



## **Avionics Appliances for Ethernet Networks**

### **Real-Time MIL-STD-1553 and ARINC 429 “Appliances” Bridge to Ethernet Networks**

**Richard Wade**  
Chief Technical Officer  
Alta Data Technologies  
May 2011

#### **Introduction**

Modern avionics systems find switched Ethernet the dominant network architecture with MIL-STD-1553 (1553) and ARINC implemented for high reliability and legacy communications. For commercial off the shelf (COTS) systems, the most common method to integrate 1553 or 429 networks into the system is to add an interface card to the computer’s backplane (PCI or PCI Express). With Ethernet dominating the modern landscape, these legacy interfaces are still popular and will not be going away anytime soon (see new Airbus A350, China 919, Next Gen 747/737, F-35, etc...), but the method by which they are incorporated in the total system is evolving.

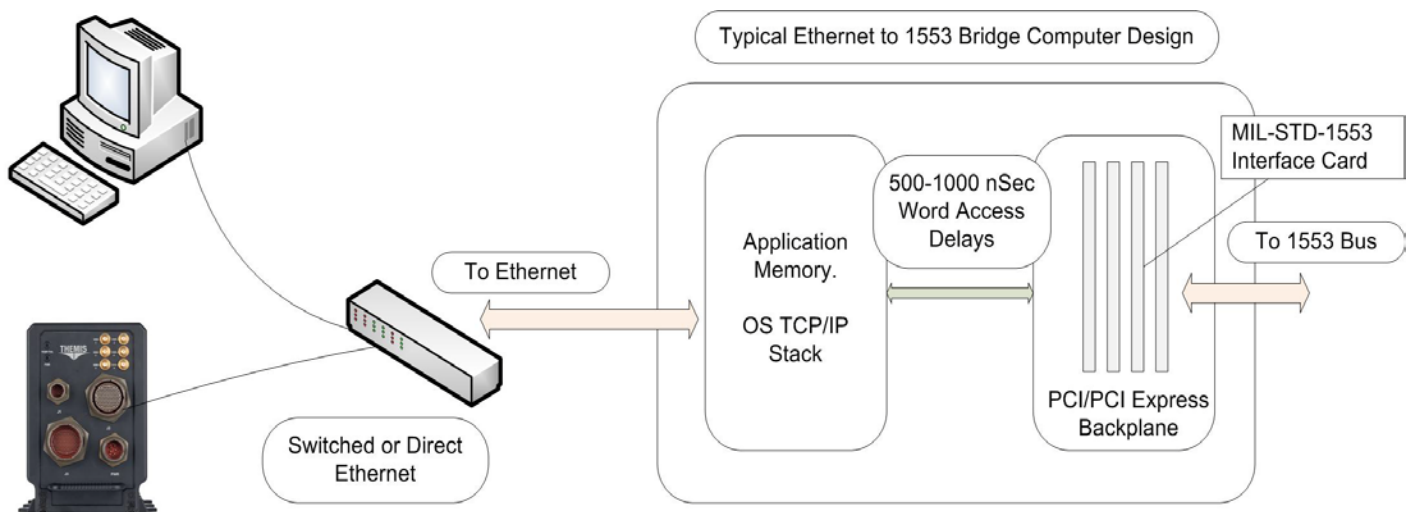
For the purpose of this paper, we’ll only further mention 1553 as a legacy network topology, but the same talking points apply to ARINC 429 systems. PCI will refer to both PCI, PCI Express and other common parallel computer backplanes for 3<sup>rd</sup> party vendor I/O and network interfaces.

With COTS test and embedded computer designs, Ethernet ports are a common part of the motherboards (or Single Board Computers – SBCs), but lower volume 1553 avionics ports are typically PCI/PMC/PCI Express commercial off the shelf (COTS) interface cards. There are several issues with new computers and 3<sup>rd</sup> party interface cards: Many computer systems are limiting expansion card availability; embedded systems frequently want to upgrade processing capability but legacy I/O requirements and changing backplane standards can limit upgrade options; and wiring bundles of legacy I/O can be difficult to manage and add weight. A more desirable system design approach may be to remote the I/O closer to the source via an Ethernet connection or use Ethernet as the backplane within the system. This method offers the advantages of better software portability, simpler hardware configurations, power savings and less wiring. Before exploring this remote networking concept further, here are some basic networking and I/O COTS design facets.

## Traditional Avionics Test and Embedded Architectures

A typical 1553 or ARINC COTS interface card has three main circuits: a Protocol Engine (PE) that is usually an FPGA or micro processor to perform network protocol off-load processing, a PCI backplane connection to the computer and some pseudo dual-port RAM to the PE and backplane to provide real-time access for both the PE and the user's application. The user's application reads and writes to these data buffers via backplane memory mapping through a custom device driver for the target Operating System (OS). Typical read/write speeds are 500-2000 nSec for a single word to or from application memory and the interface card. So to update a single word in a 1553 packet from a control application may take only 1 μSec via a backplane interface.

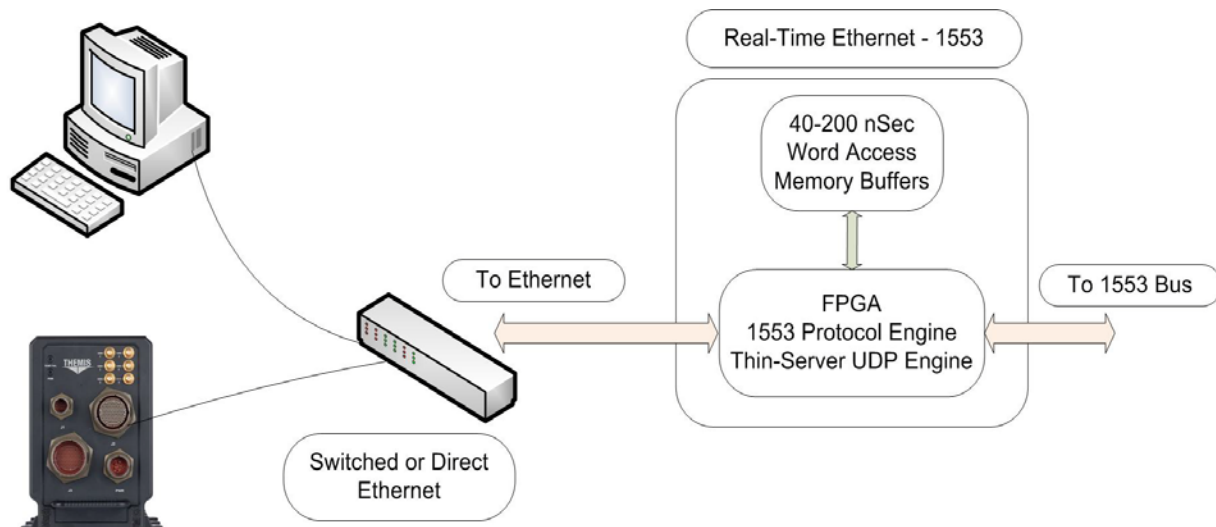
In the last decade, the most common method for remoting 1553 had been to wedge a COTS card into a small computer system (such as PC104, mini computers or proprietary boxes) and then run another Ethernet application to translate between the two domains. These box-box designs are typically running commercial OSES (Windows, Linux, RTOS) where the TCP/IP/UDP network stack will require 50-500 μSec to process each packet (worse with USB 2.0 where devices are limited to 1 mSec polling rates). Ethernet ↔ 1553 packets must be processed through OS IP/TCP stacks and one or more PCI backplane transactions. This overhead can add significant system delays and application jitter. See Figure 1 for an example architecture of traditional Ethernet to avionics system designs (computer box style).



**Figure 1: Typical Ethernet <-> 1553 Bridging Designs**

## Network Centric Avionics Architectures

A solution that eliminates one layer of process delays is to embed a real-time, thin-server directly in the 1553 FPGA protocol engine to replace the traditional PCI host interface. At Alta Data Technologies (Alta), we recently released this design in a product called eNet-1553™. The eNet-1553 has a thin-server IP/UDP protocol engine as the “backplane” interface and can provide memory packet accesses at nearly the same rate as PCI backplanes. This thin-server approach bypasses at least one layer of IP stack and PCI backplane translations saving half of the total round trip transmission time. Figure 2 shows a thin-server, real-time IP/UDP design.



**Figure 2: Next Generation Real-Time Ethernet <-> 1553 Design**

Besides real-time data access, another key advantage of an Ethernet appliance is software portability. Because IP/UDP communications (sockets) are built into almost every OS (even new tablet devices), this means a client application can be easily be ported to almost any computer system and is not locked-in on a custom device driver (even DO178 operating systems usually have a qualified Berkley Socket capability). A 1553 appliance device can provide generic maintenance, monitoring/data logging and programming (memory loader verifier – MLV) ports for deployed systems, and in many cases, eliminate the need for a 1553 card in a rugged notebook computer – the customer could use any computer device that supports Ethernet.

Alta’s eNet-1553 is a rugged, small appliance (similar to a 4-stub 1553 bus coupler) with standard 5-30 VDC, 10/100/1000 Ethernet and optional Power Over Ethernet (POE). Alta has added advanced features such as Signal Capture (to capture the actual raw 1553 signal for integrity analysis), auto BC/RT image loading to streamline setup and provides an auto Bus Monitor mode to provide real-time bridging between 1553 and Ethernet networks. Figure 3 shows a picture of Alta’s eNet-1553 product.



**Figure 3: Alta's New eNet-1553 Network Appliance**

### Design Trade-offs

There are always trade-offs with system architectures and in this case we are trading real-time PCI backplane memory accesses (with an interface card in a computer) versus a packet based network topology. 1553 is only one mega bit/second and most typical Ethernet systems are 1000 mega bit/second. This would appear to be a slam dunk to bridge the networks, but 1553 BC and RT communications cannot be directly translated to Ethernet or any network because of the required 4-12  $\mu$ Sec command-response protocol between heterogeneous computers (unlike most TCP/UDP applications that are homogenous peer-to-peer). So there will often be at least one packet (or frame) delay in bridging 1553 and Ethernet applications.

Further, many computer OS IP stacks and LAN switches cannot process above 5,000-50,000 packets per second, which means packet loss in the ether. This delay through the OS stack or switch routing is known as "path delay". Although rare, 1553 can transmit 2000-20,000 small packets per second and the user's application may need to perform both a read/write function for each data packet. So the application packet frequency could easily be 2x the 1553 packet rate and could easily overload a typical computer TCP/IP stack processing capability. The eNet-1553 device can turn around Ethernet memory read/write requests in real-time but the designer must know their systems OS TCP/IP stack and/or LAN switch processing capabilities (path delays) to match-up with application requirements.

In our testing at Alta on dozens of different Microsoft Windows™ and Linux systems, we have derived a sample formula that helps guide customers on packet processing capability of their system/LAN. The following formula can be used to estimate the number of 1553 messages per second (MsgsPerSecond) that you can expect to process with an eNet-1553 device connected to your computer:

$$\text{MsgsPerSecond} = 1 / (10 * \text{PathDelayInSeconds})$$

For example, if we have a Path Delay of 100 microseconds then we can expect to keep up with a message rate of  $\leq 1000$  messages per second. Note that this is a widely general estimate and you can monitor data at higher message rates but you may start to drop messages if you cannot read the messages faster than they are received on the 1553 bus. (Hint: our testing has shown dramatic reduction in path delays with 1000 based Ethernet and 64-bit Windows and Linux OS verse their 32-bit versions. Alta recommends 64-bit OS with 1000 based Ethernet whenever possible).

## Summary

As modern avionics system designs advance, it will be important to adapt 1553 networks to an Ethernet-centric topology. Avionics appliances like eNet-1553 help fulfill the requirement. A thin-server design, where the I/O buffers are directly available via IP/UDP requests removes one layer of packet processing and can greatly enhance system design options, maximize portability of the application and increase data application performance. These next generation appliances can provide flexibility in system design topologies and reduce both initial and recurring development costs.

## About the Author

Richard Wade is a founder and CTO at Alta Data Technologies. Richard is former Sr. Application Engineer at Condor Engineering, General Manager at SBS Technologies and a Major (retired) in the US Army. Richard holds a BSEE from New Mexico State University. You may contact him at [rich.wade@altadt.com](mailto:rich.wade@altadt.com) or visit [www.altadt.com](http://www.altadt.com).