



CycloneTCP is a dual IPv4/IPv6 stack dedicated to embedded applications. CycloneTCP conforms to RFC standards and offers seamless interoperability with existing TCP/IP systems. By supporting IPv6, CycloneTCP eases deployment of next-generation Internet. The stack is distributed as a full ANSI C and highly maintainable source code. CycloneTCP is available either as open source (GPLv2) or under a commercial license.

Main Features

- Dual stack (IPv4 and/or IPv6)
- Built-in support for multiple network interfaces
- BSD style socket API
- Blocking/non-blocking socket operation and event-driven functions (select and poll)
- Efficient data transfer through zero copy
- Well-crafted TCP module with selective acknowledgement (SACK) and congestion control
- Raw socket interface
- Multicast support (IGMPv2 and MLDv1)
- IP fragmentation and reassembly support
- Flexible memory footprint. Built-time configuration to embed only the necessary features
- High throughput
- Off-load checksum calculation (when supported by hardware) to accelerate IP/TCP/UDP/ICMP checksum generation and verification
- Configurable memory model : Static memory pool or heap memory allocation
- Dialog-based configuration wizard
- Portable architecture (no processor dependencies)

- Straightforward port to any RTOS
- Debugging and trace functionality to ease development and integration
- Highly maintainable source code
- Supports industry-standard microcontrollers with built-in MAC as well as standalone Ethernet controllers

Add-On Modules

- WebSocket client and server
- Auto-IP for dynamic configuration of IPv4 address
- DHCP client
- SLAAC for dynamic IPv6 address assignment and auto-configuration
- DHCPv6 client and relay agent
- Host name resolution (DNS, mDNS and NetBIOS Name Service)
- mDNS and NetBIOS responder
- DNS service discovery (DNS-SD)
- SMTP client for sending e-mails
- FTP client
- Secure FTPS client (both implicit and explicit security modes are supported)
- FTP server
- HTTP server with Server-Sides Includes and CGI scripting for dynamic contents
- Secure Web server (HTTPS)
- MQTT v3.1.1 client (TCP, SSL/TLS, WebSocket and secure WebSocket transport layers supported)
- SNMPv1/v2c/v3 agent (MD5/SHA-1 authentication and DES/AES privacy protocols are supported)
- SNTP client (Network time synchronization)
- TFTP server
- Standard Internet services (Echo, Discard, Chargen and Daytime)
- PPP (Point-to-Point) protocol
- Icecast/SHOUTcast client for streaming audio over the Internet

Related products

- CycloneSSL (lightweight SSL/TLS library)
- CycloneCrypto (cryptographic toolkit)

Supported Devices

MCUs with 10/100 Ethernet MAC:

Atmel®	AVR32 UC3A SAM3X SAM4E SAM7X SAM9263 SAME70 SAMV71
Freescale®	Coldfire V2 Kinetis K6x Kinetis K7x Kinetis KV5x
Infineon®	XMC4500 XMC4700 XMC4800
Microchip®	PIC32MX PIC32MZ EC PIC32 MZ EF
Microsemi®	SmartFusion SmartFusion2
NXP®	LPC1700 LPC1800 LPC2300 LPC4300
Renesas®	RX62N RX63N RX64M Synergy S7
Spansion®	FM4
ST®	STM32F107 STM32F207/217 STM32F407/417 STM32F427/437 STM32F429/439 STM32F746/756 STR912

MCUs with 10/100 Ethernet MAC+PHY:

Texas Instruments®	Stellaris LM3S6000 Stellaris LM3S9000 Tiva TM4C129X
--------------------	---

MPUs with 10/100 Ethernet MAC:

Atmel®	SAMA5D2 SAMA5D4
Renesas®	RZ/A1L RZ/A1M RZ/A1H

MPUs with Gigabit Ethernet MAC:

Atmel®	SAMA5D3
Texas Instruments®	Sitara AM335x

DSPs with 10/100 Ethernet MAC:

Texas Instruments®	Concerto F28M35x OMAP-L138
--------------------	-------------------------------

Processor IP cores with 10/100 Ethernet MAC:

Cortus®	APS1/3/5 FPS6
---------	------------------

Standalone 10/100 Ethernet Controllers:

Davicom®	DM9000A/B
Micrel®	KSZ8851
Microchip®	ENC28J60 ENC424J600 ENC624J600

Wi-Fi® modules (802.11b/g/n):

Atmel®	WILC1000 WINC1500
Broadcom®	BCM43362
Microchip®	MRF24WG0MA/B
Murata®	SN8000 SN8205
MXCHIP®	EWM3162

Ethernet PHY transceivers:

Davicom®	DM9161
Micrel®	KSZ8031 KSZ8041 KSZ8051 KSZ8081 KSZ8721
National Semiconductor®	DP83620 DP83848
Renesas®	uPD6061x
SMCS®	LAN8710 LAN8720 LAN8740 LAN8742
ST®	ST802RT1A

Gigabit Ethernet PHY transceivers:

Atheros®	AR8031
Micrel®	KSZ9031

Reference Standards

Network Layer (IPv4)

- RFC 791: Internet Protocol Specification
- RFC 792: Internet Control Message Protocol Specification
- RFC 815: IP Datagram Reassembly Algorithms
- RFC 826: Ethernet Address Resolution Protocol
- RFC 1112: Host Extensions for IP Multicasting
- RFC 1122: Requirements for Internet Hosts - Communication Layers
- RFC 2113: IP Router Alert Option
- RFC 2236: Internet Group Management Protocol, Version 2

Network Layer (IPv6)

- RFC 2460: Internet Protocol, Version 6 (IPv6) Specification
- RFC 2464: Transmission of IPv6 Packets over Ethernet Networks
- RFC 2710: Multicast Listener Discovery (MLD) for IPv6
- RFC 3484: Default Address Selection for Internet Protocol version 6 (IPv6)
- RFC 3493: Basic Socket Interface Extensions for IPv6
- RFC 4291: IP Version 6 Addressing Architecture
- RFC 4294: IPv6 Node Requirements
- RFC 4443: Internet Control Message Protocol Version 6 (ICMPv6) Specification
- RFC 4861: Neighbor Discovery for IP version 6 (IPv6)
- RFC 4862: IPv6 Stateless Address Autoconfiguration

Transport Layer

- RFC 768: User Datagram Protocol
- RFC 793: Transmission Control Protocol
- RFC 2018: TCP Selective Acknowledgment Options
- RFC 5681: TCP Congestion Control
- RFC 6298: Computing TCP's Retransmission Timer

Application Layer

- RFC 959: File Transfer Protocol (FTP)
- RFC 1035: Domain Names – Implementation and Specification
- RFC 1945: Hypertext Transfer Protocol - HTTP/1.0
- RFC 2131: Dynamic Host Configuration Protocol
- RFC 2132: DHCP Options and BOOTP Vendor Extensions
- RFC 2616: Hypertext Transfer Protocol - HTTP/1.1
- RFC 2617: HTTP Authentication: Basic and Digest Access Authentication
- RFC 2818: HTTP Over TLS
- RFC 3207: SMTP Service Extension for Secure SMTP over Transport Layer Security
- RFC 3315: Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
- RFC 3646: DNS Configuration options for DHCPv6
- RFC 4954: SMTP Service Extension for Authentication
- RFC 5321: Simple Mail Transfer Protocol
- RFC 6762: Multicast DNS