XRADIO WLAN Config Developer Guide

Revision 1.0

Nov 23, 2019

Declaration

THIS DOCUMENTATION IS THE ORIGINAL WORK AND COPYRIGHTED PROPERTY OF XRADIO TECHNOLOGY ("XRADIO"). REPRODUCTION IN WHOLE OR IN PART MUST OBTAIN THE WRITTEN APPROVAL OF XRADIO AND GIVE CLEAR ACKNOWLEDGEMENT TO THE COPYRIGHT OWNER.

THE PURCHASED PRODUCTS, SERVICES AND FEATURES ARE STIPULATED BY THE CONTRACT MADE BETWEEN XRADIO AND THE CUSTOMER. PLEASE READ THE TERMS AND CONDITIONS OF THE CONTRACT AND RELEVANT INSTRUCTIONS CAREFULLY BEFORE USING, AND FOLLOW THE INSTRUCTIONS IN THIS DOCUMENTATION STRICTLY. XRADIO ASSUMES NO RESPONSIBILITY FOR THE CONSEQUENCES OF IMPROPER USE (INCLUDING BUT NOT LIMITED TO OVERVOLTAGE, OVERCLOCK, OR EXCESSIVE TEMPERATURE).

THE INFORMATION FURNISHED BY XRADIO IS PROVIDED JUST AS A REFERENCE OR TYPICAL APPLICATIONS, ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS DOCUMENT DO NOT CONSTITUTE A WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. XRADIO RESERVES THE RIGHT TO MAKE CHANGES IN CIRCUIT DESIGN AND/OR SPECIFICATIONS AT ANY TIME WITHOUT NOTICE.

NOR FOR ANY INFRINGEMENTS OF PATENTS OR OTHER RIGHTS OF THE THIRD PARTIES WHICH MAY RESULT FROM ITS USE. NO LICENSE IS GRANTED BY IMPLICATION OR OTHERWISE UNDER ANY PATENT OR PATENT RIGHTS OF XRADIO.THIRD PARTY LICENCES MAY BE REQUIRED TO IMPLEMENT THE SOLUTION/PRODUCT. CUSTOMERS SHALL BE SOLELY RESPONSIBLE TO OBTAIN ALL APPROPRIATELY REQUIRED THIRD PARTY LICENCES. XRADIO SHALL NOT BE LIABLE FOR ANY LICENCE FEE OR ROYALTY DUE IN RESPECT OF ANY REQUIRED THIRD PARTY LICENCE. XRADIO SHALL HAVE NO WARRANTY, INDEMNITY OR OTHER OBLIGATIONS WITH RESPECT TO MATTERS COVERED UNDER ANY REQUIRED THIRD PARTY LICENCE.

Revision History

Version	Date	Summary of Changes
1.0	2019-11-23	Initial Version

Table 1- 1 Revision History

Contents

Tables

Figures

1 概述

配网，即 WLAN 设备快速入网配置技术，主要在如下场景中使用：

1.待接入互联网的设备不具备输入输出能力，如空调、空气净化器、烟雾报警器等。

2.用户不具备通过设备热点的方式进行配置的能力，如老人、家庭主妇等缺乏相关 IT 知识的用户人群。

目前 Xradio SDK 支持 4 种配网方式，分别为 AirKiss 配网、SmartConfig 配网、VoicePrint 配网和 SoftAp 配网。并且为了兼容多种配网方式同时启动，sdk 提供了配网助手 sc_assistant，来协调各个配网模块的运行。本文主要介绍这 4 种配网方式和配网助手 sc_assistant。

2 AirKiss 配网

2.1 配网原理

AirKiss 是微信硬件平台为 WLAN 设备提供的微信配网、局域网发现和局域网通讯的技术。XRADIO 为开发者提供了通过微信客户端对 WLAN 设备配网，以及通过微信客户端局域网发现 WLAN 设备的功能。其本质是通过 WLAN 收发包来达到传递信息的目的。

WLAN 设备有一种 monitor 模式，可以监听空口中的所有 WLAN 数据包。WLAN 设备处于正常 station 模式下，设备是会过滤掉目标 mac 地址与自身 mac 地址不匹配的数据包，而在 monitor 模式下，WLAN 设备可以接收所有符合 802.11 协议的数据包。Airkiss 配网就是借用了 WLAN 设备的这种模式，来实现配网的目的。

当启动 Airkiss 配网后，手机 app 端会在固定的信道发送特定的数据包。而 WLAN 设备会启动 monitor 模式，监听接收空口中的所有 WLAN 数据包，且设备会在接收识别到 app 端发送的特定数据包前，不断切换 WLAN 信道，从而锁定 app 端信道，并在该信道进行后续的通信。

2.1.1 通信协议

2.1.1.1物理层协议

Airkiss 利用 802.2 SNAP（802.11 的物理层协议）的数据帧进行信息传递。其中该数据帧的格式如下：

DA	SA	Length	LLC	SNAP	DATA	FCS
6bytes	6bytes	2bytes	3bytes	5bytes	38-1492bytes	4bytes

图 2-1 802.2 SNAP 帧格式

其中 DA 表示目的 mac 地址，SA 表示源 mac 地址，Length 表示数据的长度，LLC 表示 LLC 头部，SNAP 包括了厂商代码和协议类型标识，DATA 为有效数据区，对于加密信道来说是密文，FCS 表示帧校验区域。

从无线信号监听方的角度来说，不管无线信道有没有加密，DA、SA、Length、LLC、SNAP、FCS 字段总是明文状态，因此监听方便可以从这 6 个字段获取到一定的信息。但从发送方的角度来说，由于操作系统的限制，DA、SA、LLC、SNAP、FCS 五个字段的控制需要很高的控制权限，发送方很难取得控制权进行修改。因此，只能利用 Length 字段进行信息传递。所以，只要制定出一套利用长度字段编码的通信协议，就可利用 802.2 SNAP 数据包中的 Length 字段进行信息传递。

在实际应用中，airkiss 采用 UDP 广播包作为信息的载体。信息发送方向空间中发送一系列的 UDP 广播包，其中每一包的 Length 字段都按照 AirKiss 通信协议进行编码，信息接收方利用 monitor 模式监听接收空间中的无线信号，并从数据链路层获取 802.2 SNAP 格式数据包，便可得到已编码的 Length 字段，随后接收方便可根据 Air Kiss 通信协议解析出需要的信息。

2.1.1.2链路层协议

Airkiss 通信协议利用 UDP 数据包的长度进行编码，即有效数据都存储在 UDP 包的 length 字段，每次最多可使用 9bit 的数据，因为每个数据包只能传输 1byte 数据。发送的数据可以分成 4 类：Magic code、Prefix SSID code、Prefix PSK code、Sequence * N，其组织结构如下图所示：

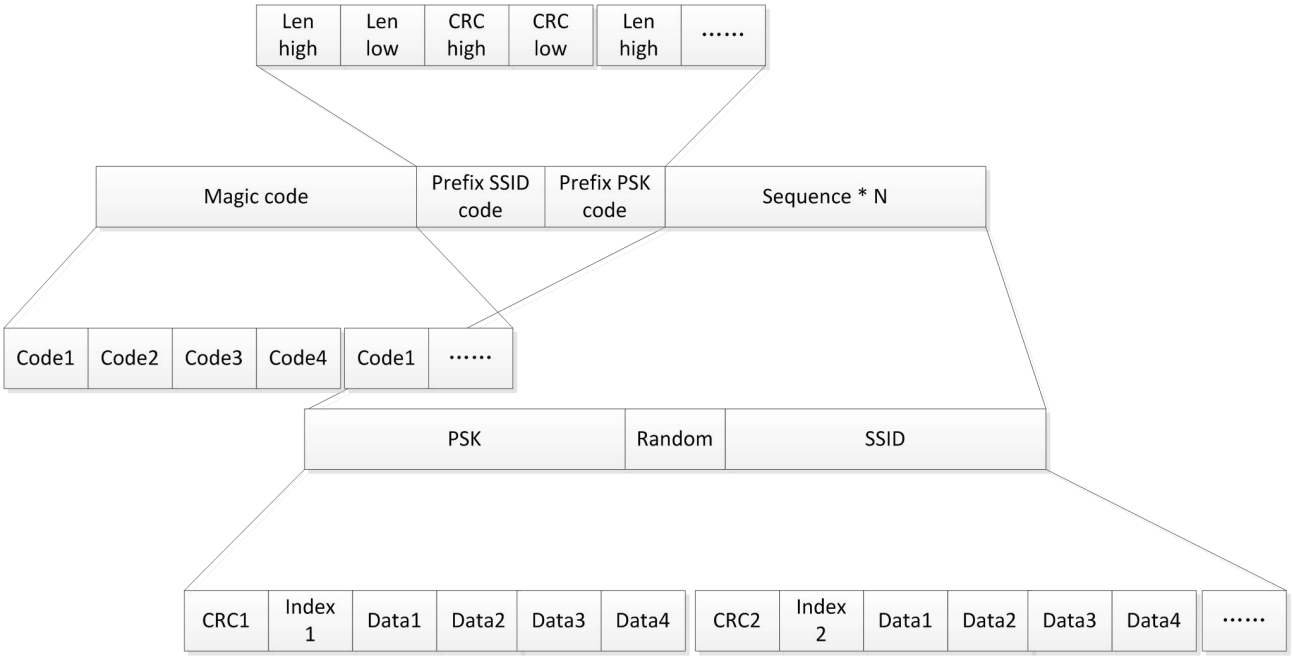

图 2-2 Airkiss 通信协议格式

1、Magic code

Magic code 为 4 个连续的 9bit 数据，用于计算后续数据的基准偏移，例如 4 个值为 61,62,63,64，则偏移为第一个值减 1，即 60.

2、Prefix SSID/PSK code

Prefix SSID code 和 prefix PSK code 格式相同，都是以 4 个 9bit 数据为一组，每个数据以高 5bit 存储 index，前两个数据的低 4bit 为 length，后两个的数据的低 4bit 为 CRC。

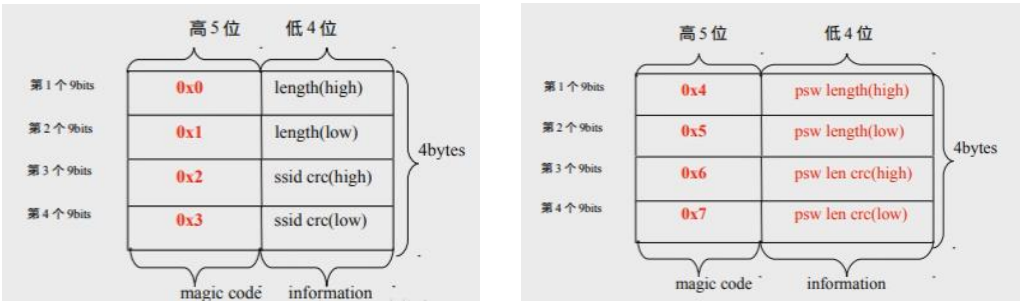

图 2-3 Prefix SSID/PSK code

3、Sequence

Sequence 包含了 PSK、random 值以及 SSID 的具体内容，airkiss 将其分成 4byte 一组，每一组为其加上 2byte 的 CRC 和 index，所以最后发出来的数据为 6byte 一组。需要注意的是，最后的数据如果没有 4byte 对齐，airkiss 不会自动补上数据，而是保持不对齐的格式进行解析。

2.2 配网流程

为了结合配网助手，xradio sdk 提供的配网流程和配网接口都使用配网助手进行了一定的封装。其中 Airkiss 配网流程如下图：

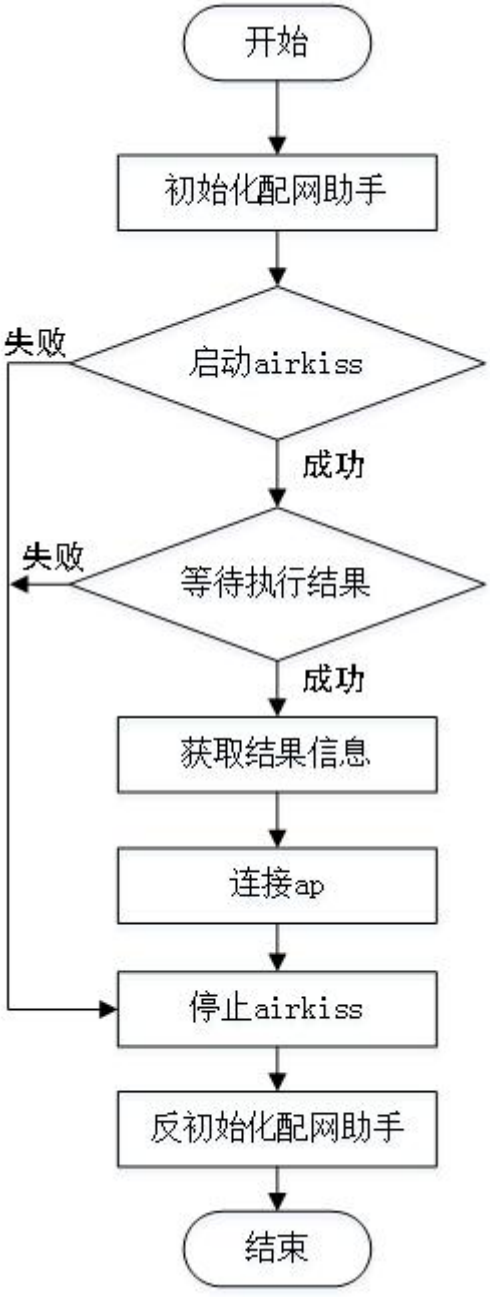

图 2-4 Airkiss 配网流程

2.3 接口说明

相关 api 描述如下：

表 2-1 Airkiss 函数接口描述

function	detail
sc_assistant_get_fun;	声明：void sc_assistant_get_fun(sc_assistant_fun_t *fun); 目的：获取配网助手操作函数 参数：fun：获取到的操作接口 返回值：无
sc_assistant_init;	声明：int sc_assistant_init(struct netif *nif, sc_assistant_fun_t *cb, sc_assistant_time_config_t *config); 目的：初始化配网助手 参数：nif：网络接口；cb：配网助手的操作函数；config：配置参数 返回值：返回状态
wlan_airkiss_start;	声明：wlan_airkiss_status_t wlan_airkiss_start(struct netif *nif, char *key); 目的：启动 airkiss 参数：nif：网络接口；key：airkiss 传输使用的密码 返回值：返回状态
wlan_airkiss_wait;	声明：wlan_airkiss_status_t wlan_airkiss_wait(uint32_t timeout_ms); 目的：等待 airkiss 执行结束 参数：ms：等待超时时间 返回值：返回状态
wlan_airkiss_get_status;	声明：airkiss_status_t wlan_airkiss_get_status(void); 目的：获取配网的状态 参数：无 返回值：返回配网状态
wlan_airkiss_get_result;	声明：wlan_airkiss_status_t wlan_airkiss_get_result(wlan_airkiss_result_t *result); 目的：获取配网的结果

	参数：配网结果结构体的指针 返回值：返回状态
wlan_airkiss_connect_ack;	声 明 ： wlan_airkiss_status_t wlan_airkiss_connect_ack(struct netif *nif, uint32_t timeout_ms, wlan_airkiss_result_t *result); 目的：连接 ap，并返回 ack 给手机端 app 参数：nif：网络接口；timeout_ms：超时时间；result：获取到的配网结果 返回值：返回状态
wlan_airkiss_stop;	声明：int wlan_airkiss_stop(void); 目的：停止 airkiss 模块 参数：无 返回值：返回状态
sc_assistant_deinit;	声明：int sc_assistant_deinit(struct netif *nif); 目的：反初始化配网助手 参数：nif：网络接口； 返回值：返回状态

2.4 使用示例

请参考示例工程 example/airkiss

3 SmartConfig 配网

3.1 配网原理

SmartConfig 配网是 xradio 基于自己的通信协议开发出来的配网技术，其本质是通过 WLAN 收发包来达到传递信息的目的。

3.2 配网流程

为了结合配网助手，xradio sdk 提供的配网流程和配网接口都使用配网助手进行了一定的封装。其中 SmartConfig 配网流程如下图：

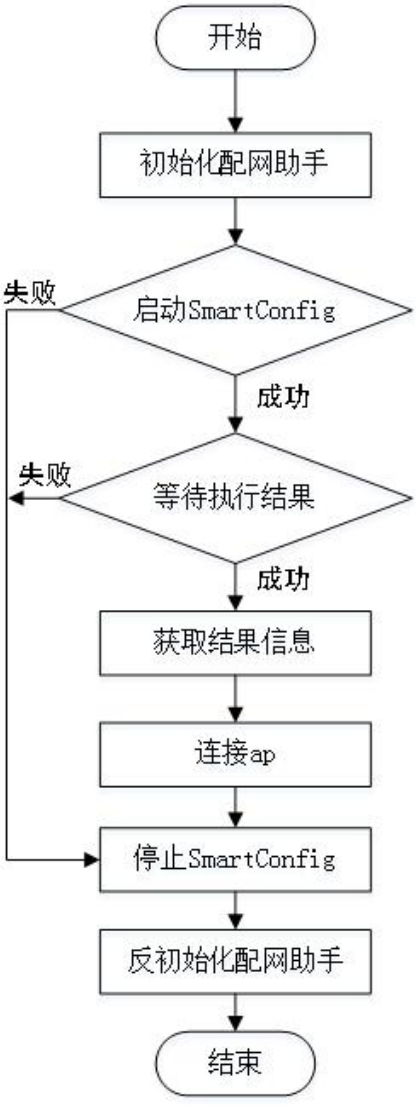

图 3-1 SmartConfig 配网流程

3.3 接口说明

相关 api 描述如下：

表 3-1 SmartConfig 函数接口描述

function	detail
sc_assistant_get_fun;	声明：void sc_assistant_get_fun(sc_assistant_fun_t *fun); 目的：获取配网助手操作函数 参数：fun：获取到的操作接口 返回值：无
sc_assistant_init;	声明：int sc_assistant_init(struct netif *nif, sc_assistant_fun_t *cb, sc_assistant_time_config_t *config); 目的：初始化配网助手 参数：nif：网络接口；cb：配网助手的操作函数；config：配置参数 返回值：返回状态
wlan_smart_config_start;	声明：wlan_smart_config_status_t wlan_smart_config_start(struct netif *nif, char *key); 目的：启动 airkiss 参数：nif：网络接口；key：smartConfig 传输使用的密码 返回值：返回状态
wlan_smart_config_wait;	声明：wlan_smart_config_status_t wlan_smart_config_wait(uint32_t timeout_ms); 目的：等待 SmartConfig 执行结束 参数：ms：等待超时时间 返回值：返回状态
wlan_smart_config_get_status;	声明：SMART_CONFIG_STATUS_T wlan_smart_config_get_status(void); 目的：获取配网的状态 参数：无 返回值：返回配网状态
smartconfig_get_result;	声明：wlan_smart_config_status_t smartconfig_get_result(wlan_smart_config_result_t *result);

	目的：获取配网的结果 参数：配网结果结构体的指针 返回值：返回状态
wlan_smart_config_connect_ack;	声 明 ： wlan_smart_config_status_t wlan_smart_config_connect_ack(struct netif *nif, uint32_t timeout_ms, wlan_smart_config_result_t *result); 目的：连接 ap，并返回 ack 给手机端 app 参数：nif：网络接口；timeout_ms：超时时间；result：获取到的配网结果 返回值：返回状态
wlan_smart_config_stop;	声明：int wlan_smart_config_stop(void); 目的：停止 SmartConfig 模块 参数：无 返回值：返回状态
sc_assistant_deinit;	声明：int sc_assistant_deinit(struct netif *nif); 目的：反初始化配网助手 参数：nif：网络接口； 返回值：返回状态

3.4 使用示例

请参考示例工程 example/smartconfig

4 VoicePrint 配网

4.1 配网原理

Voice print 是使用声波对 SSID 和 PSK 数据进行传输，WLAN 设备对收到的声波数据进行解码，解析出 SSID 和 PSK 数据，然后再去连接该 AP 即可。

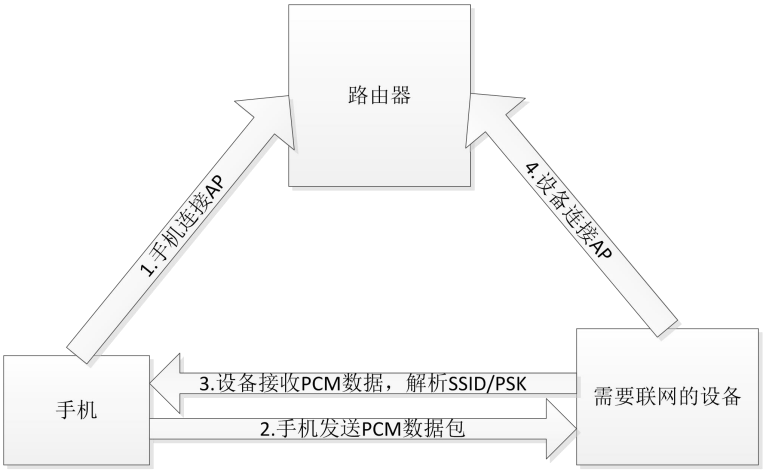

图 4-1 VoicePrint 配网架构图

4.2 配网流程

为了结合配网助手，xradio sdk 提供的配网流程和配网接口都使用配网助手进行了一定的封装。其中 VoicePrint 配网流程如下图：

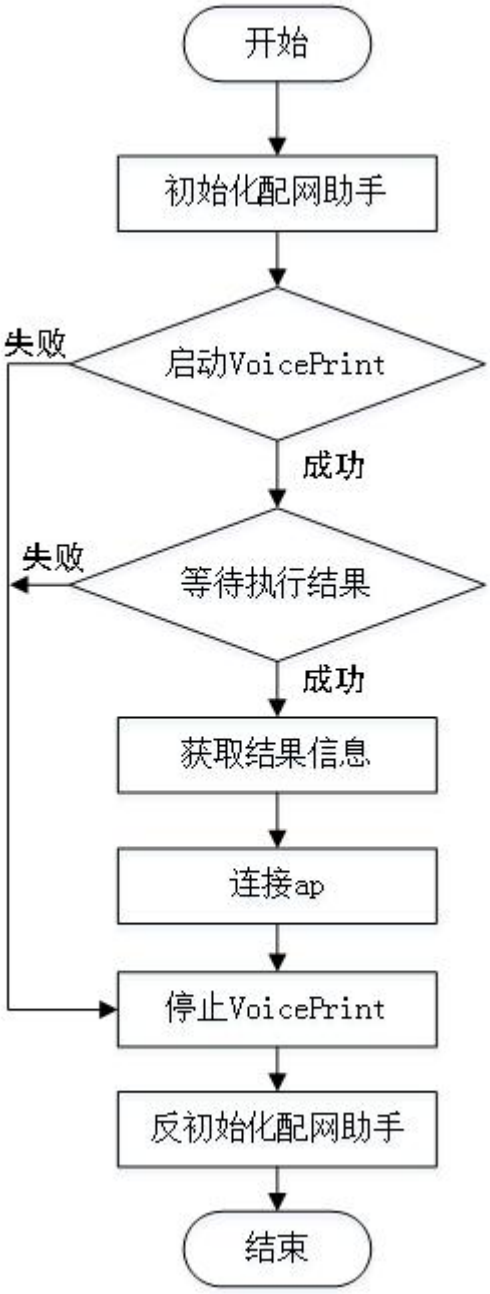

图 4-2 VoiceP rint 配网流程

4.3 接口说明

相关 api 描述如下：

表 4-1 VoicePrint 函数接口描述

function	detail
sc_assistant_get_fun;	声明： void sc_assistant_get_fun(sc_assistant_fun_t *fun);

	目的：获取配网助手操作函数 参数：fun：获取到的操作接口 返回值：无
sc_assistant_init;	声明：int sc_assistant_init(struct netif *nif, sc_assistant_fun_t *cb, sc_assistant_time_config_t *config); 目的：初始化配网助手 参数：nif：网络接口；cb：配网助手的操作函数；config：配置参数 返回值：返回状态
voiceprint_start;	声明：int voiceprint_start(voiceprint_param_t *param); 目的：启动声波配网 参数：param：配网参数 返回值：返回状态
voiceprint_wait;	声明：voiceprint_ret_t voiceprint_wait(uint32_t timeout_ms); 目的：等待 VoicePrint 执行结束 参数：ms：等待超时时间 返回值：返回状态
voiceprint_get_status;	声明：voiceprint_status_t voiceprint_get_status(void); 目的：获取配网的状态 参数：无 返回值：返回配网状态
wlan_voiceprint_get_raw_result;	声明：voiceprint_ret_t wlan_voiceprint_get_raw_result(char *result, int *len); 目的：获取配网结果的裸数据。该数据需要经过解析才能得到 ssid 和 password 参数：result：裸数据的存放地址；len：裸数据的长度 返回值：返回状态
voiceprint_stop;	声明：int voiceprint_stop(uint32_t wait); 目的：停止 VoicePrint 模块 参数：wait：是否等待 VoicePrint 线程退出 返回值：返回状态

sc_assistant_deinit;	声明：int sc_assistant_deinit(struct netif *nif); 目的：反初始化配网助手 参数：nif：网络接口； 返回值：返回状态

4.4 使用示例

请参考示例工程 example/voiceprint

5 SoftAp 配网

5.1 配网原理

softap 配网是将设备切换为 AP 模式，然后创建一个 webserver，手机通过浏览器输入 AP 的 ip 地址来访问 webserver，手机浏览器通过 post 的方式将填写好的 ssid 和 psk 发送到 webserver，设备接收到 ssid 和 psk 后，切换为 STA 模式去连接网络。softap 配网实现 webserver 的方式有两种：

1. 通过 socket 方式创建 webserver，socket 接收到浏览器发送的 request 后，解析 request，然后构建 response（response 中包含需要发送的 html 网页内容），发送 response 到浏览器中。该方式实现起来比较繁琐、复杂。

2. 通过 shttpd 模块来创建 webserver，所有的解析和构建操作都有 shttpd 来完成。该方式实现起来比较简单。

本文采用的方式 1 来构建 webserver。

5.2 配网流程

softap 基本流程就是：创建 socket->监听端口->等待连接->获取客户端请求->解析请求->根据请求执行不同的操作并返回不同的页面。

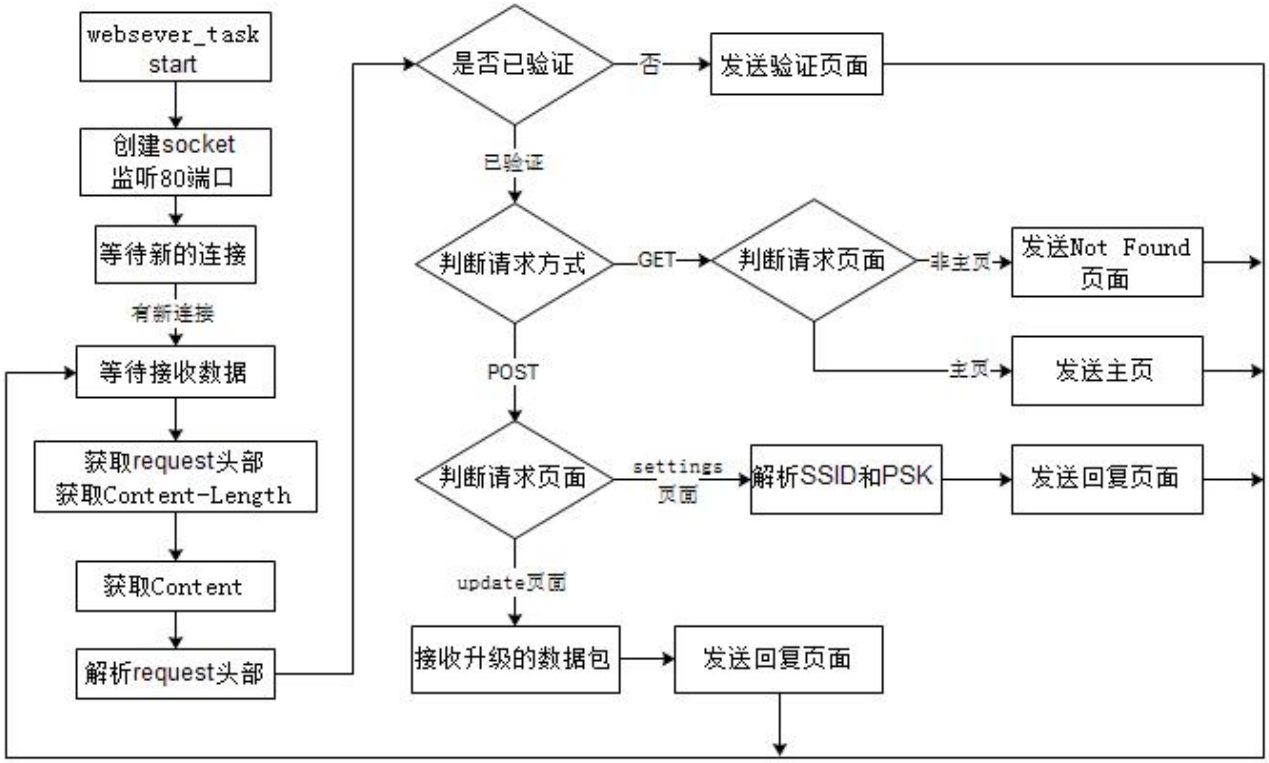

图 5-1 Softap 配网流程

5.3 接口说明

SoftAp 配网模块使用到的结构体主要有以下两个，分别用来记录配网的结果信息和配网状态，描述如下：

```
typedef struct {
    char ssid[32]; //获取到的 SSID
    char psk[32]; //获取到的 PSK
} soft_ap_config_result; //配网结果结构体
```

```
typedef enum {
    SOFT_AP_CONFIG_STOP, //配网停止
    SOFT_AP_CONFIG_START, //配网开始
    SOFT_AP_CONFIG_COMPLETE, //配网完成
} SOFT_AP_CONFIG_STA; //配网的状态
```

相关的函数接口如下

表 5-1 Softap 函数接口说明

function	detail
soft_ap_config_start;	声明：int soft_ap_config_start(void); 目的：启动 softap 配网 参数：无 返回值：返回状态 (0: 成功；-1：失败)
soft_ap_config_stop;	声明：int soft_ap_config_stop(void); 目的：停止 softap 配网 参数：无 返回值：返回状态 (0: 成功；-1：失败)
soft_ap_config_set_cb;	声明：int soft_ap_config_set_cb(soft_ap_config_cb cb); 目的：设置配网的回调函数，配网完成后，会调用该回调函数 参数：cb：回调函数指针 返回值：返回状态 (0: 成功；-1：失败)
soft_ap_config_get_result;	声明：int soft_ap_config_get_result(soft_ap_config_result *result); 目的：获取配网的结果，只有在配网完成后，才能获取，否则返回-1 参数：result：配网结果结构体指针 返回值：返回状态 (0: 成功；-1：失败)

soft_ap_config_get_state;	声明：int soft_ap_config_get_state(void); 目的：获取配网的状态 参数：无 返回值：返回配网状态

5.4 使用示例

SoftAp 配网相关代码位于以下目录：

文件类型	代码位置
source	sdk/project/example/soft_ap_config/soft_ap_config.c
header	sdk/project/example/soft_ap_config/soft_ap_config.h

SoftAp 配网使用示例可参考工程 example/soft_ap_config

6 配网助手 sc_assistant

由于多种配网方式有可能会被同时启动，为了兼容这种情况，需要一个额外的模块对各种配网方式进行管理，于是便引入了配网助手模块 sc_assistant.

配网助手 sc_assistant 并不是一种配网技术，而是用于协调各个配网方式的运行，对整个配网功能起到一种优化的作用。

使用 sc_assistant 模块，对比普通的配网流程有较大优势，如下列出：

1.可同时使用多种配网方式，兼容各个配网流程；

2.优化了切换信道的时机，增大配网成功的概率；