AN OCEANOGRAPHIC BUOY WITH INTERACTIVE SATELLITE COMMUNICATIONS

J.N. Shaumeyer, Carl C. Gaither III, Peter H. Young, John M. Borden,
Erik Mollo-Christensen, and David Provost
Jackson & Tull Aerospace Engineering Division
7375 Executive Place, Suite 200
Seabrook, MD 20706 USA

Walter Paul and Jim Irish
Woods Hole Oceanographic Institution
Woods Hole, MA 02543 USA

I. INTRODUCTION

Jackson and Tull (J&T) and the Woods Hole Oceanographic Institution (WHOI) have collaborated since 1995 on developing a modular design for a moored, oceanographic buoy with two-way satellite communications for telemetry and commanding via low-earth orbiting (LEO) satellites. The use of these new LEO satellites allows the possibility of increasing the data throughput from remote data platforms by many orders of magnitude over the currently used Argos or GOES systems. In addition, our communication link allows commands to be sent to the buoy for failure recovery or dynamic response to interesting phenomena.

The ability to retrieve large amounts of data within hours of its being collected, coupled with the ability to send commands to the data platform, has the potential of changing the way remote experimentation is done in the oceanography community. Oceanographic research programs are typically designed around the limits on how much data can be retrieved. Removing that barrier should allow more and better sensors and more dense data rates such that researchers can look at ocean processes in much greater detail.

II. BUOY DESIGN

Our approach has been to create a buoy system with a modular design so that the suite of sensors and the data-gathering software can be easily reconfigured in support of a variety of scientific experiments. For the platform, our starting point was the buoy designed for the GLOBEC program by WHOI.

Shown in cross-sectional schematic in Fig. 1, the platform is described in detail in an accompanying paper [Irish, Paul, & Borden]. The electronics well, 24” in diameter and about 36” deep, houses batteries, the data system and interface electronics, and the radio electronics. Access to the electronics well is through a hinged lid. Waterproof bulkhead connectors around the top of the well connect external devices to internal electronics. Inside the well, the electronics are mounted on aluminum trays which slide into place along guide rails, so that they are easily lifted out and replaced.

Paper appeared in *Proceedings Ocean Community Conference ‘98* (The Marine Technology Society Annual Conference), Baltimore, MD, November 1998, pp. 119-123.

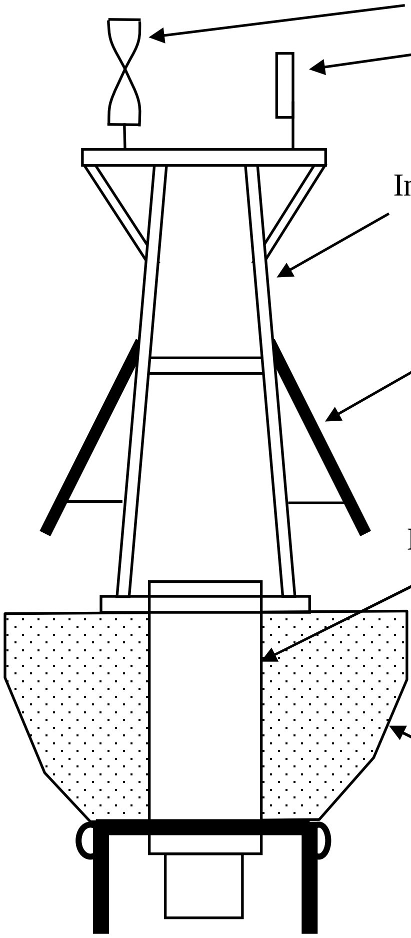

Figure 1. A cross-sectional schematic of the buoy platform.

The instrument tower, approximately 7' tall, is the structure supporting the 4, 64-Watt solar panels, meteorological sensors, radar reflector, guard light, GPS antenna, and the radio's omnidirectional quadrifilar antennas. Buoyancy comes from the 6'-diameter, Surlyn-foam hemisphere around the instrument well. The overall height of the platform is about 12'.

The buoy data system, which is described in much more detail elsewhere in these proceedings [Gaither, *et al.*] is built around a PC/104-bus embedded microcomputer system manufactured by WinSystems, Inc. The computer has an 80486DX4-100 microprocessor, 8 Mbytes of RAM, and 2 Gbytes of disk storage. Data interfaces are either through A/D converters or serial (RS-232) ports. The computer runs Linux, a freely available, Unix-like operating system.

III. 2-Way Communication System

The connection between the researcher and the remote buoy consists of three components: the buoy's data system and radio, the satellite link, and the ground station.

A. PACSAT Satellites

The buoy communication system is designed to use certain Little LEO microsatellites (with masses of 10 to 100 kg) known as UoSATs, built and operated by the University of Surry's Spacecraft Engineering Research Group. For our research buoy we have been operating with a single satellite, PoSAT-1, which was built for a consortium of Polish companies and universities. Launched in 1993, PoSAT-1 is in quasi-polar orbit at an inclination of 98.6 degrees and an altitude of about 800 km.

Messages are sent between satellite and surface using packet (digital) radio technology. Full-duplex operation is possible since uplink (to the satellite) and downlink are performed at different frequencies; with PoSAT-1, the uplink carrier frequency is 149 Mhz, and the downlink frequency is 430 Mhz, with effective baud rates of 9600.

Across this radio link the packet-level transport layer is provided by the AX.25 Link-Layer Protocol, long used by radio amateurs. AX.25 packets perform a similar function to IP packets in the TCP/IP suite of protocols, with the major difference being that source and destination addresses in AX.25 packets are radio callsigns. Two AX.25 modes are used with the UoSATs: datagram mode, in which packets are transmitted unacknowledged and may be lost, and connected mode, in which packets are acknowledged and the communication channel is guaranteed to be error free.

As the application-level layer these satellites use the PACSAT suite of protocols (written by Jeff Ward and Harold Price of Surry). PACSAT protocols define high-level mechanisms for transferring files to and from the satellite, requesting directory listings from the satellite, and other operations.

PACSAT satellites operate in a store-and-forward mode, much like an electronic bulletin board. Data files with suitable PACSAT-defined headers can be uploaded from the buoy to the satellite where they are stored in the file system, typically for several days. At a later time in the satellite's orbit another user, our groundstation in this case, commands the satellite to broadcast its directory listing and looks for messages addressed to it, which it then downloads. In a similar fashion the groundstation can uplink files addressed to the remote buoy, which the buoy will then download and process as appropriate. In order to manage the limited storage space on the satellite, files are automatically deleted after a set number of days, or even hours if necessary.

B. Buoy Radio

The buoy radio has two components: the actual RF receive and transmit circuits and radio modem, and the buoy data system, which controls the radio and implements the AX.25 and PACSAT protocols.

The radio electronics are described elsewhere [Gaither, *et al*.]. The radio is operated automatically by several programs running on the buoy computer. Tracking software running on the data system predicts satellite visibility so that the radio can be operated accordingly. The main purpose of the GPS receiver on this platform is to provide accurate time and location updates to the data-system clock for satellite tracking.

When a satellite is coming into view, a software process is started which begins by suspending all data collection processes for the duration of the pass (to conserve power and to avoid corrupting data with RF transmissions). A separate process is then started which uploads a single file, repeating the process for each file in the upload queue. When the satellite pass has ended the upload processing stops and data collection resumes. The upload process implements the necessary PACSAT protocols for file upload; the AX.25 link-layer protocol is implemented in the Linux kernel, a unique feature of Linux and the basis of our choosing it for this system.

Simultaneously with the upload processing, a separate process is run during the pass which listens to all messages broadcast by the satellite. In particular, updated directory information is captured as it is broadcast (in response to a request from the buoy, if necessary), and the directory listing is searched to see whether there are files addressed to the buoy which should be downloaded. If so, the process requests the download. (Files are downlinked in AX.25 datagram mode, not in connected mode, so the downloading process must verify that the file has been received intact.)

Although we are using only one satellite at this time, the radio electronics can be easily extended with programmable receive and transmit frequencies to use multiple satellites. The tracking software already can track multiple satellites, and the remaining upload and download software is satellite independent.

C. Groundstation

In most respects concerning radio operations, the groundstation operates the same as the buoy radio and data system. The groundstation runs the Linux operating system, and we use the same satellite tracking and uploading and downloading software. The difference between the buoy system and the groundstation is that the latter functions as the interface between the researcher operating the buoy and the buoy itself.

The groundstation collects all data uplinked from the buoy (or multiple buoys) to the satellite (or satellites), processes the files as necessary to reconstruct the actual data, and provides the researcher with Internet access to the incoming data. These data can either be added to a database accessible, say, through a World-Wide Web page, or they could be mailed electronically to the researcher.

In addition, the groundstation is the single point through which all commands pass on their way to a remote buoy. Although we have not yet implemented this operation, we plan to virtualize the interface to the buoy so that the researcher simply sends text files by e-mail to the buoy using standard Internet style (RFC 822 compliant) addressing.

IV. Data & Command Handling

Although the amount of data that can be received through the satellite link is bandwidth limited to some 100 kbytes/day, the rate of data collection by the buoy itself can be determined independently and is not subject to the same limit. The large hard disk which is part of the data system allows us to collect data at a higher rate than it can be transmitted, and these rates can be determined separately for each sensor.

Data from each sensor are collected at their own rates into separate files, with each file being either subsampled or averaged at rates which are compatible with the bandwidth of the satellite link. When the averaged time-series file has reached a suitable size for transmission (about 4 kbytes) it is prepared for uplink. Meanwhile, the file segments that constitute the original time series are archived on the data system's hard disk for retrieval when the buoy is recovered.

For uplinking, each file is first compressed, typically to one third its original size. Then the PACSAT header is added, particularly containing the destination address (the groundstation callsign). Finally, each file is then moved into a directory on the file system reserved for the uplink queue.

When the satellite is in view, the uplink process sorts the queued files and attempts to uplink as many as possible during the satellite pass. For each file to uplink, the process begins by opening the AX.25 connected-mode channel to the satellite (using the Linux socket IPC interface). With the connection established, a PACSAT message asking for upload permission is acknowledged with a file number assigned to the file by the satellite, and the uplink process begins sending PACSAT packets to the satellite. At the end of the uplink an end-of-file message is sent, it is acknowledged by the satellite if the file has been uploaded successfully, and the connection is surrendered. If the satellite is still in view, the uplink process begins the sequence again for the next file in the queue.

This same uplink procedure is followed by the groundstation to send command files to the buoy. Both buoy and groundstation follow the same downlink procedure, as described in the previous section. When the buoy downlinks a file, it is stored in the downlink directory and later examined to determine whether it is a command to be executed. Files downlinked by the groundstation are processed to recover the original data as transmitted, which are then disseminated as appropriate.

V. PERFORMANCE NOTES

In general, there are 4 or 5 orbital passes when a PACSAT satellite is visible at a terrestrial location. Of these, typically only two are at an elevation and in a direction to be more than marginally useful. Each pass lasts for about 15 minutes, depending on the elevation. During a good pass, our system is able to upload about 50 kbytes of data, amounting to about 100 kbytes/day. We would like to increase that throughput to at least 1 Mbyte/day.

For a variety of reasons such as interfering signals, obstructing structures, or nulls in antenna patterns, the link seen by the AX.25 layer in connected mode is a noisy one. File packets are transmitted in bursts of up to 7 packets (the "window", as in TCP/IP), and backtracking is common. Although the connected-mode link is guaranteed error free, it is not an efficient channel. Compared to the packet roundtrip time, the radio spends so little time actually transmitting, we've found that throughput is maximized by using the largest window size allowed by the protocol.

There is also a tradeoff in the size of the individual file being uploaded. If the file is too large, it runs the risk of having its transfer interrupted by noise or the end of a pass, thus failing to upload completely. If the file is too short, too much time is spent trying to establish AX.25 connected mode for each file compared to the time it takes to transmit the file, and throughput drops. For our typical links to PoSAT-1, the optimal file size seems to be between 1 and 4 kbytes.

With current protocols and the characteristics of the PACSAT satellites, our best strategy for increasing the throughput of our buoy-satellite-researcher communication link is by using more than one satellite. With a suitable increase in groundstations (at separated geographic locations), the number of files that the system can transmit is proportional to the number of satellites that we can utilize in the link. Using more satellites and groundstation locations can also be effective in reducing the latency time between file upload and download. Our next step then is to make the few alterations necessary to our system to allow us to use multiple satellites.

ACKNOWLEDGEMENTS

This work was supported in part by NASA SBIR contract NAS5-97057.

REFERENCES

Gaither, Carl C. III, J.N. Shaumeyer, Peter H. Young, and John M. Borden, “A Linux-Based Automated Data Collection System for Moorings”, proceedings *Oceans ’98,* 15—19 Nov., Baltimore, MD, USA, 1998.

Irish, James D., Walter Paul, and John Borden, “Next Generation Ocean observing Buoy in Support of NASA’s Earth Science Enterprise”, proceedings *Oceans ’98,* 15—19 Nov., Baltimore, MD, USA, 1998.