CycloneTCP

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.

Main Features

- Dual stack (IPv4 and/or IPv6)
- Built-in support for multiple network interfaces
- Flexible memory footprint (built-time configuration to embed only the necessary features)
- Configurable memory model (static memory pool or heap memory allocation)
- Portable architecture (no processor dependencies)
- Straightforward port to any RTOS
- Highly maintainable source code
- Debugging and trace functionality to ease development and integration
- 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
- IP fragmentation and reassembly support
- Support for virtual interfaces (multiple MAC addresses per physical interface)
- Support for multi-homed hosts (multiple IPv4 addresses per interface)
- Ethernet port multiplication using VLAN tagging (SMSC switches) or tail tagging (Micrel switches)
- VLAN support (802.1q and 802.1ad)
- USB Device RNDIS class driver (for STM32 microcontrollers)

Supported Protocols

- DNS client
- NetBIOS client and responder
- LLMNR client and responder
- mDNS client and responder
- DNS-SD responder (DNS-based service discovery)
- DHCP client and server
- Auto-IP (dynamic configuration of IPv4 link-local addresses)
- DHCPv6 client and relay agent
- SLAAC (IPv6 stateless address autoconfiguration)
- Multicast support (IGMPv2 and MLDv1)
- FTP / FTPS client and server (implicit TLS and explicit TLS modes supported)
- HTTP / HTTPS client
- HTTP / HTTPS server with SSI, CGI scripting and WebSocket support
- HTTP/2 client (including HPACK compression, server push and https scheme)
- MQTT v3.1.1 client (TCP, TLS, WebSocket and secure WebSocket transport layers supported)
- MQTT-SN client (UDP and DTLS transport layers supported)
- CoAP client and server (DTLS-secured CoAP, Observe and Block-Wise Transfers supported)
- SMTP client
- SNTP client (network time synchronization)
- SNMP agent (SNMPv1, SNMPv2c and SNMPv3 supported)
- Remote management of SNMP users and access rights (SNMP-USM-MIB and SNMP-VACM-MIB)
- Standard MIBs: MIB-II, IF-MIB, IP-MIB, TCP-MIB, UDP-MIB, SNMPv2-MIB
- TFTP client and server
- Modbus/TCP client and server (Modbus/TCP security supported)
- WebSocket client and server (WebSocket connections tunneled over SSL/TLS supported)
- PPP (Point-to-Point Protocol)

Supported Processors

- ARM7TDMI / ARM926EJ-S
- ARM Cortex-M3
- ARM Cortex-M4
- ARM Cortex-M7
- ARM Cortex-R4
- ARM Cortex-A5
- ARM Cortex-A8
- ARM Cortex-A9
- RISC-V
- MIPS M4K
- MIPS microAptiv
- PowerPC e200
- Coldfire V2
- RX600
- AVR32
- Xtensa LX6

Supported Compilers / Toolchains

- GNU GCC / Makefile
- Atollic TrueSTUDIO
- IAR Embedded Workbench
- Keil MDK-ARM
- Microsoft Visual Studio
- Segger Embedded Studio
- AC6 System Workbench for STM32 (SW4STM32)
- Atmel Studio
- Infineon DAVE
- Microchip MPLAB X
- NXP MCUXpresso
- Renesas e2Studio
- ST STM32CubeIDE
- TI Code Composer Studio (CSS)

Supported Operating Systems

- Amazon FreeRTOS
- ChibiOS/RT
- CMSIS-RTOS
- CMSIS-RTOS2 (RTX v5 and FreeRTOS)
- Keil RTX
- Micrium µC/OS-II
- Micrium µC/OS-III
- Segger embOS
- SYS/BIOS (TI-RTOS)
- Bare Metal programming (without RTOS)

Data Link Layer (PPP)

- RFC 1332: The PPP Internet Protocol Control Protocol (IPCP)
- RFC 1334: PPP Authentication Protocols
- RFC 1661: The Point-to-Point Protocol (PPP)
- RFC 1662: PPP in HDLC-like Framing
- RFC 1994: PPP Challenge Handshake Authentication Protocol (CHAP)
- RFC 2472: IP Version 6 over PPP

Network Layer (IPv4)

- RFC 791: Internet Protocol Specification
- RFC 792: Internet Control MessFFage 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
- RFC 3927: Dynamic Configuration of IPv4 Link-Local Addresses
- RFC 5227: IPv4 Address Conflict Detection

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
- RFC 6106: IPv6 Router Advertisement Options for DNS Configuration

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 1157: A Simple Network Management Protocol (SNMP)
- RFC 1213: Management Information Base for Network Management of TCP/IP-based internets (MIB-II)
- RFC 1350: The TFTP Protocol (Revision 2)
- RFC 1769: Simple Network Time Protocol (SNTP)
- RFC 1905: Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)
- 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 2863: The Interfaces Group MIB
- RFC 3207: SMTP Service Extension for Secure SMTP over Transport Layer Security
- RFC 3315: Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
- RFC 3410: Introduction and Applicability Statements for Internet Standard Management Framework
- RFC 3411: An Architecture for Describing SNMP Management Frameworks
- RFC 3412: Message Processing and Dispatching for the SNMP
- RFC 3413: Simple Network Management Protocol (SNMP) Applications
- RFC 3414: User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)
- RFC 3415: View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)
- RFC 3418: Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)
- RFC 3584: Coexistence between Version 1, Version 2, and Version 3 of SNMP Framework
- RFC 3646: DNS Configuration options for DHCPv6
- RFC 3826: AES Cipher Algorithm in the SNMP User-based Security Model
- RFC 4022: MIB for the Transmission Control Protocol (TCP)
- RFC 4113: MIB for the User Datagram Protocol (UDP)
- RFC 4293: MIB for the Internet Protocol (IP)
- RFC 4795: Link-local Multicast Name Resolution (LLMNR)
- RFC 4954: SMTP Service Extension for Authentication
- RFC 5321: Simple Mail Transfer Protocol
- RFC 6455: The WebSocket Protocol
- RFC 6762: Multicast DNS
- RFC 6763: DNS-Based Service Discovery
- RFC 7252: The Constrained Application Protocol (CoAP)
- RFC 7540: Hypertext Transfer Protocol Version 2 (HTTP/2)
- RFC 7541: HPACK Header Compression for HTTP/2
- RFC 7641: Observing Resources in the Constrained Application Protocol (CoAP)
- RFC 7860: HMAC-SHA-2 Authentication Protocols in the User-based Security Model
- RFC 7959: Block-Wise Transfers in the Constrained Application Protocol (CoAP)