

密级状态：绝密( ) 秘密( ) 内部( ) 公开( ☒ )

## PX3SE\_SDK\_Wi-Fi 开发说明文档

(技术部，第二系统产品部)

文件状态：  [ ] 正在修改  [ <input checked="" type="checkbox"/> ] 正式发布	当前版本：	V1.0
	作 者：	王剑辉、陈锦森
	完成日期：	2017-12-19
	审 核：	赵子初、邓训金、陈海燕
	完成日期：	

福州瑞芯微电子股份有限公司

Fuzhou Rockchips Semiconductor Co., Ltd

(版本所有, 翻版必究)

## 版 本 历 史

版本号	作者	修改日期	修改说明	备注
V1.0	wjh、cjs	2017-12-19	发布初始版本	

## 目 录

1	概述.....	1
2	KERNEL 配置.....	1
2.1	DTS 配置.....	1
2.2	驱动配置.....	1
3	应用层 WI-FI 使用.....	2
3.1	启动 STATION 模式.....	2
3.2	启动 AP 模式.....	6

## 1 概述

PX3SE WIFI 模块目前支持的功能有 station, AP 和 WIFI DISPLAY。

该文档主要介绍 WIFI 相关的接口和相关的代码逻辑。

## 2 Kernel 配置

### 2.1 DTS 配置

文件位于（以 emmc 大容量的为例）：arch/arm/boot/dts/px3se-sdk.dts

```
wireless-wlan {
    compatible = "wlan-platdata";
    wifi_chip_type = "rtl8723ds";
    sdio_vref = <1800>; /*1800mv or 3300mv*/
    WIFI,host_wake_irq = <&gpio3 GPIO_C7 GPIO_ACTIVE_HIGH>;
    WIFI,poweren_gpio = <&gpio3 GPIO_D3 GPIO_ACTIVE_HIGH>;
    WIFI,vbat_gpio = <&gpio1 GPIO_A3 GPIO_ACTIVE_HIGH>;
    status = "okay";
};
```

wifi\_chip\_type 为使用的 wifi 型号。

WIFI,poweren\_gpio 为 wifi 的使能脚，根据实际原理图进行配置，SDK 默认为 GPIO2\_A2。

WIFI,host\_wake\_irq 为 wifi 的中断脚，根据实际原理图进行配置，SDK 默认为 GPIO0\_A3。

### 2.2 驱动配置

Make menuconfig→

Device Drivers --->

[\*] Network device support --->

[\*] Wireless LAN --->

[\*] Realtek Wireless Device Driver Support --->

<\*> Realtek 8723D SDIO or SPI WiFi

根据实际硬件选择驱动，SDK 默认配置 rtl8723ds。

Realtek wifi 驱动默认是不加载的，可以按以下操作加载和卸载驱动：

加载：echo 1 > /sys/class/rkwifi/driver

卸载：echo 0 > /sys/class/rkwifi/driver

### 3 应用层 Wi-Fi 使用

PX3SE 应用层借助 wpa\_supplicant 对 wifi 进行操作，wpa\_supplicant 是一个开源软件项目，它实现了 Station 对无线网络进行管理和控制的功能，以下凡是对 wpa\_supplicant 都简称为 WPAS。

WPAS 实际的工作通过 socket 与驱动交互并上报数据给用户，而用户可以通过 socket 发送命令给 wpa\_supplicant 调用驱动来对 wifi 芯片进行操作。WPAS 是 C/S 结构中的 Server 端，Linux/UNIX 平台中，Client 端利用 Unix 域 socket 与其通信。linux 平台目前常用的 Client 端 wpa\_cli（无界面的命令程序）。

PX3SE 应用层 WI-FI 操作对应的代码逻辑在 app/settings/wlan 中，其中 wpaserviceutil.h 对应 Wi-Fi 芯片 STATION 与 AP 模式的初始化操作以及使能状态切换，wpamanager.h 中借助 WPAS 工具封装了在 Wi-Fi Station 模式下的一些操作（扫描、连接、AP 配置保存等）。

#### 3.1 启动 station 模式

WPAS 以 C/S(客户端-服务端)方式工作，WPAS 进程运行在后台接收来自客户端的 Wi-Fi 操作请求。因此，应用层首先需要确保 WPAS 后台程序的存在。

下面是 WPAS 的启动命令：

```
/usr/sbin/wpa_supplicant -Dnl80211 -iwlan0 -c /etc/wpa_supplicant.conf
```

其中 -i 指定端口，-c 指定配置文件，-D 指定使用的 wifi 驱动；

Conf 配置文件细节因系统而异，以下是 PX3SE 使用的配置文件：

```
ctrl_interface=/var/run/wpa_supplicant
update_config=1
ap_scan=1
```

其中，ctrl\_interface 匹配 WPAS 使用的 ctrl 接口目录，一般不需要修改；

update\_config=1，表明允许 WPAS 更新覆盖配置文件；

ap\_scan 对应 WPAS 接入点的扫描和选择。

完整的应用层 Wi-Fi Station 模式初始化请参阅: wpa\_serviceutil.h — wifi\_start\_supPLICANT()函数；

wifi station 模式操作的几个函数接口：

说明：以下函数中有几个参数为 Qt 数据类型，可转换成传统 C 或 C++数据类型；

#### 1、Wi-Fi Station 模式初始化

接口名称	
int wifi_start_supPLICANT(void)	
接口作用	
Wi-Fi Station 模式初始化	
参数名称	参数说明
@void	NULL
返回值	
@ return: 设置成功返回 0，否则返回-1。	

#### 2、关闭 Wi-Fi Station 模式

接口名称	
int wifi_stop_supPLICANT(void)	
接口作用	
关闭 Wi-Fi Station 模式	
参数名称	参数说明
@void	NULL

### 返回值

@ return: 设置成功返回 0 , 否则返回-1。

#### 3、修改并保存要连接的 AP 名称和密码

### 接口名称

void connectNetwork(const QString &ssid, const QString &password);

### 接口作用

用于修改并保存 STA 的 SSID 和 PASSWORD

### 参数名称

### 参数说明

@const QString &ssid

要连接的 AP 的名称

@const QString

要连接的 AP 的密码

&password

### 返回值

@ return: void

#### 4、列出所有保存的 AP 信息

### 接口名称

QList<netWorkItem> getConfiguredNetWork(void)

### 接口作用

用于获取所有当前保存的配置 AP 列表

### 参数名称

### 参数说明

@ void

NULL

## 返回值

@ return: 获取到的 AP List

### 5、移除保存的 AP 信息

## 接口名称

void removeNetwork(int networkId)

## 接口作用

用于移除在 WPAS 配置内存中保存的 AP 信息

## 参数名称

## 参数说明

@ int networkId

AP 配置项 id

## 返回值

@ return: void

### 6、获取当前连接上的 AP 名称和密码

## 接口名称

bool getConnectedItem(netWorkItem \*item);

## 接口作用

用于获取当前连接上的 AP 名称和密码

## 参数名称

## 参数说明

@ netWorkItem \*item

networkItem 指针，用于保存获取到的信息

## 返回值

@ return: 设置成功返回 true，否则返回 false。

也可以手动进入 station 模式，假设要连接的 AP SSID 为 RK\_PX3SE，密码为 123456789



操作如下：

```
# ifconfig wlan0 up
# /usr/sbin/wpa_supplicant -Dnl80211 -iwlan0 -c /etc/wpa_supplicant.conf
# udhcpc -i wlan0 -b
```

/etc/wpa\_supplicant.config 为 STA 模式的配置文件，参考配置如下所示：

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
network={
proto=RSN
key_mgmt=WPA-PSK
pairwise=CCMP TKIP
group=CCMP TKIP
ssid=RK_PX3SE
psk=123456789
}
```

### 3.2 启动 AP 模式

PX3SE 应用层使用 hostapd 工具实现 Wi-Fi 的无线接入功能，也就是无线路由功能。hostapd 是一个带加密功能的无线接入点程序，是 Linux 操作系统上构建无线接入点的一个比较方便的工具，支持 IEEE 802.11 协议和 IEEE 802.1X/WPA/WPA2/EAP/RADIUS 加密。

值得注意的是，默认情况下，Wi-Fi 芯片只能在 Station 与 AP 模式中的其中一种，所以开启 AP 模式之前，需先关闭 Station 模式。

wifi AP 模式操作的几个函数接口：

1、根据给定的接入点名称与密码创建 hostapd 配置文件

接口名称	
int creat_hostapd_file(const char* name, const char* password)	
接口作用	
用于根据给定的接入点名称与密码创建 hostapd 配置文件	
参数名称	参数说明

@const char* name	接入点名称
@const char* password	接入点密码
返回值	
@ return: 创建成功返回 0 , 否则返回-1。	

## 2、使用 hostapd 开启 AP 模式

接口名称	
int wifi_start_hostapd(void)	
接口作用	
使用 hostapd 开启 AP 模式	
参数名称	参数说明
@void	NULL
返回值	
@ return: 启动成功返回 0 , 否则返回-1。	

## 3、关闭 Wi-Fi AP 模式

接口名称	
int wifi_stop_hostapd(void)	
接口作用	
关闭 Wi-Fi AP 模式	
参数名称	参数说明
@void	NULL

## 返回值

@ return: 启动成功返回 0 , 否则返回-1。

也可以手动进入 ap 模式，假设热点名称为：RK\_PX3SE\_AP，密码为：123456789

操作如下：

```
# ifconfig wlan0 up
# ifconfig wlan0 192.168.100.1 netmask 255.255.255.0
# echo 1 > /proc/sys/net/ipv4/ip_forward
# iptables --flush
# iptables --table nat --flush
# iptables --delete-chain
# iptables --table nat --append POSTROUTING --out-interface eth0 -j MASQUERADE
# iptables --append FORWARD --in-interface wlan0 -j ACCEPT
# /usr/sbin/hostapd /tmp/hostapd.conf -B &
```

/tmp/hostapd.conf 决定了 AP 的 SSID、PASSWORD 等信息，参考配置如下所示：

```
interface=wlan0
driver=nl80211
ssid=RK_PX3SE_AP
channel=6
hw_mode=g
ignore_broadcast_ssid=0
auth_algs=1
wpa=3
wpa_passphrase=123456789
wpa_key_mgmt=WPA-PSK
wpa_pairwise=TKIP
rsn_pairwise=CCMP
```