

在 **ESP8266** 上使用  
**ESP8266-HomeKit** 搭建 **iOS**  
**Homekit** 指南

Phodal Huang

September 8, 2017

目录

步骤 1: 基本要求 .....	3
步骤 2: 构建 .....	3
目录结构 .....	3
编译 .....	4
wolfSSL .....	5

玩点什么: <https://www.wandianshenme.com>

原文链接:<https://www.wandianshenme.com/play/esp8266-setup-esp8266-homekit-as-homekit-se>

最近在探索基于 ESP8266 构建的智能家居系统, 之前尝试过使用 ESP8266 连接到 Amazon Echo, 便开始寻找适合 ESP8266 的 Homekit 方案。寻觅中, 便找到了 ESP8266-HomeKit, 它在 ESP8266 上采用 API 方法来运行 HomeKit 服务。该项目使用 ESP8266\_RTOS\_SDK 和 WolfCrypt 3.9.8 作为加密。但是, 它不会提供认证的 HomeKit 设备。

本文翻译自官方的构建指南: [Build instructions for ESP8266-HomeKit](#)

## 步骤 1: 基本要求

为了搭建 ESP8266-HomeKit, 我们需要准备下面的这些工具:

- xtensa-lx106-elf-gcc 工具链。推荐的安装方式是在你的计算机上安装 [esp-open-sdk] (<https://github.com/pfalcon/esp-open-sdk>)。使用 `make toolchain esptool libhal STANDALONE=n` 安装
- [ESP8266\\_RTOS\\_SDK](#)

## 步骤 2: 构建

目录结构

### ESP8266/RTOS SDK

1. 如果你是使用 git 克隆 ESP8266\_RTOS\_SDK, 首先编辑: ESP8266\_RTOS\_SDK/.git/info/exclude 在其中添加 /ESP8266-HomeKit\*/。

```
1 esp-open-sdk$ git clone https://github.com/espressif/ESP8266_RTOS_SDK.git
2 esp-open-sdk$ sudo nano ESP8266_RTOS_SDK/.git/info/exclude
```

如下图所示:

*esp8266-homekit-edit-homekit.png*

2. 然后在 ESP8266\_RTOS\_SDK 目录中, 使用 git 克隆 [ESP8266-HomeKit](#) 及 [ESP8266-HomeKit-Demo](#)

```
1 esp-open-sdk$ cd ESP8266_RTOS_SDK
2 ESP8266_RTOS_SDK$ git clone
https://github.com/HomeACcessoryKid/ESP8266-HomeKit.git
```

```
3 ESP8266_RTOS_SDK$ git clone
  https://github.com/HomeACcessoryKid/ESP8266-HomeKit-Demo.github.com
```

3. 这时, 您的文件夹结构应如下所示:

```
1 ESP8266_RTOS_SDK
2 |—— ESP8266-HomeKit
3 |—— ESP8266-HomeKit-Demo
4 |—— LICENSE
5 |—— Makefile
6 |—— README.md
7 |—— bin
8 |—— documents
9 |—— driver_lib
10 |—— examples
11 |—— extra_include
12 |—— include
13 |—— ld
14 |—— lib
15 |—— third_party
16 |—— tools
```

## 编译

跳转到 **ESP8266-HomeKit** 目录, 然后执行 `deploy.sh` 脚本。

```
1 ESP8266_RTOS_SDK$ cd ESP8266-HomeKit
2 ESP8266-HomeKit$ ./deploy.sh
```

它将创建下面的文件:

- **ESP8266\_RTOS\_SDK/lib/libhkc.a**
- **ESP8266\_RTOS\_SDK/include/hkc.h**

如果一切顺利, 你应该会看到:

```
1 ...
2 xtensa-lx106-elf-ar: creating .output/eagle/debug/lib/libhkc.a
3 rm -f -r _libhkc
```

```
4
5 ##### success #####
6 deployed lib/libhkc.a and include/hkc.hESP8266-HomeKit$
```

现在再按照 [ESP8266-HomeKit-Demo wiki](#) 的说明进行下一步。

## wolfSSL

为了方便,需要在 wolfSSL 3.9.8 一个小节里,包含 src (路径: hkc/include/wolfssl) 和 hkc (路径: hkc/wolfcrypt) 文件。

```
1 .
2 └── wolfcrypt
3     ├── COPYING
4     ├── LICENSING
5     ├── Makefile
6     ├── include
7     │   └── wolfssl
8     │       ├── ssl.h
9     │       ├── version.h
10    │       └── wolfcrypt
11    │           ├── arc4.h
12    │           ├── asn.h
13    │           ├── asn_public.h
14    │           ├── chacha.h
15    │           ├── chacha20_poly1305.h
16    │           ├── curve25519.h
17    │           ├── ed25519.h
18    │           ├── error-crypt.h
19    │           ├── fe_operations.h
20    │           ├── ge_operations.h
21    │           ├── hash.h
22    │           ├── hmac.h
23    │           ├── integer.h
24    │           ├── logging.h
25    │           ├── memory.h
26    │           └── misc.h
```

```

27 |         |—— mpi_class.h
28 |         |—— mpi_superclass.h   |         |—— poly1305.h
29 |         |—— random.h
30 |         |—— rsa.h
31 |         |—— settings.h
32 |         |—— settings.h.0
33 |         |—— sha.h
34 |         |—— sha256.h
35 |         |—— sha512.h
36 |         |—— srp.h
37 |         |—— types.h
38 |         |—— visibility.h
39 |         |—— wc_port.h
40 |—— src
41 |   |—— Makefile
42 |   |—— chacha.c
43 |   |—— chacha20_poly1305.c
44 |   |—— curve25519.c
45 |   |—— ed25519.c
46 |   |—— fe_operations.c
47 |   |—— ge_operations.c
48 |   |—— ge_operations.c.0
49 |   |—— hash.c
50 |   |—— hmac.c
51 |   |—— integer.c
52 |   |—— misc.c
53 |   |—— misc.c.0
54 |   |—— poly1305.c
55 |   |—— random.c
56 |   |—— sha256.c
57 |   |—— sha512.c
58 |   |—— srp.c

```

我们必须更改 3 个文件，这些文件将记录在文件树下面。请下载您自己的副本并验证文件的相等性 (\* .o 是原件)

```
1 $ diff ESP8266-HomeKit/wolfcrypt/src/ge_operations.c.0
```

```
ESP8266-HomeKit/wolfcrypt/src/ge_operations.c
2 770c770< static ge_precomp base[32][8] = {
3 ---
4 > static ge_precomp ICACHE_RODATA_ATTR base[32][8] = {
5 2225c2225
6 < static ge_precomp Bi[8] = {
7 ---
8 > static ge_precomp ICACHE_RODATA_ATTR Bi[8] = {
9
10 $ diff ESP8266-HomeKit/wolfcrypt/src/misc.c.0
    ESP8266-HomeKit/wolfcrypt/src/misc.c
11 48,50c48,50
12 < #if !defined(WOLFSSL_MISC_INCLUDED) && !defined(NO_INLINE)
13 <     #error misc.c does not need to be compiled when not defined NO_INLINE
14 < #endif
15 ---
16 > // #if !defined(WOLFSSL_MISC_INCLUDED) && !defined(NO_INLINE)
17 > //     #error misc.c does not need to be compiled when not defined
    NO_INLINE
18 > // #endif
19
20
21 $ diff include/wolfssl/wolfcrypt/settings.h.0
    include/wolfssl/wolfcrypt/settings.h
22 33a34,35
23 > #define WOLFSSL_USER_SETTINGS
24 >
```

请注意: 应删除其他 `src` 文件, 以防止使用过多的 `irom` 大小。

原文链接: <https://www.wandianshenme.com/play/esp8266-setup-esp8266-homekit-as-homekit-ser>