FIREWIRE: THE NEW 1553?

Michaela Blott (blott@acracontrol.com)
ACRA CONTROL, Dublin 14, Ireland

KEYWORDS: IEEE-1394, MIL-STD-1553, COTS, DAU, FibreChannel, Ethernet

ABSTRACT

MIL-STD-1553 has served the flight community well. However, in recent years several new high-speed bus standards have emerged that outperform 1553 in various respects such as data throughput and increased address space. During this time, mission requirements - including video and audio - have become more data intensive.

Although some of these busses were not initially designed for the avionics industry (such as Ethernet, FireWire, and FibreChannel), they are potentially of interest as high-speed commercial off-the-shelf (COTS) solutions for both set-up and data acquisition.

These busses offer not only improved overall system performance, in terms of aggregate sampling rates, but also simplify existing data acquisition system architectures. They require fewer high-bandwidth links which can serve for both set-up and data. This paper examines some of these issues, focusing in particular on IEEE1394, better known as FireWire.

1. INTRODUCTION

Over the past 20 years, MIL-STD-1553 has become the most widely deployed data bus in the avionics industry. With its reliability and deterministic behaviour, it is well suited to carry mission critical information between sensors, weapons, computing modules, data acquisition systems, recorders and transmitters. However, with more sensor data going digital, sampling rates increasing and demand for digital video transfers growing, new mission requirements have become more data intensive and exceed its maximum throughput by far. Meanwhile, various new high-speed networking and bus standards have evolved in the communications and desktop industry, among them Fast and Gigabit Ethernet, FibreChannel, FDDI, ATM, and FireWire. Due to their wide adoption, these busses offer low cost solutions with increased lifetime and interoperability with third party modules. Furthermore, they outperform MIL-STD-1553 in terms of data throughput, address space, and scalability.

Although there are many potential candidates, and the success of one over another depends on its technical edge, industry support and popularity, this paper will focus in particular on the IEEE1394 standard with its amendments and examine its suitability as a replacement for 1553 as a backbone avionics bus from a technical point of view. Section 2 gives an introduction to 1394b including driving applications and main characteristics. The third section evaluates FireWire in respect to a number of properties that we consider relevant to the Flight Test Community. The last part of this section includes a high level comparison between 1553, Common Airborne

Instrumentation System (CAIS), FireWire, FibreChannel and Ethernet which is then followed by conclusions.

2. THE IEEE1394 STANDARD

The IEEE1394 standard, better known as FireWire, was originally introduced in the mid 80s by Apple Computer and was accepted as a standard in 1995. Its purpose was to provide the means for high speed serial communication in a desktop environment, offering data rates from 100Mbit/sec up to 3.2Gbit/sec. Driven by applications such as video conferencing, high speed printers and mass storage, the primary objectives were the following:

- High data throughput
- Support for isochronous applications
- Ease of use - plug and play
- Increased address space
- Low cost

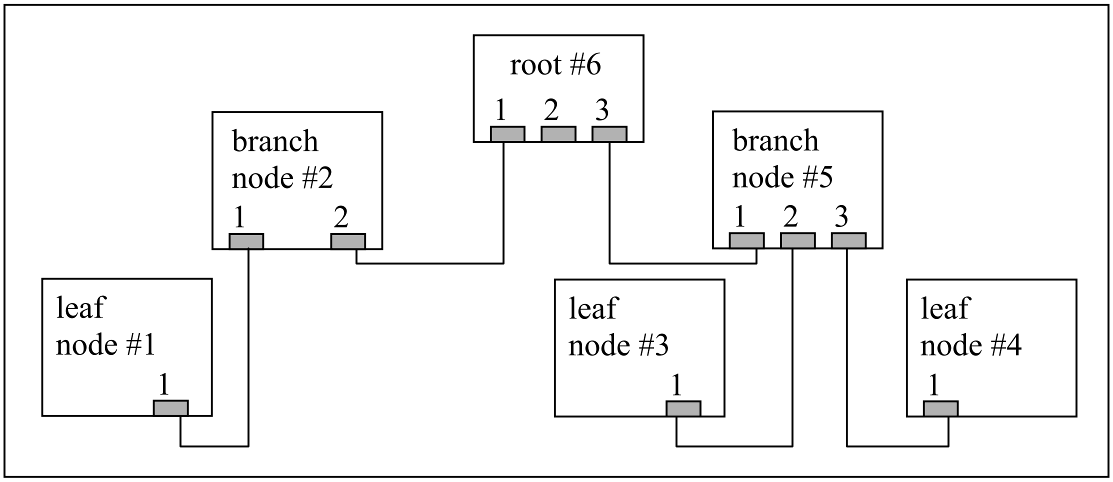

Figure 1: Example of a FireWire Topology

To achieve high data throughput, FireWire uses a bus topology built on point-to-point connections between individual nodes. Nodes that provide more than one port are referred to as branch nodes and repeat incoming traffic to all other ports. Nodes with one port only, so called leaves, discontinue the bus. After an initial bus configuration, one designated node is assigned as root, which has the highest priority on the bus and is responsible for arbitration and the dynamic self-identification process. FireWire supports up to 1024 busses connected through bridges which only forward traffic that is addressed to a node on a remote bus. Figure 1 depicts a simple FireWire network, consisting of one root node, 2 branch nodes, and three leaves. Since branch nodes repeat incoming signals to all other ports, any data transfers between node #1 and #2 are also visible to leaf node #3. For this reason, FireWire behaves as a bus although it is based on point-to-point connections.

Support for multiple synchronous videos, as required for video conferencing, makes quality of service imperative for the bus. FireWire supports these via its isochronous transactions. FireWire guarantees pre-allocated bandwidth, fixed latency and small predictable jitter for isochronous traffic, which makes this mode of data transport not only ideal for synchronous video applications but also for data acquisition as will be discussed later on.

With the objective "plug and play" in mind, FireWire supports automatic configuration, which means that the bus enumerates itself without the intervention from a CPU and eliminates the need for address switches. Somewhat like PCI, configuration is performed dynamically as the bus recognizes the removal or attachment of a node on a physical layer. Furthermore, the nodes are hot-pluggable which implies that they can be removed or inserted while the bus stays powered on. Properties such as hot-pluggable, and automatic configuration make FireWire user-friendly but might pose additional challenges in other circumstances such as flight test instrumentation (see section 3.6)

FireWire supports a 64bit address space to accommodate address intensive applications such as mass storage. 16bits are used to identify bus and node whereas the remaining 48bits represent a byte address within a node. With that, the overall addressable memory amounts to 16petabytes per bus, 256terabytes per node and 16exabytes overall.

Communication is based on a shared memory model and data can be transferred not only in an isochronous manner but also asynchronously. Asynchronous transactions, such as read, write and lock, are reliable in that they are acknowledged whereas isochronous transactions remain unconfirmed.

The initially standardized FireWire is a serial bus consisting of two differential signal pairs, and an optional pair for supplying power to and from peripherals [1,2]. Shielded twisted pair (STP) is used as media interface with a maximum of 4.5m cable length between individual nodes. This proved to be insufficient to many applications such as home networking and significantly restricted the scope of the bus. Since then it has evolved substantially: The 2nd amendment (IEEE1394b) introduced additional physical interfaces with a new encoding scheme (beta mode signalling) that improves signal integrity and allows for faster transmission rates on the legacy STP interface up to 3.2Gbps. Newly introduced media interfaces include two optical ones, as well as a category 5 unshielded twisted pair (UTP), that offers with its transformer coupled interfaces electrical isolation between separate units and benefit from being compatible with existing Ethernet infrastructure.

3. FIREWIRE IN THE FLIGHT TEST COMMUNITY

In the last section, we gave an introduction to FireWire and described the main characteristics that make it user-friendly, fast, and attractive to many desktop applications. But is FireWire suitable for the avionics industry? In the following discussion we will examine FireWire in respect to the following list of criteria, which are important characteristics for instrumentation networks and conclude with a brief comparison with other emerging high-speed bus standards.

- Reliability
- Redundancy
- Real time behaviour/ quality of service

- Isochronicity and synchronicity
- Data throughput and reach
- Address space
- System topology
- Scalability and room for growth

3.1 Reliability

Reliability has to be addressed on several levels:

- Electrical interfaces need to be rugged and electrical isolation between devices is desirable.
- Individual data transfers should support some level of error checking such as parity, CRC or checksum.
- Information exchange for control and set-up transactions must be confirmed.

Although the initial 1394a-1995 standard may not be satisfactory in respect to the electrical interface, the 1394b amendment specifies a transformer-coupled interface over unshielded twisted pair (UTP) as well as two optical interfaces, which provide the desired electrical isolation between nodes. Optical media implementations alleviate emission and interference problems. All beta mode implementations, including the UTP interface, benefit from scrambling and an 8/10 bit encoding scheme which enforces DC balance and achieves a high level of signal integrity. The bit error ratio is aimed to be less than 10^{-12}.

Similarly, the in 1394a proposed 6pin and 4pin sockets are suitable in a desktop environment, but lack environmental integrity which is required for flight equipment. However, the RJ45 connectors used for the UTP implementation are available in robust versions. Alternatively, custom connectors could be considered to work around vibration, emission and temperature requirements.

On a packet level, FireWire supports 32bit header and data CRCs, as well as a checksum for acknowledgement, initialisation and bus management packets. Set-up and control transactions typically require guaranteed delivery, which is supported via FireWire's concept of asynchronous transactions. In this mode of data transfer, each transaction, such as a read or a write, is individually acknowledged on a subaction and a transaction level.

In comparison to 1553, FireWire supports in beta mode over UTP the same amount of electrical reliability and outperforms 1553 in terms of error checking

3.2 Redundancy

Redundancy is compulsory to protect the system from a single point of failure. Due to the fact that FireWire is not a genuine bus, but rather a bus based on point-to-point connections, it is even more sensitive to failures. Considering the example in figure 1, failure of branch node #5 disrupts all traffic to nodes #3 and #4. For this reason it appears that deploying backup links is even more critical to FireWire than 1553 where the failure of a remote terminal wouldn't impact the overall bus operation. To alleviate this problem, the power supply through the FireWire bus could be used as backup power source for the PHY interface on all branch nodes. With that, communication to nodes beyond a powered down branch is still supported even though this particular node itself is down. Alternatively, star bus topologies, where up to 27 leaves are directly connected to the root, could be considered.

Redundancy is not built into the FireWire standard, but could be achieved by deploying backup networks and handling error detection and duplication at different system levels. Similar approaches have been taken within the industry for other bus standards such as Ethernet within the AFDX specification.

3.3 Real Time Characteristics / Quality of Service

To achieve quality of service, the bus needs to guarantee access to link bandwidth as well as provide for fixed and short latencies with small predictable jitter. The response time and jitter requirements differ between applications. Whereas for some control operations short and predictable latencies with low jitter are imperative, data transfer to mass storage devices are less critical. Quality of service can be accomplished in a number of ways. In the 1553 world, where real time behaviour is often associated with predictability and determinism, there is a single control over the bus through the bus controller, which assigns bus ownership to the nodes or remote terminals that need to transfer or receive data. Remote terminals only communicate when prompted. With that bandwidth guarantees can be given. The effective jitter depends entirely on the remote terminal's physical implementation rather than current bus utilization. [4]

FireWire utilizes a very different approach. The bus access is time sliced into equal intervals of 125 microseconds indicated by a cycle start packet. Up to 80% of these intervals (100 microseconds) can be allocated as channels for isochronous transactions. These are then maintained through a so-called isochronous resource manager. The isochronous traffic has guaranteed bandwidth and fixed latency properties. The bandwidth can be allocated in multiples of 20.345 nanoseconds which is the time required to transfer 4bytes at 1600Mbit/s. Latency depends on the number of hops between source and sink node and negotiated transmission speeds. [3]

The remaining amount of available bandwidth is utilized for asynchronous transactions for which the individual nodes compete while fairness between the participants is ensured and a built-in natural priority is present. It is important to note that asynchronous transactions that are issued at the end of a cycle time can delay the cycle start packet and introduce a skew of up to 50 microseconds. This doesn't affect overall system synchronization, since the cycle start packet contains the current time stamp including the delay, which allows each node to derive the encountered skew. But as a result of this, it would be advantageous to control the sampling from a system internal timer, which resides in each data acquisition unit, rather than sampling on basis of the cycle start packet itself. In this scenario, the broadcasted system time would then seed the internal timers.

Response times for actual asynchronous applications depend on a number of factors:

- Distance to sink (number of hops)
- Transmission speed (100Mbit/s or 1.6Gbit/s)
- Legacy signalling or full-duplex beta mode signalling

- Type of transaction (concatenated, split or unified)
- Arbitration service (legacy, fly-by, priority, immediate arbitration)
- Payload size

The concrete latency for a given application is a function of the specific system set-up and can be improved by changing the topology, using different arbitration services or varying the packet size. Alternatively, one could use two isochronous channels to implement confirmed operations as part of a higher level protocol. One channel carries the request and the other the response similar to CAIS. This preserves the quality of service properties associated with isochronous transactions while providing confirmed data transfers.

Although FireWire uses a very different implementation approach than 1553, it offers excellent real-time characteristics on the basis of its isochronous transactions.

3.4 Isochronicity and Synchronicity

In a distributed data acquisition system, it is important to sample at the same time. When sensors are located on physically separate units, it is advantageous if the interconnect provides built-in support for a common time and a common clock to avoid additional wiring. On a FireWire bus, a common system time that resides within the cycle master is broadcast at the beginning of each cycle interval and synchronizes all nodes every 125 microseconds. A 24.576MHZ clock controls the system timer. Furthermore, the PHY devices contain a PLL that adjusts the node clock during incoming traffic to data and strobe signals. The maximum encountered drift between two individual nodes on the bus is derived from the precision of the utilized crystals. However, the standard requires +/- 100ppm for the 49.152MHz crystal for 1394a and b. Therefore, the maximum encountered drift during a 125 microsecond interval (200ppm x 125microseconds) is 25 nanoseconds. All nodes are being resynchronised with the next cycle start packet. With that the maximum drift between any two nodes within a FireWire bus can’t exceed 25 nanoseconds. Consequently, isochronous and synchronous sampling throughout a distributed data acquisition system on basis of a pure FireWire infrastructure can be achieved.

In comparison to this, MIL-STD-1553 achieves common time via the synchronize mode code command. When used with data word (mode code 17), the current cycle time is included within the mode code data word. The jitter depends on the chosen format, as well as the precision of the involved crystals. However, 1553 signalling is unsuitable to clock peripherals and requires additional infrastructure to achieve synchronicity throughout a distributed system.

3.5 Data Throughput and Reach

The current IEEE1394 standard specifies signalling rates of up to 3200Mbit/s. The actual data throughput is subject to the type of arbitration services used, transaction types, (unified, concatenated or split), as well as the percentage of isochronous over asynchronous traffic which introduces significantly more overhead. Considering a sample application such as a digital video, which requires a bandwidth of 55Mbps (in case of a 320 x 240, 24bit true colour, 30frames/sec) shows that FireWire can easily accommodate a multiple of this, especially when used in conjunction with standard compression schemes such as MPEG2.

The maximum cable length between individual nodes is a function of the chosen media interface. Legacy and beta mode signalling over shielded twisted pair limit the distance between nodes to 4.5m, which could potentially enforce the introduction of many repeater nodes within a larger scale airplane. The remaining media types support sufficient cable length. In comparison, MIL-STD-1553 limits the maximum cable length to 300 feet, which equals the reach of FireWire's UTP interface.

3.6 Address Space and Dynamic Node Id Assignment

Future data acquisition systems face the challenge of increasing demand for address space. For example, a single video application with 640 x 480 pixels true colour occupies 921.6Kbytes of address space alone on the data bus. Furthermore, set-up or programming bus address requirements increase as overall systems become more and more complex. For example, a linearization table for a PT100 consists of a 16bitx16bit lookup resulting in 128Kbytes of required EEPROM space. This already exceeds the overall available address space on 1553 without support for expanded sub addresses.

FireWire offers 16exabyte addressable memory, which is divided into 1024 busses of up to 63 nodes each. Each node has 256terabytes of memory address space allocated to it, and although some if it is reserved for bus management (CSR architecture), it still can comfortably accommodate set-up and data requirements.

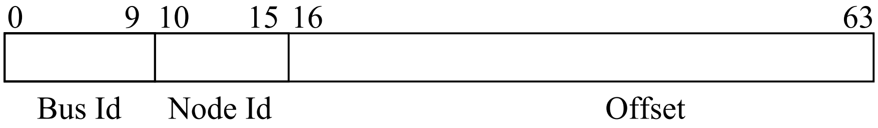

Figure 2: FireWire 64bit Address

In the 1553 world, there have been numerous attempts to increase the address space. Initially, with mode code 17 expanded sub addresses, the addressable memory increased from 59.5Kbyte (31 x 30 x 32 x 16 bit) to 7.6Mbytes (31 x 128 x 30 x 32 x 16 bit). Later introduced protocols such as the mass data transfer as defined in the MIL-STD-1760C, achieve a significant improvement with support for approximately 962Mbytes (255 files x 255 record x 255 blocks x 29 words x 16 bit). [4,5]

One of the initial FireWire design goals was hot-pluggable nodes on a self-configuring bus. The bus always undergoes a tree-identification process after a node is removed or added to the bus with the effect that the physical IDs are assigned dynamically. Although this is beneficial for consumer markets, it poses additional challenges for the avionics industry in that the nodes can change their addresses when another node on the bus fails. To illustrate this, consider the following example in Figure 3 of a FireWire network with its root node, branch nodes #2 and #4, and two leaf nodes #1 and #3. Leaf node #1 fails which is noticed by branch node #2. Node #2 will then initiate an overall bus reset. After the reset, the FireWire bus will first undergo a so-called tree-identification process, where all nodes determine which node is their parent/child, and the root node itself is assigned. A topology/speed map is created residing in the root node, which reflects the child/parent relationships between all nodes together with their speed capabilities.

Then the self-identification process begins. Although it is a deterministic algorithm that will always establish the same address assignment for the same topology, in the unfortunate event of failure on node #1 in the topology below, it has the effect that all nodes change their physical ID. Summarizing, the physical ID reflects the location of a node within the topology in respect to the root rather than identifies itself.

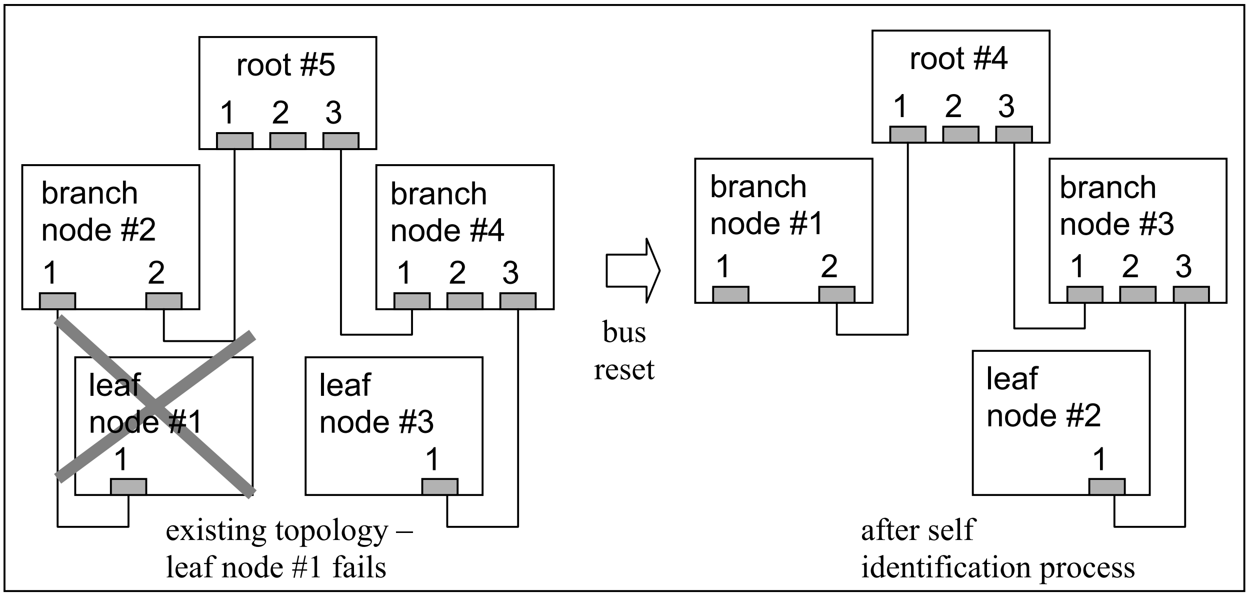

Figure 3: Dynamic Assignment of Physical IDs on the FireWire Bus

One way to resolve this issue would be to introduce hardwired addresses. This address would then uniquely identify a specific node within the bus. However, the under FireWire dynamically assigned physical IDs and this newly introduced hard configured node ID must coexist. This requires the introduction of an address resolution protocol that maintains the mapping between the two of them. The same problem is encountered in Fibre Channel's Arbitrated Loop where dynamic addressing is performed. A unique 64bit address (WorldWideName) is assigned by the IEEE standards organization that allows identification via a higher level protocol.

3.7 System Topology and Network Convergence

As discussed in the previous paragraphs, FireWire is suitable not only for data but also set-up, control and timing. With that a unified avionics interconnect can be established, that handles all aspects of data acquisition. This simplified infrastructure is attractive in that it reduces not only the amount of wiring, but also complexity and cost. Although the same is true for 1553, due to the limited number of supported nodes per bus, low data rate, and small address space, the system designer is forced to introduce numerous busses interconnected through gates, whereas with FireWire the number of bus bridges could be drastically reduced.

3.8 Scalability and Room for Growth

Scalability is a desirable characteristic for a new bus standard. However all serial busses which have no built-in concept of switch fabrics basically fail in this respect. There is a limit of maximum data throughput and maximum number of nodes that share the overall available bandwidth. Furthermore, with a high percentage of asynchronous traffic in a FireWire network, a cut-off point in the bus utilization will be encountered, as is typical for half duplex Ethernet networks. Once the utilization exceeds this point, the amount of arbitration conflicts (or collisions to use the Ethernet BaseT analogy) dramatically increases. As a result of this, the bus performance sinks. However, room for growth can still be provisioned for in that data throughput, maximum number of nodes and address space exceed current requirements substantially. Whereas MIL-STD-1553 cannot accommodate today's requirements, FireWire meets those easily and with its 64bit address space, up to 63 nodes per bus and up to 3.2Gigabit/sec data rate, it is capable of sustaining those of years to come.

3.9 Comparison

	MIL-STD-1553	**CAIS**	**FireWire**	**FibreChannel**	**Ethernet**
Origins/ Target Applications	Avionics bus control, set-up and data acquisition	FTI set-up and data acquisition	desktop industry video, audio, mass storage, entertainment	Storage Area Networks	Local Area Networks
Features	deterministic, predictable	deterministic, predictable faster and more address space	isochronous, big address space,	high speed SCSI, real-time, long reach	popular, cheap,
Physical Layer					
Coupling	transformer	transformer	transformer	transformer	transformer
Media	copper	copper	copper of fibre	copper or fibre	copper or fibre
Reach (point to point)	~100m	~100m	4.5m (STP) 50m (POF) 100m (GOF)	30m (copper) 500m (MM fibre) 10km (SM fibre)	100m (UTP) 500m (MM fibre) 10km (SM fibre)
Topology	daisy chain or star	daisy chain	daisy chain	ring or switched network	daisy chain or switched network
Address Space					
Busses	not inherent	not inherent	10bit	16bit (switched)	
Nodes	5bit	6bit	6bit	8bit (FC-AL: 126 only)	48bit
Address per Node	17bit (with MC17)	22bit set-up 17bit data	48bit		
Overall	7.6Mbyte	512Mbyte	16ExaByte	64MByte minimum	256TByte minimum
Communication model	shared memory	shared memory	shared memory	message passing	message passing
Speed	1Mbit/sec	10Mbit/sec	3.2Gbit/sec	4Gbit/sec	10Gbit/sec
Scalability	No	no	no	yes (switched)	yes (switched)
Reliability					
Redundancy	inherent	not inherent	not inherent	not inherent	not inherent but built into AFDX
Error checking	parity	parity	header + data CRC+checksum	packet CRC	header + data checksum
BER	10^{-12}	10^{-9}	10^{-12}	10^{-12}	10^{-12} (1000BaseX)
Confirmed Transfer	supported	supported	supported	supported	supported (TCP/IP)
Live at Power-up	Yes	yes	no	no	no
Timing					
Isochronicity	yes	not inherent	yes	not inherent but built into FC-AE [8]	not inherent
Synchronicity	no	yes	yes	yes	yes
Quality of Service	yes	yes	yes	yes	not inherent

Table 1: High Level Comparison of 1553, CAIS, FireWire, FibreChannel, Ethernet

Table 1 compares FireWire to other bus standards competing as a 1553 replacement for the avionics community. Although there are many other candidates, we have limited the scope of this comparison to FibreChannel, CAIS, and Ethernet. Address space refers only to addressing on a physical layer. For example, Ethernet and FibreChannel define different addressing schemes in embedded higher-level protocols. Reach depends on the chosen media interface, and in the case of Ethernet only some of the supported media interfaces are listed. The same applies to bit error ratio (BER) and speed which are a function of the physical layer. The terms isochronicity and synchronicity mean common time and common clock respectively. [6,7] Live at power-up reflects the fact that under FireWire, FibreChannel, and Ethernet (when used with dynamically assigned IP addresses as in DHCP), addressing needs to be resolved before data transmission can commence. Additional overhead is encountered in the FireWire environment due to the bus enumeration process which has to be terminated before transactions can resume.

4. CONCLUSIONS

There are numerous competing bus technologies that could potentially replace 1553. The success of one individual technology does not only depend on its technical superiority, but on the level of acceptance within the industry, which in return drives the maturity of a standard, the cost effectiveness of the components and its life time. This paper focussed on FireWire in particular and examined its suitability from a purely technical point of view. We found that the electrical interfaces specified in the 2nd amendment (UTP, POF, GOF) in terms of reliability, data rate and reach should meet the industry requirements, whereas the initially proposed DS signalling and the later introduced beta mode signalling over shielded twisted pair might prove to be insufficient due to limited reach, lack of electrical isolation between nodes, and lack of environmental integrity.

FireWire excels in its support for quality of service, error checking, isochronicity, and provides room for growth with its 16exabytes of address space and support for data rates of up to 3.2Gbit/s for future applications. It is especially suitable for digital video applications, but also supplies acknowledged traffic in the form of asynchronous transactions that could carry control and set-up operations. Furthermore, all peripherals on the FireWire bus can derive a clock from the bus signals. This is beneficial in that it avoids additional wiring while providing full synchronization within a distributed acquisition system.

However, additional work is required to handle the issue of dynamic node identification and redundancy. To ensure interoperability with other FireWire devices, it would be advantageous if the industry could come together to agree on a higher level protocol that handles the mapping of the dynamically assigned ID to a fixed address, as well as duplication and error checking when deploying a backup network to achieve dual redundancy.

Despite these few shortcomings that can be dealt with on other system levels, FireWire has the technical potential to play an important role as avionics bus or flight instrumentation network in future data acquisition systems.

REFERENCES:

[1] *IEEE Standard for a High Performance Serial Bus,* 1996,
IEEE Std 1394-1995, Institute of Electrical Engineers, New York
[2] *IEEE Standard for a High Performance Serial Bus 2nd Amendment,*
2002,

IEEE Std 1394b, Institute of Electrical Engineers, New York
[3] *FireWire System Architecture*
MindShare Inc, Don Anderson, Addison-Wesley, 2nd Edition
[4] *MIL-STD 1553, Designer's Guide Handbook*
ILC Data Device Corporation, 5th Edition
[5] *MIL-STD-1760C*
Department of Defense Interface Standard for Aircraft/Store
Electrical Interconnection System
[6] *IEEE Std 802.3 - 2000*
Institute of Electrical Engineers, New York
[7] www.fibrechannel.org
[8] *Technical Report for Information Technology – Fibre Channel-Avionics Environment (FC-AE)*
American National Standards Committee, INCITS, Information Technology,
TR-31-2002