a8800 sdk WiFi API

本文档为aic8800M/A/F/MC/FC/M40B/M80系列芯片WiFi模块API手册

配置接口

API文件

- tgt_cfg_wifi.h
- rwnx_msg_tx.h
- fhost_config.h
- fhost_config.c

API参数

1) TX power档位设置

- AIC8800M/A/F 使用pwridx索引值设置功率，取值范围0~11，CONFIG_WIFI_TXPWR_IDX_ENABLE使能功率设置
- AIC8800MC/FC/M40B/M80 使用pwrlvl设置功率，单位是dBm，建议取值<=30，CONFIG_WIFI_TXPWR_LVL_ENABLE使能功率设置

2) TX power偏移设置

- pwrofst用于不同信道功率调平，2.4G 频段可设置3组值，5G 频段可设置4组值
- pwrofst默认使用存于flash中的校准值，如果CONFIG_WIFI_TXPWR_OFST_ENABLE使能，偏移值会被强制修改

3) TX 自适应测试

- CONFIG_WIFI_TX_ADAPTIVITY_ENABLE默认关闭，进行测试时开启

4) 协议栈参数

- Listen Interval & Don't Wait BCMC

```
void set_deepsleep_param(unsigned int listen_interval, unsigned int
dont_wait_bcmc);
```

- EDCA parameters

```
int rwnx_send_set_edca(int fhost_vif_idx, rwnx_edca_param_t *edca_param);
```

5) 国家码

- CONFIG_WIFI_COUNTRY_CODE 默认设置"00"，其他选择支持："CA", "CN", "US", ...

6) hwconfig: TXOP, channel_access, mac_timescale, cca_threshold, bwmode等

```
int rwnx_send_vendor_hwconfig_req(void *config_param);
```

常用操作接口

API文件

- wlan_if.h

API函数

1 关联Station

```
 /**
 ********************************************************************************
*********
 * @brief * wlan_start_sta : Start STA to connect to router, until obtain IP
address
 *                           or timeout
 *
 * @param[]  ssid: router name
 *           pw  : router's password(if open, set NULL)
 *           timeout_ms: 0 -> 10000, -1 -> no timeout, (timeout_ms > 0 && bit0
is 1) -> wep
 * @return value: -1:password < 8, -6: dhcp fail
 ********************************************************************************
*********
 */
int wlan_start_sta(uint8_t *ssid, uint8_t *pw, int timeout_ms);
```

2 断开Station

```
/**
********************************************************************************
********
* @brief * wlan_disconnect_sta : STA disconnect
*
* @param[]  idx: 0
* @return value: 1: fail, 0: success
********************************************************************************
********
*/
int wlan_disconnect_sta(uint8_t idx);
```

3 启动 AP

```
/**
********************************************************************************
********
* @brief * wlan_start_ap : Create softAP
*
* @param[]  band: 0 -> 2.4G, 1 -> 5G
*           ssid: name
*           pw  : password(if open, set NULL)
* @return value: -1: fail, 0: success
********************************************************************************
********
*/
int wlan_start_ap(uint8_t band, uint8_t *ssid, uint8_t *pw);
```

4 关闭 AP

```
/**
*****************************************************************************
********
* @brief * wlan_stop_ap : Stop softAP
*
* @param[]  none
* @return value: -1: fail, 0: success
*****************************************************************************
********
*/
int wlan_stop_ap(void);
```

5 获取 RSSI

两种方式获取RSSI数值：

1) 从扫描结果中获取（Scan Result）

```
fhost_get_scan_results
```

2) 获取数据包RSSI

```
data_pkt_rssi_get
```

6 AP参数设置

这些参数需要在启动ap之前设置

```
/**
*****************************************************************************
********
* @brief * set_ap_ip_addr : Set IP start address for softAP
*
* @param[]  new_ip_addr: IP address
*                        Default: (192 | (168 << 8) | (88 << 16) | (1 << 24))
*****************************************************************************
********
*/
void set_ap_ip_addr(uint32_t new_ip_addr);
/**
*****************************************************************************
********
* @brief * set_ap_subnet_mask : Set subnet mask for softAP
*
* @param[]  new_mask: subnet mask
*                     Default: 255.255.255.0 -> 0x00FFFFFF
*****************************************************************************
********
*/
void set_ap_subnet_mask(uint32_t new_mask);
/**
*****************************************************************************
********
* @brief * set_ap_bcn_interval : Set Beacon interval for softAP
*
```

```
* @param[]  bcn_interval_ms: beacon interval(unit:ms)
*                            Default: 100
*****************************************************************************
********
*/
void set_ap_bcn_interval(uint32_t bcn_interval_ms);

/**
*****************************************************************************
********
* @brief * set_ap_channel_num : Set channel number for softAP, if not set, auto
select
*
* @param[]  num: channel nuber, 2.4G(1~14), 5G(depend on 'fhost_chan')
*****************************************************************************
********
*/
void set_ap_channel_num(uint8_t num);
/**
*****************************************************************************
********
* @brief * set_ap_hidden_ssid : Set hidden ssid for softAP
*
* @param[]  val: 0 -> unhidden, 1 -> hidden
*****************************************************************************
********
*/
void set_ap_hidden_ssid(uint8_t val);
/**
*****************************************************************************
********
* @brief * set_ap_enable_he_rate : Enable HE rate for softAP
*
* @param[]  en: 1 -> enable, 0(Default) -> diable
*****************************************************************************
********
*/
void set_ap_enable_he_rate(uint8_t en);
/**
*****************************************************************************
********
* @brief * set_ap_allow_sta_inactivity_s : Set max inactivity time for connected
STA (if
*                                          long time not receive/transmit, send
a NULL
*                                          frame to detect it. If detected fail,
*                                          disassociate it).
*
* @param[]  s: default -> 60s
*****************************************************************************
********
*/
void set_ap_allow_sta_inactivity_s(uint8_t s);
```

7 AP切换信道

```
/**
*****************************************************************************
********
* @brief * wlan_ap_switch_channel : softAP switch to specified channel
*
* @param[]  chan_num: channel number
* @return value: -1: fail, 0: success
*****************************************************************************
********
*/
int wlan_ap_switch_channel(uint8_t chan_num);
```

8 AP断开STA

```
/**
*****************************************************************************
********
* @brief * wlan_ap_disassociate_sta : softAP disassociate specified STA
*
* @param[]  macaddr: mac address of specified STA
* @return value: -1: fail, 0: success
*****************************************************************************
********
*/
int wlan_ap_disassociate_sta(struct mac_addr *macaddr);
```

9 获取STA当前连接的热点信息

```
/**
*****************************************************************************
********
* @brief * int wlan_get_sta_connected_ap_info(struct connected_ap_info_t **info)
*
* @param[out] info: info of ap associated
* @return value: -1: fail, 0: success
*****************************************************************************
********
*/
int wlan_get_sta_connected_ap_info(struct connected_ap_info_t **info);
```

10 获取STA当前连接的IP信息

```
/**
 *****************************************************************************
*********
 * @brief Get IPv4 address of an interface
 *
 * Set to NULL parameter you're not interested in.
 *
 * @param[in]  net_if Pointer to the net_if structure
 * @param[out] ip     IPv4 address
 * @param[out] mask   IPv4 network mask
 * @param[out] gw     IPv4 gateway address
 * @return 0 if requested parameters have been updated successfully and !=0
otherwise.
```

```
 ********************************************************************************
*********
 */
int net_if_get_ip(net_if_t *net_if, uint32_t *ip, uint32_t *mask, uint32_t *gw);
```

注：第一个参数net_if 可以通过net_if_t *net_if_find_from_wifi_idx(unsigned int idx);获得

11 设置 scan vendor ie

```
int rwnx_send_vendor_ie_req(uint8_t *ies, uint16_t ie_len);
```

Raw Data接口

API文件

- rawdata_if.h

API函数

1 设置MAC

```
/**
 ********************************************************************************
*********
 * @brief * rawdata_if_set_mac_addr : Set own mac address.
 *
 * @param[]  Notice: MUST called before 'rawdata_if_init'
 ********************************************************************************
*********
 */
void rawdata_if_set_mac_addr(uint8_t *macaddr);
```

2 设置信道

```
/**
 ********************************************************************************
*********
 * @brief * rawdata_if_set_channel : Select channel to tx/rx raw data.
 *
 * @param[]  ch_num: channel num (2.4G : 1 ~ 14, 5G : 36 ~ 165)
 ********************************************************************************
*********
 */
void rawdata_if_set_channel(uint8_t ch_num);
```

3 接收回调

```
/**
 ****************************************************************************************
*********
 * @brief * rawdata_if_rx_data_cb_register : Register API to get rx data.
 *
 * @param[]  eg.
 *               static void raw_data_handler(uint8_t *data, uint32_t size)
 *               {
 *                   dump_buf(data, size);
 *               }
 *               rawdata_if_rx_data_cb_register(raw_data_handler);
 ****************************************************************************************
*********
 */
void rawdata_if_rx_data_cb_register(wifi_raw_data_handler_t handler);
```

4 发送数据包

```
/**
 ****************************************************************************************
*********
 * @brief * rawdata_if_tx_data : TX raw data(802.11).
 *
 * @param[]  frame: data to tx
             frame_len : data length
 ****************************************************************************************
*********
 */
int rawdata_if_tx_data(uint8_t *frame, uint16_t frame_len);
```

Monitor混杂模式

API文件

- fhost_cntrl.h

API函数

1 设置monitor参数

```
/**
 ****************************************************************************************
*********
 * @brief Configure a monitor interface
 *
 * Send message to control task to configure a monitor interface. The interface
type
 * must have been changed to VIF_MONITOR (using @ref fhost_set_vif_type) before
calling
 * this function.
 *
 * @param[in] link           Link with CNTRL task to use
 * @param[in] fhost_vif_idx  Index of the FHOST vif
 * @param[in] cfg            Configuration to apply
 *
```

```
 * @return 0 on success and != 0 if error occurred.
 ******************************************************************************
*********
 */
int fhost_cntrl_monitor_cfg(struct fhost_cntrl_link *link, int fhost_vif_idx,
                            struct fhost_vif_monitor_cfg *cfg);
```

2 设置filter过滤

```
int fhost_cntrl_mm_set_filter(uint32_t value);
```

3 callback函数处理

```
/**
 ******************************************************************************
*********
 * @brief callback function
 *
 * Extract received packet infomations.
 *
 * @param[in] info           RX frame information
 * @param[in] arg            Not used
 ******************************************************************************
*********
 */
void user_monitor_callback(struct fhost_frame_info *info, void *arg);
```