wifi配网库使用说明

一：编译说明

代码在主干 master/alexa/thirdapps/wifi

进入wifi目录下，执行./build.sh就可以进行编译

库名称是：libnet\_config.a

二：api接口说明

#define API\_NET\_CONFIG\_AIR\_KISS\_TYPE 0 **//airkiss配网**

#define API\_NET\_CONFIG\_WAVE\_TYPE 1 **//声波配网**

#define API\_NET\_CONFIG\_BT\_TYPE 2 **//ble配网**

#define API\_NET\_CONFIG\_ERROR\_CONNECT\_FINISH 0 **// device connect wifi ok**

#define API\_NET\_CONFIG\_ERROR\_START\_CONNECT 1 **// device start connect wifi**

#define API\_NET\_CONFIG\_ERROR\_START\_SCAN 2 **// device start net setting**

#define API\_NET\_CONFIG\_ERROR\_TIME\_OUT 3 **// device net setting time out**

#define API\_NET\_CONFIG\_ERROR\_LOCK\_OK 4 **// only for airkiss, device lock ok**

#define API\_NET\_CONFIG\_ERROR\_GET\_SSID\_PWD\_OK 5 **// get ssid and pwd ok**

#define API\_NET\_CONFIG\_ERROR\_ERROR 6 **// error**

#define API\_NET\_CONFIG\_ERROR\_EXIT 7 **// device setting exit ok**

1 void api\_net\_config\_start(int type);

开始启动配网任务。异步。内部会创建线程，该函数会立刻返回

Type可以使用：API\_NET\_CONFIG\_AIR\_KISS\_TYPE，API\_NET\_CONFIG\_WAVE\_TYPE，API\_NET\_CONFIG\_BT\_TYPE

要启动哪个就调用一次，要同时启动三个，就调用三次

任务启动后，会有超时等待。如果配网成功，失败，或者超时，会自动退出，释放相关资源，并通过回调返回：API\_NET\_CONFIG\_ERROR\_EXIT 消息

2 void api\_net\_config\_stop(int type);

主动停止配网任务任务。应该层想自动停止的话，使用该接口。但是该接口也是异步的，调用后，不代表就立刻停止了，必须根据 API\_NET\_CONFIG\_ERROR\_EXIT 消息来判断

3 void api\_net\_config\_regist\_callback(net\_config\_callback callback);

应用层注册回调

4 void api\_net\_config\_voice\_input(int type, void \*audio, size\_t size);

声波配网专用，用来输入语音数据，16k 16bit

三：配网app

在微信上搜索公众号：我的小芯宝贝

里面有集成：airkiss和ble配网（还是测试阶段，需要内部注册使用）

四：demo演示

static int app\_net\_callback(int type, int error\_type, char \*ssid, char \*pwd)

{

(void)type;

(void)error\_type;

(void)ssid;

(void)pwd;

return 0;

}

int main(void)

{

api\_net\_config\_regist\_callback(app\_net\_callback);

printf("please inpt 1-3 to test, 1: airkiss, 2:wav, 3:bt\n");

char input = getchar();

#if 1 // airkiss test

if(input == '1')

{

api\_net\_config\_start(API\_NET\_CONFIG\_AIR\_KISS\_TYPE);

sleep(10);

printf("airkiss test exit\n");

api\_net\_config\_stop(API\_NET\_CONFIG\_AIR\_KISS\_TYPE);

}

#endif

#if 1 // wav test

if(input == '2')

{

FILE \*fpp = fopen("/out.pcm", "rb");

g\_flag = 0;

api\_net\_config\_start(API\_NET\_CONFIG\_WAVE\_TYPE);

while(1)

{

char buffer[512] = {0};

int ret = fread(buffer, 1, 512, fpp);

if(ret > 0)

{

api\_net\_config\_voice\_input(API\_NET\_CONFIG\_WAVE\_TYPE, buffer, ret);

}

usleep(1000\*10);

if(g\_flag == 1)

break;

}

printf("wav test exit\n");

api\_net\_config\_stop(API\_NET\_CONFIG\_WAVE\_TYPE);

}

#endif

#if 1 // bt test

if(input == '3')

{

g\_flag = 0;

api\_net\_config\_start(API\_NET\_CONFIG\_BT\_TYPE);

while(1)

{

usleep(1000\*10);

if(g\_flag == 1)

break;

}

printf("bt test exit\n");

api\_net\_config\_stop(API\_NET\_CONFIG\_BT\_TYPE);

}

#endif

sleep(5);

return 0;

}

#endif