



Fusion Embedded™ SNTTP

Many devices connected to a network run applications that rely on time and date in order to operate. Not only must the date and time be set, it must also be accurate.

Most embedded devices do not have a time-of-day clock, so they rely on a timer that generates a clock-tick to simulate the advancement of time. Using a clock-tick to generate time of day results in time-drift and causes applications to be out of synchronization.

Unicoi Systems has the solution for this dilemma in the form of Simple Network Time Protocol (SNTTP). Fusion Embedded SNTTP offers client and server services to obtain and set accurate date and time in network connected devices.

It operates in three selectable modes:

Unicast

A client sends a request to a designated time server at its unicast address and receives a response from which it determines the time.

Multicast

A time server periodically sends a message containing the current time to a designated multicast group address. The client “listens” on this multicast address.

Anycast

A client sends a request to a designated local broadcast address or multicast group address. One or more time servers reply with their own unicast address and the client binds to the first one received. The client continues operation in unicast mode.

With Fusion Embedded SNTTP you can adjust the receive and poll rates, and synchronize your networked devices as often as necessary; and all functions are performed using high level APIs which reduce your product development time significantly. This makes Fusion Embedded SNTTP versatile, easy to use and the perfect solution for any device, from handheld wireless to enterprise networks.

API Function Overview

All it takes to implement SNTTP functionality into your product is seven simple API calls, and Fusion Embedded SNTTP does the rest:

`fns_sntp_client_configure()`
initializes and starts the SNTTP client.

`fns_sntp_get_time()`
is used by the application program to get the current time in standard NTP format (seconds and fractions of seconds).

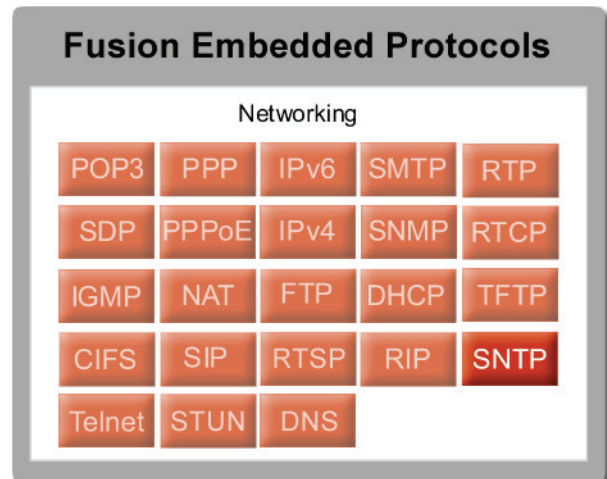
`fns_sntp_set_time()`
sets the internal NTP time, performs time zone correction and stores UTC time.

`fns_sntp_clock_configure()`
is used to set local time zone correction.

`fns_sntp_clock_update()`
is used to update the internal time value.

`fns_sntp_time_milliseconds()`
converts time in milliseconds to time in NTP format.

`fns_sntp_client_terminate()`
is issued by the application program to cease SNTTP operation. To facilitate your product development Fusion SNTTP comes with an example application program.





Fusion Embedded SNTP Features

- Multi-mode time synchronization support multicast, unicast and anycast
- Compatible with NTP and SNTP v3 and v4 clients and servers
- DSP and microprocessor support (16-bit, 32-bit, and 64-bit)
- Easily configurable through use of macros
- Adjustable poll rates
- Uses simple API calls for all functions - no low level programming required
- Custom built code, optimized for embedded systems
- Completely portable (OS/ processor/compiler independent)
- Completely ROMable with re-entrant code
- Mature code deployed in a wide range of applications and backed by many years of embedded networking experience

Fusion Embedded Products

| Networking | | | | | Web | | Security | Voice | File | Reference Designs |
|------------|-------|------|------|------|---------|------|----------|--------------|------|-------------------|
| POP3 | PPP | IPv6 | SMTP | RTP | Browser | DOM | SSL/TLS | Algorithms | NOR | IP Media |
| SDP | PPPoE | IPv4 | SNMP | RTCP | HTML UI | SAX | IPsec | Codecs | NAND | VoIP Phone |
| IGMP | NAT | FTP | DHCP | TFTP | HTTP | SOAP | IKE | Voice Engine | SD | Terminal Adapter |
| CIFS | SIP | RTSP | RIP | SNTP | | | S RTP | | SDHC | VoIP Gateway |
| Telnet | STUN | DNS | | | | | SIPS | | CIFS | |