

# Turbo Treck TCP/IP

product information

www.elmic.com

## Included protocols:

- TCP
- UDP
- ICMP
- IPv4
- ARP
- Ethernet
- SLIP
- RIPv2 Listener
- Ping Application

## The Seven Layer OSI Reference Model:

|  |                           |
|--|---------------------------|
| Application Layer<br><i>User Applications</i>            | <b>Turbo Treck TCP/IP</b> |
| Presentation Layer<br><i>Upper Layer Protocols (RIP)</i> |                           |
| Session Layer<br><i>BSD 4.4 Sockets</i>                  |                           |
| Transport layer<br><i>TCP, UDP</i>                       |                           |
| Network Layer<br><i>ARP, IP, ICMP</i>                    |                           |
| Data Link Layer<br><i>Ethernet</i>                       |                           |
| Physical Layer<br><i>Network Interface Hardware</i>      |                           |

*Turbo Treck TCP/IP incorporates the protocols that you need to get your device real-time connected to the Internet. Optional Data Link protocols—such as Turbo Treck PPP—can be used to establish dial-up connections. You can also extend Turbo Treck TCP/IP with your own Data Link Protocol.*

## Highest Performance Available

- Zero copy to and from the application.
- Zero copy inside of the protocol stack (even for TCP).
- Zero copy to and from the device driver; uses advanced features of device drivers, such as scattered data write (if supported by the device).
- Minimal RTPS/Kernel calls.
- Optimized for speed along the send/receive code paths.
- Extremely fast checksum computation
- Turbo Treck Learn to allow the protocol stack to learn your performance requirements at run-time.
- Extremely fast TCP code (compatible with any other TCP implementation).

## Designed for Embedded Systems

- Very small critical sections (most of the time less than 5 assembly instructions).
- Small code footprint (as small as 32K).
- Re-entrant and “Romable”.
- All of the RAM used by the stack is initialized by the stack.
- Easy to integrate RTOS/Kernel interface.
- Easy to use device driver API.
- Can run without an RTOS.
- Can run as its own task.
- Can run within the context of other tasks as a shared library.
- High bandwidth/high delay support (e.g. 32 bit window support) needed for set-top.

## Highest Quality Embedded TCP/IP Available Anywhere

- Fully tested and qualified with over 300 different tests using many different compilers and processors.
- RFC2000 compliant
- Easy to integrate with your RTOS/Kernel.
- No porting required. You will never need to change a single line of our source code of the protocol stack to integrate Turbo Treck TCP/IP into your environment.
- Everything can be set at run-time, including adding device drivers.
- All default settings can be set at run-time and/or compile time.
- Fully ANSI “C” compliant.
- Warning-free Build (even with lint checking).
- Can be built for size or speed.
- Single Header file for application code, interface code, and device drivers,

## Requirements to use Turbo Treck TCP/IP in your design

- ANSI compliant “C” compiler
- Any CISC/RISC processor
- Sufficient ROM/FLASH for the protocol stack.
- Sufficient RAM for the maximum number of concurrent connections you will support.
- A network device
- A timer to signal the protocol.

*For more information, please visit [www.elmic.com](http://www.elmic.com).*

