\wpa_supplicant\events.c
```

external\wpa\_supplicant\_8\wpa\_supplicant\p2p\_supplicant.c

## 5.2. Patch for WifiP2pService.java

Add or modify the following code segments of WifiP2pService.java. For the specific line number and code segments, please reference our reference code:

frameworks/base/wifi/java/android/net/wifi/p2p/WifiP2pService.java

- **Line 487:**

```
class P2pDisabledState extends State {  
    @Override  
    public void enter() {  
        if (DBG) logd(getName());  
        try {  
            mNwService.wifiFirmwareReload(mInterface, "STA");  
        } catch (Exception e) {  
            loge("Failed to reload sta firmware " + e);  
        }  
    }  
    ...  
    ...  
    ...  
}
```

- **Line 973:**

```
...  
...  
    break;  
    case WifiMonitor.NETWORK_DISCONNECTION_EVENT:  
        if (DBG) logd("Network connection lost");  
        sendMessage(WifiP2pManager.REMOVE_GROUP);  
        break;  
    case DhcpStateMachine.CMD_POST_DHCP_ACTION:  
        ...  
        ...
```

- **Line 1017:**

```

...
...
try {
    mNwService.clearInterfaceAddresses(mGroup.getInterface());
    mNwService.disableIpv6(mGroup.getInterface());
} catch (Exception e) {
    loge("Failed to clear addresses or disable ipv6" + e);
}
...

```

- **Line 1060:**

```

...
...
case WifiMonitor.P2P_PROV_DISC_PBC_REQ_EVENT:
    if (mGroup.isGroupOwner())
        notifyP2pProvDiscPbcRequest((WifiP2pDevice) message.obj);
    break;
case WifiMonitor.P2P_PROV_DISC_ENTER_PIN_EVENT:
    if (mGroup.isGroupOwner())
        notifyP2pProvDiscPinRequest((WifiP2pDevice) message.obj);
    break;
...

```

### 5.3. Patch for WifiP2pDevice.java

Add or modify the following code segment in WifiP2pDevice.java to fix the string parsing bug. For the specific line number and code segments, please reference our reference code:

frameworks/base/wifi/java/android/net/wifi/p2p/WifiP2pDevice.java

- **Line 307:**

```

private String trimQuotes(String str) {
    str = str.trim();
    if (str.startsWith("\"") && str.endsWith("\"")) {
        if(str.length()==1)
            return new String();
        return str.substring(1, str.length()-1);
    }
    return str;
}

```

#### 5.4. Patch for WirelessSettings.java

Add or modify the following code segment in WifiP2pService.java to show Wifi-Direct Setting UI. For the specific line number and code segments, please reference our reference code:

packages/apps/Settings/src/com/android/settings/WirelessSettings.java

- **Line 158:**

```

if (!getPackageManager().hasSystemFeature(PackageManager.FEATURE_WIFI_DIRECT)) {
    getPreferenceScreen().removePreference(wifiP2p);
} else {
    mWifiP2pEnabler = new WifiP2pEnabler(activity, wifiP2p);
}
//getPreferenceScreen().removePreference(findPreference(KEY_WIFI_P2P_SETTINGS));

```