GE
Intelligent Platforms

Lightweight Internet Protocol Stack

Ideal Choice for High-Performance HFT and Telecom Packet Processing Applications

Introduction

The number of devices connected to IP networks will be nearly three times as high as the global population in 2016. There will be nearly three networked devices per capita in 2016, up from over one networked device per capita in 2011. Driven in part by the increase in devices and the capabilities of those devices, IP traffic per capita will reach 15 gigabytes per capita in 2016, up from 4 gigabytes per capita in 2011 (Cisco VNI). **Figure 1** shows the anticipated growth in IP traffic and networked devices.

The IP traffic is increasing globally at the breath-taking pace. The rate at which these IP data packets needs to be processed to ensure not only their routing, security and delivery in the core of the network but also the identification and extraction of payload content for various end-user applications such as high frequency trading (HFT), has also increased. In order to support the demand of high-performance IP packet processing, users and developers are increasingly moving to a novel approach of combining PCI Express packet processing accelerator cards in a standalone network server to create an accelerated network server. The benefits of using such a hardware solution are explained in the GE Intelligent Platforms white paper, *Packet Processing in Telecommunications – A Case for the Accelerated Network Server*.

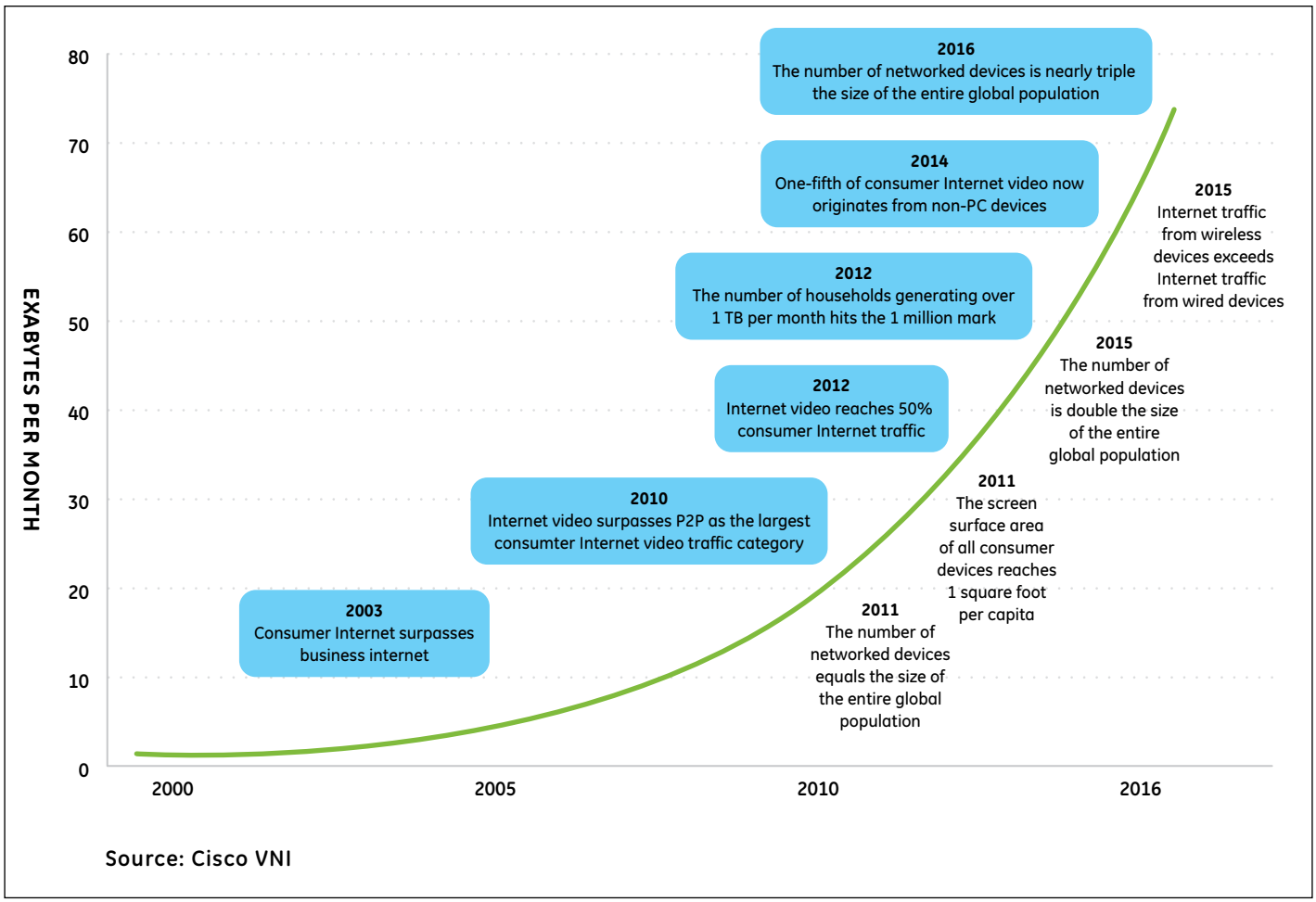

Figure 1 Projected growth of IP traffic and networked devices by 2016

Developers and users are increasingly seeing the advantage of reduced time of development, with subsequent lower cost and shorter time-to-market, by combining PCI Express packet processing accelerator in a standalone network server to create comprehensive commercial off-the-shelf (COTS) accelerated network server platform. However, they usually spend valuable time integrating a high-performance TCP/IP stack on the multiple packet processing cores to create a multi-port compute engine required for their applications.

A PCI Express packet processor usually consists of one or more multicore processors, memory, host (PCI Express) and network I/O ports, and embedded software running on the cores to process the packets. If these cores are loaded with Linux to create the developers application, then the TCP/IP stack is included with the Linux operating system (OS) and is available for use by kernel or user mode applications. However, if the developer wants to use "bare metal" OSs on their processor cores, to minimize processing overhead and enable line-rate performance, the use of a conventional TCP/IP stack can be problematic. Conventional TCP/IP stacks are usually designed for larger systems, and carry overhead and an extensive feature set that can affect performance and latency. There are alternatives to the conventional TCP/IP stack that are designed to operate in an embedded environment and can be ported to a multicore processor for bare metal OS use. One such embedded TCP/IP stack is the lightweight IP (lwIP) stack.

This white paper highlights the benefits of using the lwIP stack on the multicore processor for bare metal OS use. It will also highlight the benefits of having a pre-integrated lwIP stack on one or more PCI Express packet processing cards in a standalone network server to create a high performance, cost-effective and comprehensive application-ready COTS accelerated network server platform.

The lwIP Embedded TCP/IP stack

The lwIP stack is a small and independent implementation of the TCP/IP protocol suite that was originally developed by Adam Dunkels at the Computer and Networks Architectures (CNA) lab at the Swedish Institute of Computer Science (SICS), and is now developed and maintained by a network of developers worldwide. The focus of the lwIP TCP/IP implementation is to reduce processor resource usage while still providing a full-scale protocol stack. This makes lwIP suitable for use in embedded systems with low memory requirements. lwIP is open source, licensed under the Berkley Software Distribution (BSD) license model and supports a complete set of network protocols including IP, UDP, ICMP and TCP.

lwIP provides the network stack function for processor-based product development that has, as one of its goals to achieve Ethernet network connectivity. Since packets of traffic on a network are transported via protocols like IP, UDP and TCP, a network stack like lwIP is deployed to handle these protocols and provide encapsulated packet data to and from the network and processor. lwIP has the advantage of meeting the embedded usage requirements described above, and can be implemented in the bare metal OS environments of the multi-core packet processors to provide network support.

As the developer and maintainers of lwIP considered this embedded network stack implementation, a variety of feature goals, specific to embedded applications, guided their work, including:

- Modularity – Protocols and other elements of lwIP are implemented in their own software modules, with concessions in separation of modules to achieve maximum performance for the stack in embedded applications.
- Extensibility – lwIP runs with or without an operating system, can use threading or not, supports Little or Big Endian architectures, and can be successfully ported to many different processors from 8-bit all the way up to multicore 64-bit.
- Common Data Structures – Packet source and destination addresses, as well as other packet parameters discovered and stored during execution, are shared between functions like the IP and TCP modules to reduce complexity and size.
- Common Buffer usage – Because the user application and lwIP stack can share a common memory segment, buffers created and used during the input/output phases of the networking operation are reused by lwIP's protocol and functional modules, increasing performance by avoiding memory copy cycles and decreasing memory usage in the target embedded systems.
- Small overall memory footprint – With the embedded application goal, the amount of processor memory used by the lwIP stack needed to be small enough to fit into available memory space while not sacrificing important stack features that applications would need.
- Latency - The modularity, common buffer usage and small memory footprint also supported the goal of keeping packet latency through the stack low.
- Applications Programming Interface – A choice of three application interfaces (raw, sequential and socket) are available in lwIP. With this flexibility, developers can decide how best to deploy the lwIP stack in their application based on compatibility requirements, available OS resources or interface expectations. For example, the GE Intelligent Platforms lwIP port uses the raw API interface to support operation with the Cavium Simple Executive on multiple cores of an OCTEON® II processor.

Achieving these goals, the lwIP TCP/IP stack becomes an excellent choice for embedded multicore applications that need protocol-based network traffic support.

lwIP Deployment Examples

Being open-source, lwIP does have to be ported to the specific processor and OS of choice for use. The architecture of lwIP is such that it is operating system independent, with application-side and line-side interfaces to facilitate porting. Many silicon vendors have taken the opportunity to expand their software offering by porting and supplying lwIP as part of their software development packages. Users can then add networking to their application development effort, and subsequently support multi-media or industrial automation/control applications that require network connectivity. Similarly, many high-end FPGA suppliers include ports of the lwIP stack in their development tools to support embedded processor usage and allow development flex-ibility as well as the extension of end-use applications to support embedded networking capability and Internet accessibility.

Another novel approach to embedded TCP/IP, utilized by GE Intelligent Platforms, is to offer a port of the lwIP stack for our Cavium OCTEON II-based WANic PCI Express packet processor product. This embedded stack facilitates the creation of a packet processing node supporting a network server, to create an accelerated network server. The packet processor, oper-ating in one or more PCI Express slots of the server, provides the 1Gbit or 10Gbit Ethernet network I/O interface(s) for the server, performs packet functions like classification, encrypt/decrypt and traffic management, then executes the payload processing applications. Typically a developer would use the packet processor, with its multiple processing cores, to perform control and data processing work in these cores such that one or two cores execute the control plane tasks while the rest of the cores performed the data plane processing. The control plane cores use a Linux OS (with associated TCP/IP stack built-in) and operate at a relatively low level of performance associated with the control functions implemented. The data plane cores, however, are required to handle network traffic, for the array of functions required, at wire speed. Data plane cores would there-fore use a low- overhead bare metal OS and seek to minimize the number of instructions per packet to maintain performance and

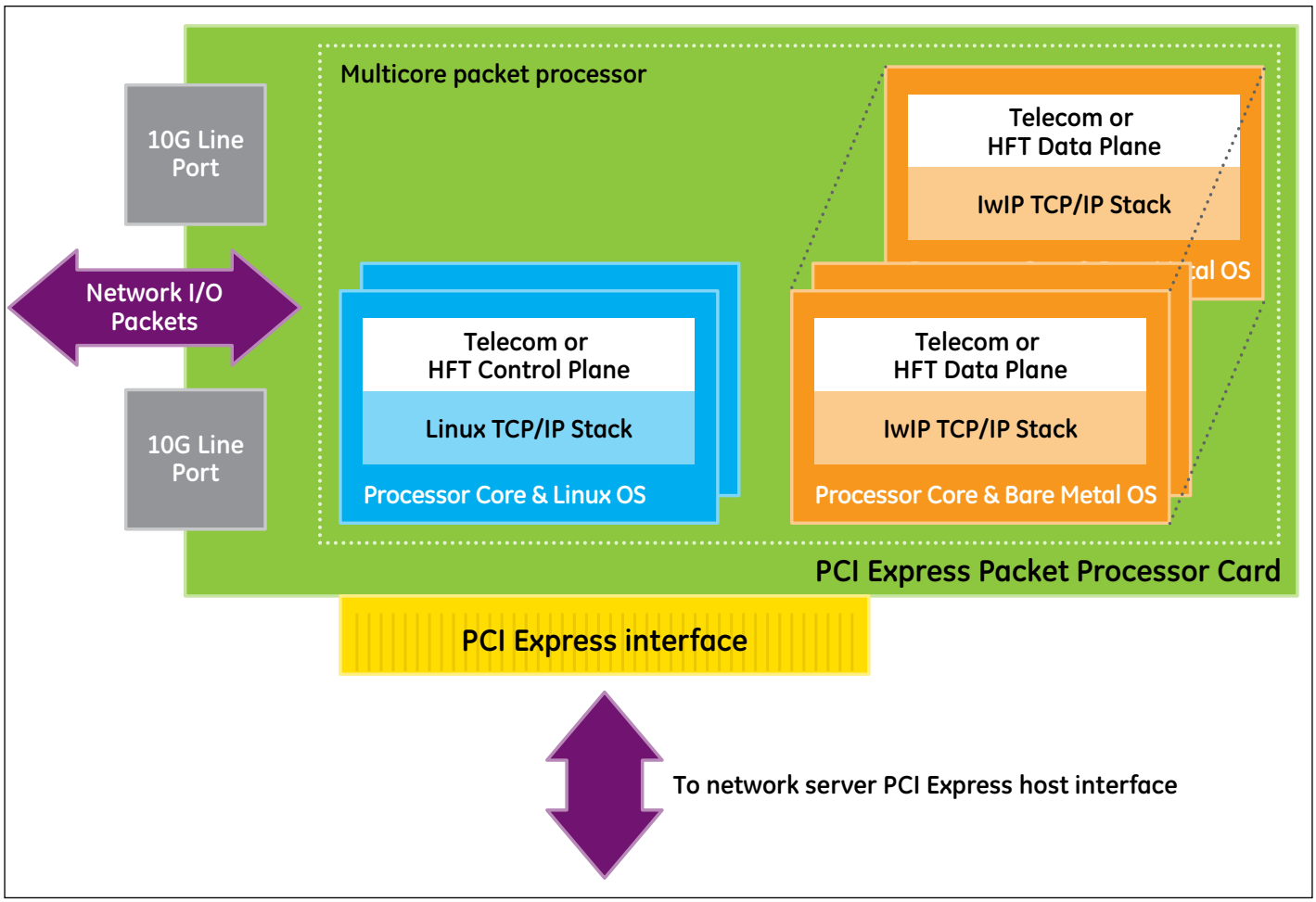

Figure 2 Example packet processing lwIP embedded TCP/IP Stack Implementation

keep latency low. Unfortunately, to receive and transmit network I/O traffic, these data plane cores would still need to deal with the protocols that are used to transport this traffic, and that is where lwIP comes in. A more conventional network stack would consume as much or more memory and processor resources as the bare metal OS itself, but lwIP, with its lower memory footprint and comprehensive feature support, provides network protocol support to the data plane applications while maintaining the high performance and low latency required to gain wire-speed processing performance. Telecommunications and financial/HFT applications take advantage of the protocol handling capability in the lwIP stack while requiring the lowest latency contribution from the stack in their applications. Figure 2 is a block diagram of the WANic 66512 with its processor cores allocated to the Linux control plane and lwIP-based data plane functions.

As GE Intelligent Platforms considered the need for a pre-integrated TCP/IP stack for IP packet support, we evaluated the array of stack options and elected to port the lwIP TCP/IP stack as an extension of our WANic PCI Express packet processor software development kit. We considered the value that having such an extension of the product development tool set brought to our customer base, and the following is what we identified as the benefits of using lwIP:

- **Provides a shorter time to market** – Developers who want to deploy a networking application always consider the amount of time it will take them to develop, test and release their product. The lwIP stack, included in development tool sets, solves problems in the development cycle by allowing the user to concentrate on their end application without having to worry about the details of network protocol interface; lwIP does this for them. This results in a shorter time to working hardware and software, ease of test and validation, and quicker availability to potential customers.
- **High performance in embedded applications** – The lwIP stack is small in memory size, having undergone an optimization of protocol and feature support in its development to target embedded applications. This optimization, along with a buffer memory architecture that allows modules to share resources and data, provides a significant increase performance. Such performance increase is available without sacrifice of necessary embedded TCP/IP stack features.
- **Lower packet latency** - Another advantage of lwIP's code optimizations and resulting small memory footprint is the reduced latency for packets flowing through the stack. Depending on specific protocol used, packet in-to-out latencies can be as low as 2 to 4uS, which are especially beneficial for HFT applications.
- **Reduced complexity** – The lwIP stack is open source and available on the Internet for those wishing to do their own porting. Full source code, documentation, diagnostic capability and example applications are part of the open source package, and there is an ecosystem of developers and users to engage when needed. lwIP is also available as a TCP/IP stack for embedded applications, already ported to a variety of processor products. As noted above, processor and FPGA suppliers, as well as product companies like GE Intelligent Platforms have already done the software porting of lwIP for their products that customers can use.
- **Lower total cost of ownership** – The ability to use the lwIP stack in target applications, achieve functional and performance goals and get the end product to market quickly helps control the cost of development, and the open source nature of lwIP allows developers to understand their product better, handle both engineering and customer problems with efficiency and minimize the costs of end-customer support.

Conclusion

When packet processing is added to a network server to create an accelerated network server, the packet processor needs a pre-integrated TCP/IP stack to service the IP traffic provided by target applications. This embedded stack must provide the network protocol support required while coexisting with the other embedded elements of the packet processing, maintaining performance and low latency and not consuming more memory than can be allocated to this stack function.

The lightweight IP stack achieves these goals and is available as open source for developers to use directly, or is provided already ported to the target processor so application developers can concentrate on their end-use and get to market sooner. GE Intelligent Platforms provides an lwIP TCP/IP stack for the WANic PCI Express packet processor family of products, and has demonstrated the high performance and low latency benefits in product development and use.

GE Intelligent Platforms Contact Information

Americas: **1 800 433 2682** or **1 434 978 5100**
Global regional phone numbers are listed by location on our web site at **www.ge-ip.com/contact**

www.ge-ip.com

11.12 GFT-847