



SCM1612

Wi-Fi 6 和 BLE 5 低功耗 SoC

SNTP 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。

Senscomm Confidential

版本历史

版本	日期	描述
0.1	2024-3-13	初稿

Senscomm Confidential

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 设置和拆除.....

9

2.2 启动时间同步操作.....

10

3 示例.....

12

3.1 连接到 AP.....

12

3.2 按名称配置 SNTP 服务器.....

14

3.3 启动 SNTP 服务.....

14

1 引言

本文档旨在帮助实现需要运行 [SNTP](#) 的应用程序。

1.1 概述

SCM1612 SDK 使用 [lwIP 的 SNTP 端口](#)：

- API 和 CLI 位于: lib/net/sntp

1.2 构建

要使用 SNTP API 和 CLI，用户应在构建配置中启用相应功能。

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

导航至 **Kernel -> Networking support -> IPv4 support -> IP: SNTP support** 并启用它。

```

.config - WISE Configuration
+ Kernel -> Networking support -> IPv4 support -> IPv4 support

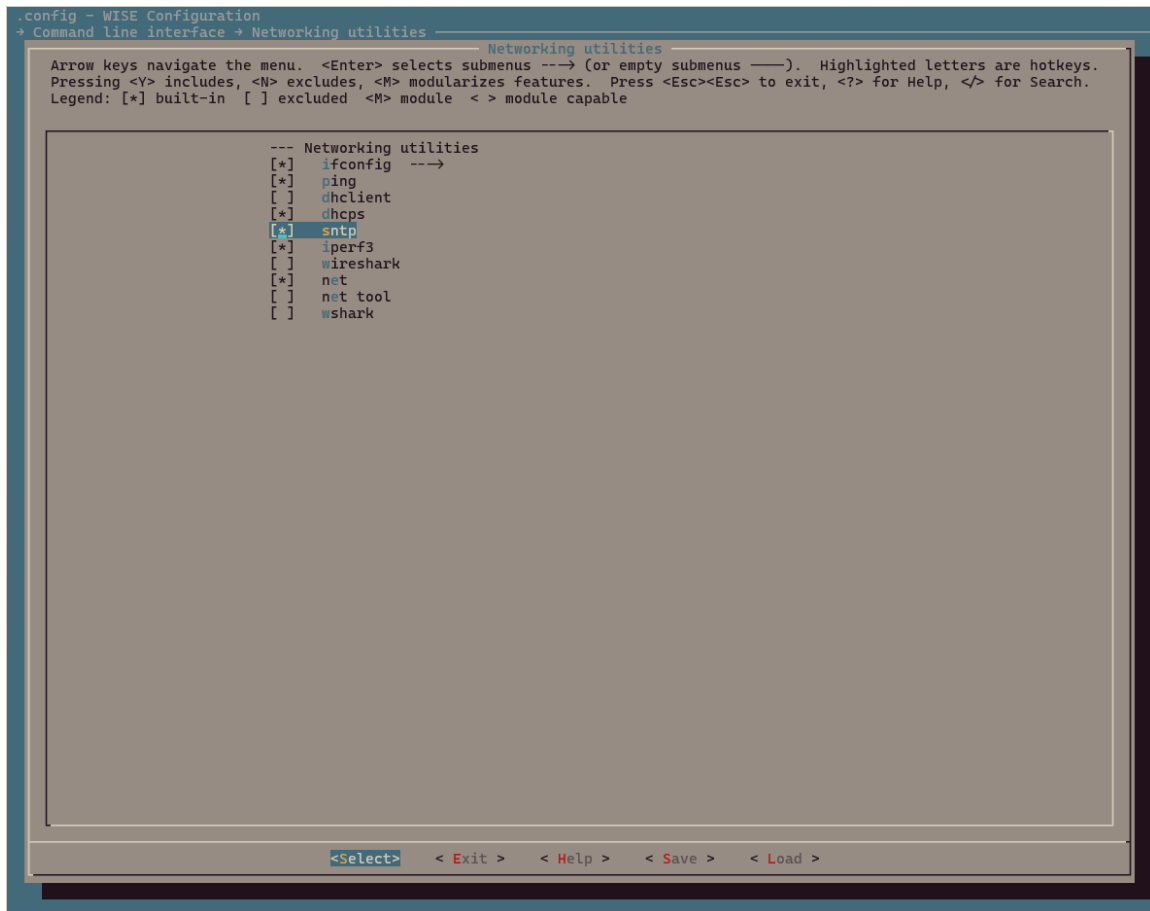
Arrow keys navigate the menu. <Enter> selects submenus --> (or empty submenu ---). Highlighted letters are hotkeys.
Pressing <Y> includes, <N> excludes, <M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </> for Search.
Legend: [*] built-in [ ] excluded <M> module < > module capable

(4) The maximum number of SACK values to include in TCP segments
[*] TCP: effective MSS for transmission
(8) TCP: send buffer space (in unit of MSS)
(2) TCP: send buffer space (in number of pbufs)
[ ] TCP: enable the backlog option for tcp listen pcb
(255) TCP: The maximum allowed backlog for TCP listen netconns
(1) TCP: pre-allocated send buffer size
[ ] TCP: timestamp
[ ] TCP: window scaling
(0) TCP: receive window scaling factor
[ ] TCP: altcp API
[ ] TCP: TLS support for altcp API
[ ] TCP: checksum offloading
-- 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
[*] IP: SNTP support

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

```

同样，导航至 **Command Line Interface -> Networking utilities -> sntp** 并启用它。



退出并保存，然后构建 **wise-mcuboot.bin**:

\$ make

请参考 **SDK 入门指南** 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 - 15:48:27 -0700)
Hello world!
$
$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
$
```

Senscomm Confidential

2 API

SNTP API 提供以下一组函数来设置 SNTP 模块并发送查询以获取网络时间。

- `sntp_setservername`
- `sntp_init`
- `sntp_set_time_sync_notification_cb`
- `sntp_stop`

2.1 设置和拆除

- `void sntp_setservername(u8_t idx, const char *server)`

通过名称初始化其中一个 NTP 服务器。

参数	描述
Idx	要设置的 NTP 服务器的索引，必须小于 NTP_MAX_SERVERS。
server	要设置的 NTP 服务器的 DNS 名称，将在联系时解析。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 15:48:27 -0700)
Hello world!
$
$ sntp
Usage: sntp_setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
$
```

- `void sntp_set_time_sync_notification_cb(void (*callback)(uint32_t sec, uint32_t us))`

安装一个回调函数，当 SNTP 服务器的更新可用时将被执行。

参数	描述
callback	当网络时间可用时将被执行的回调函数

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 15:48:27 -0700)
Hello world!
$
$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
$
```

- void sntp_stop(stop)

停止 SNTP 模块。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 15:48:27 -0700)
Hello world!
$
$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
$
```

2.2 启动时间同步操作

- void sntp_init(void)

初始化此模块并立即发送请求或在 SNTP_STARTUP_DELAY_FUNC) 之后

发送请求。

相应的 CLI 命令：

```
WISE 2018.02+ (Mar 13 2024 - 15:48:27 -0700)
Hello world!
$
$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
$
```

3 示例

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

虽然 SNTP API 会处理 SNTP 协议本身，但应该有一种方法将其与系统时间同步。为此，用户应用程序可以使用 `gettimeofday` 和 `settimeofday` 函数，这也将在此 CLI 示例中展示。

3.1 连接到 AP

Wi-Fi STA CLI 命令可以用于连接站点接口，即 `wlan0`，到 AP。参考 `SCM1612_Wi-Fi_Software_Development_Guide` 以使用 Wi-Fi 站点 CLI 命令。

```
$ 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
```

3.2 按名称配置 SNTP 服务器

使用 CLI 命令 `sntp setserver [name]` 设置 SNTP 服务器。

```
$ I (42514) SCM_CLI: STA_CONNECTED
I (42515) SCM_API: AP SSID: Xiaohu_ASUS
I (42516) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (42517) SCM_API: AP CH: 11
I (42518) SCM_API: AP RSSI: -30
I (42518) SCM_API: AP Country : AA
I (42519) SCM_API: Status: CONNECTED
I (44737) SCM_CLI: WIFI GOT IP

$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
$
$ sntp setserver pool.ntp.org
$
$
```

3.3 启动 SNTP 服务

使用 CLI 命令 `sntp init` 启动 SNTP 服务。

```
$ I (42514) SCM_CLI: STA_CONNECTED
I (42515) SCM_API: AP SSID: Xiaohu_ASUS
I (42516) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (42517) SCM_API: AP CH: 11
I (42518) SCM_API: AP RSSI: -30
I (42518) SCM_API: AP Country : AA
I (42519) SCM_API: Status: CONNECTED
I (44737) SCM_CLI: WIFI GOT IP
```

```
$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop

$ sntp setserver pool.ntp.org

$ sntp init
```

现在，SNTP 客户端请求将被发送到配置的服务器，该服务器将返回当前的网络时间。

CLI 命令 `sntp init` 还注册了一个回调函数 `sntp_set_time_sync_notification_cb`，展示了如何将系统本地时间与 SNTP 服务器返回的网络时间同步。

```
void cli_time_sync_callback(uint32_t sec, uint32_t us)
{
    struct timeval tv;

    /* do nothing for cli test */
    printk("sntp sync: %d.%d\n", sec, us);

    tv.tv_sec = sec;
    tv.tv_usec = us;

    settimeofday(&tv, NULL);
}
```

在此回调函数被调用之后，系统时间将与网络时间同步，这将通过调用 `settimeofday` 来检索。这由 CLI 命令 `sntp time` 如下所示演示：

```
$ I (42514) SCM_CLI: STA_CONNECTED
I (42515) SCM_API: AP SSID: Xiaohu_ASUS
I (42516) SCM_API: AP BSSID: 50:eb:f8:19:88:a0
I (42517) SCM_API: AP CH: 11
I (42518) SCM_API: AP RSSI: -30
I (42518) SCM_API: AP Country : AA
I (42519) SCM_API: Status: CONNECTED
I (44737) SCM_CLI: WIFI GOT IP
```

```
$ sntp
Usage: sntp setserver [server name]
or: sntp getserver
or: sntp init
or: sntp request
or: sntp time
or: sntp stop
```

```
$ sntp setserver pool.ntp.org
```

```
$ sntp init
```

```
$ sntp time
```

```
UTC Time : 2024-03-14 01:20:39
```