



# SCM1612

## Wi-Fi 6 和 BLE 5 低功耗 SoC

### HTTP 服务器开发指南

---

文档版本 0.1

发布日期 2024-3-11

#### 联系方式

速通半导体科技有限公司 ([www.senscomm.com](http://www.senscomm.com))

[江苏省苏州市工业园区苏州大道西 2 号国际大厦 303 室](#)

销售或技术支持，请发送电子邮件至

[support@senscomm.com](mailto:support@senscomm.com)

### 免责声明和注意事项

本文档仅按"现状"提供。速通半导体有限公司保留在无需另行通知的情况下对其或本文档中包含的任何规格进行更正、改进和其他变更的权利。

与使用本文档中的信息有关的一切责任，包括侵犯任何专有权利的责任，均不予承认。此处不授予任何明示或暗示、通过禁止或以其他方式对任何知识产权的许可。

本文档中的所有第三方信息均按"现状"提供，不对其真实性和准确性提供任何保证。

本文档中提及的所有商标、商号和注册商标均为其各自所有者的财产，特此确认。

© 2024 速通半导体有限公司. 保留所有权利.

## 版本历史

版本	日期	描述
0.1	2024-3-11	初始草案

# 目录

版本历史.....	3
1 开发指南.....	5
1.1 概述.....	5
1.2 示例.....	5
1.2.1 按如下设置构建配置: .....	5
1.2.2 按如下设置 Wi-Fi 参数: .....	5
1.2.3 构建 wise-mcuboot.bin.....	6
1.2.4 运行示例:.....	6
1.2.5 测试 URI:.....	6

# 1 开发指南

本文档旨在帮助实现需要运行 HTTP 服务器的应用程序。

## 1.1 概述

SCM1612 SDK 使用 [ESP-IDF](#) 的 [esp\\_http\\_server](#) 模块：

- API 位于：`lib/net/esp\_http\_server`
- 示例位于：`api/examples/protocols/http\_server`

## 1.2 示例

要运行 HTTP 服务器示例，请按照以下步骤操作：

### 1.2.1 按如下设置构建配置：

- 选择 HTTP 服务器示例作为主应用程序。  
\$ make scm1612s\_defconfig  
\$ make menuconfig
- 导航至 **Applications -> Protocols Demo**
- 选择 **Protocols Demo -> HTTP Server Demo**
- 退出并保存。

### 1.2.2 按如下设置 Wi-Fi 参数：

\$ make menuconfig

- 导航至 **Applications -> Common -> include WI-FI Configuration**
- 输入 **DEMO WI-FI Configuration** 中的参数（如有需要，使用帮助菜单）
- 退出并保存。

### 1.2.3 构建 wise-mcuboot.bin.

\$ make

- 参考 [SDK\\_Getting\\_Started\\_Guide](#) 下载镜像并在 SCM1612 EVK 上运行。

```
WISE 2018.02+ (Mar 11 2024 - 14:50:05 -0700)
I (3114) HTTPD: WIFI CONNECTED
I (3115) SCM_API: AP SSID: Xiaohu_ASUS
I (3116) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (3117) SCM_API: AP CH: 11
I (3118) SCM_API: AP RSSI: -43
I (3119) SCM_API: AP Country : AA
I (3119) SCM_API: Status: CONNECTED
I (9650) HTTPD: WIFI GOT IP
I (9650) HTTPD: Starting webserver
I (9650) HTTPD: Starting server on port: '80'
I (9652) HTTPD: Registering URI handlers
$
$
```

### 1.2.4 运行示例:

- 本示例注册了 3 个 URI, 每个 URI 服务于不同的目的:
  - /hello: 用于响应客户端的 GET HTTP 方法的 URI。
  - /echo: 用于响应客户端的 POST HTTP 方法的 URI。
  - /ctrl: 用于响应客户端的 PUT HTTP 方法的 URI。
- 本示例需要一个 HTTP 客户端, 并且它应该与托管 HTTP 服务器的 SCM1612 EVB 处于同一网络中, 以便它们之间可以连接并交换 HTTP 消息。
- 运行 HTTP 客户端可能有许多不同的选择。在此示例中, 使用了在与 SCM1612 EVB 连接到同一网络的 Linux PC 上的 [curl](#) 命令。

### 1.2.5 测试 URI:

- /hello: 通过发送 GET 请求进行测试。

```
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D: ~  
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl 192.168.51.66:80/hello  
Hello World!thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
```

```
WISE 2018.02+ (Mar 12 2024 - 11:45:53 -0700)  
I (3128) HTTPD: WIFI CONNECTED  
I (3129) SCM_API: AP SSID: Xiaohu_ASUS  
I (3129) SCM_API: AP BSSID: 50:eb:f8:19:88:a0  
I (3130) SCM_API: AP CH: 11  
I (3131) SCM_API: AP RSSI: -37  
I (3132) SCM_API: AP Country : AA  
I (3133) SCM_API: Status: CONNECTED  
I (3171) HTTPD: WIFI GOT IP  
I (3172) HTTPD: Starting webserver  
I (3172) HTTPD: Starting server on port: '80'  
I (3173) HTTPD: Registering URI handlers  
$ I (64944) HTTPD: Found header => Host: 192.168.51.66  
I (64946) HTTPD: Request headers lost
```

- /echo: 通过发送 POST 请求进行测试。

```
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D: ~  
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl -X POST -d "hello, world" 192.168.51.66:80/echo  
hello, worldthomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
```

```
WISE 2018.02+ (Mar 12 2024 - 11:45:53 -0700)
I (3128) HTTPD: WIFI CONNECTED
I (3129) SCM_API: AP SSID: Xiaohu_ASUS
I (3129) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (3130) SCM_API: AP CH: 11
I (3131) SCM_API: AP RSSI: -37
I (3132) SCM_API: AP Country : AA
I (3133) SCM_API: Status: CONNECTED
I (3171) HTTPD: WIFI GOT IP
I (3172) HTTPD: Starting webserver
I (3172) HTTPD: Starting server on port: '80'
I (3173) HTTPD: Registering URI handlers
$ I (64944) HTTPD: Found header => Host: 192.168.51.66
I (64946) HTTPD: Request headers lost

$ I (234434) HTTPD: ===== RECEIVED DATA =====
I (234434) HTTPD: hello, world
I (234435) HTTPD: =====
```

- /ctrl: 通过发送 PUT 请求进行测试。将 "0" 放入 /ctrl 将取消注册 /hello 和 /echo URI，而将 "1" 放入 /ctrl 将注册这些 URI。

```
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D: ~
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl -X PUT -d "0" 192.168.51.66:80/ctrl
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl 192.168.51.66:80/hello
/hello URI is not availablethomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl -X POST -d "hello, world" 192.168.51.66:80/echo
/echo URI is not availablethomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl -X PUT -d "1" 192.168.51.66:80/ctrl
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl 192.168.51.66:80/hello
Hello World!thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ curl -X POST -d "hello, world" 192.168.51.66:80/echo
hello, worldthomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
```



```
WISE 2018.02+ (Mar 12 2024 - 11:45:53 -0700)
I (24760) HTTPD: WIFI CONNECTED
I (24761) SCM_API: AP SSID: Xiaohu_ASUS
I (24762) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (24763) SCM_API: AP CH: 11
I (24763) SCM_API: AP RSSI: -37
I (24764) SCM_API: AP Country : AA
I (24765) SCM_API: Status: CONNECTED
I (24818) HTTPD: WIFI GOT IP
I (24819) HTTPD: Starting webserver
I (24819) HTTPD: Starting server on port: '80'
I (24820) HTTPD: Registering URI handlers
$
$ I (28296) HTTPD: Found header => Host: 192.168.51.66
I (28299) HTTPD: Request headers lost
I (54201) HTTPD: ===== RECEIVED DATA =====
I (54203) HTTPD: hello, world
I (54203) HTTPD: =====
I (73127) HTTPD: Unregistering /hello and /echo URIs
W (78940) httpd_uri: httpd_uri: URI '/hello' not found
W (78942) httpd_txrx: httpd_resp_send_err: 404 Not Found - /hello URI is not available
W (83224) httpd_uri: httpd_uri: URI '/echo' not found
W (83225) httpd_txrx: httpd_resp_send_err: 404 Not Found - /echo URI is not available
I (92089) HTTPD: Registering /hello and /echo URIs
I (96107) HTTPD: Found header => Host: 192.168.51.66
I (96110) HTTPD: Request headers lost
I (100623) HTTPD: ===== RECEIVED DATA =====
I (100625) HTTPD: hello, world
I (100625) HTTPD: =====
```