



XR806 Blink 配网应用 开发指南

版本号：1.0

发布时间：2020-08-12

版本历史

版本	日期	责任人	版本描述
1.0	2020-08-12	AWA 1451	创建文档。

目录

版本历史.....	i
目录.....	ii
表格目录.....	iii
1 前言.....	1
1.1 文档简介.....	1
1.2 目标读者.....	1
1.3 适用范围.....	1
1.4 文档约定.....	1
1.4.1 标志说明.....	1
2 概述.....	2
2.1 背景说明.....	2
2.2 规格特性.....	2
2.3 文件位置.....	2
3 技术说明.....	3
3.1 模块框架.....	3
3.2 使用流程.....	3
4 应用说明.....	5
4.1 应用简述.....	5
4.2 配置说明.....	5
4.3 接口说明.....	5
4.3.1 wlan_blink_start.....	6
4.3.2 wlan_blink_wait_once.....	6
4.3.3 wlan_blink_get_status.....	7
4.3.4 wlan_blink_connect.....	7
4.3.5 wlan_blink_stop.....	7
5 示例说明.....	8
5.1 示例简介.....	8
5.2 关键实现.....	8
5.3 效果展示.....	11

表格目录

表 2-1	XR806 wlanblink 模块代码位置.....	2
表 4-1	wlanblink 模块配置列表.....	5
表 4-2	wlan_blink_start 接口函数说明.....	6
表 4-3	wlan_blink_param_t 结构体的成员说明.....	6
表 4-4	wlan_blink_ret_t 枚举类型说明.....	6
表 4-5	wlan_blink_wait_once 接口函数说明.....	6
表 4-6	wlan_blink_get_status 接口函数说明.....	7
表 4-7	wlan_blink_status_t 枚举类型说明.....	7
表 4-8	wlan_blink_connect 接口函数说明.....	7
表 4-9	wlan_blink_stop 接口函数说明.....	7
表 5-1	ble_demo 示例工程的命令列表.....	8

1 前言

1.1 文档简介

本文档介绍了 wlanblink 配网模块的使用方法,你可以通过该文档了解如何基于 Blink 模块开发配网功能。

1.2 目标读者

想了解 Blink 配网在多种配网方式共存场景下开发的用户。

1.3 适用范围

此文档适用于 XR806 SDK, 支持 XR806 系列芯片产品。

1.4 文档约定

1.4.1 标志说明

本文档采用各种醒目的标志来表示在操作过程中应该特别注意的地方, 这些标志的含义如下:

标识	说明
 说明	为准确理解文中指令、正确实施操作而提供的补充或强调信息。

2 概述

2.1 背景说明

在实际 IoT 配网场景中，多种配网方式可能同时被启动，如 XR806 SDK 支持 airkiss、smartconfig、声波配网等多种方式同时配网。Blink 配网为了适用这种配网场景，基于配网助手，在 Blink 模块基础上封装一层应用层，即 wlanblink。

2.2 规格特性

支持基于 XR806 配网助手进行 BLE 配网。

2.3 文件位置

wlanblink 模块属于 SDK 的中间件模块，以 SDK 包为根目录，其代码位置如表 2-1 所示。

表 2-1 XR806 wlanblink 模块代码位置

模块	文件类型	代码位置
wlanblink	source	./src/smartlink/blink/wlan_blink.c
	header	./include/smartlink/blink/wlan_blink.h

3 技术说明

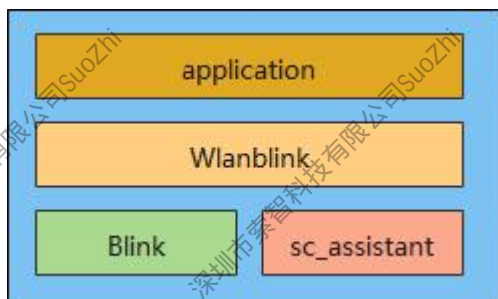
3.1 模块框架

wlanblink 模块的系统层次如图 3-1 所示。

wlanblink 基于 Blink 模块和 sc_assistant 模块实现,其通过 Blink 模块获取配网所需要的 SSID 和密码信息,通过 sc_assistant 模块进行网络连接。

wlanblink 的应用(多种配网共存场景)通过调用 wlanblink 模块提供的接口便可以新增了 Blink 配网方式。

图 3-1 wlanblink 系统层次



说明

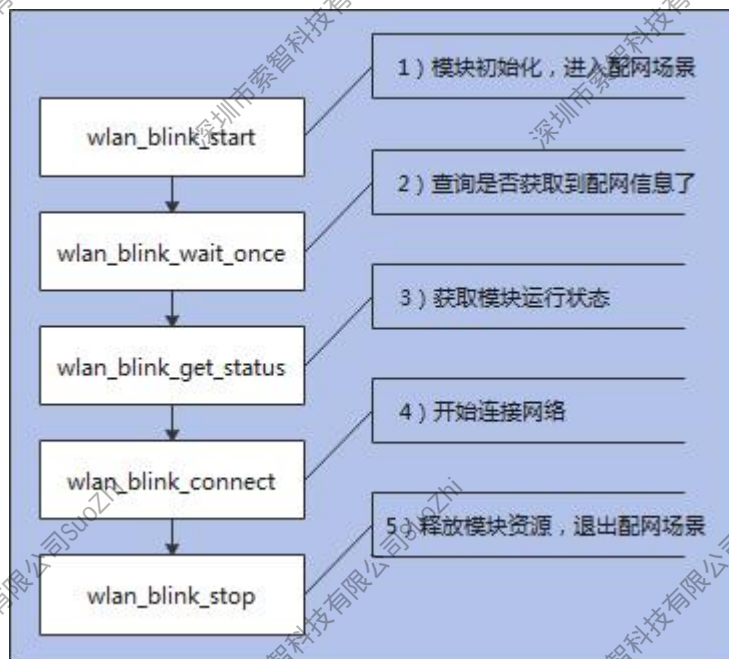
- 1、Blink 模块的使用介绍,请查看文档《XR806_Blink 配网中间件_开发指南》。
- 2、smartlink 的实现可以参考 `sdk/project/common/cmd/cmd_smartlink.c`。
- 3、sc_assistant 是配网助手模块,用于协调各个配网方式的运行。

3.2 使用流程

wlanblink 模块的接口调用流程如下图所示：

- 1) 调用 `wlan_blink_start` 接口进行模块初始化,进入配网场景。
- 2) 在配网场景中,调用 `wlan_blink_wait_once` 去查询是否获取到配网的信息了,若获取到了则通知到 sc_assistant 配网助手,否则可以继续调用该接口等待配网信息的获取。
- 3) 通过 `wlan_blink_get_status` 获取模块的运行状态,例如若获取到模块已经拿到配网信息了,则可以进行网络连接了。
- 4) 在步骤 3) 若模块成功获取到配网信息了,则可以调用 `wlan_blink_connect` 进行网络连接了,网络连接的结果也会通知到 APK 端。
- 5) 配网完成后,调用 `wlan_blink_stop` 接口释放模块资源,退出配网场景。

图 3-2 模块调用流程



4 应用说明

4.1 应用简述

wlanblink 配网方案已经在 XR806 SDK 中实现了，用户通过 wlanblink 模块提供的接口即可开发出基于配网助手的 Blink 配网功能。

4.2 配置说明

wlanblink 配网方式的启用需要使能 SDK 的 BLE 和 WLAN 功能，具体配置项如下表所示。

表 4-1 wlanblink 模块配置列表

配置项	配置说明
BLE 功能使能	设置说明： 此项配置用于在 SDK 中启用 BLE 功能。 设置方式： 在控制台执行命令“make menuconfig”，将以下功能项打开： [*] BLE Controller [*] Ble Host [*] Peripheral Role support [*] Security Manager Protocol support [*] Accept or reject pairing initiative [*] Overwrite oldest keys with new ones if key storage is full [*] Allow unauthenticated pairing for paired device [*] ATT and GATT Options [*] GATT dynamic database support [*] Enable ECDH key generation support [*] Use TinyCrypt library for ECDH
WLAN 功能使能	设置说明： 此项配置用于在 SDK 中启用 WLAN 功能。 设置方式： - 在 prj_config.h 文件，添加或修改以下定义，其中 1 为使能 NET 功能，0 为不使能 NET 功能。 /* network and wlan enable/disable */ #define PRJCONF_NET_EN 1 - 在控制台执行命令“make menuconfig”，将“NET and WLAN”功能项打开。

4.3 接口说明

wlanblink 模块总共提供 5 套接口，主要功能点涵括了进入配网、等待获取配网信息、获取模块运行状态、连接网络、退出配网等过程。

4.3.1 wlan_blink_start

表 4-2 wlan_blink_start 接口函数说明

信息项	说明
原型	wlan_blink_ret_t wlan_blink_start(wlan_blink_param_t *param);
功能	wlanblink 模块的初始化，调用该接口进入配网场景。
参数	param 含义解释：模块的初始化参数。 使用说明：param 结构体的说明，请查看表 4-3。
返回值	wlan_blink_ret_t 类型，关于 wlan_blink_ret_t 的说明请查看表 4-4。

表 4-3 wlan_blink_param_t 结构体的成员说明

成员项	说明
struct netif *nif;	含义解释：指向 netif 结构体对象。

表 4-4 wlan_blink_ret_t 枚举类型说明

成员项	说明
WLAN_BLINK_SUCCESS	执行成功
WLAN_BLINK_FAIL	通用错误
WLAN_BLINK_TIMEOUT	执行超时
WLAN_BLINK_INVALID	无效的参数

4.3.2 wlan_blink_wait_once

表 4-5 wlan_blink_wait_once 接口函数说明

信息项	说明
原型	wlan_blink_ret_t wlan_blink_wait_once(void);
功能	查询等待模块是否获取配网信息成功。若获取成功，函数内部也会通知配网助手更新“获取完成”的状态。
参数	无。
返回值	WLAN_BLINK_SUCCESS 执行成功，其他的失败。

4.3.3 wlan_blink_get_status

表 4-6 wlan_blink_get_status 接口函数说明

信息项	说明
原型	wlan_blink_status_t wlan_blink_get_status(void);
功能	获取模块运行状态。
参数	无。
返回值	wlan_blink_status_t 枚举类型，请查看表 4-7。

表 4-7 wlan_blink_status_t 枚举类型说明

成员项	说明
WB_STATUS_NOTREADY	表示模块还未初始化
WB_STATUS_NORMAL	表示模块初始化了，进入配网场景
WB_STATUS_COMPLETE	表示模块已获取到配网信息了
WB_STATUS_BUSY	表示模块状态忙，即模块正处于连接网络

4.3.4 wlan_blink_connect

表 4-8 wlan_blink_connect 接口函数说明

信息项	说明
原型	wlan_blink_ret_t wlan_blink_connect(uint32_t timeout_ms);
功能	发起网络连接，连接的结果会通知到 Blink 配网工具 APK 端。
参数	timeout_ms 含义解释：网络连接的超时时间。 使用说明：单位是 ms。
返回值	WLAN_BLINK_SUCCESS 执行成功，其他的失败。

4.3.5 wlan_blink_stop

表 4-9 wlan_blink_stop 接口函数说明

信息项	说明
原型	wlan_blink_ret_t wlan_blink_stop(void);
功能	Blink_app 模块的反初始化，调用该接口释放资源、退出配网场景。
参数	无。
返回值	WLAN_BLINK_SUCCESS 执行成功，其他的失败。

5 示例说明

本章节以 cmd_smartlink 添加 Blink 配网为例，简要说明 wlanblink 模块接口的使用。

5.1 示例简介

本示例通过演示 smartlink 配网过程简要介绍 wlanblink 接口的基本使用方法（初始化、获取信息、退出等操作），演示的功能是通过运行 wlan_ble_demo 示例工程，输入命令进入 smartlink 配网，与 Blink APK 配合完成联网操作。

wlan_ble_demo 示例工程代码，位于 XR806 SDK 的/project/demo/bluetooth/wlan_ble_demo 目录。

wlan_ble_demo 示例工程的硬件准备有如下。

1. 评估板：运行示例工程代码。
2. USB 线：用于板子供电
3. 串口线：连接评估板的 Uart0 插针，用于 console 控制台的输入输出。
4. PC 机：用于镜像烧录和 console 控制的输入输出。

wlan_ble_demo 示例工程编译烧写后，复位启动后需要输入命令进入 smartlink 配网，命令说明请查看表 5-1。

表 5-1 ble_demo 示例工程的命令列表

命令	说明
进入 smartlink 配网	命令格式： smartlink start 命令解释： 开始进入 smartlink 配网
结束 smartlink 配网	命令格式： smartlink stop 命令解释： 结束 smartlink 配网

5.2 关键实现

本节主要介绍了如何往 smartlink 添加 BLE 配网方式，代码在 cmd_smartlink.c 文件中实现，具体操作步骤如下：

- 1) 在 smartlink_start 函数中添加 wlan_blink_start 接口，实现了 Blink 配网服务的注册：

```
static int smartlink_start(void)
{
    int ret = 0;
    ...
#ifdef SMARTLINK_USE_BLINK
    wlan_blink_param_t wb_param;
    wlan_blink_status_t wb_status;
#endif

    //注册 blink 配网
#ifdef SMARTLINK_USE_BLINK
    cmd_memset(&wb_param, 0, sizeof(wb_param));
    wb_param.nif = g_wlan_netif;
    wb_status = wlan_blink_start(&wb_param);
    if (wb_status != WB_STATUS_COMPLETE) {
        CMD_DBG("blink start fail!\n");
    }
#endif
    ...
}
```

2) 在 smartlink_task 函数的 while 循环添加 wlan_blink_wait_once 接口，等待数据获取：

```
static void smartlink_task(void *arg)
{
    ...
    while (!(thread_run & SL_TASK_STOP) &&
           OS_TimeBefore(OS_JiffiesToMsecs(OS_GetJiffies()), end_time) &&
           sc_assistant_get_status() < SCA_STATUS_COMPLETE) {
#ifdef SMARTLINK_USE_VOICEPRINT
        voiceprint_wait_once();
#endif

        //等待查询配网信息获取状况
#ifdef SMARTLINK_USE_BLINK
            wlan_blink_wait_once();
#endif

        OS_MSleep(100);
    }
    ...
}
```

3) 在 smartlink_task 函数添加 wlan_blink_connect 接口进行网络连接：

```
static void smartlink_task(void *arg)
{
    ...

    while (!(thread_run & SL_TASK_STOP) &&
           OS_TimeBefore(OS_JiffiesToMsecs(OS_GetJiffies()), end_time) &&
           sc_assistant_get_status() < SCA_STATUS_COMPLETE) {
#ifdef SMARTLINK_USE_VOICEPRINT
        voiceprint_wait_once();
#else
#ifdef SMARTLINK_USE_BLINK
        wlan_blink_wait_once
#endif
        OS_MSleep(100);
    }
    ...

    //网络连接
#ifdef SMARTLINK_USE_BLINK
        if (wlan_blink_get_status() == WB_STATUS_COMPLETE) {
            if (wlan_blink_connect(SMARTLINK_TIME_OUT_MS) != WLAN_BLINK_SUCCESS) {
                CMD_DBG("connect fail\n");
                goto out;
            }
        }
    #endif

    out:
    ...
#ifdef SMARTLINK_USE_BLINK
        wlan_blink_stop();
    #endif
    ...
}
```

4) 在 smartlink_task 函数的出口位置添加 wlan_blink_stop 接口进行 Blink 配网模块资源释放：

```
static void smartlink_task(void *arg)
{
    ...

    #ifdef SMARTLINK_USE_BLINK
        if (wlan_blink_get_status() == WB_STATUS_COMPLETE) {
            if (wlan_blink_connect(SMARTLINK_TIME_OUT_MS) != WLAN_BLINK_SUCCESS) {
                CMD_DBG("connect fail\n");
            }
        }
    #endif
}
```

```

        goto out;
    }
}
#endif

out:
...
//退出 BLE 配网，释放模块资源
#ifdef SMARTLINK_USE_BLINK
    wlan_blink_stop();
#endif
...
}

```

5.3 效果展示

在评估板中运行 wlan_ble_demo 示例程序后，往控制台输入进入 smartlink 配网命令：

```
$ smartlink start
```

控制台打印如下：

```

$ smartlink start
...
<ACK> 200 OK
[cmd] smartlink_task getting ssid and psk...

```

接着使用 Blink.apk 连接设备，往设备发送配网信息，可以看到设备端收到信息后开始进行配网：

```

[cmd] smartlink_task get ssid and psk finished
...
[net INF] no need to switch wlan mode 0
[net INF] msg <wlan scan success>
en1: Trying to associate with 7e:b5:9b:36:3d:9d (SSID='AWTest' freq=2437 MHz)
en1: Associated with 7e:b5:9b:36:3d:9d
WAR drop=1093, fctl=0x00d0.
en1: WPA: Key negotiation completed with 7e:b5:9b:36:3d:9d [PTK=CCMP GTK=TKIP]
en1: CTRL-EVENT-CONNECTED - Connection to 7e:b5:9b:36:3d:9d completed [id=0 id_str=]
[net INF] msg <wlan connected>
[net INF] netif is link up
[net INF] start DHCP...
[net INF] netif is up
[net INF] address: 192.168.0.16
[net INF] gateway: 192.168.7.1

```

```
[net INF] netmask: 255.255.248.0
[net INF] msg <network up>
[net INF] no need to switch wlan mode 0
WAR drop=1093, fctl=0x00d0.
```

通过日志可以看到已经配网成功了，同时 APK 端也显示网络连接成功了。

如果想中途退出 smartlink 配网场景，可以往控制台输入“smartlink stop”命令退出。

著作权声明

版权所有©2020 广州芯之联科技有限公司。保留一切权利。

本文档及内容受著作权法保护，其著作权由广州芯之联科技有限公司（“芯之联”）拥有并保留一切权利。

本文档是芯之联的原创作品和版权财产，未经芯之联书面许可，任何单位和个人不得擅自摘抄、复制、修改、发表或传播本文档内容的部分或全部，且不得以任何形式传播。

商标声明



（不完全列举）均为广州芯之联科技有限公司的商标或者注册商标。在本文档描述的产品中出现的其它商标，产品名称，和服务名称，均由其各自所有人拥有。

免责声明

您购买的产品、服务或特性应受您与广州芯之联科技有限公司（“芯之联”）之间签署的商业合同和条款的约束。本文档中描述的全部或部分产品、服务或特性可能不在您所购买或使用的范围内。使用前请认真阅读合同条款和相关说明，并严格遵循本文档的使用说明。您将自行承担任何不当使用行为（包括但不限于如超压，超频，超温使用）造成的不利后果，芯之联概不负责。

本文档作为使用指导仅供参考。由于产品版本升级或其他原因，本文档内容有可能修改，如有变更，恕不另行通知。芯之联尽全力在本文档中提供准确的信息，但并不确保内容完全没有错误，因使用本文档而发生损害（包括但不限于间接的、偶然的、特殊的损失）或发生侵犯第三方权利事件，芯之联概不负责。本文档中的所有陈述、信息和建议并不构成任何明示或暗示的保证或承诺。

本文档未以明示或暗示或其他方式授予芯之联的任何专利或知识产权。在您实施方案或使用产品的过程中，可能需要获得第三方的权利许可。请您自行向第三方权利人获取相关的许可。芯之联不承担也不代为支付任何关于获取第三方许可的许可费或版税（专利税）。芯之联不对您所使用的第三方许可技术做出任何保证、赔偿或承担其他义务。