# Ethernet for Space Flight Applications

Evan Webb
NASA-Goddard Space Flight Center
Code 561: Flight Electronics Branch
Greenbelt, MD 20771
301-286-2667
Evan.Webb@gsfc.nasa.gov

*Abstract*– NASA's Goddard Space Flight Center (GSFC) is adapting current data networking technologies to fly on future spaceflight missions. The benefits of using commercially based networking standards and protocols have been widely discussed and are expected to include reduction in overall mission cost, shortened integration and test (I&T) schedules, increased operations flexibility, and hardware and software upgradeability/scalability with developments ongoing in the commercial world. The networking effort is a comprehensive one encompassing missions ranging from small University Explorer (UNEX) class spacecraft to large observatories such as the Next Generation Space Telescope (NGST). Mission aspects such as flight hardware and software, ground station hardware and software, operations, RF communications, and security (physical and electronic) are all being addressed to ensure a complete end-to-end system solution.

One of the current networking development efforts at GSFC is the SpaceLAN (Spacecraft Local Area Network) project, development of a space-qualifiable Ethernet network. To this end we have purchased an IEEE 802.3-compatible 10/100/1000 Media Access Control (MAC) layer Intellectual Property (IP) core and are designing a network node interface (NNI) and associated network components such as a switch. These systems will ultimately allow the replacement of the typical MIL-STD-1553/1773 and custom interfaces that inhabit most spacecraft.

In this paper we will describe our current Ethernet NNI development along with a novel new space qualified physical layer that will be used in place of the standard interfaces. We will outline our plans for development of space qualified network components that will allow future spacecraft to operate in significant radiation environments while using a single onboard network for reliable commanding and data transfer. There will be a brief discussion of some issues surrounding system implications of a flight Ethernet. Finally, we will show an onboard network architecture for a proposed new mission using Ethernet for science data transport.

0-7803-7231-X/01/$10.00/©2002 IEEE

TABLE OF CONTENTS

## 1. INTRODUCTION

Spacecraft designs for scientific exploration tend to be one-off solutions to specific problems. Each mission presents a set of unique challenges that are met by a design team that consists of spacecraft system engineers and specialists in various subsystems. The solutions for basic problems such as electrical and data interfaces between the subsystems are generally approached by designing with piece parts that are acceptable to the parts program being used by the mission. Radiation tolerance, vibration, shock, and other requirements intended to improve reliability of the spacecraft systems are stringent and expensive to achieve for electronics manufacturers. Furthermore, the market for these high reliability parts is small and as a result there is a constantly changing pool of suppliers and parts that are used from mission to mission. The use of Commercial Off The Shelf (COTS) electronics can help for certain missions but parts screening is required and tiny changes in a manufacturing process to improve yield can have disastrous effects on parts reliability in the extreme conditions of spaceflight.

### *What's the Problem?*

This set of circumstances results in a considerable amount of reinventing the wheel. There have probably been a thousand different versions of the ubiquitous clock and data interface used on NASA and other space missions; each one involved a few engineers sending emails back and forth about set-up and hold times, whether the enable is active-high or active-low, what the frame length is, how to specify the min and max timing, etc. One unlucky person gets volunteered to write the ICD and several pots of coffee later the draft version is circulated and various more senior engineers correct spelling errors, update logos and names of organizations, and change what is bolded and italicized. Finally a good enough version floats up the signature chain of the groups responsible for each side of the interface.

Breadboard interfaces are prototyped and tested at temperature for design verification. Then the document is put under configuration control and it takes five more signatures to, say, change the set-up time spec from 12 ns to 14 ns.

This is, of course, a colossal waste of time. Everyone recognizes that fact but the solution would involve reusing designs with parts that the program can't get anymore or were designed on a CAD system that is no longer supported at the organization, or... Admittedly there are numerous examples of "series" spacecraft such as GOES that reuse hardware designs but for new missions trying to aggressively achieve science goals with very limited budgets (mass, power, cost) this is difficult to do.

Standardization on an electrical bus has been a major step to simplify the sending and receiving of commands and telemetry onboard spacecraft. The MIL-STD-1553 (and occasionally the fiber optic 1773 version) bus is the de facto standard for the command bus, and in some cases a small amount of science data can also be transported within its narrow 1 MHz (roughly 600 kbps throughput) confines. On a number of missions, however, separate high-speed interfaces for science data, discrete controls for devices not on the bus, and discrete interfaces for timing are used. These are the interfaces that hog spacecraft power, and can result in unsightly (and heavy!) trunks of cables snaking around the spacecraft. Cable mass is a significant percentage on every spacecraft; the American Institute of Aeronautics and Astronautics (AIAA) recommends allocations of 5% to 8% of the on-orbit dry mass budget for planetary or other scientific spacecraft [1]. Each interface must be carefully tested, a process which consumes hundreds of hours at the board, box, and spacecraft/observatory level due to the (potentially) hundreds or thousands of individual pins that must be probed for each safe-to-mate procedure. Rework of these cable bundles is similarly difficult and expensive in cost and schedule.

The limitations of 1553 have caused (some) engineers to fantasize about how wonderful it would be to have a network onboard the spacecraft that was as good as their lowly Ethernet local area network (LAN) which connects all their dirt-cheap Ground Support Equipment (GSE). So that is what we in the Flight Electronics Branch at Goddard Space Flight Center (GSFC) proposed to do in the Spring of 2000 with the SpaceLAN (Spacecraft Local Area Network) technology development effort. SpaceLAN is intended to develop a spaceworthy version of Ethernet and demonstrate its performance and suitability to replace the typical combination of MIL-STD-1553 and custom interfaces now flying.

*An Ethernet Primer*

*IEEE 802.3 Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications* (the Ethernet standard) [3] was first released in 1980. The standard was based on work originally done by Dr. Robert M. Metcalfe in the 1970s at the Xerox Palo Alto Research Center. It described a frame-based shared media network running over thick coax, and used the CSMA/CD algorithm to regulate use of network resources (i.e. bandwidth) without the need for central arbitration. In short, this algorithm works by having each station listen on the bus to make sure no one else is transmitting, and if not then transmit its frame. If two or more stations try to transmit at close to the same time the collision is detected and all stations transmit a jam signal and then back off for a random (but exponentially increasing for successive collisions) period of time before trying again. With a small number of stations on the network this scheme works fine, but the probability of a collision increases with more stations and /or more frequent transmissions. Obviously at some point if a bus is overloaded then each station will encounter more and more collisions and the bus utilization (effective throughput) drops to a very low figure. A general rule of thumb for CSMA/CD networks is to keep average throughput below 50% of capacity to ensure stability.

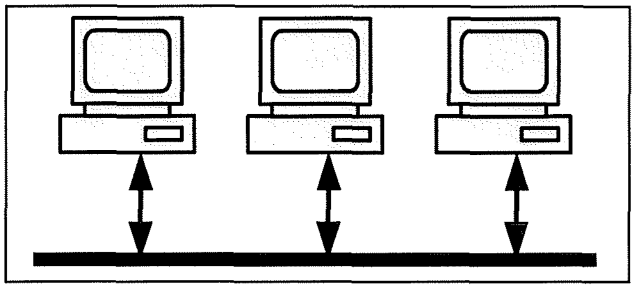


Figure 1 - CSMA/CD Shared Media Network

Since that first version of the standard many revisions have added performance and features to Ethernet resulting in its nearly universal adaptation for business Local Area Networks (LAN). Speeds up to 1 Gigabit per second (with 10 Gbps currently being finalized in committee) and other physical layers such as twisted pair and fiber optics have been incorporated into the standard. Also, the use of switches has allowed full duplex networks to become common. Switches have a switch fabric (usually implemented as a bus, shared memory, or crosspoint matrix) that connect each port and allow temporary circuits to exist between stations for the duration of the frame. This allows the full bandwidth in each direction (100 Mbps in the case of Fast Ethernet) to be used on each transmission through the network, multiplying its effective throughput.

0-7803-7231-X/01/$10.00/©2002 IEEE

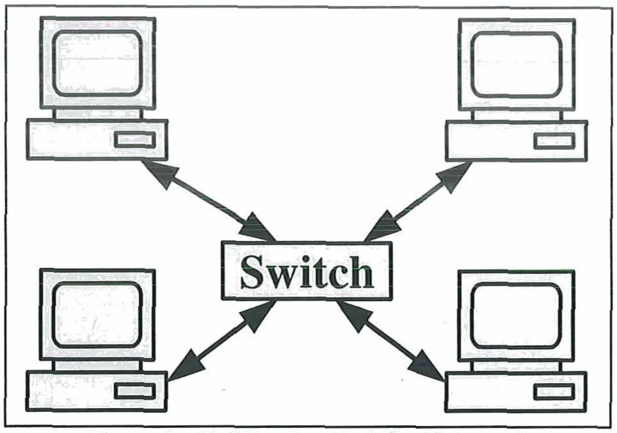


Figure 2 - Full Duplex Switched Network

Our SpaceLAN effort is currently working towards having a full duplex 100 Mbps Ethernet flying on an upcoming NASA mission. Each node in the network will be one of the spacecraft subsystems or science instruments.

## 2. GSFC Network Components Development

*Network Node Interface Development*

Our first step towards implementing Ethernet aboard spacecraft was to develop the network node interface (NNI), i.e. the Ethernet "chip" that any subsystem would have for communicating with the onboard LAN. In PC networks the NNI is usually implemented on a single card and is therefore referred to as the network interface card (NIC), but our development is obviously aimed at an embedded application in the spacecraft components. Our strategy was to make use of the wide range of intellectual property (IP) cores available from commercial developers. We found a 10/100/1000 MAC (Media Access Control) core and decided to concentrate on the 10/100 version first with a gigabit interface as a future step.

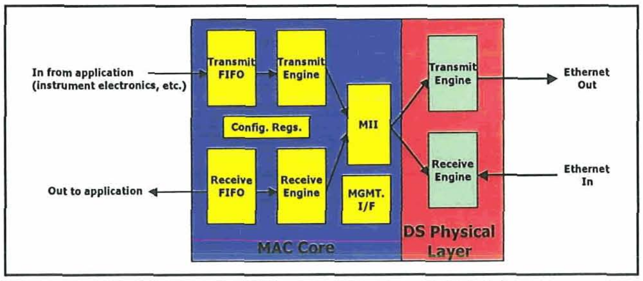


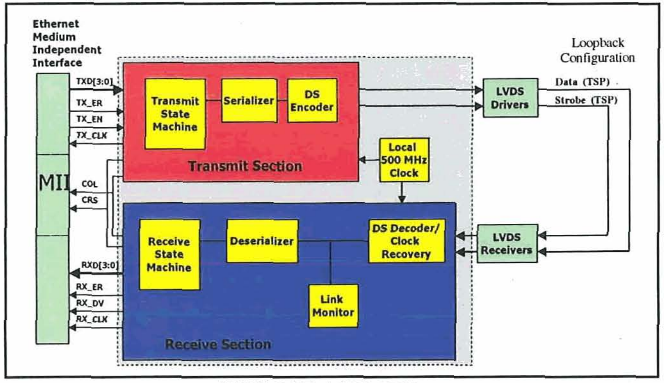
Figure 3 - Ethernet Protocol FPGA

The MAC core consists of the standard elements called out in the 802.3 standard: backend (application) interface, transmit and receive protocol engines, Medium Independent Interface (MII) to the physical layer, and management interface for configuration and maintenance of SNMP (Simple Network Management Protocol) data objects. The protocol FPGA (Field Programmable Gate Array) including the MAC core and physical layer protocol is shown in Figure 3.

*LVDS-based Physical Layer*

One of the keys to adapting Ethernet to space use in significant radiation environments such as the typical imaging satellite 700 km sun-synchronous orbits, other polar low-earth orbits (LEO), medium earth orbits (MEO), geosynchronous orbits, or planetary missions, is using an appropriate physical layer. An all-digital system is frequently more robust to timing variations induced by total dose radiation degradation than those with analog-based clock recovery. We have developed our own physical layer that interfaces at the Medium Independent Interface (MII). In this way we can easily test the physical layer separately similar to plugging in an external transceiver. The physical layer sublayers (Physical Coding (PCS), Physical Medium Attachment (PMA), and Physical Medium Dependent (PMD)) are divided very similarly to 10/100BASE-T, and with the Data-Strobe (DS) encoding we are unofficially calling this network 10/100BASE-DS.

Figure 4 - LVDS-based Ethernet Physical Layer



(10/100BASE-DS)

Another GSFC Flight Electronics Branch technology development effort, the SpaceWire network (derived from the IEEE-1355 standard) uses the Low Voltage Differential Signaling (LVDS) drivers and receivers with DS encoding and will be flying on the upcoming Swift mission [4]. These parts are available in very high-speed radiation hardened versions so we decided to investigate their use as a new physical layer for Ethernet. LVDS uses a current mode driver and a termination of 100 - 120 ohms (matched to the cable) at the receiver to produce the differential voltage at the receiver of approximately 400 mV peak-to-peak. Each interface is point-to-point with only one driver and receiver for best signal quality. We plan to use rugged DB-9 connectors in place of the RJ-45 used in TX systems, and signal quality testing will hopefully validate this choice.

The DS encoding method is similar to Manchester encoding: the strobe is toggled at each rising clock edge if the data value remains the same as the previous value. This encoding ensures a transition either on the data line or the

0-7803-7231-X/01/$10.00/©2002 IEEE

strobe line every clock period, allowing for reliable clock recovery and zero DC bias at the receiver. As shown in Figure 5, the (half-speed) transmitted clock can be recovered at the receiver by simply XORing the data and strobe together. The DS encoding is attractive in several ways compared to other alternatives; a lower frequency of transitions than an explicit clock and data interface (narrower power spectral density, extending the allowable cable length), and reliable clock recovery for much less complexity at the receiver than a single twisted pair using line codes such as Manchester, pseudorandomized RZ/NRZ, or various biphase formats.

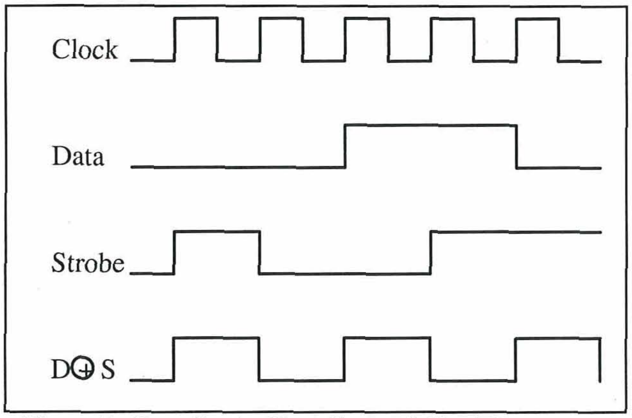


Figure 5 - Data-Strobe Encoding and Clock Recovery

*Compact-PCI Test Board*

We decided to implement our first test board using the Compact-PCI standard in the 6U form factor. Our test system is a Motorola MCP-750 with a 366 MHz Power-PC 750 processor. We selected this system for two principal reasons: compatibility with the test system of another group on center (the Flight Software Branch, who would be providing software support to our effort) adapting the standard realtime flight software suite to work with various new technologies including Internet Protocol (vs. CCSDS (Consultative Committee for Space Data Systems) packets) and the Linux operating system; and the radiation-hardened version of the PowerPC 750, the RAD750, is expected to become the processing platform of choice for many new missions in the future.

We used an "everything but the kitchen sink" approach and included on our card a number of different transceivers and physical layer drivers. By so doing we could verify the correct operation of both the Ethernet protocol FPGA and (independently) the LVDS PHY. The different interfaces included a commercial 10/100BASE-TX transceiver, a commercial Gigabit Ethernet transceiver, the native LVDS drivers on the Altera FPGA, and the flight-qualified UTMC (United Technologies Microelectronics Center) LVDS drivers. The block diagram is shown in Figure 6.

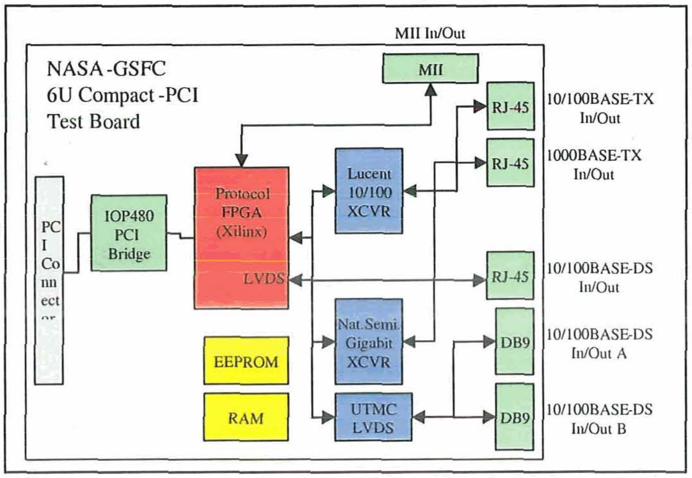


Figure 6 - Compact PCI Flight Ethernet Test Board

*GSFC Network Switch Development*

Since the 10/100BASE-DS network will not support multiple nodes on a single wire we needed a central network device to connect all the nodes. A collision-mode hub was considered but a full-duplex switch was selected for its inherent performance benefits. A side benefit of the switch is simplified testing without having to test (initially) the parts of the MAC algorithm involved with the collision-backoff process. A block diagram of the switch is shown in Figure 7.

The switch is intended to be a single-card design to fit on a Compact-PCI 6U card that will fit in the same enclosure as the spacecraft Command and Data Handling (C&DH) system. The switch architecture is a crosspoint-matrix-based (crossbar switch) switch fabric, store-and-forward design. We selected the crosspoint matrix for ease of design, simple multicast compatibility using multiplexers, and less reliance on large memories that can be expensive for flight hardware. The downside of the crossbar implementation is difficulty in scaling to large numbers of ports, but on a spacecraft relatively few (probably 10 - 20) ports are necessary. We are generally following the recommendation in the IEEE 802.1D MAC Bridges specification with some simplifications. The design is rather low-tech in comparison with current office environment fare, but the LAN on a spacecraft has no need for many of the features used in business LANs. Features such as dynamic learning, VLAN support, etc. are left out to allow a processor-less (i.e. state machine based) design. Since the network is very small and will be manually configured we are not implementing Spanning Tree support, but for future missions involving constellations with crosslink communications this will be necessary. In our simple test case on-orbit reconfiguration is most likely to be a result of a collision with space junk that would have bad (!) consequences for the mission as a whole. In any case we can add new features in a future version if they are required.

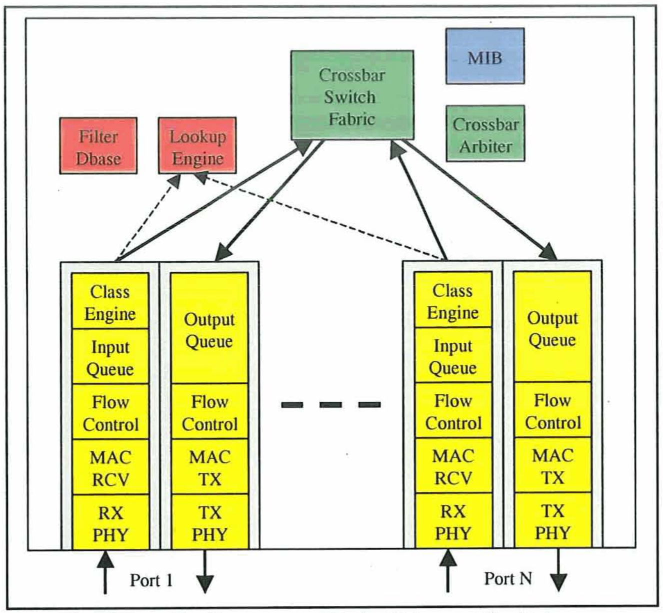


Figure 7 - Prototype Flight Ethernet Switch

*Development Roadmap*

The next step for the 10/100 speed flight Ethernet is to combine the protocol FPGA and LVDS drivers onto a single ASIC. The need for this will be determined according to the requirements of the flight mission planning to use Ethernet. After our 10/100 efforts have progressed to the point where a flight project can simply take these parts and build a network we intend to start development of space-qualified Gigabit Ethernet. The LVDS drivers do not map very well to the 1.25 Gigasymbols/sec required for the 8B/10B encoded data rate so we will pursue adapting rad-hard fiber optics technology for the physical layer. A proprietary parallel fiber technology developed under a NASA Small Business Innovative Research (SBIR) is currently under consideration for this purpose. An improved switch design to support the gigabit network will also be required. The development roadmap for the SpaceLAN project is shown in Figure 8.

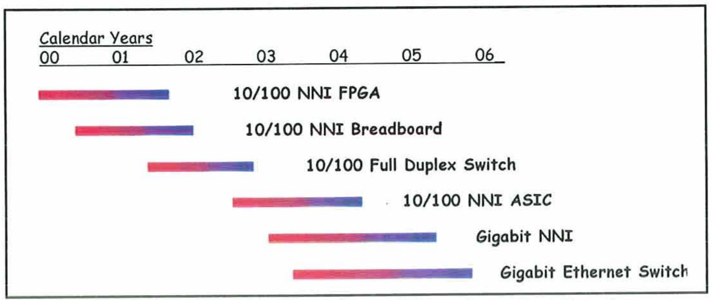


Figure 8 - GSFC Flight Ethernet Development Roadmap

## 3. SYSTEM IMPLEMENTATION ISSUES

At this point many old-school engineers will raise two important objections to using Ethernet to replace 1553, namely:

1. Ethernet does not have deterministic latency for the frames like the rigidly-scheduled 1553. How can you ensure that the frames will arrive at their destination in a fixed period of time?

2. Ethernet is not a redundant network like 1553. What if something goes wrong? Do my commands and telemetry disappear for awhile?

The short answers to these questions are 1) you can't, but you sort of can, and 2) not necessarily.

*Bounded Deterministic Latency*

In the original CSMA/CD Ethernet each station contended for the bus with an exponentially increasing penalty for collisions and it was anyone's guess when your frame would actually go out. But for the full-duplex links there is no arbitration (for the link) and each frame goes out immediately, so on an (unrealistic) network of only two nodes there is no latency in either direction. For a real network of several nodes connected by full duplex links through a switch then the source of latency becomes the transit time through the switch itself. This is due to delays in the various steps the switch must perform to route the frame internally. A detailed discussion of this process is beyond the scope of this paper, but the classification engine, the lookup engine, the switch fabric arbitration, and the possible use of a PAUSE command are all factors in switch latency. The solution to this problem lies in the switch design, the network design, and avoiding the temptation to over-specify the latency requirement. We can briefly discuss a few of the most important issues relating to latency through the network.

The first item to consider is the relative speed difference between Fast Ethernet (100 Mbps) and MIL-STD-1553 (<1Mbps). The performance advantage of Ethernet can be used to provide frames that are not strictly deterministic in their arrival times, but are deterministic within some bounds for each link based on factors such as frame length, switch design, and the average and peak throughput of the network. Frames can range in length from 72 to 1526 bytes (for preamble, header, payload, and CRC), which on Fast (100Mbps) Ethernet is a transmission time of 5.76 to 122.08 µs. These transmission times are short compared to the principal contributors to latency, delays in switch fabric arbitration and queueing. In comparison, an 1100 byte CCSDS VCDU (Virtual Channel Data Unit) transmitted over 1553 requires over 14.6 milliseconds; in that time some 120 maximum size Fast Ethernet frames would have been transmitted.

The usual tradeoff in networking involves reduced latency by using short frames vs. increased efficiency with longer frames (except for the special case of a cut-through switch), so for applications with a low latency requirement frames are kept as short as is practical. For a network with twelve nodes using (on average) a medium length frame (say, 500 bytes) with transmission times of 40 μs per frame a typical arbitration delay for the switch fabric (assuming a crosspoint matrix) would be 12 * 40 = 480 μs, or about a half a millisecond. This assumes a round-robin arbitration scheme with no priority. If a priority scheme is implemented at the arbiter then delays can be even less; priority can be assigned on a port or frame basis to improve the frame's standing with respect to the switch fabric arbitration.

The other major source of latency in the switch is queueing delay. This is a function of the traffic characteristics of the source and destination. If the source is sending a lot of data to multiple destinations then it is likely that delays at the input queue of the switch will be higher, similarly if the destination node receives traffic from multiple sources then delays at the output queue of that node will also be higher. At the other extreme if the destination node only receives traffic from a single source then its output queue will essentially be empty at all times. Typically in switch designs the input queue is small compared to the output queue and the goal is to rapidly move the frame from the input to the output queue. To do this the internal throughput of the switch must be at least N times (where N is the number of ports) higher than the speed of each link.

As an example, consider an attitude control system (ACS) actuator that requires a 10 Hz loop to the C&DH. This means 100 ms between frames. Assume for sake of argument that 50 ms (half the available time) is required for processing at the C&DH to calculate a value and produce the new frame. Then 50 ms is available to propagate the frame from C&DH to actuator. This requires only 11.5 μs (5.76 μs times two for the C&DH to switch and switch to actuator links) to transmit, so in essence there is still 50 ms available for the switch latency. We can be conservative and assign 1 ms for miscellaneous other delays within the switch (lookup, classification) and using 0.5 ms for the arbitration delay from above we have 48.5 ms available for the queueing delay. The port output queue in this case will be empty since the actuator only receives commands from the C&DH, so the output queue delay is negligible (microseconds). Now we have 48.5 ms for the input queue delay, which is equivalent to transmitting about 400 maximum length frames and would seem to be much more delay than we would actually encounter. The actual delay from frame to frame may vary widely (based on the state of the switch when each frame arrives), but as long as it fits in the timing constraints for the control loop the overall performance is acceptable. This is obviously not a rigorous analysis (some would even call it hand-waving) but I hope that it indicates that the question of latency is not a showstopper for the use of switched Ethernet in tight timing loops.

While this analysis suggests that switched Ethernet can be appropriate for many missions there may be some situations where the switched Ethernet model will not meet a difficult latency requirement. In that case, however, 1553 would not meet the requirement either. We plan to do performance testing of a model onboard network to provide the real data that spacecraft designers will need.

*Redundancy, Redundancy, Redundancy, Redundancy,..*

Redundant networks are often required to maintain high reliability and availability over long mission lifetimes. Ethernet is not an inherently redundant network, but there are various ways that a flight mission could implement redundancy, and the method chosen will depend on the mission requirements. There are a myriad of choices in types of redundant networks, but we will concentrate on two basic choices: cold standby sparing (primary network powered and running, backup network not powered) and hot standby sparing (both networks powered and running).

For the case of systems that are non-critical to the health and safety of the spacecraft it may be acceptable to trade a delay between recognizing a network fault and correcting it for other benefits. A cold standby sparing system requires that the backup network be powered up and configured in the case of a fault before communications are reestablished. This may require several seconds if the spacecraft has the autonomy to perform this action on its own, or it may have to wait until the next ground contact minutes or hours later if ground commands are required. The benefit of this type of sparing is a reduction in power consumption since the backup network is unpowered. The exact configuration of this type may vary: it could include a duplicate interfaces for each node with a separate switch for each network, or merely have a second interface for certain nodes on extra ports on a single switch (or a redundant switch) that can be powered up and used as necessary. Other system questions remain such as how to recognize the fault and how to switch to the backup network when the primary is down.

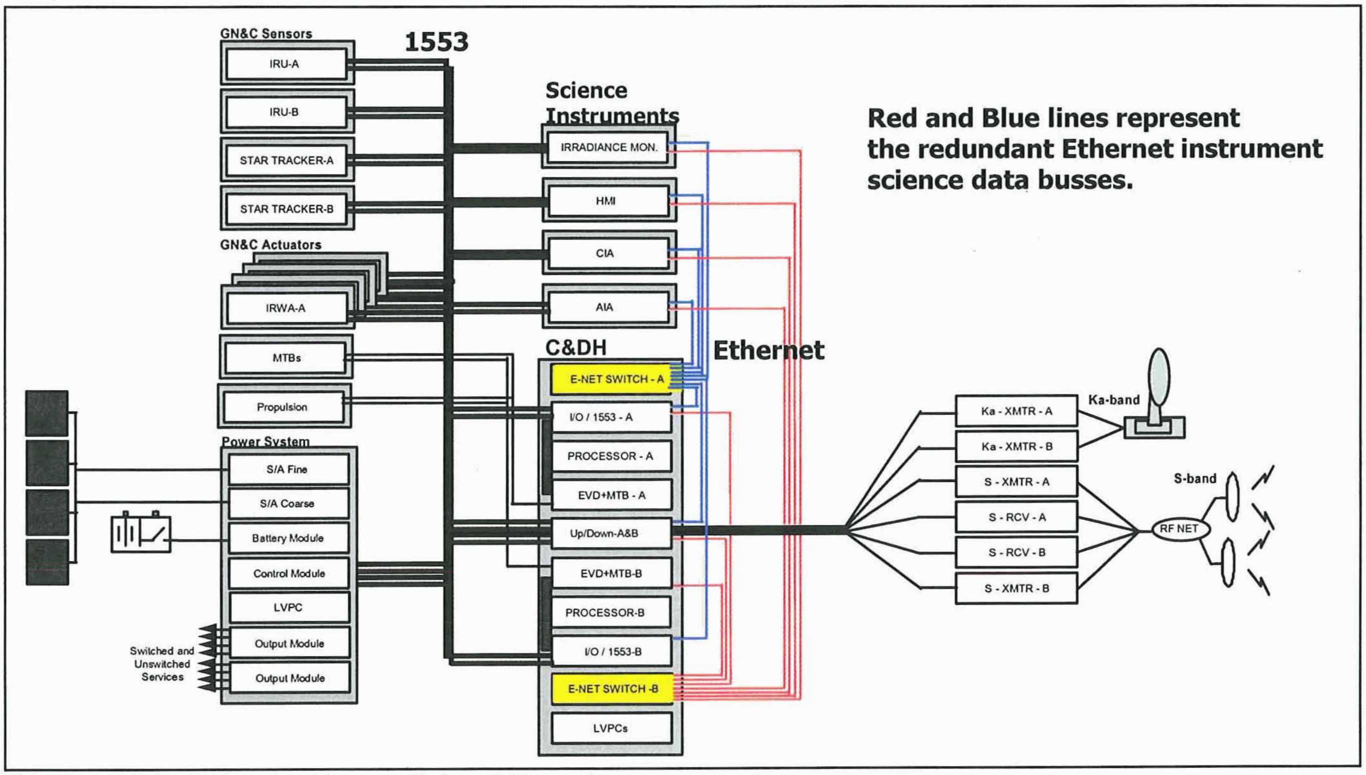
If a 1553-style retransmit on a separate bus is required then two independent Ethernet networks are required with both powered and the host system at each node must be capable of receiving commands simultaneously on either interface. This requires two switches and adds complexity at each node. A scheme to designate the "currently active" network is required, and one node must be able to send configuration commands to the others. This can be accomplished with application software at each node and could conceivably replicate the protocol followed on 1553. This added degree of complication is justified when Ethernet is to be used as the spacecraft command network with mission-critical commands and data flowing across it.

## 4. APPLICATION: SOLAR DYNAMICS OBSERVER

As with any new technology being applied to the spaceflight arena there is a certain amount of hesitancy before completely switching from a proven quantity. MIL-STD-1553 is still being baselined for many future missions for the onboard data/command network, but the area where Ethernet is likely to have its first significant application is for science instrument data transmission. Here the bandwidth requirements are generally much higher, and there is the potential for replacing multiple custom interfaces with a single standardized bus.

The first space mission to consider using our flight Ethernet is the Solar Dynamics Observer (SDO). This mission is a proposal from GSFC to NASA Headquarters and is still awaiting final approval. Launch is scheduled (in the proposal) for mid-2006. SDO, if approved, will be a geosynchronous satellite monitoring solar phenomena with a suite of four instruments. We have proposed using 100 Mbit Ethernet to connect the instruments (via a switch) with the C&DH through the processor card, uplink/downlink formatter card, and the input/output/1553 interface card (also known as the "junk" card). As it is currently configured this mission will require the network switch to have seven ports, although we will probably have ten ports available for additional capacity. This will be a fully redundant, cold-spare system consistent with the reliability plan for a five year mission.

Data rates for the instruments have not been finalized but are expected to be in the 60 - 100 Mbps range. If any single instrument exceeds the 100 Mbps limitation of a single link then a second link will be required for that instrument, with an additional port on the switch. It will be a fairly straightforward requirement to have the instruments output electronics write in a ping-pong scheme to the two network nodes. Ultimately since SDO is in a geosynchronous orbit with 24/7 real-time data delivery the RF downlink rate is the limiting factor for system throughput. The SDO onboard command and data network block diagram is shown in Figure 9.



**Red and Blue lines represent the redundant Ethernet instrument science data busses.**

Figure 9 - Solar Dynamics Observer Onboard Network

## 5. CONCLUSIONS

Ethernet is the most popular networking technology worldwide for LANs today. The concept of the spacecraft LAN is a natural fit when considering the number of nodes, cable lengths, and bandwidth requirements. At the Flight Electronics Branch of the NASA Goddard Space Flight Center we are actively working towards applying Ethernet technology combined with experience in developing spaceflight electronics to achieve orders of magnitude improvements in performance of onboard communications with reductions in system mass, power consumption, and cost.

0-7803-7231-X/01/$10.00/©2002 IEEE

## ACKNOWLEDGMENTS

The author would like to thank Phil Luers, Bob Stone, and Terry Smith of GSFC Code 561 and Keith Hogie, CSC, for their assistance and review of this paper.

## REFERENCES

[1] American Institute for Aeronautic and Astronautics, Guide for Estimating and Budgeting Weight and Power Contingencies, AIAA-G-020-1992.

[2] Charles E. Spurgeon, *Ethernet: The Definitive Guide*, Sebastopol, CA: O'Reilly, 2000.

[3] Institute of Electrical and Electronics Engineers, Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications, IEEE Std 802.3: New York, New York, IEEE, 2000.

[4] European Cooperation for Space Standards, *SpaceWire - Links, Nodes, Routers and Networks, ECSS-E-50-12 Draft 0*: Noordwijk, The Netherlands, European Space Agency (ESA), 2000.

## BIOGRAPHY

***Evan Webb*** *is a computer engineer at the NASA Goddard Space Flight Center in Greenbelt, MD. He is the Lead Engineer and Project Manager for the SpaceLAN technology development effort, and was previously the Lead System Engineer for the EO-1 WARP solid state recorder. Immediately prior to joining GSFC at Greenbelt Mr. Webb worked at the NASA Wallops Flight Facility on Wallops Island, VA in the Aircraft Programs Branch. There he was responsible for aircraft data system development and also served as project engineer and mission manager for numerous airborne geoscience remote sensing missions both within the U.S. and abroad. As an undergraduate Mr. Webb was a co-op student at Booz, Allen and Hamilton working in the Space Electronic Warfare Group performing modeling and threat assessment under contract from the Strategic Defense Initiative Organization (SDIO). Mr. Webb also worked at the University of Maryland Space Physics Laboratory as an electronics technician building the Low Energy Ion Composition Analyzer (LEICA) instrument for the SAMPEX mission. Mr. Webb has a BSEE from the University of Maryland (1990), an MSEE from Johns Hopkins University (1995), and an Advanced Master's Certificate in Computer Engineering from Johns Hopkins University (1997).*

0-7803-7231-X/01/$10.00/©2002 IEEE