A STANDARD LINK-LAYER PROTOCOL FOR SPACE MISSION COMMUNICATIONS [1]

Ron Parise, Keith Hogie, Ed Criscuolo
Computer Sciences Corp

Rick Schnurr
NASA Goddard Space Flight Center

John Wesdock, Mark Burns
ITT Industries

ABSTRACT

A necessary step for using Internet Protocols in space is to establish the basic link-layer framing protocol for delivering Internet datagrams over satellite RF links. This paper discusses the low-level data link issues related to using the ISO standard High-level Data Link Control (HDLC) protocol to support spacecraft communications. A major driver for using HDLC is its very wide usage in the Internet today and the large amount of commercially available network equipment and test equipment.

The results of a high-fidelity link simulation using HDLC are presented along with results of tests performed in 2000-2001 using Internet protocols over HDLC on the UoSAT-12 spacecraft. A rationale is provided for the selection of HDLC/Frame-Relay framing along with the IETF multi-protocol encapsulation. It also discusses the historical usage of HDLC on over 70 satellite missions.

The paper also describes how HDLC relates to various applications of forward-error-correction (FEC) coding techniques, such as convolutional coding and Reed-Solomon. It describes approaches for using these techniques in ways that are independent of the protocols used at the data link layer and above. It covers issues primarily related to layer 2 (Data Link) and its relationship to layer 1 (Physical). It does not cover layer 3 (Network) and above.

INTRODUCTION

The use of the ubiquitous IP protocol suite on space-to-ground links has many wide ranging advantages. These include the availability of low cost network hardware, application software, and experienced programmers. In order to take maximum advantage of the recent explosion in technology, the space-to-ground links must be compatible with ground based technology down to at least the link layer. There have been a number of groups experimenting with encapsulating IP into the currently used CCSDS transfer frame format as the link layer. Although perfectly viable, this approach has the disadvantage of not being compatible with any commercial network hardware currently on the market. A non-standard gateway is then required to receive the encapsulated IP and relay it in more commonly used ethernet or frame-relay format. This negates many of the advantages of using IP since specialized hardware and software is still needed in all phases of the mission life from development through operations.

In this paper we consider the HDLC/Frame-Relay link layer protocol as a possible candidate for space missions having downlink data rates less that 45Mb/s. Frame-Relay is commonly used in commercial networks for wide area network connections and is supported by many vendors. Our investigation has involved several phases. A detailed laboratory simulation comparing the performance of the current CCSDS protocol with HDLC in both the uncoded and convolutionally coded cases was performed using an existing TDRSS link simulator developed by ITT industries. A flight test of IP over the Frame-Relay protocol was also performed by porting an IP stack with some basic applications to the UoSAT-12 spacecraft that is owned and operated by Surrey Satellite Technology Ltd.

AN OVERVIEW OF THE HDLC PROTOCOL

Based on its near-universal use on the terrestrial Internet, NASA's Operating Missions as Nodes on the Internet (OMNI) project chose HDLC framing as a candidate link-layer protocol on space-to-ground links. This allows simple, straightforward interfacing with existing commercial routers in the ground station. HDLC has been used in communication equipment for over 30 years and provides basic framing for many serial line protocols such as IBM's synchronous data link control (SDLC), Frame Relay, X.25, SLIP, PPP, LAPM, and LAPB.

As indicated in figure 1, at the physical link layer, HDLC framing is extremely simple, consisting of only a 1-byte flag pattern, a variable number of data bytes, and a 2-byte CRC. During any idle time, successive flag bytes are output until the next frame begins. Flag bytes consist of a zero bit, 6 one bits, and a zero bit (01111110). In order to prevent this pattern from occurring in the data, the HDLC hardware performs "bit stuffing" when sending data. Any sequence of 5 one bits in the data automatically has a zero bit inserted after it, thus insuring that any sequence of 6 consecutive one bits *must* be a flag byte. On receipt, these extra zero bits are automatically removed from the data by the hardware.

While the primary purpose of "bit stuffing" is to ensure the uniqueness of the flag byte, it also has an additional benefit. It prevents long unbroken strings of ones from being sent to the transmitter. These periodic transitions are important at the receiver, where a bit-synchronizer depends on them to extract the clock and data bitstreams from the raw signal. Along the same lines, the use of standard non-return-to-zero (NRZI) coding for the HDLC output will insure that an unbroken sequence of zero bits in the data stream becomes transformed into an alternating sequence of ones and zeros. Thus, the use of "bit stuffing", idle flag bytes, and NRZI coding insures that the transmitter will never send an unmodulated carrier, and the receiver will see a transition *at least* once every 6 bit times. Some modern receivers however have difficulty maintaining lock on signals containing a repeating data pattern. Standard bit-scrambling algorithms that are currently used to avoid patterns in a bitstream are also applicable here. It is important to note that these "space specific" requirements can be met by standard COTS hardware and protocols without inventing any "space specific" solutions. It should be further noted that these solutions are isolated to the lowest layer and are transparent to the upper layers. None of the protocols layers need to worry about generating "fill packets" or "fill frames".

The OMNI project considered various commercially available encapsulation mechanisms for use over HDLC. There were two major criteria for selecting the encapsulation method to use:

- the encapsulation could not require full-duplex links since full-duplex links might not be available during a spacecraft emergency
- the encapsulation must be interoperable between many vendors routers since no group can ensure that all routers at all ground stations will come from the same vendor

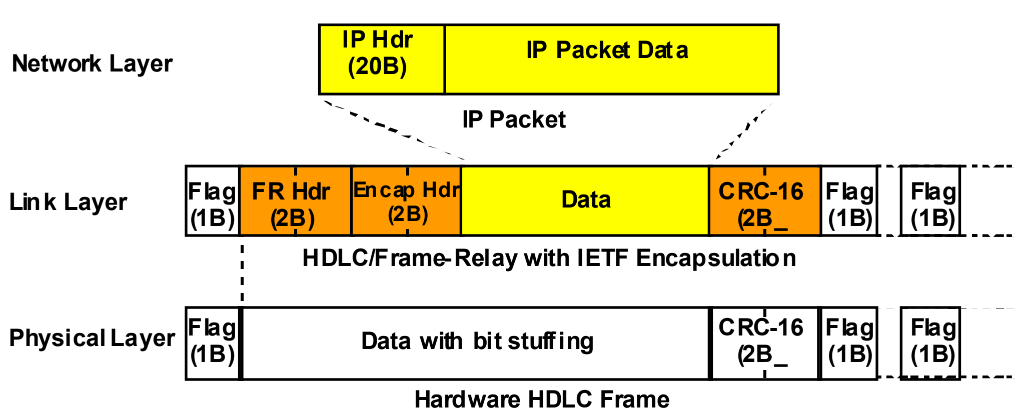

Figure 1 - HDLC/Frame Relay/IP formats

The first criteria ruled out protocols like Serial Line IP (SLIP) and Point-to-Point Protocol (PPP) because they need full-duplex links for parameter negotiation at startup. The second criteria ruled out protocols such as Cisco's default HDLC encapsulation which uses a Cisco specific HDLC header.

This led to the choice of the IETF encapsulation for multi-protocol over frame-relay/HDLC specified in RFC 2427. In the OMNI tests with UoSAT-12 the actual header format consisted of simply inserting 4 bytes of fixed information at the start of each HDLC frame. The first 2 bytes are a standard Frame Relay header with a few status bits and a virtual channel number or Data Link Connection Identifier (DLCI). Also, since this is a standard Frame-Relay header, a spacecraft could actually use the DLCI to provide additional channelization and routing in addition to the IP capabilities. This could be used along with standard Frame-Relay equipment at the ground station. The next 2 bytes in the header simply indicate that the contents of this frame are an IP packet. There are also standard IETF definitions that allow the transport of other protocols in the data area of the frame.

This data link framing provides capabilities identical to those used by current spacecraft. An application level science or telemetry packet inside of a User Datagram Protocol (UDP) packet with an IP header and HDLC is delivered through space exactly like current data. The main difference is that by using IP and HDLC headers the data leaving the spacecraft is in a format that can be directly ingested by COTS Internet equipment on the ground.

Supporting data rates over 45 Mbps using commercial routers requires using a framing technique other than just HDLC. Commercial routers have interfaces that support data rates up to 45 Mbps using HDLC framing over High-Speed Serial Interfaces (HSSI) but shift to Synchronous Optical Network (SONET) interfaces for data rates of 155 Mbps, 622 Mbps and 2.4 Gbps. These interfaces have traditionally used Asynchrounous Transfer Mode (ATM) cells to frame IP packets over SONET.

One objection to using ATM for science satellite communication is the 10% overhead imposed by the ATM cell format. ATM cells contain 48 bytes of data with an additional 5 bytes of cell header. IP packets must be broken into 48 byte pieces with some additional information added to help the receiver reassemble the packet. This process of splitting the IP packet adds complexity and results in additional error cases where the loss of a single ATM cell results in the loss of the entire IP packet. In an environment like ground fiber links with large amounts of bandwidth these issues have traditionally been accepted. However, as the Internet grows and users want more and more bandwidth, alternatives to ATM cells have arisen.

One of the more popular alternatives to ATM cells for high-speed IP support is to bypass the overhead of ATM and put IP packets into SONET. This format is called Packet over SONET (POS). There is still some framing needed but the framing has gone back to the traditional mode of using HDLC framing to put one IP packet in one HDLC frame and carry that over SONET. A PPP header is also added and the end result is very similar to the multi-protocol over Frame Relay format described above.

One concern the authors have with this format is that PPP requires a full-duplex link so it can negotiate some parameters. This presents a problem for spacecraft use because there must be a way to send blind commands to a spacecraft without any two-way communication. This is necessary for spacecraft emergency situations when normal two-way communication with the spacecraft is not available.

However, spacecraft with this type of high-rate downlink normally have multiple transmitters operating at both low and high rates. They also would not normally be attempting any high-rate downlink if the spacecraft was in trouble. A choice of link protocols for data rates above 45 Mbps needs further work to determine the applicability of Packet over SONET for spacecraft.

A major concern for satellite system engineers is both the processing overhead and byte overhead associated with protocols. This is not a major issue for onboard LAN protocols where bandwidth is not as severely limited. Overhead is an issue on the space-to-ground link where bandwidth is often limited due to standard RF link budgets affected by power, error rate, signal quality, and distance.

The overhead of HDLC is very minimal with only the following fields

- 1-byte flag or sync byte
- 4-byte Frame Relay and IP encapsulation header
- 2-byte CRC for error detection
- overhead imposed by bit-stuffing

This framing overhead is as small as other space framing formats used today but there are still concerns about the variable overhead generated by the bit-stuffing function. The extreme case would be an overhead of 20%, which would result from a frame containing all one bits and a zero bit would be inserted after every fifth bit. However this scenario is very unlikely since a frame containing all ones contains no information. Examination of several data files from the WIND, POLAR, and SOHO spacecraft indicates a realistic HDLC bit-stuffing overhead is in the 1-3% range.

Another approach to dealing with potential erroneous bit recovery on these links is to include additional bits that the receiver can use to detect and correct damaged bits. This type of coding is referred to a forward-error-correction (FEC) since the error correction information is passed forward with the data. Various FEC coding schemes have been devised over the years. Some of the most common FEC techniques are convolutional coding and Reed-Solomon (R-S) coding.

The major difference between these two coding techniques is that convolutional coding operates on a serial bitstream with no specific byte boundaries while Reed-Solomon coding operates on fixed size blocks of data. A convolutional encoder accepts individual bits, adds additional coding bits based on a predictable algorithm, and passes out the encoded bitstream. A convolutional decoder reverses this process by identifying the original pattern, removing the additional bits, and passing out the original bitstream. The additional bits provide sufficient information so that some errors can be detected and corrected by the decoder.

Reed-Solomon coding does not insert bits into the middle of the data but appends check symbols to a whole block of data. These symbols can later be used to detect and correct errors that may have been introduced in the data. Since RS coding operates on a block of data the receiver must locate the RS synchronization pattern at the beginning of the code block. The CCSDS Reed-Solomon coding specification uses a 4-byte synchronization pattern (0x1acffc1d) to delimit the code blocks and a (223,32) coding scheme. Using a 4-byte pattern and fixed length blocks provides a robust sync detection in more severe bit error environments. The long sync pattern is less likely to spuriously occur due to bit errors and the fixed length blocks allow the receiver to "flywheel" or assume where a sync pattern should be and continue processing data without dropping lock.

The Intelsat Technical Note TN309.5 specifies a Reed-Solomon code for commercial carriers to use and it has a 4-byte sync pattern (0x5a0fbe66) and Reed-Solomon code parameters of (219,201,9). It also specifies an interleaving scheme to distribute burst errors over wider areas of data and increase the probability of error correction. A common use of these Intelsat communication links is to provide WAN connectivity between switches and routers transmitting HDLC frames. Another commercial application of Reed-Solomon coding is in Digital Video Broadcasting (DVB) which uses yet another Reed-Solomon coding algorithm. The main point is that many communication applications use forward error correction techniques today but it is used to simply provide better link quality and is independent of any data link framing implemented by higher level users.

This is different from many current spacecraft systems where the RS framing is also used as the data link framing. However, this then forces each data link frame to be fixed length to match the RS code block length. The main problem with this is that science and engineering data packets are normally not the same size as the RS frame.

Fitting various length packets into fixed length RS frames means that additional information must be included along with the packets. This information indicates where the first packet starts in a frame and how long each packet is. Since the various packet sizes do not fit evenly into RS frames, packets are also split between frames.

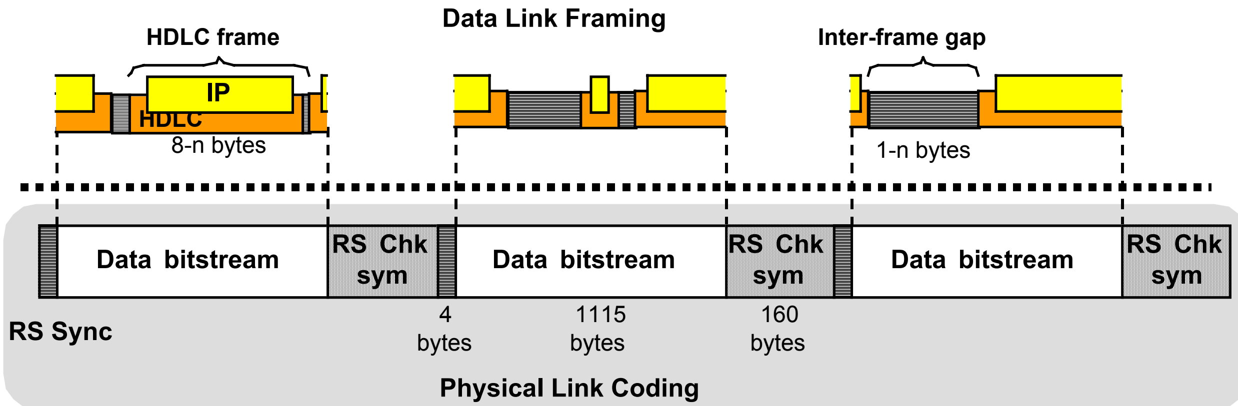

Figure 2 - Separation of HDLC Framing and RS Coding

If there are too many bit errors in a frame the Reed/Solomon coding will not be able to correct the bits. In this case the frame is discarded along with the part of the packet from the previous and following frames.

One of the most important issues in this paper is to note that unlike current space communication systems, commercial network products perform forward error correction (FEC) coding, such as Reed-Solomon or convolutional, *independently* from the data link framing. This is in accordance with the OSI layered model of networking, where framing is carried on at the data link layer and coding is down at the physical layer. The coding simply treats the data link frames as a bit-stream to be protected. This is a key difference between the current data formats used in many space missions and the OMNI architecture.

This separation, as illustrated in figure 2, is the standard way Internet connectivity is deployed across commercial satellite links. Commercially available satellite modems support many modulation and coding techniques to improve the bit error rate (BER) of bits passed through communication satellites. However, the inputs and outputs of these modems are simply a clock and data bitstream. This allows users to connect whatever network equipment they want and use any framing protocol desired. There is no relationship between the users data link framing and any framing that might be used over the RF link. This approach allows future spacecraft to use new and better coding schemes by only changing the FEC processor in their transmitters/receivers without any changes in the rest of the installed equipment onboard or in ground systems.

Reed-Solomon coding is also commonly used as a bit level FEC mechanism for many other applications such as cable modems, ADSL, cell phones, direct-broadcast TV, and CD-ROMs. These applications do not use the RS code block for data link framing but simply to provide better data quality to the bitstream being delivered.

Finally, separating the Reed/Solomon code block framing from the data link framing eliminates the current need for fill frames and fill packets. Since the space link uses synchronous clocking, conditions occur where there is no upper layer data to be sent but frames must still be output. Current protocols implement fill packets to be used to fill out frames to meet frame output timing requirements. This added complexity goes away when RS coding is separated from data link framing.

The Reed-Solomon coding simply operates on a bit level and is constantly accepting bits without any relationship to whether the upper layers are sending frames or not. This is the way Reed/Solomon coding is used in all other commercial applications. This is also the way that Reed/Solomon coding has been used on the WIND and POLAR spacecraft for the last 5 years.

A DETAILED SIMULATION

ITT Industries developed and operates a high fidelity end-to-end link simulator for NASA's Tracking and Data Relay Satellite (TDRS) system. The simulator is capable of providing accurate bit error rate and throughput estimates for CCSDS frames relayed through a TDRSS satellite link. In an effort to provide an accurate comparison of the throughput at various bit error rates between the CCSDS transfer frame and Frame-Relay as link layer protocols, ITT developed an HDLC simulation model to incorporate into their simulator. The addition of the HDLC simulation model has allowed a side by side throughput comparison to be performed between CCSDS and HDLC at a variety of bit error rates and packet sizes.

The overall architecture of the simulation model is illustrated in Figure 3. The model consists of four main components. They are:

- Customer Spacecraft Transmitter
- TDRS Transponder
- Ground Station Receiver
- Metric Estimator

The customer satellite model generates IP packets that are passed to a framing model which performs either CCSDS or HDLC framing. Options for both Reed-Solomon and convolutional encoding are available before the data is passed to the physical layer model for transmission to TDRS.

The TDRS transponder model incorporates the effects of linear and nonlinear distortions encountered during frequency translation, filtering, and amplification of the customer spacecraft signal. The simulation adds thermal noise to the customer spacecraft signal before and after the TDRS transponder to account for the space-to-space and space-to-ground link effects.

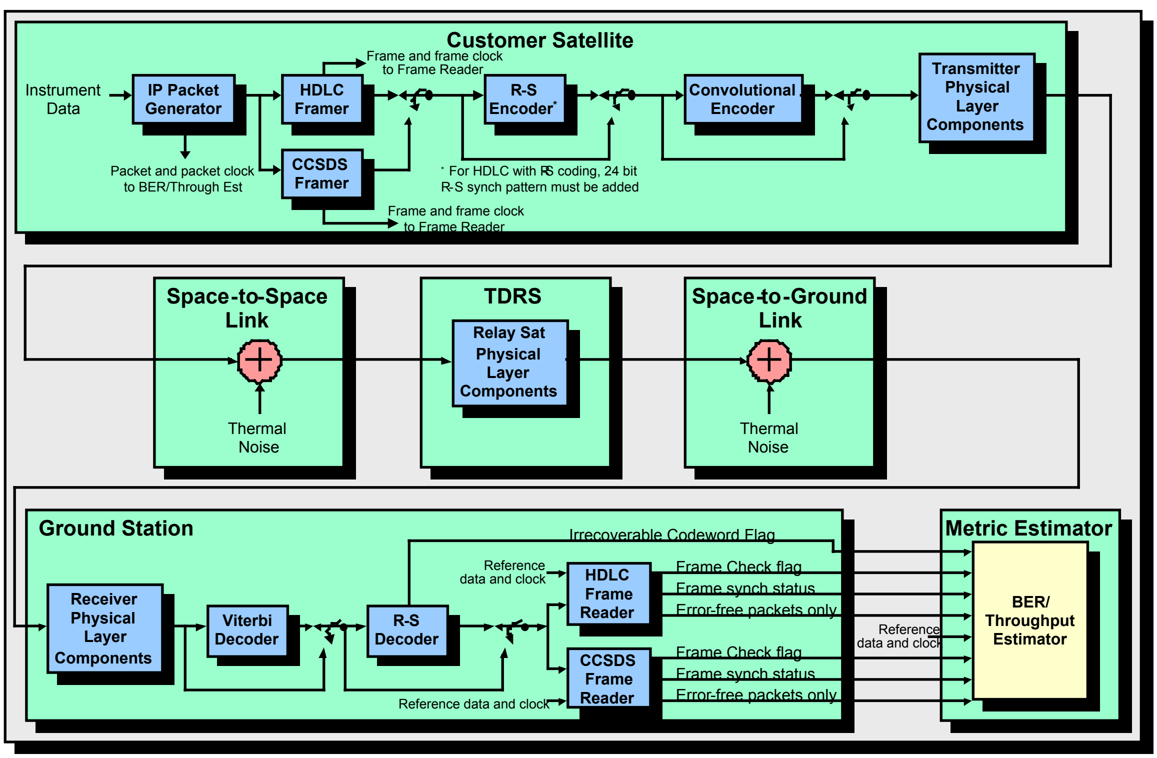

Figure 3 – Functional Diagram of the Simulation Model

The ground station model simulates the physical layer characteristics of the receiver equipment. It also incorporates optional Viterbi and Reed-Solomon decoders to match the FEC options selected in the Customer Satellite model. The resulting data stream is then passed to either the CCSDS or HDLC frame readers for data recovery. The metric estimator then compares the resulting output data with the original input data to produce the desired bit error rate and throughput statistics.

The scenarios that were modeled for this study combined a variety of packet sizes in both uncoded and convolutionally coded cases. The relationship between packet size and throughput is a function of bit error rate and protocol overhead. At low bit error rates many small packets have more overhead than a few large packets. Therefore large packets produce higher throughput than small packets. On noisy links bit errors typically cause the loss of an entire packet. Therefore small packets produce higher throughput since less data is lost for each bit error.

Forward error correction techniques typically produce a fixed improvement in the BER. Although both Reed-Solomon and convolutional coding are available in the simulation model, this study was performed using only the convolutional coding option due primarily to time constraints. .. The final matrix of cases that were simulated included packet lengths of 64, 256, 1024, 1500, and 4096 bytes both uncoded and convolutionally coded for both CCSDS and HDLC protocols.

Figure 4 summarizes the results for the uncoded cases. At the advertised BER for the TDRS system of 1e-5, the HDLC protocol slightly outperforms CCSDS for the larger packet sizes. As the BER increases HDLC continues to provide improvement of a few percent over CCSDS. At lower BER's however the CCSDS performance increases providing slightly better performance than HDLC. These performance differences are only on the order of a few percent at best and cannot be considered significant when choosing a protocol to use.

Figure 5 summarizes the results for the convolutionally coded cases. As one might expect the results are similar. The improvement in BER afforded by the coding tends to shift the results toward higher BER but does not change the relative performance between the individual cases. Although not examined in this study, the throughput results for the concatenated coding case (convolutional coding plus Reed-Solomon) would be expected to trend similar to those for the convolutional coding case.

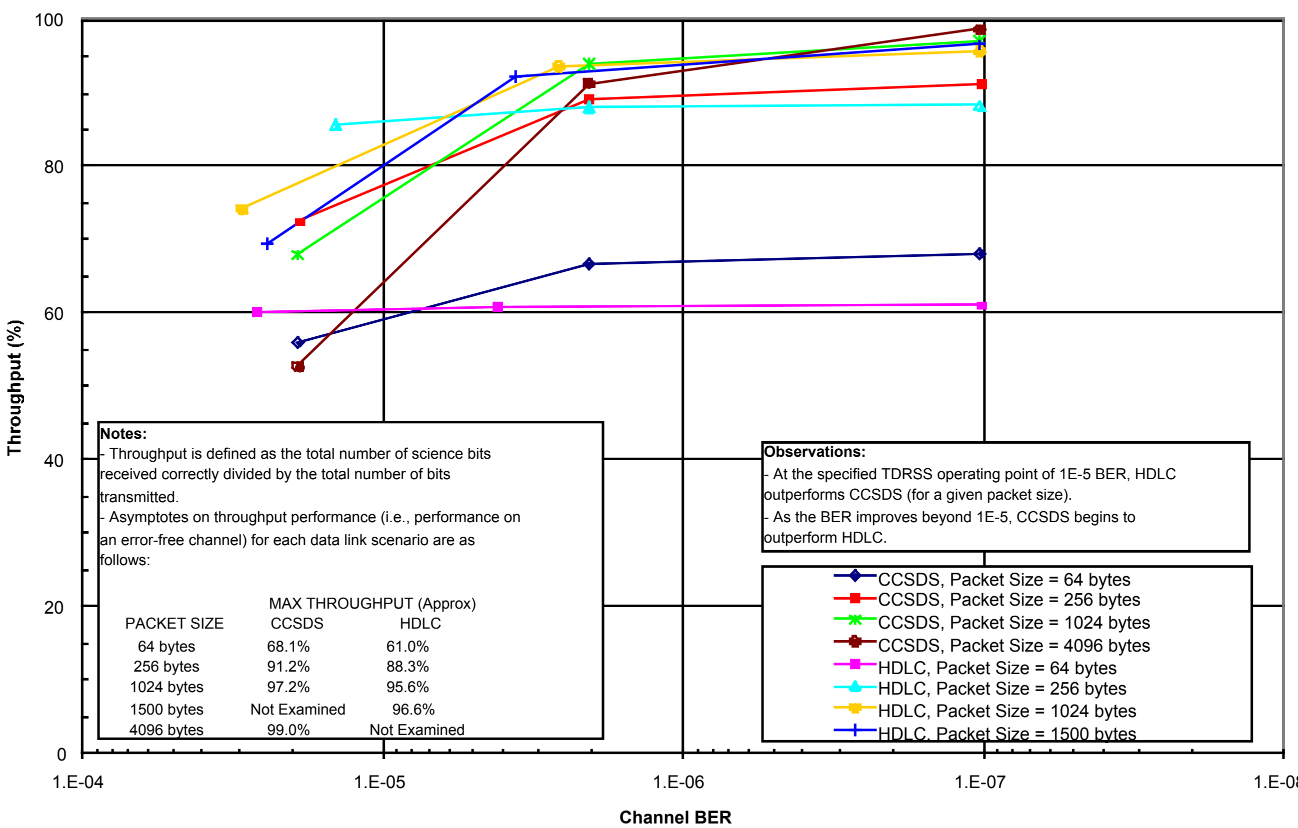

Figure 4 – Throughput versus BER for the uncoded test cases

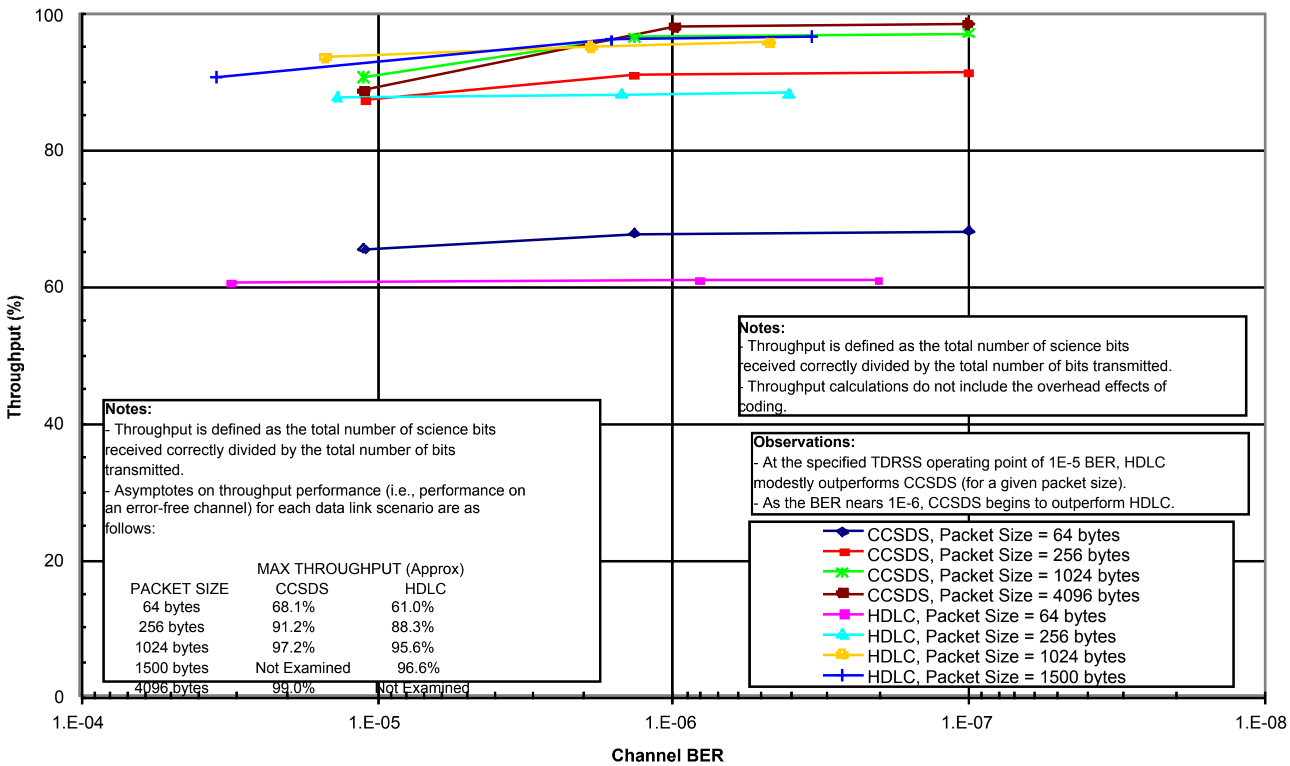

Figure 5 – Throughput versus BER for the convolutionally coded test cases

A SPACE-BASED DEMONSTRATION

In late 1999 the OMNI project had been looking for opportunities to test these "Internet in Space" concepts on an orbiting spacecraft. However, many of the spacecraft candidates were deemed unsuitable due primarily to their onboard communication hardware. The key issue was to find a spacecraft that could support HDLC framing in hardware to allow simple, straightforward interfacing with existing commercial routers. These requirements made UoSAT-12, a spacecraft launched in May 1999 by Surrey Satellite Technology Ltd. (SSTL), an ideal test platform, as it already used HDLC framing to carry its AX.25 protocol. The AX.25 protocol and HDLC framing have been used on over 70 spacecraft over the last 20 years. Since HDLC interface hardware was already present on-board, only flight software changes would be required to adapt UoSAT-12 to use IP. Changes to the ground station would also be minimal, requiring only the addition of a standard commercial router and a programmable switch.

Since the SSTL ground station already supported HDLC framing, a standard Internet router was the only addition needed. Figure 8 indicates the basic components of the ground station and where the router was added in parallel with the existing AX.25 communication front-end. The only station reconfiguration required was to select which system is connected to the transmitter. This is done with a controllable switch, which supports fully automated passes for either the IP or AX.25 mode.

The SSTL ground station is built on an Ethernet LAN with firewalls and router connectivity to the Internet. Two addresses were used on the ground station LAN to support these tests. One address was used for the Ethernet interface on the router and the other address was assigned to the spacecraft.

In February 2000 work was initiated to port a standard IP stack to the SpaceCraft Operating System (SCOS) used on the UoSAT-12 spacecraft. In April 2000 the first basic connectivity tests using IP to a spacecraft were performed. Standard ICMP echo request (PING) packets were sent from both GSFC and the Surrey ground station to the spacecraft. The packets passed through a standard router at the Surrey ground station and were transmitted to the UoSAT-12 spacecraft. The standard IP stack onboard

UoSAT-12 returned echo response packets addressed to the separate sources. Those packets then passed through the ground station router and were delivered to their respective destinations using standard Internet routing. These tests verified proper operation of both the end-to-end IP routing and the HDLC framing on the space-to-ground link.

Once the end-to-end connectivity was operational, additional tests were performed to have the spacecraft automatically set its clock using the Network Time Protocol (NTP) by referencing a time server (tick.usno.navy.mil) at the US Naval Observatory (USNO). Tests were also performed using the standard File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), and real-time telemetry and blind commanding with UDP packets.

The downlink data rate for UoSAT-12 is only 38.4Kb/s. With such a slow data rate it is difficult to accumulate enough data to produce an accurate BER measurement below 1e-5. Figure 6 illustrates a rough estimate of the BER versus elevation angle for several passes. The BER estimate is based on a comparison of the number of good and bad frames reported by the router for the serial port receiving the spacecraft data. The assumption was also made that a corrupted frame was the result of a single bit error. Our ping tests showed a usable link nearly to the horizon. A visual inspection of the BER at low elevation angles on Figure 6 shows that the HDLC framing was functioning at BER's of 1e-4.

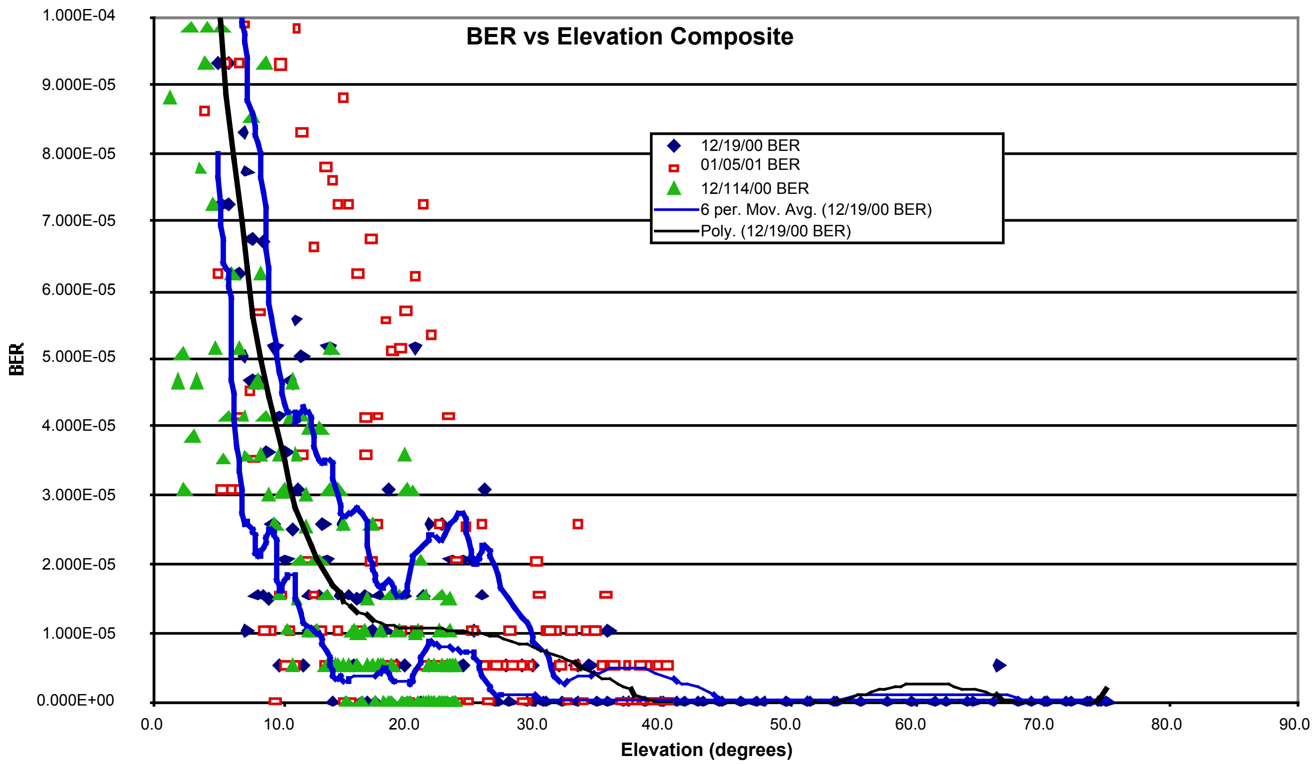

Figure 6 – Bit error rate estimates from several UoSat-12 passes

THE HISTORY OF HDLC IN SPACE

The past twenty years has seen considerable use of the HDLC protocol in space missions. The national space agencies of large spacefaring countries have chosen not to use commercial protocols on their space-to-ground links and developed the CCSDS protocol. However many educational institutions, amateur groups, and smaller countries have and are continuing to use HDLC as a viable link layer protocol. This is primarily due to the availability of low cost hardware and software required to support their data transmission and distribution systems.

A survey of small experimental satellites launched over the past twenty years and many that are planned for the next few years revealed seventy two spacecraft that either used, are currently using, or are planning to use HDLC on their missions. Most of these spacecraft were developed and operated by 23 universities, 8 amateur groups, and 7 commercial space entities in 24 countries. Also represented are NASA, ESA, the US Air Force, the US Navy, and the Chilean Air Force. Earth resources satellites by Germany and Turkey, and a Disaster Monitoring Constellation by Algeria, Nigeria, and the UK are currently in development and will use HDLC. A list of these spacecraft can be found at http://ipinspace.gsfc.nasa.gov/documents/hdlcsat.xls.

CONCLUSIONS

The OMNI project at NASA/GSFC has demonstrated the viability of using the commercial Frame-Relay/HDLC framing mechanism as a link layer protocol for space-to-ground links. The last 2 years of tests and demonstrations have shown that HDLC framing provides a very simple and flexible communication mechanism for space communication. HDLC framing is well supported in a wide range of COTS products and has been used on spacecraft for over 20 years. Detailed modeling performed by ITT Industries shows that HDLC performs comparably to the currently used CCSDS protocol for data transmissions from a low earth orbiting spacecraft to the ground via a TDRSS relay. Also, HDLC requires no modifications to operate in intermittent space link conditions.

HDLC framing provides a minimal byte overhead along with a link level error check. The variable length of HDLC framing also results in very simple data packing and unpacking since one IP packet normally ends up in one HDLC frame. A large UDP packet can be sent, causing IP fragmentation, but this is under the application programmer's control and can be completely avoided if desired. The biggest benefit of using HDLC is that it is supported on virtually any communication hardware that has serial interfaces.

Using the IETF multiprotocol over frame relay encapsulation has proven to be very robust and supported on every piece of communication equipment we have worked with. We have mixed equipment from different vendors on serial links, and there have been no compatibility problems. Frame relay equipment can also be used to provide basic forwarding of frames without any IP processing involved. This provides additional flexibility in deploying communication systems.

The major missing pieces are components for the spacecraft. Technologies like Ethernet and HDLC are currently in use on some low-earth orbit spacecraft where radiation is not a major issue. More work is needed to develop fully space-qualified components for onboard serial interfaces to the RF equipment and for onboard LANs.

The next flight test that will be made using an HDLC link will be the Communication And Networking Demonstration On Shuttle (CANDOS). The CANDOS experiment will be flown as a Hitchhiker payload on the STS-107 Space Shuttle mission. In this experiment an ITT Low Power Transceiver (LPT) with an integral computer running Linux will provide an orbital test platform to further test IP over HDLC on space-to-ground links. Test objectives include mobile IP, the UDP based reliable file transfer protocol Multicast Dissemination Protocol (MDP), reliable and blind commanding, UDP based real-time telemetry, and spacecraft clock synchronization using the Network Time Protocol (NTP).

ACKNOWLEDGMENTS

The research described in this paper was carried out by personnel from Computer Sciences Corporation working for NASA's Goddard Space Flight Center under contract GS-35F-4381G task S-43981-G, and personnel from ITT Industries under contract S-87070-4. The authors would like to thank Dave Israel and GSFC code 450 for their pioneering IP work on the SPTR project, Chris Jackson of Surrey Space Technologies Ltd. and Harold Price of VyTek Wireless for their support with UoSAT-12, and Cisco Systems for the loan of a Cisco 1601 router for use in the SSTL ground station. The authors would also like to thank of Chitra Patel, Abby Kassa and Leonardi Tran of ITT Industries who contributed so greatly to the development of the simulation models and evaluation of the raw simulation results.

Current information on test results and future activities will be posted on the OMNI project web site at http://ipinspace.gsfc.nasa.gov/