CRITICAL I/O WHITE PAPER

1Gb Ethernet XGE Performance Using the Critical I/O XGE Sockets Software Library

Abstract
Critical I/O's XGE 1Gb Ethernet TOE is designed specifically for data intensive real-time applications. The XGE 1Gb Ethernet hardware implements full TCP/IP offload in silicon, providing the highest perfromance, low host CPU loading, and highly deterministic operation.

This whitepaper discusses the performance aspects of Critical I/O XGE hardware and XGE Sockets Software Library, and compares this performance with traditional Gigabit Ethernet implementations.

Revision 3/17/2011

CRITICAL io

TCP/IP Performance Using the Critical I/O Sockets Software Library

Gigabit Ethernet is not generally viewed as suitable for transporting time-critical data in a real-time system. This is because traditional Ethernet has a well deserved reputation for long message latencies, low performance, unpredictability, and consuming lots of CPU cycles in the complex software stacks.

So what is the problem with traditional Ethernet? The problem is NOT the network switching hardware or the basic transport protocols. The transport protocols are actually well suited to data intensive real-time applications, and even a $100 gigabit Ethernet switch will yield excellent network performance and determinism. The problem with traditional Ethernet implementations is the reliance on a TCP/IP software stack and its complex interaction with traditional Ethernet NIC hardware.

Critical I/O XGE is Designed for Real Time

Critical I/O's XGE Gigabit Ethernet hardware and software is designed specifically for real-time applications. The XGE hardware implements full TCP/IP offload in silicon. Full offload provides ultra high performance, very low CPU loading, and perhaps most important for real time systems, highly deterministic operation. Yet Critical I/O XGE is fully interoperable with traditional Ethernet networks, devices, and applications.

The XGE hardware is supported by XGE software components that include the XGE Sockets Library, and XGE Sockets Drivers. Both of these components offer complete compatibility with standard Ethernet sockets communications. The Sockets Drivers tie into the host operating systems socket layer, maintaining full compatibility with existing socket based applications. The XGE Sockets Library provides a socket-like API that allows applications to also bypass the host socket layer entirely for the highest possible performance, as shown in figure 1.

The XGE Sockets Software Library is the focus of this white paper, but the majority of the comments apply to the XGE Socket Drivers as well.

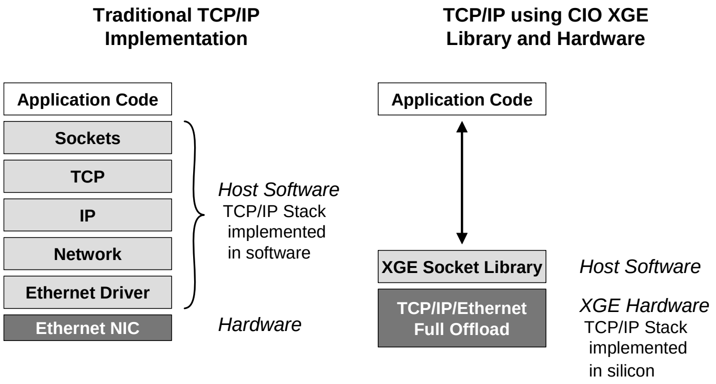
Traditional TCP/IP Implementation
TCP/IP using CIO XGE Library and Hardware
Application Code
Sockets
TCP
IP
Network
Ethernet Driver
Ethernet NIC
Host Software
TCP/IP Stack implemented in software
Hardware
Application Code
XGE Socket Library
TCP/IP/Ethernet Full Offload
Host Software
XGE Hardware
TCP/IP Stack implemented in silicon

Figure 1. Traditional TCP/IP Implementations vs. Critical I/O XGE

An Overview of Socket Communications

Sockets are the principle data transport abstraction used with Ethernet. A socket is a communication end-point that established between two Ethernet devices, and is associated with a specific pair of TCP or UDP ports. Socket connections are established between two nodes over a network, with the IP address/port pair for each node defining the socket connection. Sockets may be created as stream sockets, using the TCP protocol, or as datagram sockets, using the UDP protocol. Any Ethernet node that supports standard socket communication is fully compatible with all other socket based remote Ethernet nodes and network applications.

The Critical I/O Sockets Software Library

The XGE Sockets Software Library provides an easy to use API that can be easily accessed directly from application programs, without the need for an intervening driver. The Sockets Library API allows the user application to directly call CIO Sockets Library functions set up connections (for TCP) with remote nodes then call additional Sockets Library routines to send and/or receive data to/from the remote nodes using TCP, IP, or Ethernet protocols. The Sockets Library supports a novel feature called Named Buffer TCP Receives, which allows the TOE to place incoming TCP data directly into unique application defined buffers on a per-connection basis.

Direct use of the Sockets Library provides the highest performance sockets compatible interface, sending and/or receiving data at up to 250 MByte/s using just a few percent of the host CPUs throughput, with *application to application* latencies of as low as 40 usec. The library is fully compatible with any remote nodes that use a standard sockets interface. Both polled and interrupt driven modes of operation are supported.

A Performance Comparison: Critical I/O XGE TCP/IP vs. Traditional TCP/IP

The performance of traditional TCP/IP implementations is dominated by the characteristics of the traditional software TCP/IP stacks and Ethernet hardware. The stack is a large, complex software implementation that has a tremendous amount of low level interaction with the Ethernet NIC. This yields poor performance, and the host CPU (the stack) must touch the data multiple times, to copy form user buffers, to checksum, to segment, etc. The host CPU must also form headers at several levels, and deal with a multitude of interrupts from the Ethernet NIC. On receive the issues are much the same, but in reverse.

All of this host activity described above also results in large message latencies, and what is often an even more significant problem, large variations in latency. Partial offload NICs (for example, checksum offload) are sometimes viewed as a solution, but this only solves a very small part of the problem. Checksum offload, for example does reduce the CPU overhead slightly, but only addresses one part of the traditional TCP software stack bottleneck. Note also that most RTOS TCP/IP stacks do not even support partial offload NICs.

Because Critical I/O XGE implements full TCP/IP offload in hardware for any size data transfer, the host CPU is only involved once to set up a send of any size or to process a receive of any size. This provides a tremendous CPU overhead reduction, especially for larger transfer sizes, as illustrated in the overhead comparison chart below. This chart shows the number of CPU cycles consumed (i.e. overhead) to send each and every byte of data, for a variety of transfer sizes. Data is shown both for transfers using a traditional (VxWorks) socket implementation, and using the XGE Socket Library and XGE hardware.

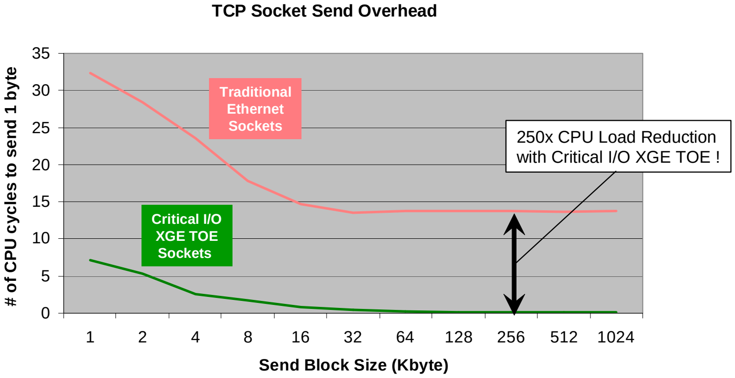

Figure 2. A Comparison of TCP Overhead: Traditional Ethernet vs. XGE

XGE TCP Send Performance

XGE offloads the full TCP/IP stack in hardware. Full offload means that a user can, with a single library call, initiate a TCP send of any size. The XGE library queues the send with the XGE hardware, and from then on, the send is performed entirely by the XGE offload engine. A single host interrupt occurs when the entire block of data (which may be many Mbytes, if desired) has been sent, and acknowledged by the receiver. The XGE library, in turn, will call the user's specified notification function only once, for the entire block send.

The XGE TCP send process is illustrated in the figure 3. The full offload send capability provides excellent TCP/IP send performance, with very low CPU utilizations, as shown in figure 4

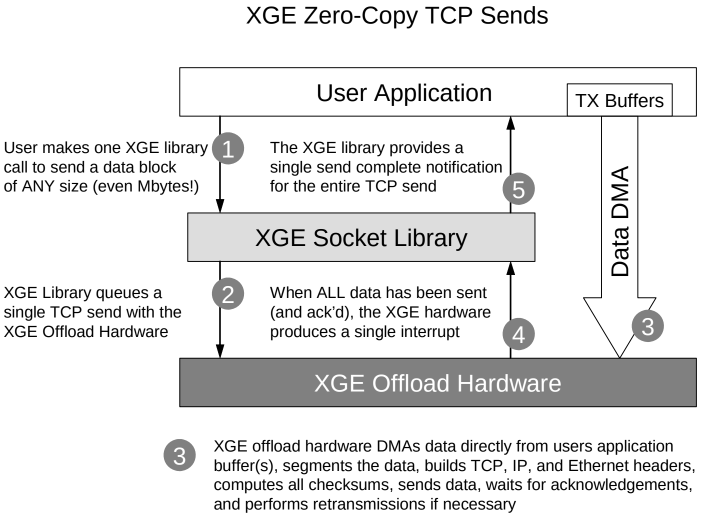
XGE Zero-Copy TCP Sends
User Application
TX Buffers
User makes one XGE library call to send a data block of ANY size (even Mbytes!)
1
The XGE library provides a single send complete notification for the entire TCP send
5
Data DMA
XGE Socket Library
XGE Library queues a single TCP send with the XGE Offload Hardware
2
When ALL data has been sent (and ack'd), the XGE hardware produces a single interrupt
4
3
XGE Offload Hardware
3
XGE offload hardware DMAs data directly from users application buffer(s), segments the data, builds TCP, IP, and Ethernet headers, computes all checksums, sends data, waits for acknowledgements, and performs retransmissions if necessary

Figure 3. XGE TCP Send Flow

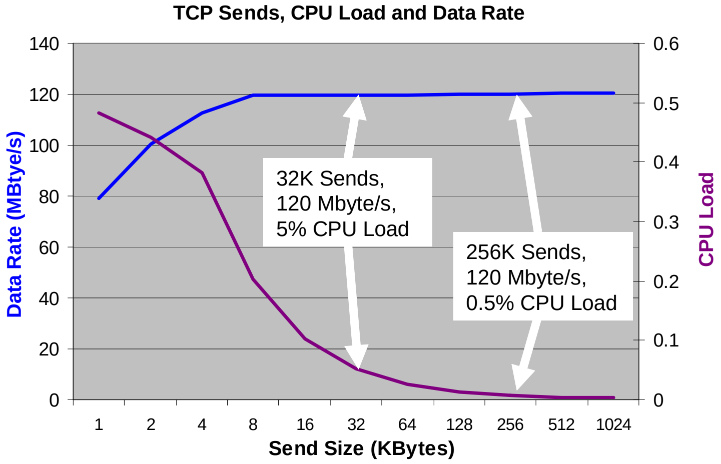

Figure 4. XGE TCP Send Performance (750 MHz PPC host)

XGE TCP Receive Performance

Full offload also applies to TCP receive operations. This means that a user can register application level *named buffers* (of any size) with the XGE library to be filled with received TCP/IP data. The user application simply allocates a receive buffer, and then provides a pointer to the buffer to the XGE library. When data is receive on the specified connection, the XGE hardware will DMA the data directly into the user buffer, without any host CPU interaction required. Only when the user buffer has been completely filled with data will the XGE hardware interrupt the host CPU, and the XGE Library will notify the user application that its buffer has been filled. If data received on an open connection and the user has not yet provided any buffers to be filled, the data will be buffered in the XGE on-board SDRAM until a user buffer is provided. Note that in addition to the named buffer mechanism described above, the XGE Library also provides an anonymous buffer receive mechanism, were a common pool of receive buffers are utilized for received data on designated connections. The XGE receive mechanism is illustrated in figure 5, and typical receive performance is shown in figure 6.

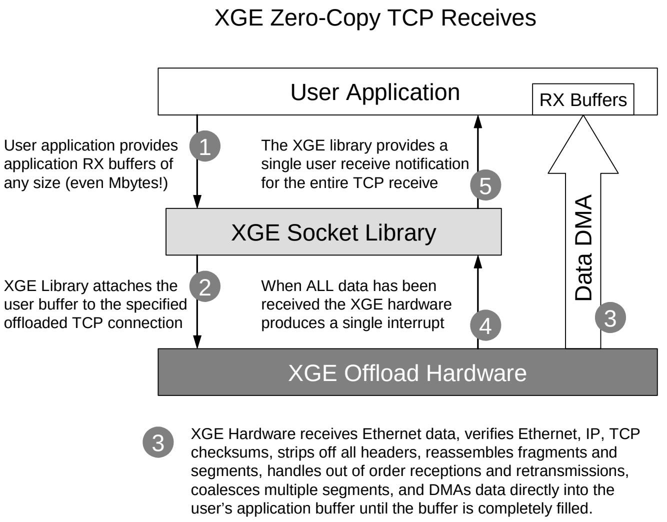
XGE Zero-Copy TCP Receives
User Application
RX Buffers
User application provides application RX buffers of any size (even Mbytes!)
1
The XGE library provides a single user receive notification for the entire TCP receive
5
XGE Socket Library
Data DMA
XGE Library attaches the user buffer to the specified offloaded TCP connection
2
When ALL data has been received the XGE hardware produces a single interrupt
4
3
XGE Offload Hardware
3
XGE Hardware receives Ethernet data, verifies Ethernet, IP, TCP checksums, strips off all headers, reassembles fragments and segments, handles out of order receptions and retransmissions, coalesces multiple segments, and DMAs data directly into the user's application buffer until the buffer is completely filled.

Figure 6. XGE TCP Send Performance (750 MHz PPC host)

XGE TCP Latency and Determinism

While data rates and CPU loads are of critical importance to data intensive systems, latency and determinism are equally important in many systems. Because Critical I/O XGE implements TCP/IP in hardware, message latencies are quite low, and very deterministic. The chart in figure 7 shows application to application latencies measured over 240 trials, using both traditional TCP/Gigabit Ethernet, and XGE hardware and the XGE Socket Library. The data measures application to application message latency; that is, the time from a user application on node A initiating a message send operation to the time that a user application on node B actually receives the message. The XGE data (pink) shows a very low latency and a very low variation in latency, while the traditional TCP/Ethernet implementation (blue) shows a much higher latency and a dramatic increase in the latency variation.

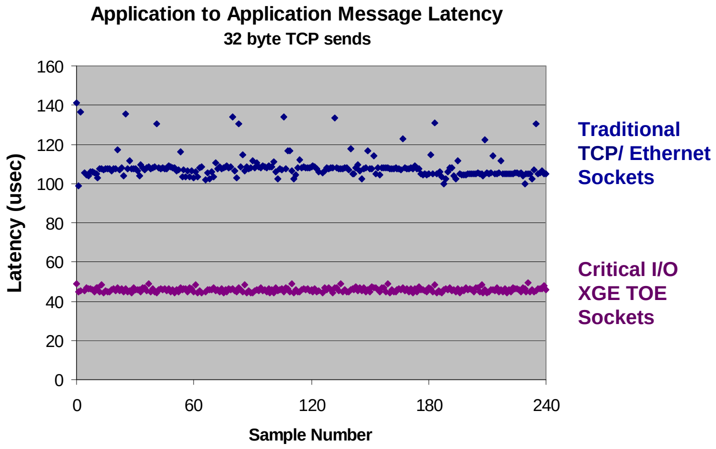

Figure 7. XGE TCP Message Latency and Determinism Comparison

Conclusion

There are many advantages in using Gigabit Ethernet in data intensive real-time systems. The use of Gigabit Ethernet allows interoperability with a wide variety of standard, low cost Ethernet hardware and protocols, while also providing the *potential* of high performance operation where needed.

But traditional Ethernet implementations are not up to the task of providing high performance operation. They suffer from low data rates, high latencies, poor determinism, and high host CPU loads, largely due to the complex software TCP/IP stack and its time-consuming interaction with traditional Ethernet NIC hardware.

Critical I/O's XGE hardware and the XGE Socket Library are designed specifically for real-time systems, and thus eliminate all of the disadvantages associated with traditional Ethernet. XGE hardware and software provide very high performance, very low CPU loading, very low message latencies, and highly deterministic operation. In addition, Critical I/O XGE maintains full interoperability with traditional Ethernet devices, networks, and applications.

36 Executive Park, Suite 150
Irvine, CA 92614
Telephone: (949) 553-2200 Fax: (949) 553-1140
www.criticalio.com