eZ80Acclaim!® Family of Microcontrollers

Zilog Full-Feature TCP/IP Software Suite

Product Brief

PB015411-0910

Introduction

Zilog TCP/IP (ZTP) solution is an integrated, pre-emptive, and a multitasking operating system and TCP/IP protocol software suite that has been optimized for embedded systems. ZTP works in conjunction with the award-winning eZ80Acclaim!® family of Flash microcontrollers. It also provides standard network connectivity in a wide range of applications, such as industrial control, automation, facility management, IP appliances, and remote systems communication.

ZTP software suite is optimized to allow development of the low-cost systems. Powered by new-Zilog Real-Time Kernel (RZK), ZTP offers full-feature operating system services in addition to network services while occupying minimal program memory. The applications can be integrated with ZTP via the easy-to-use and well-documented application programming interface (API).

ZTP supports the following TCP/IP protocols:

- IPv4, TCP, UDP, ARP, and RARP
- ICMP, IGMP, and DHCP/BOOTP

ZTP supports three physical interfaces:

1. Ethernet 10/100
2. PPP
   (a) on serial link with HDLC
   (b) on Ethernet with PPPoE
3. 802.11 b/g Wireless LAN (Wireless interface supports WEP, WPA, and WPS2)

Building upon these field-proven core protocols, ZTP offers a rich-set of higher-level TCP/IP services to accelerate application development, and it includes the following:

- HTTP/HTTPS server, TFTP client, SNMP agent, TELNET server and client, SMTP client, DNS client, TimeP client, and SNTP client.
- FTP server and client, and an embedded Flash file system that works within Flash and static RAM memories.
- Local or remote run-time debugging operating system command shell.

Industry-standard socket interface and native RZK APIs are the primary interface available to the applications to interface with the TCP/IP stack. These interfaces accelerate development time for network programmers who are already familiar with the socket interface. For new TCP/IP programmers, the native RZK APIs shorten the learning process. In addition, an internal mapping layer maintains a set of backward-compatible APIs to previous ZTP releases.

ZTP is royalty-free, configurable, scalable, and modular in design; it provides a rich-set of features via easy-to-use and well-documented APIs. In addition, features have been highly optimized to stringent memory and performance requirements of the typical 8-bit embedded applications.

ZTP is available in a standard release package with an object/library. It is also available in a source release package with full source code.

ZTP is currently available for Zilog's eZ80Acclaim! family of devices, including the eZ80F91, eZ80F92 and eZ80F93 microcontrollers, and the eZ80L92 microprocessors.

Architecture

ZTP software system is designed to be highly modular and scalable. It is broadly classified into three major subsystems:

1. **ZTP Drivers—**The ZTP driver subsystem is comprised of drivers for devices, such as Ethernet, UART, I^2C, SPI, RTC, WLAN and protocol drivers such as, PPP and HDLC.
2. **ZTP Core—**The ZTP core subsystem is comprised of TCP/IP protocols and Zilog's real-time operating system. ZTP supports stream connections (TCP), datagram connections (UDP), and secure connections (SSL).
3. **ZTP Applications—**Most ZTP applications are comprised of high level application protocols that use core components to perform standard network functions. An optional run-time shell interface is provided so that you can initiate shell commands using either a local RS-232 terminal or a remote terminal via Telnet.

The shell implements a rich-set of informative debugging commands. In addition, custom shell commands can be added. Many of the ZTP functions are initiated or monitored via the shell during run-time. For example, if a remote host initiates a Telnet client request, the ZTP shell provides a network virtual terminal to the remote host.

Another component of the ZTP applications subsystem is an embedded Flash file system, which supports multiple-disk volumes and operates in the available Flash and RAM memories. The Flash file system facilitates easy handling of files over a network connection such as FTP and HTTP/HTTPS, and can be employed by the application to manage data files and directories. The Flash file system supports standard C APIs, such as `fopen`, `fclose`, etc.

ZTP application developers have the flexibility to use either the native RZK programming interface or a BSD socket-like interface to communicate with the network. The highly configurable nature of ZTP thereby enables you to include only those components that are appropriate to the application and exclude components that are not required to build the final binary image.

Figure 1 displays the ZTP architecture.

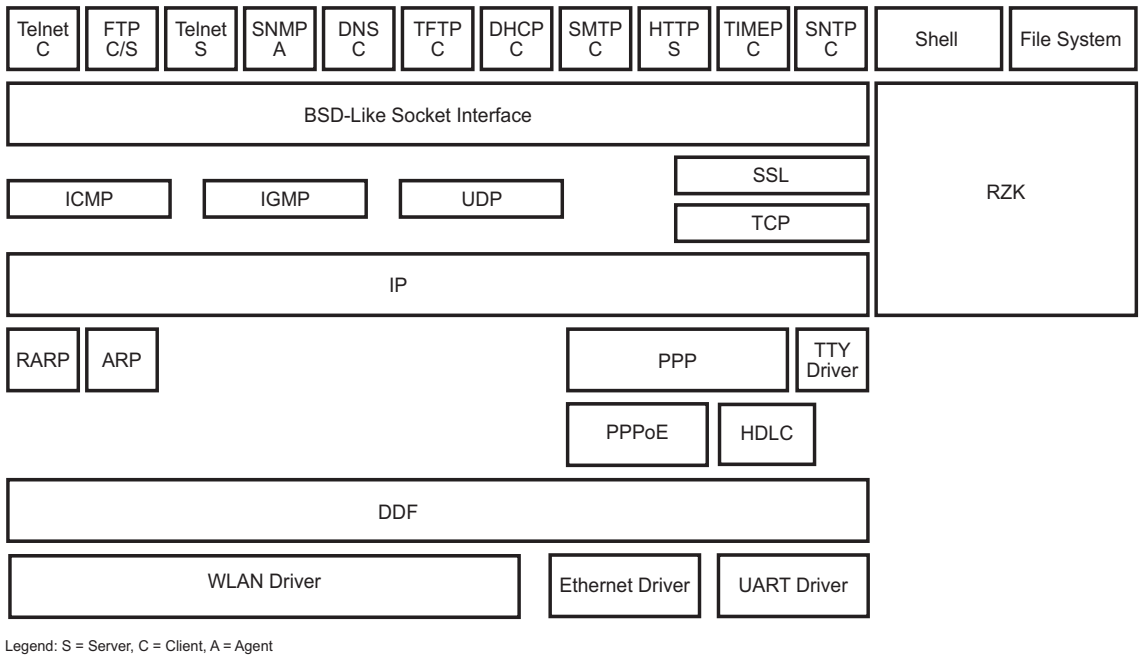

Figure 1. ZTP Architecture Block Diagram

Features

ZTP has been optimized with the requirements of typical 8-bit applications in mind. In addition to its industry standard RFC-compliant protocols, the key features of ZTP include:

HTTP Server—An application protocol to retrieve hypertext documents from the World Wide Web. Java Applets, CGI calls as well as many standard website objects/file types are supported. It also supports HTTP1.1 Keep alive mechanism and HTTP1.1 basic and message digest (MD5 based) scheme authentication.

HTTPS Server—An application protocol that adds data privacy, data integrity and authentication features to HTTP using SSL technology (optional: included in separate SSL plug-in package).

Telnet Client—An application protocol that makes possible for you to connect and run programs on remote hosts. The protocol implements options such as echo and suppress go ahead.

Telnet Server—An application protocol that allows remote users to Telnet into the system.

FTP Client/Server—An application protocol that allows you to transfer files from one system to another system over a TCP/IP network. This protocol operates seamlessly with RZK embedded Flash file system.

SNMP Agent—An application protocol that makes network management information available to SNMP managers. ZTP Implements SNMPv1, SNMPv2, and SNMPv3. SNMPv2 includes an additional GetBulk command and SNMPv3 implements authentication and encryption.

TFTP Client—Trivial file transfer protocol (TFTP) is a simple form of the FTP. The protocol is implemented on top of user datagram protocol (UDP). Since it is small, it lacks most of the features of FTP.

DHCP Client—An application protocol that dynamically requests an IP address from an available DHCP server on a TCP/IP network.

SMTP Client—An application protocol that is used to transfer mail reliably and efficiently. Supports challenge response (CRAM-MD5 based) digest scheme authentication.

TimeP—An application protocol that retrieves a time reference from a remote TimeP server on a TCP/IP network.

SNTP Client—An application protocol that gets time from the time server.

PPPoE Client—PPPoE uses the features of Ethernet and PPP standards. If PPPoE is used, 'dial' from one machine to another over an Ethernet network using the DSL line, and establishes a point-to-point connection between the host and target.

Refer to the relevant ZTP documentation for complete descriptions of the above features.

High Performance and Fast Context Switching

ZTP is powered by RZK, which can perform a context switch between threads in less than 12.5 µs (measured on a 50 MHz eZ80F91 MCU running with 1 wait-state in eZ80 bus mode). On this host platform, ZTP's high-performance core can transmit data on a UDP socket with a speed of 6 Mbps. In the current ZTP release, this performance is reflected in the performance of the higher-layer network. Refer to the latest *ZTP Release Notes* for updated information.

Small Memory Footprint

The memory footprint for the basic ZTP core is less than 60 KB. The memory footprint of ZTP running application protocols varies based on the components being included. RAM requirements for ZTP depends on, but are not limited to, actual configuration, such as the number of application threads and the number of TCP/UDP connections supported at run-time.

Modular, Scalable, and Configurable

ZTP is highly configurable and scalable. You can derive a light-to full-featured version per application requirements. The modular design of ZTP enables you to include only those library objects

that are required by the application, thereby saving memory space. The standard ZTP package is released as a set of libraries.

TCP/IP parameters, such as the number of UDP/TCP connections and the size of the UDP/TCP/ buffer, are configurable. System parameters, such as the number of objects required (threads, semaphores, etc) can be specified at compile time. ZTP allocates memory for the object control blocks based on the number of objects specified in the configuration file. All of these compile-time configuration options make ZTP fully configurable, thereby enabling to fine-tune it to the requirements.

For complete information about ZTP configuration, refer to *Zilog TCP/IP Software Suite Programmer's Guide (RM0041)*.

Portability

ZTP applications can be easily migrated/ported to a custom eZ80® CPU-based hardware platform. Extensive user documentation is available from the standard ZTP release. For complete information about ZTP configurations, refer to the *relevant ZTP manual*.

Powerful Kernel

Zilog's latest RZK supports both preemptive and cooperative multitasking. The RZK scheduler runs tasks of equal priority in round-robin sequence. Thread priority can be changed at run-time. RZK offers the following mechanisms for inter-thread communication and synchronization:

Semaphores—RZK provides both binary as well as counting semaphores. RZK implements a priority inheritance protocol to solve priority inversion issues.

Message Queues—RZK provides message queues for asynchronous communication between threads. High-priority messages can be placed in front of the queue.

Event Groups—Event groups are provided for control synchronization and do not carry any information. An event group can accommodate a maximum of 24 events.

Memory Management—Fixed-size memory pools are provided for deterministic memory allocation, which are very useful for time-critical applications. RZK also supports variable-size memory allocation.

For complete information about the RZK kernel, refer to the following documents available for download at www.zilog.com.

- Zilog Real-Time Kernel Product Brief (PB0155)
- Zilog Real-Time Kernel User Manual (UM0075)
- Zilog Real-Time Kernel Reference Manual (RM0006)

Demonstration Examples

The standard ZTP package includes sample applications that demonstrate a number of ZTP features for easy reference. These demo applications can be used as a reference to start a new project with a new configuration. These samples are simple interactive applications that are used to demonstrate:

- Socket programming for network data transfer over Ethernet and PPP interfaces.
- Raw data transfer ability of ZTP over serial peripherals such as UART, I^2C, and SPI using RZK's DDF.
- A simple webserver application.
- An FTP client demo.
- A minimal configuration demo.
- A Ping API utility demo.
- A TCP and UDP socket programming demo.

These examples are available with ZDS II tool chains for the entire family of eZ80Acclaim!® microcontrollers.

Development Tools

ZTP is built upon Zilog's Integrated Development Environment (ZDS II) which provides code editing, assembler/C-compiler/linker features,

source-level debugging, and project managing facilities for the quick and efficient development of embedded applications. ZDS II is included with all eZ80Acclaim!® development kits.

Packaging

The standard ZTP package for eZ80Acclaim! MCUs is supplied as C object library modules along with the necessary source files and application examples. A package with full source code is also available. For part number references, refer to *Zilog TCP/IP Stack API Reference Manual (RM0040)*.

Documentation

The following documents comprehensively describe all the features, functions, and usage of ZTP, and are available with each release.

Zilog TCP/IP Software Suite Quick Start Guide (QS0049)—Enables you to install and get started with ZTP. It guides you through running sample applications.

Zilog TCP/IP Software Suite Programmer's Guide (RM0041)—Explains in detail different ZTP configurations, features, and also gives a detailed explanation of the sample applications. It also contains a section on ZTP performance numbers and FAQ.

Zilog TCP/IP Stack API Reference Manual (RM0040)—A complete API Reference Manual.

Ordering Information

Software Package	Part Number	Description	Support
Standard Release			Free support via www.zilog.com; Free updates
Source Release	eZ800000100KTS	Full ZTP source package integrated with ZDS II	Free support via www.zilog.com; One year of free updates

Note: ZTP is integrated with ZDS II—eZ80Acclaim! 5.0.0 and available in a single installer.

Warning: DO NOT USE IN LIFE SUPPORT

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer

 Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.