



SCM1612

Wi-Fi 6 和 BLE 5 低功耗 SoC

mDNS 开发指南

文档版本 0.1

发布日期 2024-3-13

联系方式

速通半导体科技有限公司 (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

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。

Senscomm Confidential

版本历史

版本	日期	描述
0.1	2024-3-13	初稿

目录

版本历史.....	3
1 引言.....	5
1.1 概述.....	5
1.2 构建.....	5
2 API.....	9
2.1 初始化和设置 mDNS 响应器.....	9
2.2 管理 mDNS 响应器的服务.....	11
2.3 开始服务查询.....	13
2.4 停止服务查询.....	14
3 示例.....	16
3.1 连接到 AP.....	16

1 引言

本文档旨在帮助实现需要运行 [mDNS](#) 的应用程序。

1.1 概述

SCM1612 SDK 使用 [lwIP's 的 mdns 端口](#):

- API 和 CLI 位于: `lib/net/mdns`

SCM1612 的 mDNS 模块可以设置并充当 mDNS 响应器，也可以作为 mDNS 查询器启动服务查询作为可选功能。

1.2 构建

要使用 mDNS API 和 CLI，用户应在构建配置中启用相应功能。

```
$ make scm1612s_defconfig  
$ make menuconfig
```

选择 `Kernel -> Networking support -> IPv4 support -> IP: mDNS responder support`

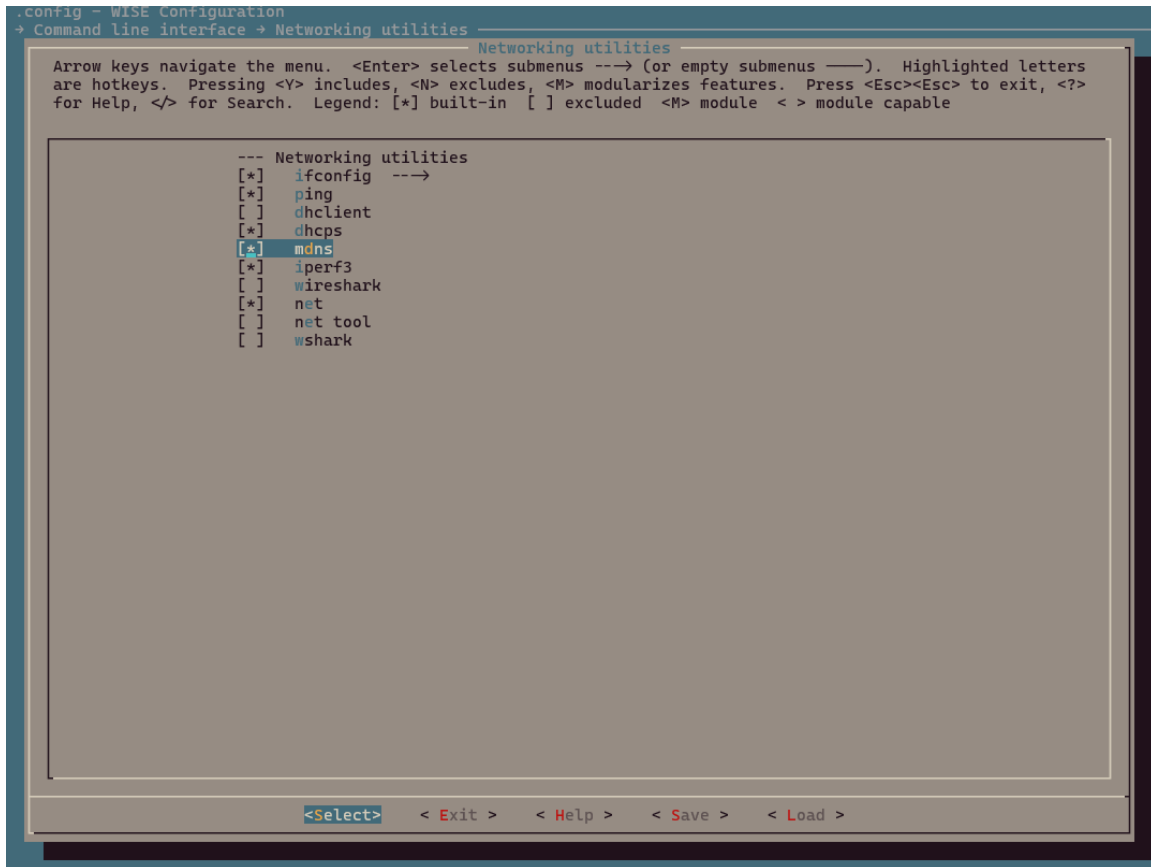
```
.config - WISE Configuration
→ Kernel → Networking support → IPv4 support → IPv4 support

Arrow keys navigate the menu. <Enter> selects submenus --> (or empty submenus ---). Highlighted letters are
hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help,
</> for Search. Legend: [*] built-in [ ] excluded <M> module <> module capable

--
[*] IP: applicaiton-level IPv4 protocol (raw socket)
(255) Raw socket time-to-live
[*] IP: DHCP client support
[ ] DHCP: check ARP on the offered address
[*] DHCP: start DHCP only when the network interface is up
[ ] DHCP: store offered_si_addr and boot_file_name
[ ] DHCP: request NTP servers with discover/select
(1) DHCP: max number of NTP servers requested
(2) DHCP: max number of DNS servers requested
[ ] Use DHCP_OPTION_HOSTNAME with netif's hostname field (?)
(68) DHCP: max number of bytes for options
[*] IP: DHCP server support
(100) DHCP: maximum number of leases
(120) DHCP: lease duration in minutes
[ ] IP: Auto IP support
[ ] IP: enable both DHCP and Auto IP
[*] IP: DNS support
(4) DNS: max DNS entries
(80) DNS: max host name length
(2) DNS: max number of DNS servers
(4) DNS: max DNS retries
[ ] DNS: check name between the query and the response
(7) DNS: security level
[ ] DNS: enable local host-to-address list
[ ] DNS: enable dynamic host list
[ ] DNS: enable mDNS queries (obsolete)
[*] IP: mDNS responder support
(1) Max. number of services (NEW)
(250) Max. number of services (NEW)
(4) Max. number of stored packets (NEW)
[ ] Enable mDNS search for services (NEW)
[ ] IP: SNTP support

<Select> < Exit > < Help > < Save > < Load >
```

选择 `Command Line Interface -> Networking utilities -> mdns`



如果需要处理以.local 结尾的本地域名，还需要
选择 `Kernel->Networking support->Enable lwIP TCP/IP stack->IPv4
support->NDS support->enable mDNS queries (obsolete)`

```
--- IP: DNS support
(4)  DNS: max DNS entries
(80) DNS: max host name length
(2)  DNS: max number of DNS servers
(4)  DNS: max DNS retries
[ ]  DNS: check name between the query and the response
(7)  DNS: security level
[ ]  DNS: enable local host-to-address list
[ ]  DNS: enable dynamic host list
[*]  DNS: enable mDNS queries (obsolete)
```

退出并保存。

构建 wise-mcuboot.bin.

\$ make

请参考 《SDK 入门指南》以下载镜像并在 SCM1612 EVK 上运行。您将能够确认相关的 CLI 命令是否可用。

```
WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$
```


2 API

mDNS API 提供以下一组函数来初始化和设置 mDNS 服务器并发送查询以搜索邻近设备上可用的服务。

- `mdns_resp_init`
- `mdns_resp_add_netif`
- `mdns_resp_remove_netif`
- `mdns_resp_add_service`
- `mdns_resp_del_service`
- `mdns_search_service`
- `mdns_search_stop`

需要注意的是，这些函数必须由 lwIP 核心锁保护，因为有共享资源将由 lwIP TCP/IP 核心线程访问。参考 `mdns.c` 中的 `mdns()` CLI 入口函数以获得参考。

2.1 初始化和设置 mDNS 响应器

- `void mdns_resp_init(void)`

初始化 MDNS 响应器。将在端口 5353 上打开 UDP 套接字。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$
```

- `err_t mdns_resp_add_netif(struct netif *netif, const char *hostname)`

为网络接口激活 MDNS 响应器。

参数	描述
netif	要激活的网络接口。
hostname	要使用的名称。对于主机名的查询， <code>.local</code> 将用该接口的 IP 地址回答。

返回值:
如果 netif 被添加，则为 <code>ERR_OK</code> ，否则为 <code>err_t</code> 。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$
```

- `err_t mdns_resp_remove_netif(struct netif *netif)`

停止在此接口上响应 MDNS 查询，离开多播组，并释放帮助结构及其任何服务。

参数	描述
netif	要移除的网络接口。

返回值
如果 netif 被移除，则为 <code>ERR_OK</code> ，否则为 <code>err_t</code> 。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$
```

2.2 管理 mDNS 响应器的服务

- `s8_t mdns_resp_add_service(struct netif *netif, const char *name, const char *service, enum mdns_sd_proto proto, u16_t port, service_get_txt_fn_t txt_f, void *txt_data)`

为选定的网络接口添加一个服务。

参数	描述
<code>netif</code>	要在其上发布此服务的网络接口。
<code>name</code>	服务的名称。
<code>service</code>	服务类型，如 <code>"_http"</code> 。
<code>proto</code>	服务协议，对于 TCP 为 <code>DNSSD_PROTO_TCP</code> (<code>"_tcp"</code>)，对于其他为 <code>DNSSD_PROTO_UDP</code> (<code>"_udp"</code>)。
<code>port</code>	服务监听的端口。
<code>txt_fn</code>	获取 TXT 数据的回调。每次创建 TXT 回复时都会调用它，以允许动态回复。
<code>txt_data</code>	<code>txt_fn</code> 的用户数据指针。

返回值
如果服务被添加到 netif，则为 Service_id ，否则为 err_t 。

相应的 CLI 命令：

```

WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$

```

- err_t mdns_resp_del_service(struct netif *netif, u8_t slot)

在选定的网络接口上删除一个服务。

参数	描述
netif	服务应该被移除的网络接口。
slot	mdns_resp_add_service 返回的服务槽号。

返回值
如果服务从 netif 中被移除，则为 ERR_OK ，否则为 err_t 。

相应的 CLI 命令：

```

WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$

```

2.3 开始服务查询

- err_t mdns_search_service(const char *name, const char *search, enum mdns_sd_proto proto, struct netif *netif, search_result_fn_t result_fn, void *arg, u8_t *request_id)

在网络上搜索特定服务。

参数	描述
name	服务的名称。
service	服务类型，如 "_http"。
proto	服务协议，对于 TCP 为 DNSSD_PROTO_TCP ("tcp")，对于其他为 DNSSD_PROTO_UDP ("udp")。
netif	发送搜索请求的网络接口。
result_fn	发送收到的答案的回调。对于匹配发送的请求的响应帧的每个答案，都会调用它。
arg	result_fn 的用户数据指针。
request_id	返回的请求标识符，以允许停止它。

返回值
如果搜索请求被创建并发送，则为 <code>ERR_OK</code> ，否则为 <code>err_t</code> 。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$
```

这两个 CLI 命令都使用这个 API，但参数不同。

2.4 停止服务查询

- void mdns_search_stop(u8_t request_id)

停止一个搜索请求。

参数	描述
request_id	要停止的搜索请求。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 08:53:14 -0700)
Hello world!
$
$ mdns
Unknown interface ?
Usage: mdns init
or: mdns ifname start
or: mdns ifname stop
or: mdns ifname service add <name> <service> <proto> <port>
or: mdns ifname service del <id>
or: mdns ifname search browse
or: mdns ifname search info <service> <proto>
or: mdns ifname search stop <id>
$
```

3 示例

目前没有专门的 mDNS 示例应用程序。相反，可以使用上面介绍的 mDNS CLI 命令来测试其功能。

3.1 连接到 AP

Wi-Fi STA CLI 命令可用于连接站点接口（例如 wlan0）到 AP。

参考 《SCM1612_Wi-Fi_Software_Development_Guide》 以使用 Wi-Fi 站点 CLI 命令，因为 Wi-Fi API 函数和 Wi-Fi CLI 命令之间几乎总是一一对应的。


```
$ wifi help
wifi sta_cfg <ssid> <auth> <key> <bssid> <pairwise> <hidden ap>
or: wifi sta_connect
or: wifi sta_disconnect
or: wifi sta_get_connect
or: wifi sta_set_reconnect <enable> <timeout> <period> <count>
or: wifi sta_fast_connect <ssid> <auth> <bssid> <pairwise> <psk> <channel>
or: wifi sta_start
or: wifi sta_get_psk
or: wifi sta_scan
or: wifi sta_advance_scan <scan_type> <channel>|<ssid>|<bssid>
or: wifi sta_scan_results <max_ap_num>
or: wifi sap_start
or: wifi sap_stop
or: wifi sap_cfg <ssid> <key> <ch> <hidden> <auth> <pairwise>
or: wifi sap_beacon <interval>
or: wifi sap_dtim <period>
or: wifi sap_deauth <sta_mac>
or: wifi sap_show
or: wifi sap_showsta
or: wifi ip_set <ifn> <ip> [nm] [gw]
or: wifi dhcp_start/dhcp_stop
or: wifi dhcps_start/dhcps_stop
or: wifi set keepalive <enable> <interval>
or: wifi set powersave <enable> <interval>
or: wifi reg_evt_cb

I (44626) SCM_CLI: help OK (0)
$
$ wifi reg_evt_cb
I (47730) SCM_CLI: reg_evt_cb OK (0)
$
$ wifi sta_start
I (47796) SCM_CLI: STA_STOP
I (47800) SCM_CLI: ifname: wlan0
I (47800) SCM_CLI: sta_start OK (0)
$ I (47801) SCM_CLI: STA_START

$ wifi sta_cfg Xiaohu_ASUS 0 0 00:00:00:00:00:00 1 0
I (47833) SCM_CLI: sta_cfg OK (0)
$
$ wifi sta_connect
I (47855) SCM_CLI: sta_connect OK (0)
$ wifi dhcp_start
I (47857) SCM_CLI: dhcp_start OK (0)
$ I (49543) SCM_CLI: STA_CONNECTED
I (49544) SCM_API: AP SSID: Xiaohu_ASUS
I (49544) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (49545) SCM_API: AP CH: 11
I (49546) SCM_API: AP RSSI: -28
I (49547) SCM_API: AP Country : AA
I (49547) SCM_API: Status: CONNECTED
I (49568) SCM_CLI: WIFI GOT IP
```

现在，mDNS 响应器将回答同一网络上邻近设备的查询。为了查看这一点，可以使用来自与 SCM1612 EVB 连接到同一局域网的 Linux PC 的 [avahi-browse](#)，如下所示。

```
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ avahi-browse -d local -a -r
+ wlp1s0 IPv4 smart_bulb Web Site local
= wlp1s0 IPv4 smart_bulb Web Site local
  hostname = [wise-wlan0.local]
  address = [192.168.51.66]
  port = [1919]
  txt = ["owner=thomas"]
+ enx8e43bb23ad3 IPv4 RT-AX86U-55A0 _alexa._tcp local
= enx8e43bb23ad3 IPv4 RT-AX86U-55A0 _alexa._tcp local
  hostname = [RT-AX86U-55A0.local]
  address = [192.168.50.1]
  port = [80]
  txt = ["version=1" "skillSetupId=8b18386c-1353-4612-9626-714937decf3e"]
+ wlp1s0 IPv4 RT-AX53U-88A0 _alexa._tcp local
= wlp1s0 IPv4 RT-AX53U-88A0 _alexa._tcp local
  hostname = [RT-AX53U-88A0.local]
  address = [192.168.51.1]
  port = [80]
  txt = ["version=1" "skillSetupId=8b18386c-1353-4612-9626-714937decf3e"]
```

由于已启动 mDNS 响应器，因此同一局域网上的任何对等设备（例如这台 Linux PC）都可以使用它的本地主机名，如下所示。

```
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ avahi-browse -d local -a -r
+ enx8e43bb23ad3 IPv4 RT-AX86U-55A0 _alexa._tcp local
+ wlp1s0 IPv4 RT-AX53U-88A0 _alexa._tcp local
+ wlp1s0 IPv4 smart_bulb Web Site local
= wlp1s0 IPv4 smart_bulb Web Site local
  hostname = [wise-wlan0.local]
  address = [192.168.51.66]
  port = [1919]
  txt = ["owner=thomas"]
= enx8e43bb23ad3 IPv4 RT-AX86U-55A0 _alexa._tcp local
  hostname = [RT-AX86U-55A0.local]
  address = [192.168.50.1]
  port = [80]
  txt = ["version=1" "skillSetupId=8b18386c-1353-4612-9626-714937decf3e"]
+ wlp1s0 IPv4 RT-AX53U-88A0 _alexa._tcp local
  hostname = [RT-AX53U-88A0.local]
  address = [192.168.51.1]
  port = [80]
  txt = ["version=1" "skillSetupId=8b18386c-1353-4612-9626-714937decf3e"]
^CGot SIGINT, quitting.
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$ ping wise-wlan0.local
PING wise-wlan0.local (192.168.51.66) 56(84) bytes of data:
64 bytes from 192.168.51.66 (192.168.51.66): icmp_seq=2 ttl=255 time=19.8 ms
64 bytes from 192.168.51.66 (192.168.51.66): icmp_seq=3 ttl=255 time=5.25 ms
64 bytes from 192.168.51.66 (192.168.51.66): icmp_seq=4 ttl=255 time=5.23 ms
64 bytes from 192.168.51.66 (192.168.51.66): icmp_seq=5 ttl=255 time=6.09 ms
64 bytes from 192.168.51.66 (192.168.51.66): icmp_seq=6 ttl=255 time=6.66 ms
64 bytes from 192.168.51.66 (192.168.51.66): icmp_seq=7 ttl=255 time=8.38 ms
^C
--- wise-wlan0.local ping statistics ---
7 packets transmitted, 6 received, 14.2857% packet loss, time 6023ms
rtt min/avg/max/mdev = 5.230/8.574/19.831/5.144 ms
thomas@thomas-900X3C-900X3D-900X3E-900X4C-900X4D:~$
```

‘wise-wlan0’ 是通过 CLI 命令提供给 `mdns_resp_add_netif` 函数的主机名。它可以在用户应用程序中更改为任何合适的名称。